NOVEL HIGH SPEED WITH LOW POWER CONFIGURABLE ADDER FOR APPROXIMATE COMPUTATIONS

M Srikanth Reddy (M.tech)[1]
V Srinivas (M.tech (Assistant Professor))[2]
Malla Reddy Engineering College, Maisammaguda, Dhulapally, Ranga Reddy, Telangana, INDIA[1, 2]
srikanthreddy.2646@gmail.com[1] sreenivas43@gmail.com[2]

ABSTRACT: Surmised registering is a productive methodology for mistake tolerant applications since it can exchange off precision for control. Expansion is a key essential capacity for these applications. In this paper, we proposed a low-control yet high speed precision configurable viper that additionally keeps up a little plan region. The proposed viper depends on the ordinary convey look-ahead viper, and its configurability of precision is acknowledged by veiling the convey proliferation at runtime. Looked at with the ordinary convey look-ahead snake, with just 14.5% zone overhead, the proposed 16-bit viper diminished power utilization by 42.7%, and basic way delay by 56.9% most as indicated by the exactness setup settings, individually. Besides, contrasted and other recently examined adders, the exploratory outcomes exhibit that the proposed viper accomplished the first motivation behind upgrading both power and speed at the same time without decreasing the exactness.

Keywords: NULL Convention Logic (NCL), Index Terms—Asynchronous circuits, low-power electronics, power gating.

I. INTRODUCTION

Applications that have recently emerged (such as image recognition and synthesis, digital signal processing, which is computationally demanding, and wearable devices, which require battery power) have created challenges relative to power consumption. Addition is a fundamental arithmetic function for these applications [1] [2]. Most of these applications have an inherent tolerance for insignificant inaccuracies. By exploiting the inherent tolerance feature, approximate computing can be adopted for a tradeoff between accuracy and power. At present, this tradeoff plays a significant role in such application domains [3]. As computation quality requirements of an application may vary significantly at runtime, it is preferable to design quality Configurable systems that are able to tradeoff computation quality and computational effort according to application requirements [4] [5]. The previous proposals for configurability suffer the cost of the increase in power [5] or in delay [12]. In order to benefit such application, a low-power and high-speed adder for configurable approximation is strongly required. In this paper, we propose a configurable approximate adder, which consumes lesser power than [5] does with a comparable delay and area. In addition, the delay observed with the proposed adder is much smaller than that of [12] with a comparable power consumption. Our primary contribution is that, to achieve accuracy configurability the proposed adder achieved the optimization of power and delay simultaneously and with no bias toward either. We implemented the proposed adder, the conventional carry look-ahead adder (CLA), and the ripple carry adder (RCA) in Verilog HDL using a 45-nm library. Then we evaluated the power consumption, critical path delay, and design area for each of these implementations. Compared with the conventional CLA, with 1.95% mean relative error distance (MRED), the proposed adder reduced power consumption and critical path delay by 42.7% and 56.9%, respectively. We provided a crosswise comparison to demonstrate the superiority of the proposed adder. Moreover, we implemented two previously studied configurable adders to evaluate power consumption, critical path delay, design area, and accuracy. We also evaluated the quality of these accuracyconfigurable adders in a real image processing application.

