

Design of Hierarchy Multiplier Based on Vedic mathematics using CSLA and BEC

¹B.Farshana, ²P.Nagarajan, ³S.Manikandan

¹PG Student [VLSI], ²Assistant Professor, ³ Assistant Professor

¹Department of ECE, ¹Vivekanandha College of Engineering for women, Namakkal, India

Abstract—Hierarchy multiplier has the ability to carry the multiplication operation within one clock cycle. The existing hierarchical multipliers occupy more area and also results in more delay. A method to lower the computation delay of hierarchy multiplier by using Carry select adder (CSLA) and Binary to Excess 1 Converter (BEC) is proposed. The BEC removes the $n/4$ number of adders, existing in the conventional addition scheme, where n indicates the multiplier input width. Then the area of the hierarchy multiplier is determined by its base multiplier, the base multiplier is realized with the proposed Vedic multiplier, this has small area and operates with less delay than the conventional multipliers. Furthermore the reduction of power consumption in the hierarchy multiplier can be confirmed by implementing the designed multiplier with full swing Gate Diffusion Input (GDI) logic. The Cadence SPICE simulator using 45 nm technology model has used to analyses the performances of the proposed and also the existing multipliers.

Index Terms— Vedic multiplier, Binary to Excess 1 Converter, Carry Select Adder.

I. INTRODUCTION

Multiplier-Accumulation operation is used repeatedly the algorithm of Digital Signal Processing (DSP), such as convolution, correlation, FFT (Fast Fourier Transform), DCT (Discrete Cosine Transform). Since multiplication uses too much time, MAC often becomes the bottleneck of DSP Processor and bounds the performance of DSP core. Various applications need different data width multipliers. For example voice, audio and video processors have different precisions, and some processors manipulate data in the different precision.

Hierarchical multipliers are considered as feasible means for achieving orders of magnitude speed up in computer intensive applications through the use of fine grained parallelism. They are used in various fields of numerical and scientific computations, communication, and Image processing, cryptographic computation and so on [1–5]. To design n bit hierarchical multiplier, four $n/2$ base multipliers are important which generate $2n$ bit output, where n indicates hierarchical multiplier input width. All the base multipliers are allowed to perform the task in parallel. Due to that, the performance of the hierarchy multiplier is determined from the accumulation delay of its base multipliers output bits. But this is a time consuming task as it needs more number of additions and considered a bottleneck for the hierarchy multiplier performance. The multiplication process needs several intermediate stages to get the final result due to which critical path gets lengthened. Also, these intermediate stages involves additional hardware which leads to increase in area and power consumption. To overcome these disadvantages, multipliers based on Vedic Mathematics technique have been proposed.

In this work, an approach to perform this accumulation with less number of addition process is proposed. The following are the contributions discussed in the paper.

- (i) For the area and delay efficient implementation of base multiplier, a new design is proposed based on Vedic mathematics concept.
- (ii) To decrease the accumulation delay of base multiplier output bits, Carry select adder and Binary to Excess 1 Converter are introduced.
- (iii) To understand the hierarchy multiplier with small area it is realized using Full Swing Gate Diffusion Input (FS-GDI) logic.

The rest of the paper is organized as follows. Section ii overviews prior work related to the multiplier proposed in this paper. Section iii describes the proposed hierarchy multiplier architecture. Section iv presents experimental results. Section v concludes the discussion.

II. EXISTING METHOD

Multipliers with large width are necessary for the implementation of cryptography and error correction circuits for more reliable transmission over highly insecure and/or noisy channels in networking and multimedia applications. The hierarchical principle helps to realize fast large bit multiplier, except that it requires a large width adder for performing the addition task, which poses limitation on the performance and increases area of the designed multiplier [6–7]. Over the last few decades, a lot of works have been dedicated, at the algorithmic and implementation level, to improve the performance of hierarchical multiplier. The delay in the addition process of the hierarchy multiplier is reduced with the parallel execution of ripple carry adder [8]. However, this method requires twice the number of adders thus results in increased area. On the other hand, the delay is reduced with the deployment of carry look ahead adder for the addition process but this increases the interconnection complexity [9]. Not only delay and area, the power consumption of the hierarchy multiplier also has to be reduced because the existing designs appending more zeros to equalize the number of bits in order to make them suitable for parallel computation [10]. This might increase the spurious activities and thus increases the power consumption.

