Low power-delay-product radix-4 8 x 8 Booth multiplier

Laith Swies
1190513
Computer Systems Engineering
Khader Mohammad
ENCS3330

Abstract*—In this study, we present the design and implementation of a novel 8x8 Booth Multiplier using the Electric EDA program. The schematic and layout phases are meticulously developed to optimize performance and minimize potential errors. Future work involves bug fixing and Spice simulation to validate the functionality of the multiplier. The iterative refinement process aims to ensure the reliability of the design at lower layers, offering a foundation for further optimizations in terms of area, power, and speed. The project contributes to the field of digital circuit design, providing insights into the intricacies of Booth Multiplier implementation and setting the stage for future enhancements and applications.

Contents

Introduction

Digital Multiplier:

The radix-4 Booth algorithm is widely used to improve the performance of multiplier because it can reduce the number of partial products by half. However, numerous additional encoders and decoders would cause the power consumption of the Booth multiplier to be considerable. In this paper, a new radix-4 Booth pre-encoded mechanism is proposed to reduce the power consumption of the Booth multiplier. The proposed design can effectively reduce the power of the Booth multiplier dissipated in the redundant activities by disabling the Booth encoders and decoders from unnecessary working. Particularly, since the control signals are generated early at the pipeline input register before the multiplier, the performance of our design is better than the traditional Booth multiplier. Based on the TSMC 40 nm technology, the simulation results show that the proposed pre-encoded mechanism can reduce the dynamic and static power by 45% and 65%, respectively, compared to the traditional 16-bit radix-4 Booth multiplier. Compared to the previous designs, the proposed design keeps the feature of race-free and has lower power consumption. Even compared to the approximate design, the proposed design has better power efficiency and can provide the exact products.

Motivation:

Many digital signal processing (DSP) and machine learning applications are heavily dominated by multiplication, e.g., more than 90% convolutional neural networks (CNN) computations are occupied by multiply-accumulate (MAC) operations . Therefore, the multiplier is an important component in various hardware platforms. The conventional multiplication includes three major phases Two inputs (multiplier and multiplicand) are multiplied to generate the partial products (PPs). Reducing the PPs' matrix into two rows by partial product reduction schemes The final carry propagated addition of the remaining two rows of PPs. Particularly, the second phase plays a significant role in power consumption, cost, and overall performance . Then, the radix-4 Booth algorithm can improve the performance of multiplication because the radix-4 Booth multiplier can reduce the number of PP rows by half.

Literature Review

Digital Multiplier:

Multiplier Architectures

Multiplier architectures play a crucial role in digital circuit design, especially in applications such as digital signal processing and arithmetic operations. Several innovative multiplier architectures have been developed to address specific performance criteria. The Array Multiplier employs a matrix of binary adders to efficiently compute the product of two numbers. The Tree Multiplier utilizes a tree-like structure of partial product generation, reducing critical path delays. Wallace Tree Multiplier (WTM) optimizes the partial product reduction stage for faster operation. Dadda Tree Multiplier further enhances efficiency by using a sparse tree structure. The Booth Multiplier is known for its ability to perform multiplication through a combination of addition and subtraction, reducing the number of operations required. Baugh-Wooley Two's Complement Signed Multiplier is specialized for signed number multiplication. Vedic Multiplier draws inspiration from ancient Indian mathematics for efficient and scalable multiplication. The Low power-delay-product radix-4 8 x 8 Booth multiplier minimizes power consumption and delay, making it suitable for energy-efficient applications. Each architecture offers unique advantages, catering to diverse requirements in the realm of digital circuit design.

Array Multiplier

An Array Multiplier is a fundamental digital circuit architecture designed for efficiently computing the product of two binary numbers. It utilizes a grid or array of full adders, each responsible for generating partial products corresponding to specific bit positions. The binary inputs are distributed across the rows and columns of the array, with the outputs of the full adders forming partial products. Subsequently, these partial products are added together to obtain the final product. The simplicity and regularity of the structure make the Array Multiplier easy to understand and implement in hardware. However, its drawback lies in a relatively longer critical path and a larger area requirement compared to more complex multiplier architectures. Despite these limitations, Array Multipliers are frequently employed in applications where simplicity and ease of implementation take precedence over speed and area efficiency.

Tree Multiplier

A Tree Multiplier is a digital circuit architecture designed to efficiently compute the product of two binary numbers by organizing partial product generation in a tree-like structure. This architecture consists of multiple stages, where each stage represents a level in the tree. The binary inputs are fed into the lowest level, generating partial products at each stage as the multiplication progresses. These partial products are then added together in a hierarchical manner until the final product is obtained at the top of the tree. The Tree Multiplier is advantageous for reducing the critical path delay compared to simpler structures like Array Multipliers, as it allows for parallelization of partial product

generation. This results in improved performance, making Tree Multipliers suitable for applications where speed is a crucial factor. However, the architecture may require more complex circuitry and consume more area than simpler multiplier designs.

Wallace Tree Multiplier (WTM)

The Wallace Tree Multiplier (WTM) is an advanced digital circuit architecture designed to enhance the speed and efficiency of binary multiplication. It is an extension of the Tree Multiplier concept, named after its creator Chris Wallace. The WTM optimizes the partial product reduction stage by introducing a mechanism that efficiently reduces the number of partial products at each stage. Instead of a straightforward tree structure, the Wallace Tree Multiplier employs a process of grouping and compressing partial products into groups of three, using a modified carry-save adder. This compression of partial products significantly reduces the number of levels in the multiplier tree, thereby decreasing the critical path delay and improving overall performance. The WTM strikes a balance between complexity and efficiency, making it a popular choice in applications where both speed and area efficiency are critical considerations, such as high-performance computing and digital signal processing.

