International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

High Speed Vedic Multiplier for 16 Bits Numbers

M. Narasimharao[1], R. V. Shashanka[2]

[1]Department of ECE, Amrita Sai Institute of Science & Technology, Paritala, Andhra Pradesh, India

[2]Assistant Professor, Department of ECE, Amrita Sai Institute of Science & Technology, Paritala, Andhra Pradesh, India

Abstract: ***Speed and occupational area are key in RISCs (Reduced Instruction Set Computers), DSPs (Digital Signal Processors). Knowing that most of the operations involved in processing signal are multiplications since fundamental process in communication "modulation" is multiplication. This paper introducing architecture to perform high speed multiplication using one of the methods "Urdhva-tiryakbhyam" from Vedic maths technique. Here in this a 4:2 compressor, 7:2 compressors and 15:2 compressors technique are used to increase speed. Upon comparison, the compressor based multiplier has improvement in speed and area with some stranded methods like array and booth***

Keywords: Compressor, array, Booth's multiplier, Urdhwa Tiryakbhyam Sutra, Vedic Mathematics

1. Introduction

Stranded methods like array and booth techniques are suffering from lack of speed. Most of the techniques will not have more concentration on effective resource utilization in terms of using less LUT'S or not using a better LUT which is suitable for most of operations. If we cut down number LUT'S usage will decrement architectural space (*area*).

In concern to speed they suffer with one waiting for carry, two large number of intermediate partial products, three lack of n-bit adders led to large consumption time. If there is a method which can produce products with less independent on waiting for carry, intermediate partial products and good n-bit adder can improve operational time of multiplier (*speed*).

Vedic mathematics fulfilling above needs. Vedic mathematics are come to live by one of popular mathematician, sir Bharati Krishna Tirthaji. He explained total techniques into 16 different simple methods known as sutras. In this paper, one method from total 16 namely Urdhva-tiryakbhyam which suitable for our present engineering scenario is selected with additional improvement of compressor for n-bit addition to improve speed as well as reduction in size.

2. Vedic Mathematics

Urdhva-tiryakbhyam

This is one of technique to perform multiplication in easy way taken from ancient Vedic mathematics. This is a Sanskrit word resembles a meaning of urdhva as vertical, tiryakbhyam as crosswise. Here in this method multiplier and multiplicand bit are multiplied, in vertical and crosswise as shown in fig 1 for a 3 bit number

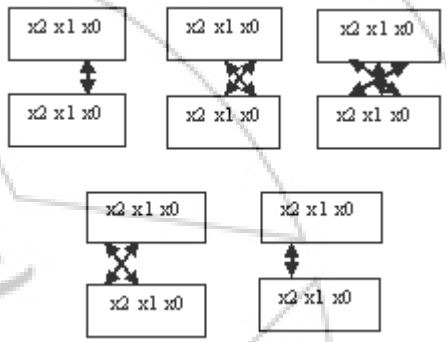

Figure 1: Multiplication process in urdhva tiryakbhyam

In this paper proposed method multiple two 16 bit numbers, which has equations as shown below

p0=a [0]&b[0]----------**(1)**

p1=a [0]&b[1]+a[1]&b[0]----------**(2)**

cp2=a[0]&b[2]+a[1]&b[1] +a[2]&b[0]----------**(3)**

p3=a[0]&b[3]+a[1]&b[2]+ a[2]&b[1]+a[3]&b[0]-**(4)**

p4=a[0]&b[4]+ a[1]&b[3] +a[2]&b[2]+ a[3]&b[1]+ a[4]&b[0]----------**(5)**

cp5=a[0]&b[5]+ a[1]&b[4] +a[2]&b[3] +a[3]&b[2] +a[4]&b[1] +a[5]&b[0] ----------**(6)**

p6=a[0]&b[6]+ a[1]&b[5] +a[2]&b[4] +a[3]&b[3] +a[4]&b[2] +a[5]&b[1] +a[6]&b[0]----------**(7)**

p7=a[0]&b[7] +a[1]&b[6] +a[2]&b[5] +a[3]&b[4] +a[4]&b[3]+a[5]&b[2] +a[6]&b[1] +a[7]&b[0]--**(8)**

p8=a[0]&b[8] +a[1]&b[7] +a[2]&b[6] +a[3]&b[5]+a [4]&b[4] +a[5]&b[3] +a[6]&b[2] +a[7]&b[1]+a[8]& b[0] ----------**(9)**

