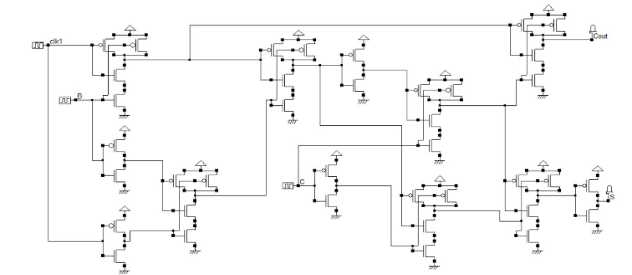
Graphical Abstract



**Guide for Design of a 2-bit Binary Parallel Ripple Carry Adder Using**

**CMOS NAND Gates with Microwind (Process 65nm) and EDA tool**

**for verification**

*Khader Mohammed, Walaa Kashou, Aziz Qaroush, Raha Zabade*

Leave this area blank for abstract info.

|  |  |  |
| --- | --- | --- |
|  | Engineering Science and Technology, an International Journal  journal homepage: www.elsevier.com/locate/jestch | C:\Users\0002402\Desktop\Jestch.jpg |

Design of a 2-bit binary parallel ripple carry adder using CMOS NAND gates with Microwind (process 25nm) and EDA tool for verification

|  |  |  |
| --- | --- | --- |
| ARTICLE INFO |  | ABSTRACT |
| Article history:  Received  Received in revised form  Accepted  Available online |  | With the advancements of online learning especially in the computer and semiconductor industry, teaching undergraduate students on how to design, simulate and implement in layout one of the most critical parts in almost every digital signal processing application is crucial. This paper addresses the implementation of adder design employing CMOS 2-bit Binary Parallel Ripple Carry Adder Using CMOS NAND Gates with Microwind tools with minimum noise margin and lower speed degradation which leads to significant improvement in the speed of the overall circuitry. The investigation is carried out with simulation runs on the HSPICE environment using 65 nm process technology at 25 °C using the Microwind layout tool. Finally, the design guidelines are derived to select the most suitable topology for the desired applications. Investigation reveals that the proposed design using carry adder with CMOS NAND proves to be more speed efficient in comparison with the other two considered design strategies. |
| Keywords:  2-bit parallel ripple carry adder  CMOS Logic gates  Microwind tool  Verification  efficiency |  |
|  |  |  |

# Introduction

With the advancement of automotive, artificial intelligence, machine learning, and high data science, complementary metal-oxide silicon (CMOS) logic circuits are extensively used in Very-Large-Scale-Integration (VLSI) chips. The trend of CMOS technology improvement continues to be driven by the need to integrate more functions within a given silicon area, reduce the fabrication cost, increase operating speed, and dissipate less power as proposed in CMOS was used due to momentous characteristics such as high performance, low power consumption, small size, lightweight and has the flexible computational ability. CMOS has also inherited characteristics of having a lesser number of transistors and is been used to build VLSI chips which constitute the brain of recent Integrated circuits (ICs). For low power and real-time applications, computationally intensive digital signal processing algorithms are implemented in dedicated VLSI systems, adders are one of the most crucial circuits used in such systems. The complexity of the design directly depends on the speed of circuit computation. High-speed requirement results in increased complexity of the circuit; hence a larger number of transistors will be required in the design which further results in high power dissipation. So, there is a tradeoff between speed, area, cost, and power dissipation for adder design.

Adder performance characteristics directly affect the functioning of the entire system. Therefore, improvement in the performance of adder architecture is a prime concern as reported [1]. Various techniques can be employed externally or internally to improve the overall performance of

\*Corresponding author. Tel.: +9720592961614

e-mail: khamadawwad@birzeit.edu

any system. External techniques involve dealing with input data characteristics whereas internal techniques are concerned with the logic, circuit design, and architecture of multiplier which depends on adder design is reported [2–5].

The main goal of this paper is to teach and to help the student to learn and get the optimum design, at the same time to use this paper as a learning tool to do all implementation steps such as schematic and logic creation, simulation, create the layout and run verification for adder design. The design methodology carried out in this paper demonstrated in detail starts with the proposed improvement of circuit diagram, design verification, and the simulation using Microwind and The DSCH program which is a logic editor and simulator. You can download it from the website that referenced as 6.

