ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Special Issue 10, March 2016

DESIGN AND IMPLEMENTATIONOF BZ-FAD USING JOHNSON COUNTER

Dr.M.Meenakumari[1] M.E.,Ph.D.,Associate prof /ECE, SNS College of Engineering
S.Rajesh Kannan[2], A.Pradeep Kumar[2], P.Praveen[2] , S.Vinoth Kumar[2]
[2]UG Scholar, Department of ECE, SNS College of Engineering
Email: mnakumari@gmail.com, srajeshkannan1212@gmail.com, apradeepkumar00712@gmail.com
praveensankar734@gmail.com,billavinoth33@gmail.com

Abstract:

Multiplication is one of the basic arithmetic operations and it requires substantially more hardware resources and processing time than addition and subtraction. The latency increases in the proposed architecture are low power multiplier, this structure is also called as Bypass Zero, Feed A directly (BZ - FA D) for shift - and - add architecture. The architecture reduces the switching activity of the conventional multipliers. The modifications to the multiplier which multiplies A by B include the removal of shifting the B register , direct feeding of A to the adder , bypassing the adder whenever possible, using a ring counter instead of a binary counter and removal of the partial product shift. Simulation results for 16 - bit multiplier that the BZ- FAD architecture lowers the total switching. The proposed multiplier can be used for low - power applications where the speed is not a primary design parameter. The architecture makes use of a low power ring counter proposed in this work. Multipliers have large area, long latency and consume considerable power. At the algorithm and architecture level, this dissertation addresses low-power multiplier design systematically from two aspects: internal efforts considering multiplier architectures and external efforts considering input data characteristics. By using new algorithms or architectures, it is even possible to achieve both power reduction and area/delay reduction, which the strength of high-level optimization. The trade off between power, area and delay is also considered.

Introduction:

In VLSI implementation low power concept is necessary to meet Moore's law and to produce consumer electronics with more back up and less weight. To save a significant power consumption of a VLSI design, it is a good direction to reduce its dynamic power which is the major part of power dissipation. Multiplication occurs frequently finite impulse response filters, fast Fourier transforms discrete cosine transform and other important DSP and multimedia kernels.