p9=a[0]&b[9] +a[1]&b[8] +a[2]&b[7] +a[3]&b[6] +a[4]&b[5] +a[5]&b[4] a[6]&b[3] +a[7]&b[2] +a[8]&b[1] +a[9]&b[0]---------- **(10)**

p10=a[0]&b[10]+ a[1]&b[9] +a[2]&b[8] +a[3]&b[7] +a[4]&b[6] +a[5]&b[5] +a[6]&b[4] +a[7]&b[3] +a[8]&b[2] +a[9]&b[1]+a[10]&b[0]---------- **(11)**

p11=a[0]&b[11]+a[1]&b[10] +a[2]&b[9] +a[3]&b[8] +a[4]&b[7] +a[5]&b[6] +a[6]&b[5] +a[7]&b[4]+ a[8]&b[3] +a[9]&b[2] +a[10]&b[1] +a[11]&b[0]**(12)**

p12=a[0]&b[12]+ a[1]&b[11] +a[2]&b[10] +a[3]&b[9] +a[4]&b[8] +a[5]&b[7] +a[6]&b[6] +a[7]&b[5] +a[8]&b[4] +a[9]&b[3] +a[10]&b[2] +a[11]&b[1] +a[12]&b[0] ----------**(13)**

p13=a[0]&b[13] +a[1]&b[12]+ a[2]&b[11] +a[3]&b[10] +a[4]&b[9] +a[5]&b[8] +a[6]&b[7] +a[7]&b[6] +a[8]&b[5] +a[9]&b[4] +a[10]&b[3] +a[11]&b[2] +a[12]&b[1] +a[13]&b[0]---------**(14)**

p14=a[0]&b[14]a[1]&b[13]+a[2]&b[12]+a[3]&b[11] +a[4]&b[10] +a[5]&b[9] +a[6]&b[8] +a[7]&b[7] +a[8]&b[6] +a[9]&b[5] +a[10]&b[4] +a[11]&b[3] +a[12]&b[2] +a[13]&b[1]+a[14]&b[0]----------**(15)**

p15=a[0]&b[15]+ a[1]&b[14] +a[2]&b[13] +a[3]& b[12] +a[4]&b[11] +a[5]&b[10] +a[6]&b[9] +a[7] &b[8] +a[8]&b[7]+a[9]&b[6]+a[10]&b[5]+a[11] &b[4] +a[12]&b[3] +a[13]&b[2] +a[14]&b[1] +a[15]&b[0]----------**(16)**

p16=a[15]&b[1] a[14]&b[2]+ a[13]&b[3] +a[12]& b[4] +a[11]&b[5]+a[10]&b[6] +a[9]&b[7] +a[8]& b[8] +a[7]&b[9] +a[6]&b[10]+a[5]&b[11]+a[4] &b[12] +a[3]&b[13] +a[2]&b[14] +a[1]&b[15] -**(17)**

p17= a[15]&b[2] a[14]&b[3] +a[13]&b[4] +a[12]& b[5] +a[11]&b[6] +a[10]&b[7] +a[9]&b[8]+a[8]& b[9] +a[7]&b[10]+a[6]&b[11]+a[5]&b[12]+a[4]& b[13] +a[3]&b[14] +a[2]&b[15] ----------**(18)**

p18=a[15]&b[3] +a[14]&b[4] + a[13]&b[5] +a[12]&b[6]+a[11]&b[7] +a[10]&b[8] +a[9]&b[9] a[8]&b[10] +a[7]&b[11] +a[6]&b[12] +a[5]&b[13] a[4]&b[14]+a[3]&b[15]----------**(19)**

p19= a[15]&b[4] +a[14]&b[5] +a[13]&b[6] +a[12]&b[7] +a[11]&b[8] +a[10]&b[9]+ a[9]&b[10] +a[8]&b[11] +a[7]&b[12] +a[6]&b[13] + a[5]&b[14] +a[4]&b[15]+ ----------**(20)**

p20= a[15]&b[5]+ a[14]&b[6] +a[13]&b[7] +a[12] &b[8] +a[11]&b[9]+a[10]&b[10]+a[9]&b[11]+a[8] &b[12]+a[7]&b[13] +a[6]&b[14]+a[5]&b[15]---**(21)**

p21=a[15]&b[6] +a[14]&b[7] +a[13]&b[8] +a[12] &b[9]+a[11]&b[10]+ a[10]&b[11] +a[9]&b[12] +a[8]&b[13] +a[7]&b[14] +a[6]&b[15]---------**(22)**

