International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

DESIGN AND IMPLEMENTATION OF HIGH SPEED PARALLEL PREFIX ADDER BASED MULTIPLIER ARCHITECTURE

[1]Rajeev Sunkara, [2]Dr.M.Raghava Naidu
[1]Assistant professor, Dept. of ECE, Krishna University, Machilipatnam, Krishna District, Andhra Pradesh, India.
[2]Assistant professor, Dept. of CSE, Krishna University, Machilipatnam, Krishna District, Andhra Pradesh, India.

ABSTRACT: Multipliers are the essential computational capacities that are broadly utilized in DSP based Applications. The complexity of the circuit depends mainly on the multiplication count. In any VLSI design mainly power, area and speed play a massive role, by improving any of these parameters the overall performance will be improved. Among various multiplication algorithms, multiplication algorithm is beneficial in terms of speed of operation. This paper presents design and implementation of high speed Parallel Prefix Adder based Multiplier architecture. In the proposed structure, the final addition stage of partial products is performed by parallel prefix adders (PPAs). The use of PPA reduces the complexity required by the sums of the partial product lines. The entire operation of proposed system depends upon three stages they are multiplier partial product generation, reduction stages and parallel prefix adder. The delay gets reduced by achieving low logical depth in the system. The designed multiplier is simulated and synthesized using Xilinix ISE 13.2. The obtained simulation results prove that the described structure performance is better in terms of Propagation delay, memory used when compared with the other multipliers.

KEYWORDS: Parallel Prefix Adder, Multiplier, Partial Product Generation, Xilinix ISE 13.2.

I. INTRODUCTION

As we know that in the areas of system on chip and VLSI designs, the low power circuit designs is an important issue. In the sub micro regions, the transistor produces the leakage currents which are very significant in nature [1]. This leakage current is controlled by introducing the new designs. Two logics are used mainly to improve the performance, adiabatic logic and asynchronous logic. Compared to the static CMOS logic design, the adiabatic Logic uses less power to perform the operations.

So adiabatic logic is most used in low power applications. As we know that addition, multiplication and subtraction operations are performed in any system. While executing the multiplication operations, large numbers of problems are obtained related to computations. Because of this changes are occurred in the speed of the system. In DSP systems multiplier plays an important role. Several multiplication algorithms are present such as Binary multiplier, array multiplier, Booth's multiplier [2], Dadda multiplier.

A multiplier is a hardware implementation of binary multiplier, a digital circuit that multiplies two integers [3]. It uses a selection of full adders and half adders to sum partial products in stages until two numbers are left. Multipliers reduce as much as possible on each layer, whereas Dadda multipliers try to minimize the required number of gates by postponing the reduction to the upper layers. Operation of WTM is same as the first stage of multiplication and it generates the partial product. In traditional modal half adders and full adders are used for the reduction of partial products. It adds the first three rows of partial product. Then generated sum and carry is added with next row of partial products. This process continues until the final stage which has only two rows of sum and carry. Final stage is also reduced by adders like carry look ahead adder, carry select adder

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

parallel prefix etc. thus adders plays a important role in the generation of final product terms. The speed of addition is going to effect the operation of the multiplication [4].

Basically, parallel prefix adder produces high speed multi operands. The scaling down of device dimensions into the Nano-meter range is likely to result in significantly higher defect rates during the manufacturing process of IC's [5]. With significantly increased defect rates, defect tolerance mechanisms are necessitated to guarantee a reasonable yield. Post manufacturing reconfiguration techniques to bypass defects are already applied in memory systems and FPGA's. However, such low-cost defect tolerance techniques rely heavily on the relative independence of operations of the homogeneous components, such as LUT and memory cells. Logic systems, on the other hand, usually constitute heterogeneous components with strong dependencies among each other. This makes it hard to realize fine-grained, low-cost defect tolerance schemes for a high level of defect rate.