III. PROPOSED HIERARCHY MULTIPLIER

An approach for efficient implementation of n bit hierarchy multiplier with minimum delay is discussed. As an example, the architecture for 16 bit multiplier design is explained. Further, a new design is suggested for the hierarchy multiplier building block namely, base multiplier based on Vedic mathematics. Following that, the discussion of CSLA, Binary to Excess 1 Converter and multiplexer is carried out in this section.

In general, the hierarchy multiplier speed is determined from the computation delay of base multiplier output bits addition. This delay can be decreased by reducing the number of additions without affecting the functionality. The following approach is incorporated in the proposed n bit hierarchy multiplier multiplication procedure to reduce the delay:

Step 1: The multiplier inputs and output are represented as X , Y and Z , respectively.

Step 2: Divide n bit multiplier inputs i.e., X and Y , into equal two halves. For the input X , it is divided into $(X_{n/2-1}, \dots, X_0)$, $(X_n, \dots, X_{n/2})$, which are assigned as XL and XH , respectively. The same procedure is also adopted for another multiplier input Y .

Step 3: After dividing both the inputs, they are formed into four groups like (XL, YL) , (XH, YL) , (XL, YH) and (XH, YH) .

Step 4: The multiplication is realized using four $n/2$ bit base multipliers namely, a_0 , a_1 , a_2 and a_3 .

Step 5: The multiplier product bits $Z_{n/4-1}, \dots, Z_0$ is obtained from 0 to $n/2-1$ output bits of a_0 .

Step 6: The resultant bits of a_1 , a_2 and concatenation of a_0 ($n/2$ to n), a_3 (0 to $n/2-1$) are formed an array of carry save format which are processed by carry save adder.

Step 7: The resultant sum and carry from carry save adder are becoming the inputs for CSLA of n bit adder. Also, the sum output of CSLA adder are assigned as multiplier resultant bits for the range of $Z_{n+n/2-1}, \dots, Z_{n/2}$.

Step 8: BEC takes the input from a_3 ($n/2$ to n bit) and its output bits are available prior to CSLA and they are passed to the multiplexer.

Step 9: The multiplier output bits $Z_{2n} \dots Z_{n+n/2}$ is obtained from the multiplexer, based on the carry output of CSLA adder, if it is one then the BEC output are becoming the output otherwise the product bits of a_3 ($n/2$ to n bit).

Based on this algorithm, 16 bit (n) hierarchy multiplier architecture is designed as shown in Fig 3.1. The multiplier inputs are X , Y of 16 bit width and produces the output Z of 32 bit. First, the inputs X and Y are divided into equal two halves namely, XH and XL , YH and YL and they are multiplied by 8 bit base multiplier. As seen in Fig 4.1, the symbols of a_0 , a_1 , a_2 and a_3 denote the base multiplier for the multiplication of $(XL$ and $YL)$, $(XH$ and $YL)$, $(XL$ and $YH)$ and $(XH$ and $YH)$, respectively.

Once these multiplication processes is over, then their output bits will form a carry save array as per step 6, which in turn processed by carry save adder thus results into two rows of 16 bit output. These bits are further added with the help of 16 bit CSLA adder to produce the Z_{24}, \dots, Z_8 multiplier output bits. Meanwhile, the BEC also computed its output and fed to multiplexer as one of the inputs. Another input for the multiplexer is from a_3 output (half of the output bits i.e., $n/2$ to $n-1$). Finally, the multiplexer selects, either BEC or a_3 output bit as $Z_{24}-Z_{31}$, based on CSLA adder's carry. As a result of introduction of BEC in the hierarchy multiplier, $n/4$ adders are eliminated.

Due to the parallel computation of BEC and CSLA output, the processing delay for multiplier output bits i.e., $Z_{24}-Z_{31}$ is minimized significantly. As seen from the architecture of proposed hierarchy multiplier, given in Fig.1, the critical path of the proposed architecture consists of one base multiplier, one bit adder, one CSLA adder and multiplexer only. Further, the implementation details of building components of the hierarchy multiplier namely, base multiplier, CSLA adder and BEC converter are described in the following subsection.

