

# Design and Analysis of Hybrid Tree Multipliers for Reduction of Partial Products

SAFIA BIBI\*, MUHAMMAD OBAID ULLAH\*\*, AND MUHAMMAD ALI SHAMI\*\*\*

RECEIVED ON 24.02.2017 ACCEPTED ON 29.05.2017

## ABSTRACT

This paper confers one of the three phases of tree multipliers, i.e. the Partial Product reduction phase. In this paper four types of hybrid tree multipliers are studied and proposed using parallel counters (full adders and half adders) for reduction of Partial Products in multiplication operation. After ANDing the bits of multiplier and multiplicand, the Partial Products are arranged into two groups for reduction, each group uses a different technique for reduction of Partial Products, resulting in a fewer gates than the parent tree reduction techniques. The results of the proposed tree reduction techniques are then tabulated and compared with the parent tree multipliers. The performance comparison is done in terms of number of gate counts of half adder and full adders used in the Partial Product reduction phase. Four types of hybrid tree multipliers are presented using CSA (Carry Save Adder) Array multiplier, Wallace Tree multiplier, Modified Wallace Tree multiplier and Dadda Tree Multiplier. The results show significant reduction in number of full adders and half adders with the slight overhead of increased final addition stage of the hybrid multiplier. The proposed multipliers can prove to be the better choice for digital signal processing designs, image processing designs and processor architecture.

**Key Words:** Carry Save Adder Array, Wallace Tree Multiplier, Modified Wallace Tree Multiplier, Dadda Tree Multiplier, Partial Products

## 1. INTRODUCTION

Multipliers are crucial in modern electronic systems like image and signal processing, ALU (Arithmetic Logic Unit) and VLSI (Very Large Scale Integration) Design applications [1-2]. With the progress in technology, a lot of work has been done on various reduced area, fast multiplier schemes as well as to provide appropriate design algorithms for various high speed, low power and dense VLSI implementations

[3-5]. Multiplication produces Partial Product matrix which is reduced and then added to produce final result [6]. Any multiplier unit has 3 phases:

- (i) Generation of Partial Products
- (ii) Reduction of Partial Products
- (iii) Final Addition

Authors E-Mail: (safia.bibi@uettaxila.edu.pk, obaid.ullah@uettaxila.edu.pk, ali.shami@bui.edu.pk)

\* Department of Electronics Engineering, University of Engineering & Technology Taxila, Sub Campus Chakwal.

\*\* Department of Electrical Engineering, University of Engineering & Technology, Taxila.

\*\*\* Department of Electrical Engineering, Bahria University, Islamabad.

An  $N \times N$  bit multiplier needs  $N^2$  AND gates for multiplication forming  $N$  rows of Partial Product matrix having  $N$  bits each, and the final sum produces  $2N+1$  bits result [7].

The Partial Products are generated by ANDing every bit of the multiplier with all the bits of multiplicand [8]. Subsequently, after generation of Partial Products there is reduction of Partial Product matrix to two rows. This is achieved by splitting the Partial Products into groups of three or two. The groups of three or two bits are then reduced by using CSA. The third step in multiplication is the addition of reduced rows of Partial Products by using CPA (Carry Propagate Adder) to get the final product. The multiplication process can be accelerated on the basis of these three expedients.

Multipliers can be optimized in generation of Partial Products, reduction of Partial Product matrix and addition of final two rows of reduced Partial Product matrix.

The second step, i.e. Partial Product reduction is the main parameter that determines the performance of the multiplier as it adds most to the overall delay.

In this paper, four types of Hybrid Tree Multipliers are proposed and designed using conventional CSA Array, Wallace Tree, Dadda Tree and Modified Wallace Tree Reduction Schemes. The proposed designs exploit the advantages of their parent multipliers and reduce the number of compressors in the Partial Product reduction phase of tree multipliers for fast and low area applications like Signal and Image Processing.

