DESIGN AN ADDER USING PARALLEL PREFIX LADNER FISCHER TECHNIQUE

G. Lakshmi Prasanna,M.Tech Scholar,Chaitanya Institute of Science and Technology, Madhavapatnam.
Email:prasanna.prassu431@gmail.com
G.Jaya Lakshmi, Assistant prof.,Dept.of E.C.E,Chaitanya Institute of Science and Technology, Madhavapatnam.
Email:jayalakshmi433ece@gmail.com

Abstract— Parallel-Prefix adders or Carry tree adders are the kind of adders that uses prefix operation in order to do efficient addition. Nowadays Parallel-Prefix adders are the frequently used adders due to the high speed computation properties. So called carry tree adder uses the prefix operation to do the arithmetic addition with way greater speed than the simple parallel adders that is ripple carry adder, carry skip adder, carry select adder etc. Here in this paper we will discuss about the various parallel-prefix adders and analyses there delay. However, performance of kogge stone adder through black cell takes huge memory. So, kogge stone adder is replaced with ladnerfischer adder with simplified structure, which gives the Efficiency in Ladner-Fischer Adder. In general adder each bit of addition operation is waited for the previous bit addition operation. In Ladner - Fischer adder, addition operation does not wait for previous bit addition operation and modification is done at gate level to improve the speed and to decreases the memory used.
INDEX TERMS: Ripple carry adder, Ladner–Fischer adder, Black cell, Gray cell

I.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.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 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 $N-1^{th}$ bit full adder operation carry will be given to the Nth bit full adder operation present in the ripple carry adder. 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.
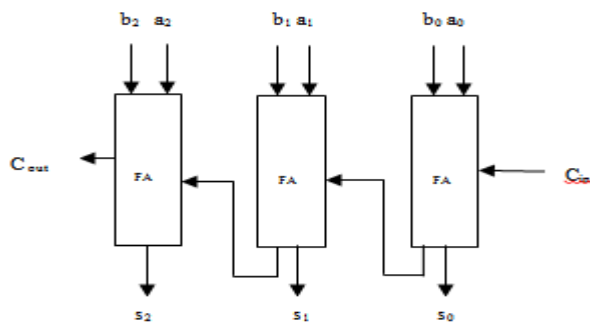

Fig1: Three Bit Ripple Carry Adder

II KOGGE STONE ADDER

The kogge stoneis the parallel prefix adder used to perform the addition operation. It is looking like tree structure to perform the arithmetic operation. Kogge stone adder is used for high performance addition operation. This adder consists of black cells and gray cells. Each black cell consists of two AND gates and one OR gate. Multiplexer is combinational circuit which consists of multiple inputs and a single output. Each gray cell consists of only one AND gate.P_i denotes propagate and it consists of only one AND gate given in equation 1. G_i denotes generate and it consists of one AND gate and OR gate given in equation 2.

$$P_i=B_i \text{ AND } B_{i-1} \text{ -------------- (1)}$$
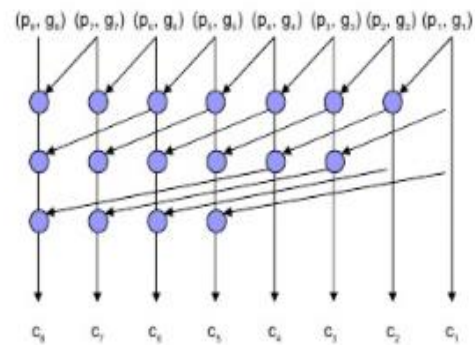$$G_i=A_i \text{ OR } [B_i \text{ AND } A_{i-1}] \text{ ---- (2)}$$

Fig 2: kogge Stone Adder

G_i denotes generate and it consists of one AND gate and OR gate given in equation 3.

$$G_{i-1}=A_{i-2} \text{ OR } [B_{i-2} \text{ AND } A_{i-1}] \text{ --- (3)}$$

III LADNER FISCHER ADDER

The 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.

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

Pre-Processing Stage:

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

$$P_i = A_i \text{ XOR } B_i \text{ ------ (4)}$$
$$G_i = A_i \text{ AND } B_i \text{ ------ (5)}$$

Carry Generation Stage:

In this stage, carry is generated for each bit and this is called as carry generate (C_g). 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 6 & 7.

$$C_p = P_1 \text{ AND } P_0 \text{ ----------------- (6)}$$
$$C_g = G_1 \text{ OR } (P_1 \text{ AND } G_0) \text{ ------ (7)}$$

The above carry propagate C_p and carry generation C_g in equations 6&7 is black cell and the below shown carry generation in equation 8 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 8.

$$C_g = G_1 \text{ OR } (P_1 \text{ AND } G_0) \text{ ------ (8)}$$

Post-Processing Stage:

It is the final stage of an Ladner-Fischer adder, 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.

$$S_i = P_i \text{ XOR } C_{i-1} \text{ ------- (9)}$$

It is used for two sixteen bit addition operations and each bit carry is undergoes post-processing stage with propagate, gives the final sum.

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. The step by step process of Ladner-Fischer adderis shown in Fig 3.

