International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 04 | Apr 2020 www.irjet.net p-ISSN: 2395-0072

A SPECULATIVE APPROXIMATE ADDER FOR ERROR RECOVERY UNIT

S.Sireesha[1], Dr.T. Lalith Kumar[2], T.Vijaya Nirmala[3]

[1]PG student, Dept. of ECE, Annamacharya Institute of Technology and Sciences, Kadapa, Andhra Pradesh, India
[2]professor, Dept. of ECE, Annamacharya Institute of Technology and Sciences, Kadapa, Andhra Pradesh, India
[3]Asst.professor, Dept. of ECE, Annamacharya Institute of Technology and Sciences, Kadapa, Andhra Pradesh, India

---

Abstract - *In this paper, a low delay consumption block-based carry speculative approximate adder is expected. Its design is based on partitioning the adder into some non-overlapped summation blocks whose designs may be selected from both the carry propagate and parallel-prefix adders. Here, the carry output of each block is speculated based on the input operands of the block itself and those of the next block. In this adder, the length of the carry chain is reduced to two blocks (worst case), where in most cases only one block is employed to calculate the carry output leading to a lower average delay. In addition, to reduce the output error rate an lower delay is also achieved through error detection and recovery mechanism.*

Key Words: Approximate computing, Low delay, Speculative Adder.

I. INTRODUCTION

In stream digital networks, one of the code restraint is the thermal style power (TSP) could maximize the concert of digital networks. One of the approaches will aid to gain the most out of this restraint in the usage of the accurate computing approaches. It may be usaged for application arena such as multimedia and image processing, digital signal processing, wireless communication, machine learning, and data mining which are naturally error-resilient. The approach may be used to realize more energy reduction and/or concert at the charge of some accuracy loss. In current years, different accurate computing approaches at various applications for software/hardware levels have been schedule. Illustration includes thread fusion and tunable kerenals, accurate accelerator, impresice logic/arthimetic unit and approximate instruction set architecture(ISA). In this task, we deal with approximate adders which are utilized as the staple operator in Concerting of the arthimetic operations such as substraction, multiplication and division. Approximate adders have been collected many regard by the designers.In the state-of-the-art approximate adders, where most of them are based on the carry propagate designs, the energy and speed gains have been attained by hardware manipulation, logic simplification, and voltage over scaling. While some of the adders were based on a configurable output accuracy others had a fixed accuracy level. The accuracy configurability enforced few overheads in terms of delay, area and power which could maximized their usage in some applications where such re-configurability is not required.In this paper, we propose a high performance yet low delay block-based carry speculative approximate adder structure which is called BCSA adder. In this design, the adder is partitioned into some non-overlapped parallel blocks, which in the worst-case, the carry output of a block is dependent on the carry output of the preciding block. To deduce the critical path more, we suggest an technique to foretell the carry output of a block based on its signals as well as of the succeeding block.The provision has a low hardware complication dominant a high delay (on average, about one block) and a rather high quality. To reach a lower error rate, an error detection and recovery mechanism is introduced, which propose the high output error rate . The effectiveness of this adder is compared with some of the state-of-the-art approximate adders. Finally, the efficiency of the adder studied using two image processing applications. The rest of the paper is unionized as follows.

II. RELATED WORKS

In this section, some of the earlier works in the area of the approximate adders are shortly reviewed.

A High Accuracy Block-based Approximate adder (HABA) was suggested in [9]. This adder proposed an error correction unit, which operated based on using the generate signal to decrease the accuracy loss. Although the power consumption and error rate were low, the vital path delay was even high due to its ripple carry propagation pattern.Reconfigurable approximate carry look-ahead adder (RAP- CLA), which is an approximate adder produced based on the exact carry look ahead adder, was anticipated[6].This adder was able to swap between the exact and approximate operating styles during the runtime. In RAP-CLA, fixed-size intersection sub-blocks (windows) were used to calculate the carry output and sum bits. In,[13]an accuracy surely degrading adder (GDA) was proposed which used $[n/l]$ l-bits blocks to calculate the output. GDA utilized multiplexers between sub-blocks to select between the correct and approximate input carries The proposed error correction unit is a sequential circuit, which depending on the number of sub-adders, takes various cycles for correcting the outputs. In, [13] an error liberal adder (ETA-I), where the add operation was divided into two independent parts such that the most important sum bits were calculated by exact FAs while the least compelling sum bits were generated by the XOR gates, was

suggested. For the least significant part, modified XOR gates were used. Although this adder reduced the power consumption, the error rate/distance was large.