## **1.1 RELATED WORK**

A variety of multipliers have been discussed and proposed in the literature to get an efficient design [9]. In 1964 Wallace proposed column compression scheme for fast multiplication with the entire delay which is relative to the logarithmic of the length of the word in the operand [10]. The proposed column compression multiplier is faster

than the array multiplier because of the fact that in array multiplier, as the operand word length increases, the delay increases linearly. In 1965 Dadda refined the approach of reducing the number of adders in column compression scheme anticipated by Wallace [11]. Baugh and Wooley [12] worked on a parallel array multiplier algorithm. Townsend et. al. [13] compared the delays of Dadda and Wallace Multipliers. Waters and Swartzlander [14] proposed a reduced complexity Wallace multiplier, which is alike to the conventional Wallace multiplier but utilizes least number of half adders in reduction of Partial Products. Karthick et. al. [15] proposed different type of 3:2, 4:2 and 5:2 compressors which were based on XOR-XNOR for reduction of Partial Products in Wallace tree multiplier. Gahlan et. al. [16] designed reduced delay Wallace tree multiplier by means of 3:2, 4:2, 5:2, 6:2 and 7:2 compressors with a slight over head of power and area. Mokrian et. al. [17] worked on 4:2 compressors to reduce Partial Products. Ravi et. al. [18] suggested a Sklansky adder Wallace tree multiplier. It produces a high fan out but increased delay. Ramkumar et. al. [19] proposed a new segmented design method of Partial Products for Dadda multiplier which with long word size, proved enhanced improvement in the speed, area and power. Ramanathan et. al. [20] proposed decomposition logic based high speed multiplier with improved speed and parallelism but little increase in power. Krishna and Kumar [21] proposed a parallel hybrid multiplier in which the Partial Products are reduced using Booth Recording Algorithm with less delay. Akhter [22] implemented fast multipliers in VHDL (Hardware Description Language). Vedic and Booth-Wallace multiplication techniques are also implemented to address the issues of power and delay [23-25]. In this paper, four types of Hybrid Tree Multipliers are proposed and designed using conventional CSA Array, Wallace Tree, Dadda Tree and Modified Wallace Tree Reduction Schemes. The aim is to lessen the number of full and half adders involved in reduction phase by exploiting the better attributes of the parent multipliers discussed later in the paper.

## 2. MATERIALS AND METHOD

The importance of a fast, area efficient multiplier cannot be denied in any design and signal processing application. For this purpose the paper proposed hybrid multiplier design for reduction of PPs.

### 2.1 Carry Save Array

When several numbers are added consecutively, there is no need to circulate the carries through each addition. As an alternative, the carries produced in each addition can be kept as partial carries for the next column and then added with the subsequent operand in the next addition step. To be precise; we can speed up each addition by suspending the propagation of previous carry propagation. It shows the approach to the carry save addition technique. The numbers can be added by using a sequence of carry save adders, tracked by a carry propagate addition to get the final result. For multiple operands addition it takes only one carry propagate adder.

The basic idea of carry save reduction technique is that the numbers are reduced to 2 using full adder or half adder while keeping the carry and the sum separate. It leads to the ease of adding all the Partial Products in parallel using a set of adders without waiting for the carries to propagate. It relieves the addition from dependence on the result of the previous column without any carry-chaining. Fig. 1 shows the carry save adder array reduction technique.

In first stage, the CSA array adds up the set of three or two Partial Products and results in partial sum and partial carry. In second stage the same approach is used for incomplete sum and incomplete carry from the earlier stage with operands and turns out into a new partial sum and partial carry and so on until the whole matrix of Partial Products is reduced to two.

### 2.2 Wallace Tree Reduction

In 1964, an Australian Computer Scientist Chris Wallace proposed a method for rapid multiplication which was rooted in adding the Partial Product matrix on parallel by means of a tree of counters (most commonly 3:2 counter) for which it is recognized as the Wallace tree multiplier [6]. In Wallace tree design, the entire bits of the Partial Product matrix in all columns are summed up together using a set of CSA (i.e. full adders/half adders) in parallel without circulating any carries rather saving them for the next column.

Another set of adders then reduces this new matrix which may include the sum of the same column, carry from the last column in the previous stage and the Partial Product bit. This reduction is carried until a two row matrix is reached. The ultimate two rows are added using CPA. In the multiplication of two numbers using Wallace method, as Fig. 2 states, the rows of Partial Product matrix are condensed to two by grouping the Partial products in two or three. Using the carry save adder, three Partial Product terms of same weight are added and reduced to two terms, i.e. the carry and the sum. If there are two Partial Product terms of the same weight, half adder reduces them into two. The sum term is then used by the carry save adder of subsequent level of the same weight.

