International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)
Web Site: www.ijettcs.org Email: editor@ijettcs.org, editorijettcs@gmail.com
Volume 2, Issue 4, July – August 2013 ISSN 2278-6856

A Review of Different Type of Multipliers and Multiplier-Accumulator Unit.

Soniya[1], Suresh Kumar[2]

[1]M.Tech. Student U.I.E.T., M.D.U. Rohtak
[2]Assistant Professor, U.I.E.T., M.D.U. Rohtak

Abstract: ***High speed and low power MAC unit is utmost requirement of today's VLSI systems and digital signal processing applications like FFT, Finite impulse response filters, convolution etc. In this paper, we have discussed different types of multipliers like booth multiplier, combinational multiplier, Wallace tree multiplier, array multiplier and sequential multiplier. Each multiplier has its own advantages and disadvantages. Different types of techniques are presented for improving the speed and low power consumption like pipelined booth multiplication technique in which pipelining is used in booth multiplier to reduce the delay of each stage. In SPST technique useless portion of the data is removed for reducing the power consumption. And block enabling technique is also used for low power consumption.***

KEYWORDS: Multiplier and accumulator unit, Booth multiplier, Wallace tree multiplier, Sequential multiplier, Pipelining booth multiplication, SPST, Block enabling technique.

1. INTRODUCTION

Multipliers play an important role in today's digital signal processing and various other applications .In high performance systems such as microprocessor, DSP etc addition and multiplication of two binary numbers is fundamental and most often used arithmetic operations. Statics shows that more than 70% instructions in microprocessor and most of DSP algorithms perform addition and multiplication [1]. So, these operation dominates the execution time. That's why, there is need of high speed multiplier. The demand of high speed processing has been increasing as a result of expanding computer and signal processing applications.

Low power consumption is also an important issue in multiplier design. To reduce significant power consumption it is good to reduce the number of operation thereby reducing dynamic power which is a major part of total power consumption So the need of high speed and low power multiplier has increased. Designer mainly concentrate on high speed and low power efficient circuit design. The objective of a good multiplier is to provide a physically packed together, high speed and low power consumption unit. In this paper first we describe different types of multipliers: Booth multiplier, Sequential multiplier, combinational multiplier, Wallace tree multiplier. In next section we will discuss different techniques used in MAC for efficient operations.

2. DIFFERENT MULTIPLIERS

An efficient multiplier should have following characteristics:-

Accuracy:- A good multiplier should give correct result.

Speed:- Multiplier should perform operation at high speed.

Area:- A multiplier should occupies less number of slices and LUTs.

Power:- Multiplier should consume less power.

Multiplication process has three main steps[2]:

1. Partial product generation.
2. Partial product reduction.
3. Final addition.

For the multiplication of an *n*-bit multiplicand with an *m*-bit multiplier, *m* partial products are generated and product formed is *n* + *m* bits long. Here we discuss about four different types of multipliers which are

1. Booth multiplier.
2. Combinational multiplier.
3. Wallace tree multiplier.
4. Array multiplier.
5. Sequential multiplier.

2.1 Booth multiplier:-

Booth multiplication algorithm gives a procedure for multiplying binary integers in signed -2's complement representation.

Following steps are used for implementing the booth algorithm:-Let X and Y are two binary numbers and having m and n numbers of bits(m and n are equal) respectively.

Step 1 Making booth table: In booth table we will take four columns one column for multiplier second for previous first LSB of multiplier and other two (U and V) for partial product accumulator (P)[3].

1. From two numbers, choose multiplier (X) and multiplicand (Y).
2. Take 2's complement of multiplicand (Y).
3. Load X value in the table.
4. Load 0 for X-1 value.
5. Load 0 in U and V which will have product of X & Y at the end of the operation.
6. Make n rows for each cycle because we are multiplying m and n bits numbers.

Table1: Booth Table

U	V	Xi	Xi-1

Step2 Booth algorithm: Booth algorithm requires examination of the multiplier bits, and shifting of the partial product(P). Prior to the shifting, the multiplicand may be added to P, subtracted from the P, or left unchanged according to the following rules:

1. Xi Xi-1
0 0 Shift only
1 1 Shift only
0 1 Add Y to U and shift
1 0 Minus Y from U and shift

2. Take U & V together and shift arithmetic right shift which preserves the sign bit of 2's complement number. So, positive numbers and negative numbers remains positive and negative respectively.

