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

DESIGN A HIGH SPEED AND HIGH EFFICIENT ACCURATE MULTIPLIER FOR COMMUNICATION APPLICATIONS

[1]V. LATHA MADHURI, [2]K. MANOJ PAVAN KUMAR, [3]SRINIVASARAO
[1]M.Tech Scholar, Dept of ECE, Chebrolu Engineering College, Chebrolu, Andhra Pradesh, India

ABSTRACT:
The design of accurate multipliers which were applied in image processing has many applications and they lead to less delay and power. In any general purpose processor the use of conventional full precision multipliers results in increase in the power, area and computational time. So, multipliers being the basic key element of any computation unit take its own importance in decreasing the power as well as increase in the speed. In this project, design of high speed and high efficient accurate multiplier for communication applications is proposed. The two input multiplier and multiplicand are divided in to two groups of bits each according to their computational significance in product generation. Scaling of these bits and ordering can be done by using a shifter. Then the multiplication can be performed and generates partial products that are aligned. After that an accurate adder is used to add the products and carry propagation is also performed in order to obtain an accurate final product.

KEY WORDS: VLSI, Digital signal processing (DSP), Partial products, Barrel Shifter and Adder, look-up-table

I. INTRODUCTION

Numerous inexorably well known applications, for example, picture preparing and acknowledgment, are inalienably tolerant of little mistakes. These applications are computationally requesting and increase is their major number juggling capacity, which makes a chance to exchange off computational exactness for decreased power utilization. Basically, fault tolerant system produces the unpleasant figuring in the applications of fault tolerant. Because of this the trade of f delay and power will provide accurate results. In some applications, the fault tolerant application requires distinctive precision.

This will mainly used in the program stages of fault tolerance system. Here the programs stages will run depend upon the precision control. If the accuracy is fixed then the precision is not required. In the same way the accuracy is not fixed then the precision is required. Hence for diverse applications the precision requirement is necessary. There should be reconfigurable of multipliers in various program stages applications. So in this paper we designed a multiplier which will control the precision in effective way.

Multiplication is one of the basic arithmetic operations, used extensively in the domain of digital signal and image processing [1-2]. FPGA vendors, such as Xilinx and Intel, provide DSP blocks to achieve fast multipliers [3]. Despite the high performance offered by these DSP blocks, their usage might not be efficient in terms of

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

overall performance and area requirements for some applications. For small applications, it may be possible to perform manual Floor planning to optimize an application's overall performance. However, for complex applications with contending requirements for FPGA resources, it may not be possible to optimize the placement of required FPGA resources manually to enhance performance gains.

In earlier period multiplication was generally implemented by the use of addition, subtraction, and shift operations [5]. Multiplication can be painstaking as a series of repetitive additions. The product is obtained by adding the number i.e., multiplier with another number many times i.e., multiplicand. Partial product is generated at each step of addition. In many processors, the operand frequently contains the same number of bits. When the operands are considered as integers, the product length is generally twice the length of operands due to No loss of information contain. This repetition addition increases the length of product so as to area by the arithmetic algorithm that is slow which has to replace by an algorithm that makes use of positional representation. It can be possible to divide multipliers into two parts [6]. The first part is assigned to the generation of partial products term, and the second one collects and adds them. The basic multiplication principle is having two main processes i.e. estimation of partial products and gathering of the shifted partial products. It is performed by the successive additions of the columns of the shifted partial product matrix [7]. The 'multiplier' is effectively shifted and gets the suitable bit of the 'multiplicand'. The product bits are from by added them again. Multiplication is, therefore, a multi operand operation. To broaden the multiplication to both signed and unsigned numbers, a convenient number system would be the representation of numbers in two's complement format.

II. LITERATURE SURVEY

K. Deergha Rao;Ch. Gangadhar;Praveen K Korrai, *et.al.*, 2016 [8], complex numbers multiplication is a key arithmetic operation to be performed with high speed and less consumption of power in high performance systems such as wireless communications. Hence, in this paper, 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. The architectures for the four multiplier solution and three multiplier solution of complex multiplier for 32 × 32 bit complex numbers multiplication are coded in VHDL and implemented through Xilinx ISE 13.4 navigator and Modelsim 5.6.

