

An Efficient Dual Stage Compressor for High Speed Digital Applications

Submitted in partial fulfilment of requirements for the award of
Bachelor of Engineering in Electronics and Communication Engineering

by

B HARI PRASAD(39130160)

D JANITH CHARAN(39130179)



DEPERTMENT 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 "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of B HARI PRASAD (39130160) and D JANITH CHARAN (39130170) who carried out the project entitled "AN EFFICIENT DUAL STAGE COMPRESSOR FOR HIGH SPEED DIGITAL APPLICATIONS" under our supervision from Jan 2023 to April 2023.

Internal Guide

Dr. MATHAN N., M.Tech., Ph.D

Assistant Professor

Head of the Department

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

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

We, B HARI PRASAD (39130160) and D JANITH CHARAN (39130179) hereby declare that the Project Report entitled **“AN EFFICIENT DUAL STAGE COMPRESSOR FOR HIGH SPEED DIGITAL APPLICATIONS”** done by us under the guidance of Dr. MATHAN N., M.Tech., Ph.D, Assistant Professor is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Electronics and Communication Engineering.

1. 
2. 

DATE: 15.04.2023

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATES

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.MATHAN N., M.Tech., Ph.D** for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

We 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

The overhead on computation units in a processor to deliver high performance and execution efficiency can be leveraged by introducing approximation. Speed of operation which is inversely proportional to the delay of the system requires immense parallel operations that incur huge hardware and power dissipation. Energy and area efficient systems can be realized by relaxing the precision and reliability of the system. In order to maintain the balance between delay, area and power, approximate computing has emerged as a promising solution. Approximate application is a common operation used in approximate computing methods for high performance and low power computing. These applications deliver high performance at the cost of reduction in accuracy. Furthermore, such implementations reduce the complexity of the system architecture, delay and power consumption. Power efficient circuits for approximate multiplication can be realized with an approximate 4:2 compressor. This paper proposes a novel design that uses a modification of a previous approximate 4:2 compressor design and adds an error recovery module, delay and power with comparable accuracy when compared with the existing architecture. The proposed design, even with the additional error recovery module, is more accurate requires less hardware, and consumes less power than previously proposed 4:2 compressor – based approximate multiplier design. These designs are implemented using 130nm CMOS technology and efficiency of the proposed designs have been extensively verified and projected on scales of area, delay, power, Power Delay Product (PDP). A novel high speed, area – efficient, low power 4:2 compressor architecture is proposed. The proposed compressors are utilized to implement in multipliers. The analysis is further will be extended for high speed applications.

TABLE OF CONTENTS

CHAPTER No.	TITLE	PAGE No.
	ABSTRACT	V
	LIST OF FIGURES	IX
	LIST OF SYMBOLS AND ABBREVIATIONS	XII
1	INTRODUCTION	1
	1.1 INTRODUCTION	1
	1.2 VLSI DESIGN FLOW	3
	1.3 4:2 COMPRESSOR	4
	1.4 APPROXIMATE MULTIPLIERS	6
	1.5 MUX	7
	1.5.1 AND GATE	8
	1.5.2 OR GATE	9
	1.5.3 XOR GATE	9
	1.5.4 XNOR GATE	10
	1.5.5 NOT GATE	11
	1.5.6 NAND GATE	11
	1.5.7 NOR GATE	12
2	LITERATURE SURVEY	13
	2.1 LITERATURE SURVEY	13
	2.2 LITERATURE SUMMARY	24
	2.3 PROBLEM DEFINITION	24
	2.4 OBJECTIVES OF THE PROJECT	24
3	EXISTING SYSTEM	25
	3.1 DUAL STAGE APPROXIMATE 4:2 COMPRESSOR	25

	3.2 HIGH SPEED AREA EFFICIENT 4:2 COMPRESSOR	28
	3.3 MODIFIED DUAL STAGE ARCHITECTURE FOR 4:2 COMPRESSOR	29
4	PROPOSED SYSTEM	32
	4.1 STACK TECHNIQUE	32
	4.2 MODIFIED DUAL STAGE 4:2 COMPRESSOR USING STACK TECHNIQUE	33
	4.3 2*1 MUX USING STACK TECHNIQUE	34
	4.4 OR GATE USING STACK TECHNIQUE	35
	4.5 AND GATE USING STACK TECHNIQUE	35
	4.6 XOR GATE USING STACK TECHNIQUE	36
5	RESULTS AND DISCUSSION	
	5.1 SIMULATION RESULTS OF AREA EFFICIENT 4:2 COMPRESSOR USING FULL STACK 130NM	37
	5.2 SIMULATION RESULTS OF DUAL STAGE 4: 2 COMPRESSOR USING FULL STACK 130NM	38
6	CONCLUSION	41
	REFERENCES	42

LIST OF FIGURES

FIGURE No.	TITLE	PAGE No.
1.1	FLOW CHART OF VLSI DESIGN	4
1.2	4:2 COMPRESSOR	5
1.3	MULTIPLEXER	7
1.4	2:1 MUX USING TRANSMISSION LOGIC GATE	8
1.5	AND GATE	8
1.6	OR GATE	9
1.7	XOR GATE	10
1.8	XNOR GATE	10
1.9	NOT GATE	11
2.0	NAND GATE	11
2.1	NOR GATE	12
2.2	DUAL STAGE 4:2 APPROXIMATE COMPRESSOR	27
2.3	BASIC BUILDING BLOCK OF MODIFIED DUAL STAGE 4:2 APPROXIMATE COMPRESSOR	29
2.4	MODIFIED DUAL STAGE ARCHITECTURE FOR 4:2 COMPRESSOR	31
2.5	STACK DIAGRAM	33
2.6	2*1 MUX USING STACK TECHNIQUE	34
2.7	OR GATE USING STACK TECHNIQUE	35
2.8	AND GATE USING STACK TECHNIQUE	35
2.9	XOR GATE USING STACK TECHNIQUE	36

LIST OF TABLES

TABLE NO	TITLE	PAGE NO.
5.1	Performance Analysis of 130nm Compressor	39
5.2	Performance Analysis of 130nm Modified Compressor using Full-Stack	39
5.3	Performance Analysis of 32nm Compressor	40
5.4	Performance Analysis of 32nm Modified Compressor using Full-Stack	40

LIST OF SYMBOLS AND ABBREVIATIONS

PDP	POWER DELAY PRODUCT
ER	ERROR RATE
ED	ERROR DISTANCE
AOC	ACCURATE OUTPUT COUNT
MED	MEAN ERROR DISTANCE
MRED	MEAN RELATIVE ERROR DISTANCE

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

High speed multimedia applications have paved way for a whole new area in high speed error-tolerant circuits with approximate computing. These applications deliver high performance at the cost of reduction in accuracy. Furthermore, such implementations reduce the complexity of the system architecture, delay and power consumption. This paper explores and proposes the design and analysis of two approximate compressors with reduced area, delay and power with comparable accuracy when compared with the existing architectures. The proposed designs are implemented using 130nm CMOS technology and efficiency of the proposed designs have been extensively verified and projected on scales of area, delay, power, Power Delay Product (PDP), Error Rate (ER), Error Distance (ED), and Accurate Output Count (AOC). The proposed approximate 4 V 2 compressor shows 56.80% reduction in area, 57.20% reduction in power, and 73.30% reduction in delay compared to an accurate 4 V 2 compressor. The proposed compressors are utilised to implement 8x8 and 16x16 Dadda multipliers.

These multipliers have comparable accuracy when compared with state-of-the-art approximate multipliers. The analysis is further extended to project the application of the proposed design in error resilient applications like image smoothing and multiplication. The overhead on computation units in a processor to deliver high performance and execution efficiency can be leveraged by introducing approximation. Speed of operation which is inversely proportional to the delay of the system requires immense parallel operations that incur huge hardware and power dissipation. Energy and area efficient systems can be realised by relaxing the precision and reliability of the system. In order to maintain the balance between delay, area and power, approximate computing has emerged as a promising solution. Approximation in arithmetic operations result in faster systems with lesser design complexity and power consumption. The trade off would be reduction in accuracy, which does not necessarily affect the normal operation for

machine learning and multimedia applications. Such applications effectively take advantage of the The associate editor coordinating the review of this manuscript and approving it for publication was Yizhang Jiang inability of human eye to detect variation inner details within images and videos.

This level of error tolerance is used to design approximate arithmetic circuits for Artificial Intelligence (AI) and Digital Signal Processing (DSP) applications. Extensive research has been carried out to enhance the efficiency of approximate arithmetic units . In multiplication operation, partial product summation has in-arguably been the major contributor to the power dissipation and delay in the system . Research shows that compressors can reduce the delay associated with the partial product summations . Compressors estimate the count of logic 1 in the input using half adders and/or full adders. The various commonly used topologies for compressors are 7 :3, 5 :2, 4 V 2, and 3 V 2 . A 4 V 2 compressor is preferred over any other topology due to the regularity achieved while cascading. Also, it has been widely used to design Dadda multipliers .Due to the considerable reduction in delay using transmission gates when compared to traditional CMOS based logic, optimised design with transmission gates are explored in literature. But, the major disadvantage is the inconsistency

in the rise and fall times for different inputs .

In this paper, two novel 4 V 2 compressor architectures are presented. The contributions of the paper are listed below.

- A novel high speed area-efficient, low power 4 V 2 compressor architecture is proposed.
- Overall error rate is 25% with equal number of C1 and 1 ED. This leadsto reduced Mean Error Distance (MED) and Mean Relative Error Distance (MRED) which is ideal for Multiply Accumulate(MAC) operations.
- Dadda multiplier is implemented with the proposed 4 V 2 compressor.
- A modified dual-stage compressor design is proposed to reduce area, delay and power dissipation in multipliers in which more than two stages of cascaded compressors are required for partial product accumulation.

- Proposed designs were verified with the implementation of multiplication and smoothing of images.

1.2 VLSI Design Flow

Very-large-scale integration (VLSI) is the process of creating an **integrated circuit** (IC) by combining thousands of **transistors** into a single chip. VLSI began in the 1970s when complex **semiconductor** and **communication** technologies were being developed. The **microprocessor** is a VLSI device.

- Before the introduction of VLSI technology, most ICs had a limited set of functions they could perform. An **electronic circuit** might consist of a **CPU, ROM, RAM** and other **glue logic**. VLSI lets IC designers add all of these into one chip.
- The electronics industry has achieved a phenomenal growth over the last few decades, mainly due to the rapid advances in large scale integration technologies and system design applications. With the advent of very large scale integration (VLSI) designs, the number of applications of integrated circuits (ICs) in high-performance computing, controls, telecommunications, image and video processing, and consumer electronics has been rising at a very fast pace.
- The current cutting-edge technologies such as high resolution and low bit-rate video and cellular communications provide the end-users a marvelous amount of applications, processing power and portability. This trend is expected to grow rapidly, with very important implications on VLSI design and systems design.
- The VLSI IC circuits design flow is shown in the figure below. The various levels of design are numbered and the blocks show processes in the design flow.
- Specifications comes first, they describe abstractly, the functionality, interface, and the architecture of the digital IC circuit to be designed.

