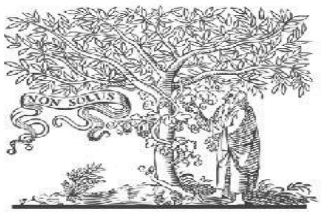
International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

COPY RIGHT



ELSEVIER
SSRN

IJIEMR Transactions, online available on 5th Oct 2021. Link
:http://www.ijiemr.org/downloads.php?vol=Volume-10&issue=ISSUE-10

DOI: 10.48047/IJIEMR/V10/I10/34

Title **DESIGN AND EVALUATION OF HIGH SPEED APPROXIMATE REVERSE CARRY PROPAGATE ADDER**

Volume 10, Issue 10, Pages: 202-206

Paper Authors
ALLAMSETTI TANUJA, N.RAMESH BABU

USE THIS BARCODE TO ACCESS YOUR ONLINE PAPER

To Secure Your Paper As Per UGC Guidelines We Are Providing A Electronic Bar Code

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

DESIGN AND EVALUATION OF HIGH SPEED APPROXIMATE REVERSE CARRY PROPAGATE ADDER

[1]ALLAMSETTI TANUJA, [2]N.RAMESH BABU

[1]M.Tech Scholar, Dept of ECE, St. Mary's Womens Engineering College, Budampadu Post, Guntur, Andhra Pradesh 522017

[2]Assistant Professor, Dept of ECE, St. Mary's Womens Engineering College, Budampadu Post, Guntur, Andhra Pradesh 522017

ABSTRACT: In any Digital Signal Processing (DSP) systems multiplier is the major key block. Adder plays a prominent role in Digital filtering, Digital communications and spectral analysis. This project presents the Design and Evaluation of High Speed Approximate Reverse Carry Propagate Adder. The inputs of the adder are obtained from the memory location and it will be given to the multiplier block. In the RCPA structure, the carry signal propagates in a counter-flow manner from the most significant bit to the least significant bit; hence, the carry input signal has higher significance than the output carry. The final product output can be obtained through the register.

KEYWORDS: VLSI, Multiplier, Reverse Carry Propagate Adder (RCPA).

I.INTRODUCTION

As we know that Addition is one of the operation done in digital signal processing and control systems. Different types of adders are proposed before. Depend upon the parameters adder is used. But if the speed is main constraint then we supposed to use this parallel prefix adders. In very large scale integrated circuits, parallel prefix adder is mostly used. The parallel prefix structures allow trade-offs to obtain required logic levels. Depending upon the adder performance the digital signal processor produces accurate results [1].

Basically, ripple carry adder is obtained from the general purpose processors and DSP processors. Repetitive additions are performed by using multiplier operand. But using propagation, the quality, Delay, and area is measured. In ripple carry adder first the carry propagation is overlapped and next addition operation is performed.

But the ripple carry adder produces large delay in the system. To overcome this delay problem parallel prefix adder is invented. This adder produces faster operations, reduces power consumption and delay. To add more number of numbers together powerful adders should be used [2].

Basically, parallel prefix adder produces high speed multi operands. The scaling down of device dimensions into the Nano-meter range is likely to result in significantly higher defect rates during the manufacturing process of IC's. With significantly increased defect rates, defect tolerance mechanisms are necessitated to guarantee a reasonable yield. Post manufacturing reconfiguration techniques to bypass defects are already applied in memory systems and FPGA's.

However, such low-cost defect tolerance techniques rely heavily on the relative independence of operations of the homogeneous components, such as LUT and memory cells. Logic systems, on the other hand, usually constitute heterogeneous components with strong dependencies among each other. This makes it hard to realize fine-grained, low-cost defect tolerance schemes for a high level of defect rate [3].

Among the various adders, PPA provides a general form to represent a wide range of

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

adder design choices. Reliable PPA designs have mostly been done on the particular form of ripple carry adder. Structure and hardware redundancy. For performance purposes, the hardware of a typical parallel prefix adder is divided into two disjoint groups of the even bits and the odd bits. This provides a natural way to make the defects isolated: errors caused by the defects in one group will not affect the results produced by the other group [4-5]. Furthermore, the in-built redundancy in a parallel prefix adder allows each group to be capable of generating the results for the other group, with a small hardware and time overhead.

II.RELATED WORK

In electronics adder performance addition operations. Adder exits in ALU, some Arithmetic logic units contain multiple adders. Adder can construct based on numerical expressions, such as Excess-3 or binary coded decimal (BCD), the most the adders operates on binary numbers. For single bit addition, two types of adders are there. Consider A and B are two inputs of half adder and outputs are sum S and Carry Co. S is the XOR operation of two inputs A and B, and Co is the AND operation of two inputs A and B. perhaps output of Half adder is sum of two bit numbers, and Co be the most significant number. Second type of adder is full adder; it contains three inputs A, B and Ci. A full adder is constructed based on two half-adders.