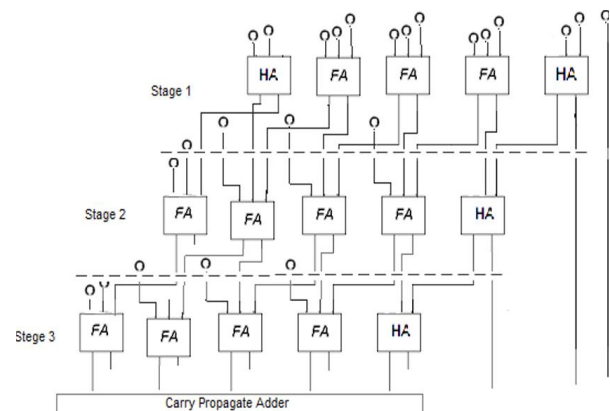


FIG. 1. 5x5 CARRY SAVE ADDERS ARRAY

The carry term corresponds to the adder used in making next output bit of the higher weight. If there lefts only one wire, it is connected to the subsequent layer. Hence the partial product matrix of  $N$  rows for a  $N \times N$  multiplier is reduced to two rows.

For  $8 \times 8$  multiplier, the reduction is achieved in four phases (each phase having the delay of one full adder because of parallel counters used in each level of reduction) with 38 full adders and 15 half adders. The final stage requires 11-bit wide final adder.

### 2.3 Dadda Tree Reduction

In 1965, Dadda presented an alternate multiplier which is similar to Wallace scheme but performs less reduction i.e. whenever required, to get the limits [11]. The Dadda Tree multiplier has the common reduction stages as the Wallace Tree. It groups the Partial Product matrix in groups of three and two and does least reduction necessary at all stage to get the limits. Dadda uses fewer adders during

reduction of the Partial Product rows than that of Wallace Multiplier which reduces Partial Product matrix as soon as possible. Fig. 3 shows the dot notation representation of  $8 \times 8$  Dadda Tree Multiplier. For  $8 \times 8$  bits, Dadda reduction is completed in four phases, identical to the Wallace reduction, using 35 full adders and 7 half adders. The last stage requires a 14-bit extensive adder.

It can be seen that the Dadda multiplier draws on three fewer full adders and eight fewer half adders than Wallace multiplier in Partial Products reduction stage but involves a somewhat larger carry propagating adder in final stage. Hence, Dadda multiplier is less costly in reduction stage but requires larger CPA as it contains more number of bits in each reduction stage.

### 2.4 Modified Wallace Tree Reduction

The Modified Wallace Tree multiplier utilizes half adders and full adders in Partial Product matrix reduction as conventional Wallace Tree multiplier but