In PPAs, a carry generation tree is present which generates carry for all preceding stages which improves the speed of operation. The carry generation tree mainly consists of two components-black cell and grey cell. The black cell and grey cell are interconnected to from carry tree network. Carry generation tree block is also called as parallel carry generation block as it generates carry bits for all stages at a time parallel. There are different types of PPAs whose classification mainly depends on two factors:

i) Number of black and grey cells in carry generation tree

ii) Interconnection of black and grey cells in carry generation tree

This paper is organized as into various sections –section II explains the literature survey, Section III is composed of many sub sections which describes in depth operation of PPA, different PPAs used in the design of multiplier and the structure of proposed multiplier using PPAs. The result analysis is clearly explained in section IV. Section V is conclusion of design proposed in this paper.

II. LITERATURE SURVEY

Gunjan Jain, Meenal Jain, Gaurav Gupta, et.al. [6], Design of radix-4,16,32 approx booth multiplier using Error Tolerant Application. This multiplier makes significant improvements in power, delay, and area at the expense of reducing accuracy. The approximate multiplier is to use less power and area than conventional truncated multiplier, and keep high speed. Its main purpose is to speed up the speed of arithmetic circuits as well as spend low power. This Error Tolerant Application easily restrict on accuracy and achieve various improvement in power consumption and speed accuracy.

K. Deergha Rao;Ch. Gangadhar, Praveen K Korrai, et. al. [7], presents FPGA implementation of complex multiplier using minimum delay Vedic real multiplier architecture. two possible architectures are proposed for a Vedic real multiplier based on the URDHVA TIRYAKBHYAM (Vertically and cross wise) sutra of Indian Vedic mathematics and an expression for path delay of an N×N Vedic real multiplier with minimum path delay architecture is developed. Then, architectures of four Vedic real multipliers solution, three Vedic real multipliers solution of complex multiplier are presented. The architecture of Vedic real multiplier with minimum path delay is used in the implementation of complex multiplier. Finally, the results are compared with that of the four and three real multipliers solutions using the conventional Booth and Array multipliers.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

B. Dinesh, V. Venkateshwaran, P. Kavinmalar, M. Kathirvelu, et. al. [8] presents comparartive analysis. Multipliers are key components of many high performance systems such as FIR filters, microprocessors, digital signal processors, etc. Among the tree based multipliers Dadda multipliers have a slight advantage over tree multipliers in terms of performance. The Modified booth multiplier is comparatively inefficient for bits lesser than or equal to 4, due to the increased area involved for realization of the booth encoder and booth selector blocks. The analysis shows that for lower order bits Dadda reduction is the most efficient.

Ravindra P. Rajput;M.N. Shanmukha Swamy, et.al. [9], they describe the design and implementation of signed-unsigned Modified Booth Encoding (SUMBE) multiplier. The modified Booth Encoder circuit generates half the partial products in parallel. By extending sign bit of the operands and generating an additional partial product the SUMBE multiplier is obtained. The Carry Save Adder (CSA) tree and the final Carry Look ahead (CLA) adder used to speed up the multiplier operation. Since signed and unsigned multiplication operation is performed by the same multiplier unit the required hardware and the chip area reduces and this in turn reduces power dissipation and cost of a system.

Jaberipur G., and Kaivani A., et. al., [10], have examined the previous two solutions and the effect of PPR method on the design of the parallel decimal multiplier and selected the best possible approach. A complete new PPG approach was generated and CSA technique is used to lower down the effect of PPR technique. The multiplicand's 2, 5, 8 and 9 multiples are computed directly using this proposed method which makes the computation quite fast. This PPR approach is applied to both 16-bit as well as the 32-bit decimal multipliers. The delay incurred due to this PPR approach is much low than compared to any other previous one. The PPR result is represented in double-BCD format. This approach improves the speed by 16% when compared to the previous design.

