Photocopy and Use Authorization

In presenting this dissertation in the partial fulfillment of the requirements for an advanced degree at Idaho State University, I agree that the library shall make it freely available for inspection. I further state that the permission for extensive copying of this dissertation for scholarly purposes may be granted by the Dean of the Graduate Studies, Dean of the College of Science & Engineering or by the University Librarian. It is understood that any copying or publication of this dissertation for financial gain shall not be allowed without my written permission.

Signature _____

Date _____

HIGH PERFORMANCE CMOS CIRCUITS FOR FAST CARRY GENERATION WITH POWER OPTIMIZATION

By

Naga Spandana Muppaneni

A Dissertation

submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy in Engineering and Applied Science

in the Department of Electrical Engineering,

Idaho State University

December 2019

To the Graduate Faculty:

The members of the committee appointed to examine the dissertation of Naga Spandana Muppaneni find it satisfactory and recommend that it be accepted.

_____
Dr. Steve C.Chiu,
(Major Advisor)

_____
Dr. Gene Stuffle,
Committee Member

_____
Dr. Stevan G. Hunter,
Committee Member

_____
Dr. Ikhoon Shin,
Committee Member

_____
Dr. Chikashi Sato
Graduate Faculty Representative

DEDICATION

I would like to dedicate this dissertation to my husband Kunal Nadella, my parents Kusuma Muppaneni and Narendranath Muppaneni, my sisters Anusha Muppaneni and Prathusha Muppaneni for encouraging and supporting me in every possible way.

ACKNOWLEDGEMENTS

First of all, I would like to offer my sincere reverence to Almighty whose blessings have enabled me to complete an important phase of my life.

I would like to express my sincerest gratitude and special thanks to my major advisor, Dr. Steve C. Chiu for his continued support, invaluable guidance, constant encouragement, immense patience, useful discussion and peerless criticisms throughout this research and also, I would like to express my deep appreciation for his readiness to always make time for me and I have always cherished meeting with him in person for discussions. Thank you very much Dr.Chiu.

I am highly grateful to Dr. Gene Stuffle for your constant support and encouragement throughout my years of study at ISU and for believing in me, that I could teach and giving me my first opportunity to teach in the United States. Thanks for everything Dr. Stuffle, especially for helping me with my internship paperwork.

My immense gratitude to Dr. Stevan G. Hunter, for his ceaseless support and guidance since the beginning days of my research and also referring me for my internship position. Thank you, Dr. Hunter, for always providing me your time and encouragement.

I would also like to thank Prof. Farshad Dailami, for his valuable guidance and for helping me in my endeavors. I always enjoyed his lecture classes and gained a rewarding experience working with him.

I would like to thank Dr. Ikhoon Shin for providing your time and insightful feedback during my research presentation.

I would also, like to thank Dr. Chikashi Sato for accepting to be the GFR on my committee.

I am profoundly grateful to the Department of Electrical Engineering at ISU, for supporting me financially and otherwise as part of my study over the years. Also, would like to thank Ellen and

Laura for always coming forward to help with any kind of paperwork at ISU.

Finally, and most importantly, I am very grateful to my family for their love, support, and prayers. I would like to wholeheartedly thank my husband, for always being by my side, and believing that I could do it. I would also like to thank my colleagues and friends, Anupama, Sabin, Bryan Baker, Harman, Ashwini, and Srujana for their moral and emotional support throughout the years of my study.

Contents

List of Figures

List of Tables

List of Abbreviations

ALU	Arithmetic Logic Unit
CLA	Carry Look-ahead Adder
CMOS	Complementary Metal Oxide Semiconductor Transistor
ECAD	Electronic Computer-Aided Design
FET	Field Effect Transistor
FPGA	Field Programmable Logical Gate Array
GPU	Graphical Processing Unit
HDL	Hardware Description Language
IC	Integrated Circuit
NMOS	N-channel Metal Oxide Semiconductor
PC	Program Counter
PMOS	P-channel Metal Oxide Semiconductor
RTL	Register Transfer Level
SPICE	Simulation Program with Integrated Circuit Emphasis
VLSI	Very Large-Scale Integrated Circuits

High Performance CMOS Circuits for Fast Carry Generation with Power Optimization

Dissertation Abstract – Idaho State University (2019)

Carry Look-ahead adders (CLA) are used in the broad spectrum of applications from electronics to medical industry. The critical parameters for the designers of CLA are their speed, size and power. CLA are the core of the arithmetic logic units, graphic processing units of the microprocessors and to perform higher bit addition using CLA in these units of the computers, cascaded structures are commonly used. In this kind of structures, for an instance, a 16-bit addition could be performed by connecting four of 4-bit adders in series. The trade-offs of these structures are, the propagation time of the carry signal is higher and they need more transistors to realize the logic. Increase in the transistor count leads to higher switching activity and higher power dissipation. This research presents new stand-alone design topologies for CLA carry generation circuits for 1, 2, 4, and 8-bit addition using complementary metal oxide semiconductor technology. These designs have both input and output signals connected on a single net, in a single architecture. They provide better efficiency compared to the cascaded structures. This work presents, the schematics and the layouts of the proposed stand-alone CLA carry generation implementations and reference stand-alone CLA implementations at various CMOS technology nodes. The proposed stand-alone CLA carry generation circuits are evaluated and compared in terms of their transistor implementation and propagation delay with the reference stand-alone CLA circuits which have been published. Both analytically and practically the proposed stand-alone CLA carry generation circuits outperform, the reference stand-alone CLA carry generation circuits.

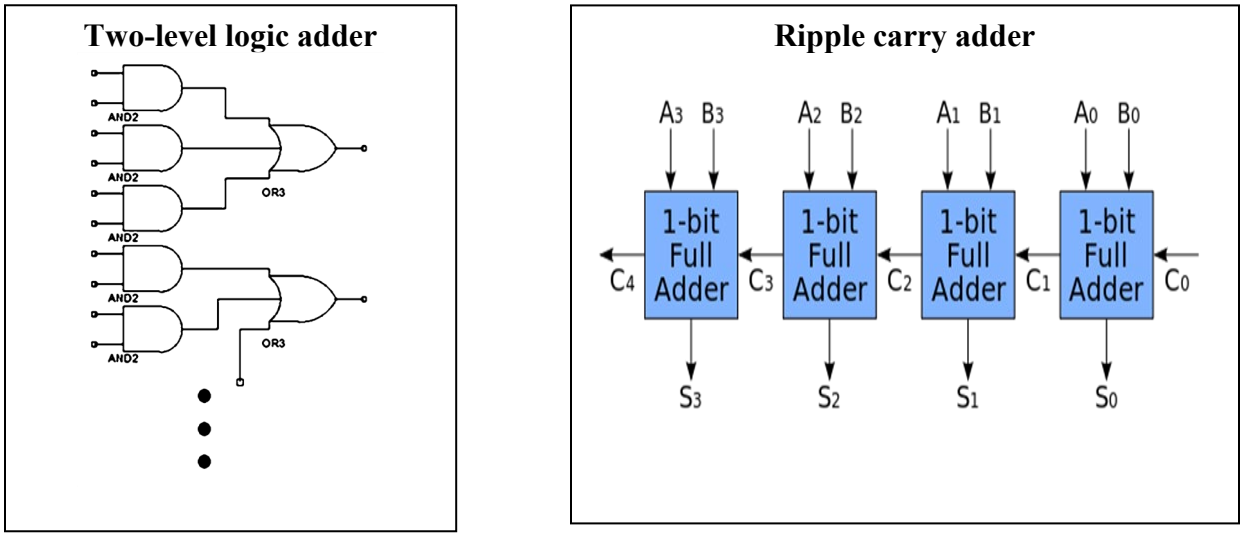
Key words: Carry Look-ahead Adder (CLA), Carry Generation Circuits, Propagation Delay, Stand-alone Design Topology, Complementary Metal Oxide Semiconductor (CMOS) and Speed.

Chapter 1. Introduction

1.1 General Introduction:

Over the years, digital adders have been widely used in a range of applications from electronics, controls, automobiles, and telecommunications to the medical and aviation industries. They are the building blocks of the Arithmetic Logic Unit (ALU) and Graphic Processing Unit (GPU) of various types of computers and digital signal processors.

Digital adders play a vital role in high-performance computing. There are various types of digital adders in the market. As the number of bits to be added increases, the fastest digital addition is possible by direct addition of the augends and addends, using a two-level logic adder. This is shown in Figure 1.1 (a).



(a) (b)

Figure 1.1: (a) Two-level logic adder [20]; (b) Block diagram of a 4-bit Ripple carry adder [15]

A two-level logic adder is implemented by feeding a level of AND gates into the level of OR gates. It is hard to implement the high-level bit addition logic using a two-level logic adder since it requires millions of transistors to implement it at the circuit level. For instance, a two-level logic

adder requires two million transistors to implement a 16-bit adder [3] [20]. Hence, in a two-level logic adder, the transistor count increases exponentially as the number of bits to be added increases. This increases the size of the device.

One solution for reducing the size of the device is to use a ripple carry adder instead of a two-level logic adder. Ripple carry adders use full-adder logic blocks for processing each bit addition and the carry bits between each adder are cascaded and propagated. Figure 1.1 (b) represents a 4-bit ripple carry adder, where $A_3A_2A_1A_0$ and $B_3B_2B_1B_0$ represent two 4-bit binary inputs, $S_3S_2S_1S_0$ and $C_4C_3C_2C_1$ denote the output sum and carry bits of each corresponding bit addition. The addition takes place from right to left in the order of least significant bits to the most significant bits. Figure 1.1 (b) shows that the carry-out signal of each full-adder serves as the carry-in input signal of the next most significant full-adder. This rippling of the carry signal from one stage to the other does not allow the full-adders to operate simultaneously. Each full-adder has to wait for its input carry signal to be available from its preceding full-adder block, which leads to the propagation delay of the carry signal from input to output. Hence, the ripple carry adder reduces the size of the device but escalates the propagation delay. As, the propagation delay of the adder increases, its speed decreases.

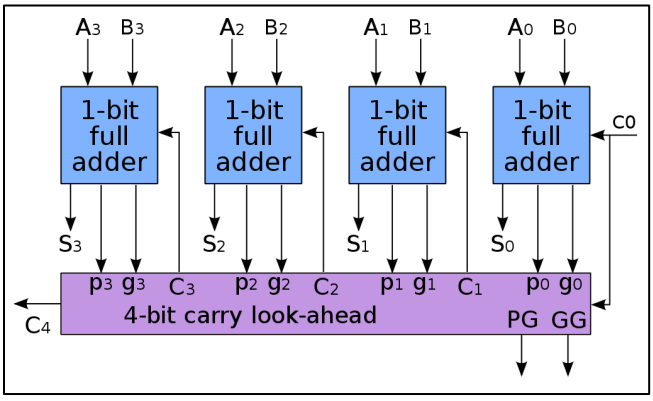


Figure 1.2: Block diagram of a 4-bit carry look-ahead adder [4] [7]

The Carry-Look Ahead Adder (CLA) is one of the fastest digital adders. Figure 1.2 represents a 4-bit CLA. It computes the carry signals ahead of the sum signals. Each full-adder block addition in the CLA is independent of the output carry signal of its preceding full-adder block as shown in Figure 1.2. This reduces the propagation delay of the carry signal compared to the ripple carry signal. Apart from the sum ($S_3S_2S_1S_0$) and carry ($C_4C_3C_2C_1$) output signals, the CLA uses additional circuitry to improve the speed of the adder. It generates two additional signals, generate ($G_3G_2G_1G_0$) and propagate ($P_3P_2P_1P_0$) signals as seen in Fig 1.3. This adder acts as a compromise between the speed of the two-level logic adder and the size of the ripple carry adder. Hence the CLA is considered throughout this study.

A CLA can be implemented using several technologies. Among them, Complementary Metal Oxide Semiconductor (CMOS) technology is the most widely used. High noise immunity and low static-power dissipation are the striking features of CMOS technology. These features help CMOS technology to stand-out from other technologies and make it popular among designers to realize various logic specifications. In this dissertation, all schematics are implemented in CMOS technology. CMOS ON Semi C5 process technology node and 180nm process technology node are used to design the schematics and perform the simulations. Simulation is the process of using **s**imulation software (a simulator) to verify the functional correctness of a digital design that is modeled using an HDL (hardware description language) like Verilog.

1.2 Problem Statement:

The addition operation of most ALU's of microprocessors and digital signal processors is realized physically by using a CLA [7]. The major factors influencing the performance of the CLA are propagation delay, layout area, and dynamic power dissipation.

The propagation delay of the carry signal is a measure of the time taken for an input carry signal

to traverse through the circuit to the output end. The propagation delay factor of CLA also depends on the number of transistors used to implement the logic and the design topology of the adder circuit. As the transistor count increases, the propagation delay of the CLA increases and it affects the speed of the circuit.

The layout size of the CLA also largely depends on the transistor count of the design and the structure of the CLA. As the layout size of the circuit increases, it occupies more area on the chip. This makes the chip costlier.

Dynamic power dissipation of any CMOS circuit depends largely on the switching activity of the circuit. Switching activity of the circuit is a factor of the number of transitions taking place in the circuit. Dynamic power dissipation of the CLA circuit can be reduced by decreasing the transistor count of the design and turning off unwanted signals during the circuit operation.

For implementing the addition of a higher number of bits using CLA, cascaded carry chain structures are used. In these cascaded circuits, the output carry signal is generated after all the intermediate carry signals are computed from the preceding blocks as shown in Figure 1.3. This concept of determining the intermediate carry signals increases the number of transistors required to implement the carry look-ahead logic in CLA. As the number of bits to be added rises, the propagation delay of the carry signal, layout size and the dynamic power dissipation of the cascaded CLA carry generator circuit increase significantly. These issues make the modern IC design more complex.

The purpose of this dissertation is to reduce propagation delay, layout size and dynamic power dissipation of CLA carry generator circuits by introducing proposed stand-alone CLA carry generation circuits. These proposed stand-alone CLA carry generation circuits are circuits whose input and output carry signals are within a single architecture, with simple circuitry. Here the carry

signal of CLA directly propagates from input to output, within a single circuit. In this study, a variety of stand-alone CLA carry generation circuit topologies are introduced.

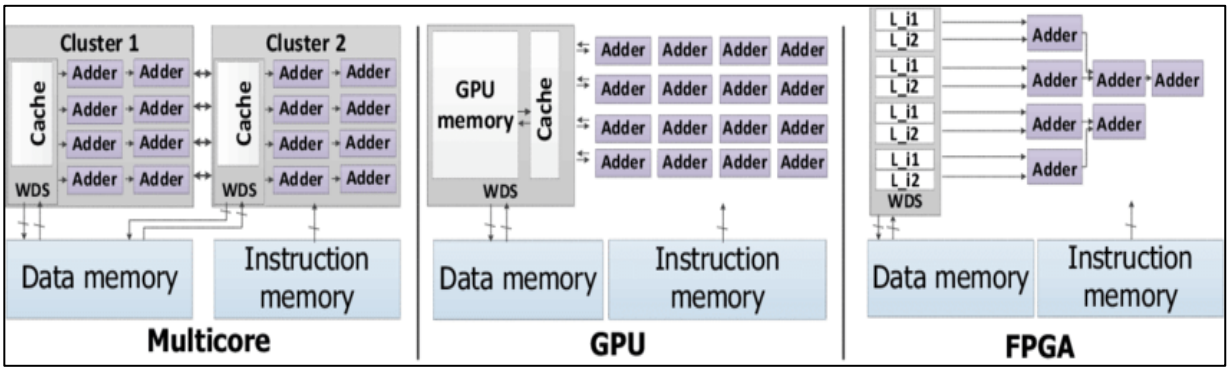


Figure 1.3**:** Block diagrams of multicore, GPU processors and FPGA with adders [5]

Similarly, in graphically-related applications, the GPU needs faster adders for higher performance, and also in FPGA's in their configurable logic blocks, and in a multicore processor. Adders are also used to generate memory addresses that help the Program Counter (PC) to point to the next instruction to be executed It is typical that adders take a large area on the circuit as seen in Figure 1.3. All of the example circuits in Figure 1.3 have the same issues discussed in this dissertation. The performance of the circuits in Figure 1.3 can be improved by incorporating the proposed stand-alone CLA carry generation circuits.

1.3 Dissertation Objective and Outline

The objective of this dissertation is to reduce the carry propagation delay by introducing proposed stand-alone CLA carry-generation circuits for 1-bit, 2-bit, 4-bit, 8-bit and 16-bit CLA. To demonstrate the benefits of proposed stand-alone CLA carry generation designs, they are compared with current CLA carry implementation designs by simulation.

The outline of this dissertation is as follows:

Chapter 2: History and the State of Art in CLA, provides an overview of CLA operation, it is generated and propagate signals, and the mathematical equations of CLA. The previous

implementations of CMOS CLA carry-generation circuits for 1, 2, 4, 8 and 16-bit carry-generation circuits and their critical path delays are compared and their tradeoffs are described.

Chapter 3: Proposed Stand-alone Carry Generation Circuits. In this section of the dissertation, the schematics of the design implementations of proposed stand-alone carry generation circuits are introduced. They are designed using Electronic Computer-Aided Design (ECAD) tools and are analyzed based on their propagation delay, layout size, and power dissipation. The rise and fall delays are computed across each transition path of the proposed stand-alone circuits. Throughout the process, the performance of the designs in various CMOS technology nodes is analyzed. Transistor resizing is done in the process of optimization. Layouts of the proposed design configurations are presented. Critical path propagation delays are computed both empirically and theoretically.

Chapter 4: Comparative Results and Analysis, presents comparative results of the proposed stand-alone CLA carry-generation design topologies and the previous CLA design implementations based on their propagation delay, layout area, and dynamic power dissipation. It explains the benefits of proposed stand-alone CLA carry generation circuits over the previous CLA carry generation implementations in a complete CLA adder circuit and its application in the adder circuit.

Chapter 5: Conclusion and Outlook. This chapter summarizes and clarifies the work done in this dissertation by elucidating the importance of the proposed stand-alone CMOS CLA carry generation circuits and by a brief outlook of potential applications.

Chapter 6: Recommended Future Work, lists areas for further study, and enhancements that could be made to these designs and their implementation using various design technologies and nodes.

1.4 Glossary

For the sake of clarity, the basic terminology used throughout the dissertation is defined here:

Rise Time (t_r): It is defined as the time is taken for a signal to rise from 10% to 90% of its power supply voltage. In this dissertation, it is used to analyze the speed of proposed stand-alone circuits.

Fall Time (t_f): It is defined as the time taken for a signal to fall from 90% to 10% of its power supply voltage. In this dissertation, it is used to analyze the speed of proposed stand-alone circuits.

Propagation Delay (t_p): It is defined as the time taken for the logic signal in a digital circuit to travel from its input to its output. It is computed by the difference in the transition times of the input and output signals at 50% of its power supply voltage. In this dissertation, the average propagation delay is considered throughout.

Carry Rippling: The traverse of the carry signal from one circuit to another.

Critical Path: The longest delay path across the circuit from the input to the output.

Mobility of the transistor: It is defined as the speed of movement of the charge carriers per unit voltage gradient across the channel. For an NMOS, transistor, electrons are the charge carriers and for PMOS transistor, holes are the charge carriers.

Switching Speed of the transistor: It is defined as the frequency at which the transistor switches the logic from one state to the other state.

Design Metrics: The parameters used by the designers to evaluate the performance of the design topology.

Aspect Ratio: The ratio of width to length of a transistor is known as the aspect ratio of the transistor.

1.5 Tools Used

Throughout this study, various tools have been used for design implementation, simulation, and

analysis.

Tanner Tools: Professional ECAD tool. During this study, S-editor, T-editor, and L-editor tools are used for schematic implementation, net-list generation and layout correspondingly on 0.25µm processing node.

Electric VLSI: An open-ended ECAD tool used for drawing schematics and integrated layout. In this dissertation, all proposed stand-alone carry generation circuits are designed and laid out using this tool.

LTspice: A freeware Spice tool used for circuit design. All the schematics and layouts implemented in this study are designed using Electric VLSI tools and simulated using LTspice.

Chapter 2. Background and Literature Review

2.1 Arithmetic Logic Unit (ALU):

Various types of computers and electronic circuits use ALU to compute all the arithmetic and logical operations. It is a digital circuit which is comprised of adder, shifter and multiplexer components. All the addition operations in ALU are performed by a digital adder. There are many progressive adders that have come into existence for this operation. CLA is one among them, which is considered to be one of the optimal digital adders. It is known for its look-ahead carry computation feature. The major chip area of ALU is occupied by adder circuitry as seen in Figure 2.1.

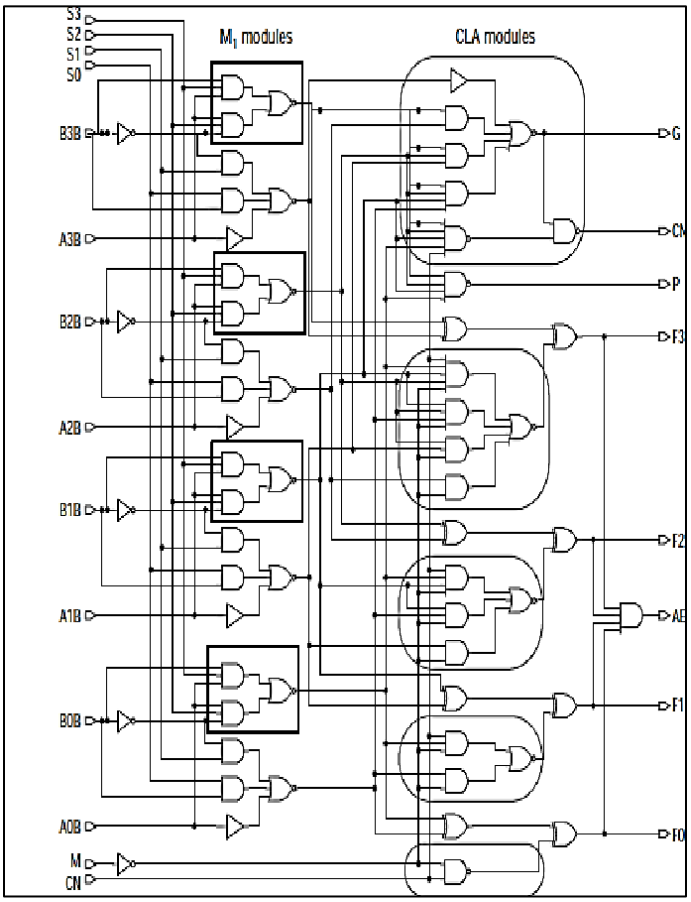


Figure 2.1**:** Logical Diagram of 4-bit ALU [29]

Figure 2.1 represents the logical gate-level implementation of 4-bit ALU with a 4-bit carry look-

ahead adder in it. This chapter illustrates the mathematical implementation, applications and the design issues of CLA.

2.2 Literature Research

This section illustrates the glimpses of literature research of this study.

2.2.1 Intel 8008 Microprocessor

Intel's first 8-bit 8008 microprocessor includes 8-bit CLA in its architecture for its arithmetic operations. Figure 2.2 shows the block diagram of this microprocessor with CLA highlighted.

Limitation: The CLA circuit in this block occupied as much die space as the ALU itself, which is the major drawback of this processor [30].