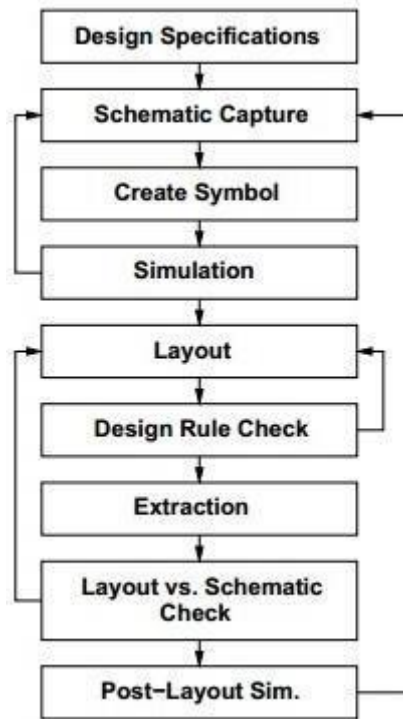


Fig 1.1 Flow chart of Vlsi design

- Behavioral description is then created to analyse the design in terms of functionality, performance, compliance to given standards, and other specifications.
- RTL description is done using HDLs. This RTL description is simulated to test functionality. From here onwards we need the help of EDA tools.
- RTL description is then converted to a gate-level netlist using logic synthesis tools. A gate level netlist is a description of the circuit in terms of gates and connections between them, which are made in such a way that they meet the timing, power and area specifications.

1.3 4:2 COMPRESSOR

A compressor is a device which is mostly used in multiplier to reduce the operands while adding terms of partial products. A typical M-N compressor takes M

equally weighted input bits and produces N-bit binary number. The simplest and the most widely used compressor is the 3-2 compressor which is also known as a full adder. It has Three inputs to be summed up and provides two outputs. Similarly, a 4-2 compressor can also be built from two Cascaded 3-2 compressor circuits. The conventional implementation of a 4-2 compressor is composed of two serially connected full adders. Different structures of 4-2 compressors are reported in literature and these are governing by the basic equation as Follows:

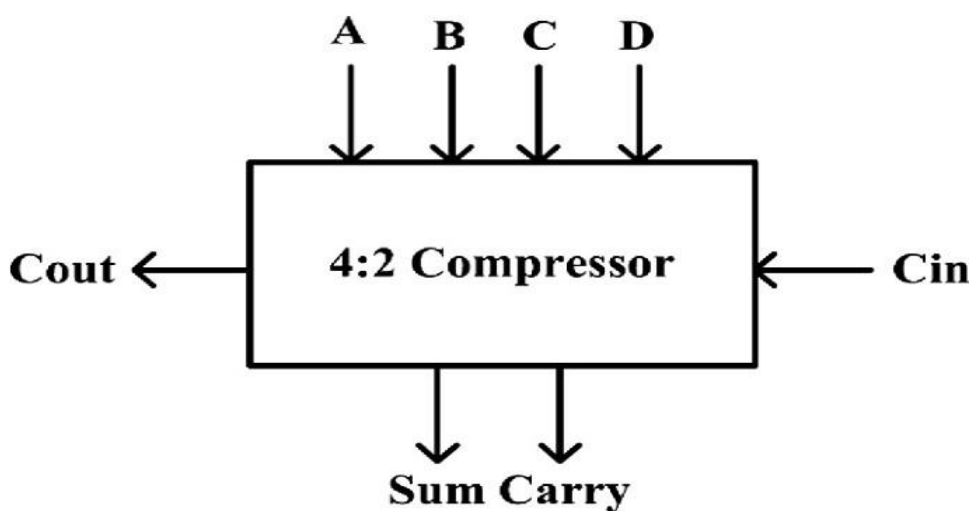


Fig 1.2 4:2 compressor

One of the major speed enhancement techniques used in modern digital circuits is the ability to add numbers with minimal carry propagation. The basic idea is that three numbers can be reduced to 2, in a 3:2 compressor, by doing the addition while keeping the carries and the sum separate. This means that all of the columns can be added in parallel without relying on the result of the previous column, creating a two output "adder" with a time delay that is independent of the size of its inputs. The sum and carry can be recombined in a normal addition to form the correct result. This process may seem more complicated and pointless, but the power of this technique is that any amount, number of additions can be added together in this manner. It is only the final recombination of the final carry and sum that requires a carry propagating addition. 3:2 compressor is also known as full adder. It adds three

one bit binary numbers, a sum and a carry. The full adder is usually a component in a cascade of adders. The carry input for the full adder circuit is from the carry output from the cascade circuit. Carry output from full adder is fed to another full adder. The characteristics of the 4:2 compressors are:

- The outputs represent the sum of the five inputs, so it is really a 5 bit adder.
- Both carries are of equal weighting (i.e. add "1" to the next column)
- To avoid carry propagation, the value of Cout depends only on A, B, C and D. It is independent of Cin.
- The Cout signal forms the input to the Cin of a 4:2 of the next column.

The common implementation of a 4-2 compressor is accomplished by utilizing two full-adder (FA) To add binary numbers cells.4:2 compressor is composed of two serially connected full adders. With minimal carry propagation we use compressor adder instead of other adder. Compressor is a digital modern circuit which is used for high speed with minimum gates requires designing technique. This compressor becomes the essential tool for fast multiplication adding technique on fast processor and lesser area.

1.4 APPROXIMATE MULTIPLIERS

Multiplication is unquestionably a performance determining operation in AI and DSP applications. These applications demand high speed multiplier architectures to necessitate high speed parallel operations with acceptable levels of accuracy. Introduction of approximation in multipliers leads to realisation of faster computations with reduced hardware complexity, delay and power, with accuracy in desirable levels.

Partial product summation is the speed limiting operation in multiplication due to the propagation delay in adder networks. In order to reduce the propagation delay, compressors are introduced. Compressors compute the sum and carry at each level simultaneously. The resultant carry is added with a higher significant sum bit in the next stage. This is continued until the final product is generated.

1.5 MUX

A Multiplexer or mux is a combinational circuits that selects several analog or digital input signals and forwards the selected input into a single output line. A multiplexer of 2^n inputs has n selected lines, are used to select which input line to send to the output.

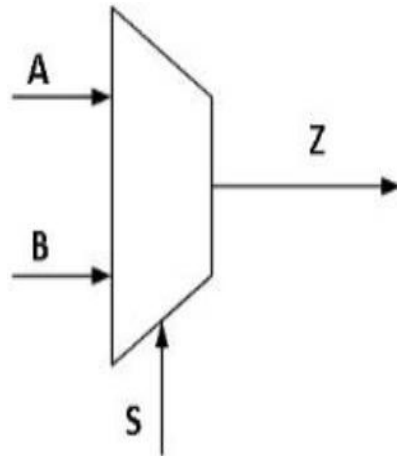


Fig 1.3 Multiplexer

A transmission gate is an electronic element and good non mechanical relay built with CMOS technology. It is made by parallel combination of nMOS and pMOS transistors with the input at the gate of one transistor (C) being complementary to the input at the gate () of the other.

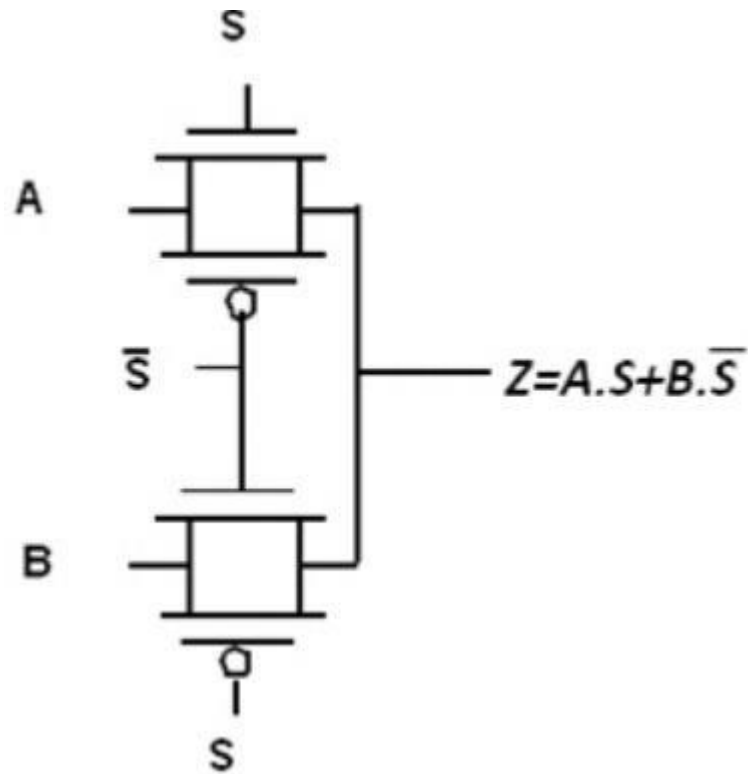


Fig 1.4 2:1 mux using transmission gate logic

1.5 .1 AND GATE

The Logic AND Gate is a type of digital logic circuit whose output goes HIGH to a logic level 1 only when all of its inputs are HIGH



Fig 1.5 And gate

The output state of a digital logic AND gate only returns “LOW” again when **ANY** of its inputs are at a logic level “0”. In other words for a logic AND gate, any LOW input will give a LOW output.

The logic or Boolean expression given for a digital AND gate is that for *Logical Multiplication* which is denoted by a single dot or full stop symbol, (.) giving us the Boolean expression of: $A.B = Q$.

Then we can define the operation of a digital 2-input AND gate as being:

“If both A and B are true, then Q is true”

1.5.2 OR GATE

The Logic OR Gate is a type of digital logic circuit whose output goes HIGH to a logic level 1 only when one or more of its inputs are HIGH

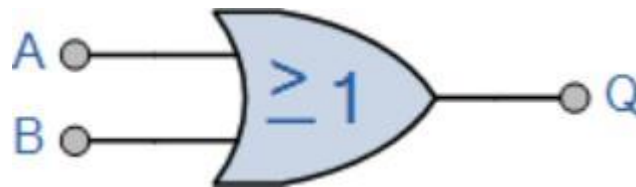


Fig 1.6 Or gate

The output, Q of a “Logic OR Gate” only returns “LOW” again when **ALL** of its inputs are at a logic level “0”. In other words for a logic OR gate, any “HIGH” input will give a “HIGH”, logic level “1” output.

The logic or Boolean expression given for a digital logic OR gate is that for *Logical Addition* which is denoted by a plus sign, (+) giving us the Boolean expression of: $A+B = Q$.

Thus the OR gate can be correctly described as an “Inclusive OR gate” because the output is true when both of its inputs are true (HIGH). Then we can define the operation of a 2-input logic OR gate as being:

“If either A or B is true, then Q is true”

1.5.3 XOR GATE

The Exclusive-OR logic function is a very useful circuit that can be used in many different types of computational circuits

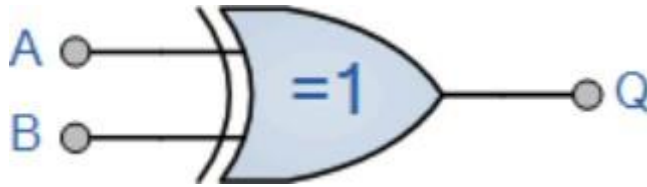


Fig 1.7 Xor gate

The “Exclusive OR Gate” is another type of digital logic gate commonly used in arithmetic operations since it can be used to give the sum of two binary numbers as well as error-detection and correction circuits.

1.5.4 XNOR GATE

The Exclusive-NOR Gate function is a digital logic gate that is the reverse or complementary form of the Exclusive-OR function

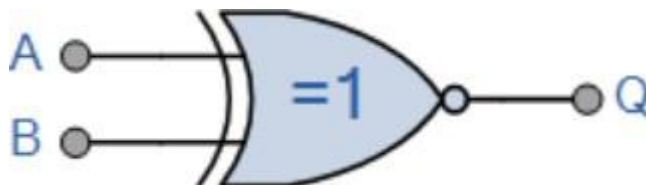


Fig 1.8 Xnor gate

Basically the “Exclusive-NOR Gate” is a combination of the Exclusive-OR gate and the NOT gate but has a truth table similar to the standard NOR gate in that it has an output that is normally at logic level “1” and goes “LOW” to logic level “0” when **ANY** of its inputs are at logic level “1”.

