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

# Implementation of MAC UNIT Using Efficient Adders

**U.Anusha**
**Department of ECE,**
**Chalapathi Institute of Engineering and Technology, Guntur.**

**M.Sirisha**
**Assistant Professor,**
**Department of ECE,**
**Chalapathi Institute of Engineering and Technology, Guntur.**

**M.Kalpana Chowdary**
**Assistant Professor,**
**Deportment of ECE,**
**Chalapathi Institute of Engineering and Technology, Guntur.**

**Abstract:**
A design of high performance 64 bit Multiplier-and-Accumulator (MAC) is implemented in this paper. MAC unit performs important operation in many of the digi-tal signal processing (DSP) applications. The multiplier is designed using modified Wallace multiplier and the adder is done with carry save adder. The total design is coded with Synthesize and simulate by verilog-HDL.

**Keywords:**
Wallace multipliers, Carry save adder, multi-plier and accumulator (MAC).

## I. INTRODUCTION:

MAC unit is an inevitable component in many digital signal processing (DSP) applications involving multi-placations and/or accumulations. MAC unit is used for high performance digital signal processing systems. The DSP applications include filtering, convolution, and inner products. Most of digital signal processing methods use nonlinear functions such as discrete co-sine transform (DCT) or discrete wavelet transforms (DWT).

Because they are basically accomplished by repetitive application of multiplication and addition, the speed of the multiplication and addition arithmetic determines the execution speed and performance of the entire cal-culation [1]. Multiplication-and-accumulate operations are typical for digital filters. Therefore, the function-ality 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 pro-cessing of huge amount of digital data. The Fast Fou-rier Transform (FFT) also requires addition and multi-plication. 64 bit can handle larger bits and have more memory.A MAC unit consists of a multiplier and an ac-cumulator containing the sum of the previous succes-sive 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. This paper is divided into six sections. In the first sec-tion the introduction about MAC unit is discussed. In the second section discuss about the detailed opera-tion of MAC unit. The third and fourth section deals with the operation of modified Wallace multiplier and carry save adder respectively. In the fifth section, the obtained result for the 64 bit MAC unit is discussed and finally the conclusion is made in the sixth section.

## II. MAC OPERA TION:

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. 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.

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

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 [4]:

F= IPjQj (1)

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 pro-duces all the output values in parallel, PIPO register is used where the input bits are taken in parallel and out-put 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 figure 1 shows the basic archi-tecture of MAC unit.

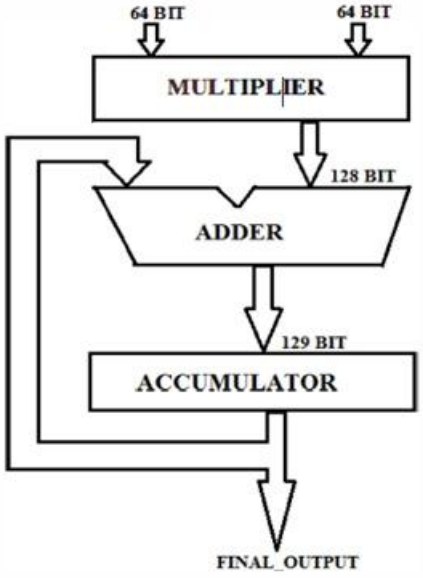



**Figure:-1 Basic architecture of MAC unit**

## III. WALL ACE MULTIPLIER:

AWall ace multiplier is an efficient hardware implementation of digital circuit multiplying two in-tegers. Generally in conventional Wallace multipliers many full adders and half adders are used in their reduc-tion phase. Half adders do not reduce the number of partial product bits. Therefore, minirnizing the number of half adders used in a multiplier reduction will reduce the complexity [2]. Hence, a modification to the Wal-lace Reduced complexity Wall ace multiplier reduction con-sists of three stages [2]. First stage the N x N product matrix is formed and before the passing on to the sec-ond 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 figure 2, 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 reduc-tion phase is calculated by the formula

rj+ 1= 2[ri/3]+rjmod3
(2) If rj mod3 = 0, then rj+ 1 = 2r/3
(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 formed 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 on to the third stage. During the third stage the output of the second stage is given to the carry propagation adder to gener-ate the final output. 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 [2].

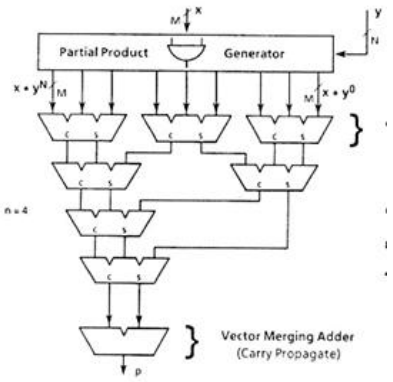



**Figure:-2 Wallace Tree Mutiplier**

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

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 conven-tional Wallace multiplier.

## IV. KOGGE STONE ADDER:

The Kogge–Stone adder is a parallel prefix form carry look-ahead adder. Other parallel prefix adders include the Brent-Kung adder, the Han Carlson adder, and the fastest known variation, the Lynch-Swartzlander Spanning Tree adder. The Kogge–Stone adder takes more area to implement than the Brent–Kung adder, but has a lower fan-out at each stage, which increases performance for typical CMOS process nodes. However, wiring congestion is often a problem for KoggeStone adders. The Lynch-Swartzlander design is smaller, has lower fan-out, and does not suffer from wiring congestion; however to be used the process node must support Manchester Carry Chain implementations. The general problem of optimizing parallel prefix adders is identical to the variable block size, multi level, carry-skip adder optimization problem, a solution of which is found in.

