AREA EFFICIENT HIGH PERFORMANCE MAC FOR DSP APPLICATIONS

[1]CHILIVERU PAVANI, [2]Y.SREENIVASULU, [3]DEVARAPALLI RAVIKISHORE

[1]152W1D6803 GVR&S College of Engineering and Technology
[2]Prof. Dept.Of ECE GVR&S College of Engineering and Technology
[3]162W1D6803 GVR&S College of Engineering and Technology
E-mail: [1]chpavani413@gmail.com., [2]yerraboina@yahoo.co.in, [3]023kishore@gmail.com

Abstract- In many DSP application multipliers are playing major role. For any embedded based DSP wireless application we propose Vedic method for multiplication of binary numbers which strikes a difference in the actual process of multiplication itself. It enables parallel generation of partial products and eliminates unwanted multiplication steps. The proposed algorithm is modeled using Verilog, a hardware description language. The propagation time of the proposed architecture Vedic multiplier is 3.433ns. Implementation has been done for the ALTERA FPGA device, CYCLONE-III. We can run this core up to several MHz under normal operating conditions and with all the input signals having normal inputs. The results shows that multiplier implemented using Vedic multiplication is efficient in terms of area and speed compared to its implementation using Array and Booth multiplier architectures.

I. INTRODUCTION

Multiplication is an important fundamental function in arithmetic operations. Multiplication-based operations such as Multiply and Accumulate(MAC) and inner product are among some of the frequently used Computation- Intensive Arithmetic Functions(CIAF) currently implemented in many Digital Signal Processing (DSP)applications such as convolution, Fast Fourier Transform(FFT), filtering and in microprocessors in its arithmetic and logic unit [1]. Since multiplication dominates the execution time of most DSP algorithms, so there is a need of high speed multiplier. Currently, multiplication time is still the dominant factor in determining the instruction cycle time of a DSP chip. The demand for high speed processing has been increasing as a result of expanding computer and signal processing applications.

Higher throughput arithmetic operations are important to achieve the desired performance in many real-time signal and image processing applications [2]. One of the key arithmetic operations in such applications is multiplication and the development of fast multiplier circuit has been a subject of interest over decades. Reducing the time delay and power consumption are very essential requirements for many applications [2, 3].This work presents different multiplier architectures.

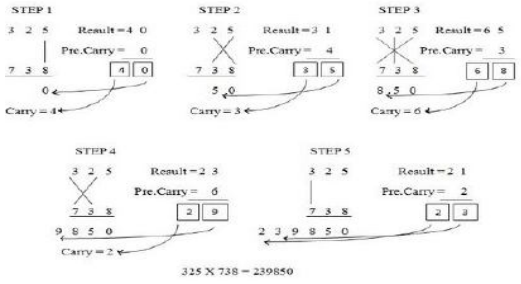


Fig.1: Multiplication of two decimal numbers

II. VEDIC MATHEMATICS

Multiplier based on Vedic Mathematics is one of the fast and low power multiplier. Minimizing power consumption for digital systems involves optimization at all levels of the design. This optimization includes the technology used to implement the digital circuits, the circuit style and topology, the architecture for implementing the circuits and at the highest level the algorithms that are being implemented. Digital multipliers are the most commonly used components in any digital circuit design. They are fast, reliable and efficient components that are utilized to implement any operation. Depending upon the arrangement of the components, there are different types of multipliers available. Particular multiplier architecture is chosen based on the application. In many DSP algorithms, the multiplier lies in the critical delay path and ultimately determines the performance of algorithm.

The speed of multiplication operation is of great importance in DSP as well as in general processor. In the past multiplication was implemented generally with a sequence of addition, subtraction and shift operations. There have been many algorithms proposals in literature to perform multiplication, each offering different advantages and having tradeoff in terms of speed, circuit complexity, area and power consumption. The multiplier is a fairly large block of a computing system. The amount of circuitry involved is directly proportional to the square of its resolution i.e. A multiplier of size n bits has n gates.

A. DIGITAL MULTIPLIERS

Digital multipliers are the core components of all the digital signal processors (DSPs) and the speed of the DSP is largely determined by the speed of its multipliers. Two most common multiplication

algorithms followed in the digital hardware are array multiplication algorithm and Booth multiplication algorithm.

The computation time taken by the array multiplier is comparatively less because the partial products are calculated independently in parallel. The delay associated with the array multiplier is the time taken by the signals to propagate through the gates that form the multiplication array. Firstly, least significant bits are multiplied which gives the least significant bit of the product (vertical). Then, the LSB of the multiplicand is multiplied with the next higher bit of the multiplier and added with the product of LSB of multiplier and next higher bit of the multiplicand (crosswise). The sum gives second bit of the product and the carry is added in the output of next stage sum obtained by the crosswise and vertical multiplication and addition of three bits of the two numbers from least significant position.