II.RELATED WORK

Gupta et al. [6] discussed how to simplify the complexity of a conventional mirror adder cell at the transistor level. Mahdiani et al. [7] proposed a lower-part-OR adder, which utilizes OR gates for addition of the lower bits and precise adders for addition of the upper bits. Venkatesan et al. [8] proposed to construct an equivalent untimed circuit that represents the behavior of an approximate circuit. The above static approximate designs [6-8] with fixed accuracy may fail to meet the quality requirements of applications or result in wastage of power when high quality is not required. Kahng et al. [4] proposed an accuracy-configurable adder (ACA), which is based on a pipeline structure. The correction scheme of the ACA proceeds from stage 1 to stage 4, if the most significant bits of the results are required to be correct, all the four stages should be performed. Motivated by the above, Ye et al. [5] proposed an accuracy gracefully-degrading adder (GDA) which allows the accurate and approximate sums of its sub adders to be selected at any time. Similar to [5], our adder proposed in this paper does not consider a pipeline structure either. To generate outputs with different levels of computation accuracy and to obtain the configurability of accuracy, some multiplexers and additional logic blocks are required in [5]. However, the additional logic blocks require more area. Furthermore, these blocks will cause power wastage when their outputs are not used to generate a sum. This problem was addressed by [12] based on a low-power configurable adder that generates an approximate sum by using OR gates. Similar to [12], the proposed adder also uses OR gates to generate an approximate sum, but [12] focuses on only power consumption and its delay is large. Thus, it may fail to meet the speed requirement of an application.

III.PROPOSED ACCURACY-CONFIGURABLE ADDER

Typically, a CLA consists of three parts: (1) half adders for carry generation (G) and propagation (P) signals preparation, (2) carry look-ahead units for carry generation, and (3) XOR gates for sum generation. We focus on the half adders for G and P signals preparation in part 1. Consider an n-bit CLA; each part of it can be obtained as follows:

$$P_i = A_i \oplus B_i, \ \ G_i = A_i \cdot B_i, \tag{1}$$

$$C_i = G_i + P_i \cdot C_{i-1}, \tag{2}$$

$$S_i = P_i \oplus C_{i-1}. \tag{3}$$

Where i is denoted the bit position from the least significant bit. Note that owing to reuse of the circuit of Ai XOR Bi for Si generation, here Pi is defined as Ai XOR Bi instead of Ai OR Bi. Because C0 is equal to G0 , if G0 is 0, C0 will be 0. From (2), we find that C1 is equal to G1 when C0 is 0. In other words, if G0 and G1 are equal to 0, C0 and C1 will be 0. By expanding the above to i , Ci will be 0 when G0, G1, … , Gi are all 0. This means that the carry propagation from C0 to Ci is masked. From (3), we can obtain that Si is equal to Pi when Ci-1 is 0.

From the perspective of approximate computing, if G is controllable and can be controlled to be 0, the carry propagation will be masked and S (=P) can be considered as an approximate sum. In other words, we can obtain the selectivity of S between the accurate and approximate sum if we can control G to be A AND B or 0. Evidently, we can achieve selectivity by adding a select signal. Figure 1(a) is a conventional half adder and Fig. 1(b) is a half adder to which the select signal has been added. Compared with the conventional half adder, we add a signal named “M_X” as the select signal and use a 3-input AND gate to replace the 2-input one. When M_X = 1, the function of G is the same as that of a conventional half adder; when M_X = 0, G is equal to 0.

Consider the condition when the inputs Ai and Bi are both 1, when M_Xi = 1, the accurate sum Si and carry Ci will be 0 and 1 ({Ci , Si} = {1,0}); when M_X0 , M_X1 , … , M_Xi are all 0, Si is equal to Pi (= Ai XOR Bi = 0) as an approximate sum and Ci is equal to 0 ({Ci , Si} = {0, 0}) as discussed above. Here {,} denotes concatenation. This means that the difference between the accurate and approximate sum is 2. Toward better accuracy results for the approximate sum, we use an OR function instead of an XOR function for P generation when M_X = 0. Thus, the difference will be reduced to 1. A 2-input XOR gate can be implemented by using a 2-input NAND gate, a 2-input OR gate, and a 2-input AND gate. An equivalent circuit of the conventional half adder is shown in Fig. 2. This is called a carry mask able half adder (CMHA). The dashed frame represents the equivalent circuit of a 2-input XOR (M_X = 1). We can obtain the following: P is equal to A XOR B, and G is equal to A AND B when M_X = 1; when M_X = 0, P is equal to A OR B and G is 0. Thus, M_X can be considered as a carry mask signal.

