ENERGY EFFICIENT LOW DENSITY ADDER FOR ENHANCED DSP APPLICATIONS

1. G.HINDUJA, 2. S.KOTESWARI

1. M. Tech, Dept. of ECE, DNR College of Engineering and Technology, Bhimavaram, W.G, A.P.

2. Professor & H.O.D, Dept. of ECE, DNR College of Engineering and Technology, Bhimavaram, W.G, A.P.

ABSTRACT:

In this paper, a reverse carry propagate adder (RCPA) is presented. 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. This method of carry propagation leads to higher stability in the presence of delay variations. Three implementations of the reverse carry propagate full-adder (RCPFA) cell with different delay, power, energy, and accuracy levels are introduced. The proposed structure may be combined with an exact (forward) carry adder to form hybrid adders with tunable levels of accuracy. Further this project is enhanced by using square root modified carry select adder. This enhancement improves the area time optimizations in adder implementations.

KEYWORDS: Reverse Carry Propagate Adder, Inexact adder, Accuracy, Digital Signal processing, Carry select adder.

INTRODUCTION: The challenge of the verifying a large design is growing exponentially. There is a need to define new methods that makes functional verification easy. Several strategies in the recent years have been proposed to achieve good functional verification with less effort. Recent advancement towards this goal is methodologies. The methodology defines a skeleton over which one can add flesh and skin to their requirements to achieve functional verification. The report is organized as two major portions; first part is brief introduction and history of the functional verification of regular Carry skip adder which tells about different advantages of Carry skip adder and RCA architecture and in this Regular model, there is a drawback and in order to overcome that complexity, the modified architecture of CSKA has been designed. The electronics industry has achieved a phenomenal growth over the last two decades, mainly due to the rapid advances in integration technologies, large-scale systems design due to the advent of VLSI. The number of applications of integrated circuits in high-performance computing, telecommunications and consumer electronics has been rising steadily and at a very fast pace. Typically, the required computational power of these applications is the driving force for the fast development of this field.

LITERATURE SURVEY:

In 1962, O.J.Bedrij [1] described the extremely fast digital adder with sum selection and multiple-radix carry. He compared the amount of hardware and the logical delay for a 100-bit ripple-carry adder and a carry-select adder. The problem of carry-propagation delay was overcome by independently generating multiple-radix carries and using these carries to select between simultaneously generated sums. In this adder system, the addend and augend were divided into sub addend and sub augend sections that were added twice to produce two sub sums. One addition was done with a carry digit forced into each section, and the other addition combined the operands without the forced carry digit. The selection of the correct sub sum from each of the adder sections depended upon whether or not there actually was a carry into that adder section. T.Y.Chang and M.J.Hsiao [3], suggested that instead of using dual ripple carry adders, a carry select adder scheme using an add one circuit to replace one ripple carry adder requires 29.2% fewer transistors with a speed penalty of 5.9% for bit length n=64. If speed was important for this 64 bit adder, then two of carry-select adder blocks could be substituted by the proposed scheme with a 6.3% area saving and the same speed. The B.Ramkumar, H.M.Kittur, and P. M. Kannan in 2010 [3] suggested a very simple approach to improve the speed of addition. Based on this approach a 16, 32 and 64-bit adder architecture was developed and compared with conventional fast adder architectures. In many parallel multipliers to speed up the final addition, CLA was arranged in the form of Carry Select adder (CSLA) & was used. But due to the structure of the CSLA it occupied more chip area, because it uses multiple pairs of RCA‟s to generate the partial sum and carry by considering Cin=0 and Cin=1.Thus the complexity of the final adder structure was high. So they replaced the RCA (CLA) with Cin=1 with BEC logic, which reduced the maximum area but delay is increased in the final adder structure. Ramkumar and Harish 2011 [4] propose BEC technique which is a simple and efficient gate level modification to significantly reduce the area and power of square root CSLA. Veena nair in 2013 suggested a new approach in with D-latch is used with enabled signal instead of BEC [6]. Based on this approach a 16, 32 and 64-bit adder architecture was developed and compared with conventional fast adder architectures .The new structure as a result reduces the delay of the structure. Shanthala et al (2009) investigated various pipelined MAC architectures and circuit and the design techniques which are suitable for the implementation of high throughput signal processing algorithms. The goal of this project is to design and VLSI implementation of pipelined MAC for highspeed DSP applications at 180nm technology. For designing the pipelined MAC, various architectures of multipliers and one bit full adders are considered. The static and dynamic one bit full adder was implemented as the basic block. For checking the functionality of the whole system, spice code is written using the HSPICE by defining all the blocks in the circuit as the sub circuits. To improve the speed of VLSI signal processing systems, a new architecture for a high-speed Multiply-Accumulate (MAC) unit optimized for digital filters is proposed. This unit is designed as a coprocessor for the LEON2 RISC processor. In this work, four parallel MAC units with two dualport coefficient register files and a three-port general register file and a control unit are included in the co processing block. With the

