IF : 4.547 | IC Value 80.26 Volume-6, Issue-3, March - 2017 • ISSN No 2277 - 8160

Original Research Paper Engineering

FPGA IMPLEMENTATION OF HIGH SPEED LOW POWER 16 BIT MULTIPLIER

P.RAMA KRISHNA Assistant Professor, Dept of VLSI, ANURAG GROUP OF INSTITUTIONS, TS ,India

G.PRIYANKA PG Scholar, Dept of VLSI, ANURAG GROUP OF INSTITUTIONS, TS, India

ABSTRACT The performance of multiplication is very important in terms of speed and power in Digital Signal Processing (DSP) applications. In general the multiplication in terms of more number of bit require a lot of time and even the speed of the multiplication gets reduced .To overcome this we design a 16x16 multiplier in this paper. Here the multiplication of 2 multiplier bits each bit consist of 16 input bits individually are taken and produce a 32 bit output.

However ,for the present day application Vedic Multiplier based on Vedic Mathematics are presently under focus due to their low power consumptions and less delay product .In this paper we propose a design of 16 bit multipliers using 4 multipliers each multiplier consist of 2bitswhich consist of 8 bits each as input and obtain an output of 32 bits. In this paper we use different types of fast adders such as Carry Save Adder, Carry Select Adder, Brent Kung Adder. This paper is even implemented on Z-Board.

KEYWORDS : Vedic multiplier, carry save array, power-delay product,carry-select adder

INTRODUCTION

Multiplier plays an important role in the design of many digital based applications such as in DSP Application.SuchApplicationsare convolution,FastFourieTransform(FFT),Discrete Cosine Transform(DCT) and filtering. The speed in DSP applications largely depends on the multiplier Block. As the multiplier block plays a key role the demand for it is more. Over the past few years, many researchers have come of with many types of multipliers where they used different types of algorithms such as Array, Booth algorithms Carry Save, Wallace tree and modified Booth algorithms .where a number of multiplier architectures also have been proposed based on these algorithms that include parallel, serial, and serial –parallel multipliers.

Vedic multipliers are based on Vedic Sutras. In Sanskrit word 'Veda' stands for 'knowledge'. Vedic mathematics is believed to be reconstructed from Vedas by Sri Bharti Krishna Tirathaji between the years 1911 to 1918. The Vedic mathematics has been divided into sixteen different Sutras which can be applied to any branch of mathematics like algebra, trigonometry, geometry etc. Its methods reduce the complex calculations into simpler ones because they are based on methods similar to working of human mind thereby making them easier. Since speed is our major concern, utilizing such type of architectures is not a feasible approach since it involves several time consuming operations.

In order to address the disadvantages with respect to speed of the above mentioned methods, and explored a new approach to multiplier design based on ancient Vedic Mathematics.

A. Urdhva Tiryakbhyam

This sutra is based on "Vertically and Crosswise"technique. It makes almost all the numeric computations faster and easier. The advantage of multiplier based on this sutra over the others is that with the increase in number of bits, area and delay increase at a smaller rate in comparison to others.

Higher binary multiplications can also be obtained with the help of lower multiplication units and the adder unit. The individual multiplication products are obtained by same partitioning method, ultimately using the 2X2 bit multiplication method. For NXN multiplication unit, we require four N/2 bit multipliers, two N bit full adders, one half adder and N/2 bit full adder to add the sum and carry of half adder. High speed of multiplier depends highly upon speed of adder units used.

DESCRIPTION

Let us consider two inputs *X*1*X*0 and *Y*1*Y*0, each of 2 bits. *P*3*P*2*P*1*P*0 represents each bit of the final computed product. The result is obtained after generating partial products and adding them as per the basic method of multiplication shown below.

