SQUARE OPERATION IMPLEMENTATION ON RECONFIGURABLE HARDWARE LOGIC TO ATTAIN HIGH SPEED, AREA OPTIMIZATION AND LOW POWER CONSUMPTION

Avinash Patil¹, S. C. Patil², D. S. Bormane³, Sushma Wadar⁴

¹Research Scholar, E&TC Department, Rajarshi Shahu College of Engineering, Pune, MS-India
²Professor, Dept of Information Tech., Rajarshi Shahu College of Engineering, Pune, MS-India
³Principal, AISSMSCOE, Pune, MS-India,
⁴Research Scholar, E&TC Department, AISSMSIOIT, Pune, MS-India

¹avispatil@yahoo.co.uk, ²Shailaja.patil11@gmail.com, ³bdattatraya@yahoo.com, ⁴sushma97in@yahoo.co.in

Corresponding Author: Avinash Patil

https://doi.org/10.26782/jmcms.spl.9/2020.05.00004

Abstract

The contribution made by authors in the research work carried out on square operation is brought forward operated on a four and eight-bit number using duplex property of number based on Vedic mathematics. The conventional method of computing square of a number follows the polynomial multiplication of the same number to find the square. The said method requires the area and power consumption is not sufficiently optimized considering today’s low power application needs. The proposed method of computing the square of a number presented here is based on the Dwandva yog of Vedic mathematics which also called as duplex property of a number. The duplex method of calculating the square of number gives the online solution which can be easily calculated mentally and the efforts were to prove the same with the electronic circuit. The implementation of the square algorithm using polynomial multiplication and Vedic mathematics based duplex property for square operation is carried out with VHDL coding on the Xilinx Vivado 2015 ISE tool and the FPGA used is Artix7 device: 7a35tcpg236-1. The results were compared with 4-bit as well as 8-bit operation using both algorithms for a square operation are it is observed that the speed of operation is improved by 20 % whereas the hardware resources utilized were reduced by 66 %.

Copyright reserved © J. Mech. Cont.& Math. Sci.
Avinash Patil et al.
Conference on “Emerging Trends in Applied Science, Engineering and Technology” Organized by MDSG Research Group, Malaysia
Keywords: Vedic mathematics; Urdhva-Tiryagbhyam sutra; Dwandva Yog; Duplex Property.

I. Introduction

A square operation is generally calculated based on polynomial multiplication. The equation for polynomial multiplication is, for example if \( m \) and \( n \) are two numbers, then \( m \) and \( n \) are given by equations 1 and 2. The multiplication is given as the product of \( m \) and \( n \) which is equals to \( m*n \) as given in equation 3.[IX]

\[
m(p) = \sum_{i=0}^{d} m^i p^i
\]  
\( (1) \)

And

\[
n(p) = \sum_{i=0}^{d} n^i p^i
\]  
\( (2) \)

then,

\[
y(p) = m(p)n(p) = \sum_{i=0}^{d} \sum_{j=0}^{d} m_in_j
\]  
\( (3) \)

But a square of a number is the product of an integer with itself i.e. if 25 is a square number then it can be written as 5 x 5. Usually the notation for the square of a number \( m \) is instead of showing as product \( m \times m \); it is represented as equivalent exponentiation \( m^2 \), and is pronounced as "m squared". Square numbers are non-negative and the expression of the square is given as equation (II).

\[
m^2 = \sum_{j=1}^{m} (2j-1)
\]  
\( (4) \)

Arithmetic operation square of a number is an important operation in many applications such as encryption and decryption process, signal and image processing includes an arithmetic operation which is today part of many digital systems. The application of square of a number is in digital signal processing applications which include pattern recognition, image compression and so on [V]. Squaring is also used in cryptography algorithms, graphics processors [V].

Almost in all the calculations based on Conventional Method also called Parallel square algorithm, the multiplier unit is used to calculate the square of a number so it may be in processors, DSP, DIP applications or in encryption-decryption operations. As the square is required very frequently instead of using a multiplier to perform the
operation it is best to use dedicated square hardware which improves the speed of operations and reduction in power consumption significantly. A parallel square unit was based on urdhva triyakbyam sutra (UT) of Vedic mathematics Vertical and Crosswise proposed in [III] having reduced area and higher speed of operation. With the use of Vertical and crosswise method of multiplication, finding a square of 8-bit number is already a faster approach. Thus, there is a significant reduction in the device utilization, further; a square can be computed much faster than a square with a normal multiplier. The square architecture proposed is proving an excellent reduction in hardware over multipliers with conventional method or even using Vedic multiplier in which the ‘n’ bit multiplication can be performed [I][VII].

II. Methodology

Proposed algorithm for calculation of square of a number

The algorithm proposed by authors to calculate the square of a number is to incorporate the Duplex property of a number explained in Vedic mathematics, which is explained for decimal number system is also applicable for binary numbers[II] and Vertical and crosswise method of multiplication also called Urdhva Tiryakbyam sutra of multiplication. To calculate the square of a given 8-bit number the “Duplex” D property of numbers which is proposed in Vedic mathematics is here applied on binary numbers. The Duplex property of a number works as explained,

If a number is a single bit $B$

$B$ is the same number i.e. $\text{Dup}(B_0) = B_0$.