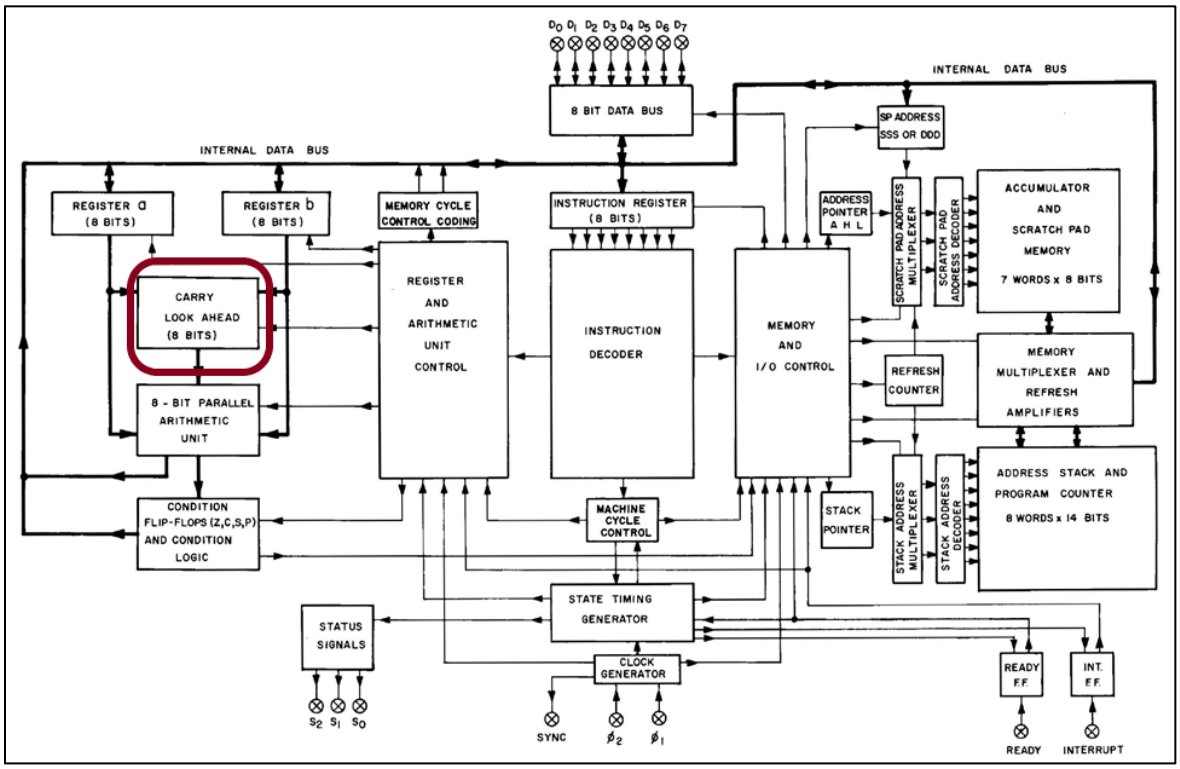


Figure 2.2: Block diagram of Intel 8008 microprocessor [30]

2.2.2 74181 Bit-Slice Arithmetic Logic Unit

74181 is a 4-bit parallel ALU chip, which performs 16 arithmetic and 16 logical operations. It uses a look-ahead logic to realize all the arithmetic operations. To satisfy this feature CLA is

incorporated into 74181 ALU, which could be noticed from the figure 2.3 [31] [32].

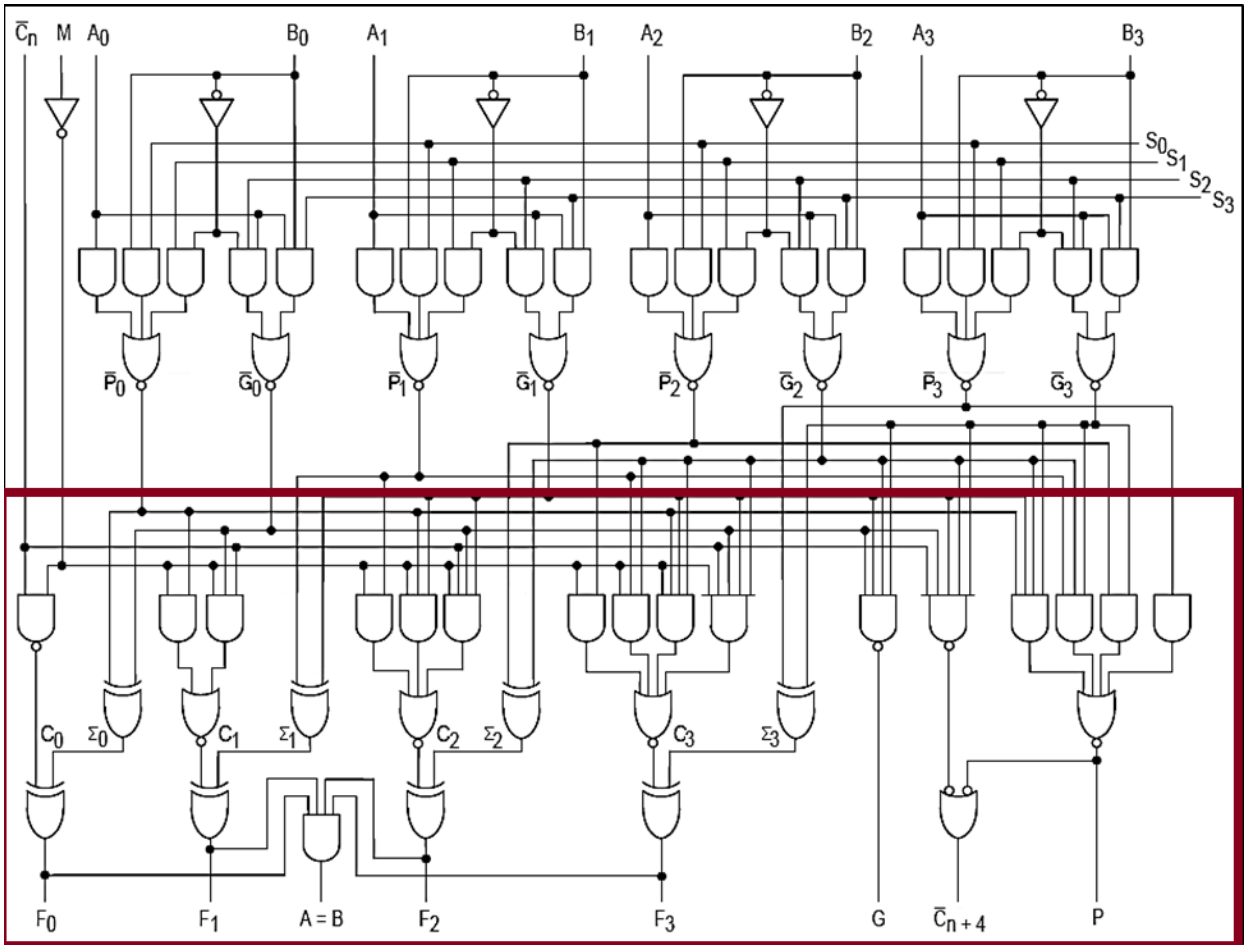


Figure 2.3**:** Logical Circuit Diagram of 74181 Integrated Circuit. [31] [32]

74181 ALU is also used in several processors such as Data General Nova, the Xerox Alto, the VAX-11/780 and Texas Instruments-990.

Limitation: For 8-bit and 16-bit operations, multiple 4-bit slices of 74181 ALU chip are connected serially or cascaded [32]. Hence for higher bit 74181 ALU circuitry, the size and the delay of the integrated circuit increases to a greater extent.

2.2.3 Zilog Z80 microprocessor

Z80 microprocessor is an 8-bit processor. The program counter updates, increment, decrement of the instructions and loop counter functions are served by using a 16-bit incrementer/ decrementer.

It uses CLA to generate the carry from bits 0-7, 7-11 and 12-14. The size of the Z80 ALU is 4-bits, so each 8-bit operation requires two iterations to complete the processing through ALU.

Limitation: The limitation of this processor is it requires higher processing time to complete the execution of higher bit operations. The proposed high–bit stand-alone CLA designs in this study, would eliminate this drawback by replacing 4-bit CLA structures with 8-bit CLA structures.

2.3 Operation of CLA

The addition operation of CLA is similar to its other contemporary adders, but it stands out from other adders in computing the carry generate and propagate signals ahead of the sum computation. Figure 1.1 presents the 4-bit CLA block diagram. The sum computation in this figure is done by using full-adder blocks for each bit addition discretely and the carry computation is done based on the following principle. The carry signals of each full-adder stage are calculated independently of the carry signals generated in their preceding stages. This principle can be clearly explained by the logical carry equations of the CLA [3] discussed below.

A general Boolean equation for carry generation of CLA is

$$C_{i+1} = G_i + (P_i \cdot C_i) \tag{2.1}$$

Where *i* represents the number of the adder block, G_i and P_i denote generate and propagate signals respectively, they are defined based on the input bits of the corresponding adder block.

$$G_i = A_i \cdot B_i \tag{2.2}$$

$$P_i = A_i + B_i \tag{2.3}$$

Where A_i and B_i represent the inputs signals of each adder block.

For a 4-bit CLA, as seen in Figure 1.1, the carry signal equations for each full-adder block, are as follows:

$$C_1 = G_0 + (P_0 \cdot C_0) \tag{2.4}$$

$$C_2 = G_1 + (P_1 \cdot C_1) \tag{2.5}$$

$$C_3 = G_2 + (P_2 \cdot C_2) \tag{2.6}$$

$$C_4 = G_3 + (P_3 \cdot C_3) \tag{2.7}$$

Equations (2.4) to (2.7), can be re-written by using equations (2.2) and (2.3) as follows [5] [6] [7]:

$$C_1 = G_0 + (P_0 \cdot C_0) \tag{2.8}$$

$$C_2 = G_1 + (G_0 \cdot P_1) + (P_0 \cdot P_1 \cdot C_0) \tag{2.9}$$

$$C_3 = G_2 + (G_1 \cdot P_2) + (G_0 \cdot P_1 \cdot P_2) + (P_0 \cdot P_1 \cdot P_2 \cdot C_0) \tag{2.10}$$

$$C_4 = G_3 + (G_2 \cdot P_3) + (G_1 \cdot P_2 \cdot P_3) + (G_0 \cdot P_1 \cdot P_2 \cdot P_3) + (P_0 \cdot P_1 \cdot P_2 \cdot P_3 \cdot C_0) \tag{2.11}$$

From all the above equations, it is evident that the carry signals of each stage are independent of the carry signals computed in their previous stages. They only depend on generate and propagate signals, which are computed by the input bits and hence, all the carry signals are anticipated ahead of time, overcoming the limitation of the carry signal rippling in ripple-carry adders. Logical gate-level implementation for regular 4-bit CLA for equations (2.8) to (2.11) shown in Figure 2.4 [8].

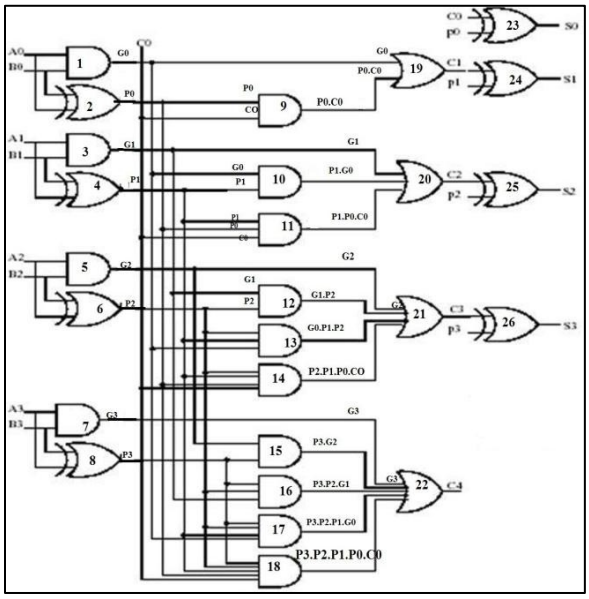


Figure 2.4: Gate-level implementation of 4-bit CLA [8]

2.4 CMOS Implementations of CLA

Digital design flow of any circuit starts from a design specification and then its behavioral description of the specification, which is then converted to Register Transfer Level (RTL) description using Hardware Description Language (HDL) to gate-level implementation using logic gates. This implementation is further converted to transistor-level implementation normally by logic synthesis function.

Designers often describe a process by its feature size. Feature size refers to the minimum transistor length. All the schematics designed and simulations performed in this chapter are done using ON Semi C5 process technology node, which is CMOS 500nm technology. The minimal channel length of this technology node is 0.6μm. So, all the PMOS and NMOS transistors of the schematics presented in this chapter maintain the same gate length of 0.6μm.

2.4.1 Basic CMOS Inverter

CMOS transistor technology is one of the most popularly used transistor technologies to realize the logic specifications. The features like low dynamic power dissipation, high fan-out capability, high operating speed and ease of implementing the logic at transistor level make CMOS, a proven technology. The basic CMOS structure used for implementing logic design is the inverter structure. Simple CMOS inverter schematic is seen in Figure 2.5.

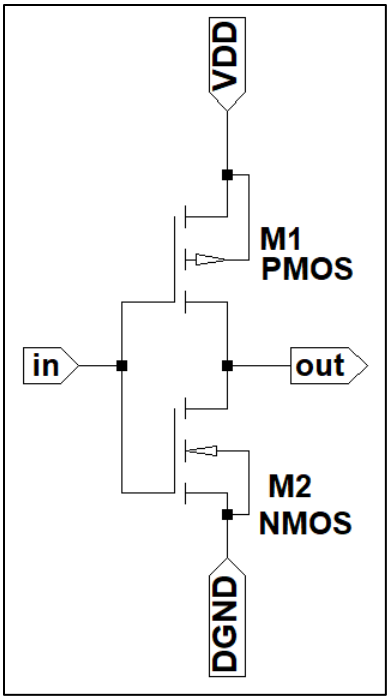


Figure 2.5**:** Schematic of CMOS Inverter

It comprises of both PMOS and NMOS transistors, which are complementary in nature. They are used in tandem. In Figure 2.5 'VDD' represents the power supply voltage, for ON Semi C5 process VDD is equal to 5 volts. 'DGND' represents the digital ground signal, which by default is equivalent to 0 volts. A common input signal is applied to their gate terminal and the drain terminals of both PMOS and NMOS transistors tied together to form an output side. The source terminal of the PMOS transistor is connected to the power supply voltage, 'VDD', whereas the source terminal of NMOS is connected to ground. PMOS transistor is known as pull-up transistor since it pulls-up the output to power supply voltage when turned on. NMOS transistor is known as a pull-down transistor since it pulls-down the output to the ground when activated.

Normally, the mobility of NMOS transistor is higher than the PMOS transistor and hence this increases the resistance of the PMOS transistors. To have symmetrical rise and fall times of the output voltage signal in a CMOS inverter circuit and to achieve the desired switching speed of the

transistors, the size of the PMOS transistor is maintained higher than NMOS.

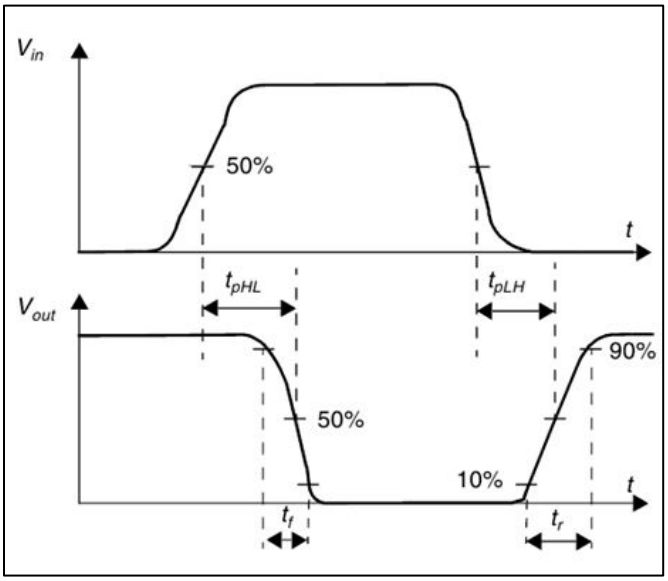


Figure 2.6: Input and Output waveforms of a CMOS inverter

The propagation delay of a CMOS inverter defines how quickly the inverter responds to the transitions in the input. It is measured by the difference in the time taken for an input and output signal of the inverter to reach 50% of the VDD. It can be noticed from Figure2.6 that the propagation delay from high to low (t_{pHL}) is the delay when output switches from high-to-low when input switches from low-to-high and the propagation delay from low-to-high (t_{pLH}) is the opposite of it. The average propagation delay (t_p) is computed by taking the average of t_{pHL} and t_{pLH}. In this study, the average propagation delay is computed throughout for all the schematics designed.

The dynamic power dissipation of a CMOS inverter largely depends on the switching activity factor. The switching activity is the factor of the frequency at which the transistors in the inverter switch from one logic state to the other logic state. In other CMOS circuits, it also relies on the number of transistors required to implement the logic. In this study, the dynamic power dissipation of the CLA carry generator circuits is reduced by reducing the number of transistors required to

implement the carry-generation logic and by reducing the switching activity.

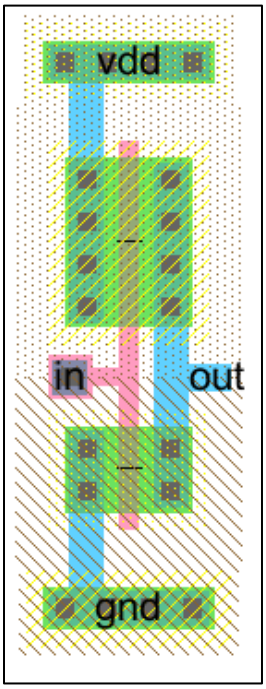


Figure 2.7: Layout of a CMOS inverter

The layout of a CMOS inverter is represented in Figure 2.7. Layout is a representation of an integrated circuit in terms of planar geometrical shapes, which correspond to metal, oxide and semiconductor layers that make up the components of an integrated circuit. In Figure 2.6, for an NMOS design, green color represents n-diffusion layer for both source and drain, pink represents polysilicon wire which is used for representing the gate. When polysilicon layer crosses across the n-diffusion layer it forms an NMOS transistor and when it crosses the p-diffusion layer it forms a PMOS transistor. The blue color in Figure 2.7 represents the metal-1 layer. Metal1 layer is used for connecting the transistors to the power supply and ground rails. The input signal is connected to the polysilicon wire, which acts as a common gate for both PMOS and NMOS transistors. The drains for both PMOS and NMOS transistors are tied together using a metal1 layer and output is measured across it.

2.4.2 Cascaded 4-bit CLA

In order to realize 4-bit CLA logic seen in Figure 1.3, initially each 1-bit CLA is implemented at CMOS transistor level and then they are cascaded together to form a 4-bit cascaded CLA. The sub-circuits for producing generate, propagate, sum and carry generation signals of 1-bit CLA are seen in Figures 2.8, 2.9, 2.10 and 2.11. They are further connected together.

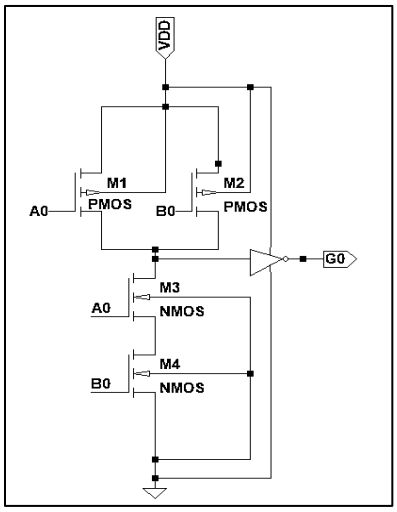


Figure 2.8: Schematic for realizing a generate signal

In Figure 2.8, A_0 and B_0 represent the input signals of the full- output end inverts the output signal coming from the schematic and provides a non-inverted output adder in the stage-1 of 4-bit CLA block-diagram seen in Figure 1.3. The inverter connected at the generate signal G_0. Figure 2.8 realizes the logical equation (2.2).

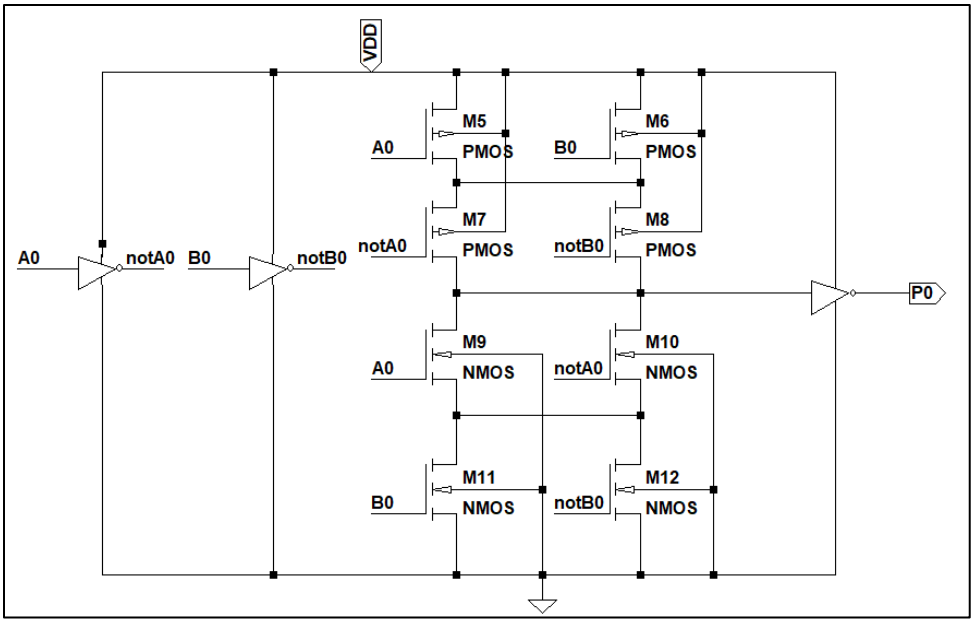


Figure 2.9: Schematic for realizing the propagate signal

Figure 2.9 presents the schematic required to implement the logic of equation (2.3). The inverters for isolation and signal clean up are added at the input and output end of the schematics. The inverters at the beginning of the schematic generate the inverted input signals, which are directly applied to the transistors in the propagate circuitry. This schematic enables the output to propagate signal P_0 when either of the input signals is high.

The sum of each bit of CLA can be logically implemented by

$$S_i = A_i \oplus B_i \oplus C_i \tag{2.12}$$

Substituting equation (2.5) in equation (2.12) and re-writing it we get

$$S_i = P_i \oplus C_i \tag{2.13}$$

The sum circuitry realizing the logic in equation (2.13) is represented in Figure 2.10. Similar to Figure 2.9, this schematic has inverters in the input and at the output end for isolation and signal cleanup. The signal C0 in the summing circuitry represents the input carry signal of the stage-1 of 4-bit CLA in Figure 1.3, and signal P0, is the propagate signal output derived from Figure 2.9. The

output signal S0 here represents the sum of the stage-1 full-adder block in the 4-bit CLA block diagram in Figure 1.1. The inverter at the output end here acts as a capacitive load, to simulate the effect of additional circuitry in an actual implementation.

