

# Area and Power Analysis of Various Adders for Addition and Subtraction in ALU

Bhavani S<sup>1</sup>, Bhavyashree B G<sup>2</sup>, Geetha S<sup>3</sup>, Harshitha M<sup>4</sup>, Manjula B B<sup>5</sup>

U.G. Student, Department of Electronics and Communication Engineering, East West Institute of Technology,  
Anjananagar, Bangalore, India<sup>1,2,3,4</sup>

Associate Professor, Department of Electronics and Communication Engineering, East West Institute of Technology,  
Anjananagar, Bangalore, India<sup>5</sup>

**ABSTRACT:** In this paper, we have compared the different types of adders like Ripple Carry Adder(RCA), Carry Save Adder(CSA) and Carry Look-Ahead Adder (CLA) for performing addition and subtraction operation in the Arithmetic Logic Unit(ALU), the operations are performed on two 32-bit integer numbers which produces 32-bit integer output. The different adders are compared on the basis of power and area constraint and the best adder for performing addition and subtraction is chosen. The code is written in Verilog HDL language. Simulation is done using Modelsim. Simulation results are verified and synthesized in the Cadence Tool.

**KEYWORDS:**Ripple Carry Adder, Carry Save Adder, Carry Look Ahead Adder, Power and Area Constraints.

## I. INTRODUCTION

Adders are the basic building blocks of most of the digital systems. Fast and accurate operation of digital system depends on performance of adder. In most of the computers, processors and also in the arithmetic logic unit (ALU), adders become a most important hardware unit for the efficient implementation of arithmetic unit. In many applications, adders are not only used in the arithmetic logic unit, but also in other part of the processors.

The most primitive adder is known as Half Adder that has two inputs and two outputs namely, sum and carry. A device capable of computing the sum of 3 bits and producing two outputs which are sum and carry is called as Full Adder. But, these adders can compute for a single set of bits at a time. For multi bit addition, usage of full adders in parallel block is preferred. When computation is done in parallel, the sum of each bit is done simultaneously and carry is forwarded to the next block as an input, and the carry output is calculated from the last adder block. Various adders used in this paper are RCA, CSA and CLA. Ripple Carry Adder consists of full adder connected in parallel, in Carry Save Adder three bit are added in parallel, Carry Look Ahead Adder is created after observing the lower order bits and they are added if higher order bits generate a carry. Based upon the parameters like area and power constraints, the best adder is chosen for addition and subtraction operation.

The remaining portion of the paper is organized as follows: Section II will provide a brief knowledge about the Related Work. Section III gives the comparative study of various types of adders and the Methodology. Section IV specifies the simulation and synthesis results. Section V brief out the conclusion and future scope.

## II. RELATED WORK

The primary aim of fast growing technologies is to speed up the device but increasing the speed alone does not serve the purpose, the size of device is also a matter of concern that is to be reduced to an optimum level of mobility issues other major problem is power i.e., usage of more and more power from a single device demonstrating mobility as the balance between the process and power supply also needs to be managed with more care. So, if summarize the main problem facing is from time to time are the managed between delay, area and power constraint, these are the main problem of our adders nowadays is associate with, making it faster with low power consumption and lesser on-chip-area for mobility, so we need to modify the adders taking this point into consideration[1]. All complex adders are constructed from the basics of half- adder and full-adder circuitry[2]. Approximate adders have gain interest in case of

error tolerance application which enhance circuit performance on trading-off accuracy. CLA adder reduces delay by generating all the carry bits at a time[4]. In ripple carry adder, it does not execute the addition of operands until and unless the previous carry is obtained. The architecture of RCA becomes complicated during the addition of higher bits. To overcome this carry must be generated in parallel[6]. The two main operation that are important in multiplication i.e., partial product generation and partial product addition. For higher bit multiplication addition of larger no. of PPR's affects the performance of multiplication. So the best approach is addition of PPR's using CLA based architecture which reduces the number of partial products and yields good result[5]. The different adders are efficient in various types of work platform they are ripple carry adder, carry save adder, carry look ahead adder[9] and among the ripple carry adder happens to be the simplest adder with less complexity[7]. But it has high amount of propagation delay for higher bit operation associated with it[10]. Every single full adder in the first stage operation on three inputs and save the carry for the next stage. As it saves the carry and propagates to the next stage that is why it is called as carry save adder. Among ripple carry adder and carry save adder, carry look ahead adder is the faster adder[2].

### III. METHODOLOGY

In this section, the implementation details of ripple carry adder, carry save adder and carry look ahead adder is presented to perform 32-bit integer addition and subtraction.