B. Dines;V. Venkateshwaran;P. Kavinmalar;M. Kathirvelu, *et.al.*, 2014 [9], multipliers are key components of many high performance systems such as FIR filters, microprocessors, digital signal processors, etc. A system's performance is generally determined by the performance of the multiplier as the multiplier is generally the slowest element in the system. The analysis of performance parameters of different multiplier logics is essential for

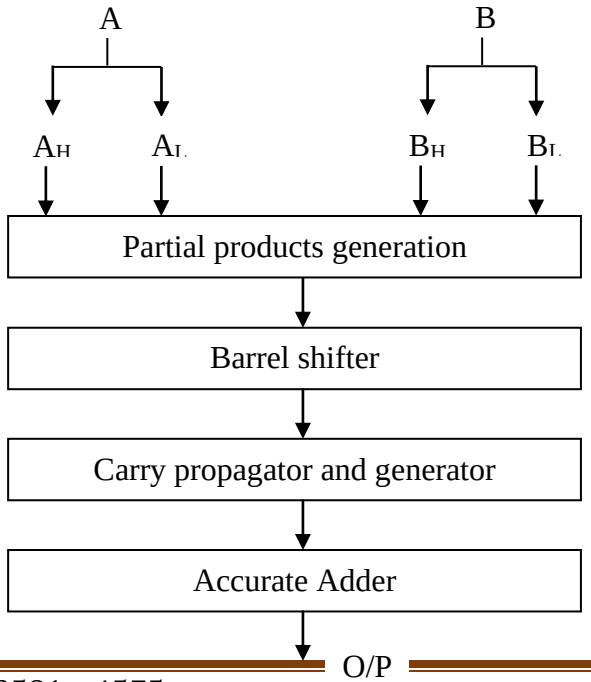
International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

design of a system intended for a specific function with constraints on Power, Area and Delay. The paper presents a detailed analysis of all the serial-parallel and parallel architectures. The multipliers are designed for 4 bit multiplication using DSCH tool and the corresponding layouts are obtained using Microwind 3.5 tool using 45nm technology. From the analysis it is observed that the array multipliers provide a regular routing structure which will be optimum for FPGA based systems. Among the tree based multipliers Dadda multipliers have a slight advantage over Wallace tree multipliers in terms of performance.

Leandro Zafalon Pieper;Eduardo A. C. da Costa;José C. Monteiro, *et.al.*, 2013 [10], they describe the design of efficient 2's complement 64-bit array multipliers. We propose the combination of radix-2^m dedicated multiplier blocks and adder compressors that leads to the reduction of partial product lines, and hence to the higher performance and least power consumption. The flexibility of the architecture allows for the easy construction of multipliers for different values of m. With the use of optimized radix-2^m dedicated multiplication blocks the multiplication can be naturally extended up for radix-256 multiplication. Since in the radix-2^m multiplier the number of partial lines is reduced, by the multiplication of m bits at a time, we have used a combination of 4:2, 8:2 and 16:2 adder compressors in order to speed-up the addition of the simultaneous operands. We present results of area, delay and power consumption by using SIS and SLS (Switch Level Simulator) tools. The results show that combining the use of dedicated multiplier modules with adder compressors the radix-2^m array multipliers are more efficient in terms of delay and power consumption when compared with both a Radix-2^m array multiplier with Ripple Carry Adders (RCA) in the partial product lines, previously presented in literature, and the Modified Booth multiplier.

III. PROPOSED ACCURATE MULTIPLIER

The below figure (1) representing the architecture of accurate multiplier for digital signal processing applications. The entire system is divided into following modules. The following modules are inputs X and Y, partial products (A_H, A_L and B_H, B_L), shifter, accurate adder, Carry propagation and register. First the input numbers are divided into two parts each based on their significance level while product computation as lower significant and higher significant bits. Now shifter will shift the data words by sequential bits. Then the alignment of partial products generator will be done. Now these bits perform the addition operation using accurate adder and give the final product by subsequently performing the carry propagation. After that the partial products takes this registers and generates final output.



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

Fig. 1: BLOCKDIAGRAM OF PROPOSED ACCURATE MULTIPLIER

Here firstly, the operands are loaded in the multiplier in which say A and B are input nits they are divided in to the higher significant bits and lower significant bits as A_H, A_L and B_H, B_L. 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 utilized for data transfer in multiplicand register.

