International Journal of Scientific Engineering and Technology Research

ISSN 2319-8885
Vol.03,Issue.48
December-2014,
Pages:9794-9797

www.ijsetr.com

Implementation of 64-Bit MAC Unit for High Speed Applications

S.KRUSHNA[1], P.HASEENABHANU[2]

[1]PG Scholar, Dept of VLSI Design, Anantha Lakshmi Institute of Technology and Sciences, Anantapur, AP, India, E-mail: krishnasandep@gmail.com.

[2]Associate Professor, Dept of ECE, Anantha Lakshmi Institute of Technology and Sciences, Anantapur, AP, India, E-mail: p.bhanu111@gmail.com.

Abstract: MAC unit is an inevitable component in many digital signal processing (DSP) applications involving multiplications and/or accumulations. MAC unit is used for high performance digital signal processing systems. The DSP applications include filtering, convolution, and inner products. Multiplication-and-accumulate operations are typical for digital filters. Therefore, the functionality of the MAC unit enables high-speed filtering and other processing typical for DSP applications. Since the MAC unit operates completely independent of the CPU, it can process data separately and thereby reduce CPU load. The application like optical communication systems which is based on DSP, require extremely fast processing of huge amount of digital data. A MAC unit consists of a multiplier and an accumulator containing the sum of the previous successive products. The MAC inputs are obtained from the memory location and given to the multiplier block. The design consists of 64 bit modified Wallace multiplier, 128 bit carry save adder and a register.

Keywords: MAC, Carry Save Adder, Modified Wallace.

I. INTRODUCTION

Binary adders are one of the most essential logic elements within a digital system. In addition, binary adders are also helpful in units other than Arithmetic Logic Units (ALU), such as multipliers, dividers and memory addressing. Therefore, binary addition is essential that any improvement in binary addition can result in a performance boost for any computing system and, hence, help improve the performance of the entire system. The major problem for binary addition is the carry chain. As the width of the input operand increases, the length of the carry chain increases. Fig.1 demonstrates an example of an 8- bit binary add operation and how the carry chain is affected. This example shows that the worst case occurs when the carry travels the longest possible path, from the least significant bit (LSB) to the most significant bit (MSB). In order to improve the performance of carry-propagate adders, it is possible to accelerate the carry chain, but not eliminate it. Consequently, most digital designers often resort to building faster adders when optimizing a computer architecture, because they tend to set the critical path for most computations. The binary adder is the critical element in most digital circuit designs including digital signal processors (DSP) and microprocessor data path units. As such, extensive research continues to be focused on improving the power delay performance of the adder.

In VLSI implementations, parallel-prefix adders are known to have the best performance. Reconfigurable logic such as Field Programmable Gate Arrays (FPGAs) has been gaining in popularity in recent years because it offers improved performance in terms of speed and power over DSP-based and microprocessor-based solutions for many practical designs involving mobile DSP and telecommunications applications and a significant reduction in development time and cost over Application Specific Integrated Circuit (ASIC) designs.

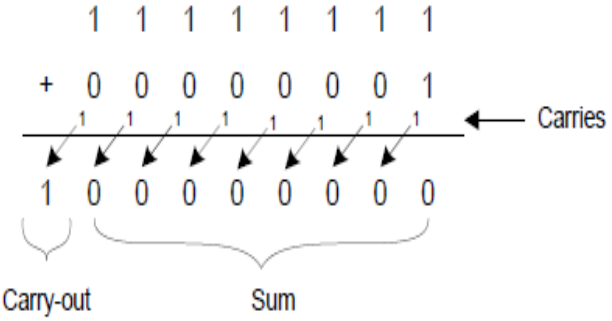

Fig.1. Binary Adder Example.

The power advantage is especially important with the growing popularity of mobile and portable electronics, which make extensive use of DSP functions. However, because of the structure of the configurable logic and routing resources in FPGAs, parallel-prefix adders will have a different performance than VLSI implementations. Adders form an almost obligatory component of every contemporary integrated circuit. This section presents some of the adder topology designs. Adders in VLSI digital systems use binary notation. In that case, add is done bit by bit using Boolean equations as shown in Fig.2.

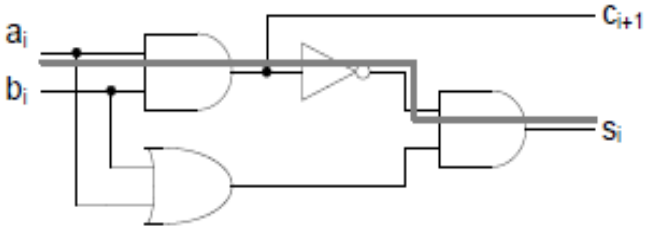

Fig.2. 1-bit Half Adder.

A. Ripple Carry Adder

Ripple carry adder is an n-bit adder built from full adders. Fig.3 shows a 4-bit ripple carry adder. One full adder is responsible for the addition of two binary digits at any stage of the ripple carry. The carryout of one stage is fed directly to the carry-in of the next stage. Even though this is a simple adder and can be used to add unrestricted bit length numbers, it is however not very efficient when large bit numbers are used.

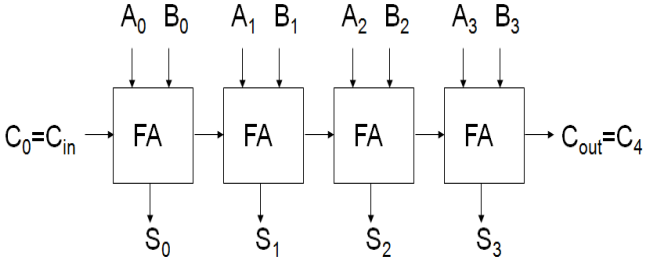

Fig.3. 4-b Ripple Carry Adder.

B. Carry Look-Ahead Adder

Lookahead carry algorithm speed up the operation to perform addition, because in this algorithm carry for the next stages is calculated in advance based on input signals. In CLA, the carry propagation time is reduced to O(log2(Wd)) by using a tree like circuit to compute the carry rapidly. Fig.4 shows the 4-bit Carry Look-Ahead Adder.

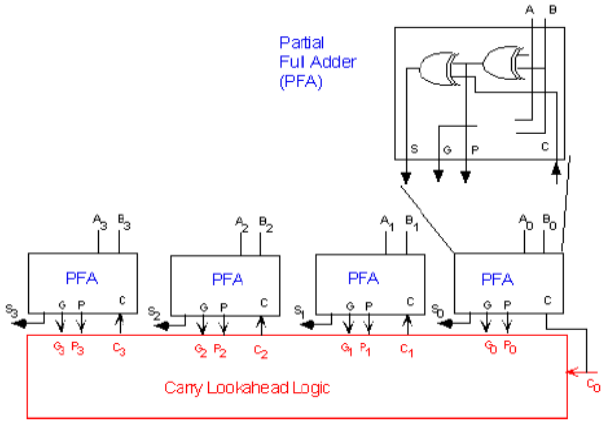

Fig.4. 4-bit Carry Look Ahead Adder.

The CLA exploits the fact that the carry generated by a bit-position depends on the three inputs to that position. If 'X' and 'Y'are two inputs then if X=Y=1, a carry is generated independently of the carry from the previous bit position and if X=Y= 0, no carry is generated. Similarly if X ≠ Y, a carry is generated if and only if the previous bit-position generates a carry. 'C' is initial carry, "S" and "Cout" are output sum and carry respectively, then Boolean expression for calculating next carry and addition is:

II. MAC OPERATION

The Multiplier-Accumulator (MAC) operation is the key operation not only in DSP applications but also in multimedia information processing and various other applications. As mentioned above, MAC unit consist of multiplier, adder and register/accumulator. In this paper, we used 64 bit modified Wallace multiplier as shown in Fig.5. The MAC inputs are obtained from the memory location and given to the multiplier block. This will be useful in 64 bit digital signal processor. The input which is being fed from the memory location is 64 bit. When the input is given to the multiplier it starts computing value for the given 64 bit input and hence the output will be 128 bits. The multiplier output is given as the input to carry save adder which performs addition. The function of the MAC unit is given by the following equation:

$$F = \sum P_i Q_i \quad (1)$$

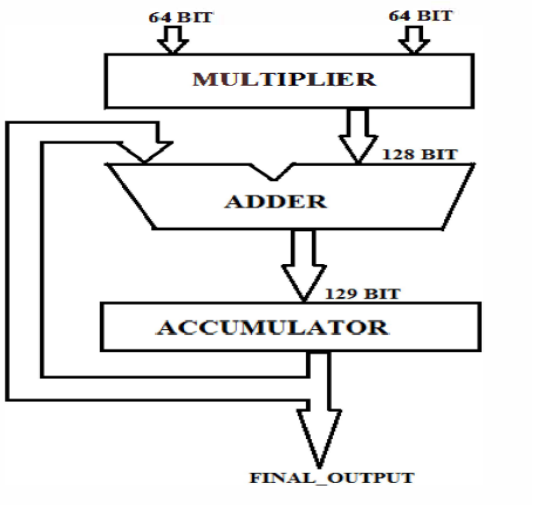

Fig.5. Basic Architecture of MAC unit.

The output of carry save adder is 129 bit i.e. one bit is for the carry (128bits+ 1 bit). Then, the output is given to the accumulator register. The accumulator register used in this design is Parallel In Parallel Out (PIPO). Since the bits are huge and also carry save adder produces all the output values in parallel, PIPO register is used where the input bits are taken in parallel and output is taken in parallel. The output of the accumulator register is taken out or fed back as one of the input to the carry save adder. The fig.5 shows the basic architecture of MAC unit.

III. MODIFIED WALLACE IMPLEMENTATION

A modified Wall ace multiplier is an efficient hardware implementation of digital circuit multiplying two integers. Generally in conventional Wallace multipliers many full adders and half adders are used in their reduction phase.

Half adders do not reduce the number of partial product bits. Therefore, minimizing the number of half adders used in a multiplier reduction will reduce the complexity. Hence, a modification to the Wallace reduction is done in which the delay is the same as for the conventional Wallace reduction. The modified reduction method greatly reduces the number of half adders with a very slight increase in the number of full adders. Reduced complexity Wall ace multiplier reduction consists of three stages. First stage the N x N product matrix is formed and before the passing on to the second phase the product matrix is rearranged to take the shape of inverted pyramid. During the second phase the rearranged product matrix is grouped into non-overlapping group of three as shown in the fig.6, single bit and two bits in the group will be passed on to the next stage and three bits are given to a full adder. The number of rows in the in each stage of the reduction phase is calculated by the formula

$$r_{j+1} = 2[r_i/3] + r_j \bmod 3 \quad (2)$$

$$\text{If } r_j \bmod 3 = 0, \text{ then } r_{j+1} = 2r/3 \quad (3)$$

If the value calculated from the above equation for number of rows in each stage in the second phase and the number of row that are fonned in each stage of the second phase does not match, only then the half adder will be used. The final product of the second stage will be in the height of two bits and passed onto the third stage. During the third stage the output of the second stage is given to the carry propagation adder to generate the final output.

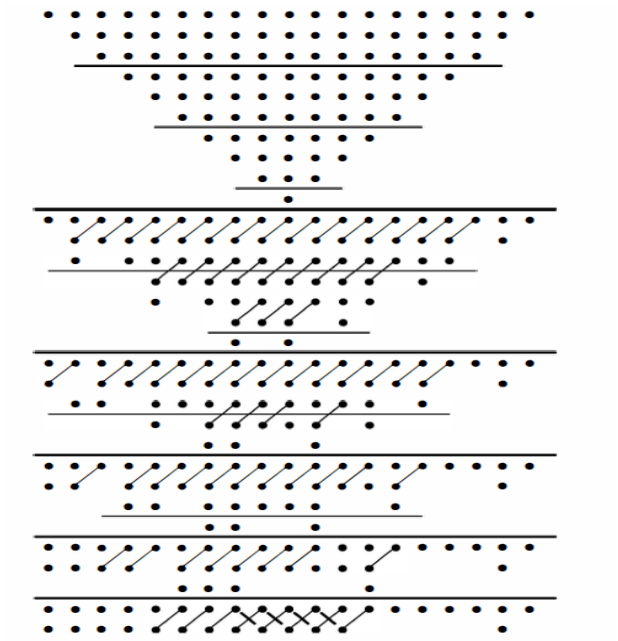

Fig.6. Modified Wall ace 10-bit by 10-bit reduction.

Thus 64 bit modified Wallace multiplier is constructed and the total number of stages in the second phase is 10. As per the equation the number of row in each of the 10 stages was calculated and the use of half adders was restricted only to the 10th stage. The total number of half adders used in the second phase is 8 and the total number of full adders that was used during the second phase is slightly increased that in the conventional Wallace multiplier. Since the 64 bit modified Wallace multiplier is difficult to represent, a typical lO-bit by 10-bit reduction shown in fig.6 for understanding. The modified Wallace tree shows better performance when carry save adder is used in final stage instead of ripple carry adder. The carry save adder which is used is considered to be the critical part in the multiplier because it is responsible for the largest amount of computation.

IV. RESULTS

Results of this paper is shown in bellow Figs.7 to 9.

A. Schematic

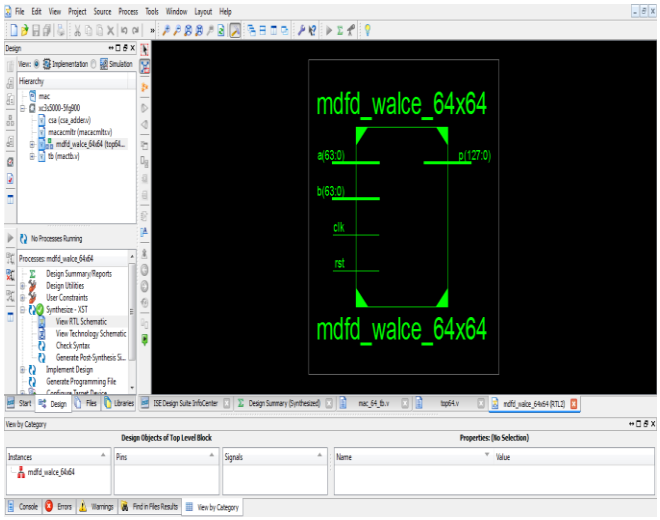

Fig.7. Schematic.

B. RTL Schematic

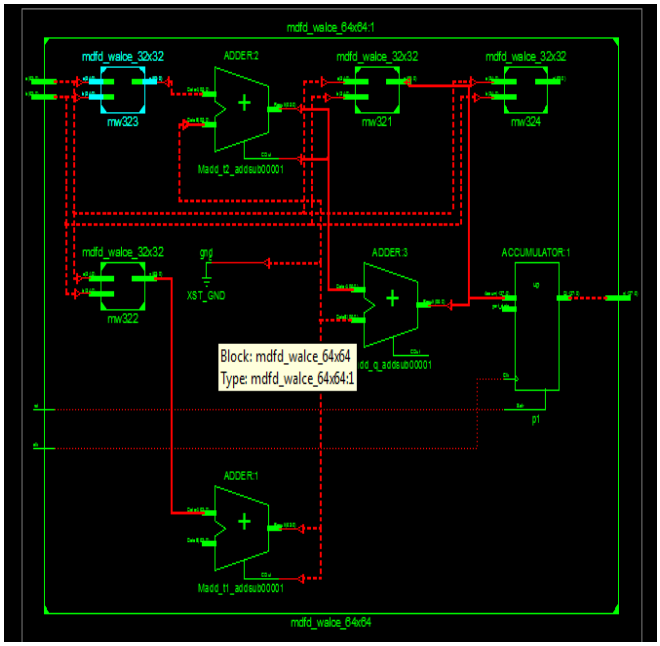

Fig.8. RTL Schematic.

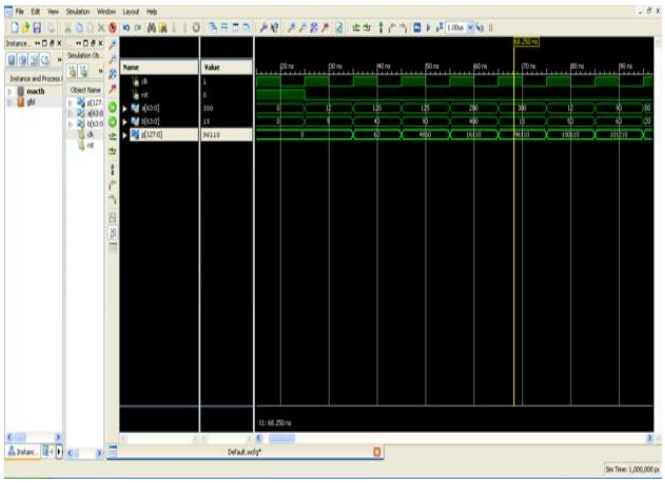

Fig.9. Waveform.

V. CONCLUSION

The Design of high performance 64 bit Multiplier-and-Accumulator (MAC) was implemented in this project. The total MAC unit operates at a frequency of 217 MHz's The total power dissipated by 64 bit MAC unit is 177.732 mW. Since the delay of 64 bit MAC is less, this design can be used in the system which requires high performance in processors involving large number of bits of the operation. The functionality of the MAC is verified using XILINX ISE and synthesized using XILINX synthesizer.

VI. REFERENCES

[1].Young-Ho Seo and Dong-Wook Kim, "New VLSI Architecture of Parallel Multiplier-Accumulator Based on Radix-2 Modified Booth Algorithm," IEEE Transactions on very large scale integration (vlsi) systems, vol. 18, no. 2,february 20 10.

[2] Ron S. Waters and Earl E. Swartzlander, Jr., "A Reduced Complexity Wall ace Multiplier Reduction, " IEEE Transactions On Computers, vol. 59, no. 8, Aug 20 10.

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

[4]. Shanthala S, Cyril Prasanna Raj, Dr.S.Y.Kulkarni, "Design and VLST Implementation of Pipelined Multiply AccumulateUnit," IEEE International Conference on Emerging Trends in Engineering and Technology, ICETET-09.

[5]. B.Ramkumar, Harish M Kittur and P.Mahesh Kannan, "ASIC Implementation of Modified Faster Carry Save Adder ", European Journal of Scientific Research, Vol. 42, Issue 1, 2010.

[6]. R.UMA, Vidya Vijayan, M. Mohanapriya and Sharon Paul,"Area, Delay and Power Comparison of Adder Topologies",International Journal of VLSI design & Communication Systems (VLSICSj Vo1.3, No.1, February 2012.

[7]V.G.Oklobdzija, "High-Speed VLSI Arithmetic Units: Adders and Multipliers", in "Design of High-Performance Microprocessor Circuits", Book edited by A.Chandrakasan, IEEE Press,2000.

[8]. Dadda, "Some Schemes for Parallel Multipliers," Alta Frequenza, vol. 34, pp. 349-356, 1965.

[9]. C.S. Wall ace "A Suggestion for a fast multipliers," IEEE Trans. Electronic Computers, vol. 13, no.l,pp 14-17, Feb. 1967.

[10]. L.Dadda, "On Parallel Digital Multiplier", Alta Frequenza,vol. 45, pp. 574-580, 1976.