Mux Based Novel Counter Design for Low Power Application

Divya.G
Vengatesh Kumar S ME..(p.hD)
Mohamed Sathak Engineering College

Abstract
Parallel counters are key elements in many arithmetic circuits,especially fast multipliers.a new binary counter design is proposed.new binary counter design is proposed It uses mux based full adder circuit, which group all of the "1" bits together In the proposed structure, one XOR block in the conventional full adder is replaced by a multiplexer block so that the critical path delay is minimized . Counter-based mechanisms have been proposed for usein built-in test set embedding. A single counter or multiple countersmay be used with one or multiple seeds. In addition, counters may becombined with ROM's. Proposed system coded in Verilog HDL and simulated using Xilinx 12.

1INTRODUCTION

An (n, m) parallel counter is a circuit which provides an m-bit count of the number of the n-inputs that are logic ONES. A counter differs from a compressor in that compressors have carry inputs and carry outputs in addition to the "normal" inputs and outputs, while counters do not have carry inputs and outputs. The most widely used parallel counters are full adders which are (3:2) counters and half adders which are (2:2) counters. Larger parallel counters are especially useful in the implementation of widely used signal processing elements such as multipliers, convolves, etc. The speed of multipliers is a critical issue in determining the performance of microprocessors The rest of this brief introduces the counters in section I. Next the related works and fast binary counter using symmetrical stack are considered in Section II. Then, in Section III, the proposed is presented. Section IV presents a experimental Results and performance analysis to illustrate the effectiveness of the hardware security approach. Finally, the conclusions are summarized in Section V.

In this paper [1] column compression multiplier to design a method for column minimized according to the compressor based multipliers. So the main drawback is the higher hardware complexity. But the Energy Delay Product (EDP) is slightly higher than then lower order compressors. Wallace Tree multiplier using high speed counter to be designed for the counter methodology. This compressor uses counter property. So that the output of compressor gives number of 1's at input. So the main drawback for delay overhead. [2].

In the novel architectures followed by the wallance Tree multiplier circuit. Thus the efficient (m,n) parallel counters are the basic mux counter then the mux based approach adds additional of the complex [3]. Feed forward and feedback interconnections are derived by the methods used in the passivity for the interconnected in the symmetrically distributed system. Thus the feed forward methods to be held at the routing complexity [4]. In the high speed multiplication of 8-4 and 9-4 compressors using the higher order of the compressor can be effectively used for high speed multiplications. Thus the multiplication is a fundamental operation in most of the signal processing algorithm [5]. In this paper followed by the high speed operations of the full adder based counter. Thus the counter to be analyzed by the architecture of the Wallace multipliers. So the main drawback for this paper using high power consumption [6]. In the fast arithmetic circuits using low power, low voltage (5:2) compressors cell. Then the compressor cell is accordingly performed by low power compressors. In this regard many innovative designs for basic logic functions using pass transistors and transmission gates have appeared in the literature recently. This may also results in unreliable operations as the power supply voltage is scaled down. Hence the 16 transistor added cell (16T) given is not a good choice for low voltage operations [7]. The part of detection and grouping methods using multi-scale symmetric. Its recovering and grouping symmetric parts [8]. In this paper to design a efficient reversible multiplier using tanner EDA. Thus the method can be performed as low power designing by reversible logic gates. So it can be occupy the large input, output requirements [9].

I. COUNTER IMPLEMENTATION

In most cases, the design of (7:3) parallel counter is shown. Generally such counters are large

enough to illustrate the concepts without getting bogged down with too much detail. In some cases, counters of other sizes are shown to illustrate the method.

The threshold gate counters multipliers included a design for a (7:3) counter implemented with inverting threshold gates provides a definition of inverting threshold gates. There are n binary inputs, each with a corresponding weight. If the sum of the weights of the inputs that are logic ONES exceeds the threshold of the gate the output is a logic ZERO, otherwise the output a logic ONE. Then the realization of a (7:3) parallel counter with three inverting threshold gates. This circuit uses three threshold gates with seven, eight and nine inputs and thresholds of four, six and seven, respectively. The delay of the least significant bit of the count, y2, is three gate delays, the delay of the middle bit, y1, is two gate delays, and the delay of the most significant bit, y0, is one gate delay. This approach has not been widely used, perhaps because of the difficulty of realizing large threshold gates with accurate thresholds.

