Design of High Performance Adder with Tolerable Error and Acceptable Output and It's Application in Image Processing

L.V.Sivarao [1] , S.Vidyarani [2]

[1]*Department of ECE, Narasaraopet Engineering College, Narasaraopet, Ap, India.*

[2]*Assistant.proffesor Department of ECE, Narasaraopet Engineering College, Narasaraopet, Ap, India.*

Abstract— **The design of adder is the most focused area in VLSI design of processing units. So far there are a variety of such adders like RCA, CSA, CLA and ETA. ETA is the Error Tolerant Adder and is the latest of the adders which has better performance when compared with the other adders in terms of power consumption, delay etc.**

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.

KEYWORDS: Adders, digital signal processing (DSP), error tolerance, high-speed integrated circuits, low-power design, VLSI.

I. INTRODUCTION

In conventional digital VLSI design, one usually assumes that a us-able circuit/system should always provide definite and accurate results. But in fact, such perfect operations are seldom needed in our nondigital worldly experiences. 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 er-rors. 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 im-portant in today's digital IC design [2].

Based on the characteristic of digital VLSI design, some novel con-cepts and design techniques have been proposed.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 [3]. The "imperfect" attribute seems to be not appealing.To deal with error-tolerant problems, some truncated adders/multi-pliers have been reported [4], [5] but are not able to perform well in either its speed, power, area, or accuracy. The "flagged prefixed adder" performs better than the nonflagged 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 con-cept. 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) sys-tems that process signals relating to human senses such as hearing, sight, smell, and touch, e.g., the image processing and speech processing sys-tems, the error-tolerant circuits may be applicable [3], [6], [7].

II. ERROR-TOLERANT ADDER

Before detailing the ETA, the definitions of some commonly used terminologies shown in this paper are given as follows.

- *Overall error (OE):OE=*$\|Rc - \mathrm{Re}\|$, where Re is the result obtained by the adder, and Rc denotes 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 adder is used to indicate how "correct" the output of an adder is for a particular input.It's defined as: $Acc = (1-(OE/Rc))\times 100\%$. Its value ranges from 0% to 100%.
- *Minimum acceptable accuracy (MAA):* Although some errors are allowed to exist at the output of an

ETA, the accuracy of an ac-ceptable output should be "high enough" (higher than a threshold value) to meet the requirement of the whole system. Minimum ac-ceptable accuracy is just that threshold value. The result obtained whose accuracy is higher than the minimum acceptable accuracy is called acceptable result.

- *Acceptance probability (AP):* Acceptance probability is the prob-ability that the accuracy of an adder is higher than the minimum acceptable accuracy. It can be expressed as $AP = P(ACC > MAA)$, with its value ranging from 0 to 1.

A. Need for Error-Tolerant Adder

Increasingly huge data sets and the need for instant response re-quire 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) [6], carry-select adder (CSL) [7], and carry-look-ahead adder (CLA) [18], have been developed. Also, there are many low-power adder design techniques that have been proposed [9]. 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.

B. Proposed Addition Arithmetic

In a conventional adder circuit, the delay is mainly attributed to the carry propagation chain along the critical path, from the least signif-icant bit (LSB) to the most significant bit (MSB). Meanwhile, a sig-nificant proportion of the power consumption of an adder is due to the glitches that are caused by the carry propagation. Therefore, if the carry propagation can be eliminated or curtailed, a great improvement in speed performance and power consumption can be achieved. In this paper, we propose for the first time, an innovative and novel addition arithmetic that can attain great saving in speed and power consumption. This new addition arithmetic can be illustrated via an example shown in Fig. 1.

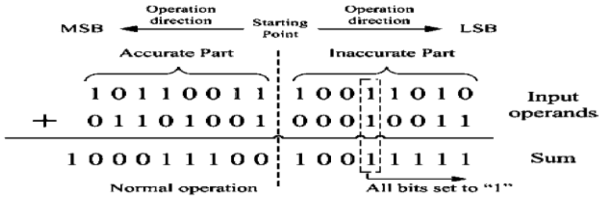


Fig. 1. Proposed addition arithmetic.

