AN EFFICIENT THREE STAGES 32-BIT LADNER-FISCHER ADDER

B.Raghavaiah[1] Omprakash[2]
[1]Research Scholar, Shri JJT University, Rajasthan, India
[2]Shri JJT University, Rajasthan, India

Abstract
Ladner –Fischer adder is one of the parallel prefix adders. Parallel prefix adders are used to speed up the process of arithmetic operation. These adders are introduced to eliminate the carry propagation delay that rises in ripple carry adder. Ladner-Fischer adder is parallel prefix adder that performs the addition operation. It consists of three stages they are pre-processing stage, carry propagation and generation and post propagation. Pre-processing stage comprises of generate and propagate. This paper proposed the design and analysis of a 32-bit Ladner-Fischer adder

Keywords: Ladner-fischer adder, Post propagation, Carry generation, Verilog HDL, Xilinx

1. INTRODUCTION

Addition operation is the main operation in digital signal processing and control systems. The fast and accuracy of a processor or system depends on the adder performance. In general purpose processors and DSP processors the addition operation addresses are taken from simple ripple carry adder .

Ripple carry adder is used for the addition operation i.e., if N-bits addition operation is performed by the N-bit full adder. In ripple carry adder each bit full adder operation consists of sum and carry, that carry will be given to next bit full adder operation, that process 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. For 16-bit ripple carry adder, the first bit carry is given to second bit full adder, second bit carry is given to the third bit full adder, similarly the operation is continue till fifteenth bit carry is given to sixteenth bit full adder. The addition operation is performed from least significant bit to most significant bit in ripple carry adder. Configuration logic and routing resources in Field Programmable Gate Array.

PARALLEL PREFIX ADDERS:

Description

In a PPA, a prefix operation is constructed that permits the computation of intermediate carries. In conjunction with generate and propagate signals, the prefix operator allows PPAs to obtain an advantageous latency of *O(log2N)* instead of *O(N)* (like in a Ripple Carry adder), where N is the word length. Figure 5 (below) illustrates a tree graph of an 8-bit Kogge-Stone PPA.

Functionality

The PPA functions properly by successfully going through three stages. The first stage is a pre-computation stage, which generates the generate and propagate signals, and is governed by the equations:

$$g_i = G_{i:i} = a_i \; AND \; b_i \text{ ----- } p_i = P_{i:i} = a_i \; XOR \; b_i \text{ ---} \mathbf{(1)}$$

where a_i and b_i are inputs bits. Additionally,

$$g_0 = G_{0:0} = C_{in} \text{ ----- } p_0 = P_{0:0} = 0 \text{ ---} \mathbf{(2)}$$

The prefix operation serves as the middle stages in this high performance adder topology. A dot operator denotes the prefix operation:

$$(G_{j:mv} \; P_{j:mv}) = (G_{j:k} \; P_{j:k}) \bullet (G_{l:mv} \; P_{l:m}) \text{ ---} \mathbf{(3)}$$

$$G_{j:m} = G_{j:k} \; OR \; (P_{j:k} \; AND \; G_{l:m}) \text{ ----- } P_{j:m} = P_{j:k} \; AND \; P_{l:m} \text{ ---} \mathbf{(4)}$$

where $j \leq (k, l) \leq m$. The dot operator is associative and idempotent, but not commutative.

The Carry logic is also utilized in the middle stages. By utilizing equations (3) and (4), the prefix tree can determine the carry:

$$c_i = G_{i:0} \text{ ---} \mathbf{(5)}$$

The last stage in this high performance PPA is the sum computation. The sum is governed by the following equation:

$$s_i = c_i \; XOR \; p_i \text{ ---(6)}$$

2 Ladner-Fischer Adder:

The Ladner-Fischer is the parallel prefix adder used to perform the addition operation. It is looking like tree structure to perform the arithmetic operation [4]. Ladner-Fischer adder is used for high performance addition operation. The Ladner-Fischer adder consists of black cells and gray cells [3]. Each black cell consists of two AND gates and one OR gate [2]. Multiplexer is combinational circuit which consists of multiple inputs and a single output. 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 [6] given in equation 2.

