

Digital Signal Processing Using High Speed Low Power Tolerant Adder

Abstract: Low power is an essential requirement to process various signal processing algorithm and architecture used for portable multimedia device. In modern VLSI technology the occurrence of all kind of error has become ineluctable. The useful information gathered by human being for multimedia application has some faulty output. Therefore there is no need to produce exactly correct numerical output. Previous research in context is based on the considering tradeoff between power and speed. The concept of error tolerance compromises with correctness, a large reduction in power consumption and improvement in speed can be achieved. In this paper the tolerant adders used for digital signal processing. The world accepts “analog computation,” which generates “good enough” results rather than totally accurate results [1]. The data processed by many digital systems may already contain errors.

Preeti Arora

Lecturer Physics
MR International School, Gurgaon
preetiara20@gmail.com

Keywords: Approximate adder, Low Power ,DSP application, tolerant adder

I. INTRODUCTION

In applications such as a communication system, the analog signal coming from outside world must first be sampled before it can be converted to digital data at the front end of the system. The digital data is then processed and transmitted in a noisy channel before being converted back to the analog signal at the back end. During this process, errors may occur everywhere. Furthermore, due to the advances in transistor size scaling, the previously insignificant factors such as noise and process variations are becoming important impacts in today’s digital IC design [2]. Based on the characteristic of digital VLSI design, some novel concepts and design techniques have been proposed. The concept of error tolerance (ET) has proposed in [3]–[10]. According to the definition, a circuit is error tolerant if: 1) it contains defects that cause internal and external errors and 2) the system that includes this circuit produces acceptable results [3] not accurate but approximate. The “imperfect” result not appealing for the system attribute. However, the need for the error-tolerant circuit [3]–[10] was foretold in the 2003 International Technology Roadmap for Semiconductors (ITRS) [2].

To deal with error-tolerant problems, some truncated adders/multipliers have been reported [11], [12] but are not able to perform well in either its speed, power, area, or accuracy. The “flagged prefixed adder” [11] 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” [12], 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.

II. TOLERANT ADDER

In this section, discussion of different methodologies for designing approximate adders and use ripple carry adders (RCAs) and carry select adders CSAs throughout our subsequent discussions in all sections of this paper. Since the Mirror adder MA [13] is one of the widely used economical implementations

of an full adder FA [14], and use it as our basis for proposing different approximations of an FA cell.

Approximation Strategies for the MA

In this section, and explain step-by-step procedures for coming up with various approximate MA cells with fewer transistors. Removal of some series connected transistors will facilitate faster charging/discharging of node capacitances. Moreover, complexity reduction by removal of transistors also aids in reducing the αC term (switched capacitance) in the dynamic power expression $P_{dynamic} = \alpha CV^2 DDf$, where

