

VLSI Design of a Squaring Architecture Based on Yavadunam Sutra of Vedic Mathematics

A.Deepa¹,

¹Assistant Professor, Department of ECE,
J.K.K.Munirajah College of Technology,
Erode, Tamilnadu, India.
adeepaeaswaramoorthy@gmail.com

Dr.C.N.Marimuthu²

² Prof. & Dean Department of ECE, Nandha
Engineering College, Erode, Tamilnadu,
India.
muthu_me2005@yahoo.co.in

Abstract— In this digital epoch, the thirst for high speed is fulfilled by the accomplishment of digital multipliers. Multipliers play a fundamental role in many high-speed applications where the complex multiplications are carried out by squaring operations. Vedic Mathematics is a part of Atharva Veda which deals with the easiest methodology for all types of arithmetic calculations. Yavadunam is one of the

squaring algorithms of Vedic Mathematics. But currently, there is a lack of implementation hardware for Yavadunam for squaring binary numbers. In this paper, a squaring architecture is designed by implementing Yavadunam algorithm. The design is simulated and realized with the help of Xilinx Spartan kit.

Keywords— *Squaring, Vedic Mathematics, Yavadunam Sutra*

1 INTRODUCTION

Squaring architecture plays an essential task in many fields like Adaptive filtering, Cryptography, image compression etc. [3]. In the era of the wireless world, everything around us is getting digital. Consequently, our focus is towards the speed rather than other facets [2]. Due to the rising trend in high-speed applications, the researchers focus on squaring operations [3]. On the whole, the squaring operation is conceded out by the multiplier. However, the design of a multiplier entails large area and consume a considerably huge amount of power for every calculation. The hardware implementation of a squaring architecture for binary numbers is emphasized in this paper. Here only a single Vedic multiplier is used as an alternative of four multipliers for calculating the square of N bit binary numbers. The architecture is designed based on Yavadunam sutra, a squaring sutra of an ancient Indian Vedic Mathematics.

The Vedic Mathematics calculation is carried out based on one of the 16 sutras and the 13 up sutras, which is word formulae depicting standard methods of solving an

entire scope of Mathematical problems [8]. The outcome of the realization of Vedic Mathematics in the design of a binary squaring architecture is the increase in speed with reduced delay when compared with the conventional one. The implementation is carried out with the aid of Xilinx Spartan 3E Kit.

2 VEDIC MATHEMATICS

The word ‘Veda’ means an unconstrained storehouse of all knowledge. The Vedas are four in number Rig, Yajur, Sama and Atharva. Also, they have four upavedas Ayurveda, Gandharvaveda, Dhanurveda and Sthapatyaveda, all of which form an undividable corpus of divine facts as it once was and as it may be exposed. The upaveda ‘sthapatya’ or engineering includes all sorts of architectural and structural human endeavor and all visual craftsmanship. Arithmetic or Mathematics falls under this class. These 16 sutras frame some portion of a ‘Parisista’ of the Atharva Veda [8].

Vedic Mathematics is a souvenir given to us by the ancient sages of India. When compared to the

mathematics of today's era it makes all the arithmetic calculations easy and simpler in a pleasing way. Between 1911 and 1918 Sri. Bharati Krishna Tirthaji Maharaj revived this ancient method of computation from the Vedas. The most significant benefit of this mental mathematics is that the learner can formulate their method and they are not restricted to one technique. This creates inventive, fascinated and intellectual pupils. Tirthaji developed methods and techniques for amplifying the squaring method can without much of a stretch be turned around to create one-line square root of a decimal number. Vedic science is exceptionally direct to learn.