Multi-bit adders are several types in that ripple carry adder is simple, as well as slowest, since it propagate each full adder. Carry look ahead adder is work by creating and propagate from a few significant bit position. In some cases P is nothing but sum of output of a half adder and G is carry output for some adder. P andG create carries of bit position. Multi bit architecture breaks the adder into blocks. To reduce time blocks required the carry length of circuits, this block based on carry bypass adder. Here each bit is determined by p and g values from each block.

a) Half adder:

A and B are two single bit numbers , Half adder is used to add these two numbers and it produces sum ‘S’ and Carry out Co. Half adder uses only two single digit numbers. For lager adding circuits like Full adders it may be used as stating building block. Boolean expressions are

$S = A \oplus B = A^1B + AB^1$ (1)

$Co = AB$ (2)

b) Full adder:

Consider A, B and C are three inputs of Full adder and outputs are Sum ‘S’ and Carry ‘Co’. Full adder is constructed in series connection by using two half adder. The sum of A and Bare fed to second half- adder, which then adds it to the carry in C to generate final output S. the Carry out , Co, is the result of an OR operation obtained from the carry outputs of both half adders.

c) Parallel adder:

Parallel adders are constructed with basic digital circuits which compute the addition operation of binary equivalent in parallel manner.

III. PROPOSED SYSTEM

The sum and carry of each literal full adder are developed by the below equation

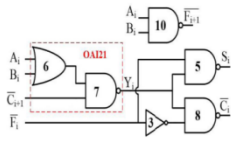
$$2C_{i+1} + S_i = A_i + B_i + C_i \text{ ------ (3)}$$

Where A_i and B_i are the inputs of the i^{th} bit corresponding to A and B, C_i, $C_{i+1}$are the input and output carry and S_i is the sum of the i^{th} bit. Depending on the above equation sum and carry of the i^{th} stage is related to the inputs A, B of the ith bit position and the carry out from the previous stage. Eq. (3) can be rearranged by moving C_i and C_{i+1} to their opposite sides.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

$$S_i - C_i = A_i + B_i - 2C_{i+1} \text{ --------- (4)}$$

Considering the above equation, the working principle of RCPA architecture depends on the input of the current stage and the carry output of the next stage. In this structure, the outputs sum and carry have the same significance. The input carry C_{i+1} to the current i^{th} stage is set up by the $(i+1)^{th}$ stage FA. The selection of exact output corresponding to the input is from the set $\{-2,-1, 0, 1, 2\}$. But while considering the significance of the output, the selection of output can be done only from $\{-1, 0, 1\}$ set, leads to improper output. Especially, when the RHS of (4) is -2 or 2 then the output becomes unreliable. And either (0, 0) or (1, 1) may be considered for S_i and C_i, when the RHS of Eq.(4) is zero. One solution to select one of the two answers is to use the auxiliary signal developed by the inputs of $(i-1)^{th}$ stage. Based on the above details the full adder structure for RCPFA is shown in Fig. 1. The RCPFA has 4 inputs and 3 outputs. The inputs to this full adder are the A_i, B_i, forecast signal F_i, and the carry out from the next stage C_{i+1}. The sum S_i, carry C_i and forecast signal $F_{i-1}$are termed to be its output signals.

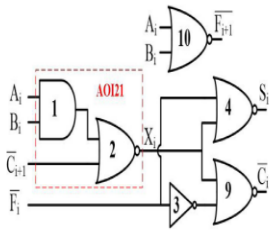


Item	A i	B i	C i+1	F i	S i	C i	F i+1
1	0	0	0	0	0	0	0
2	0	0	0	1	1	1	0
3	0	0	1	X	0	1	0
4	0	1	0	0	0	0	1
5	0	1	0	1	1	1	1
6	0	1	1	X	0	1	1
7	1	0	0	0	0	0	1
8	1	0	0	1	1	1	1
9	1	0	1	X	0	1	1
10	1	1	X	0	0	0	1
11	1	1	x	1	1	1	1

Fig. 1: THE INTERNAL STRUCTURE AND TRUTH TABLE OF RCPFA

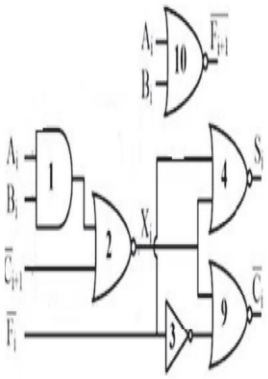
When the RHS of Eq. (4) is zero then the F_i signal is used to select one value from 2 pairs. The n-bit RCPA structure is shown in Fig. 2. In this n stage RCPA, the carry input C_n for MSB stage is accepted to be equal to the output F_n of that stage. Due to this some inexact result may be generated in the approximate adder. F_0 for the LSB stage is assumed to be equal to the C_0 of the n-bit RCPA, because there is no preceding stage for the zeroth stage. The analytical flow of operation is shown in Fig. 2.

