International Journal of Electrical and Electronics Research ISSN 2348-6988 (online)
Vol. 3, Issue 3, pp: (131-140), Month: July - September 2015, Available at: www.researchpublish.com

Pipelined Linear Convolution Based On Hierarchical Overlay UT Multiplier

Pranav K, Pramod P

[1]PG scholar (M Tech VLSI Design and Signal Processing) L B S College of Engineering Kasargod, Kerala, India
[2]Assistant professor (Dept. of ECE) L B S College of Engineering Kasargod, Kerala, India

Abstract: **Convolution is one of the fundamental operations of the signal processing system and it can be employed by types of multiplication. Here linear convolution is performed by Vedic multiplier which is based on one of the sixteen sutras in Vedic mathematics, called Urdhava Triyagbhyam Sutra. Urdhava Triyagbhyam multiplier provides better result in speed compared to other conventional multiplier. Pipelining architecture in the multiplier increases the timing performance of the multiplier hence such multiplier is used to compute linear convolution of two sequences. This is a novel approach to compute convolution using pipelining architecture. The simulation and synthesis are performed by using ModelSim and Xilinx ISE Design suite 13.4 and power analysis is performed by using Xilinx Xpower Analyzer. Simulated result for proposed pipelined convolution shows 24.24% lesser propagation delay compared to convolution without using pipelining.**

Keywords: **Convolution, Compressor, Pipelining, Urdhava Tiryagbhyam.**

I. INTRODUCTION

Vedic mathematics is a stream of mathematics, based on four Vedas (wisdom of knowledge). It is part of Upaveda (supplement) of Atharva Veda called Sthapatya-Veda (book on civil engineering and architecture). It gives explanation of several mathematical fields including geometry (plane, co-ordinate), arithmetic, trigonometry, quadratic equations, factorization and even calculus. Advancement of recent technology can happen by applying Vedic mathematic techniques especially in VLSI design field. Recently signal processing has a pivotal role in VLSI design. Discrete convolution is one of the major computation blocks in the Digital signal processors such as filters. Surabhi jain and Sandeep Saini have implemented a one of the efficient linear convolution by using one sutra in Vedic mathematics. This was the advanced version of the fast convolution method. This method was similar to multiplication of two numbers and which was very easy to understand [1]. Pierre and John W presented novel approach for calculating linear convolution sum by using graphical method. This graphical method is shown as an easy way for learning linear convolution sum [2].

Vedic Mathematics is the name given to the ancient system of Indian Mathematics which was rediscovered from the Vedas between 1911 and 1918 by Sri Bharati Krisna Tirthaji (1884-1960). According to his research all of mathematics is based on sixteen Sutras (formulae). The advantage of Vedic mathematics lies in the fact that it changes some typical calculations in conventional mathematics to very simple ones. Vedic formulae increases the speed of calculations because it claimed to be based on the natural principles on which the human mind works [3]. There are five sutra for fast calculation of multiplication, among that Urdhava Triyagbhaym (UT) Sutra and Nikhilam Sutra are more efficient. This Sutra had been traditionally used for the multiplication of two numbers. UT Sutra is a general multiplication formula applicable to all cases of multiplication. It means "Vertically and crosswise". Nikhilam Sutra means "all from 9 and last from 10". It is also applicable to all cases of multiplication; it is more efficient when the numbers involved are large [4]. Urdhva Tiryakbhyam (UT) Sutra is most efficient Sutra (Algorithm), giving minimum delay for multiplication of all types of numbers; either small or large [5].

The three important considerations for VLSI design are power, area and delay. Currently, multiplication time is still a dominant factor in determining the instruction cycle time of a DSP chip. The conventional multiplication techniques like array multiplier, Wallace tree multiplier and booth algorithm multiplier are provide greater delay compared to Vedic

multiplier [6]. The great advantages of Vedic multipliers are regularity in the structure and its lesser gate delay. That means a 2-bit Vedic multiplier is used for employing 4-bit multiplication, 4-bit multiplier is used for 8-bit multiplication and so on. An efficient multiplier based on decomposition provides better than traditional multipliers [7], [8]. It gives efficient multiplier architecture and increases the speed of multiplication and reduces the time delay. Convolution is fundamental operation of most of the signal processing systems. It is necessity of time to speed up convolution process at very appreciable extent. Filtering of signals is very important in order to determine which one to accept and which one to reject and all of that is done by convolution [9], [10]. The speed of the processor can be increased by ensuring pipelined architecture. Harish et *al* [11] gives a novel method for pipelined architecture of Vedic multiplier. UT multiplier in the convolution shows better result by employing pipelined architecture in adding section of partial products. The compressors are used to employing addition of partial products instead of Ripple Carry Adder (RCA) or Carry Save Adder (CSA) [12]. Hence the propagation delay of the linear convolution decreases by 12.21% compared to the existing convolution method but total number of delays used in such design is very large. Hence pipelined convolution is employed by using hierarchical overlay UT multiplier. It provides lesser propagation delay and lesser number of delay elements.

II. UT MULTIPLIER

Vedic mathematics is mainly based on 16 Sutras (or aphorisms) dealing with various branches of mathematics like arithmetic, algebra, geometry etc. Nikhilam sutra and UT sutra are mainly concern with the multiplication among these two UT sutra is more efficient for multiplication of digits.

