

VLSI DESIGN OF HIGH PERFORMANCE MAC UNIT

Ajitha C^{#1}, V. FemilaSavio^{#1}

¹ Final year ME-Applied electronics, St.Xavier's Catholic College of Engineering

² Assistant Professor, Dept of ECE, St.Xavier's Catholic College of Engineering

ABSTRACT- We proposed a new architecture of multiplier -and- accumulator unit (MAC) for high-speed arithmetic and low power consumption is proposed here. Multiplication occurs frequently in finite impulse response filters, fast Fourier transforms, discrete cosine transforms, convolution, and other important DSP and multimedia kernels. A carry-skip adder (also known as a carry-bypass adder) is an adder implementation that improves on the delay of a ripple-carry adder with little effort compared to other adders. The improvement of the worst-case delay is achieved by using several carry-skip adders to form a block-carry-skip adder. To reduce the adder's delay and power consumption, the adder is divided into variable-sized blocks that balance the inputs to the carry chain. The adder architecture decreases power consumption by reducing the number of logic levels, glitches, and transistors. The objective of a good multiplier and accumulator (MAC) is to provide a physically compact, good speed and low power consuming chip. To save significant power consumption of a VLSI design, it is a good direction to reduce its dynamic power that is the major part of total power dissipation. The goal of this project is to design and implement the MAC unit for high-speed DSP applications. MAC unit is designed by using carry skip adders. The MAC unit architecture is implemented using VHDL code and it is synthesized and simulated using Xilinx ISE.

Keywords- Carry Skip Adder, Array Multiplier, Accumulator, MAC Unit, Block Enable

I. INTRODUCTION

Very Large Scale Integration (VLSI) is the process of creating an Integrated Circuit (IC) by combining thousands of transistors into a single chip. VLSI began in the 1970s when complex semiconductor and communication technologies were being developed. VLSI is one of the basic building blocks of today's higher end or advance technology. The introduction of VLSI technology most ICs had a limited set of functions they could perform. An electronic circuit consists of a CPU, ROM, RAM and other logic. VLSI lets IC makers to add all of these into one chip. As integrated circuit technology has improved to allow more and more components on a chip, digital systems have continued to grow in complexity. As digital systems have become more complex, detailed design of the systems at the gate and flip-flop level has become very tedious and time consuming. For this reason, use of hardware description languages in the digital design process continues to grow in importance. A hardware description

language allows a digital system to be designed and debugged at a higher level before conversion to the gate and flip-flop level. DSP processors are microprocessors designed to perform digital signal processing the mathematical manipulation of digitally represented signals. Digital signal processing is one of the core technologies in rapidly growing application areas such as wireless communications, audio and video processing and industrial control. Digital signal processing (DSP) applications constitute the critical operations which usually involve many multiplications and accumulations. Hence, high throughput multiplier accumulator (MAC) is always a key element to achieve a high-performance digital signal processing application. In the last few years, the main consideration of MAC design has been to enhance its speed. This is because speed and throughput rate are always the concerns of digital signal processing systems. Due to the increase of portable electronic products, low power designs have also become major considerations for the VLSI application.

The limited battery energy of these portable products restricts the power consumption of the system. Therefore, the motivation behind this project is to investigate various pipelined MAC architectures and circuit and the design techniques which are suitable for the implementation of high throughput signal processing algorithms. The goal of this project is to design the VLSI implementation of pipelined MAC for high speed DSP applications. For designing the MAC, various architectures of carry adders are considered. The total process is coded with Verilog to describe the hardware.

Carry Skip Adder (CSKA) uses skip logic in the propagation of carry. It is designed to speed up the addition operation by adding a propagation of carry bit around a portion of entire adder. The carry-in bit designated as Ci. The output of RCA (the last stage) is Ci+4. The Carry Skip circuitry consists of two logic gates.

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 bit. The partial products are shifted according to their bit orders and then added. The addition can be performed with normal carry propagate adder. N-1 adders are required where N is the multiplier length. This method is simple and the addition is done serially as well as in parallel. To improve on the delay and area the ripple carry adder are

replaced with the carry save adders, in which every carry sum signal is passed to the adders of the next stage. Final product is obtained in a final adder by any fast adders. In array multiplication we need to add, as many partial products as there are in multiplier.

II. LITERATURE SURVEY

The design of 8X8 MAC unit has been implemented using block enable technique to reduce the power consumption. This MAC unit has 18 bit output and its operation is to add repeatedly the multiplication results. All the basic blocks of MAC unit are identified and analyzed its performance is analyzed. Power and delay is calculated for the blocks[1].design 1 bit MAC unit is designed and power consumption is calculated based on block enable technique. This block is extended to N bit MAC unit the power consumption of this N bit MAC unit is calculated. The improvement of power-delay product of the MAC unit can be used in high speed DSP application[2].