However, an output “1” is only obtained if **BOTH** of its inputs are at the same logic level, either binary “1” or “0”. For example, “00” or “11”. This input

combination would then give us the Boolean expression
of: $Q = (A \oplus B) = A.B + A.B$

1.5.5 NOT GATE

The Logic NOT Gate is the most basic of all the logical gates and is often referred to as an Inverting Buffer or simply an Inverter

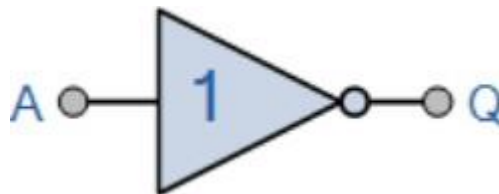


Fig 1.9 Not gate

Inverting NOT gates are single input devices which have an output level that is normally at logic level “1” and goes “LOW” to a logic level “0” when its single input is at logic level “1”, in other words it “inverts” (complements) its input signal. The output from a NOT gate only returns “HIGH” again when its input is at logic level “0” giving us the Boolean expression of: $A = Q$.

1.5.6 NAND GATE

The Logic NAND Gate is a combination of a digital logic AND gate and a NOT gate connected together in series



Fig 2 Nand gate

The NAND (Not – AND) gate has an output that is normally at logic level “1” and only goes “LOW” to logic level “0” when **ALL** of its inputs are at logic level “1”.

The **Logic NAND Gate** is the reverse or “*Complementary*” form of the AND gate we have seen previously.

1.5.7 NOR GATE

The Logic NOR Gate gate is a combination of the digital logic OR gate and an inverter or NOT gate connected together in series

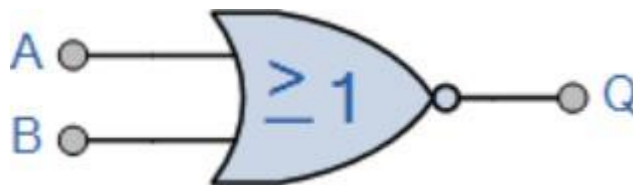


Fig 2.1 nor gate

The inclusive NOR (Not-OR) gate has an output that is normally at logic level “1” and only goes “LOW” to logic level “0” when **ANY** of its inputs are at logic level “1”. The **Logic NOR Gate** is the reverse or “*Complementary*” form of the inclusive OR gate.

CHAPTER 2

LITERATURE SURVEY

This chapter discusses about extensive literature survey on dual stage compressor for high speed digital applications .

2.1 LITERATURE SURVEY

PRANOSE J.EDVOOR (2020) et.al Determines that high-speed multimedia applications have paved way for a whole new area in high-speed error tolerant circuits with approximate computing. These applications deliver high performance at the cost of a reduction in accuracy. Furthermore, such implementations reduce the complexity of the system architecture delay, and power consumption. This paper explores and proposes the analysis of two approximate computing. These applications deliver high performance at the cost of reduction in accuracy. To balance the between delay ,area and power ,approximate computing has implemented Furthermore, such implementations reduce the complexity of the system architecture, delay and power consumption. This paper develops the design and analysis of two approximate compressors with less area and low power consumption with accurate compared to the exists architectures. The proposed designs are implemented using 45 nm CMOS technology and efficiency of the proposed designs have been extensively verified and projected on scales of area, delay, power, Power Delay Product (PDP), Error Rate (ER), Error Distance (ED), and Accurate Output Count (AOC). The proposed approximate 4:2 compressors reduced the area into 56.80% and reduced power into 57.20% and reduced delay into 73.30% compared to an accurate 4 : 2 compressor. The proposed compressors are utilized to implement 8×8 and 16×16 Dadda multipliers. These multipliers have comparable accuracy when compared with state-of-the-art existing approximate multipliers. The analysis is further extended to project the applications of multiplication. The proposed design is in error correction applications like image smoothing.

Omid Akbari (2017) et.al determines the four 4:2 compressors, which have the flexibility of switching between the exact and approximate operating modes. These dual-quality compressors provide higher speed applications in the approximate mode and low power consumption at the lower accuracy in the

approximate mode. These compressors have their level of accuracy in the approximate mode and different delays. These compressors have power dissipations in the approximate mode and exactly in the other mode. Using these compressors in the structures of parallel multipliers provides configurable multipliers whose accuracies (as well as their powers and speeds) may change dynamically during the runtime. The efficiencies of these compressors in a 32-bit Dadda multiplier are evaluated in a 45-nm standard using CMOS technology by comparing their parameters with those of the state-of-the-art approximate multipliers. The results of the comparison indicate 46% of average lower delay and 68% of power consumption in the approximate mode. Also, the effectiveness of these compressors is estimated in some image processing applications. the proposed compressors in 32-bit Dadda multiplier provided, on average, about 33% lower NED compared with the state-of-the-art compressor-based approximate multipliers. When comparing with non compressor-based approximate multipliers, the errors of the proposed multipliers were higher while the design parameters were considerably better.

Swaroop Ghosh (2010) et.al. determines various arithmetic units for possible use in high-speed, high-yield ALUs operated at scaled supply voltage with adaptive clock stretching. We demonstrate that careful logic optimization of the existing arithmetic units (to create hybrid units) indeed makes them further flexible to supply voltage scaling. Such hybrid units result from mixing the right amount of fast arithmetic into the slower ones. Simulations on different *hybrid* adders and multipliers in BPTM 70 nm technology show 18%–50% improvements in power compared to standard adders with only a 2%–8% increase in die-area at iso-yield. These optimized data path units can be used to construct voltage scalable robust ALUs that can operate at high clock frequency with minimal performance degradation due to occasional clock stretching. Variable latency functional units using adaptive clock stretching can allow aggressive scaling of supply voltage while maintaining rated frequency with little performance degradation and also proposed hybrid adder design (by mixing a small amount of fast arithmetic into the slower ones) that can be utilized for improving yield or scaling the supply voltage further.

Dursun Baran et.al. (2011) determine the suitability of energy-efficient serial and parallel multiplier structures for low and ultra-low power designs is evaluated. 16×16-bit serial and state-of-art parallel multipliers are compared in 45nm CMOS. A multiplier structure is proposed by optimizing the architecture, gate sizes, and voltage supply. The proposed structure provides 15% more throughput as compared to a two-cycle parallel multiplier with the same energy consumption for high-speed applications. In the low-speed design region, it provides 3.7X energy reduction compared to the serial multiplier. The parallel multipliers will be more energy-efficient than the serial multiplier in low throughput design regimes by optimizing voltage supply and gate sizes. A two-cycle Non-Booth parallel multiplier that is obtained by balancing the complexities of pipeline stages is proposed. The proposed design outperforms the parallel structures and it also provides up to 3.7X energy reduction for ultra-low power applications compared to low-cost serial multipliers.

Sparsht Mittal et.al (2016) determines Approximate computing trades off computation quality with the effort expended and as rising performance demands confront plateauing resource budgets, approximate computing has become, not merely attractive, but even imperative. In this paper, we present a survey of techniques for approximate computing(AC). We discuss strategies for finding approximable program portions and monitoring output quality, techniques for using AC in different processing units (e.g., CPU, GPU, and FPGA), processor components, memory technologies, etc., and programming frameworks for AC. We classify these techniques based on several key characteristics to emphasize their similarities and differences. The aim of this paper is to provide insights to researchers into the working of AC techniques and inspire more efforts in this area to make AC the mainstream computing approach in future systems. Using approximation in a single processor component alone can have an unforeseen effect on the operation of other components and is likely to lead to erroneous or myopic conclusions.

HONGLAN JIANG et.al.(2017) determine the most important arithmetic modules in a processor, adders, multipliers, and dividers determine

the performance and the energy efficiency of many computing tasks. The demand for higher speed and power efficiency, as well as the feature of error resilience in many applications (e.g., multimedia, recognition, and data analytics), have driven the development of an approximate arithmetic design. In this article, a review and classification are presented for the current designs of approximate arithmetic circuits including adders, multipliers and dividers. A comprehensive and comparative evaluation of their error and circuit characteristics is performed for understanding the features of various designs. By using approximate multipliers and adders, the circuit for an image processing application consumes as little as 47% of the power and 36% of the power-delay product of an accurate design while achieving a similar image processing quality. Improvements in delay, power, and area are obtained for the detection of differences in images by using approximate dividers.

Jinghang Liang et.al. (2013) determine that Addition is a fundamental function in arithmetic operation several adder designs have been proposed for implementations in exact computing. These adders show different operational profiles some of them are approximate in nature while others rely on probabilistic features of nanoscale circuits. However, there has been a lack of appropriate metrics to evaluate the efficacy of various inexact designs. In this paper, new metrics are proposed for evaluating the reliability as well as the power efficiency of approximate and probabilistic adders. Reliability is analyzed using the so-called sequential probability transition matrices (SPTMs). Error distance (ED) is initially defined as the arithmetic distance between an erroneous output and the correct output for a given input. The mean error distance (MED) and normalized error distance (NED) are then proposed as unified figures that consider the averaging effect of multiple inputs and the normalization of multiple-bit adders. It is shown that the MED is an effective metric for measuring the implementation accuracy of a multiple-bit adder and that the NED is a nearly invariant metric independent of the size of an adder. The MED is, therefore, useful in assessing the effectiveness of an approximate or probabilistic adder implementation, while the NED is useful in characterizing the reliability of a specific design. Since inexact adders are often used for saving power, the product of power and NED are further utilized for evaluating the tradeoffs between power consumption and precision. Although illustrated using adders, the proposed metrics are potentially useful in assessing other arithmetic

circuit designs for applications of inexact computing. In most digital systems, sequential circuits are utilized, so this paper primarily deals with sequential adders.

Reza Zendegani et.al. (2017) determines an approximate multiplier that is high speed yet energy efficient. The approach is to round the operands to the nearest exponent of two. This way the computationally intensive part of the multiplication is omitted improving speed and energy consumption at the price of a small error. Both signed and unsigned multiplications are applicable to the approach in the approximate multiplier. There are three hardware implementations of the approximate multiplier. Includes one for the unsigned operations and another two for the signed operations. The efficiency of the proposed multiplier is evaluated by comparing its performance with those of some approximate and accurate multipliers using different design parameters. In addition, the efficacy of the proposed approximate multiplier is studied in two image processing applications, i.e., image sharpening and smoothing.

Hoglan Jiang et.al.(2016) determine The Booth multiplier has been widely used for high-performance signed multiplication by encoding and thereby reducing the number of partial products. The radix-8 algorithm is very efficient due to the ease of partial product generation in the multiplier. whereas the radix-8 Booth multiplier is slow due to the complexity of generating the odd multiples of the multiplicand. In this paper, this issue is alleviated by the application of approximate designs. An approximate 2-bit adder is carefully designed for calculating the sum of 1 and 2 of a binary number. This adder requires a small area, a low power, and a short critical path delay. Subsequently, the 2-bit adder is employed to implement the less significant section of a recording adder for generating the triple multiplicand with no carry propagation. In the pursuit of a trade-off between accuracy and power consumption, two signed 16X16 bit approximate radix-8 Booth multipliers are designed using the approximate recoding adder with and without the truncation of a number of less significant bits in the partial products. The proposed approximate multipliers are faster and more power efficient than the accurate Booth multiplier; moreover, the multiplier with 15-bit truncation achieves the best overall performance in terms of hardware and accuracy when compared to other approximate Booth multiplier designs. Finally, the approximate multipliers are applied to the design of a

low-pass FIR filter and they show better performance than other approximate Booth multipliers.