We first split the input operands into two parts: an accurate part that includes several higher order bits and the inaccurate part that is made up of the remaining lower order bits. The length of each part need not necessary be equal. The addition process starts from the middle (joining point of the two parts) toward the two opposite directions simultaneously. In the example of Fig.1, the two 16-bit input operands A="1011001110011010" (45978) and B="0110100100010011" (26899), are divided equally into 8 bits each for the accurate and inaccurate parts.

The addition of the higher order bits (accurate part) of the input operands is performed from right to left (LSB to MSB) and normal addition method is applied. This is to preserve its correctness since the higher order bits play a more important role than the lower order bits. The lower order bits of the input operands (inaccurate part) require a special addition mechanism. No carry signal will be generated or taken in at any bit position to eliminate the carry propagation path.To minimize the overall error due to the elimination of the carry chain,a special strategy is adapted, and can be described as follow: 1) check every bit position from left to right (MSB to LSB); 2) if both input bits are "0" or different, normal one-bit addition is performed and the operation proceeds to next bit position; 3) if both input bits are "1,"the checking process stopped and from this bit onward, all sum bits to the right are set to "1." The addition mechanism described can be easily understood from the example given in Fig. 1 with a final result of "10001110010011111" (72863).

The example given in Fig. 1 should actually yield "10001110010101101" (72877) if normal arithmetic has been applied. The overall error generated can be computed as OE=72877-72863=14. The accuracy of the adder with respect to these two input operands is ACC=(1-(14/72877))X100%.=99.98%. .By eliminating the carry propagation path in the inaccurate part and performing the addition in two separate parts simultaneously, the overall delay time is greatly reduced, so is the power consumption.

C. Hardware Implementation

The block diagram of the hardware implementation of such an ETA that adopts our proposed addition arithmetic is provided in Fig. 2. This most straight forward structure consists of two parts: an accurate part and an inaccurate part. The accurate part is constructed using a conventional adder such as the RCA, CSK, CSL, or CLA. The carry-in of this adder is connected to ground. The inaccurate part constitutes two blocks: a carry-free addition block and a control block. The control block is used to generate the control signals, to determine the working mode of the carry-free addition block. In Section III, a 32-bit adder is used as an example for our illustration of the design methodology and circuit implementation of an ETA.

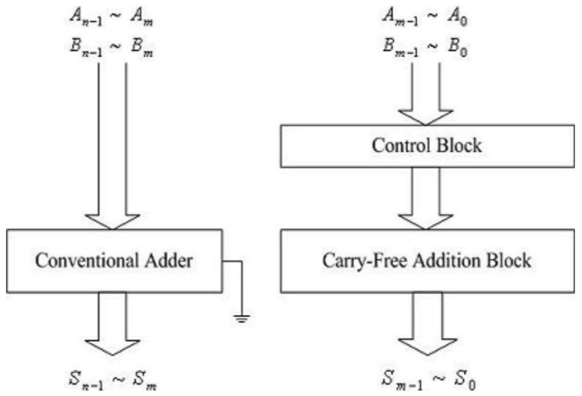


Fig. 2. Hardware implementation of the proposed ETA.

III. DESIGN OF A 32-BIT ERROR TOLERANT ADDER

A. Strategy of Dividing the Adder

The first step of designing a proposed ETA is to divide the adder into two parts in a specific manner. The dividing strategy is based on a guess-and-verify stratagem, depending on the requirements, such as accuracy, speed, and power. First, we define the delay of the proposed adder as, where is the delay in the accurate part and is the delay in the inaccurate part. With the proper dividing strategy, we can make approximately equal to and hence achieve an optimal time delay.

With this partition method defined, we then check whether the ac-curacy performance of the adder meets the requirements preset by de-signer/customer. This can be checked very quickly via some software programs. For example, for a specific application, we require the min-imum acceptable accuracy to be 95% and the acceptance probability to be 98%. The proposed partition method must therefore have at least 98% of all possible inputs reaching an accuracy of better than 95%. If this requirement is not met, then one bit should be shifted from the inaccurate part to the accurate part and have the checking process repeated.