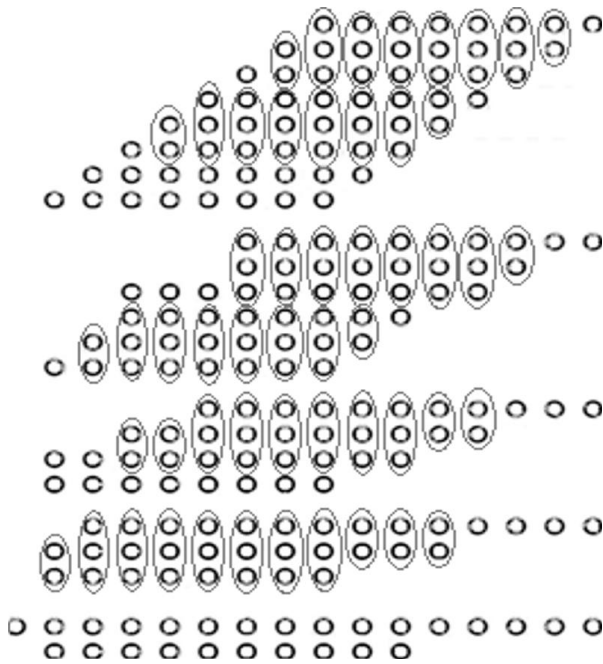


FIG. 2.  $8 \times 8$  WALLACE TREE MULTIPLIER

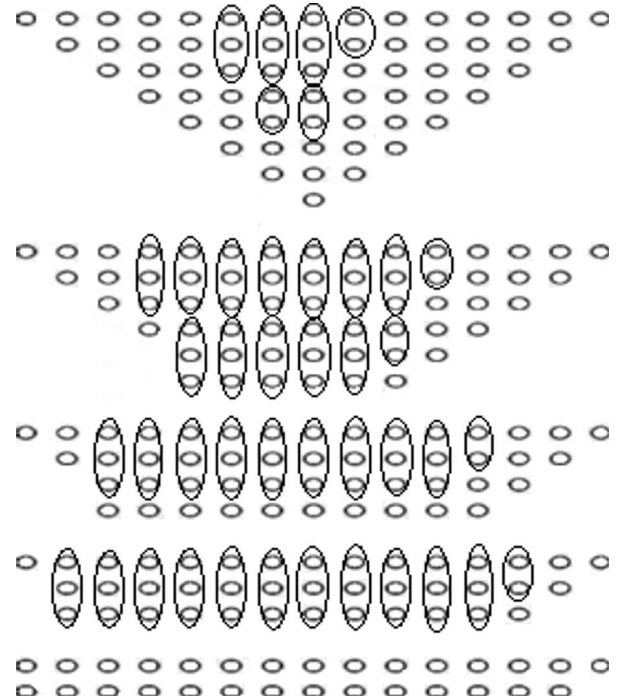


FIG. 3.  $8 \times 8$  DADDA TREE MULTIPLIER

differs in the logic that it uses half adders only when it is to assure that the number of reduction stages is similar as for usual Wallace Tree multiplier. The Modified Wallace Tree initially forms a pyramidal of partial products, divides the structure in three rows and reduces the Partial Products to two rows using fewer numbers of half adders and slightly increased number of full adders than conventional Wallace Tree. It uses slightly large carry propagate adder than conventional Wallace Tree Reduction for the final addition of two rows [10]. Fig. 4 shows the dot notation of Modified Wallace Multiplier.

Modified Wallace Partial Product reduction is done in four stages, like the Wallace & Dadda reduction, using 39 full adders and 3 half adders. The final stage requires a 14-bit wide adder. This reduction method greatly reduces the implementation to 80 percent less half adders than conventional Wallace Tree multiplier by a little increment in number of full adders in Partial Product reduction.

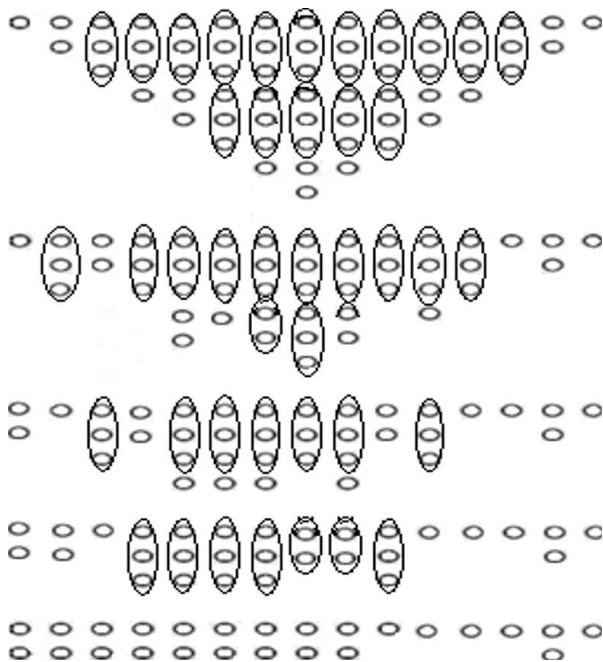


FIG. 4. 8x8 MODIFIED WALLACE MULTIPLIER

## 2.5 Hybrid Multipliers, The Proposed Techniques

Following four types of Hybrid Tree Multipliers are proposed in this paper using conventional CSA Array, Wallace Tree, Dadda Tree and Modified Wallace Tree Reduction Schemes.

- (i) Hybrid of Modified Wallace and Dadda Tree Multiplier
- (ii) Hybrid of CSA Array and Modified Wallace Tree Multiplier
- (iii) Hybrid of CSA Array and Dadda Tree Multiplier
- (iv) Hybrid of CSA Array and Wallace Tree Multiplier

As mentioned earlier, for a  $N \times N$  multiplier, there are  $N$  rows of partial product matrix each having  $N$  bits. In the proposed designs  $N$  rows of Partial Product matrix are reduced using one reduction scheme while the other half are reduced using some other scheme. The results of all the hybrid multipliers are compared in terms of number of full adders and half adders and gate counts. Fig. 5 shows one of the proposed 8x8 hybrid multiplier using Modified Wallace and Dadda Reduction schemes. Same approach is used for other proposed hybrid multipliers as shown in Fig. 5.

