DOI 10.29042/2019-5275-5281

Helix Vol. 9 (5): 5275- 5281

Novel Method to Compute Cube Confirming Low Device Utilization on FPGA

[1]Avinash Patil, [2]S C Patil

*[1]Research Scholar, E&TC Department, Rajarshi Shahu College of Engineering, Pune, MS-India

[2]Professor, Dept of IT, Rajarshi Shahu College of Engineering, Pune, MS-India

Email: [1]avispatil@yahoo.co.uk, [2]Shailaja.patil11@gmail.com

Received: 13th September 2019, Accepted: 30th September 2019, Published: 31st October 2019

Abstract

The contribution made by authors in a calculation of a cube of a number includes the digital architectures leading to a computation of cube of an 8- bit binary number which produces the result with reduced hardware utilization on FPGA. The cube is a prominent operation required in many mathematical operations, signal and digital image processing applications. The cube also required in many cryptographic applications as well as trigon arithmetic operations such as taylor series and Mclauren series. The conventional method of finding the cube of a number is faster but required dedicated multiplier block on FPGA and or a large amount of digital gate level resource. The architectural novelty is proposed which can be used at gate level as well as algorithmic level. The cube proposed by authors include features of Vedic mathematics Anurupya sutra, Vertical and crosswise sutra also used, leading to a new better architecture. The cube operation is carried out in VHDL and is compared over conventional as well as Vertical and crosswise based multiplier architectures and provide out better in terms of area and speed optimization. The targeted device used is Spartan XC3S400PQ208 on Xilinx platform.It is observed that the area optimization is achieved with proposed architecture at 2131 gate count compared over 2250 with Vertical and crosswise based cube operation and 8000 gate count with conventional cube calculation method whereas speed enhancement registered over Vertical and crosswise method as proposed method reports 38.348 ns and second one report 43.309 ns.

Keywords

Vedic Mathematics, Vertical and Crosswise, Duplex Method, Anurupya Sutra, Cube of a Number, VHDL.

Introduction

The cube of a number is mathematically its third power, means if b is a number then its cube is the number b multiplier by itself three times. And the result is represented as given in equation 1.

$b^3 = b \times b \times b.$ (1)

And the same is calculated as a square of a number multiplied by b directly as given in equation 2.

$b^3 = b \times b^2.$ (2)

The same equations is also a formula for calculating the volume of a cuboid shape with having length, breadth, and height as b. Cube is an odd function as its output is as given in equation 3.

$(-b)^3 = -(b^3).$ (3)

As the cube is an odd function it gives the graph which is a parabola in nature and is plotted against b verses b^3 as shown in Fig 1. The graph shows that the curve has no axis of symmetry.

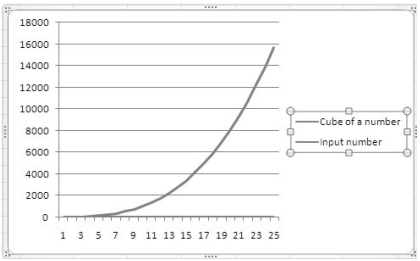

Fig 1: Cube Parabola

A positive integer b is a perfect cube in geometrical point of view when its possible to arrange b solid cube units into a larger cube.

Example, 27 small cubes can make a large cube as $3 \times 3 \times 3 = 27$. The consecutive cube numbers differce by expressed as follows:

$b^3 - (b-1)^3 = 3(b-1)b + 1$. Or $(b+1)^3 - b^3 = 3(b+1)b + 1$.

 Helix E-ISSN: 2319-5592; P-ISSN: 2277-3495

Finding cube of a number is an arithmetic operation which is today part of many digital systems. The application of cube of a number is in digital signal processing applications which include pattern recognition, image compression and so on[7]. Cube operation is also used in cryptography algorithms, graphics processors, The cube calculation is required in Taylors series and Mclaurens series for calculation of Trigon aritmetics[7].