existence of four parallel units, several SIMD format instructions have been added to LEON2 instruction set. Each MAC unit has two 16-bit inputs, 32-bit output register and a programmable Round-Saturate block The MAC unit uses a new architecture which embeds the accumulate module within the partial products summation tree of the multiplier with minimum overhead. The experimental results demonstrate a high performance in implementation of digital filters at elevated speeds of up to 33 millions of input samples per second.

REVERSE CARRY PROPAGATE ADDER:

The conventional FA which is the key building block of the carry propagate adders has three inputs with the same weight. Moreover, it has two outputs for a summation result with the same weight as that of the inputs and a carry output with twice the weight. The carry propagation delay (tCP) is the most important timing parameter in an FA due to the fact that it determines the delay of the critical path of multibit adders (and multipliers). In the worst case, the delay of the carry propagation adder is $n \times t$CP where n is the bit width of the adder. Hence, a clock period smaller than $n \times t$CP can result in a setup time violation and hence a potential error. A small short-delay violation may lead to a large amount of error owing to the fact that the error occurs on the MSBs of the summation. This is the result of the generation and propagation of the carry input of the MSBs through small significant bit FAs. Based on this reasoning, if the order of the carry propagation is reversed, one may expect that the amount of error due to the timing violation decreases. This has inspired us with conceiving approximate FAs in which the carry propagation takes place in the reverse order (counter-flow direction).

EXISTING TECHNIQUE:

REVERSE CARRY PROPAGATE FULL-ADDER CELL

Each exact FA generates its carry output and sum signals using

$$2Ci+1 + Si = Ai + Bi + Ci \quad (1)$$

where Ai (Bi) is the i th bit of the input A (B), Ci (Ci+1) is the carry input (output), and Si is the i th bit of the sum. Based on this equation, the output signals in the i th bit position depends on the i th bits of the inputs A and B and the carry output of