This counter based upon a switching tree implemented with relays stated to be an extension of an idea that is due to Falk off. The first counter input controls a single pole double throw relay, the second input controls a double pole double throw relay, the third input controls a triple pole triple throw relay, etc. The complexity of this approach grows as the square of the number of inputs, making the cost prohibitive for the realization of large counters.

The relay switches can be implemented with superconducting cryotrons. In VLSI, CMOS domino logic is also suitable for the realization of switching tree. Switching trees for the three bits of a (7:3) parallel counter. The realization of the switching trees with domino logic. Although traditional design rules would not allow as many levels of transistors between the V_{dd} and V_{ss} supplies as the eight levels of this circuit, simulations show that performance of the structure does not suffer.

Foster and Stockton developed a counter implemented with full and half adders. This methods (31:5) parallel counter which consists of grouping the counter inputs into [n/3] sets of three lines each which are input to full adders (where: [x] denotes the largest integer less than or equal to x). The [n/3] sum and [n/3] carry outputs of the full adders (and any unprocessed inputs to the counter) are grouped into threes and applied to second (smaller) sets of full adders, etc. The number of full adders to realize an (n, m) parallel counter is derived as less than or equal to n.

A parallel counter design was developed that combines a pair of small, (n, m) counters (usually (3:2) counters) using an m -bit parallel adder to make a (2n + 1, m + 1) counter. A pair of these (2n + 1, m + 1) counters and an m + 1-bit adder produce a (4n+3, m + 2) counter, etc. A (15:4) parallel counter constructed with methods. When the parallel adders are realized with ripple carry adders, the number of full adders to implement an (n, m) parallel counter is n–m assuming that n = 2m – 1 where m is an integer.

The delay for the least significant bit of the count is m – 1 full adder delays (assuming that the delay from any input to any output of a full adder is 1 full adder delay) and the delay for the most significant bit of the count is 2m – 3 full adder delays. It is indicated that the parallel adders can alternatively be realized with fast adders. It was suggested that read only memories could be used, but carry lookahead adders or conditional sum adders would be a more appropriate choice with current technology. If the parallel adder has delay comparable to a full adder delay, this approach gives all bits of an (n, m) parallel counter in m – 1 full adder delays.

II. METHODOLOGY

A binary counter design is proposed. It uses 3-bit stacking circuits, which group all of the "1" bits together, followed by a novel symmetric method to combine pairs of 3-bit stacks into 6-bit stacks. The bit stacks are then converted to binary counts, producing 6:3 counter circuits with no xor gates on the critical path. This avoidance of xor gates results in faster designs with efficient power and area utilization. In VLSI simulations, the proposed counters are faster than existing parallel counters and also consume less power than other higher order counters. Additionally, using the proposed counters in existing counter-based Wallace tree multiplier architectures reduces latency and power consumption.

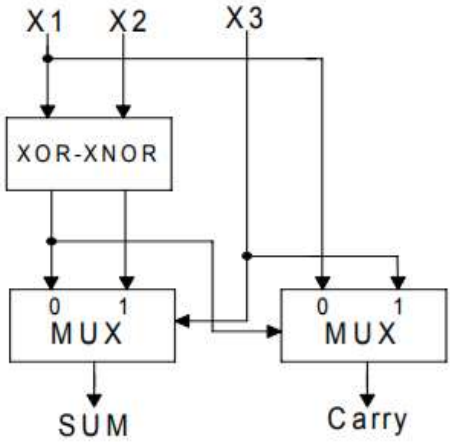

Fig 3.1 Conventional counter

This is the basic full adder circuit. It can be used conventional counter for the design. They are

having XOR-XNOR for the input X1, X2 and X3 is connected by the direct input for mux. The conventional counter designs gives from sum and carry output.