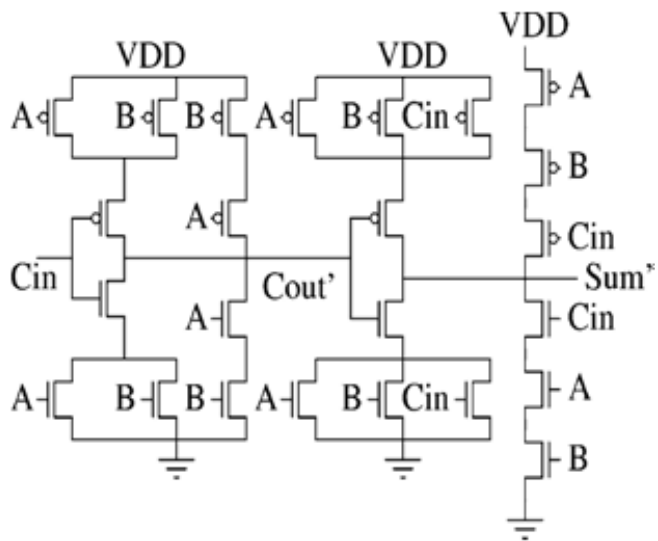


Fig. 1: Conventional Mirror Adder

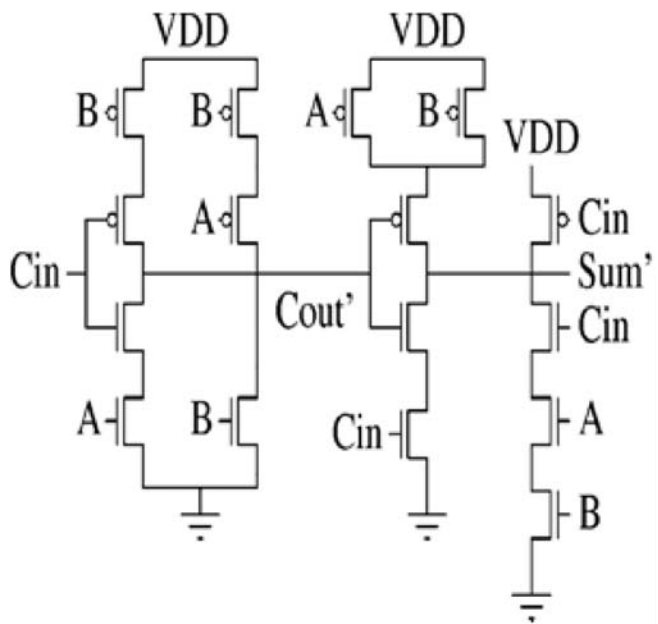


Fig. 2: Mirror Adder approximation 1

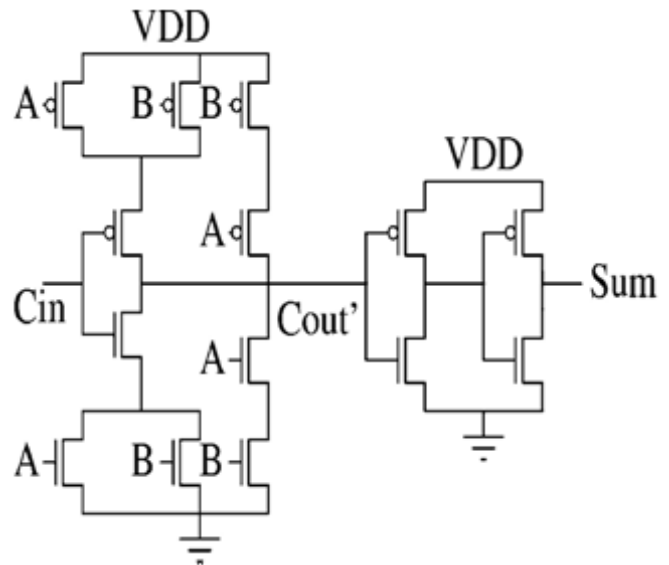


Fig. 3: Mirror Adder approximation 2

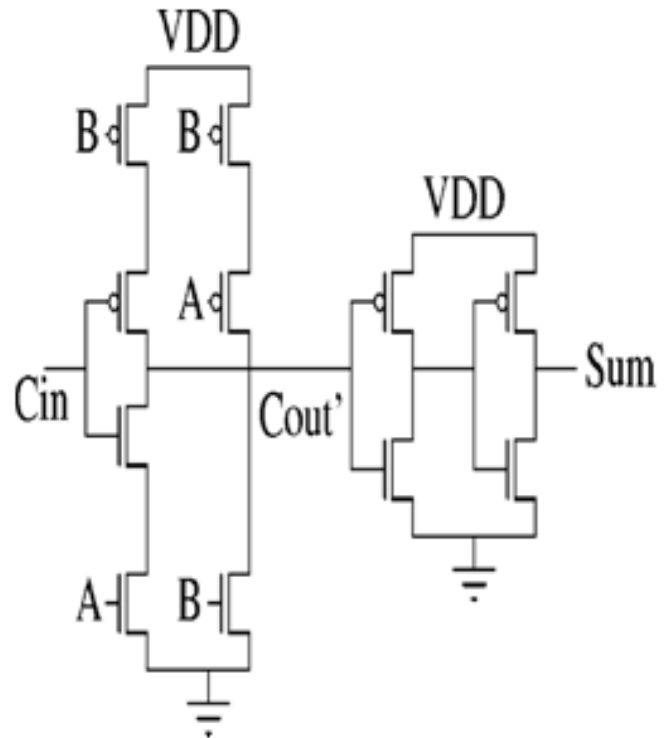


Fig. 4: Mirror Adder approximation 3

activity or average number of switching transitions per unit time and C is the load capacitance being charged/discharged. This directly results in lower power dissipation. Area reduction á is the switching is also achieved by this process. Now, let us discuss the conventional MA implementation followed by the proposed approximations

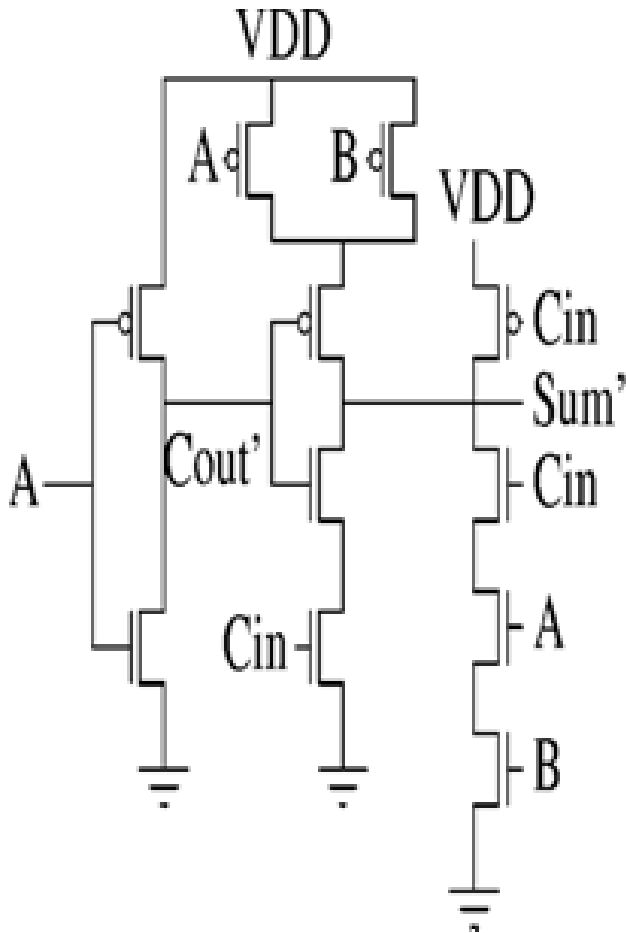


Fig. 5: Mirror Adder approximation 4

- 1) Conventional MA: Fig. 1 shows the transistor-level schematic of a conventional MA [13], which is a popular way of implementing an FA. It consists of a total of 24 transistors. Since this implementation is not based on complementary S logic, it provides a good opportunity to design an approximate version with removal of selected transistors.
- 2) Approximation 1: In order to get an approximate MA with fewer transistors, start to remove transistors from the conventional schematic one by one. However, it cannot be done this in an arbitrary fashion and thus needed to make sure that any input combination of A , B and C in does not result in short circuits or open circuits in the simplified schematic. Another important criterion is 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 Fig. 2, which is called approximation 1.

Clearly, this schematic has eight fewer transistors compared to the conventional MA schematic. In this case, there is one error in C_{out} and two errors in Sum , as shown in Table I. A tick mark denotes a match with the corresponding accurate output and a cross denotes an error.

