Effective Performance Analysis of Multiplier Using CLAA and CSLA Through a Literature Review

Anu Mangal[1], Prof. Amrita Khera[2] & Prof. Saurabh Sharma[3]
[1]M-Tech Research Scholar, [2]Research Guide, [3]HOD, Department of Electronics & Communication
Trinity Institute of Technology & Research, Bhopal

Abstract- A multiplier is one of the basic building blocks in the most digital systems, FIR filters, and microprocessors digital signal processors etc. With advances in technology, several researchers have tried to design multipliers which can provide high speed, low power consumption, regularity of layout and less area or even a combination of them in a multiplier. For different applications designing of various high speeds, low power and compact VLSI implementations is needed. Area and speed are two conflicting constraints. Multiplication goes in two basic steps partial product and then addition. Therefore in this review paper we have tried to analyze the adders and compare their speed and complexity of circuit. Carry Select Adders are having high speed but posses a larger area. And a Carry Look Ahead Adder is in between the spectrum having a proper trade-off between time and area complexities.

Keywords: Carry Look Ahead Adder, Carry Select Adder (CSLA) ,Delay, Area.

I. INTRODUCTION

Addition is the most common and often used arithmetic operation in microprocessor, digital signal processor, especially digital computers. Also, it serves as a building block for synthesis all other arithmetic operations. Therefore, regarding the efficient implementation of an arithmetic unit, the binary adder structures become a very critical hardware unit. In any book on computer arithmetic, someone looks that there exists a large number of different circuit architectures with different performance characteristics and widely used in the practice. Although we may get many of them dealing with the binary adder structures have, the studies based on their comparative performance analysis are only a few. Qualitative evaluations of the classified binary adder architectures are given. Among the huge member of the adders that wrote VHDL (Hardware Description Language) code for Ripple-carry, Carry-select and Carry-look ahead to emphasize the common performance properties belong to their classes. With respect to asymptotic delay time and area complexity, the binary adder architectures can be categorized into four primary classes. The first class consists of the very slow ripple-carry adder with the smallest area. In the second class, the carry-skip, carry-select adders with multiple levels have small area requirements and shortened computation times. From the third class, the carry-look ahead adder and from the fourth class, the parallel prefix adder represents the fastest addition schemes with the largest area complexities.

The module generator offer a wide variety of adder architectures that can be selected according to the power and performance needs of the design. The implementations include ripple carry adder, carry select, Carry Look-Ahead adders and Brent and Kung adder. A carry can either be generated in an adder stage or propagated from the previous one. CLAAs use these generate (G) and propagate (P) values of all previous stages to determine the carry of a particular stage. This combination of previous stage GP values can be done in many ways depending on the architecture used. Accordingly there are three variants of Carry Look-Ahead adders having power-delay-area trade-offs.

Ripple Carry Adder

This is the simplest implementation of adder which is formed by cascading full adders in series as given in fig. 1.1. A full adder computes the sum and carry for each stage. The Carry Out of a full adder stage is applied as a Carry In to the next stage.

The main advantage of this architecture is its small area; though, it has large delay that increases linearly with the bit width. Both, delay and area are of order O (n).

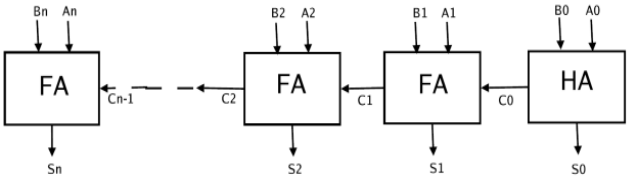

Fig. 1.1: Ripple Carry Adder

II. SYSTEM MODEL

Carry Select Adder (CSLA)

Carry select adder is based on anticipation of output carry for two possible values of input carry. Once the real value of the incoming carry is known, the correct result is easily selected with a simple multiplexer stage [1]. Carry select adder can be

implemented in two different ways 1) Linear carry select adder 2) Square-root carry select adder. Consider the block of adders, which is adding bits k to k+3. Instead of waiting on the arrival of the output carry of bit k-1, both the 0 and 1 possibility is analyzed. A multiplexer can be used to select either of the results when $C_{0k-1,}$ settles. The hardware overhead of the carry select adder is an additional carry path and a multiplexer. A basic structure of linear carry select adder is given in Fig. 1.2.