A. UT Sutra:

The given Vedic multiplier based on the Vedic multiplication formulae (Sutra). This Sutra has been traditionally used for the multiplication of two numbers. UT Sutra is a general multiplication formula applicable to all cases of multiplication. It means "Vertically and Crosswise". The digits on the two ends of the line are multiplied and the result is added with the previous carry. When there are more lines in one step, all the results are added to the previous carry. The least significant digit of the number thus obtained acts as one of the result digits and the rest act as the carry for the next step. Initially the carry is taken to be as zero. The line diagram for multiplication of two 4-bit numbers is as depicted in Fig.1.

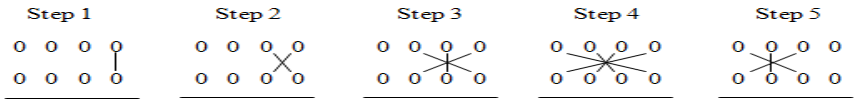

Figure 1: Line diagram for two 4-bit numbers

The main advantage of Vedic multiplier regularity and symmetry is used here. That is each Multiplication operation is an embedded parallel 4x4. 8-bit multiplier was designed using 4bit multiplier, l6-bit multiplier using 8bit multiplier and 32-bit multiplier using l6-bit multiplier.

B. 2x2 bit Vedic multiplier:

Let A_1A_0 and B_1B_0 are two 2-bit numbers say multiplier and multiplicand then the output can be of four bits, say S_3 S_2 S_1 S_0. As per line diagram depicted above, lsb of the result obtained by direct multiplication and all other bits are obtained by adding internal partial products. The architecture of 2 bit multiplier is shown in Fig.2.

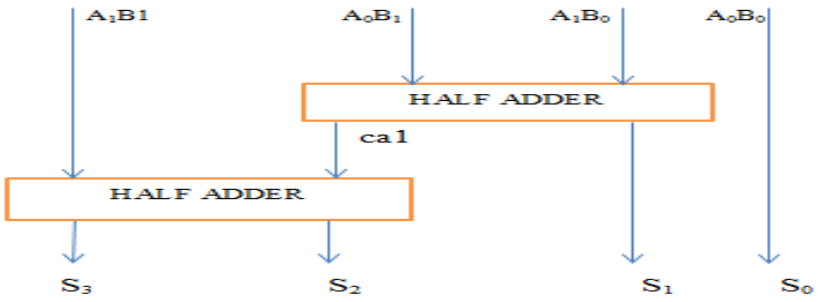

Figure 2: 2-bit multiplier

C. 4x4 bit Vedic multiplier:

Let $A_3\ A_2\ A_1\ A_0$ and $B_3\ B_2\ B_1\ B_0$ are two four bit numbers and $S_7\ S_6\ S_5\ S_4\ S_3\ S_2\ S_1\ S_0$ is the multiplication result of two. Here 4-bit multiplication are obtained by parsing A and B into two parts, say $A_3\ A_2$ & $A_1\ A_0$ for A and $B_3\ B_2$ & $B_1\ B_0$ for B. Using the fundamental of Vedic multiplication, two bit Vedic multiplications are performed by using 2-bit Vedic Multiplier block, we can have the following structure for multiplication shown in Fig.3

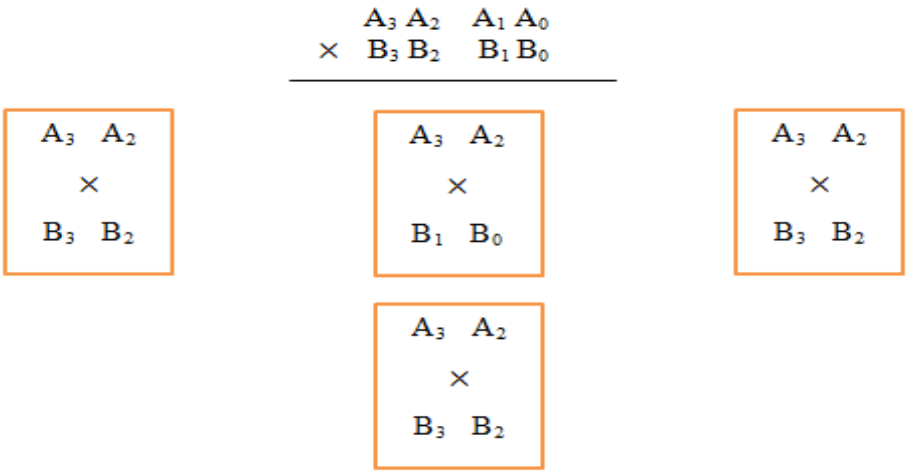

Figure 3: Block diagram representation of 4x4 multiplications

So the final result of multiplication, which is of 8 bit, $S_7\ S_6\ S_5\ S_4\ S_3\ S_2\ S_1\ S_0$, can be interpreted as given in Fig.4

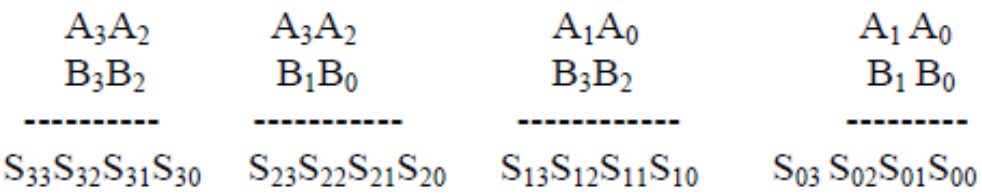

Figure 4: Interpretation of 4x4 bit Vedic Multiplier

For the final result, add the middle product term along with other terms shown in Fig.5.