## 2.6 Methodology

The multiplication process of the proposed  $N \times N$  hybrid multipliers starts with the production of Partial Product matrix using  $N^2$  AND gates. Half rows of the Partial Product matrix are reduced to two rows using X reduction technique (one of the mentioned PP reduction techniques) and the remaining half Partial Products rows are reduced using Y (other) reduction technique. After the reduction of rows of Partial Product matrix of both

the X and Y reduction schemes, the two rows of both the reduction techniques are summed using carry propagate adders and the final sum of the multiplication is concluded using another carry propagate adder. The proposed hybrid techniques considerably decrease number of full adder and half adders than the parent techniques but use total of three smaller carry propagate adders. Fig. 6 shows the methodology of the proposed 8x8 Hybrid Multipliers.

The proposed hybridization of multipliers focuses on reduction of Partial Products in multiplication. The increase in number of carry propagate adders can be handled by using any fast, area efficient adder. Hybrid of ripple carry adder and carry look ahead adder is the best choice in the final phase of multiplication i.e. for the final sum.

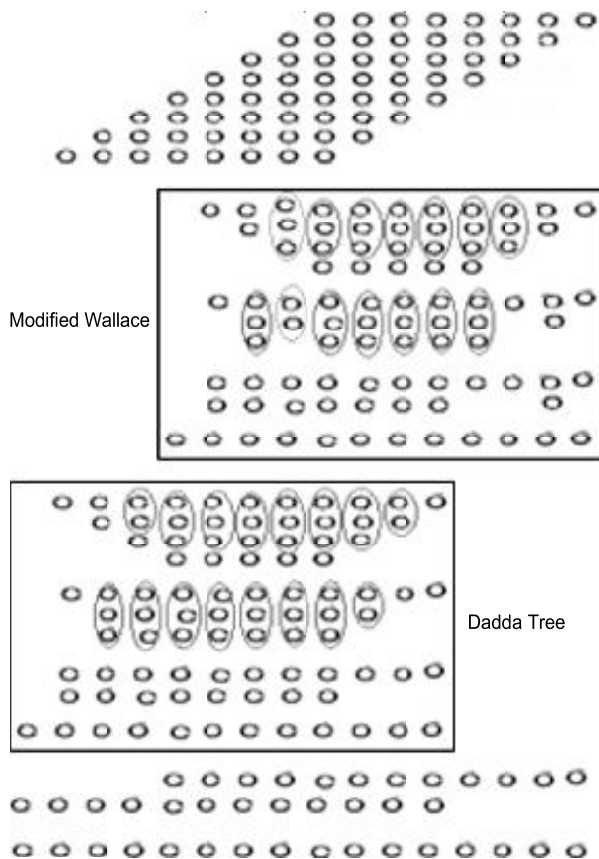


FIG. 5. ONE OF THE PROPOSED HYBRID TREE MULTIPLIERS

### 3. RESULTS AND DISCUSSION

The second phase of multiplication i.e. Partial Product reduction is analyzed for 8, 12 and 16 bits using the hybridization of conventional CSA Array, Wallace Tree, Dadda Tree and Modified Wallace Tree Reduction Multipliers. The results shown in Fig. 7 reveal that the proposed Hybrid

Reduction Multipliers show significant decrease in number of full adders, half adders and hence the total gates in reduction phase of multiplication than the parent reduction techniques. Number of full adders, half adders, and overall no. of gates for parent multipliers are given in Table 1 for the sake of reference.

The total gate count shown in Tables 1-2 is found from gate level designs; where full adders are realized using nine gates and half adders are realized using four gates (each XOR gate is equal to three gates). Only reduction phase is included in the tables, neither the  $N^2$  AND gates of partial product generation nor the carry propagating adders of final sum are included.

The results of Table 2 show that all of the proposed Hybrid Multipliers for reduction of Partial Products are better in number of gate counts and use fewer full adders and half adders than their respective parent multiplier. Hybrid of Modified Wallace and Dadda shows the best results amongst all other hybrid multipliers. The proposed multipliers can require a slight larger adder in the final addition phase of multiplication process. The Verilog implementation of proposed technique having fewest gates than all the proposed and parent Tree multipliers is simulated and synthesized on Xilinx 14.2 and results are shown in Fig. 8. Design parameters (i.e. area, delay etc) calculated using Xilinx are presented in Table 3.

