

HIGH SPEED VLSI ARCHITECTURE FOR DIGITAL APPLICATIONS

A Project Report

Submitted in partial fulfilment of requirements for the award of

Bachelor of Engineering in

Electronics and Communication Engineering

by

LENKA LAKSHMI (39130257)

ASAM SUJITHA (39130041)



DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

SCHOOL OF ELECTRICAL AND ELECTRONICS

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI – 600119

APRIL– 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119

www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICAITON ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **LENKA LAKSHMI (39130257)** and **ASAM SUJITHA (39130041)** who carried out the project entitled "**HIGH SPEED VLSI ARCHITECTURE FOR DIGITAL APPLICATIONS**" under our supervision from November 2022 to April 2023.

Internal Guide

Dr. R NARMADHA, M.E, Ph.D.,

Head of the Department

Dr.T. RAVI M.E., Ph.D.,

Submitted for Viva voce Examination held on: 15.4.2023

Dr.Magthelin Therase, M.E., Ph.D.

Internal Examiner

Dr.N.Krishnamoorthy, M.E., Ph.D.

External Examiner

DECLARATION

I, **LENKA LAKSHMI (39130257)** and **ASAM SUJITHA (39130041)** hereby declare that the Project Report entitled “**HIGH SPEED VLSI ARCHITECTURE FOR DIGITAL APPLICATIONS**” done by us under the guidance of **Dr. R NARMADHA, M.E, Ph.D.**, Department of ECE is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Electronics and Communication Engineering.

DATE:13.4.2023

PLACE:Chennai

SIGNATURE OF THE CANDIDATES

1.



2.



ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr.N.M. NANDHITHA M.E., Ph.D., Dean, School of Electronics and Communication Engineering** and **Dr.T. RAVI M.E., Ph.D. Head of the Department, Department of Electronics and Communication Engineering** for providing me necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our **Project Guide Dr. R NARMADHA, M.E, Ph.D.,** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

wish to express our thanks to all Teaching and Non-teaching staff members of the Department of Electronics and Communication Engineering who were helpful in many ways for the completion of the project.

ABSTRACT

Three-operand binary adder is the basic functional unit to perform the modular arithmetic in various cryptography and pseudorandom bit generator (PRBG) algorithms. Carry save adder (CS3A) is the widely used technique to perform the three-operand addition. However, the ripple-carry stage in the CS3A leads to a high propagation delay of $O(n)$. Moreover, a parallel prefix two-operand such as Han-Carlson (HCA) can also be used for three-operand addition that significantly reduces the critical path delay at the cost of additional hardware. Hence, a new high-speed and area-efficient adder architecture is proposed using pre-compute bitwise addition followed by carry prefix computation logic to perform the three-operand binary addition that consumes substantially less area, low power and drastically reduces the adder delay to $O(\log_2 n)$. The proposed architecture is implemented on the FPGA device for functional validation and also synthesized with the commercially available 32nm CMOS technology library. The post-synthesis results of the proposed adder reported 3.12, 5.31 and 9.28 times faster than the CS3A for 32-, 64- and 128- bit architecture respectively. Moreover, it has a lesser area, lower power dissipation and smaller delay than the HC3A adder. Also, the proposed adder achieves the lowest ADP and PDP than the existing three-operand adder techniques.

INDEX

CHAPTER No.	TITLE	PAGE No.
	ABSTRACT	V
	LIST OF TABLES	IX
	LIST OF FIGURES	IX
	LIST OF SYMBOLS AND ABBREVIATIONS	XII
1	INTRODUCTION	1
	1.1 INTRODUCTION	1
	1.2 INTRODUCTION OF THREE OPERAND ADDERS	1
	1.3 CMOS TECHNOLOGY	4
	1.3.1 NMOS	4
	1.3.2 PMOS	5
	1.3.3 CMOS GATE	6
	1.3.4 CMOS LOGIC GATES	7
2	LITERATURE SURVEY AND OBJECTIVES	8
	2.1 LITERATURE SURVEY	8
	2.2 LITERATURE SUMMARY	16
	2.3 PROBLEM DEFINITION	16
3	REQUIREMENTS ANALYSIS	17
	3.1 MULTIPLEXER	17
	3.1.1 BASIC MULTIPLEXING SWITCH	17
	3.1.2 2x1 MULTIPLEXER	29

	3.1.3 4x1 MULTIPLEXER	20
	3.2 ADDER	21
	3.2.1 HALF ADDER	23
	3.2.2 FULL ADDER	26
	3.3 MOSFET TECHNOLOGY	28
	3.4 CNFET TECHNOLOGY	30
	3.5 HSPICE SOFTWARE	31
	3.6 COSMOSCOPE	33
	3.6.1 POWER	34
	3.6.2 DELAY	35
	3.6.3 POWER DELAY PRODUCT	36
4	EXISTING METHODOLOGY	37
	4.1 MDCLCG	37
	4.2 LCG	38
	4.3 LCG CIRCUIT DESIGN	38
	4.4 ADVANTAGES OF LCG	39
	4.5 DISADVANTAGES OF LCG	40
	4.6 D-FLIP FLOP	40
	4.7 EXISTING D-FLIP FLOP	41
5	PROPOSED SYSTEM	42
	5.1 PROPOSED LOW POWER D-FLIP FLOP	42
6	RESULT AND DISCUSSION	44
	6.1 STIMULATION RESULTS	44
	6.2 PERFORMANCE ANALYSIS	45

	6.2.1 LCG-MOSFET	45
	6.2.2 LCG-CNFET	46
7	CONCLUSION	48
	REFERENCES	50
	STANDARDS AND CONSTRAINTS	52
	APPENDIX	53
	A. CERTIFICATES OF RESEARCH PAPER	53
	B. RESEARCH PAPER	54

LIST OF TABLES

TABLE No.	TITLE	PAGE No.
3.1.3.1	4x1 MULTIPLEXER	21
3.2.1.1	TRUTH TABLE OF HALF ADDER	24
3.2.2.1	TRUTH TABLE OF FULL ADDER	28
4.6.1	TRUTH TABLE OF D-FF	40
6.2.1.1	LCG-MOSFET UNDER 130NM	46
6.2.1.2	LCG-MOSFET UNDER 32NM	46
6.2.2.1	LCG-CNFET UNDER 32NM	47

LIST OF FIGURES

FIGURE No.	TITLE	PAGE No.
1.3.1.1	NMOS TRANSISTOR	5
13.2.1	PMOS TRANSISTOR	5
1.3.3.1	CMOS LOGIC GATE	6
1.3.4.1	CMOS INVERTER	6
1.3.4.2	CMOS NAND GATE	7
3.1.1.1	BASIC MULTIPLEXING SWITCH	17
3.1.1.2	MULTIPLEXER	19
3.1.2.1	THE SCHEMATIC DIAGRAM OF 2x1 MULTIPLEXER	20
3.1.3.1	4x1 MULTIPLEXER	22
3.2.1.1	BLOCK DAIGRAM OF ADDER	23
3.2.1.2	BLOCK DIAGRAM OF HALF ADDER	23
3.2.1.3	LOGIC CIRCUIT FOR HALF ADDER	25
3.2.2.1	BLOCK DAIGRAM OF FULL ADDER	26
3.3.1	MOSFET	28
3.3.2	SCHEMATIC DAIGRAM OF A MOSFET	29
3.4.1	BLOCK DAIGRAM OF CNFET	31
3.5.1	HSPICE SOFTWARE	32
3.5.2	INPUT, OUTPUT SELECTING SCREEN	33
3.6.1	COSMOSCOPE SOFTWARE	34
3.6.2	RESULT GRAPH	34
3.6.2.1	OUTPUT OF DELAY	36

4.1.1	MDCLCG	37
4.3.1	BLOCK DAIGRAM OF LCG OPERATION	38
4.3.2	GENARAL CIRCUIT DESIGN OF LCG	39
4.6.1	D-FLIP FLOP	40
4.7.1	CMOS TECHNOLOGY DFF	41
5.1.1	MODIFIED CMOS TECHNOLOGY DFF	43
6.1.1	STIMULATION OF PROPOSED LCG IN CNFET	44
6.1.2	STIMULATION OF PROPOSED LOW POWER D FLIP FLIOP	44
6.1.3	STIMULATION OF PROPOSED LCG	45

LIST OF SYMBOLS AND ABBREVATIONS

PRBG	PSEUDO-RANDOM BIT GENERATOR
LCG	LINEAR CONGRUENTIAL GENERATOR
CLCG	COUPLED LCG
MDCLCG	MODIFIED DUAL CLCG

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

To achieve optimal system performance while maintaining physical security, it is necessary to implement the cryptography algorithms on hardware. Modular arithmetic such as modular exponentiation, modular multiplication and modular addition is frequently used for the arithmetic operations in various cryptography algorithms. Therefore, the performance of the cryptography algorithm depends on the efficient implementation of the congruential modular arithmetic operation. The most efficient approach to implement the modular multiplication and exponentiation is the Montgomery algorithm whose critical operation is based binary addition is also a primary arithmetic operation in the linear congruential generator (LCG) based pseudo-random bit generators (PRBG) such as coupled LCG (CLCG), modified dual-CLCG (MDCLCG) and coupled variable input LCG (CVLCG). Modified dual-CLCG (MDCLCG) is the most secure and highly random PRBG method among all the LCG-based and other existing PRBG methods. It is polynomial-time unpredictable and secure if for bit greater than 32-bits. Therefore, the security of the MDCLCG enhances with the increase of operand size. However, the area and critical path delay increases linearly since its hardware architecture consists of four three operand modulo- 2^n adders, two comparators, four multiplexer's area in previous papers. Hence, the performance of the MDCLCG can be improved by the efficient implementation of the three-operand adder.

1.2 INTRODUCTION TO THREE OPERAND ADDERS

The three-operand binary addition can be carried out either by using two two-operand adders or one three-operand adder. Carry-save adder (CS3A) is the area-efficient and widely adopted technique to perform the three-operand binary addition in the modular arithmetic used in cryptography algorithms to enhance the security operations. However, the longer carry propagation delay in the ripple-carry stage of CS3A seriously influences the performance of the MDCLCG and other cryptography architectures on IoT based hardware devices. In order to shorten the critical path delay, a parallel prefixed two-operand adder such as Han-Carlson (HCA) can also

be used for three-operand binary addition. It reduces the critical path delay in the order of $O(\log^2 n)$ but increases the area in the order of $O(n \log^2 n)$. Therefore, it is necessary to develop an efficient VLSI architecture to carry out the fast three-operand binary addition with minimum hardware resources.

Random numbers have been used in daily activities since long times ago. Nowadays, a small and cheap kid's toy containing a random number circuit inside it. For example, in a toy like (that mimic) mobile phone will ring different types of sound when the same button is pressed more than one time.

Several random numbers theory have been introduced in the last several decades. Linear congruential generator (LCG) that introduced 1954 by Lehmer is the oldest and the most widely used pseudorandom number generator (PNG). Park & Miller suggest good parameters for LCG. The suggestion is used in Matlab for generating uniform random numbers.

Many other random number generators have been proposed and also used in many applications. Blum BlumShub, Wichmann-Hill, Complementary multiply with carry, Inversive congruential generator, ISAAC (cipher), Lagged Fibonacci generator, Linear feedback shift register, Maximal periodic reciprocals, Mersenne twister, Multiply-with-carry, Naor-Reingold Pseudorandom Function, RC4 PRGA, Well Equidistributed Long-period Linear, and Xor shift are some of the well-known methods.

Hardware for generating random number also available as well as its algorithm. The hardware has been used since 2008. LETech is the fastest among all hardware for computing random numbers, this hardware has been developed since 2008. Research for finding the suitable algorithm of generating random number is well establish field until now. Many researchers use field programmable gate arrays (FPGA) for testing their ideas. Several of these has been realized into hardware and sell into the market.

Initially, the algorithm of LCG combined with MonteCarlo method has been used for generating non uniform random numbers using Matlab. After that, we develop the circuit of random number generator and implemented in FPGA.

The increment factor (c) has been ignored ($c = 0$). Besides, we present novel design and FPGA implementation of LCG algorithm without ignoring the increment. The rest of this work report are organized as follows. Section II deals with theory of LCG algorithm. The design circuit of LCG for FPGA implementation and nets connections are covered. It provides some implementations data and analysis.

The three-operand binary addition is also a primary arithmetic operation in the linear congruential generator (LCG) based pseudo-random bit generators (PRBG) such as coupled LCG (CLCG), modified dual-CLCG (MDCLCG) and coupled variable input LCG (CVLCG) [11]. Modified dual-CLCG (MDCLCG) is the most secure and highly random PRBG method among all the LCG-based and other existing PRBG methods. It is polynomial-time unpredictable and secure if $n \geq 32$ -bits. Therefore, the security of the MDCLCG enhances with the increase of operand size. However, the area and critical path delay increases linearly since its hardware architecture consists of four three-operand modulo- 2^n adders, two comparators, four multiplexers area. Hence, the performance of the MDCLCG can be improved by the efficient implementation of the three-operand adder.

The three-operand binary addition can be carried out either by using two two-operand adders or one three-operand adder. Carry-save adder (CS3A) is the area-efficient and widely adopted technique to perform the three-operand binary addition in the modular arithmetic used in cryptography algorithms and PRBG methods. However, the longer carry propagation delay in the ripple-carry stage of CS3A seriously influences the performance of the MDCLCG and other cryptography architectures on IoT based hardware devices. In order to shorten the critical path delay, a parallel prefixed two-operand adder such as Han-Carlson (HCA) can also be used for three-operand binary addition.

It reduces the critical path delay in the order of $O(\log_2 n)$ but increases the area in the order of $O(n \log_2 n)$ [15]. Therefore, it is necessary to develop an efficient VLSI architecture to carry out the fast three-operand binary addition with minimum hardware resources. Hence, a new high-speed area-efficient adder technique is proposed using pre-compute bitwise addition followed by carry-prefix computation logic to perform the three-operand addition in this paper that consumes considerably less gate area while minimizing the propagation delay in comparison to the HCA-based three-operand adder (HC3A). Furthermore, the proposed adder architecture

is implemented with the Verilog HDL, and then synthesized with commercially available 32nm CMOS technology library. Also, the area-delay and power-delay products of the proposed adder technique are measured and compared with respect to the existing CS3A and HC3A three-operand adder techniques.