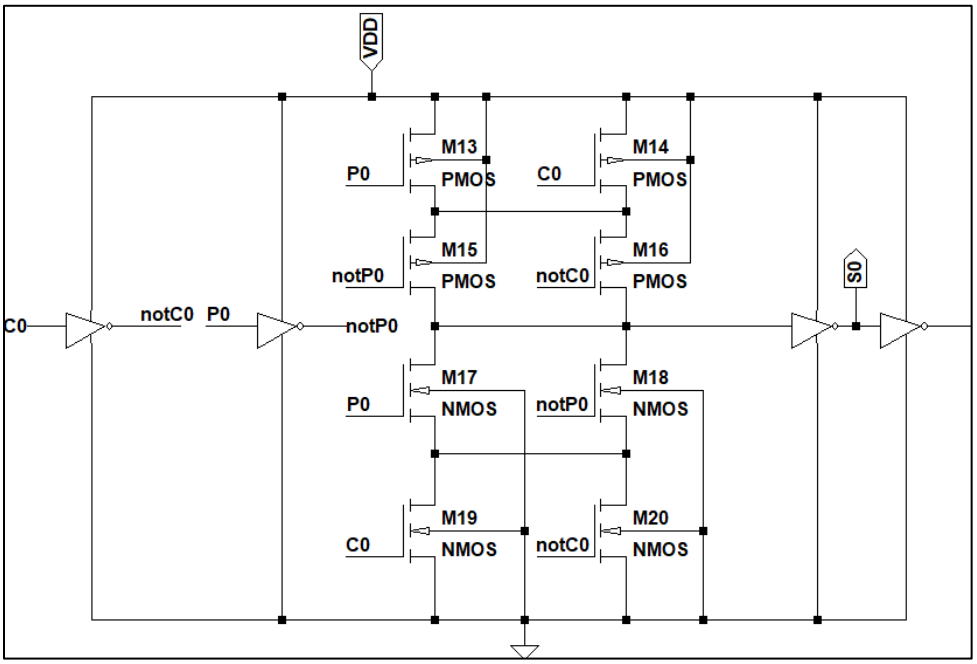


Figure 2.10: Schematic for realizing the sum signal

The output carry signal of the stage-1 in the block diagram of 4-bit CLA is generated and propagated by the schematic in Figure 2.11. Equation (2.4) is realized in Figure 2.11. The input signals G0 is derived from Figure 2.8. The output signal C1 in this schematic denotes the output carry signal of the first stage 1-bit adder. The propagation delay of this output carry signal is measured at port C1 in Figure 2.11 and is 0.43ns. The inverter at the output end adds a capacitive load to the schematic. Combining the circuitry in Figures 2.8 to 2.11 forms a 1-bit CLA.

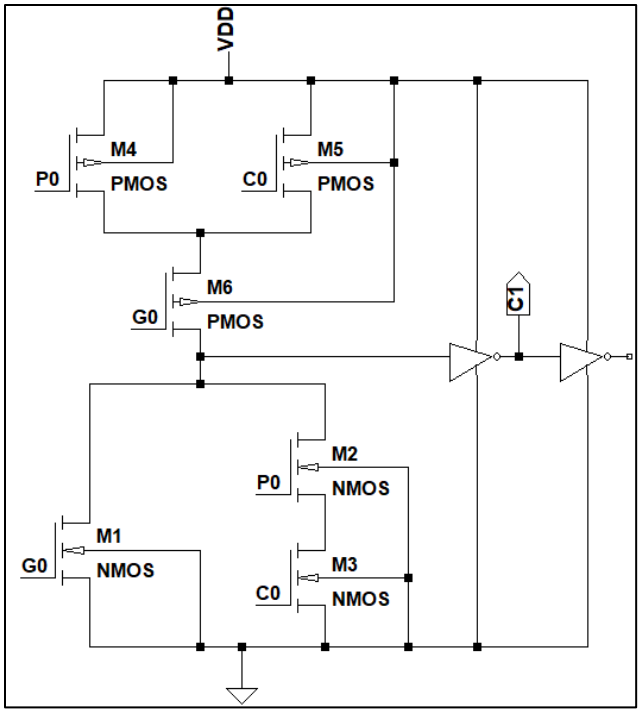


Figure 2.11: Schematic for realizing the carry signal

The schematic of the cascaded 4-bit CLA is designed using LTspice and is seen in Figure 2.12. This schematic cascades four 1-bit adder blocks to form a 4-bit cascaded CLA. This adds two 4-bit binary inputs A3A2A1A0 and B3B2B1B0. Each block diagram in Figure 2.12 represents the combined circuitry of generate, propagate and sum signals from Figures 2.8 to 2.11 for corresponding input bits. The corresponding input binary bits in level0, level1, level 2 and level3 are A0B0, A1B1, A2B2, and A3B3. This combined circuitry of the block diagram is shown in Figure 2.13. Once the input signals are given, it generates the output sum signal S3S2S1S0 and output carry signal C4 along with the intermediate carry signals C1, C2, and C3.

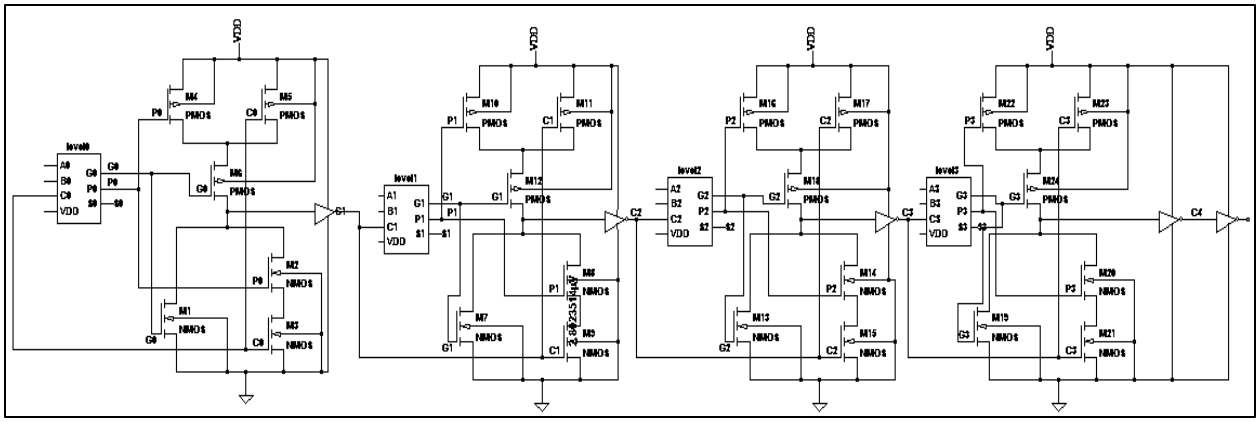


Figure 2.12: Schematic of 4-bit cascaded CLA

The block diagram of each level in Figure 2.12 is shown in Figure 2.13.

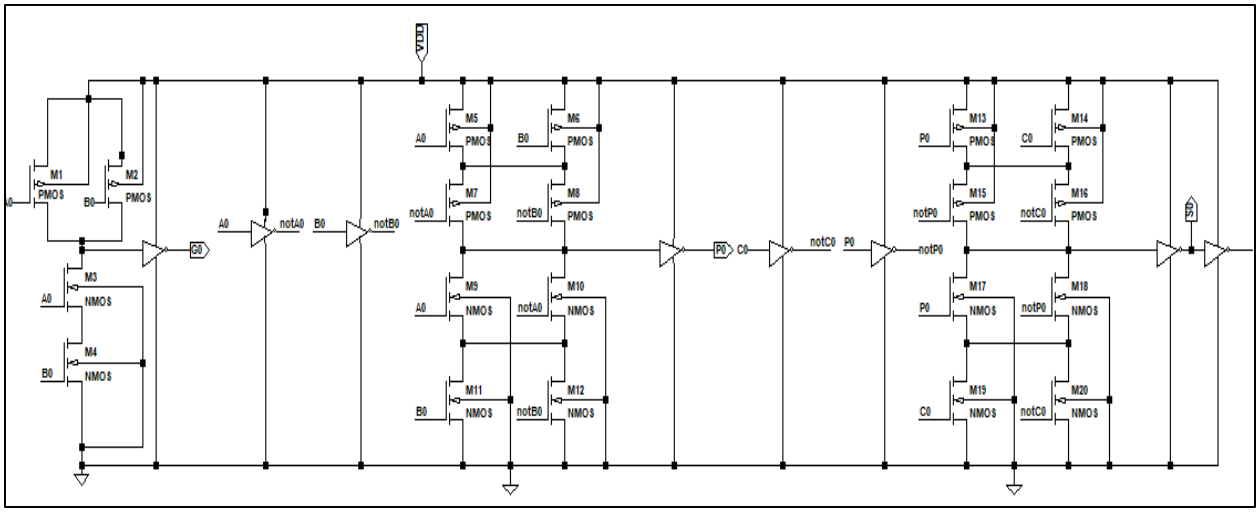


Figure 2.13: Block Diagram for Sum Computation in Cascaded-CLA

In this case, the consecutive carry signals are computed with a carry signal from intermediate stages as inputs. This leads to the delay in the carry-generation and carry-propagation of CLA. The propagation delay across the cascaded 4-bit CLA is computed through transient analysis. The transient analysis of the four-bit cascaded CLA is presented in Figure 2.14. Here V (c0) represents the input carry signal and V (c4) denotes the output carry signal generated from the 4-bit cascaded CLA in Figure 2.12. A piecewise linear function is given as an input carry signal voltage source, V (c0). The measured propagation delay of the output carry signal V (c4) in Figure 2.12 is 0.95ns.

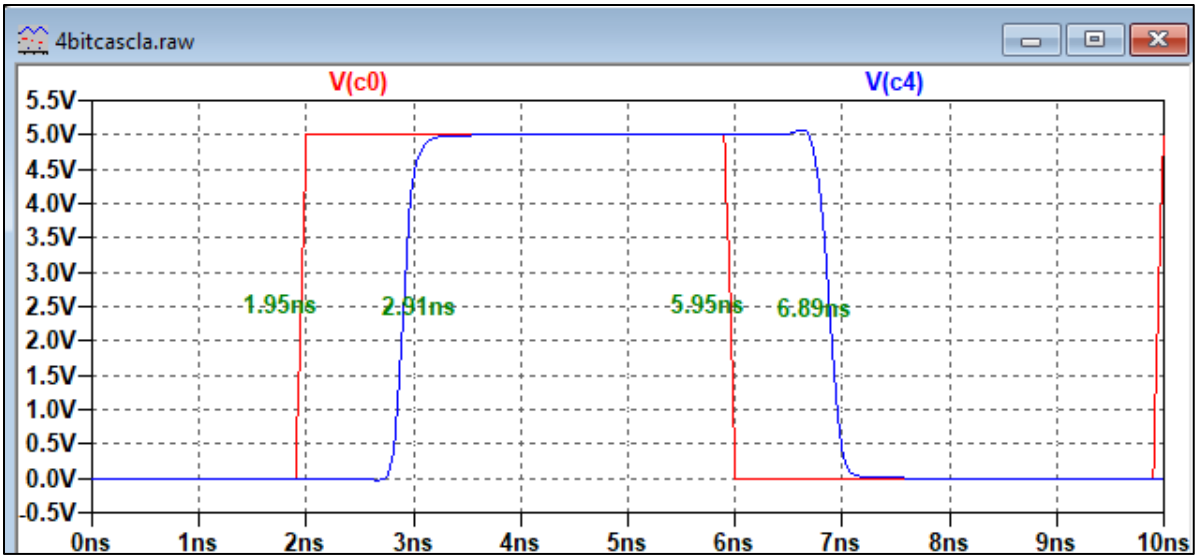


Figure 2.14: Transient Analysis of 4-bit cascaded CLA.

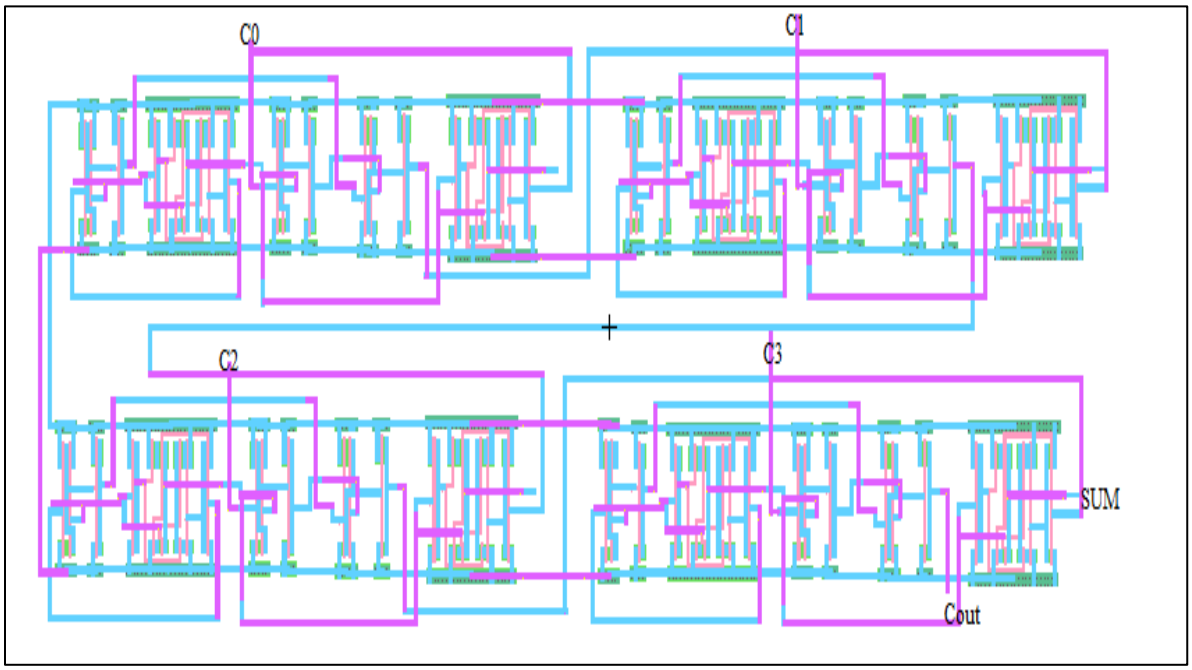


Figure 2.15: Layout of 4-bit cascaded CLA [5]

The layout of the 4-bit cascaded CLA schematic represented in Figure 2.12 is presented in Figure 2.15 [5]. The component color representations in Figure 2.15 are similar to the color representations used in CMOS inverter layout as seen in Figure 2.7.

It is apparent from the above 4-bit cascaded structure of CLA that, it takes time for the input to

carry signal ‘C0’ to propagate all the way to the output carry signal ‘Cout’, propagating through all the level of the CLA represented in Figure 2.12. This increases the propagation delay of the carry signal to a large extent. Also, the cascaded 4-bit CLA requires 104 CMOS transistors to implement the logic, which is high. This leads to an increase in the layout size, further increasing the chip cost. As the transistor count of the 4-bit cascaded CLA is high, the switching activity of the adder is higher. This increases the dynamic power dissipation of the CLA. Continuous research has been going on to augment the performance of CLA by reducing the propagation delay of the carry signal, decreasing the layout area and dynamic power dissipation

2.5 Previous Work

In this section, the state of art of the CLA design with respect to the above-mentioned issues are described, along with elucidating, how this dissertation research gives better results than the previous work done so far in those parameters.

2.5.1 Replacing AND gates with NAND gates in CLA architecture

In a structured approach for optimizing a 4-bit CLA AND gates in Figure 2.4 are replaced by NAND gates in [10] to reduce the propagation delay, dynamic power dissipation, and power-delay product. Power-delay product is also known as the switching energy. It largely depends on the switching activity factor in the circuit, which is directly related to the number of transistors as explained in chapter 1. This replacement of the AND with NAND gates reduce the number of transistors required to implement the 4-bit CLA logic. Generally, a CMOS implementation of the CLA provides inverted output signals, since they are implemented with CMOS inverter structure as a base. An inverter is used at the output end to get a non-inverted output signal. Replacing AND gates with NAND gates eliminates the need of using inverters at the end of the AND gates used throughout the logical circuit diagram of CLA. This replacement decreases the transistor count

required to implement the CLA compared to the conventional implementation of CLA shown in Figure 2.4 and which further reduced the layout size. This reduction in the number of transistors reduces the switching activity and the dynamic power dissipation of CLA in this case. In [10] all the components except the propagate (P) and sum (S) outputs, which are implemented using XOR gates are replaced by NAND gates. In this case, the delay caused by the carry-generation and propagation signals in the cascaded structure used for the implementation is not focused. It focuses on reducing the delay through the full adder circuits of CLA. This is the major drawback of this research in [10] since the delay caused by XOR gates used and the propagation delay of the carry signal in CLA is not reduced by the greater extent. As this design is extended to 8-bit, the factor of improvement in the design issues wouldn't be significant.

Table 2.1: Design issues comparison for [10]

Method Adopted	Propagation Delay	Layout Size	Dynamic power dissipation
Replacing AND with NAND gates	----	ok	ok

2.5.2 Realizing CLA using NAND gates

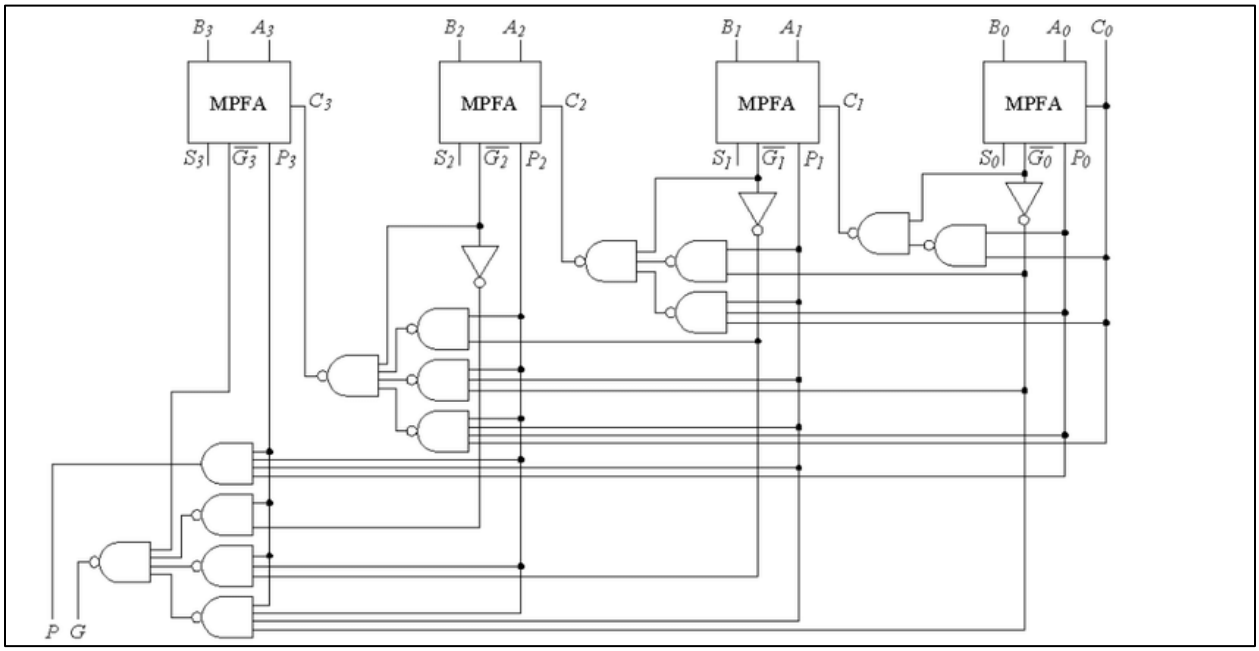


Figure 2.16: Implementation of CLA using NAND gates [11]

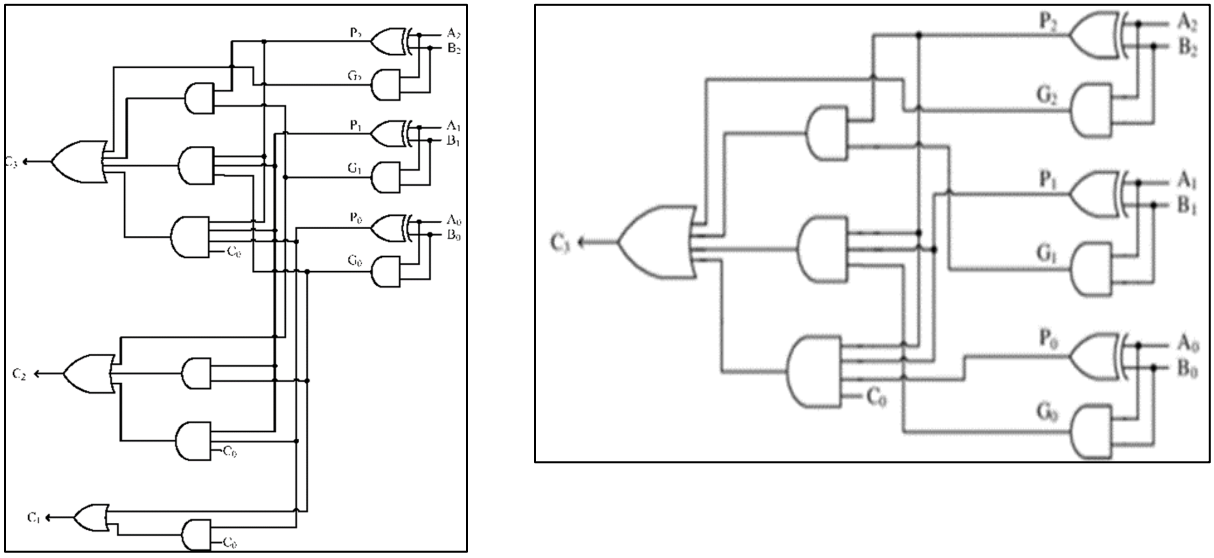
In the work discussed in the [11], all the AND, OR, XOR gates used in Figure 2.4 are swapped with NAND gates as shown in Figure 2.16. In Figure 2.16, MPFA stands for Modified Proposed Full-Adder structure, which is a basic arithmetic adder circuit implemented with NAND gates. NAND is a universal logic gate, as any digital gate logic can be implemented by a NAND gate. As, stated in section 2.4.2, replacing the AND gates in Figure 2.4 with NAND gates reduces the transistor count, similarly swapping OR and XOR gates with NAND gates also reduces the transistor count. This reduces the layout size, better than the research discussed in section 2.4.1. As the transistor count reduces, the switching activity reduces and dynamic power dissipation decreases. Also, the 8-bit CLA implementation in [11] is done by using recursive circuits. Here two 4-bit CLA's are cascaded to design an 8-bit CLA. The author in [11] mentions that using recursive circuits by cascading reduces the propagation delay of the CLA carry signal. But clear evidence showing the drop in the propagation delay is not provided. It is obvious that using

recursive circuits the layout size increases and dynamic power dissipation increases. This is the drawback of using recursive circuits.

Table 2.2: Design issues comparison for [11] [12]

Method Adopted	Propagation Delay	Layout Size	Dynamic power dissipation
Replacing AND, OR and XOR with NAND gates	----	ok	--

2.5.3 Block-based carry in CLA



(a) (b)

Figure 2.17: (a) Carry generator of 3-bit cascaded CLA; (b) Carry generator circuit of sectional carry based CLA [13]

The block-based carry in CLA [13] introduces the section-carry based CLA topology as a high-speed alternative to the cascaded CLA structure. The sum computation circuitry here is similar to the cascaded CLA structure. In [13], the cascaded structure CLA, discussed in Figure 2.12 is referred to as conventional CLA. In the sectional- carry based CLA, the look-ahead carry signal

computation is done in the 3-bit block architecture. In this case, the computation of intermediate carry signals for 1-bit and 2-bit are avoided. It generates the carry signal for 3-bit in a single circuit, without cascading intermediate carry signals. Figure 2.17 shows the architectural difference between the carry signals generator logic implementation in cascaded-3 bit CLA and in 3-bit sectional-carry based CLA. For instance, in section-based CLA an 8-bit carry look-ahead logic is implemented by cascading two 3-bit carry blocks and a 2-bit carry block.

