Optimization and Analysis area of Carry lock Ahead Adder

Mohita Sharma, Himanshu Joshi

ABSTRACT:
An adder is a hardware blocks in almost digital and high performance system with advance in technology. Many researchers have tried and trying to design adder which offer either of high speed, low power consumption and low delay and small area. Thus making them suitable for various parameters
In this research paper carry Look Ahead Adder and Ripple Carry Adder both Adders are 32 bit adders. These adders defined the low delay, small area and high speed. In this research paper all the simulation done on XILINX 14.2 Design suit and ALTER QUARTU II S12.2 software.

Keywords— Communication, CPU Execution time, Xillinx, Tanner Tools EDA.

I. INTRODUCTION

Digital electronics, or digital circuits, represent signals by discrete bands of analog levels, rather than by a continuous range. All levels within a band represent the same signal state relatively small changes to the analog signal levels due to manufacturing tolerance, signal attenuation or noise do not leave the discrete envelope, and as a result are ignored by signal state sensing circuitry. In most cases the number of these states is two, and they are represented by two voltage bands: one near a reference value and a value near the supply voltage, corresponding to the false (0) and true (1) values of the Boolean domain respectively. Digital techniques are useful because it is easier to get an electronic device to switch into one of a number of known states than to accurately reproduce a continuous range of values. Digital electronic circuits are usually made from large assemblies of logic gates, simple electronic representations of Boolean logic functions. Digital systems are used extensively in computation and data processing, control systems, communications, and measurement. Because digital systems are capable of greater accuracy and reliability than analog systems, many tasks formerly done by analog systems are now being performed digitally. In a digital system, the physical quantities or signals can assume only discrete values, while in analog systems the physical quantities or signals may vary continuously over a specified range. For example, the output voltage of a digital system might be constrained to take on only two values such as 0 volts and 5 volts, while the output voltage from an analog system might be allowed to assume any value in the range 10 volts to 10 volts. Because digital systems work with discrete quantities, in many cases they can be designed so that for a given input, the output is exactly correct. For example, if we multiply two 5-digit numbers using a digital multiplier, the 10-digit product will be correct in all 10 digits. On the other hand, the output of an analog multiplier might have an error ranging from a fraction of one percent to a few percent depending on the accuracy of the components used in construction of the multiplier. Furthermore, if we need a product which is correct to 20 digits rather than 10, we can redesign the digital multiplier to process more digits and add more digits to its input. A similar improvement in the accuracy of an analog multiplier would not be possible because of limitations on the accuracy of the components. The design of digital systems may be divided roughly into three parts—system design, logic design, and circuit design. System design involves breaking the overall system into subsystems and specifying the characteristics of each subsystem. For example, the system design of a digital computer could involve specifying the number and type of memory units, arithmetic units, and input-output devices as well as the interconnection and control of these subsystems. Logic design involves determining how to interconnect basic logic building blocks to perform a specific function. An example of logic design is determining the interconnection of logic gates and flip-flops required to perform binary addition. Circuit design involves specifying the interconnection of specific components such as resistors, diodes, and transistors to form a gate, flip-flop, or other logic building block. Most contemporary circuit design done in integrated circuit form using appropriate computer aided design tools to layout and interconnect the components on a chip of silicon.[1]

II. PROBLEM DEFINITION AND REQUIREMENT ANALYSIS

According the first research paper i observed some conclusion and result based on, no of slices, levels of

logics and processing time. This is shown in table no.1 for different 8- bit and 16- bit adder with the help of xillen 9.2 software. According the above table Ripple Carry Adder ,Carry Look Ahead Adder and Carry Skip Adder ,these adder is design by XOR Gate and this paper results shows that Ripple Carry Adder consist total no1 of 16 XOR Gate as an better than the Carry Look Ahead Adder and Carry Skip Adder. Now when I consider the processing time than i got the less processing time of Carry Look Ahead Adder with other adders likes Ripple Carry Adder and Carry Skip Adder. Because in Ripple Carry Adder the propagation delay is greater than the Carry Look Ahead Adder, means that in Ripple Carry Adder excution of pervious carry input to next carry input takes more time as compare to Carry Look Ahead Adder. [5]

TABLE I
COMPARISON OF DIFFERENT ADDERS

S. no	Parameters	Ripple carry Adder	Carry-look Ahead Adder	Carry Skip Adder
1	Xor 1 bit	16	32	32
2	No of Slices	18/960	18/960	21/960
3	Level of Logic	18	18	15
4	Processing time	3.77s	3.555s	4.67s

According to second research paper I observed some result and conclusion for different 4-bit and 8-bit Adders, which is shown in below graph. This all simulation is done on TANNER EDA TOOL version 13.1. In this paper the performance of the CLA has been measured by comparing the results in terms of propagation delay, power dissipation and Power Delay Product. [6]

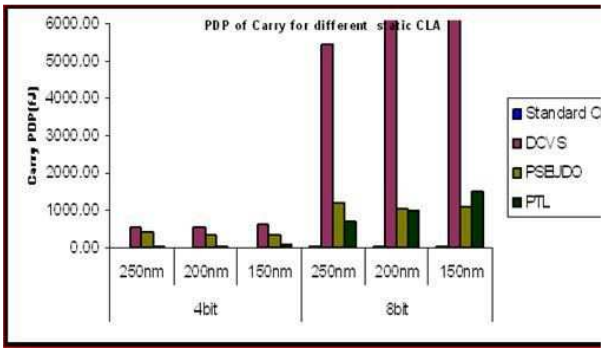

Fig. 1 Power delay for 4-Bit and 8-Bit on different channel length

In this paper design a high speed and low area consumption carry look ahead adder. So for excution of this simulation I have required some software's. This software is VERILOD HDL, XILLEN, ALTERA and MODEL SIM.