1.3 CMOS TECHNOLOGY

CMOS technology is a predominant technology for manufacturing integrated circuits. CMOS stands for “Complementary Metal Oxide Semiconductor”. Microprocessors, batteries, and digital sensors among other electronic components make use of this technology due to several key advantages. This technology uses both NMOS and PMOS to realize various logic functions. Both N and P MOSFET channels are designed to have matching characteristics.

Before CMOS, PMOS and NMOS logic were widely used for implementing logic gates. PMOS was then replaced by the NMOS Technology, which used to be the standard IC fabrication technology. Initially, CMOS was slower and more expensive than NMOS. The main advantages of NMOS technology are simple physical process, functional density, processing speed and manufacturing efficiency.

The main disadvantages of NMOS technology are its electrical asymmetry and static power dissipation. These drawbacks are minimized by using CMOS Technology. The main advantage of CMOS is the minimal power dissipation as this only occurs during circuit switching. This results in much better performance as it allows integrating more CMOS gates on an IC.

1.3.1 NMOS

N-channel MOSFET consists of an N-type source and drain diffused on a P-type substrate. The majority carriers are electrons. When the applied voltage to the gate is high enough, the NMOS will conduct; otherwise, it will not. Since the majority carriers (electrons) travel faster than holes, NMOS are considered to be faster than PMOS.

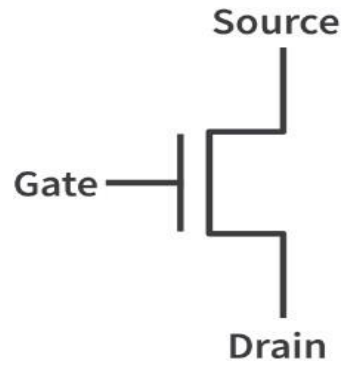


Fig 1.1: NMOS transistor

1.3.2 PMOS

P-channel MOSFET also has a Source and Drain diffused on a substrate. The Source is P-type while the substrate is N-type. The majority carriers are holes. PMOS will conduct when a low voltage is applied. When a high voltage is applied to the gate, the PMOS will not conduct.

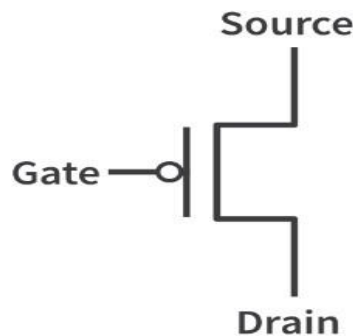


Fig 1.2: PMOS Transistor

1.3.3 CMOS

Since CMOS technology uses both N-type and P-type transistors to design logic functions, a signal which turns ON a transistor type is used to turn OFF the other transistor type. This eliminates the need for pull-up resistors in favour of simple switches. In CMOS logic gates N-type MOSFETs are arranged in a pull-down network between the output and the low voltage supply rail (V_{SS} or ground) while P-type MOSFETs are in a pull-up network between the output and the higher-voltage rail (often V_{DD}). Thus, the N-type MOSFET will be ON when the P-type MOSFET is

OFF, and vice-versa. For any input pattern, one of the networks is ON and the other is OFF.

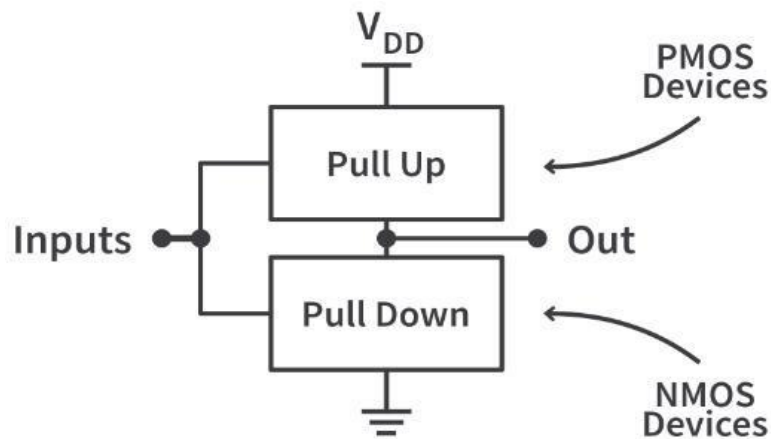


Fig 1.3 CMOS logic gate

CMOS advantages are high speed, low power dissipation, high noise margins in both states, and a wide range of source and input voltages (fixed source voltage).

1.3.4 CMOS Logic Gates

1) CMOS Inverter

The CMOS inverter is the simplest CMOS logic gate. The circuit consists of PMOS and NMOS FET. Input A serves as the gate voltage for both transistors while Y is the output.

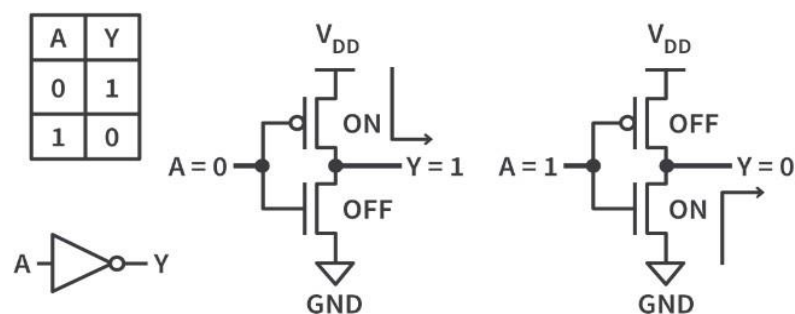


Fig 1.4: CMOS Inverter

The NMOS transistor has an input from V_{SS} or ground and the PMOS transistor has an input from V_{DD} . When the input (A) is low ($<V_{DD}$, ~ 0 V, Logic 0), the NMOS is OFF while the PMOS is ON. V_{DD} will appear at the output through the P-channel MOSFET path. Hence, there is output (Logic 1) with the circuit pulled up to V_{DD} . When the input is high ($\sim V_{DD}$, Logic 1), the PMOS is OFF while the NMOS is ON. The output is pulled down and is therefore low (Logic 0).

2) CMOS NAND Gate

A 2-input NAND gate has two N-channel MOSFETs connected in series between Y (output) and GND and two P-channel MOSFETs connected in parallel between V_{DD} and Y.

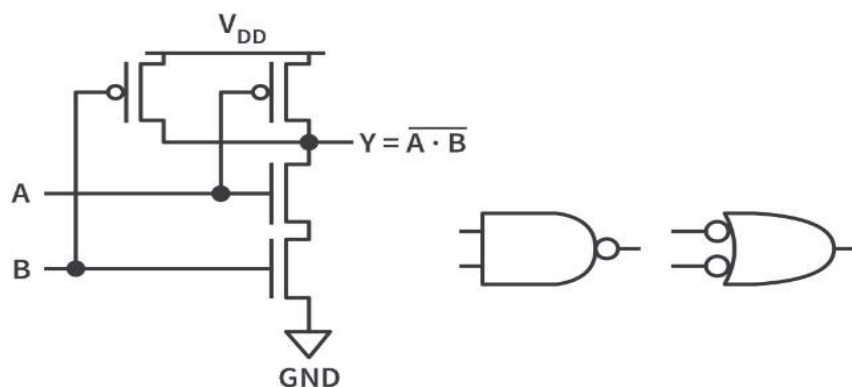


Fig 1.5: CMOS NAND Ga

If either A or B is low (Logic 0), at least one of the NMOS transistors will be OFF. This breaks the path from Y to GND since the NMOS transistors are connected in series. But in this case at least one of the PMOS transistors is ON, completing a path from Y to V_{DD} . This makes the output Y high (Logic 1). For Y to be low, both A and B should be high to ensure that both NMOS transistors are ON so that the path from Y to GND is complete. For all the other combinations of the inputs, Y will be high. The truth table of NAND logic gate is given below.

CHAPTER 2

LITERATURE SURVEY

This chapter discusses about the designing and implementation of linear congruential generator.

2.1 LITERATURE SURVEY

K. Panda and K. C. Ray et.al., (2019) determines that pseudorandom bit generator (PRBG) is an essential component for securing data during transmission and storage in various cryptography applications. Among popular existing PRBG methods such as linear feedback shift register (LFSR), linear congruential generator (LCG), coupled LCG (CLCG), and dual-coupled LCG (dual-CLCG), the latter proves to be more secure. This method relies on the inequality comparisons that lead to generating pseudorandom bit at a non-uniform time interval. Hence, a new architecture of the existing dual CLCG method is developed that generates pseudorandom bit at uniform clock rate. However, this architecture experiences several drawbacks such as excessive memory usage and high-initial clock latency, and fails to achieve the maximum length sequence. To implement the design by using carry save adder for the addition of three operand in the design Therefore, a new PRBG method called as “modified dual-CLCG” and it is very largescale integration (VLSI) architecture are proposed in this paper to mitigate the aforesaid problems. The novel contribution of the proposed PRBG method is to generate pseudorandom bit at uniform clock rate with one initial clock delay and minimum hardware complexity. The proposed architecture is implemented using Verilog-HDL and prototyped on the commercially available FPGA device.

A.K. Panda and K. C. Ray et.al., (2018), the necessity of hardware security for internet-of things applications demands a low hardware area, high speed and secure pseudorandom bit generator (PRBG). Amongst various PRBGs, Blum-Blum-Shub (BBS) is the proven cryptographically secure PRBG because of its large prime factorize problem. The efficient implementation of BBS method relies on the large integer modular multiplication which makes it computationally expensive. Montgomery algorithm is a very efficient solution to perform the modular multiplication which replaces the critical trial division with series of shift and

additions. However, the clock latency and critical path delay are increased with increase of modular size. Therefore, in this paper, a modified radix-2 iterative Montgomery modular multiplier is used for efficient hardware implementation of 1024-bit BBS generator. It replaces two two-operand adders with one three-operand adder. Carry-save adder is the commonly used technique for three-operand addition which experiences high critical path delay. Hence, the critical path delay is further reduced by employing a fast parallel prefix Han-Carlson adder for three-operand addition in the proposed architecture. The proposed architecture is designed using Verilog HDL and prototyped on the Virtex5 FPGA device. The physical implementation results report that the proposed 1024-bit BBS architecture can work at a maximum frequency.

Rezai and P. Keshavarzi et.al., (2015), modular exponentiation with a large modulus and exponent is a fundamental operation in many public-key cryptosystems. This operation is usually accomplished by repeating modular multiplications. Montgomery modular multiplication has been widely used to relax the quotient determination. The carry–save adder has been employed to reduce the critical path. This paper presents and evaluates a new and efficient Montgomery modular multiplication architecture based on a new digit serial computation. The proposed architecture relaxes the high-radix partial multiplication to a binary multiplication. It also performs several multiplications of consecutive zero bits in one clock cycle instead of several clock cycles. Moreover, the right-to-left and left-to-right modular exponentiation architectures have been modified to use the proposed modular multiplication architecture as its structural unit. We provide the implementation results on a Xilinx Virtex 5 FPGA demonstrating that the total computation time and throughput rate of the proposed architectures outperform most results so far in the literatures.

K. S. Pandey, D. K. B. N. Goel, and H. Shrimali et.al., (2019), Parallel Prefix adders are arguably the most commonly used arithmetic units. They have been extensively investigated at architecture level, register transfer level (RTL), gate level, circuit level as well as layout level giving rise to a plethora of mathematical formulations, topologies and implementations. This paper contributes significantly to the understanding of these parallel prefix adders in a couple of ways. Firstly, it attempts to describe various such parallel prefix adders in elegant and consistent

formulations. Secondly, a new family of parallel prefix adders is proposed at architecture level. The estimates of the area-throughput characteristics for an instance of this family are also presented. While the speeds achieved by this instance match those achieved by the state-of-the-art adders, their area characteristics improved than existing but still it has more hardware complexity.

S. S. Erdem, T. Yanik, and A. Celebi et.al., (2017), this paper presents an efficient digit-serial hardware architecture for Montgomery algorithm. As many previous works, carry-save adders are used to accumulate partial products to avoid carry propagation delay. The Montgomery algorithm is a fast modular multiplication method frequently used in cryptographic applications. This paper investigates the digit-serial implementations of the Montgomery algorithm for large integers. A detailed analysis is given and a tight upper bound is presented for the intermediate results obtained during the digit serial computation. Based on this analysis, an efficient digit-serial Montgomery modular multiplier architecture using carry save adders is proposed and its complexity is presented. In this architecture, pipelined carry select adders are used to perform two final tasks: adding carry save vectors representing the modular product and subtracting the modulus from this addition, if further reduction is needed.

S.-R. Kuang, J.-P. Wang, K.-C. Chang, and H.-W. Hsu et.al., (2013), modular exponentiation in the Rivest, Shamir, and Adleman cryptosystem is usually achieved by repeated modular multiplications on large integers. To speed up the encryption/decryption process, many high-speed Montgomery modular multiplication algorithms and hardware architectures employ carry-save addition to avoid the carry propagation at each addition operation of the add-shift loop. In this paper, we propose an energy-efficient algorithm and its corresponding architecture to not only reduce the energy consumption but also further enhance the throughput of Montgomery modular multipliers. The proposed architecture is capable of bypassing the superfluous carry-save addition and register write operations, leading to less energy consumption and higher throughput. In addition, we also modify the barrel register full adder (BRFA) so that the gated clock design technique can be applied to significantly reduce the energy consumption of storage elements in BRFA. Experimental results show that the proposed approaches consume less area and high throughput improvement for 1024-bit Montgomery multiplier.

S.-R. Kuang, K.-Y. Wu, and R.-Y. Lu et.al., (2016), this paper proposes a simple and efficient Montgomery multiplication algorithm such that the low-cost and high-performance Montgomery modular multiplier can be implemented accordingly. The proposed multiplier receives and outputs the data with binary representation and uses only one-level carry-save adder (CSA) to avoid the carry propagation at each addition operation. This CSA is also used to perform operand pre-computation and format conversion from the carry save format to the binary representation, leading to a low hardware cost and short critical path delay at the expense of extra clock cycles for completing one modular multiplication. To overcome the weakness, a configurable CSA (CCSA), which could be one full-adder or two serial half-adders, is proposed to reduce the extra clock cycles for operand pre-computation and format conversion by half. In addition, a mechanism that can detect and skip the unnecessary carry-save addition operations in the one-level CCSA architecture while maintaining the short critical path delay is developed. As a result, the extra clock cycles for operand pre-computation and format conversion can be hidden and high throughput can be obtained. Experimental results show that the proposed Montgomery modular multiplier can achieve higher performance and significant area–time product improvement when compared with previous designs.

