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

IJIEMR Transactions, online available on 13th Dec 2019. Link :http://www.ijiemr.org/downloads.php?vol=Volume-08&issue=ISSUE-12

Title **LOW POWER APPROXIMATE ADDER FOR ENERGY EFFICIENT**

Volume 08, Issue 12, Pages: 101-109.

Paper Authors

SIDDAVATAM VINAYPRAKASH REDDY, P.KUMAR

SIR C.V. RAMAN Institute of Technology & Science, AP, India

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

LOW POWER APPROXIMATE ADDER FOR ENERGY EFFICIENT

SIDDAVATAM VINAYPRAKASH REDDY[1], P.KUMAR[2]
[1]PG Scholar, Dept of ECE, SIR C.V. RAMAN Institute of Technology & Science, AP, India
[2] Assistant Professor, Dept of ECE, SIR C.V. RAMAN Institute of Technology & Science, AP, India

Abstract: Many signal processing blocks, especially those meant for video and speech, are error tolerant which makes it possible to use inaccurate arithmetic units. This is exploited in systems to save power and area as well as to reduce the delay. Approximation is mainly done using voltage over scaling and architectural approximation Approximate circuit design has gained significance in recent years targeting applications like media processing where full accuracy is not required. In this paper, we propose an approximate adder in which the approximate part of the sum is obtained by finding a single optimal level that minimizes the mean error distance. Therefore hardware needed for the approximate part computation can be removed, which effectively results in very low power consumption. We compare the proposed adder with various approximate adders in the literature in terms of power and accuracy metrics

INTRODUCTION

Approximate adders are the building block for any arithmetic circuit used in exact computing Approximate adders are derived from accurate adders based on various approximations which have incorrect outputs for sum (S) and carry out (Cout) for some input combinations. The approximation reduces the number of transistors in adders as compared to accurate adders. Thus approximate adders also decrease power consumption and propagation delay as compared to accurate adders. Various approximate adders have been reported in the literature . However approximate adders using Complementary Pass Transistor Logic (CPL) suffers from low output voltage swing. Yang proposed Transmission Gate (TG) based approximate adders which offered good output voltage swing but consumed more power than approximate adders based on CPL. It required a large number of transistors and also increased the power consumption and delay. The design of approximate adder using Complementary Pass Transistor Logic(CPL) which offer the advantage in terms of power and delay .Another XOR/XNOR based approximate adder reported in was derived from ten transistors (10T) accurate adder. This approximate adder suffered from low output swing as compared to approximate adders based on CPL. Buffers are required to restore the output swing which again increases the power consumption and delay. The approximate adders also decrease power consumption and propagation delay as compared to accurate adders. The approximate adder based on 14T requires offers good output voltage swing and requires less number of transistors than approximate mirror adder design and approximate adders based on TG and CPL.

The proposed 1-Bit approximate adder is also compared with existing designs based on total error distance which measures the accuracy of the circuit adders. It required a large number of transistors and also increased the power consumption and delay.

