ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Parallel multiplier-accumulator based on radix-2 modified Booth algorithm by using a VLSI architecture

Baile Shruthi
M Tech Student
JNIT, Hyderabad.

K Venkateswarlu, M Tech,
Associate Professor,
JNIT, Hyderabad.

ABSTRACT

In this paper, we proposed a new architecture of multiplier-and-accumulator (MAC) for high-speed arithmetic. By combining multiplication with accumulation and devising a hybrid type of carry save adder (CSA), the performance was improved. Since the accumulator that has the largest delayed in MAC was merged into CSA, the overall performance was elevated. The proposed CSA tree uses 1's-complement-based radix-2 modified Booth's algorithm (MBA) and has the modified array for the sign extension in order to increase the bit density of the operands. The CSA propagates the carries to the least significant bits of the partial products and generates the least significant bits in advance to decrease the number of the input bits of the final adder. Also, the proposed MAC accumulates the intermediate results in the type of sum and carry bits instead of the output of the final adder, which made it possible to optimize the pipeline scheme to improve the performance. The proposed architecture was synthesized with 250, 180 and 130 /xm, and 90 nm standard CMOS library. Based on the theoretical and experimental estimation, we analyzed the results such as the amount of hardware resources, delay, and pipelining scheme. We used Sakurai's alpha power law for the delay modelling. The proposed MAC showed the superior properties to the standard design in many ways and performance twice as much as the previous research in the similar clock frequency. We expect that the proposed MAC can be adapted to various fields requiring high performance such as the signal processing areas.

I.INTRODUCTION

With the recent rapid advances in multimedia and communication systems, real-time signal processing likeaudio signal processing, video/image processing, or large-capacity data processing are increasingly being demanded. The multiplier and multiplier-and-accumulator (MAC) [1] are the essential elements of the digital signal processing such as filtering, convolution, and inner products. Most digital signal processingmethods use nonlinear functions such as discrete cosine transform (DCT) [2] or discrete wavelet transforms (DWT) [3]. Because they are basically accomplished by repetitive application of multiplication and addition, the speed of the multiplication and addition arithmetic determines the execution speedand performance of the entire calculation. Because the multiplierrequires the longest delay among the basic operationalblocks in digital system, the critical path is determined by themultiplier, in general. For high-speed multiplication, the modifiedradix-4 Booth's algorithm (MBA) [4] is commonly used.

However, this cannot completely solve the problem due to thelong critical path for multiplication [5], [6].In general, a multiplier uses Booth's algorithm [7] and arrayof full adders (FAs), or Wallace tree [8] instead of the arrayof FAs., i.e., this multiplier mainly consists of the three parts:Booth encoder, a tree to compress the partial products such asWallace tree, and final adder [9], [10]. Because Wallace tree isto add the partial products from encoder as parallel as possible,its operation time is proportional to , where is thenumber of inputs. It uses the fact that counting the number of 1'samong the inputs reduces the number of outputs into. Inreal implementation, many (3:2) or (7:3) counters are used toreduce the number of outputs in each pipeline step. The mosteffective way to increase the speed of a multiplier is to reducethe number of the partial products because multiplication proceedsa

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

series of additions for the partial products. To reducethe number of calculation steps for the partial products, MBAalgorithm has been applied mostly whereWallace tree has takenthe role of increasing the speed to add the partial products. Toincrease the speed of the MBA algorithm, many parallel multiplicationarchitectures have been researched [11]–[13]. Amongthem, the architectures based on the Baugh–Wooley algorithm(BWA) have been developed and they have been applied to variousdigital filtering calculations [14]–[16].