3 RELATED WORK

To obtain the square of the binary number multipliers like Braun Array, Baugh Wooley methods of two's complement, Booth multiplier, Wallace trees, Dadda, etc are incredible [6]. However, currently they happen to be outmoded and many innovative techniques are emerging for even faster operations. In [6] Urdhva Tiryagbhyam Sutra is used to design a high-speed Vedic multiplier. They have concentrated on delay reduction for speedy operation. [7] proposed a method of squaring with the help of Urdhva Triyambhakam algorithm. The realization has been made with the simple gate and adder circuits which leads the squaring unit to be speedy. This squaring unit is compared with the conventional multipliers and proves to save area and provides faster computational speed. [7] also mean that the designing of Vedic multiplier using array multiplier provides less delay and are treated as high-speed multipliers as compared to Booth algorithm using recorded multipliers and Wallace trees [6]. In [5] a simple way of squaring using Peasant multiplication is presented. Here one of the multiplicands generally the larger one is decomposed into the sum of powers of two and a table is created for doubling the second multiplicand. [4] used dwandwa yoga property of Urdhvatriyagbhyam Sutra to design a squaring architecture. Since now there has been no hardware or architecture available for the squaring sutra, Yavadunam[11]. This made us work on this sutra and design architecture for the existing Yavadunam sutra[11].

4 VEDIC SUTRAS

The Vedic Mathematics comprises of certain sets of formulae or rules which are simple and easy to realize. Those simple Mathematical formulae are known as Sutras. There are 16 Sutras and 13 Upa Sutras. These sutras can be applied to all the branches of arithmetic. Vedic Sutras can be useful to all types of calculations whether it is multifarious or has many numbers of Mathematical operations. When compared to that of our conventional methods which are in current mode the problem solving with these sutras is prompt, pioneering, coherent and lucid. Sutras perk up the logical thinking inclusive of elucidating on the ways to work out the Mathematical problems [8]. The sutras have their relevance in solving different problems in different

principles contained in sutras and up sutras and called it Vedic Mathematics [8]. The most vital component of the old Indian Vedic arithmetic is its consistency. It is critical to specify that the organization of the old science is shrewdly sorted out and joined together. In Vedic Calculations, the common multiplication operation can undoubtedly turned around to accomplish the one-line division of decimal numbers. Likely the simple perspectives. They can be used by combining with other sutras. The following are the 16 sutras constructed by Swamiji:

1. (Anurupyey) Shunyamanyat
2. Chalana-Kalanabyham
3. Ekadhikina Purvena
4. Ekanyunena Purvena
5. Gunakasamuchyah
6. Gunitasamuchyah
7. Nikhilam Navatashtcaramam Dashatah
8. Paraavartya Yojayet
9. Puranapuranaabyham
10. Sankalana- vyavakalanabyham
11. Shesanyankena Charameṇa
12. Shunyam Saamyasamuccaye
13. Sopantyadvayamantyam
14. Urdhva-Tiryagbhyam
15. Vyashtisamanstih
16. Yavadunam

The above techniques can directly be applied to trigonometry, geometry, applied mathematics and even calculus [8].

5 DESIGN OF A SQUARING ARCHITECTURE

Squaring is a special case of multiplication. And hence any method that has been tenable for multiplication can be utilised for squaring. Yavadunam Sutra is a squaring sutra the gist of this sutra is whatsoever the deficits lessen by that amount and set up the square of the deficiency [8],[10]. Yavadunam Tavadunikrtya Vargancha Yojayet is an Up sutra of Yavadunam [10]. This Sutra can be applied to find the square of any number closer to the base of powers of 10. The following are the steps to be followed:

- Step 1:** Find the deficiency with the nearest base.
Step 2: Square the deficiency and place at the right side.
Step 3: Add or subtract the Deficiency from the number
Step 4: Result = [Number – Deficiency + carry over] &
[Square of Deficiency]

Let the sutra be explained with the help of some examples.

- Example:** 1) Let us find the square of the number 11.
• Here the base is 10.

- The number is 1 excess from the base.1 is stated as a deficiency.
- Square of 1 is 1 and this is the RHS.
- Next, add the deficiency to the given number $11+1=12$ let it be LHS.
- Now just concatenate RHS and LHS. Square of 11 is 121.

Example: 2) Square of 997

- Here the base is 1000.
- Deficiency= $1000-997=3$
- Square of deficiency =9 since the base is 1000 the RHS=009
- Subtract the deficiency from the given number $997-3=994$, let this be LHS.
- Square of 997=994009.