All of the proposed hybrid multipliers exploit the good traits of the parent multipliers, like in Hybrid of Modified Wallace and Dadda Multiplier, the Modified Wallace Multiplier uses fewer half adders than Dadda Multiplier but requires somehow lengthy adder in the final stage of the multiplication, and Dadda Multiplier requires a slight

more half adders than the Modified Wallace Multiplier but requires adder of shorter length for final stage of addition.

Both of the techniques combine to make an appropriate scheme which uses less full and half adders for reduction than the parent multipliers.

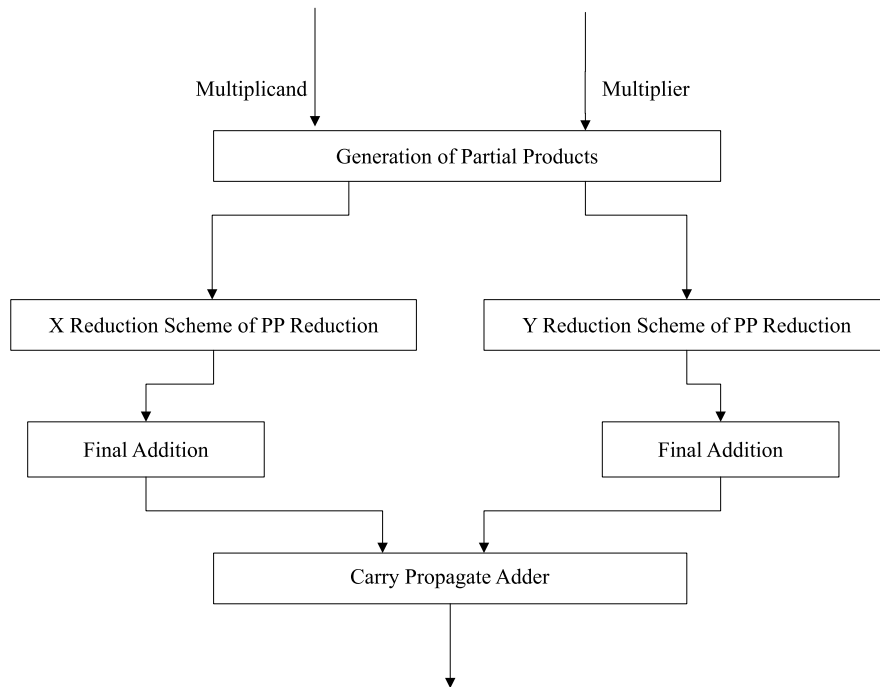


FIG. 6. PROPOSED ARCHITECTURE OF 8x8 HYBRID MULTIPLIERS

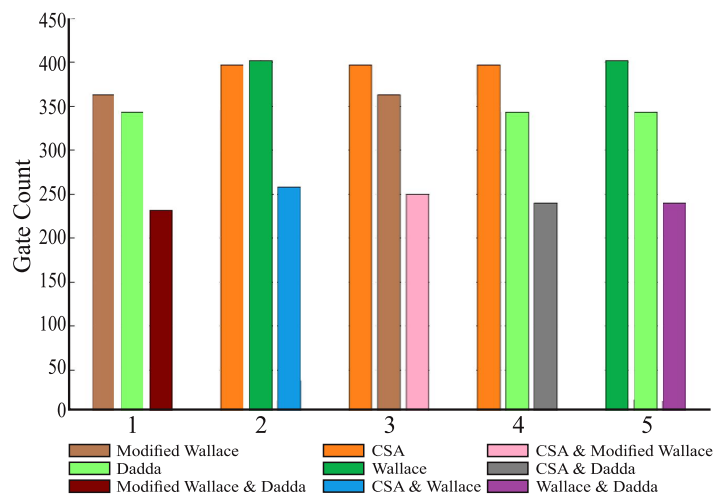


FIG. 7. COMPARISON OF GATE COUNTS OF 8x8 PROPOSED MULTIPLIERS WITH THEIR PARENT MULTIPLIERS

TABLE 1. GATE COUNT RESULTS OF PARENT TREE MULTIPLIERS

	Input Size (N)	8-Bit	12-Bit	16-Bit
Dadda	Full Adders	35	99	195
	Half Adders	7	11	15
	Total Gate	343	935	1,815
Wallace	Full Adders	38	102	200
	Half Adders	15	23	52
	Total Gate	402	1,010	2,008
Modified Wallace	Full Adders	39	104	201
	Half Adders	3	6	9
	Total Gate	363	9,60	1,845
CSA Array	Full Adders	41	109	209
	Half Adders	7	11	15
	Total Gate	396	1,025	1,941

