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

IJIEMR Transactions, online available on 4th Dec 2018. Link :http://www.ijiemr.org/downloads.php?vol=Volume-07&issue=ISSUE-12

Title: **A LOW-POWER HIGH PERFORMANCE CONFIGURABLE ADDER DESIGN FOR APPROXIMATE COMPUTING**

Volume 07, Issue 12, Pages: 819–826.

Paper Authors

RAMADEVI VEMULAPALLI, KETAVATH SAKRU NAIK
Guntur Engineering College, yanamadala, AP, 522019.

.

International Journal for Innovative Engineering and Management Research

PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL

www.ijiemr.org

A LOW-POWER HIGH PERFORMANCE CONFIGURABLE ADDER DESIGN FOR APPROXIMATE COMPUTING

[1] RAMADEVI VEMULAPALLI, [2] KETAVATH SAKRU NAIK

[1]M.Tech student, Dept of ECE, Guntur Engineering College, yanamadala, AP, 522019

[2]Assistant Professor, Dept of ECE, Guntur Engineering College, yanamadala, AP, 522019

ABSTRACT: Approximate computing is an efficient approach for error-tolerant applications because it can trade off accuracy for power. Addition is a key fundamental function for these applications. In this paper, we proposed a low-power yet high speed accuracy-configurable adder that also maintains a small design area. The proposed adder is based on the conventional carry look-ahead adder, and its configurability of accuracy is realized by masking the carry propagation at runtime. Compared with the conventional carry look-ahead adder, with only 14.5% area overhead, the proposed 16-bit adder reduced power consumption by 42.7%, and critical path delay by 56.9% most according to the accuracy configuration settings, respectively.Furthermore, compared with other previously studied adders, the experimental results demonstrate that the proposed adder achieved the original purpose of optimizing both power and speed simultaneously without reducing the accuracy.

KEY WORDS: Approximate computing; accuracy-configurable adder; high-speed adder; low-power adder.

I.INTRODUCTION

Power constraints are a well-known challenge in advanced VLSI technologies. Low power techniques for the conventional exact computing paradigm have been already extensively studied. A comparatively new direction is approximate computing, where errors are intentionally allowed in exchange for power reduction. In many applications, such as audio, video, haptic processing and machine learning, occasional small errors are indeed acceptable. Such error-tolerant applications are found in abundance in emerging applications and technologies.

A great deal of approximate computing research has been concentrated on Arithmetic circuits, which are essential building blocks for most of computing on Hardware. In particular, several approximate adder designs have been developed. One such design achieves 60% power reduction for DCT (Discrete Cosine Transform) computation without making any discernible difference to the images being processed. In realistic practice, accuracy requirements may vary for different applications. In mobile computing devices, different power modes may entail different accuracy constraints even for theSame application. Specifically, arithmetic accuracy can be adjusted at runtime using methods such as dynamic voltage and frequency scaling (DVFS) to obtain the best accuracy power trade-off. The benefit of runtime accuracy adjustment is demonstrated but their approximation is realized by voltage over-scaling, where errors mostly occur at the timing critical path associated with the most significant bits, i.e, errors are often large.Addition is one of the four elementary operations in mathematics, the other being subtraction, multiplication and division. In digital systems, addition forms the most important operation. This is primarily because we can perform operations like subtraction, multiplication and division using the addition operation. Hence the design of a very fast, accurate and a lower power consumption adder directly results in the increased speed of the device for faster computational purpose as well as an improved life. While designing an adder, lot of constraints comes into picture. The trade-off between speed and size being the most important one. The most basic adder has a very small area and is very easy to implement. But the delay involved in obtaining the output is very huge. Hence the new designs came to picture. But in current age the power consumed by the device is also a very important constraint. Hence designing an adder which is very close in meeting all these requirements is a very important. A few accuracy configurable adder designs that use approximation schemes other than voltage over-scaling have been proposed. An early work, called ACA, starts with an Approximate adder and augments it with an error detection and correction circuit, which can be configured to deliver varying approximation levels or accurate computing. Its baseline approximate adder contains significant redundancy and the error detection/correction circuit further increases area overhead. The ACA design is generalized to a flexible framework GeAr. In both ACA and GeAr, the error correction must start from the least significant bits and hence accuracy improves slowly in the progression of configurations. The work of Accurus modifies ACA/GeAr to overcome this drawback and achieves graceful degradation. However, in ACA, GeAr as well as Accurus, the error correction circuit is pipelined, implying that the computation in accurate mode takes multiple clock cycles and causes data stallsAn alternative direction of accuracy configurable adder design is represented by GDA and RAP-CLA. These methods start with an accurate adder and use carry prediction for optional approximation. As such, they no longer need error detection/correction and do not incur any data stall. In addition, they intrinsically support graceful degradation. TheGDA design is composed by accurate CRA (Carry Ripple Adder) and extra configurable carry prediction circuitry, similar as the carry look-ahead part of CLA (Carry Look-ahead Adder). Thus, its area is generally quite large. RAP-CLA is based on accurate CLA design and reuses a portion of the carry look-ahead circuit as carry prediction. This leads to an overall area that is less than GDA but greater than CLA. The carry-prediction-based approach is shown to be superior to error-correction-based design. To reduce the overall error, a few approximate designs have been developed by intentionally allowing errors in lower bits with shorter carry chain in addition operation. a design that considers only the previous k inputs instead of all input bits can approximate the result with the benefit in half of the logarithmic delay. Reliable variable latency carry select adders shows a speculation technique that introduces carry chain truncation and carry select addition as a basis. A series of error tolerant adders (ETAI, ETAII, and ETAIIM), which truncate the carry propagation chain by dividing the adder into several segments, have been proposed. Correlation-aware speculative adder relies on the correlation between MSBs of input data and carry-in values. Another approximate adder that exploits the generate signals for carry speculation is presented. These designs focus on static approximation, which pursue almost correct results at the required accuracy. However, in some applications such as image processing or audio/video compression, the required accuracy might vary during run time. To meet the need for runtime accuracy adjustment, a series of designs are developed to implement accuracy-configurable approximation, which could be reconfigured online to save more power.