Example: 3) Square of 999993

- Here the base is 1000000
- Deficiency= $1000000-999993=7$
- Square of deficiency= 49. Now LHS=000049
- Subtract the deficiency from the given number $999993-7=999986=RHS$
- Concatenating LHS and RHS, the square of 999993=999986000049

Given the above examples designing a unique architecture is impossible since the inputs are from a different range. Thereby the sutra fails to give the correct outcome. Also if the numbers to be squared are not in the range of the same base value then the output will not be correct. To overcome this similar analysis can be extended for the binary numbers. A new architecture have been proposed where the deficiency is calculated from the nearest base value of 2 i.e. 2^N for binary calculations based on Yavadunam sutra [1],[10],[11].The squaring using Yavadunam sutra is precise and rapid; if the same is implemented over the binary numbers they can find extensive applications in high speed signal processors. In order to demonstrate this coveted squaring design in binary number system, here a squaring of 4 bits binary number $A_3A_2A_1A_0$ is considered. Since the number is a 4 bits number the base is 16.Squaring is a special case of multiplication; therefore the output will be more than 4 bits. Since the input is 4 bits the output will be 8 bits and is denoted as $B_7...B_2B_1B_0$.Squaring of binary numbers has two modes. One of the modes is when the deficiency is positive i.e. if the given number is greater than the base value. For example if number is 12 the base value is 10 and the deficiency is +2.And the other mode is when the deficiency is negative i.e. if the given number is less than the base value. For example if number is 8 the base value is 10 and the deficiency is -2.The deficiency is calculated by taking two's complement of the input number.

Mode: 1 The given number is greater than 2^{N-1} .
Mode: 2 The given number is lesser than 2^{N-1} .

A modified version of Yavadunam architectures is proposed which was explained in the previous papers [1],[10],[11].

5.1 Yavadunam Architecture

It is the design of architecture based on the existing Yavadunam sutra which is explained in the previous papers [1],[10] and [11]. Deficiency is computed by two's complementing the given input. The deficiency is then squared by the N bit square unit. The least N bits of the squared deficiency is considered as the RHS part of the output and the rest of the bits are fed as they carrier to the LHS side [10],[11]. The deficiency is subtracted from the input and it is added or subtracted with the carry bit depending on the MSB of the input. If the MSB is '0' then the carry bits are subtracted from the subtractor output[11]. If the MSB is '1' then the carry bits are added with the subtractor output. The output of the adder/subtractor is considered as the LHS part of the output. On concatenating the LHS and the RHS the square of the given input can be obtained [11].

Algorithm for Mode: 1

INPUT: $A_3A_2A_1A_0$
 OUTPUT: $B_7...B_2B_1B_0$

- Step:1) Compute the deficiency by two's complementing $A_3A_2A_1A_0$.
- Step:2) Let the deficiency be $D_3D_2D_1D_0$
- Step:3) Square the deficiency,
 output of squaring= $X_7...X_2X_1X_0$
 carry = Q = $X_7X_6X_5X_4$ and
 RHS = P = $X_3X_2X_1X_0 =$
 $B_3B_2B_1B_0$
- Step:4) Subtract the deficiency from the number
 $A_3A_2A_1A_0 . D_3D_2D_1D_0 = Y_3Y_2Y_1Y_0$
- Step:5) Add the above output to the carry of the
 RHS $E=Y_3Y_2Y_1Y_0 + X_7X_6X_5X_4 =$
 $B_7B_6B_5B_4 = LHS$
- Step:6) Concatenating LHS and RHS ,the output is
 $B_7B_6B_5B_4B_3B_2B_1B_0 = \text{square of } A_3A_2A_1A_0$

Algorithm for Mode: 2

INPUT: $A_3A_2A_1A_0$
 OUTPUT: $B_7...B_2B_1B_0$

- Step:1) Compute the deficiency by two's complementing $A_3A_2A_1A_0$.
- Step:2) Let the deficiency be $D_3D_2D_1D_0$
- Step:3) Square the deficiency,
 output of squaring= $X_7...X_2X_1X_0$
 carry = Q= $X_7X_6X_5X_4$ and
 RHS = P= $X_3X_2X_1X_0 =$
 $B_3B_2B_1B_0$
- Step:4) Subtract the deficiency from the number
 $A_3A_2A_1A_0 . D_3D_2D_1D_0 = Y_3Y_2Y_1Y_0$
- Step:5) Subtract the above output from the carry of the
 $RHS=E=X_7X_6X_5X_4-Y_3Y_2Y_1Y_0$

