A Survey on Different Multiplier Techniques

Y.RamaLakshmanna[#1], G.V.S.Padma Rao[#2], N . Udaya Kumar[#3], K. Bala Sindhuri[#4]
[#1] PG Student, SRKR Engineering College, Bhimavaram, India.
[#2]Professor, SRKR Engineering College, Bhimavaram, India.
[#3]Professor, SRKR Engineering College, Bhimavaram, India.
[#4] Assistant Professor, SRKR Engineering College, Bhimavaram, India

Abstract –

A multiplier has a significant role in various arithmetic operations in digital processing applications which include digital filtering, digital communications and spectral analysis. With the advancement in semiconductor technology, chip density and operating frequency are increasing, so the power consumption in VLSI circuits has become a major problem of consideration. Rapidly growing technology has raised demands for fast and efficient real time digital signal processing applications. A large number of multiplier designs have been developed to enhance their speed. This paper presents a study on some important multiplier techniques.

Keywords: *Multiplier, Vedic Mathematics; Wallace Tree*

I. INTRODUCTION

In the design of systems using digital signal processing and other applications multiplier is an important basic building block. Many researchers are continuously trying to design multiplier with high speed, low power consumption, regular structure, such that it occupies less area for compact VLSI implementation.

Many algorithms are proposed in the past to perform multiplication process. Every algorithm offerings its own advantages and having tradeoff between themselves by means of their speed area, power consumption and circuit complexity.

Add and shift multiplication process is the common method. The main parameter in parallel multipliers that determines the performance of the multiplier is the number of products to be added. By the use of increased parallelism, the amount of shifts between intermediate sums and the partial products to be added will increase which may gives in reduction in speed. Due to irregularity of structure silicon area increases and also due to increase in interconnect resulting from complex routing power consumption increased. Depends on the nature of application, the selection of a parallel or serial multiplier is made.

The organization of this paper is as follows: Section II deals with literature survey. In this section an introduction to serial multiplier, Shift and Add multiplier, Array multiplier, Booth Multiplier, Constant coefficient multiplier, Vedic multiplier and Wallace tree multiplier is presented. A conclusion is given in section III.

II. LITERATURE SURVEY

A. Serial Multiplier

The serial multiplier is used where the area and power is most important and delay can be tolerated. In this one adder is used to add partial products. The multiplier and multiplicand inputs are arranged such that they synchronized with the circuit behavior. Depending on the length of the multiplier and multiplicand, the inputs can be presented at different rates. Here two clock signals are used. One for data and another for reset operation. The main drawback of serial multiplier algorithm is not suitable for large values of multiplier and multiplicand.

B. Shift and Add Multiplier

For standard add shift algorithm, every multiplier bit gives one multiple of the multiplicand which is added to partial products. A more number of multiplicands are added, if multiplier is very large. In this situation the number of additions to be performed determines the delay of multipliers. The performance will get better, if the number of additions are minimum.

The value of multiplicand to be accumulated and added depends on the value of multiplier LSB. At each clock cycle the multiplier is shifted one bit to the right and its value is tested. If it is a 0, then only a shift operation is performed. If the value is a 1, then the multiplicand is added to the accumulator and is shifted by one bit to the right. After all the multiplier bits have been tested the product is in the accumulator. The accumulator is 2Q (P+Q) in size and initially the Q, LSBs contains the Multiplier. The delay is Q cycles maximum. Serial multipliers consume more power. So power is an important criterion there we should prefer parallel multipliers like booth multipliers to serial multipliers.

The parallel multipliers like booth multiplier perform the computations using very few adders and very few iterative steps. As a result of which they cover minimum space as compared to the serial multiplier.

C. Array Multipliers

Laxman S, Darshan Prabhu R, Mahesh S Shetty ,Mrs. Manjula BM, Dr. Chirag Sharma have presented this algorithm. The detailed study of different multipliers based on Array Multiplier, Constant coefficient multiplication (KCM) and multiplication based on vedic mathematics is introduced in this work[1].