Given inputs X0, X1, and X2, a 3-bit stacker circuit will have three outputs Y0, Y1, and *Y*2 such that the number of "1" bits in the outputs is the same as the number of "1" bits in the inputs, but the "1" bits are grouped together to the left followed by the "0" bits. It is clear that the outputs are then formed by

$$Y0 = X0 + X1 + X2$$
$$Y1 = X0X1 + X0X2 + X1X2$$
$$Y2 = X0X1X2.$$

Namely, the first output will be "1" if any of the inputs is one, the second output will be "1" if any two of the inputs are one, and the last output will be one if all three of the inputs are "1." The Y1 output is a majority function and can be implemented using one complex CMOS gate.

The symmetric stacking method can be used to create a 7:3 counter as well. The 7:3 counters are desirable as they provide a higher compression ratio. The design of the 7:3 counter involves computing outputs for *C*1 and *C*2 assuming both *X*6 = 0 (which matches the 6:3 counter) and assuming *X*6 = 1. We compute the *S* output by adding one additional XOR gate.

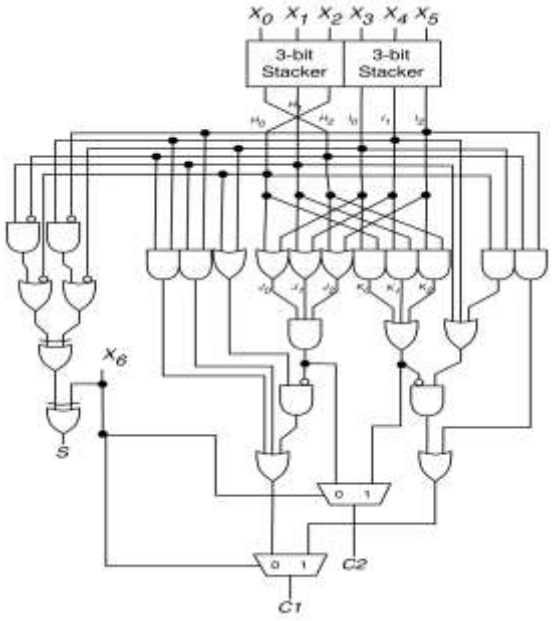

Fig 3.3 7:3 Counters based on symmetric stacking.

This is the basic full adder circuit. It can be used conventional counter for the design. They are having XOR-XNOR for the input X1, X2 and X3 is connected by the direct input for mux. The conventional counter designs gives from sum and carry output.

The proposed modified full adder circuit as shown in figure.3.5. It consists of two 2:1 MUX and an XOR gate. In the proposed structure, one XOR block in the conventional full adder is replaced by a multiplexer block so that the critical path delay is minimized. As can be seen, the critical path delay (i.e.) delay= XOR + MUX

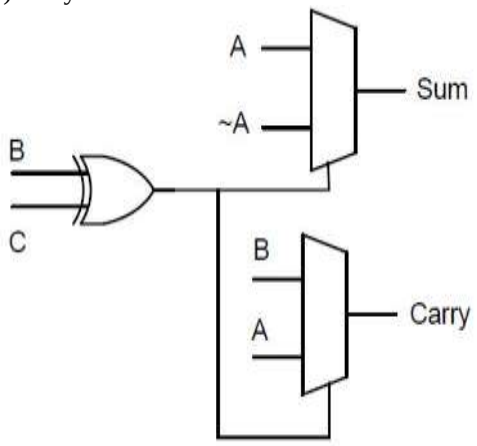

Fig 3.5 Proposed Full Adders

The other two inputs are given to XOR gate, the output of which will act as a select line to both the multiplexers. The inputs of the second multiplexer are, the bits other than the carry bit. This unique way of designing leads to the reduction of the switching activity, which in turn reduces the power. In addition to this, the critical path delay is also reduced compared to the existing designs discussed in literature, which leads to reduction in delay and thus increasing the speed. Operation of the proposed full adder can be explained as follows:

a) When both B and C are zero or one, sum = A;
b) When either of B or C is one and another is zero, sum=A;
c) When both B and C are zero or one, carry= B;
When either of B or C is one and another is zero, carry=A; l

This can be implemented by using second MUX with XOR output as selection line. Since XOR employs most of the power consumption in the adder circuit, by reducing number of XOR gates, power consumption of the full adder can be reduced. The proposed full adder is applied into array multiplier reduction stage to validate the effectiveness. In array structure the partial products is divided into certain levels. In each level, whenever there are three bits, full adder has to be used. Out of the three inputs, one input and its complement is provided as inputs to the first multiplexer.