II.RELATEDWORK

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. Most of these applications have an inherent tolerance for insignificant inaccuracies. By exploiting the inherent tolerance feature, approximate computing can be adopted for a trade-off between accuracy and power. At present, this trade-off plays a significant role in such application domain. As computation quality requirements of an application may vary significantly at runtime, it is preferable to design quality configurable systems that are able to trade-off computation quality and computational effort according to application requirements. The previous proposals for configurability suffer the cost of the increase in power or in delay. 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 does with a comparable delay and area. In addition, the delay observed with the proposed adder is much smaller than that of with 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 accuracy configurable adders in a real image processing application. 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 where *i* is denoted the bit position from the least significant bit.Note that owing to reuse of the circuit of A*i* *XOR* B*i* for S*i* generation, here P*i* is defined as A*i* *XOR* B*i* instead of A*i* *OR* B*i*. Because C0 is equal to G0 , if G0 is 0, C0 will be 0. 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* , C*i* will be 0 when G0, G1, … , G*i* are all 0. This means that the carry propagation from C0 to C*i* is masked. We can obtain that S*i* is equal to P*i* when C*i*-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. 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 A*i* and B*i* are both 1, when M_X*i* = 1, the accurate sum S*i* and carry C*i* will be 0 and 1 ({C*i* , S*i*} = {1,0}); when M_X0 , M_X1 , … , M_X*i* are all 0, S*i* is equal to P*i* (= A*i* *XOR* B*i* = 0) as an approximate sum and C*i* is equal to 0 ({C*i* , S*i*} = {0, 0}) as discussed above.

III.EXISTEDSYSTEM

The structure of the Existed16-bit adder is shown in Fig. 1 as an example. The error rates calculated by our method match well with experimental results, which demonstrates the correctness of our mathematical analysis. But the propagation of inaccurate carry bit may cause error in higher bit, which also be counted in the calculation of lower bit. Therefore, we need

To exclude those errors to avoid over calculation in the total error.