p22=a[15]&b[7] +a[14]&b[8] +a[13]&b[9] +a[12] &b[10] +a[11]&b[11] +a[10]&b[12] +a[9]&b[13] +a[8]&b[14] +a[7]&b[15]---------- **(23)**

p23= a[15]&b[8]+ a[14]&b[9] +a[13]&b[10] +a[12] &b[11] +a[11]&b[12] +a[10]&b[13] +a[9]&b[14] +a[8]&b[15]----------**(24)**

p24=a[15]&b[9] +a[14]&b[10] +a[13]&b[11] +a[12] &b[12] +a[11]&b[13] +a[10]&b[14] +a[9]&b[15]---------- ----------**(25)**

-

p25=a[15]&b[10] +a[14]&b[11] +a[13]&b[12] +a[12]&b[13] +a[11]&b[14] +a[10]&b[15]+ ----**(26)**

p26=a[15]&b[11] +a[14]&b[12] +a[13]&b[13] +a[12]&b[14] +a[11]&b[15]---------- **(27)**

p27= a[15]&b[12]+ a[14]&b[13] +a[13]&b[14] +a[12]&b[15]----------**(28)**

p28=a[15]&b[13] +a[14]&b[14] +a[13]&b[15]--**(29)**

p29=a[15]&b[14]+a[14]&b[15]----------**(30)**

p30=a[15]&b[15]----------**(31)**

3. Compressor Adder

3.1 Compressors

A compressor adder is a logical circuit which can add 4 bits at a time with previous carry which increases speed of operations. Here half adder and full adder are replaced with compressor for the addition of partial products which plays key role to enhance speed of operations. Block diagram for it is as shown in fig 2. and architecture is as shown in Fig: 3. proposed architectures an equivalent circuit, using full adders and half adders is as shown in fig 4.

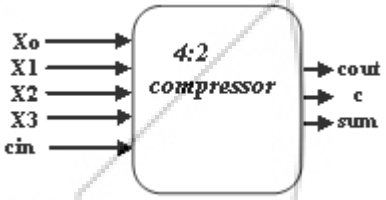

Figure 2 4:2: compressors: block diagram

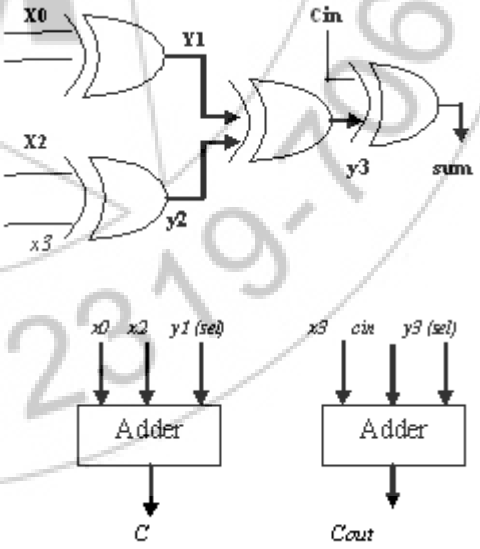

Figure 3 4:2: Compressors proposed architecture:

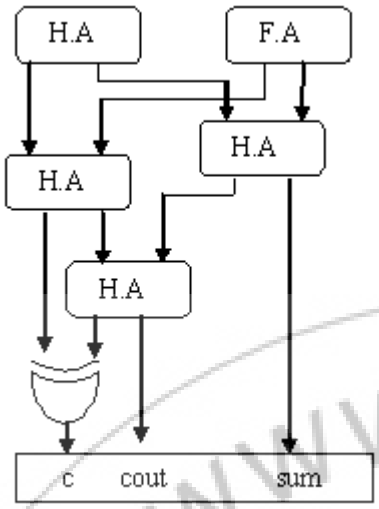

Figure 4 4:2: Compressors with half and full adders:

Let us consider the propagation delay of a gate to be 1ns. It is well known that a full adder has a total propagation delay of 2tp and a half adder has a propagation delay of tp. Considering this, the total propagation delay of a 4:2 adder using full adders and half adders can be calculated as 5tp and can be seen in Fig.4. On the other hand, it can be seen from Fig. 3. that the propagation delay of a 4:2 compressor remains only 3tp.Therefore, a 66.6% increase in speed can be recorded in comparison with an equivalent circuit made of full and half adders, proving to be a highly efficient architecture for addition.