If a number is two bit $B_1B_0$

$B$ is twice their product i.e. $\text{Dup}(B_1B_0) = 2 * B_1 * B_0$

If a number is three bit $B_2B_1B_0$

$B$ is twice the product of the outer pair + the middle bit

i.e. $\text{Dup}(B_2B_1B_0) = 2 * B_2 * B_0 + B_1$.

If a number is four bit $B_3B_2B_1B_0$

$B$ is twice the product of the outer pair + twice the product of the inner pair

i.e. $\text{Dup}(B_3B_2B_1B_0) = 2 * B_3 * B_0 + 2 * B_2 * B_1$

Thus, if a square of a 4 bit number is to be computed then,

$\text{Dup} (a) = a;$

$\text{Dup} (ab) = 2 * a * b;$

$\text{Dup} (abc) = 2 * a * c + b;$

$\text{Dup} (abcd) = 2 * a * d + 2 * b * c;$

So to calculate the square of a two digit number denoted as $ab$ by the Duplex method is as given by equation 5

Copyright reserved © J. Mech. Cont. & Math. Sci.
Avinash Patil et al.
\[(ab)^2 = a^2 + 2ab + b^2 \]  

(5)

And the \(a^2\) and \(2ab\) and \(b^2\) calculated by the Urdhva Tiryakbyam based multiplication techniques instead of using polynomial multiplication. So the proposed system provides dual advantages such as reducing area, delay and power consumption by using UT based multiplier but also further enhances the parameters with use of the duplex method. Urdhva Triyakbyam method of multiplication is generalized multiplication technique applicable for all numbers and the Nikhilam Navatascaramam Dasatah and Ekadhi kena Purvena sutras are applicable for special cases of numbers. If any number is multiplied by 11.

\[\text{Res}=ab \times 11\]
\[\text{Res}=a \& (a+b) \& b\] where \& is a concatenation operator or mathematically,
\[\text{Res}=100(a) +10(a+b) +b\]

And if carry generated by middle term will be added to most significant digit i.e. a

e.g. 1. 34*11=3&(3+4) &4=3&7&4=347

\[\text{Res}=100(a) +10(a+b) +b\]

2. 67*11=6&(6+7)&7=6&13&7=737 as carry 1 from 13 propagated and added into most significant digit 6. The example proves that for special case any number multiply by 11 can be solved by only 2 adders than to use 8 bit multiplier circuit. Similarly, the architecture shown in Figure 1 is for 2 bit UT multiplier where

\[R_0=A_0B_0\]
\[C_1R_1=A_1B_0+A_0B_1\]
\[C_2R_2=C_1+A_1B_1\]
\[R_3=C_2\]

The final result will be \(R_3R_2R_1R_0\) which are calculated in parallel propagation delay is reduced significantly.

III. System

The VHDL code is written for both the algorithms of calculation of square of an eight-bit number based on conventional multiplication and Urdhva Tiryakbyam based multiplier with Duplex property based square circuit. And both the codes were tested on Artix 7 FPGA 7a35tcpu236-1. The results were generated are post synthesis results.
Using the 2 bit UT multiplier architecture the 4 bit multiplier is designed and used with the carry look ahead adder which will produce all the 6 bit output in parallel and Carry look ahead adder helps to produce the carry propagated through with a constant delay of 2 gate delays.

Fig No. 2 shows the flowchart of a proposed algorithm for calculation of square of an 8-bit number based on Duplex property and Vertical and crosswise method of multiplication.

![Flowchart of calculation of square of an 8-bit number using duplex property and UT multiplier.](image)

\[
\begin{align*}
R_2 &= (A_1 \times B_1) \text{ xor } C_0 \quad R_1 = (A_1 \times B_0) \text{ xor } (A_0 \times B_1) \quad R_0 = A_0 \times B_0 \\
R_3 &= A_1 \times B_1 \times C_0
\end{align*}
\]

Copyright reserved © J. Mech. Cont.& Math. Sci.
Avinash Patil et al.
V. Verification and Implementation

The VHDL coding of proposed work and its simulation was carried out on Xilinx Vivado 2015 platform. The post-implementation simulation results were generated for the algorithms coded in VHDL using Xilinx Vivado 2015.4 ISE simulator tool. The design is optimized for speed and area using Xilinx Vivado design constraints and device considered is Artix7 FPGA 7a35tcepg236-1