Gagan Abbot and Dhruv Sharma et.al., (2020), VLSI architecture design of DSP focuses on design techniques for the realization of a dedicated Very Large Scale Integrated (VLSI) systems for signal processing, image processing, and other communication applications. The VLSI design techniques and systolic architectures will be used for exploring the speed–area–power trade-offs for different DSP applications. Memory-based structures are a pertinent and fitting choice for a large number of signal processing implementations that implicate multiplication with a certain set of coefficients. In this paper, however, we show a memory-based approach that can be advantageous for reduced-latency and area reducing implementations in which memory processing time is shorter than the normal computation-time effectuated in traditional multipliers. The key factor of our paper is lookup table (LUT) optimization which reduces the area and power, also a pipelined version of the memory-based multiplier reduces the combinational path delay.

Sudipta Bose, Arijit De, and Indrajit Chakrabarti et.al., (2021), low-complexity, high-speed and re-configurability are the primary requirements of the finite impulse

response (FIR) filters employed for the processing of acquired seismic signal in real-time seismic-alert-system. The common sub-expression elimination (CSE) technique is employed widely to reduce the hardware complexity by minimizing the logic operators (LOs) and logic depths (LDs) in digital FIR filter. In this brief, a novel matrix grouped CSE (MCSE) algorithm has been proposed which outperforms the existing CSE algorithms in terms of LOs and LDs minimization. Moreover, a new half-unit biased (HUB) based rounding technique is incorporated in the proposed design to reduce the truncation error while maintaining low-complexity and the cut-set retiming technique is employed to reduce the critical-path-delay (CPD). Two hardware efficient FIR filter architectures (I and II) involving the proposed canonical signed digit (CSD) based MCSE algorithm, HUB rounding and cut-set retiming approach have been presented. Further, the architecture II represents a hardware efficient realization of a reconfigurable FIR filter. The hardware implementation of the architectures is performed on both FPGA and ASIC platforms. The hardware implementation of the proposed architecture I yields more than 25%, 49%, 38%, 36% and, 31% reduction in LOs, CPD, effective latency, area-delay-product (ADP) and, power-delay-product (PDP), respectively, over the state-of-the-art CSE based architectures. Whereas the reconfigurable architecture II exhibits nearly 48%, 12% and, 13% reduction in CPD, ADP and, PDP over the counterpart.

Tresa Joseph, Bindiya T. S et.al., (2021), this paper proposes a multiplexer-based technique for accelerating matrix vector multiplication (MVM) in long short-term memory networks. The concept of multiple constant multiplication with separate generation and selection of partial products is used in this design. The most important benefit of the proposed architecture is the reduced implementation complexity in terms of cell area and power efficiency. A truncating method is effectively utilized with a guaranteed reduction in the hardware complexity and power consumption. For inner product computations, the input coefficients are truncated instead of being shifted, which in turn make use of combinational circuits replacing the sequential designs. Also, to improve the overall clock period, a novel design for inner product computations is envisaged, which uses modified multiplexer architecture instead of the existing method. The results show a reduction in the implementation complexity of the MVM with a significant power reduction of 14%.

Amit Kumar Panda, Rakesh Palisetty, and Kailash Chandra Raym et.al., (2020), three-operand binary adder is the basic functional unit to perform the modular arithmetic in various cryptography and pseudorandom bit generator (PRBG) algorithms. Carry-save adder (CS3A) is the widely used technique to perform the three-operand addition. However, the ripple-carry stage in the CS3A leads to a high propagation delay of $O(n)$. Moreover, a parallel prefix two-operand adder such as Han-Carlson (HCA) can also be used for three-operand addition that significantly reduces the critical path delay at the cost of additional hardware. Hence, a new high-speed and area-efficient adder architecture is proposed using pre-compute bitwise addition followed by carry-prefix computation logic to perform the three-operand binary addition that consumes substantially less area, low power and drastically reduces the adder delay to $O(\log_2 n)$. The proposed architecture is implemented on the FPGA device for functional validation and also synthesized with the commercially available 32nm CMOS technology library. The post-synthesis results of the proposed adder reported 3.12, 5.31 and 9.28 times faster than the CS3A for 32-, 64- and 128- bit architecture respectively. Moreover, it has a lesser area, lower power dissipation and smaller delay than the HC3A adder. Also, the proposed adder achieves the lowest ADP and PDP than the existing three-operand adder techniques.

Guilherme Paim, and Sergio Bampi et.al., (2021), Efficient Kalman filter (KF) designs for real-time mobile applications, such as nano-drones navigation, robots' localization, spacecraft orbit control, GPS positioning, image recognition, and multi sensor data fusion for wearable systems, are key technology goals. The KF is a compute-intensive kernel composed of consecutive complex matrix operations, like multiplications and matrix inversions. The most complex block in the KF is the Kalman gain (KG) function, which involves matrices inversion at each iteration, applying the determinant matrix calculation and division operations. In this article, we combine architectural solutions of different types, for which balancing conflicting low-power and high-performance requirements aiming at real-time KF processing is a key design issue. The key finding in our architectural exploration herein presented is that the KF architectures in semi parallel and sequential forms offer the best balance of circuit area size, power dissipation, and processing speed. Compared to the state-of-the-art solutions, our KF architecture is more efficient, with 2.8 times

fewer arithmetic operators, requiring 3.3 times fewer clock cycles. The usefulness of the developed KF in digital signal processing (DSP) is shown herein by simulations of system identification, noise elimination, and state estimation applications. These figures highlight the results of the KF architecture: the speed of adaptation for the system identification applications with root mean square error (RMSE) of 0.01 after 12 samples, precision level in noise elimination applications with RMSE of 0.13, and reliability in state estimation processes with RMSE less than 10% of system peak response.

Nikhil Advait Gudala, Trond Ytterdal, and John J. Lee et.al., (2021), it is one of the most substantial areas of research in VLSI is the design of power-efficient and high-speed data path logic systems. The speed of addition is constrained in digital adders by the time taken to propagate a carry through the adder. When the previous bit has been added and the carry generated from this addition is propagated onto the next bit, the sum for each bit in an elementary adder is generated sequentially in this manner. In several computer systems, the Carry Select Adder (CSLA) is used to mitigate the issue of carry propagation delay by generating multiple carry bits and then selecting a carry for the desired output. The CSLA, however, is not area-efficient since it utilizes several Ripple Carry Adders (RCA) pairs to produce partial sum and carry by considering carry data, and then multiplexers pick the final sum and carry (mux). The core concept of this study is to use the Binary to Excess-1 Converter (BEC) to achieve high speed and low-power consumption instead of the RCA in the regular CSLA. The results have been analyzed and compared for implementation of three adders (conventional CSA, CSA with RCA, CSA with BEC). The results from the performance evaluations of the adders are compared with each other. All the simulation is carried out in 45nm technology. The delay of CSA and CSA with RCA are same, but the main difference is in reduction of area and power. Similarly, when CSA(BEC) and CSA(RCA) are compared, the area has been reduced by approximately 18.67% and power has been reduced by 25.85%.

Spoorthi H R and Dr. Narendra C P et.al., (2019), this paper presents a high pace and low power data path architecture for Multiply – Accumulate (MAC) unit using Baugh – Wooley multiplier and with the 4:2 Compressor. Generally, MAC contains Partial – Product generation terms (PP unit) and a reduction tree in first

stage. In the following stage there exists an accumulation stage of an adder with the sign extension. A lower operating frequency for the proposed architecture can be used to down size the gates in the available time slack, resulting in a reduction in lower power, delay, and area. We propose a new architecture of MAC with compressor and it efficiently performs either multiply-accumulate or multiply operations for N bit operands. The new proposed architecture was realized with respect to the ISE Xilinx Plan Ahead tool 14.7 and Cadence RTL compiler by giving the comparison between Ripple carry adder, Carry save adder and Brent-Kung adder.

Abhiram T, Ashwin T, Sivaprasad B, Aakash S, Anita J.P et.al., (2017), the factors which play a pivotal role in VLSI system design are power, delay and area. In order to carry out fast arithmetic functions in many data processing processors, Carry Select Adders (CSLAs) are widely used because they are one amongst many of the fast adders that could be used for the high-speed data processing. CSLAs stand out from the remaining conventional adders in terms of performing faster operations. The scope for reduction in area can be observed from the structure of CSLA. Hence a gate level modification has been proposed for the regular CSLA. Based on this modification 16-b, 32-b, 64-b and 128-b SQR T CSLA has been designed. The comparison between regular and modified CSLA showed a reduction in area and power. This work proposes the use of simple, optimized and an efficient gate-level modification to significantly reduce the area and power of carry select adder with a slight increase in delay.

K. Pavan and P. Rahul Reddy et.al., (2021), this paper deals with a comparator with N bits which is efficient in terms of area and operates at high speed with power dissipation as low as possible. Area and power necessities can be scaled down by this proposed digital comparator. In our proposed model comparison is carried out bit wise up to least significant from most significant bit, when bits are equal only then comparison forgoes, which is called as parallel prefix tree structure is exercised. This proposed methodology of comparator structure comprises of two independent modules. Comparison evaluation unit (CEU) and final unit (FU) named first and second units respectively. By depending on the outcomes of CEU, FU module authenticates the result. The presence of structured VLSI methodology in the mooted architecture grants the area derivation through analytical method

relating transistors count and entire total delay can be seen in terms of operand bit width. By making use of 0.18micron CMOS technology at 1 GHz spectre simulations have been performed. Using this proposed design, power consumption reduced by 77.76% and operating speed had increased by 31.92% when compared with existing model.

2.2 LITERATURE SUMMARY

To implement the Modified Dual-CLCG, the three-operand used is carry save adder which consumes less delay and more area. In a parallel prefix Han-Carlson adder for three-operand which is used in proposed consumes less delay and drawback is more area. While we implementing the Montgomery modular multiplication by using carry save adder which consumes more critical path delay with less area. In various parallel prefix adder have been proposed like Ling adder with different topology and Weinberger-Smith Recurrence which consists of less delay and more area. Multiply – Accumulate (MAC) unit using Baugh – Wooley multiplier and with the 4:2 Compressor presents a high pace and low power data path architecture. Generally, MAC contains Partial – Product generation terms (PP unit) and a reduction tree in first stage. The Carry Select Adder (CSLA) is used to mitigate the issue of carry propagation delay by generating multiple carry bits and then selecting a carry for the desired output. The core concept of this study is to use the Binary to Excess-1 Converter (BEC) to achieve high speed and low-power consumption instead of the RCA in the regular CSLA. Multiple constant multiplication with separate generation and selection of partial products is used for designing. The most important benefit of the proposed architecture is the reduced implementation complexity in terms of cell area and power efficiency.

2.3 PROBLEM DEFINITION

To analyse an efficient VLSI architecture for high-speed applications.

CHAPTER 3

REQUIREMENTS ANALYSIS

3.1 MULTIPLEXER

A multiplexer makes it possible for several input signals to share one device or resource, for example, one analog-to-digital converter or one communications transmission medium, instead of having one device per input signal. Multiplexers can also be used to implement Boolean functions of multiple variables. A multiplexer is a combinational circuit that has 2^n input lines and a single output line. Simply, the multiplexer is a multi-input and single-output combinational circuit. The binary information is received from the input lines and directed to the output line. On the basis of the values of the selection lines, one of these data inputs will be connected to the output.

Unlike encoder and decoder, there are n selection lines and 2^n input lines. So, there is a total of 2^N possible combinations of inputs. A multiplexer is also treated as Mux.

Multiplexing is the generic term used to describe the operation of sending one or more analogue or digital signals over a common transmission line at different times or speeds and as such, the device we use to do just that is called the multiplexer.

3.1.1.1 BASIC MULTIPLEXING SWITCH

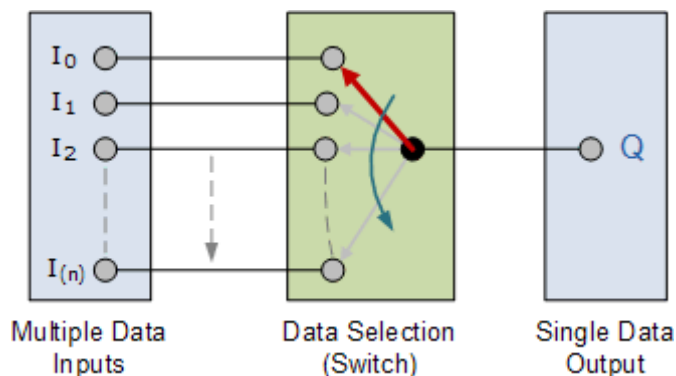


Fig 3.1.1.1: Basic Multiplexing Switch

The rotary switch, also called a wafer switch as each layer of the switch is known as a wafer, is a mechanical device whose input is selected by rotating a shaft. In other words, the rotary switch is a manual switch that you can use to select individual data or signal lines simply by turning its inputs “ON” or “OFF”. So how can we select each data input automatically using a digital device.

In digital electronics, multiplexers are also known as data selectors because they can “select” each input line, are constructed from individual Analogue Switches encased in a single IC package as opposed to the “mechanical” type selectors such as normal conventional switches and relays.

They are used as one method of reducing the number of logic gates required in a circuit design or when a single data line or data bus is required to carry two or more different digital signals. For example, a single 8-channel multiplexer.

Generally, the selection of each input line in a multiplexer is controlled by an additional set of inputs called *control lines* and according to the binary condition of these control inputs, either “HIGH” or “LOW” the appropriate data input is connected directly to the output. Normally, a multiplexer has an even number of 2^n data input. Note that multiplexers are different in operation to *Encoders*. Encoders are able to switch an n-bit input pattern to multiple output lines that represent the binary coded (BCD) output equivalent of the active input. We can build a simple 2-line to 1-line (2-to-1) multiplexer from basic logic NAND.

The *multiplexer*, shortened to “MUX” or “MPX”, is a combinational logic circuit designed to switch one of several input lines through to a single common output line by the application of a control signal. Multiplexers operate like very fast acting multiple position rotary switches connecting or controlling multiple input lines called “channels” one at a time to the output.

Multiplexers, or MUX’s, can be either digital circuits made from high speed logic gates used to switch digital or binary data or they can be analogue types using transistors, MOSFET’s or relays to switch one of the voltage or current inputs through to a single output.