#### A. RIPPLE CARRY ADDER

Ripple Carry Adder is the simplest form of adder. One Full Adder performs the addition operation of two one-bit input, but when multi-bit addition are to be carried out, we need Ripple Carry Adder. The Ripple Carry Adder is designed simply by cascading Full Adder blocks. The carry-out of any Full Adder is propagated to the next stage as carry-in and the calculation continues in the next block. It is the simplest adder among all the multi-bit adder circuit. The carry-out of every block is propagated to the next block and the computation waits for the carry bit to arrive which adds up to the delay. One of the most important point to be considered in this carry adder is the final output is known only after the carry outputs are generated by each full adder stage and forwarded to its next stage. So the last adder has to wait for all other adders to complete the operation which increases the delay.

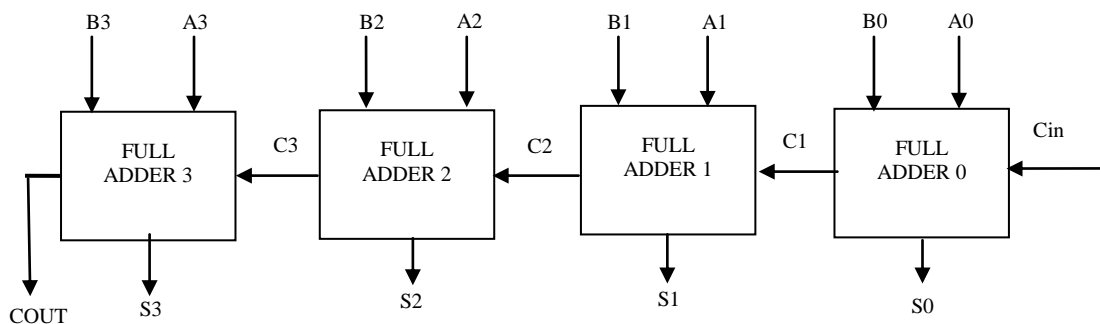


Fig 1. Ripple Carry Adder

#### B. CARRY SAVE ADDER

The drawback of the Ripple Carry Adder can be overcome by Carry Save Adder. Carry Save Adder works on the principle of delaying carry resolution until the end, in other words, saving carries. When calculating the sum of three numbers – A, B, C, for example. Carry Save Adder splits it into two parts:  $A + B + C = S + C$ . Carry Save Adder (CSA) has n number of full adders which perform the single summation and generates carry individually. It does its first operation of summation in the first stage simultaneously. As it saves the carry and propagates to the next stage it is called as Carry Save Adder. To add three numbers, each of n bit operand need  $2nT$  delay in Ripple carry Adder. As delay in each full adder is 'T'. The same operation in Carry save adder take  $(n+1)T$  because the first stage of full adder will operate at same time and save the carry to the next level. So only one full adder from the first stage and the total full adder in the second stage is responsible for total delay. Entire sum can be calculated after shifting the carry sequence by one place to the left side. The main application of Carry Save algorithm is, well known for computer microarchitecture.