This kind of block representation of the carry-generator circuit of CLA paves the way for the reduction of the propagation delay and improvement in the throughput of the circuit. This study acts as a good initiative for this kind of design topologies for carry look-ahead computation in CLA. The drawback of this section-based CLA topology is as, the number of bits to be added increases, the design metrics like layout area and dynamic power dissipation also increase. These are the metrics used to evaluate the design. Since, the cascaded-structures are still used here, for higher bit additions 3-bit carry generation blocks are cascaded, which requires more layout area. The cost of the chip using this adder circuitry for its applications increases. As, the layout area increases, the dynamic power dissipation also increases.

Table 2.3: Design issues in block-based design [13]

Method Adopted	Propagation Delay	Layout Size	Dynamic power dissipation
Block-based carry signal computation	Better	---	----

2.5.4 Reference CLA Design Representations

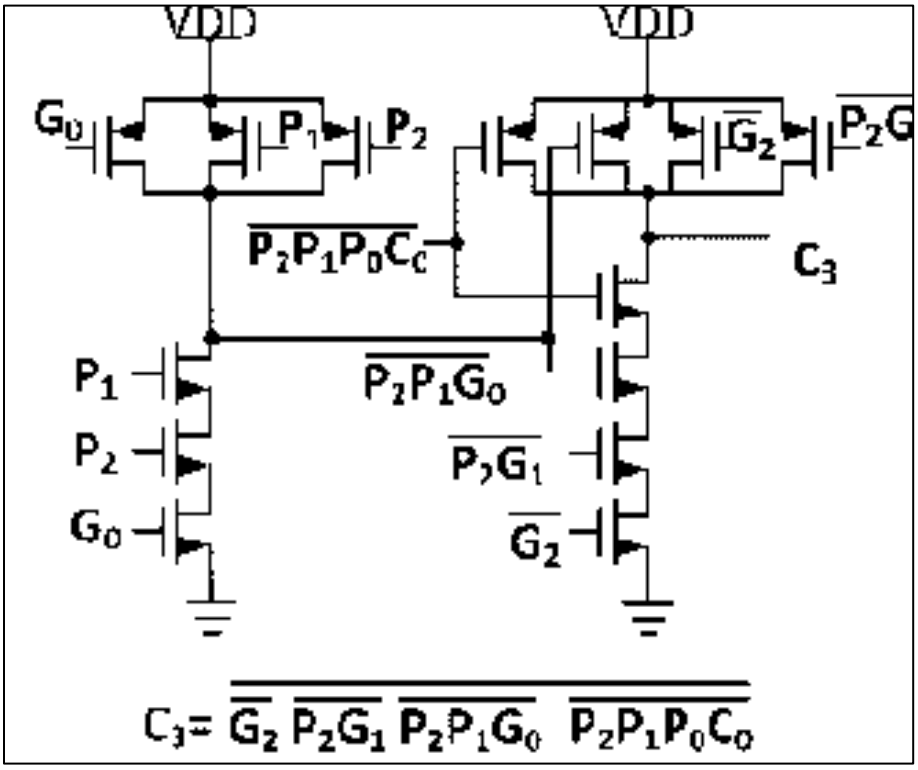


Figure 2.18: Representation of a 3-bit CLA [14]

Based on the different architectures of CLA discussed accordingly, it is apparent that the performance of the CLA largely depends on its design metrics. In a novel implementation of 4-bit CLA [14], a new design topology for carry-generation is introduced. The sum circuitry in CLA requires full-adders for each bit addition. This study focuses on accelerating the carry generation logic in CLA. The 3-bit carry generator design topology introduced in [14] is shown in Figure 2.18. Though it is mentioned as a dynamic CMOS logic structure in [14], it is a static CMOS structure, as one of the input signals acts as a clock signal here. In dynamic logic structures, a clock signal is used to enable the operation of CLA and disable the operation of CLA. The carry signal generation design topology introduced as seen in Figure 2.18 doesn't have a specific clock signal as an input, and it considers one of the input signals as the clock signal. It is a static CMOS structure design. As static CLA structures optimization is described throughout this dissertation, the above-

mentioned topology can be considered for reference. In Figure 2.16, though the logic structure of carry-generator of CLA looks simplified, the logic circuitry to realize the input signals G_2', (P_2G_1)', $(P_2P_1G_0)$' and $(P_2P_1P_0\ G_0)$' is not presented.

During the literature review of this dissertation, the design topology used in Figure 2.16 is considered and the 2-bit carry generation logic from [14], represented by the equation (2.14)

$$C_2 = \overline{\left[\overline{G_1}(\overline{\overline{P_1} + \overline{G_0}})\right]} + \overline{\left[\overline{P_1} + \overline{P_0}C_0\right]} \tag{2.14}$$

is taken and a 2-bit CLA reference carry generation circuit is developed. This schematic is represented in Figure 2.17. It requires 19 transistors to realize 2-bit carry generation logic using the design configuration used in [14] and all the input signals are generated individually, which needs more transistors.

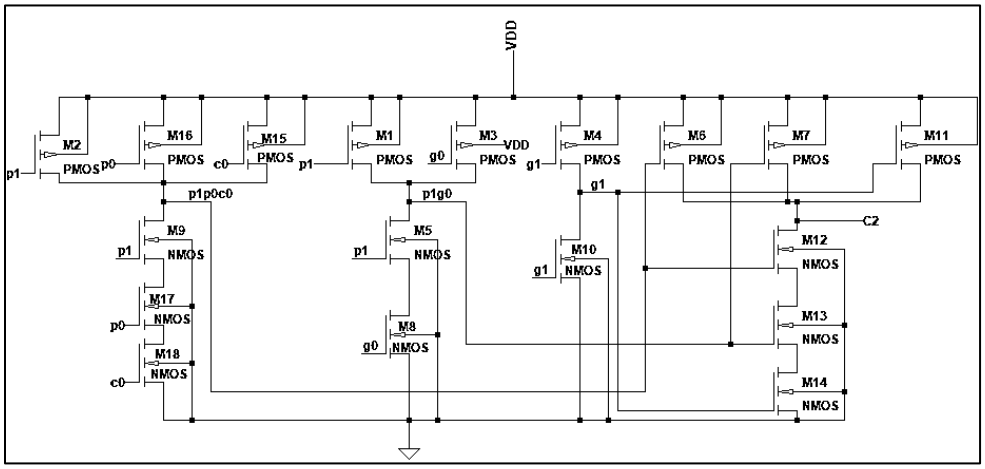


Figure 2.19: Representation of a 2-bit CLA reference carry generation circuit

This design topology requires a higher transistor count to implement the carry generation logic. Even the 2-bit cascaded structure does not require 19 transistors to implement a 2-bit design. In this case, implementing a 4-bit CLA carry generation circuit using the design topology in Figure 2.19, would be more complex, in terms of the transistor count, and dynamic power dissipation. As the transistor count increases the layout size of the device increases and the dynamic power

dissipation increases. This is the drawback of this design topology.

Table 2.4: Design issues comparison for design topology in [14]

Method Adopted	Propagation Delay	Layout Size	Dynamic power dissipation
Design topology in [14]	ok	---	----

A static CMOS implementation of carry generation and propagation circuit in [15] is represented in Figure 2.20. Figure 2.20 represents a 2-bit CLA carry generator and propagator circuit

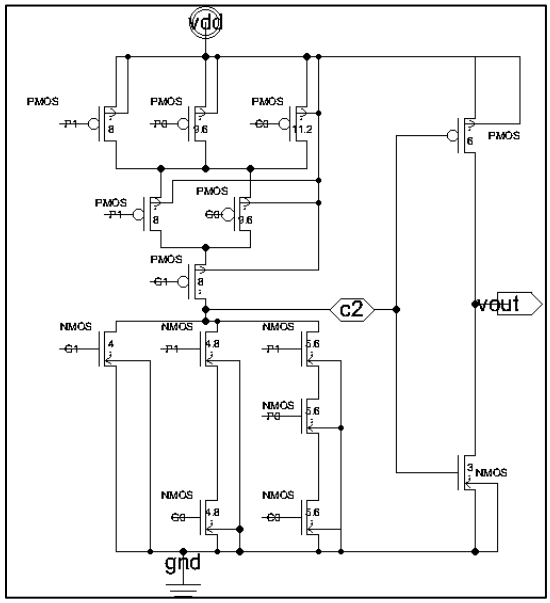


Figure 2.20: Representation of a reference 2-bit CLA carry generator and propagator

The 2-bit CMOS CLA carry generator circuit represented in Figure 2.20 needs 12 transistors to implement the logic excluding the inverter at the output end. In the aspect of the transistor count, this design is better than the design topology described in Figure 2.19. As the transistor count reduces, the switching activity of the design reduces and hence the dynamic power dissipation is low. The propagation delay value of this 2-bit CLA carry generation design measured through

simulation is 0.23ns, which is better than the design topology in Figure 2.19. The sum circuitry in the CLA is the direct implementation using full-adder circuits. The major complexity lies in the carry look-ahead logic block,

Table 2.5: Design issues comparison for design topology in [15]

Method Adopted	Propagation Delay	Layout Size	Dynamic power dissipation
Design topology in [15]	ok	ok	ok

The current state of the art in the carry based adder designs using any technology focuses on the optimization of their design metrics. As all three design metrics focused in this dissertation are better compared to the other previous works discussed in this section, this design is considered as the reference topology throughout this dissertation. It is used to evaluate the performance of the proposed stand-alone CLA carry generation circuits described in the further sections.

2.6 CMOS Implementation of 1-bit, 4-bit, 8-bit and 16-bit reference stand-alone circuits

Considering the design topology in Figure 2.20 [15] as a base, 1, 4, 8 and 16-bit reference stand-alone CLA carry generation circuits are developed. Stand-alone carry generation circuits here represent a single architecture circuit, with both input and output carry signals connected in the same circuit. These circuits do not generate intermediate carry signals. 2-bit reference stand-alone carry-generation circuit is seen in Figure 2.20. Figure 2.21 to Figure 2.25 represent the schematics of reference stand-alone carry generation circuits for 1, 4, 8 and 16-bits accordingly.

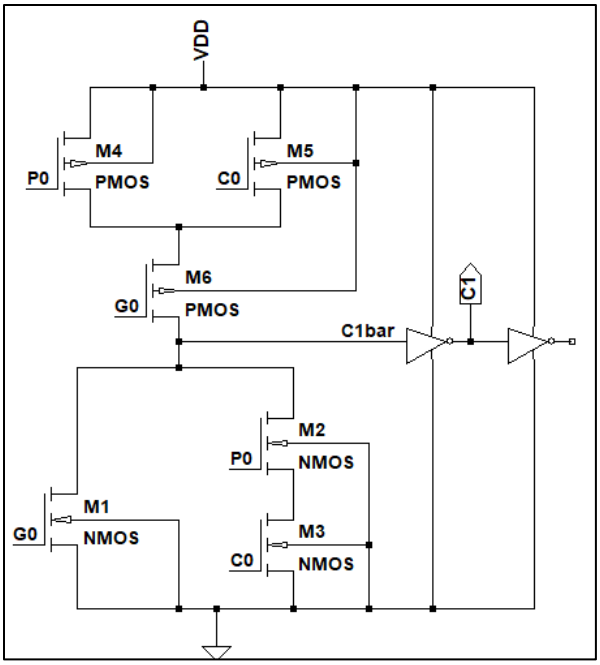


Figure 2.21: 1-bit reference CLA stand-alone carry generation circuit

In Figure 2.21, G0, P0, and C0 represent the input carry generate, carry propagate and the input carry signals. The output carry signal generated in this circuit is C1. The propagation delay is measured across the port C1. The inverter at the output end acts as a capacitive load for the actual implementation of the circuit. It requires 6 transistors to implement the one bit carry generation logic presented by equation (2.8).

Figure 2.22 presents the 4-bit reference CLA stand-alone carry generation circuit. It realizes the logic presented in equation (2.11). Nine input signals G0, G1, G2, G3, P0, P1, P2, P3 and C0 are required to implement this design. It requires 30 transistors to implement the carry generation logic.

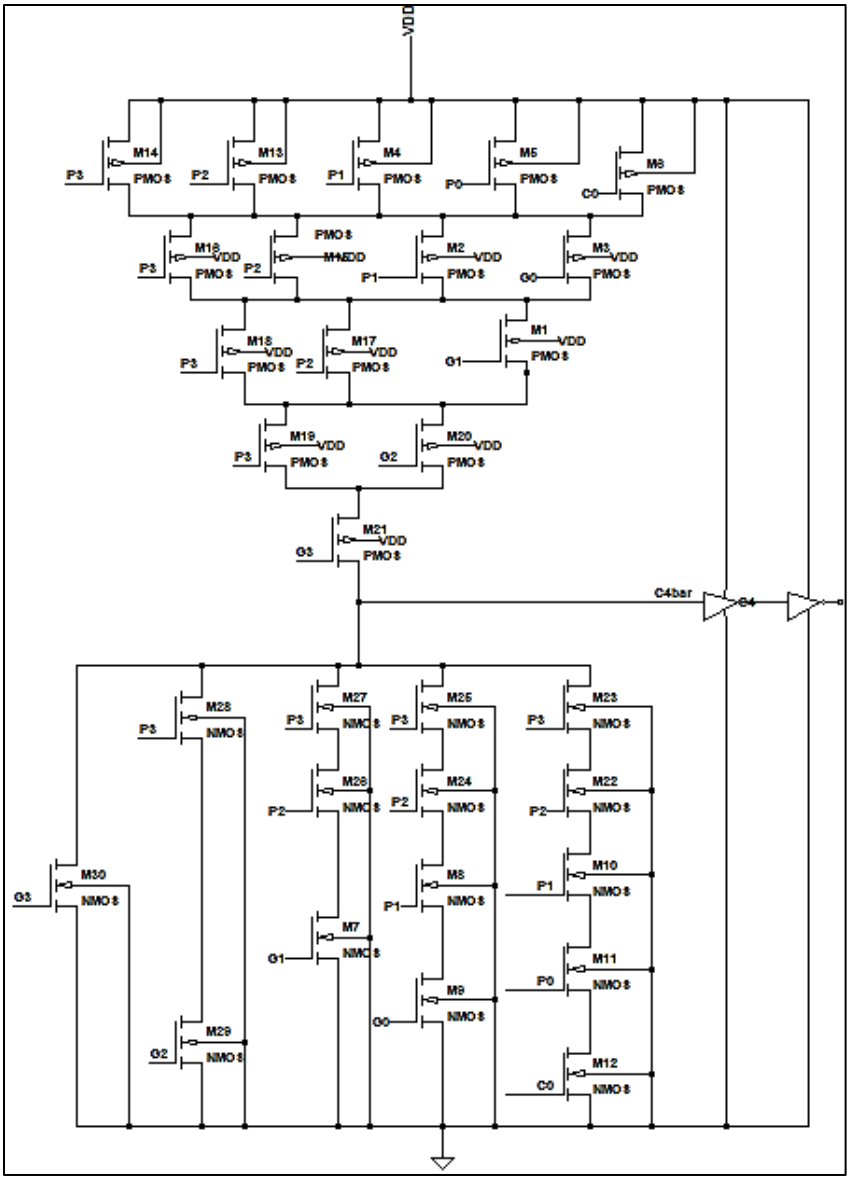


Figure 2.22: 4-bit CLA reference stand-alone carry generation circuit

Figure 2.23 represents 8-bit CLA reference stand-alone carry generation circuit. It requires 90 transistors to implement the carry generation logic. The inputs signals, in this case, are doubled, as of 4-bit representation seen in Figure 2.22. It requires to generate inputs from G7 down to G0, propagate inputs from P7 down to P0 and an input carry signal C0. All the output signals in Figure 2.21 to Figure 2.25 correspondingly, drive an external inverter load.

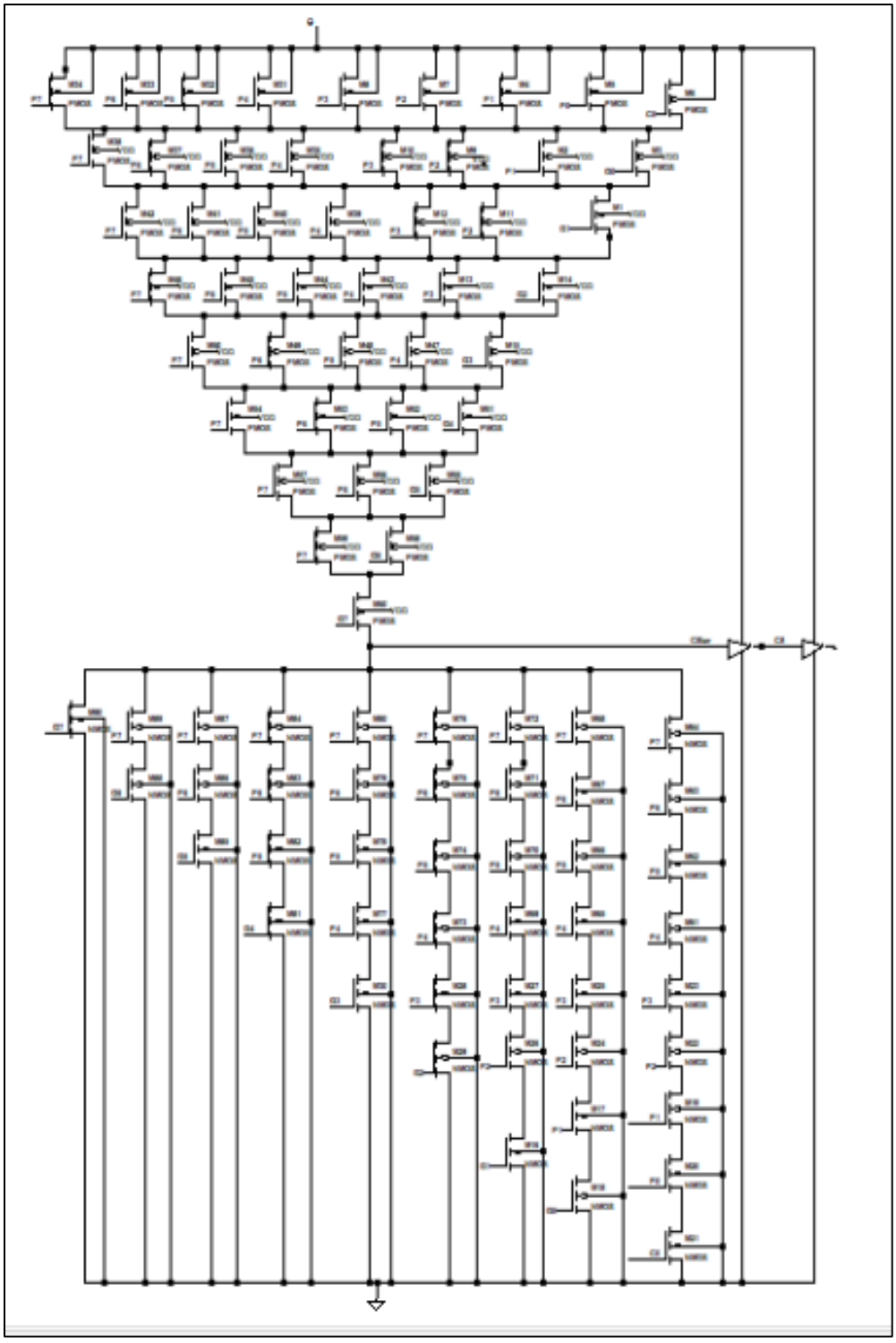


Figure 2.23: 8-bit CLA reference stand-alone carry generation circuit

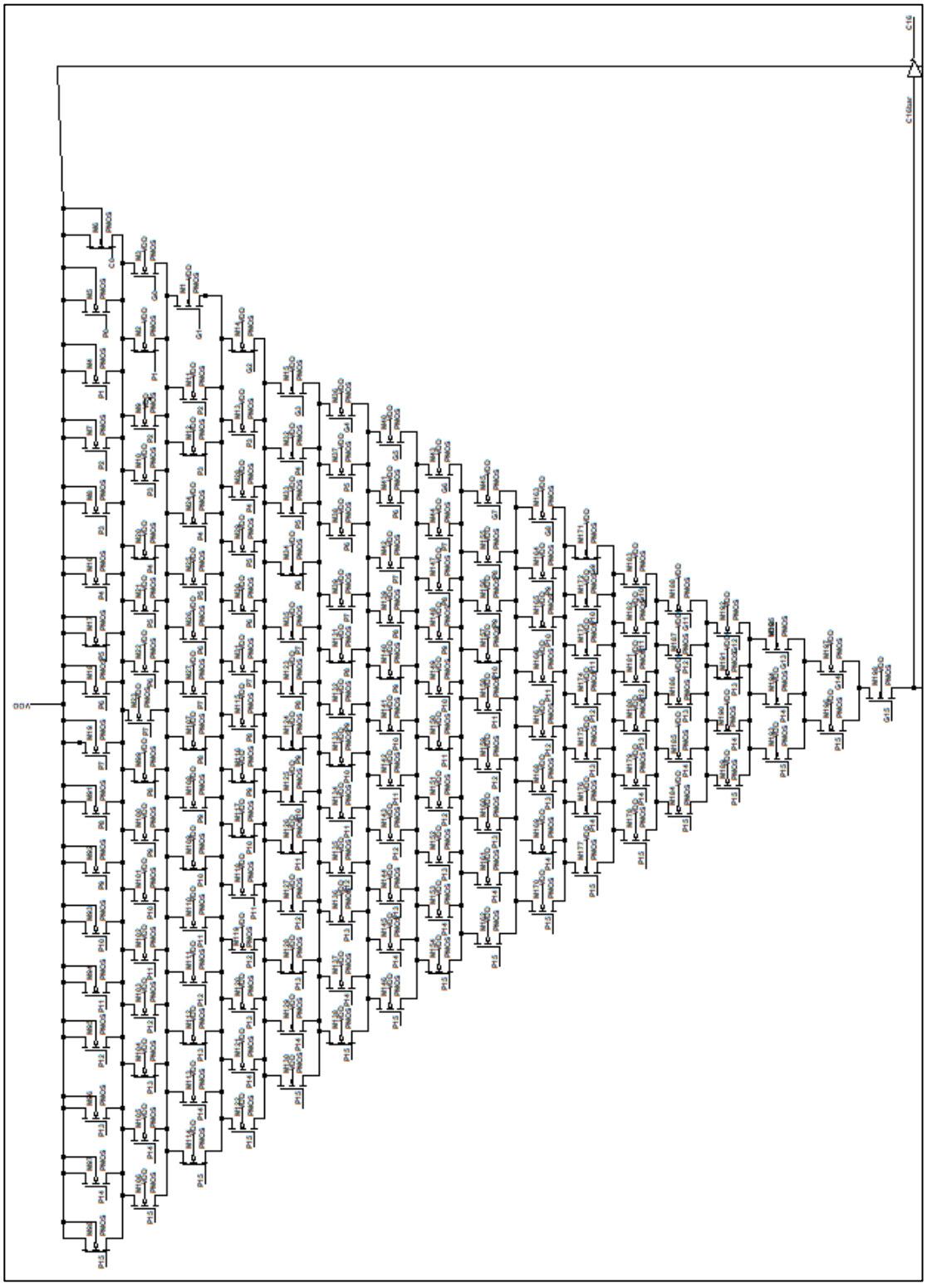


Figure 2.24: 16-bit pull-up block of CLA reference stand-alone carry generation circuit