III. MULTIPLIER USING PARALLEL PREFIX ADDER

The Fig. 1 shows the architecture of proposed system. Estimation of partial products and gathering of the shifted partial products are two main processes of basic multiplication principle It is performed by the successive additions of the columns of the shifted partial product matrix. The 'multiplier' is effectively shifted and gets the suitable bit of the 'multiplicand'. The product bits are from by added them again. Therefore, Multiplication is a multi operand operation. The entire system is divided into different modules. The following modules are multiplier, multiplicand, pre-processing stage, hybrid carry generation stage and post processing stage. Now multiplier and multiplicand bits are given as input to pre-processing stage. Then the alignment of partial products generator will be done. Now these bits perform the addition operation using hybrid model of Brent–Kung and Ladner-Fischer, and give the final product by subsequently and saved in the post processing stage.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

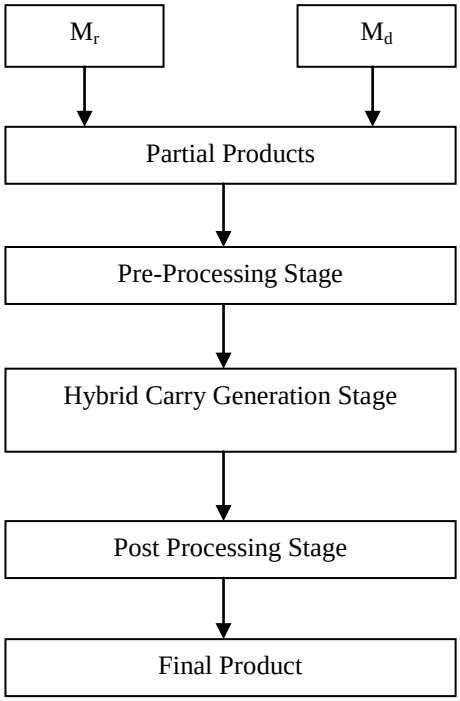

Fig. 1: BLOCKDIAGRAM OF MULTIPLIER OPERATION

A and B are two operands loaded in the multiplier at first and considered as input. The arithmetic operations like addition and multiplication operations are performed. The obtained result of this will be saved in the shifter. Here irreducible polynomial function is not used in the system. The main intent of register multiplier is to store the bit representation and give polynomial output a(t). Here parallel load operation is performed in the most significant bit position. In the same way left shift operations are performed in MSB bit. The multiplicand bit is used b(t) value to store the value in register. The parallel load operation is also applied in the multiplicand. The obtained value is stored in the register. The right shift operation is performed in the multiplicand register block. Crypto core processor is used to transfer the data in multiplicand register.

It computation a pair of signals in this phase namely generate and propagate signals, which corresponds to each i^{th} state of A and B input sets. Propagate and generate signal are represent as shown below:

$$P_i = A_i \; XOR \; B_i \; \ldots . (1)$$

$$G_i = A_i \; AND \; B_i \; \ldots . . (2)$$

Result register consists of output and saves the entire arithmetic result. The both a(t) and b(t) values are assigned in the barrel shifter blocks. The obtained values in the barrel shifter block will shift the bits to adder block. This block will perform the addition operation. After performing particular operation, the bits are shifted to the result register. This result register will save the output as product. At last the barrel shifter will perform the parallel operation in effective way.

Parallel Prefix Adder is used to find the arithmetic sum of two numbers which is more than one bit in length and corresponding pairs of bits are processed in parallel form. The Ladner-Fischer is the parallel prefix adder used to perform the addition operation. It is looking like tree structure to perform the arithmetic operation. The Brent–Kung adder is a parallel prefix adder (PPA) form of carry-lookahead adder (CLA). A Brent–Kung adder is a parallel adder made in a regular layout with an aim of minimizing the chip area and ease of manufacturing.

The efficiency of a squaring circuit can be improved by using parallel adders. A carry and sum are generated by the parallel adder at last the MSB to be added. An ‘n’ bit parallel adder needs ‘n’ full adders to perform the operation. In this phase the carries are computed much earlier using the two cells described below.