Jie Han et al. (2015) determines that approximate computing is an attractive paradigm for digital processing at nanometric scales. Inexact computing is particularly interesting for computer arithmetic designs. This paper deals with the analysis and design of two new approximate 4-2 compressors for utilization in a multiplier. These designs rely on different features of compression, such that imprecision in computation can meet with respect to circuit-based figures of merit of a design (number of transistors, delay, and power consumption). Four different schemes for utilizing the proposed approximate compressors are proposed and analyzed for a Dadda multiplier. Extensive simulation results are provided and an application of the approximate multipliers to image processing is presented. The results show that the proposed designs accomplish significant reductions in power dissipation, delay, and transistor count compared to an exact design; moreover, two of the proposed multiplier designs provide excellent capabilities for image multiplication with respect to average normalized error distance and peak signal-to-noise ratio (more than 50 dB for the considered image examples).

Minho Ha and Sunggu Lee (2018) determine Approximate multiplication is a common operation used in approximate computing methods for high performance and low power computing. Power-efficient circuits for approximate multiplication can be realized with an approximate 4-2 compressor. This letter presents a novel design that uses a modification of a previous approximate 4-2 compressor design and adds an error recovery module. The proposed design, even with the additional error recovery module, is more accurate, requires less hardware, and consumes less power than previously proposed 4-2 compressor-based approximate multiplier designs. The most important operations in a digital signal processor are addition and multiplication. Focusing on the latter, this letter has proposed a novel 4-2 compressor design that can be used in the partial product compression phase of a binary multiplier.

Yi Guo et.al (2018) determine that Approximate computing is applicable to improve hardware performance by sacrificing some accuracy for error-tolerant applications, where multiplication is a key arithmetic operation. In this paper, we propose a low-cost approximate multiplier design by employing a new probability-driven inexact 4:2, 6:2, 8:2 compressors and inexact half-adders. This compressor design is explored to reduce the height of the partial product matrix into two rows. Different levels of accuracy can be achieved through a grouped error recovery scheme that employs different numbers of error compensation vectors for error reduction. The mean relative error distance (MRED) of the proposed multiplier design is from 1.07% to 7.86%. Compared with the Wallace multiplier using SMIC 40nm process, the most accurate variant of the proposed design reduces power by 50.52%, area by 52.46%, and delay by 33.90%. The proposed multiplier design has a better accuracy-performance trade-off than other designs. Moreover, the efficiency of approximate multipliers is assessed in an image processing application.

Ihsen Alouani et.al (2018) determine Approximate computing is a design paradigm considered for a range of applications that can tolerate some loss of accuracy. In fact, the bottleneck in conventional digital design techniques can be eliminated to achieve higher performance and energy efficiency by compromising accuracy. In this letter, a new architecture that engages accuracy as a design parameter is presented, where an approximate parallel multiplier using heterogeneous blocks is implemented. Based on design space exploration, we demonstrate that introducing diverse building blocks to implement the multiplier rather than cloning one building block achieves higher precision results. We show experimental results in terms of precision, delay, and power dissipation as metrics and compare them with the three previous approximate designs. Our results show that the proposed heterogeneous multiplier achieves more precise outputs than the tested circuits while improving performance and power trade-offs. After design space explorations, we realized that introducing different elementary architectures to implement the circuit leads to lower ERs compared to the classical homogeneous designs. In fact, the proposed design benefits from the masking mechanisms within logic elements in different cases to limit the overall deviation from the exact results.

Yen-jen chang et.al.(2019) determine that approximate computing can be used in error-tolerant applications since it can provide meaningful results with lower power consumption. In this study, the authors proposed a novel imprecise 4-2 compressor which is used in the multipliers of image processing applications. Besides the output values, they also consider the pattern distribution to synthesize the 4-2 a compressor in imprecise style. Compared to the precise compressor, the proposed imprecise 4-2 compressor can reduce power consumption and delay by 56 and 39%. Compared to the precise multiplier, the simulation results show that the multiplier which uses the proposed imprecise 4-2 compressor can achieve 33 and 30% improvement in power consumption and delay, respectively. In addition, the image quality of their design is good for human perception because peak signal-to-noise ratio PSNR values are more than 33, and mean structural similarity values are more than 0.99. Even compared to the related imprecise works, their design has a better error rate to improve the quality of images while maintaining both high power efficiency and low circuit complexity. Compared to the precise multiplier, the multiplier which uses the proposed imprecise design can save 33% power consumption and achieve a 30% improvement in delay. Multiplier M3 which adopts the proposed imprecise 4-2the compressor can achieve PSNR and MSSIM above 33 and 0.99 on average, respectively. In addition, multiplier M3 has the smallest $vPDP$ -MRED product that shows the proposed design can improve the accuracy and maintain both the high power efficiency and low circuit complexity.

Karri Manikantta Reddy et.al (2019) determines that approximate computing has received significant attention as an attractive paradigm for error-tolerant applications to reduce power consumption, delay, and area with some trade-off in accuracy. This paper proposes the design of a novel approximate 4–2 compressor. A modified architecture of the Dadda Multiplier is presented for the effective utilization of the proposed compressor and to reduce the error at the output. Through extensive experimental evaluation, the efficiency of the proposed compressor and multiplier are evaluated in a 45 nm standard CMOS technology and their parameters are compared with those of the state-of-the-art approximate multipliers. The results show that the proposed compressor accomplishes a significant reduction in error rate compared to other approximate compressors available in the literature. In

addition, the proposed multiplier shows a 35%, 36%, and 17% reduction in power consumption, delay, and area respectively compared to those of the exact multiplier. The effectiveness of the multiplier is assessed by some of the image processing applications. On average, the proposed multiplier processes images with 85% structural similarity compared to the exact output image. The performance of this approximate compressor for the aforesaid accuracy metrics is assessed using a modified Dadda multiplier.

Aloke Saha et.al.(2018) determine a novel multi-bit counter for efficient binary multiplication. A 7:3 counter is proposed, customized, and optimized by a 3-pronged approach, firstly by group-wise parallel addition using half-adders, secondly by eliminating redundant carry-generators from the design, and finally, by optimizing the resultant hardware. The circuit is designed and optimized on standard static-CMOS. Corner analyses with TT, FF, and SS for PVT variation have been performed on the proposed design to study reliability and robustness. A benchmarking exercise on Power-Delay-Product (PDP) with reported candidate designs demonstrates the superiority of the proposed design. The study reveals that the 7:3-counter, designed using the proposed strategy, has up to 36% less PDP, compared to the best candidate design. Next, the proposed counters are employed, first to design an 8-b 8-b Column-Compression (CC) multiplier and thereafter, using decomposition-logic on thus-designed-8-b 8-b-multipliers, to design a 16-b 16-b multiplier. The multipliers are optimized on 90-nm standard-CMOS technology and compared for speed-power performance with reported candidate designs at 500 MHz. Simulations show that the presented design of 16-b 16-b multiplier using the proposed 7:3 counter offers 55% less PDP, compared to the best candidate design under identical conditions. Once again, all simulations are performed on TSMC 90-nm CMOS technology at 25 C temperature and 1.0 V supply rail.

Darjn Esposito et.al.(2018) determine approximate computing is an emerging trend in digital design that trades off the requirement of exact computation for improved speed and power performance. This paper proposes novel approximate compressors and an algorithm to exploit them for the design of efficient approximate multipliers. By using the proposed approach, we have synthesized approximate multipliers for several operand lengths using a 40-nm library. Comparison with

previously presented approximated multipliers shows that the proposed circuits provide better power or speed for a target precision. For each operand size, we developed four approximate multiplier versions, with different precision vs electrical performance trade-offs. The circuits developed in this paper and previously proposed approximated multipliers have been synthesized by using a 40nm CMOS technology. Syntheses show that our circuits provide a very good error-electrical performance trade-off, compared with previously proposed approximate multipliers. We have also investigated the use of approximate multipliers in two applications: image filtering and LMS system identification. In both cases, proposed approximate multipliers show remarkably good results, compared with the state of the art.

Soheil Hashemi et.al.(2015) determines that the applications for signal processing, computer vision and machine learning show an inherent tolerance to some computational error. This error resilience can be exploited to trade off accuracy for savings in power consumption and design area. Since multiplication is an essential arithmetic operation for these applications, in this paper we focus specifically on this operation and propose a novel approximate multiplier with a dynamic range selection scheme. We design the multiplier to have an unbiased error distribution, which leads to lower computational errors in real applications because errors cancel each other out, rather than accumulate, as the multiplier is used repeatedly for a computation. Our approximate multiplier design is also scalable, enabling designers to parameterize it depending on their accuracy and power targets. Furthermore, our multiplier benefits from a reduction in propagation delay, which enables its use on the critical path. We theoretically analyze the error of our design as a function of its parameters and evaluate its performance for a number of applications in image processing, and machine classification introducing bounded errors with a Gaussian distribution with near-zero average and standard deviations of 0.45% – 3.61%. We also report power savings of up to 58% when using the proposed design in applications. We show that our design significantly outperforms other approximate multipliers recently proposed in the literature.

Srinivasan Narayanamoorthy et.al.(2015) determines various digital signal processing (DSP) and classification applications on energy-constrained devices have steadily grown. Such applications often extensively perform matrix

multiplication using fixed-point arithmetic while exhibiting tolerance for some computational errors. Hence, improving the energy efficiency of multiplications is critical. In this brief, we propose multiplier architectures that can trade off computational accuracy with energy consumption at design time. Compared with a precise multiplier, the proposed multiplier can consume 58% less energy/op with an average computational error of $\sim 1\%$. Finally, we demonstrate that such a small computational error does not ably impact the quality of DSP and the accuracy of classification applications.

The approximate multiplier that can trade off accuracy and energy/op at design time for DSP and recognition applications. Our proposed approximate multiplier takes m consecutive bits (i.e., an m -bit segment) of an n -bit operand either starting from the MSB or ending at the LSB and applying two segments that include the leading ones from two operands (i.e., SSM) to an $m \times m$ multiplier. Compared with an approach that identifies the exact leading one positions of two operands and applies two m -bit segments starting from the leading one positions (i.e., DSM), ours consumes much less energy and area than PM and DSM. This improved energy and area efficiency comes at the cost of slightly lower compute accuracy than PM and DSM. However, we demonstrate that the loss of small computes accuracy using SSM does not notably impact QoC of image, audio, and recognition applications we evaluated. On average, 16×16 ESSM8 $\times$ 8 can achieve 99% computational accuracy, respectively, with negligible degradation in QoC for audio, image, and recognition applications.

Minho Ha et.al.(2018) determines that approximate multiplication is a common operation used in approximate computing methods for high performance and low power computing. Power-efficient circuits for approximate multiplication can be realized with an approximate 4-2 compressor. This letter presents a novel design that uses a modification of a previous approximate 4-2 compressor design and adds an error recovery module. The proposed design, even with the additional error recovery module, is more accurate, requires less hardware, and consumes less power than previously proposed 4-2 compressor-based approximate multiplier designs. The most important operations in a digital signal processor are addition and multiplication. Focusing on the latter, this letter has proposed a novel 4-2 compressor design that can be used in the partial product compression phase of a

binary multiplier. The proposed design uses a small modification of an existing 4-2 compressor design that changes the error profile such that all errors result in slightly decreased values (over the exact values). The resulting error profile is such that all such error points can be easily located with a simple 2-input AND. A simple error recovery module can then be used to recover from such errors when desired. An approximate multiplier can be designed with accurate and approximate regions.

2.2 LITERATURE SUMMARY

The common implementation of a 4-2 compressor is accomplished by utilizing two full-adder (FA) To add binary numbers cells.4:2 compressor is composed of two serially connected full adders. With minimal carry propagation we use compressor adder instead of other adder. Compressor is a digital modern circuit which is used for high speed with minimum gates requires designing technique. This compressor becomes the essential tool for fast multiplication adding technique on fast processor and lesser area.