A. Vedic multiplier

The proposed Vedic multiplier is based on the Vedic multiplication formulae (Sutras). These Sutras have been conventionally used for the multiplication of two numbers in the decimal number system. The multiplier is based on Urdhva Triyakbhyam (Vertical & Crosswise) is one of the sutra of ancient Indian Vedic Mathematics.

Vedic multiplier is created on a novel concept through which the generation of all partial products can be done with the concurrent addition of these partial products. The parallelism in generation of partial products and their summation is achieved using Urdhva Triyakbhyam explained. To illustrate the multiplication algorithm, let us consider the multiplication of two binary numbers $a_3a_2a_1a_0$ and $b_3b_2b_1b_0$. As the result of this multiplication would be more than 4 bits, we express it as $r_3r_2r_1r_0$. The 8×8 Vedic multiplier designed along with three ripple carry adders. Ripple carry adder is designed using multiple full adders to add 8-bit numbers. Each full adder inputs a C_{in} , which is the C_{out} of the previous adder. The adder is known as a ripple-carry adder, since each carry bit "ripples" to the next full adder.

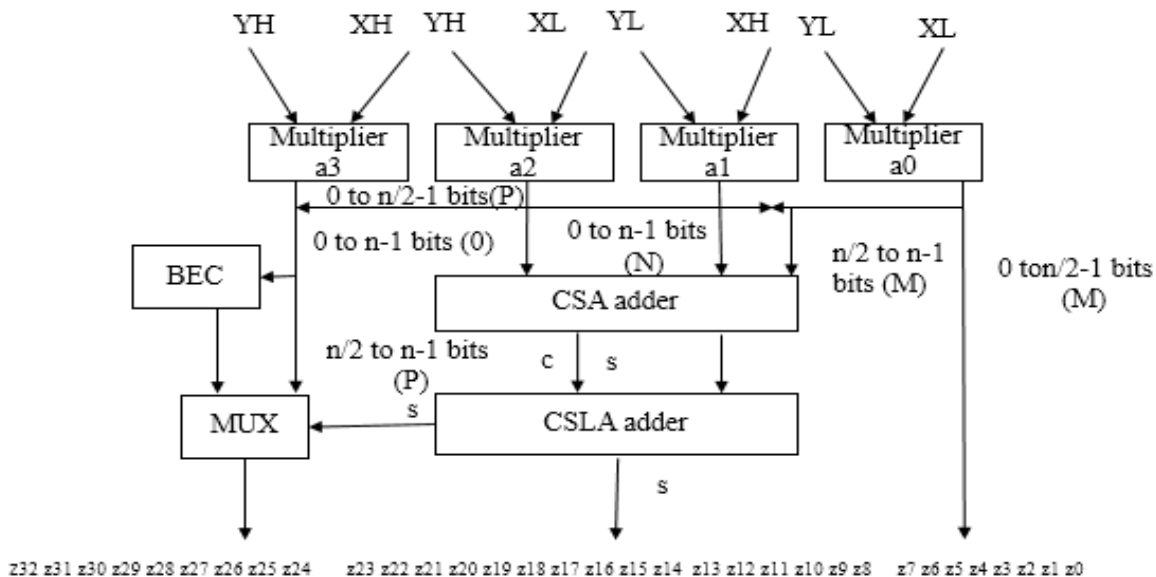


Fig 1

proposed 16 bit hierarchy multiplier

B. Base multiplier

The performance of the hierarchy multiplier is obtained by its base multiplier. In the conventional multiplication techniques, the intermediate computation present in the multiplier operation decreases the speed exponentially in accordance with the number of bits present in multiplier input. This becomes serious issue for more number of input bits. But this issue can be moderated by the parallel addition of partial products which is an inherited principle of Vedic multiplication method.

Wallace multiplication uses random placement of counters for the efficient partial product accumulation due to this, the design becomes complex than the conventional scheme. Hence the Vedic multiplication is considered as an alternative way of performing the multiplication operation without increasing the circuit complexity and power consumption Figure 2 shows the example of Vedic algorithm.

The digits on the two ends of the line are multiplied and the result is added with the previous carry. When the number of lines are more in one step then all the results are added to the previous carry. The least significant digit (LSB) of the number thus achieved acts as one of the result digits and the rest act as the carry for the next step. In the beginning the carry is taken to be as zero.