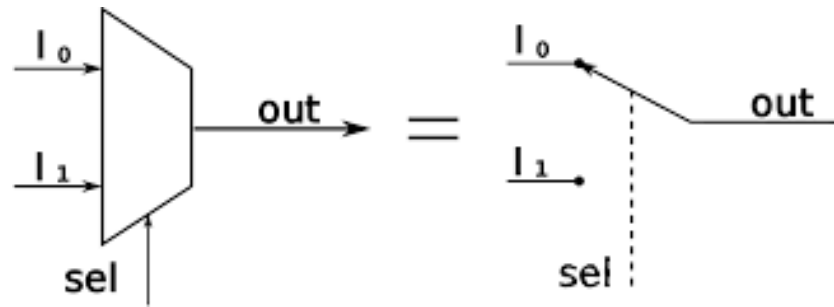


Fig 3.1.1.2: Multiplexer

There are various types of the multiplexer which are as follows:

3.1.2 2x1 MULTIPLEXER:

In 2×1 multiplexer, there are only two inputs, i.e., A_0 and A_1 , 1 selection line, i.e., S_0 and single outputs, i.e., Y . On the basis of the combination of inputs which are present at the selection line S^0 , one of these 2 inputs will be connected to the output. The block diagram and the truth table of the 2×1 multiplexer are given below.

The input A of this simple 2-1 line multiplexer circuit constructed from standard NAND gates acts to control which input (I_0 or I_1) gets passed to the output at Q .

From the truth table above, we can see that when the data select input, A is LOW at logic 0, input I_1 passes its data through the NAND gate multiplexer circuit to the output, while input I_0 is blocked. When the data select A is HIGH at logic 1, the reverse happens and now input I_0 passes data to the output Q while input I_1 is blocked.

So, by the application of either a logic “0” or a logic “1” at A we can select the appropriate input, I_0 or I_1 with the circuit acting a bit like a single pole double throw (SPDT) switch.

As we only have one control line, (A) then we can only switch 2^1 inputs and in this simple example, the 2-input multiplexer connects one of two 1-bit sources to a common output, producing a 2-to-1-line multiplexer. We can confirm this in the following Boolean expression.

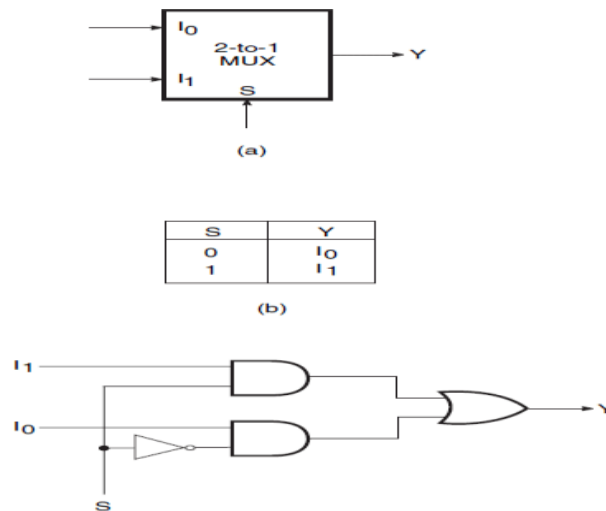


Fig 3.1.2.1: The schematic diagram of 2x1 multiplexer

3.1.3 4x1 MULTIPLEXER

4x1 Multiplexer has four data inputs A_3 , A_2 , A_1 & A_0 , two selection lines s_1 & s_0 and one output Y . The block diagram of 4x1 Multiplexer is shown in the following figure.

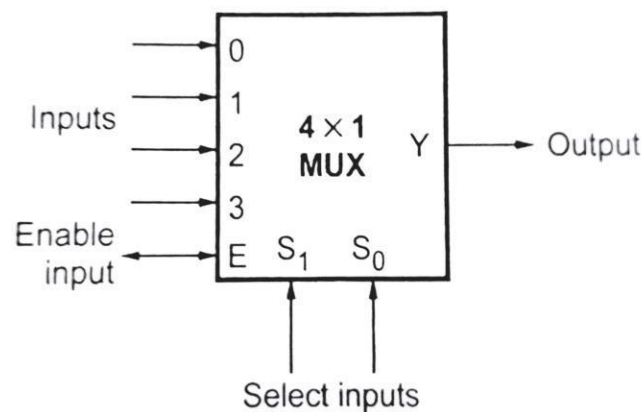


Fig 3.1.3.1: 4x1 Multiplexer

One of these 4 inputs will be connected to the output based on the combination of inputs present at these two selection lines. Truth table of 4x1 Multiplexer is shown below.

Table 3.1.3.1: 4x1 Multiplexer

INPUTS		Output
S_1	S_0	Y
0	0	A_0
0	1	A_1
1	0	A_2
1	1	A_3

3.2 ADDER

A combinational circuit that performs the addition of bits is called an Adder. Each computer has an adder located in its CPU(ALU) that is responsible for the process of addition. There are two types of Adder. They are also used in other parts of the processor, where they are used to calculate addresses, table indices, increment and decrement operators, and similar operations.

Adder is used to add binary numbers. Adder circuit is basically a combinational logic circuit. It is a memory less circuit and performs an operation assigned to it logically by a Boolean expression. The output depends upon the present input at any given time.

An adder or summer is a digital circuit that performs addition of numbers. In many computers and other kind of processors, adders are used not only in the arithmetic logic units, but also in other parts of the processor, where they are used to calculate addresses, table indices, increment and decrement operators, and similar operations.

Although adders can be constructed for many numerical representations, such as binary-coded decimal or excess-3, most common adders operate on binary numbers. In cases where two's complement or one's complement is being used to represent negative numbers; it is trivial to modify an adder into an adder subtractor. Other signed number representations require more logic around the basic adder.

Binary carry-propagate adders have been extensively published, heavily attacking problems related to carry chain problem. Binary adders evolve from linear

adders, which have a delay approximately proportional to the width of the adder, example ripple-carry adder (RCA), to logarithmic-delay adders, such as the carry-look ahead adder (CLA), there are some additional performance enhancing schemes, including the carry-increment adder and the Ling adder that can further enhance the carry chain, however, in Very Large Scale Integration (VLSI) digital system. The most efficient way of offering binary addition involves utilizing parallel-prefix trees, this occurs because they have the regular structure that exhibit logarithmic delay. Parallel-prefix adders compute the sum bit in two steps: one to obtain the carry at each bit, with the next to compute the sum bit based on the carry bit. Unfortunately, prefix trees are algorithmically slower than fast logarithmic adders, such as the carry propagate adders, however, their regular structure promote excellent results when compare to traditional CLA adders.

This happens within VLSI architectures because a carry-look ahead adder, such as the one implemented in one of Motorola's processors, tends to implement the carry chain in the vertical direction instead of a horizontal one, which has a tendency to increase both wire density and fan-in/out dependence. Therefore, although logarithmic adder structures are one of the fastest adders algorithmically, the speed efficiency of the carry-look ahead has been hampered by diminishing returns given the fan-in and fan-out dependencies as well as the heavy wire load distribution in the vertical path. In fact, a traditional carry-look ahead adder implemented in VLSI can actually be slower than traditional linear-based adders, such as the Manchester carry adder.

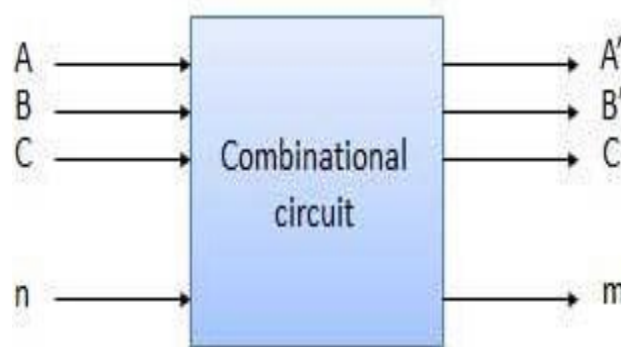


Fig 3.2.1: Block diagram of Adder

3.2.1 HALF ADDER

A half adder is a type of adder, an electronic circuit that performs the addition of numbers. The half adder is able to add two single binary digits and provide the output plus a carry value. It has two inputs, called A and B, and two outputs S (sum) and C (carry). The common representation uses a XOR logic gate and an AND logic gate.

A half adder is used to add two single-digit binary numbers and results into a two-digit output. It is named as such because putting two half adders together with the use of an OR gate results in a full adder. In other words, it only does half the work of a full adder.

The adder works by combining the operations of basic logic gates, with the simplest form using only a XOR and an AND gate. This can also be converted into a circuit that only has AND, OR and NOT gates. This is especially useful since these three simpler logic gate ICs (integrated circuits) are more common and available than the XOR IC, though this might result in a bigger circuit since three different chips are used instead of just one.

So, coming to the scenario of half adder, it adds two binary digits where the input bits are termed as augend and addend and the result will be two outputs one is the sum and the other is carry. To perform the sum operation, XOR is applied to both their inputs, and AND gate is applied to both inputs to produce carry.

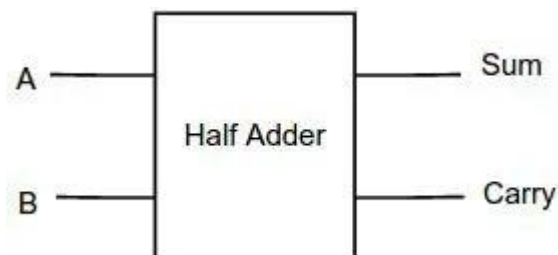


Fig 3.2.1.1: Block Diagram of Half Adder

Whereas in the full adder circuit, it adds 3 one-bit numbers, where two of the three bits can be referred to as operands and the other is termed as bit carried in. The produced output is 2-bit output and these can be referred to as output carry and

sum. By using a half adder, you can design simple addition with the help of logic gates. Let's see an example of adding two single bits.

Table 3.2.1.1: Truth Table of Half Adder

Inputs		Outputs	
A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

in the above table,

1. 'A' and 'B' are the input states, and 'sum' and 'carry' are the output states.
2. The carry output is 0 in case where both the inputs are not 1.
3. The least significant bit of the sum is defined by the 'sum' bit.

$$0+0 = 0$$

$$0+1 = 1$$

$$1+0 = 1$$

$$1+1 = 10$$

These are the least possible single-bit combinations. But the result for 1+1 is 10, the sum result must be re-written as a 2-bit output. Thus, the equations can be written as

$$0+0 = 00$$

$$0+1 = 01$$

$$1+0 = 01$$

$$1+1 = 10$$

The output '1' of '10' is carry-out. 'SUM' is the normal output and CARRY is the carry-out. Now it has been cleared that a 1-bit adder can be easily implemented with the help of the XOR Gate for the output 'SUM' and an 'AND' Gate for the Carry. For instance, when we need to add, two 8-bit bytes together, then it can be implemented by using a full-adder logic circuit. The half-adder is useful when you want to add one binary digit quantities.

A way to develop two-binary digit adders would be to make a truth table and reduce it. When you want to make a three binary digit adder, the half adder addition operation is performed twice. In a similar way, when you decide to make a four-digit adder, the operation is performed one more time. With this theory, it was clear that the implementation is simple, but development is a time taking process.

The simplest expression uses the exclusive OR function:

$$\text{Sum} = A \text{ XOR } B \text{ and Carry} = A \text{ AND } B$$

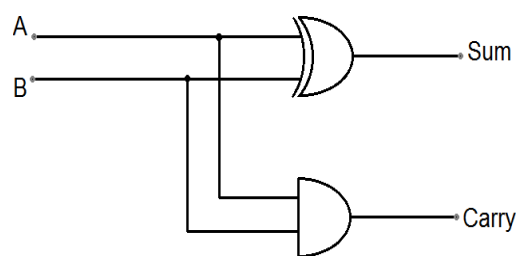


Fig 3.2.1.2: Logic Circuit For Half Adder

And an equivalent expression in terms of the basic AND, OR, and NOT is:

$$\text{SUM} = A.B + A.B'$$

The implementation of half adder can be done through high-speed CMOS digital logic integrated circuits like 74HCxx series which include the SN74HC08 (7408) & SN74HC86 (7486).

The main reason to call these binary adders like Half Adders is, that there is no range to include the carry bit using an earlier bit. So, this is a main limitation of

has once used like binary adder particularly in real-time situations which involve adding several bits. So, this limitation can be overcome by using the full adders.

3.2.2 FULL ADDER

The half adder is used to add only two numbers. To overcome this problem, the full adder was developed. The full adder is used to add three 1-bit binary numbers A, B, and carry C. The full adder has three input states and two output states i.e., sum and carry.

A full adder is a circuit that has two AND gates, two EX-OR gates, and one OR gate. The full adder adds three binary digits. Among all the three, one is the carry that we obtain from the previous addition as C-IN, and the two are inputs A and B. It designates the input carry as the C-OUT and the normal output as S (or SUM).

Just like the Half Adder, the Full Ladder is a combinational type of logic circuit- meaning, it has no storage element. But it has additional logic gates. Thus, it adds the previous carry to generate the complete output. Thus, it is called the Full Adder. One can also designate a Full Adder using one OR gate and two Half Adders. The OR gate here generates a carry that it obtains after the addition. We obtain the sum of these digits in the form of output from the second Half Adder. The equation for the output that you can obtain by the EX-OR gate is the sum of all the binary digits. Here, the output that you obtain from the AND gate is the carry that you obtain by addition. This equation is in the form of a logical operation.

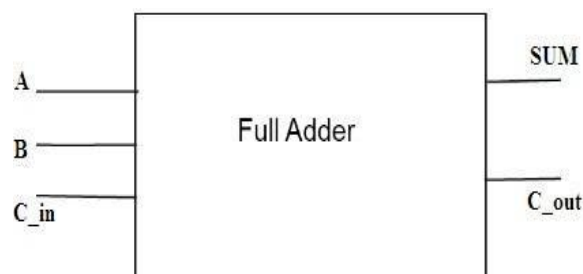


Fig 3.2.2.1: Block Diagram of Full Adder

The full-adder has three inputs and two outputs, whereas half adder has only two inputs and two outputs. The first two inputs are A and B and the third input is an

input carry as C-IN. When a full-adder logic is designed, you string eight of them together to create a byte-wide adder and cascade the carry bit from one adder to the next.