This design showed how we reduced the number of transistors used to build the circuit using the 65nm process to achieve a power consumption reduction of 32% compared to power consumption being used in earlier design as proposed in [7]. The area of the system is 59.3µm2 and the delay for the whole system is 207ps.

The remainder of this paper is organized as follows. In section 2, the background of adders is addressed using different logic styles in section 3, circuit diagram, and simulation. Section 4 presents the Design results and Comparison. Finally, in section 5, the conclusions are drawn.

# Background

|  |  |  |  |
| --- | --- | --- | --- |
| Symbol | Truth Table | | |
| 2-input logic NAND gate    2-input NAND Gate | B | A | Q |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

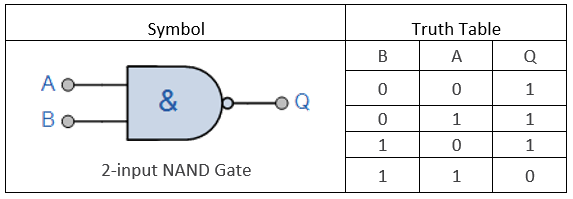
**Fig. 1.** 2-input NAND gate symbol and Truth Table.

This section presents procedures along with optimization for adders which is considered in this paper.

Adder is the basic architecture commonly used in every arithmetic circuit. In a multiplier and many arithmetic units, design adders are used for partial product addition and thus contribute toward the largest part of delay associated with whole multiplication and other arithmetic operations. For example, as presented in [5] and 8-bit Multiplier is designed with three different adder architectures i.e. CMOS full adder, Double Pass Transistor logic (DPL) as proposed by References [8] and [9] and multioutput Carry Lookahead (CLA) adder as demonstrated by Reference [10]

|  |  |  |
| --- | --- | --- |
| Symbol | Truth Table | |
| Invertor Gate | A | Q |
| 0 | 1 |
| 1 | 0 |

**Fig. 2.** Invertor symbol and Truth Table



**Fig. 1.** 2-input NAND gate symbol and Truth Table.

## Logic background

Logic design flow in this paper starts with the specification and ends with the verification. Typically, the stages start with adder specification then with architecture design which can predict the performance and the power of design. Then the functional design identified the main functional units of the adder. After that, the logic design and circuit design implemented. Finally, the adder verification completed using the EDA playground tool.

Design a ripple adder using CMOS, NAND, and inverter gates being introduced in many types of research while the challenge used to be achieving higher design efficacy. This paper introduced a new design methodology that enhanced efficiency by using a fewer number of CMOS gates which incorporate less power consumption, less heat dissipation, smaller size, and lighter weight. All of these factors being improved in the proposed design.

A ripple carries adder is a digital circuit that produces the arithmetic sum of two binary numbers. It can be constructed with full adders connected In cascade with the carry output from each full adder connected to the carry input of the next full adder in the chain as demonstrated by [11]. The 2-inputs CMOS NAND gate consists of 2 PMOS and 2 NMOS transistors. Fig. 1 shows the symbol for 2-inputs CMOS NAND gate and truth table for 2-inputs CMOS NAND gate.

In addition to the CMOS NAND gates, an inverter gate is required to accomplish the functionality. A simple NOT gate constructed using a PMOS and NMOS transistors. Fig. 2 shows the symbol and truth table for the inverter gate.

Arithmetic sum combinational circuit of three bits A, B, and Cin where Cin is a carry bit. The corresponding output sum bit S and a carry-out bit Cout. Fig.3. shows the logic diagram for the full carry ripple adder.

Two single-bit full adders are used in a cascade combination that ripples the Cout bit to a Cin bit for the next adder. Table 1 shows the truth table of a 1-bit full adder as proposed by [12]

Each 1-bit adder used to be built is using three basic logic components (AND, OR & XOR). In the design proposed in this paper, it’s built using NAND universal gate and inverter gate. Optimized adder calculation carried out and the proposed design shown below in Fig. 4 where seven of 2-inputs NAND gates and 5 inverters used only.