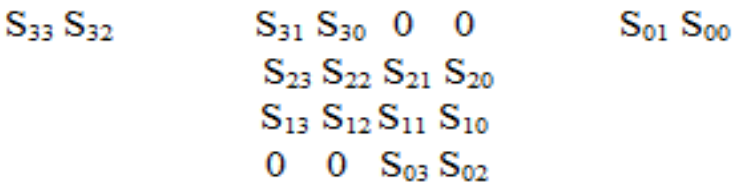

Figure 5: Arrangement of Partial Product

The first two outputs S_0 and S_1 are same as that of S_{00} and S_{01}.Result of addition of the middle terms by using two, 4 bit full adders will forms output line from $S_5\ S_4\ S_3\ S_2$. One of the full adder will be used to add ($S_{23}\ S_{22}\ S_{21}\ S_{20}$) and ($S_{13}\ S_{12}\ S_{11}\ S_{10}$) and then the second full adder is required to add the result of 1st full adder with ($S_{31}\ S_{30}\ S_{03}\ S_{02}$). The respective sum bit of the 2nd full adder will be $S_5 S_4\ S_3\ S_2$. Now the carry generated during 1st full adder operation and that during 2nd full adder operation should be added using half adder so that the final carry and sum to be added with next stage i.e. with$S_{33}\ S_{32}$ to get $S_7\ S_6$.4-bit multiplication can also be employed by using compressors. Compressors are used to add partial products and give final result. Design of 4-bit UT multiplier using compressors is shown in Fig.6. Architecture of 4:3 Compressor, 5:3 Compressor and 6:4 Compressor are shown in Fig.6, Fig.7 and Fig.8 respectively.