2.3 PROBLEM DEFINITION

To analyze an efficient dual stage compressor for high speed digital applications

2.4 OBJECTIVES OF THE PROJECT

- To increase the efficiency of the Dual Stage Compressor for Low Power applications
- In order to improve Efficiency Low Power Techniques will be applied in all Circuits
- To compare, the characteristics of MOSFET Technology in terms of Power, Delay, and Power Delay Product
- To identify the Feasibility of Dual Stage Compressor

CHAPTER 3

EXISTING SYSTEM

3.1 DUAL-STAGE APPROXIMATE 4 :2 COMPRESSOR

As a measure to optimise the hardware utilisation of the proposed design, this paper proposes an alternate architecture for multipliers with more than three stages of cascaded compressors. In the high speed area-efficient compressor architecture apart from the MUX, one *XOR*, one *AND* and two *OR* gates are required. *OR* and *AND* gates each need 6 transistors in CMOS logic implementation. In order to reduce the transistor count, this paper proposes an architecture with *NAND* and *NOR* gates as shown in Figure. Even though the *SUM* and *CARRY* generated by the modified architecture is not as same as that of the proposed 4: 2 compressor architecture, with cascading of the compressor in multiples of 2, the error is nullified. This is explained with the help of Figure. The given figure has a two level cascading of proposed high speed area-efficient 4 : 2 compressors.

A compressor is a device which is mostly used in multipliers to reduce the operands while adding terms of partial products. A typical M-N compressor takes M equally weighted input bits and produces N-bit binary number. The simplest and the most widely used compressor is the 3-2 compressor which is also known as a full adder. It has Three inputs to be summed up and provides two outputs. Similarly, a 4-2 compressor can also be built from two Cascaded 3-2 compressor circuits. The conventional implementation of a 4-2 compressor is composed of two serially connected full adders.

One of the major speed enhancement techniques used in modern digital circuits is the ability to add numbers with minimal carry propagation. The basic idea is that three numbers can be reduced to 2, in a 3:2 compressor, by doing the addition while keeping the carries and the sum separate. This means that all of the columns can be added in parallel without relying on the result of the previous column, creating a two output "adder" with a time delay that is independent of the size of its inputs. The sum and carry can be recombined in a normal addition

to form the correct result. This process may seem more complicated and pointless, but the power of this technique is that any amount, number of additions can be added together in this manner. It is only the final recombination of the final carry and sum that requires a carry propagating addition. 3:2 compressor is also known as full adder. It adds three one bit binary numbers, a sum and a carry. The full adder is usually a component in a cascade of adders. The carry input for the full adder circuit is from the carry output from the cascade circuit. Carry output from full adder is fed to another full adder. The characteristics of the 4:2 compressors are:

- The outputs represent the sum of the five inputs, so it is really a 5 bit adder.
- Both carries are of equal weighting (i.e. add "1" to the next column)
- To avoid carry propagation, the value of Cout depends only on A, B, C and D. It is independent of Cin.
- The Cout signal forms the input to the Cin of a 4:2 of the next column.

A 4:2 compressors are capable of adding 4 bits and one carry, in turn producing a 3 bit output. The 4-2 compressor has 4 inputs X1, X2, X3 and X4 and 2 outputs Sum and Carry along with a Carry-in (Cin) and a Carry-out (Cout). The input Cin is the output from the previous lower significant compressor. The Cout is the output to the compressor in the next significant stage. The critical path is smaller in comparison with an equivalent circuit to add 5 bits using full adders. However, like in the case of 3-2 compressor, the fact that both the output and its complement are available at every stage is neglected. Thus replacing some XOR blocks with multiplexer's results in a significant improvement in delay as . Also the MUX block at the SUM output gets the select bit before the inputs arrive and this minimizes the delay to a considerable extent.

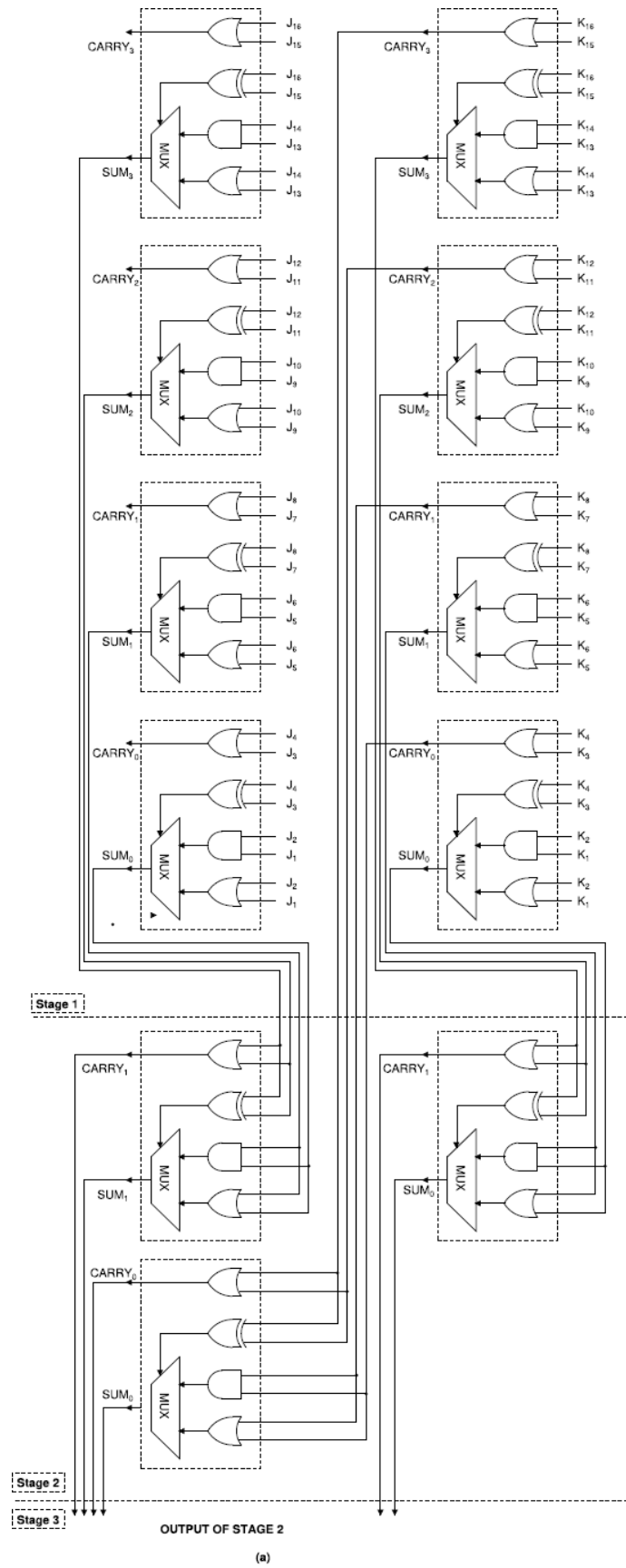


Fig 2.2 Dual stage approximate 4:2 compressor

The proposed compressors are implemented using 45-nm CMOS technology, at an operating frequency of 1 GHz and supply voltage of 1 V. Implementation efficiency metrics analyses area, power, delay, and PDP. Area refers to how well the proposed design optimises the hardware, which makes the design compact. Delay refers to the time taken by a design to perform its intended operation and it determines the maximum speed at which the circuit can operate. An ideal design must have optimised parameters like area, delay and power for a comparable range of accuracy.

3.2 HIGH SPEED AREA-EFFICIENT APPROXIMATE 4 : 2 COMPRESSOR

The proposed high speed area-efficient 4 V 2 approximate compressor is shown in Figure 6. The compressor inputs are A_1 , A_2 , A_3 and A_4 , outputs are *CARRY* and *SUM*. A multiplexer (MUX) based design approach is used to generate *SUM*.

Output of *XOR* gate acts as the select line for the *MUX*. When select line goes high, (A_3A_4) is selected and when it goes low, ($A_3 \oplus A_4$) is selected. By introducing an error with error distance 1 in the truth table of the exact compressor, the proposed 4 V 2 compressor is able to reduce carry generation logic to an *OR* gate. The logical expressions for realisation of *SUM* and *CARRY* are given below.

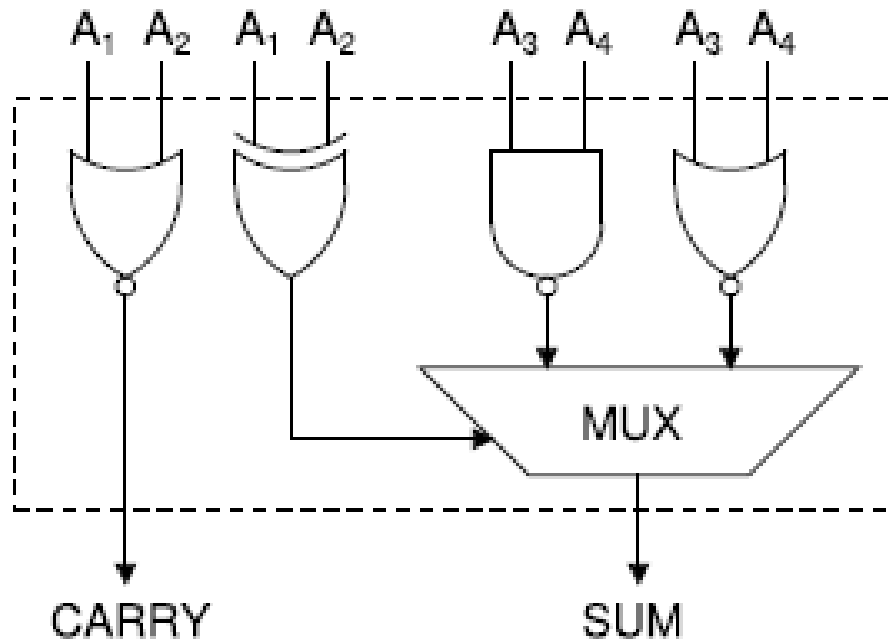


Fig 2.3 Basic building block for proposed modified dual stage approximate 4:2 compressor

Proposed 4 :2 compressor , it can be observed that the error has been introduced for the input values {0011}, {0100}, {1000} and {1111}, so as to ensure that equal positive and negative deviation with $ED D 1$ (minimum) is obtained.

3.3 Modified Dual Stage architecture for 4:2 compressor

As a measure to optimise the hardware utilisation of the proposed design, this paper proposes an alternate architecture for multipliers with more than three stages of cascaded compressors. In the high speed area-efficient compressor architecture , apart from the MUX, one *XOR*, one *AND* and two *OR* gates are required. *OR* and *AND* gates each need 6 transistors in CMOS logic implementation. In order to reduce the transistor count, this paper proposes an architecture with *NAND* and *NOR* gates as shown in Figure . Even though the *SUM* and *CARRY* generated by the modified architecture is not as same as that of the proposed 4 : 2 compressor architecture, with cascading of the compressor in multiples of 2, the error is nullified. This is explained with the help of Figure has a

two level cascading of proposed high speed area-efficient 4 : 2 compressors. two level cascading of modified dual-stage 4 : 2 compressors. The outputs at the Stage 1 differ for both the architectures, but the occurrence of negation in the order of an integral multiple of two (in Stage 1 and Stage 2) in the modified dual-stage 4 : 2 compressor will ensure that the outputs at Stage 2 are same.

The modified dual-stage 4 : 2 compressor reduces area, delay and power dissipation compared to the proposed high speed area-efficient 4 : 2 compressor and other compressors in the literature due to the reduction in transistor count. Analyses the output of the two proposed architectures at different stages in a 2 stage cascaded structure. Modified dual-stage 4 :2 compressor has the same output in cascaded architectures with 18.23% and 14.84% reduction in area and power respectively when compared to proposed high speed area efficient 4 :2 compressor. The accuracy remains unaffected with the modifications introduced.