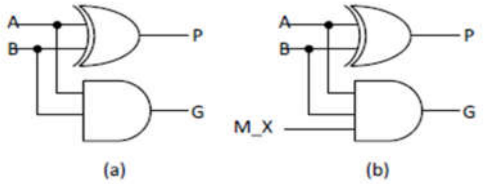


Fig. 1. (a) An accurate half adder, and (b) a half adder with a select signal.

Consider an n-bit CLA, whose half adders for G and P signals preparation are replaced by CMHAs. In this case, an nbit carry mask signal for each CMHA is required. To simplify the structure for masking carry propagation, we group four CMHAs and use a 1-bit mask signal to mask the carry propagation of the CMHAs in each group. The structure of a group with four CMHAs is shown in Fig. 3 as an example. A3-0, B3-0, P3-0, and G3-0 are 4-bit-length signals and represent {A3, A2, A1, A0}, {B3, B2, B1, B0}, {P3, P2, P1, P0}, and {G3, G2, G1, G0,}, respectively. M_X0 is a 1-bit signal and is connected to the four CMHAs to mask the carry propagation simultaneously. When M_X0 = 1, P3-0 = A3-0 XOR B3-0, and G3-0 = A3-0 AND B3- 0; when M_X0 = 0, P3-0 = A3-0 OR B3-0, and G3-0 = 0. We proposed an accuracy-configurable adder by using CMHAs to mask the carry propagation.

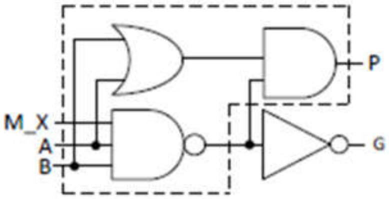


Fig. 2. A carry-mask able half adder.

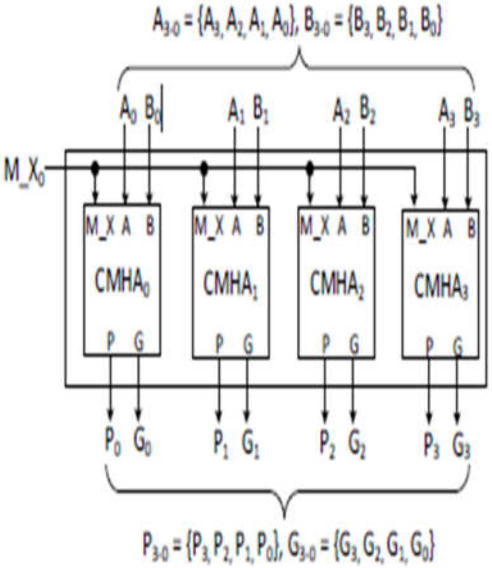

A3-0 = {A3, A2, A1, A0}, B3-0 = {B3, B2, B1, B0}
A0 B0
A1 B1
A2 B2
A3 B3
M_X0
M_X A B
CMHA0
CMHA1
CMHA2
CMHA3
P G
P0 G0
P1 G1
P2 G2
P3 G3
P3-0 = {P3, P2, P1, P0}, G3-0 = {G3, G2, G1, G0}

Fig. 3. Structure of a group with four CMHAs.