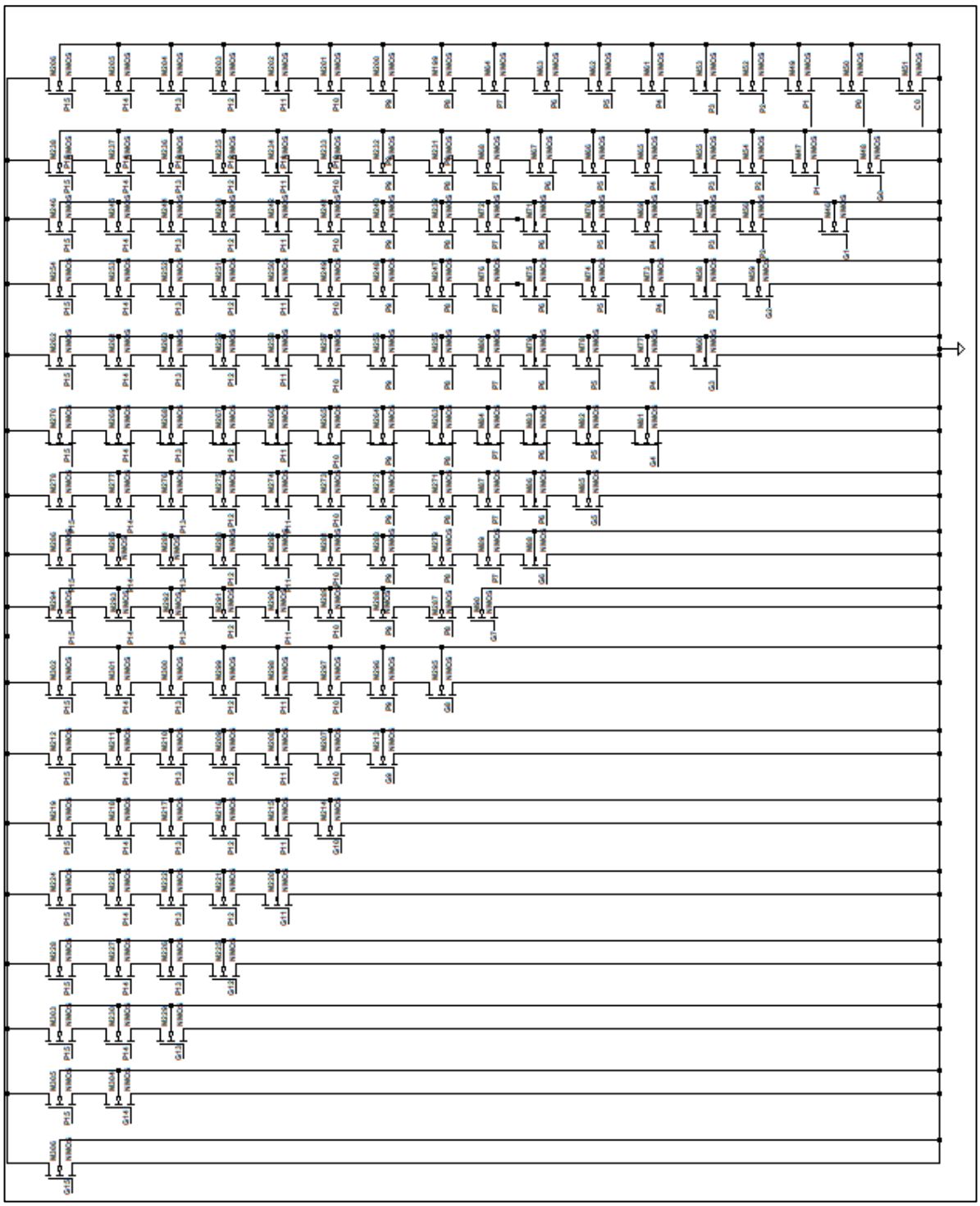


Figure 2.25: 16-bit pull-down block of CLA reference stand-alone carry generation circuit

The 16-bit CLA reference stand-alone carry generation circuit requires 306 transistors, each of 103 transistors in its pull-up and pull-down blocks. Figure 2.22 and Figure 2.23 represent the pull-up and pull-down blocks accordingly.

The sizing of the reference stand-alone carry generation configurations is done based on their placement. The transistors driving the output have higher widths compared to the other transistors in the circuit. The transistor count and the propagation delay of all the schematics of reference stand-alone carry generation circuits are tabulated in Table 2.1.

Table 2.6: Reference Stand-alone Carry Generation Circuits Preliminary Data

Reference Stand-alone Carry Generation Circuits	No. of Transistors	Propagation Delay (ns) of the schematic at ON Semi C5 technology (500nm)
1-bit	6	0.15
2-bit	12	0.23
4-bit	30	0.81
8-bit	90	2.74
16-bit	306	10.5

It could be observed from Table 2.1, that as the number of bits to be added, the transistor count and the propagation delay of the circuit increases exponentially. This would also lead to higher dynamic power dissipation. This might diminish the performance of the circuit for the higher bit carry generation circuits, beyond 4-bit. Hence, proposed stand-alone carry generation and propagation circuits are introduced in the later chapters of this dissertation, to optimize the carry generation circuits of CLA by resolving the design issues, propagation delay, layout size, and

dynamic power dissipation to a large extent. Proposed stand-alone carry generator and propagator circuits described further in chapter 3 are supposed to eliminate the cascading structure problem discussed in section 2.3.

2.7 R-C Delay Model:

The delay across the CMOS circuits is modeled theoretically in various ways [17]. One among them is the distributed RC delay. The analytical RC delay model provides the estimation of the timing analysis of the circuit.

2.7.1 R-C Delay of a CMOS NOR logic gate.

In R-C delay model, all the transistors of the CMOS circuit, are replaced by a switch in series with a resistor. Each transistor comprises a gate capacitance and source and drains diffusion capacitances. The realization of the CMOS circuit using R-C delay model can be demonstrated in this sub-section with an example.

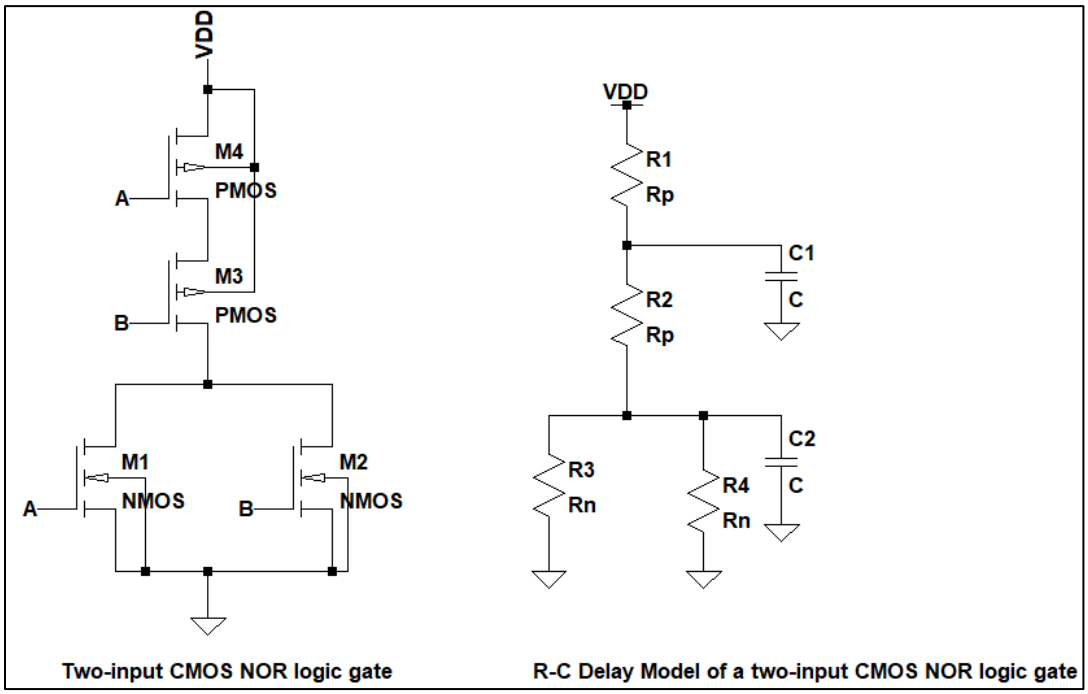


Figure 2.26: R-C Delay model of a two-input CMOS NOR logic gate

Figure 2.26 represents the two-input CMOS NOR logic gate and its equivalent RC delay model. It can be noticed that each transistor is replaced by a resistor with the corresponding diffusion capacitance across each node. The resistance, capacitance and propagation delay calculations in the delay model are discussed in the later sub- sections. The R-C delay models of a digital circuit are also known as R-C transmission networks.

2.7.2 Resistance and Capacitance Computations in the RC delay model

Normally, according to the semiconductor physics, the resistance of the transistor relies inversely on the width of the transistor, whereas the diffusion capacitance varies directly with the width of the transistor. During the computations of the resistance and capacitance of the RC network structure, the following assumptions are made:

- PMOS transistor with the minimum length and width in the circuit is defined as a Unit PMOS transistor.
- NMOS transistor with the minimum length and width in the circuit is defined as a Unit NMOS transistor. The resistance of the Unit PMOS transistor is known as the PMOS effective switching resistance (R_p) and likewise, the resistance of the Unit NMOS transistor is known as the NMOS effective switching resistance (R_n).

The mathematical equations for calculating the effective switching resistance and effective switching capacitance are seen in the equations (2.15) and (2.16)

$$R_n{}' = \left[\frac{Vdd}{(K_{pn}/2)\times(W_n/L_n)\times(Vdd-V_{thn})^2}\right] \tag{2.15}$$

$$R_p{}' = \left[\frac{Vdd}{(K_{pp}/2)\times(W_p/L_p)\times(Vdd-|V_{thp}|)^2}\right] \tag{2.16}$$

Here, K_{pn} and K_{pp}denote the NMOS and PMOS trans-conductance parameters and are defined by$K_{pn} = \mu_n C_{ox}$, $K_{pp} = \mu_p C_{ox}$ where $\mu_n$and $\mu_p$represent the mobility of electrons in NMOS and PMOS transistors respectively. W_n/L_n and $W_p/L_p$are the aspect ratios of the Unit NMOS and Unit PMOS transistors.$V_{thn}$and V_{thp} are the threshold parameters of the NMOS and PMOS transistors. The resistance across each PMOS and NMOS transistors are computed using equations (2.17) and (2.18)

$$R_n = R_n' \times (L_n/W_n) \tag{2.17}$$

$$R_p = R_p' \times (L_p/W_p) \tag{218}$$

And the diffusion capacitance of NMOS and PMOS transistor are calculated using below equations

$$C_{oxn} = C_{ox} \times scale \times W_n \times L_n \tag{2.19}$$

$$C_{oxp} = C_{ox} \times scale \times W_p \times L_p \tag{2.20}$$

For ON Semi C5 process technology node used to design all the delay models, all the necessary parameter values to compute the delay are tabulated in Table 3.3.

Table 3 List of parameters

Parameter	Value
ε_{ox}	34.5aF/μm
t_{ox}	0.14μm
C_{ox}	2.48fF/μm^2
R_n	22k
R_p	38.5k

2.7.3 Elmore Delay

Elmore delay is a widely used model to compute the propagation delay of the signals in the digital circuits. It provides the approximation of the delay through the R-C transmission networks. The propagation delay time from high-to-low transition and low to high-transition of the R-C delay model according to Elmore model is computed using the formula shown in below equations:

$$t_{plh} = 0.7 \times \sum(Ri. \times Ci) \tag{2.21}$$

$$t_{phl} = 0.7 \times \sum(Ri. \times Ci) \tag{2.22}$$

In the above equation the product of $(Ri. \times Ci)$ denotes time constants across each node 'i' in the pull-up and pull-down networks of CMOS circuits. The time constant across the node in the circuits is computed by the product of the resistance and capacitance across the node. The average propagation delay of the CMOS circuit using Elmore model is computed using equation 2.8.

$$t_{delay} = 0.35 t_p = 0.35 \times (\tau_{plh} + \tau_{phl}) \tag{2.23}$$

Chapter 3. Proposed Stand-alone CLA Carry Generation Circuits

3.1 Research approach

This chapter illustrates the research procedure and the historical data that led to the fulfillment of the dissertation goal.

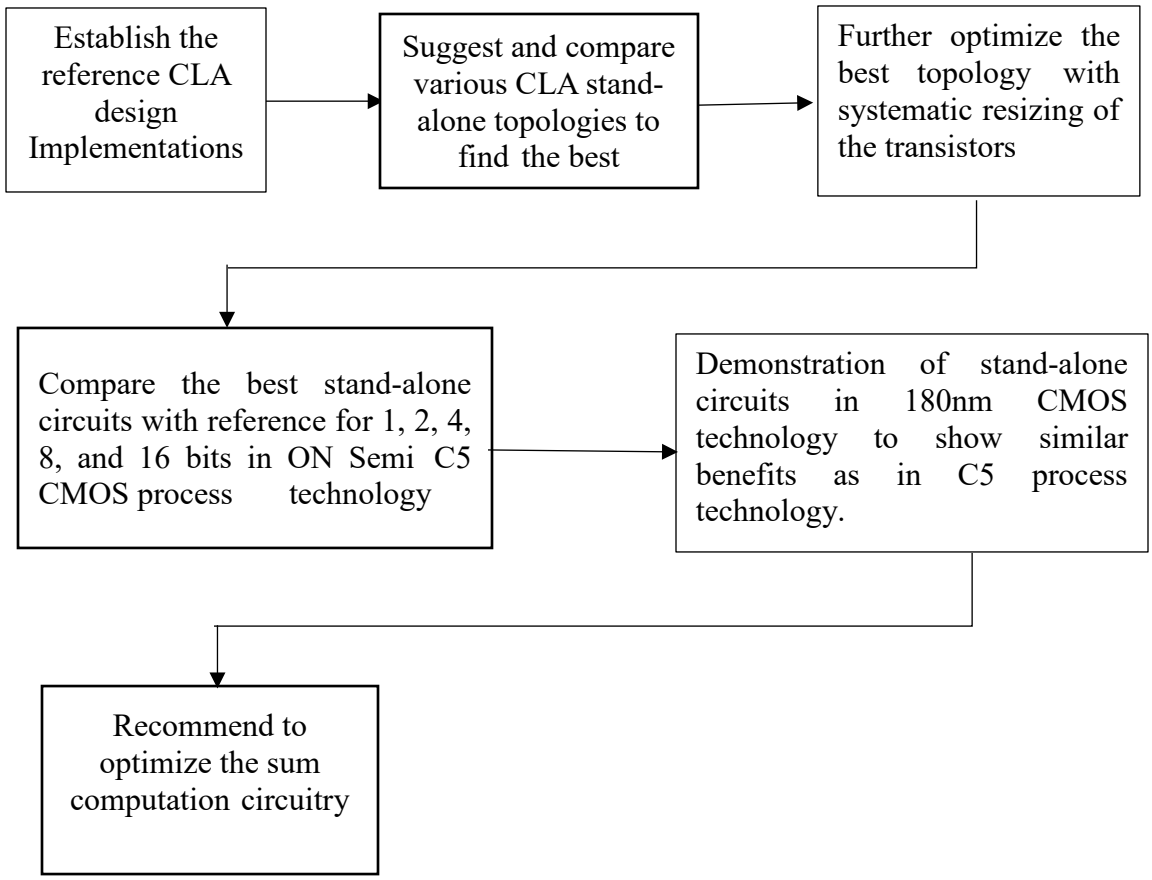


Figure 3.1: Flow-chart of the research procedure.

Figure 3.1 represents the flow-chart of the research procedure followed in this dissertation. Initially, during the literature research the reference CLA design implementations are established using CMOS technology. Then, the four different stand-alone design topologies for various CLA carry-chains are introduced. They implemented using CMOS ON Semi C5 process technology node and their performance is compared in terms of the propagation delay of the input carry signal to the output carry node. Based on their performance one of the proposed stand-alone topologies

is considered to be the best. This topology is further optimized by systematic resizing of the transistors in the schematic and extended for 4, 8 and 16-bits. Further, they are compared with reference CLA design implementations in terms of the transistor count and power dissipation analytically and practically through simulations. The proposed designs are re-implemented using 180nm CMOS technology node to show the similar advantages as of C5 process technology. Finally, the future research to optimize the proposed stand-alone CLA designs is recommended.

3.2 Proposed Stand-alone Design Topologies

In digital design, a simple Boolean logical equation can be represented in infinite ways without modifying its logic. Similarly, the carry generation logic stated in equation (2.1) can be presented in many fashions. In this study, taking CMOS inverter design as a base, initially, four different stand-alone design configurations have been developed and proposed to realize the 2-bit CLA carry generation logic noted in equation (2.5). They are presented from Fig. 3.4, to Fig.3.7 sequentially. Considering, the design topologies both in the pull-up and pull-down structures, the designs are named using alphabets ‘A’ and ‘V’. Figure 3.2 and 3.3 represent the PMOS blocks which represent the alphabet ‘A’ and ‘V’ respectively in shape. Similarly, the NMOS blocks are also named.

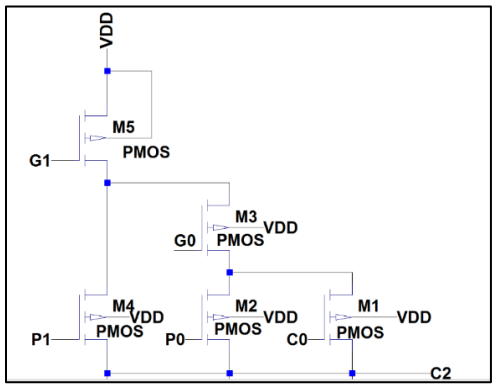


Figure 3.2: PMOS block, type ‘A’ [2]

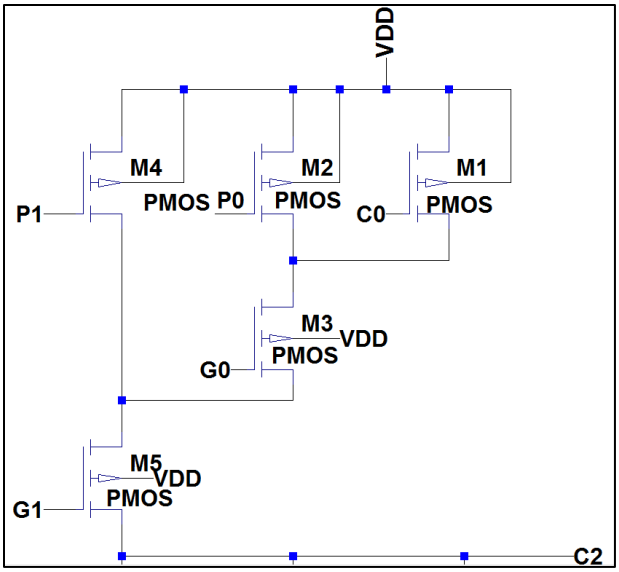


Figure 3.3**:** PMOS block, type 'V' [2]

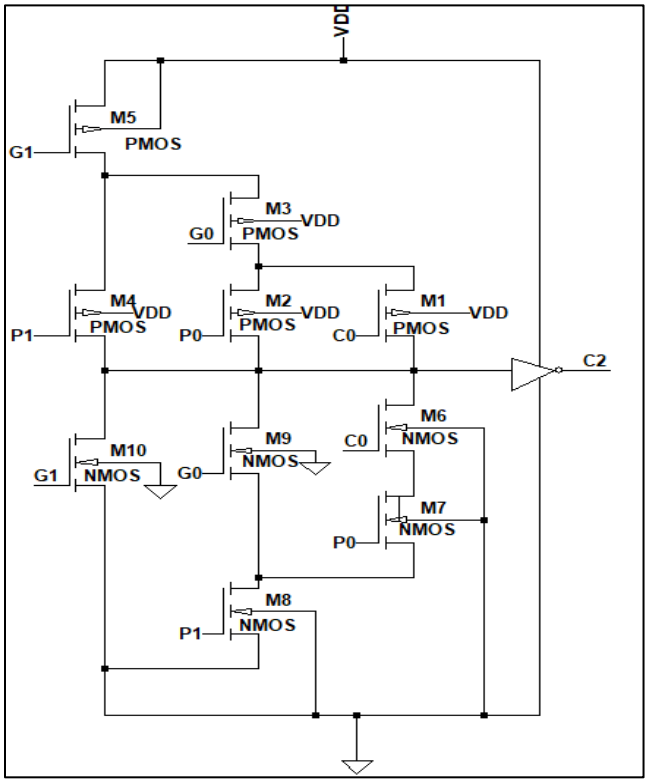


Figure 3.4: Stand-alone 2-bit 'A_pV_n' CLA carry-generation Circuit

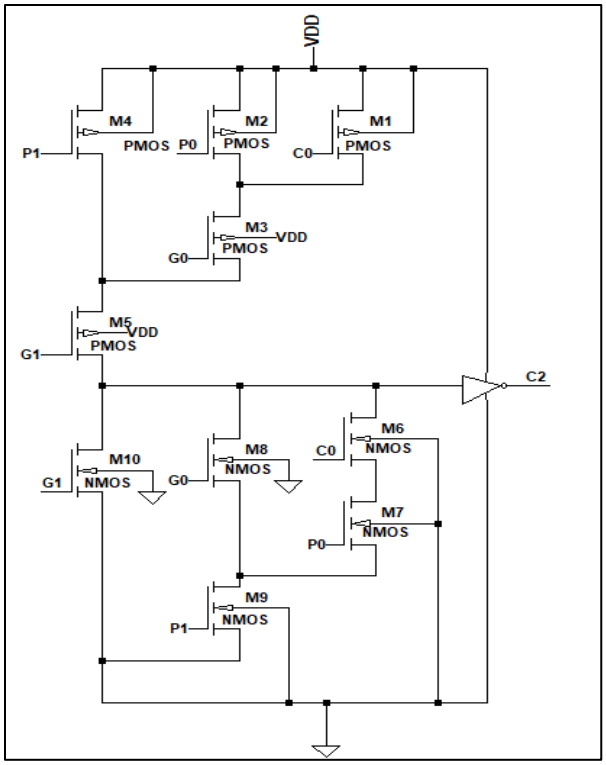


Figure 3.5: Stand-alone 2-bit 'V_pV_n' CLA carry-generation Circuit

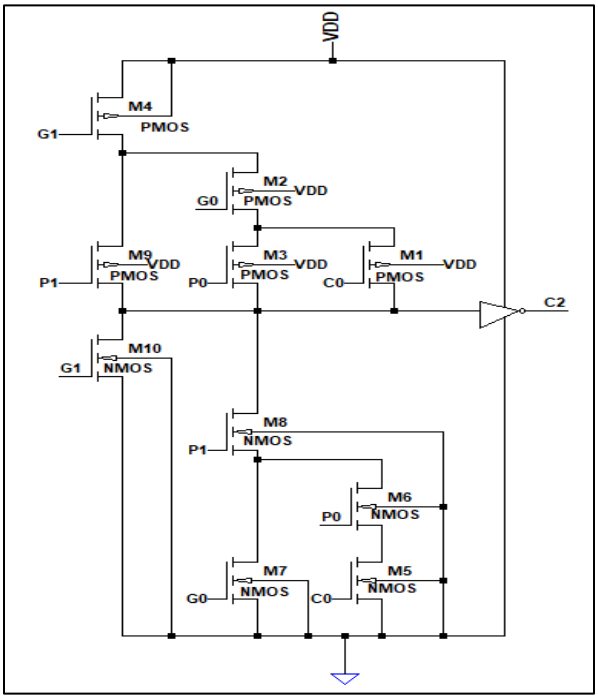


Figure 3.6: Stand-alone 2-bit 'A_pA_n' CLA carry-generation Circuit

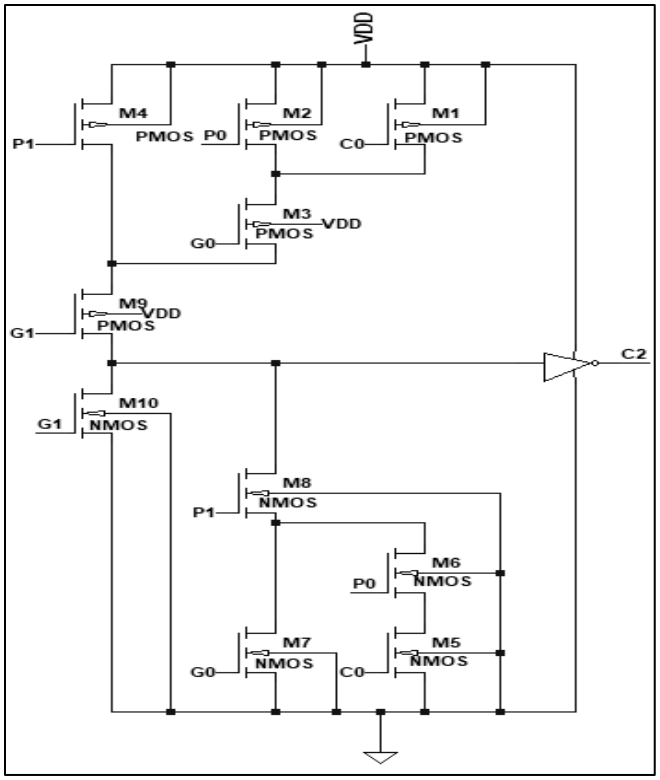


Figure 3.7: Stand-alone 2-bit 'V_pA_n' CLA carry-generation Circuit

Proposed stand-alone carry generation circuits are named based on the pull-up and pull-down structure design topologies. For instance, the configuration with PMOS block representing alphabet 'V' and NMOS block representing alphabet 'A' is named as V_pA_n configuration.