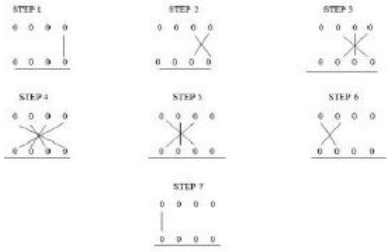


Fig. 2: Line diagram for multiplication of two 4 - bit numbers.

Next, all the four bits are processed with crosswise multiplication and addition to give the sum and carry. The sum is the corresponding bit of the product and the carry is again added to the next stage multiplication and addition of three bits except the LSB. The same operation continues until the multiplication of the two MSBs to give the MSB of the product. For example, if in some intermediate step, we get 110, then 0 will act as result bit (referred as rn) and 11 as the carry (referred as cn). It should be clearly noted that cn may be a multi-bit number. Thus we get the following expressions:

- r0=a0b0
- c1r1=a1b0+a0b1
- c2r2=c1+a2b0+a1b1 + a0b2
- c3r3=c2+a3b0+a2b1 + a1b2 + a0b3
- c4r4=c3+a3b1+a2b2 + a1b3
- c5r5=c4+a3b2+a2b3
- c6r6=c5+a3b3

B. ALGORITHM FOR DIVIDE AND CONQUER APPROACH

To illustrate the multiplication algorithm, let us consider the multiplication of two binary numbers a3a2a1a0 and b3b2b1b0. As the result of this multiplication would be more than 4 bits, we express it as... r3r2r1r0. Line diagram for multiplication of two 4-bitnumbers is shown in Fig. 2.2 which is nothing but the mapping of the Fig.2.1 in binary system. For the simplicity, each bit is represented by a circle. Least significant bit r0 is obtained by multiplying the least significant bits of the multiplicand and the multiplier. The process is followed according to the steps shown in Fig. 2

III. HARDWARE IMPLEMENTATION:

This hardware design is very similar to that of the famous array multiplier where an array of adders is required to arrive at the final product. All the partial products are calculated in parallel and the delay associated is mainly the time taken by the carry to propagate through the adders which form the multiplication array. Clearly, this is not an efficient algorithm for the multiplication of large numbers as a lot of propagation delay is involved in such cases. To deal with this problem, we now discuss Karatsuba Sutra which presents an efficient method of multiplying two large numbers.

IV. MODIFIED BOOTH ENCODER:

In order to achieve high speed multiplication, multiplication ALGORITHMS using parallel counters, such as the modified Booth algorithm has been proposed, and some multipliers based on the algorithms have been implemented for practical use. This type of multiplier operates much faster than an array multiplier for longer operands because its computation time is proportional to the logarithm of the word length of operands.Booth multiplication is a technique that al lows for smaller,faster multiplication circuits, by reco ding the numbers that aremultiplied. It is possible to reduce the number of partial products by half, by using the technique of radix-4 Booth recoding.

The basic idea is that, instead of shifting and adding for every column of the multiplier term and multiplying by 1 or 0, we only take every second column, and multiply by ±1, ±2, or 0, to obtain the same results. The advantage of this method is the halving of the number of partial products. To Booth recode the multiplier term, we consider the bits in blocks of three, such that each block overlaps the previous block by one bit. Grouping starts from the LSB, and the first block only uses two bits of the multiplier. Figure 3 shows the grouping of bits from the multiplier term for use in modified booth encoding.

Block	Re - coded digit	Operation on X
000	0	0 X
001	+1	+1 X
010	+1	+1 X
011	+2	+2 X
100	-2	-2 X
101	-1	-1 X
110	-1	-1 X
111	0	0 X