One of the most advanced types of MAC for general-purposedigital signal processing has been proposed by Elguibaly [17].It is an architecture in which accumulation has been combinedwith the carry save adder (CSA) tree that compresses partialproducts. In the architecture proposed in [17], the critical pathwas reduced by eliminating the adder for accumulation and decreasingthe number of input bits in the final adder. While ithas a better performance because of the reduced critical pathcompared to the previous MAC architectures, there is a needto improve the output rate due to the use of the final adder results for accumulation. An architecture to merge the adder blockto the accumulator register in the MAC operator was proposedin [18] to provide the possibility of using two separate /2-bitadders instead of one -bit adder to accumulate the –bitMACresults. Recently, Zicari proposed an architecture that took amerging technique to fully utilize the 4–2 compressor [19]. It also took this compressor as the basic building blocks for the multiplication circuit.

In this paper, a new architecture for a high-speed MAC is proposed. In this MAC, the computations of multiplication and accumulation are combined and a hybrid-type CSA structure is proposed to reduce the critical path and improve the output rate. It uses MBA algorithm based on 1's complement number system. A modified array structure for the sign bits is used to increase the density of the operands. A carry look-ahead adder (CLA) is inserted in the CSA tree to reduce the number of bits in the final adder. In addition, in order to increase the output rate by optimizing the pipeline efficiency, intermediate calculation results are accumulated in the form of sum and carry instead of the final adder outputs.

This paper is organized as follows. In Section II, a simple introduction of a general MAC will be given, and the architecture for the proposed MAC will be described in Section III. In Section IV, the implementation result will be analyzed and the characteristic of the proposed MAC will be shown. Finally, theconclusion will be given in Section V.

II. OVERVIEW OF MAC

In this section, basic MAC operation is introduced. A multipliercan be divided into three operational steps. The first isradix-2 Booth encoding in which a partial product is generatedfrom the multiplicand (X) and the multiplier (Y). The secondis adder array or partial product compression to add all partialproducts and convert them into the form of sum and carry.The last is the final addition in which the final multiplicationresult is produced by adding the sum and the carry. If theprocess to accumulate the multiplied results is included, aMAC consists of four steps, as shown in Fig. 1, which showsthe operational steps explicitly.

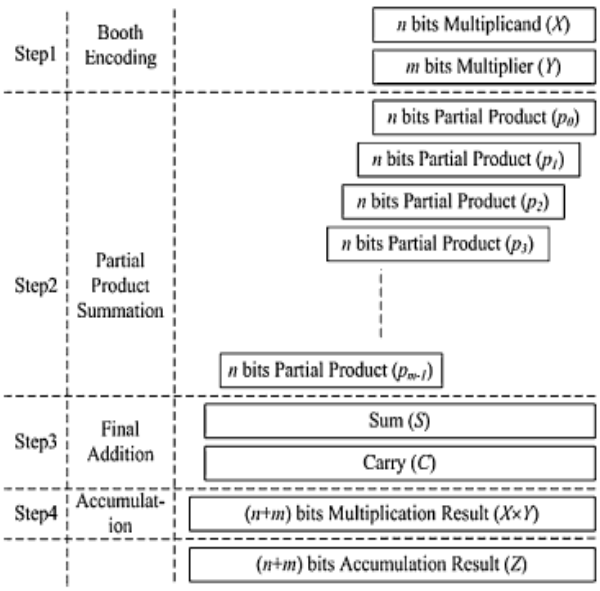

Fig.1. Basic arithmetic steps of multiplication and accumulation.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

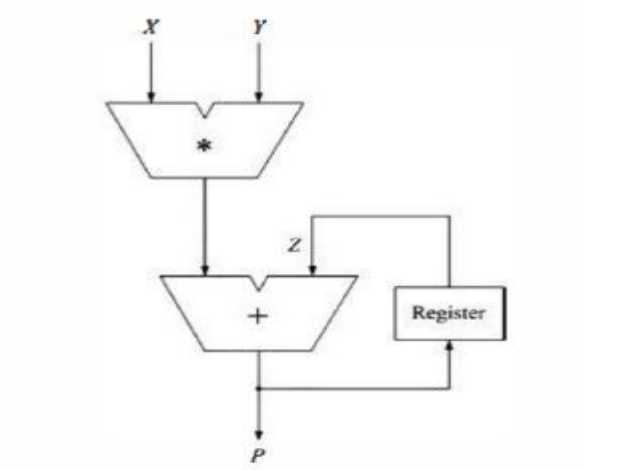

Fig.2.Hardware architecture of general MAC.