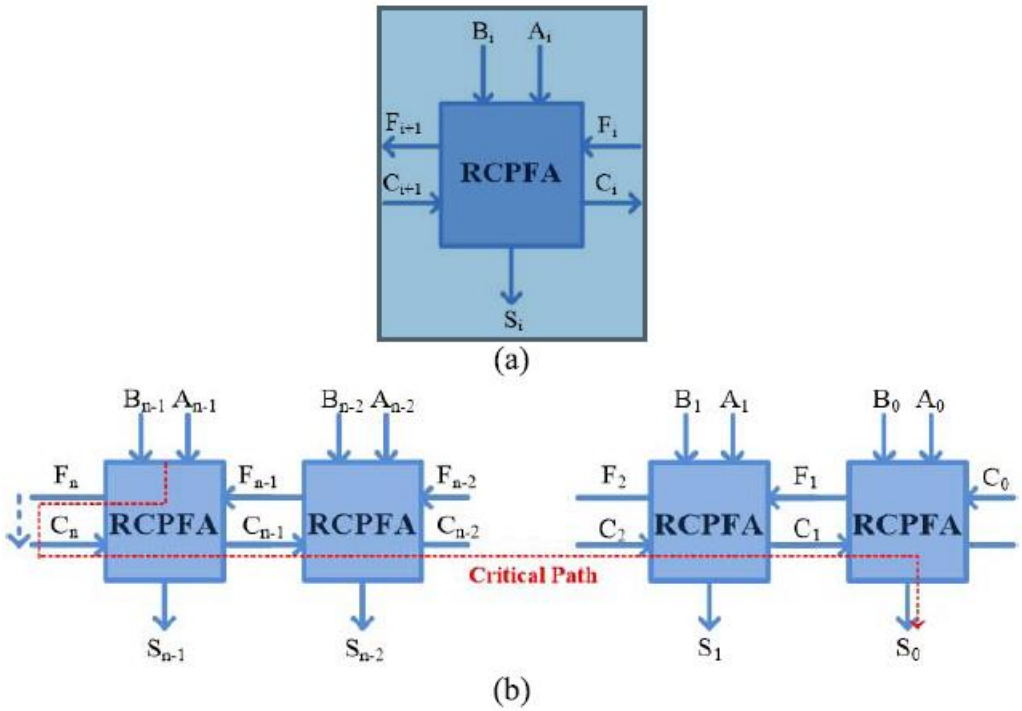


Fig. (a) Block diagram of the RCPFA. (b) *n*-bit RCPA. the previous position (*Ci)*. By moving the term *Ci (Ci*+1*)* to the left (right) side of the equation, one may write *Si* − *Ci* = *Ai* + *Bi* − 2*Ci*+1. (2) Considering (2), one may think of a full adder as a structure which operation depends on the carry output of the (*i*+1)st bit position (*Ci*+1*)* and its input operand bits. For this structure, the outputs are the sum and the carry signals with the same weights. Notice that the carry input of the *i* th bit position (*Ci*+1*)*, should be generated by the FA in the (*i* + 1)st bit position. Based on the input bits, the exact output range for *Si* − *Ci* is from the set {-2, −1, 0, 1, 2}. On the other hand, based on the weights of the output signals, the output range can be only from the set {−1, 0, 1}, which makes the output inexact. More specifically, the output becomes imprecise when the right side of (2) becomes −2 or 2. In addition, when the right side of (2) becomes 0, either of (0,0) and (1,1) may be considered for (*Si* ,*Ci)*. One of the ways to select between these two solutions is to use an auxiliary signal created byusing the inputs of the (*i* − 1)st bit position. Based on the above discussion, we suggest a family of fulladders for the RCPFA shown in Fig. As shown in Fig, these full adders have four inputs and three outputs. The inputs are the input operands (*Ai* and *Bi)*, the carry output of the next bit position (*Ci*+1*)*, and a forecast signal (*Fi)*. The RCPFA determines the summation result (*Si)*, carry (*Ci)*, and the forecast signal (*Fi*+1*)* as its output signals. Signal *Fi* is employed to select one of the two pairs when the right-hand side of (2) is zero. Fig. 2(b) indicates an *n*-bit RCPA. In this structure, the most significant carry input (*Cn)* is assumed to be equal to output *F* of the most significant RCPFA. This may introduce some inaccuracies in the suggested approximate adder. Also, since there is no previous stage for generating *F* for the 0th stage, the carry input of the *n*-bit adder (*C0)* is used as *F* of the LSB full-adder. The critical path for this adder is also shown in Fig. 2(b). In addition to the intrinsic error of the RCPA, similar to the conventional RCA, an incomplete carry

propagation causes some error. As mentioned before, the advantage of the RCPA is that the value of the error is in the direction of decrease