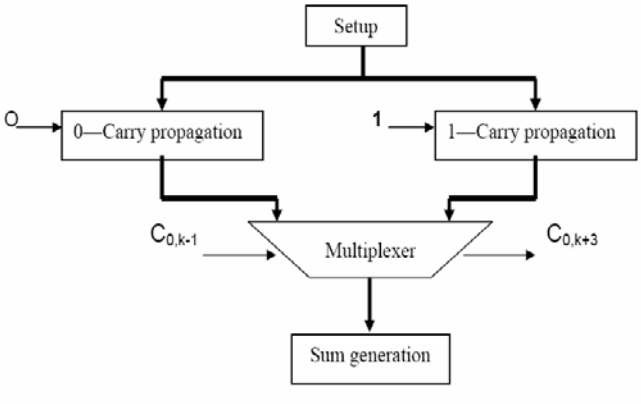

Fig. 1.2: Carry select module

A full carry select adder is now built using a chain of equal-length adder stages, as in carry bypass adder. The propagation delay of N-bit adder with M-bit stages can be determined as follows.

$$T_{add} = t_{setup} + Mt_{carry} + [N/M]\, t_{max} + t_{sum} \qquad 1.1$$

The carry select adder can be optimized to reduce the delay of critical path by making the adder stages progressively longer than the previous ones. This architecture results in square root dependence and is called square root carry select adder. For example, the first stage can add 2 bits, the second consists of 3, and the third has 4.

A carry select adder for n bits is divided into n/s stages of s bits each. The sum and carry for each stage is first computed separately considering input carry as '0' as well as '1' and then one of these is selected depending on the actual Carry In from the previous stage [7]. This circuit has more area (almost twice) than ripple carry adder as each stage consists of an extra s-bit adder and a multiplexer. Carry Select has a lower delay than ripple carry as the carry and sum are computed for each block in advance.
A block diagram for n-bit carry select adder with s-bit blocks is given in Fig. 1.3. Here, 0-Carry and 1-Carry are s-bit adders with input carry as 0 and 1 respectively.

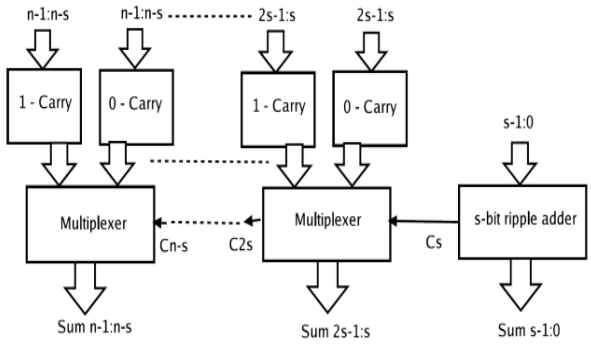

Fig. 1.3: Carry Select Adder

Carry look-ahead adder (CLA)

The concept behind the CLA is to get rid of the rippling carry present in a conventional adder design. The rippling of carry produces unnecessary delay in the circuit. For a conventional adder the expressions for sum and carry signal can be written as follows.

$$S = A \oplus B \oplus C \qquad 1.2$$

$$C_0 = AB + BC + AC \qquad 1.3$$

It is useful from an implementation perspective to define S and Co as functions of some intermediate signals G (generate), D (delete) and P (propagate) [1]. G =1 means that a carry bit will be generated, P=1 means that an incoming carry will be propagated to C_0. These signals are computed as

$$G = AB \qquad 1.4$$
$$P = A \oplus B \qquad 15$$

Now write S and C0 in terms of G and P.

$$C_0\,(G,P) = G + PC \qquad 1.6$$
$$S(G,P) = P \oplus C \qquad 1.7$$

For an N-bit adder, the following relation holds for the carry signal.

$$C_{0,k} = G_k + P_k C_{0,k-1} \qquad 1.8$$

In a fully expanded form [1], then

$$C_{0,k} = G_k + P_k(G_{k-1} + P_{k-1}\left(\ldots + P_1\left(G_0 + P_0\, C_{i,0}\right)\right)) \qquad 1.9$$

For k =4, the logic expressions are as

$$C_4 = G_3 + G_2 P_3 + G_1 P_2 P_3 + G_0 P_1 P_2 P_3 + C_0 P_0 P_1 P_2 P_3 = G_3 + P_3 C_3 \quad 1.9$$

$$C_3 = G_2 + P_2 + G_0 P_1 P_2 + C_0 P_0 P_1 P_2 = G_2 + P_2 C_2 \quad 1.10$$

$$C_2 = G_1 + G_0 P_1 + C_0 P_0 P_1 = G_1 + P_1 C_1 \quad 1.11$$

$$C_1 = G_0 + C_0 P_0 \quad 1.12$$

This expanded relationship is used to implement N-bit adder. The carry and sum outputs are independent of the previous bits. The ripple effect has been eliminated [1]. The logic design of a 4-bit CLA is given in Fig. 1.4.