*X*1 *X*0
× *Y*1 *Y*0
————————————
*X*1*Y*0 *X*0*Y*0
*X*1*Y*1 *X*0*Y*1
————————————
*P*3 *P*2 *P*1 *P*0

In Vedic mathematics, the vertical multiplication of bits *X*0 and *Y*0 gives product *P*0. The product term *P*1 is obtained by the addition of crosswise bit multiplication i.e. *X*1 & *Y*0 and *X*0 & *Y*1. *P*2 is addition of the vertical product of bits *X*1 & *Y*1 along with the carry generated from the previous addition during *P*1. *P*3 output is the carry generated from the previous calculation of *P*2. Fig. 2 shows the module that generate 4 outputs from the two 2-bit inputs by using AND gates and half adder and it is known as 2×2 multiplier block.

According to Vedic mathematics, initially vertical multiplication of *XL* & *YL* is carried out, which gives partial product outputs *p*0[3 : 0]. Then, the middle one shows crosswise multiplication of two, 2 × 2 multiplier with inputs *XH* & *YL* and *XL* & *YH*, generates *p*1[3 : 0] and *p*2[3 : 0] partial product outputs respectively. Finally, the last block inputs *XH* & *YH* multiplies vertically and generates the partial product outputs as *p*3[3 : 0]. The first two final product outputs *P*0 and *P*1 are same as that of the partial products *p*0[0] and *p*0[1]. The remaining product terms are obtained by using three conventional adders as shown in Fig. 3. The inputs for adder1 are *{p*3[3 : 0], 00*}* and *{*00, *p*2[3 : 0]*}* and for adder2 are *p*1[3 : 0] and *{*00, *p*0[3 : 2]*}*. The outputs of adder1 and adder2 are given inputs to the adder3, that generates the final product terms *P*[7 : 2]. Thus, the final product terms are obtained by parallel multiplication using sub multiplier blocks and addition as *P*7*P*6*P*5*P*4*P*3*P*2*P*1*P*0, that can reduces the delay of the multiplier.

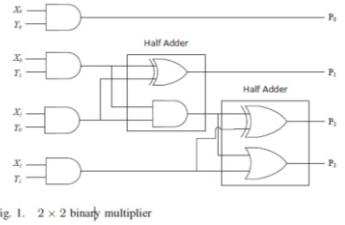

Fig. 1. 2 × 2 binary multiplier

Here for a 2x2 multiplier we only require 4 AND gates ,2 XOR and one OR gates are only required where here one XOR and AND gate makes a half adder her the operation is very simple .Here 2 input bits are given where each input represent a 2bits of data. Here the product is stored at p0,p1,p2and the carry generated is stored at p4.

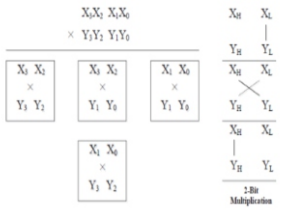

Fig. 2. Block diagram representation of 4 × 4 multiplier and 2 × 2 multiplier

Here coming to 4x4 multiplier it requires 2, 2x2 multiplier blocks where each input represent a 4bits of data and here we obtain 8 binary bits as the output bits. Here the vertical and crosswise multiplication is done we get the final output by decreasing the partial products.

OPERATION:

Vedic Multiplier for 4x4 bit :

Divide the no. of bits in the inputs equally in two parts. Let's analyze 4x4 bit multiplication, say multiplicand A=A3A2A1A0 and multiplier B= B3B2B1B0. Following are the output line for the multiplication result, S7S6S5S4S3S2S1S0. Let's divide A and B into two parts, say "A3 A2" & "A1 A0" for A and "B3 B2" & "B1B0" for B. Using the fundamental of Vedic multiplication, taking two bit at a time and using 2 bit multiplier block, we can have the following structure for 4x4 bit multiplication as shown in fig 2

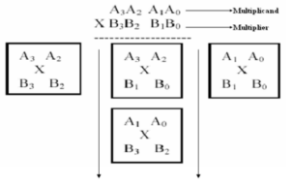

Figure 2: Structure for 4x4 bit Multiplication

Each block as shown above is 2x2 bit multiplier. First 2x2 multiplier inputs are "A1 A0" and "B1 B0". The last block is 2x2 bit multiplier with inputs "A3 A2" and "B3 B2". The middle one shows two, 2x2 bit multiplier with inputs "A3A2" & "B1B0" and "A1A0" & "B3B2". So the final result of multiplication, which is of 8 bit, "S7S6S5S4S3S2 S1S0".