3. Circularly right shift X because this will prevent us from using two registers for the X value.

Repeat the same steps until n no. of cycles are completed. In the end we get the product of X and Y.

2.2 Combinational Multiplier:

Combinational Multipliers do multiplication of two unsigned binary numbers .This multiplier is also used for the multiplication of two signed number. Each bit of the multiplier is multiplied against the multiplicand, the product is associated according to the position of the bit within the multiplier, and the resulting products are then added to form the final result. Main advantage of binary multiplication is that the generation of intermediate products are easy. If the multiplier bit is a 1, the product is an correctly shifted copy of the multiplicand; if the multiplier bit is a 0, the product is simply 0.In most of the systems combinational multipliers are slow and take a lot of area [4].

2.3 Wallace Tree Multiplier:

A Wallace tree multiplier is an efficient hardware implementation of a digital circuit that multiplies two integers devised by an Australian computer scientist Chris Wallace in 1964. Wallace tree reduces the no. of partial products and use carry select adder for the addition of partial products.

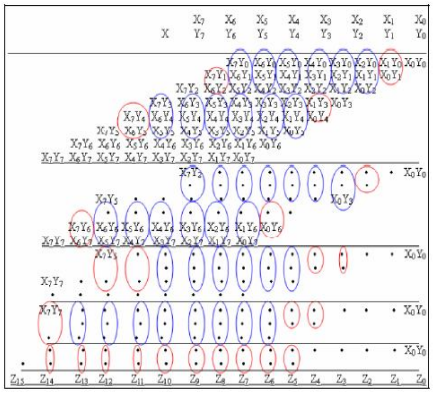

Figure1: Example of 8 bit×8 bit Wallace tree multiplier [11].

In this figure blue circle represent full adder and red circle represent the half adder.

Wallace tree has three steps:-

1. Multiply each bit of multiplier with same bit position of multiplicand. Depending on the position of the multiplier bits generated partial products have different weights.

2. Reduce the number of partial products to two by using layers of full and half adders.

3. After second step we get two rows of sum and carry, add these rows with conventional adders.

Explanation of second step:-

. As long as there are three or more rows with the same weight add a following layer:

1. Take any three rows with the same weights and input them into a full adder. The result will be an output row of the same weight i.e sum and an output row with a higher weight for each three input wires i.e carry.
2. If there are two rows of the same weight left, input them into a half adder.
3. If there is just one row left, connect it to the next layer.

The advantage of the Wallace tree is that there are only O(log n) reduction layers (levels), and each layer has O(1) propagation delay. As making the partial products is O(1) and the final addition is O(log n), the multiplication is only O(log n), not much slower than addition (however, much more expensive in the gate count). For adding partial products with regular adders would require $O(\log n^2)$ time.

2.4 Array Multiplier:

Array multiplier is well known due to its regular structure. Multiplier circuit is based on repeated addition and shifting procedure. Each partial product is generated by the multiplication of the multiplicand with one multiplier digit. The partial product are shifted according to their bit sequences and then added. The summation can be performed with normal carry propagation adder. N-1 adders are required where N is the no. of multiplier bits.

2.5 Sequential Multiplier:

If we want to multiply two binary number (multiplicand X has n bits and multiplier Y has m bits) using single n bit adder, we can built a sequential circuit that processes a single partial product at a time and then cycle the circuit m times. This type of circuit is called sequential multiplier. Sequential multipliers are attractive for their low area requirement. In a sequential multiplier, the multiplication process is divided into some sequential steps. In each step some partial products will be generated, added to an accumulated partial sum and partial sum will be shifted to align the accumulated sum with partial product of next steps. Therefore, each step of a sequential multiplication consists of three different operations which are generating partial products, adding the generated partial products to the accumulated partial sum and shifting the partial sum. Figure 1 shows partial product generation and addition in a sequential multiplier.