Incomplete carry propagation causes some errors in RCPA as in RCA and in addition to intrinsic error. The main advantage of RCPA is the reduction in error as the bit significance increases. That is the error due to delay variation during carry propagation is lower for the most significant bit



Item	A i	B i	C i+1	F i	S i	C i	F i+1
1	0	0	X	0	0	0	0
2	0	0	X	1	1	1	0
3	0	1	0	X	1	0	0
4	0	1	1	0	0	0	0
5	0	1	1	1	1	1	0
6	1	0	0	X	1	0	0
7	1	0	1	0	0	0	0
8	1	0	1	1	1	1	0
9	1	1	0	X	1	0	1
10	1	1	1	0	0	0	1
11	1	1	1	1	1	1	1

Fig. 2: THE INTERNAL STRUCTURE AND TRUTH TABLE OF RCPFA-II



Item	A i	B i	C i+1	F i	S i	C i	F i+1	X i	Y i
1	0	0	0	0	0	0	0	0	1
2	0	0	0	1	1	1	0	0	1
3	0	0	1	X	0	1	0	1	1
4	0	1	0	X	1	0	0	0	0
5	0	1	1	X	0	1	0	1	1
6	1	0	0	X	1	0	1	0	0
7	1	0	1	X	0	1	1	1	1
8	1	1	0	X	1	0	1	0	0
9	1	1	1	0	0	0	1	0	1
10	1	1	1	1	1	1	1	0	1