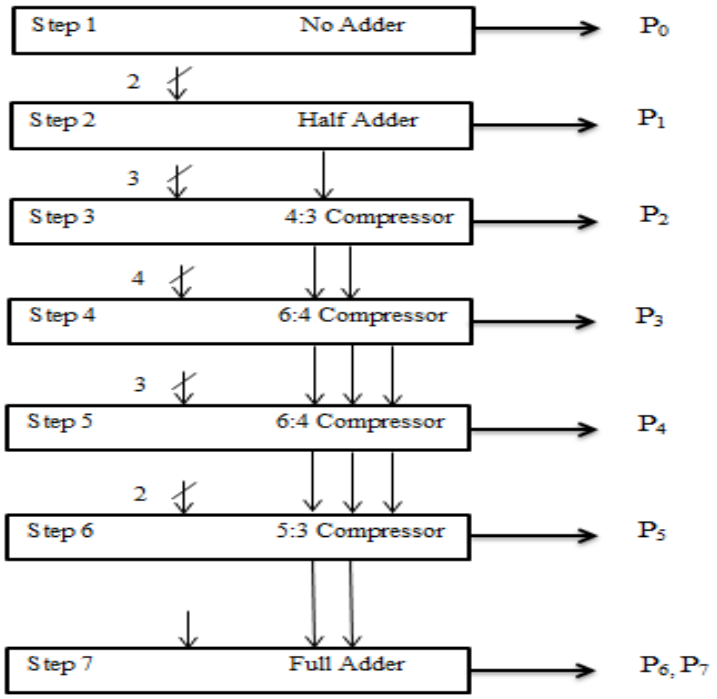

Figure 6: Partial Products Addition using Compressors in 4-bit UT multiplier

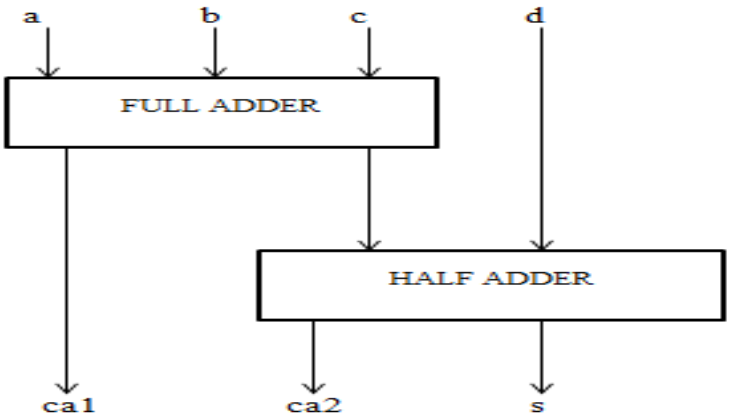

Figure 7: 4:3 Compressor

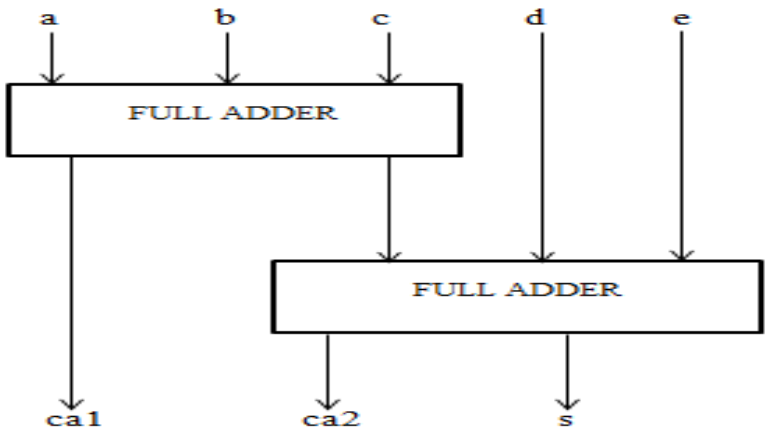

Figure 8: 5:3 Compressor

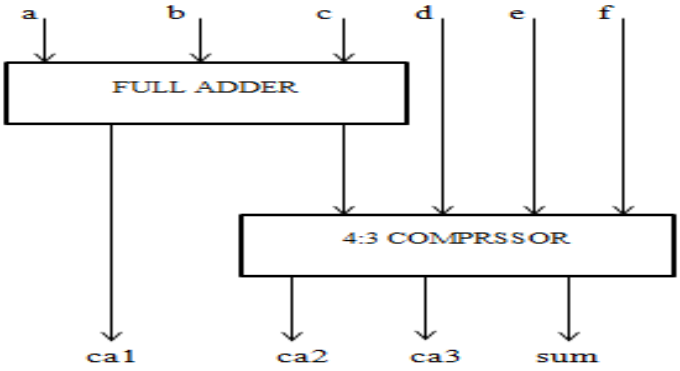

Figure 9: 6:4 Compressor

Where a, b, c, d, e and f are inputs, s is sum, ca1, ca2 and ca3 are carries.

III. CONVOLUTION

Linear convolution can be solved in different methods such as graphical method, matrix method and by using linear equation. The linear convolution of sequence $f(n)$ and $g(n)$ is given below

$$y(n) = f(n) * g(n) \quad (1)$$

$$y(n) = \sum_{k=-\infty}^{\infty} f(k)g(n-k) \quad (2)$$

The new approach for calculating the convolution sum is set up like multiplication where the convolution of $f(n)$ and $g(n)$ is performed as shown in Fig.10. Where $f(n) = \{4\ 2\ 3\}$ and $g(n) = \{4\ 5\ 3\ 4\}$:

$g(n)$: 4 5 3 4
$f(n)$: * 4 2 3

		12	15	09	12
	08	10	06	08	
16	20	12	16		
$y(n)$: 16	28	34	37	17	12

Figure 10: Convolution by proposed method

Tow sequence can be convolved in high speed when it uses the advantages of UT Vedic multiplier. Let two sequence are $\{a_4\ a_3\ a_2\ a_1\}$ and $\{b_4\ b_3\ b_2\ b_1\}$, elements in the sequence are 4-bit each and their convolution result sequence is $\{c_7\ c_6\ c_5\ c_4\ c_3\ c_2\ c_1\}$.Convolution is carried out by performing the multiplication, but carries are not performed out of a column. The decimal is positioned just as with multiplication and then the answer is rewritten as a sequence as shown in Fig.11.

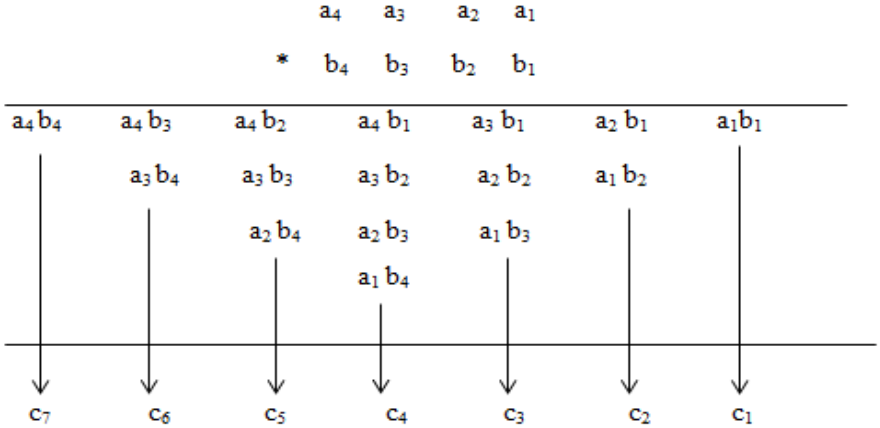

Figure 11: Convolution of two sequences

IV. CONVOLUTION USING PIPELINED UT MULTIPLIER

This section depicts idea of pipelining in UT multiplier and how these multipliers provide high speed convolution. The steps included in a 4-bit UT multiplier are depicted in Fig.12.