VI. Result and Discussion

The results generated were tabulated below in Table no 1. Device utilization report of Square operation on an 8-bit number using polynomial multiplication with Urdhva Tiryakbhyam sutra of Multiplication is generated. The report shows the device utilization in terms of number of LUTs while Fig no 3 shows the Simulation Result of Square operation on an 8-bit number using polynomial multiplication with Urdhva Tiryakbhyam sutra of Multiplication. Fig no 4 shows the RTL generated due to VHDL code for Square operation on 8-bit number using polynomial multiplication with Urdhva Tiryakbhyam sutra of Multiplication. Fig no 5 Shows Power Consumption report of Square operation performed on an 8-bit number using polynomial multiplication with Urdhva Tiryakbhyam sutra of Multiplication. Power report indicates the dynamic power consumed is 13.433 w. The Table no 2 shows the device utilization report of Square operation on an 8-bit number using Urdhva Tiryakbhyam sutra of Multiplication along with Duplex property of number. Device utilization in terms of number of LUTs used is indicated while Fig no 6 shows the simulation result. Fig no 7 shows the RTL generated by the code and Fig no 8 shows the power consumption report. Power report indicates the dynamic power consumed is 8.091 w which is further lesser than the first algorithm. The final Fig no 9 concludes the comparison of a square calculation based on Vedic with the inclusion of urdhva triyakbyam based multiplier gives the best result when compared with the parameters such as power consumption and device utilization and speed of operation.

Table 1: Device utilisation report of Square operation on an 8-bit number using polynomial multiplication with UrdhvaTiryakbhyam sutra of Multiplication

| Site Type    | Used | Available | % Utilization |
|--------------|------|-----------|---------------|
| Slice LUTs   | 104  | 20800     | 0.50          |
| LUT as Logic | 104  | 20800     | 0.50          |

Copyright reserved © J. Mech. Cont.& Math. Sci.
Avinash Patil et al.
Fig. 3: Simulation Result of Square operation on a 8-bit number using polynomial multiplication with Urdhva Tiryakbhyam sutra of Multiplication

Fig. 4: RTL Generated due to VHDL code for Square operation on a 8-bit number using polynomial multiplication with Urdhva Tiryakbhyam sutra of Multiplication

Fig. 5: Power Consumption report of Square operation on a 8-bit number using polynomial multiplication with Urdhva Tiryakbhyam sutra of Multiplication
Table 2: Device utilisation report of Square operation on a 8-bit number using Urdhva Tiryakbhyam sutra of Multiplication along with Duplex property of number.

| Site Type     | Used | Available | % Utilization |
|---------------|------|-----------|---------------|
| Slice LUT     | 35   | 20800     | 0.17          |
| LUT as Logic  | 35   | 20800     | 0.17          |
| F7 Muxes      | 4    | 16300     | 0.02          |

Fig. 6: Simulation Result of Square operation on a 8-bit number using Urdhva Tiryakbhyam sutra of Multiplication along with Duplex property of number.

Fig. 7: RTL Generated due to VHDL code for Square operation on a 8-bit number using Urdhva Tiryakbhyam sutra of Multiplication along with Duplex property of number.
VII. Conclusion

The results achieved for square operation circuit of an 8 bit number by Duplex property and vertical and crosswise multiplier is compared over with Conventional Polynomial multiplier based square circuit and with Vedic based square multipliers circuit. The results were generated by using Xilinx Vivado 2015 licensed ISE tool on a reconfigurable hardware Artix7 7a35tcpg236-1. The results for finding square of a 8 bit number is observed to be far better by proposed method when compared with conventional with the speed of operation is improved by 20 % whereas the hardware resources utilized were reduced by almost 66 %.
Reference

I. Avinash Patil¹, Dr Shailaja Patil², Dr D S Bormane³, Sushma Wadar⁴ “Novel Technique of Finding Square of Number to Reduce the Resources Utilized on Reconfigurable Hardware Logic”, 7th-8th Sept 2018 at GRT institute of Technology, Chennai, IEEE conference

II. Avinash Patil¹, Dr Shailaja Patil², Y V Chavan², Sushma Wadar³ “Division operation based on Vedic Mathematics”, 2nd-3rd Dec 2016 at RSCOE JSPMS, IEEE conference

III. Avinash Patil¹, Y V Chavan², Sushma Wadar³ “Performance analysis of Multiplication operation based on Vedic mathematics” 21st, 22nd Oct 2016, Allahabad, IEEE conference

IV. C. M. Kim, “Multiplier design based on ancient indian mathematics,” IEEE, vol. 11, pp. 3686–3689, 2008.

V. 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.

VI. 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.

VII. Jagadguru Swami Sri Bharati Krishna Tirthji Maharaja, “Vedic Mathematics”, Motilal Banarsidas, Varanasi, India, 1986, Book.

VIII. 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.

IX. 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, IIT Kharagpur

X. Stuart F. Obermann and Michael J. Flynn, Division algorithms and implementations,” IEEE Transactions on Computers, 46(8):833–854, August 1997.

XI. Sudhanshu Mishra, Maoranjan Pradhan, “Synthesis Comparison of Karatsuba Multiplier using Polynomial Multiplication, Vedic Multiplier and Classical Multiplier”. International Journal of Computer Applications (0975 – 8887) Volume 41– No.9, March 2012

XII. T. Prabakar, “Design and fpga implementation of binary squarer using vedic mathematics,” IEEE, 2013.