The barrel shifter cotains root and load mr and this are taken as input to this block. The multiplier register is generally attached to the finite field arithmetic circuit. In the same way, multiplicand register consists of shift, data_in and load_md bits which are taken as input to the barrel shifter. It will shift the data and as well as load the data in effective way. Result register consists of output and saves the entire arithmetic result. Compared with existed system, the proposed system provides effective results.

The result multiplier and multiplicand is saved in the result barrel shifter block. 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.

3.1 Partial-Products Generation

An alternative method to solve the problems of multi – digit multiplications is Partial – Product Multiplication. It is a strategy which is based on multiplication property of distributive (grouping). LSB of multiplier generates the first partial products; second bit in the multiplier generates the second partial product, etc. Final products are added by accurate adder circuit. Firstly computation is performed on 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

$$P_i = A_i \; XOR \; B_i$$
$$G_i = A_i \; AND \; B_i$$

Partial-Product Multiplication is an alternative method for solving multi-digit multiplication problems. This is a strategy that is based on the distributive (grouping) property of multiplication. The first partial product is created by the LSB of the multiplier, the second partial product is created by the second bit in the multiplier, etc. The final partial products are added with a accurate adder circuit.

3.2 Barrel Shifter

A digital circuit which will shift a data word with a specified number of bits utilizing only pure combination logic, without using any

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

sequential logic is a barrel shifter. A barrel shifter is a digital circuit that can shift a data word by a specified number of bits without the use of any sequential logic, only pure combinational logic. The barrel shifter consists of root and load mr and this are taken as input to this block. The multiplier register is generally attached to the finite field arithmetic circuit. In the same way, multiplicand register consists of shift, data_in and load_md bits which are taken as input to the barrel shifter. It will shift the data and as well as load the data in effective way. Result register consists of output and saves the entire arithmetic result. Compared to existed system, the proposed system gives effective results. 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

3.3 Carry Propagator and Generator

Parallel adders are 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 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.

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

$$G = G_i OR\ (P_i AND P_j)$$

$$P = P_i AND P_j$$

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

$$G = G_i OR\ (P_i AND P_j)$$

3.4 Accurate Adder

In this proposed architecture, accurate adder block utilizes half adders, full adders & approximate adders for providing improved results to the bits of higher order. For n- bits, this block is extended from MSB to LSB, to provide optimized PDP (power delay product) at the least error cost in the overall output. The application of architecture or system level decides the accurate adder. If the architecture/system level selection is either difficult or unclear then for such conditions a self configuration technique has been proposed. Through many consecutive bits carry is propagated due to large actual path delay. If the actual carry propagation chain is short, then approximation configuration is not required, which is intended for cutting carry chain shorter. Here first inputs are given to the propagator and generator unit. This unit will generate and propagated the input signals to adder logic. This logic will control the overall operation of the system. The adder will perform the addition operation in parallel form. Because of this the speed of operation is reduced.

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

$$S_i = P_i XOR\ C_{i-1}$$

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

IV. RESULTS

The Xilinx design environment was used to implement and examine the developed algorithm. The FPGA architecture of proposed accurate multiplier design is shown in Fig. 2 and Fig. 3. The below Fig. 2 and Fig. 3 show the RTL schematic and technology schematic of proposed Proposed high speed and high efficient accurate multiplier. RTL (Register-transfer logic) schematic is the combination of inputs and outputs. Technology schematic is the combination of Look up tables, Truth Tables, K-Map and equations.