In conventional digital VLSI design, one usually assumes that a usable circuit/system should always provide definite and accurate results. But in fact, such perfect operations are seldom needed in our non digital world experiences. The world accepts "analog computation," which generates "good enough" results rather than totally accurate results . The data processed by many digital systems may already contain errors. In many applications, such as a communication system, the analog signal coming from the outside world must first be sampled before being converted to digital data. The digital data are then processed and transmitted in a noisy channel before converting back to an analog signal. During this process, errors may occur anywhere. Furthermore, due to the advances in transistor size scaling, factors such as noise and process variations which are previously insignificant are becoming important in today's digital IC design. Based on the characteristic of digital VLSI design, some novel concepts and design techniques have been proposed. The concept of error tolerance (ET) and the PCMOS technology are two of them. According to the definition, a circuit is error tolerant if: 1) it contains defects that cause internal and may cause external errors and 2) the system that incorporates this circuit produces acceptable results . The "imperfect" attribute seems to be not appealing. However, the need for the error-tolerant circuit was foretold in the 2003 International Technology Roadmap for Semiconductors (ITRS). To deal with error-tolerant problems, some truncated adders/multipliers have been reported but are not able to perform well in either its speed, power, area, or accuracy. The "flagged prefixed adder" performs better than the non flagged version with a 1.3% speed enhancement but at the expense of 2% extra silicon area. As for the "low-error area-efficient fixed-width multipliers" it may have an area improvement of 46.67% but has average error reaching 12.4%. Of course, not all digital systems can engage the error-tolerant concept. In digital systems such as control systems, the correctness of the output signal is extremely important, and this denies the use of the error tolerant circuit. However, for many digital signal processing (DSP) systems that process signals relating to human senses such as hearing, sight, smell, and touch, e.g., the image processing and speech processing systems, the error-tolerant circuits may be applicable. Increasingly huge data sets and the need for instant response require the adder to be large and fast. The traditional ripple-carry adder (RCA) is therefore no longer suitable for large adders because of its low-speed performance. Many different types of fast adders, such as the carry-skip adder (CSK), carry-select adder (CSL) , and carry-look-ahead adder (CLA), have been developed. Also, there are many low-power adder design techniques that have been proposed. However, there are always trade-offs between speed and power. The error-tolerant design can be a potential solution to this problem. By sacrificing some accuracy, the ETA can attain great improvement in both the power consumption and speed performance Error Tolerant Addition: The commonly used terminologies in Error Tolerant addition are as follows: Overall error (OE): OE=|Rc-Re |, where Re is the result obtained by the Error tolerant addition technique, and Rodents the correct result (all the results are represented as decimal numbers). Accuracy (ACC): In the scenario of the error tolerant design, the accuracy of an addition process is utilized to indicate how “correct” the output of an adder is for a particular input. It is defined as ACC%=(1-(OE/Rc)) x 100. Its value ranges from 0-100%.Addition Arithmetic: In the conventional adder circuit, the delay is mainly attributed to the carry propagation chain along the critical path, from the least significant bit (LSB) to the most significant bit (MSB). Also glitches in the carry propagation chain dissipate a significant proportion of dynamic power dissipation. Therefore, if the carry propagation can be eliminated or curtailed, a great improvement in speed performance and power consumption can be achieved. This new addition arithmetic can be illustrated via an example shown below.

2. Existing System

Which is a popular way of implementing an FA. It consists of a total of 24 transistors. Note that this implementation is not based on complementary CMOS logic, and thus provides an opportunity to cleverly design an approximate version with removal of selected transistors.

Approximation 1: In order to get an approximate MA with lesser transistors, we start to remove transistors from the conventional schematic one by one. In doing so, we need to ensure that any input combination of A,B and C in does not result in short circuits or open circuits in the simplified schematic. We also impose another criterion that the resulting simplification should introduce minimal errors in the FA truth table. A judicious selection of transistors to be removed (ensuring no open or short circuits) results in a schematic shown in Figure 2. Clearly, this schematic has 8 less transistors compared to the conventional MA schematic. A close observation of the truth table of an FA shows that Sum=Cout for 6 cases out of 8, except for the input combinations A=0,B=0,Cin=0andA=1,B=1,Cin=1.Now,inthe conventional MA ,Cout is computed in the first stage. Thus an elegant way of simplifying the MA further is to discard the Sum circuit completely. Although one can directly set Sum=Cout as shown in Figure 1, we introduce a buffer stage after Cout (see Figure 3) to implement the same functionality. The reason for this can be explained as follows. If we set Sum=Cout as it is in the conventional MA, the total capacitance at the Sum node would be a combination f4 source-drain diffusion and 2gate capacitances. This is an appreciable increase compared to the conventional case. Such a design would lead to a delay penalty in cases where two or more multi-bit approximate adders are connected in a chained fashion, which is a common scenario in DSP applications. Thus we combine the simplified circuit for Cout in Figure 2 with the idea that Sum=Cout for 6 cases out of 8. Figure 3 shows the simplified MA obtained using this technique. This introduces 1 error in Cout and 3 errors in Sum, as shown inTableI.3) Approximation 2 Again, a careful observation of the FA truth table shows that Cout= A for 6 cases out of 8. Similarly, Cout=B for 6 cases out of 8. Since A and B are interchangeable, we consider Cout=A. Thus we propose a second approximation(approximation2) where we just use an inverter with input A to calculate Cout and Sum is calculated similar to the simplified MA in Figure 2.Figure 4 shows the simplified circuit obtained using this technique. This introduces 2 errors in Cout and 3 errors in Sum.