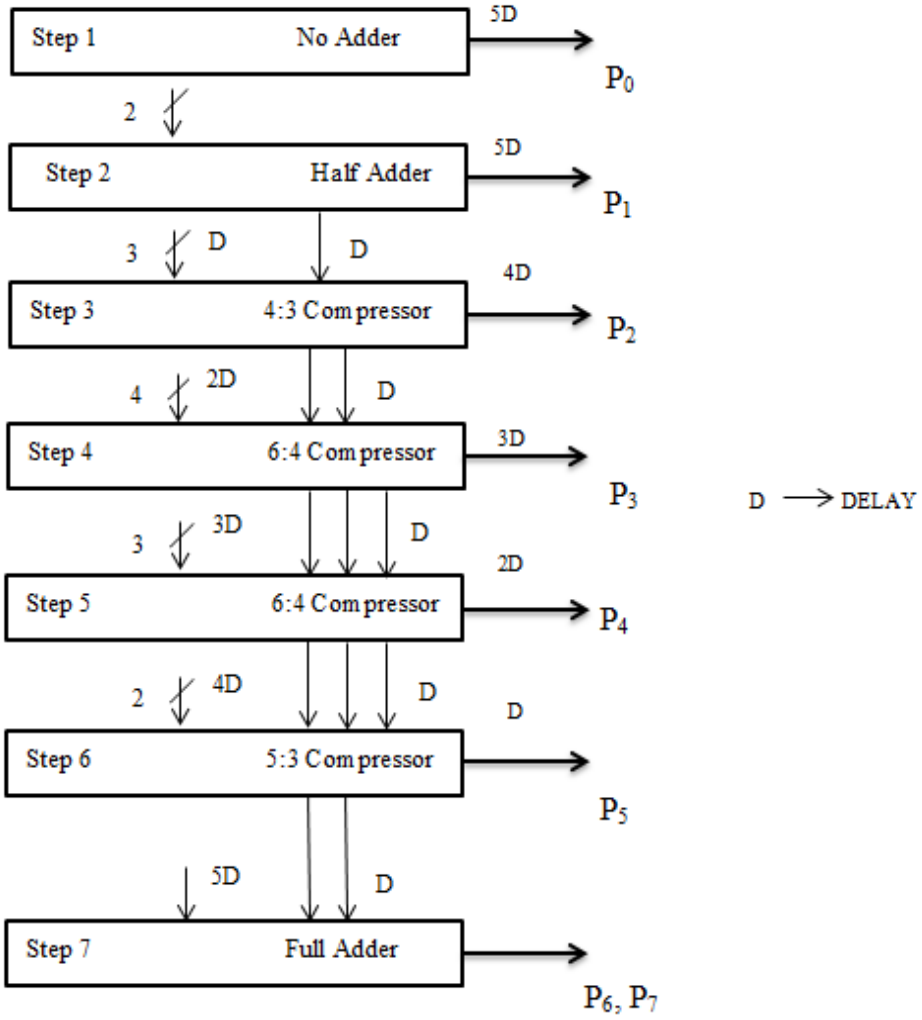

Figure 12: Block diagram of pipelined architecture

Pipelining in UT Multipliers are designed as above by inserting appropriate number of registers in between two stages. Convolution of two numbers can be easily computed by adding partial products in such a way that there is no propagation of carry from one stage to other. Here each digit is 4-bit and whose product is calculated by using pipelined UT multiplier. The arrangements of pipelined multiplier and convolution operations depicted in Fig.13. Here $\{a_0, a_1, a_2, a_3\}$ and $\{b_0, b_1, b_2, b_3\}$ are two input sequence where each digit is 4-bit length and $\{c_1\ c_2\ c_3\ c_4\ c_5\ c_6\ c_7\}$ is the convolution result of it. Total number of delays used in the pipelined UT multipliers 68; such 16 pipelined UT multipliers are used for employing convolution. Hence total 1088 number of delay elements are used for calculating linear convolution of two sequences by using pipelined UT multiplier.

V. PIPELINED ARCHITECTURE FOR LINEAR CONVOLUTION BASED ON HIERARCHICAL OVERLAY UT MULTIPLIER

Linear convolution of two sequences can be calculated by using 4-bit UT multiplier in such a condition that each input digits in the sequence is 4-bit length and it provides a convolution output of sequence having digits length varying from eight to ten bits. The performance of the convolution improved by pipelining architecture by additional delay in between two stages. Pipelined architecture of convolution uing UT multiplier is shown in Fig.14. Here total two delays are placed in between input and output so latency of the system is reduced to two instead of five in the previous case.