In Booth multiplication the partial products reduction is possible but the encoding and decoding mechanism involved in this method increases the circuit complexity thereby power consumption. In the multiplication process, the partial products are collected at every step as opposed to the conventional multiplication schemes. Hence the speed of this multiplier can be improved by reducing its partial product accumulation delay.

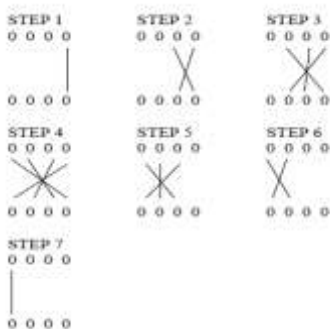


Fig 2 Example of Vedic algorithm

C. Binary to Excess 1 Converter

Code converters are very important in digital systems. To reduce the delay of partial products addition in the hierarchy multiplier, this work uses BEC instead of adder for the output bits of $Z_{2n-1}, \dots, Z_{2n+n/2}$. For n bit input width, n + 1 bit BECs are required. The main advantage of BEC comes from the less number of logic gates.

D. Carry Select Adder

In the modified CSLA, the carry computation part uses part of the half adder output as input thereby delay is getting increased. This issue can be overcome by making independent carry computation. Though the gate counts are increasing due to the requirement of separate circuits for carry output, the proposed CSLA total layout area will be small due to FSGDI logic based implementation.

E. Multiplexer

Multiplexer is also known as Universal element or Data Selector. Multiplexers used to increase the amount of data that can be sent over the network A Multiplexer has of 2^n inputs have n select lines in it. MUX operation based on the select lines. Depending upon the select line the input is Send to the output.

IV. RESULTS AND DISCUSSIONS

The simulation results of the 16 bit Hierarchy multiplier with the basic modules namely Vedic multiplier, carry select adder, Binary to Excess 1 Converter and Multiplexer were designed. The performance parameters are evaluated using spice simulation using 45 nm technology. The proposed multiplier has smaller delay compared to other existing implementations. Due to the implementation of BEC converter in the base multiplier output bits accumulation the number of adders are reduced thus decreases the delay significantly. The simulation results shows that the Vedic multiplier has minimum path delay compared to array multiplier.

Figure 5 shows the delay output for 16 bit hierarchy multiplier. The Delay(v(a0),v(p0)) is 990.62ps. Then Delay(v(a0),v(p31)) is 836.1ps. Then Delay(v(a15),v(p0)) is 990.62ps and delay(v(a15),p(31)) is 836.1ps. The Delay(v(b0),p(0)) is 1.4906ns and Delay(v(b15),p(31)) is 1.336ns. Here the given value the output obtained with the Delay(v(b15),v(p0)) is 1.4906ns and Delay (v(b15), v(p31)) is 1.336ns. For 16x16 bit Vedic multiplier consider two 16 bit numbers are A and B such that the individual bits can be represented as the A [15:0] and B [15:0].

The final output can be obtained as the C16S [31:0]. The partial products are calculated in parallel and hence delay obtained is decreased enormously for the increase in the number of bits. The Least Significant Bit (LSB) S0 is achieved easily by multiplying the LSBs of the multiplier and the multiplicand. After performing all the steps the result (Sn) and carry (Cn) is obtained and in the same way at each step the previous stage carry is forwarded to the next stage and the process goes on.