Methodolgy

1. Conventional Method: Parallel Cube Algorithm

Almost in all the calculations, the multiplier unit is used to calculate the cube of a number so it may be in processors, DSP, DIP applications or in encryption-decryption operations. As the cube is required very frequently instead of using a multiplier to perform the operation it is best to use dedicated cube hardware which improves the speed of operations and reduction in power consumption significantly.

A parallel multiplier unit was based on Vertical and crosswise sutra(VC) of Vedic mathematics (Vertical and Crosswise) proposed in [3] [14] having reduced area and higher speed of operation[4][5]. With the use of Vertical and crosswise (UT)[1][2] method of multiplication finding the cube of an 8- bit number is already a faster approach. Thus, there is a significant reduction in the device utilization, Further, a cube can be computed much faster than a cube with a normal multiplier. The standard multipliers considered are booths multiplier, array multiplier[10][12].

2. Proposed Algorithm for Calculation of Cube of a Number

To calculate the cube of a given 8 bit number the anurupya sutra[8] of Vedic mathematics is applied on binary numbers. The proposed cube is based on the Anurupya Sutra of Vedic Mathematics which states "If you start with the cube of the first digit and take the next three numbers(in the top row) in a Geometrical Proportion (in the ratio of the original digits themselves) you will find that the 4th figure (on the right end) is just the cube of the second digit". The algebraic explanation is as follows: If a and b are two digits, then according to Anurupya Sutra[7][8].,

$(ab)^3=$

$a^3+a^2b+ab^2+b^3$ $2a^2b+2ab^2$
$a^3+3a^2b+3ab^2+b^3$

which is equal to $(a+b)^3$ and is used in the calculation of cube of a number. The number ab which is 8- bit is split into two 4 bits to calculate by the above method, as a and b. The Anurupya Sutra is applied to find the cube of the number. In the algebraic explanation of the Anurupya Sutra, it is required to calculate a^3 and b^3 and $2a^2b$ and $2ab^2$. An example of how Anurupya sutra works is as given below.

$(15)^3=3375$

$(15)^3=$	1	5	25	125	
		10	50		
	3	3	7	5	

$(23)^3=12147$

$(23)^3=$	8	12	18	27	
		24	36		
	12	1	4	7	

The cube architecture proposed is proveing an excellent reduction in hardware over multipliers with conventional method or event using Vedic multiplier in which the n bit multiplication can be performed. In continuation with the Vedic Anurupya sutra based method of finding the cube of a number, adding a VC based multiplier gives added advantage and anurupya sutra based method having a base multiplier as a VC multiplier is totally a new way to compute the cube of a number. Vertically and crosswise is the general formula applicable to all cases of multiplication of a large number. Consider multiplicand"s digits be a, b, c and d. This is multiplied with p, q, r. The steps involved in a calculation of vertical and crosswise multiplication also called Vertical and crosswise based multiplier is as shown in Fig no 2 and 3. Fig no 4 shows the 4 bit multiplier based on Vertical and crosswise.