Being one among the functional components of many digital systems the reduction of power dissipation in multipliers should be as much as possible. The proposed BZFAD architecture lowers the power dissipation and area when compared to a conventional shift and add multiplier. A multiplexer with one hot encoded bus selector is used for avoiding the switching activity due to the shifting of the multiplier register. Feeder and bypass registers are used for avoiding the unnecessary additions. The BZFAD architecture makes use of a low power ring counter [1]. The DPST has been applied on both the modified Booth decoder and the compression tree of multipliers to enlarge the power reduction. This paper provides the experience of applying an advanced version of our former dynamic power suppression technique (DPST) on multipliers for high-speed and low-power purposes. To filter out the useless switching power, there are two approaches, i.e., using registers and using AND gates, to assert the data signals of multipliers after the data transition. The simulation results show that the DPST implementation with AND gates owns an extremely high flexibility on adjusting the data asserting time which not only facilitates the robustness of DPST but also leads to a 40% speed improvement. Because they are basically accomplished by the repetitive application of multiplication and addition, the speed of the multiplication and addition arithmetic's determines the execution speed and performance of the entire calculation. Because the multiplier requires the longest delay among the basic operational blocks in a digital system, the critical path is determined by the multiplier, in general. For high-speed multiplication, the modified radix-4 Booth's algorithm (MBA) is commonly used. The most effective way to increase the speed of a multiplier is to reduce the number of the partial products because multiplication proceeds a series of additions for the partial products [2]. In Multi-rate Signal processing studies used in Digital Signal processing systems include sample rate conversion. This technique is used for systems with different input and output sample rates. Interpolation and Decimation are very effective and popular in multi rate signal processing applications. This paper proposes a high speed, area and power efficient VLSI architecture for a polyphase decimation filter with decimation factor of three (D=3) using Booth multiplier. By using booth multiplier to multiply signed numbers also. Various key performance metrics such as the number of slices, maximum operating frequency, number of LUT's, input, output bonds, power consumption, setup time, hold time, propagation delay between source and destinations are estimated for the filter of length nine (N=9). The power dissipation is reduced in the polyphase decimation filter using Booth multiplier which consumes low-power when compared to the conventional multiplier. The speed is improved by using carry look-ahead adder. It was observed that the proposed scheme provides an increase in speed, reduction in area and slight reduction in power dissipation when compared to conventional and BFD multiplier and low complexity [3].The proposed delay buffer uses several new techniques to reduce its power consumption. Since delay buffers are accessed sequentially, it adopts a ring-counter addressing scheme. In the ring counter, double-edge-triggered (DET) flip-flops are utilized to reduce the operating frequency by half and the C-element gated-clock strategy is proposed. A novel gated-clock-driver tree is then applied to further reduce the activity along the clock distribution network. Moreover, the gated-driver-tree idea is also employed in the input and output ports of the memory block to decrease their loading, thus saving even more power. In a digital processing chip of mobile communications, the delay buffer takes up a large portion of the circuit layout. If the power consumption of the delay buffer could be reduced significantly, the overall power consumption of the digital processing chip could be reduced significantly as well. On the other hand, as these chips are working at even higher operation frequencies, a new, low-power delay buffer should be operable under high frequencies. A straight ring counter or Overbeck counter connects the output of the last shift register to the first shift register input and circulates a single one (or zero) bit around the ring [4]. A low power 8x8 shift-add multiplier architecture called Universal Shift Register (USR)-Multiplier is proposed. The proposed shift-add multiplier architecture concentrates on minimizing the switching activities of partial products in conventional shift-add multiplier.The switching activities are minimized by bypassing the multiplication operations of Zero's ('0') in Multiplier (A) with Multiplicand (B). This technique of bypassing reduces power consumed by the multiplier architecture. The proposed low power 8x8 multiplier adopts a Universal Shift Register (USR) with clock control unit such that the bypassing of zero's in 'B' is realized.The power consumption of the conventional Shift-and Add multiplier, BZ-FAD shift-and add multiplier and ET shift-and add multiplier for the 8-bit multiplication results. Several computer arithmetic techniques exist to implement a digital multiplier and most among them involve computing a set of partial products, and then summing the partial products together to get the product of the two input values. There are more techniques for reducing the partial product in order to make the operation faster. Numerous algorithms and methods are proposed for efficient multiplier implementation such as Booth's Algorithm. Area and speed are usually inversely proportional to each other , hence improving the speed of the system results in an increase in area [5].

Existing System:

This architecture is to reduce the number of partial products to be added into 2 final intermediate results. To modify the regular adder process and to optimize the partial product generator architecture circuit complexity level.Exiting system is used to optimize the critical path section and to reduce the overall tree based structure work.A fast process of multiplication of two numbers was developed by Wallace. Using this method, a three step process is used to multiply two numbers; the bit products are formed, the bit product matrix is reduced to a two row matrix where the sum of the row equals the sum of bit products, and the two resulting rows are summed with a fast adder to produce a final product. Three bit signals are passed to a one bit full adder ("3W") which is called a three input Wallace tree circuit and the output of sum signal is supplied to the next stage full adder of the same bit. The carry output signal is passed to the next stage full adder of the same no of bit, and the carry output signal thereof is supplied to the next stage of the full adder located at a one bit higher position. Wallace tree is a tree of carry-save adders (CSA) arranged as shown. A carry save adder consists of full adders like the more familiar ripple adders, but the carry output from each bit is brought out to form second result vector rather being than wired to the next most significant bit. The carry vector is 'saved' to be combined with the sum later. In the Wallace tree method, the circuit layout is not easy, although the speed of the operation is high since the circuit is quite irregular. Wallace tree is known for their optimal computation time, when adding multiple operands to two outputs using carry-save adders. The Wallace tree guarantees the lowest overall delay, but requires the largest number of wiring tracks. The number of wires tracks is a measure of wiring complexity. To improve speed, Wallace Tree algorithm can be used to reduce the number of sequential adding stages. On the other hand ―serial-parallel multipliers compromise speed to achieve better performance for area and power consumption. The selection of a parallel or serial multiplier is actually depends on the nature of the application. In this paper, we introduce the multiplier using Vedic algorithm and its architecture and compare them in terms of time delay. In microprocessor, DSP, etc., addition and multiplication of two binary digits is the basic and most commonly used arithmetic operations. Statics show that more than 70% instructions in microprocessor and most of DSP algorithms perform addition and multiplication. So, addition and multiplication operations dominate the execution time. That‘s why; there is need of high speed multiplier. The high speed processor demand has been increasing as a result of increasing computer and signal processing applications. The consumption of power is also an important issue in multiplier design. In order to reduce power consumption, it is good to reduce the number of operations, thereby reducing dynamic power which is a major part of total power consumption so the need for high speed and low power multiplier has increased. Designed mainly concentrates on high speed and low power, efficient circuit design. A good multiplier should be compactly packed and provide high speed and low power consumption unit.Multiplication is an important fundamental function in arithmetic logic operation. Computational performance of a DSP system is limited by its multiplication performance and since, multiplication dominates the execution time of most DSP algorithms; therefore the high-speed multiplier is much desired. Currently, multiplication time is still the dominant factor in determining the instruction cycle time of DSP chip. With an ever-increasing quest for greater computing power on battery-operated mobile devices, design emphasis has shifted from optimizing conventional delay time area size to minimize power dissipation while still maintaining the high performance. Traditionally shift and add algorithm has been implemented to design, however this is not suitable for VLSI implementation and also from delay point of view. Some of the important algorithms proposed in literature for VLSI implementable fast multiplication are Booth multiplier, array multiplier and Wallace tree multiplier. This paper presents the fundamental technical aspects behind these approaches. The low power and high speed VLSI can be implemented with different logic style. The three important considerations for VLSI design are power,area and delay. There are many proposed logics (or) low power dissipation and high speed and each logic style has its own advantages in terms of speed and power. They aim at offering higher speed and lower power consumption, even while occupying the reduced silicon area. This makes them compatible for various complex and portable VLSI circuit implementations. However, the fact remains that the area and speed are two conflicting performance constraints. Hence, innovating increased speed, always results in larger area. In this paper, we arrive at a better trade-off between the two, by realizing a marginally increased speed performed through a small rise in the number of transistors. The new architecture enhances the speed performance of the widely acknowledged Wallace tree multiplier. The structural optimization is performed on the conventional Wallace multiplier, in such a way that the latency of the total circuit reduces considerably. The Wallace tree basically multiplies two unsigned integers. The conventional Wallace tree multiplier architecture comprises of an AND array for computing the partial products, a carry save adder for adding the partial products so obtained and a carry propagate adder in the final stage of addition.

Disadvantage of the existing system is its performance is too low and it consumes more power and it consumes large amount of area. .

Proposed Multipier design with Johnson counter:

The architecture of a conventional shift - and - add multiplier, which multiplies A by B. There are six major sources of switching activity in the multiplier. These sources, which are marked with dashed ovals in the figure, are: (a) Shifts of the B register.(b) Activity in the counter.(c) Activity in the adder.(d) Switching between “0” and A in the multiplexer.(e) Activity in the mux - select controlled by B (0).(f) Shifts of the partial product (PP) register. Note that the activity of the adder consists of required transitions (when the B (0) is nonzero) and unnecessary transitions (when the B (0) is zero). By removing or minimizing any of these switching activity sources, one can lower the power consumption. Since some of the nodes have higher capacitance, reducing their switching will lead to more power reduction. As an example, B (0) is the selector line of the multiplexer which is connected to k gates for a k - bit multiplier. If we somehow eliminate this node, a noticeable power saving can be achieved. Next, we describe how we minimize or possibly eliminate these sources of switching activity. A low - power structure called bypass zero, feed A directly (BZ - FAD) for shift - and - add multipliers is proposed. The architecture considerably lowers the switching activity of conventional multipliers. The modifications to the multiplier which multiplies A by B include the removal of the shifting the register, direct feeding of A to the adder, bypassing the adder whenever possible, using a johnson counter instead of a binary counter and removal of the partial product shift. The architecture makes use of a low - power johnson counter proposed in this work.In the traditional architecture, to generate the partial product, B (0) is used to decide between A and 0.

