Design of Carry Free Error Tolerant Adder Using Verilog HDL

B.Rama Krishna
PG-Scholar, VLSI,
Department of ECE,
Krishnamurthy Institute of
Technology and Engineering.

CH.Bhavani, M.Tech
Assistant Professor,
Department of ECE,
Krishnamurthy Institute of
Technology and Engineering.

T.Ravichandra Babu, Ph.D
Associate Professor & HOD,
Department of ECE,
Krishnamurthy Institute of
Technology and Engineering.

Abstract:

In the era of low power, high performance digital systems are needed to boost up the technology revolution in nano-electronics. Realization of new digital logic is essential for making revolutionary changes in low power and high speed performance. In modern VLSI technology, the occurrence of all kinds of errors has become inevitable. By adopting an emerging concept in VLSI design and test, error tolerance (ET), a novel error-tolerant adder (ETA) is proposed. The ETA is able to ease the strict restriction on accuracy, and at the same time achieve tremendous improvements in both the power consumption and speed performance. A robust and efficient error-tolerant adder (ETA) is proposed in this project and it is compared with its conventional counterparts, with respect to power consumption and high speed.

Index Terms:

Error tolerant Adder, low-power and high speed circuit, VLSI.

I. INTRODUCTION:

In conventional digital VLSI design, most digital data are processed and transmitted in a noisy channel. During this process, errors may occur anywhere in the processing chain. Error-tolerant concept cannot be adopted for all systems – In digital control systems, the correctness of the output signal is extremely important, and this denies the use of the error tolerant circuit.

In modern VLSI technology, the occurrence of all kinds of errors has become inevitable. By adopting an emerging concept in VLSI design and test, error tolerance (ET), a novel error-tolerant adder (ETA) is proposed. The ETA is able to ease the strict restriction on accuracy, and at the same time achieve tremendous improvements in both the power consumption and speed performance. A robust and efficient error-tolerant adder (ETA) is proposed in this project and it is compared with its conventional counterparts, with respect to power consumption and speed. The proposed ETA is then applied for a typical signal processing application (image processing and speech processing systems, the error-tolerant circuits may be applicable) and tested for its performance in terms of error tolerance.

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 – But all of them suffer a trade-off between speed and power.ETA can attain great improvement in both the power consumption and speed performance. Implementation of FPGA based error tolerant adder is done by two parts. First part is that PC communication with FPGA board through RS 232 cable.

Second one is to design the low power high speed adder and apply and check image processing application. The first part has three separate blocks. Clock divider, Transmitting section and another one is receiving section. clockdivider is work as an incoming clock frequency 20 MHZ is divided by a factor 1042and again divide by two to achieve 9600 baud is used for both transmission and reception of pc & FPGA. For that receiving section the received data from PC is 9600 baud rate. The received data is used for image processing application using low power high speed adder. Reception is done by accumulating one bit at a time than shifting left. Next, the transmitting blocks performing the reverse operation of receiving block. Each bit is separated from data output and shifted right to achieve serial transmission. This process is repeated eight times.

II.LOW POWER HIGH SPEED ERRORTOLERANT ADDER

A. Terminology

The ETA, the definitions of some commonly used terminologies are given as follows.

Overall error (OE): OE = | Rc – Re |

Where Re is the result obtained by the adder, and Rc denotes the correct result. Accuracy (ACC): The accuracy of an adder is used to indicate how "correct" the output of an adder is for a particular input. Minimum acceptable accuracy (MAA):The accuracy of an acceptable output should be high enough to meet the requirement of the whole system.MAA is just that threshold value. Acceptance probability (AP): Acceptance probability is the probability that the accuracy of an adder is higher than the minimum acceptable accuracy.

B. Reason for Delay & Power Consumption in Digital Arithmetic Circuits

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). Significant 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.

C. Proposed Error Tolerant Adder Addition in Arithmetic

Step -1: 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 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.

Step – 2: Addition Process for Accurate part 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.

Step-3: Addition Process for In Accurate part 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 a final result of “10001110010011111” (72863).The example given in should actually yield“10001110010101101” (72877) if normal arithmetic has been applied. The overall error generated can be computed as OE=14. The block diagram of the ETA Figure.1.that adopts our proposed addition arithmetic is provided in Figure. This most straightforward 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.

