JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

DESIGN OF LOW POWER HIGH SPEED PARALLEL PREFIX ADDER

[1] Varikunta Adbhuta Teja, [2] K.Sangeetha,M.Tech

[1]M.Tech PG Scholar, [2]Asst.Professor
[1,2]Deptartment Of ECE,
[1,2] SKR College of Engineering and Technology,Konduru satram (v),Manubolu(m),SPSR Nellore(dt),Andhra Pradesh.

Abstract : The proposed adder presents a 64 bit low power high speed parallel prefix adder. The 64-bit proposed adder Design is compared with various implemented standard models like Sklansky and Brent Kung with respect to their power consumptions, delay and speed. We introduce a low power high speed full adder, that pre generates the carry which will reduce the time requirement. Full adder is the basic component for an ALU. By reducing the power and increasing the speed of Full Adder, the ALU power also be reduced. Compared with other existing standards like Sklanshy and Brent Kung, the Proposed Full Adder is more efficient and eliminates the conductance overlap between n MOS and p MOS and to reduce the short circuit power dissipation. The Simulation is carried out using Xilinix ISE 12.1 and Micro wind technology, and compared with preexisting standards which shows that design of Proposed Parallel Prefix Adder using Gate Diffusion Input Technology is more efficient

I.INTRODUCTION

Very large scale integration (VLSI) is the process of creating an integrated circuit by combining thousands of transistors into a single chip. VLSI begin in the 1970s when complex semiconductor communication technology being developed. The microprocessors is a VLSI device. Before the introduction of VLSI technology, most ICs had a limited set of function they could perform. Electronic circuit might consist of a CPU, ROM, RAM and other glue logic. VLSI lets IC designers add all of these into one chip.

The electronic industry has achieved a phenomenal growth over the last few decades, mainly due to the rapid advances in large scale integration technology and system design applications. With the advent of very large scale integration (VLSI) designs, the number of applications of integrated circuits (ICs) in high performance computing, controls, telecommunication, images and video processing and consumer electronics has been rising at a very fast pace.

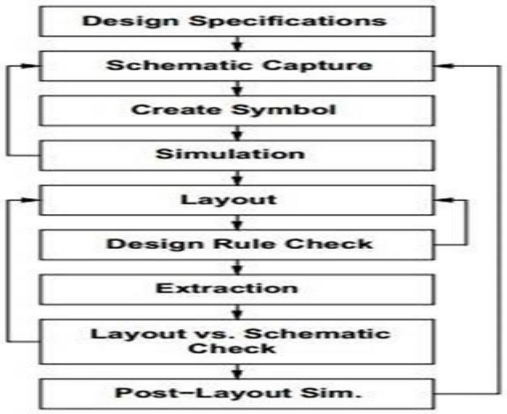


Fig 1.1 Basic block diagram of VLS

The current cutting-edge technology such as high resolution and low bit rate video and cellular communication provide the end-users a marvellous amount of application, processing power and portability this trend is expected to grow rapidly, with very important application on VLSI design and system design.

DEALING WITH VLSI CIRCUIT:

The way normal blocks like latches and gates are implemented is different from what students have seen so far, but the behaviour remains the same. All the miniaturization involves new things to consider. A lot of thought has to go into actual implementations as well as design.

1.1.CIRCUIT DELAYS:

Large complicated circuits running at very high frequencies have one big problem to tackle-the problem of delays in propagation of signals through gates and the wire even or areas a few micrometres across! The operation seed is so large that as the delays add up, they can actually become comparable to the clock speeds.

Effect of high operation frequencies is increased consumption of power. This has two-fold effect-devices consume batteries faster, and heat dissipation increases. Coupled with the fact that surface areas have increased, heat possess a major threat to the Stability of the circuit itself.

1.2.LAYOUT:

Laying out the circuit components is task common to all branches of electronics. What"s so special in our case is that they are many possible way to do this; there can be multiple layers of different materials on the same silicon, there can be different arrangements of the smaller parts for the same component and soon. The choice between the two is determined by the way we choose the layout the circuit components. Layout can also affect the fabrication of VLSI chips, making it either easy or difficult to implement the components on the silicon.

1.3.INTRODUCTION TO VHDL:

A digital system can be described at different levels of abstraction and from different points of view. An HDL should faithfully and accurately model and describe a circuit, whether already built or under development, from either the structural or behavioural views, at the desired level of abstraction. Because HDLs modelled after hardware, their semantics and views are very different from those of traditional programming languages.

1.4.LIMITATIONS OF TRADITIONAL PROGRAMMING LANGUAGES:

There are wide varieties of computer programming languages, from Front end to C to JAVA. Unfortunately, they are not adequate to model digital hardware. To understand their limitations, it is beneficial to examine the development of the language. A programming language is characterized by its syntax semantics.

The syntax comprises the grammatical rule used to write a program, and the semantics is the "meaning" associated with language constructs. When new computer language is developed, the designers first study the characteristics of the underline processes and the developed syntactic constructs and their associated semantics to model and express these characteristics

Most traditional general-purpose programming languages such as C, are modelled after a sequential process. In this process, operations are performed in sequentially order, one operation at a time. Since an operation frequently depends on the result of an earlier options, the order of execution cannot be altered at will. The sequential process model has two major benefits. At the abstract level, it helps the human thinking process to develop an algorithm step by step. At implementation level, the sequential process resembles the operation of a basic computer model and does allows efficient translation from an algorithm to machine instructions.

The characteristics of digital hardware, on the other, are very different from those of the sequential model. The typical digital system is normally built by smaller parts, with customised wiring that connects the input and output ports of these parts. When signal changes, the ports connected to the signal are activated and a new set of operation is initiated accordingly. The operation are performed concurrently, and each operation will a specific amount of time, which generates the concept of propagation delay and timing. The sequential model used in traditional programming language cannot capture the characteristics of digital hardware, and there is a need for special language.

II. PARALLEL PREFIX ADDERES

Adders are one of the critical elements in VLSI chips because of their variety of usages such as ALUs, floating point arithmetic units, memory addressing and program counting. Among them, prefix adders are based on parallel prefix circuit theory which provides a solid theoretical basis for wide range of design trade-offs between delay, area and wiring complexity. Parallel prefix adder is the most flexible and widely-used binary adder for ASIC designs. Many high-level synthesis techniques have been developed to find optimal prefix structures for specific applications.

Carry look-ahead adder is designed to overcome the latency introduced by the rippling effect of the carry bits. Parallel prefix adders allow more efficient implementations of the carry look-ahead technique. These are nothing but a two level carry look-ahead adders. They are among the best adders with respect to area, time, cost, performance and especially for high-speed addition of large numbers. The most important aspect of prefixes is that, they, and their sub-terms can be computed in parallel. This follows from the property of associatively.