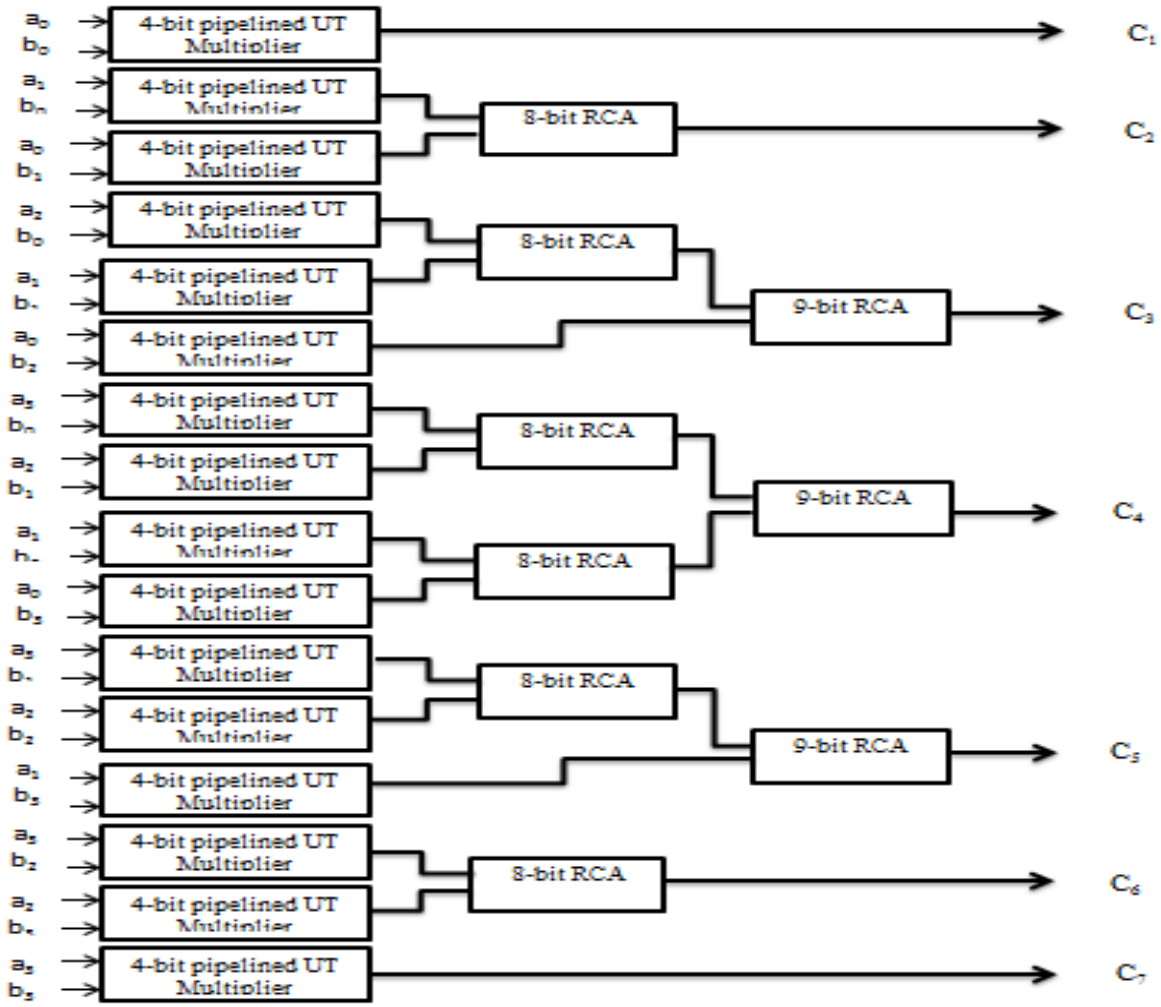

Figure 13: Convolution using pipelined UT multiplier

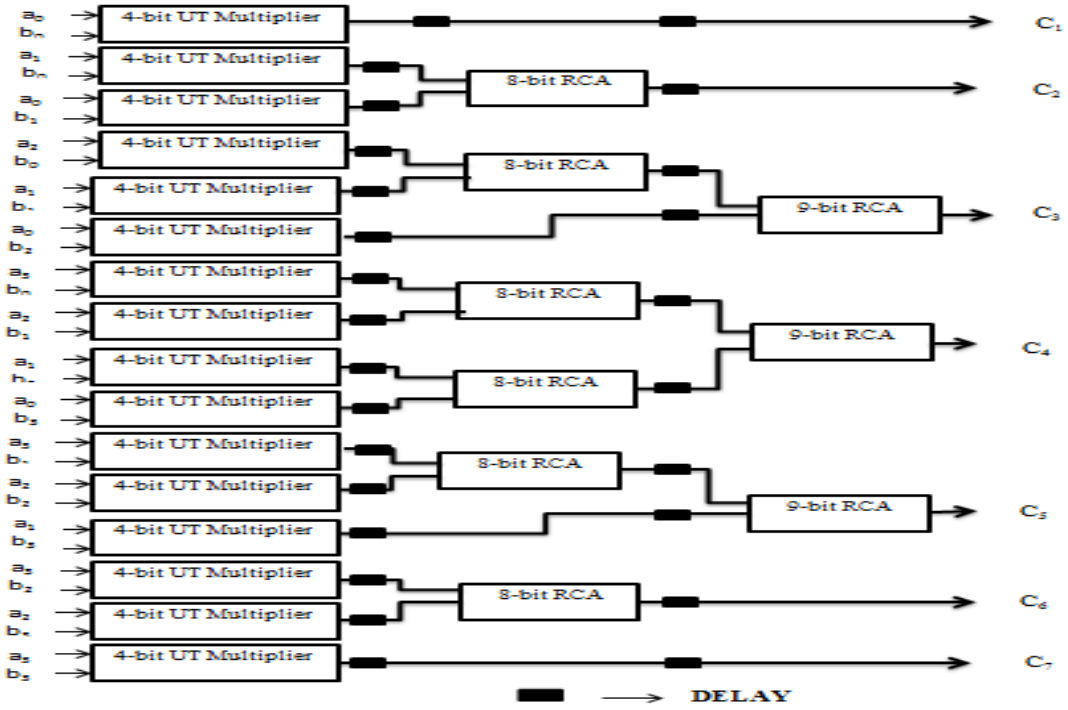

Figure 14: Pipelined Architecture for linear convolution using UT multiplier

VI. EXPERIMENT RESULT

The convolution using pipelined UT multiplier and without pipelined UT Multiplier are designed and implemented using VHDL code. The models are simulated and verified using Xilinx 13.4 and ModelSim 6.3f. The power of the designs are estimated by using Xilinx Xpower Analyzer. Simulation result of convolution using UT multiplier without pipelining, with pipelined UT multiplier and pipelining in the overlay architecture are shown in Fig.15, Fig.16 and Fig.17 respectively. Fig.15 depicts convolution of two sequence using UT multiplier(without pipelining) so latency of such convolution is zero but in Fig.16 shows the timing diagram of convolution using pipelined UT multiplier and latency is five. That means output for corresponding inputs is generate only after five clock pulses, it's clearly shown in the Fig.16. Simulation result of pipelined linear convolution using UT multiplier is shown in Fig.17 and from this clear that latency of the system is reduced to two.