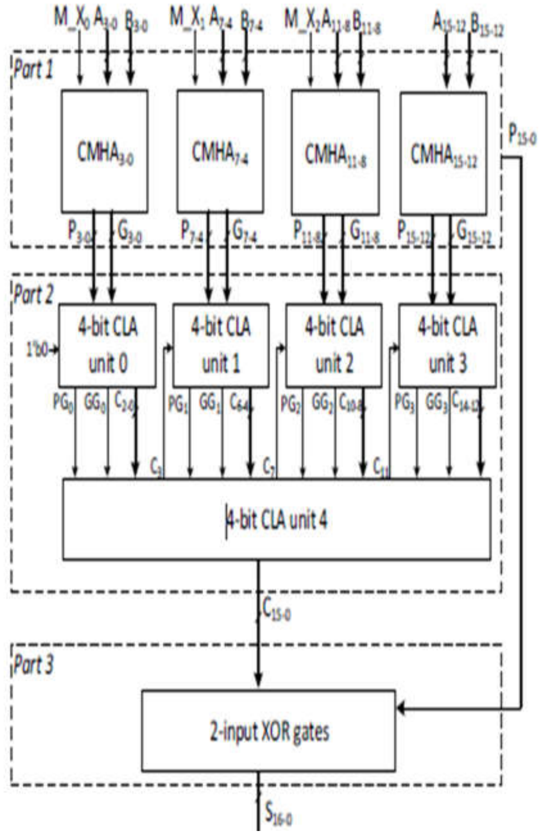


Fig. 4. Structure of the proposed 16-bit adder.

The structure of the proposed 16-bit adder is shown in Fig. 4 as an example. Four groups (CMHA3-0, CMHA7-4, CMHA11- 8, and CMHA15-12) are used to prepare the P and G signals. Each group comprises four CMHAs There is no mask signal for CMHA15-12 in this example; therefore, accurate P15-12 (= A15-12 XOR B15-12) and G15-12 (= A15-12 AND B15-12) are always obtained. P15-0 and G15-0 are the outputs from Part 1 and are connected to Part 2. Note that P15-0 is also connected to Part 3 for sum generation. In Part 2, four 4-bit carry look-ahead units (unit 0, 1, 2, 3) generate four PGs (PG0, PG1, PG2, and PG3), four GGs (GG0, GG1, GG2, and GG3), and 12 carries (C2-0, C6- 4, C10-8, and C14-12) first, and then the carry look-ahead unit 4 generates the remaining four carries (C3, C7, C11, and C15) by using the PGs and GGs. C15-0 is the output of Part 2 and is connected to Part 3. The fifteen 2-input XOR gates in Part 3 generate the sum.

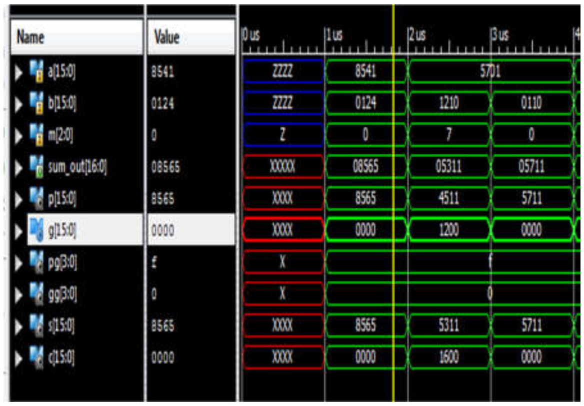
IV. RESULTS

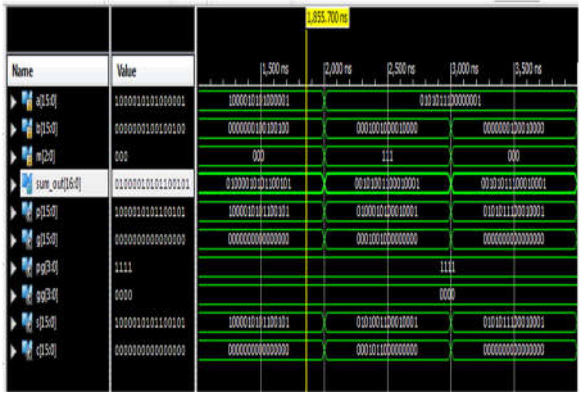
SYNTHESIS RESULTS:

The developed project is simulated and verified their functionality. Once the functional verification is done, the RTL model is taken to the synthesis process using the Xilinx ISE tool. In synthesis process, the RTL model will be converted to the gate level net list mapped to a specific technology library. Here in this Spartan 3E family, many different devices were available in the Xilinx ISE tool. In order to synthesis this design the device named as "XC3S500E" has been chosen and the package as "FG320" with the device speed such as "-5".