$P_i = B_i$ AND B_{i-1}

$G_i = A_i$ OR [B_i AND A_{i-1}]

Where

G_i denotes generate and P_i denotes propagate

3 PROPOSED LADNER-FISCHER ADDER

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

In ripple carry adders each bit wait for the last bit operation. In parallel prefix adders instead of waiting for the carry propagation of the first addition, the idea here is to overlap the carry propagation of the first addition with the computation in the second addition, and so forth, since repetitive additions will be performed by a multioperand adder.

Research on binary operation elements and motivation gives development of devices. Field programmable gate arrays [FPGA's] are most popular in recent years because they improve the speed of microprocessor based applications like mobile DSP and telecommunication. The construction of efficient Ladner-Fischer adder consists of three stages. They are pre-processing stage, carry generation stage, post-processing stage.

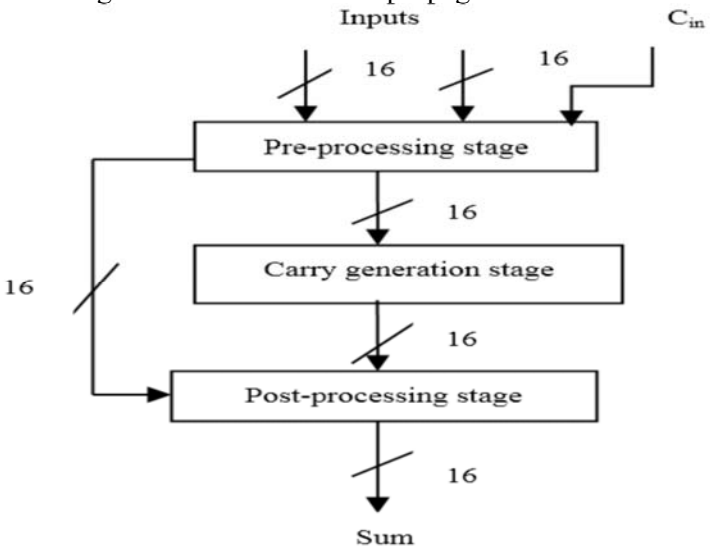


Fig 3: Block diagram proposal three stages Ladner-Fischer adder

3.1 Pre-Processing Stage:

In this stage generate and propagate are done for each stage of inputs. Propagate is done through XOR operation and generate is done through AND operation.

$P_i = A_i$ XOR B_i

$G_i = A_i$ AND B_i

3.2 Carry Generation :

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

$C_p = P_i$ AND P_{i-1}

$C_g = G_i$ OR (P_i AND G_{i-1})

The above carry propagate Cp and carry generation Cg in equations above is black cell and the below shown carry generation in equation is gray cell. The carry propagate is generated for the further operation but final cell present in the each bit operation gives carry. The last bit carry will help to produce sum of the next

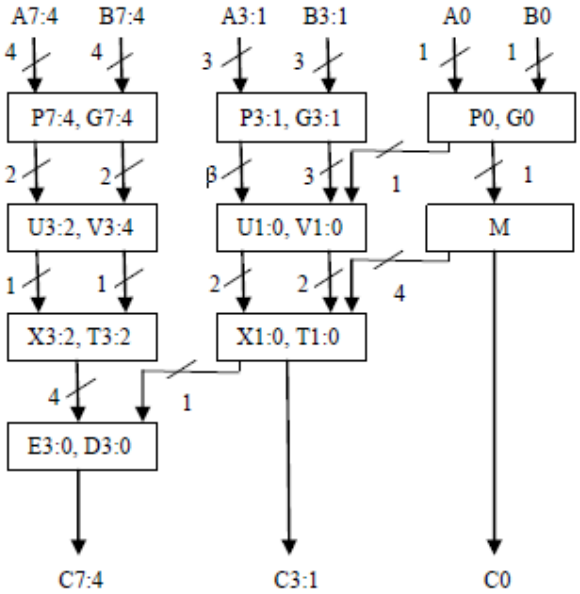
bit simultaneously till the last bit. This carry is used for the next bit sum operation, the carry generate is given in below equations .