VLSI architecture and a limited-resource scheduling algorithm for MAC-level DWT processor have been presented in this paper. The scheduling algorithm has been successfully proven that a variety of DWT processors can be efficiently realized by tuning the parameters[7][10]. Given the architecture constraints and DWT parameters, the scheduling algorithm can generate four schedule matrices and enable the data path to perform the DWT computation. In this research, the performance and memory size of LRS algorithm are also investigated. It is clear that the performance is optimized for DWT processor with the limited number of MACs[12].

VLSI architecture for low power MAC unit. The basic building blocks for the MAC unit are identified and each of the blocks is analyzed for its performance. 1-bit MAC unit will be designed with enable to reduce the total power consumption based on above proposed techniques. Using this block, the N-bit MAC unit will be constructed and the total power consumption will be calculated for the MAC unit [3][4]. The dynamic power which is determined by the equation where α is the switching activity factor, C is the capacitance, V is the supply voltage, and f is the clock frequency. To achieve low power in circuits one or more of the parameters must be minimized. The MAC unit designed in this work can be used in filter realizations for High speed DSP applications[5][15].

A design and implementation of low power MAC using the Cadence tool is achieved. This paper also investigates on various architectures of multipliers and adders also on the various techniques to reduce power consumption which are suitable for implementation of high throughput signal processing and at the same time to achieve low power consumption[14][13]. The whole MAC chip is operated at 125 MHz using 1.8 V power supply. The power factor is reduced by 11.28% using Block Enabling Technique. Block Enabling proves to be one of the efficient technique to reduce power, hence one can

implement this to various logic circuits to reduce power[7][6].

Shanthala S et al (2013), proposed an 8x8 multiplier-accumulator (MAC) unit in which a full-adder circuit based mux is used for MAC architecture. The basic building blocks for the MAC unit are identified and each of the blocks is analyzed for its performance. Power and delay is calculated for each blocks. The N-bit MAC unit is constructed and the total power consumption is calculated for the MAC unit. The power reduction techniques adopted in this work, 27% of power is saved. The MAC unit designed in this work can be used in filter realizations for High speed DSP applications. The Full custom design has been carried out for the proposed work and verified using cadence tools[11][12].

Nagaraj Rishna Naik et al (2013), proposed a MAC unit based on the of DSP operation. Mainly it is used in digital filter design. The complexity of the filter response dictates the number of MAC operations required per sample period. Digital filter involve signals in the digital domain (discrete time signals) and are used extensively in application such as digital image processing, pattern recognition and spectral analysis[8][9]. A full adder circuit based on MUX is used in this design. Compared to all other full adder. This calculates delay in each block, control logic is set for the selection of each block in MAC for the selection of block after certain delay. The full custom design is carried out for the proposed chapter and verified using CADENCE VIRTUSO tool[13].

III. PROPOSED MAC UNIT

The increase in demand of portable devices makes Low power device design and it becomes an important field of research. Power dissipation is one of the fundamental design objectives in integrated circuit, after speed. Design of low area, delay and power forms the largest systems in VLSI system design. These three parameters i.e. power, area and speed are always traded off. However, area and speed are usually conflicting constraints, so that improving speed results mostly in larger areas. The addition and multiplication of two binary numbers are the fundamental and most frequently used arithmetic operation in microprocessors, digital signal processors, and data-processing application specific integrated circuits. In Multiplier Accumulator unit addition and multiplication forms the main blocks. High speed and low power MAC units are required for applications of digital signal processing like Fast Fourier Transform, Finite Impulse Response filters, convolution etc. Area and speed of MAC unit are the most significant factors, but sometimes, increasing speed also increases the power consumption, so there is an upper bound of speed for a given power criteria.

Since the various filter designs found in the Digital Signal Processing applications, require computationally efficient multiply and Accumulate

operations so the blocks with the desired characteristics have to be chosen carefully. The target of this thesis is to design and analysis various adder and multiplication schemes for high-speed, area efficient and low power operation Multiplier Accumulator unit. Our proposed MAC architecture consists of 3 sub designs.

1. Design of 16×16-bit array multiplier.
2. Design of CSK.
3. Design of accumulator which integrates both multiplier and adder stages.