  Helix E-ISSN: 2319-5592; P-ISSN: 2277-3495

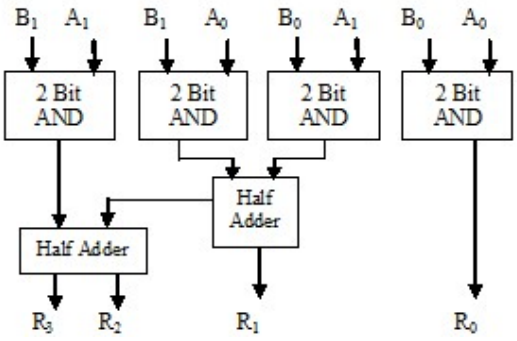

Fig 2: Architecture of Two Bit VC Multiplier

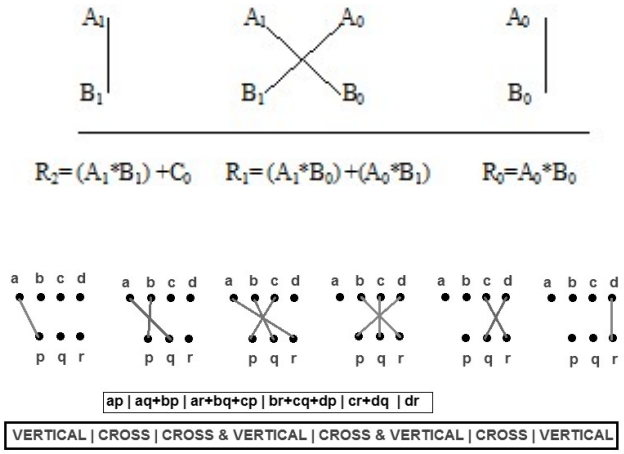

Fig 3: Urdhva Tiryakbhyam Multiplier also called Vertical and Crosswise (VC) Multiplier

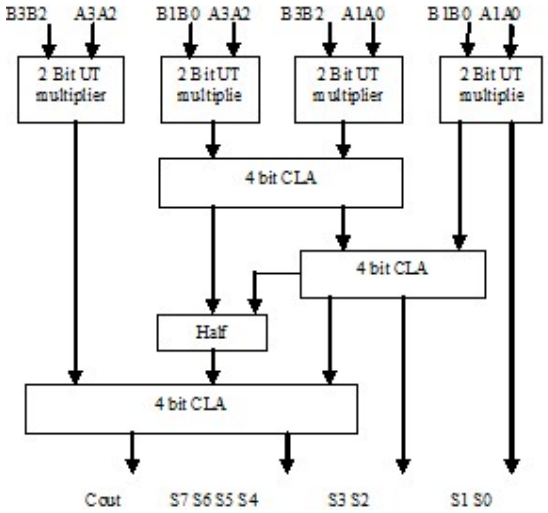

Fig 4: 4 BIT Vertical and Crosswise (VC) Multiplier

The VHDL code is written for the algorithms of calculation of cube of a eight bit number based on conventional multiplication and Vedic mathematics VC based multiplier to calculate cube and Anurupya sutra based with inclusion of VC based multiplier to calculate cube and VHDL coding is carried on Xilinx tool for a targeted

  Helix E-ISSN: 2319-5592; P-ISSN: 2277-3495

device Spartan3 XC3S400PQ208. The coding style used is mixed mode style of modeling and results are generated module wise and then connected in an architecture. The results were generated are post implemented synthesis results. Fig No. 5 shows the flowchart for calculation of cube of a 8- bit number based on Vedic mathematics Anurupya sutra with VC multiplier. The proposed work and its algorithm are coded in VHDL on Xilinx tool suite and post implementation simulation results were generated using for a targeted device Spartan3 XC3S400PQ208. The design is optimized for area and speed using Xilinx

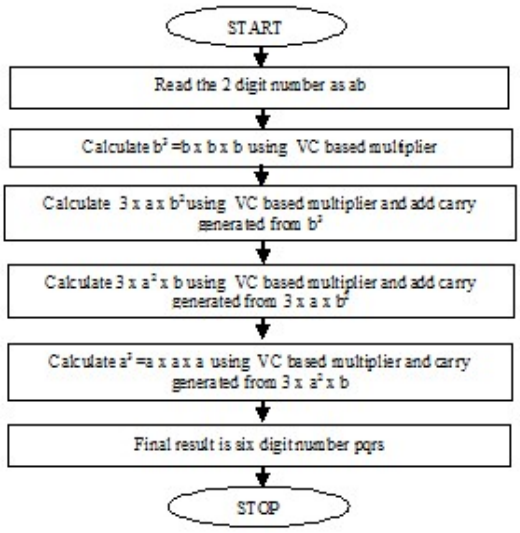

Fig 5 : Flowchart of Calculation of Cube.