With an array multiplier two binary numbers will be multiplied by using of an array of half adders and full adders. Simultaneously addition of the different product terms is done in this array. By using an array of AND gates, the partial product terms are formed..Following this an array of AND gates, the adder array is used. The hardware structure for an pxq bit multiplier is described as (pxq) AND gates (p-1)q adders .Here q Half adders and (p-2).q Full adders. Array multiplier doing the multiplication process in traditional way. It looks like regular structure. Hence wiring and the layout are done in a much simplified manner. Add and shift algorithm is employed in an array multiplier. Implementation of this multiplier is simple but it requires larger area, with considerable delay also[2].

Instead of Ripple Carry Adder (RCA), in this multiplier Carry Save Adder (CSA) is used for adding each group of partial product terms, because RCA is the slowest adder among all other types of adders available. In case of multiplier with CSA , partial product addition is carried out in Carry save form and RCA is used only in final addition. In this algorithm , no waiting is necessary until all the partial products have been formed before summing them. As soon as the partial products formed immediately the addition of partial product can be done[3].

The major advantage of the array multiplier is that it has a regular structure. Another advantage of the array multiplier is its ease of design for a pipelined architecture. Major limitation of array multiplier is its size. As operand sizes increase, arrays grow in size at a rate equal to the square of the operand size. It comes under conventional multiplier.

D. Booth Multiplier

Ruchi Sharma have presented this algorithm**.** In this work[4] different multiplier architectures are implemented in Xilinx FPGA and compared for their performance It is a powerful algorithm for signed-number multiplication, which treats both positive and negative numbers uniformly.

To overcome the limitation of array multiplier the speed of the multiplier is increased by the booth algorithm. Booth algorithm reduces the number of partial products. Here, the multiplier considers two number of bits at a time for the multiplication process. The multiplication process for both signed and unsigned numbers can be done in this booth multiplier. This multiplier considers the 2's compliment of the given multiplicand and multiplier. It is based on radix-2 computation. In add-shift operation each multiplier bit multiply with the multiplicand and to be added to the partial product.

For very large multiplier, a large number of multiplicands to be added. In this multiplier, number of additions can decide the multiplier delay. Booth algorithm can easily reduce the no.of multiplicand multiplies.For a n-bit number can be represented as n/2-digit radix 4 number, a n/3- digit radix 8 number and so on.

Major limitation of array multiplier is its size. As operand sizes increase, arrays grow in size at a rate equal to the square of the operand size ,hence speed of multiplier reduces .In order to increase the speed of multiplier booth algorithm is used. The Booth multiplier makes use of Booth encoding algorithm in order to reduce the number of partial products by considering two bits of the multiplier at a time, thereby achieving a speed advantage over other multiplier architectures. This algorithm is valid for both signed and unsigned numbers. It accepts the number in 2's complement form, based on radix-2 computation[4,10]

The low power consumption quality of booth multiplier makes it a preferred choice in designing different circuits. By implementing both Radix-2 & Radix -4 multiplier using booth algorithm their computation speed increases so much.

E. Constant Coefficinet Multiplier

Mohammed Hasmat Ali, Anil Kumar Sahani have presented this algorithm. In this work[5] introduces the multiplier is based on an algorithm Urdhva Tiryakbhyam (Vertical & Crosswise) of ancient Indian Vedic Mathematics. This method is based on ROM approach. In conventional KCM, one input is fixed but here, both the inputs for the multiplier can be variables. ROM approach is employed in this method. The two inputs are variables for this multiplier. By averaging, squaring and subtraction this two variables multiplication is performed. The averaging process is done by right shifting the sum by one bit. Here ROM stores the squares of the numbers, finally the result will be calculated instantaneously.

In KCM, the multiples are stored in ROM. In this method, the difference of two given numbers is decided as even or odd. For even difference, the two variable multiplication is performed by averaging squaring and subtraction. For odd difference, the average is a floating point number,the square of numbers end with 5 is found by the Vedic sutra – Ekadikena purvena.These squares of averages and difference is stored in two port memory simultaneously to reduce memory accessing time[6].

Thus, division and multiplication operations are effectively converted to subtraction and addition operations using Vedic Maths. Square of both Average and Deviation is read out simultaneously by using a two port memory to reduce memory access time.