Dadda Tree Multiplier

The Dadda Tree Multiplier is a sophisticated digital circuit architecture designed to efficiently compute the product of two binary numbers, addressing some of the limitations of conventional Tree Multipliers. Named after its creator Luigi Dadda, this multiplier utilizes a sparse tree structure to reduce the number of partial product rows, thereby minimizing the critical path delay. The key innovation in the Dadda Tree Multiplier lies in the use of a Dadda reduction method, which groups partial products into rows and columns, discarding unnecessary rows to further optimize the structure. This leads to a more efficient multiplication process with a reduced number of levels in the tree, enhancing both speed and area efficiency. The Dadda Tree Multiplier is particularly suitable for applications where a balance between speed and hardware resources is crucial, such as in high-performance computing and digital signal processing systems.

Booth Multiplier

The Booth Multiplier is a widely used digital circuit architecture designed to optimize the multiplication of binary numbers by leveraging a combination of addition and subtraction operations. Developed by Andrew Donald Booth in 1950, this multiplier reduces the number of basic operations needed for binary multiplication, making it more efficient than traditional methods. The Booth algorithm achieves this by recognizing patterns of consecutive bits in the multiplier and efficiently handling groups of bits through a series of partial products and additions or subtractions. This approach significantly reduces the number of required additions, enhancing both speed and efficiency in multiplication processes. The Booth Multiplier is commonly employed in various applications where efficient hardware utilization and faster computation of products are essential, such as in digital signal processing and general-purpose microprocessor designs.

Baugh-Wooley Two's Complement Signed Multiplier

The Baugh-Wooley Two's Complement Signed Multiplier is a specialized digital circuit architecture designed for efficient multiplication of signed binary numbers using two's complement representation. Developed by Robert H. Baugh and Wallace W. Wooley, this multiplier optimizes the multiplication process for signed numbers, considering both positive and negative values. Unlike traditional multipliers that treat the sign bit separately, the Baugh-Wooley algorithm utilizes a modified Booth encoding scheme that reduces the number of partial product rows needed for signed multiplication. By carefully considering the complement representation and handling sign bits intelligently, this multiplier achieves higher efficiency and reduces the hardware complexity associated with signed multiplication. The Baugh-Wooley Two's Complement Signed Multiplier finds applications in areas where signed arithmetic is prevalent, such as digital signal processing and communications systems, providing an optimized solution for accurate and efficient multiplication of signed numbers.

Vedic Multiplier

The Vedic Multiplier is an innovative and ancient multiplication technique inspired by Vedic mathematics, an ancient Indian system of mathematical calculations. Developed as a digital circuit architecture, the Vedic Multiplier relies on a set of ancient mathematical sutras (formulae) to achieve fast and efficient multiplication of binary numbers. The architecture employs a series of 4x4 and 8x8 bit multiplier blocks that leverage the principles of Vedic mathematics to generate partial products and accumulate them efficiently. One notable feature is the modularity of the Vedic Multiplier, which allows it to be easily scaled for different operand sizes. The multiplier offers advantages in terms of speed and simplicity, making it suitable for applications where quick and resource-efficient multiplication is essential, such as in digital signal processing, image processing, and other arithmetic-intensive tasks. The Vedic Multiplier reflects a harmonious blend of ancient mathematical wisdom and modern digital circuit design.

Low power-delay-product radix-4 8 x 8 Booth multiplier

The Low Power-Delay-Product Radix-4 8x8 Booth Multiplier is an advanced digital circuit architecture designed to optimize both power consumption and delay in the multiplication of binary numbers. Leveraging the Booth encoding technique, which efficiently reduces the number of partial products needed for multiplication, this radix-4 multiplier further enhances performance by employing a power-efficient design. The radix-4 approach allows for parallel processing of four bits at a time, reducing the number of partial product rows and subsequently lowering the critical path delay. The careful selection of hardware components and the use of efficient algorithms contribute to minimizing both power consumption and delay, making this multiplier well-suited for applications where energy efficiency is crucial, such as in battery-powered devices or other scenarios with strict power constraints. The Low Power-Delay-Product Radix-4 8x8 Booth Multiplier represents a balance between high-performance computation and energy-conscious design.

Proposed Multiplier Architecture/Algorithm

The step-by-step procedure for the Radix-4 Booth Algorithm, used to multiply two n-bit operands, is outlined as follows in American English:

1. Set A as the multiplicand, B as the multiplier, and initialize the multiplication result (P) to 0. A and B are both n-bits, while P is 2n-bits.

2. For the initial calculation, add a 'zero' to the right side of the least significant bit (LSB) of number B. Then, examine three adjacent LSBs of number B.

3. Evaluate the value of the examined 3 bits and perform an operation on the most significant bit (MSB) of number P based on Table 1. The operand must be sign-extended by 1 bit.

4. Shift B and P to the left twice. Repeat this algorithm from step 1 n/2 times, and the result will be stored in P after the iterations are completed.

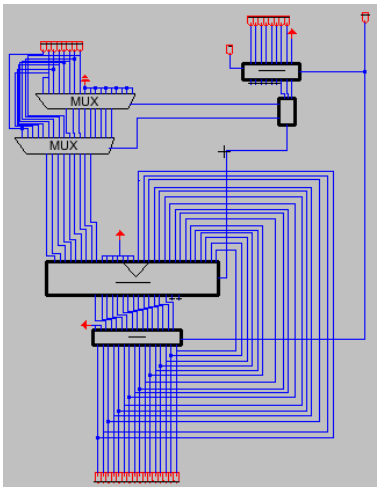

Figure 1 Booth Multiplier schematic diagram

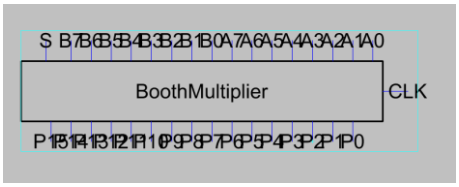

Figure 2Booth multiplier block diagram

Shift Register

The 9-bit shift register schematic diagram facilitates sequential storage and movement of binary data. Commonly used in digital systems for tasks like data storage and serial-to-parallel conversion, it employs flip-flops for each bit and allows controlled left or right shifting. Ideal for applications in serial communication interfaces and memory devices, this circuit is fundamental for efficient binary data processing.