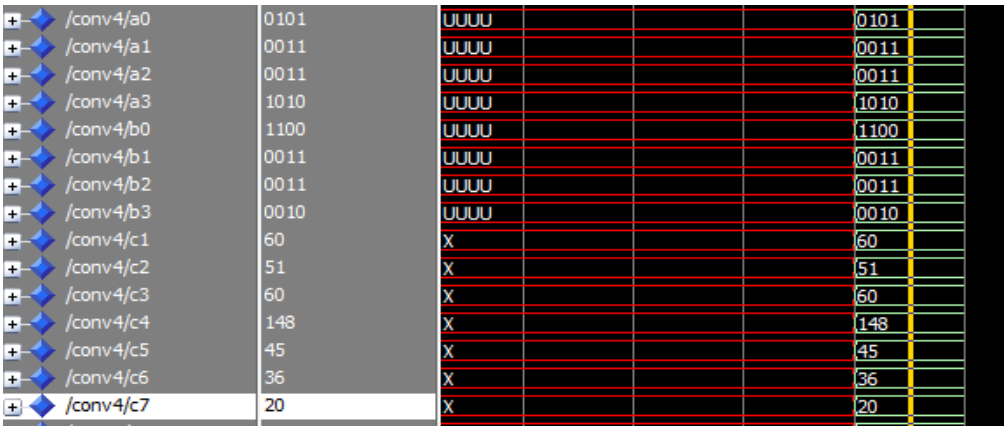

Figure 15: Time Diagram of Convolution of Two Sequence using UT Multiplier (without pipelining)

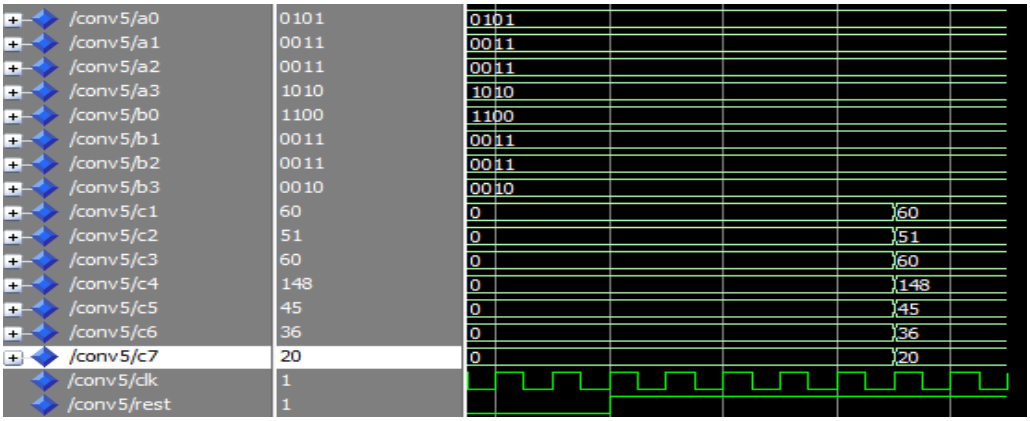

Figure 16: Convolution of two sequences

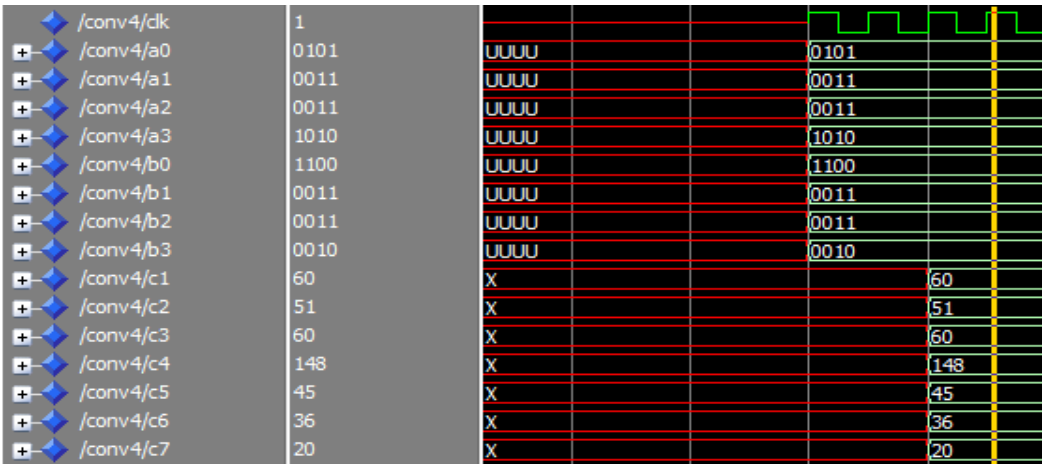

Figure 17: Timing Diagram of Pipelined Convolution of Two Sequences

Delay-Power comparison of existing and proposed convolution using pipelined UT multiplier is tabulated in Table. I and graphical representation comparisons is shown in Fig.18.