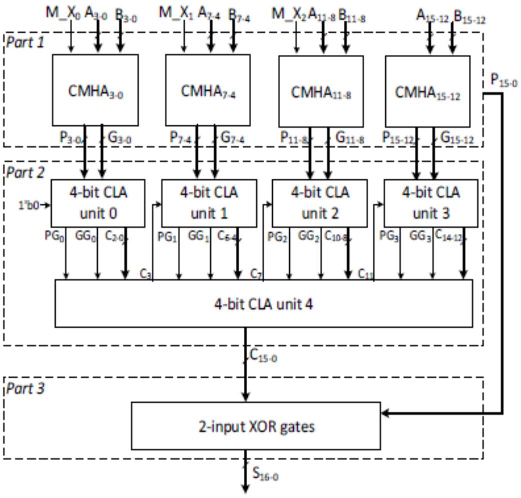


FIG. 1: EXISTED SYSTEM

The structure of the Existed16-bit adder is shown in Fig. 1 as an example. The error rates calculated by our method match well with experimental results, which demonstrates the correctness of our mathematical analysis. But the propagation of inaccurate carry bit may cause error in higher bit, which also be counted in the calculation of lower bit. Therefore, we need To exclude those errors to avoid over calculation in the total error. Four groups (CMHA3-0, CMHA7-4, and 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 3generate the sum 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. 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.Consider an n-bit CLA, whose half adders for G and P signals preparation are replaced by CMHAs. In this case, an n-bit 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. 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.

III. PROPOSED SYSTEM

In this paper, we propose a new carry-prediction-based accuracy configurable adder design: SARA (Simple Accuracy Reconfigurable Adder). It is a simple design with significantly less area than CLA, which, to the best of our knowledge, has not been achieved in the past in accuracy configurable adders. SARA inherits the advantages of all previous carry-prediction based approaches: no error correction overhead, no data stall and allowing graceful degradation. Compared to GDA, SARA incurs 50% less PDP (Power Delay Product) and can reach the same PSNR (Peak Signal-to-Noise Ratio). Moreover, SARA demonstrates remarkably better accuracy-power-delay.Any false propagate bit from the addends results in a shorter carry propagation chain. When the actual carry propagation chain is short, there is no need to use approximation configuration, which is intended to cut carry chain shorter. We propose a DAR technique: the output of a MUX in SARA is set to approximation mode only when a potentially long carry chain is detected. The main idea of self-configuration is based on the observation that the actual worst case path delay depends on addend values. Specifically, the actual path delay is large only when a carry is propagated through several consecutive bits.Trade-off than the latest, and arguably the best, previous work RAP-CLA. A delay-adaptive reconfiguration technique is developed to further improve the accuracy-power-delay Trade-off. The proposed designs are also validated by DCT computation in image processing. We review a few representative works on accuracy configurable adder design and show the relation with our method. These designs can be generally categorized into two groups: error-correction-based configurations and carry prediction- based configurations.

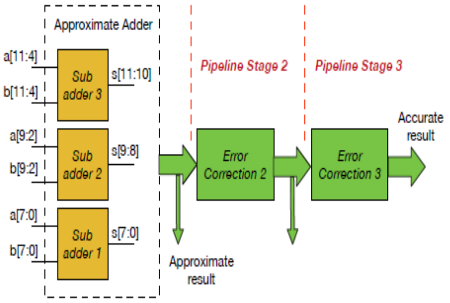


FIG. 2: PROPOSED SYSTEM

The main idea of an error-correction-based approach is shown in Figure 2. The scheme starts with an approximate adder (the dashed box), where the carry chain is shortened by using separated sub-adders with truncated carry-in. In order to reduce the truncation error, the bit-width in some sub-adders contains redundancy. For example, subadder2 calculates the sum for only bit 8 and 9, but it is an 8-bit adder using bit [9: 2] of the addends, 6 bits of which are redundant. Even with the redundancy, there is still residual error which is detected and corrected by additional circuits. In Figure 2, the errors of sub-adder2 must be corrected by error-correction2 before the errors of sub-adder3 are rectified by error-correction3. As such, the configuration progression always starts with small accuracy improvements. The redundancy and error detection/correction incur large area overhead. Since the error correction circuits are usually pipelined, an accurate computation may take multiple clock cycles and could stall entire data path, depending on the addend values.