3. PROPOSED SYSTEM

Approximate adders consist of two parts: accurate part and inaccurate part. The accurate part addition is done by the conventional adder whose input carry in connected to ground. The inaccurate part consist of two blocks. First is the Carry Free Addition Block (CFAB) and second is the Control Block (CB) which decides the working mode of the circuit of overall inaccurate-part.

Proposed Architecture of 8 Bit Approximate Adder: Here we proposed a new architecture of half adder and full adder as we know for 8 bit addition there is total 7full adder and 1 half adder is require. But in proposed approach we propose a new novel 8 bit architecture where we can put some error on lsb bit of adder. Here in approximate half and full adder there is no any carry generation unit. So on first LSB bit we are using proposed approximate half adder and on second LSB bit we use one approximate full adder for next third bit there is no any carry generate so there is no need to use one full adder so at the place of full adder we are using one half adder and after that we use 5 full adder. So as we can see with small error generation we can reduce the hardware requirement and we can make justice with SPAA matrices.

Design of Modified Error Tolerant Adder:

The Fast Fourier Transform (FFT) is very important function in Digital Signal Processing (DSP)
and image processing application. Very large number of additions and multiplications are involved in computational process of FFT. It is therefore a very good platform for embedding novel approaches like Error Tolerance (ET) in the computational process presented in this research work.

Although in today's VLSI world, exact results are always better than accurate results how ever exact results are seldom needed in analog Integrated Circuit (IC) design because of its complexity. Based on this, some novel concepts like error tolerance have come into existence. Today, there are some designs such as adders/multipliers which have been proposed but they could not perform well either in terms of power, speed, accuracy or area.

It is true that not all the digital systems can allow the concept of error tolerance. For example, application involving control systems where the accuracy of output signal is very important and hence the concept of error tolerance cannot be applied. However, there are many examples of applications like DSP applications in which human sensing signals such as sight, hearing, touch and smell are being processed, where we can use this concept of error tolerance.

3.1. Error Tolerant Adder:

Based on the characteristics of digital VLSI design, some novel concepts and design techniques have been proposed. The concept of Error Tolerance (ET) and the PCMOS technology are two of them. According to the definition, a circuit is error tolerant if: (1) it contains defect that causes internal and may cause external errors and (2) the system that incorporates this circuit produces acceptable results .

3.1.1. Necessity of Error Tolerant Adder:

Requirement for large data sets and fast response has made the adder to be large and fast. Thus, the conventional Ripple Carry Adder (RCA) cannot be used because of its low-speed performance. There are other adders such as Carry Look Ahead (CLA) adder, Carry Select (CSL) adder, Carry Skip (CSK) adder but there is always trade-off between speed and power. The Error Tolerant Adder (ETA) is potential solution to this problem. This research further improves the design model of ETA thereby reducing the chip area and cost of the design.

3.1.2. Adopted Addition Arithmetic:

The signal propagating though the different paths with unequal delay will reach the output at different time. This introduces static-0 and static-1 hazards also called glitches. Thus, in a similar way, carry propagation also decreases the speed of the adder. More so, large power is consumed mainly because of glitches occurring during this process. So, if the carry propagation is avoided, then there can be a large improvement in speed performance and power consumption. The adopted addition arithmetic from where A and B are inputs and S is output. Here, the input operand are divided into two parts. First, the accurate part that includes several higher order bits and second the inaccurate part that have the Least Significant Bits (LSBs). Equal length of both part is not necessary. Consider for an example, A and B are 16-bits input operands.

Here for ease of partition both operands are divided in equal part. Addition of accurate part is performed right to left in the conventional way in order to sustain precision as higher bits plays important role in determining accuracy. For inaccurate part, a special addition method is adopted to speed up the process with minimal power consumption. This method eliminates the carry propagation path.