TABLE 2. GATE COUNT RESULT OF PROPOSED HYBRID TREE MULTIPLIERS

	Input Size (N)	8-Bit	12-Bit	16-Bit
Modified Wallace & Dadda	Full Adders	24	81	170
	Half Adders	4	7	10
	Total Gate	232	757	1,570
CSA & Wallace	Full Adders	26	84	175
	Half Adders	6	14	22
	Total Gate	258	812	1,663
CSA & Modified Wallace	Full Adders	26	84	175
	Half Adders	6	14	22
	Total Gate	258	812	1,663
CSA & Dadda	Full Adders	26	85	176
	Half Adders	4	7	10
	Total Gate	250	793	1,624
Wallace & Dadda	Full Adders	24	80	169
	Half Adders	6	14	22
	Total Gate	240	776	1,609

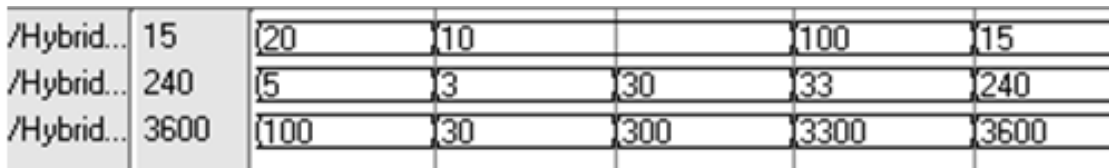


FIG. 8. SIMULATION WAVEFORM OF THE HYBRID OF MODIFIED WALLACE AND DADDA TREE MULTIPLIER



**TABLE 3. SYNTHESIS RESULTS**

	Dadda Tree Multiplier	Wallace Tree Multiplier	Hybrid Multiplier
Number of 4 Input LUTs	144	155	141
Number of Occupied Slices	85	87	78
Delay (ns)	31.974	33.413	31.885

#### 4. CONCLUSION

This paper presents the design of four hybrid tree multiplier technique using conventional CSA Array, Wallace Tree, Dadda Tree and Modified Wallace Tree Reduction Schemes. The analysis of the proposed hybrid multipliers is presented in terms of number of full adders, half adders and total gate counts in reduction phase (which lessens N rows of Partial Products to two). The hybrid multipliers use less number of gates than their parent multipliers with the penalty of the final phase carry propagate adders. All of the proposed multipliers have equal number of phases of reduction and thus the delay is likely to be very similar. Moreover, Hybrid of Modified Wallace and Dadda multiplier shows minimum number of gate counts when compared with other hybrid multipliers.

#### ACKNOWLEDGMENT

This work has been supported by The University of Engineering & Technology, Taxila, Pakistan.

#### REFERENCES

<p>[1] Kumar, G., Prasannam, D., and Christy, A., "Analysis of Low Power, Area and High Speed Multipliers for DSP Applications", International Journal of Emerging Technology &amp; Advanced Engineering, Volume 4, No. 3, pp. 278-282, India, 2014.</p> <p>[2] Kakde, S., "Design of Area and Power Aware Reduced Complexity Wallace Tree Multiplier", IEEE International Conference on Pervasive Computing, 2015.</p>	<p>[3] Vasudev, G., and Hegadi, R., "Design and Development of 8-Bits Fast Multiplier for Low Power Applications", International Association of Computer Science &amp; Information Technology, Volume 4, No. 6, pp. 774-780, 2012</p> <p>[4] Habibi, A., and Wintz, P., "Fast Multipliers", IEEE Transactions on Computers, Volume C-19, pp. 153-157, USA, 1970.</p> <p>[5] Aradhya, H.V. and Ravish, "Design and Performance Comparison of Adiabatic 8-Bit Multipliers", IEEE Distributed Computing, VLSI, Electrical Circuits and Robotics, 2016.</p> <p>[6] Gowreesrinivas, K.V., and Samundiswary, P., "Design and Implementation of Efficient Multiply Accumulate Unit Using Vedic Multiplier and CSLA", 3<sup>rd</sup> International Conference on Electrical, Electronics, Engineering Trends, Communication, Optimization and Sciences. 2016.</p> <p>[7] Oklobdzija, V., Vileger, D., and Liu, S., "A Method for Speed Optimized Partial Product Reduction and Generation of Fast Parallel Multipliers using an Algorithmic Approach", IEEE Transaction on Computers, Volume 45, No. 3, pp. 294-306, USA, 1996.</p> <p>[8] Ramanathan, P., Kowsalya, P., and Anitha, P., "Modified Low Power Wallace Tree Multiplier Using Higher Order Compressors", International Journal of Electronic Letters, Volume 5, No. 2, pp. 177-188, 2016.</p> <p>[9] Parhami, B., "Computer Arithmetic: Algorithms and Hardware Designs", Oxford University Press, New York, 2000.</p> <p>[10] Wallace, C., "A Suggestion for a Fast Multiplier", IEEE Transactions on Electronic Computers, Volume EC-13, pp. 14-17, 1964.</p>
---	--