= $B_7B_6B_5B_4 = \text{LHS}$
Step:6) Concatenating LHS and RHS ,the output is
 $B_7B_6B_5B_4B_3B_2B_1B_0 = \text{square of } A_3A_2A_1A_0$

Both the modes are combined into a single architecture. The number greater than or lesser than 2^{N-1} can be identified from the Most Significant Bit of the numbers. The control signal to select the adder or subtractor depends on the Most Significant Bit of the input i.e. A_{N-1} . When the number is greater than 2^{N-1} , A_{N-1} will be 1 and adder will be selected. In other case if the number is less than 2^{N-1} , A_{N-1} will be 0 and the subtractor will be selected.

5.2. Proposed Yavadunam Architecture

The architecture is designed based on the algebraic proof of the sutra, which states that the square of any number is equal to the sum of the square of the deficiency and the product of the difference between the deficiency and the number to be squared [8]. The above can be simply stated by the below formula,

$$G=A^2 = (A \ll N) + (B*B) - (B \ll N) ----- (1)$$

Where,

- A – Input, the number to be squared
- G – output, square of the input
- B - deficiency, (input-base)
- N – number of bits

Algorithm:

INPUT: $A_3A_2A_1A_0$

OUTPUT: $G_7\dots G_2G_1G_0$

Step:1) Compute the deficiency by taking

Messages	1100	0100	1000	1100	1111	0010
/toplevelm2ex/a	1100	0100	1000	1100	1111	0010
/toplevelm2ex/b	10010000	00010000	10000000	10010000	11100001	00000100
/toplevelm2ex/d	0100	1100	1000	0100	0001	1110
/toplevelm2ex/x	00010000	10010000	01000000	00010000	00000001	11000100
/toplevelm2ex/p	0000	0000	0000	0000	0001	0100
/toplevelm2ex/q	0001	1001	0100	0001	0000	1100
/toplevelm2ex/y	1000	1000	0100	1000	1110	1100
/toplevelm2ex/e	1001	0001	1000	1001	1110	0000

Fig.1 Simulation output of the Existing Yavadunam Architecture

Zoom Full Messages	1111	0001	0011	1001	1011	1111
/toplevel/a	1111	0001	0011	1001	1011	1111
/toplevel/g	11100001	00000001	00001001	01010001	01111001	11100001
/toplevel/b	0001	1111	1101	0111	0101	0001
/toplevel/c	00000001	11110001	10101001	00110001	00011001	00000001
/toplevel/d	11110000	00010000	00110000	10010000	10110000	11110000
/toplevel/e	00010000	11110000	11010000	01110000	01010000	00010000
/toplevel/f	11110001	11110001	11011001	11000001	11001001	11110001

Fig.2 Simulation output of the Proposed Yavadunam Architecture

two's complement of $A_3A_2A_1A_0$.
Step:2) Let the deficiency be $B_3B_2B_1B_0$
Step:3) Multiply the deficiency by itself,
 output = $C_7\dots C_2C_1C_0$
Step:4) Shift the input left N times,
 $E=A_3A_2A_1A_00000$
Step:5) Shift the deficiency left N times,
 $D=B_3B_2B_1B_0\ 0000$
Step:6) Add $A_3A_2A_1A_00000$ and $C_7\dots C_2C_1C_0$,
 let it be $F_7\dots F_2F_1F_0$
Step:7) $A^2=G=F-D$

6 RESULT AND DISCUSSION

The hardware implementation of the squaring architecture for binary numbers is designed. The same is implemented with Xilinx Spartan 3E kit. The simulated output of a 4bit squaring of the Existing Yavadunam Architecture is shown in the Fig.1 and the simulation output of a 4bit squaring of the proposed Yavadunam Architecture is shown in Fig.2. The RLT schematic of the Existing and the proposed Yavadunam Architecture are depicted in Fig.3 and Fig.4 respectively. The synthesis report of the two architectures is depicted in Table.1. Table.1 portrays that the proposed architecture is faster than the existing one. Fig.5.represents the comparisons of the delay of the proposed algorithm with that of the conventional architectures for various bit values. From the Fig.5 it is clear that the Vedic squaring architecture produces less delay and hence it is efficient in speed when compared to the conventional one.