III. INTERNAL DESIGN OF THE PROPOSED ADDER

The general design of an n-bit speculative approximate adder improved by a carry predictor unit is pictorialed in Fig1.The add operation is executed by [n/l]l-bit summation blocks working in parallel where l is bit length of each summation blocks. Each summation block admit an l-bit subadder, a carry predict unit , and a select unit .In this arrangement, the carry input of the i^{th} subadder, is chosen by$(i-1)^{th}$ select unit from the carry generator. In this erection, the carry input of the *i*th sub-adder, is chosen by $(i-1)^{th}$ Select unit from the carry signal generated by the $(i-1)^{st}$ Carry Predictor unit and the one generated by the $(i-1)^{th}$ sub-adder.Choosing the carry output of the Carry Predictor unit leads to a shorter critical path and lower energy consumption. In this case, the output between the blocks are cut at the cost of some accuracy loss. Thus, the accuracy of the add firm depends on the accuracy of the Carry Predictor unit, and also, the scheme of the carry output signal option. In our expected arrangement, the worst-case is the length of a carry chain which is equal to two blocks (*i.e.*,2).In most of the state-of-the-art approximate adders, the carry input of each block is selected only by the basics of the input signals of preciding blocks (see, *e.g.*, [10][11]). In this task, however, we introduced a speculative approximate adder that the carry input.

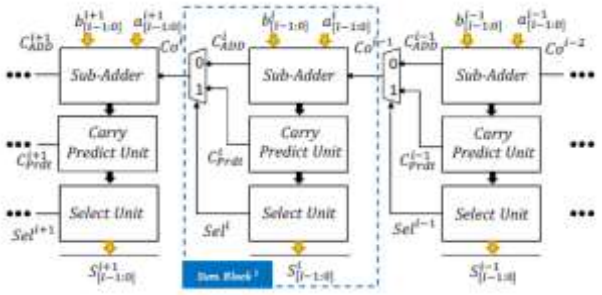

Fig 1: The general design of an approximate adder employed with carry prediction unit.

of its ith block is determined based on some input signals of the stream block and those of the succeeding one. As we will show, this approach results in a reasonable accuracy advance equivalenced to the other approximate adders. In BCSA(see Fig. 1), the carry output of the *i*th summation block ($\mathbf{CO^i}$) is obtained from

$$\mathbf{CO^i = sel^i.C^i_{ADD} + sel^i.C^i_{prdt}} \quad (7)$$

Where the $\mathbf{sel^i}$ ($\mathbf{C^i_{prdt}}$) is the output signal of the select (carry predictor)unit. Also , $\mathbf{C^i_{add}}$ is the carry output of the sub-adder of the i^{th} summation block analyzing that the carry input of the block is zero. Thus, in the worst case, the carry is propogated through two blocks(generated in the first bit place of the i^{th} block and propogated in the (i^{th} and $(i+1)^{th}$blocks).Now, in this task, we propose to determine the $\mathbf{sel^i}$,$\mathbf{C^i_{prdt}}$, and $\mathbf{C^i_{exact}}$ using

$$\mathbf{Sel^i = K^{i+1}{}_0 + G_{l-1}{}^i} \quad (8)$$
$$\mathbf{C^i_{prdt} = G^i_{l-1}} \quad (9)$$
$$\mathbf{C_{add} = P^i_{l-1}G_{l-2}{}^i + P_{l-1}{}^i P^i_{l-2} G_{l-3}{}^i + \ldots + \pi_{k=1} \cdot P^i_k . G_0{}^i} \quad (10)$$