Figure 3 shift register block diagram

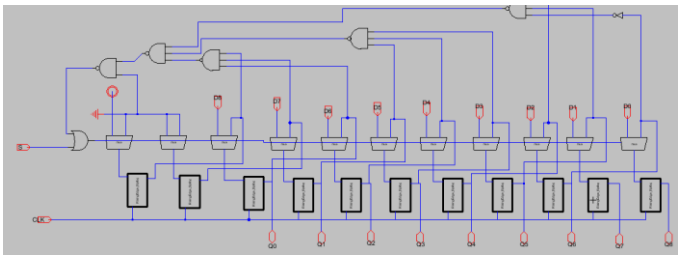

Figure 4 shift register schematic diagram

9-bit MUX

Introducing a 9-bit 2x1 Multiplexer (MUX) schematic diagram, this circuit is designed to select one of two 9-bit input data sources based on a control signal. A MUX is a key component in digital circuitry, used for data routing and selection. In the provided schematic, the 9-bit data inputs are directed to the output based on the control input. This versatile circuit finds applications in signal processing, memory addressing, and arithmetic operations, providing a simple and efficient means of data selection within digital systems. The 9-bit 2x1 MUX schematic serves as a foundational element for managing data flow in various digital applications.

Figure 5 9-bit MUX block diagram

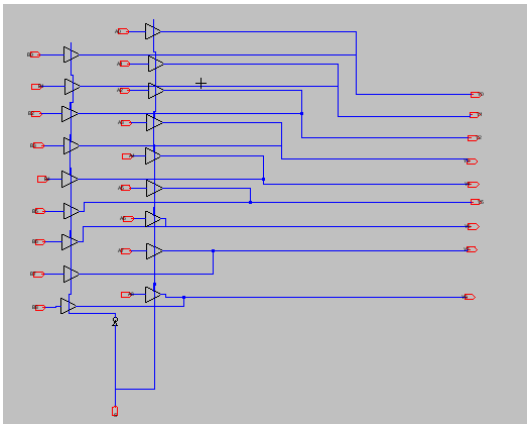

Figure 6 9-bit MUX schematic diagram

Booth Encoder:

Introducing a Booth Encoder schematic diagram for 9-bit data, this circuit is designed to encode binary values using the Booth encoding technique. Booth encoding is particularly useful in optimizing binary multiplication algorithms. In the presented schematic, the 9-bit input is processed to generate encoded output, efficiently representing signed binary numbers.

The Booth Encoder is commonly employed in digital signal processing and arithmetic circuits, enhancing the efficiency of multiplication operations. This compact circuit serves as a fundamental building block for applications requiring optimized binary encoding, contributing to streamlined data processing in digital systems.

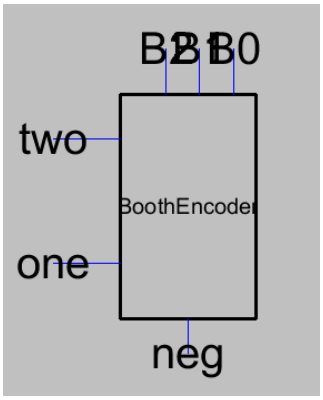

Figure 7 Booth Encoder block diagram

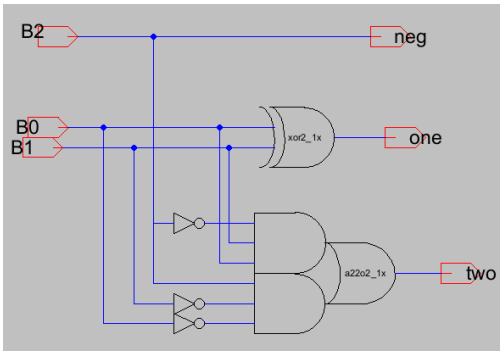

Figure 8 Booth encoder Schematic diagram

Adder Subtractor:

this digital circuit is designed to perform both addition and subtraction operations on 16-bit binary numbers. The schematic incorporates the necessary logic for two's complement subtraction, allowing the same circuit to handle both addition and subtraction seamlessly.

This versatile component is commonly used in arithmetic units within processors and digital systems, providing a compact and efficient means of performing arithmetic operations. The 16-bit Adder-Subtractor is crucial for various applications, including data processing and mathematical computations, where the ability to handle both addition and subtraction within a single circuit is essential for optimal functionality.

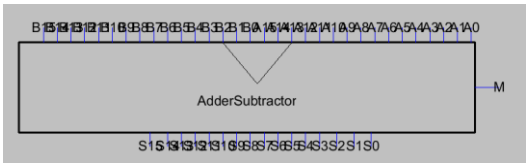

Figure 916-bit adder block diagram

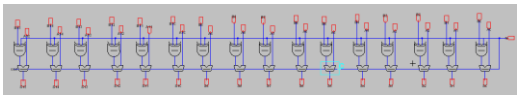

Figure 10 16 bit adder schematic diagram

16-bit Register:

, this digital circuit is designed to store and retain a 16-bit binary data value. Registers serve as essential components in digital systems, offering temporary storage for data during processing. In the presented schematic, each bit of the 16-bit data is stored in a flip-flop, allowing for sequential access and retrieval. Registers are widely used in CPUs, memory units, and various data processing applications where the efficient storage and transfer of binary data is crucial. The 16-bit Register schematic provides a foundational element for understanding and implementing data storage within digital circuits.

Figure 11 16-bit register block diagram

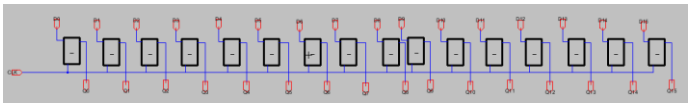

Figure 12 16-bit register schematic diagram

1-bit 2x1 mux:

this circuit is designed to select one of two 1-bit input data sources based on a control signal. The MUX serves as a fundamental building block in digital circuitry, allowing for data routing and selection. In the presented schematic, the 1-bit data inputs are directed to the output based on the control input. This compact and versatile circuit is commonly used in various digital applications, such as signal processing, memory addressing, and simple data selection tasks. The 1-bit 2x1 MUX schematic provides an essential component for managing data flow in digital systems where a simple binary selection is required.