Also, due to the simplified circuit structure and the elimination of switching activities in the inaccurate part, putting more bits in this part yields more power saving. By considering the above, we divided the 32-bit adder by putting 12 bits in the accurate part and 20 bits in the inaccurate part.

B. Design of the Accurate Part

In our proposed 32-bit ETA, the inaccurate part has 20 bits as op-posed to the 12 bits used in the accurate part. The overall delay is determined by the inaccurate part, and so the accurate part need not be a fast adder. The ripple-carry adder, which is the most power-saving conventional adder, has been chosen for the accurate part of the circuit.

C. Design of the Inaccurate Part

The inaccurate part is the most critical section in the proposed ETA as it determines the accuracy, speed performance, and power consumption of the adder. The inaccurate part consists of two blocks: the carry-free addition block and the control block. The carry-free addition block is made up of 20 modified XOR gates, and each of which is used to generate a sum bit. The block diagram of the carry-free addition block and the schematic implementation of the modified XOR gate are presented in Fig. 3. In the modified XOR gate, three extra transistors, M1, M2, and M3, are added to a conventional XOR gate. CTL is the control signal coming from the control block of Fig. 4 and is used to set the operational mode of the circuit. When, M1 and M2 are turned on, while M3 is turned off, leaving the circuit to operate in the normal XOR mode. When, M1 and M2 are both turned off, while M3 is turned on, connecting the output node to VDD, and hence setting the sum output to “1.”

The function of the control block is to detect the first bit position when both input bits are “1,” and to set the control signal on this position as well as those on its right to high. It is made up of 20 control signal generating cells (CSGCs) and each cell generates a control signal for the modified XOR gate at the corresponding bit position in the carry-free addition block. Instead of a long chain of 20 cascaded GSGCs, the control block is arranged into five equal-sized groups, with additional connections between every two neighboring groups. Two types of CSGC, labeled as type I and II in Fig. 4(a), are designed, and the schematic implementations of these two types of CSGC are pro-vided in Fig. 4(b). The control signal generated by the leftmost cell of each group is connected to the input of the leftmost cell in next group. The extra connections allow the propagated high control signal to “jump” from one group to another instead of passing through all the 20 cells. Hence, the worst case propagation path [shaded in gray in Fig. 4(a)] consists of only ten cells.