The process is defined in the following three steps: (1) Check every bit position from left to right.(2) If both input bits combination are other than Is, conventional 1-bit addition is performed. (3)If both the input bits found to be ' 1 ' , then all the sum bits to the right of the first occurrence of input bits 'I' are set to '1' irrespective of the inputs.

Consider for an example, A = " 1011001II0011010" and B = "0110100100010 11". This addition should actually yield "100011 lOOlOIOI 101" (72877) for a normal addition operation but it gives"10001110010011111" (72863) with this method. Thus, the Total Error (TE) can be computed as TE = 172877

- 72870] = 14 and Accuracy (ACC) ACC = (1 - (14/72877)) * 100 % = 99.99 % [8].Thus, by eliminating the carry propagation in inaccurate part there can be a huge reduction in power consumption as compared to conventional fast adders.

3.1.3. Hardware Implementation:

The block diagram of a modified ETA. This consist of two parts: accurate part and inaccurate part. The accurate part addition is done by the conventional adder whose input carry is connected to ground. The inaccurate part consist of two blocks. First is the Carry Free Addition Block (CFAB) and second is the Control Block (CB) which decides the working mode of the circuit of overall inaccurate-part.

Concept of Error Tolerant Adder (ETA) has already been introduced by Zhu et al. This research further improves the design of this adder which reduces the area and thereby decreasing

the power consumption. In the present work, we further improved the modified XOR block of ETA and subsequently the XOR logic is replaced by improved modified OR block.

3.1.4. Dividing the Adder:

The dividing method is based on the accuracy, speed and power. For more accuracy, there are less number of bits in the inaccurate part and vice-versa . With the method used in we can verify the accuracy performance of the adder and the partition is then adjusted to meet the requirements. Thus having less bits in the accurate part can greatly reduce the power consumption. Considering the mentioned partition method but for the ease of division procedure, 16-bits were divided equally in accurate part and inaccurate part in the present work.

3.1.5. Design of the Accurate Part:

In our proposed 16-bit modified ETA, inaccurate part consist of 8-bits and accurate part contains

remaining 8-bits. Thus, the overall delay can be determined by inaccurate part and hence the adder required for accurate part need not to be faster. The most power saving Ripple Carry Adder (RCA) is chosen for the accurate part.