F. Vedic Multiplier

Mohammed Hasmat Ali, Anil Kumar Sahani have presented this algorithm. In this work[7,11] introduces the multiplier is based on an algorithm Urdhva Tiryakbhyam (Vertical & Crosswise) of ancient Indian Vedic Mathematics. In Vedic mathematics, two of sixteen sutras are mainly used for multiplication process. One is Urdhva-tiryagbhyam sutra and other is Nikhilam Navatascaramam dasatahs. Urdhva – Tiryagbyam performs the operation of two decimal numbers multiplication. It is applicable ao all types of multiplication between two large numbers. It is also referred as “Vertically and crosswise algorithm”. This can solve the multiplication of larger number (N X N bits) by breaking it into smaller sizes. Vedic multiplier gives the improved speed than the conventional multiplier and reduces the system memory. Very small area is needed for this multiplier. For binary and decimal number multiplication this multiplier is used exclusively. To solve complex calculations by simple techniques Vedic Mathematics is used. The strategy applied for developing a 64 x 64-bit Vedic multiplier is to design a 2 x 2- bit Vedic multiplier as a basic building module for the system. The development a 4 x 4-bit multiplier is designed using 2 x 2-bit Vedic multiplier. Further in the same manner 8 x 8, 16 x 16 and 32 x 32- bit Vedic multiplier is designed. This type of Multiplier plays a very important role in today's digital circuits[12].

The first step in multiplication is vertical multiplication of LSB of both multiplicands, and then second step is crosswise multiplication and additions of the partial products. Third step involves vertical multiplication of MSB of the multiplicand and addition with the carry propagated from step 2.

Vedic multiplier gives the improved speed than the conventional multiplier and reduces the system memory. Very small area is needed for this multiplier as compared to other multiplier architecture. For binary and decimal number multiplication this multiplier is used and also it is used in unsigned and signed number multiplication. The Vedic Urdhva multiplier is much more efficient than Array and Constant Coefficient Multiplier (KCM) in terms of execution time. The main Disadvantage of this multiplier is system becomes complex for complex multiplications[8].

G. Wallace Tree Multiplier

Vidhi Gupta[1], J. S. Ubhi[21]Scholar have presented this algorithm. In this work[7,9] introduces the Analysis And Comparison Of Various Parameters For Different Multiplier Designs. Wallace tree is an efficient hardware implementation of a digital circuit that multiplies two integers. Wallace trees are irregular structure in that the informal description does not specify a systematic method for the compressor interconnections. But still it is an efficient implementation of adding partial products in parallel. Using this method, a three step process is used to multiply two integer numbers. First step is to multiply each bit of one of the arguments, by each bit of the other, yielding n^2 results. Based on the position of the multiplied bits, the wires carry different weights. The second step is to reduce the number of partial products to two by layers of full and half adders. The third step is to group the wires in two numbers, and then add them with conventional adder . There are two different architectures of Wallace tree multiplier are available. First one is designed using only half adder and full adder, while the second one uses a more sophisticated carry skip adder (CSA).

Wallace Tree Multiplier using only Full and Half Adders is employed. With this idea of Wallace tree method is to reduce the number of adders by minimizing the number of the half adder in any multiplier. The first partial product is the least significant bit in the output of the multiplier result. After that, moving to the next column of the partial product if there are any adders from the previous product, the full Adder is used otherwise a half adder is used and so on[13].

Wallace Tree Multiplier using Carry Skip Adder is also employed. A carry-skip adder consists of a simple ripple carry-adder with a special speed up carry chain called a skip chain. The purpose of using the CSA is to improve the worst case path delay. A 4 -bit CSA is used for implementing the Wallace tree multiplier. The carry output from the first addition is the carry input in the second addition. The advantage of using CSA is to increase the maximum frequency.

The Wallace tree multiplier using CSA occupies smallest area while the Vedic multiplier using KSA consumes large area. The power consumption of the four multipliers is convergent. In the parallel FIR filter architecture, Wallace tree multiplier using CSA has the minimum critical path whereas the Vedic multiplier using conventional adder has the maximum delay.

III. CONCLUSION

It is concluded that the parallel multipliers are much option than the serial multiplier. In case of parallel multipliers, the total area is much less than that of serial multipliers. Hence the power consumption is also less.