The overall performance of the multiplier mainly depends on the optimization of the second module. The 8 x8 unsigned Dadda multiplier is utilized to assess the impact of using the proposed approximate compressors in the design of multipliers. The architecture of the Dadda multiplier designed using exact 4:2 compressors is presented in . To design an approximate multiplier, since the proposed approximate compressor does not have C in and C out, the reduction circuitry is modified . Each dot in the figure represents a partial product obtained from AND gates.

The main objective of replacing all the exact compressors with approximate designs is to reduce the delay and power consumption significantly. If the approximate compressors are used only at the least significant columns (rightmost columns in then the performance of approximate multipliers improves in terms of accuracy. The proposed approximate compressor along with the optimized exact compressor and other existing approximate compressors in literature are simulated. For an effective comparison, only 4:2 compressor designs are considered and all these designs are implemented using same CMOS technology

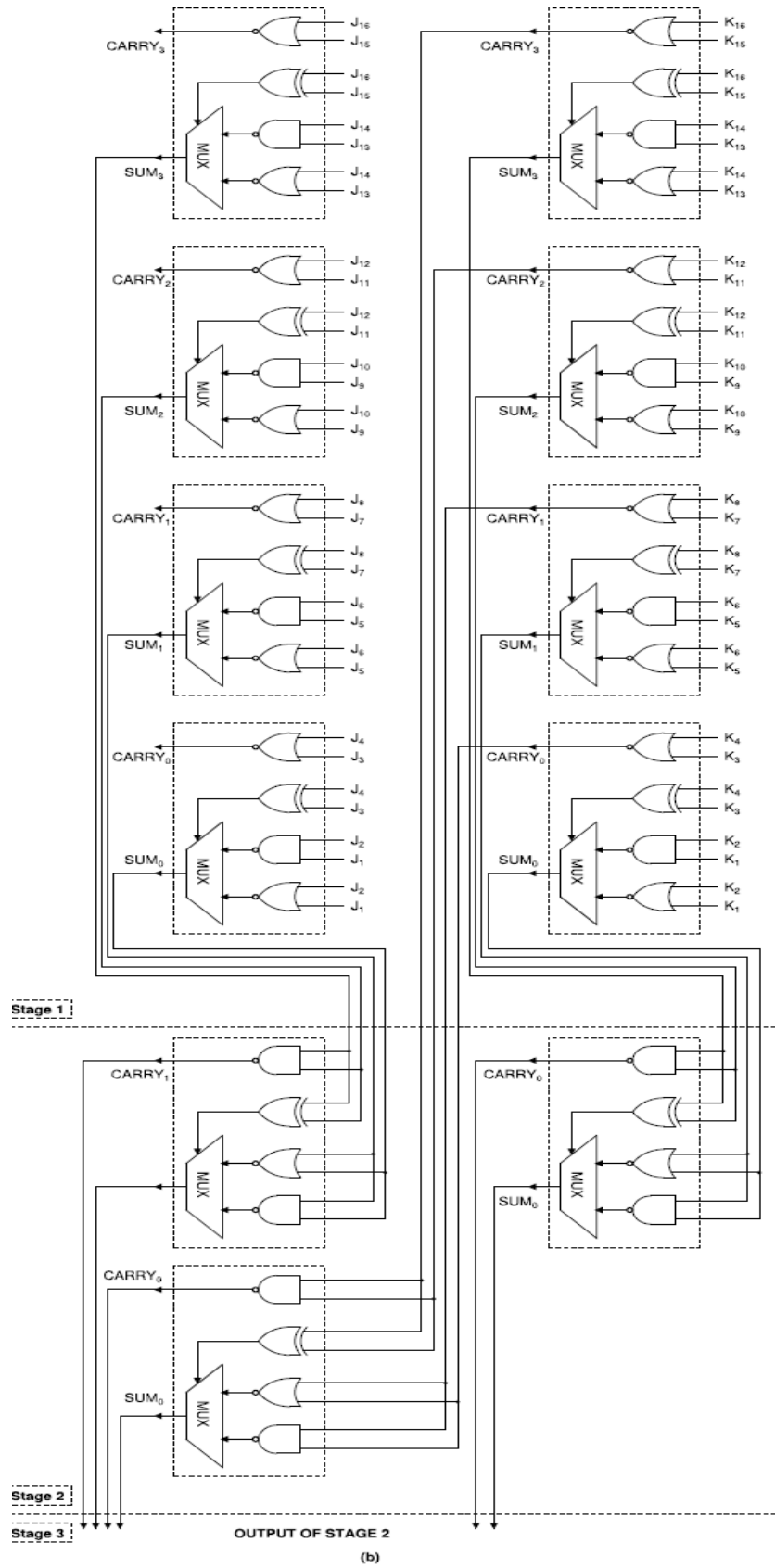


Fig 2.4 : Modified dual stage architecture for 4:2 compressor

CHAPTER 4

PROPOSED SYSTEM

In order to improve efficiency, existing system as dual stage compressor will be analyzed using 130nm, 32nm CMOS technology. Further, dual stage compressor will be modified and analyzed using 130nm, 32nm CMOS Technology in terms of power, speed and power delay product. This outperforms an efficient dual stage compressor for high speed and low power applications.

In existing work it is designed using 4:2 compressors architecture. In the proposed system the 4:2 compressor is designed by implementing forced stack technique which give low power and area compared to existing system. By using stack method, it trade-offs to reduce to low power and it reduces the leakage of current when flows from one transistor to other transistor.

4.1 STACK TECHNIQUE

In forced stack technique, which forces a stack affect by breaking down an existing transistor into two half size transistors. Subthreshold leakage is exponentially related to the threshold voltage of the device, and the threshold voltage changes due to body effect. From these tow facts, one can reduce the subthreshold leakage in the device by stacking two or more serially.

Forced stack is also known as Back Biasing to Reduce Sub Threshold Leakage. Leakage is well known to be a major issue in nanometre VLSI circuits, as it has become a major portion of the overall power budget. The problem is particularly critical in circuits and cells that intrinsically have a considerably high leakage under a given transistor size (e.g., inverters, since they cannot benefit from stack effect). Leakage is exacerbated in circuits driving heavily loaded nodes, as it occurs for buffers, which are usually oversized to achieve a targeted delay. This work is focused on circuits where stack forcing is employed to reduce leakage, as in inverters and heavily loaded tapered buffers. In low-standby current applications, forced stacks are widely adopted to reduce the leakage of each inverter (and hence of the overall buffer). In this approach, a single transistor is replaced by two stacked transistors with halved aspect ratio.

Stack forcing tends to be less effective in downscaled technologies, hence low-standby power applications typically require the adoption of additional techniques to further reduce leakage. This happens due to the reduction of gate to source voltage and reduction of DIBL coefficient due to lower drain to source potential thereby further reducing leakage.

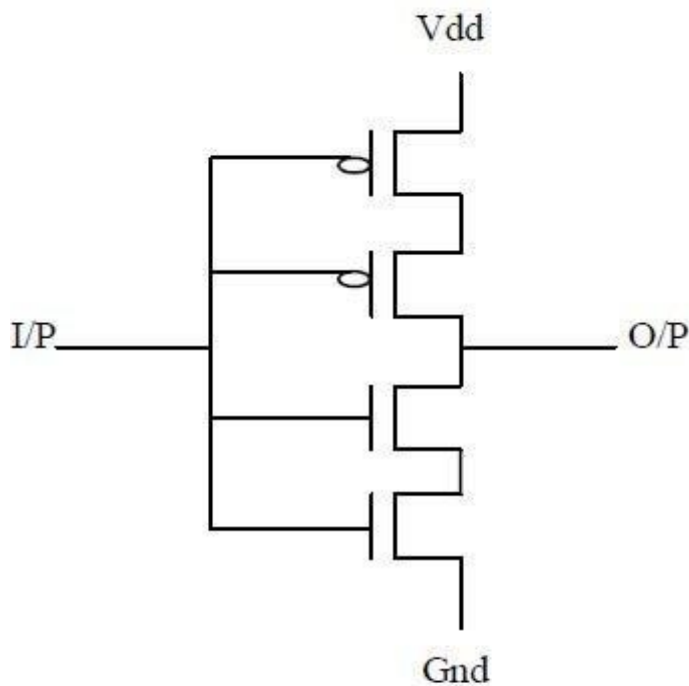


Fig 2.5 :Stack Diagram

Leakage power consumption is an important issue in CMOS VLSI circuits. The main contribution of Power Dissipation in CMOS circuit increases with the reduction of channel length, Threshold voltage and gate oxide thickness.

Subthreshold leakage current (I_{sub}) in MOS transistors, which occurs the gate voltage is below the threshold voltage and mainly, consists of diffusion current.

4.2 : MODIFIED DUAL STAGE 4:2 COMPRESSOR USING STACK TECHNIQUE

In this approach, stack effect takes place by breaking down an existing transistor into two half size transistors. Figure shows its structure. When the two transistors are in off state at the same time, induced reverse bias between the two transistors results in sub threshold leakage current reduction. But these divided Therefore, in proposed system to compare the results better, have changed the length of each

gate level into 130nm and 32nm and observed the parameters such as area, power and power delay product. The dependence of subthreshold current is exploited by the transistor stacking effect and an increase in the source voltage V_s of the transistor reduces the sub-threshold leakage current exponentially. More leakage power savings can be obtained by increasing the number of transistors connected in stack structure. For circuits without stacking structure, forced stacking can be implemented. In forced stacking, a single transistor of width " W " can be replaced by two transistors of width " $W/2$ " each. This results in two transistors switching off at the same time reducing leakage current.