The rise time delay, fall time delay and the propagation delay of all the four proposed configurations across the critical path are listed in Table 3.1. All the transient analysis simulations are performed using ON Semi C5 process technology. All the NMOS transistors in all the four circuits are uniformly sized with a gate length of 0.6u and width of 1.2u maintain the aspect ratio of 2:1, whereas all the PMOS transistors are sized with a gate length of 0.6u and width of 2.4u, maintaining the aspect ratio of 4:1. The rise delay, fall delay and propagation delay across each critical path are computed.

Table 3.1: Delay analysis of 2-bit proposed stand-alone CLA carry generation circuits

CLA Configuration	Delay (ns)		
	Rise Time (t_r)	Fall Time (t_f)	Propagation Delay (t_d)
A_pV_n	1.09	0.68	0.17
V_pV_n	**0.75**	0.77	0.13
A_pA_n	1.12	**0.63**	0.17
V_pA_n	0.77	0.66	0.11

3.2.1 Delay analysis of the proposed stand-alone CLA Configurations

The analysis of the delay results recorded in Table 3.1, is done extensively to select a better topology between the four proposed stand-alone CLA carry generation circuits. From Table 3.1, it is evident that amongst the rise time delays of all the four configurations, V_pV_n and V_pA_n took lower time to raise the output signal from 0.1VDD to 0.9VDD and among both the design topologies the common factor is that their pull-up blocks denote pattern 'V'. The fall time delays are better in A_pA_n and V_pA_n design configurations. The time taken for the output signal to transmit from 0.9VDD to 0.1VDD is lesser in these topologies compared to the other two topologies. In both A_pA_n and V_pA_n, the NMOS block denotes pattern 'A'. Hence, the design configuration that has its PMOS block representing pattern 'V' and NMOS block representing pattern 'A' gave the finest results in terms of the speed, amidst the proposed stand-alone CLA carry generation circuits. It is obvious from Table 3.1, that VpAn, took lesser time to propagate from input carry signal to output carry signal and hence, this design topology is considered for the further experiments of this dissertation. The same kind of analysis is done with 4-bit representations of the proposed stand-alone CLA carry generation circuits. This whole study of 3.2.1 is also done using CMOS

0.25u technology [2].

3.2.2 'V_pA_n' CLA Carry Generation Circuit Implementations for 1, 4, 8 and 16-bits

In view of the analysis from sub-section 3.2.1, the topology 'V_pA_n' is considered and developed for realizing 1-bit,4-bit, 8-bit and 16-bit carry generation logic of CLA. The logic in the equation (2.8) for 1-bit CLA is realized and the schematic is shown in Fig. 3.8, and the equation (2.11) of 4-bit is implemented and is presented in Fig. 3.9. Here G0, P0, and C0 represent the input generate, propagate and carry signals.

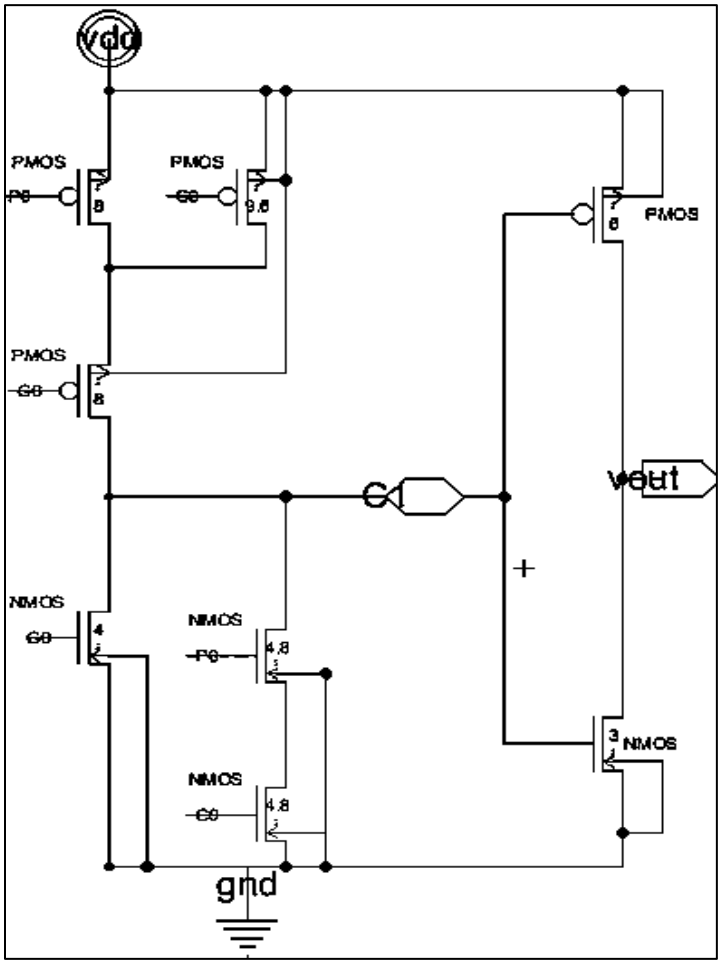


Figure 3.8: 1-bit stand-alone ' V_pA_n' design implementation

Figure 3.9 represents the 4-bit stand-alone design implementation. This design requires 18 transistors to implement the 4-bit CLA carry generation logic. Here, the critical path for raising

the signal from low to high across the pull-up block is through G3-G2-G1-G0-C0 and the critical path for pulling down the output signal from high to low is through P3-P2-P1-P0-C0.

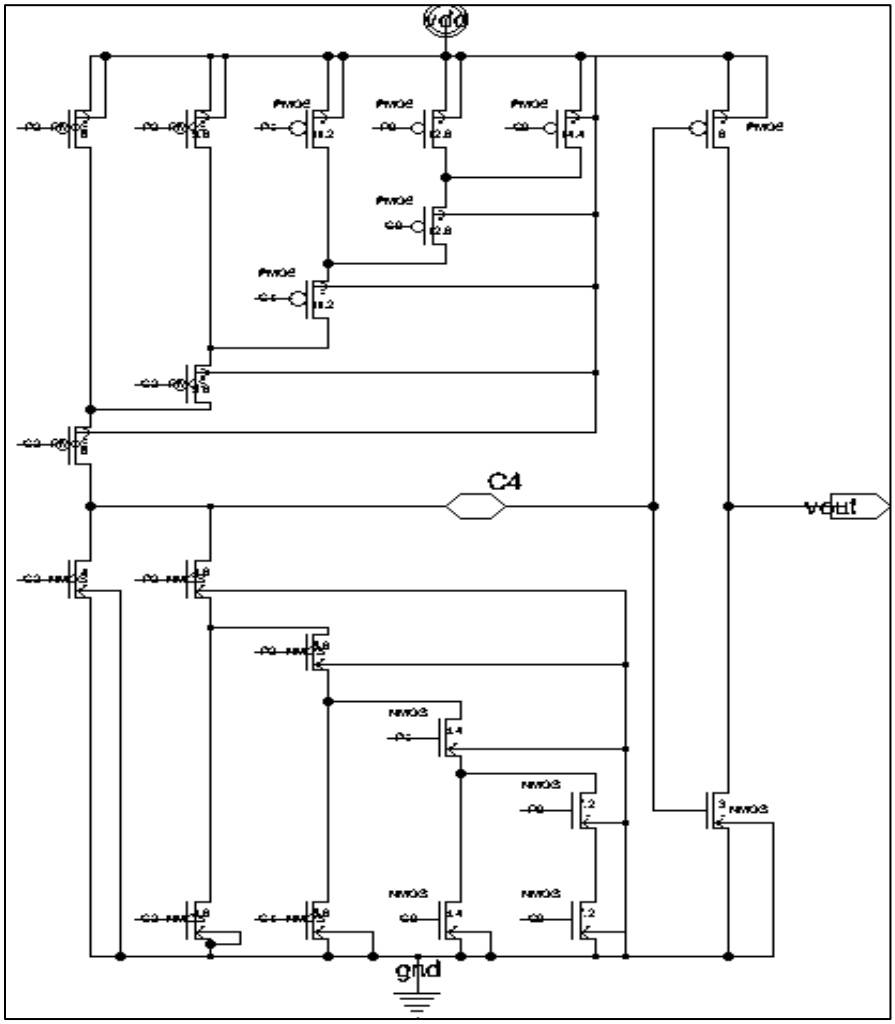


Figure 3.9**:** 4-bit stand-alone 'V_pA_n' design implementation

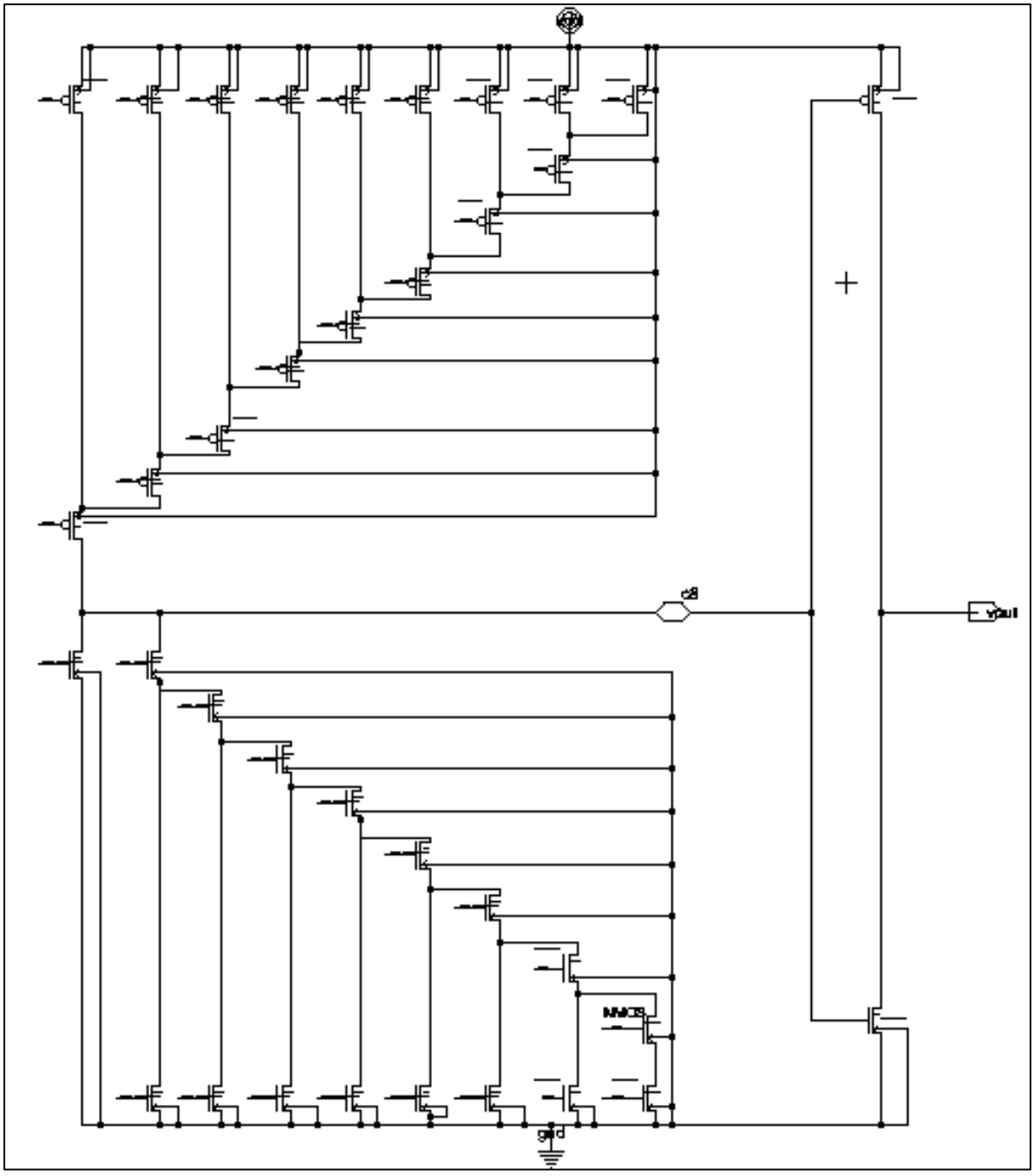


Figure 3.10**:** 8-bit stand-alone 'V_pA_n' design implementation

Figure 3.10 represents the 8-bit stand-alone 'V_pA_n' design implementation. This schematic requires 34 transistors to implement the carry generation logic. The critical path across the pull-up structure is G7-G6-G5-G4-G3-G2-G1-G0-C0 and for the pull-down structure, it is P7-P6-P5-P4-P3-P2-P1-P0-C0. The transistor count across this schematic in Figure 3.10 is small compared to the 8-bit

design schematics described in chapter 2. This reduction in the transistor count reduces the switching activity and power dissipation in the circuit.

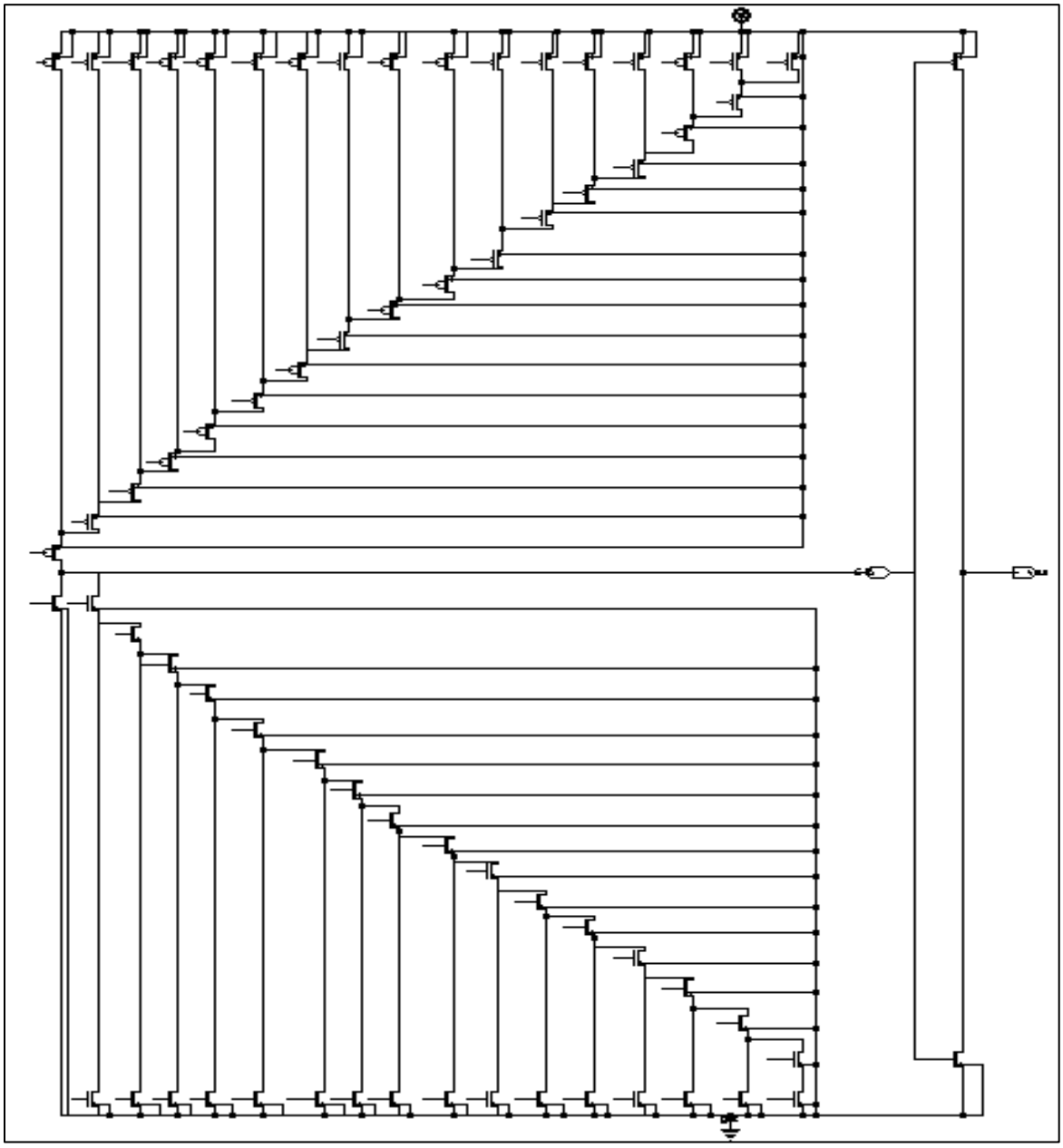


Figure 3.11: 16-bit stand-alone 'V_pA_n' design implementation

Figure 3.11 represents the 16-bit stand-alone 'V_pA_n' design implementation. This design requires 66 transistors for the 16-bit CLA carry generation design implementation. The critical path across the pull-up structure in Figure 3.10 is G15-G14-G13-G12-G11-G10-G9-G8-G7-G6-G5-G4-G3-

G2-G1-G0-C0 and for the pull-down structure, it is P15-P14-P13-P12-P11-P10-P9-P8- P7-P6-P5-P4-P3-P2-P1-P0-C0.

All the schematics in this section are designed using the Electric VLSI Design System, an ECAD tool, which allows the circuit design from schematic to layout. Initially, all the 'V_pA_n' design configurations are simulated by maintaining the aspect ratio and physical dimensions of the PMOS and NMOS transistors the same as in section 3.2.1, uniformly.

3.2.3 Transistor Resizing

As in the course of optimization of the proposed 'V_pA_n' design configurations, for reducing the propagation delay, the transistor-resizing is done all over. The transistors are re-sized depending upon the placement of the transistors. The transistors that are driving the output, the ones which are closer to the outputs have higher widths compared to the transistors that are placed farther from the output. The resistance of the transistor is inversely proportional to its width, higher the width of the transistor, lower is its resistance. For an instance for a 4-bit 'V_pA_n' CLA circuit, the aspect ratio of PMOS transistors of 4:1 with a gate length of 0.6u, is raised to 7.2:1 by augmenting the width of 0.2u across each consecutive vertical path [2]. The corresponding NMOS transistors across each vertical path maintain half the width of the PMOS transistors in that vertical path.

Simulations of all the design configurations discussed in this dissertation are done by raising the generate signals to 1 and zeroing the propagate signals, to see how quick the carry output signals are generated and propagated across the circuit. This method of turning off the unwanted gate signals in the non-critical path also reduces the dynamic power dissipation. Transient analysis of the schematic is performed for all the Proposed 'VpAn' Stand-alone design configurations at ON Semi C5 process technology node and CMOS 0.18um technology node. The propagation delay and the transistor count of each proposed 'VpAn' schematic is listed in Table 3.2.

Table 3.2**:** Schematic ‘VpAn’ Simulation Delay

Proposed Design (VpAn)	No. of Transistors	Propagation Delay (ns) of the schematic at ON Semi C5 process	Propagation Delay (ns) of the schematic at 0.18um
1-bit	6	0.11	0.05
2-bit	10	0.13	0.06
4-bit	18	0.25	0.08
8-bit	34	0.67	0.19
16-bit	66	1.96	0.57

3.2.4 Layout Designs of Proposed ‘VpAn’ Design Configurations

All the proposed stand-alone ‘VpAn’ design configurations are laid out using the Electric VLSI ECAD tool. Fig. 3.12, to Fig. 3.16 present the layout designs of 1, 2, 4, 8 and 16-bit designs respectively.