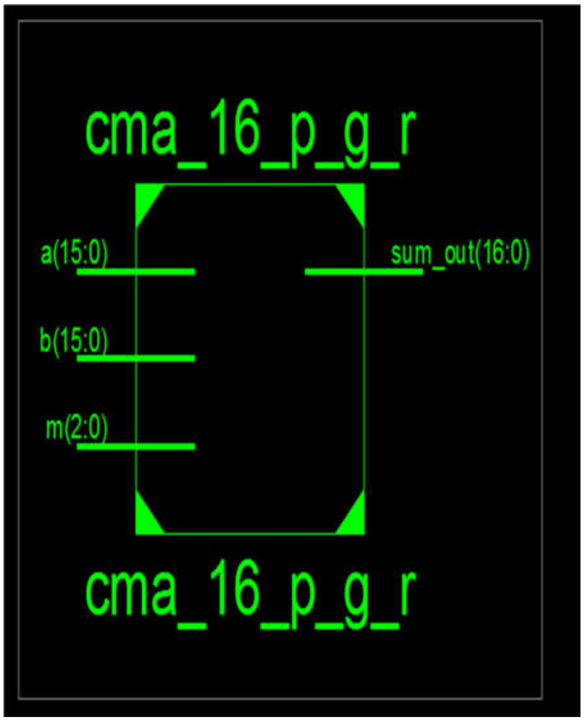
This design is synthesized and its results were analyzed as follows:

SIMULATION RESULTS OF 16 BIT CMHA

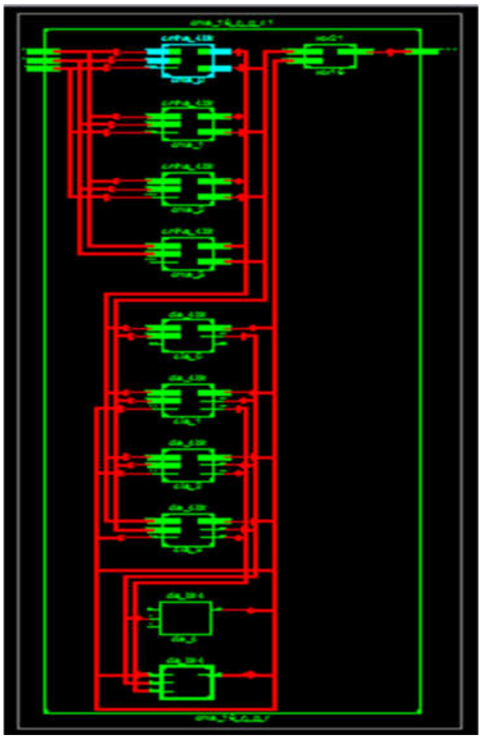

Name
Value
0 us
1 us
2 us
3 us
a[15:0] 8541 ZZZZ 8541 5701
b[15:0] 0124 ZZZZ 0124 1210 0110
m[2:0] 0 Z 0 7 0
sum_out[16:0] 08565 XXXXX 08565 05311 05711
p[15:0] 8565 XXXX 8565 4511 5711
g[15:0] 0000 XXXX 0000 1200 0000
pg[3:0] f X f
gg[3:0] 0 X 0
s[15:0] 8565 XXXX 8565 5311 5711
c[15:0] 0000 XXXX 0000 1600 0000