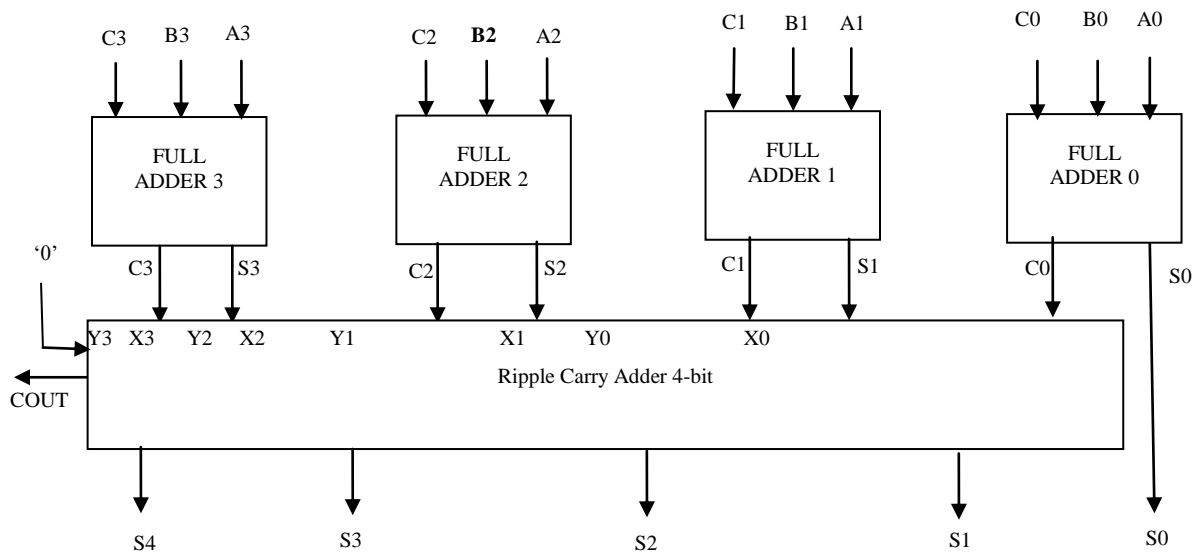


Fig 2. Carry Save Adder

### C. CARRY LOOK AHEAD ADDER

Carry Look Ahead Adder or fast adder is a type of adder using a digital logic. A Carry Look Ahead Adder improves speed as it reduces the amount of time required to determine the carry bits. The block is designed such that the carry logic over fixed groups of bits of the adder is reduced to two-level logic. For each bit in a binary sequence to be added, the carry look-ahead adder logic will determine whether that bit pair will generate a carry or propagate a carry. This allows the logic to process the two numbers being added to determine the carry ahead of time. Hence, it reduces the wait time, power consumed and also area consumed.

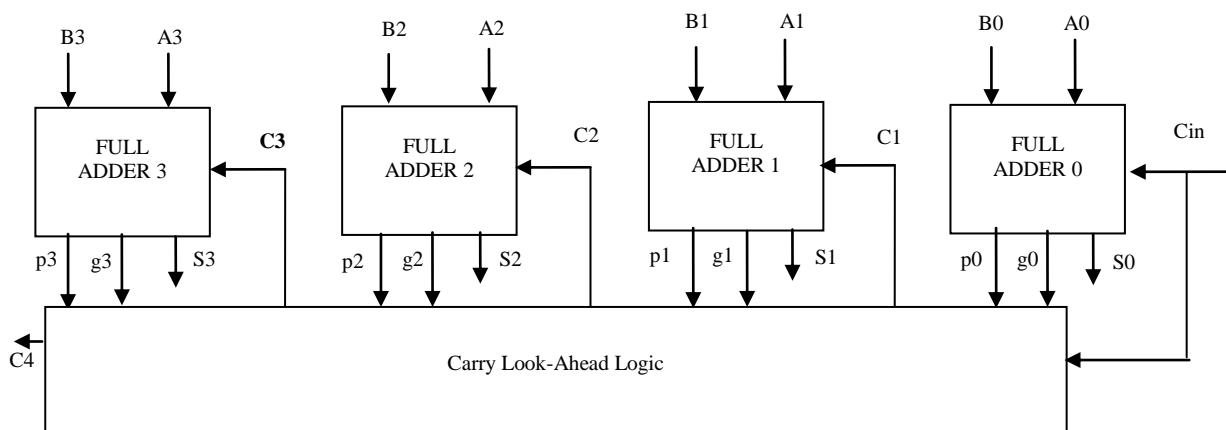


Fig 3. Carry Look Ahead Adder

## IV. EXPERIMENTAL RESULTS

The design is developed using Verilog HDL. Simulation is done in Modelsim and the result obtained is again verified in the Cadence. Fig 4 shows the simulation result of Ripple Carry Adder. Fig 5 shows the simulation result of the Carry Save Adder and Fig 6 shows the simulation result of Carry Look Ahead adder.