Grey Cell: It consumes two pairs of generate and propagate signals (Gi, Pi) and (Gj, Pj) as inputs and Computes a single generate signal G as output for the current stage

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

$$G = G_i OR\left(P_i AND P_j\right) \ldots\ldots (3)$$

Black Cell: It consumes two pairs of generate and propagate signals (Gi, Pi) and (Gj, Pj) as input and computes a pair of signals as generate and propagate signals (G, P) as output for the current stage:

$$G = G_i OR\left(P_i AND P_j\right) \ldots\ldots (4)$$

$$P = P_i AND P_j \ldots. (5)$$

The last phase to compute sum is common to all adders of PPA adder family. This stage involves the computation of sum bits which is given by

$$S_i = P_i XOR\ C_{i-1} \ldots (6)$$

IV. IMPLEMENTATION RESULTS

Multiplier structures using five different PPA's (hybrid model of Brent–Kung and Ladner-Fischer adder, Sklansky adder, Kogge stone adder and Ladner Fischer adder) are designed for input size of 16-bits in this paper using Verilog HDL. All the designed multipliers are simulated and synthesized using Xilinix ISE 13.2. The simulation waveforms for proposed multipliers are shown in below figures. Fig. 2 shows the RTL Schematic of described multiplier.

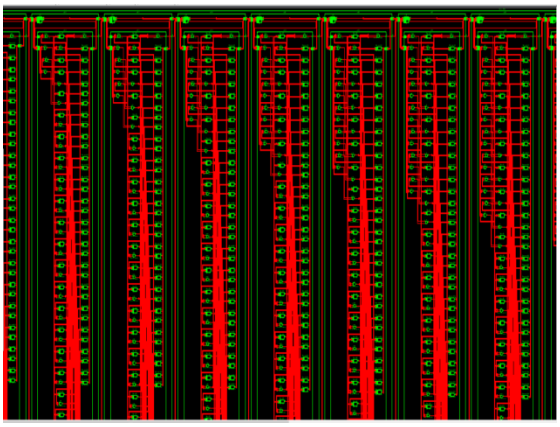

Fig. 2: RTL SCHEMATIC

Fig. 3 shows the simulation output waveforms of described model.

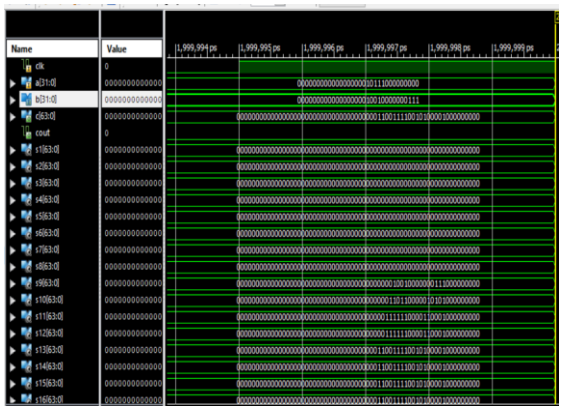

Fig. 3: OUTPUT WAVEFORMS

Fig. 4 shows the technology schematic of described multiplier.

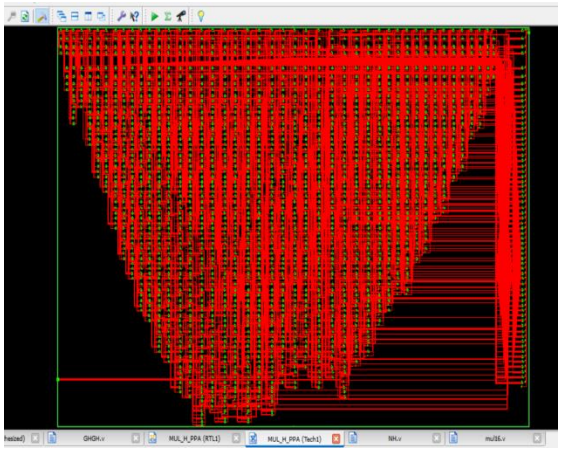

Fig. 4: TECHNOLOGY SCHEMATIC

As shown in Table 1, total four multipliers i.e., the proposed multipliers using PPAs in this paper are three and one is traditional multiplier are synthesized using Xilinx XST synthesizer.