## Design Verification

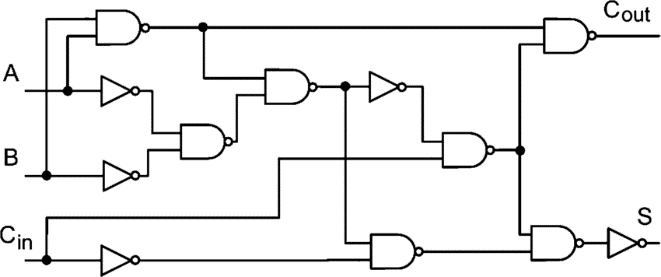
This section introduces the verification that the design which is shown in Fig. 4. is working properly for different cases. EDA playground used to write a Verilog design and testbench code for this adder as shown in this [link](https://www.edaplayground.com/x/5s5g?fbclid=IwAR3Qh5SdEu9M3_4P9gepHckNnkhpaaElcxppLF1yxrrHOG4VUWXpqT26QXQ). A video prepared to show the steps which demonstrate EDA Playground for design verification. [(Link)](https://www.youtube.com/watch?v=T0yFAWawSBM).

## نتيجة بحث الصور عن ‪full adder circuit‬‏

**Fig.3.** Full adder logical diagram.

**Table 1.** 1-bit full adder truth table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **An** | **Bn** | **Cn-1** | **Sn** | **Cn** |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |



**Fig. 4.** Realization of sum and carry by using NAND and invertor gates.

## Circuit Implementation

A comparison with another design which as in [7] that attains the same functionality proves the new design improvements. First, the number of transistors used to build the circuit dropped from 56 to 38 devices which means 32% off power consumption reduction. Second, the variety of gate types has been reduced which incorporates smaller circuit sizes. Furthermore, a more compact design reduces the adder delay. Table 2 shows the number of gates used in each design. From the truth table, it’s clear that the SUM (S) output is the result of the Exclusive-OR gate. Also, the Cout is the result of the AND gate. Then the boolean expression for adder is as follows:

For the Sum bit:

Sum = A XOR B = A ⊕ B

For the Carry bit:

**Table 3.** Performance for both designs.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Common design | Introduced design |  |
| Area | 600 µm2 | 59.3µm2 | Area |
| Delay | 431ps | 207ps | Delay |
| Power | 67.968µw | 25.797 µw | Power |

Carry = A AND B = A.B

Transistor level for the system built and simulated for all basic gates uses. Fig. 5 shows transistor level for a 2-bit adder

# System flow and simulation

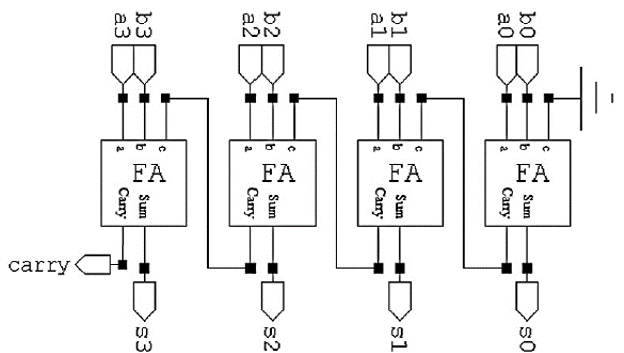
Carry Ripple adder for N-bit adder, there must be N number of full adder circuits. The carry-out of each full adder is the carry-in of the next most full adder. Fig. 6 shows the block diagram for the adder. To build schematic, layout and run simulations please watch this video ([Link](https://www.youtube.com/watch?v=LNHdnLLSHz8&fbclid=IwAR0ItOW9thSKAnfNyrCWxqC3UQfxE7L-YgaVjDsZBcBrETnzCjoC2FcLwJM)).

Table 3 shows the summary of performance, power, and area compared with other designs referenced as [7]. This was obtained by using the Microwind tool. To ensure system performance each part was simulated and testes as standalone before system interconnection. A schematic, layout, and time simulation for each part was demonstrated. An inverter gate schematic diagram completed using Dsch software while the Microwind tool used to generate the layout as shown in Fig. 7-a & Fig. 7-b respectively. A simulation carried out on Microwind software which generates the timing diagram of the inverter, refer Fig. 7-c.