- 3) *Approximation 2:* The truth table of an FA shows that $Sum = C_{out} + 1$ for six out of eight cases, except for the input combinations $A = 0, B = 0, C_{in} = 0$ and $A = 1, B = 1, C_{in} = 1$. Now, in the conventional MA, C_{out} is computed in the first stage. Thus, an easy way to get a simplified schematic is to set $Sum = C_{out}$. However, it introduces a buffer stage after C_{out} in Fig. 3 to implement the same functionality. The reason for this can be explained as follows. If $Sum = C_{out}$ as it is in the conventional MA, the total capacitance at the Sum node would be a combination of four source-drain diffusion and two gate capacitances. This is a considerable increase compared to the conventional case or approximation 1. Such a design would lead to a delay penalty in cases where two or more multi-bit approximate adders are connected in series, which is very common in DSP applications. Fig. 3 shows the schematic obtained using the above approach and call this approximation 2. Here, Sum has only two errors, while C_{out} is correct for all cases, as shown in Table I.
- 4) *Approximation 3:* Further simplification can be obtained by combining approximations 1 and 2. Note that this introduces one error in C_{out} and three errors in Sum , as shown in Table I. The corresponding simplified schematic is shown in Fig. 4.
- 5) *Approximation 4:* A close observation of the FA truth table shows that $C_{out} = A$ for six out of eight cases. primarily, $C_{out} = B$ for six out of eight cases. Since A and B are interchangeable, consider $C_{out} = A$. Thus fourth approximation where just use an inverter with input A to calculate C_{out} and Sum is calculated similar to approximation 1. This introduces two errors in C_{out} and three errors in Sum , as shown in Table I. The corresponding simplified schematic is shown in Fig. 5. In all these approximations C_{out} is calculated by using an inverter with C_{out} as input.

Table 1: Truth Table for Conventional FA and Approximations 1-4

Input			Accurate output		Approximate output							
A	B	Cin	Sum	Cout	Sum 1	Cout 1	Sum 2	Cout 2	Sum 3	Cout 3	Sum 4	Cout 4
0	0	0	0	0	-✓	0✓	1×	0✓	1×	0✓	0✓	0✓
0	0	1	1	0	1✓	0✓	1✓	0✓	1✓	0✓	1✓	0✓
0	1	0	1	0	0×	1×	1✓	0✓	0×	1×	0×	0✓
0	1	1	0	1	0✓	1✓	0✓	1✓	0✓	1✓	1×	0×
1	0	0	1	0	0×	0✓	1✓	0✓	1✓	0✓	0×	1×
1	0	1	0	1	0✓	1✓	0✓	1✓	0✓	1✓	0✓	1✓
1	1	0	0	1	0✓	1✓	0✓	1✓	0✓	1✓	0✓	1✓
1	1	1	1	1	1✓	1✓	0×	1✓	0×	1✓	1✓	1✓

IV. RESULT

While simulating these adder using cadence tool at 180nm technology it is found an approximate result of tolerant adder for addition but the power consumption of fourth approximate adder is very low, delay is also less. This approximate adder consumes less power and fast in response. No.of transistors used in tis approximation is less than the half of the conventional adder. Comparison of power and delay response is shown in Table 2.

V. IMAGE COMPRESSION USING TOLERANT ADDER

The DCT and inverse discrete cosine transform (IDCT) are integral components of a Joint Photographic Experts Group (JPEG) image compression system [25]. One-dimensional integer DCT $y(k)$ for an eight-point sequence $x(i)$ is given by [15]

$$y(k) = \sum_{i=0}^7 \alpha(k, i) x(i), k = 0, 1, \dots, 7.$$

Here, $a(k, i)$ are cosine functions converted into equivalent integers [8]. The integer outputs $y(k)$ can

then be right shifted to get the actual DCT outputs. A similar expression can be found for 1-D integer IDCT [9]. Thus alter the integer coefficients $a(k, i), k = 1, \dots, 7$ so that the multiplication $a(k, i)x(i)$ is converted to two left shifts and an addition (using an RCA). Since $a(0, i)$ corresponds to the dc coefficient, which is most important, and leave it unaltered. The multiplication $a(0, i)x(i)$ then corresponds to an addition of four terms. This is done using a carry-save tree using a 4:2 compressor followed by an RCA. Also, each integer DCT and IDCT output is the is the sum of eight terms. Thus, these outputs are calculated using a carry-save tree using an 8:2 compressor followed by an RCA. Thus, the whole DCT-IDCT system now consists of RCAs and CSAs. In our design, all RCAs and CSAs are approximate, which use the approximate FA cells proposed earlier. Three cases were considered, where use of approximate FA cells for 7-9 LSBs. FA cells corresponding to other bits in each case are accurate. According to our adders everywhere in DCT and IDCT is considered to be the base case.