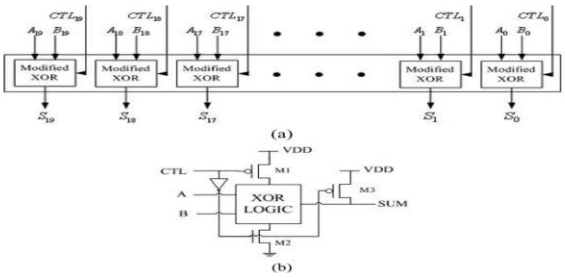
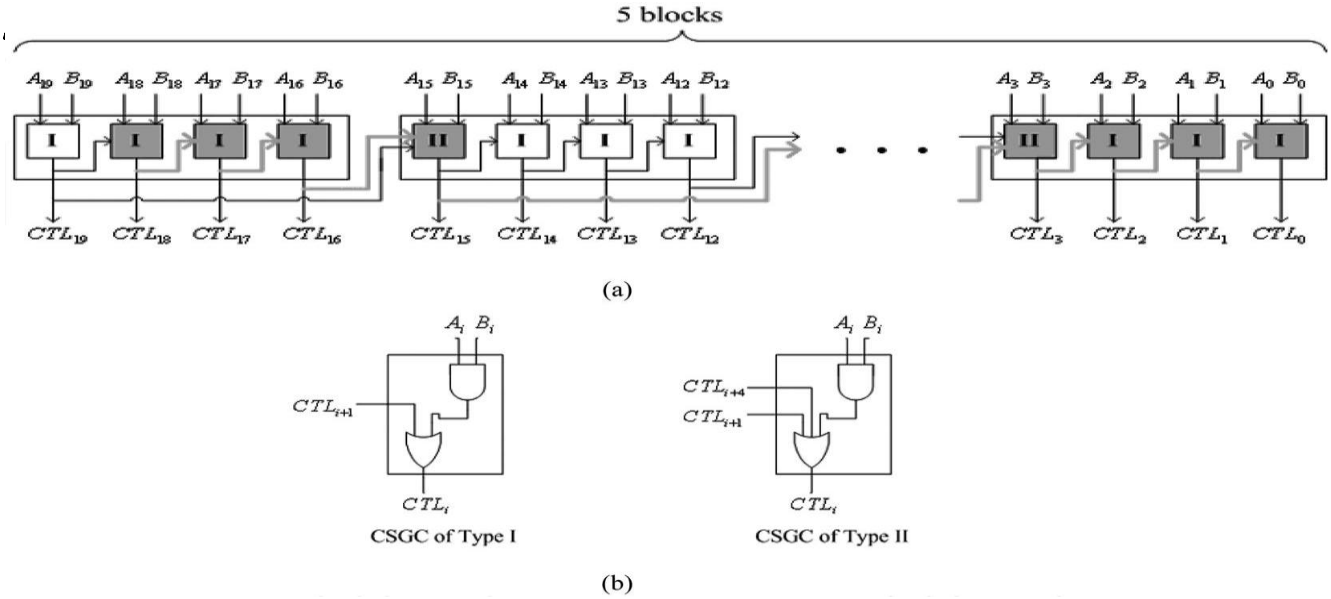


Fig. 3. Carry-free addition block. (a) Overall architecture and (b) schematic diagram of a modified XOR gate


5 blocks
A_{19} B_{19} A_{18} B_{18} A_{17} B_{17} A_{16} B_{16} A_{15} B_{15} A_{14} B_{14} A_{13} B_{13} A_{12} B_{12} A_3 B_3 A_2 B_2 A_1 B_1 A_0 B_0
I
II
CTL_{19} CTL_{18} CTL_{17} CTL_{16} CTL_{15} CTL_{14} CTL_{13} CTL_{12} CTL_3 CTL_2 CTL_1 CTL_0
A_i B_i
CTL_{i+1}
CTL_i
CTL_{i+4}

(a)

CSGC of Type I

CSGC of Type II

(b)

Fig. 4. Control block. (a) Overall architecture and (b) schematic implementations of CSGC