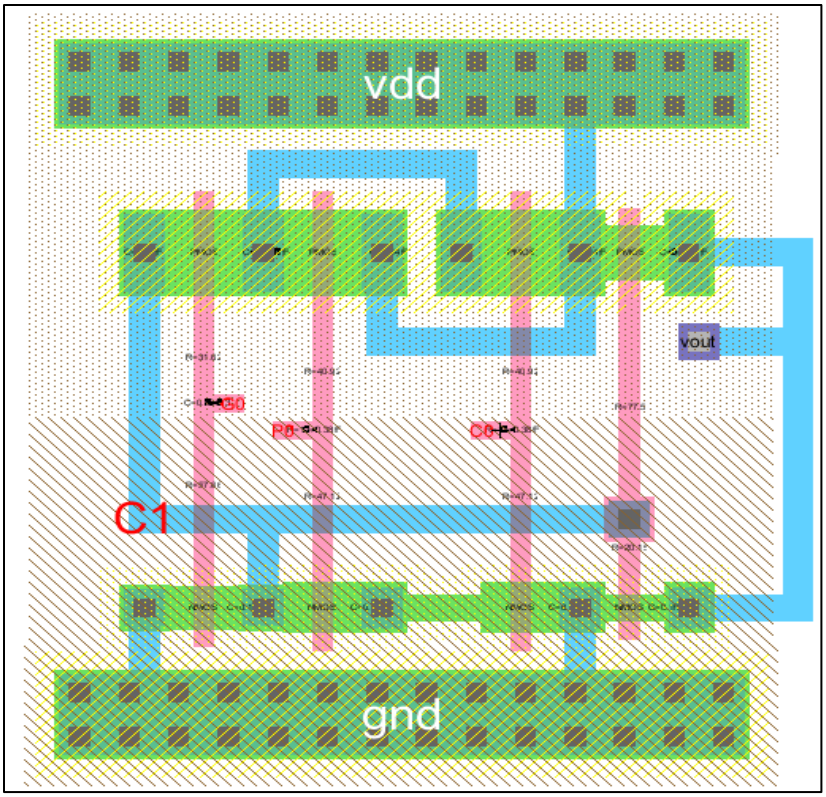


Figure 3.12: 1-bit Stand-alone 'V_pA_n' Layout Implementation

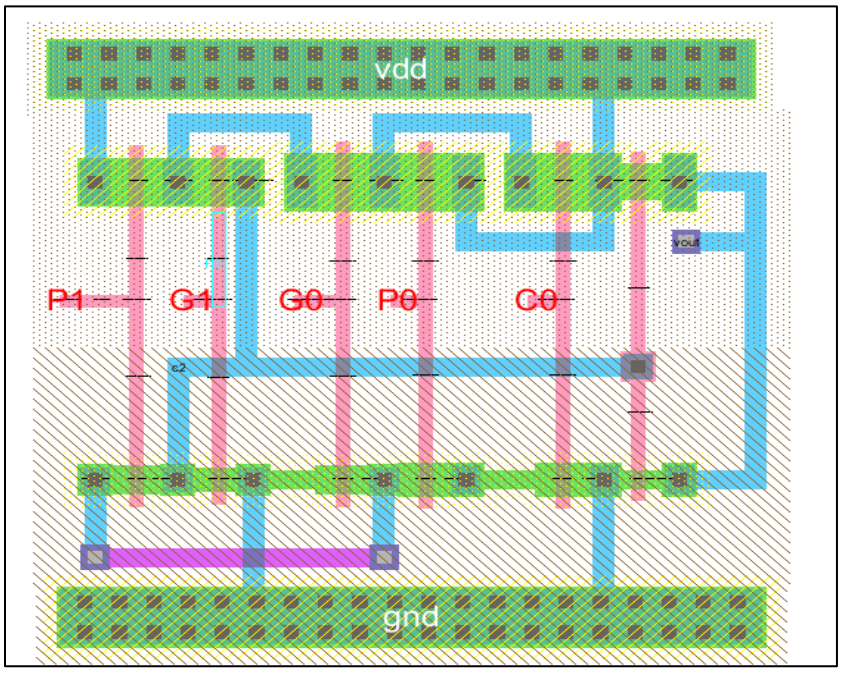


Figure 3.13: 2-bit Stand-alone 'V_pA_n' Layout Implementation

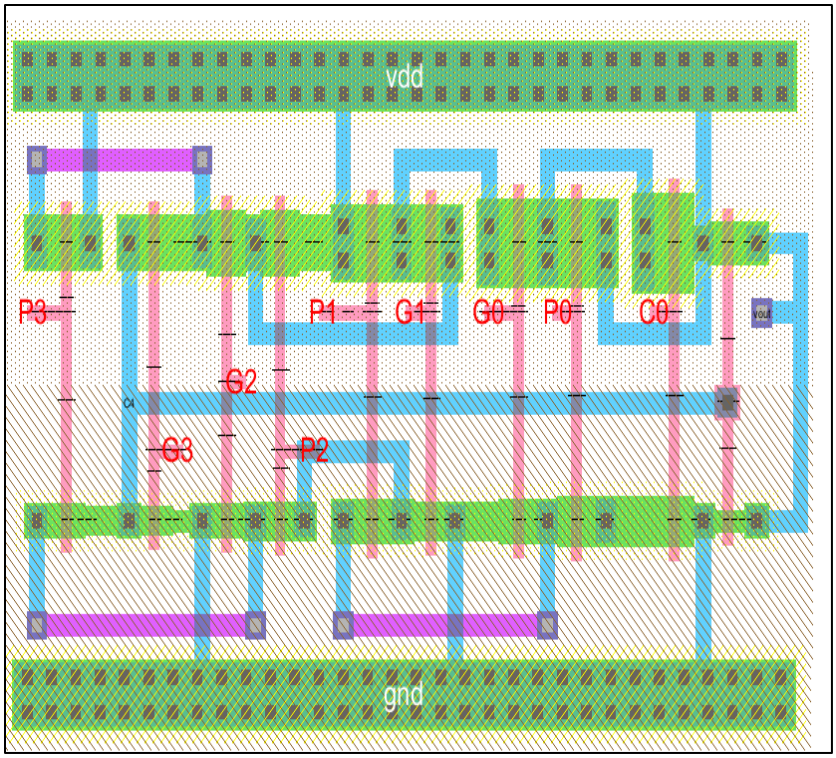


Figure 3.14**:** 4-bit Stand-alone 'V_pA_n' Layout Implementation

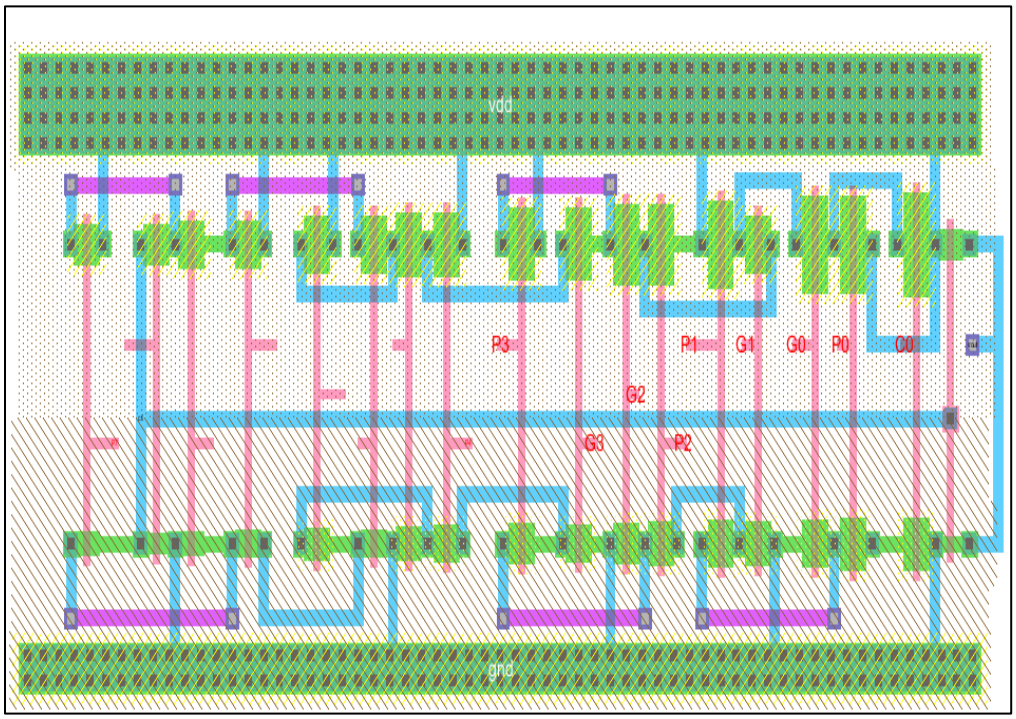


Figure 3.15: 8-bit Stand-alone 'V_pA_n' Layout Implementation

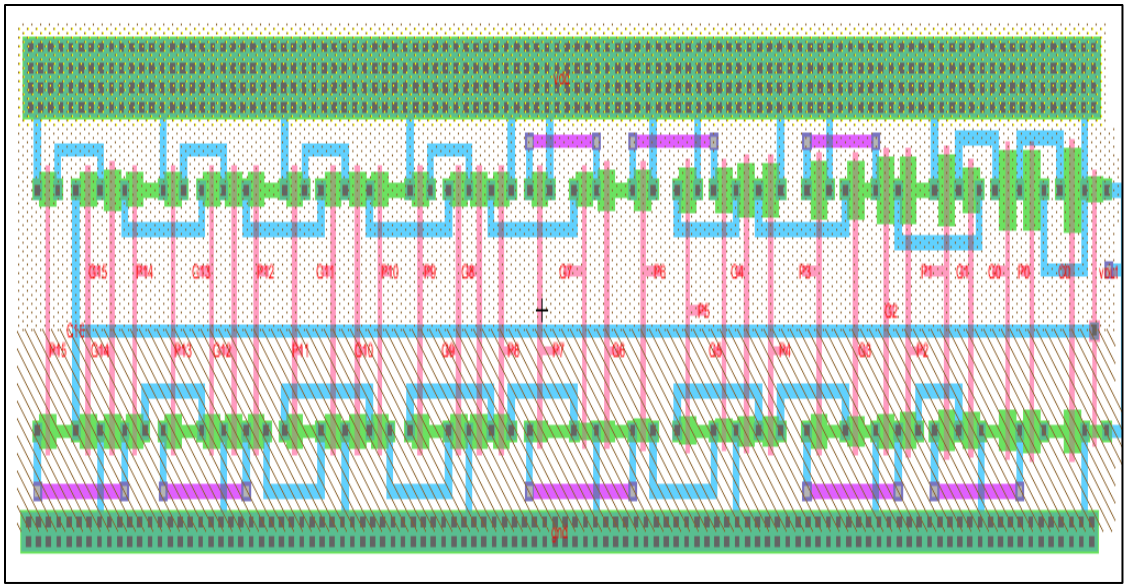


Figure 3.16: 16-bit Stand-alone 'V_pA_n' Layout Implementation

The layout simulations of all the 'VpAn' configurations are performed at both On Semi C5 process technology and at CMOS 0.18um technology node for the propagation delay with and without including RC delay, caused by the device are computed and recorded in Table 3.2.2.

Table 3.3**:** Layout 'VpAn' Simulation Delay at 500nm CMOS technology

Proposed Design (VpAn)	**Propagation Delay (ns) of the layout without device parasitics**	**Propagation Delay (ns) of layout including device parasitics**
1-bit	0.22	0.30
2-bit	0.36	0.57
4-bit	0.77	1.25
8-bit	1.80	3.27

Table 3.4**:** Layout 'VpAn' Simulation Delay at 180nm CMOS technology

Proposed Design (VpAn)	**Propagation Delay (ns) of the layout without device parasitics**	**Propagation Delay (ns) of layout including device parasitics**
1-bit	0.12	0.14
2-bit	0.19	0.22
4-bit	0.38	0.45
8-bit	1.05	1.26

3.3 Analytical RC delay of proposed stand-alone CLA carry generation circuits

Propagation delay of the CMOS circuits can by computed analytically through R-C delay model. The simulated propagation delays are calibrated and recorded as discussed in section 3.1 for the proposed stand-alone CLA carry generation circuits.

In this section, the analytical delay models for the proposed stand-alone CLA carry generation circuits is developed from an existing model for an extensive analysis of the proposed stand-alone circuits. Based on the R-C delay model presented in the section 2.6.1, the delay models of the CLA carry generation circuits are developed.

3.3.1 RC Delay Model for 2-bit CMOS Stand-Alone 'V_pA_n' Configuration

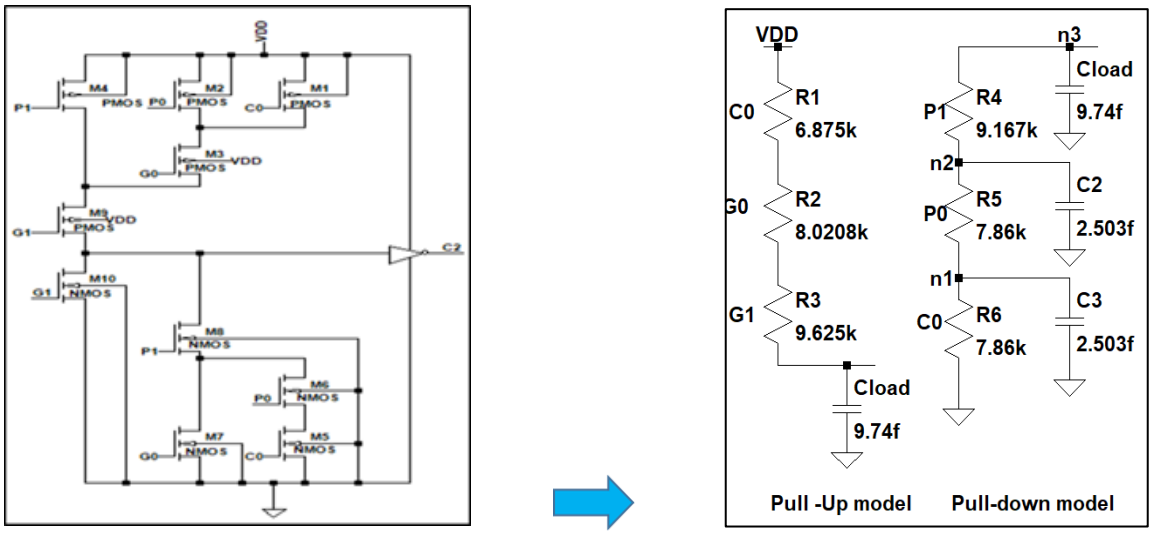


Figure 3.17**:** (a) 2 -bit stand-alone 'V_pA_n' CMOS design implementation and its

(b) RC delay model across the critical path

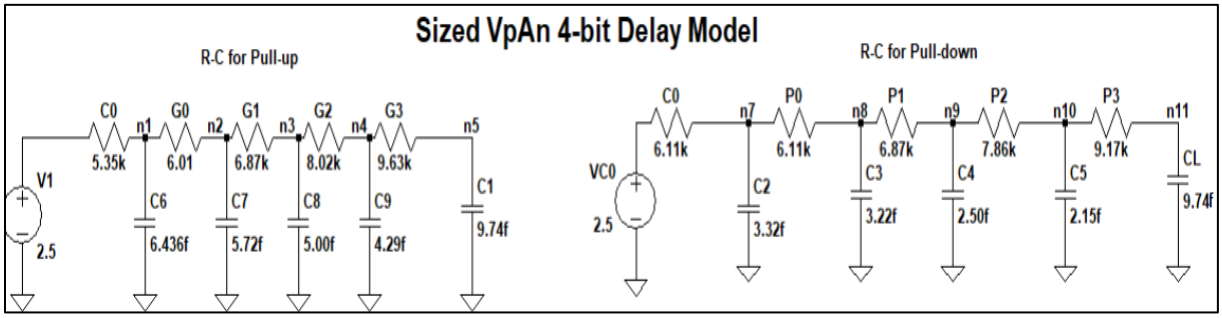


Figure 3.18: RC delay model for 4-bit proposed stand-alone CLA carry generation circuits

Fig. 3.16(b), represents the RC-delay model for the 2-bit 'V_pA_n' design configuration across the critical path of the pull-up network, G_1-G_0-C_0 and across the critical path of the pull-down network P_1-P_0-C_0. Likewise, the RC-Delay models of the critical paths of 4-bit, 8-bit Pull-up and Pull-down networks of proposed stand-alone CLA carry generation circuits are represented in Fig. 3.17 to 3.19 correspondingly.

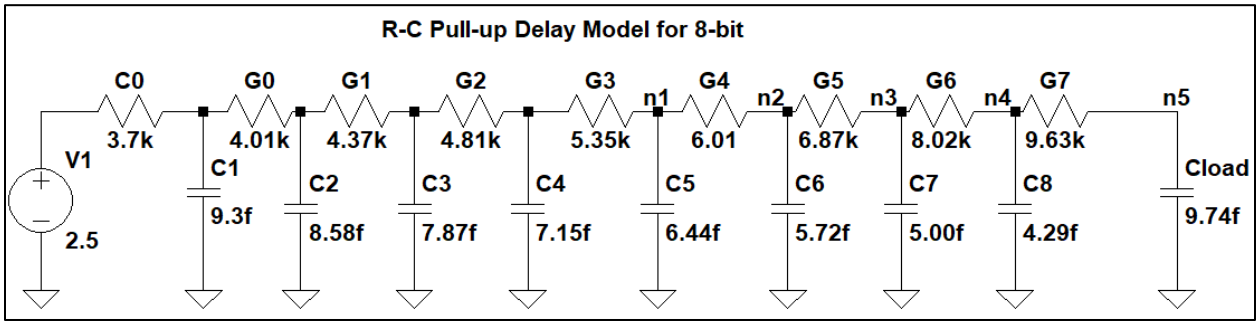


Figure 3.19**:** RC delay model for 8-bit proposed stand-alone CLA carry generation circuits circuit pull-up structure

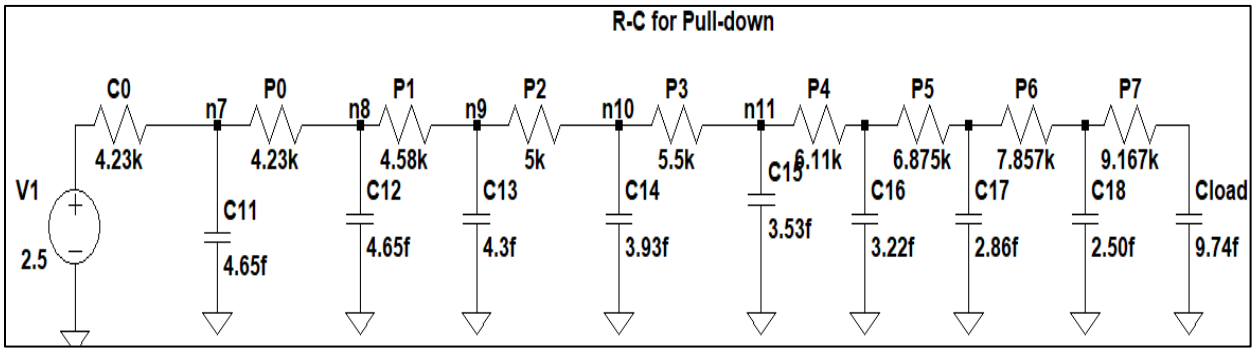


Figure 3.20: RC delay model for 8-bit proposed stand-alone carry generation circuit pull-down structure

The time constant across the pull-up and pull-down networks of the 2-bit 'V_pA_n' circuit in Fig. 3.17 (b), is calculated by the summation of the RC time constants across the network. Rise-time constant and the fall-time constant across the pull-up network in Fig. 3.17 are

$$\tau_{Plh} = R_1C_1 + (R_1 + R_2)C_4 + (R_1 + R_2 + R_3)C_{load} = 0.33ns \quad (3.1)$$

$$\tau_{Phl} = R_6C_3 + (R_6 + R_5)C_2 + (R_6 + R_5 + R_4)C_{load} = 0.3ns \quad (3.2)$$

The propagation delay across the 2-bit stand-alone 'V_pA_n' circuit is

$$t_p = 0.35 \times (\tau_{plh} + \tau_{phl}) = 0.22ns \quad (3.3)$$

Using the same method, the analytical propagation delay values across 1-bit, 4-bit and, 8-bit proposed stand-alone CLA carry generation circuits are computed and listed in Table 3.4.

Table 3.5: Analytical delay of proposed stand-alone CLA carry generation circuits

Proposed Design (VpAn)	Analytical Delay (ns) of the schematics designed at C5 process node
1-bit	0.14
2-bit	0.22
4-bit	0.43
8-bit	0.92

Chapter 4. Comparative Results and Analysis

This chapter delves deep into the results of the proposed stand-alone CLA carry generation circuits and the reference stand-alone CLA carry generation circuits and comprehensive analysis of their results is presented.

4.1 Comparative analysis of the size

Post-recording the simulated data of both reference stand-alone CLA carry generation circuits and proposed stand-alone CLA carry generation circuits, a comparative analysis is done based on their propagation delay and layout size. Table 4.1, represents the number of transistors required to implement the CLA carry generation and propagation logic for 2, 4,8 and 16-bits carry chain length using ON Semi C5 process technology in the proposed pattern and the reference pattern. It also shows the percentage decrease in the device size.

Table 4.1: Size of reference stand-alone CLA carry generation circuits and proposed stand-alone CLA carry generation circuits

Carry chain length (bits)	Reference stand-alone CLA carry generation circuits	Proposed stand-alone CLA carry generation circuits	Decrease in transistor count (%)
2	12	10	17
4	30	18	40
8	90	34	62
16	306	66	78

It could be observed from Table 4.1 that for all the carry-chain lengths implemented, the proposed

stand-alone CLA circuits require a lesser number of transistors compared to the reference stand-alone CLA carry generation circuits. There is a significant decrease in the layout size of the CLA carry-generator, as the carry-chain length increases. Majorly, a notable difference in the sizes could be identified in the 8-bit and 16-bit.

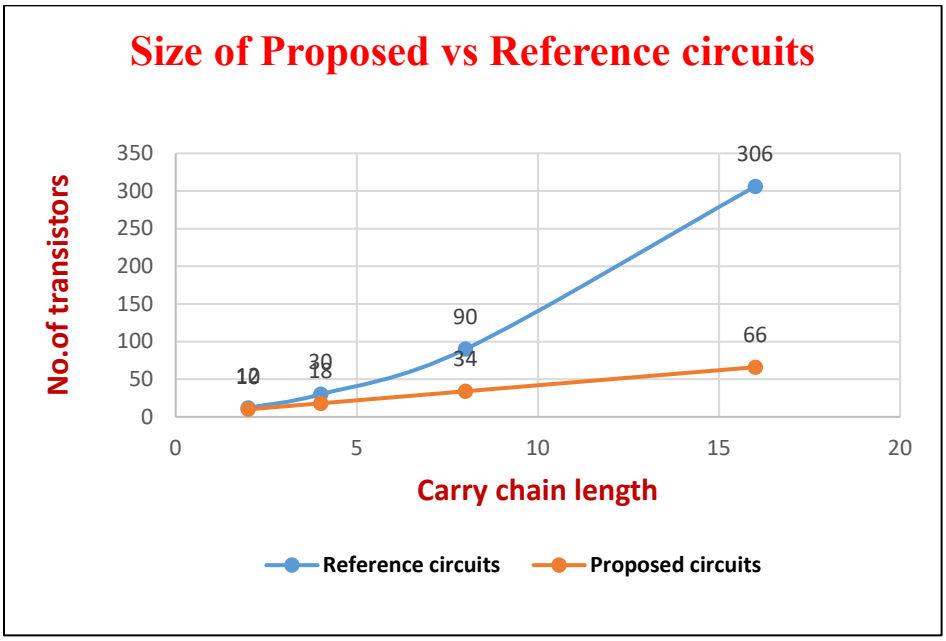


Figure 4.1: Graphical Analysis of reference stand-alone CLA carry generation circuits and proposed stand-alone CLA carry generation circuits

The graphical analysis of the data provided in Table 4.1 is presented in Figure 4.1. It could be seen from Figure 4.1 that there is a significant decrease in the transistor count of the proposed stand-alone CLA carry-generation circuits. The switching activity in these circuits is very less compared to reference stand-alone CLA carry-generation circuits as the transistor count decreases significantly. Hence, the power dissipation in the proposed stand-alone CLA carry generation circuits is very less compared to the reference stand-alone CLA carry generation circuits.

4.2 Comparative Analysis of the speed

Table 4.2: Propagation Delay of reference stand-alone CLA carry generation circuits and proposed stand-alone CLA carry generation circuits.

Carry Chain length(bits)	Reference stand-alone CLA carry-generation circuits	Proposed stand-alone CLA carry-generation circuits	Improvement in speed (%)
2	0.16	0.13	19
4	0.57	0.25	56
8	1.91	0.67	65
16	8.16	1.96	76

The propagation delay measurements for the various carry-chain lengths of 2, 4, 8 and 16-bits recorded and listed in Table 3.2 and Table 3.5 are compared in Table 4.2. Table 4.2 also, represents the percentage improvement in the speed of the CLA carry generators by using the proposed stand-alone circuits. Observing Table 4.2, it is evident that the propagation delay of the proposed circuits is much lower than the reference CLA circuits. Similar to the size of the circuit, using 'V_pA_n' design topology compared to the reference stand-alone CLA carry generation circuits, there is a significant increase in the speed of the carry generation and propagation, as the carry-chain length increases.