3.3 Post processing stage:

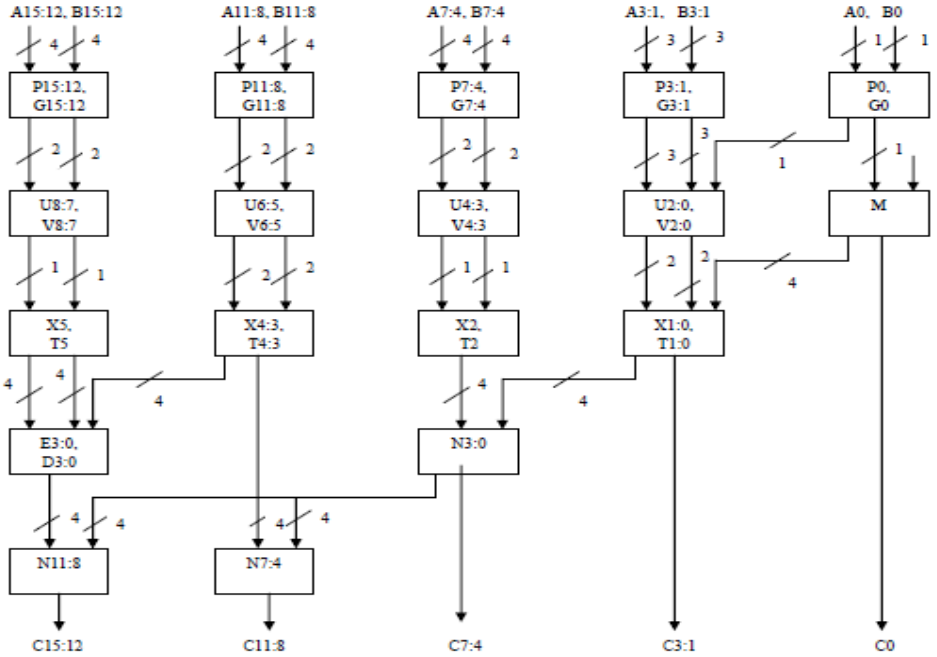
The first input bits goes under pre-processing stage and it will produce propagate and generate. These propagates and generates undergoes carry generation stage produces carry generates and carry propagates, these undergoes post-processing stage and gives final sum.

$S_i = P_i$ AND C_{i-1}.

The below figure represents 8-bit ladner fischer adder



The below figure represents 16bit Ladner-Fischer adder

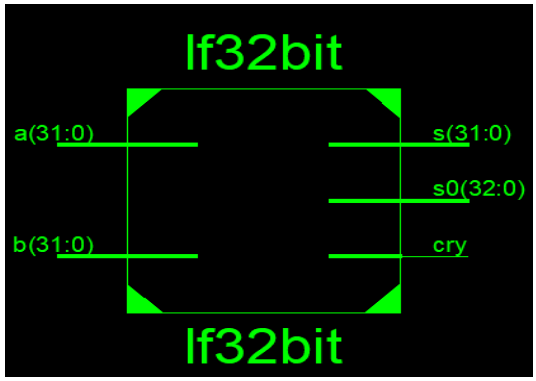


Research on binary addition motivates gives development of devices. Many parallel prefix networks describe the literature of addition operation. The parallel prefix adders are Brent-kung, Kogge-stone, ladner-Fischer, Sklansky, etc,. The fast and accurate performance of an adder is used in the very large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

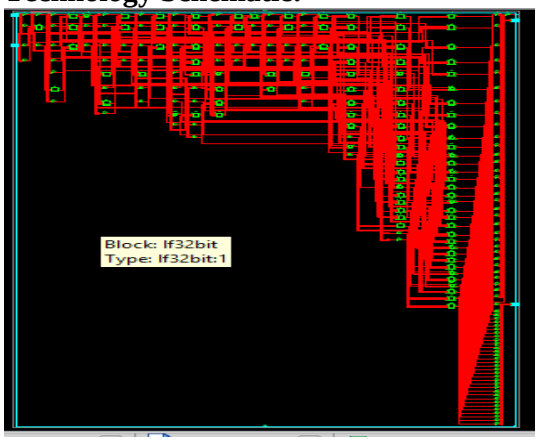
4 SIMULATION RESULTS :

The Ladner-Fischer adder is designed on Verilog HDL (verification Logic hardware description language). Xilinx project navigator 14.7 is used for synthesis. Simulation results are shown in Fig below.

