RESEARCH ARTICLE OPEN ACCESS

Review on Multiply-Accumulate Unit

Roshani Pawar*, Dr. S. S. Shriramwar**
** M.Tech. Student, Department of Electronics Engineering, Priyadarshini College of Engineering, Nagpur*
*** Head of Department of Electronics Engineering, Priyadarshini College of Engineering, Nagpur*

ABSTRACT
In present day MAC unit is demanded in most of the Digital signal processing. Function of addition and multiplication is performed by the MAC unit. MAC operates in two stages. Firstly, multiplier computes the given number output and the result is forwarded to second stage i.e. addition/accumulation operates. Speed of multiplier is important in MAC unit which determines critical path as well as area is also of great importance in designing of MAC unit. Multiplier plays an important roles in many digital signal processing (DSP) applications such as in convolution, digital filters and other data processing unit. Many research has been performed on MAC implementation. This paper provides analysis of the research and investigations held till now.

Keywords: Carry save adder , MAC, Urdhav Triyagbhyam, Vedic Mathematics, VLSI

I. INTRODUCTION

Multiplier is an important basic building block in designing of systems using digital signal processing and in other applications such as microprocessors, microcontroller and other data processing unit. Many researchers are continuously trying to design multiplier with low power consumption, high speed, regular structure, reduced delay for compact VLSI implementation. Multiplication dominates the execution time of the most DSP and hence, it’s the overall operation of the systems. Multiplication of binary numbers is usually implemented by using repeated addition and shift operations. Since the binary adders are designed to add only two binary numbers at a time, instead of adding all the partial products at the end, results in increased delay. Thus the systems speed gets reduced and consume more power.

Many algorithms are designed to perform the multiplication process. Every algorithms has its own advantages and disadvantages in terms of their area, delay, speed, circuit complexity and power consumption. Multiplier is the essential element in the MAC unit. The MAC architecture consists of multiplier, adder and an accumulator. In order to improve speed and reduction in delay of the MAC, there are two bottlenecks. The first is to reduce the partial products and the second is the accumulators. Since multipliers in MAC are rather complex circuits and must typically operate at a high system clock rate, reducing the delay of a multiplier is an essential part of satisfying the overall design.

The main goal of the MAC unit is to increase the speed which in turn reduced the delay and consumes less power. In order to increase the speed of the adders and multiplier, the number of the partial products generated must be reduced. During the operations, the partial products determine the number of clock cycles which generates the systems delay. In digital signal processing, speed and throughput are the two important parameters and hence designing of MAC with enhance speed is need of now. The MAC unit is briefly described and its operation is introduced.

II. LITERATURE SURVEY

Meenu S Ravi1, R H Khade and Ajit Saraf [1] in this paper, a floating point multiply and accumulate unit is designed using ancient mathematics that reduces the number of partial products to be added as well as increases the speed of accumulation of partial products by reducing the number of stages of partial products that needs to be added thereby making it a high performance unit. The output of this unit is simulated using simulation software ModelSim SE plus 6.2C and the language used is VHDL. G.Indira, G. Madhusudhana Rao, P.Jaya Babu, M. Ravi Kiran [4] in this paper, Digital signal processing is the application of mathematical operations to digitally represented signals. MAC is the most important block in DSP system. High throughput multiplier accumulator (MAC) is always a key element to achieve a high-performance digital signal processing application for real time signal processing applications. This is because speed and throughput rate are always the concerns of digital signal processing systems.

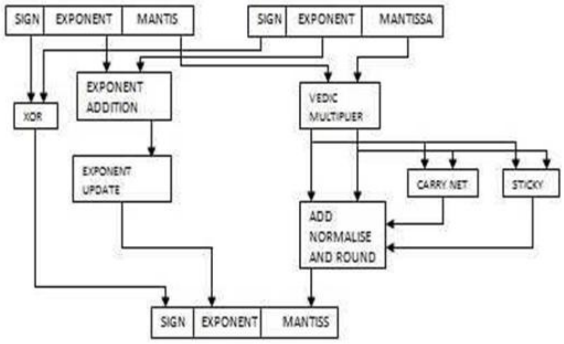


Fig. 1 Block diagram representation of floating point multiplier

The key blocks of the MAC unit are multipliers and adders, in which multiplier is the one which occupies the major silicon area and consumes more power. In general, the multiplication operations are performed by the shift and add logic. Most of the DSP applications demand faster adders for its arithmetic computations. Carry Select Adder (CSLA) is a well-known adder for its faster computation time. The goal of this project is to design the VLSI implementation of MAC for high-speed DSP applications. In this project, the design of 8X8 bits MAC unit has been implemented using Wallace tree multiplier to reduce the area. This MAC unit has 16 bit output and its operation is to add repeatedly the multiplication results. All the basic blocks of MAC unit are identified and analyzed through its performance. The improvements achieved in low power consumption of the MAC unit can be used in high speed DSP applications.