Results and Discussion

The results are tabulated below in Table no 1 for Cube operation of a 8-bit number based on a conventional method which shows the device utilization in terms of the number of LUTs and Multipliers utilized.

Logic Distribution	Used	Available
No of Bonded IO	32	141
Number of MULT18X18	2	16
Total equivalent gate count	8000	--
JTAG ggate count for IOBs	1536	--

Table 1: Device Utilization for Finding Cube of an 8- bit Number using a Conventional Method

Fig no 6 shows the simulation result,

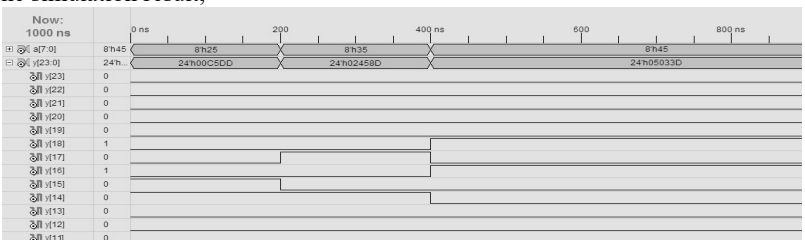

Fig. 6: Simulation Result by Conventional Method

Fig no 7 shows the RTL generated by the code.

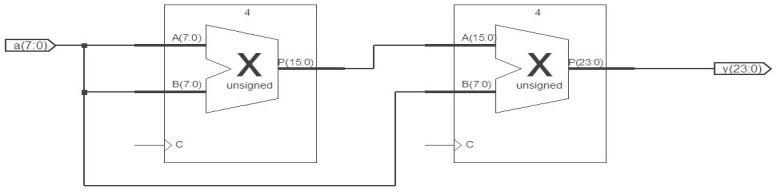

Fig. 7: RTL Generated by Conventional

  Helix E-ISSN: 2319-5592; P-ISSN: 2277-3495

The results are tabulated below in Table no 2 for Cube of a 8- bit number based on Vedic mathematics VC based multiplier which shows the device utilization in terms of number of LUTs utilized.

Logic Distribution	Used	Available
No of Bonded IO	32	141
Number of MULT18X18	--	16
Total equivalent gate count	2250	--
JTAG ggate count for IOBs	1536	--

Table 2: Finding Cube of a 8- bit Number using VC Based Multiplier

Fig no 8 shows the simulation result,

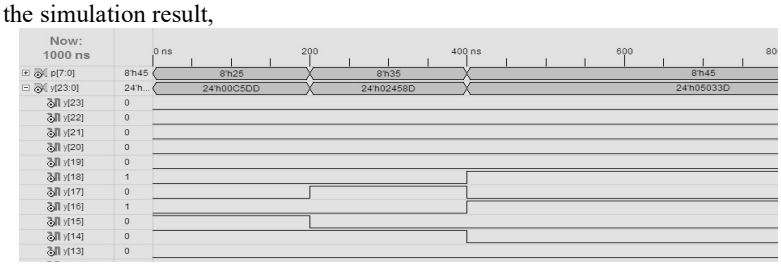

Fig. 8: Simulation Result of Cube by VC Multiplier Method

Fig no 9 shows the RTL generated by the code.

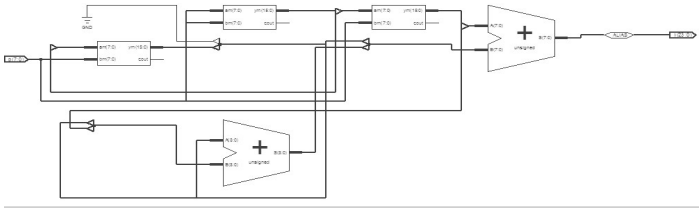

Fig. 9: RTL Generated by VC Multiplier Method

The results are tabulated below in Table no 3 for Cube of a 8 bit number based on Vedic mathematics Anurupya sutra based cube operation with incorporating the VC based multiplier which shows the device utilization in terms of number of LUTs

