ISSN: 2321-2152

IJMECE

International Journal of modern electronics and communication engineering

E-Mail
editor.ijmece@gmail.com
editor@ijmece.com

www.ijmece.com

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

DESIGN OF LOGICALLY OBFUSCATED N-BIT ALU FOR ENHANCED SECURITY

MRS. RUPALI TOSHNIWAL[1], SANJANA THAKUR[2], MOHAMMED BAQAR QUADRI[3], MOHAMMAD ABU NASER[4], MOHAMMAD ABDUL KAREEM GHORI [5]

Assistant professor [1], Dept of ECE, Lords Institute of Engineering and Technology, Hyderabad, TS

BTech Student[2, 3, 4, 5], Dept of ECE, Lords Institute of Engineering and Technology, Hyderabad, TS

Abstract: Technology in this digital age relies on A.L.U.'s activities to determine system performance. Since an Arithmetic Logic Unit (ALU) is a basic component of any Central Processing Unit (CPU), its necessity is just as significant as that of a computer. Since very few devices exist without an ALU, encryption of an ALU is therefore absolutely necessary for the device's security.

In this study, an n-bit ALU is designed using HDL, a hardware description language that is physically modelled to increase the device's flexibility and reusability. Additionally, the obfuscation logic is suggested to increase its security without sacrificing the device's optimum efficiency and proper operation with quantum dot cellular automation. Verified and analyzed by Xilinx, the results demonstrate that the obfuscation module may be included without significantly increasing the area or power overhead of the ALU, which is implemented in VERILOG HDL and generated using Xilinx.

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

I. INTRODUCTION:

Battery-operated and power-sensitive applications have grown significantly as a result of the current extraordinary combination of chip size reduction and circuit expansion, which has fueled the development of the new sector of low power electronics. Since static power reduction at the architectural level will soon dominate the overall amount of power dissipated in SOCs (System On Chip), we have focused on this topic in our article. To maximize the static power being dissipated, we have suggested synthesizing the POWER GATING TECHNIQUE, namely the fine-grained approach. When not in use, the inputs to the gates are blocked using NMOS in this method, which reduces needless input usage and significantly lowers power consumption. Therefore, the main idea of our entire study is to reduce static power at the architectural level, starting with 1 bit and going up to 8 bits, while also reducing power consumption. An essential component of any digital system is a processor. Additionally, one of a microprocessor's primary parts is an ALU. To put it simply, the CPU functions as the brain of any system, and the ALU functions as the brain of the CPU. Thus, it is a computer's brain. They feature meticulously optimized structures and are made up of quick dynamic logic circuits. The CPU is responsible for a large amount of a processor's overall power usage. Due to its greatest clock speed and constant activity, the ALU also contributes to one of the processor's highest power-density areas, which causes thermal hotspots and abrupt temperature changes within the execution core. As a result, this greatly encourages us to create energy-efficient ALUs that meet high performance standards while lowering average and peak power dissipation. [1,2] An ALU is essentially a combinational circuit that operates logically and arithmetically on a pair of n-bit operands, such as A [31:0] & B [31:0] for 32 bits. An 832-bit ALU's typical internal structure is displayed.

platforms, the proposed ALU is rigorously evaluated for performance, area, and energy metrics. The goal is to demonstrate the viability of combining traditional computation with modern design innovations to meet the growing demands of next-generation digital systems.

II. SCOPE OF THE PROJECT:

1. Design and implementation of a fully functional N-bit ALU supporting basic arithmetic and logical operations.
2. Integration of key-based obfuscation **logic** (e.g., XOR/XNOR gates) to secure the functionality against tampering or unauthorized access.
3. Simulation and testing of the obfuscated ALU using hardware description languages like Verilog or VHDL to ensure correctness and security.
4. Performance analysis in terms of area, delay, and power to evaluate the impact of obfuscation on system resources.
5. Security analysis to demonstrate the ALU's resistance to reverse engineering and functional attacks.

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

6. Targeted applications include secure hardware design for embedded systems, IoT devices, and processors where logic protection is crucial.