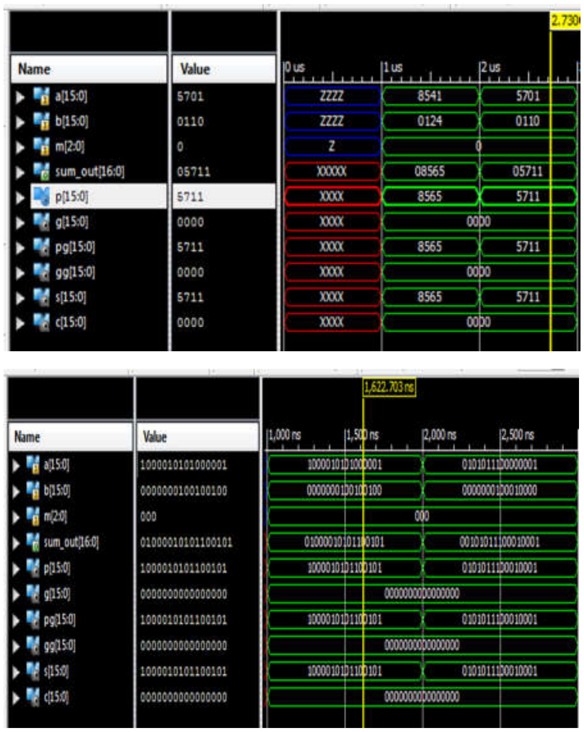
RTL BLOCK DIAGRAM OF 16 BIT CMHA



RTL SCHEMATIC DIAGRAM OF 16 BIT CMHA



TECHNOLOGICAL SCHEMATIC OF 16 BIT CMHA



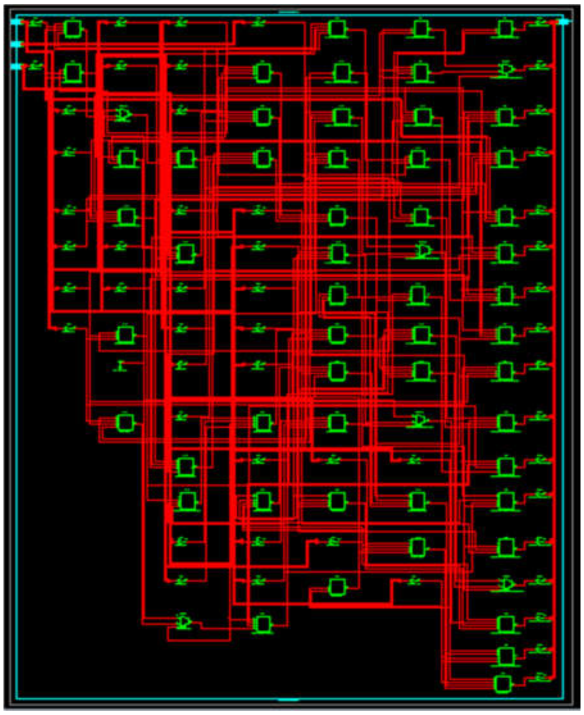
2.730

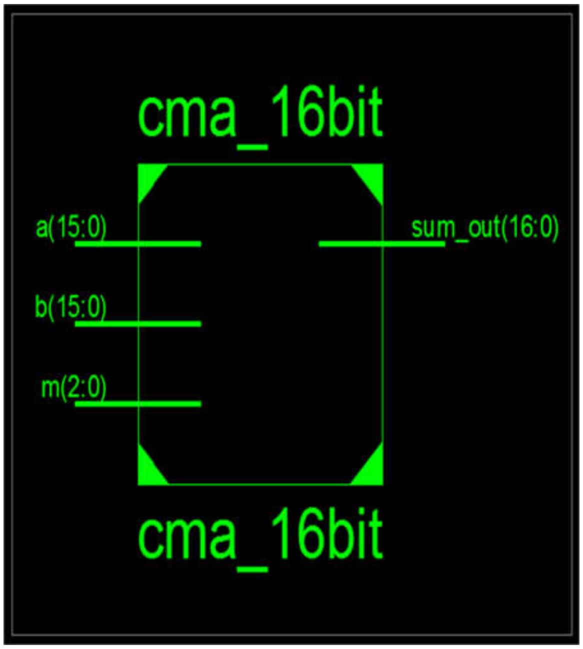
Name	Value	0 us	1 us	2 us
a[15:0]	5701	ZZZZ	8541	5701
b[15:0]	0110	ZZZZ	0124	0110
m[2:0]	0	Z	0	
sum_out[16:0]	05711	XXXXX	08565	05711
p[15:0]	5711	XXXX	8565	5711
g[15:0]	0000	XXXX	0000	
pg[15:0]	5711	XXXX	8565	5711
gg[15:0]	0000	XXXX	0000	
s[15:0]	5711	XXXX	8565	5711
c[15:0]	0000	XXXX	0000	