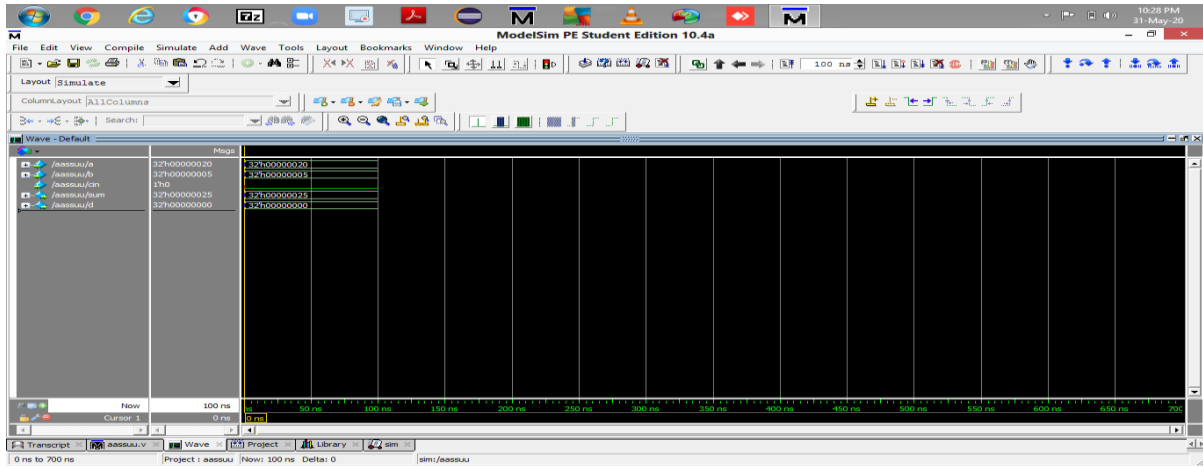


Fig 4. Simulation Result of Ripple Carry Adder

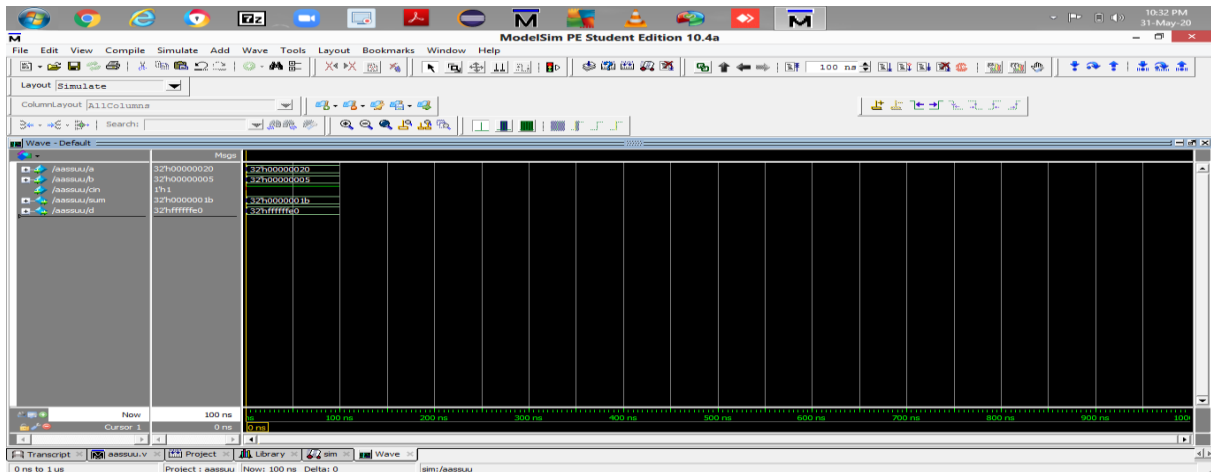


Fig 5. Simulation Result of Carry Save Adder

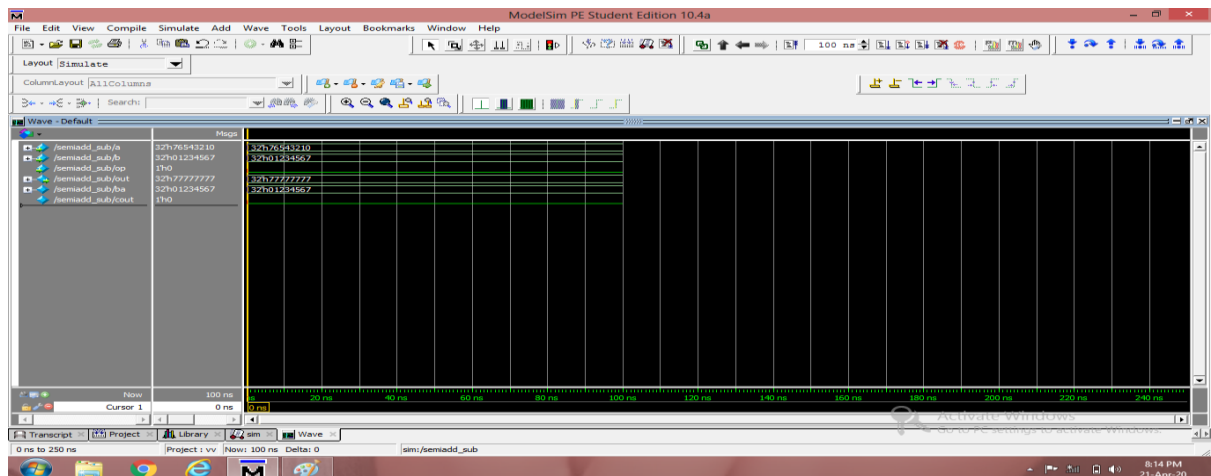


Fig 6. Simulation Result of Carry Look Ahead Adder

TABLE I gives the details of area and power for each adder which performs both addition and subtraction. By the simulation and synthesis result we can observe that Carry Look Ahead Adder is the best for performing addition and subtraction in the Arithmetic and Logic unit.