2.1.PARALLEL PREFIX ADDDER TOPOLOGIES:

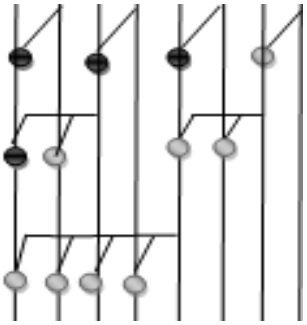
- Sklansky adder
- Kogge stone adder
- Ladner Fischer
- Brent-kung adder
- Han Carlson adder
- Knowles adder
- Reduced inter stage Sklansky adder by Ghaznavi.
- Ramanathan adder

A detailed description on each adder is given below.

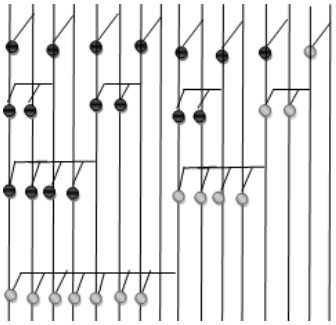
2.2.1.SKLANSKY ADDER:

The conditional-sum addition is a fast addition paving a logarithmic speed-up. It provides a platform for a parallel, high speed addition for digital computers. One of the main features is that the addition rate is invariant with the length of the summands due to its sequential mode of operation. The scheme of conditional sum addition may be applied to the simultaneous addition of more than two summands. It is possible to apply conditional sum addition to higher –radix number systems .This adder stands ahead because realization of the adder with integrated devices or modules is possible. The sklansky adder reduces the delay to $\log_2 N$ stages by computing intermediate prefixes along with large group prefixes. The logical depth of this adder topology is O ($\log_2 N$), the number of nodes per stages

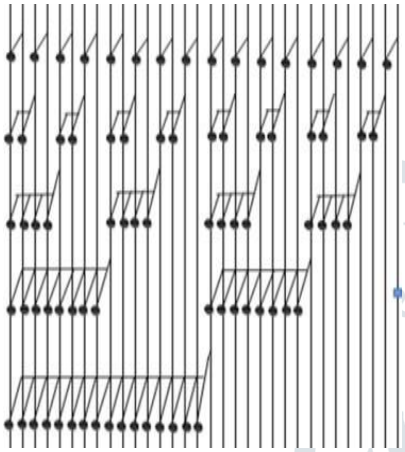
=N/2 and total cost function is C(K)=N/2 $\log_2 N$. Therefore, this adder presents a minimum logic depth and the least routing tracks. Due to the large fan out, the area circuit speed is also affected. The entire architecture for 8, 16, 32 and 64 bit is depicted below in fig2.1.



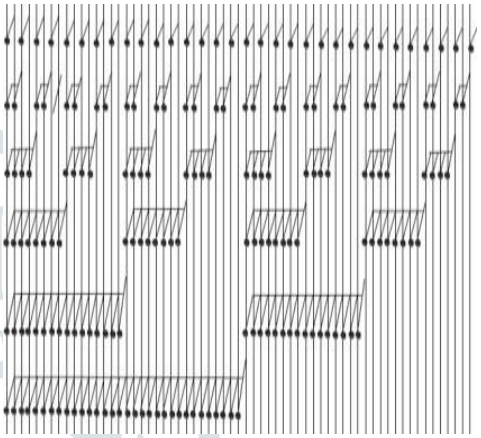
a)**sklansky 8-bit architecture**



b)sklansky 16-bit architecture



c)Sklansky 32-bit architecture

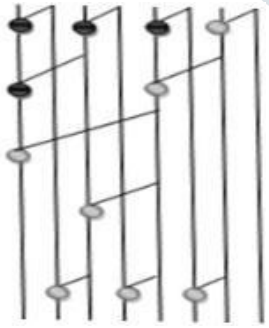


d)sklansky 64-bit architecture

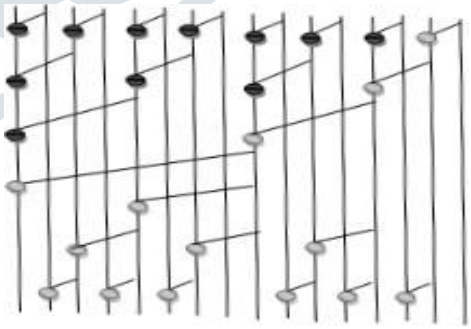
Fig 2.2.1 architecture of Sklansky adder

2.2.2 BRENT KUNG ADDER:

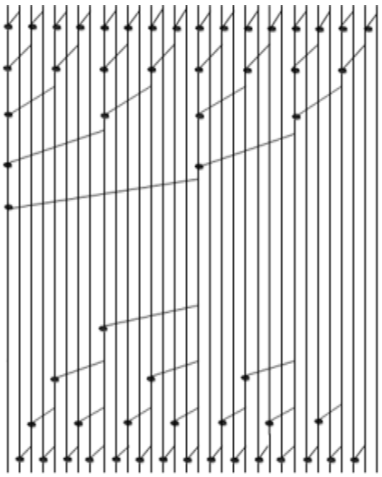
The Brent–Kung adder is one of the better advanced designs. The Brent–Kung adder is a parallel prefix form carry look-ahead adder. It has a high logic depth. Itis considered as one of the better tree adders for minimizing wiring tracks, fan out, and gate count and it has the minimum number of nodes possible. It was originally proposed as a simple and regular design of a parallel adder that addresses the problems of connecting gates in a way to minimize chip area. The operational delay is due to the lengths of interconnection (logic depth) and also the critical path in this adder is long. The logical depth of this topology is $2\log_2 N-1$, the number of nodes per stages $=n/2^i$and the total cost function is C (K) $=N/2\log_2 N$.



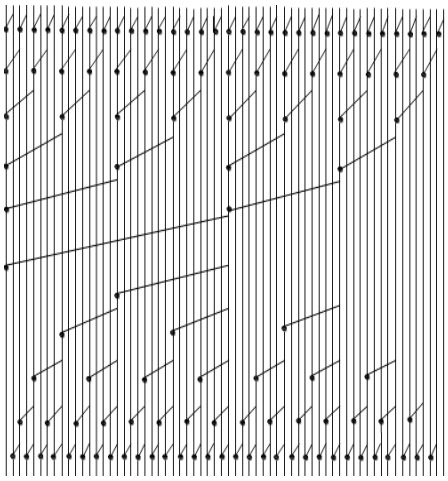
a)brent-kung 8-bit architecture



b)brent-kung 16-bit architecture



c)brent-kung32-bit architecture



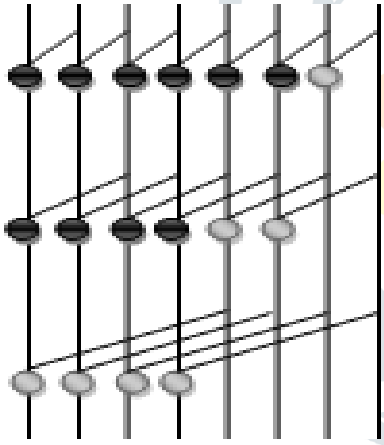
d)brent-kung 64-bit architecture

Fig 2.2.2 architecture of Brent Kung adder

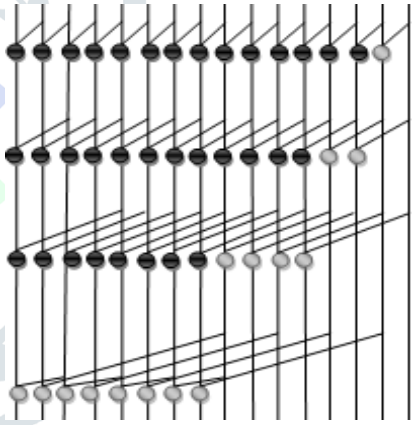
A higher number of buffers are inserted in the Brent Kung structure. Even though the area consumption is convex, it does not affect the energy consumption.The main disadvantage is the presence of too many logic levels. The architecture for 8, 16, 32 and 64 bit Brent Kung adder is depicted.