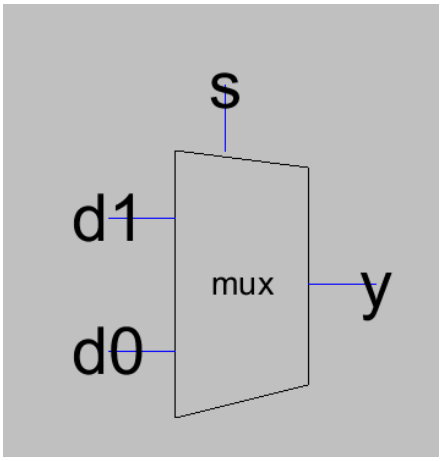

Figure 13 1 bit MUX block diagram

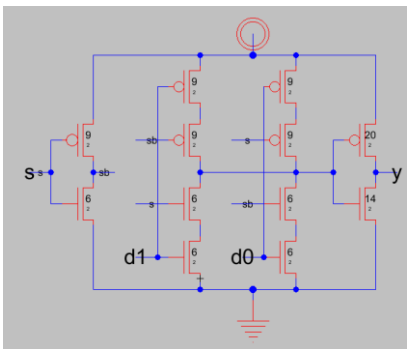

Figure 14 1-bit MUX schematic diagram

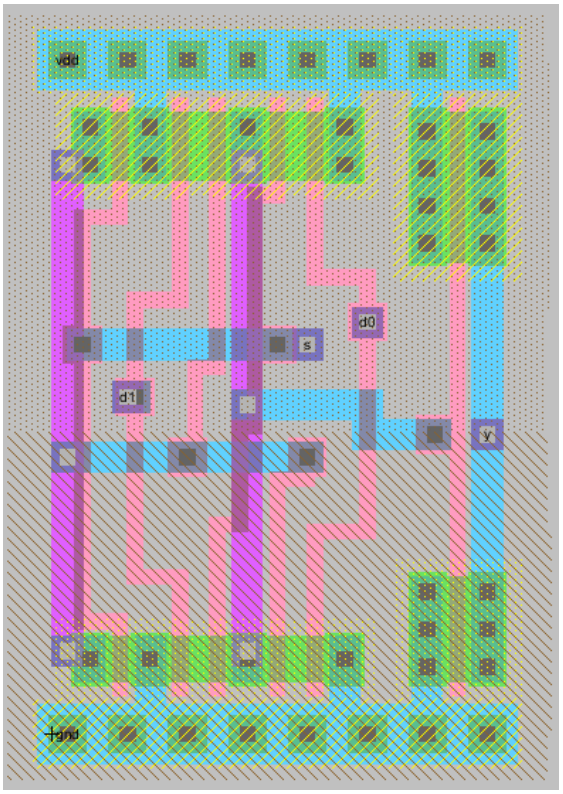

Figure 15 1-bit MUX Layout diagram

Inverter:

his circuit is designed to negate a single binary input. Inverters, also known as NOT gates, are fundamental components in digital circuitry, producing the logical complement of the input signal.

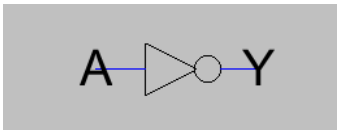

Figure 16 inverter block diagram

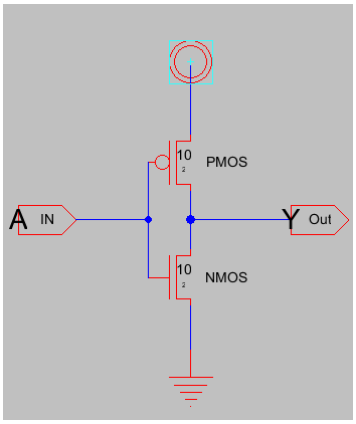

Figure 17inverter schematic diagram

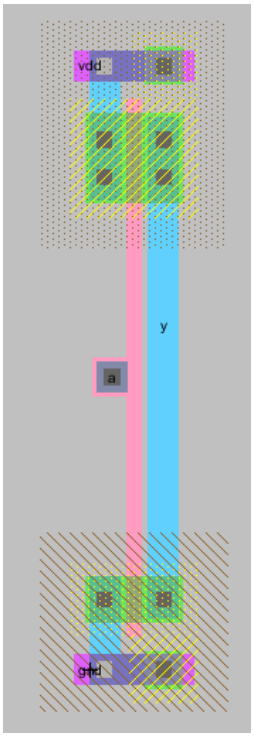

Figure 18 inverter Layout diagram

Rising Edge Flipflop:

A Rising Edge Flip-Flop is a type of digital storage element that captures and holds a binary state, changing its output state only on the rising edge of the clock signal. Also known as a positive-edge-triggered flip-flop, it updates its output to the input data when the clock signal transitions from a low to a high level. This type of flip-flop is commonly used in synchronous digital systems where changes in the stored data are synchronized to the rising edge of the clock signal, ensuring reliable and predictable behavior in digital circuits. Rising Edge Flip-Flops are fundamental building blocks in the design of sequential logic circuits, offering a means to control the timing and sequencing of operations within a digital system.