TABLE I. Synthesis Result of Adders

Operation	Adders	Power(mW)	Area( $\mu\text{m}^2$ )
Addition-Subtraction	RCA	1.5012	8114
	CSA	1.2092	3342
	CLA	0.0627	917

Figure 7 shows the graphical comparison of area among Ripple Carry Adder, Carry Save Adder and Carry Look Ahead Adder. From the graph, we can observe that Carry Look Ahead Adder consumes less area when compared to Ripple Carry Adder and Carry Save Adder. Figure 8 shows the graphical comparison of power among ripple carry adder.

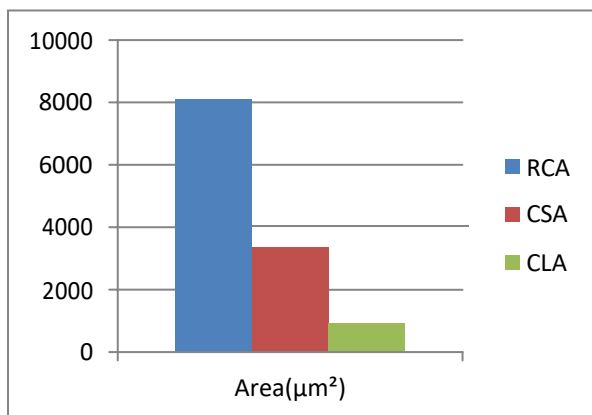


Fig 7. Graph of Area for RCA, CSA and CLA

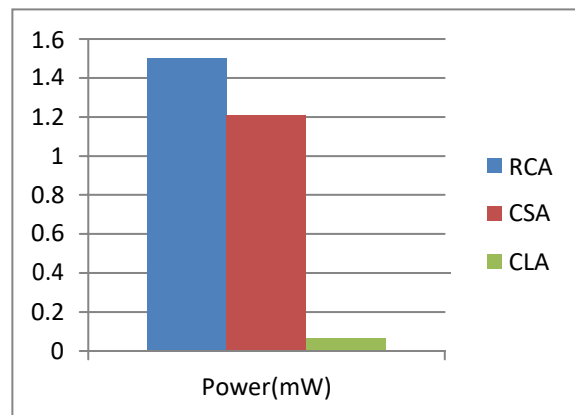


Fig 8. Graph of Power for RCA, CSA and CLA

### V. CONCLUSION

The proposed work describes the comparison of adders like Ripple Carry Adder, Carry Save Adder and Carry Look Ahead Adder. Simulation and Synthesis results of these adders are mentioned. Upon the comparison we conclude that Carry Look Ahead Adder is the best adder with least area and power. Low power consumption results to low heat dissipation and increases the durability of the system. Lesser area helps in the portability. This work can be further extended to perform multiplication and division operation using Carry Look Ahead Adder. It can also be used in other processors to identify the addresses, indices etc.

### REFERENCES

- [1] ShubhamSarkar, SujanSarkar, “Comparison of various adder and their VLSI implementation”, IEEE paper-2018.
- [2] MarojuSaikumar, Dr. P Samuundiswary, “Design and Perform Analysis of Various Adders Using Verilog”, IJCSMC-2013.
- [3] BenishaBennet, Dr. S Maflin, “Modified Energy Efficient Carry Save Adder”, ICCPCT-2015.
- [4] MehediHasan, ParagBiswas, “High Speed and Ultra Low Power Design of Carry-Out Bit of 4-Bit Carry Look-Ahead Adder” IEEE paper-2019.
- [5] SmrutiBokade, PravinDakhole, “CLA Based 32-bit Signed Pipelined Multiplier”, IEEE paper-2016.
- [6] A Krishna Vamshi, N Uday Kumar, “A Systematic delay and Power Dominant Carry Save Adder Design”, ICSSIT-2018.
- [7] Shubin, “Analysis and comparison of ripple carry adder by speed”, IEEE paper-2010.



- [8] Jayeeta Berman, Vinaykumar, “Approximate Carry Look Ahead Adder(CLA) for Error Tolerant Applications, ICOEI-2018.
- [9] Jasbirkaur, lalitsood, “Comparision between various types of adder topologies”, IEEE paper-2015.
- [10] Ravikumar A Javali, Ramanath J Nayak, “Design of high speed carry save adder using carry look ahead adder”, IEEE paper-2014.
- [11] SauvagyaRanjanSahoo, Kamala KantaMahapatra, “Design of low power and high speed ripple carry adder”, IEEE paper-2012.