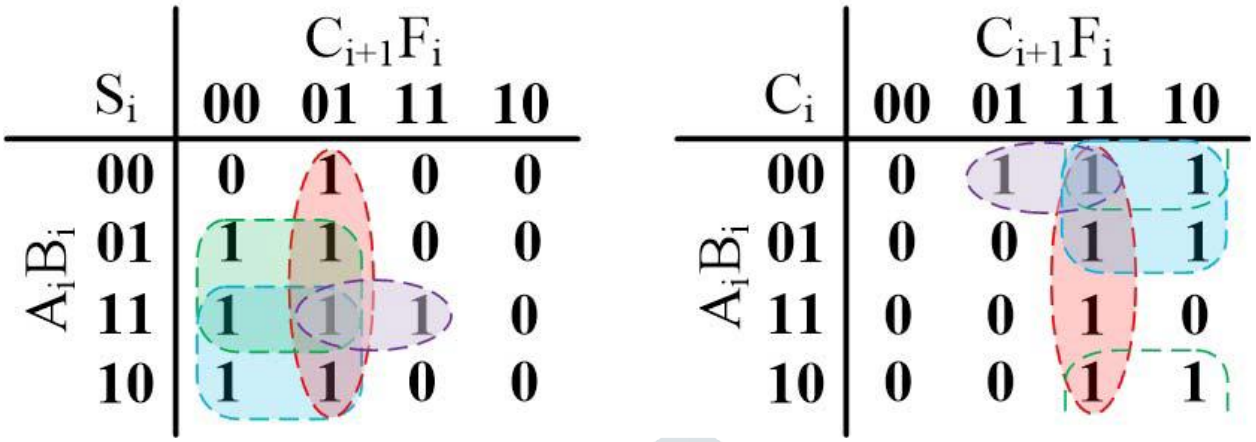


Fig. Karnaugh maps for signals *Si* and *Ci* of the general form of RCPFA.

in the bit significance. This means that the cumulative impact of the error (e.g., due to the delay variation) during the carry propagation is lower for bits with higher significances.

B. Internal Structure of RCPFA

To determine a structure for RCPFA, the Karnaugh maps of the summation result (*Si*) and carry (*Ci*) were drawn based on (2) and considering the forecast signal as an input (Fig. 3). The Boolean relations between inputs for generating *Si* and *Ci* are obtained as

$Si = Ci{+}1Fi + Ci{+}1\ Ai + Ci{+}1Bi + Ai\ Bi\ Fi$ (3)

$Ci = Ci{+}1Fi + Ci{+}1\ Ai + Ci{+}1Bi + Ai\ Bi\ Fi$. (4)

An optimized gate-level structure for implementing RCPFA may be achieved by simplifying (3) and (4) as

$Si = Fi\ (Ci{+}1 + Ai\ Bi\ ) + Ci{+}1(Ai + Bi\ ) = Fi\ Xi + Yi$ (5)

$Ci = Fi\ (Ci{+}1(Ai + Bi\ )) + (Ci{+}1 + Ai\ Bi\ ) = FiYi + Xi$. (6)

In this adder structure, the accuracy and performance of RCPFA depend on the signal *F* whose generation leads to some overheads. This means that optimizing the generation of the forecast signal may simplify (optimize) the general form of the RCPFA structure. In this paper, three different generation mechanisms for signal *F* are presented. The truth table and the optimized gate-level structures of these RCPFAs are provided in Fig. 4. In the first RCPFA type (RCPFA-I), which is the general form obtained from (5) and (6), one of the input

operands, e.g., *Ai* is considered as the output *F*. In the second type (RCPFA-II), the signal *F* is the carry generate signal (*Ai* **AND** *Bi*), while in the third type (RCPFA-III), the signal *F* is the carry alive signal (*Ai* **OR** *Bi*). By choosing the carry alive signal as signal *F*, some states of the truth table that *Xi* = 1 does not happen. Hence, by replacing *Xi* by zero, the general structure can be simplified.

