COMPARISON OF 32-BIT HYBRID ADDERS IN VHDL

Mr. Viraj V. Gotmare[1], Dr. Pankaj Agrawal[2]
Department of Electronics and Communication Engineering
GHRAET, Nagpur, Maharashtra, India

Abstract-This paper describes the comparison between the hybrid adders. Adders are always used in many data-processing systems to perform fast arithmetic operations. The carry select adder (CSA) is a high speed adder. It provides good compromise between RCA and CLA. The ripple carry adder (RCA) has a most compact design but it takes longer computation time. The time critical applications uses carry look-ahead adder (CLA) to derive fast result but it required a large area. In this work we compared hybrid adders on the basis of delay, power and area. This design has been synthesized by Spartan 3 family with XC3S400 device.

Keywords- Adder, carry select adder, Ripple carry adder, Carry look-ahead adder, VHDL code.

I. INTRODUCTION

Adders are widely used in digital integrated circuits. High-speed adders are the necessary components in Microprocessors and Digital signal processors. For adding two binary numbers, there are several adder structures based on different design. There is much binary adder architecture to be implemented in such applications. The easiest type of adder to build is a ripple carry adder, which uses parallel connected one bit full adders to generate its output. The Ripple Carry Adder (RCA) gives the most compact design but it requires longer computation time. The time critical applications use Carry Look-ahead scheme (CLA) to derive fast results but lead to increase in area. In mobile electronics, reducing area and power consumption are key factors in increasing portability and battery life. Even in servers and desktop computers, power consumption is an important design constraint. Design of area and power-efficient high-speed data path logic systems are one of the most substantial areas of research in VLSI system design. In digital adders, the speed of addition is limited by the time required to propagate a carry through the adder. Adders are extensively used in processing units such as the ALU's (Arithmetic Logic Unit) or in DSP (Digital Signal Processing) applications. A two-operand adder is used not only when performing additions and subtractions, but also often employed when executing more complex operations like multiplication and division [3].

A. Adders-

A two-operand adder is used not only when performing additions and subtractions, but also often employed when executing more complex operations like multiplication and division. Consequently, a fast and area efficient two- operand adder is essential. RCA design occupies the small area but takes longer computing time. The CLA offers a way to eliminate the ripple effect. For every bit, sum and carry is independent of the previous bits. CLA is faster than RCA but consumes large area [3].

Adders are basically of two type's i.e.

I. Homogeneous adders [1].
II. Heterogeneous/Hybrid adders [1].

Homogeneous adder consists of design of ripple carry adder (RCA) and carry look-ahead adder (CLA) [1]:-

1. Ripple carry adder-

Figure 1 depicts an 8-bit RCA, which is formed by a cascade of full adder modules. The full adder is an arithmetic building block that adds an augend and addend bit (say, a0 and b0) along with any carry input (Carry_in) and produces two outputs, namely, sum (Sum) and carry overflow (Carry_out). Since there is a rippling of carry from one full adder stage to another, the propagation delay of the RCA varies linearly in proportion to the adder width. This is called a RCA, since the carry signal "ripple" from the least significant bit position to the most significant bit position. The carry of this adder traverses longest path called worst case delay path through N stages[1].

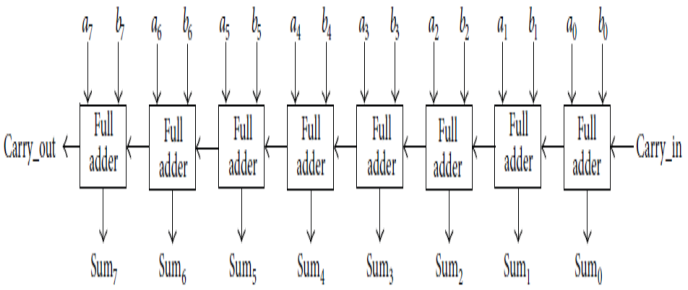

Fig. 1: 8-BIT RCA [1].

2. Carry look-ahead adder-

It is well known that a CLA is faster than a RCA. Although the concept of carry look-ahead is widely understood, the concept of section-carry based carry look-ahead may not be that well known, and hence to explain the distinction between the two, sample 4-bit look-ahead logic realized using these two approaches is portrayed in Fig. 2 for an illustration [1].