3.2 Compressors

Similar to its 4:2 compressors, the 7:2 compressors as shown in Fig. 5. is capable Of

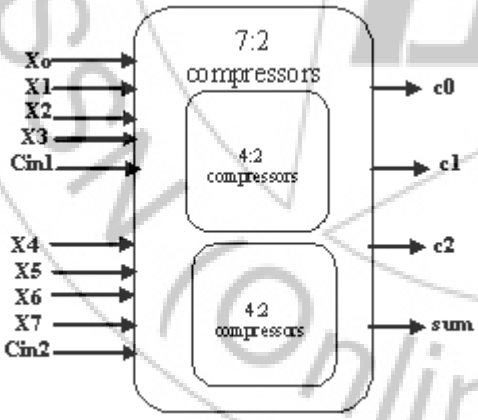

Figure 5 7:2: compressors: block diagram

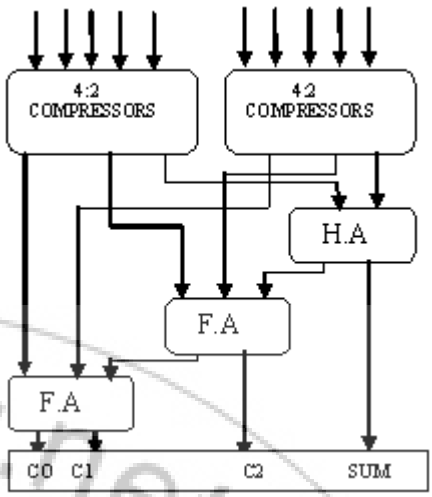

Figure 6 7:2: Compressors Proposed Architecture

Adding 8 bits of input and 2 carry's from the previous stages, at a time. It is constructed from two 4:2 compressors block diagram is show in fig: 5 and architecture is show in fig: 6.

15:2 Compressors:

Similar to its 4:2 and 7:2 compressors, the 15:2 compressors as shown in Fig. 7, is capable of adding 16 bits of input and 4 carry's from the previous stages, at a time. It is constructed from two 7:2 compressors block diagram is show in fig: 7 and architecture is show in fig: 8

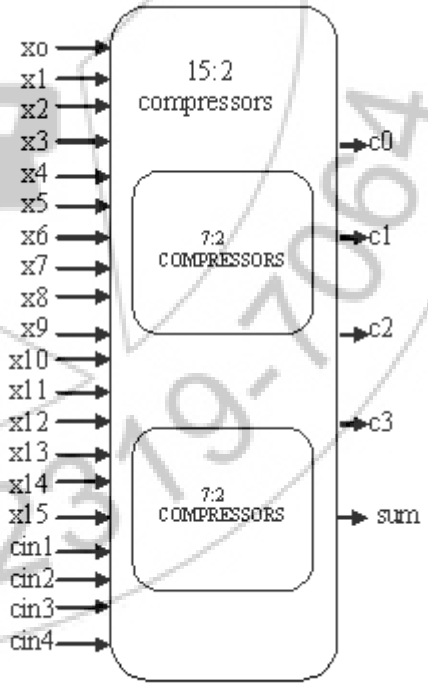

Figure 7 15:2: compressors: block diagram

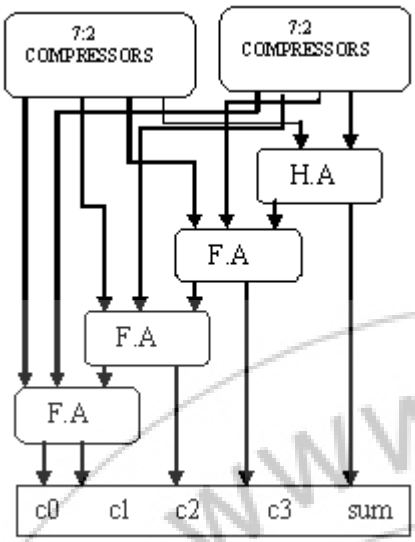

Figure 8 15:2: compressors proposed architecture:

4. Compressor Based Urdhwa Multiplier

As mentioned in Section II, the multiplier based on Urdhwa method of multiplication requires several full adders and half adders to add the necessary partial products. This in turn leads to a large propagation delay due to the reasons explained in the previous section. The compressor based Urdhwa multiplier requires only 12 parallel stages as opposed to 15 which was in the case of the conventional Urdhwa Tiryakbhyam multiplier. This is a major improvement with respect to high speed multiplier design. Also, it can be seen that, many of the stages have now been reduced to a logical XOR operation, with an initiative to reduce area. An analysis on the area and speed occupied by the new design is compared with other multiplication has been presented in the next section.