Where the $\mathbf{K_0^{i+1}}$ is the kill signal of th first bit position of the $(i+1)^{th}$ block (i.e. $\mathbf{a^{l+1}{}_0 \,.\, b_0{}^{l+1}}$),$\mathbf{G^{l-1}}$ is the generate signal of the last bit position of the i^{th} block (i.e. $\mathbf{a_{l-1}{}^i \,.\, b^i_{l-1}}$), and $\mathbf{P^i_{l-1}}$ is the generate signal of the last bit position of the i^{th} block (i.e$\mathbf{.. a_{l-1}{}^i \wedge b^i_{l-1}}$). Based on the design of the excepted model is depicted in Fig. 2 where the carry propagate and parallel prefix adder designs(*e.g.* CLA, and RCA) could be covered for the sub-adders.Based on the carry output of the *i*th block is determined under four cases where, for each of them, either or is choosen. In the proposed approach, we focus on reducing the error as much as likely by selecting a precise carry input on the later block. Therefore, in each of these events, the carry output is selected with the highest possible accuracy. Thus, in the following paragraphs, we discuss the idea behind the carry output selection for the four cases of

- In the first case ($\mathbf{K_0^{i+1} = 0}$ **and** $\mathbf{G_{l-1}{}^i = 0}$), since $\mathbf{G_{l-1}{}^i}$ = 0, to reduce the probability of the error propogation in the proposed adder, in this case , the select unit circuit, chooses the $\mathbf{C^i_{add}}$ whose error probability is smaller than $\mathbf{C^i_{prdt}}$.

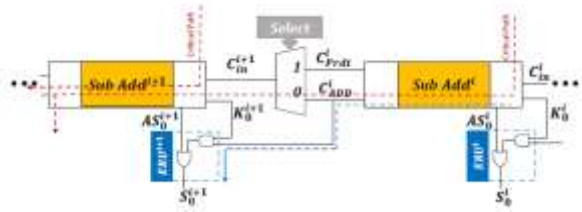

Fig.2: The structure of the proposed adder with Error Recovery Unit (ERU).

- In the second case($\mathbf{K_0^{i+1} = 0 and G_{l-1}{}^i = 1}$),because$\mathbf{G_{l-1}{}^i}$is1,the speculative carry($\mathbf{C^i_{prdt}}$) is correct. Therefore, for this case, the $\mathbf{C^i_{prdt}}$ is selected as the $\mathbf{C_{in}{}^{i+1}}$.
- In the third case ($\mathbf{K^{i+1}{}_0 = 1}$ **and** $\mathbf{G_{l-1}{}^i = 0}$), because $\mathbf{G_{l-1}{}^i}$ is 1, independent from accuracy of the carry input of the $(i+1)^{th}$ block, for shortening the critical path, we suggest to select $\mathbf{C^i_{prdt}}$ as the carry output of the i^{th} block.
- In the fourth case ($\mathbf{K_0^{i+1} = 1}$ **and** $\mathbf{G_{l-1}{}^i = 1}$),similar to the second case, since $\mathbf{G^i_{l-1}}$ is 1, the predicted output of the block. Therefore, in the proposed approach, $\mathbf{C_{prdt}{}^i}$ is selected as the $\mathbf{C^i_{in}}$.

Between these cases, only in the first case, the carry is propagated in two blocks. Therefore, on average, the length of the carry propagation is close to one block.

In the third case, although the carry input of the block is killed and is not propagated, the carry input is employed to determine the first summation bit of the block. Therefore, if the carry input in this case is wrong, it impacts on the output accuracy of the summation. Hence, for improving the accuracy of the proposed adder, we suggest an error recovery unit which generates the first summation bit of the i^{th} block (S_0^i) by

$$S_0^{i+1} = (K^{i+1}_0 . C_{add}^i) + (P_0^{i+1} \wedge C^{i+1}_{in}) \quad (11)$$

Note that $P_0^{i+1} \wedge C^{i+1}_{in}$ $(C_{in}^{i+1} = C^i_{prdt})$ is the approximate summation output in the first bit of the $(i+1)^{th}$ block denoted as AS_0^{i+1} in Fig. 2. Since the ERU is not on the critical path of adder, using the Error Recovery Unit (ERU) track to developing the accuracy with increasing the delay of the proposed adder structure. Fig. 3 shows the functionality of the proposed speculative approximate adder with and without the ERU. The error has been reduced by the error reduction unit. The ERU imposes only about 3% and 2% delay and area.