1,622.703 ns

Name	Value	1,000 ns – 1,500 ns	2,000 ns – 2,500 ns
a[15:0]	1000010101000001	1000010101000001	0101011100000001
b[15:0]	0000000100100100	0000000100100100	0000000100010000
m[2:0]	000	000	
sum_out[16:0]	01000010101100101	01000010101100101	00101011100010001
p[15:0]	1000010101100101	1000010101100101	0101011100010001
g[15:0]	0000000000000000	0000000000000000	
pg[15:0]	1000010101100101	1000010101100101	0101011100010001
gg[15:0]	0000000000000000	0000000000000000	
s[15:0]	1000010101100101	1000010101100101	0101011100010001
c[15:0]	0000000000000000	0000000000000000	

RTL BLOCK DIAGRAM OF 16 BIT NOVAL CMHA

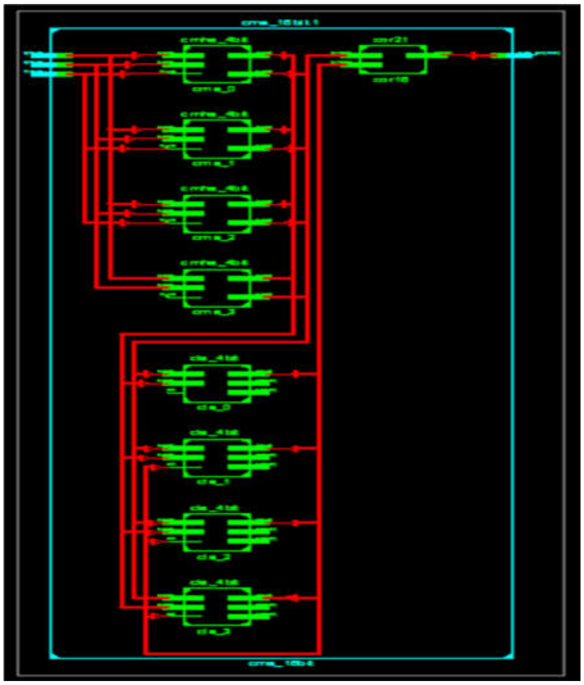




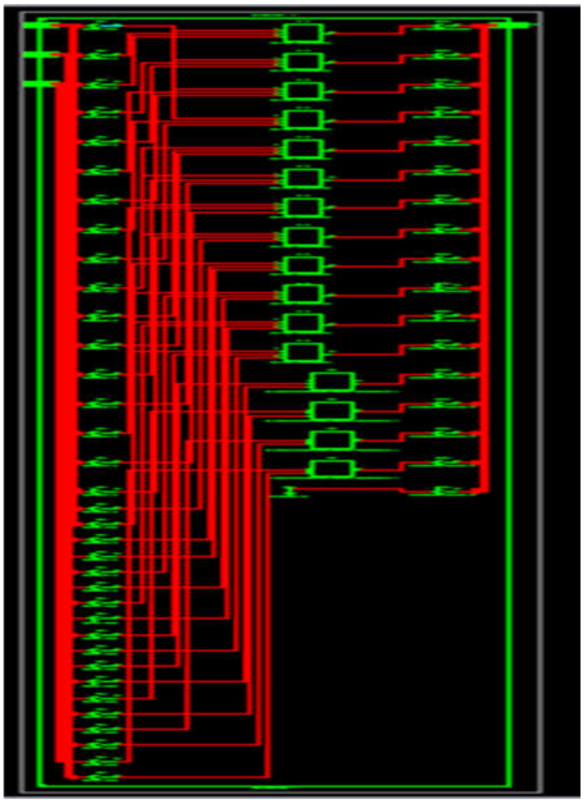
RTL SCHEMATIC DIAGRAM OF 16 BIT NOVAL CMHA

EXTENSION RESULTS

SIMULATION RESULTS OF 16 BIT NOVAL CMHA



TECHNOLOGICAL SCHEMATIC OF 16 BIT NOVAL CMHA



V. CONCLUSION

In this paper, an exactness configurable adder without enduring the expense of the expansion in power or in postponement for configurability was proposed. The proposed adder depends on the ordinary CLA, and its configurability of precision is acknowledged by covering the carry proliferation at runtime. The trial results show that the proposed adder conveys huge power reserve funds and speedup with a little zone overhead than those of the traditional CLA. Moreover, contrasted and recently contemplated configurable adders, the exploratory outcomes exhibit that the proposed adder accomplishes the first reason for conveying a fair upgraded result among power and postponement without relinquishing exactness. It was likewise discovered that the quality prerequisites of the assessed application were not traded off.

VI. REFERENCES

[1] S. Cotofana, C. Lageweg, and S. Vassiliadis, “Addition related arithmetic operations via controlled transport of charge”, IEEE Transactions on Computers, vol. 54, no. 3, pp. 243-256, Mar. 2005.