Fig. 3: THE INTERNAL STRUCTURE AND TRUTH TABLE OF RCPFA-III

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

IV. RESULTS

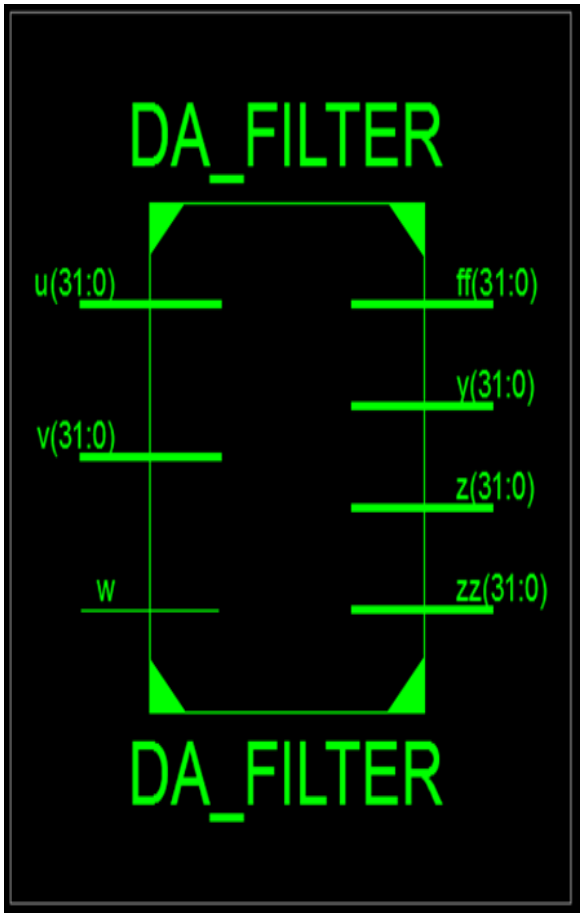


Fig. 4: RTL SCHEMATIC

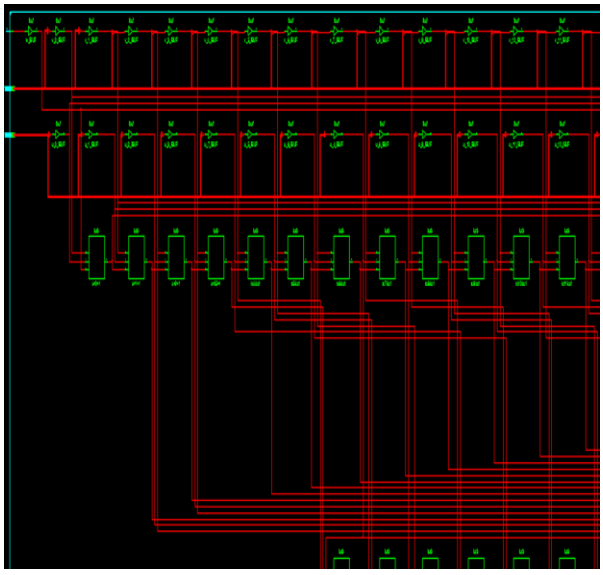


Fig. 5: TECHNOLOGY SCHEMATIC

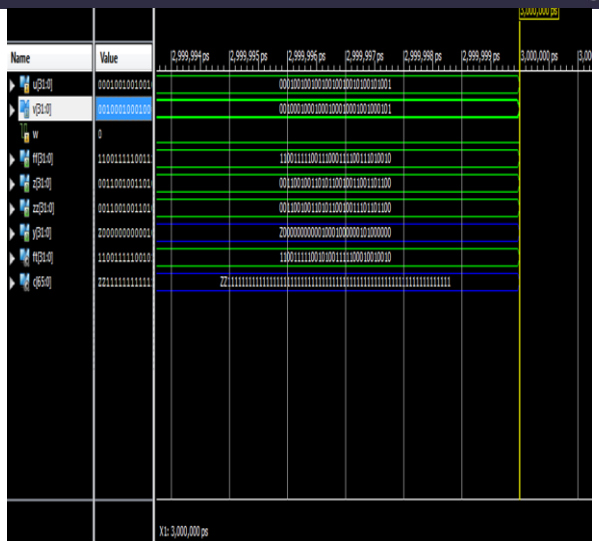


Fig. 6: OUTPUT WAVEFORM

V.CONCLUSION

Here, Design and Evaluation of High Speed Approximate Reverse Carry Propagate Adder was implemented. The reverse carry propagate performs addition operation in which the carry propagation in counter flow manner. The final result from the RCPA is stored in the shift register that provide final multiplication output. The reverse carry propagation provides higher stability for delay variations. The design implementation of the proposed approximate multiplier using approximate RCPA was observed from the simulated result analysis in Xilinx ISE design suite 14.7.

VI.REFERENCES

[1] J. Kung, D. Kim, and S. Mukhopadhyay, "On the impact of energyaccuracy tradeoff in a digital cellular neural network for image processing," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 34, no. 7, pp. 1070–1081, Jul. 2015.

2] T. Moreau, A. Sampson, and L. Ceze, "Approximate computing: Making mobile systems more efficient," IEEE Pervasive Comput., vol. 14, no. 2, pp. 9–13, Apr. 2015.

[3] A. Madanayake et al., "Low-power VLSI architectures for DCT/DWT: Precision vs

approximation for HD video, biomedical, and smart antenna applications," IEEE Circuits Syst. Mag., vol. 15, no. 1, pp. 25–47, 1st Quart., 2015.

[4] S. Ghosh, D. Mohapatra, G. Karakonstantis, and K. Roy, "Voltage scalable high-speed robust hybrid arithmetic units using adaptive clocking," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 18, no. 9, pp. 1301–1309, Sep. 2010.

[5] N. Zhu, W. L. Goh, W. Zhang, K. S. Yeo, and Z. H. Kong, "Design of low-power high-speed truncation-error-tolerant adder and its application in digital signal processing," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 18, no. 8, pp. 1225–1229, Aug. 2010.

[6] H. R. Mahdiani, A. Ahmadi, S. M. Fakhraie, and C. Lucas, "Bio-inspired imprecise computational blocks for efficient VLSI implementation of soft-computing applications," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 57, no. 4, pp. 850–862, Apr. 2010.

[7] V. Gupta, D. Mohapatra, A. Raghunathan, and K. Roy, "Low-power digital signal processing using approximate adders," IEEE Trans. Comput.- Aided Design Integr. Circuits Syst., vol. 32, no. 1, pp. 124–137, Jan. 2013.

[8] Z. Yang, A. Jain, J. Liang, J. Han, and F. Lombardi, "Approximate XOR/XNOR-based adders for inexact computing," in Proc. 13th IEEE Int. Conf. Nanotechnol. (NANO), Aug. 2013, pp. 690–693.

[9] Z. Yang, J. Han, and F. Lombardi, "Transmission gate-based approximate adders for inexact computing," in Proc. IEEE/ACM Int. Symp. Nanosc. Archit. (NANOARCH), Jul. 2015, pp. 145–150.

[10] H. A. F. Almurib, T. N. Kumar, and F. Lombardi, "Inexact designs for approximate low power addition by cell replacement," in Proc. Design, Autom. Test Eur. (DATE), Mar. 2016, pp. 660–665.