III. PROPOSED SECURED *N*-BIT ALU

The inputs A, B, and Cin are subjected to arithmetic and logical operations by the ALU. Based on the selection value (S1S0), as indicated in Table 1, the mux allocates any one of the calculated values to the output (result).There are n complete adders in the adder and subtractor structure. One OR gate and two half adders make up a full adder. Behavioral modeling is used to create the comparator. The multiplier is made up of the necessary number of AND gates, structural modeling of half adders and full adders.The full adder, half adder, OR, and AND gates are the core building blocks of the ALU.

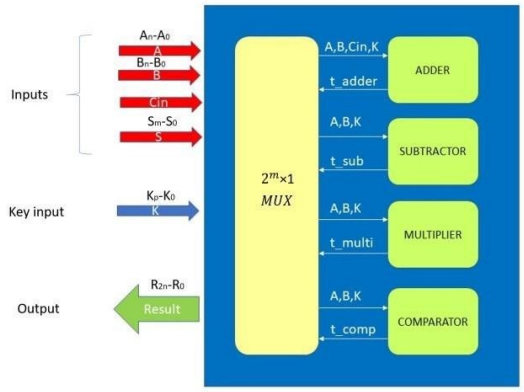


FIG 1:System overview

S_1	$S0$	Function
0	0	Addition
0	1	Subtraction
1	0	Comparison
1	1	Multiplication

TABLE 1:Alu function

IV.ENCRYPTION LOGIC

Structural modeling is used in the design of an n-bit ALU, increasing the circuit's flexibility and reusability and broadening its use. By creating an encryption logic for a module with more connections and influence, the generalized n-bit ALU is encrypted. That impacts the majority of the outputs on the circuit at the following level. This aids in identifying the node that contributes or influences the output the most. Key gates are placed in these nodes after they are determined to be the critical nodes. Each ALU submodule is made up : With the exception of the comparator, all adders are common. As a result, the entire adder exhibits the ALU's encryption. One OR gate and two half adders make up a full adder. The half adder's output is sum and carry. Therefore, as seen in FIG. 3, we will introduce two key gates, key gate 1 (KG1) and key gate 2 (KG2), to encrypt the total and carry of the first half adder. The encryption logic for KG1 and KG2 is provided in Table 2 and Table 3, respectively.