Fig. 3: Truth table of modified booth algorithm

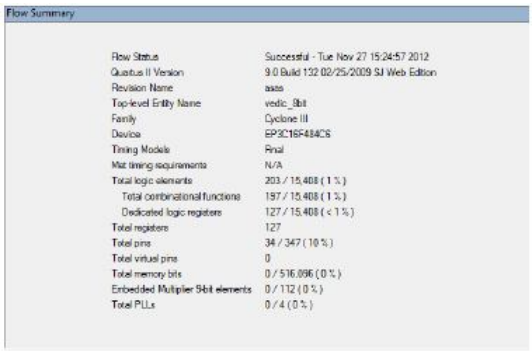


Fig. 4: Proposed system area utilization report

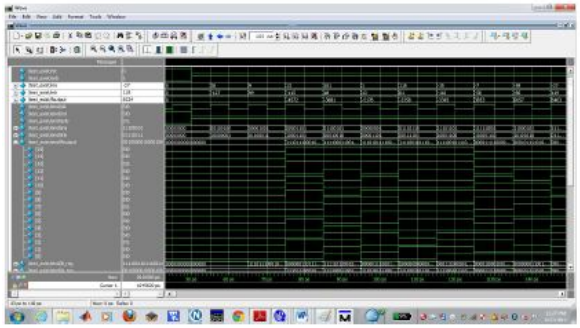


Fig.5: simulated output

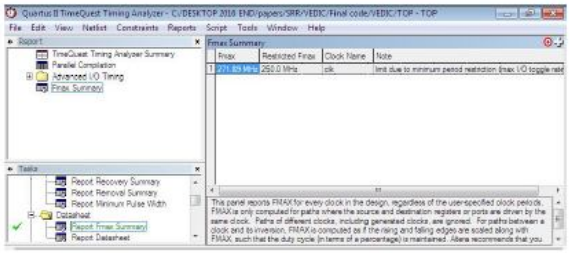
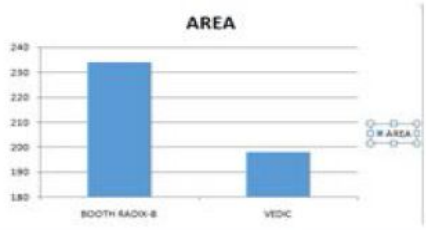


Fig. 6: performance report

Table I complexity comparison report booth radix-8 Vs Vedic method



Multiplier Type used	AREA
BOOTH	234
VEDIC-Karatsuba	198

CONCLUSIONS

The Delay of the proposed Vedic multiplier is far less as compared to efficient high performance radix-8 booth algorithm. Hence, the maximum operating frequency of the proposed multiplier is found to be less under normal operating conditions and with all the input signals having normal inputs. It is evident from the summary that divide and conquer approach is required for the proposed Architecture. The advantages of this proposed architecture is efficient in speed and area(1ess resources used, such as less number of multipliers and adders) and is Flexible in design Example, the same architecture can be extended for 16 -bit, 64bit etc multiplication.

REFERANCES:

[1] H.Guilt, "Fully Iterative Fast Array for Binary Multiplication", Electronics Letters, vol. 5, p. 263, 1969.
[2] Vadiraj Sagar, Shripad Sagar, Sudhindracharya, Vedavyas Mathad, Subhash Kulkarni "Vhdl Implementation of Vedic Mathematical Sutras" Department of Electronics & Communication, P DA College of Engineering.
[3] "Lifting Scheme Discrete Wavelet Transform Using Vertical and Crosswise Multipliers" Anthony O'Brien and Richard Conway, ISSC, 2008,Galway, June 18-19.
[4] Zhijun Huang, Milos D. Ercegovac, "High-Performance Lefl-to-Right Array Multiplier Design," arith, pp.4, 16th IEEE Symposium on Computer Arithmetic (ARITH-16 '03), 20 03
[5] Ramalatha, M Dayalan, K D Dharani, P Priya, and S Deborah, "High speed energy efficient ALU design using Vedic multiplication techniques", ICACTEA, 2009. pp. 600-3, Jul 15-17, 2009.
[6] H ThapJiyal, M B Srinivas, and H R Arabnia, "Design and Analysis of a VLSI Based High Performance Low Power Parallel Square Architecture", in Proc. Int. Con! Algo. Math. Compo Sc., Las Vegas, pp. 72-6, Jun. 2005.
[7] Harpreet Singh Dhillon Abhijit Mitra, "A Digital Multiplier Architecture using Urdhava Tiryabhyam Sutra of Vedic Mathematics", Indian Institue of Technology, Guwahatti.
[8] Purushottam D. Chidgupkar and Mangesh T. Karad, "The Implementation of Vedic Algorithms in Digital Signal Processing on 8085/8086", Global J. of Engng. Educ., Vol.8 No.2 © 2004 UICEE Published in Australia.
[9] Himanshu Thapliyal and Hamid R. Arabnia, "A Time-Area- Power Efficient Multiplier and Square Architecture Based on Ancient Indian VediC", Department of Computer Science, The University of Georgia, 415 Graduate Studies Research Center Athens, Georgia 30602-7404, U.S.A.
[10] M.M.Mano, "Computer system Architecture", Englewood Cliffs, NJ: Prentice-hall,1982
[11] Jagadguru Swami Sri Bharati Krishna Tirthji Maharaja," Vedic Mathematics", Motilal Banarsidas, Varanasi, India, 1986 .
[12] Harpreet Singh Dhillon and Abhijit Mitra, "A Reduced- Bit Multiplication Algorithm for Digital Arithmetics ", International Journal of Computational and Mathematical Sciences, 2008.

★★★