IV. RESULTS

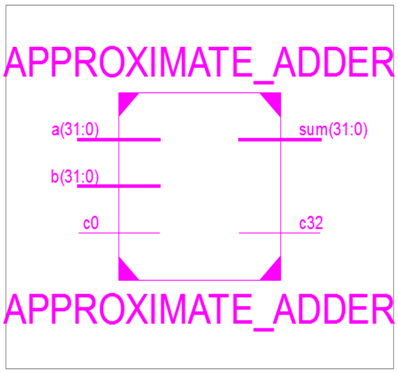


FIG. 3: RTL SCHEMATIC

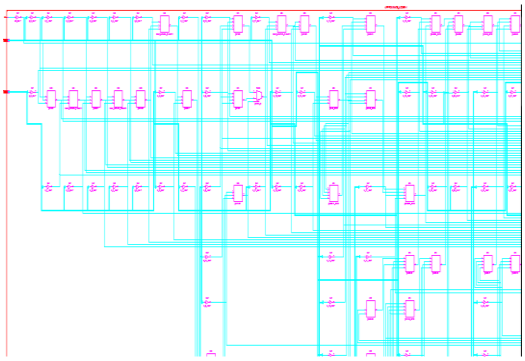
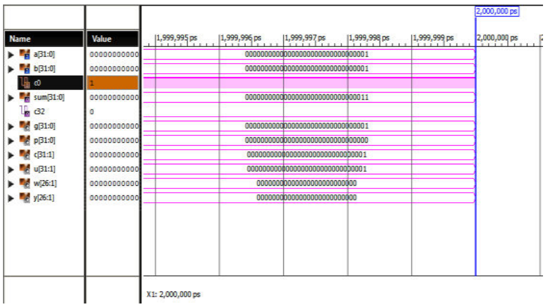
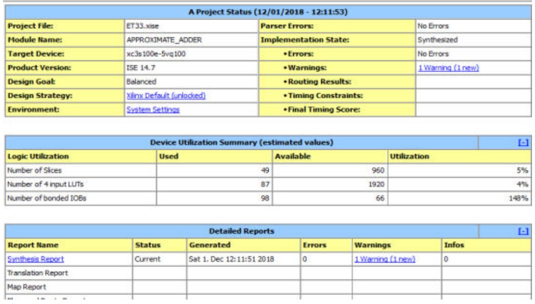


FIG. 4: TECHNOLOGY SCHEMATIC



```
Name        Value
a[31:0]     00000000000   00000000000000000000000000000001
b[31:0]     00000000000   00000000000000000000000000000001
c0          1
sum[31:0]   00000000000   00000000000000000000000000000011
c32         0
g[31:0]     00000000000   00000000000000000000000000000001
p[31:0]     00000000000   00000000000000000000000000000000
c[31:1]     00000000000   0000000000000000000000000000001
u[31:1]     00000000000   0000000000000000000000000000001
w[26:1]     00000000000   00000000000000000000000000
y[26:1]     00000000000   00000000000000000000000000
```

FIG. 4: OUTPUT WAVEFORM



A Project Status (12/01/2018 - 12:11:53)			
Project File:	ET33.xise	Parser Errors:	No Errors
Module Name:	APPROXIMATE_ADDER	Implementation State:	Synthesized
Target Device:	xc3s100e-5vq100	•Errors:	No Errors
Product Version:	ISE 14.7	•Warnings:	1 Warning (1 new)
Design Goal:	Balanced	•Routing Results:	
Design Strategy:	Xilinx Default (unlocked)	•Timing Constraints:	
Environment:	System Settings	•Final Timing Score:	

Device Utilization Summary (estimated values)			[-]
Logic Utilization	Used	Available	Utilization
Number of Slices	49	960	5%
Number of 4 input LUTs	87	1920	4%
Number of bonded IOBs	98	66	148%

Detailed Reports					[-]
Report Name	Status	Generated	Errors	Warnings	Infos
Synthesis Report	Current	Sat 1. Dec 12:11:51 2018	0	1 Warning (1 new)	0
Translation Report					
Map Report					

FIG. 5: REPORT

VI. CONLUSION

In this paper, an accuracy-configurable adder without suffering the cost of the increase in power or in delay for configurability was proposed. The proposed adder is based on the conventional CLA, and its configurability of accuracy is realized by masking the carry propagation at runtime. The experimental results demonstrate that the proposed adder delivers significant power savings and speedup with a small area overhead than those of the conventional CLA. Furthermore, compared with previously studied configurable adders, the experimental results demonstrate that the proposed adder achieves the original purpose of delivering an unbiased optimized result between power and delay without sacrificing accuracy. It was also found that the quality requirements of the evaluated application were not compromised.

VII. 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 Microarchitecture (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 Comptuer-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.