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

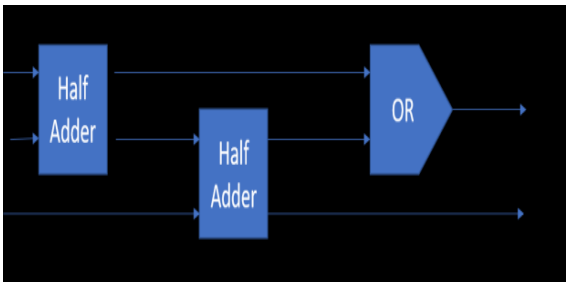


FIG 2: Full adder

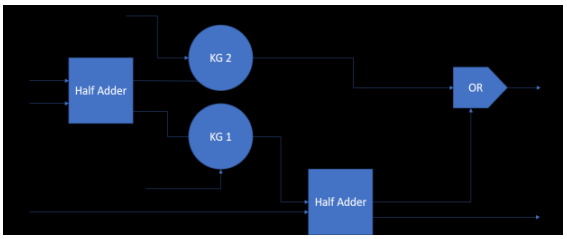


FIG 3: Encrypted Full Adder

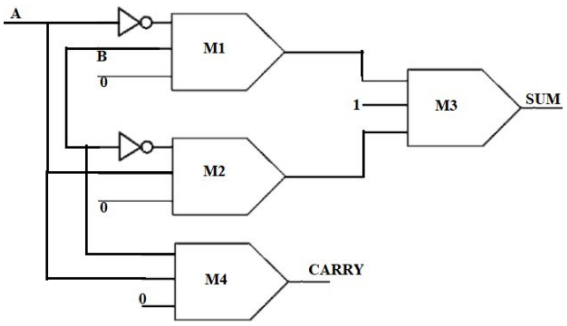


FIG 4:Half adder using QCA majority gates

S_1	K_0	O_1
0	0	S^-
0	1	S
1	0	S^-
1	1	S

TABLE 2:ENCRYPTION LOGIC FOR KG1

C_1	K_1	O_2
0	0	C
0	1	C^-
1	0	C
1	1	C^-

TABLE3:ENCRYPTION LOGIC FOR KG2

Arithmetic Units:- Designing a low power, high performance system will be made easier by using quick and effective adders in the arithmetic logic unit. The internal hardware of other operations, such subtraction and multiplication, is nearly identical to addition hardware, although it is nevertheless used in their operations. For potential application in ALUs, a number of adder families have previously been proposed that trade off power, area, and speed. The implementation of a dynamic adder is necessary due to the ALU's performance criticality. The most efficient adders in terms of transistor count, speed, and power consumption are those in the dynamic logic family. The design of a three-bit adder utilizing complementary logic is covered in this paper. The subtractor unit is implemented using the same adder unit. This conserves space and makes use of the hardware we currently have for the adder.

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

Four primary arithmetic units are embedded in the system to perform core computational functions:

1. To determine the product of two binary values, a binary multiplier can be utilized as an electronic circuit in digital electronics, such as computers. Binary multipliers use a carbon-copy of the standard multiplication process, whereby the multiplicand is multiplied by each multiplier bit starting with the least significant bit. A 2-bit binary multiplier can be implemented using two half adder (HA) modules. A digital multiplier can be operated using a variety of computer arithmetic computations. Many of these methods involve calculating a series of partial products and then adding up all of the partial products that are produced. A 1 fig 5

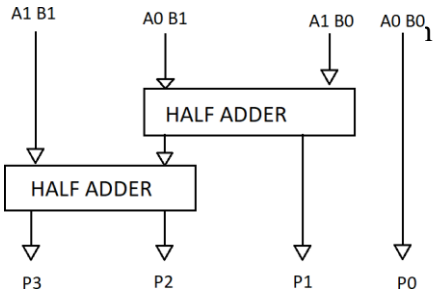


FIG 5: 2x2 Binary Multiplier

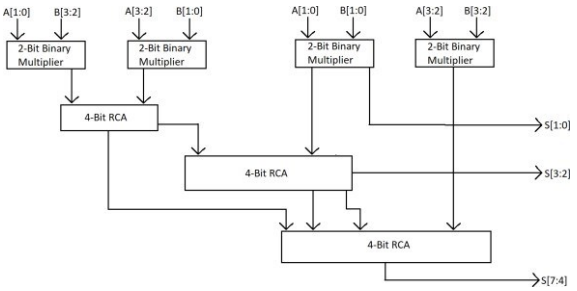


FIG 6: 4-Bit Vedic Multiplier.

2. Standard AND and Ex-ORgates can be used to create a simple Binary Adder circuit that enables us to "add" two single-bit binary values, A and B. According to the guidelines for binary addition, adding these two digits yields two outputs: the SUM of the addition and the CARRY or Carry-out, (COUT) bit. Arithmetic and counting circuits are among the primary applications for the Binary Adder. The two denary (base 10) numbers below can be added simply.

123	A	(Augend)
+ 789	B	(Addend)
912	SUM	

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

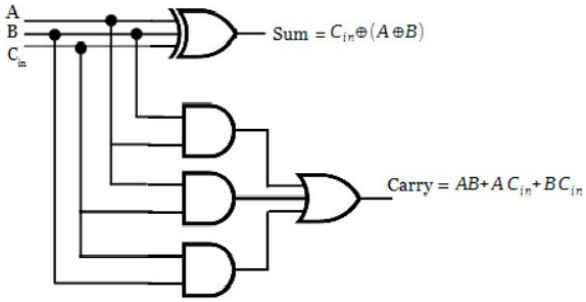


FIG 7:full adder circuit diagram

3. A combinational circuit that compares two digital or binary numbers to determine whether one is equal to, less than, or greater than the other is called a magnitude digital comparator. We rationally create a circuit with two inputs, one for A and one for B, and three output terminals: one for the condition A > B, one for the condition A = B, and one for the condition A < B.