To understand the concept, the block diagram of 4x4 bit Vedic multiplier is shown in Figure 3. To get final product S7S6S5S4S3S2S1S0 four, 2-bit Vedic multiplier and three 4-bit Ripple Carry (RC) Adders are required. In this proposal, the first 4-bit RC Adder is used to add two 4-bit operands obtained from cross multiplication of the two middle 2x2 bit multiplier modules. The second 4-bit RC Adder is used to add two 4-bit operands, i.e. concatenated 4-bit ("00" & most significant two output bits of right hand most of 2x2 multiplier module as shown in Figure 3) and one 4-bit operand we get as the output sum of first RC Adder. Its carry "ca1" is forwarded to third RC Adder. Now the third 4-bit RC Adder is used to add two 4-bit operands, i.e. concatenated 4 -bit (carry ca1, "0" & most significant two output sum bits of 2nd RC Adder as shown in Figure 3) and one 4-bit operand we get as the output sum of left hand most of 2x2 multiplier module. Early literature speaks about Vedic multipliers based on array multiplier structures. The arrangement of Ripple Carry Adder as shown in Figure 6 helps us to reduce delay.

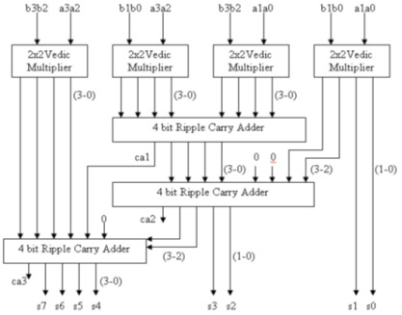

Figure 3: Block Diagram of 4x4 bit Vedic Multiplier (VM)

Vedic 8x8 multiplier:We here use 16,2x2 bit multipliers ,2 4x4 multipliers. Here we give a input of 2bitsof data each having 8-bits as inputs and produces a output of 16 bits of data.

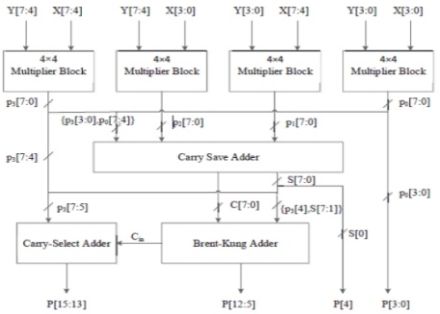

Fig: proposed 8x8 vedic multiplier

To obtain 8 × 8 multiplier block, the multiplicand and multiplier bits are decomposed equally as *XH*=*X7X6X5X4* and *XL*=*X3X2X1X0* for *X* and *YH*=*Y7Y6Y5Y4* and *YL*=*Y3Y2Y1Y0* for *Y* as shown in Fig. 5. The first four final product outputs *P*[3 : 0] are same as that of the partial products *p*0[3 : 0]. The other partial products which includes *p*1[7 : 0],

*p*2[7 : 0] and {*p*3[3 : 0], *p*0[7 : 4]} are added using carry save adder which can generates the sum and carry output bits as *S*[7 : 0] and *C*[7 : 0] respectively. The product output *P*[4] is the output of sum *S*[0]. Then, the sum and carry bits of carry save adder are added by using Brent-Kung adder along with the partial product bit *p*3[4] that generates the final product term *P*[12 : 5] and a carry bit *Cin*.

VEDIC 16X16 MULTIPLIER: In this we propose a two 16 bits of data as input and get an output of 32 bits for it .