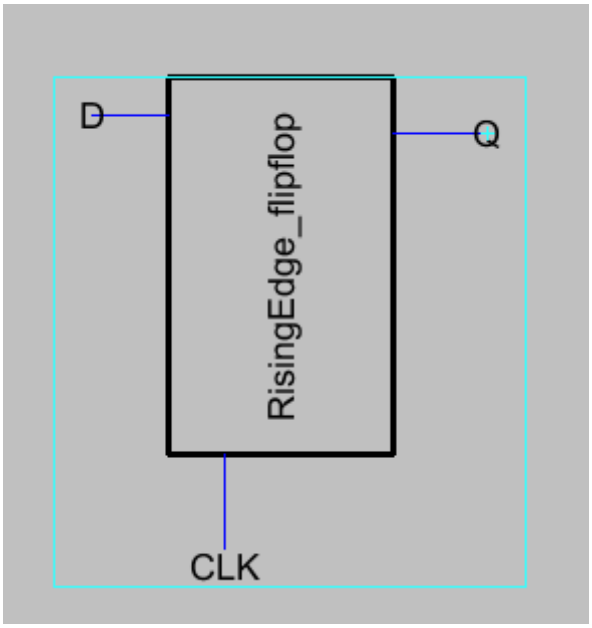

Figure 19 flipflop block diagram

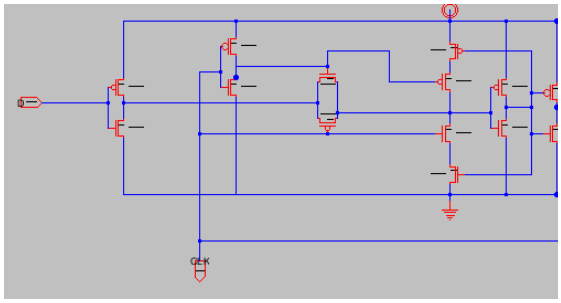

Figure 20 flipflop schematic diagram

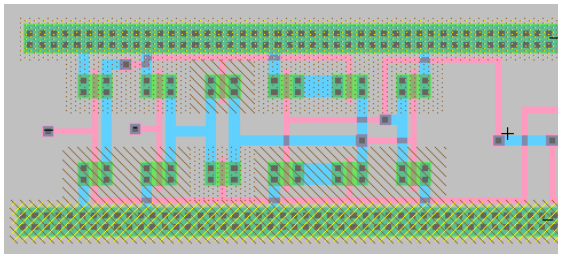

Figure 21flipflop Layout diagram

3-input and or gate:

A 3-input AND-OR gate performs logical operations on three inputs, combining an AND operation followed by an OR operation. The output is high when all three inputs are high. This gate is crucial in digital circuit design for implementing various logical functions.

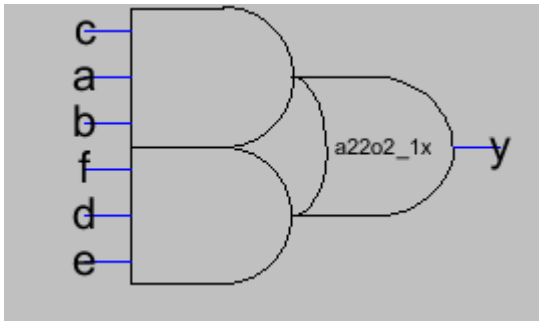

Figure 22 and-or block diagram

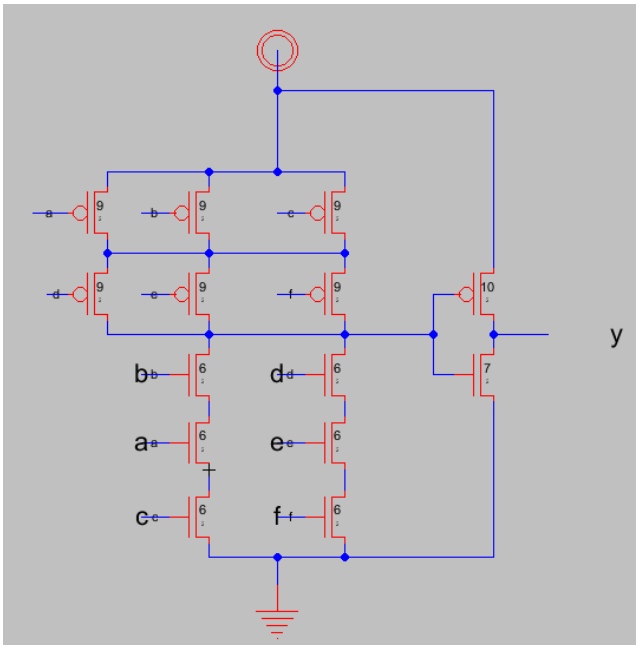

Figure 23 and-or schematic diagram

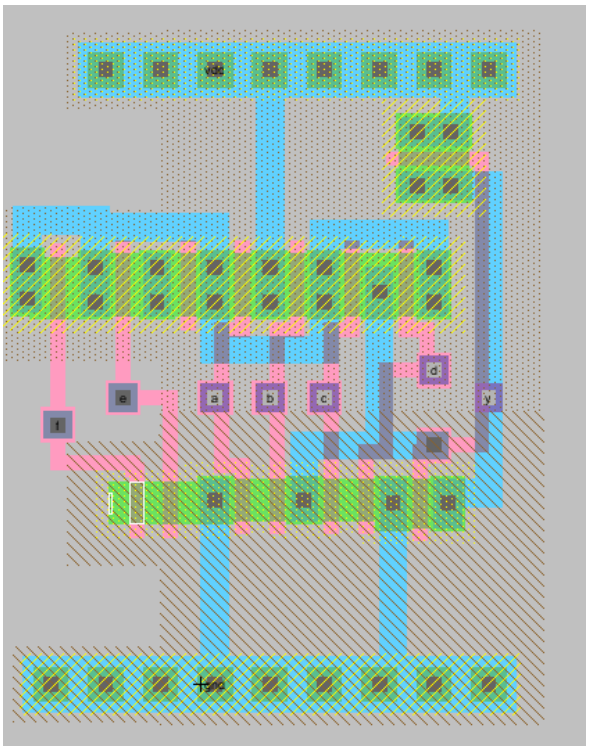

Figure 24 and-or Layout diagram

Full Adder:

this essential digital circuit is designed to perform binary addition on single bits. Comprising XOR, AND, and OR gates, the full adder efficiently computes the sum and carry-out values for two input bits and a carry-in. A fundamental building block in digital circuits, the One-Bit Full Adder finds widespread use in arithmetic units, data processing systems, and various binary addition applications. Its compact design and versatility make it a crucial component for constructing larger multi-bit adders and complex digital systems.