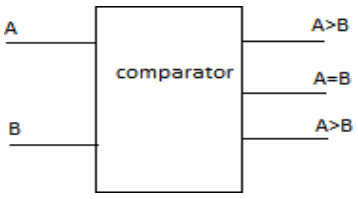


FIG 8 : block diagram of 1bit comparator

A	B	A>B	A=B	A<B
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

Table 4: 1bit comparator truth table

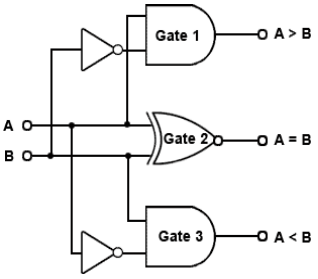
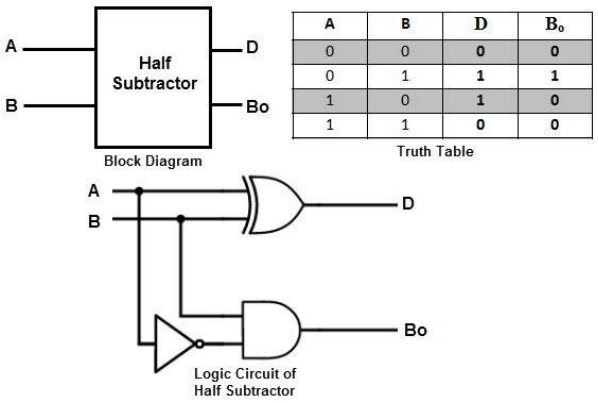


FIG 9: 1 Bit ComparatoR

4. Subtraction is another fundamental mathematical operation that digital computers can execute. The mathematical process of subtracting one integer number from another to get the equivalent quantity is known as subtraction. "Minuend" is the number from which another number is to be subtracted, and "Subtrahend" is the number that is subtracted from the minuend.

- 0 – 0 = 0
- 0 – 1 = (Borrow)1 1
- 1 – 0 = 1
- 1 – 1 = 0

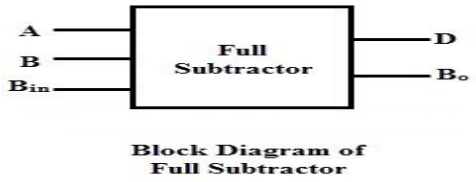


A	B	D	B_o
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Truth Table

FIG 10:Half Subtractors

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025



Block Diagram of Full Subtractor

A	B	Bin	D	Bo
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Truth Table

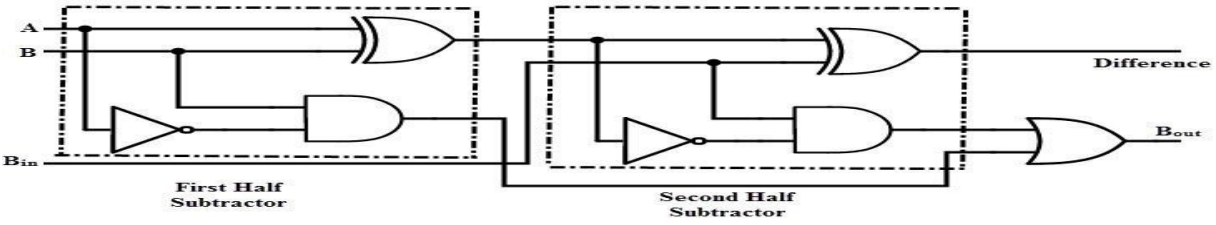


FIG 11:Full Subtractor

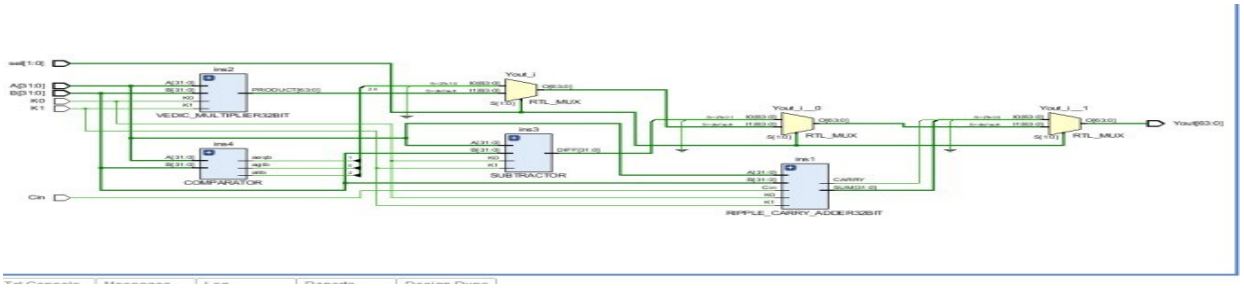


FIG 12: ALU

V. RESULT: -

The flexibility and reusability of individual sub-circuit modules are improved by extending the behaviorally modelled 32-bit ALU to a structurally modelled n-bit ALU. QCA majority gates were used in the design of each individual module. Because QCA may achieve excellent performance in terms of clock frequency, device density,