**Fig. 5. transistor level for 2-bit adder.**

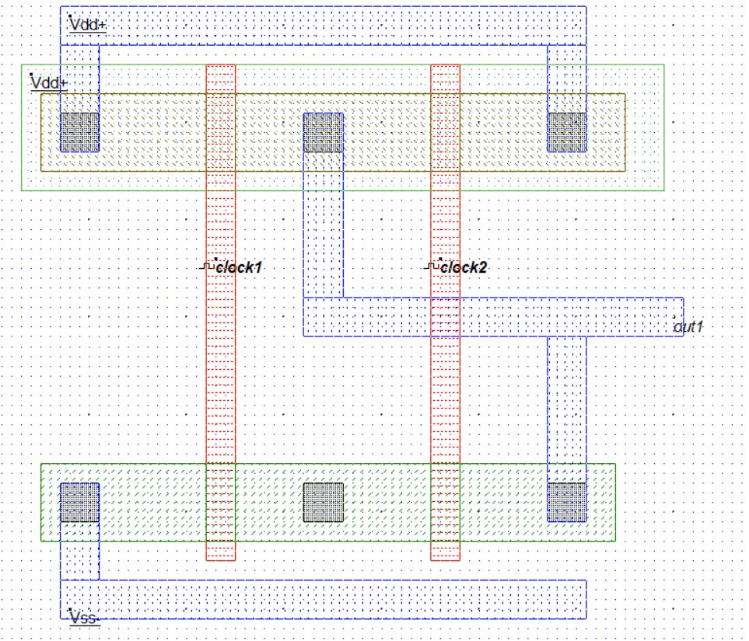
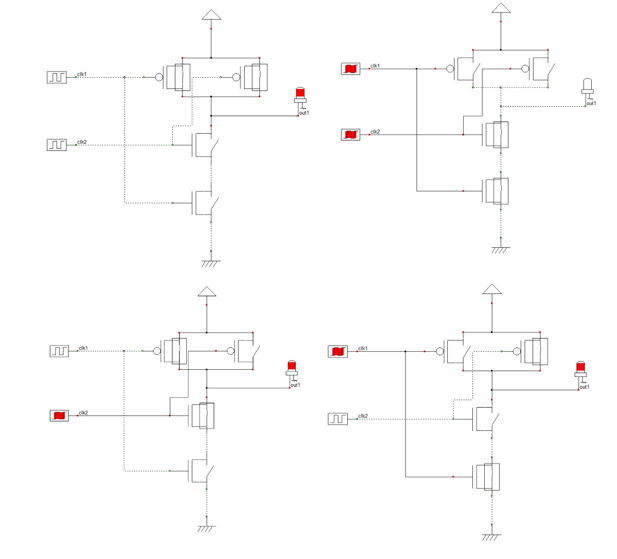
The 2-inputs CMOS NAND gates schematic diagram, layout, and time diagram shown in Fig. 8-a, 8-b & 8-c respectively.



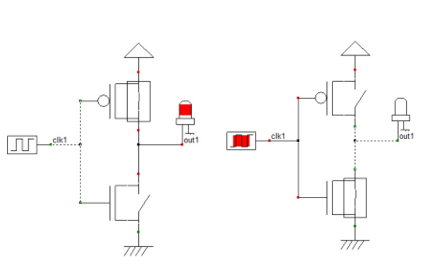
**Fig. 6.** 4-bit carry ripple adder block diagram.

**Table 2.** number of used gates in both designs

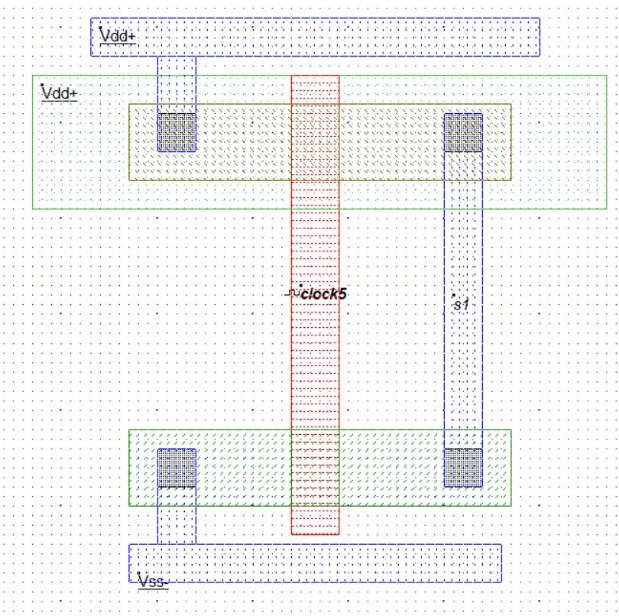
|  |  |  |  |
| --- | --- | --- | --- |
|  | *Nu. of transistor per gate* | *Common design used gates Nu.* | *Introduced design used gates Nu.* |
| *2-input NAND* | *4* | *3* | *7* |
| *3-input NAND* | *6* | *5* | ***0*** |
| *4-input NAND* | *8* | *1* | ***0*** |
| *CMOS inverter* | *2* | *3* | *5* |