III. PROPOSED CARRY LOCK AHEAD ADDER

In this paper defined the proposed carry lock ahead using Xillinx software. In this paper present the speed, delay and area. Carry lock ahead based on Xillinx software and using 32 bit Xor gate. Carry lock ahead delay is high compare to other 2adder like ripple carry adder but CPU execution time is low.

Carry- Look ahead Adder (CLA) is designed to overcome the latency introduced by the repelling effect of the carry bits in RCA. The CLA improves speed by reducing the amount of time required to determine carry bits. Carry look ahead logic uses the concepts of generating (G) and propagating (P) carries. Its work is based on two signals called P and G for each bit position. The P and G are shown bellow.

$C_i +1 = G_i + P_i.C_i$

Here $G_i = A_i.B_i$

$S = P_i$ xor C_i

and $P_i = (A_i$ xor $B_i)$

$S_i = A_i$ xor B_i xor C_i

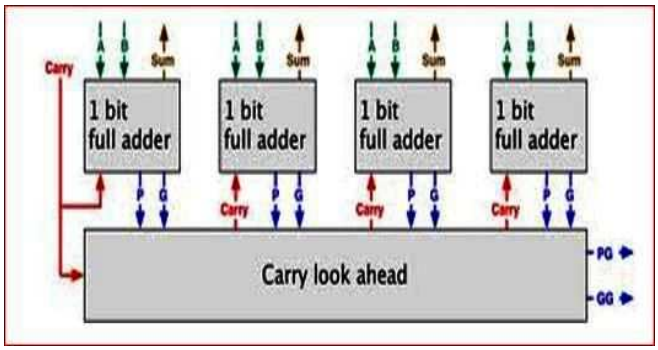

Fig. 2 Carry lock ahead diagram

The Si and Ci+1 represent the sum and carry from ith full adder respectively. The carry-look ahead adder can be broken up in two modules: (1) The Partial Full Adder, PFA, which generates Si, Pi and Gi. (2) The Carry Look- Ahead Logic, which generates the carry-out bits. The structure of CLA for 4-bit adder is shown in figure 2. The addition of two numbers in parallel implies that all bits of the augends and the addend are available for computations at the same time. In a parallel adder the carry output of each stage is connected to the carry input of the non integer order stage (ripple carry) . Therefore the sum and carry output of any stage cannot be produced until the input carry occurs. This lead output of any stage cannot be

produced until the input carry occurs. This lead to a time delay in the addition process .This delay is known as carry propagation delay. Since each bit sum output depends on the value of inputs carry, the value of Si in any stage in the adder will be in its steady state final value only after the inputs carry to that stage has been propagated . The carry propagation time is the limiting factor on the speed with which two numbers are added in parallel. Although a parallel adder will always have same value at its output terminals. The output not is correct unless the signals are given enough time to propagate through the connected from the input to the outputs'.

(1) One method for reducing the carry propagation delay time is to employ faster gates with reduced delays.

(2) the most widely used techniques employed the principle of look ahead carry. This method utilizes logic gates to look at the lower order bits of the augends and addend if a higher order carry is to add. Its uses two functions: carry generate and carry propagate.[1]

IV. RESULT

According to table no2 I have observed and calculated the different results parameter.

TABLE III
COMPARISON BETWEEN CARRY LOCK AHEAD AND RIPPLE CARRY ADDER

Adder	**Real time to Xst Completion (sec)**	**CPU time Xst Completion (sec)**	**Total Delay**	**Memory usage**	**Level of Logic**
Ripple carry adder	17.00	17.63	7.312	416692	18
Carry Lock Ahead Adder	17.00	17.17	8.488	420020	18

The ripple carry adder taken 17.63 nsec processing time and Carry Look Ahead Adder takes 17.17 nsec processing time means the Carry Look Ahead Adder is best one and the processing speed is higher than the Ripple Carry Adder. Third column shows the total delay known as routing delay .The routing delay time for Carry Look Ahead Adder is greater than Ripple Carry Adder. Fourth column shows the total memory usage by Ripple Carry Adder and Carry Look Ahead Adder in kilobyte. And last column defines how many level of logics used by both Adders.

V. CONCLUSIONS

The conclusions of this research I have design High Speed and low Delay Carry Look Ahead Adder. In this research I used XILINX 14.2 design suit software and for FPGA implementation use ALTERA QUARTS 12.2 design suit software. After implementation and simulation on these software I get High speed Carry look Ahead Adder. In this paper define the CPU execution time. Carry lock ahead adder CPU execution time is low compare to ripple carry adder.

ACKNOWLEDGMENT

Mohita Sharma would like to thank Mr. Himanshu Joshi Asst.Prof in the Department of ECE who had been guiding throughout the project and supporting me in giving technical ideas about the paper and motivating me to complete the work efficiently and successfully

REFERENCES

[1] Stephen brown book " Fundamental of digital logic with verilog"

[2] Mohammad Hassena Begum, V .Varmsi Mohan ,Krishna " Design and verification of low power and area efficient kogge – stone carry select adder". IJERT Page no.462 vol-2 (2013)

[3] Paolo Zicari, Stefania Perri " A fast carry chain adder for virtex-5 FPGA" IEEE Journal page no. 304 (2010)

[4] R.UMA.P.Dhavachelvan "Logic optimization using technology independent Mux based adders in FPGA" VLSICS vol-3 (2012)

[5] Prashant Gurjar , Rashmi Solanki, Pooja Kansliwal, Mahendar Vucha " VLSI implementation of adders for high speed ALU" IJCA vol-29(2011)

[6] Jagannath Samanta ,Mausam Halder , Bisbnu Prasad De " Performance Analysis of high speed ,low power carry look ahead adder using different logic style" IJSCE vol no. 2 (2013)