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

and power consumption when compared to identical implementations using traditional logic gates, its usage on the nanoscale has a bright future. The encryption logic is implemented at the basic sub-circuit module to secure the planned ALU. This is common to all higher level modules that have a bigger influence on the output, making encryption a significant factor.

The output of the encrypted ALU is evaluated for various inputs for all four key combinations (K1K0) using a two-bit key as the input. The ALU's function is determined by the selection line inputs (S1S0), and the accuracy of the outcome is determined by the key inputs (K1K0). Only when the correct key is used, "K1K0 = 01," is the output accurate. The output differs significantly from the desired response for incorrect key combinations, which confuses the adversary.

Simulation Output

In contrast to the schematic, which verifies the connections and blocks, the simulation is the procedure that is referred to as the final verification with regard to its operation.

FIG 13: Simulated Waveforms of existed ALU

Simulated Waveforms of existed ALU

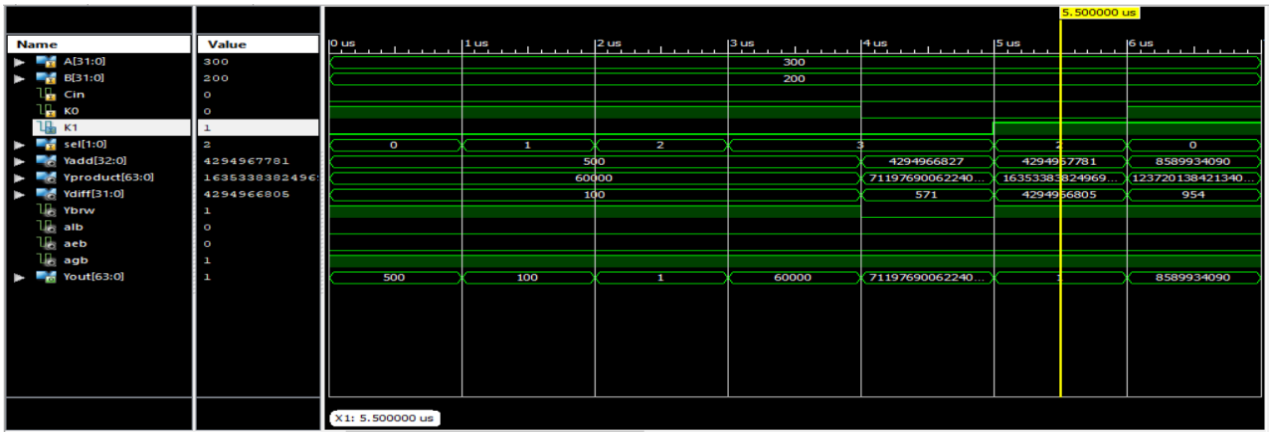


FIG 14: Simulated Waveforms of proposed ALU

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

PARAMETERS:-

Consider in VLSI the parameters treated are area ,delay and power ,based on these parameters one can judge the one architecture to other. here the consideration of delay is considered the parameter is obtained by using the tool XILINX 14.7 and the HDL language is verilog language.