III. RESULTS AND DISCUSSION

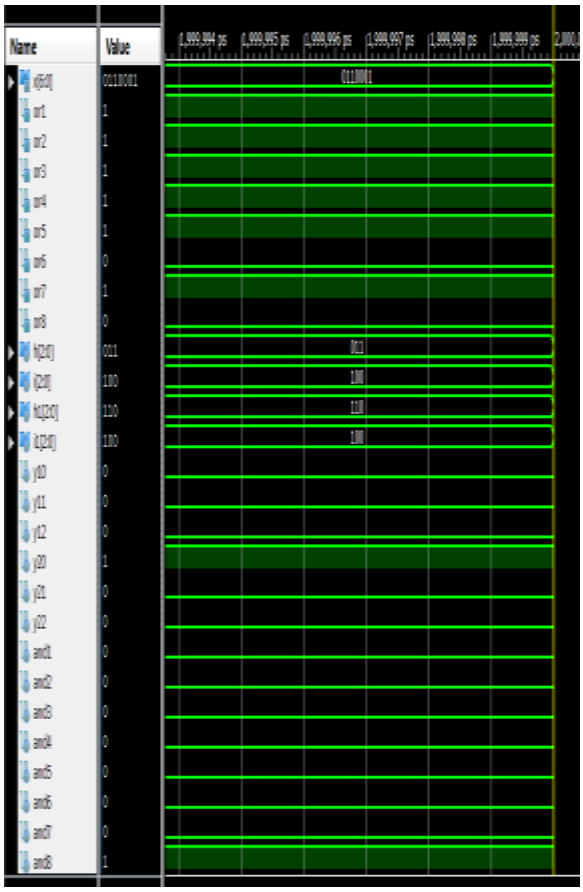

Fig 5.3 7:3.New Full Adder (NFA) Counters

Fig.5.5 Design Summary of .New Full Adder (NFA) Counters

The figure.5.5 shows that design summary of the proposed system. It gives the estimated values of the function.

Fig.5.8 Power Analysis New Full Adder (NFA) Counters of System

The Power analysis of proposed system is shown in above fig 5.7. This analysis mainly concentrated on the number IOs in the entire system to obtain the output. In the entire system, the available numbers of IOs are 12.0 and the IOs used to obtain the output are 0.008. The total available power in the system is 0.081 W and hence the utilization of power is 0.2 W.

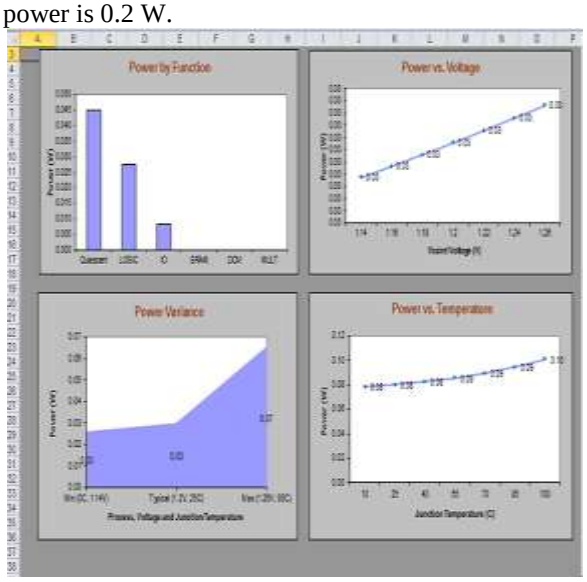

Fig.5.9 Power New Full Adder (NFA) Counters

The proposed has been simulated and the synthesis report can be obtained by using Xilinx ISE 12.1i. The various parameters used for computing existing and proposed systems with Spartan-3 processor are given in the table 7.1

S.NO	Parameter	Existing	Proposed
1	Slice	9	4
2	Lut	15	8

Table.5.1 Comparison of existing and proposed system

The Figure.5.12 given below is shown that there is a considerable reduction in time and area based on the implementation results which have been done by using Spartan-3 processor. The proposed algorithm significantly reduces area consumption when compared to the existing system.