Wallace tree multiplier exhibits good features as compared to Array multiplier. It has lower power dissipation both static and dynamic. It has lesser delay and good noise immunity

The performance of both sequential and parallel micro programmed FIR filters using Wallace tree and Vedic multipliers for different number of taps. Wallace tree multiplier delivers better performance than Vedic multiplier for both the FIR filter architectures.

The Wallace tree multiplier using regular full and half adder has minimum area while Vedic using KSA has maximum area. In terms of power consumption, the four multiplier architectures are very close. The Wallace tree multiplier has the smallest critical path delay as compared to Vedic multiplier.

REFERENCES

[1] Laxman S, Darshan Prabhu R, Mahesh S Shetty , Mrs. Manjula BM, Dr. Chirag Sharma "FPGA Implementation of Different Multiplier Architectures" International Journal of Emerging Technology and Advanced Engineering Volume 2, Issue 6, June 2012 ISSN: 2250- 2459.

[2] S.P.Pohokar, R.S.Sisal, K.M.Gaikwad, M.M.Patil, Rushikesh Borse "Design and Implementation of 16x16 Multiplier using vedic mathematics" International Conference on Industrial Instrumentation and control(ICIC) College of Engineering Pune, India, May 28-30,2015.

[3] V.Priyanka Brahmaiah ,L.Dhrma Teja, Dr Y.Padma sai "Study on comparison of various multipliers" International Journal of Electronics and communication engineering & technology(IJECET), ISSN 0976-6464 volume 4 Issue 5 September – October, 2013 pp-132-142.

[4] Ruchi Sharma "Analysis of Different Multiplier with Digital Filters Using VHDL Language" International Journal of Engineering and Advanced Technology (IJEAT) ISSN: – 8958, Volume-2, Issue-1, October 2012

[5] Mohammed Hasmat Ali, Anil Kumar Sahani " Study, Implementation and Comparison of Different Multipliers based on Array, KCM and Vedic Mathematics Using EDA Tools" International Journal of Scientific and Research Publications, Volume 3, Issue 6, June 2013 ISSN 2250-3153

[6] S Venkateswara Reddy "Design and Implementation of 32 Bit multiplier using vedic mathematics" International journal of advanced research in electrical electronics and Instrumentation engineering Volume 2 , Issue 8 August 2013.

[7] Vidhi Gupta, J. S. Ubhi "Analysis And Comparison Of Various Parameters For Different Multiplier Designs" International Journal Of Scientific Research And Management Studies (IJSRMS) ISSN: 2349-3771Volume 1 Issue 4, Pg: 136-147

[8] Dipika Chauhan, Prof. Kinjal vagadia, Prof.Kirit patel " design and Implementation of High speed 64-bit multiply and Accumulator Unit using FPGA" International Journal of Innovative research in Computer and Communication Engineering Volume 3 Issue 5 May 2015.

[9] Aditya Kumar Singh, Bishnu Prasad De, Santanu Maity "Design and Comparison of Multipliers Using Different Logic Styles" International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-2, May 2012.

[10] S.dhivya, Mr.T.Nallusamy "radix-8 modified booth recoder for high speed add-multiply operator" international conference on engineering trends and science & humanities (icetsh-2015) issn: 2348 – 8549 www.internationaljournalssrg.org page 1

[11] S Muneer Ahamed 1, V Madhuri"Implementation Of High Speed 8-Bit Vedic Multiplier Using Barrel Shifter" SSRG International Journal of Electronics and Communication Engineering (SSRG-IJECE) – volume1 issue10 Dec 2014 ISSN: 2348 – 8549 www.internationaljournalssrg.org Page 20

[12] Sulakshna Thakur#1, Pardeep Kumar"Area-Efficient & High Speed Ripple Carry based Vedic Multiplier" SSRG International Journal of Electronics and Communication Engineering (SSRG-IJECE) – EFES April 2015 ISSN: 2348 - 8549 http://www.internationaljournalssrg.org Page 6

[13] M.Rohini, M.Madhumetha, S.Ranjith"An Survey on Adders & Multiplier For High Speed & Low Power Application" International Conference on Futuristic Trends in Computing and Communication (ICFTCC-2015) ISSN: 2348 – 8549 www.internationaljournalssrg.org Page 55