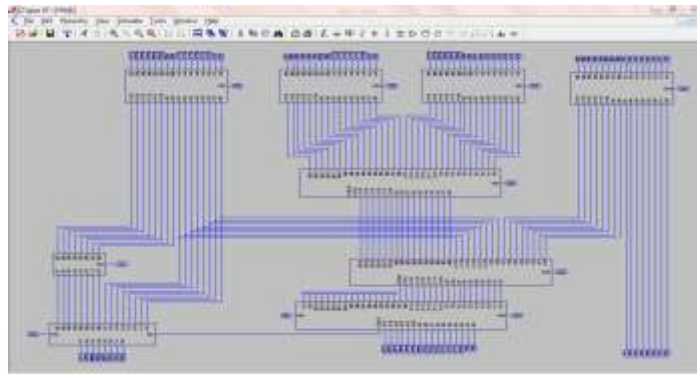


Fig 3 schematic view of 16 bit Hierarchy multiplier



Fig 4 Simulation output for 16 bit Hierarchy multiplier

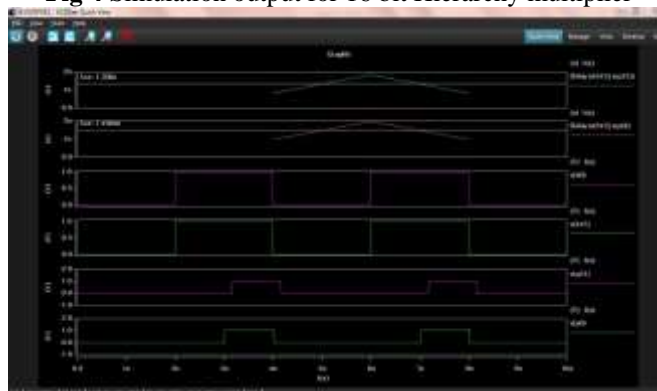


Fig 5 Delay for 16 bit Hierarchy multiplier

V. CONCLUSION AND FUTURE WORK

The BEC converter based hierarchy multiplier architecture is proposed which operates with less delay due to the removal of $n/4$ number of adders, presented in the existing hierarchy multiplier. Moreover, the delay incurred by BEC is not affecting the hierarchical multiplier because it is not included in the critical path of the multiplier. Furthermore, a new design for base multiplier is proposed, based on Vedic mathematics, which is having less delay and area compared with other multipliers. The major outcome of the proposed design is the number of adders is reduced. Then the realization of proposed multiplier using FS-GDI logic lowers the power consumption and area. Thus, area-power and delay efficient hierarchy multiplier is designed. The performances delay and power consumption of the existing and the proposed hierarchy multipliers are calculated through SPICE simulation using 45 nm technology models. In future instead of carry select adder static energy recovery full adder can be used. Because it has minimum of 10 Transistors.

REFERENCES

- [1] Jhamb Garima M. and Lohani H. (2016), 'Design, implementation and performance comparison of multiplier topologies in power-delay space', *Eng. Sci. Technol., Int. J.*, vol.19, pp.355–363.
- [2] Zakaria Z. and Abbasi S.A. (2013), 'Optimized multiplier based upon 6 input LUTs and Vedic mathematics', *World Acad. Sci. Eng. Technol.*, vol.7, pp.26–30.
- [3] Quan G., Davis J.P., Devarkal P. and Buell D.A. (2005), 'High level synthesis for large bit width multipliers on FPGAs: a case study', in: *Proc. Int. Conf. Hardware/ Software Codesign Syst Synth.*, pp.213–218.
- [4] Shi J., Jing G., Di Z. and Yang S. (2011), 'The design and implementation of reconfigurable multiplier with high flexibility', in: *Proceedings of the International Conference on Electronics, Communications and Control*, pp.1095–1098.
- [5] Quan S., Qiang Q. and Wey C.L. (2005), 'A novel reconfigurable architecture of low power unsigned multiplier for digital signal processing', in: *Proceedings of the International Symposium on Circuits and Systems*, pp.3327–3330.
- [6] Abbasi S.A., Zulhelmi and Alamoud A.R.M. (2015), 'FPGA design, simulation and prototyping of 32 bit pipeline multiplier based on Vedic mathematics', *IEICE Electron. Exp.*, vol.12, pp.1–12.
- [7] Pushpangadan R., Sukumaran V. and Innocent R. (2009), 'High speed Vedic multiplier for digital signal processors', *IETE J. Res.*, vol.55, pp.282–286.
- [8] Ronisha Prakash A. and Kirubaveni S. (2013), 'Performance evaluation of FFT processor using conventional and Vedic algorithm', in: *Proceedings of the International Conference Emerging Trends in Computing, Communication and Nanotechnology*, pp. 89–94.
- [9] Sethi K. and Panda R. (2015), 'Multiplier less high speed squaring circuit for binary numbers', *Int. J. Electron.*, vol.102, pp.433–443.
- [10] Ramalatha M. and Thanushkodi K. (2009), 'A novel time and energy efficient cubing circuit using Vedic mathematics for finite field arithmetic', in: *Proceedings of the International Conference on Advances in Recent Technologies in Communication and Computing*, pp. 873–875.