Ku. Shweta N. Yengade, Associate Prof. P. R. Indurkar [2] in this paper, shows different approaches of designing the MAC unit. Performance is analyzed based on the parameters such as area, delay, power. For 8 bit MAC, Baugh-Wooley Multiplier has increased delay and power is very low as compared to other techniques. For 32 bit MAC using Baugh-Wooley with HPM reduction exhibit comparable delay, less power dissipation and smaller area than Modified Booth multiplier. For 16 bit MAC, Proposed 2C-MAC has less power and less delay. By using Baugh-Wooley Multiplier the delay has been reduced .And if pipelining technique used, the power consumption also very less as compared to other multiplier techniques.

Shaik Nasar, K. Subbarao [3] in this paper, designs a novel 4x4 bit reversible multiplier circuit using Peres gates and HNG gates. Table.no.1 demonstrates that the proposed reversible multiplier circuit is better than the existing designs in terms of hardware complexity, number of gates, garbage outputs and constant inputs.

Multiplier Design	No of Logic Gates	No of Garbage outputs	No of Constant Inputs	Total Logical Calculations
Peres gates and HNG gates	28	52	28	80a+36ß
Nanotechnology	28	56	32	92a+52ß+36d
TSG gates	29	58	34	110a+103ß+71d
BME and MHNG gates	40	56	31	80a+100ß+68d

Table No.1 Comparative results of various reversible multipliers

Saleh, Hani, and Earl E. Swartzlander [5] in this paper, a floating-point fused add-subtract unit describes the performance of continuous floating-point add and subtract operations on a common pair of single-precision data in about the same time that it takes to perform a single addition with a conventional floating-point adder. The fused add-subtract unit is about 40% larger than a conventional floating-point adder.

Samy, Rodina, Hossam AH Fahmy, RamyRaafat, Amira Mohamed, Tarek ElDeeb, and Yasmin Farouk [6] in this paper, implementation of a fully parallel decimal floating-point fused multiply-add unit. It is performing the operation ±(A × B) ± C on decimal floating-point operands. The design is fully compliant with the IEEE 754-2008 standard and supports the two standard formats decimal 64 and decimal128. The decimal floating-point FMA may be pipelined so that a complete resultant decimal floating-point is available each clock cycle. [14]

Michael F. Cowlishaw [7] in this paper, decimal arithmetic is used in human calculations, and human-centric applications must use a decimal floating-point arithmetic to achieve the same results. In this paper, it introduces a new approach to decimal floating-point which not only provides the strict results which are necessary for commercial applications but also meets the constraints and requirements of the IEEE 854 standard. A hardware implementation of this arithmetic is in development, and it is expected that this will significantly accelerate a wide variety of applications.

A. D. Robison [8] in this paper, integer division on modern processors is expensive compared to multiplication. In this paper it presents, and proves, a hybrid of previous algorithms that replaces n + 1 bit multiplication with a single fused multiply-add operation on n-bit operands, thus reducing any n-bit unsigned division to the upper n bits of a multiply-add, followed by a single right shift. The advantage is that the prerequisite calculations are simple and fast. [16]

Singh, Harpreet, and Chakshu Goel [9] in this paper, reversible logic has shown considerable acceptance and growth in the research fields like quantum computing, Nano computing and optical computing promising lower power dissipation. This paper proposes an optimized design single-bit reversible comparator called SKAR gate with a purpose of reducing quantum cost. This paper describes two designs, one with the use of SKAR gate and other one using a derivative gate constructed from SKAR gate. Both the proposed designs for single-bit and four-bit reversible comparator are compared with other existing designs on the basis of elementary parameters of reversible logic.

DIFFERENT MULTIPLIER:

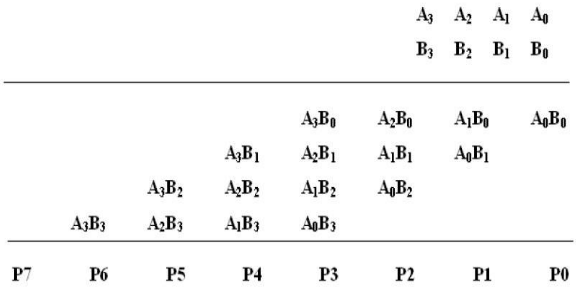
The fundamental operation of MAC unit is multiplications. The major issues in the multiplier unit are the power consumption, dissipation, area, speed and latency. Thus, to avoid these issues, we opt for the fast multiplier in various applications of DSP.