**Fig. 8-a.** NAND gate Schematic. **Fig. 8-b.** NAND gate Layout.



**Fig. 7-a.** Invertor Schematic

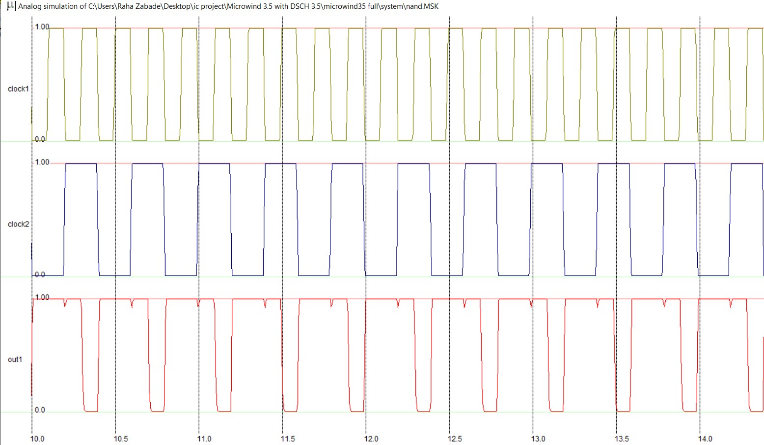


**Fig. 7- b.** Invertor Layout.



**Fig.7. (c)** Invertor Timing diagram.

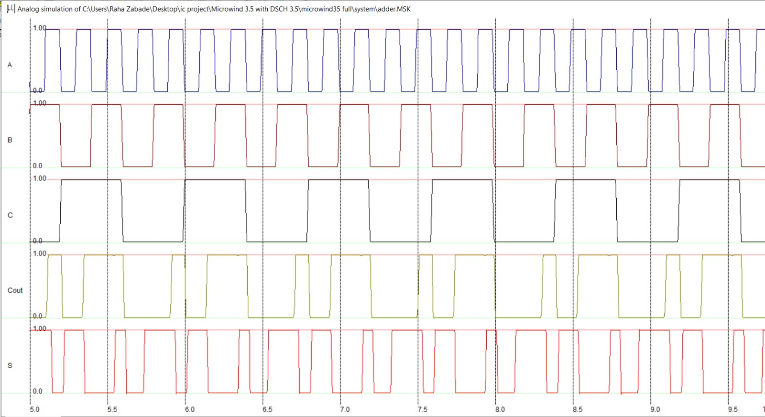
‘



**Fig. 8-c.** NAND gate Timing Diagram.

2-bit binary parallel ripple carries adder constructed as per the design proposed in Fig.4. The adder schematic diagram is shown in Fig.5 Adder time diagram shown in Fig.9 and Adder Layout shown in Fig.10 respectively.

The 2-bit binary parallel ripple carries adder time diagram simulated again using DSCH software, shown in Fig. 11. The graph was matching with Microwind software output presented in Fig. 9.



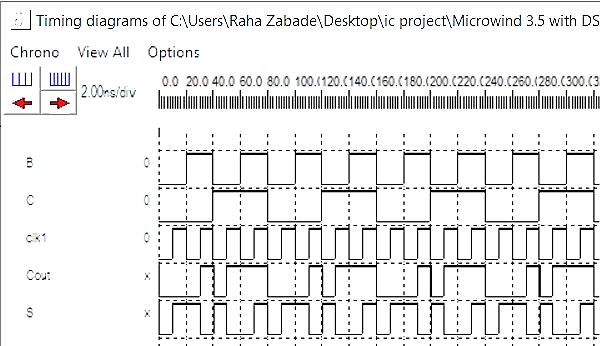
**Fig. 9.** Adder Timing Diagram.