Logic Distribution	Used	Available
No of Bonded IO	32	141
Number of MULT18X18	--	16
Total equivalent gate count	2131	--
JTAG ggate count for IOBs	1536	--

Table 3: Finding Cube of a 8- bit Number Using Anurupya Sutra and VC Based Multiplier

while Fig no 10 shows the simulation result,

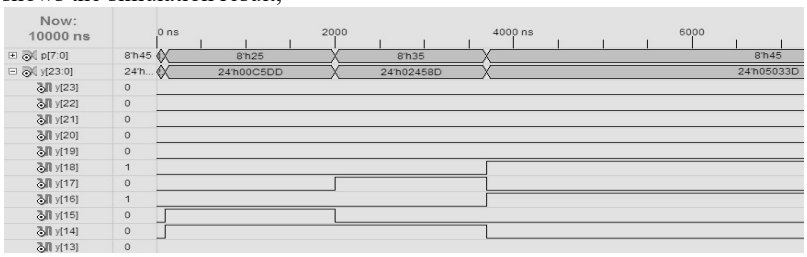

Fig. 10: Simulation Result for Cube Operation by Anurupya Sutra

Fig no 11 shows the RTL generated by the code.

  Helix E-ISSN: 2319-5592; P-ISSN: 2277-3495

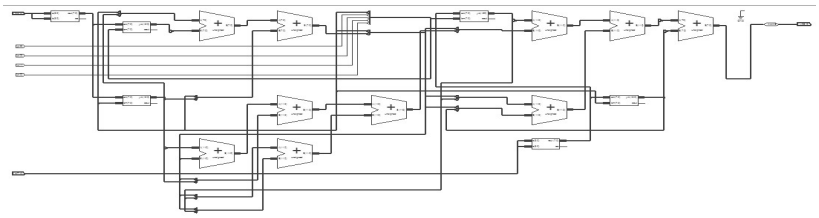

Fig. 11: RTL Generated for Cube Operation by Anurupya Sutra

The final Table no 4 and Fig no 12 concludes the comparison of cube calculation based on conventional and with the inclusion of VC based multiplier and a proposed method, where proposed method gives the best result when compared with the parameters such as are optimization. And also provides better speed when comparing with a 2nd method of calculation of cube.

Logic Distribution	Method 1	Method 2	Proposed method
LUT's Used	--	367	289
MULT18X18	2	--	--
Equivalent gate count	8000	2250	2131
Delay in ns	14.630	43.309	38.348

Table 4: Comparision Analysis between Cube Operation by Method 1: Conventional , Method 2: VC Based Cube Operation and Method 3: Proposed Anurupya Sutra Based Cube Operation

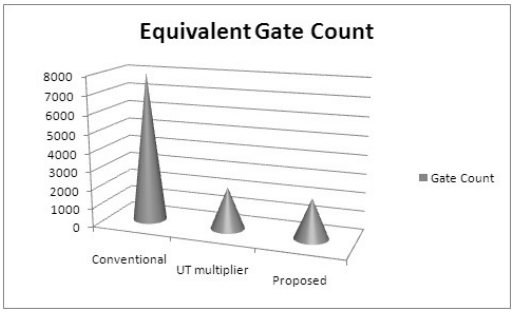

Fig. 12: Result Comparison

Conclusion

The cube operation implemented on Xilinx platform for a targeted device Spartan3 XC3S400PQ208. And the results were generated for conventional method, cube calculation with VC based multiplier and cube calculation by Anurupya sutra with VC based multiplier. The results obtained are compared and it is reported as the proposed method is proved better compared with rest two methods when it comes to device utilization and for speed provides marginal benefits.