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	×	0	1	0	1	1
4	0	1	0	×	1	0	0	0	0
5	0	1	1	×	0	1	0	1	1
6	1	0	0	×	1	0	1	0	0
7	1	0	1	×	0	1	1	1	1
8	1	1	0	×	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

(a)

Item	A_i	B_i	C_{i+1}	F_i	S_i	C_i	F_{i+1}
1	0	0	×	0	0	0	0
2	0	0	×	1	1	1	0
3	0	1	0	×	1	0	0
4	0	1	1	0	0	0	0
5	0	1	1	1	1	1	0
6	1	0	0	×	1	0	0
7	1	0	1	0	0	0	0
8	1	0	1	1	1	1	0
9	1	1	0	×	1	0	1
10	1	1	1	0	0	0	1
11	1	1	1	1	1	1	1

(b)

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	×	0	1	0
4	0	1	0	0	0	0	1
5	0	1	0	1	1	1	1
6	0	1	1	×	0	1	1
7	1	0	0	0	0	0	1
8	1	0	0	1	1	1	1
9	1	0	1	×	0	1	1
10	1	1	×	0	0	0	1
11	1	1	×	1	1	1	1

(c)

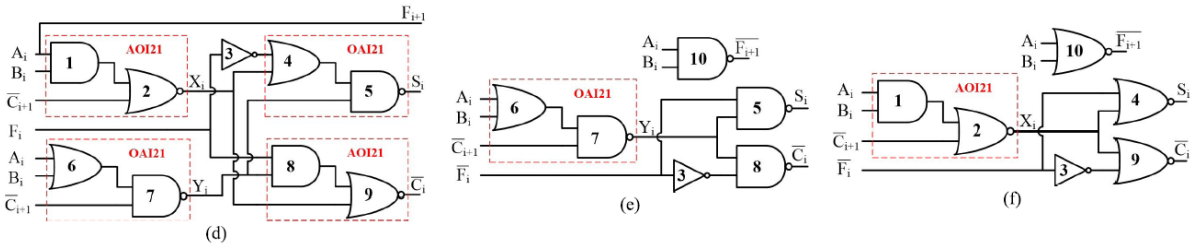


Fig. Truth tables of (a) RCPFA-I, (b) RCPFA-II, and (c) RCPFA-III, and the internal structures of the (d) RCPFA-I, (e) RCPFA-II, and (f) RCPFA-III. employed in [9] and [10]. This leads to fewer transistor counts. In this paper, to achieve highest reliability and speed, we use standard CMOS gates for implementing RCPFAs. Therefore, the combinational gates like AND–OR–Invert (AOI21) and OR–AND–Invert (OAI21) are utilized, which consist of six transistors. The conventional mirror FA has two (eight) more transistors compared to that (those) of RCPFA-I (RCPFA-II and RCPFA-II).

16-BIT CARRY SELECT ADDER:

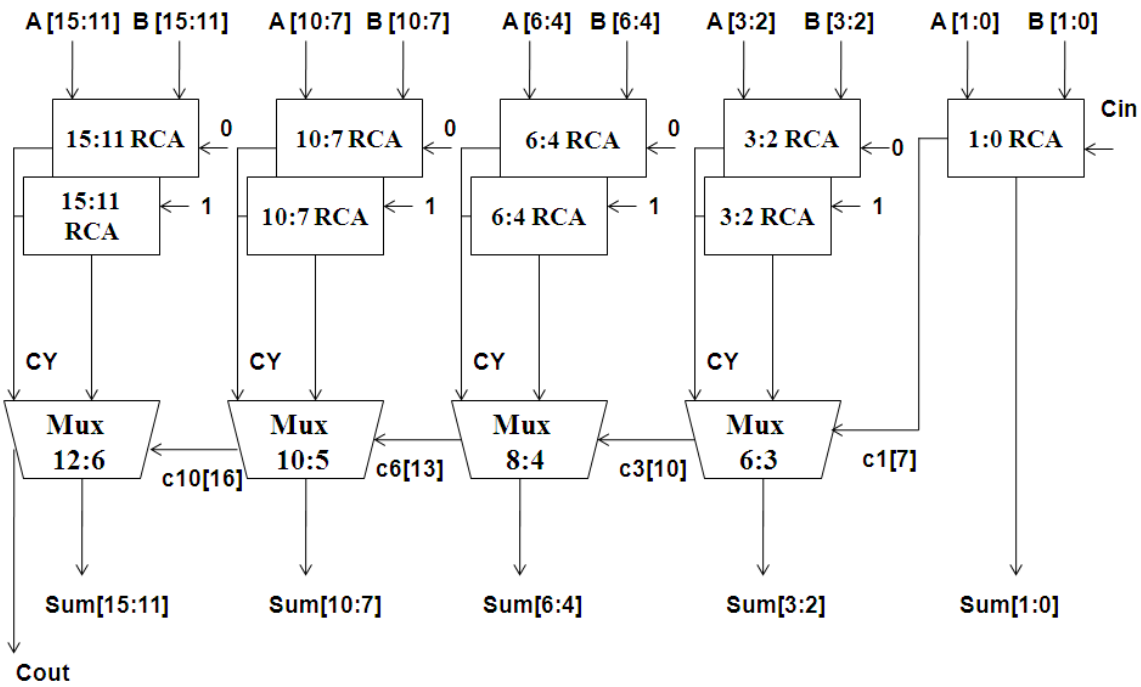


Fig: Existing system (Regular 16-bit Carry select adder)

The carryselect adder generally consists of two ripple carry adders and a multiplexer. Adding two n-bit numbers with a carry-select adder is done with two adders (therefore two ripple carry adders) in order to perform the calculation twice, one time with the assumption of the carry being zero and the other assuming one. After the two results are calculated, the correct sum, as well as the correct carry, is then selected with the multiplexer once the correct carry is known.

BEC

BEC stands for Binary to Excess-1 Converter.

Design of CSLA with BEC:

As stated above, the main idea of this work is to use BEC instead ofthe RCA with Cin=1 in order to reduce the area and power consumptionof the regular CSLA. To replace the n-bit RCA, an (n+1)-bit BECis required.

The structure and the function table of a 4-b BEC are shownin the figure 3.7 and Table 3.3 respectively.

Structure of BEC:

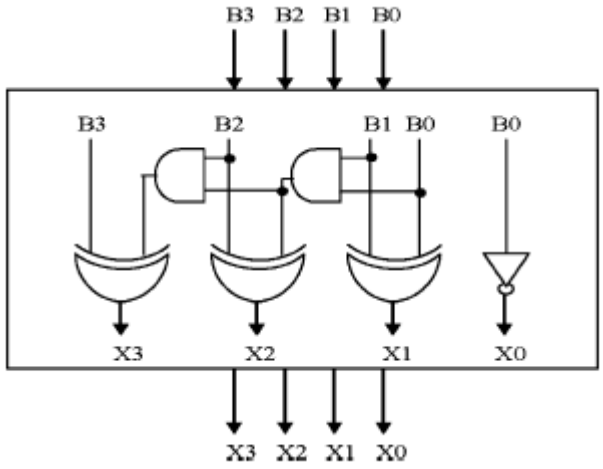


Fig : Structure of 4-bit BEC

B [3:0] (Binary inputs)	X[3:0] (Excess-1 outputs)
0000	0001
0001	0010
0010	0011
0011	0100
0100	0101
0101	0110
0110	0111
0111	1000
1000	1001
1001	1010
1010	1011
1011	1100
1100	1101
1101	1110
1110	1111
1111	0000

Table: Function table of BEC

The Boolean expressions of the 4-bit BEC are listed as below:

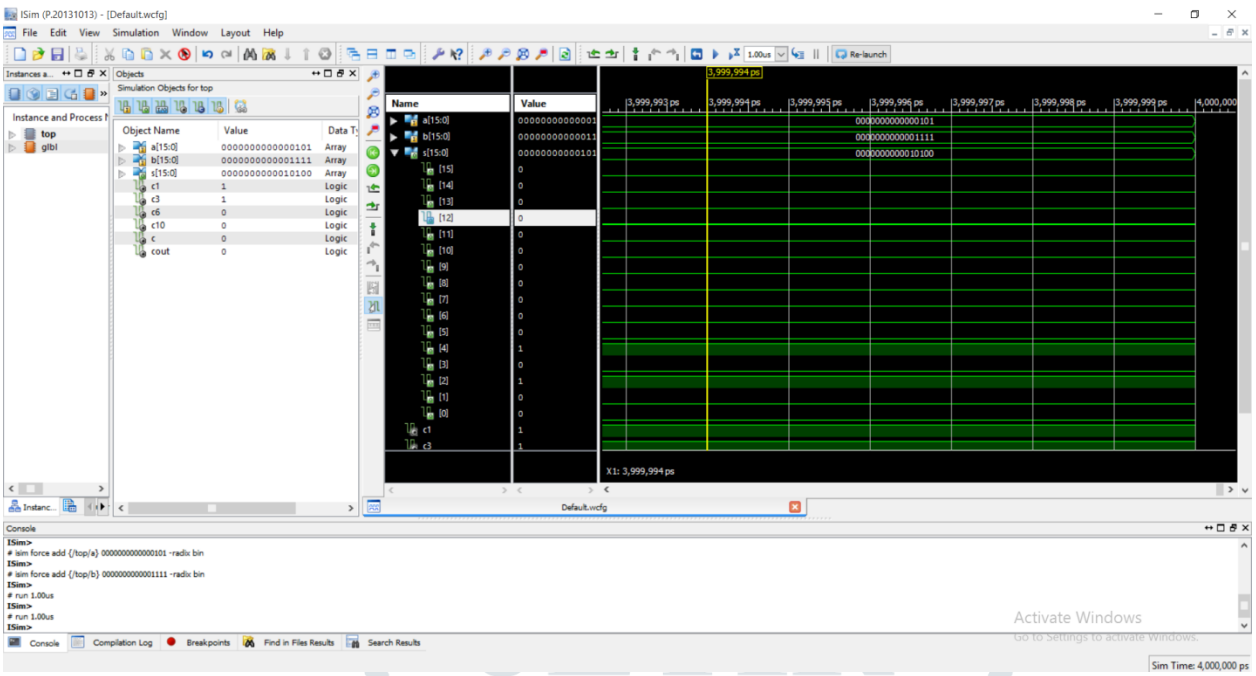
X0 = ~B0

X1 = B0 ^ B1

X2 = B2 ^ (B0 & B1)

X3 = B3 ^ (B0 & B1 & B2)

RESULT:



COCNLUSION:

In this paper, we proposed approximate RCPFAs which propagate carry from most significant to LSBs. The reverse carry propagation provided higher stability in delay variation. The efficacy of the proposed approximate FAs and the hybrid adders which realized them has been studied in XILINX ISE 14.7. The results indicated that utilizing the proposed RCPFAs in the hybrid adders provides most efficient optimizations. Enhanced results shows efficient implementation of modified carry select adder for many applications.

REFERENCES

[1] W. Dally, J. Balfour, D. Black-Shaffer, J. Chen, R. Harting, V. Parikh, J. Park, and D. Sheffield, "Efficient embedded computing," *Computer*, vol. 41, no. 7, pp. 27–32, Jul. 2008.

[2] P. Kulkarni, P. Gupta, and M. Ercegovac, "Trading accuracy for power with an underdesigned multiplier architecture," in *Proc. 24th IEEE Int. Conf. VLSI Design*, Jan. 2011, pp. 346–351.

[3] R. Hegde and N. Shanbhag, "Energy-efficient signal processing via algorithmic noise-tolerance," in *Proc. IEEE/ACM Int. Symp. Low Power Electron. Design*, Aug. 1999, pp. 30–35.