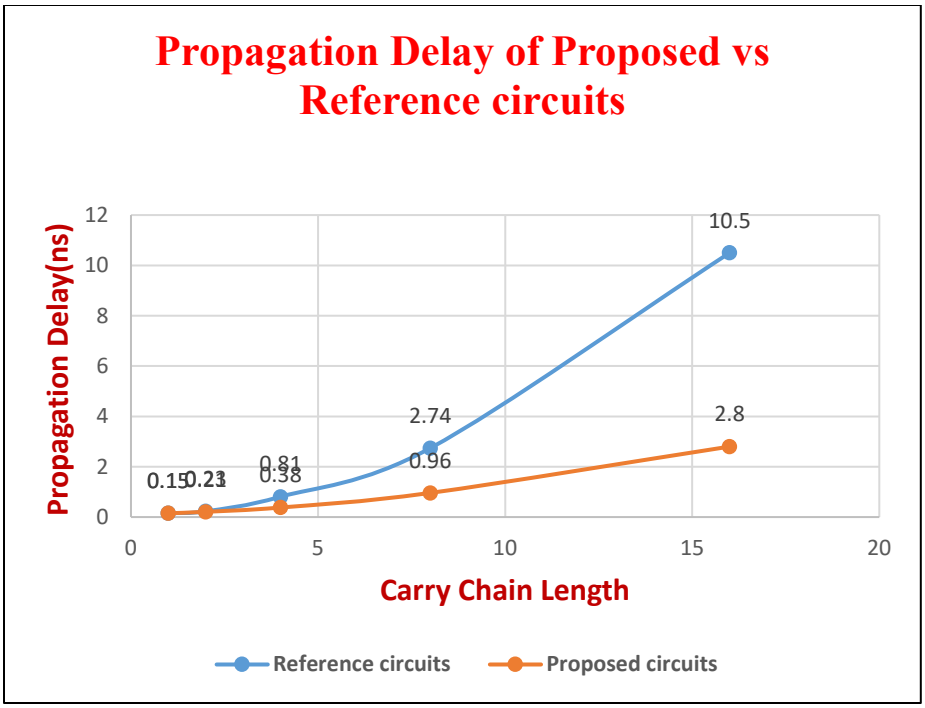


Figure 4.2: Graphical analysis of propagation delay of reference stand-alone CLA carry generation circuits and proposed stand-alone CLA carry generation circuits

The difference in the propagation delay of the reference stand-alone CLA carry generation circuits and proposed stand-alone CLA carry generation circuits is shown in Fig. 4.2. As, the propagation delay of the reference stand-alone CLA carry generation circuits grows exponentially, for the increase in the carry-chain length, the propagation delay of the proposed stand-alone CLA carry generation circuits grows linearly.

The proposed stand-alone CLA carry generation circuits when connected along with the adder logic circuitry of CLA, perform the high-speed logic addition compared to the reference stand-alone CLA carry generation circuits when connected to the adder block.

4.3 Application of a Proposed Carry Chain Circuit in a 4-bit Adder

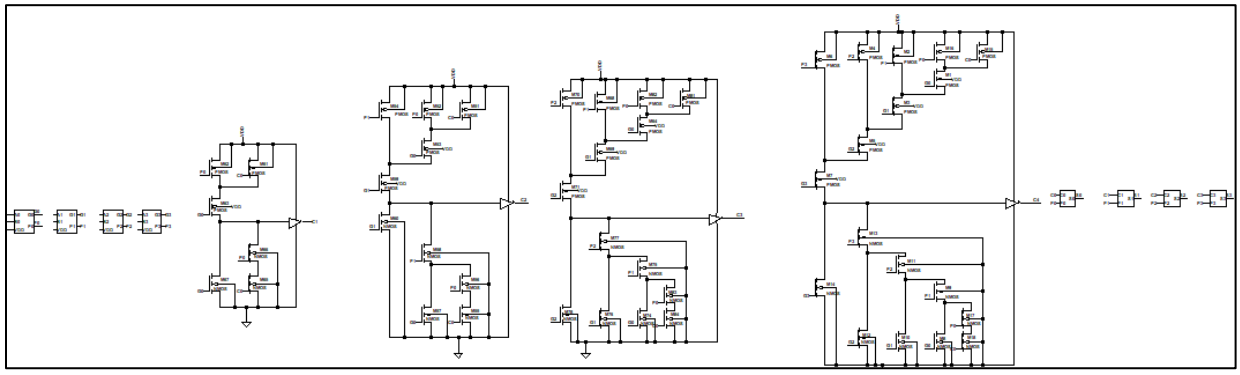


Figure 4.3: Four-bit Carry Look-ahead Adder using Proposed Carry Chain Circuit.

A four-bit carry look-ahead adder with proposed carry chain circuit for carry generation and propagation is designed using LTspice tool and is presented in Fig. 4.3. Here, the initial blocks represent the schematic of carry generation and propagation signals, and an instance of a block diagram of level0 is shown here in Fig. 4.4.

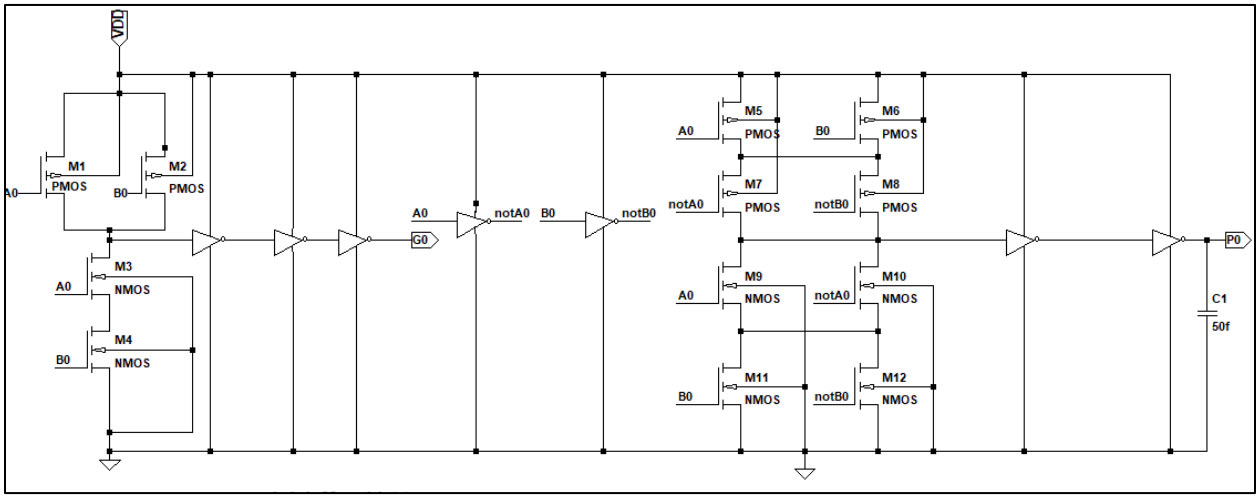


Figure 4.4: Generate and Propagate Signals Schematic Block Instance

The proposed single stand-alone CLA carry generation circuit is used in this adder and it generates the output carry signal 'C_4' using the schematic in Fig. 3.13. The sum computation of level0 is shown using the circuitry in Fig.4.8. Similarly, the sum bits across each level can be generated

using the logic hardware used in Fig 4.8, with the corresponding inputs of that particular level. The transient analysis of Fig. 4.5, is performed and the input and output carry signal waveforms are represented in Fig 4.6.

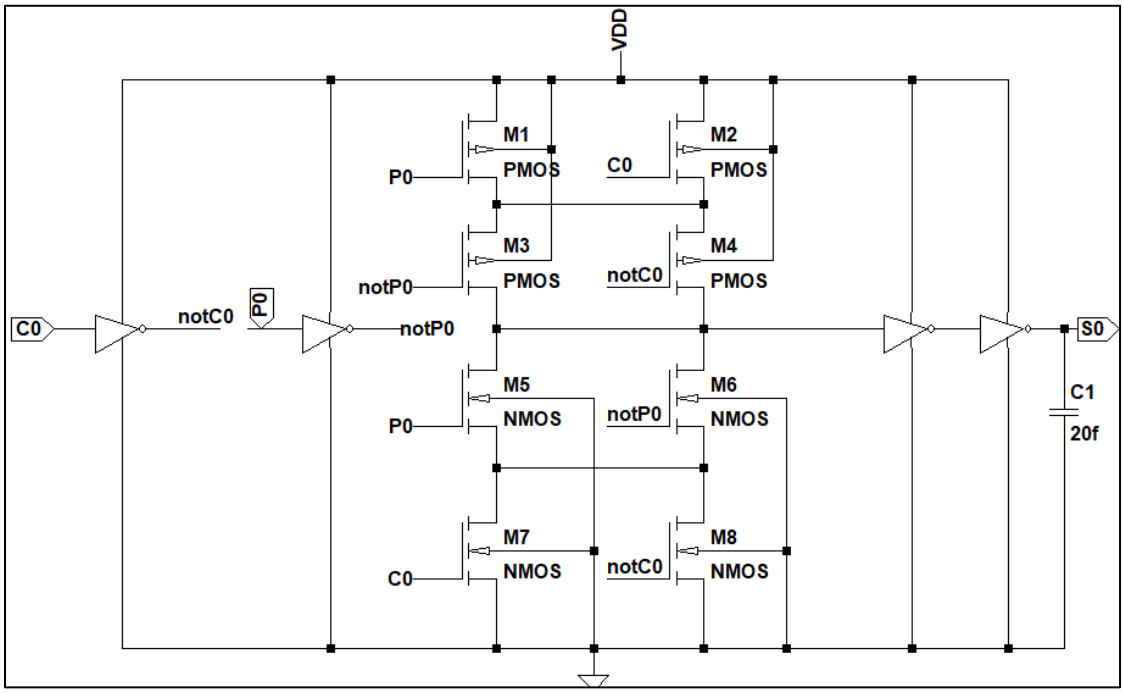


Figure 4.5: Sum Computation Circuitry

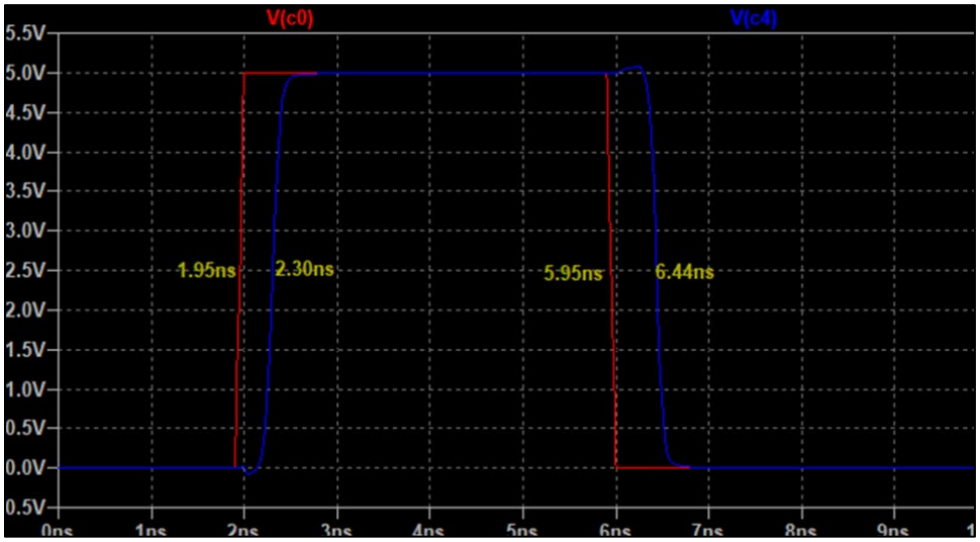


Figure 4.6: Transient analysis of a 4-bit Adder with stand-alone carry chain circuit.

The average propagation delay of the output carry signal, in this case, is 0.42ns.

Table 4.3: Propagation Delay of 4-bit Cascaded CLA and 4-bit adder with proposed stand-alone CLA carry generation circuit

Propagation Delay of 4-bit Cascaded-CLA	Propagation Delay of 4-bit CLA with proposed stand-alone CLA carry generation circuits	Difference	Improvement in Speed (%)
0.95ns	0.42ns	0.53ns	**56**

By, the insertion of 4-bit proposed stand-alone CLA carry generation circuit into the cascaded adder block the delay of the carry look-ahead Adder carry signal is reduced by 0.53ns and the speed is improved by 56% as seen in Table 4.3. It is evident from this table that proposed stand-alone CLA carry generation circuits could be used in the applications wherever adder is needed for enhanced performance.

Chapter 5. Conclusion and Outlook

In the current world of microelectronics, there is always a need for high performance, lower power consumption, and portable computing devices. High-performance digital devices have become a part of our daily life and the processors used today are measured based on their ability to perform computations. The addition is the vital operation performed in the computational devices like ALU, GPU and DSP architectures and hence, digital adders are widely used. CLA is one of the high-speed digital adders. The cascaded carry generator structure used for carry generation and propagation operation in multi-bit CLA adders require more time to generate and propagate a signal from input to output. The design topology used for cascading needs more amount of the transistors. As the transistor count increases, the switching activity increases. This increases the layout size, propagation delay of the carry signal and power dissipation factors Theses parameters raise the design complexity to the designers. This is the trade-off of the CLA carry generation circuit design. Proposed stand-alone CLA carry generator circuits have a single architecture to generate and propagate the carry signals. The proposed stand-alone CLA carry generation circuits for 2-bit, 4-bit, 8-bit and 16-bit have been introduced in this work with a lesser number of transistors compared to the reference CLA structures used.

5.1 Summary

This dissertation presents an approach to reduce the propagation delay, the layout size of the carry generation circuits. The patterns proposed does not rely on the carry of the preceding block. It helps to reduce the delay caused due to the propagation of the signals across large circuits. The major contribution of the work is introducing the stand-alone carry generation circuits and analyzing their performance both analytically and empirically. Optimization of the proposed stand-alone CLA circuits is done as part of the design by resizing the transistors repeatedly. During this

research, the proposed circuits are designed, simulated and their delays are calibrated using various CMOS process technologies from 1um, ON Semi C5 process design, 0.25um to 180nm. Physical layouts of the proposed circuits are designed. The time taken by these designs to propagate the carry signals from input to the output and generate the carry signals including the device parasitic is measured. Analytical RC delay models are developed for the proposed circuits. Reference CLA carry generation circuits are developed, optimized and their delays are calibrated, and the performance of the proposed stand-alone CLA circuits is compared to the reference stand-alone CLA circuits based on their design issues.

5.2 Contributions to the research problem

- The overall goal of the research is to introduce single architecture carry generator circuits of the CLA of various carry-chain lengths for better performance and the device size, to overcome the limitations of the cascaded carry generator circuits. This is achieved by developing four different stand-alone patterns and investigating them in the aspects of their propagation delay, layout size, and dynamic power dissipation.
- The performance of the proposed pattern carry generation circuits for various carry chain lengths are compared to the reference carry generation circuits in virtue of showing their improvement in the context of their layout size and speed. This is a significant research question which is answered.
- The application of the proposed carry generation and propagation circuit in the complete CLA adder unit is realized and its carry output is evaluated.

5.3 Outlook

Optimization of the carry look-ahead logic structures is an interesting and advantageous field of research. Even though the research in this area has been going on for years, there is still scope for the research in solving the design issues. Proposed stand-alone CLA carry generation circuits solves some of these design issues. The CLA adders used in ALU's, GPU's for memory generation, in PC are supposed to have high-efficiency and performance. The author of this dissertation believes that using CLA with the proposed stand-alone carry generator circuits would be a stepping-stone to provide a concrete solution for the limitations created due to vast cascaded-CLA structures. As the designers designing the IC's use the library cells for their design, this study would help design researches to synthesize new library cells for carry-generation in CLA. During the process of optimizing the performance of CLA carry-chain circuits, the sum generation circuit isn't much concentrated. Enhancing the speed of the adder circuit, and integrating it with the proposed stand-alone CLA carry generation circuits, would make this CLA best fit for any high-performance processing applications in the variety of computers.

Chapter 6. Recommended Future Work

In this study, the proposed stand-alone CLA carry generation circuits are investigated at various CMOS process technologies and they facilitated the speed and the layout size-reduction requirements of the cascaded-structures. These patterns enhanced the performance of the CLA carry generator circuits to a greater extent.

Optimizing the summing circuitry in terms of its design issues is a highly-recommended future work for some researcher who is interested to continue this study. This will enhance the performance of many application chips using the CLA structures for their arithmetic applications. Implementation of these proposed circuits on silicon is a potential area of work to be done. This can be done by submitting this design to MOSIS, a fabrication service which prototypes the design on a shared mask with other designs at a low cost to the designers and academicians.

References

[1] N. S. Muppaneni, S. C. Chiu, V. Kantabutra and F. Dailami, "Comparative Analysis of Various Design Implementations of CLA Carry Chains," *2018 IEEE International Conference on Electro/Information Technology (EIT)*, Rochester, MI, 2018,pp.0713-0718.doi:10.1109/EIT.2018.8500112

[2] Muppaneni, N.S.; Chiu, S, "Augmented Efficiency of CLA Logic Through Multiple CMOS Configurations," 2017 Future technologies Conference (FTC), Vancouver, Canada, 2017.

[3] Binggeli, M.; Denton, S.; Muppaneni, N.S.; Chiu, S, "Optimizing carry-lookahead logic through a comparison of PMOS and NMOS block inversions," 2015 IEEE International Conference on Electro/Information Technology (EIT), Dekalb, IL,2015, pp. 641 – 646

[4] Amita, A., & Sachdeva, N. (2014). Design and Analysis of Carry Look Ahead Adder Using CMOS Technique. IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 9, Issue 2, Ver. VII (Mar-Apr. 2014), PP 92-95

[5] H. A. D. Nguyen, L. Xie, M. Taouil, R. Nane, S. Hamdioui and K. Bertels, "Computation-in-memory based parallel adder," *Proceedings of the 2015 IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH'15)*, Boston, MA, 2015, pp. 57-62.doi:10.1109/NANOARCH.2015.7180587

[6] *Samanta, Jagannath and Mousam Halder. "Performance Analysis of High-Speed Low Power Carry Look-Ahead Adder Using Different Logic Styles." (2013).* International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-6, Jan- 2013

[7] Lily Kanoriya, Aparna Gupta, Dr. Soni Changlani, "Layout Designing and Transient Analysis of Carry Look-ahead Adder Using 300nm Technology-A Review", International Journal of Engineering Development and Research (IJEDR), ISSN:2321-9939, Volume.4, Issue2,pp.467-472,May2016,
Available at: http://www.ijedr.org/papers/IJEDR1602084.pdf

[8] Yu-Ting Pai and Yu-Kumg Chen, "The fastest carry look-ahead adder," *Proceedings. DELTA 2004. Second IEEE International Workshop on Electronic Design, Test and Applications*, Perth WA, Australia,2004,pp.434-436.doi:10.1109/DELTA.2004.10071

[9] J. Miao and S. Li, "A novel implementation of 4-bit carry look-ahead adder," *2017 International Conference on Electron Devices and Solid-State Circuits (EDSSC)*, Hsinchu, 2017,pp.1-2.doi:10.1109/EDSSC.2017.8126457

[10] Wei-Cheng and Jianping Hu; ”A structured approach for optimizing 4-bit Carry look-ahead adder”, The Open Electrical & Electronics Engineering Journal, 2014, 8, 133-142

[11] T.Archanca, K.Arunkumar, AH Malini; ” Delay Depreciation and Power Efficiency Carry Look-Ahead Adder Using CMOS”, Journal of Chemical and PharmaceuticalSciences, ISSN:974,2115

[12] K. Seshadri, A. Pontarelli, G. Joglekar and G. E. Sobelman, "Design techniques for Pulsed Static CMOS," *2004 IEEE International Symposium on Circuits and Systems (IEEE Cat.No.04CH37512)*, Vancouver, BC, 2004, pp. II-929.doi:10.1109/ISCAS.2004.1329425

[13] Balasubramanian, P. and Nikos E. Mastorakis. “ASIC-based Implementation of Synchronous Section-Carry Based Carry Lookahead Adders.” CoRR abs/1603.07961 (2016).

[14] J. Miao and S. Li, "A novel implementation of 4-bit carry look-ahead adder," *2017 International Conference on Electron Devices and Solid-State Circuits (EDSSC)*, Hsinchu, 2017, pp.1-2.doi: 10.1109/EDSSC.2017.8126457

[15] Mr. “Design and Implementation of Low Power 8-bit Carry-look Ahead Adder Using Static CMOS Logic and Adiabatic Logic.” (2013). I.J. Information Technology and Computer Science, 2013, 11, 78-92 Published Online October 2013 in MECS (http://www.mecs -press.org/).doi: 10.5815/ijitcs.2013.11.09

[16] A Subodh Wairya, A Rajendra Kumar Nagaria, A Sudarshan Tiwari, “New Design methodologies for high-speed mixed-mode CMOS full adder circuits” International Journal of VLSI Design & Communication Systems,2011 Academy & Industry Research Collaboration Center (AIRCC)

[17] F. Vahid, *Digital Design: with RTL Design, VHDL, and Verilog. 2nd Ed*. Hoboken, New Jersey: John Wiley & Sons, Inc., 2011

[18] *Chopra, Mohit P. “Design Optimization of Double Gate Based Full Adder.” (2015),* Innovative Systems Design and Engineering, ISSN 2222-1727 (Paper) ISSN 2222-2871 (Online) Vol.6, No.9, 2015

[19] L. Bisdounis, S. Nikolaidis and O. Koufopavlou, "Analytical transient response and propagation delay evaluation of the CMOS inverter for short-channel devices," in *IEEE Journal of Solid-State Circuits*, vol. 33, no.2,pp.302-306,Feb.1998. doi: 10.1109/4.658636

[20] Ting-Wei Chiang, C. Y. R. Chen and Wei-Yu Chen, "An efficient gate delay model for VLSI design," *2007 25th International Conference on Computer Design*, Lake Tahoe, CA,2007,pp.450-455.doi: 10.1109/ICCD.2007.4601938

[21] https://www.mosis.com/what-is-mosis

[22] Kumar, Arvind, and Anil K. Goyal. “Study of Various Full Adders using Tanner EDA Tool.” (2012*).* ISSN : 0976-8491 (Online) | ISSN : 2229-4333 (Print) IJCST Vol. 3, Issue 1, Jan. - March 2012

[23] J. Luu *et al*., "On Hard Adders and Carry Chains in FPGAs," *2014 IEEE 22nd Annual International Symposium on Field-Programmable Custom Computing Machines*, Boston MA, 2014, pp.52-59.doi: 10.1109/FCCM.2014.25

[24] M. Rashid and A. Muhtaroğlu, "Power delay product optimized hybrid full adder circuits," *2017 International Artificial Intelligence and Data Processing Symposium (IDAP)*, Malatya,2017,pp.1-4.doi: 10.1109/IDAP.2017.8090319

[25] M. Sarkar, G. S. Taki, Prerna, R. Sengupta, and S. N. Ray, "Design of ripple carry adder using CMOS output wired logic-based majority gate," *2017 8th Annual Industrial Automation and Electromechanical Engineering Conference (IEMECON)*, Bangkok, 2017,pp.328-331.doi: 10.1109/IEMECON.2017.8079617

[26] P Yadav, G Kumar, S Gupta, “Design and implementation of 4-bit arithmetic and logic unit chip with the constraint of power consumption,” *IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 9, Issue 3, Ver. V (May - Jun. 2014), PP 36-43 www.iosrjournals.org*

[27] Intel 80088-bit parallel central-processing unit user’s manual

[28] Fairchild Semiconductor DM 74LS181 4-bit Arithmetic Logic Unit

[29] Z. Guan, W. Li, W. Ding, Y. Hang and L. Ni, "An Arithmetic Logic Unit design based on reversible logic gates," *Proceedings of 2011 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, Victoria, BC, 2011, pp. 925-931. doi: 10.1109/PACRIM.2011.6033020