- 1) *Output Quality*: The measure of the output quality of the decoded image after IDCT using the well-known metric of peak signal-to-noise ratio

Table 2: Table for power and delay of Conventional FA and Approximations 1-4

	Conventional	Approximation 1	Approximation 2	Approximation 3	Approximation 4
Power	45.786 pw	30.055 μw	28.335μw	33.60μw	25μw
Delay	5.75isec	7.8 isec	4.25 isec	4.98 isec	3.75 isec
No.of Transistor	24	16	14	11	11

(PSNR). The output PSNR for the base case is 31.16 Db Fig. 6 shows the output images for the base case, truncation, and approximation 5 and can be seen severe blockiness in the output images using truncation. This suggests that truncation is a bad idea when more LSBs are approximate. Fig. 1 shows the output quality for truncation and different approximations when 7–9 LSBs are approximated. Truncation leads to an appreciable decrease in PSNR for all cases. On the other hand, using approximate FAs in the LSBs can make up for the lost quality to a large extent, and also provide substantial power savings.

V. CONCLUSION

In this paper, it is proposed several imprecise or approximate adders that can be effectively utilized to trade off power and quality for error-resilient DSP systems. Our approach aimed to simplify the complexity of a conventional MA cell by reducing the number of transistors and also the load capacitances. When the errors introduced by these approximations were reflected at a high level in a typical DSP algorithm, the impact on output quality was very little. Note that our approach differed from previous approaches where errors were introduced due to VOS [3]–[10]. A decrease in the number of series connected transistors helped in reducing the bits in each case are accurate. According to our experiments, using approximate FA cells beyond

effective switched capacitance and achieving voltage scaling and also derived simplified mathematical models for error and power consumption of an approximate RCA using the approximate FA cells. Using these models, it is discussed how to apply these approximations to achieve maximum power savings subject to a given quality constraint. This procedure has been illustrated for two examples, DCT and FIR filter. It is believed that the proposed approximate adders can be used on top of already existing low-power techniques like SDC and ANT to extract multifold benefits with a very minimal loss in output quality.

REFERENCES

- [1]. A. B. Melvin, "Let's think analog," in *Proc. IEEE Comput. Soc. Annu. Symp. VLSI*, 2005, pp. 2–5.
- [2]. International Technology Roadmap for Semiconductors, latest edition available online at <http://public.itrs.net/>.
- [3]. A. B. Melvin and Z. Haiyang, "Error-tolerance and multimedia," in *Proc. 2006 Int. Conf. Intell. Inf. Hiding and Multimedia Signal Process.*, 2006, pp. 521–524.
- [4]. M. A. Breuer, S. K. Gupta, and T. M. Mak, "Design and error-tolerance in the presence of massive numbers of defects," *IEEE Des. Test Comput.*, vol. 24, no. 3, pp. 216–227, May-Jun. 2004.
- [5]. M. A. Breuer, "Intelligible test techniques to support error-tolerance," in *Proc. Asian Test Symp.*, Nov. 2004, pp. 386–393.