IV. SIMULATION RESULTS

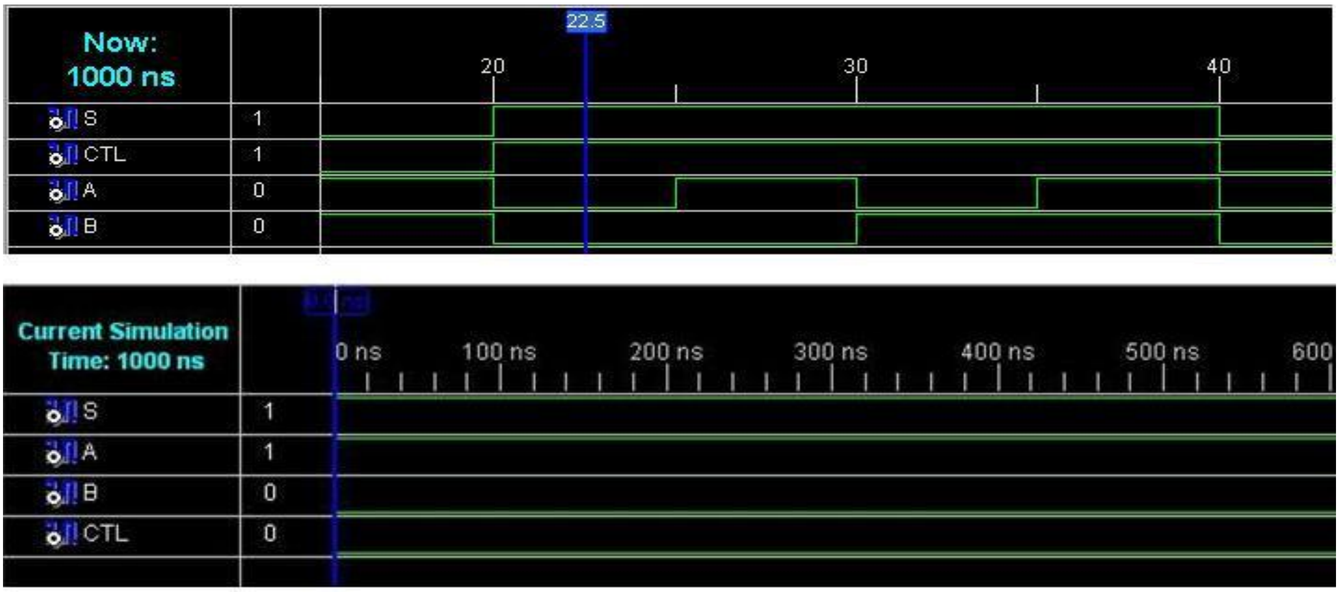


Fig.5. Simulation Waveforms of Modified X-OR

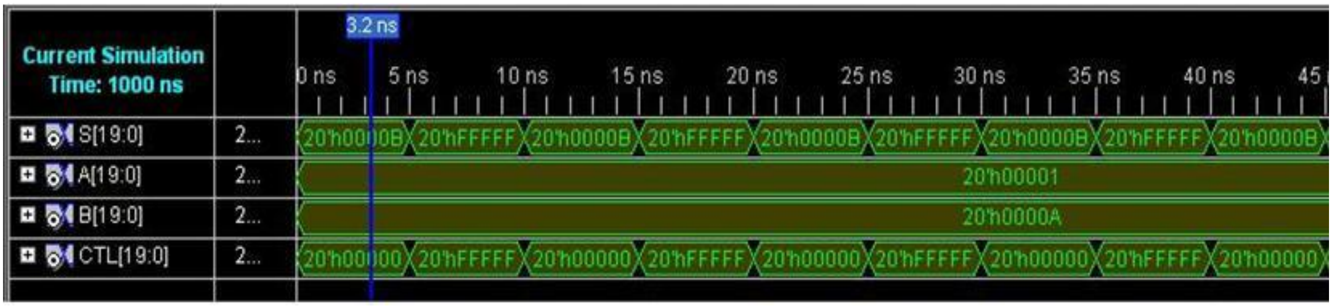


Fig.6. Simulation Waveforms of Modified Carry Free Adder

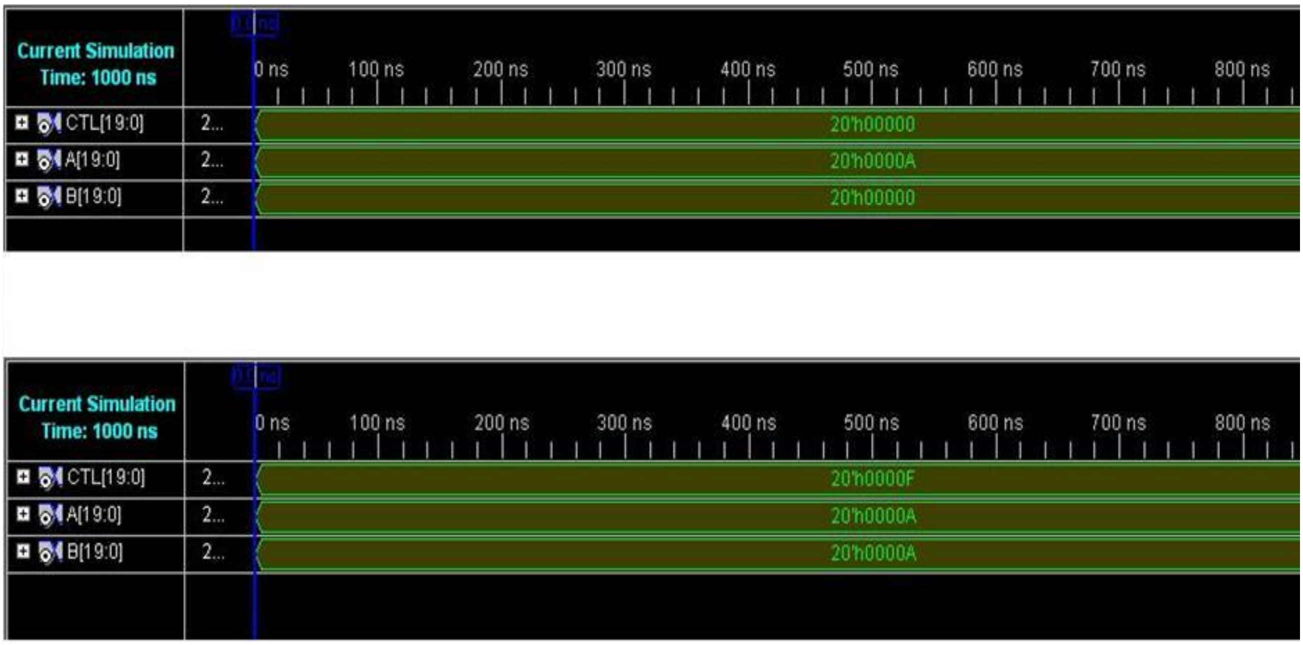


Fig.7.Simulation Waveforms of Control Block

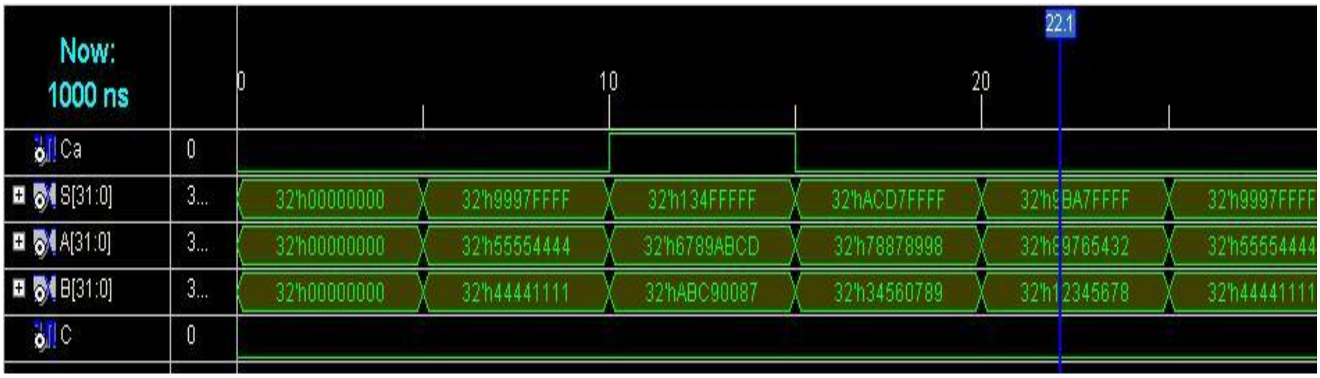


Fig.8.Simulation Waveforms of Error Tolerant Adder

V. APPLICATION OF ERROR-TOLERANT ADDER IN DIGITAL SIGNAL PROCESSING

In image processing and many other DSP applications, fast Fourier transformation (FFT) is a very important function. The computational process of FFT involves a large number of additions and multiplications. It is therefore a good platform for embedding our proposed ETA.To prove the feasibility of the ETA, we replaced all the common additions involved in a normal FFT algorithm with our proposed addition arithmetic. As we all know, a digital image is represented by a matrix in a DSP system, and each element of the matrix represents the color of one pixel of the image. To compare the quality of images processed by both the conventional FFT and the inaccurate FFT that had incorporated our proposed ETA, we devised the following experiment. An image was first translated to a matrix form and sent through a standard system that made used of normal FFT and normal reverse FFT. The matrix output of this system was then transformed back to an image and presented in Fig. 9(a). The matrix of the same image was also processed in a system that used the inaccurate FFT and inaccurate reverse FFT, where both FFTs had incorporated the 32-bit ETA described in Section III, with the processed image given in Fig.9 (b).