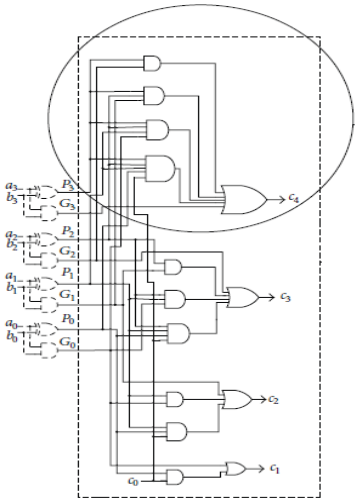

Fig. 2: CLA [1].

The section-carry based carry look-ahead generator shown enclosed within the circle in Figure 4 produces a single look-ahead carry signal corresponding to a "section" or "group" of the adder inputs (hence the term "section-carry"), while the conventional carry look-ahead generator encapsulated within the rectangle produces multiple look-ahead carry signals corresponding to each pair of augends and addend primary inputs. The section-carry based carry look-ahead generator differs from the traditional carry look-ahead generator in that bit-wise look-ahead carry signals are not required to be computed for the former [1].

The XOR and AND gates used for producing the necessary propagate and generate signals (P3 to P0 and G3 to G0) are highlighted using dotted lines in Fig. 4.We can calculate the generate bit, propagate bit, sum and carry in carry look-ahead generator form following equation [1].

$C_{4 =} G_i + P_i + C_i$
$G_i = a_i + b_i$
$P_i = a_i \oplus bi$
$SUM_i = P_i \oplus C_i$

Where, G is a generate bit and P is a propagate bit.
Heterogeneous/Hybrid adders consist of carry select adder along with the combination of ripple carry adder and carry look-ahead adder. The Heterogeneous/Hybrid adders are as follows:-

I. RCA_CSA
II. CLA_CSA
III. RCA_CLA_CSA

1. Ripple carry adder along with carry select adder (RCA_CSA)-

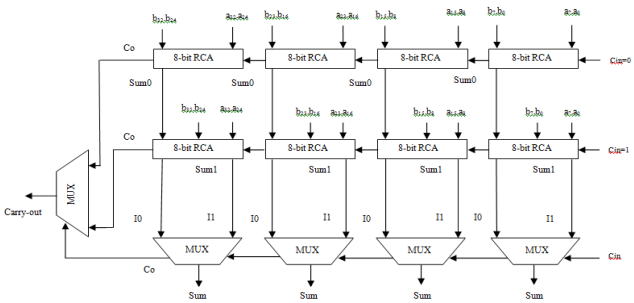

Fig. 3: BLOCK DIAGRAM OF RCA_CSA

The RCA and homogeneous CSA architectures are shown in Fig. 3 for an example case of 8-bit addition. Fig. 3 depicts an 8-bit RCA, which is formed by a cascade of full adder modules, the full adder is n arithmetic building blocks that add two input bits along with carry input and produce two output bit i.e. sum and carry out. Since there is a rippling of carry from one stage to another, the delay of RCA is varies linearly in proportion to the adder width. The CSA basically partitions the input data into groups and addition within the groups is carried out in parallel, that is, the CSA is composed of partitioned and duplicated RCA [1].

It can be seen from Fig. 3 that the least significant 8-bit adder stages of RCA and CSA are identical. However, the carry produced by least significant bit is simply propagated through the more significant bit in the case of RCA bit-by-bit, while the carry corresponding to the least significant bit serves as the selection input for MUXes present in the more significant position in the case of CSA [1].

2. Carry look-ahead adder along with carry select adder(CLA_CSA)-

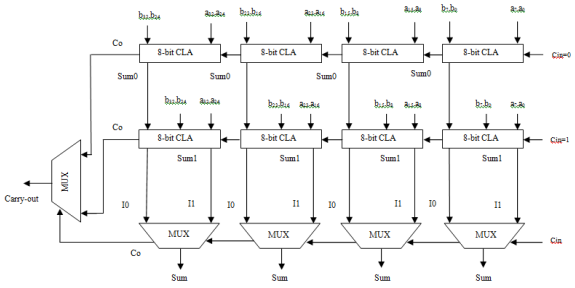

Fig. 4: BLOCK DIAGRAM OF CLA_CSA

It is well known that CLA is faster than a RCA, and hence it may be worthwhile to have a CLA as a replacement for the least significant RCA in the CSA structure. The CLA along with CSA in Fig. 4. The section-carry based carry look-ahead generator differs from a carry look-ahead generator in that bit-wise look-ahead carry signal are not required to be computed for the former [1].