IV. RESULTS AND DISCUSSION

A. Error Metrics Evaluation

As early discourse, the Select and Carry Predictor units determine the accuracy of the approximate adders. In this section, the accuracy of the proposed adder is evaluated compared to the four latest state-of-the-art approximate adders. These adders include RAP-CLA, HABA and the BCSA. HABA equipped with the ERU unit proposed in (HABAERU). Note that the ERU circuit in HABAERU is different from the one we suggested for BSCAERU. For this study, three error metrics have been considered including Error Rate (ER), Normalized Mean Error Distance (NMED), and Mean Relative Error Distance (MRED). The NMED and MRED for an n-bit adder are obtained by

$$\mathbf{NMED} = \frac{1}{2^n} \sum_{i=1}^{|N|} \frac{|S_i - S_i'|}{2^{|N|}} \quad (12)$$

$$\mathbf{MRED} = \frac{1}{|N|} \sum_{i=1}^{|N|} \frac{|S_i - S_i'|}{S_i} \quad (13)$$

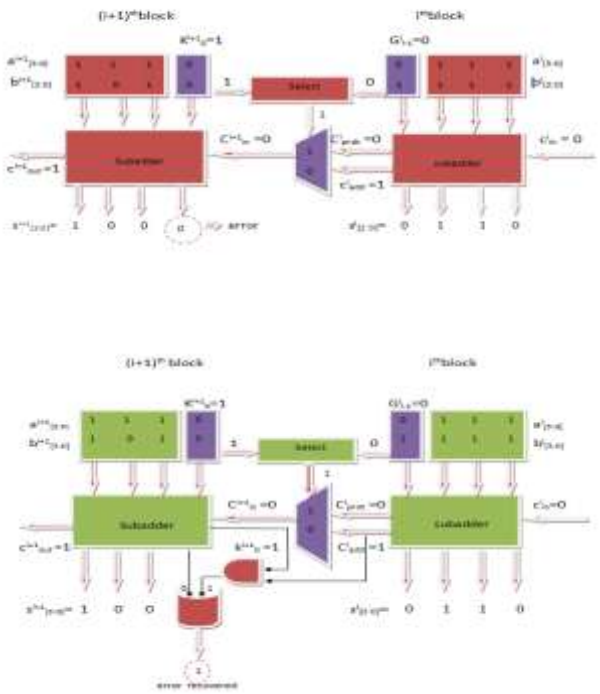

Fig 3: An example of functionality of the speculative approximate adder, (a) without ERU(BCSA W/O ERU), and (b) with ERU

where |N| shows the number of the input data samples, and S_i (S_i') is the true output of the add operation. The ER, NMED, and MRED of the considered adders under varius block sizes and input operand widths are recorded in TABLE I. While only the error prosody for the cases of block sizes of 2, 4 and 8 are reported in this table, we have performed this study for the block sizes from8 and 16-bit.

These metrics have been extracted by applying 65,536 (10 million) reliable odd numbers in the case of 8-bit adders. As the results show, in the case of 8-bit adder, the ER, NMED, and MRED of the BCSAERU are 0 meaning that BCSAERU is perfect. Among the deliberate adders, BCSAERU and HABA have the lowest ER compared to the other ones. On average, the ER of the BCSA is about 80% larger than HABA . On the other hand, the NMED and MRED of the BCSAERU (BCSA) are, on average, about 87% (52%) and 86% (40%) smaller than those of the other studies adders. In improver, for all the adders, by increasing the block size, the accuracy increases.

B. Design Parameters Evaluation

The hardware of BCSA, BCSAERU, HABA and RAP-CLA were described by Verilog HDL and synthesized by Synopsys Design Compiler. All the studies in this task have been performed using the typical process of the 15nm Fin FET Nan Gate technology with the operating voltage level of 0.8V and at the temperature of 25°C. The design

parameters of each adder under different block sizes have been reported in these games where the related block sizes are provided inside the circle. For extracting the power consumption (to obtain the energy consumption), up to 10M random stimuli were injected to the input of the netlist of the synthesized adders and the activity of the internal nodes of them were logged in the VCD format.