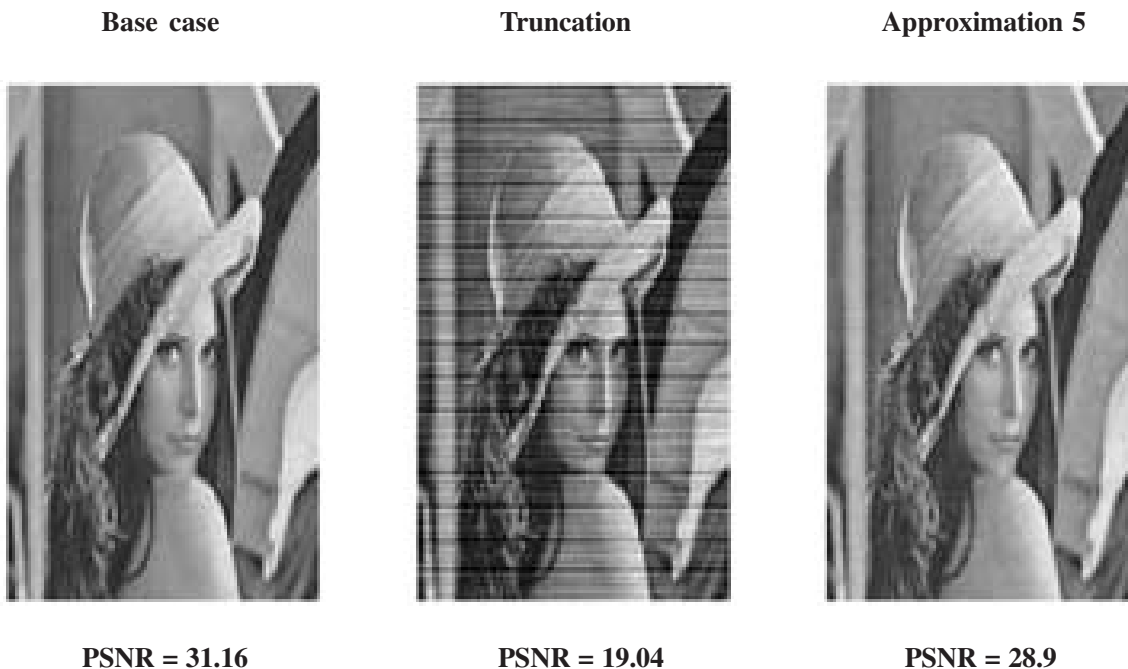


Fig. 6: Output quality when 8 LSBs are approximated

- [6]. K. J. Lee, T. Y. Hsieh, and M. A. Breuer, "A novel testing methodology based on error-rate to support error-tolerance," in *Proc. Int. Test Conf.*, 2005, pp. 1136–1144.
- [7]. I. S. Chong and A. Ortega, "Hardware testing for error tolerant multimedia compression based on linear transforms," in *Proc. Defect and Fault Tolerance in VLSI Syst. Symp.*, 2005, pp. 523–531.
- [8]. H. Chung and A. Ortega, "Analysis and testing for error tolerant motion estimation," in *Proc. Defect and Fault Tolerance in VLSI Syst. Symp.*, 2005, pp. 514–522.
- [9]. H. H. Kuok, "Audio recording apparatus using an imperfect memory circuit," U.S. Patent 5 414 758, May 9, 1995. (2002) The IEEE website. [Online]. Available: <http://www.ieee.org/>
- [10]. T. Y. Hsieh, K. J. Lee, and M. A. Breuer, "Reduction of detected acceptable faults for yield improvement via error-tolerance," in *Proc. Des., Automation and Test Eur. Conf. Exhib.*, 2007, pp. 1
- [11]. D. Shin and S. K. Gupta, "Approximate logic synthesis for error tolerant applications," in *Proc. Design, Automat. Test Eur.*, 2010, pp. 957–960.
- [12]. 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
- [13]. J. M. Rabaey, *Digital Integrated Circuits: A Design Perspective*. Upper Saddle River, NJ: Prentice-Hall, 1996/
- [14]. Lyons, V. Ganti, R. Goldman, V. Melikyan, and H. Mahmoodi, "Full-custom design project for digital VLSI and IC design courses using synopsys generic 90nm CMOS library," in *Proc. IEEE Int. Conf. Microelectron. Syst. Edu.*, Jul. 2009, pp. 45–48.
- [15]. K. K. Parhi, *VLSI Digital Signal Processing Systems: Design and Implementation*. New York: Wiley, 1999.

□