4.3 2*1 MUX USING STACK TECHNIQUE

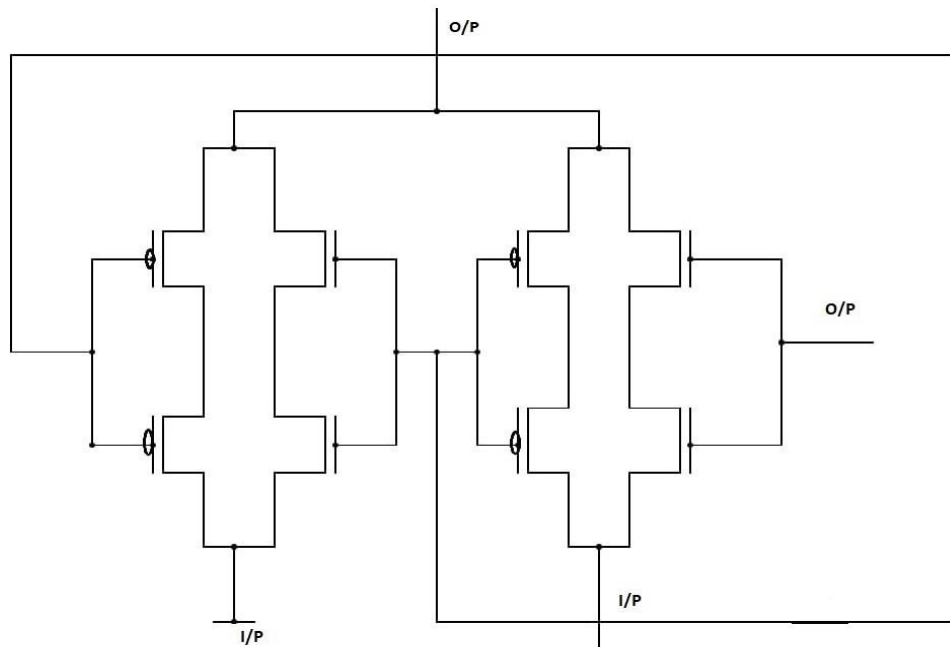


FIG 2.6 2*1 Mux Using Stack Technique

4.4 OR GATE USING STACK TECHNIQUE

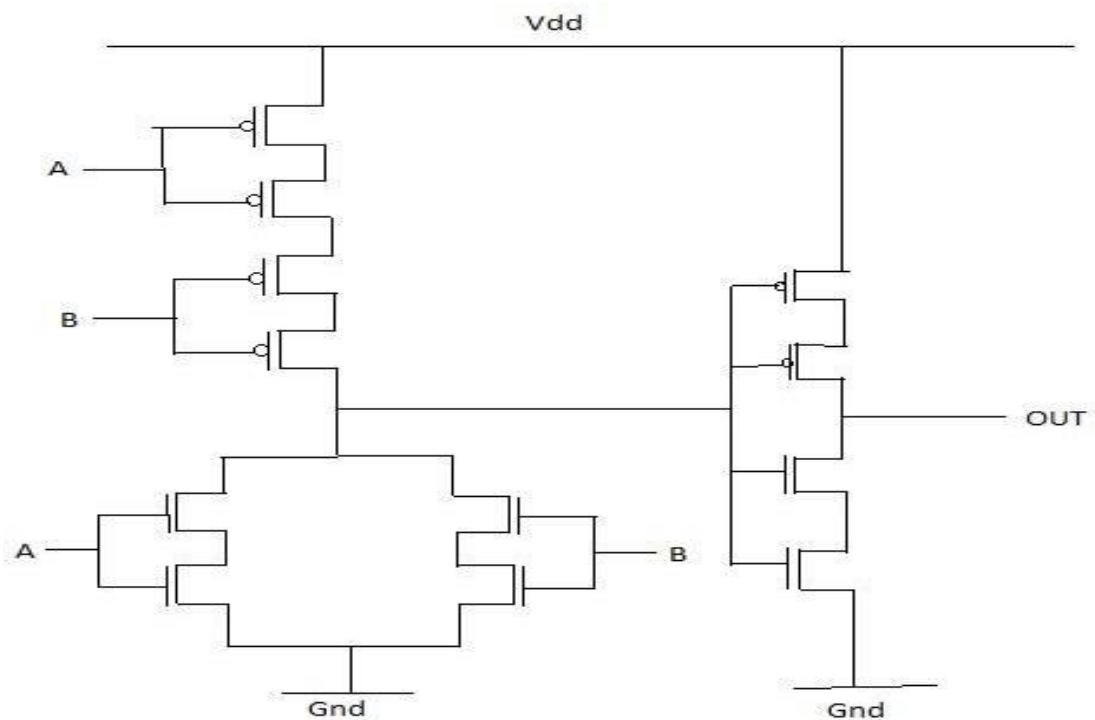


FIG 2.7 Or Gate Using Stack Technique

4.5 AND GATE USING STACK TECHNIQUE

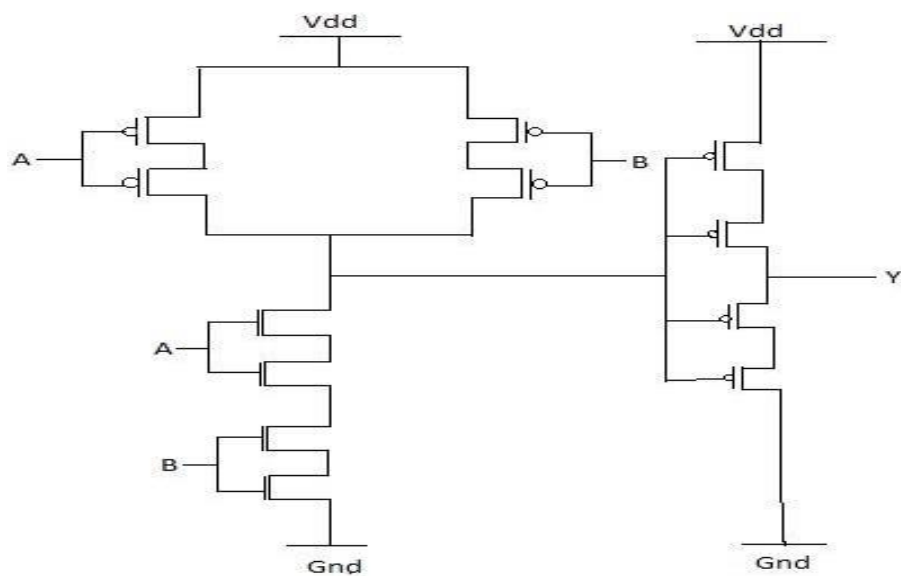


FIG 2.8 And Gate Using Stack Technique

4.6 XOR GATE USING STACK TECHNIQUE

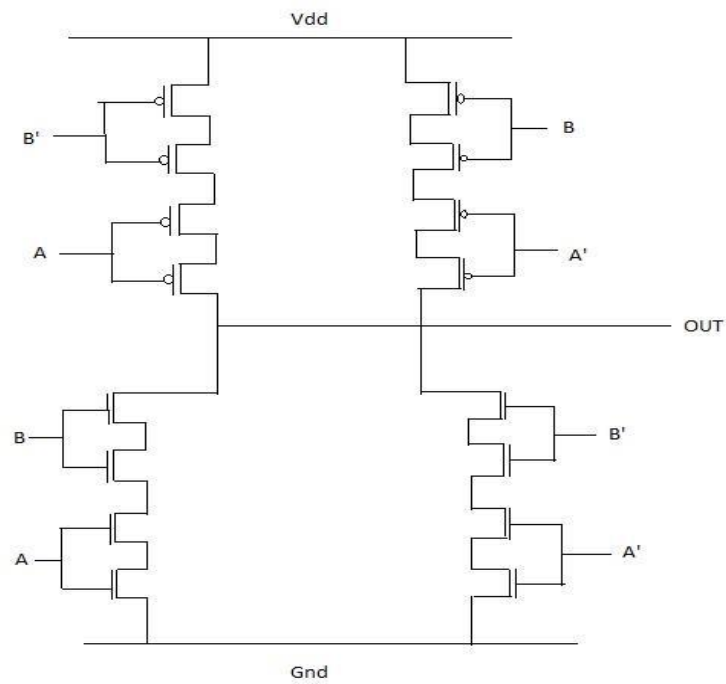


FIG 4.5 Xor Gate Using Stack Technique

CHAPTER 5

RESULT AND DISCUSSION

The power, delay and PDP reduced in modified dual stage compressor than existing. And the more power, delay and PDP is reduced in CMOS Technology under 32nm. The results and it is concluded that the proposed architecture reduces the power consumption compared with the existing system. The Synthesis is done using Hspice.

5.1 Simulation Results of area efficient 4:2 compressor using Full stack 130nm

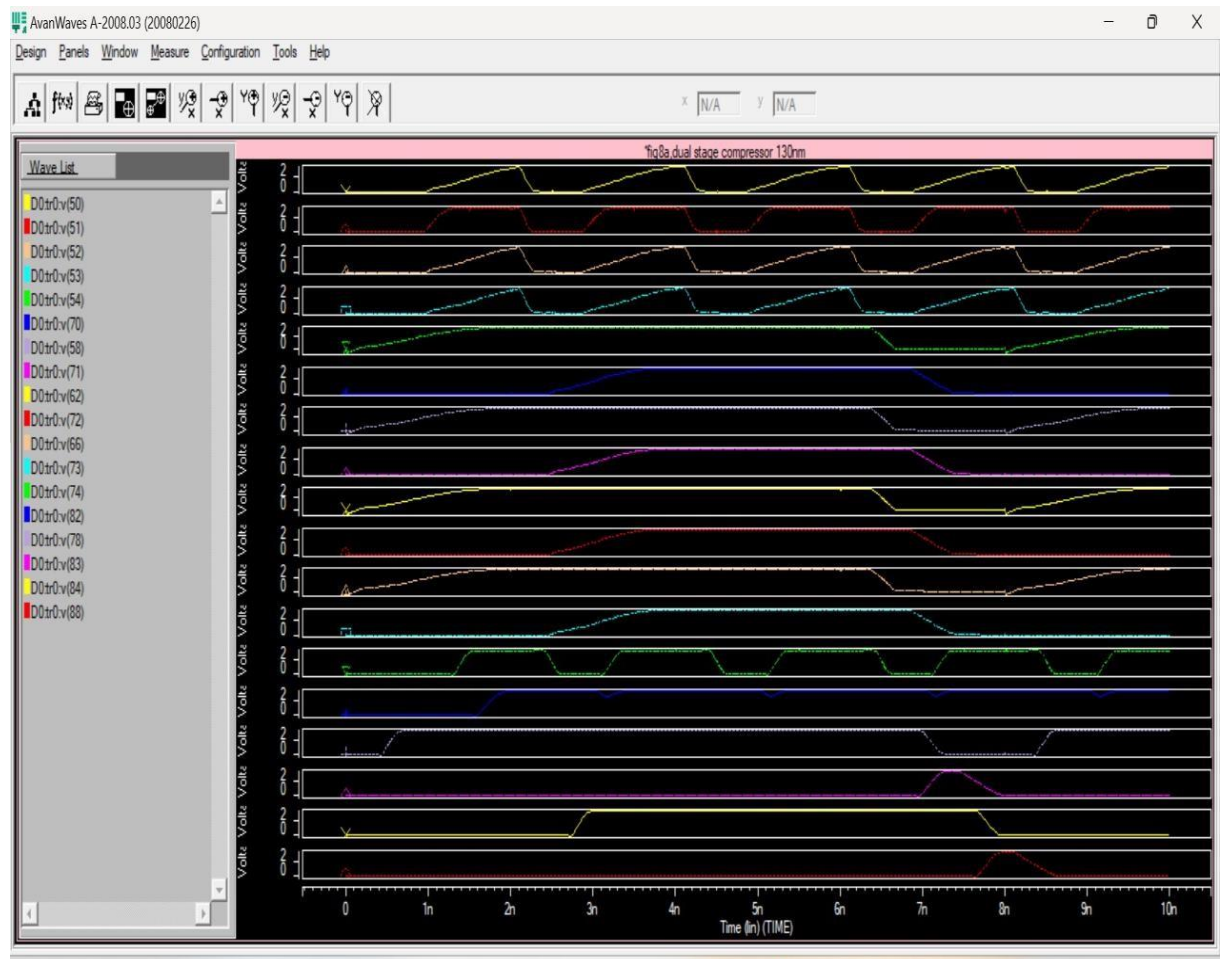


Fig 5.1 : Simulation results of Full stack 130nm area efficient 4:2 compressor

Figure 5.1 shows the simulation output of a Full stack 130nm area efficient4: 2 . The inputs given were a and b the output is taken by c and d.