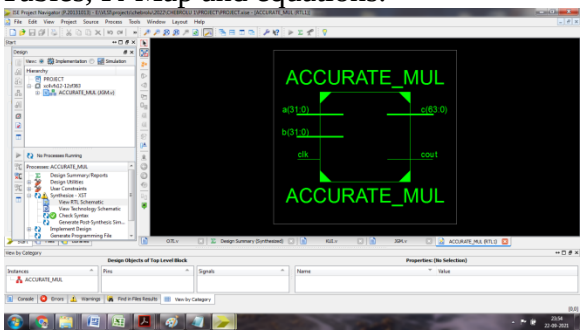


Fig. 2: RTL SCHEMATIC OF PROPOSED METHOD

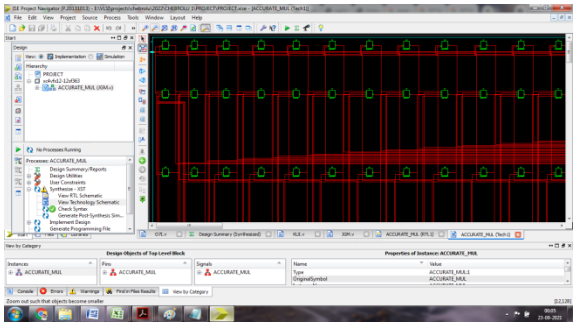


Fig. 3: TECHNOLOGY SCHEMATIC OF PROPOSED METHOD

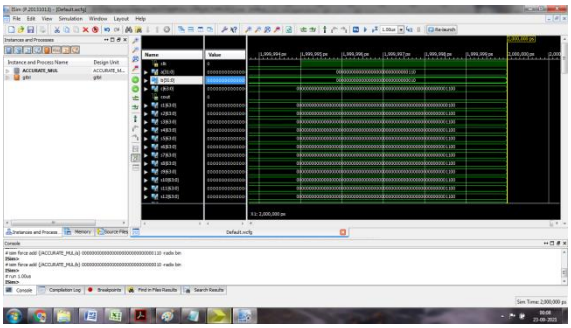


Fig. 4: OUTPUT WAVEFORMS OF PROPOSED METHOD

VI. CONCLUSION

Design of a high speed and high efficiency accurate multiplier for communication applications is presented in this document. Proposed technique divides the multiplicand & multiplier into two halves based on the significance as lower order and higher order. Each half is multiplied with the other & architecture is made accuracy-configurable. The proposed multiplier architecture is uses four-stage structures to get final output such as partial product generation, barrel shifter, carry propagator and generator and accurate adder. Shifter is used to shift the data words by sequential bits. Then resultant partial products are order in the alignment of partial products generator. The aligned partial products are then added using accurate adder and give the final product by subsequently performing the carry propagation. After that partial product takes the registers and generates final output. The proposed multipliers has a high speed of operation as well as area effective implementation because of employing divide and conquer approach in the design.

VI. REFERENCES

[1] En-Hui Zhang, Shih-Hsu Huang, "Low-Power Low-Error Fixed-Width Multiplier Design for Digital Signal Processing", 2021 IEEE International Conference on Consumer Electronics (ICCE), 2021

[2] N. Savithaa, A. Poornima, "A High speed Area Efficient Compression technique of Dadda multiplier for Image Blending Application", 2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2019

[3] Sunwoong Kim, Keewoo Lee, Wonhee Cho, Jung Hee Cheon, Rob A. Rutenbar, "FPGA-based Accelerators of Fully Pipelined Modular Multipliers for Homomorphic Encryption", 2019 International Conference on ReConFigurable Computing and FPGAs (ReConFig), 2019

[4] Tayab D Memon, Aneela Pathan, "An approach to LUT based multiplier for short word length DSP systems", 2018 International Conference on Signals and Systems (ICSigSys), 2018

[5] Praveen Yadav, Anirudha Pandey, Manikantta Reddy K., Ravi Prasad K.J., Vasantha M.H, Nithin Kumar Y.B., "Low Power Approximate Multipliers With Truncated Carry Propagation for LSBs", 2018 IEEE 61st International Midwest Symposium on Circuits and Systems (MWSCAS), 2018

[6] Anirudha Pandey, Manikantta Reddy Karri, Praveen Yadav, Nithin Kumar Y.B., Vasantha M.H., Design and Analysis of Approximate Multipliers for Error-Tolerant Applications", 2018 IEEE International Symposium on Smart Electronic Systems (iSES) (Formerly iNiS), 2018

[7] S. Venkatachalam, H. J. Lee, and S.-B. Ko, "Power efficient approximate booth multiplier," in Proc. IEEE Int. Symp. Circuits Syst. (ISCAS), Florence, Italy, May 2018, pp. 1–4

[8] 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.

[9] 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.

[10] Leandro Zafalon Pieper;Eduardo A. C. da Costa, José C. Monteiro, "Combination of radix-2m multiplier blocks and adder compressors for the design of efficient 2's complement 64-bit array multipliers", 26th Symposium on Integrated Circuits and Systems Design (SBCCI),2013.