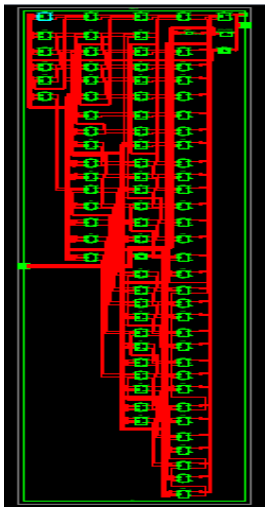
RTL-TOP:



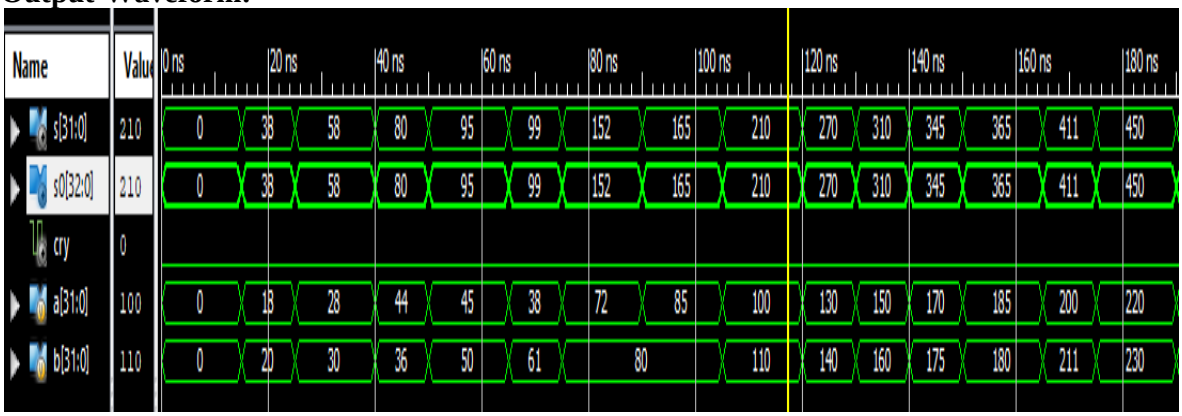
Technology Schematic:



RTL-Internal:



Output-Waveform:



Comparison Table:

TYPE OF ADDER	No. Of LUT'S	DELAY	POWER
16-bit Ladner Adder	38	16.353 ns	2.88 Watts
32-bit Ladner Adder	85	25.998 ns	6.46 Watts

5. CONCLUSION

In this paper, a new approach to design an efficient Ladner-Fischer adder concentrates on gate levels to improve the speed and decreases the memory. It is like tree structure and cells in the carry generation stage are decreased to speed up the binary addition. The proposed adder addition operation offers great advantage in reducing delay.

7. REFERENCES

[1] Pakkiraiah. Chakali, madhu Kumar. Patnala "Design of high speed Ladner - Fischer based carry select adder" IJSCE march 2013

[2] 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

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

[4] 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.

[5] Giorgos Dimitrakopoulos *and* Dimitris Nikolos, "High Speed Parallel Prefix ...*Ling adders,"* IEEE Transactions *on* Computers, vol.54, no.2, February 2005

[6] S. Knowles," A family of adders,"proc.15thsymp. Comp. Arith, pp. 277-281, June 2001.

[7] R. Brent and H. Kung, "A regular layout for parallel adders," IEEE Trans. Computers, vol.C-31, no. 3, pp. 260-264, March 1982.

[8] R.E. Ladner and M.J. Fischer, "Parallel Prefix Computation," J. ACM, vol. 27, no. 4, pages 831-838, Oct. 1980.