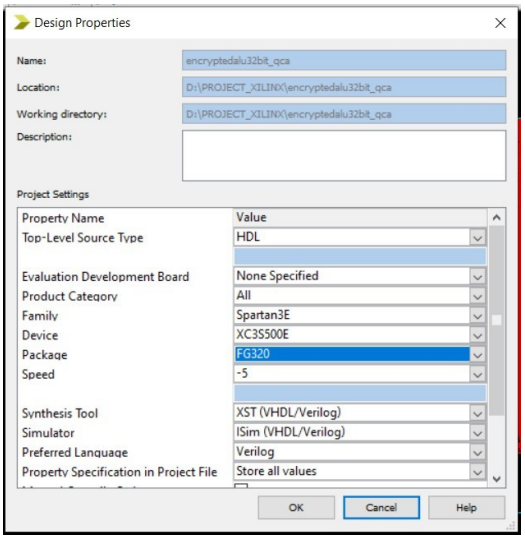


FIG 15:device used for ALU synthesis

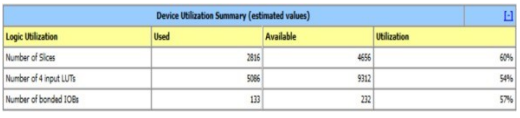


FIG 16: device used for ALU synthesis

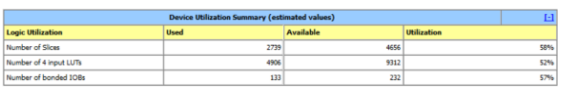


FIG 17: device used for proposed ALU synthesis

Parameter	Existed ALU	Proposed ALU
No of LUTs	5086	4906
Power (Watt)	44.240	42.674

Table 5: parameter comparison bar graph

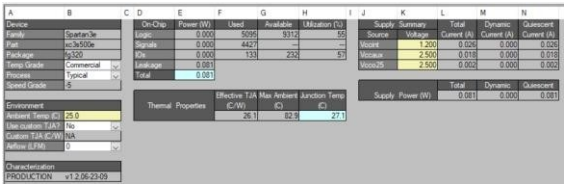


Table6 : power analyzer

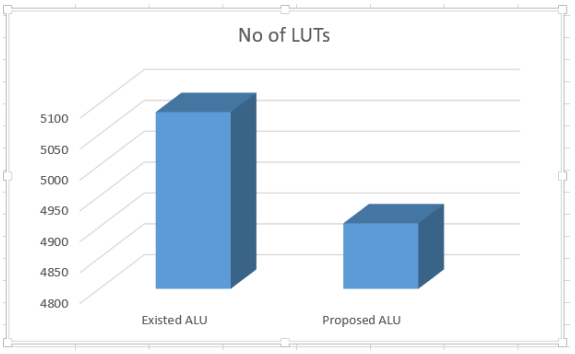


FIG18: LUTs comparison bar graph

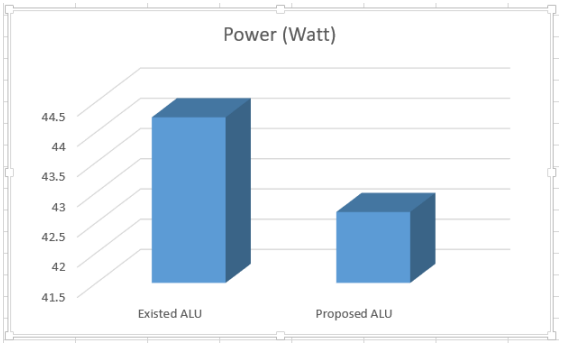


FIG 19: Power comparison bar graph

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

VI. CONCLUSION

The flexibility and reusability of individual sub-circuit modules are improved by extending the behaviorally modelled 32-bit ALU to a structurally modelled n-bit ALU. QCA majority gates were used in the design of each individual module. Because QCA may achieve excellent performance in terms of clock frequency, device density, and power consumption when compared to identical implementations using traditional logic gates, its usage on the nanoscale has a bright future. power use in contrast to analogous implementations using traditional logic gates. The encryption logic is implemented at the basic sub-circuit module to secure the planned ALU. This is common to all higher level modules that have a bigger influence on the output, making encryption a significant factor. The output of the encrypted ALU is evaluated for various inputs for all four key combinations (K1K0) using a two-bit key as the input. The ALU's function is determined by the selection lineinputs (S1S0), and the accuracy of the outcome is determined by the key inputs (K1K0). Only when the correct key is used, "K1K0 = 01," is the output accurate. The output differs significantly from the desired response for incorrect key combinations, which confuses the adversary.