Table 3.2.2.1: Truth Table of Full Adder

Inputs			Outputs	
A	B	C _{in}	Sum	Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

The difference between a half-adder and full-adder is that the output carry is designated as C-OUT and the normal output is represented as S which is 'SUM'. With the above full adder truth-table, the implementation of a full adder circuit can be understood easily. The SUM 'S' is produced in two steps: By XORing the provided inputs 'A' and 'B'. the result of A XOR B is then XORed with the C-IN, this generates SUM and C-OUT is true only when either two of three inputs are HIGH, then the C-OUT will be HIGH. So, we can implement a full adder circuit with the help of two half adder circuits. Initially, the half adder will be used to add A and B to produce a partial Sum and a second-half adder logic can be used to add C-IN to the Sum produced by the first half adder to get the final S output.

If any of the adder logic produces a carry, there will be an output carry. So, C-OUT will be an OR function of the half-adder Carry outputs. Take a look at the implementation of the full adder circuit shown below. The implementation of large logic diagram is possible with the above full adder logic a simpler symbol is mostly used to represent the operation. Given below is a simpler schematic representation of a one-bit full adder.

3.3 MOSFET TECHNOLOGY

The metal–oxide–semiconductor field-effect transistor (MOSFET, MOSFET, or MOS FET) is a type of field-effect transistor (FET), most commonly fabricated by the controlled oxidation of silicon. It has an insulated gate, the voltage of which determines the conductivity of the device. This ability to change conductivity with the amount of applied voltage can be used for amplifying or switching electronic signals. A metal-insulator-semiconductor field-effect transistor (MISFET) is a term almost synonymous with MOSFET. Another synonym is IGFET for insulated-gate field-effect transistor.

The basic principle of the field-effect transistor was first patented by Julius Edgar Lilienfeld in 1925.

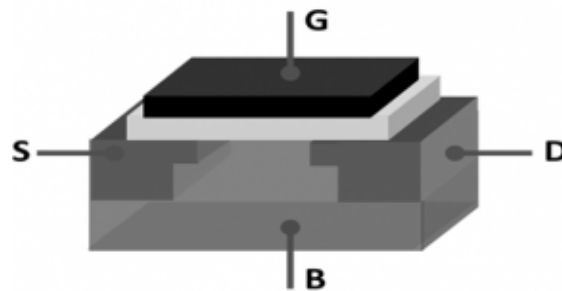


Fig 3.3.1: MOSFET

The main advantage of a MOSFET is that it requires almost no input current to control the load current, when compared with bipolar transistors (bipolar junction transistors/BJTs). In an *enhancement mode* MOSFET, voltage applied to the gate terminal increases the conductivity of the device. In *depletion mode* transistors, voltage applied at the gate reduces the conductivity.

The "metal" in the name MOSFET is sometimes a misnomer, because the gate material can be a layer of polysilicon (polycrystalline silicon). Similarly, "oxide" in the name can also be a misnomer, as different dielectric materials are used with the aim of obtaining strong channels with smaller applied voltages.

The MOSFET is by far the most common transistor in digital circuits, as billions may be included in a memory chip or microprocessor. Since MOSFETs can

be made with either p-type or n-type semiconductors, complementary pairs of MOS transistors can be used to make switching circuits with very low power consumption, in the form of CMOS logic.

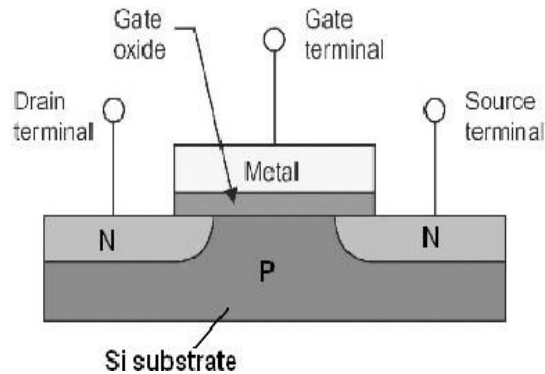


Fig 3.3.2: Schematic diagram of a MOSFET

- The p-type semiconductor forms the base of the MOSFET.
- The two types of the base are highly doped with an n-type impurity which is marked as n+ in the diagram.
- From the heavily doped regions of the base, the terminals source and drain originate.
- The layer of the substrate is coated with a layer of silicon dioxide for insulation.
- A thin insulated metallic plate is kept on top of the silicon dioxide and it acts as a capacitor.
- The gate terminal is brought out from the thin metallic plate.
- A DC circuit is then formed by connecting a voltage source between these two n-type regions.

The IGFET or MOSFET is a voltage-controlled field effect transistor that differs from a JFET in that it has a “Metal Oxide” Gate electrode which is electrically insulated from the main semiconductor n-channel or p-channel by a very thin layer of insulating material usually silicon dioxide, commonly known as glass.

This ultrathin insulated metal gate electrode can be thought of as one plate of a capacitor. The isolation of the controlling Gate makes the input resistance of

the MOSFET extremely high way up in the Mega-ohms ($M\Omega$) region thereby making it almost infinite.

As the Gate terminal is electrically isolated from the main current carrying channel between the drain and source, "NO current flows into the gate" and just like the JFET, the MOSFET also acts like a voltage-controlled resistor where the current flowing through the main channel between the Drain and Source is proportional to the input voltage.

3.4 CNFET TECHNOLOGY

A carbon nanotube field-effect transistor (CNTFET) is a field-effect transistor that utilizes a single carbon nanotube or an array of carbon nanotubes as the channel material instead of bulk silicon in the traditional MOSFET structure. First demonstrated in 1998, there have been major developments in CNTFETs since.

According to Moore's law, the dimensions of individual devices in an integrated circuit have been decreased by a factor of approximately two every two years. This scaling down of devices has been the driving force in technological advances since the late 20th century. However, as noted by ITRS 2009 edition, further scaling down has faced serious limits related to fabrication technology and device performances as the critical dimension shrunk down to sub-22 nm range.^[3] The limits involve electron tunnelling through short channels and thin insulator films, the associated leakage currents, passive power dissipation, short channel effects, and variations in device structure and doping.^[4] These limits can be overcome to some extent and facilitate further scaling down of device dimensions by modifying the channel material in the traditional bulk MOSFET structure with a single carbon nanotube or an array of carbon nanotubes.

The earliest techniques for fabricating carbon nanotube (CNT) field-effect transistors involved pre-patterning parallel strips of metal across a silicon dioxide substrate, and then depositing the CNTs on top in a random pattern. The semiconducting CNTs that happened to fall across two metal strips meet all the requirements necessary for a rudimentary field-effect transistor. One metal strip is the "source" contact while the other is the "drain" contact. The silicon oxide substrate can be used as the gate oxide and adding a metal contact on the back makes the semiconducting CNT gate able.

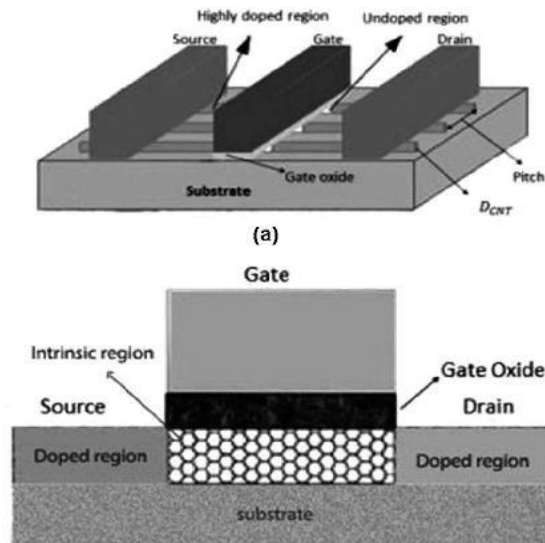


Fig 3.4.1: Block Diagram of CNFET

This technique suffered from several drawbacks, which made for non-optimized transistors. The first was the metal contact, which actually had very little contact to the CNT, since the nanotube just lay on top of it and the contact area was therefore very small. Also, due to the semiconducting nature of the CNT, a Schottky barrier forms at the metal–semiconductor interface,^[17] increasing the contact resistance. The second drawback was due to the back-gate device geometry. Its thickness made it difficult to switch the devices on and off using low voltages, and the fabrication process led to poor contact between the gate dielectric and CNT

3.5 HSPICE SOFTWARE

Hewlett Simulation Program with Integrated Circuit is the full name of HSPICE. Unlike board-level designs made up of individual components, integrated circuits cannot be breadboarded before being manufactured. Furthermore, due to the high cost of photolithographic masks and other manufacturing requirements, it is critical to design the circuit to be as close to perfect as possible before it is designed, until committing to producing an integrated circuit, the industry-standard method of verifying circuit operation at the transistor level is to simulate the circuit with SPICE.

HSPICE is an analog circuit simulator (similar to Berkeley's SPICE-3) capable of performing transient, steady state, and frequency domain analyses.

Existing SPICE decks created for SPICE3 can be easily modified to run under HSPICE, or can be rewritten to take advantage of features not available in SPICE-3. HSPICE generally has better convergence than SPICE-3 and, because it is a commercial product (from Meta-Software), is better supported. It also allows hierarchical node naming, circuit optimization, input, output, and behavioral algebraics for parameterized cells, and interactive waveform viewing with Meta Waves. This handout contains basic information about HSPICE and MetaWaves, which should be sufficient for performing simple simulations. For more detailed information, consult the HSPICE USERS MANUAL(available from MetaSoftware directly at 800-346-5953(note - ONLY use this number to order manuals: alltechnical questions should be directed to your TA)), or you can view an on-line version of the manual in the /usr/class/ee/hspice/docs/Postscript directory. To view a section of the manual, you can use one of the postscript viewers (such as ghost view, or ghost script). For example, to view the 15_mosfet_introduction.ps section from one of the elaine workstations.

SPICE is a powerful general purpose analog circuit simulator that is used to verify circuit designs and to predict the circuit behavior. This is of particular importance for integrated circuits. It was for this reason that SPICE was originally developed at the Electronics Research Laboratory of the University of California, Berkeley (1975), as its name implies.

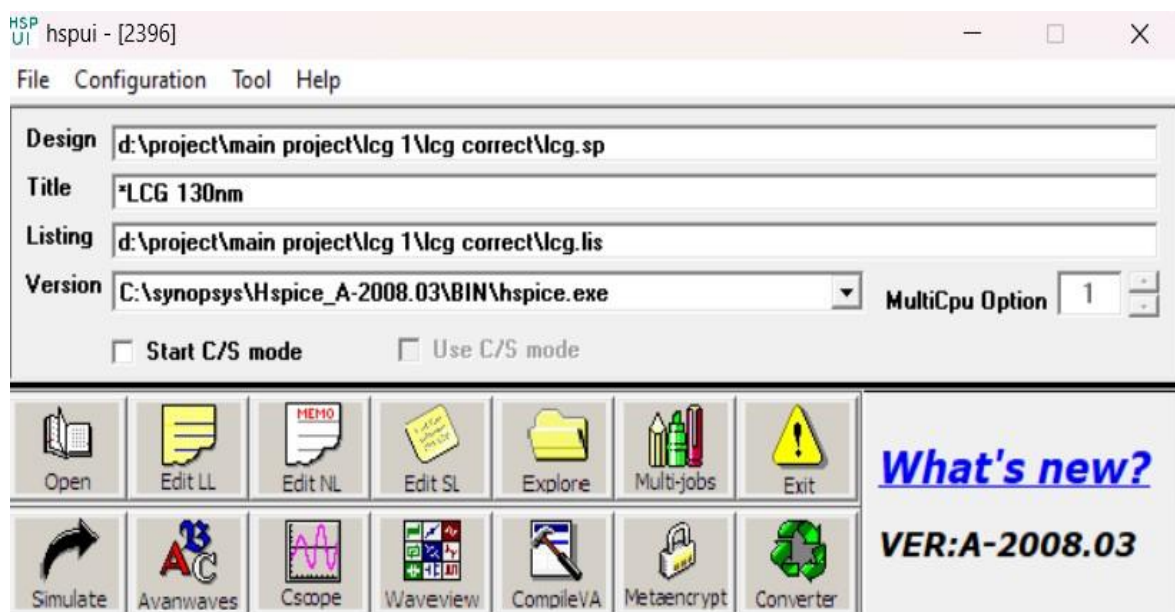


Fig 3.5.1: HSPICE Software

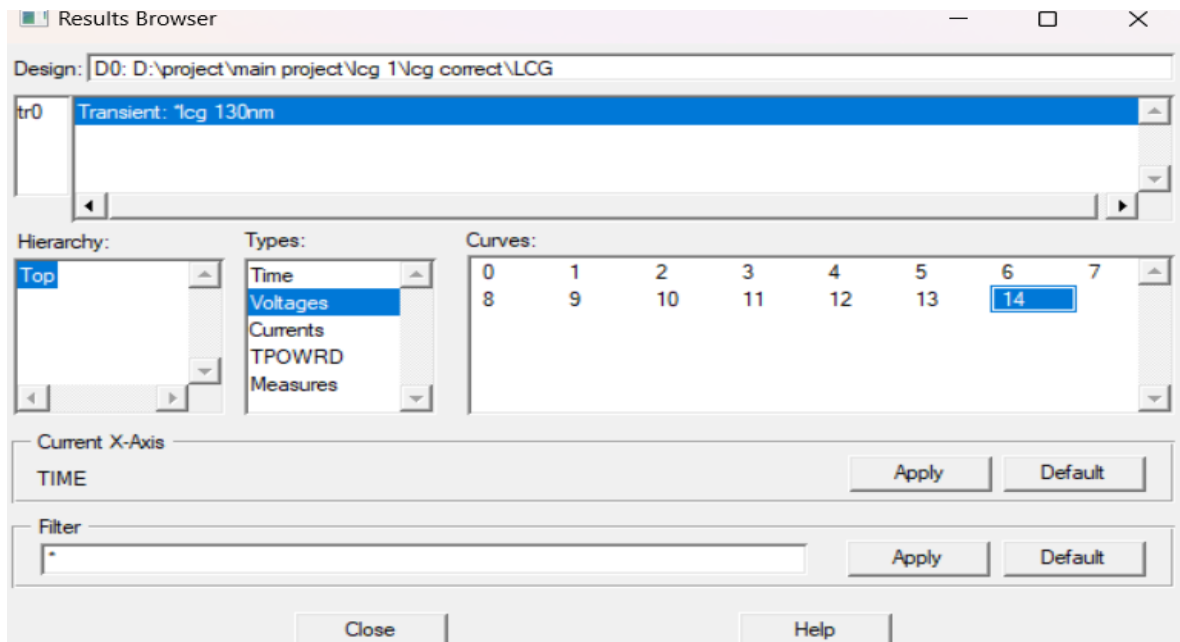


Fig 3.5.2: Selecting Screen

3.6 COSMOSCOPE

CosmoScope is a software which is used to calculate the delay value. With powerful analysis and measurement capabilities, patented waveform-calculator technology, and scripting language based on the industry standard Tcl/Tk, CosmoScope offers unparalleled capability and flexibility to analyze design performance and ensure design quality. CosmoScope supports all Synopsys and SaberHDL. CosmoScope can also import industry common formats such as FSDB, Touchstone, and VCD. Even measured or lab data can be imported thanks to CosmoScope ability to import tabular text or CSV files.

- Performs post-processing of analog and digital simulation results
- Automatically annotates graphs with design information using true
- WYSIWYG graphics, including arrows, shapes and text
- Annotates graphs with 50 types of measurements for immediate visual feedback on design performance
- Saves and restores graphs for further editing—entire CosmoScope sessions can be saved and restored to pick up where you left off
- Streamlines the design process through tight integration with Synopsys' Cosmos™ full-custom design environment and third-party design frameworks.

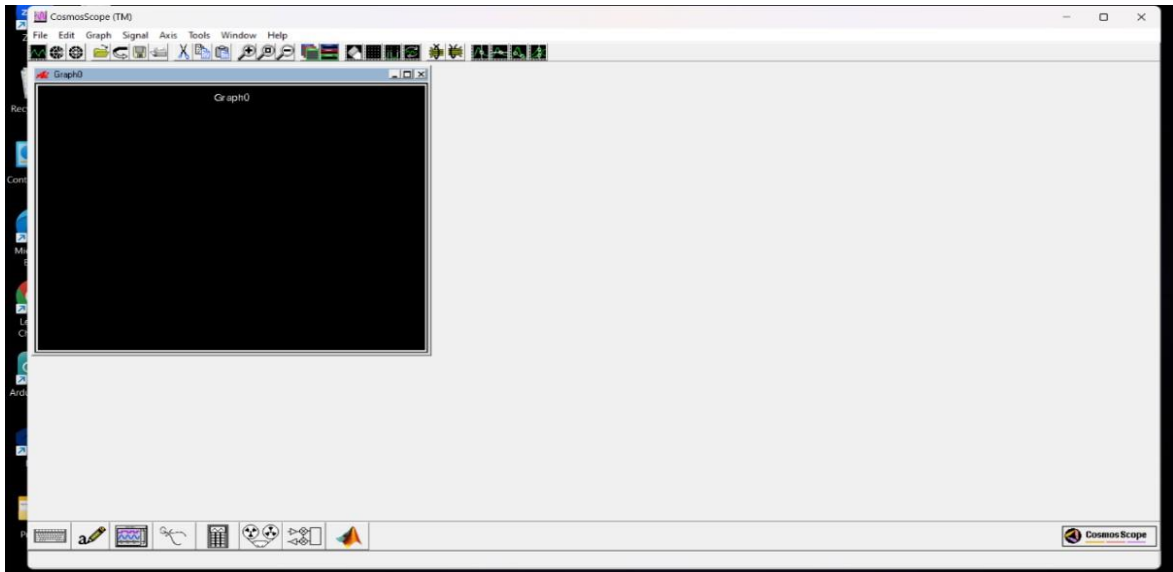


Fig 3.6.1: CosmoScope Software

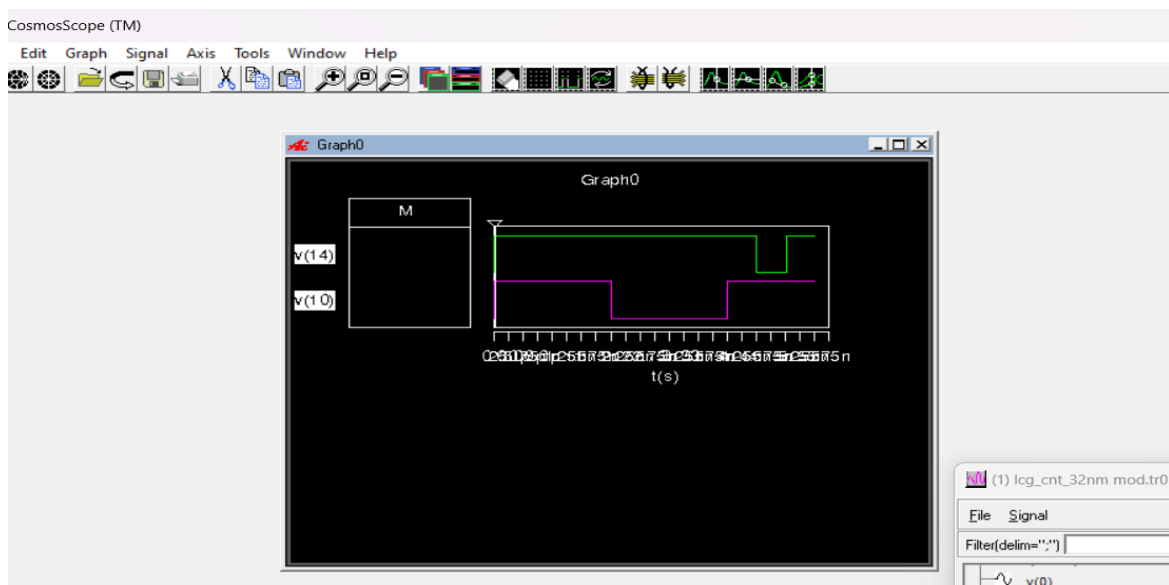


Fig 3.6.2: Result Graph

3.6.1 POWER

Electric power is the rate at which electrical energy is transmitted by an electric circuit per unit of time. It is the pace at which work is completed. Electrical power is the rate at which electrical energy is transmitted by an electric circuit per unit time in terms of an electric circuit. CMOS devices have very low static power consumption, which is the result of leakage current. This power consumption occurs when all inputs are held at some valid logic level and the circuit is not in charging states.

The upper MOSFET is off for a high voltage input, and the lower MOSFET is on given a low input. The circuit is an inverter.

The term of power leads us to the term of energy. Instantaneous power of the certain circuit element is the product of the current and voltage through this element and described by the formula. The instantaneous power is always time dependent, measured with Watts.

The unit of the Electric Power = watt

$$\text{Power} = V \cdot I (\text{W})$$

$$I = Q/t$$

Where:

W = Power in watts

V = Potential in volts

I = Current in amps

Q = Charge in coulombs

t = Time in seconds

3.6.2 DELAY

The time it takes for a signal to reach its destination is referred to as delay. The time it takes for the consequence of a shift in input to be noticeable at the output is known as the delay of a logic gate. The propagation delay of the CMOS inverter is determined by the time it takes to charge and discharge the capacitances present in the logic circuit. Figure 1b shows the circuit for analysis of the propagation delay of the inverter under condition that it is driving an identical inverter. It is defined as the delay between an input and output pair of a cell, when a near zero slew is applied to the input pin and the output does not see any load condition. It is predominantly caused by the internal capacitance associated with its transistor.

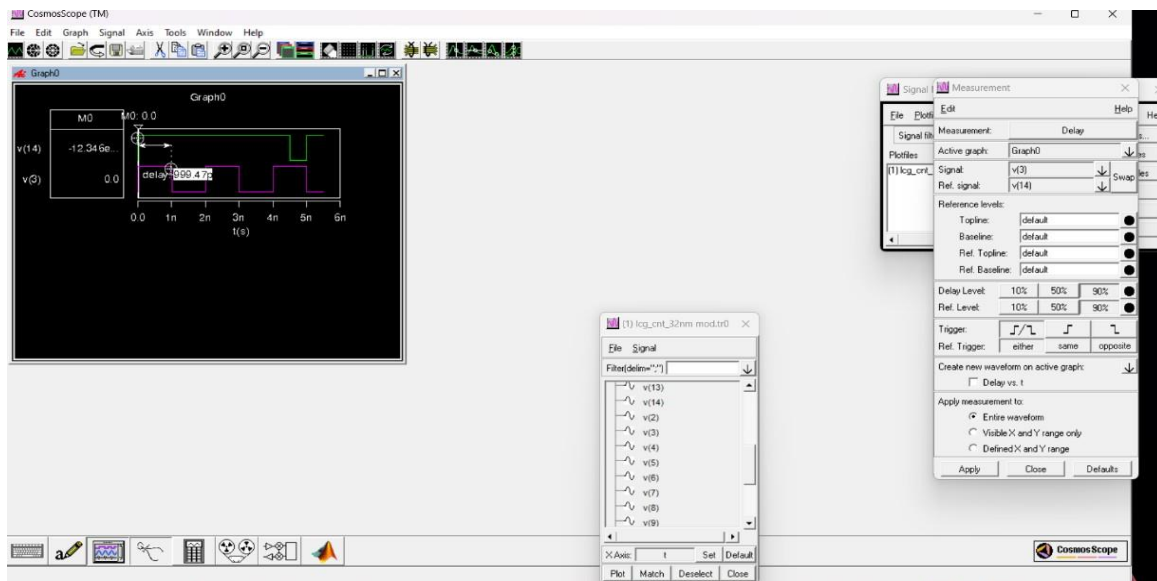


Fig 3.6.2.1: Output of Delay

3.6.3 POWER DELAY PRODUCT

Power Delay Product is the product of the power and delay, in digital electronics, the power–delay product (PDP) is a figure of merit correlated with the energy efficiency of a logic gate or logic family. Also known as switching energy, it is the product of power consumption P (averaged over a switching event) times the input–output delay or duration of the switching event D . The power-delay product measures the energy dissipated in a CMOS circuit per switching operation. Since the energy per switching event is fixed, the PDP describes a fundamental tradeoff between speed and power dissipation – if we operate at high speeds, we will dissipate a lot of power.

$$\text{PDP} = \text{Power} * \text{Delay}$$

CHAPTER 4

EXISTING METHODOLOGY

In this chapter discuss about MDCLCG algorithm to measure the average power and delay.

4.1 MDCLCG

Modified dual-CLCG (MDCLCG) is the most efficient PRBG method amongst the existing PRBG methods which is suitable for stream-cipher based hardware security. However, the security strength of the MDCLCG method linearly depends on the bit size of the congruential modulus. It is polynomial-time unpredictable and secure if $n \geq 32$ - bits. The hardware architecture of the MDCLCG method is based on LCG, as shown in Fig in which three-operand modulo- $2n$ adder is the primary computational arithmetic block.

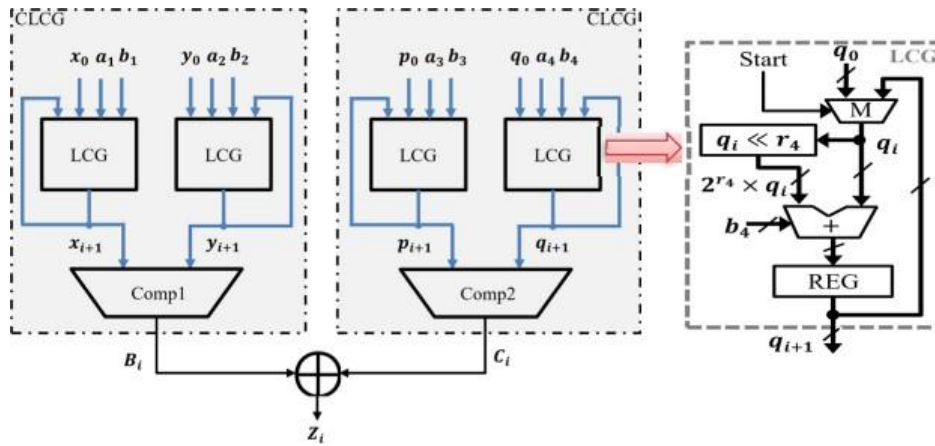


Fig 4.1: MDCLCG architecture (Modified Dual Coupled Linear Congruential generator)

The MDCLCG architecture is developed with four three-operand modulo- $2n$ carry-save adders (CS3A) and two magnitude comparators along with four registers and multiplexers. The longer carry propagation gate delay in CS3A adder influences the performance of MDCLCG architecture with an increase of bit size. Therefore, in this section, the performance metrics of the MDCLCG are measured by replacing the CS3A adder with the HHC3A and proposed adder architectures.

4.2 LCG (Linear Congruential Generator)

There is a popular method and most used to generate random number called linear congruential generator.

This idea was introduced by Lehmer according to sequential formula.

$$X_{n+1} = (aX_n + c) \bmod m$$

Where m is modulus, a is multiplier, c is increment. Parameters a , c and m have to be chosen carefully in order to avoid repetition of similar numbers before m . Park & Miller suggested a good result will be obtained by choosing $c=0$.

The modulus m should be a large prime integer, multiplier a will be an integer in the range $2, 3, \dots, m-1$. The cycle length of LCG will never exceed modulus m , but it can be maximized using three following conditions:

- c is relatively prime to modulus m ,
- multiplier $a-1$ is a multiple of every dividing modulus m ,
- multiplier $a-1$ is a multiple of four when modulus m is a multiple of four.

4.3 LCG CIRCUIT DESIGN

Fig.4.3.2 shows block diagram LCG operation in general (seed is ignored). It requires multiplier, adder, comparator and subtractor blocks. Multiplier is used to multiply previous random value X with a , then add with increment c . After that it is compared to modulus m . The number is considered as random number if it is smaller or equal to m . Otherwise, the number then subtracted with m . The result of this is then consider as random number.

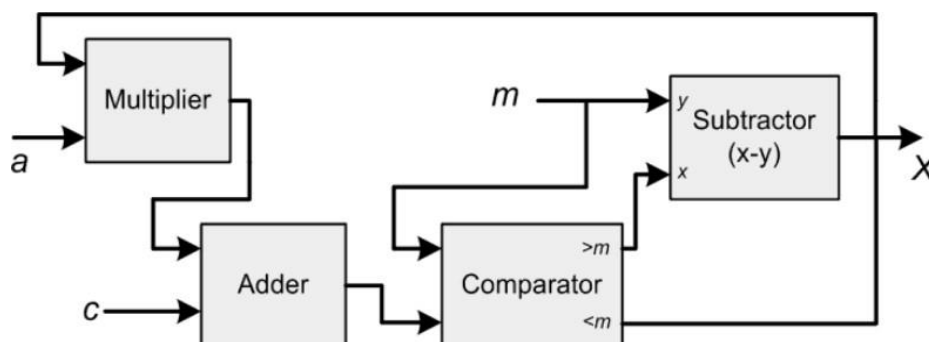


Fig 4.3.1 Block diagram of LCG operation

The block diagram of Fig. 3.3 is involving arithmetic operation such as multiplication, addition, subtraction and comparison. In order simplify the process, the circuit is designed using word lengths reduction technique that has been suggested. Then subtractor and comparator blocks can be removed, as shown in the Fig. 2.

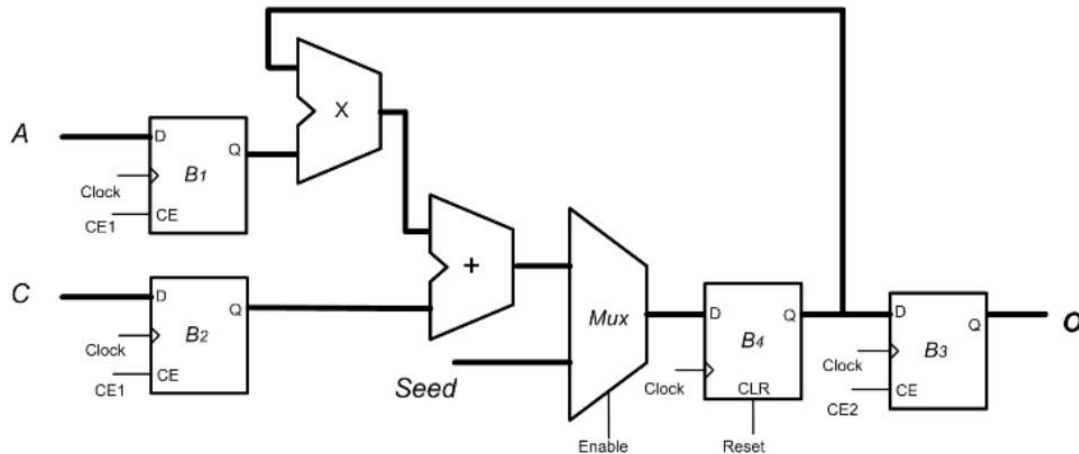


Fig 4.3.2: General circuit diagram of LCG

The designed circuit consist of a multiplexer, a multiplier, an adder, a buffer (required clear), and three buffers (required enable). Ports input Seed, A and C are used to pass initial value, multiplier and increment into the circuit. Meanwhile, port O is used to taking out the resulted random numbers.

The circuit is controlled by two signals enable and reset. Initially, signal reset has to be HIGH (enable=LOW) to clear the previous stored values in the buffer B4. Signal enable determine when the operation should be started. Pre-defined value (seed), increment and multiplier have to be available at the input ports before enable goes HIGH (reset=LOW). Then, every time clock goes HIGH, a random number resulted. Fig.3shows the circuit of the two signal controls.

4.4 ADVANTAGES OF LCG

- Very fast to implement.
- Requires no storage of the numbers, only the most recent value.
- CLCG provides an efficient way to calculate pseudo-random numbers.
- LCG algorithm in inexpensive to use.

4.5 DISADVANTAGES OF LCG

- LCGs are not intended and must not be used for cryptographic applications.
- It is not free of sequential correlation on successive calls.

4.6 D-FLIP FLOP

A D (or Delay) Flip Flop is a digital electronic circuit used to delay the change of state of its output signal (Q) until the next rising edge of a clock timing input signal occurs.

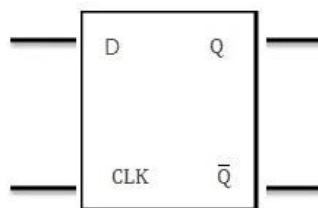


Fig 4.6.1: D-flip flop

The truth table for the D Flip Flop is shown in below table.

Table 4.6.1: Truth table of d-ff

D	CLK	$\bar{Q}$
0	Rising Edge	0
1	Rising Edge	1

The D Flip Flop acts as an electronic memory component since the output remains constant unless deliberately changed by altering the state of the D input followed by a rising clock signal.

The D Flip Flop is a building block Shift Registers. For example, by cascading eight D Flip Flops in sequence, a byte (8-bits) of information can be stored after 8 clock cycles.

By connecting the inverting output of the D Flip Flop to the D input, a simple divide by two circuit is created i.e., the D output changes state at half the frequency of the clock signal. By cascading D flip flops and through appropriate design of external combinational logic gates, a countdown timer can be created.

4.7 EXISTING D-FLIP FLOP

This paper proposes a new D flip-flop configuration based on differential cascode voltage switch with pass-gate logic. The circuit is able to reduce the transition time from the input to output. The flip-flop was implemented in 0.18 μm CMOS technology.

- D-flip flop is used in digital circuit for temporary storage of data or a delay element
- D-flip flop is written in CMOS technology MOSFET under 32nm, 130nm and CNTFET under 32nm.
- Fig 4 shows the existing CMOS technology D-flip flop which have high power, high delay and high PDP
- In the below D-flip flop eight transistors are taken (four PMOS transistors and four NMOS transistors). This DFF is taken in the LCG circuit diagram to measure the power, delay and PDP.
- The LCG circuit diagram has DFF, multiplier, adder and mux.

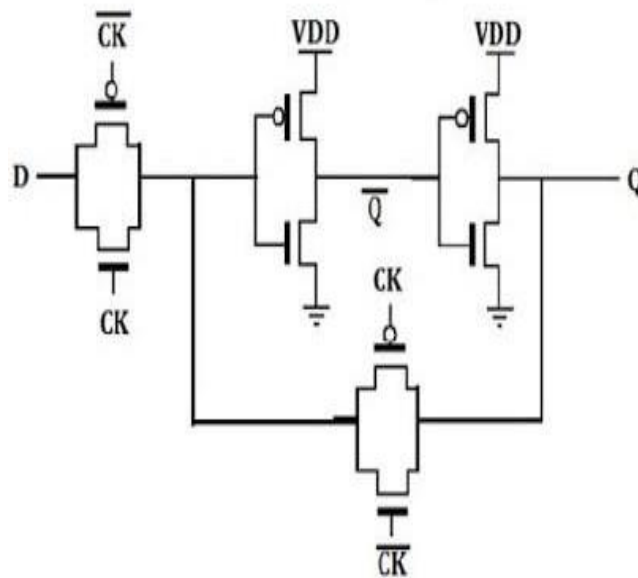


Fig 4.7.1: CMOS Technology DFF

CHAPTER 5

PROPOSED SYSTEM

In order to improve efficiency, the existing Dual Coupled Linear Congruential generator will be analyzed using 130nm and 32nm CMOS Technology by Synopsys HSpice Tool. Further the existing architecture will be modified by using low power VLSI Techniques and the results will be compared and analyzed in terms of power, speed and power delay using 130nm and 32nm technology.

5.1 PROPOSED LOW POWER D-FLIP FLOP

Low Power flip-flops are flip-flops that are designed for low-power electronics, such as smartphones and notebooks. A flip-flop, or latch, is a circuit that has two stable states and can be used to store state information.

In most VLSI devices, maximum power dissipation is due to the clock network and clocked sequential elements. It can account to anywhere between 25% - 40% of the total power according to the variation of parameters like clock frequency, deeper pipelining etc. in a design. Sequential elements, latches and flip flops dissipate power when there is switching in their internal capacitance. This may happen with every clock transition/pulse into the sequential element. Sometimes the sequential elements need to change their state, but sometimes they retain their state and their output remains the same, before and after the clock pulse. This leads to unnecessary dissipation of power due to clock transition. If flip flops are designed in such a way that they are able to gate the clock with respect to their own internal data path, only to use the clock when needed, power dissipation can be brought down.

D-flip flop is used in digital circuit for temporary storage of data or a delay element. D-flip flop is written in CMOS technology MOSFET using 130nm and 32nm technology.

The LCG circuit diagram has DFF, multiplier, adder and mux. To measure the power, delay and PDP, first we have to write the code for each device in the LCG block diagram in HSPICE. Coming to the DFF, we will write the nodes for each transistor as D=2, CK=3, CKbar=4. After giving the nodes we will the code and stimulation is done in the HSPICE software.

After simulation it has been proven that compare to existing system, proposed LCG Architecture serves efficient in terms of power, delay and power delay product. Capacitance of a diode is directly proportional to its size. This enhances its speed in turn.

Fig 5.1.1: Modified CMOS technology DFF

- The above figure 4.1 which is seen in the existing will be modified to get low power, low delay and low PDP than existing.
- We'll combine the PMOS and NMOS transistors from the present DFF into a single NMOS transistor. because NMOS is faster and more effective than PMOS. Due to their smaller junction areas than PMOS circuits, NMOS circuits are faster.
- The modified DFF has seven transistors (two PMOS transistor and four NMOS transistor) which has low power, low delay and low PDP than existing DFF
- replace the modified DFF from existing in LCG circuit diagram to measure the power, delay and PDP.
- Stimulation is done in HSPICE software
- The power, delay and PDP will reduce in MOSFET under 32nm, 130nm and CNTFET under 32nm.

CHAPTER 6

RESULT AND DISCUSSION

6.1 STIMULATION RESULTS

The power, delay and PDP reduced in modified DFF than existing. And the more power, delay and PDP is reduced in CNTFET under 32nm. So, it is better to use the CNTFET than MOSFET under 32nm, 130nm.

The below figures show the output waveform of both modified DFF and LCG circuit diagram in CNTFET under 32nm.

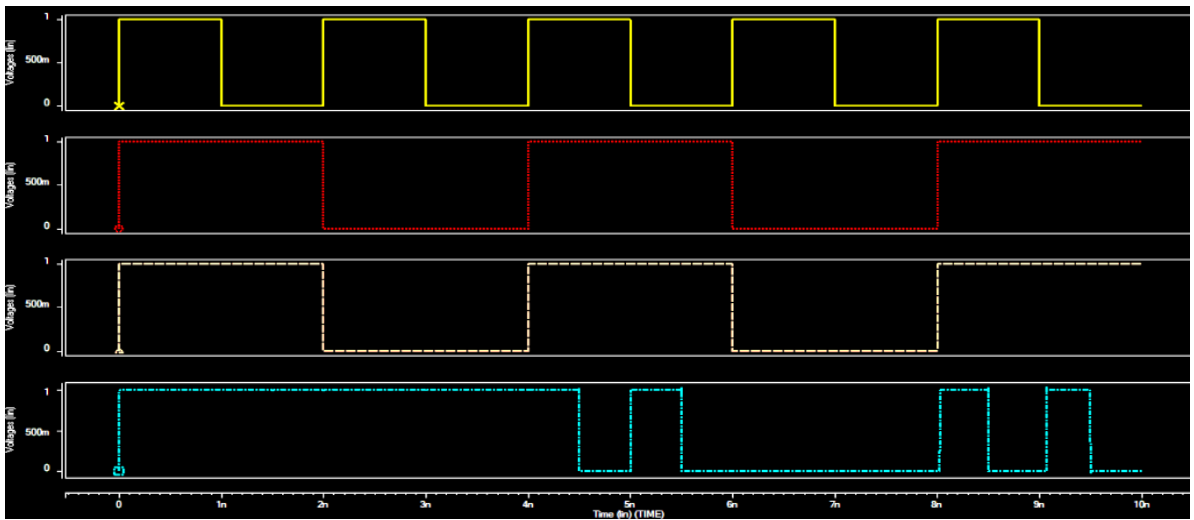


Fig 6.1.1: Stimulation of proposed LCG in CNTFET

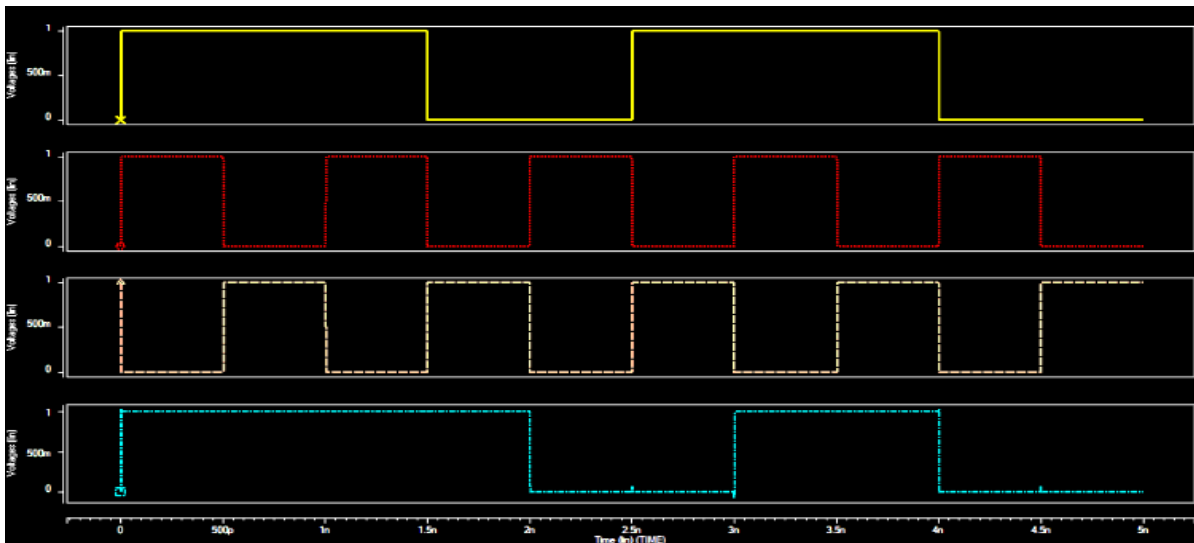


Fig 6.1.2: Stimulation of Proposed Low Power D-Flip Flop

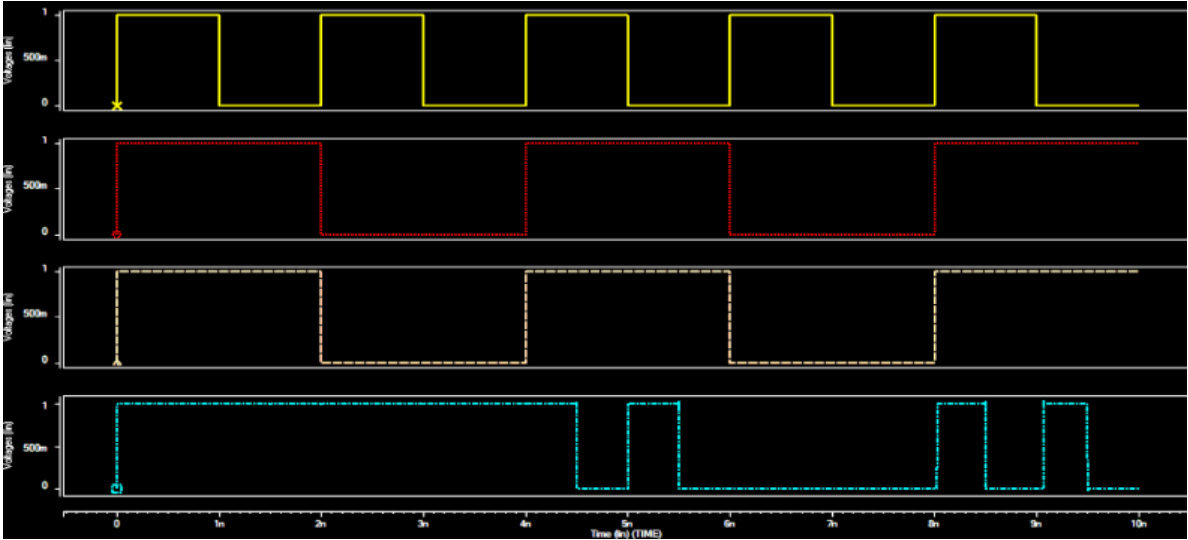


Fig 6.1.3: Stimulation of proposed LCG

The below table is readings of power, delay and PDP in MOSFET under 32nm, 130nm of adder, and, DFF, proposed DFF, mux, LCG and proposed LCG and comparisons are done between existing and proposed DFF and LCG.