5. Results

In order to perform a comparison, Urdhwa multiplier and the compressor based Urdhwa multiplier were implemented on a Xilinx Spartan 3e – XC3S100E FPGA using Verilog as the RTL language with the help of Xilinx Project Navigator 10.1. The codes were synthesized. Optimized speed and area parameters were compared. The Spartan 3e FPGA used for the experiments has a speed grade.

It can be clearly noted from Table I., that in term of speed, the compressor based Vedic maths multiplier performs exceptionally well and is nearly 1.5 times faster than the existing Vedic maths based multiplier. This method also provided an improvement of area up to 3%.

Table 1: comparison of area occupied and speed of various multiplier architectures

Algorithm	% of area occupied	Time (ns)
Array[5]	21%	217.8
Radix-2 modified Booth algorithm[5]	37%	39.69
Urdhwa- Tiryakbhyam	7%	30.51
Compressor based Urdhwa Tiryakbhyam with full and half adder	6%	27.26
Compressor based Urdhwa Tiryakbhyam with proposed architecture	5%	26.51

6. Conclusion

In this paper, we have proposed high speed Architecture for multiplication of two 16 bit numbers, combining the advantages of compressor based adders and also the ancient Vedic maths methodology. New 15:2 compressor architecture, based on 7:2 compressor architecture was also discussed. We can conclude that the compressor based Vedic maths multiplier proves to be a better option over conventional multipliers used in several expeditious and complex VLSI circuits. As a future work, the multiplier's performance could be tested within an ALU and also compared with several other existing multipliers.

References

[1] "High speed asic design of complex Multiplier using Vedic mathematics" IEEE, Proc.vol. 978-1-4244-8943-5/11

[2] "Novel high speed Vedic mathematics multiplier using compressors mathematics" IEEE, Proc.vol. 978-1-4673-5090-7/13

[3] **"**Design and FPGA implementation of compressor based Vedic multiplier" 2nd International conference on science, Engineering and management, Srinivasa Engineering college, Tamil Nadu, India, March 28-29,2014

[4] "High speed vedic multiplier" international journal of engineering research volume no.3 issue no: special 2, pp: 73-76, 22 march 2014

[5] Efficient implementation of 16-bit Multiplier-accumulator using radix-2 modified booth algorithm and spst adder using verilog. by addanki purna ramesh, Dr.a.v. n. tilak and Dr.a.m.prasad, international journal of vlsi design & communication systems (vlsics) vol.3, no.3, june 2012

[6] "Low Power High Speed 16x16 bit Multiplier using Vedic Mathematics" International Journal of Computer Applications (0975 – 8887) Volume 59– No.6, December 2012

[7] Design and implementation of Vedic multiplier international journal of engineering research and development" e-issn: 2278-067x, p-issn: 2278-800x, volume 8, issue 6 (September 2013), pp.23-28.

[8] "Implementation of Multiplier using Vedic Algorithm" by Poornima M, Shivaraj Kumar Patil, Shivukumar , Shridhar K P , Sanjay H International Journal of Innovative Technology and Exploring Engineering

(IJITEE) ISSN: 2278-3075, Volume-2, Issue-6, May 2013

[9] "Implementation of power efficient vedic multiplier" international journal of computer applications volume 43– no.16, April 2012

[10] " Design of High Speed Vedic Multiplier using Vedic Mathematics Techniques" by G.Ganesh Kumar, V.Charishma, International Journal of Scientific and Research Publications, Volume 2, Issue 3, March 2012 1 ISSN 2250-3153

[11] Tiwari, Honey Durga, et al., "Multiplier design based on ancient Indian Vedic Mathematics, " Int. SoC Design Conf., 2008, vol. 2. IEEE Proc., pp. II-65 - II-68.

[12] D. Radha krishnan and A. P. Preethy, "Low power CMOS pass logic 4-2 compressor for high-speed multiplication," Circuits and Systems, Proc. 43rd IEEE Midwest Symp., vol. 3, pp. 1296-1298, 2000.

Author Profile

M. Narasimharao, M. Tech student in amrita sai institute of science and technology, Parital. Andhra Pradesh

R. V. Shashanka ,Assistant Professor in Dept. of ECE Amrita Sai Inst.of Sci. & Tech., Paritala, Andhra Pradesh