ARRAY MULTIPLIER:

Structure of array multiplier is regular as compared to conventional multiplier. Algorithm of multiplier circuit is based on add and shift operation. Multiplication of the multiplicand with one multiplier bit generates each partial product. According to their bit orders, the partial product are shifted and then added. The addition is performed with normal carry propagate adder. N-1 adders are required where N is the multiplier length.

N partial products is generated using N*M two bit AND gates. Adding of n partial products requires most of the area of the multiplier, which is N-1, M-bit adders. By simple routing, the shifting of the partial products for their proper alignment's performed and does not require any logic.

It has very simple and efficiently pipelined layout in VLSI. The conventional array multiplier is integrated using 16T full adder cell. The existing full adder cell is made of 16T which is implemented using XNOR gate, pass transistor and transmission gate. The 16T full adder cell consumes less power as compared to the conventional CMOS full adder cell that uses 28 transistor. At the output nodes, the cell produces the full swing. When compared to other adders, this low power full adder cell has a drawback of giving large delay. Though circuit can operate with full output voltage swing, it consumes significant amount of power and have more delay compared to other adders having less transistor count. Consider the multiplication of two unsigned n-bit numbers, where A = An-1, An-2 ... Ao is the multiplicand and B =Bn-1, Bn-2..... B0 is the multiplier. The product of these two bits can be written as P7P6P5P4P3P2P1P0.Where P0 is the LSB AND P7 is the MSB.Fig1.shows multiplication of a 4*4array multiplier using carry save adders [11].



				A_3	A_2	A_1	A_0
				B_3	B_2	B_1	B_0
				A_3B_0	A_2B_0	A_1B_0	A_0B_0
			A_3B_1	A_2B_1	A_1B_1	A_0B_1	
		A_3B_2	A_2B_2	A_1B_2	A_0B_2		
	A_3B_3	A_2B_3	A_1B_3	A_0B_3			
P7	P6	P5	P4	P3	P2	P1	P0

Fig.2 4X4 Array Multiplier

The conventional array multiplier uses carry save addition to add the products. [11] The first row will be either half adders or full adders in carry save adder. With help of full adders, if the first row of the partial products is implemented, Cin will be considered „0". Then the carries of each full adder is diagonally forwarded to the next row of the adder. The resulting multiplier is said to be carry save array multiplier as the carry bits are saved for the next stage of addition rather than adding immediately. Hence the name carry save multiplier.

Total Delay in x-bit × y bit Array Multiplier:

Delay due to ANDs in partial products at all level is just one unit AND gate delay. But delay at levels 1 to (y – 1) units of n-bit adders = (y – 1) × delay of one-unit 16-bit adder. The delay in adders is very large if ripple carry adders are used. The delay in adders reduced by using carry-look ahead adders.

Advantage of Array Multiplier

1. An array multiplier― a multiplication method in which an array of identical cells generates new partial product and accumulation of it at the same time.
2. We can use pipelines at each level.
3. The delay is logarithmically equal to the bit size of multiplicand and multiplier if we use the high speed array multiplier circuit.

Disadvantage of Array Multiplier:

To design an array multiplier, we required large number of logic gates.

WALLACE TREE METHOD:

A wallace tree is a digital circuit that multiplies two integers and is one of the efficient hardware implementation, devised by Australian Computer Scientist Chris Wallace in 1964.

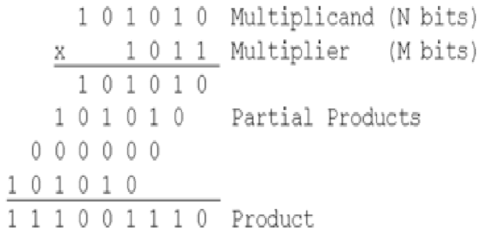
Three steps of the Wallace tree are as follows:

1. Multiply (that is – AND) each bit of one of the arguments, by each bit of the other, yielding n^2 results. The wires are carry different weights depending on position of the multiplied bits.
2. The number of partial products is reduced to two by layers of full and half adders.
3. Two numbers are group in the wires, and add them with a conventional adder.