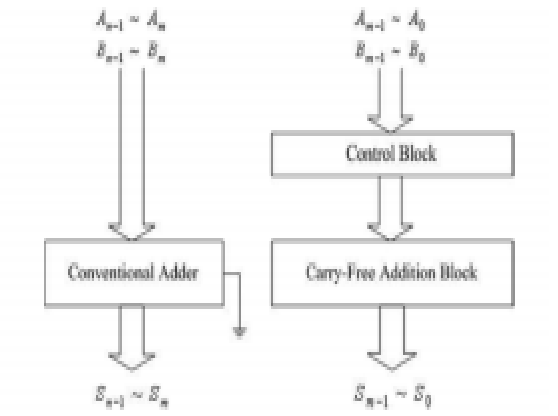


Figure.1: low power high speed error tolerant adder

D. Design of 32-Bit Low Power High Speed Error Tolerant Adder

Fault tolerance (FT) is the ability of a system to continue correct operation of its tasks after the occurrence of hardware of software faults. Defect tolerance (DT) the term refers to any circuit implementation that tolerates more defects than a non-defect-tolerant implementation. Recently the “undefined” term error-tolerance has appeared in several documents. In some cases, the term seems to refer to the fact that hardware has defects, and that one needs the ability to reconfigure a device so that only the “good” circuitry is used. This concept seems to be a merger of DT and reconfigurable FT ideas. We put forth the following new definition of error-tolerance. A circuit is error-tolerant (ET) with respect to an application or system, if (1) it contains defects that cause internal and may cause external errors, and (2) the system that incorporates this circuit produces acceptable results. Since an ET circuit may produce erroneous results, ET is different from FT and DT. The primary interest in ET is that it has the potential to significantly enhance the effective yield of a process.

As VLSI scaling continues along its traditional path, chips will have billions of devises and thousands of defects. Using only FT and DT techniques might not be sufficient to produce functionally perfect chips at affordable prices. The case is clearly made in the 2001 International Technology Roadmap for Semiconductors (ITRS). One challenge they discuss is error-tolerance. Clearly, some applications require correct computation, such as banking and the control of a nuclear power plant. Many others do not. One large domain where perfect functional operation is not required is in some multi-media and many DSP applications. One good example of an error tolerant system is a MPEG motion detection encoder that provides excellent performance in the presence of a large fraction of single stuck-at faults.

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 time delay. With this partition method defined, we then check whether the accuracy performance of the adder meets the requirements preset by designer/customer. This can be checked very quickly via some software programs. 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.

Having considered the above, we divided the 32-bit adder by putting 12 bits in the accurate part and 20 bits in the inaccurate part. In our proposed 32-bit ETA, the inaccurate part has 20 bits as opposed 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.

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. 2

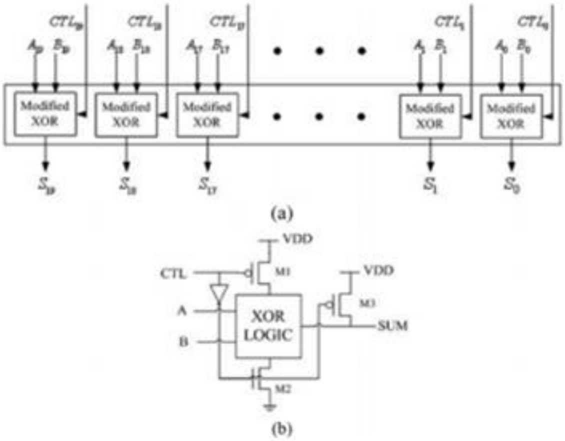


Fig. 2.Carry-free addition block.(a) Overall architecture and (b) schematic diagram of a modified XOR gate.

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.3 and is used to set the operational mode of the circuit. When CTL=0, M1 and M2 are turned on, while M3 is turned off, leaving the circuit to operate in the normal XOR mode.

When CTL=1, 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 thecarry-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 neighbouring groups.

Two types of CSGC, labelled as type I and II in Fig. 3(a), are designed, and the schematic implementations of these two types of CSGC are provided in Fig. 3(b). The control signal generated by the left most 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. 3(a)) consists of only ten cells.