2.2.3.KNOWLES:

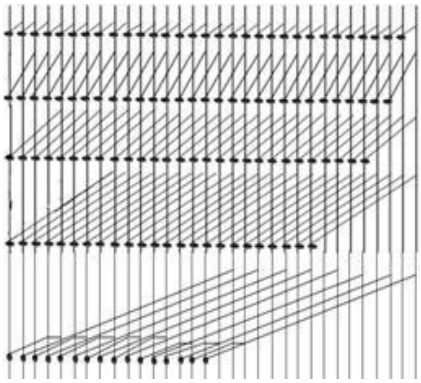
Knowles adder falls between the family of Kogge-Stone and Sklansky. This is used to construct a family of adders focusing on minimum logical depth. These adders are bound by ladner Fischer as there is minimum depth carry trees for static adders and along with Brent Kung topologies as they have minimum number of fan outs. The logical depth of this topology is log2N , the number of nodes per stages=n/2 and the total cost function is C (K) =2log2N-N+1. Due to the trade offs between the internal wiring and the fan-out strategies, a better combination of speed and area/power is achieved. The architecture of 8,16,32 and 64 bit Knowles is depicted in fig 2.2.3



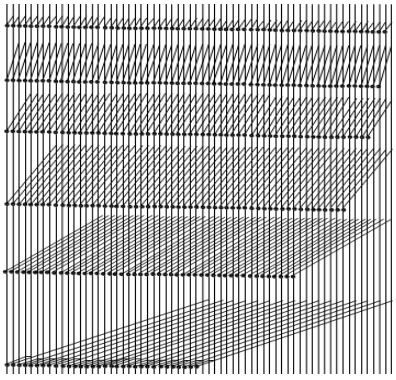
a)Knowles 8-bit architecture



b)Knowles 16-bit architecture



c)Knowles 16-bit architecture

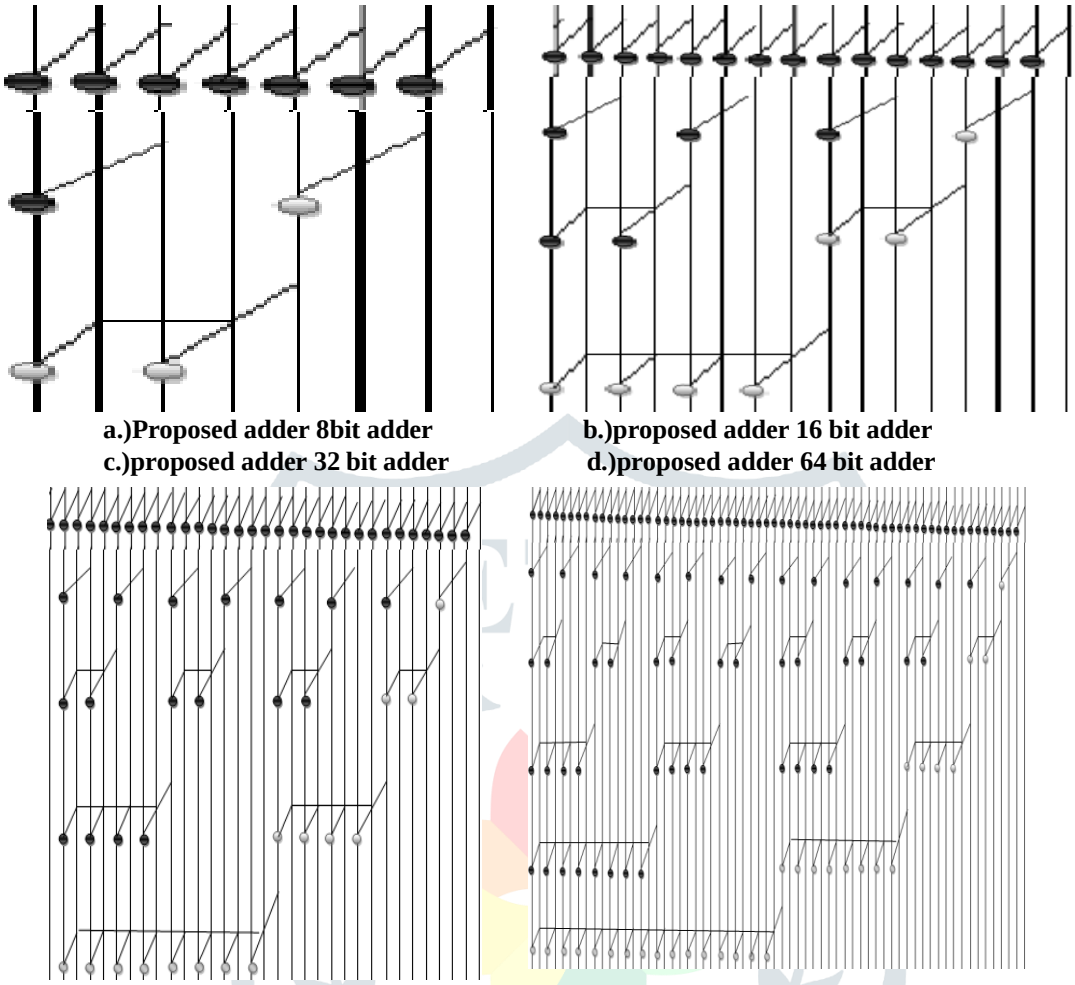


d)Knowles 64-bit architecture

Fig 2.2.3 architecture of Knowles adder

III.PROPOSED ADDER I:

The proposed adder is the hybrid adder obtained from the combination of Ladner Fischer and Han Carlson. The first two stages of the adder follow Han Carlson adder topology and the remaining stages follow the Ladner Fischer adder. The best characteristics of both the adder are adopted in order to construct this adder. It has the order of log N and the numbers of nodes are N/2-1 and N/4.



a.)Proposed adder 8bit adder **b.)proposed adder 16 bit adder**

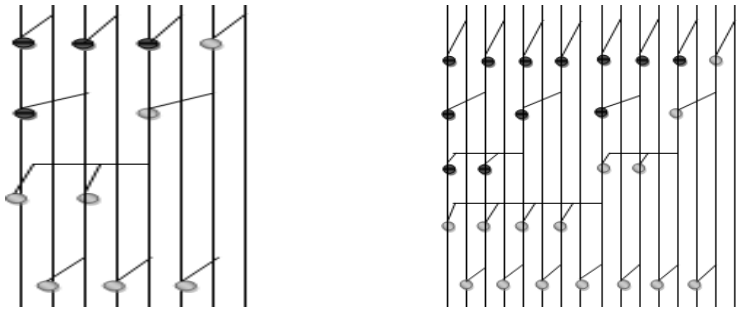
c.)proposed adder 32 bit adder **d.)proposed adder 64 bit adder**

Fig 3.1 architecture of proposed adder I

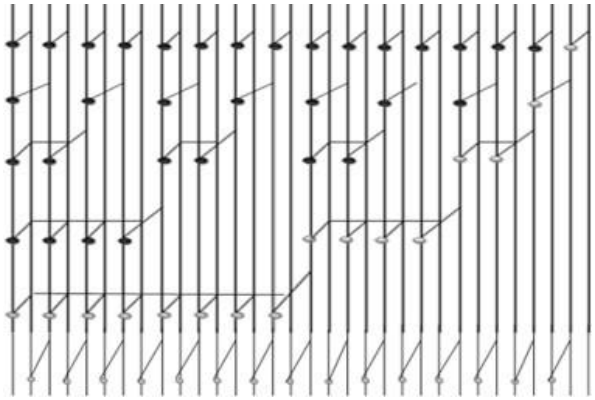
The total number of computational nodes are N-1+(log2N-2)N/4. This adder gives the most effective performance in the aspects of delay. The architecture for 8, 16, 32 and 64 bit proposed adder I is depicted in fig 3.1.

3.1.LADNER FISCHER ADDER:

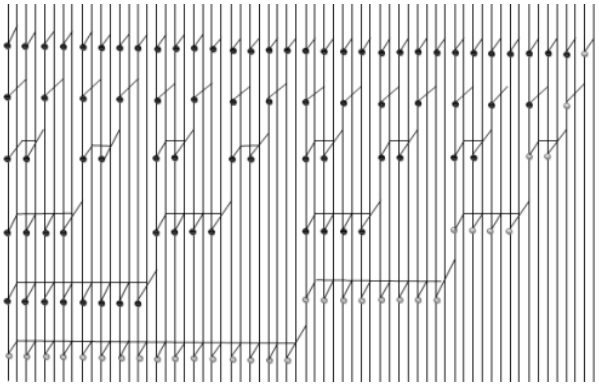
The ladner Fischer adder is the best known early prefix adder. These networks have low gate counts. This adder concentrates on the design of a parallel prefix network with a minimal depth case. The main disadvantage of this adder is that the lateral fan out of the prefix cells doubles at every levels. Thus additional buffers are used, as this drawback can adversely affect the performance. As there is reduction in the length of horizontal wires, the wire capacitance is reduced The logical depth of this topology is $\log_2 N+1$, the number of nodes per stages=n/4 and the total cost function is $C(K)=[n/2*\log_2 n]$. Adders with Ladner Fischer prefix structure require less implementation area but have unlimited fan out comparatively. The entire architecture for 8, 16, 32 and 64 bit is depicted in fig given below



a)Ladner Fischer 8-bit adder **b.)Ladner Fischer 16-bit adder**



c.)Ladner fischer 32-bit adder

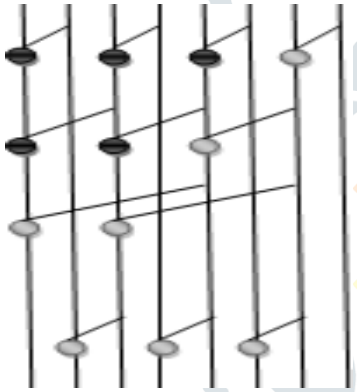


d.)Ladner fischer 64 bit adder

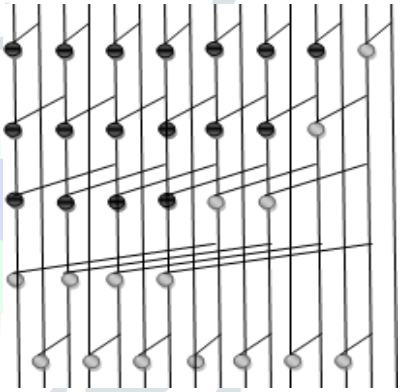
Fig 3.2 architecture of Ladner Fischer adder

3.3.HAN CARLSON:

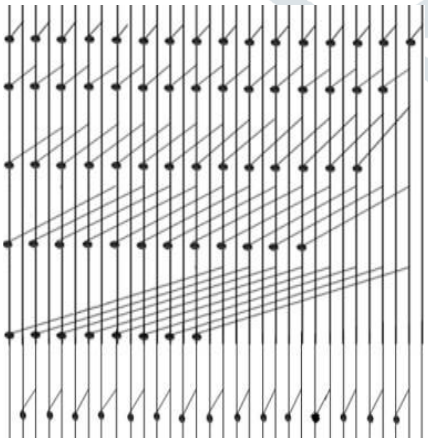
The need for a faster, area efficient and low latency binary adder paved way for deriving a new "hybrid" algorithm. The most important advantage of the creation of such a hybrid model is the designation of fast and area efficient VLSI. adders. The logical depth of this topology is $\log_2 N+1$, the number of nodes per stages = n/4 and the total cost function is $C(K)=N/2\log_2 N$. This is done by combining the Brent -Kung and Kogge -Stone algorithms. By using the new hybrid algorithm, a solution to the problem of applying carry look ahead adder in VLSI is achieved apart from the interconnection irregularity, fan in and fan out limitations. Thus, it gives a good balance between the logic depth and fan-out. This circuitry is fast as it employs on a near minimum depth prefix algorithm. The architecture of 8, 16, 32 and 64 bit Han Carlson adder is depicted in Fig 3.3.



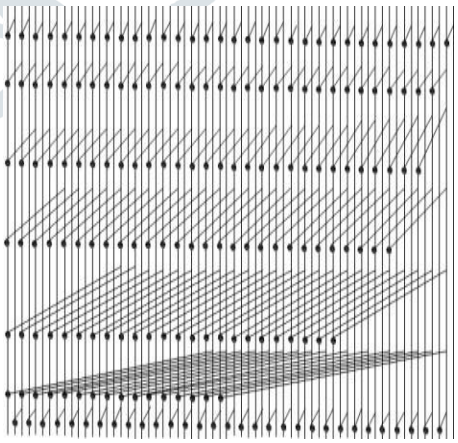
a)Hanclarson 8-bit architecture



b)Hanclarson16-bit architecture



c)Hanclarson32-bit architecture

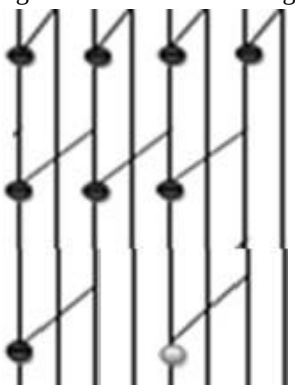


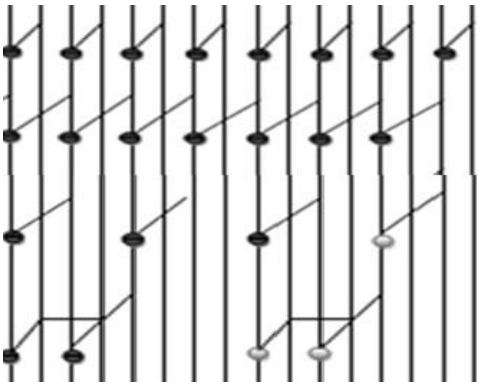
d)Hanclarson64-bit architecture

Fig 3.3 architecture of Han Carlson

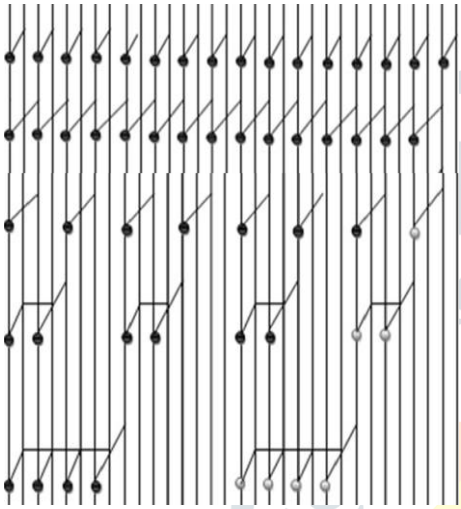
3.4.PROPOSED ADDER II

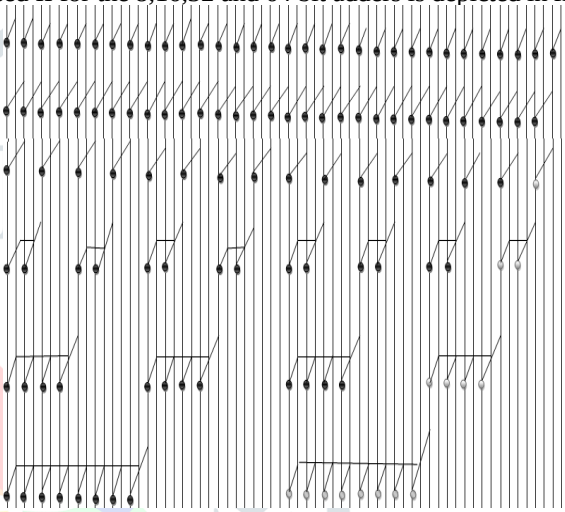
The proposed adder II is the hybrid of kogge stone adder and the ladner fischer adder. This adder follows the kogge stone adder for its first stage and then the remaining is followed by ladner fischer adder. Thus it has the fastest speed feature





of the kogge stone adder . It has the order of logN and the numbers of nodes are N/2-1 and N/4. The total number of computational nodes are N- 1+(log2N-2)N/4.Since the ladner fischer adder has very less slices, this featurealso enhances the overall improvement of the adder. The architecture for the proposed II for the 8,16,32 and 64 bit adders is depicted in fig 3.4.



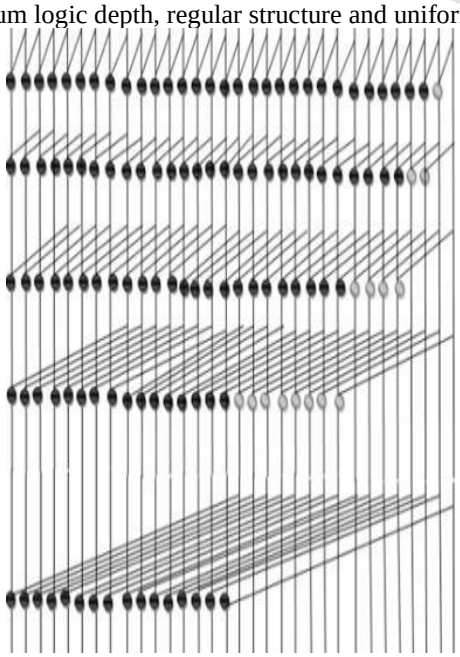


c.) proposed 2 32-bit architecture **d)proposed 2 adder 64 bit architecture**

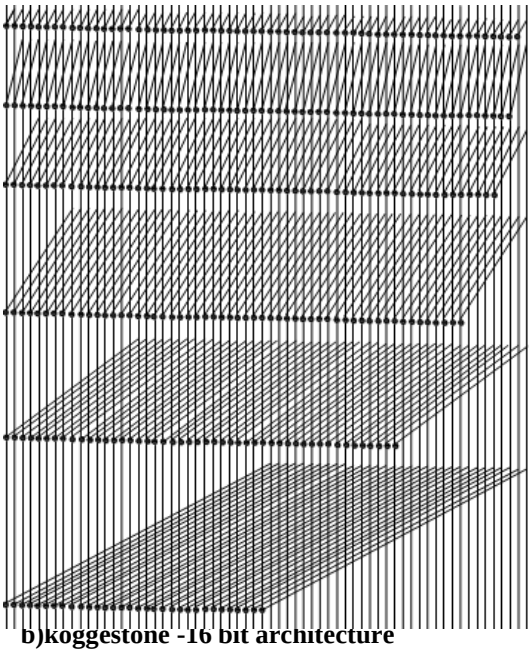
Fig 3.4 architecture of proposed adder II

3.5.KOGGE STONE:

Kogge-stone adders are the fastest prefix tree. It generates the carry signals in O(log n) time, and is widely considered the fastest adder design possible. This is the most commonly used parallel prefix topology. The main features of this adder are that, it has minimum logic depth, regular structure and uniform fan out.

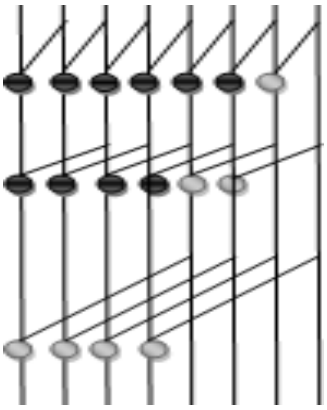


a)koggestone 8-bit architecture

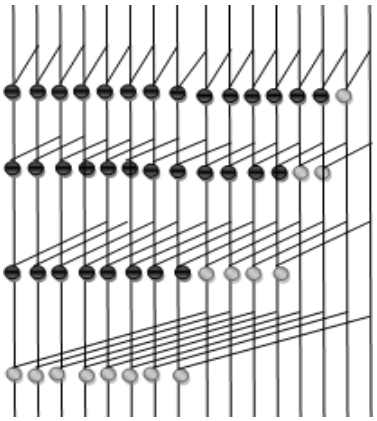


b)koggestone -16 bit architecture

The kogge stone tree achieves both $\log_2 N$ stages and fan out of 2 at eachstage. Recursive doubling is the technique used in this algorithm



c)kogge stone 32-bit architecture



d)kogge stone 64-bit architecture

Fig 3.5 architecture of Kogge Stone adder

The logical depth of this adder topology is log2N, the number of nodes per stages =N-2nwhere n=stage index (n=0,1,---n-1) and the total cost function is C(K)=Nlog2N-N+1. Though it has both optimal depth and low fan out, it produces high complex circuit realizations and also account for large number of interconnects. The main disadvantages are large number of wires, large computational nodes and high power dissipation. The entire architecture for 8, 16, 32 and 64 bit Kogge Stone adder is depicted.

VI.STIMULATION RESULT AND OUTPUT

4.1.INTRODUCTION TO SIMULATION TOOL

Introduction Simulation plays an important role in the design of integrated circuits. Using simulation, a designer can determine both the functionality and the performance of a design before the expensive and time-consuming step of manufacture. The ability to discover errors early in the design cycle is especially important for MOS circuits, where recent advances in manufacturing technology permit the designer to build a single circuit that is considerably larger than ever before possible.

Simulation is more than a mere convenience-it allows a designer to explore his circuit in ways which may be otherwise impractical or impossible. The effects of manufacturing and environmental parameters can be investigated without actually having to create the required conditions; the ability to detect manufacturing errors can be evaluated beforehand; voltages and currents can be determined without the difficulties associated with attaching a probe to a wire 500 times smaller than the period at the end of this sentence; and so on. To paraphrase a popular corporate slogan: without simulation, VLSI itself would be impossible!

To use a simulator, the designer enters a design into the computer, typically in the form of a list of circuit components where each component connects to one or more nodes. A node serves as a wire, transmitting the output of one circuit component to other components connected to the same node. The designer then specifies the voltages or logic levels of particular nodes, and calls upon the simulator to predict the voltages or logic levels of other

4.2.TOOLS USED:

The tools used for the simulation of various adder topologies like sklansky, Brent kung, Proposed adder I and Proposed adder II are Xilinx ISE and DSCH3.These are the tools used for the software implementation of various adders and to study their characteristic output like sum, carry and **delay for the corresponding inputs given.**

4.2.1XILINX ISE

Xilinx ISE (Integrated Synthesis Environment) is a software tool produced by Xilinx for synthesis and analysis of HDL designs, enabling the developer to synthesize ("compile") their designs, perform timing analysis, examine RTL diagrams, simulate a design's reaction to different stimuli, and configure the target device with the programmer. The Xilinx ISE is primarily used for circuit synthesis and design, while ISIM or the ModelSim logic simulator is used for system-level testing.

4.3.SIMULATION DESCRIPTION:

Thus the simulation process for adders helps us to study the output response and behaviour of various adder topologies and helps us to find the delay, which helps to choose the efficient adder with lesser delay by comparing it with the other existing standards.

The simulation result for some adders like Sklansky, Brent kung, proposed adder 1 and 2 are given below

4.3.1 STIMULATION RESULTS FOR SKLANSHY ADDER FOR VARIOUS BITS

The program for 8 bit, 16 bit, 32 bit and 64 bit sklansky adder is obtained and is verified and the corresponding output is noted. The output for an sklanshy adder is given below. Where the 3 input a, b and c are given as a, b and c (carry)=1 and the respective 2 outputs sum and carry are sum and carry=1. The carry generate signal g and the carry propogate signal p are generated The delay of the adder is found and noted.

Fig i. 8 bit sklanshy adder

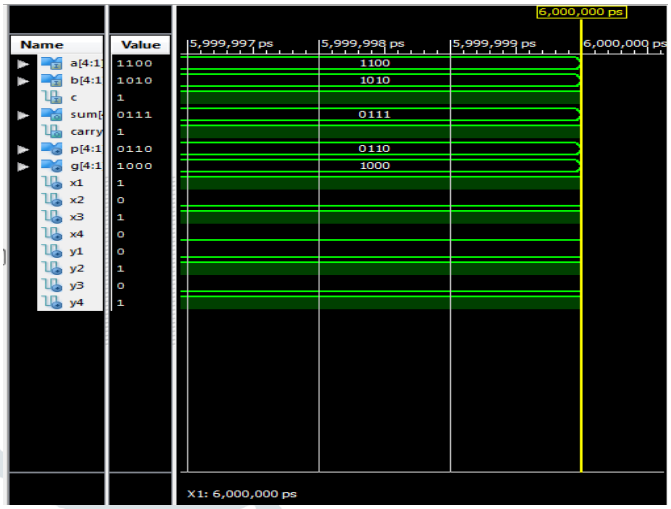


fig ii. 4 bit sklanshy adder

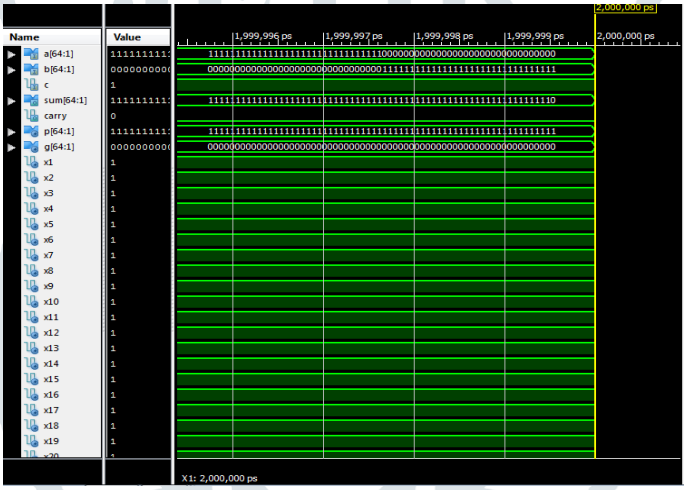


Fig iii. 64 bit sklanshy adder

STIMULATION RESULT OF BRENT KUNG

The program for 8 bit, 16 bit, 32 bit and 64 bit brent kung adder is obtained and is verified and the corresponding output is noted. The output for an brent kung adder is given below. Where the 3 input a, b and c are given and c (carry)=1 and the respective 2 outputs sum and carry are sum and carry=1. The carry generate signal g and the carry propogate signal p is found. The delay of the adder is found and noted.

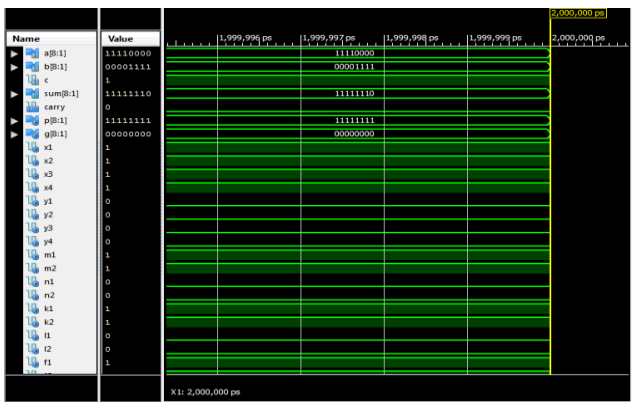


Fig i 8 bit brunt kung adder

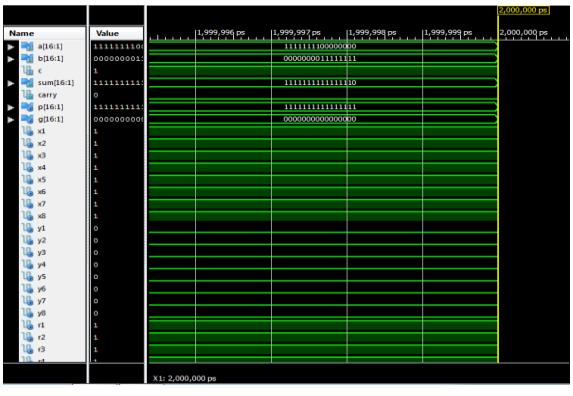


fig ii 16 bit brunt kung adder

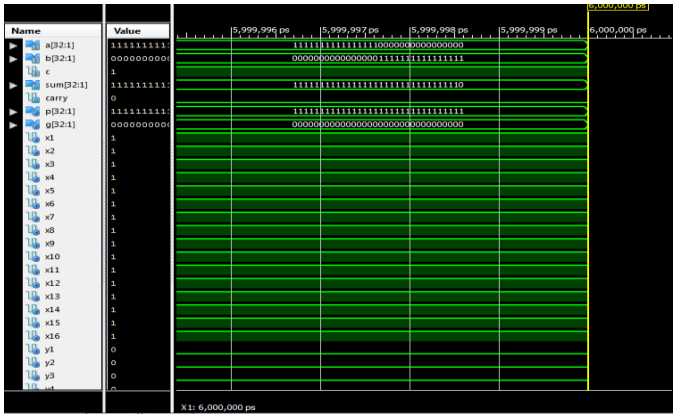


Fig iii 32 bit brunt kung adder

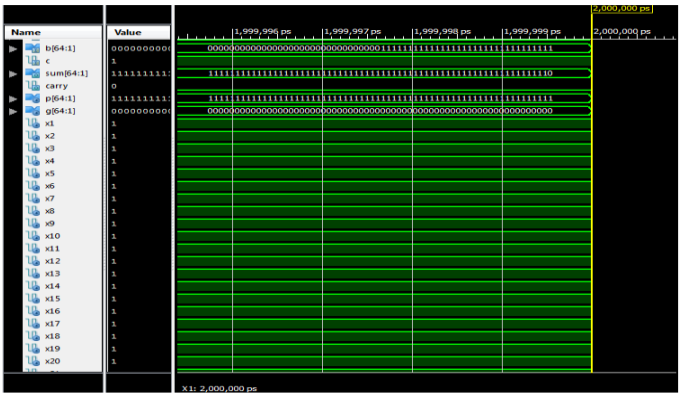


fig iv 64 bit brunt kung adder

4.3.3.STIMULATION RESULT OF 8 BIT PROPOSED ADDER 1

The program for 8 bit, 16 bit and 32 bit and 64 bit proposed adder 1 (ie. combination of ladner fisher and han carlson) is obtained and is verified and the corresponding output is noted. The output for an proposed-1 adder is given below. Where the 3 input a, b and c are given and c (carry)=1 and the respective 2 outputs sum and carry are noted. The carry generate signal g and the carry propogate signal p are found.. The delay of the adder is found.

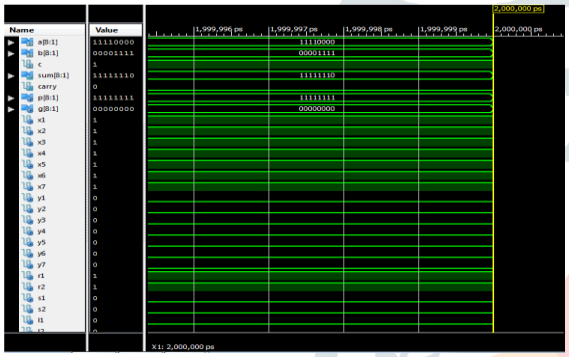


Fig i 8 bit proposed adder 1

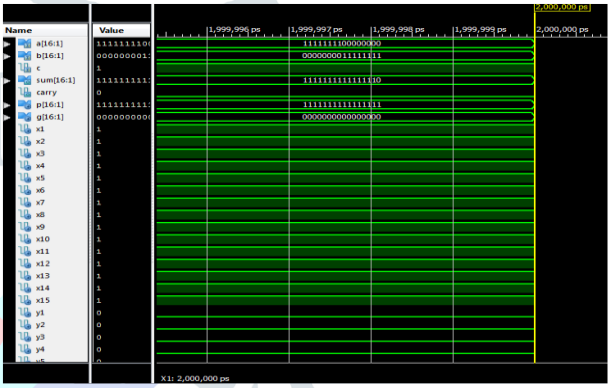


fig ii 16 bit proposed adder 1

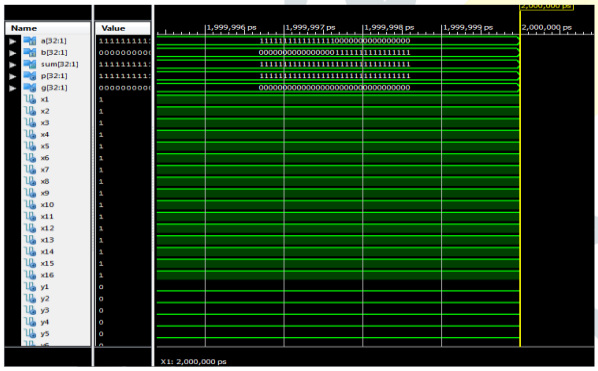


Fig iii 32 bit proposed adder 1

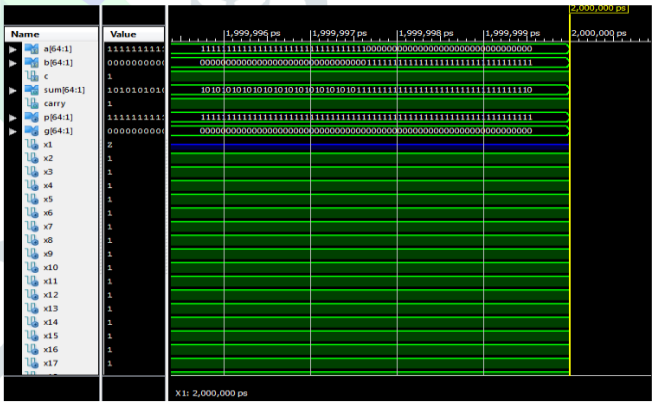


fig iv 64 bit proposed adder 1

4.3.4.STIMULATION RESULT OF 8 BIT PROPOSED ADDER 2

The program for 8 bit 16 bit 32 bit and 64 bit proposed adder 2 (ie. combination of kogge stone and ladner fischer) is obtained and is verified and the corresponding output is noted. The output for an 8 proposed-2 adder is given below. Where the 3 input a, b and c are given and the respective 2 outputs sum and carry are noted The carry generate signal g and the carry propogate signal p are found. The delay of the adder is found.

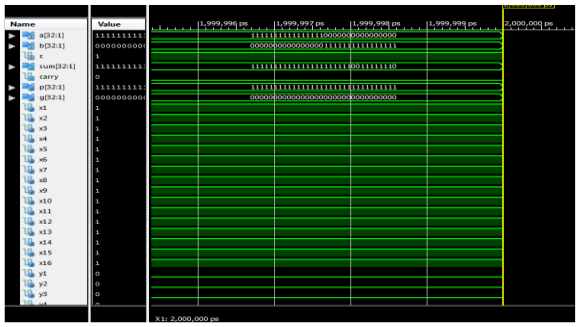


Fig i 32 bit proposed 2 adder

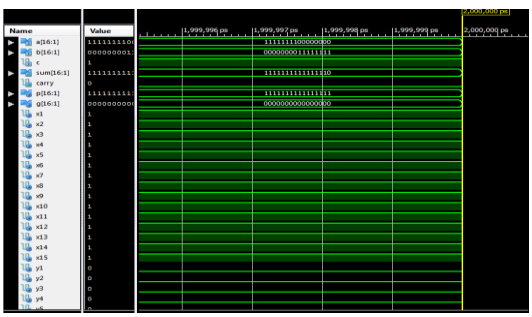


fig ii 16 bit proposed 2 adder

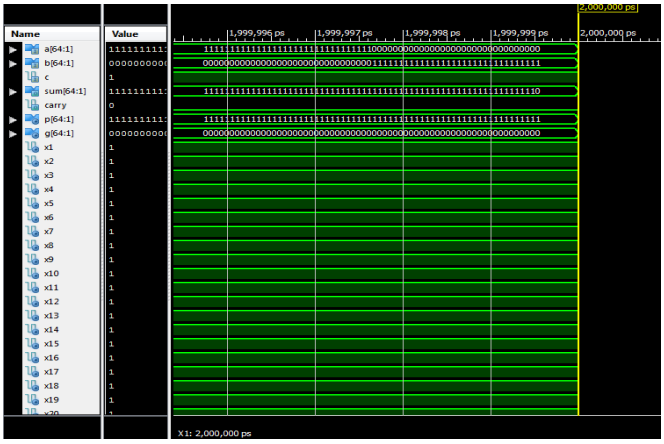


Fig iii 64 bit proposed 2 adder

V.Summary and conclusion

5.1 Summary

The work focus the design of low power high power parallel prefix adder(PPA) with minimum depth algorithm and its performance is compared with the existing architecture in terms of delay and area with Gate diffusion input technique(GDI) using XILIN ISE 12.1. Each adder type was implemented with bit size of 8-bit. Similarly when comparing the Sklansky adder and the brunt kung adder, occupies the maximum slices and delay, and the proposed adder utilise minimum slices and delay than the compared topologies.

ADDER TOPOLOGY	DELAY IN NANOSECONDS (ns)	NO OF SCLICES OCCUPIED	NO OF LUT'S	POWER (W)
Sklansky adder	25.830ns	121	218	788 W
Brent kung	30.204ns	119	220	716 W
Proposed adder I	18.36ns	117	208	694 W
Proposed adder II	22.10ns	115	202	532 W

5.2.Conclusion

Thus the parallel prefix adders were analysed from 8 to 64 bits with the help of xilinx tool in the FPGA architecture. From the already found parallel prefix adders, two adders have been proposed. These two adders shows the best performance in terms of speed and area occupation. They are very much reliable when compared to the peer parallel prefix adders. The parallel prefix adders are best suited for any kind of a circuit in order to give a high speed performance. The entire performance analysis is also made from which the above mentioned conclusions are derived. This satisfies the demand for a very high speed, less area and a much reliable performance adder circuit.From the analysis done on the above adders, it is found that all the adders with respective to the properties of ideal adder characteristics in the matter of wiring, number of computational nodes and the number of stages. In Sklansky adder, a minimum logic depth is presented with the least routing tracks. Due to the large fan out, the area circuit speed is also affected. This does not give the required decrease in the delay. Kogge stone has a large number of wirings and also the highest degree of power dissipation. But this adder gives the least delay when compared to the other parent adders. In Ladner Fischer adder, even though the wiring capacitance is very much reduced, there occurs a large amount of fan-outs. In the Brent Kung adder, there is the presence of too many logic levels and many buffers are used in this case. Therefore, the speed is not up to the mark. Han Carlson, being the hybrid of Brent Kung and Kogge-Stone gives a good performance with respect to the logic level, whereby increasing the speed of the circuit. Knowles adder gave a good improvement over the area and the power dissipation strategies. With all these parameters in mind the proposed adder was constructed which resulted in the adder giving the best performance in terms of speed and delay. This adder has the advantage of both Han Carlson adder and Ladner Fischer.

This high performance low power adder using GDI Technique can be used in the implementation of FIR filter design for DSP for low power applications. It is also in ALU which is the basic building block in VLSI design to make the arithmetic and logic operations more easier.

VI.REFERENCE:

[1]. R. Uma, Jebashini Ponnian, P. Dhavachelvan, "New low power adders in Self Resetting Logic with Gate Diffusion Input Technique", Engineering science 2017, 29th Journal of king saud university 118-134, 2017.

[2]. R. Uma, P. Vigneshwarababu, R. Nakkeeran and P. Dhavachelvan, "New Low- Power Reversible Logic Gates Using Gate Diffusion Input Technique", International conference on Emerging research computing information communication and applications, ERCICA 2013.
[3]. Uma.Ramadass and P. Dhavachelvan, "New Low Power Delay Element in Self Resetting Logic with Modified Gated Diffusion Input Technique", IEEE- ICSE 2012 Proc., 2012.
[4]. R.Uma and P. Dhavachelvan, "Modified gate diffusion input technique, A new technique for enhancing performance in full adder circuit", 2nd International conference on communication, computing and security(ICCS-2012)
[5]. R.Uma, Jebashini Ponnian, "Systolic FIR Filter Design with Various Parallel Prefix Adders in FPGA: Performance Analysis, International Symposium on Electronics System Design, 2012.
[6]. T. NagaBabu, D. Sounder, B. SubhakaraRao and P. Bose Babu, A Low Power Adder Using Reversible Logic Gates, *IJRET*, November (2012).
[7].M.Moghaddam and M. B. Ghaznavi-Ghoushchi ,"A New Low-Power, Low- area, Parallel Prefix Sklansky Adder with Reduced Inter-Stage Connections Complexity",IEEE Computer society,2011.
[8]. P.Ramanathan and P.T.Vanathi ," A Novel Power Delay Optimized32-bit Parallel Prefix Adder For High Speed Computing", International Journal of Recent Trends in Engineering, Vol 2, No. 6, November 2009.
[9].T. NagaBabu, D. Sounder, B. SubhakaraRao and P. Bose Babu, A Low Power Adder Using Reversible Logic Gates, *IJRET*, November (2012).
[10]. S Knowles, "A family of adders" Proceedings. 14th IEEE symposium on 1999.