The second phase works as follows. If three or more wires with the same weight are present add a following layer:

- Take any three wires of the same weights and input them into a full adder. The result will be an output wire of the same weight with a higher weight for each three input wires.
- If two wires of the same weight are left, then input them into a half adder.
- If only one wire is left, then connect it to the next layer.

These computations deals with gate delays and not with wire delays, which can also be very substantial.



```
      1 0 1 0 1 0  Multiplicand (N bits)
    x     1 0 1 1  Multiplier   (M bits)
      1 0 1 0 1 0
    1 0 1 0 1 0    Partial Products
  0 0 0 0 0 0
1 0 1 0 1 0
1 1 1 0 0 1 1 1 0  Product
```

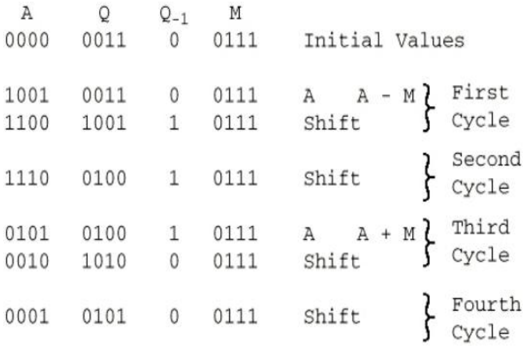
Fig. 3 Wallace Tree Method

BOOTH'S MULTIPLIER:

In multiplication algorithm that multiplies two signed binary numbers in two's complement notation is called **Booth's multiplication algorithm**. The algorithm was invented by Andrew Donald Booth in 1950.Each multiplier bit generates one multiple of the multiplicand that is added to the partial product for the standard add-shift operation. A large number of multiplicands being added, if the multiplier is very large. The number of additions to be performed determines the delay of multiplier. Reduction in the number of the additions, will give the better performance. Booth algorithm is a method that will reduce the number of multiplicand multiples. To represent numbers for a given range, a higher representation radix leads to fewer digits. Because a k-bit binary number can be represented as K/2-digit radix-4 number, a K/3-digit radix-8 number, and so on. By using high radix multiplication, it can deal with more than one bit of the multiplier in each cycle.

Booth's algorithm can be implemented by repeatedly adding (with ordinary unsigned binary addition) one of two predetermined values *X* and *Y* to a product *Z*, then performing a rightward arithmetic shift on *Z*. Let **p** and **q** be the multiplicand and multiplier, respectively; and let *a* and *b* represent the number of bits in **p** and **q**.

1. Determine the values of *X* and *Y*, and the initial value of *Z*. The length of all these numbers should be equal to (*a* + *b* + 1).
1. X: Fill the most significant (leftmost) bits with the value of **p**. Remaining (*b* + 1) bits fill with zeros.
2. Y: Fill the most significant bits with the value of (−**p**) in two's complement notation. Fill the remaining (*b* + 1) bits with zeros.
3. Z: Fill the most significant *x* bits with zeros. To the right of this, append the value of **q**. Fill the least significant (rightmost) bit with a zero.
2. Determine the two least significant (rightmost) bits of *P*.
1. If they are 01, find the value of *Z* + *X*. Ignore any overflow.
2. If they are 10, find the value of *Z* + *Y*. Ignore any overflow.
3. If they are 00, do nothing. Use *Z* directly in the next step.
4. If they are 11, do nothing. Use *Z* directly in the next step.
3. Arithmetically shift the value obtained in the 2nd step by a single place to the right. Let *Z* now equal this new value.
4. Repeat steps 2 and 3 until they have been done *y* times.
5. Drop the least significant (rightmost) bit from *Z*. This is the product of **p** and **q**.



```
A      Q      Q-1   M

0000   0011   0     0111    Initial Values

1001   0011   0     0111    A    A - M }  First
1100   1001   1     0111    Shift      }  Cycle

1110   0100   1     0111    Shift      }  Second
                                          Cycle

0101   0100   1     0111    A    A + M }  Third
0010   1010   0     0111    Shift      }  Cycle

0001   0101   0     0111    Shift      }  Fourth
                                          Cycle
```

Fig.4 Booth's Multiplier

PERFORMANCE PARAMETERS OF MAC UNIT:

Power: The three major components of power are: Transient or dynamic, short circuit and leakage power. The short circuit power is owing to the current conducting path between GND and VDD. The transient or dynamic power is due to the total number of nodes and capacitors charged/discharged in a transition which is expressed as follows:

$$P_{dynamic} = \alpha_{transition} C_{pd} f_{clk} VDD^2$$

where, $\alpha_{transition}$ is the total number of nodes active per transition (node activity factor), C_{pd} is the dynamic power capacitors, f_{clk} is the clock frequency (Input/output) and VDD is the supply voltage. So, in the MAC unit the major portion of power is contributed due to transient power that mainly depends upon the node activity factor and dynamic capacitors.