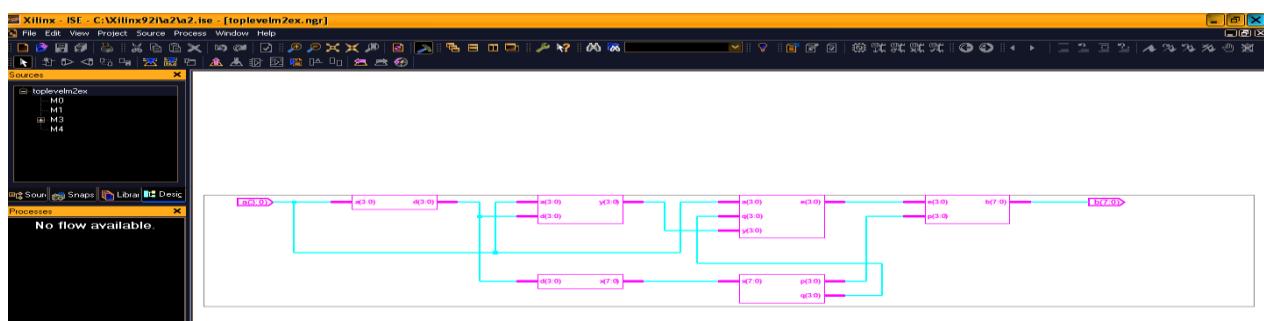


Fig.3 RTL Schematic of the Existing Yavadunam Architecture

S.NO	DEVICE UTILIZATION	EXISTING ARCHITECTURE	PROPOSED ARCHITECTURE
1.	No. of slices	10 out of 3584	4 out of 3584
2.	No. of slice FFs	3 out of 7168	-
3.	No. of 4 input LUTs	17 out of 7168	7 out of 7168
4.	No. of IOs	12	12
5.	No. of bounded IOBs	12 out of 141	12 out of 141
6.	No. of mult18*18s	1 out of 16	1 out of 16
7.	Delay in nS	17.404	15.67