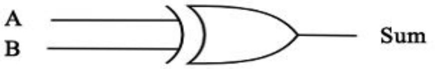


Fig 3: Approximate half adder

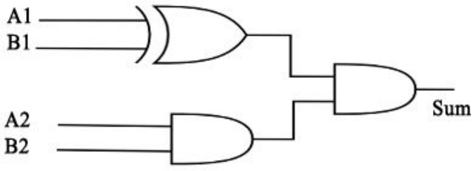


Fig 4: Approximate full adder

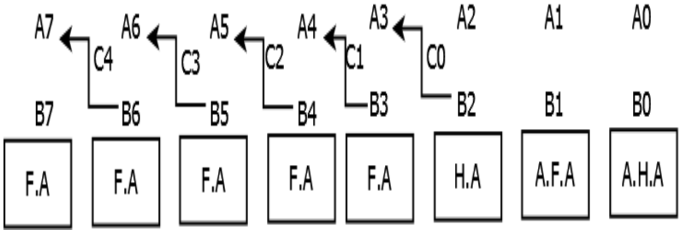


Fig 5: Approximate 8 bit adder

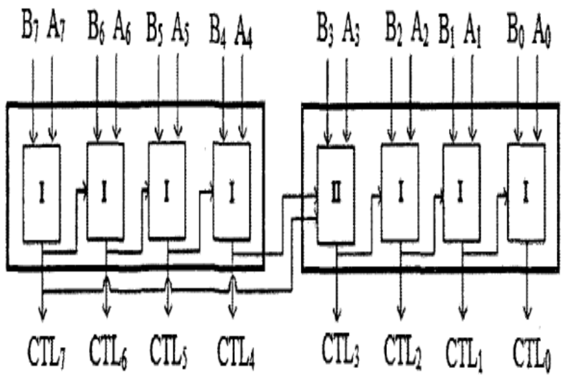


Fig 6: Approximate control signal block

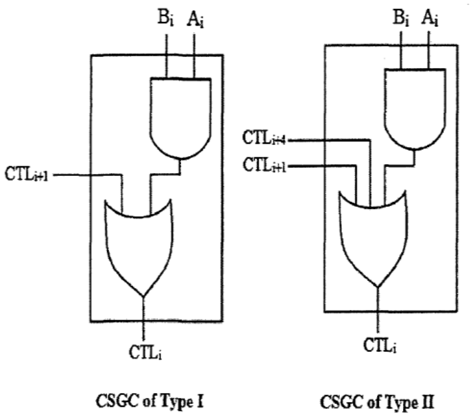


Fig 7: Approximate control signal generator type-1&type-2

3.1.6. Design of the Inaccurate Part:

This is the most critical portion of design since it determines the speed, accuracy and power consumption. It consist of two parts, namely Carry Free Addition Block (CFAB) and Control Block (CB). The schematic of modified OR block. In proposed modified OR block-1 two more transistor are used whose input depends on CTL signal, where CTL signal is a control signal coming out from Control Block (CB). When CTL = 0, Ml is on and M2 is off leaving the OR block circuit to operate in normal OR operation mode. When CTL=1, Ml is OFF and M2 is ON, hence the output sets to '1' by connecting output node to VDD. Problem with proposed block in presence of NMOS as a pass transistor in output node i.e. we knows that NMOS is poor in passing strong logic '1', so, we will get degraded value of output while connecting to VDD. To solve the problem of passing strong logic ' 1' we have proposed another modified OR block-2 by replacing NMOS with PMOS and rearranging the connection of transistors.

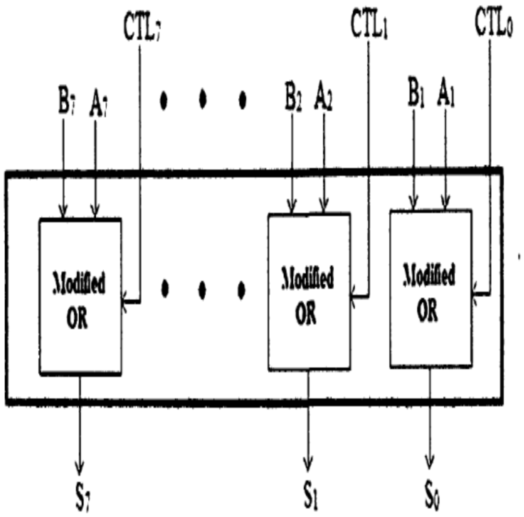


Fig 8: Modified OR block-1

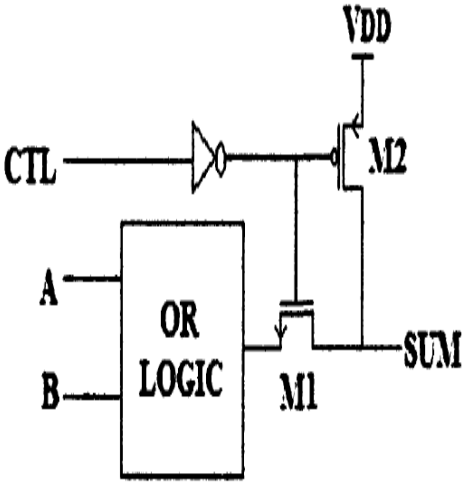
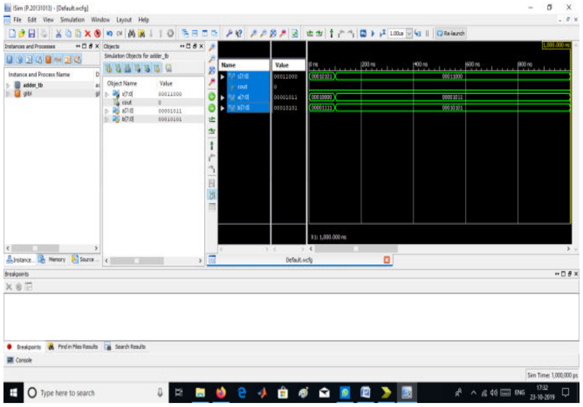