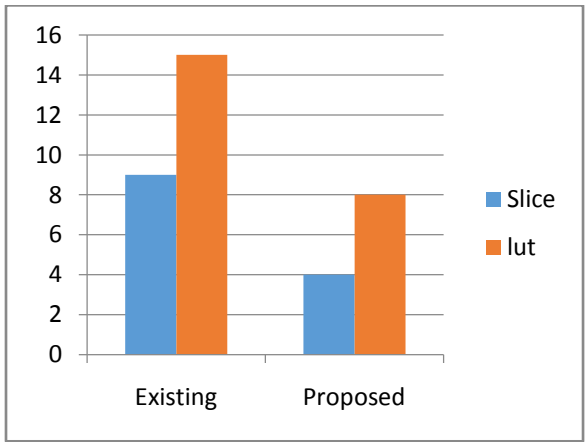

Fig.5.12 Performance Analysis

IV. CONCLUSION

In this brief, a design method for area effective and speed efficient counter is designed and simulated. A binary counter based on a novel symmetric bit Sum and carry calculation approach is proposed. We showed that this counting method can be used to implement 6:3 and 7:3 counters, which can be used in any binary multiplier circuit to add the partial products. We demonstrated that 6:3 counters implemented with this NFA technique achieve higher speed than other higher order counter designs while reducing power consumption.

V. REFERENCES

1. C. S. Wallace, “A suggestion for a fast multiplier,” IEEE Trans. Electron. Comput.vol. EC-13, no. 1, pp. 14–17, Feb. 1964.
2. Dadda, “Some schemes for parallel multipliers,” Alta Freq., vol. 34, pp. 349–356, May 1965.
3. Z. Wang, G. A. Jullien, and W. C. Miller, “A new design technique for column compression multipliers,” IEEE Trans. Comput., vol. 44, no. 8, pp. 962–970, Aug. 1995.
4. M. Mehta, V. Parmar, and E. Swartzlander, “High-speed multiplie Design using multi-input counter and compressor circuits,” in Proc. 10th IEEE Symp. Comput. Arithmetic, Jun. 1991, pp. 43–50.
5. S. Asif and Y. Kong, “Design of an algorithmic Wallace multiplier Using high speed counters,” in Proc. IEEE Comput. Eng. Syst. (ICCES), Dec. 2015, pp. 133–138.
6. S. Veeramachaneni, L. Avinash, M. Krishna, and M. B. Srinivas, “Novel Architectures for efficient (m, n) parallel counters,” in Proc. 17th AC Great Lakes Symp. VLSI, 2007, pp. 188–191.
7. S. Veeramachaneni, K. M. Krishna, L. Avinash, S. R. Puppala, and M.B. Srinivas, “Novel architectures for high-speed and low-power 3-2,4-2 and 5-2 compressors,” in Proc. 20th Int. Conf. VLSI Design HeldJointly 6th Int. Conf. Embedded Syst. (VLSID), Jan. 2007, pp. 324–329.
8. V. G. Oklobdzija, D. Villeger, and S. S. Liu, “A method for speed optimized Partial product reduction and generation of fast parallel multipliers Using an algorithmic approach,” IEEE Trans. Comput., vol. 45, no. 3 pp. 294–306, Mar. 1996
9. S. Asif and Y. Kong, “Analysis of different architectures of counter Based Wallace multipliers,” in Proc. 10th Int. Conf. Comput. Eng.Syst. (ICCES), Dec. 2015, pp. 139–144.
10. J. Gu and C.-H. Chang, “Low voltage, low power (5:2) compressor cell for fast arithmetic circuits,” in Proc. IEEE Int. Conf. Acoust., Speech,Signal Process. (ICASSP), vol. 2. Apr. 2003, pp. 661–664.