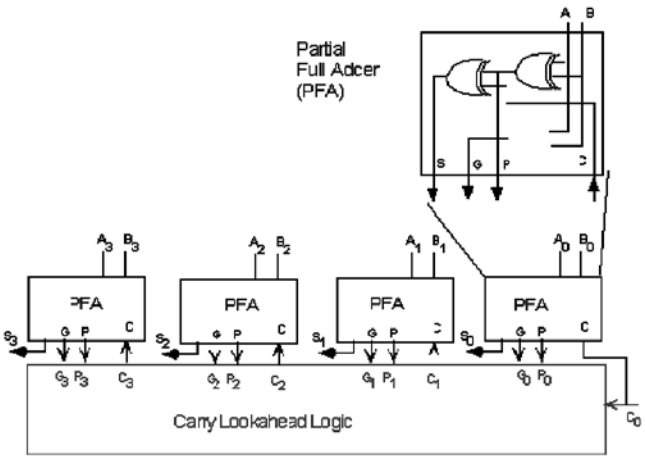

Fig. 1.4: 4-bit carry look ahead (CLA)

This expanded relationship is used to implement N-bit adder. The carry and sum outputs are independent of the previous bits. The ripple effect has been eliminated [1]. The logic design of a 4-bit CLA is given in Fig. 1.4. Full carry CLA is impractical for wide words. Since wide gates and large stacks display poor performance, the CLA computation has to be limited to up to 2 or 4 bits in practice. For example, the equation for C31 consists of 32 product terms, the largest of which contains 32 literals. Thus the required AND and OR functions must be realized by tree networks, leading to increased latency and cost [3]. Two schemes for managing this complexity are 1) High-radix addition 2) Multilevel lookahead. Multilevel lookahead is the most widely used technique for large CLAs.

III. LITERATURE REVIEW

Vijayalakshmi, V.; Seshadri, R.; Ramakrishnan, S. [1] deals with the comparison of the VLSI design of the carry look-ahead adder (CLAA) based 32-bit unsigned integer multiplier and the VLSI design of the carry select adder (CSLA) based 32-bit unsigned integer multiplier. Both the VLSI design of multiplier multiplies two 32-bit unsigned integer values and gives a product term of 64-bit values. The CLAA based multiplier uses the delay time of 99ns for performing multiplication operation where as in CSLA based multiplier also uses nearly the same delay time for multiplication operation. But the area needed for CLAA multiplier is reduced to 31% by the CSLA based multiplier to complete the multiplication operation. These multipliers are implemented using Altera Quartus II and timing diagrams are viewed through avan waves.

Xuan-Vy Luu; Trong-Thuc Hoang; Trong-Tu Bui; Anh-Vu Dinh-Duc [6] proposed architecture consists of a modified Radix-4 Booth encoder, a modified Wallace Tree adder, and a Carry Look Ahead adder. The design has been verified successfully on DE2-115 and then synthesized to ASIC implementation. The FPGA-based experimental result shows that it has the resources of 1788 ALUTs. The synthesized result occupies an area of 58.28 mm^2 with 4.13 ns total delay (i.e. 242.13MHz maximum frequency).

Balpande, R.S.; Patel, S., [9] Design of constant delay logic style for high speed adder researched on the low power microelectronics that has become more intense and low power VLSI systems having emerged as greatly in demand. For increasing number of portable applications require small area low power high throughput. High speed high throughput, small silicon area and at the same time low power consumption is the motivation behind this. This research presents an effective approach of constant delay (CD) logic style targeting at high-speed applications. Characteristic of this CD logic style without concern of the logic type makes it suitable in implementing complicated logic expressions such as addition. A “timing window” technique is also proposed to reduce the amount of excessive power dissipation in the CD approach. The concept is validated through the implemented of 32-bit Carry look ahead adder with Constant delay logic style using Tanner EDA v13.0.