Fig 9: Modified OR block-2

4. RESULT


ISim (P.20131013) - [Default.wcfg]
File Edit View Simulation Window Layout Help
Re-launch
Instances and Processes
Objects
Default.wcfg

Simulation Objects for adder_tb

Instance and Process Name
adder_tb
glbl

Object Name	Value
s[7:0]	00011000
cout	0
a[7:0]	00001011
b[7:0]	00010101

Name	Value	0 ns		
s[7:0]	00011000	00010101	00011000
cout	0		
a[7:0]	00001011	00010000	00001011
b[7:0]	00010101	00001111	00010101

X1: 1,000.000 ns

Fig 10 Simulation result

5. CONCLUSION

In this work, trade-offs between circuit area, energy requirements and accuracy of approximate adder and multiplier designs are explored for pipelined processor data paths. A two-phase architecture mapping and register balancing synthesis with adaptive timing constraints is used to obtain area-efficient gate-level implementations for a parameterizable amount of pipeline stages. The trade-off analysis shows that besides to more performance, inherent data path pipelining can also be used to implement more area- and energy-efficient approximate arithmetic units when the maximum performance is not required, obtaining area reductions of up to 20% and energy reductions of up to 11% for the same target clock period constraint. Furthermore, when higher performance is desired for a specific architecture, pipelining can be used instead of choosing another approximate unit with lower accuracy. The applied register balancing to implement pipelined de-signs does not modify the functionality of the arithmetic architecture, so there is no accuracy variation with the number of pipeline stages. In general, state-of-the-art approximate adder and multiplier units are designed for single-cycle execution, however, pipelined datapaths can be further used to structurally exploit multi-cycle designs. By including the concept to pipelining directly in the architecture description instead of using register balancing, more area- and energy-efficient adder and multiplier implementations at given performance and accuracy requirements could be developed in future work.

BIBILOGRAPHY

[1]. K. Vitoroulis and A. J. Al-Khalili, “Performance of Parallel Prefix Adders Implemented with FPGA technology,” IEEE Northeast Workshop on Circuits and Systems, pp. 498-501, Aug. 2007.

[2]. D. Gizopoulos, M. Psarakis, A. Paschalis, and Y. Zorian, “Easily Testable Cellular Carry Lookahead Adders,” Journal of Electronic Testing: Theory and Applications 19, 285-298, 2003.

[3]. S. Xing and W. W. H. Yu, “FPGA Adders: Performance Evaluation and Optimal Design,” IEEE Design & Test of Computers, vol. 15, no. 1, pp. 24-29, Jan. 1998.

[4]. M. Bečvář and P. Štukjunger, “Fixed-Point Arithmetic in FPGA,” Acta Polytechnica, vol. 45, no. 2, pp. 67-72, 2005.

[5]. P. M. Kogge and H. S. Stone, “A Parallel Algorithm for the Efficient Solution of a General Class of Recurrence Equations,” IEEE Trans. on Computers, Vol. C-22, No 8, August 1973.

[6]. P. Ndai, S. Lu, D. Somesekhar, and K. Roy, “Fine-Grained Redundancy in Adders,” Int. Symp. on Quality Electronic Design, pp. 317-321, March 2007.

[7]. T. Lynch and E. E. Swartzlander, “A Spanning Tree Carry Look ahead Adder,” IEEE Trans. on Computers, vol. 41, no. 8, pp. 931-939, Aug. 1992.

[8]. N. H. E. Weste and D. Harris, CMOS VLSI Design, 4th edition, Pearson–Addison-Wesley, 2011.

[9]. R. P. Brent and H. T. Kung, “A regular layout for parallel adders,” IEEE Trans. Comput., vol. C-31, pp. 260-264, 1982.

[10]. D. Harris, "A Taxonomy of Parallel Prefix Networks," in Proc. 37th Asilomar Conf. Signals Systems and Computers, pp. 2213–7, 2003.