The area optimization is achieved with proposed architecture at 2131 gate count compared over 2250 with Vertical and crosswise based cube operation and 8000 gate count with conventional cube calculation method whereas speed enhancement registered over Vertical and crosswise method as proposed method reports 38.348 ns and second one report 43.309 ns.

References

[1]. R.Bhaskar, Ganapathi Hegde, P.R.Vaya,"An efficient hardware model for RSA Encryption system using Vedic mathematics". Procedia Engineering Volume 30, 2012, Pages 124–128, SciVerse Science Direct, ELSEVIER.

[2]. Devika Jaina, Kabiraj Sethi, Rutuparna Panda, "Vedic Mathematics Based Multiply Accumulate Unit", 978-0-7695-4587-5/11 $26.00 © 2011 IEEE.

[3] Avinash Patil1, Y V Chavan[2] , Sushma Wadar[3] "Performance analysis of Multiplication operation based on Vedic mathematics" 21st ,22nd Oct 2016, Allahabad, IEEE conference

[4]Avinash Patil1, Dr Shailaja Patil2,Y V Chavan[2] , Sushma Wadar[3] "Division operation based on Vedic Mathematics", 2nd -3rd Dec 2016 at RSCOE JSPMS, IEEE conference

[5]. M. Ramalatha, K. Deena Dayalan, P. Dharani, S. Deborah Priya, "High Speed Energy Efficient ALU Design using Vedic Multiplication Techniques", 978-1-4244-3834-1/09, 2009 IEEE.

  Helix E-ISSN: 2319-5592; P-ISSN: 2277-3495

[6]. Honey Durga Tiwari, Ganzorig Gankhuyag, Chan Mo Kim, Yong Beom Cho, "Multiplier design based on Ancient Vedic Mathematics",978-1-4244-2599-0/08/$25.00 © 2008, IEEE.
[7]. Himanshu Thapliyal, Saurabh Kotiyal and M. B Srinivas, "Design and Analysis of A Novel Parallel Square and Cube Architecture Based On Ancient Indian Vedic Mathematics", Centre for VLSI and Embedded System Technologies, International Institute of Information Technology, Hyderabad, 500019, India, 2005 IEEE.
[8]. Jagadguru Swami Sri Bharati Krishna Tirthji Maharaja,"Vedic Mathematics", MotilalBanarsidas, Varanasi, India, 1986, Book.
[9] Prabir Saha, Arindam Banerjee, Partha Bhattacharyya, Anup Dandapat ""High Speed ASIC Design of Complex Multiplier Using Vedic Mathematics" Proceeding of the 2011 IEEE Students' Technology Symposium 14-16 January, 2011, lIT Kharagpur
[10] Y.-H. Seo, D.-W. Kim, A. New VLSI, Architecture of Parallel Multiplier– Accumulator Based on Radix-2 Modified Booth Algorithm, IEEE Trans. Very Large Scale Integration. (VLSI) Syst. 18 (2) (2010) 201–208.
[11] Stuart F. Obermann and Michael J. Flynn, "Division algorithms and implementations," IEEE Transactions on Computers, 46(8):833–854, August 1997.
[12] J. Hu, L. Wang, T. Xu, A low-power adiabatic multiplier based on modified Booth algorithm, in: Proceedings of the IEEE International Symposium on Integrated Circuits, Singapore, September 2007, pp. 489–492.
[13] C. M. Kim, "Multiplier design based on ancient indian Vedic mathematics," IEEE, vol. 11, pp. 3686–3689, 2008.
[14] S. Akhter, VHDL implementation of fast N N multiplier based on vedic mathematic, in: Proceedings of the IEEE, Eighteenth European Conference on Circuit Theory and Design, Seville, August 2007, pp. 472–475.
[15] T. Prabakar, "Design and fpga implementation of binary squarer using Vedic mathematics," IEEE, 2013.

  Helix E-ISSN: 2319-5592; P-ISSN: 2277-3495