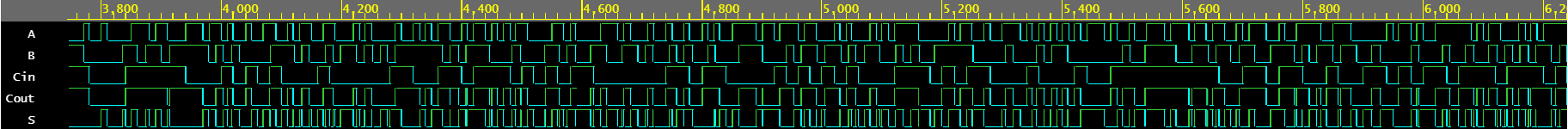
**Fig. 10.** Adder Layout.

Fig.12 shows the waveform for the testbench from the EDA playground.

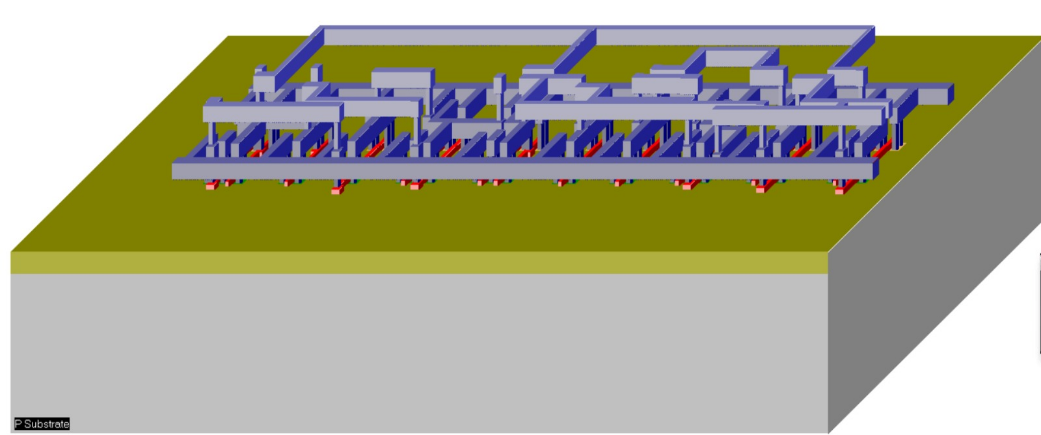
The area of the system is 59.3µm2. the worst-case scenario delay for the whole system is 207ps. Three-dimension model for the 2-bit binary parallel ripple carries adder shown in Fig. 13.



**Fig. 11.** 2-bit binary parallel ripple carry adder time diagram using DSCH software.



**Fig. 12.** wave form for the testbench from EDA playground.



**Fig. 13.** Three-dimension model for the 2-bit binary parallel ripple carry adder

# Design results and Comparison:

First, the number of transistors used to build the circuit dropped from 56 to 38 pieces which mean 32% off power consumption reduction. Second, the variety of gate types reduced which incorporates smaller circuit sizes. Furthermore, a more compact design reduces the adder delay. Table 2 shows the number of gates used in each design and Table 3 shows the summary of performance, power, and area compared with other designs that are referenced as [7].

# Conclusion:

This paper helped students and any person involved in the VLSI design to design an adder from scratch using free tools. So, in this paper, we did

not show the procedure, but also on how to optimize the design. So, we present proposal circuit technique to be used as a guide for an undergraduate to design a 2-bit binary parallel ripple carry adder using CMOS NAND Gates with Microwind (Process 65nm). In summary, by using optimized design for NAND gate the adder with high speed and less delay by making it predict the carry was built. Also, the ripple carries adder doesn’t need area and power compared to another adder. The area of the system is 59.3µm2. the worst-case scenario delay for the whole system is 207ps.

# References

[1] Z. Huang, High-level optimization techniques for low-power multiplier design (Ph.D. thesis), University of California, 2003.

[2] Y. Jiang, A. Al-Sheraidah, Y.Wang, E. Sha, J.-G. Chung, A novel multiplexer-based low-power full adder, IEEE Trans. Circ. Syst. II: Exp. Briefs 51 (2004) 345–348.