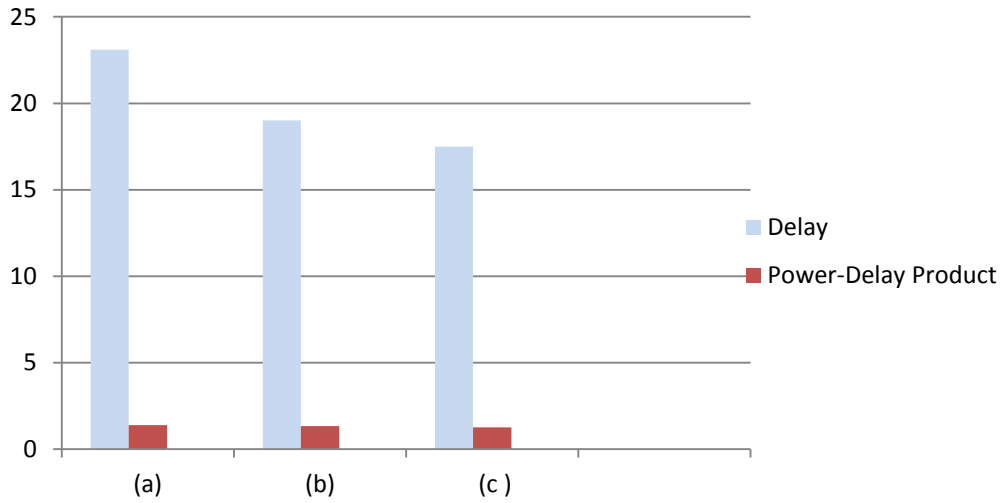

Figure 18: Power-Delay comparisons

Total number of delays used in the pipelined UT multiplier is 68; such 16 pipelined UT multipliers are used for employing convolution. Hence total 1088 delay elements are used. But in the case of pipelined convolution using UT multiplier total of 214 delays are used. Graphical representation is shown in Fig.19.

Table 1: Power-Delay Comparison

Parameter	Convolution without Pipelining (a)	Convolution using Pipelined UT Multiplier (b)	Pipelined Convolution using UT Multiplier (b)
Propagation Delay	23.097 ns	19.016 ns	17.496 ns
Power	61 mW	70 mW	72 mW
Power-Delay Product	1.385 nJ	1.331 nJ	1.259 nJ

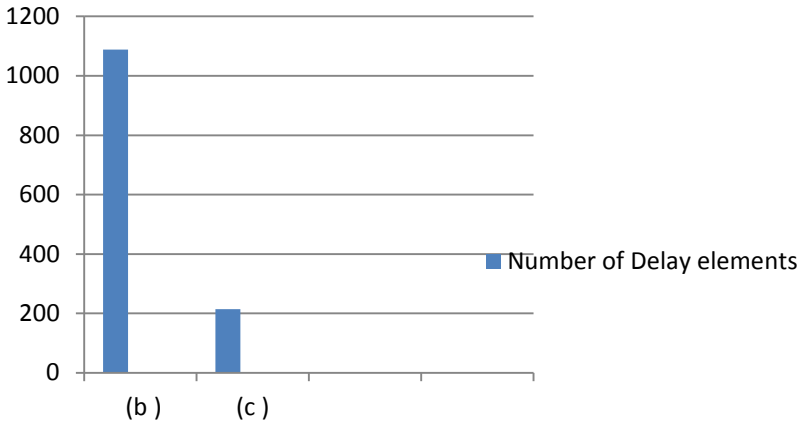

Figure 19: comparison on number of Delays

VII. CONCLUSION

This work mainly focused to introduce a method for calculating the linear convolution based on pipeline architecture with the help of Vedic algorithms that easy to learn and perform. Then this method is shown as a way of quickly computing the convolution sum and checking both their final and intermediate answers from graphical convolution. Speed of the convolution increased by employing pipelined architecture in UT multiplier. From the primary synthesis analysis it is clear that the propagation delay of pipelined convolution has reduced to 24.24% than without pipelining convolution.

Pipelined architecture for convolution based on UT multiplier gives a better result than convolution using pipelined UT multiplier and convolution using UT multiplier. Total number of delay elements used for pipelined convolution is 80.33% lesser than that of convolution using pipelined UT multiplier.

REFERENCES

[1] Surabhi Jain, Sandeep Saini, "High Speed Convolution and deconvolution Algorithm (Based on Ancient Indian Vedic Mathematics)", IEEE Conf. on. Comm. and Signal Processing (ICCSP), 2014.

[2] Pierre, John W, "A novel method for calculating the convolution sum of two finite length sequences", IEEE Transaction on 39.1, Education, 2012.

[3] Jagadguru Swami Sri Bharati Krishna Tirthaji Maharaja, "Vedic Mathematics", Motilal Banarsidass, New Delhi, India, 1994.

[4] Rudagi, J. M, et al., "Design and implementation of efficient multiplier using Vedic mathematics" in proc. IEEE Conf. on Advances in Recent Technologies in Comm. and Computing,2011, pp:162-166.

[5] Vaidya, Sumit, et al., "Delay-Power Performance Comparison of multipliers in VLSI circuit design", in proc. (IJCNC), Vol.2, No.4, 2013.

[6] Akhter, Shamim, "VHDL implementation of fast NxN multiplier based on Vedic mathematic" in Proc. IEEE European Conf. In Circuit Theory and Design (ECCTD), 2007, pp: 472-475.

[7] Thapliyal, Himanshu, and M. B. Srinivas, "High Speed Efficient NxN Bit Parallel Hierarchical Overlay Multiplier Architecture Based on Ancient Indian Vedic Mathematics" in Trans. On Engg. Computing and tech, 2004, pp: 225-228.

[8] Lomte, Rashmi K., and P. C. Bhaskar, "High Speed Convolution and Deconvolution Using Urdhva Triyagbhyam" in Proc. Int. Symp Circuits Syst. (ISCAS), 2011, pp :35-41.

[9] Hanumantharaju M, et al, "A High Speed Block Convolution using Ancient Indian Vedic Mathematics"2007 , in proc. IEEE int.conf. Computational Intelligence and Multimedia Applications, pp: 25-32.

[10] Akhalesh K. Itawadiya, et al, "Design a DSP operations using Vedic mathematics" in proc.IEEE Comm. and Signal Process. (ICCSP), 2013, pp:14-22.