3. Ripple carry adder and carry look-ahead adder along with carry select adder(RCA_CLA_CSA)-

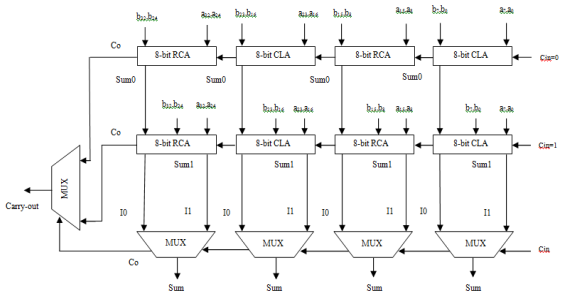

Fig. 5: BLOCK DIAGRAM OF RCA_CLA_CSA

Hybrid adder using RCA, CLA and CSA is design based on sections as shown in Fig. 5. Combination of RCA and CLA produces better results as compared to other hybrid adders.

II. SYNTHESIS

The complete Design is modeled in Pure VHDL. The syntax of the RTL design is checked using Xilinx tool. For functional verification, the design is modeled in Hardware descriptive

language (HDL).Test cases for the block level are generated in VHDL by both directed and random way.
The complete design along with all timing constraints, area utilization and optimization options are described using synthesis report. The adder design is synthesized at Spartan-3 (XC3S400).

III. RESULTS AND DISCUSSION

The comparison on the basis of critical path delay, area and power for the different hybrid adders shown in the table no. 1

Table: 1

Input Partition	Type of adder architecture (32-bit)	Delay (ns)	Area (BELs)	Power (mw)
Not Applicable	RCA	51.02	200	59.84
Not Applicable	CLA	50.94	197	59.84
8-8-8-8	CSA_RCA	23.904	257	59.84
8-8-8-8	CSA_CLA	23.918	257	59.84
8-8-8-8	CSA_RCA_CLA	22.858	258	59.84

IV. CONCLUSION

The paper describes the comparison between the hybrid adders. Adders are used in many data-processing systems to perform fast arithmetic operations. The carry select adder (CSA) is a square-root time high speed adder. It provides good compromise between RCA and CLA. The ripple carry adder (RCA) gives the most compact design but takes longer computation time. The time critical applications use carry look-ahead adder (CLA) to derive fast result but least to increase in area.
In this work we compared hybrid adders on the basis of delay, power and area. It clearly indicates that hybrid carry select adder using combination of RCA, CLA and CSA achieves fastest speed at approximately similar area and power dissipation.

REFERENCES

[1] V. Kokilavani, K. Preethi,and P. Balasubramanian, "FPGA-Based Synthesis of High-Speed Hybrid Carry Select Adders" Hindawi Publishing Corporation Advances in Electronics Volume 2015, Article ID 713843.

[2] Basant Kumar Mohanty, Senior Member, IEEE, and Sujit Kumar Patel, "Area– Delay–Power Efficient Carry-Select Adder" IEEE Transaction on circuits and systems— II: express briefs , vol .61, no.6, Jun 2014.

[3] Shivani Parmar, Kirat Pal Singh, "Design of high speed hybrid carry select adder," *IEEE Transactions on VLSI Systems*, 978-1-4673-4529-3/12 -2012.

[4] K. Preethi,and P. Balasubramanian, "FPGA Implementation of Synchronous section-carry base carry look-ahead adder" 2nd International conference on devices, circuits and systems (ICDCS), 2014 IEEE

[5] Shamim Akhter, Saurabh Chaturvedi, Kilari Pardhasardi, "CMOS implementation of efficient 16 bit square root carry select adder" 2nd International Conference on Signal Processing and Integrated Networks (SPIN), 2015 IEEE

[6] J.Monteiro, J. L. G¨untzel, and L. Agostini, "A1CSA: an energy efficient fast adder architecture for cell-based VLSI design," in Proceedings of the 18th IEEE International Conference on Electronics, Circuits and Systems (ICECS 11), pp. 442–445, Beirut, Lebanon, December 2011.