[2] V. Beiu, S. Aunet, J. Nyathi, R. R. Rydberg, and W. Ibrahim, “Serial Addition: Locally Connected Architectures”, IEEE Transactions on Circuits and Systems-I: Regular papers, vol. 54, no. 11, pp. 2564-2579, Nov. 2007.

[3] S. Venkataramani, V. K. Chippa, S. T. Chakradhar, K. Roy, and A. Raghunathan, “Quality programmable vector processors for approximate computing”, 46th Annual IEEE/ACM International Symposium on Micro architecture (MICRO), pp. 1-12, Dec. 2013.

[4] A. B. Kahng, and S. Kang, “Accuracy-configurable adder for approximate arithmetic

designs", IEEE/ACM Design Automation Conference (DAC), pp. 820-825, Jun. 2010.

[5] R. Ye, T. Wang, F. Yuan, R. Kumar, and Q. Xu, "On Reconfiguration- Oriented Approximate Adder Design and Its Application", IEEE/ACM International Conference on Computer-Aided Design (ICCAD), pp. 48- 54, Nov. 2013.

6] V. Gupta, D. Mohapatra, A. Raghunathan, and K. Roy, "Low-Power digital signal processing using approximate adders", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 32, no. 1, pp. 124-137, Jan. 2013.

7] 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 Transactions on Circuits and Systems I: Regular papers, vol. 57, no. 4, pp. 850-862, Apr. 2010.

[8] R. Venkatesan, A. Agarwal, K. Roy, and A. Raghunathan, "MACACO: modeling and analysis of circuits for approximate computing", IEEE/ACM International Conference on Computer-Aided Design (ICCAD), pp. 667-673, Nov. 2011.

[9] NanGate, Inc. NanGate FreePDK45 Open Cell Library, http://www.nangate.com/?page_id=2325, 2008

10] J. Liang, J. Han, and F. Lombardi, "New metrics for the reliability of approximate and probabilistic adders", IEEE Transactions on Computers, vol. 62, no. 9, pp. 1760-1771, Sep. 2013.

11] M. S. Lau, K. V. Ling, and Y. C. Chu, "Energy-Aware probabilistic multiplier: Design and Analysis", International Conference on Compliers, architecture, and synthesis for embedded systems, pp. 281-290, Oct. 2009.

[12] T. Yang, T. Ukezono, and T. Sato, "A Low-Power Configurable Adder for Approximate Applications", 19th International Symposium on Quality Electronic Design, Mar. 2018. 978-