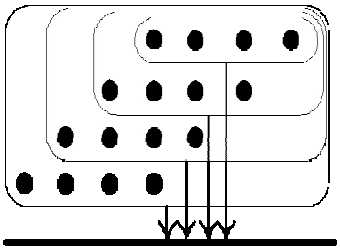

Figure2: Row by row addition in a sequential multiplier[5].

In what follows, the terms multiplicand and multiplier refer to the first and second operands of a given multiplication, respectively. In each multiplication step, one or more multiples of the multiplicand are generated and added to the partial sum through a two- or multi-operand addition operation. Sequential multipliers can take a number of clock cycles to produce a result. That's why even they can work on high clock frequency, but the latency in terms of absolute time get an output may be more or equal to that of combinational multiplier. If the numbers we want to multiply are small then a combinational multiplier is ok, as it is very easy to code.

3. CONVENTIONAL MAC UNIT

MAC are the building blocks of the processor and has a great impact on the speed of processor.MAC is composed of adder, multiplier and an accumulator. The inputs for the MAC are to be fetched from memory location and fed to multiplier block of MAC, which will perform multiplication and give the result to adder which will accumulate the result and then will store the result into a memory location.

MAC mainly consist two parts [2]:-

- Multiplier
- Accumulator.

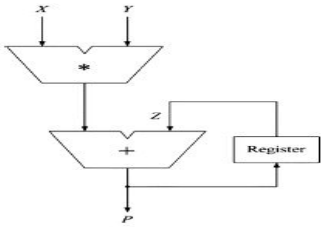

Figure3: Diagram of general MAC [7].

N-bit 2's complement binary number can be represented as $X = -2^{N-1}x_{N-1} + \sum_{i=0}^{N-2} x_i 2^i \quad , \quad x_i \in 0,1$ (1)

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

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

$$d_i = -2x_{2i+1} + x_{2i} + x_{2i-1} \quad (3)$$

If we make use of equation 2 then multiplication is

$$X \times Y = \sum_{i=0}^{N/2-1} d_i 2^{2i} Y \quad (4)$$

If all of these equations are used then multiplication-accumulation result can be expressed as

$$P = X \times Y + Z = \sum_{i=0}^{\frac{N}{2}} d_i 2^i Y + \sum_{j=0}^{2N-1} z_j 2^j \quad (5)$$

In equation (5) two terms in right hand side are calculated independently and final result is obtained by adding these two terms and this equation is called as standard equation of MAC.

3.1 Multiplier:

A multiplier can be divided into three steps. The first is radix 4 booth encoding in which a partial product is produced from the multiplier and multiplicand. The second is adder array or partial product compression to add all partial products and convert them into the form of sum and carry. The last is the final addition in which the final multiplication result is generated by adding the sum and carry [6].

Z=A*B+Z.

3.2 Accumulator:

Accumulator basically consists of register and adder. Register hold the output of previous clock from adder. Holding outputs in accumulator register can reduce additional add instruction. An accumulator should be fast in response so it can be implemented with one of fastest adder like carry look ahead adder or carry skip adder or carry select adder[4].

4. DIFFERENT DESIGNS OF MAC UNIT

4.1 Pipelined Booth Multiplier:

The pipelining is a popular technique to increase throughput of a high speed system which divides total system into several small cascade stages and add some registers to synchronize output of each stage. As the no. of stages increases, the power consumption and area gets increased. So, most of the times pipelining technique can be introduced in Wallace tree in order to improve the performance. Also, when arithmetic throughput is more important than latency, pipelined multipliers are useful

because the introduction of registers along the array reduces the unnecessary activity [8].

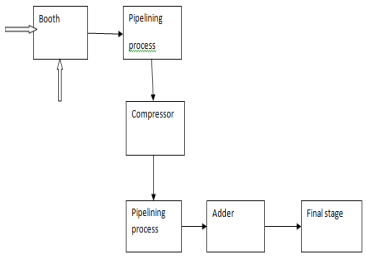

Figure4: Block diagram of pipelining booth multiplier.