5.2 : Simulation Results of dual stage 4: 2 compressor using Full stack 130nm

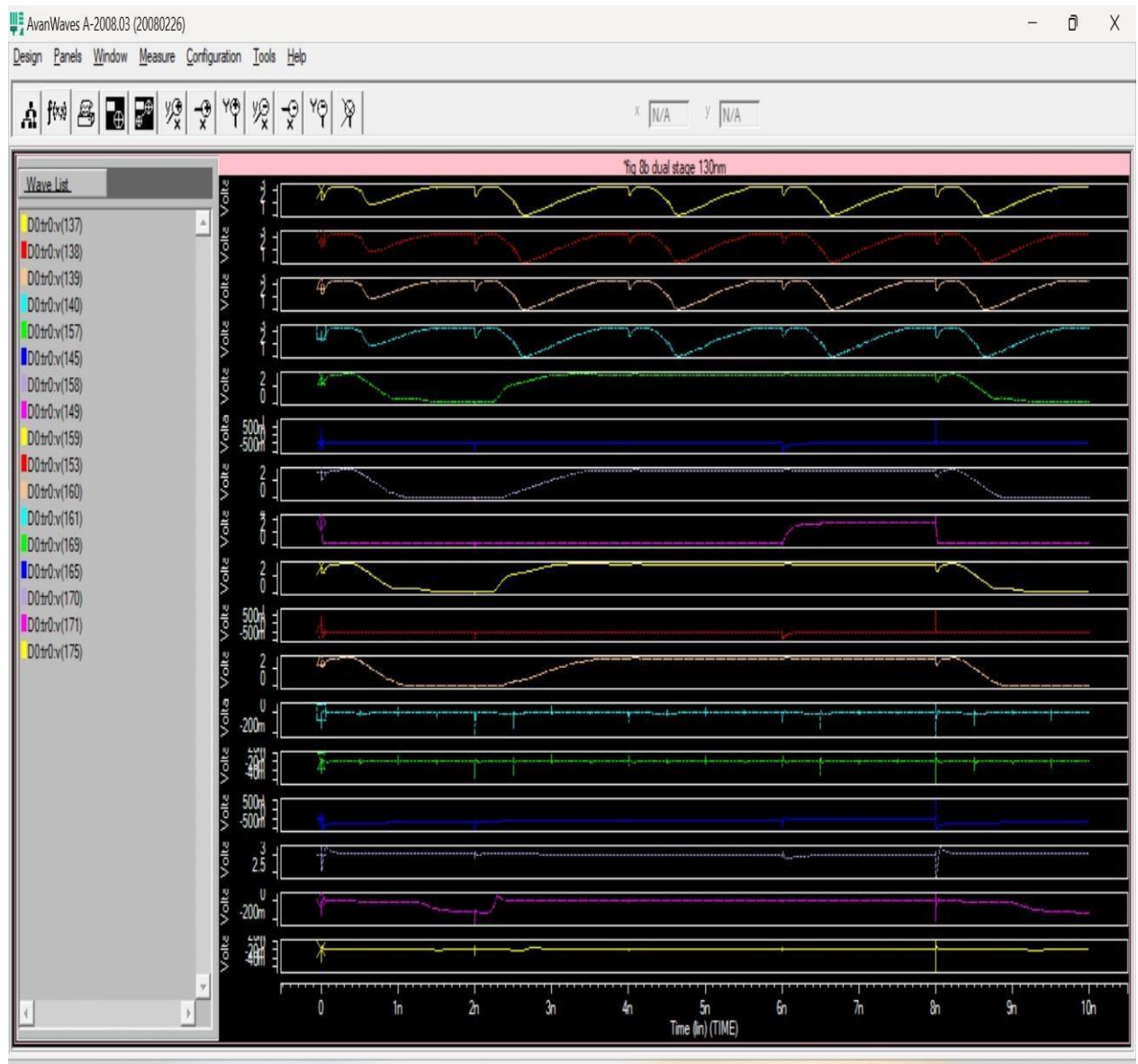


Fig 5.2 : Simulation results of Full stack 130nm dual stage 4:2 compressor

Figure 5.2 shows the simulation output of a full stack 130nm dual stage 4:2 compressor The inputs given were a, b and the output is taken by c.

5.3 PERFORMANCE ANALYSIS

The performance analysis of area efficient 4:2 compressor , dual stage 4:3 compressor in terms of power, power delay product has been given below.

5.3.1 PERFORMANCE ANALYSIS OF 130nm COMPRESSOR

DEVICE	POWER(W)	DELAY(S)	PDP (J)
OR	17.90u	54.25p	971.075a
XOR	17.15u	55.21p	946.8515a
AND	16.34u	25.69p	419.7746a
MUX	41.047u	54.21p	2225.157a
COMPRESSOR	93.80u	65.21p	6116.698a

TABLE 5.1 Performance Analysis Of 130nm Compressor

5.3.2 PERFORMANCE ANALYSIS OF 130nm MODIFIED COMPRESSOR using FULL-STACK

DEVICE	POWER(W)	DELYA(S)	PDP (J)
OR	9.59u	48.36p	463.772a
XOR	11.25u	49.58p	557.775a
AND	9.56u	19.56p	186.993a
MUX	32.59u	48.59p	1583.548a
COMPRESSOR	80.25u	56.26p	4514.865a

TABLE 5.2 Performance Analysis Of 130nm Modified Compressor using

Full-Stack

5.3.3 PERFORMANCE ANALYSIS OF 32nm COMPRESSOR

DEVICE	POWER(W)	DELAY(S)	PDP (J)
OR	7.58u	46.25p	350.575a
XOR	9.56u	48.52p	463.851a
AND	8.45u	18.25p	154.212a
MUX	30.25u	47.25p	1429.312a
COMPRESSOR	78.25u	54.25p	4245.062a

TABLE 5.3 Performance Analysis Of 32nm Compressor

5.3.4. PERFORMANCE ANALYSIS OF 32nm MODIFIED COMPRESSOR using FULL-STACK

DEVICE	POWER(W)	DELAY(S)	PDP (J)
OR	4.85u	34.12p	165.482a
XOR	6.01u	42.15p	253.321a
AND	5.56u	13.25p	73.67a
MUX	18.54u	41.55p	770.337a
COMPRESSOR	59.12u	48.25p	2852.54a

**TABLE 5.4 Performance Analysis Of 32nm Modified Compressor Using
Full-Stack**

CHAPTER 6

CONCLUSION

In this report have proposed a new technique to design stack technique using unique replacement of stack transistors, which gives better response in terms of average power and delay in comparison to the conventional stack technique. We have achieved a design of a stack technique using a 4:2 area-efficient dual stage compressor which has reduced the power to 3.125×10^{-4} watt and reduced the delay of the multiplier considerably. Since our objective was to reduce the power of the multiplier and it has been reduced by 3.125×10^{-4} watt, hence we achieved our objective. We propose a novel technique called STACK for designing High area-efficient 4:2 dual stage compressor which significantly cuts down the leakage current without increasing the dynamic power dissipation. This increases the resistance of the path from V_{DD} to ground, leading to significant decrease in leakage currents. The gate-level netlist of the given circuit is first converted into a static CMOS complex gate implementation and then STACK techniques are introduced to obtain a leakage-controlled circuit. Further, the proposed technique overcomes the limitations posed by other existing methods for leakage reduction. From the results obtained it can be verified that there is a significant amount of reduction in power dissipation of the circuits when STACK technique was used. Low power consumption in turn increases the efficiency of the overall system when compared to the existing systems. The significant feature of stack is that it works effectively in both active and idle states of the circuit, resulting in better leakage reduction compared to other techniques. Further, the proposed technique overcomes the limitations posed by other existing methods for leakage reduction. An analysis of a High area-efficient 4:2 dual stage compressor has been designed using stack 130nm and 32nm MOSFET technology.

REFERENCES

1. A. Saha, R. Pal, A. G. Naik, and D. Pal, "Novel CMOS multi-bit counter for speed-power optimization in multiplier design," *AEU-Int. J. Electron. Commun.*, vol. 95, pp. 189198, Oct. 2018.
2. D. Esposito, A. G. M. Strollo, E. Napoli, D. De Caro, and N. Petra, "Approximate multipliers based on new approximate compressors," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 65, no. 12, pp. 41694182, Dec. 2018.
3. H. Jiang, J. Han, F. Qiao, and F. Lombardi, "Approximate Radix-8 booth multipliers for low-power and high-performance operation," *IEEE Trans. Comput.*, vol. 65, no. 8, pp. 26382644, Aug. 2016.
4. H. Vasantha, Y. B. N. Kumar, and D. Dwivedi, "Design and analysis of multiplier using approximate 4-2 compressor," *AEU-Int. J. Electron. Commun.*, vol. 107, pp. 8997, Jul. 2019.
5. I. Alouani, H. Ahangari, O. Ozturk, and S. Niar, "A novel heterogeneous approximate multiplier for low power and high performance," *IEEE Embedded Syst. Lett.*, vol. 10, no. 2, pp. 4548, Jun. 2018.
6. J. Liang, J. Han, and F. Lombardi, "New metrics for the reliability of approximate and probabilistic adders," *IEEE Trans. Comput.*, vol. 62, no. 9, pp. 1760_1771, Sep. 2013.
7. K. M. Ha and S. Lee, "Multipliers with approximate 42 compressors and error recovery modules," *IEEE Embedded Syst. Lett.*, vol. 10, no. 1, pp. 69, Mar. 2018.
8. M. Ha and S. Lee, "Multipliers with approximate 42 compressors and error recovery modules," *IEEE Embedded Syst. Lett.*, vol. 10, no. 1, pp. 69, Mar. 2018.
9. P. J. Edavoor, S. Raveendran and A. D. Rahulkar, "Approximate Multiplier Design Using Novel Dual-Stage 4:2 Compressors," in *IEEE Access*, vol. 8, pp. 48337-48351, 2020.
10. R. Zendegani, M. Kamal, M. Bahadori, A. Afzali-Kusha, and M. Pedram, "RoBA multiplier: A rounding-based approximate multiplier for high-speed yet energy-efficient digital signal processing," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 2, pp. 393401, Feb. 2017.
11. S. Ghosh, D. Mohapatra, G. Karakonstantis, and K. Roy, "Voltage scalable high-speed robust hybrid arithmetic units using adaptive clocking," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 18, no. 9, pp. 13011309, Sep. 2010.

12. S. Hashemi, R. I. Bahar, and S. Reda, "DRUM: A dynamic range unbiased multiplier for approximate applications," in *Proc. IEEE/ACM Int. Conf. Computer-Aided Des. (ICCAD)*, Austin, TX, USA, Nov. 2015, pp. 418_425.
13. S. Narayanamoorthy, H. A. Moghaddam, Z. Liu, T. Park, and N. S. Kim, "Energy-efficient approximate multiplication for digital signal processing and classification applications," *IEEE Trans. Very Large Scale Integr.(VLSI) Syst.*, vol. 23, no. 6, pp. 1180_1184, Jun. 2015.
14. S. Mittal, "A survey of techniques for approximate computing," *ACM Comput. Surv.*, vol. 48, no. 4, pp. 1_33, Mar. 2016.
15. G. Zervakis, S. Xydis, K. Tsoumanis, D. Soudris, and K. Pekmestzi, "Hybrid approximate multiplier architectures for improved power-accuracy trade-offs," in *Proc. IEEE/ACM Int. Symp. Low Power Electron.Des. (ISLPED)*, Rome, Italy, Jul. 2015, pp. 79_84.
16. S.-F. Hsiao, M.-R. Jiang, and J.-S. Yeh, "Design of high-speed low-power 3-2 counter and 4-2 compressor for fast multipliers," *Electron. Lett.*, vol. 34, no. 4, p. 341, Feb. 1998.
17. Y.-J. Chang, Y.-C. Cheng, Y.-F. Lin, S.-C. Liao, C.-H. Lai, and T.-C. Wu, "Imprecise 4-2 compressor design used in image processing applications," *IET Circuits, Devices Syst.*, vol. 13, no. 6, pp. 848856, Sep. 2019.
18. Y. Guo, H. Sun, L. Guo, and S. Kimura, "Low-cost approximate multiplier design using probability-driven inexact compressors," in *Proc. IEEE Asia Pacific Conf. Circuits Syst. (APCCAS)*, Chengdu, China, Oct. 2018, pp. 291_294.
19. Z. Wang, G. A. Jullien, and W. C. Miller, "A new design technique for column compression multipliers," *IEEE Trans. Comput.*, vol. 44, no. 8, pp. 962_970, Aug. 1995.