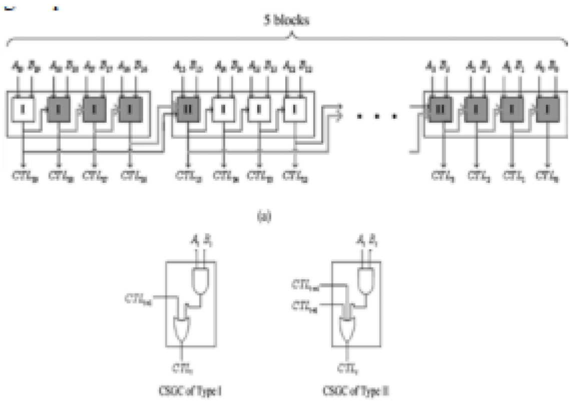


Fig. 3. Control block. (a) Overall architecture and (b) schematic implementations of CSGC.

III. SIMULATION RESULTS:

All the synthesis and simulation results are performed using Verilog HDL. The synthesis and simulation are performed on Xilinx ISE 14.4. The simulation results are shown below figures.

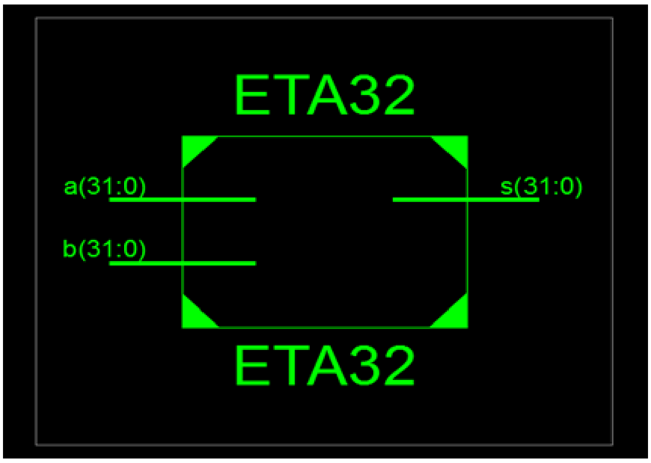


Fig 4.RTL schematic of32bit error-tolerant adder (ETA)

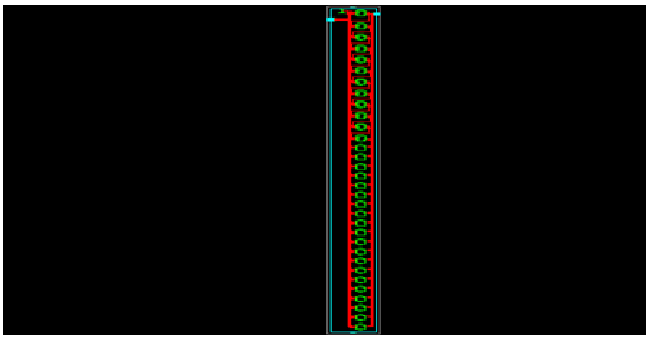


Fig 5.RTL sub schematic of32bit error-tolerant adder (ETA)

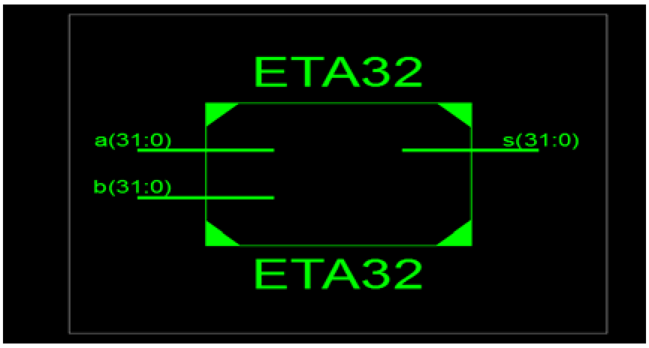


Fig 6.Technology schematic of32bit error-tolerant adder (ETA)