4.1.1 Booth Encoder and Decoder:
Radix 2 multiplier does not work well when the multiplier has isolated ones. In such cases recoded multiplier has more numbers of one's when compared to actual multiplier. To multiply multiplier and multiplicand Radix 4 booth algorithm starts from grouping multiplicand by three bits and encoding them into one of (-2,-1,0,1,2). Then multiplier is multiplied with the generated encoding bits and produces the partial product which is known as booth decoder.
Modified booth encoder is used to recode the multiplicand bit in order to reduce the number of partial product. This encoder reduces the three bits into single bit. It takes earlier, present and next bit into account to convert that bit into the single bit.

4.1.2 Compressor:
The Wallace tree construction method is usually used to add the partial products in a tree-like fashion in order to produce two rows of partial products that can be added in the last stage. The Wallace tree is fast since the critical path delay is proportional to the logarithm of the number of bits in the multiplier. The prominent method considers all the bits in each column at a time and compresses them into two bits (a sum and a carry).For compress them into two bits many type of compressor are used such as 4:2 compressor, 3:2 compressor, 5:3 compressor.

4.1.3 Pipelining:
Pipelining is a concept to reduce the delay in the critical path. It is done by adding registers or latches in the data path .By eliminating the delay in the critical path the speed and throughput is increased. Pipelining block is constructed using registers. Registers consists of latches (flip-flops). Pipelining is a popular technique to increase throughput of a high speed system, which divides total system into several small cascade stages and adds some register to synchronize outputs of each stages. Also parallel pipeline architecture is considered to be most suitable for low voltage and low power system. In a pipelining system, the maximum operating frequency is limited by the slowest stage which has the longest delay time. Mostly, D-flip flop is used as the register.

4.1.4 Final Stage Addition:
This stage is also crucial for any multiplier because in this stage addition of large size operands is performed so in this stage fast carry propagate adders like Carry-look Ahead Adder or Carry Skip Adder or Carry Select Adder and other adders such as Carry Save Adder can be used as per requirement [9].

4.2 Spurious Power Suppression Technique:
By using SPST we can reduce power consumption in addition process. In booth multiplication, when two numbers are multiplied some portion of data may be zero in partial products, so this data can be neglected. In other words saving those computations can significantly reduce the power consumption by transient signals. For this purpose SPST equipped modified booth encoder is used which is controlled by a detection unit. As shown in the figure of SPST the total power is divided in to MSB POWER and LSB POWER. By using Transient Power minimization technique we can eliminate MSB POWER, Provided MSB data should not affect the computation. Some cases are studied and in 1st case a transient state in which the spurious transitions of carry signals occur in the MSP though the final result of the MSP are unchanged. The 2nd and the 3rd cases describe the situations of one negative operand adding another positive operand without and with carry from LSP, respectively. Moreover, the 4th and the 5th cases respectively demonstrate the addition of two negative operands without and with carry-in from LSP. In all of the above cases, the results of the MSP are knowable. Therefore the computations in the MSP are useless and can be neglected. The data are separated into the Most Significant Part (MSP) and the Least Significant Part (LSP). To know whether the MSP affects the computation results or not, detection logic unit to detect the effective ranges of the inputs. The detection unit has one of the two operands as its input to decide whether the booth encoder calculates redundant computations.

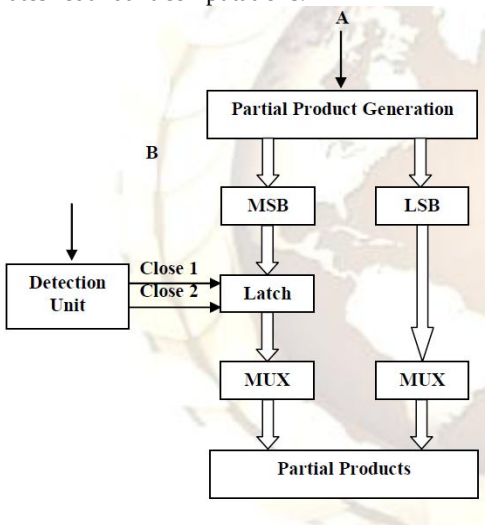

Figure5: SPST equipped modified booth multiplier [10].

In SPST equipped modified booth encoding the MSB power is suppressed based on the close1 and close2 signals and these close1 and close2 signals are generated by detection unit. If the MSB part contains all redundant terms, the total MSB power is eliminated by SPST booth encoding.