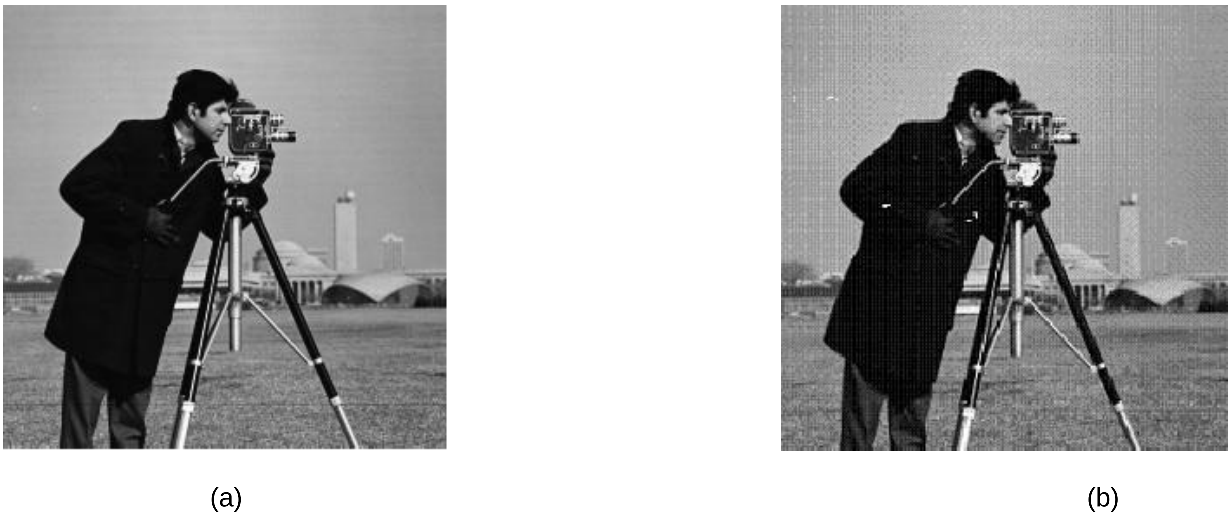


Fig. 9. Images after FFT and inverse FFT. (a) Image processed with conventional adder and (b) image processed with the proposed ETA.

Although the two resultant matrices of the same image were different, the two pictures obtained (see Fig. 9) look almost the same. Fig. 9(b) is slightly darker and contains horizontal bands of different shades of gray. With a MAA setting of 95%, the AP of the matrix representation of Fig. 9(b) is 98.3% as compared to the matrix representation of Fig. 9(a).The comparison between the two images in Fig. 9 shows that the quality loss to the image using our proposed ETA is negligible and can be completely tolerated by human eyes. These simulation results have proven the practicability of the ETA proposed in this paper.

VI. CONCLUSION

In this paper, the concept of error tolerance is introduced in VLSI design.A novel type of adder, the error-tolerant adder, which trades certain amount of accuracy for significant power saving and performance improvement, is proposed. Extensive comparisons with conventional digital adders showed that the proposed ETA outperformed the conventional adders in both power consumption and speed performance. The potential applications of the ETA fall mainly in areas where there is no strict requirement on accuracy or where superlow power consumption and high-speed performance are more important than accuracy. One example of such applications is in the DSP application for portable devices such as cell phones and laptops.

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 [Online].

[3] A. B. Melvin and Z. Haiyang, "Error-tolerance and multi-media," 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.

[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.