[3] C.-H. Chang, M. Zhang, J. Gu, A novel low power low voltage full adder cell, in: 3rd International Symposium on Image and Signal Processing and Analysis, IEEE, New York, 2003, pp. 454–458.

[4] N. Weste, D. Harris, CMOS VLSI Design, A Circuit and System Perspective,

Addison-Wesley, USA, 2004.

[5] M. Ito, D. Chinnery, K. Keutzer, Low power multiplication algorithm for

switching activity reduction through operand decomposition, in 21st. International Conference on Computer Design, 2003, pp. 21–26.

[6] Informer Technologies, Inc., (2020, August 6). Microwind Lite 3.5 Download, Retrieved from <https://microwindlite.software.informer.com/3.5/>

[7] Arif Ul Alam, Nishatul Majid and S. K. Aditya, Layout Design of a 2-bit Binary Parallel Ripple Carry Adder Using CMOS NAND Gates with Microwind. Dhaka. 20. 08. 2011.

[8] A. Kishore Kumar, D. Somasundareswari, V. Duraisamy, T. ShunbagaPradeepa, Design of Low Power Multiplier with Energy Efficient Full Adder Using DPTAAL, Hindawi Publishing Corporation, 2013, DOI:10.1155/2013/157872.

[9] R. Zimmermann, W. Fichtner, Low-power logic styles: CMOS versus passtransistor logic, IEEE J. Solid-St. Circ. 32 (7) (1997) 1079–1090.

[10] C. Efstathiou, Z. Owda, Y. Tsiatouhas, New high-speed multioutput carry

look-ahead adders, IEEE Trans. Circ. Syst. II 60 (10) (2013) 667–671.

[11] K.V. RAJENDRA PRASAD, G. NARESH, T. RAVISEKHAR, Layout Design of 4-bit Ripple Carry Adder Using NOR and NAND Logic. Tirupati, INDIA October 2013.

[12] M. Morris Mano, Michael D. Ciletti, Digital Design with an Introduction to the Verilog HDL. Fifth Edition. California State University, Los Angeles. 1991.

[13] Kang, Sung-Mo (Steve), and Leblebici, Yusuf. “CMOS Digital Integrated Circuits Analysis and Design”, Third Edition, McGraw-Hill, 2002.

[14] Neeraj Kumar Misra, Mukesh Kumar Kushwaha, Subodh Wairya, Amit Kumar, Cost-Efficient Design of Reversible Adder Circuits for Low Power Applications. Lucknow, UP. 19, May 2015.

[15] Sreedevi, M., and P. Jeno. Paul (2010). “Design and Optimization of a High-Performance Low-Power CMOS Flex Cell “, International Journal of Signal System Control and Engineering Application. Tamil Nadu, India.2010.

[16] K. Hwang, Computer Arithmetic: Principles, Architecture, and Design, Wiley, New York, NY, USA, 1979.

[17] Y. Liu, S. Furber, The Design of an Asynchronous Carry-Lookahead Adder Based on Data Characteristics: Integrated Circuit and System Design. Power and Timing Modeling, Optimization and Simulation, 15th International Workshop, PATMOS 2005, Leuven, Belgium, Springer, 2005.

[18] O. Kavehei, M. Rahimi Azghadi, K. Navi, A.P. Mirbaha, Design of robust and high-performance 1-bit CMOS Full Adder for nanometer design, IEEE Comput. Soc. Annu. Symp. VLSI ISVLSI 08 Montpellier Fr., pp. 10–15, 2008.

[19] L. Junming, S. Yan, L. Zhenghui, W. Ling, A novel 10-transistor low-power high-speed full adder cell, in 6th International Conference on Solid-State and Integrated-Circuit Technology, 2001.

[20] S.-M. Kang, Y. Leblebici, CMOS Digital Integrated Circuits Analysis, and Design, third ed., Tata McGraw-Hill, 2003.

[21] G.A. Ruiz, New static multi-output carry look-ahead CMOS adders, IEEE Proc.Circ. Dev. Syst. 1997.

[22] Mansi Jhamb, Garima, Himanshu Lohani, Design, implementation, and performance comparison of multiplier topologies in power-delay space. Dwarka, New Delhi, India.2016.