An example of a 4-bit KoggeStone adder is shown to the right. Each vertical stage produces a "propagate" and a "generate" bit, as shown. The culminating generate bits (the carries) are produced in the last stage (vertically), and these bits are XOR'd with the initial propagate after the input (the red boxes) to produce the sum bits. E.g., the first (least-significant) sum bit is calculated by XORing the propagate in the farthest-right red box (a "1") with the carry-in (a "0"), producing a "1". The second bit is calculated by XORing the propagate in second box from the right (a "0") with C0 (a "0"), producing a "0". During the addition of two numbers using a half adder, two ripple carry adder is used. This is due the fact that

Si = xi 1\ yi
(4)
Ci = xi & yi
(5)

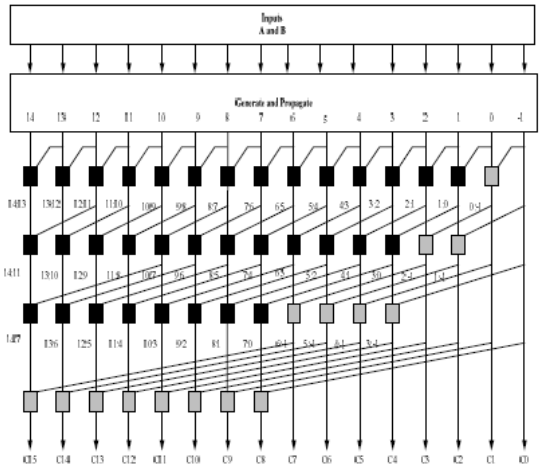


**Figure:- 3 Kogge Stone Adder**

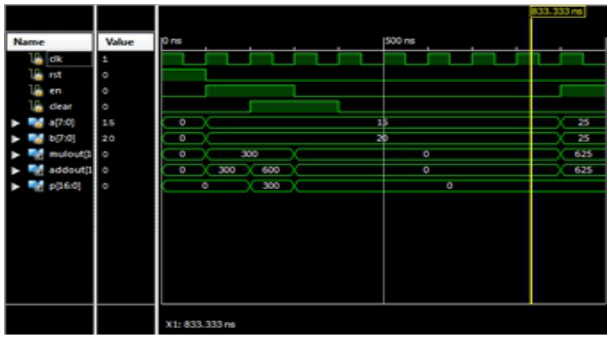
ripple carry adder cannot compute a sum bit without waiting for the previous carry bit to be produced, and hence the delay will be equal to that of n full adders. However a carry-save adder produces all the output values in parallel, resulting in the total computation time less than ripple carry adders. So, Parallel In Par-allel Out (PIPO) is used as an accumulator in the final stage.

## V. RESULT:

The design is developed using Verilog-HDL and synthe-sized in Encounter RTL compiler using typical libraries of TSMC 180nm technology. As a previous work, 8 bit MAC unit is designed using different multipliers and ad-ders. The multipliers used for comparative study are: (i) Modified Booth Aigorithm (ii) Dadda Multiplier (iii) Wallace multiplier. The different adders used in the study are: (i) Carry Look Ahead (ii) Carry Select Adder (iii) Carry Save adder.

### Simulation Results:

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




## Timing Report:

```
Timing Summary:
---------------
Speed Grade: -3

   Minimum period: 2.774ns (Maximum Frequency: 360.477MHz)
   Minimum input arrival time before clock: 17.611ns
   Maximum output required time after clock: 3.634ns
   Maximum combinational path delay: No path found
```

## Area Report:

```
Device utilization summary:
---------------------------

Selected Device : 6slx16csg324-3


Slice Logic Utilization:
 Number of Slice Registers:              17  out of  18224     0%
 Number of Slice LUTs:                  149  out of   9112     1%
    Number used as Logic:               149  out of   9112     1%

Slice Logic Distribution:
 Number of LUT Flip Flop pairs used:    150
   Number with an unused Flip Flop:     133  out of    150    88%
   Number with an unused LUT:             1  out of    150     0%
   Number of fully used LUT-FF pairs:    16  out of    150    10%
   Number of unique control sets:         1

IO Utilization:
 Number of IOs:                         37
 Number of bonded IOBs:                 37  out of    232    15%

Specific Feature Utilization:
 Number of BUFG/BUFGCTRLs:               1  out of     16     6%
```

## RTL Schematic:

The following figures are the generated RTL Schematics for Kogge stone adder

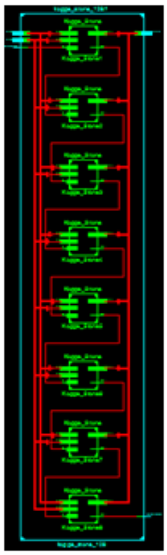


**Figure:-4 RTL Schematic**

## IV.CONLCUSION:

Since the delay of 64 bit is less, this design can be used in the sys-tem which requires high performance in processors in-volving large number of bits of the operation. The MAC unit is designed using Verilog-HDL and synthesized in Behavioral RTL Complier.Hence , If we keep this type of applications and designs in any hardware the total behavior and performance of the architecture will become efficient and reliable.

## REFERENCES:

[1].Young-Ho Seo and Dong-Wook Kim, "New VLSI Ar-chitecture 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 Re-duced 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.

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

[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 Fre-quenza, vol. 45, pp. 574-580, 1976.

[12].Fabrizio Lamberti and Nikos Andrikos, " Reducing the Computation Time in (Short Bit-Width) Two's Com-plement Multipliers", IEEE transactions on computers, Vol. 60, NO. 2, FEBRUARY 20 1 1.

[11].WJ. Townsend, E.E. Swartzlander Jr., and J.A. Abraham, "A Comparison of Dadda and Wall ace Mul-tiplier Delays," Proc. SPIE, Advanced Signal Processing Algorithms, Architectures, and Implementations XIII, pp. 552-560, 2003.