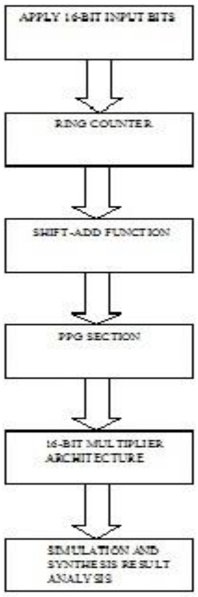

APPLY 16-BIT INPUT BITS
RING COUNTER
SHIFT-ADD FUNCTION
PPG SECTION
16-BIT MULTIPLIER ARCHITECTURE
SIMULATION AND SYNTHESIS RESULT ANALYSIS

Fig 1: System Architecture

If the bit is “1”, A should be added to the previous partial product, where as if it is “0”, no addition operation is needed to generate the partial product. Hence, in each cycle, register B should be shifted to the right so that its right bit appears at B (0); this operation gives rise to some switching activity. To avoid this, in the proposed architecture a multiplexer (M1) with one - hot encoded bus selector chooses the hot bit of B in each cycle. A johnson counter is used to select B (n) in the in the cycle. The same counter can be used for block M2 as well. The johnson counter used in the proposed multiplier is noticeably wider (32 bits versus 5 bits for a 32 - bit multiplier) than the binary counter used in the conventional architecture; therefore an ordinary johnson counter, if used in BZ - FAD, would raise more transitions than its binary counterpart in the conventional architecture. To minimize the switching activity of the counter, we utilize the low - power johnson counter. In the conventional multiplier architecture (see Fig. 1), in each cycle, the current partial product is added to A (when the B (0) is one) or to 0 (when the B (0) is zero). This leads to unnecessary transitions in the adder when B (0) is zero. In these cases, the adder can be bypassed and the partial product should be shifted to the right by one bit. This is what is performed in the proposed architecture which eliminates unnecessary switching activities in the adder. As shown in Fig. 2 , the Feeder and Bypass registers are used to bypass the adder in the cycles where B (n) is zero. In each cycle, the hot bit of the next cycle (i.e., B (n+1)) is checked. If it is 0, i.e., the adder is not needed in the next cycle, the Bypass register is clocked to store the current partial product. If B (n+1) is 1, i.e., the adder is really needed in the next cycle, the Feeder register is clocked to store the current partial product which must be fed to the adder in the next cycle. The power consumption of digital circuits has become a critically important parameter motivating many efforts in reducing the power dissipation of the logic blocks of digital systems. Among different blocks, johnson counter is one of logic components which have several applications including control units and, multiplier and divider architectures, and the arbitration circuitry (round robin arbitration) of routers. One of the important properties of a johnson counter is that its output is one - hot encoded (i.e. , there is always only a single „1‟ - valued bit in its output and all other bits are zero). This property of the johnson counter makes its output wide, especially as the counter size increases. As an example, consider a 5 - bit bin array counter, which counts from 0 to 31.

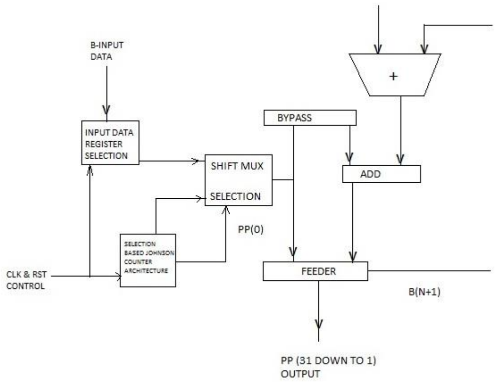
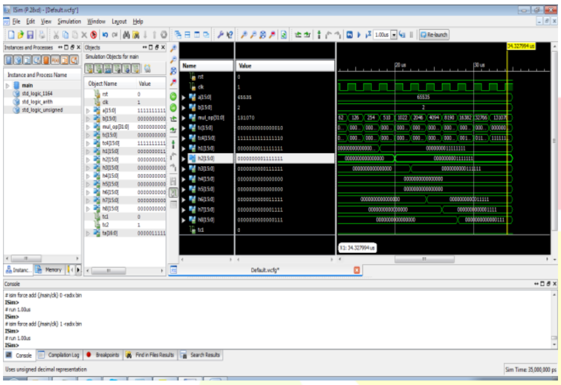