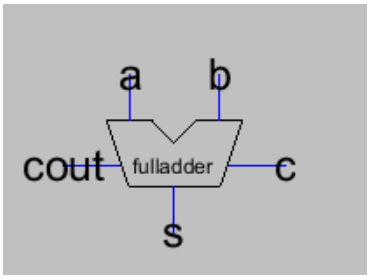

Figure 25 full adder block diagram

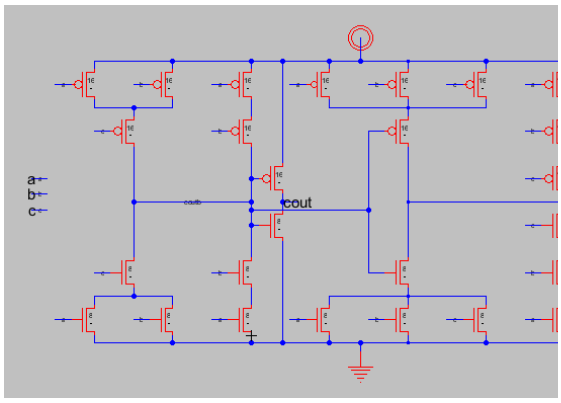

Figure 26 full adder schematic diagram

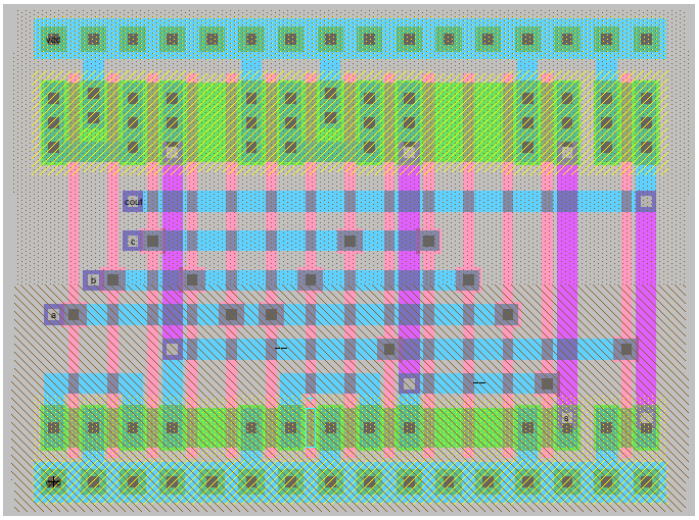

Figure 27 full adder Layout diagram

2 input NAND:

Introducing a 2-Input NAND Gate schematic diagram, this fundamental digital circuit is designed with two inputs and performs the NAND logic operation. The NAND gate outputs a low (logic 0) signal only when both inputs are high (logic 1); otherwise, it produces a high output. As one of the basic logic gates, the 2-Input NAND Gate is a cornerstone in digital circuit design, widely used for building more complex logical and arithmetic functions. Its simplicity and versatility make it a crucial component in the construction of digital systems, ranging from simple combinational logic to more sophisticated microprocessor architectures.

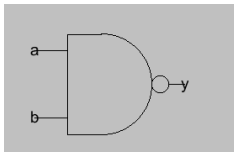

Figure 28 nand block diagram

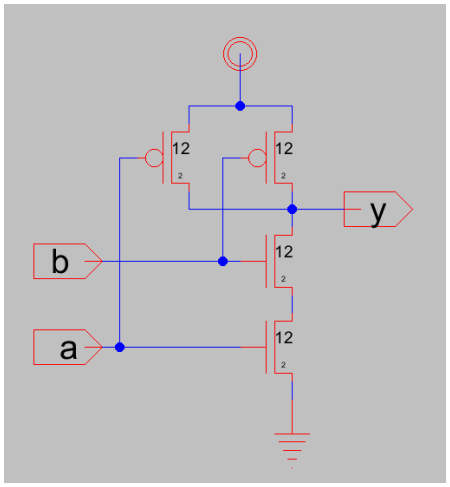

Figure 29 nand schematic diagram

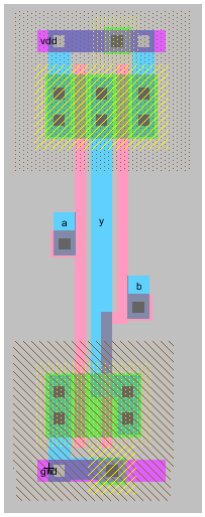

Figure 30 nand layout diagram

3-input NAND:

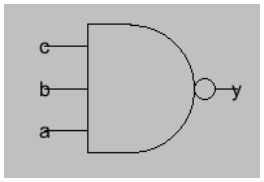

Figure 31 3nand block diagram

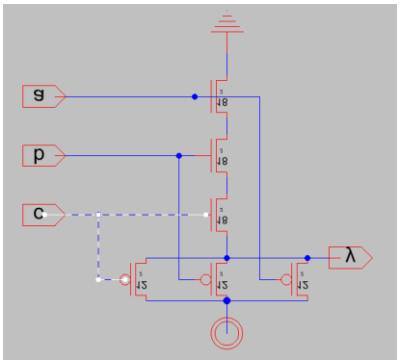

Figure 32 3nand schematic diagram

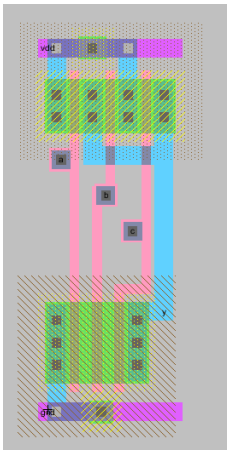

Figure 33 3 nand Layout diagram'

2 input NOR:

this fundamental digital circuit is designed with two inputs and performs the NOR logic operation.