Rajesh, A.; Madhumalini, M. [10] Carry select adder is fastest adder but it required more area and power. The modern VLSI design systems are small in size and less power consumption so the modification is need in the carry select adder to achieve the reduced area and less power consumption. Two proposed works are introduced in this paper. First method include the reduction of area and power in Carry select adder by modifying the EX-OR gate and

BINARY TO EXCESS converter. Modification of Ex-or gate is done inside the RIPPLE CARRY ADDER. Second method includes the reduction of delay by replacing CARRY LOOK AHEAD instead of Ripple carry adder. This replacement increases the area, this can be overcome by modifying the EX-OR gate in CARRY LOOK AHEAD block.

Preethi, K.; Balasubramanian, P. [11] proposed that the FPGA based realization of high-speed carry look-ahead adders based on the concept of section-carry is discussed. Three kinds of carry look-ahead adder architectures viz. Type 1, Type 2, Mixed are presented. In comparison with conventional carry look-ahead adders of sizes 16, 32 and 64-bits, the proposed section-carry based carry look-ahead adders statement improvements in speed of 14.9%, 12.1% and 13% for Type 1, Type 2 and Mixed topologies respectively, for implementations targeting a 90nm FPGA device.

Chaitanya kumar, M.V.S.; Selva kumar, J. [12] Abstract: Carry Look-ahead Adder (CLA) is implemented by using Dual Mode Logic (DML) topologies. DML logic switches between the static and dynamic mode of operations. In dynamic mode achieves higher performance with increase in power consumption and in static mode, DML logic achieves low power dissipation albeit with reduced performance. This feature allowed implementing CLA by selection of carry path based on input vectors. A 4-bit CLA was designed in 45nm TSMC technology using Cadence Virtuoso Design. Outcomes showed gain in speed albeit with increase in power and area when compared to the conventional CMOS logic.

Inoue, K.; Takeuchi, N.; Yamanashi, Y.; Yoshikawa, N., [13] authors have been studying ultra-low-power logic circuits using adiabatic quantum-flux-parametron (AQFP) logic, whose power consumption can be decreased significantly by changing their potential energies adiabatically using AC exciting currents. In this study, they designed an 8-bit carry look-ahead adder (CLA) using an AQFP cell library based on the AIST Nb 2.5 kA/cm^2 standard process (STP2) and evaluated its circuit properties. The CLA is composed of 1224 critically-dumped 50-μA Josephson junctions, which occupy the area of 1.74 × 0.99 mm^2. Simulation outcomes show that the CLA has a wide bias margin of ±29.5%, and the energy dissipation is 16.4 aJ per clock cycle at 5-GHz operation.

Efstathiou, C.; Owda, Z.; Tsiatouhas, Y., [14] Abstract: In this brief, an efficient implementation of an 8-bit Manchester carry chain (MCC) adder in multioutput domino CMOS logic is proposed. The carries of this adder are computed in parallel by two independent 4-bit carry chains. Due to its limited carry chain length, the use of the proposed 8-bit adder module for the implementation of wider adders leads to significant operating speed improvement compared to the corresponding adders based on the standard 4-bit MCC adder module.

IV. RESEARCH APPROACH

The basic reason of this research work is to study and design an Efficient, Fast and Low Power Multiplier. As the name suggests it had to go for faster and low power factor optimization simultaneously. As it is known that the basic building block of a multiplier is ADDER circuit.

Therefore we turned our focus into the CSLA and CLAA first. We studied the time delay consumed by different adders and found out a proper relation between time and area complexity of all the adders under consideration. Multiplication is an important arithmetic operation. Multipliers are much larger than adders and also more power consuming. Also, multiplication is a slow operation Multiplication is actually a process of addition of multiple partial-products. These partial products are formed by operating the multiplicand by each bit/bits of the multiplier.

V. CONCLUSION

The range of integration keeps rising; additional signal processing systems are implemented on a VLSI chips. These signal processing uses not only demand of great computation capability but also that consume considerable amount of energy for reducing the delay. While delay and area are to the two most important design parameters and the power utilization has become a significant concern in today's VLSI systems design. The requirement for low-power VLSI system arises from two main forces.

We analyzed different adders among compared them by different criteria like time and then Area-Delay Product for designing a multiplier. In this review paper we studied which adder is best suited for situation 32 bit unsigned multiplier. After comparing all CSLA and CLA we came to a conclusion that Carry Select Adders are best suited for situations where Speed is the only criteria. Carry Look Ahead Adder has the least Area-Delay product that tells us that, it is suitable for situations where both low power and fastness are criteria.