[4] R. Hegde and N. R. Shanbhag, "Soft digital signal processing," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 9, no. 6, pp. 813–823, Jun. 2001.

[5] B. Shim, S. Sridhara, and N. Shanbhag, "Reliable low-power digital signal processing via reduced precision redundancy," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 12, no. 5, pp. 497–510, May 2004.

[6] G. Varatkar and N. Shanbhag, "Energy-efficient motion estimation using error-tolerance," in *Proc. IEEE/ACM Int. Symp. Low Power Electron. Design*, Oct. 2006, pp. 113–118.

[7] D. Mohapatra, G. Karakonstantis, and K. Roy, "Significance driven computation: A voltage-scalable, variation-aware, quality-tuning motion estimator," in *Proc. IEEE/ACM Int. Symp. Low Power Electron. Design*, Aug. 2009, pp. 195–200.

[8] N. Banerjee, G. Karakonstantis, and K. Roy, "Process variation tolerant low power DCT architecture," in *Proc. Design, Automat. Test Eur.*, 2007, pp. 1–6.

[9] G. Karakonstantis, D. Mohapatra, and K. Roy, "System level DSP synthesis using voltage overscaling, unequal error protection and adaptive quality tuning," in *Proc. IEEE Workshop Signal Processing Systems*, Oct. 2009, pp. 133–138.

[10] L. N. Chakrapani, K. K. Muntimadugu, L. Avinash, J. George, and K. V. Palem, "Highly energy and performance efficient embedded computing through approximately correct arithmetic: A mathematical foundation and preliminary experimental validation," in *Proc. CASES*, 2008, pp. 187–196.

[11] A. K. Verma, P. Brisk, and P. Ienne, "Variable latency speculative addition: A new paradigm for arithmetic circuit design," in *Proc. Design, Automat. Test Eur.*, 2008, pp. 1250–1255.

[12] N. Zhu, W. L. Goh, and K. S. Yeo, "An enhanced low-power highspeed adder for error-tolerant application," in *Proc. IEEE Int. Symp. Integr. Circuits*, Dec. 2009, pp. 69–72.

[13] P. Kulkarni, P. Gupta, and M. D. Ercegovac, "Trading accuracy for power in a multiplier architecture," *J. Low Power Electron.*, vol. 7, no. 4, pp. 490–501, 2011.

[14] D. Shin and S. K. Gupta, "Approximate logic synthesis for error tolerant applications," in *Proc. Design, Automat. Test Eur.*, 2010, pp. 957–960.

[15] B. J. Phillips, D. R. Kelly, and B. W. Ng, "Estimating adders for a low density parity check decoder," *Proc. SPIE*, vol. 6313, p. 631302, Aug. 2006.

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

[17] D. Shin and S. K. Gupta, "A re-design technique for datapath modules in error tolerant applications," in *Proc. 17th Asian Test Symp.*, 2008, pp. 431–437.

[18] D. Kelly and B. Phillips, "Arithmetic data value speculation," in *Proc. Asia-Pacific Comput. Syst. Architect. Conf.*, 2005, pp. 353–366.

[19] S.-L. Lu, "Speeding up processing with approximation circuits," *Computer*, vol. 37, no. 3, pp. 67–73, Mar. 2004.

BIBILOGRAPHY:

G.HINDUJA received the bachelor's degree and currently perceiving her M. Tech from DNR College of Engineering and Technology, Bhimavaram, West Godavari, A.P. Got many prizes in various college level competitions and talent exams. Her current research interests include VLSI architectures for cryptography arithmetic algorithms and true random number generator along with image processing methodologies.

S.KOTESWARI, Professor, received the Master's degree and currently working as Head of Department in DNR College of Engineering and Technology, Bhimavaram, West Godavari, A.P. Her current research interests include wireless communications, digital Image processing and signal processing. Very much enthused about learning innovative algorithms and research works.