		8-Bit			16-Bit		
ADDER TYPE	BLOCK SIZE	ER(%)	NMED ($*10^{-4}$)	MRED ($*10^{-4}$)	ER(%)	NMED ($*10^{-4}$)	MRED ($*10^{-4}$)
HABA	2	12.42	607	621	33.77	608	623
	4	1.37	137	136	4.29	155	157
	8	0	0	0	0.10	10	10
RAP-CLA	2	15.54	606	646	36.45	626	670
	4	2.34	137	147	8.54	129	154
	8	0	0	0	034	10	11
BCSA	2	27.47	395	428	60.98	417	554
	4	5.46	34	52	21.15	56	81
	8	0	0	0	6.20	2	3
BCS-AERU	2	18.73	167	185	47.86	172	190
	4	0	0	0	5.89	20	26
	8	0	0	0	0	0	0

TABLE 1: comparison of accuracy results for 8-bits and 16-bits.

V. CONCLUSIONS

In this paper, we proposed a block-based carry speculative approximate adder (BCSA), which was based on dividing an exact adder into some non-overlapped blocks operated in parallel. Each block may be composed of any desired type of adders. In this adder, the length of carry chain was reduced to was utilized to estimate the carry. A select logic was recommended to guess the carry input of apiece block founded on some input operand bits of the current and next block. In addition, o reduce the error and delay with the declination of the accuracy loss, an error detection and recovery mechanism was suggested. Based on the results, for the different approximate performing styles, BCSAERU display on approximate adder.

REFERENCES

1. H. Jiang, C. Liu, L. Liu, F. Lombardi and J. Han, “A review, classification and comparative evaluation of approximate arithmetic circuits,” *ACM JETCAS*, vol. 13, no. 4, Article no. 60, 2017.
2. H. Esmaeilzadeh, A. Sampson, L. Ceze, and D.Burger. "Neural Acceleration for General-Purpose Approximate Programs," In Proc. *of Micro*, pp.105-115, 2012.
3. B. K. Mohanty, S. K. Patel, "Area–delay–power efficient carry select adder", IEEE TCAS-II, vol. 61, no. 6, pp. 418-422, Jun. 2014.
4. M. Bilal, S. Masud, and S. Athar, “FPGA Design for Statistics-Inspired Approximate Sum-of-Squared-Error Computation in Multimedia Applications,” IEEE TCAS-II, vol. 59, no. 8, pp. 506-510, 2012.
5. M. Kamal, A. Ghasemazar, A. Afzali-Kusha, andM. Pedram, “Improving efficiency ofextensible processors by usingapproximate custom instructions,” in Proc. *DATE*, 2014.
6. O. Akbari, M. Kamal, A. Afzali-Kusha, and M. Pedram, “RAP-CLA: A reconfigurable approximate carry look-ahead adder,” *IEEE TCAS-II*, vol. 65, no. 8, pp. 1089–1093, 2018.
7. Y. Kim, Y. Zhang, and P. Li. “An energy efficient approximate adder with carry skip for error resilient neuromorphic VLSI systems,” In Proc. *ICCAD*, 2013, pp. 130–137.
8. B. Kahng, S. Kang, "Accuracy-configurable adder for approximate arithmetic designs," In Proc. *DAC*, 2012, pp.820-825.
9. J. Hu and W. Qian, "A new approximate adder with low relative error and correct sign calculation," In Proc. *IEEE DATE*, pp. 1449-1454, 2015.
10. W. Xu, S. S. Sapatnekar, and J. Hu. "A Simple Yet Efficient Accuracy Configurable Adder Design," In Proc. *ISLPED,* 2017.
11. R. Ye, T. Wang, F. Yuan, R. Kumar and Q. Xu, “On Reconfiguration-Oriented Approximate Adder Design and Its Application,” In Proc. *ICCAD*, 2013, pp. 48-54.
12. M. Shafique, W. Ahmad, R. Hafiz and J. Henkel, "A low latency generic accuracy configurable adder," In Proc. *DAC*, 2015, pp. 1-6.
13. N. Zhu, W., et. al., “Design of low-power high-speed truncation-error-tolerant adder and its application in digital signal processing,” *IEEE TVLSI*, vol. 18, no. 8, pp. 1225–1229, 2011