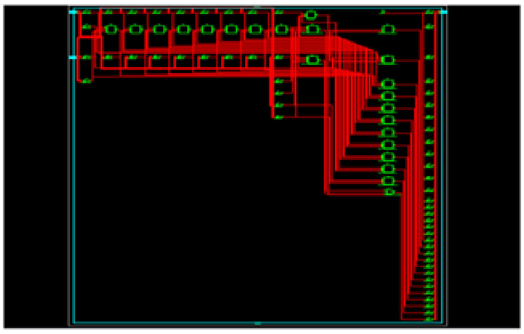
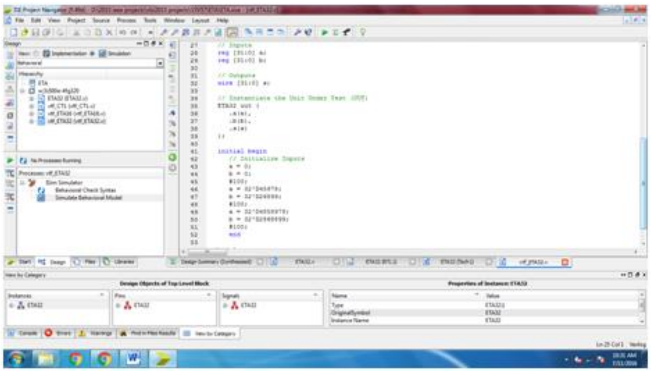


Fig 7.Sub Technology schematic of32bit error-tolerant adder (ETA)



```
// Inputs
reg [31:0] a;
reg [31:0] b;

// Outputs
wire [31:0] s;

// Instantiate the Unit Under Test (UUT)
ETA32 uut (
    .a(a),
    .b(b),
    .s(s)
);

initial begin
    // Initialize Inputs
    a = 0;
    b = 0;
    #100;
    a = 32'D[illegible];
    b = 32'D[illegible];
    #100;
    a = 32'D[illegible];
    b = 32'D[illegible];
    #100;
    end
```

Fig 8.Test Bench of32bit error-tolerant adder (ETA)

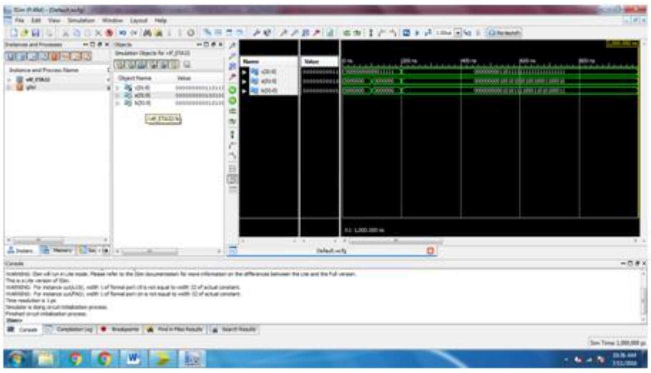


Fig 9.Simulation of 32bit error-tolerant adder (ETA)

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 super low 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.

V.REFERENCES

[1]. M. A. Breuer, “Intelligible test techniques to support error-tolerance,”in Proc. Asian Test Symp., Nov. 2004, pp. 386–393.

[2]. A. B. Melvin and Z. Haiyang, “Error-tolerance and multi-media,”in Proc. 2006 Int. Conf. Intell. Inf. Hiding and Multimedia SignalProcess., 2006, pp. 521–524.

[3]. M. A. Breuer, S. K. Gupta, and T. M. Mak, “Design and error-tolerance in the presence of massive numbers of defects,” IEEE Des. TestComput., vol. 24, no. 3, pp. 216–227, May-Jun. 2004.

[4]. Ning Zhu path, Wang Ling Goh, Weija Zhang, KiatSeng Yeo, and ZhiHuiKong,”Design of Low-Power High-Speed Truncation-ErrorTolerant Adder and Its Application in Digital Signal Processing”.

[5]. K. V. Palem, “Energy aware computing through probabilistic switching: A study of limits,” IEEE Trans. Comput., vol. 54, no. 9, pp.1123– 1137, Sep. 2005.

[6].J. E. Stine, C. R. Babb, and V. B. Dave, “Constant addition utilizing flagged prefix structures,” in Proc. IEEE Int. Symp. Circuits and Systems (ISCAS), 2005.

[7].L.-D. Van and C.-C.Yang, “Generalized lowerror area-efficient fixed width multipliers,” IEEE Trans. Circuits Syst.I, Reg.