A general hardware architecture of this MAC is shown inFig. 2. It executes the multiplication operation by multiplyingthe input multiplier X and the multiplicand Y. This is added tothe previous multiplication result Z as the accumulation step.The N-bit 2's complement binary number can be expressed as

$$X = -2^{N-1}x_{N-1} + \sum_{i=0}^{N-2} x_i 2^i, \qquad x_i \in 0, 1. \tag{1}$$

If (1) is expressed in base-4 type redundant sign digit form in order to apply the radix-2 Booth's algorithm

$$X = \sum_{i=0}^{N/2-1} d_i 4_i \tag{2}$$

$$d_i = -2x_{2i+1} + x_{2i} + x_{2i-1}. \tag{3}$$

If (2) is used, multiplication can be expressed as

$$X \times Y = \sum_{i=0}^{N/2-1} d_i 2^{2i} Y. \tag{4}$$

If these equations are used, the afore-mentioned multiplication–accumulation results can be expressed as

$$P = X \times Y + Z = \sum_{i=0}^{N/2-1} d_i 2^i Y + \sum_{j=0}^{2N-1} z_i 2^i. \tag{5}$$

Each of the two terms on the right-hand side of (5) is calculated independently and the final result is produced by adding the two results. The MAC architecture implemented by (5) is called the standard design [6].

III. PROPOSED MAC ARCHITECTURE

In this section, the expression for the new arithmetic will be derived from equations of the standard design. From this result, VLSI architecture for the new MAC will be proposed. In addition,a hybrid-typed CSA architecture that can satisfy the operationof the proposed MAC will be proposed.

A.Derivation of MAC Arithmetic

I) Basic Concept:

If an operation to multiply two TV -bit numbersand accumulate into a 27V -bit number is considered, the criticalpath is determined by the 27V -bit accumulation operation. Theoverall performance of the proposed MAC is improved byeliminating the accumulator itself by combining it with the CSAfunction. If the accumulator has been eliminated, the critical pathis then determined by the final adder in the multiplier. The basicmethod to improve the performance of the final adder is todecrease the number of input bits. In order to reduce this numberof input bits, the multiple partial products are compressed into asum and a carry by CSA. The number of bits of sums and carriesto be transferred to the final adder is reduced by adding the lowerbits of sums and carries in advance within the range in which theoverall performance will not be degraded. A 2-bit CLA is used toadd the lower bits in the CSA. In addition, to increase the outputrate when pipelining is applied, the sums and carrys from theCSA are accumulated instead of the outputs from the final adderin the manner that the sum and carry from the CSA in theprevious cycle are inputted to CSA. Due to this feedback of bothsum and carry, the number of inputs to CSA increases, comparedto the standard design and [17]. In order to efficiently solve theincrease in the amount of data, a CSA architecture is modified totreat the sign bit.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

2) Equation Derivation:

The aforementioned concept is applied to(5) to express the proposed MAC arithmetic. Then, themultiplication would be transferred to a hardware architecturethat complies with the proposed concept in which the feedbackValue for accumulation will be modified and expanded for thenew MAC

If the multiplication in (4) is decomposed , it becomes

$$X \times Y = d_0 2Y + d_1 2^2 Y + d_2 2^4 Y + \ldots + d_{N/2-1} 2^{N-2} Y \quad (6)$$

If (6) is divided into the first partial product, sum of the middle partial products, and the final partial product, it can be re-expressed as (7).

$$X \times Y = d_0 2Y + \sum_{i=1}^{N/2-2} d_i 2^{2i} Y + d_{N/2-1} 2^{N-2} Y. \quad (7)$$

The reason for separating the partial product addition as (7) is that three types of data are fed back for accumulation, which are the sum, the carry, and the preadded results of the sum and carry from lower bitsNow, the proposed concept is applied to Z in (5). If Z is firstdivided into upper and lower bits and rearranged, (8) will bederived. The first term of the right-hand side in (8) corresponds tothe upper bits. It is the value that is fed back as the sum and thecarry. The second term corresponds to the lower bits.

$$Z = \sum_{i=0}^{N-1} z_i 2^i + \sum_{i=N}^{2N-1} z_i 2^i. \quad (8)$$

The second term can be separated further into the carry term and sum term as

$$\sum_{i=N}^{2N-1} z_i 2^i = \sum_{i=0}^{N-1} z_{N+i} 2^i 2^N = \sum_{i=0}^{N-2} (c_i + s_i) 2^i 2^N. \quad (9)$$

Thus, (8) is finally separated into three terms as

$$Z = \sum_{i=0}^{N-1} z_i 2^i + \sum_{i=0}^{N-2} c_i 2^i 2^N + \sum_{i=0}^{N-2} s_i 2^i 2^N. \quad (10)$$

If (7) and (10) are used, the MAC arithmetic in (5) can beexpressed as

$$P = \left(d_0 2Y + \sum_{i=1}^{N/2-2} d_i 2^{2i} Y + d_{N/2-1} 2^{N-2} Y \right) + \left(\sum_{i=0}^{N-1} z_i 2^i 2^N + \sum_{i=0}^{N-2} c_i 2^i 2^N + \sum_{i=0}^{N-2} s_i 2^i 2^N \right) \quad (11)$$

If each term of (11) is matched to the bit position and rearranged, it can be expressed as (12), which is the final equation for the proposed MAC. The first parenthesis on the right is the operation to accumulate the first partial product with the added result of the sum and the carry. The second parenthesis is theone to accumulate the middle partial products with the sum of the CSA that was fed back. Finally, the third parenthesis expressesthe operation to accumulate the last partial product with the carry of the CSA.

$$P = \left(d_0 2Y + \sum_{i=0}^{N-1} z_i 2^i \right) + \left(\sum_{i=1}^{N/2-1} d_i 2^{2i} Y + \sum_{i=0}^{N-2} c_i 2^i 2^N \right) + \left(d_{N/2-1} 2^{N-2} Y + \sum_{i=0}^{N-2} s_i 2^i 2^N \right) \quad (12)$$

B. Proposed MAC Architecture

If the MAC process proposed in the previous section is rearranged, it would be as Fig. 3, in which the MAC is organized into three steps. when compared with Fig. 1, it is easy to identify the difference that the accumulation has been merged into the process of adding the partial products. Another big difference from Fig. 1 is that the final addition process in step 3 is not always run even though it does not appear explicitly in Fig. 3. Since accumulation is carried out using the result from step 2 instead of that from step 3, step 3 does not have to be run until the point at which the result for the final accumulation is needed.

The hardware architecture of the MAC to satisfy the process in Fig. 3 is shown in Fig. 4. The n-bitMAC inputs, X and Y , are converted into an(n+1) bit partial product by passing through the Booth encoder. In the CSA S,C and Z accumulator, accumulation is carried out along with the addition of the partial products. As a result,P[2_n-1:n]bit , and (the result from adding the

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

lower bits of the sum and carry) are generated. These three values are fed back and used for the next accumulation. If the final result for the MAC is needed, is generated by adding S and C in the final adder and combined withP[n-1:0] that was already generated.

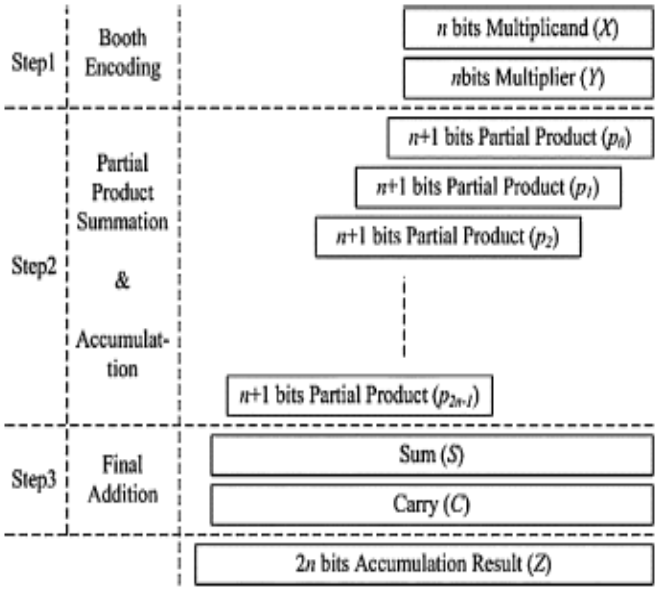

Fig.3. Proposed arithmetic operation of multiplication and accumulation

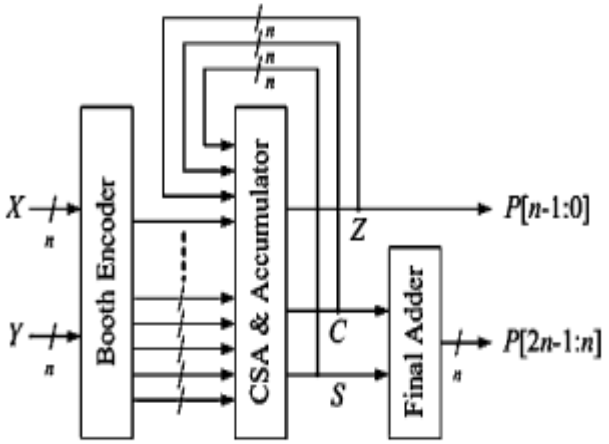

Fig.4.Hardware architecture of the proposed MAC.

C. Proposed CSA Architecture

The architecture of the hybrid-type CSA that complies with theoperation of the proposed MAC is shown in Fig. 5, whichperforms 8 x 8-bit operation. It was formed based on (12). In Fig.5, Si is to simplify the sign expansion and Ni is to compensatel's complement number into 2's complement number.S[i] and C [i] correspond to the ith bit of the feedback sum and carry.Z [i] is the ith bit of the sum of the lower bits for each partialproduct that were added in advance and Z' [i] is the previousresult.

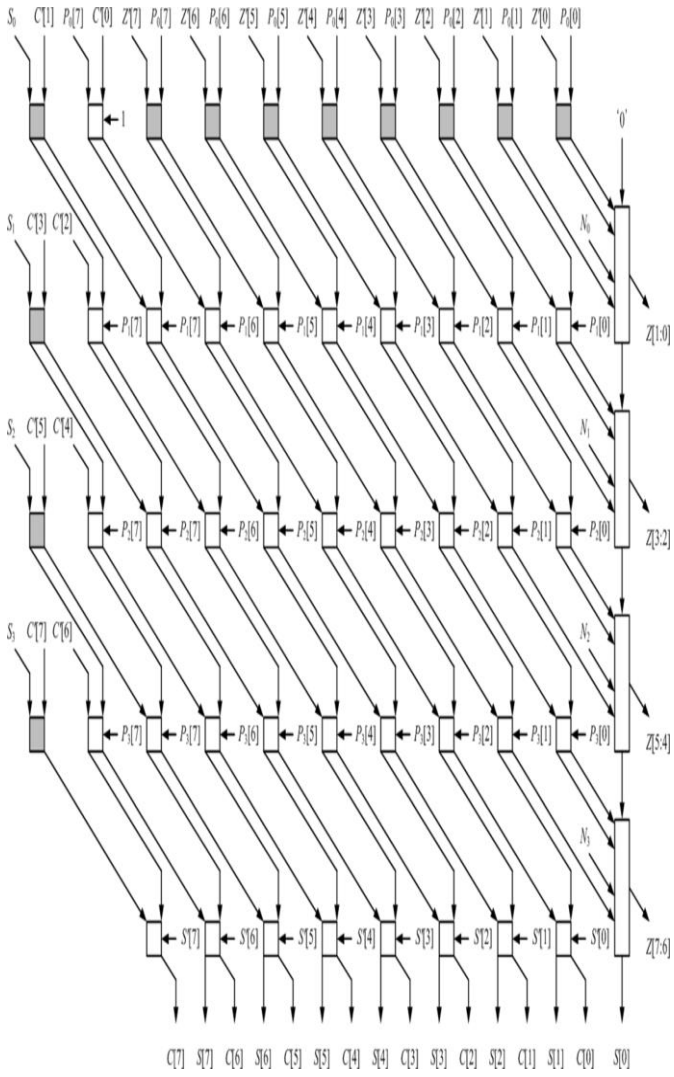

Fig.5.Architecture of the proposed CSA tree.

IV. SYNTHESIS AND SIMULATION RESULTS

We have coded the proposed MAC in Verilog HDL using the proposed design and the existing MAC designs of [6] and [7] for bit-widths 16. All the designs are synthesized in the Xilinx Synthesis XST Tool and Simulated using Xilinx ISE simulator.The synthesis and simulation results are shown in Fig.6 and Fig.7

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

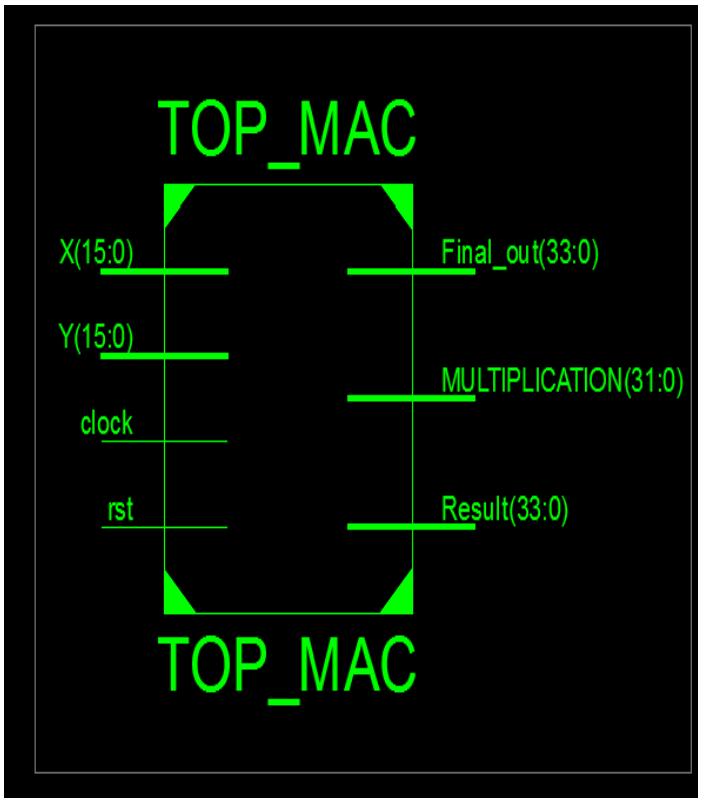

Figure 6: RTL schematic of 16x16 MAC unit.

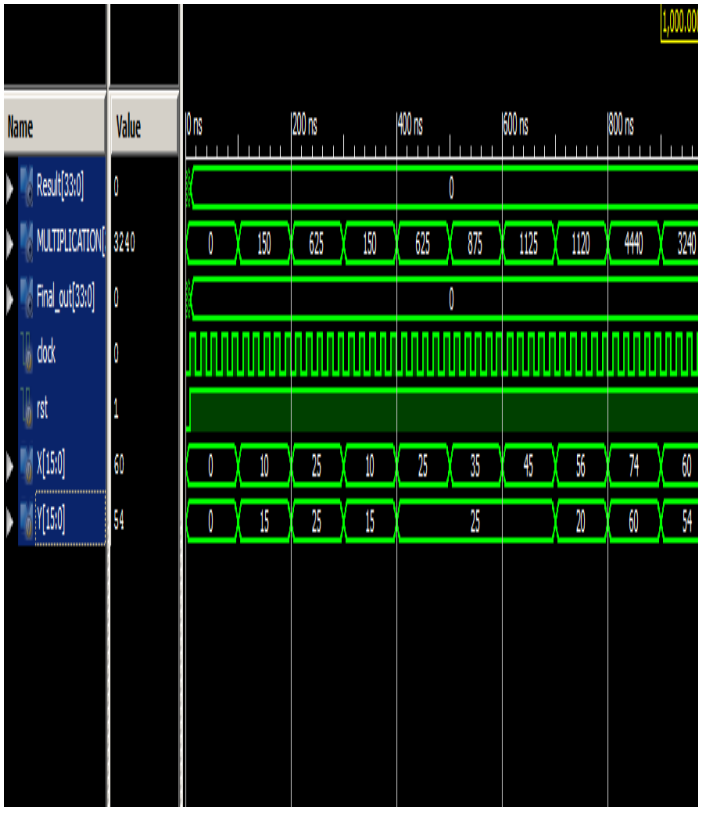

Figure 7: Simulated results of 16x16 MAC unit.

The design summary of the proposed 16x16 MAC unit.

TOP_MAC Project Status (06/23/2015 - 14:51:03)			
Project File:	mac_ise.xise	Parser Errors:	No Errors
Module Name:	TOP_MAC	Implementation State:	Synthesized
Target Device:	xc3s500e-4cp132	• Errors:	No Errors
Product Version:	ISE 14.4	• Warnings:	9 Warnings (0 new)
Design Goal:	Balanced	• Routing Results:	
Design Strategy:	Xilinx Default (unlocked)	• Timing Constraints:	
Environment:	System Settings	• Final Timing Score:	

Device Utilization Summary (estimated values)			[-]
Logic Utilization	Used	Available	Utilization
Number of Slices	694	4656	14%
Number of Slice Flip Flops	34	9312	0%
Number of 4 input LUTs	1278	9312	13%
Number of bonded IOBs	134	92	145%
Number of GCLKs	1	24	4%

Detailed Reports					[-]
Report Name	Status	Generated	Errors	Warnings	Infos
Synthesis Report	Current	Sat 27. Jun 16:43:31 2015	0	9 Warnings (0 new)	1 Info (0 new)
Translation Report					
Map Report					
Place and Route Report					

V. CONCLUSION

In this paper, a new MAC architecture to execute the multiplication-accumulation operation, which is the key operation, for digital signal processing and multimedia information processing efficiently, was proposed. By removing the independent accumulation process that has the largest delay and merging it to the compression process of the partial products, the overall MAC performance has been improved almost twice as much as in the previous work.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

REFERENCES

[I] J. J. F. Cavanagh, Digital Computer Arithmetic. New York: McGraw-Hili, 1984.

[2] Information Technology-Coding of Moving Picture and Associated Autio, MPEG-2 Draft International Standard, ISO/IEC 13818-1,2,3, 1994.

[3] JPEG 2000 Part I Finall19l Draft. ISO/IEC JTClISC29 WG 1.

[4] O. L. MacSorley, "High speed arithmetic in binary computers," Proc. IRE. vol. 49, pp. 67-91, Jan. 1961.

[5] S. Waser and M. 1. Flynn, Introduction to Arithmetic for Digital Systems Designers. New York: Holt, Rinehart and Winston, 1982.