Table 1: COMPARATIVE ANALYSIS

Multiplier structure	Area (no. of LUT's)	Delay (ns)
Multiplier using Brent–Kung and Ladner-Fischer adder	524	28.44
Multiplier using Sklansky adder	599	30

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

Multiplier using Ladner fischer adder	596	31.57
Multiplier using Kogge stone adder	590	36.7

From Table 1, it can be seen that multiplier using hybrid model of Brent–Kung and Ladner-Fischer adder is having least delay but it has more number of LUTs occupied when compared to other structures. The multiplier using ladner fischer adder is having least number of LUTs occupied where as its delay is medium i.e, 28.44ns which is less when compared with the traditional structure.

V. CONCLUSION

In this paper, design and implementation of high speed Parallel Prefix Adder based Multiplier architecture. Multipliers are the essential computational capacities that are broadly utilized in DSP based Applications. The entire operation of proposed system depends upon three stages they are multiplier partial product generation, reduction stages and parallel prefix adder. All the designed multipliers are simulated and synthesized using Xilinix ISE 13.2. Hybrid model of Brent–Kung and Ladner-Fischer adder is having least delay but it has more number of LUTs occupied when compared to other structures. The multiplier using ladner fischer adder is having least number of LUTs occupied where as its delay is medium.

VI. REFERENCES

[1] Preeti Verma, Alok K Mishra, "Implementation and Analysis of Efficient Low Power Dynamic Circuit Tec hnique", 2021 2nd International Conference on Smart Electronics and Communication (ICOSEC), Year: 2021

[2] S Rooban, M Nagesh, M V S L Prasanna, K Rayudu, G Dheeraj Sai, "Implementation of 128-bit Radix-4 Booth Multiplier", 2021 International Conference on Computer Communication and Informatics (ICCCI), Year: 2021

[3] Navin Kumar M., R.S. Adithyaa, Bhavan Kumar D., T. Pavithra, "Design Analysis of Wallace Tree based Multiplier using Approximate Full Adder and Kogge Stone Adder", 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), Year: 2020

[4] Nidhi Gaur, Anu Mehra, Pradeep Kumar, Sankalp Kallakuri, "16 Bit Power Efficient Carry Select Adder", 2019 6th International Conference on Signal Processing and Integrated Networks (SPIN), Year: 2019

[5] Shilpa K.C., Shwetha M., Geetha B.C., Lohitha D.M., Navya, Pramod N.V., "Performance Analysis of Parallel Prefix Adder for Datapath Vlsi Design", 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), Year: 2018

[6] Gunjan Jain, Meenal Jain, Gaurav Gupta, "Design of radix-4,16,32 approx booth multiplier using Error Tolerant Application", 6th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), 2017.

[7] K. Deergha Rao, Ch. Gangadhar, Praveen K Korrai, "FPGA implementation of complex multiplier using minimum delay Vedic real multiplier architecture", IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics Engineering (UPCON), 2016.

[8] B. Dinesh, V. Venkateshwaran, P. Kavinmalar, M. Kathirvelu, "Comparison of regular and tree based multiplier architectures with modified booth encoding for 4 bits on layout level using 45nm technology", International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE), 2014

[9] Ravindra P. Rajput, M.N. Shanmukha Swamy, "High Speed Modified Booth

Encoder Multiplier for Signed and Unsigned Numbers", UKSim 14th International Conference on Computer Modelling and Simulation, 2012

[10] Jaberipur G., and Kaivani A., "Improving the Speed of Parallel Decimal Multiplication", IEEE Transactions on Computers, 2009