VII. FUTURE SCOPE

1. **Scalability for Complex Operations and GreaterBit-**

WidthsThe current architecture is appropriate for integration into contemporary processors and SoCs since it can be expanded to enable 16-bit, 32-bit, or 64-bit ALUs in addition to other operations like multiplication, division, and floating-point arithmetic.

2. **Scalability for Larger Bit-Widths and ComplexOperations**

Since the existing architecture may be extended to support 16-bit, 32-bit, or 64-bit ALUs in addition to other operations like multiplication, division, and floating-point arithmetic, it is suitable for incorporation into modern CPUs and SoCs.

3. **Enhanced Key Management Techniques**

Future implementations can explore dynamic or runtime key management systems, such as Physically Unclonable Functions (PUFs), to provide hardware-level key generation, further enhancing resistance against key extraction attacks.

4. **Side-channelAttackResistance**

Additional countermeasures like power-balancing logic and randomization techniques can be integrated to protect against side-channel attacks such as Differential Power Analysis (DPA) or Electromagnetic Analysis (EMA).

5. **Formal Verification and Security Metrics**

Incorporating formal verification tools to prove the correctness and security of the obfuscated logic can improve reliability. Moreover, quantitative security metrics (like SAT attack

resistance) can be developed to compare obfuscation strength.

6. Application in IoT and EmbeddedSystems
Lightweight versions of the obfuscated ALU can be tailored for low-power and resource-constrained devices in the IoT domain, where security is crucial but hardware resources are limited.

7. AI-assisted Obfuscation Strategies
Machine learning models could be employed to automatically generate optimized obfuscation patterns based on threat models and usage scenarios, improving both security and performance.

8. Commercial IP Protection Solutions The concept can evolve into a standardized IP core that semiconductor companies can integrate into their ASIC/FPGA designs to protect proprietary logic from being cloned or pirated

VIII. REFERENCE:

[1]Vijay kumar reddy Modified High Speed Vedic Multiplier Design and Implementation The proposed research work specifies the modified version of binary vedic multiplier using vedic sutras of ancient vedic mathematics.It provides modification in preliminarilry implemented vedic multiplier S. Akhter, "VHDL implementation of fast NxN multiplier based on Vedic mathematics," in Proc. 18th European Conference on Circuit Theory and Design, 2007, pp. 472-475

[2]A. Momeni, J. Han, P. Montuschi, and F. Lombardi, "Design and Analysis of Approximate Compressors for Multiplication", Inexact (or approximate) computing is an attractive paradigm for digital processing at nanometric scales. Inexact computing is particularly interesting for computer arithmetic designs. This paper deals with the analysis and design of two new approximate 4-2 compressors for utilization in a multiplierS. Nagaraj, Dr.G.M. Sreerama Reddy and Dr.S. Aruna Mastani; A Comparative Study on Different Multipliers-SurveyJournal of Advanced Research in Dynamical and Control Systems14739-7522018Institute of Advanced Scientific Research.

[3]C. Liu, J. Han, and F. Lombardi, "A Low- Power, High-Performance Multiplier with ConFIGurable Partial Error Recovery", Proc. of IEEE Design, Automation & Test in Europe Conference & Exhibition (DATE), M.Pushpa,
S. Nagaraj, Design and Analysis of 8-bit Array, Carry Save Array, Braun,Wallace Tree and Vedic Multipliers, IEEE Sponsored International Conference On New Trends In Engineering & Technology(ICNTET 2018).

[4]G. Zervakis, et al., "Design-Efficient Approximate Multiplication Circuits Through Partial Product Perforation" Approximate computing has received significant attention as a promising strategy to decrease power consumption of inherently error tolerant

applications Nagaraj, S; Thyagarajan, K; Srihari, D; Gopi, K; Design and Analysis of Wallace Tree Multiplier for CMOS and CPL Logic2018 International Conference on Computation of Power, Energy, Information and Com- munication (ICCPEIC)006-0102018IEEE

[5]T. Yang, T. Ukezono, and T. Sato "A Low-Power High-Speed Accuracy- Controllable Multiplier Design", Multiplication is a key fundamental function for many error-tolerant applications. Josmin Thomas ; R. Pushpangadan ; S Jinesh Comparative study of performance Vedic multiplier on the Basis of Adders used 2015 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE)