International Journal of Electrical and Electronics Engineers
Vol. No. 9, Issue No. 01, January-June 2017
ISSN (O) 2321-2055
ISSN (P) 2321-2045

DESIGN AND IMPLEMENTATION 0F 64-BIT PARALLEL PREFIX BRENTKUNG ADDER

V. Jeevan Kumar[1], N.Manasadevi[2], A.Hemalatha[3], M.Sai Kiran[4], P.Jhansi Rani[5]

[1]Asst. Professor, [2,3,4,5]Student, Dept of ECE, Sri Sivani College of Engineering, JNTUK, A.P (India)

ABSTRACT

A parallel-prefix adder gives the most excellent performance in VLSI design. However, performance of Brent-kung adder through black cell takes large area. So, gray cell can be replaced instead of black cell which gives the Efficiency in Brent-kung Adder. The proposed system has two stages of operations they are preprocessing stage and generation stage. The preprocessing stage having propagate and generate. Generation stage focuses on carry generation and final result. In ripple carry adder each bit having addition operation is waited for the preceding bit addition operation. In efficient Brent – Kung adder, addition operation does not wait for preceding bit addition operation and modification is done at gate level to improve the speed and decreases the area

***Keywords*: Ripple carry adder, Efficient Brent–Kung adder, Black cell, Gray cell**

I. INTRODUCTION

Ripple carry adder is used for the addition task i.e., if N-bits addition operation is performed by the full adder with N- bits. In ripple carry adder each full adder operation consists of sum and carry, that carry will be given to next bit full adder operation, that processes is continuous till the Nth bit operation. The N-1th bit full adder operation carry will be given to the Nth bit full adder operation present in the ripple carry adder. [1]

Addition procedure is the main process in digital signal processing and control systems. The high-speed and accuracy of a processor or system depends on the adder performance. Multiplexer is combinational circuit which consists of multiple inputs and a single output. In general purpose processors and DSP processors the addition

The 3-bit ripple carry adder is shown in Fig.1. The first bit carry is given to second bit full adder and similarly the second bit carry is given to the third bit full adder. The addition process is performed from least significant bit to most significant bit in ripple carry adder[1]. Configuration logic and routing resources in Field Programmable Gate Array.

International Journal of Electrical and Electronics Engineers
Vol. No. 9, Issue No. 01, January-June 2017
ISSN (O) 2321-2055
ISSN (P) 2321-2045

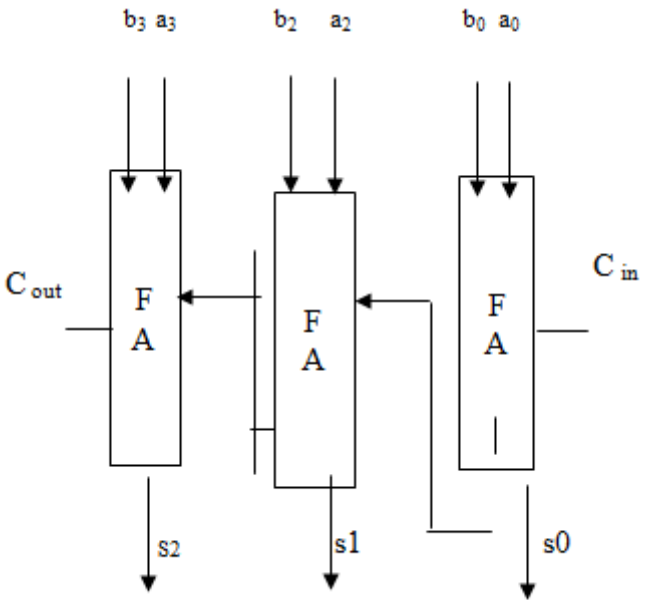


Fig1: 3-bit ripple carry adde

II. BRENT-KUNG ADDER

Brent Kung adder is used for high performance addition operation. The Brentkung is the parallel prefix adder used to perform the addition operation [3]. It is looking like tree structure to perform the arithmetic operation. The Brent-kung adder consists of black cells and gray cells. [2] Each black cell consists of two AND gates and one OR gate [4]. Each gray cell consists of only one AND gate.pi denotes propagate and it consists of only one AND gate[5] given in equation 1. gi denotes generate and it consists of one AND gate and OR gate given in equation 2. [6]

pi= Ai XOR Bi ---------- (1)

gi= Ai AND Bi ---------- (2)

Gi denotes carry generate and it consists of one AND gate and OR gate given in equation 3 used for first black cell. [8]

Gi=pi OR [gi AND cin] --- (3)

III. PROPOSEDBRENTKUNG ADDER

The proposed Brent-kung adder is flexible to speed up the binary addition and the arrangement looks like tree structure for the high performance of arithmetic operations.

Field programmable gate arrays [FPGA's] are mostly used in recent years because they improve the speed of microprocessor based applications like mobile communication, DSP and telecommunication. Research on binary operation fundamentals and motivation gives development of devices. The construction of efficient Brent-kung adder consists of two stages. They are preprocessing stage and generation stage.

3.1 Pre-Processing Stage:

International Journal of Electrical and Electronics Engineers
Vol. No. 9, Issue No. 01, January-June 2017
ISSN (O) 2321-2055
ISSN (P) 2321-2045

In the pre-processing stage, generate and propagate are from each pair of the inputs. The propagate gives "XOR" operation of input bits and generates gives "AND" operation of input bits [7]. The propagate (Pi) and generate (Gi) are shown in below equations 4 & 5.

$$Pi=Ai \text{ XOR } Bi \text{ ------ (4)}$$

$$Gi=Ai \text{ AND } Bi \text{ ------ (5)}$$

3.2 Generation Stage:

In this stage, carry is generated for each bit is called carry generate (Cg) and carry is propagate for each bit is called carry generate (Cp). The carry propagate and carrygenerate is generated for the further operation, final cell present in the each bit operate gives carry. The last bit carry will help to sum of the next bit simultaneously till the last bit. The carry generate and carry propagate are given in below equations 6 & 7.

$$Cp=P1 \text{ AND } P0 \text{ ---------- (6)}$$

$$Cg=G1 \text{ OR } (P1 \text{ AND } G0) \text{ ------ (7)}$$

The above carry propagate Cp and carry generation Cg in equations 6&7 is black cell and the below shown carry generation in equation 8 is cell i.e., gray cell. The carry propagate is generated for the further operation. The final cell present in the each bit operation gives carry. The last bit carry will lead tosum of the next bit simultaneously till the last bit. This carry is used for the next bit sum operate, the carry generate isgiven in below equations 8.

$$Cg=G1 \text{ OR } (P1 \text{ AND } G0) \text{ ------ (8)}$$

The carry of a first bit is XORed with the next bit of propagates then the output is given as sum and it is shown in equation 9.

$$Si=Pi \text{ XOR } Ci\text{-}1 \text{ ------- (9)}$$

It is used for two thirty-two bit addition operations and each bit undergoes preprocessing stage and generation stage then gives the final sum.

The first input bits goes under preprocessing stage and they will produce propagate and generate. These propagates and generates undergoes generation stage produces carry generates and carry propagates then gives final sum. The step by step process of efficientBrent-kung adderis shown in Fig.2.

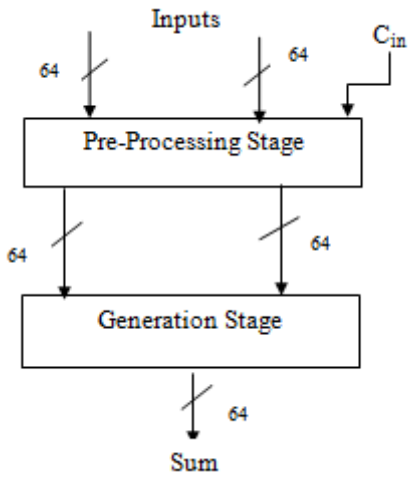


Fig.2: Block Diagram

International Journal of Electrical and Electronics Engineers
Vol. No. 9, Issue No. 01, January-June 2017
ISSN (O) 2321-2055
ISSN (P) 2321-2045

The efficient Brent-kungadder arrangement is looking like tree structure for the high performance of arithmetic operations and it is the high speed adder which focuses on gate level logic. It designs with a reduction of number of gates. So, it decreases the delay and memory used in this architecture.

The efficient Brent-kung adder is shown in fig.3 which improves the speed and decrease the area for the operation of 16-bit addition. The input bits Ai and Bi concentrates on generate and propagate by XOR and AND operations respectively. The propagates and generates undergoes the operations of black cell and gray cell and gives the carry Ci. That carry is XORed with the propagate of next bit, that gives sum.

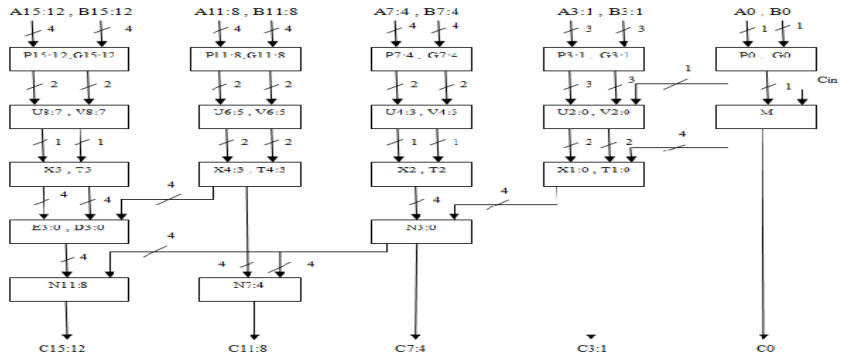


Fig.3: 16-Bit Efficient Brent-kung Adder

The properties of the operations are evaluated in parallel with accept the trees to overlap which leads to parallelization. The architecture of Efficient Brent-kung adder gives the less delay and less memory for the operation of 16-bit addition.

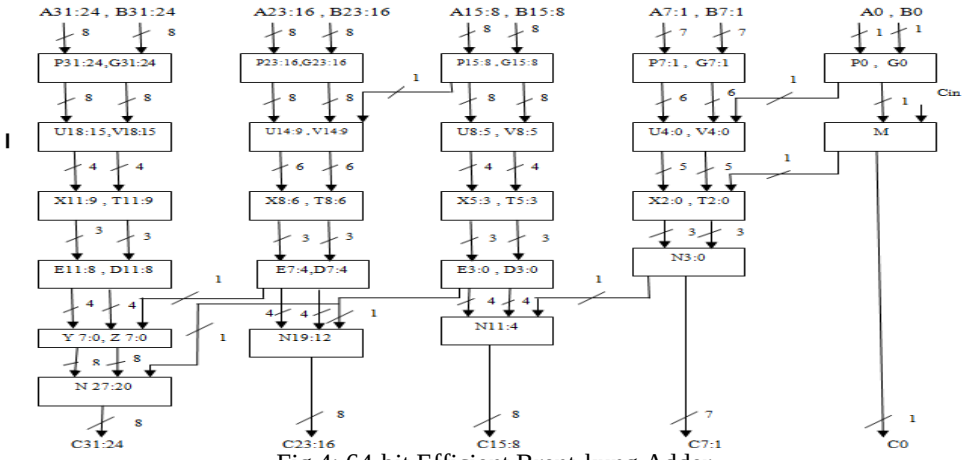


Fig.4: 64-bit Efficient Brent-kung Adder

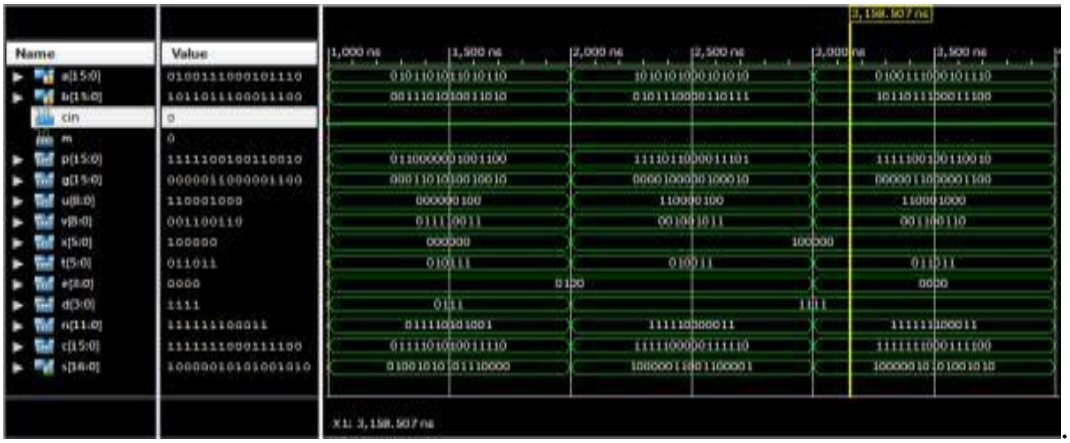
The architecture of 64-bit Efficient Brentkung adder is shown in Fig.4. The logical circuit is using multiple adders to find the ans i.e., sum of N-bit numbers. Each addition operation has a carry input (Cin) which is the previous bit carry output (Cout).

Research on binary addition innovatively motivates gives development of devices. Many parallel prefix networks describe the literature of parallel addition operation. The parallel prefix adders are Brent-kung, Kogge-stone, brent-kung, Sklansky, etc,. The fast and accurate performance of an adder gives to used in the very large scale integrated circuits design and digital signal processors.

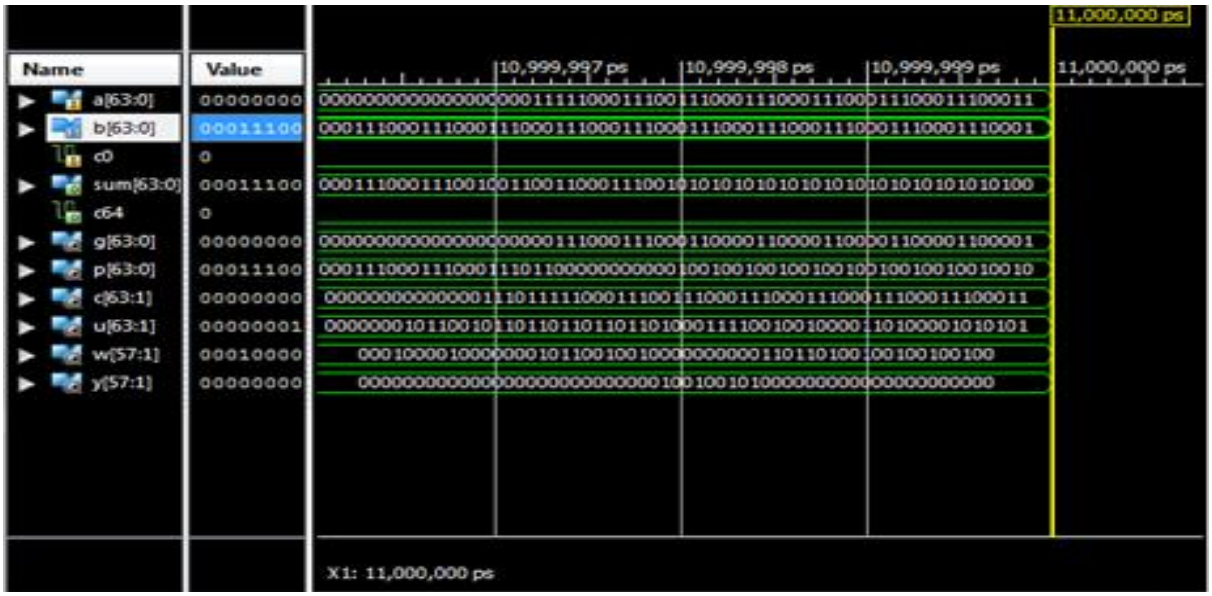
International Journal of Electrical and Electronics Engineers
Vol. No. 9, Issue No. 01, January-June 2017
ISSN (O) 2321-2055
ISSN (P) 2321-2045

IV. SIMULATION RESULTS

The Efficient Brent-kung adder is design with an VHDL (very high speed integration hardware description language). Xilinx project navigator 14.1 is used and Simulation results of 16-bit efficient Brent-kung are shown in Fig.5



The architecture of 64-bit Efficient Brentkung adder is shown in Fig.4 and the Simulation results is shown in Fig.6.



The design of adders is done on VHDL. The memory and delay performance Efficient Brent-kungadder (EBK) is shown in Table.1

International Journal of Electrical and Electronics Engineers
Vol. No. 9, Issue No. 01, January-June 2017
ISSN (O) 2321-2055
ISSN (P) 2321-2045

Adder	Delay(ns)	Memory used(MB)
8-bit Efficient Brent-kung adder	11.2	181
16-bit Efficient Brent-kung adder	12.2	184
64-bit Efficient Brent-kung adder	13.275	208.9

Table.1: Delay and memory used in EBK

V. CONCLUSION

In this paper, new approaches to design an efficient Brent-kung adder look like tree structure and cells in the carry generation stage are decreased to speed up the binary addition. It concentrates on gate levels to perk upthe speed and decreases the memory used.The proposed adder addition operation offers elude great advantage in reducing delay.

REFERENCES

[1] Pakkiraiah chakali, madhu kumar patnala "Design of high speed Brent - Kung based carryselect adder" IJSCE march 2013

[2] David h,k hoe, Chris Martinez and sri jyothsna vundavalli "Design and characterization of parallel prefix adders using FPGAs", Pages.168- 172, march2011 IEEE.

[3] K.Vitoroulis and A.J. Al-Khalili, "performance of parallel prefix adders implemented with FPGA technology," IEEE Northeast Workshop on circuits and systems, pp.498-501, Aug. 2007.

[4] Haridimos t.vergos, Member, IEEE and Giorgos Dimitrakopoulos, Member, IEEE,"On modulo 2n+1 adder design"IEEE Trans on computers, vol.61, no.2, feb 2012

[5] Giorgos Dimitrakopoulos and Dimitris Nikolos, "High-Speed Parallel-Prefix VLSI Ling Adders" IEEE Trans on computers, vol.54, no.2, Feb. 2005.

[6] S.Knowles,"Afamily ofadders," Proc.15 th Symp. Comp. Arith.,pp.277-281,June2001.

[7] R.BrentandH.Kung,"Aregular layout for parallel adders,"IEEE Trans. Computers, vol. C-31,no.3, pp. 260-264,March1982.

[8] R.E. Brent and M.J. Kung, "Parallel Prefix Computation," J. ACM, vol. 27, no. 4, pages