The Ladner-Fischer adder structure is looking like tree structure for the high performance of arithmetic operations and it is the fastest adder which focuses on gate level logic. It designs with less number of gates. So, it decreases the delay and   memory used in this architecture.
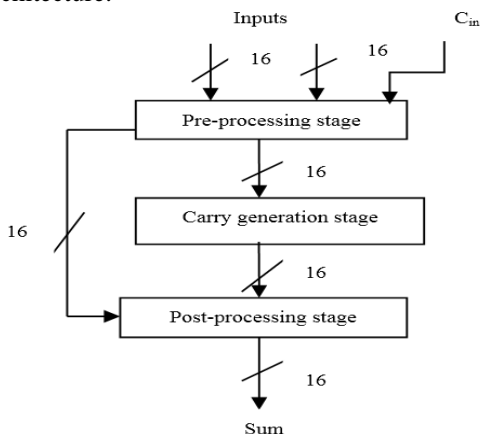

Fig 3: Block Diagram

The Ladner-Fischer adder is shown in fig 4 which improves the speed and decreases the area for the operation of 8-bit addition. The input bits A_i and B_i concentrates on generate and propagate by XOR and AND operations. These propagates and generates undergoes the operations of black cell and gray cell and gives the carry C_i. That carry is XORed with the propagate of next bit, that gives sum.
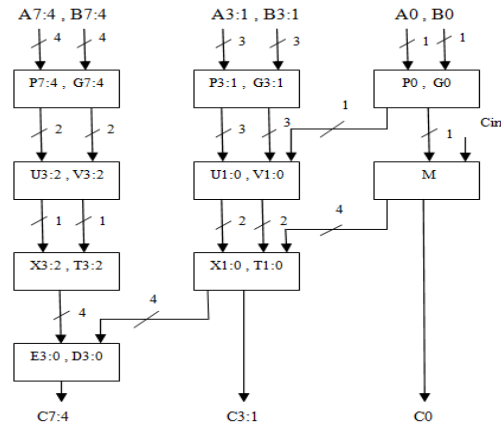

Fig 4: 8-Bit Ladner-Fischer Adder

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

The architecture of 16-bit Ladner-Fischer adder is shown in Fig 5. The logical circuit is using multiple adders to find the sum of N-bit numbers. Each addition operation has a carry input (C_{in}) which is the previous bit carry output (C_{out}).
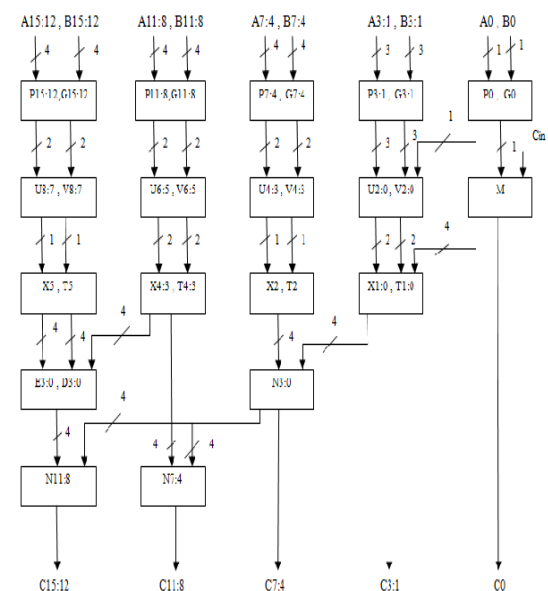

Fig 5:16-bit Ladner-Fischer adder

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.

IV SIMULATION RESULTS

The Ladner-Fischer adder is designed on VHDL (very high speed integration hardware description language). Xilinx project navigator 12.1 is used for synthesis. Simulation results are shown in Fig 6.
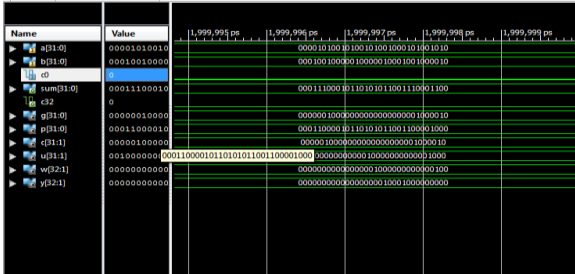

Fig 6: 32-Bit Ladner-Fischer Adder Simulation Waveform

Fig 7: Summary report

TABLE 1: COMPARISION TABLE

ADDER	DELAY[nano secs]	MEMORY [kilo bytes]
32 bit kogge stone adder	35.2	245685
32 bit ladner fischer adder	30.2	203876

V CONCLUSION

In this paper, a new approach to design an Ladner-Fischer adder concentrates on gate levels to improve the speed and decreases the area. Kogge-Stone, Han-Carlson and Knowles adders require a large number of parallel wiring for wide bit adders. Thus packing the wires close together will increase the coupling capacitance on each wire. The proposed system consists of three stages of operations they are pre-processing stage, carry generation stage, post-processing stage. The pre-processing stage focuses on propagate and generate, carry generation stage focuses on carry generation and post-processing stage focuses on final result. 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.

REFERENCES

[1] pakkirai ahchakali, madhu kumar patnala "Design of high speed Ladner – Fischer based carry select adder" IJSCE march 2013.

[2] David h,k hoe, Chris Martinez and sri jyothsna vundavalli "Designand 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]Giorgos Dimitra kopoulos and Dimitris Nikolos, "High-Speed Parallel-Prefix VLSI Ling Adders" Feb. 2005.

[5]S. Knowles, "Afamily of adders," Proc.15th Symp. Comp. Arith., pp.277-281,June2001.

[6] I. Sutherland, R. SproullandD. Harris, LogicalEffort,SanFrancisco:MorganKaufmann,1999.

[7]R.Brentand H.Kung,"A regular layout for parallel adders,"IEEETrans.Computers,vol.C-31,no.3,   pp.260-264,March1982

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

[9]P.Kogge and H.Stone,"A Parallel algorithem for the solution of a general class of recurrence relation," IEEE transactions on computers, vol. c-22, no.8, pp.786-793, Aug 1973.

[10]J.Sklansky,"Conditionalsumadditionlogic,"IRETrans.Electr onicComputers,vol.EC-9,pp.226-231,June1960.