Fig 2:Block diagram of proposed Multipier

The first step toward a low power design is to detect signals which can be temporary or locally shot off without affecting the circuit functionality. Therefore an inspection of johnson counter logic is performed. Hot Block architecture there is many flip- flops, the output of which does not go to any clock generator. This point noticeably reduces the total switching activity of the johnson counter. A johnson counter has a very interesting property of whose advantage is taken in design of Hot Block architecture: a '1' is moving through the cells of the johnson counter, and at each moment all cells of the johnson counter except one is zero(0). This property helps in removing the OR gate from the clock generator. The cited property tells us that in the partitioned johnson counter, there is always only one block the flip - flops of which should be clocked (except on some occasions where the '1' leaves a block and enters another); this block is called the Hot Block.

Results and Discussion:

The proposed method is synthesized by Xilinx 14.7 and simulated by Model sim .



```
# isim force add {/main/clk} 0 -radix bin
ISim>
# run 1.00us
ISim>
# isim force add {/main/clk} 1 -radix bin
ISim>
# run 1.00us
ISim>
```

Fig 3: Simulation Waveform of the multiplier

Conclusion:

In this project, sources of power dissipation in VLSI circuits are studied and methods to reduce power dissipation are explained. Switching activity is found to be an active source of power dissipation in conventional shift and add architecture. A low power architecture known as Bypass Zero Feed A Direct (BZFAD) architecture is proposed. Both the architectures are implemented using VHDL in Xilinx and the results thus obtained are analyzed . The proposed architecture occupies a lesser area compared to that of the conventional multiplier, but it takes more time to produce the output, hence the proposed multiplier can be used in applications where speed is not a primary concern. The power dissipated in the modified circuit is reduced by 20% compared to conventional multiplier architecture.

References:

1. C.N. Marimuthu and P. Thangaraj “Low Power Multiplier Design Using Latches and Flip-Flops” Journal of Computer Science 6 (10): 1117-1122, 2010.

2. Vagolu Aruna, P.Deepthi “High Performance Low Power Dynamic Multiplier” International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, October 2013 116.

3. Ch.D.Vishnupriya , K.Neelima “Efficient Area Minimization with High speed and Low Power Multiplier Structural Design for Multirate Filter Design” International Journal of Advanced Research in Electrical,Electronics and Instrumentation Engineering.

4. B.R.B Jaswanth, R.V.S Rayudu, K.Mani babu, R.Himaja, L.Veda kumar “A Design Of A Low Power Delay Buffer Using Ring Counter Addressing Schemes” International Journal of Technological Exploration and Learning (IJTEL).

5. S. P.Valan Arasu,Dr.S. Baulkani “Modified Universal Shift Register Based Low Power Multiplier Architecture” Journal of Theoretical and Applied Information Technology 10th July 2014. Vol. 65 No.1.

6. Wallace, C.S., “A suggestion for a fast multiplier,” IEEE Trans. Elec. Comput., vol. EC-13, no. 1, pp. 14–17, Feb. 1964.

7. Booth, A.D., “A signed binary multiplication technique,”Quarterly Journal of Mechanics and Applied Mathematics, vol. 4, pt. 2, pp. 236– 240, 1951.

8. Jagadguru Swami Sri Bharath, KrsnaTirathji, “Vedic Mathematics or Sixteen Simple Sutras From The Vedas”, MotilalBanarsidas, Varanasi(India),1986

9. A.P. Nicholas, K.R Williams, J. Pickles, “Application of Urdhava Sutra”, Spiritual Study Group, Roorkee (India), 1984.

10. Neil H.E Weste, David Harris, Ayananerjee,”CMOS VLSI Design, A Circuits and Systems Perspective”,Third Edition, Published by Person Education, PP-327-328

11. Mrs. M. Ramalatha, Prof. D. Sridharan, “VLSI Based High Speed Karatsuba Multiplier for Cryptographic Applications Using Vedic Mathematics”, IJSCI, 2007

12. Thapliyal H. and Srinivas M.B. “High Speed Efficient N x N Bit Parallel Hierarchical Overlay Multiplier Architecture Based on Ancient Indian Vedic Mathematics”, Transactions on Engineering, Computing and Technology, 2004, Vol.2.