REFERENCES

[1] Vijayalakshmi, V.; Seshadri, R.; Ramakrishnan, S., "Design and implementation of 32 bit unsigned multiplier using CLAA and CSLA," in *Emerging Trends in VLSI, Embedded System, Nano Electronics and Telecommunication System (ICEVENT), 2014 International Conference on* , vol., no., pp.1-5, 7-9 Jan. 2014.

[2] W. Stallings, Computer Organization and Architecture Designing for Peljormance, 71h ed., Prentice Hall, Pearson Education International, USA, 2006, ISBN: 0-13-185644-8.

[3] 1. F. Wakerly, Digital Design-Principles and Practices, 4th ed., Pearson Prentice Hall, USA, 2006. ISBN: 0131733494.

[4] A. Sertbas and R.S. Ozbey, "A performance analysis of classified binary adder architectures and the VHDL simulations", J Elect. Electron. Eng., Istanbul, Turkey, vol. 4, pp. 1025 1030,2004.

[5] P. S. Mohanty, "Design and Implementation of Faster and Low Power Multipliers", Bachelor Thesis. National Institute of Technology, Rourkela, 2009.

[6] Xuan-Vy Luu; Trong-Thuc Hoang; Trong-Tu Bui; Anh-Vu Dinh-Duc, "A high-speed unsigned 32-bit multiplier based on booth-encoder and wallace-tree modifications," in *Advanced Technologies for Communications (ATC), 2014 International Conference* on , vol., no., pp.739-744, 15-17 Oct. 2014.

[7] Saptalakar, B.K.; Saptalakar, S.K.; Navalagund, S.S.; Latte, M.V., "VLSI implementation of reduced resource allocation for modified carry look-ahead adder," in *Advanced Communication Control and Computing Technologies (ICACCCT), 2014 International Conference on* , vol., no., pp.559-564, 8-10 May 2014.

[8] Luu, J.; McCullough, C.; Sen Wang; Huda, S.; Bo Yan; Chiasson, C.; Kent, K.B.; Anderson, J.; Rose, J.; Betz, V., "On Hard Adders and Carry Chains in FPGAs," in *Field-Programmable Custom Computing Machines (FCCM), 2014 IEEE 22nd Annual International Symposium on* , vol., no., pp.52-59, 11-13 May 2014.

[9] Balpande, R.S.; Patel, S., "Design of constant delay logic style for high speed adder," in *Green Computing Communication and Electrical Engineering (ICGCCEE), 2014 International Conference on* , vol., no., pp.1-5, 6-8 March 2014.

[10] Rajesh, A.; Madhumalini, M., "An efficient structure of carry select adder," in *Green Computing Communication and Electrical Engineering (ICGCCEE), 2014 International Conference on* , vol., no., pp.1-5, 6-8 March 2014

[11] Preethi, K.; Balasubramanian, P., "FPGA implementation of synchronous section-carry based carry look-ahead adders," in *Devices, Circuits and Systems (ICDCS), 2014 2nd International Conference on* , vol., no., pp.1-4, 6-8 March 2014

[12] Chaitanya kumar, M.V.S.; Selva kumar, J., "Dual Mode Logic Carry Look ahead Adder," in *Advanced Communication Control and Computing Technologies (ICACCCT), 2014 International Conference on* , vol., no., pp.537-540, 8-10 May 2014.

[13] Inoue, K.; Takeuchi, N.; Yamanashi, Y.; Yoshikawa, N., "Simulation and implementation of an 8-bit carry look-ahead adder using adiabatic quantum-flux-parametron," in *Superconductive Electronics Conference (ISEC), 2013 IEEE 14th International* , vol., no., pp.1-3, 7-11 July 2013.

[14] Efstathiou, C.; Owda, Z.; Tsiatouhas, Y., "New High-Speed Multioutput Carry Look-Ahead Adders," in *Circuits and Systems II: Express Briefs, IEEE Transactions on* , vol.60, no.10, pp.667-671, Oct. 2013

[15] Suri, L.; Lamba, D.; Kritarth, K.; Sharma, G., "High performance and power efficient 32-bit carry select adder using hybrid PTL/CMOS logic style," in *Automation, Computing, Communication, Control and Compressed*

[16] Manju, S.; Sornagopal, V., "An efficient SQRT architecture of Carry Select adder design by Common Boolean logic," in *Emerging Trends in VLSI, Embedded System, Nano Electronics and Telecommunication System (ICEVENT), 2013 International Conference on* , vol., no., pp.1-5, 7-9 Jan. 2013.