Table. I Summary of Device utilization

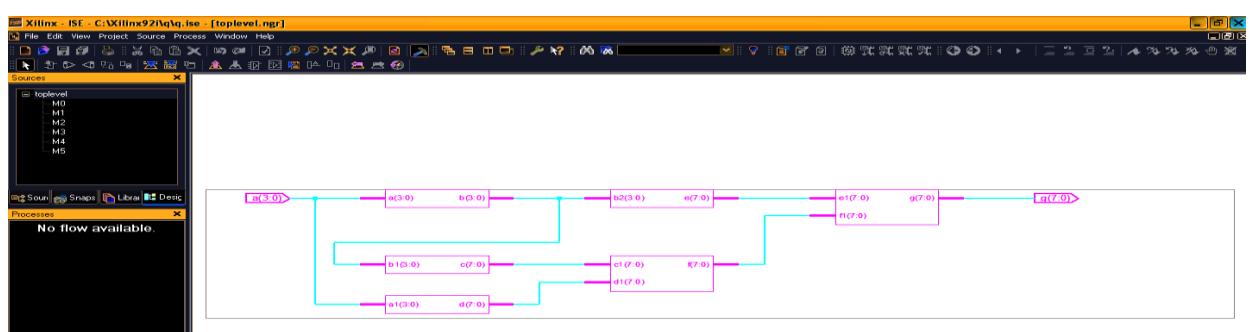


Fig.4 RTL representation of the Proposed Yavadunam Architecture

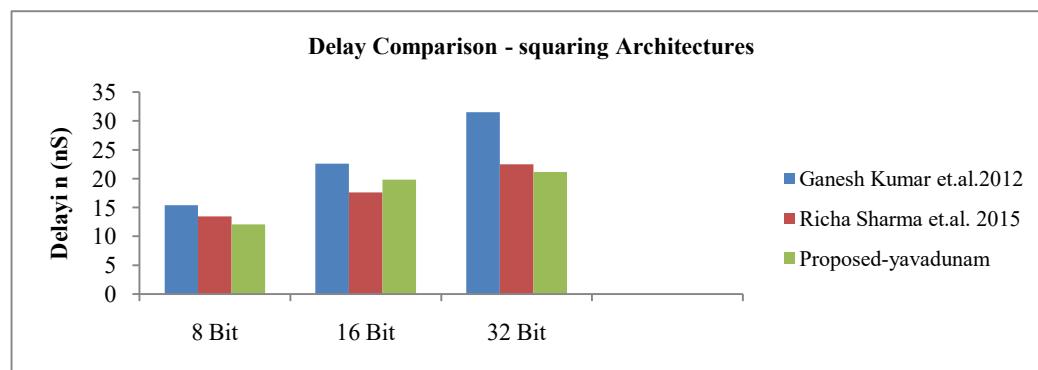


Fig.3 Delay Comparision

7. CONCLUSION AND FUTURE WORK

The problem of designing a unique squaring architecture based on Yavadunam sutra is rectified. Vedic squaring architecture is presented based on the Yavadunam algorithm. The proposed work is efficient for high-speed applications. The future work is to modify this existing Vedic squaring architecture for improved speed and design a proposed architecture for multiplication of given numbers using Yavadunam Sutra.

References

- [1] Deepa.A. and Marimuthu.C.N, "High Speed VLSI Architecture for Squaring Binary Numbers Using Yavadunam Sutra and Bit Reduction Technique", International Journal of Applied Engineering Research, Volume 13, Number 6 (2018) pp. 4471-4474
- [2] Deepa.A. and Marimuthu.C.N, "A High Speed VLSI Architecture of a Pipelined Reed Solomon Encoder for Data Storage in Communication Systems", Asian Journal of Research in Social Science and Humanities, Vol.7, No.2, pp.228-238,(2017).
- [3] Pabitra Kumar Mohapatra,siba Kumar Panda,Sambita Dalal,Shibashis Pradhan,"VHDL implementation of a Novel Low power squaring circuit using YTYY algorithm of Vedic Mathematics ", International Journal of Emerging Trends in Science and Technology, Vol.02, No.3, pp.2139-2146,(2015).
- [4] T.Maheshwar, A.Rama, A.Sreesha,"Novel Advanced Method for the Square and Cube Architectures using Vedic Sutras", International Journal of Engineering and Science, Vol.4, No.04 pp.56-59,(2014).
- [5] Shubham Mishra, Shailendra Kumar Dhakad,"A High Speed and device efficient FPGA based Squaring circuit", International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol.02, No.11,(2013).
- [6] Kabiraj Sethi and Rutuparna Panda, "An improved Squaring circuit for binary numbers", International Journal of Advanced Computer science and Applications, Vol.03, No.02,(2012).
- [7] Prabha S.Kasliwal, B.P.Patil and D.K.Gautam,"Performance evaluation of squaring operation by Vedic Mathematics,"IETE Journal of Research, pp.39-41, Published online: 01 September (2011).
- [8] Jagadguru Swami Sri Bharati Krishna Tirthaji Maharaj "Vedic Mathematics", Motilal Banarsidas Publishers Pvt.Ltd, Delhi (2009).
- [9] Ganesh Kumar, G & Charishma, V, "Design of High Speed Vedic Multiplier using Vedic Mathematics Technique", International Journal of Scientific and Research Publications, Vol. 2, no. 3, pp 1-5,(2012).
- [10] Arish, S & Sharma, R.K, 'An efficient binary multiplier design for high speed applications using Karatsuba algorithm and Urdhva-Tiryagbhyam algorithm', Proceedings of 2015 Global Conference on Communication Technologies, pp 192-196,(2015).
- [11] Sadhana -Indian Academy of Sciences, volume 44, Number 9, pp 1-10, article no 197 (2019).
- [11] Deepa.A., Marimuthu.C.N. and Murugesan, "An Efficient High Speed Squaring and Multiplier Architecture using Yavadunam Sutra and Bit Reduction Technique", Journal of Physics Conference series, volume 1432, article no 012080 (2020).