4.3 Block Enabling Technique:

The basic building block for the MAC unit are multiplier, adder, and register. In block enabling technique delay of each stage is checked and every block gets enabled only after the expected delay. So, in this technique when inputs are not enable, the successive blocks are disabled thus saving power. Each of the block in the MAC unit has an enable signal to save power[12].

The basic gate that is required to enable or disable the MAC is controlled using an AND gate. it is examined that delay reduces with increase in width. As the NAND gate has delay, the blocks connected to the output of AND gate are disable until this time and these blocks are enabled only after the output are available, hence saving power. To completely understand this technique we can take 1 bit MAC example

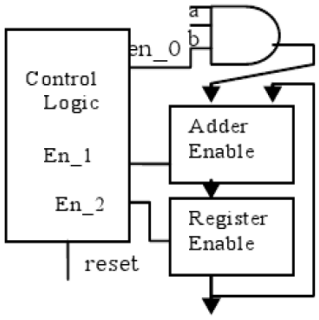

Figure6: Control logic for block enable technique [12]

1 bit MAC unit is designed with clock gating and enable pin. Initially when the input is applied, all the blocks are enabled simultaneously, the FA block would compute the result on unknown data until AND gate delay, and the register block would be receiving unknown data for register gate delay – AND gate delay and hence there is wastage of power as these data's are not actual ones. So, in this a control signal is incorporated that enable the blocks only after the outputs are available at their inputs. Hence we call this technique as block enable technique. Based on the delay of each block, a control signal is generated to enable the blocks.

5. CONCLUSION:

Here we have studied different multipliers and concluded that for large binary number multiplication sequential multiplier perform better than the combinational multiplier in term of speed and area. Considering different technique or design of MAC unit, pipelining booth multiplication gives good performance in terms of speed and SPST and block enabling technique are better in low power consumption and area.

REFERENCES:

[1] Naveen Kumar, Manu Bansal, Navnish Kumar" VLSI Architecture of Pipelined Booth Wallace MAC unit" International Journal of Computer Application(0975-8887)

[2] Fayed, Ayman A., Bayoumi, Magdy A., "A Merged Multiplier-Accumulator for high speed signal processing applications", IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp 3212 -3215, 2002.

[3] Abenet Getahun "Booth Multiplication Algorithm" Fall 2003 CSCI 401.

[4] Sarita Singh, Sachin Mittal"VHDL Design and implementation for optimum delay and area for Multiplier and Accumulator unit by 32 bit Sequential Multiplier" International Journal of engineering Trends and Technology Volume3 issue 5-2012

[5] N. Honarmand, M.R.Javaheri, N.Sedaghati-Mokhtari and A. Afzali-Kusha "PowerEfficient Sequential Multiplication Using Pre-computation" ISCAS 2006.

[6] S.Shafiulla, Syed Jahangir Badashah "Design and Implementation of Radix 4 based high speed Multiplier for ALU's using minimal partial products" International Journal of Advances in Engineering and Technology.

[7] S. Jagadeesh, S. Venkata Chary"Design of Parallel Multiplier-Accumulator based on Radix-4 Modified Booth Algorithm with SPST" International Journal of Engineering Research and Application.

[8] Priya stalin, anuradha, k ranjithkumar, n vaishnav, d vigneswara, s t santhosh "high speed multiplier with pipelining" International Journal of VLSI and Embedded Systems-IJVES .

[9] Navdeep Kaur, Rajeev Kumar Patial "Implementation of Modified Booth Multiplier using Pipeline Technique on FPGA" International Journal of Computer Applications (0975 – 8887).

[10] K. Srishylam, Prof. Syed Amjad Ali, M.Praveena " Implementation of Hybrid CSA, Modified Booth Algorithm and Transient power Minimization techniques in DSP/Multimedia Applications" International Journal of Engineering Research and Applications (IJERA) .

[11] Shanthala S, S. Y. Kulkarni "VLSI Design and Implementation of Low Power MAC Unit with Block Enabling Technique" European Journal of Scientific Research

[12] Iffat Fatima " Analysis of Multipliers in VLSI" Journal of Global Research in Computer Science.