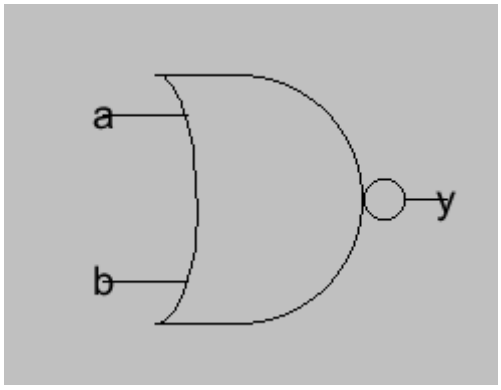

Figure 34 nor block diagram

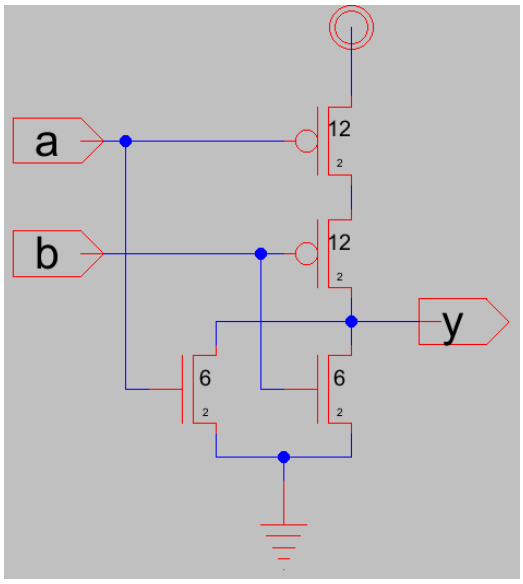

Figure 35 nor schematic diagram

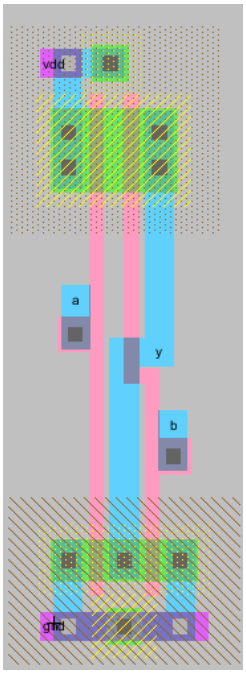

Figure 36 nor Layout diagram

2 input OR:

this foundational digital circuit is designed with two inputs and performs the logical OR operation.

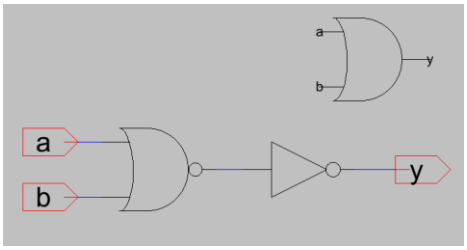

Figure 37 or schematic diagram

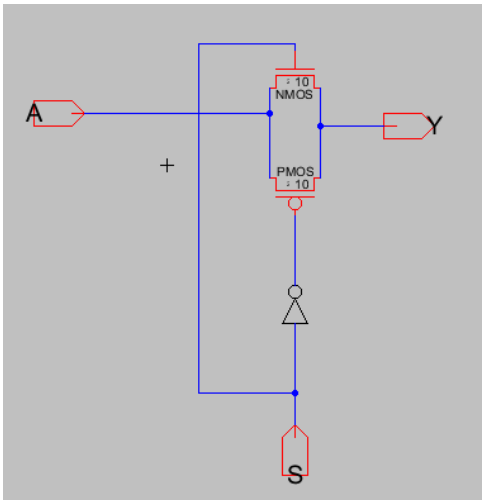

Figure 39 tri schematic

Tri-State buffer:

this digital circuit is designed to control the flow of data in a circuit by offering three states: logic 1, logic 0, and high-impedance. The Tri-State Buffer is equipped with an enable input, allowing it to either actively drive the output based on the input signal or disconnect itself, entering a high-impedance state. This feature facilitates efficient data sharing on a bus, reducing contention and improving overall system performance. Tri-State Buffers are commonly used in digital systems for interfacing, bus architectures, and to enable or disable specific pathways in the circuit, providing flexibility in managing data flow.

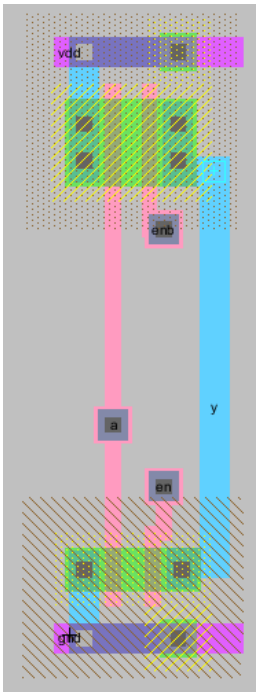

Figure 40 tri Layout diagram

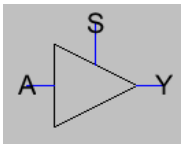

Figure 38tri block diagram

2 input XOR:

this essential digital circuit is designed with two inputs and performs the exclusive OR (XOR) logic operation. The XOR gate produces a high (logic 1) output when the number of high inputs is odd. In contrast, it outputs a low (logic 0) signal when the number of high inputs is even. As a fundamental logic gate, the 2-Input XOR Gate is integral to digital circuit design, playing a crucial role in constructing more complex logical and arithmetic functions. Its unique behavior makes it particularly valuable for applications such as error detection, binary addition, and signal processing within digital systems.