[6] A. R. Omondi, Computer Arithmetic Systems. Englewood Cliffs, NJ:

[7] A. D. Booth, "A signed binary multiplication technique," Quart. J. Math., vol. IV, pp. 236-240,1952.

[8] C. S. Wallace, "A suggestion for a fast multiplier," IEEE Trans. Electron Comput., vol. EC-13, no. I, pp. 14-17, Feb. 1964.

[9] A. R. Cooper, "Parallel architecture modified Booth multiplier," Proc. Inst. Electr. Eng. G, vol. 135, pp. 125-128, 1988.

[10] N. R. Shanbag and P. Juneja, "Parallel implementation of a 4 X 4-bit multiplier using modified Booth's algorithm," IEEE J. Solid-State Circuits. vol. 23, no. 4, pp. 1010-1013, Aug. 1988.

[11] G. Goto, T. Sato, M. Nakajima, and T. Sukemura, "A 54 X 54 regular structured tree multiplier," IEEE J. Solid-State Circuits. vol. 27, no. 9, pp. 1229-1236, Sep. 1992.

[12] 1. Fadavi-Ardekani, "M N Booth encoded multiplier generator using optimized Wallace trees," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. l,no. 2,pp. 120-125,Jun. 1993.

[13] N. Ohkubo, M. Suzuki, T. Shinbo, T. Yamanaka, A. Shimizu, K. Sasaki, and Y. Nakagome, "A 4.4 ns CMOS 54 54 multiplier using pass-transistor multiplexer," IEEE J. Solid-State Circuits, vol. 30, no. 3, pp. 251-257, Mar. 1995.

[14] A. Tawfik, F. Elguibaly, and P. Agathoklis, "New realization and implementation of fixed-point IIR digital filters," J. Circuits, Syst., Comput.,vol. 7, no. 3, pp. 191-209, 1997.

[15] A. Tawfik, F. Elguibaly, M. N. Fahmi, E. Abdel-Raheem, and P. Agathoklis, "High-speed area-efficient inner-product processor," Can. J. Electr.Comput.Eng., vol. 19, pp. 187-191, 1994.

[16] F. Elguibaly and A. Rayhan, "Overf10w handling in inner-product processors," in Proc. IEEE Pacific Rim Con] Commun..Comput.. Signal Process .. Aug. 1997, pp. 117-120.

[17] F. Elguibaly, "A fast parallel multiplier-accumulator using the modified Booth algorithm," IEEE Trans. Circuits Syst., vol. 27, no. 9, pp. 902-908, Sep. 2000.

[18] A. Fayed and M. Bayoumi, "A merged multiplier-accumulator for high speed signal processing applications," Proc. ICASSP, vol. 3, pp. 3212-3215, 2002.

[19] P. Zicari, S. Perri, P. Corsonello, and G. Cocorullo, "An optimized adder accumulator for high speed MACs," Proc. ASICON 2005.vol.2, pp. 757-760,2005.

[20] T. Sakurai and A. R. Newton, "Alpha-power law MOSFET model and its applications to CMOS inverter delay and other formulas," IEEE J. Solid-State Circuits, vol. 25, no. 2, pp. 584-594, Feb. 1990

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

[21] Soojin Kim and Kyeongsoon Cho, Design of High-speed Modified Booth Multipliers Operating at GHz Ranges, World Academy of Science, Engineering and Technology; pp. 1-4,6.1. 2010.

[22] Qingzheng L1, Guixuan LIANG, Amine BERMAK, A High-speed 32-bit Signed/Unsigned Pipelined Multiplier, IEEE 5th International Symposium on Electronic Design, Test &Applications; pp. 207-211, 2010.

[23]Rajput, R.P,SwamY,M.N.S, "High Speed Modified Booth EncoderMultiplier for signed and unsigned numbers", IEEE page 649-654, 28-30 March 2012.