- [11] Dadda, L., "Some Schemes for Parallel Multipliers", *Alta Frequenza*, Volume 34, pp. 346-356.
- [12] Baugh, C., and Wooley, B., "A Two's Complement Parallel Array Multiplication Algorithm", *IEEE Transaction on Computers*, Volume C-22, No. 12, pp. 1045-1047, USA, 1973.
- [13] Townsend, J., Earl, J., Swartzlander, E., and Abraham, A., "A Comparison of Dadda and Wallace Multiplier Delays", *Advanced Signal Processing Algorithms, Architectures, and Implementations*, Volume 5205, pp. 552-560, Austin, 2003.
- [14] Waters, R., and Swartzlander, E., "A Reduced Complexity Wallace Multiplier Reduction", *IEEE Transactions*, Volume 59, No. 8, pp. 1134-1137, USA, 2010.
- [15] Karthick, S., Karthika, S., and Valannathy, S., "Design and Analysis of Low Power Compressors", *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, Volume 1, No. 6, pp. 487-493, India, 2012.
- [16] Gahlan Kr., Shukla, P., and Kaur, J., "Implementation of Wallace Tree Multiplier Using Compressor", *International Journal of Circuit Theory and Applications*, Volume 5, No.3, pp.1194-1199, India, 2012.
- [17] Mokrian, P., Howard, G., Jullien, G., and Ahmadi, M., "On the Use of 4:2 Compressors for Partial Product Reduction", *IEEE Transaction on Computers*, Volume 45, No. 3, pp. 7781-8, USA, 2003.
- [18] Ravi, D., Sankar, R., and Ali, A., "Design of Wallace Tree Multiplier by Sklansky Adder", *International Journal of Engineering Research and Applications*, Volume 4, No. 4, pp. 1036-1040, India, 2013.
- [19] Ramkumar, B., Sreedeeep, V., and Kittur, M., "A Faster Design Technique for Dadda Multiplier", *The School of Electronics Engineering*, VIT University, Vellore, 2011.
- [20] Ramanathan, P., Vanathi, P., and Agarwal, S., "High Speed Multiplier Design Using Decomposition Logic", *Serbian Journal of Electrical Engineering*, Volume 6, No. 1, pp. 33-42, Serbia, 2009.
- [21] Krishna, D., and Kumar, V., "Convolution Using Delay Efficient Improved Hybrid Multiplier", *International Journal of Science, Engineering and Technology Research*, Volume 4, No. 6, pp. 4924-4930, India, 2015.
- [22] Akhter, S., "VHDL Implementation of Fast NxN Multiplier," *IEEE Transaction on Computers*, Volume 4, No. 7, pp. 1327-1334, India, 2007.
- [23] Poornima, M., Patil, S.K., Shivukumar, Shridhar, K.P., and Sanjay, H., "Implementation of Multiplier Using Vedic Algorithm", *International Journal of Innovative Technology and Exploring Engineering*, Volume 2, 2013.
- [24] Kumar, J., and Mahapatra, K., "Design of an Adaptive Hearing Aid Algorithm Using Booth-Wallace Tree Multiplier", *International Journal of Logic and Computation*, Volume 1, No. 1, pp. 1-17, India, 2010.
- [25] Sakthivel, R., Vanitha, M., and Singh, S., "Low Leakage Power Vedic Multiplier using Standard Cell Design", *Indian Journal of Science and Technology*, Volume 8, No. 24, pp. 1-5, India, 2015.