Clock frequency: It is one of the significant parameters persuading the functional power dissipation of MAC unit. The clock frequency is directly proportional to power factor of the MAC unit, therefore reducing clock frequency may proportionally reduce power, on the other hand, the MAC speed and throughput simultaneously reduced. In order to preserve the throughput for reduced clock frequency parallelism and pipelined architecture have to be considered. The MAC architectural power (block level) is characterized in terms of bits of the component (multipliers and adders) and their operating frequency which can be expressed as:

$$P_{functional\text{-}block} = \Delta_1 \sum_{input\ i} f_i + \Delta_2 \sum_{input\ j} f_j = \Delta_1 f_{in} + \Delta_2 f_{out}$$

where, f_{in} and f_{out} is the input and output frequency of MAC unit and Δ_1, Δ_2 are the empirical coefficients derived from gate-level simulation.

Figure of merit: The Figure of Merit (FOM) is expressed as:

$$FOM = \frac{f}{PV} \text{X}100$$

where, P is the total power of MAC unit for the given voltage (V) operating at a given frequency and this performance parameter should be minimum.

Throughput: The throughput of the MAC design is computed with respect to the clock frequency f_{clk} and latency in various pipe stages. The throughput of the MAC can be expressed as

$$\text{MAC}_{\text{operation/sec}} = \frac{\text{Frequency}}{\text{Cycle each MAC operation}} \text{XParaller MAC units}$$

III. CONCLUSION

We have seen different application of MAC unit for the various application. MAC unit designed with various multiplier among them Booth's multiplier is having highest operating speed and consumes less power. MAC unit is high in demand in Digital signal processing to provide the basic hardware for the systems. MAC unit is in use for all type of arithmetic operation such as addition multiplication, division, squares and square-root. MAC unit must be superior in terms of area, delay, power consumption, speed and complexity.

REFERENCES

[1]. Meenu S Ravi1, R H Khade and Ajit Saraf, "Design of Fast Floating Point Multiply Accumulate Unit using Ancient Mathematics for DSP Applications." *European Journal of Advances in Engineering and Technology*, pp.55-59, 2015.

[2]. Ku. Shweta N. Yengade, Associate Prof. P. R. Indurkar, "Review On Design Of Low Power Multiply And Accumulate Unit Using Baugh-Wooley Based Multiplier." *International Research Journal of Engineering and Technology (IRJET)* Volume: 04 Issue: 02 Feb -2017.

[3]. Shaik Nasar, K. Subbarao, "Design & Implementation of MAC Unit Using Reversible Logic." *International Journal of Engineering Research and Applications (IJERA)* Vol. 2, Issue5, September- October 2012, pp.1848-1855

[4]. G.Indira, G. Madhusudhana Rao, P.Jaya Babu, M. Ravi Kiran, "An Efficient Architecture of MAC Unit with LAW Multiplier." *International Journal of Advanced Scientific Technologies in Engineering and Management Sciences (IJASTEMS-ISSN: 2454-356X)* Volume.2, Issue.11, November.2016

[5]. Saleh, Hani, and Earl E. Swartzlander, "A floating-point fused add-subtract unit." In 2008 *51st Midwest Symposium on Circuits and Systems*, pp. 519-522. IEEE, 2008.

[6]. Samy, Rodina, Hossam AH Fahmy, RamyRaafat, Amira Mohamed, Tarek ElDeeb, and Yasmin Farouk, "A decimal floating-point fused-multiply-add unit." *In 2010 53rd IEEE International Midwest Symposium on Circuits and Systems*, pp. 529-532. IEEE, 2010.

[7]. Michael F. Cowlisha, "Decimal Floating-Point: Algorism for Computers." *Proceedings of the 16th IEEE Symposium on Computer Arithmetic (ARITH'03)* IEEE, 2003

[8]. A. D. Robison, "N-Bit Unsigned Division via N-Bit Multiply-Add," *Proceedings of the 17th IEEE Symposium on Computer Arithmetic*, pp. 131-139, 2005.

[9]. Singh, Harpreet, and Chakshu Goel, "Design Approaches for a Novel Reversible 4-bit Comparator." *IJMSC-International Journal of Mathematical Sciences and Computing (IJMSC)* (2015)