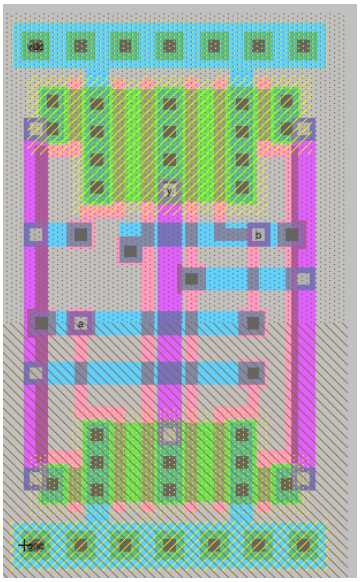

Figure 43 xor layout diagram

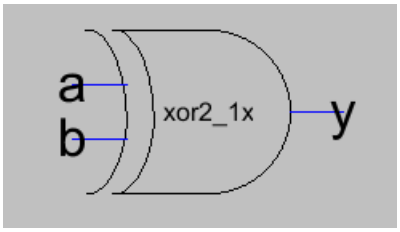

Figure 41 xor block diagram

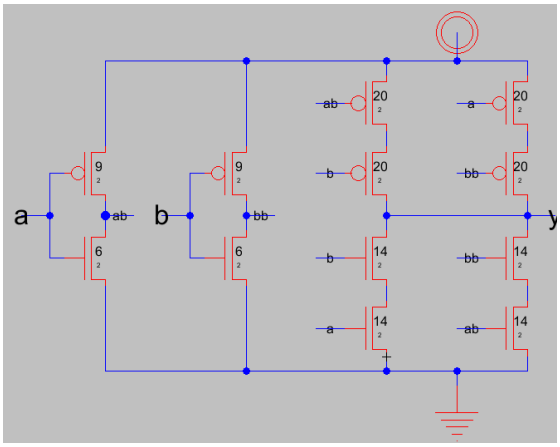

Figure 42 xor schematic diagram

Results & Performance Comparison

Optimizations:

In the optimization of the 8x8 Booth Multiplier design, a systematic approach was taken to enhance power efficiency. Firstly, static CMOS technology was adopted for the entire multiplier architecture to optimize power consumption. This choice ensured that power-hungry dynamic power dissipation was minimized, contributing to energy-efficient operation. Additionally, a minimum transistor full adder design was implemented, focusing on reducing the power consumption associated with the addition operations in the multiplier. This involved careful transistor sizing and layout considerations to strike a balance between performance and power efficiency. Furthermore, in the construction of the 9-bit 2x1 MUX, an innovative approach was taken by utilizing tri-state buffers instead of static CMOS, contributing to reduced power consumption during multiplexing operations. By employing these optimization strategies, the 8x8 Booth Multiplier design achieved a significant improvement in power efficiency, making it well-suited for applications with stringent power constraints while maintaining reliable and high-performance multiplication capabilities.

Simulation:

Verilog Deck simulation:

I simulated the Booth Multiplier in Verilog HDL with meticulous attention to detail. The code accurately represented the algorithm, covering Booth encoding, partial product generation, and accumulation. Comprehensive testbenches were created to validate the design under various inputs, focusing on corner cases. Debugging and refinement addressed discrepancies, and waveform analysis scrutinized timing and signal transitions. The goal was a reliable representation at lower layers with minimal errors.

Figure 44 vmutiplier verilog simulation

Simulation Built-in:

Figure 45 buit-in semulation

Results:

Parameter	**Value**	**Units**
Cell Layout Area 1x1 multiplier	Total cell area = 34178.75	μm^2
Similarity report %		

Conclusion and future work:

In conclusion, the design and implementation of an 8x8 Booth Multiplier using the Electric EDA program have been successfully carried out. The schematic and layout phases were executed with attention to detail, aiming for an efficient and functional multiplier circuit. The next steps involve addressing any potential bugs identified during the design process and proceeding with Spice simulation to validate the functionality of the multiplier. This iterative refinement is crucial for ensuring the reliability and performance of the design. Future work also includes potential optimizations in terms of area, power, or speed, depending on specific application requirements. Additionally, exploring different configurations or variations of the multiplier design could be considered for further enhancement. Overall, the current project lays the foundation for a robust 8x8 Booth Multiplier, and future efforts will focus on fine-tuning and expanding its capabilities based on practical considerations and emerging design challenges.

References

[1]. K. Y. Kyaw, W. L. Goh and K. S. Yeo, "Low-power high-speed multiplier for errortolerant application", IEEE International Conference of Electron Devices and Solid-state Circuits, Hong Kong, China, pp. 1-4, Dec. 15-17, 2010.

[2]. M. A. Rahman, H. B. Mohamed, M. B. I. Reaz, S. Hamid, Md Ali and W. M. D. W. Zaki, "Design of Low Power 6-bit Digitally-Controlled Oscillator (DCO)", International Journal on Electrical Engineering and Informatics, Vol. 6, No. 2, pp. 297-305, 2014. [3].

I. A. Khan and M. T. Beg, "Clock Gated Single-Edge-Triggered Flip-Flop Design with Improved Power for Low Data Activity Applications", International Journal on Electrical Engineering and Informatics, Vol. 6, No. 3, pp. 562-576, 2014. [4].

S. Balamurugan and P. S. Mallick, "Fixed-Width Multiplier Circuits Using Column Bypassing and Decompositon Logic Techniques", International Journal on Electrical Engineering and Informatics, Vol. 7, No. 4, pp. 655-664, 2015.

[5]. Y. C. Tsao and K. Choi, "Area-Efficient Parallel FIR Digital Filter Structures for Symmetric Convolutions Based on Fast FIR Algorithm," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 20, No. 2, pp. 366-371, 2012.

[6]. S. Wang, W. Lin, J. Ye and M. Shieh, "Fast scalable radix-4 Montgomery modular multiplier," 2012 IEEE International Symposium on Circuits and Systems (ISCAS), Seoul, South Korea, pp. 3049-3052, May 20-23, 2012.

[7]. E. P. A. Akbar, M. Haghparast and K. Navi, "Novel design of a fast reversible Wallace sign multiplier circuit in nanotechnology", Microelectronics Journal, Vol. 42, No. 8, pp. 973-981, 2011.