6.2 PERFORMANCE ANALYSIS

The below table is readings of power, delay and PDP in MOSFET under 32nm, 130nm of adder, and, DFF, proposed DFF, mux, LCG and proposed LCG and comparisons are done between existing and proposed DFF and LCG. Then the result will be compared of both Proposed and Existing Methodology.

Since MOSFETs can be made with n-type semiconductors, complementary pairs of MOS transistors can be used to make switching circuits with very low power consumption, in the form of CMOS logic.

6.2.1 LCG-MOSFET

The below table is Represents the output of the Proposed LCG, here the results have been compared of both 130nm and 32nm technology by using MOSFET technology, 32nm is low power, delay and PDP (Power Delay Product) than 130nm

Table 6.2.1.1: LCG-MOSFET under 130nm

Device	130nm		
	Power(W)	Delay(s)	PDP(J)
ADDER	2.18×10^{-5}	1.6162×10^{-12}	0.035×10^{-15}
AND	1.79×10^{-5}	47.475×10^{-12}	0.849×10^{-15}
DFF	1.58×10^{-5}	563.86×10^{-12}	8.908×10^{-15}
Proposed DFF	1.45×10^{-5}	547.01×10^{-12}	7.97×10^{-15}
MUX	1.49×10^{-5}	1.6529×10^{-12}	0.002×10^{-15}
LCG	23.03×10^{-5}	133.59×10^{-12}	30.76×10^{-15}
Proposed LCG	14.48×10^{-5}	126.36×10^{-12}	18.29×10^{-15}

Table 6.2.1.2: LCG-MOSFET under 32nm

Device	32nm		
	Power(W)	Delay(s)	PDP(J)
ADDER	3.51×10^{-5}	0.0076×10^{-12}	0.0002×10^{-15}
AND	0.01×10^{-5}	12.819×10^{-12}	0.0025×10^{-15}
DFF	0.52×10^{-5}	499.84×10^{-12}	2.6456×10^{-15}
Proposed DFF	0.02×10^{-5}	5.9172×10^{-12}	0.0014×10^{-15}
MUX	1.82×10^{-5}	0.2023×10^{-12}	0.0036×10^{-15}
LCG	1.77×10^{-5}	50.085×10^{-12}	0.8885×10^{-15}
Proposed LCG	1.05×10^{-5}	45.859×10^{-12}	0.4847×10^{-15}

6.2.2 LCG-CNFET

The power, delay and PDP reduced in modified DFF than existing. And the more power, delay and PDP is reduced in CNTFET under 32nm. So, it is better to use the CNTFET than MOSFET under 32nm, 130nm.

Table 6.2.2.1: LCG-CNFFET under 32nm

Device	32nm: Operating voltage: 1v, Operating frequency: 1Ghz		
	Power	Delay	PDP
ADDER	0.0105×10^{-5}	0.1262P	0.001×10^{-17}
AND	0.0007×10^{-5}	0.0575P	0.0004×10^{-17}
DFF	0.0103×10^{-5}	0.0776P	0.008×10^{-17}
Proposed DFF	0.0088×10^{-5}	0.0778P	0.006×10^{-17}
MUX	0.1037×10^{-5}	0.0966P	0.01×10^{-17}
LCG	1.658×10^{-5}	8.1163P	10.34×10^{-17}
Proposed LCG	0.1630×10^{-5}	7.5257P	1.226×10^{-17}

CHAPTER 7

CONCLUSION

To achieve optimal system performance while maintaining physical security, it is necessary to implement the cryptography algorithms on hardware. Modular arithmetic such as modular exponentiation, modular multiplication and modular addition is frequently used for the arithmetic operations in various cryptography algorithms. Therefore, the performance of the cryptography algorithm depends on the efficient implementation of the congruential modular arithmetic operation. The most efficient approach to implement the modular multiplication and exponentiation is the Montgomery algorithm whose critical operation is based binary addition is also a primary arithmetic operation in the linear congruential generator (LCG) based pseudo-random bit generators (PRBG) such as coupled LCG (CLCG), modified dual-CLCG (MDCLCG) and coupled variable input LCG (CVLCG). Modified dual-CLCG (MDCLCG) is the most secure and highly random PRBG method among all the LCG-based and other existing PRBG methods. It is polynomial-time unpredictable and secure if for bit greater than 32-bits. Therefore, the security of the MDCLCG enhances with the increase of operand size. However, the area and critical path delay increases linearly since its hardware architecture consists of four three operand modulo- $2n$ adders, two comparators, four multiplexer's area in previous papers. Hence, the performance of the MDCLCG can be improved by the efficient implementation of the three-operand adder

An efficient of linear congruential generator using proposed low power has been analysed in terms of power, delay and power delay product using 130nm and 32nm MOSFTE technology. LCG circuit diagram contains DFF, adder, multiplier and multiplexer. The readings were compared of using 130nm and 32nm between existing and proposed LCG and DFF. After analysis it is proven that proposed low power flip flop based LCG serves better performance by 72% when compare to existing LCG. Further, the proposed LCG can be used for low power and high-speed digital applications. Design and implementation of linear congruential generator was successfully. LCG circuit diagram contains DFF, adder, multiplier and mux to measure power, delay and PDP in MOSFET under 32nm, 130nm and CNTFET under 32nm. The readings are compared of MOSFET under 32nm, 130nm and CNTET under 32nm between existing and proposed LCG and DFF. Whether in the

MOSFET and CNTFET, the CNTFET is the best and it has the low power among all these. After stimulation we will see the power which is nearly 0.00158% and 0.0005293% which is high other than other DFF in MOSFET under 32nm, 130nm and 0.00001037% in CNTFET under 32nm. And the delay we get is 563.86p, 499.84p in MOSFET under 32nm, 130nm and 0.077p in CNTFET under 32nm. Finally, the product of power and delay will be the PDP (Power -Delay product).

REFERENCES

- [1] Erdem S. S, Yanik T, and A. Celebi, "A general digit-serial architecture for montgomery modular multiplication," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 5, pp. 1658–1668, May 2017.
- [2] Han T and Carlson D. A, "Fast area-efficient VLSI adders," in *Proc. IEEE 8th Symp. Comput. Arithmetic (ARITH)*, May 1987, pp. 49–56.
- [3] Harris D. L, "Parallel prefix networks that make tradeoffs between logic levels, fanout and wiring racks," U.S. Patent 0 225 706 A1, Nov. 11, 2004.
- [4] Islam M. M, Hossain M. S, Hasan M. K, M. Shahjalal, and Y. M. Jang, "FPGA implementation of high-speed area-efficient processor for elliptic curve point multiplication over prime field," *IEEE Access*, vol. 7, pp. 178811–178826, 2019.
- [5] Jackson R and Talwar S, "High speed binary addition," in *Proc. Conf. Rec. 38th Asilomar Conf. Signals, Syst. Comput.*, vol. 2. Pacific Grove, CA, USA, Nov. 2004, pp. 1350–1353.
- [6] Katti R. S and Srinivasan S. K, "Efficient hardware implementation of a new pseudo-random bit sequence generator," in *Proc. IEEE Int. Symp. Circuits Syst.*, Taipei, Taiwan, May 2009, pp. 1393–1396.
- [7] Kim T, Jao W, and Tjiang S. K, "Circuit optimization using carry-saveadder cells," *IEEE Trans. Comput. -Aided Design Integr. Circuits Syst.*, vol. 17, no. 10, pp. 974–984, Oct. 1998.
- [8] Kuang S.-R, Wang J.-P, Chang K.-C., and H.-W. Hsu, "Energy-efficient high-throughput montgomery modular multipliers for RSA cryptosystems," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 21, no. 11, pp. 1999–2009, Nov. 2013.
- [9] Kuang S.-R, Wu K.-Y, and R.-Y. Lu, "Low-cost high-performance VLSI architecture for Montgomery modular multiplication," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 24, no. 2, pp. 434–443, Feb. 2016.
- [10] Kumar Panda A and K. Chandra Ray, "A coupled variable input LCG method and its VLSI architecture for pseudorandom bit generation," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 4, pp. 1011–1019, Apr. 2020.
- [11] Ling H, "High-speed binary adder," *IBM J. Res. Develop.*, vol. 25, no. 3, pp. 156–166, Mar. 1981.

- [12] Liu Z, GroBschadl J, Hu Z, Jarvinen K, Wang H, and Verbaauwhede I, "Elliptic curve cryptography with efficiently computable endomorphisms and its hardware implementations for the Internet of Things," *IEEE Trans. Comput.*, vol. 66, no. 5, pp. 773–785, May 2017.
- [13] Liu Z, Liu D, and Zou X, "An efficient and flexible hardware implementation of the dual-field elliptic curve cryptographic processor," *IEEE Trans. Ind. Electron.*, vol. 64, no. 3, pp. 2353–2362, Mar. 2017.
- [14] Montgomery P. L, "Modular multiplication without trial division," *Math. Comput.*, vol. 44, no. 170, pp. 519–521, Apr. 1985.
- [15] Panda A. K and Ray K. C, "Modified dual-CLCG method and its VLSI architecture for pseudorandom bit generation," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 66, no. 3, pp. 989–1002, Mar. 2019.
- [16] Panda A. K and Ray K. C, "Design and FPGA prototype of 1024-bit Blum-Blum-Shub PRBG architecture," in *Proc. IEEE Int. Conf. Inf. Commun. Signal Process. (ICICSP)*, Singapore, Sep. 2018, pp. 38–43.
- [17] Pandey K. S, Goel D. K. B. N and Shrimali H, "An ultra-fast parallel prefix adder," in *Proc. IEEE 26th Symp. Comput. Arithmetic (ARITH)*, Kyoto, Japan, Jun. 2019, pp. 125–134.
- [18] Parhami B, *Computer Arithmetic: Algorithms and Hardware Design*. New York, NY, USA: Oxford Univ. Press, 2000.
- [19] Rezai A and Keshavarzi P, "High-throughput modular multiplication and exponentiation algorithms using multibit-scan–multibit-shift technique," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 23, no. 9, pp. 1710–1719, Sep. 2015.
- [20] Weste N and Eshraghian K, *Principles of CMOS VLSI Design—A Systems Perspective*. Reading, MA, USA: Addison-Wesley, 1985.

STANDARDS AND CONSTRAINTS

Standards and constraints are essential in project analysis of an efficient linear congruential generator architecture for digital applications. These helps ensure that the generator meets specific criteria and is suitable for the intended application. Standards such as IEEE and ISO provide guidelines for design, performance, and reliability. Constraints, such as power consumption, size, and speed, also play a critical role in determining the generator's feasibility and usability. The project analysis must consider and optimize both standards and constraints to develop a high-performing and efficient linear congruential generator architecture for digital applications.

APPENDIX

A. CERTIFICATES OF RESEARCH PAPER



Advancing Technology
for Humanity



22 - 24, February 2023
Erode | Tamil Nadu | India

**2023 Fifth IEEE International Conference on
Electrical, Computer and Communication
Technologies**

22 - 24, February 2023 | www.icecct.com

EC 3085
Peer Reviewed

CERTIFICATE

This certificate is presented to

Sujitha .A
Department of Electronics and Communication,
Sathyabama Institute of Science and Technology,
Chennai, India

for presenting the research paper entitled "Analysis of an Efficient Linear Congruential Generator Architecture for Digital Applications" in the 2023 Fifth IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT 2023) held at Velalar College of Engineering & Technology, Erode, Tamil Nadu, India during 22 - 24, February 2023. ICECCT 2023 has been organized jointly by the Department of Biomedical Engineering and Department of Electrical & Electronics Engineering.



Dr. Thangaprakash Sengodan
Conference Chair



Dr. Mangai S
Organizing Chair



Dr. Jayaraman M
Principal



**Velalar College of
Engineering & Technology**
Approved by AICTE, Affiliated to Anna University, Chennai, India
Established in 1984, Accredited by NAAC, A Grade



Advancing Technology
for Humanity



22 - 24, February 2023
Erode | Tamil Nadu | India

**2023 Fifth IEEE International Conference on
Electrical, Computer and Communication
Technologies**

22 - 24, February 2023 | www.icecct.com

EC 3085
Peer Reviewed

CERTIFICATE

This certificate is presented to

Lakshmi.L
Department of Electronics and Communication,
Sathyabama Institute of Science and Technology,
Chennai, India

for presenting the research paper entitled "Analysis of an Efficient Linear Congruential Generator Architecture for Digital Applications" in the 2023 Fifth IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT 2023) held at Velalar College of Engineering & Technology, Erode, Tamil Nadu, India during 22 - 24, February 2023. ICECCT 2023 has been organized jointly by the Department of Biomedical Engineering and Department of Electrical & Electronics Engineering.



Dr. Thangaprakash Sengodan
Conference Chair



Dr. Mangai S
Organizing Chair



Dr. Jayaraman M
Principal



**Velalar College of
Engineering & Technology**
Approved by AICTE, Affiliated to Anna University, Chennai, India
Established in 1984, Accredited by NAAC, A Grade