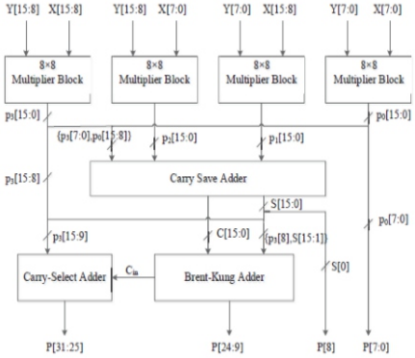

Fig : Proposed 16-bit Vedic Multiplier

We here get an output by multiplying lower order bits of X and Y bits ie; from 0to7 and one Y as lower and X as higher bits and next Y as higher bits and X as lower bits at last X, Y as two lower bits they are multiplied.

The final stage of addition is done by using carry-select adder that contains multiplexers and half adders . If the input of the carry-select adder is Cin = 0, the multiplexer gives the input of the partial product output p3[15 : 9] to the final product terms P[31 : 25] directly without any computation, that can reduces the switching power. If the input carry bit Cin = 1 then, the output of the multiplexer is the addition of the partial product output p3[15 : 9] with the carry bit using half adders. The final carry out bit (Cout) is discarded in the final product.

In the similar way of design, a 16 × 16 Vedic Multiplier is obtained by using 8×8 multiplier blocks, carry save adder, Brent-Kung adder, and

carry select adder .The usage of the fast adder in the proposed design improves the performance in terms of delay, and power-delay product.

Hence, the signal processing can be made faster using the proposed multiplier in Multiplier-Accumulator (MAC) unit,Fast Fourier Transform (FFT), convolution, and filtering. However, the major drawback of the proposed multiplier is that, it requires larger area for computation.

Tools and Hardware:

Software : Vivado Design Suite
Simulation : Vivado Design Suite Isim
Synthesis : Vivado Integrated Design Environment(IDE) Synthesis
Hardware : Zynq Board

- Device XC7Z020 ,
- Package CLG 484,
- Speed-1

ZYNQ Implementation:

ZYNQ BOARDSPECIFICATIONS:

The Zynq board used in this project has the following specifications:

Vendor	: Vivado
Family	: Zynq
Family	: XC7Z020
Package	: CLG484
Speed grade	: -1

SIMULATION RESULTS

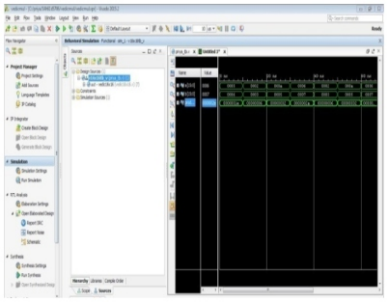

Fig: SIMULATION WAVEFORM

ADVANTAGES:

- Reduction Of Delay.
- Reduction Of Power-Delay Product.
- High Speed as 50%of bit is divided speed get increased.
- Cheap in cost
- Design is easy
- Early understandable

APPLICATIONS:

- DSP
- FFT
- DCT
- Convolution
- Filtering
- Biomedical application
- Mobile and Satellite Communication.
- In computers

CONCLUSION:

The proposed multiplier with fast adders can achieve significant improvement in delay and power-delay product when compared with the conventional multiplier architectures. Here we have given 16 bits as inputs and we get 32 outputs in digital form .Here we have done in vivado 15.2 soft ware and implemented on the z-board and the desired output ifs obtained.

REFERANCE:

[1] T. G. Noll, "Carry-save architectures for high-speed digital signal processing," Journal of VLSI signal processing systems for signal, image and video technology, vol. 3, no. 1-2, pp. 121–140, 1991.
[2] Y. Bansal, C. Madhu, and P. Kaur, "High speed vedic multiplier designsa review," in Engineering and Computational Sciences (RAECS), 2014 Recent Advances in. IEEE, 2014, pp. 1–6.
[3] S. R. Huddar, S. R. Rupanagudi, M. Kalpana, and S. Mohan, "Novel high speed vedic mathematics multiplier using compressors," in Automation, Computing, Communication, Control and Compressed Sensing (iMac4s), 2013 International Multi-Conference on. IEEE, 2013,.