A. MAC ARCHITECTURE

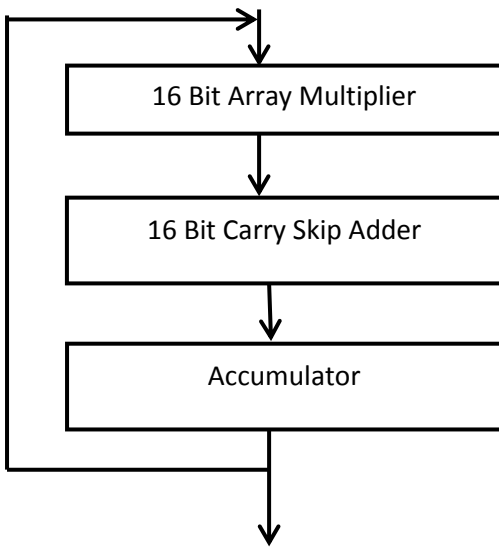


Figure : 1 Architecture of proposed MAC unit

Figure 1 shows the architecture of our proposed MAC unit. The proposed MAC unit consists of a multiplier, adder and an accumulator. First comes the array multiplier whose output is taken as input to the next block (i.e.,)the adder block and then the resultant output is given to the accumulator which is more or less similar to the register (storage device). Then for the repetition of the output values to be counted the output from the accumulator part is given as feedback to the adder block if required. By this repetition the output gets generated which reduces the factors like power consumption, usage of minimum chip area which are all the essential criterions in a VLSI implementation circuits.

B. MULTIPLIER

The composition of an array multiplier is shown in Fig 3. There is a one-to-one topological correspondence between this hardware structure. The generation of N partial products requires N x M two-bit AND gates most of the area of the multiplier is devoted to the adding of the N partial products, which requires N -1 M-bit adders. The shifting of the partial products for their proper alignment is performed by simple routing and does not require any

logic. The overall structure can easily be compacted into a rectangle, resulting in a very efficient layout. Due to the array organization, determining the propagation delay of this circuit is not straightforward. Consider the implementation of the partial sum adders are implemented as ripple-carry structures.

$$T_{mult} = [(M-1) + (N-2)]t_{carry} + (N-1)t_{sum} + t_{AND} \dots (1)$$

From equation 1 where t_{carry} is the propagation delay between input and output carry, t_{sum} is the delay between the input carry and sum bit of the full adder, and t_{AND} is the delay of the AND gate. Since all critical paths have the same length, speeding up just one of them—for instance, by replacing one adder by a faster one such as a carry-select adder—does not make much sense from a design standpoint. At critical paths have to be attacked at the same time. From the above equation, it can be deduced that the minimization of t_{mul} requires the minimization of both t_{carry} .

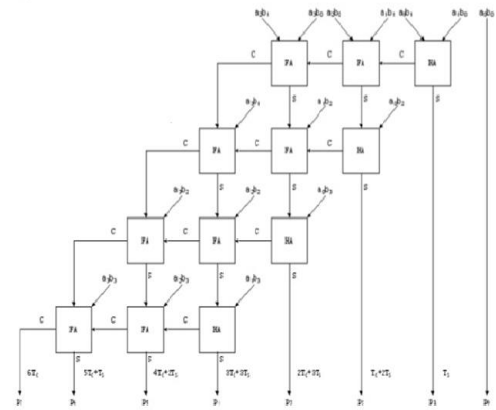


Figure 2: Array Multiplier Architecture.

The figure 2 shows the architecture of array multiplier. Older multiplier architectures employed a shifter and accumulator to sum each partial product, often one partial product per cycle, trading off speed for die area. Modern multiplier architectures use the (Modified) Baugh–Wooley algorithm, Wallace trees, or Dadda multipliers in which we have made use of Dadda multiplier to add the partial products together in a single cycle for improving the performance of the array multiplier.

C. CARRY SKIP ADDER

carry-skip adder (also known as a carry-bypass adder) is an adder implementation that improves on the delay of a ripple-carry adder with little effort compared to other adders. The critical path of a carry-skip-adder begins at the first full-adder, passes through all adders and ends at the sum-bit. Carry-skip-adders are chained to reduce the overall critical path.

a. Design of Various Carry Skip Adders

Designing a CSKA structure using the conventional CSKA (Conv-CSKA) structure. Designing a modified CSKA structure by combining the concatenation and the incrimination schemes to the conventional CSKA (Conv-CSKA). Structure for enhancing the speed and energy efficiency of the adder.

b. Conventional CSKA

The working principle of CSKA is that it operates in 2 stage, they are generating sum bits from the Ripple Carry Adder block and the carry propagation block. Propagation block uses EXOR gates to generate the propagate bits and the generated bits are fed to the AND gates. The result of propagation block is given to the selection line of multiplexer, which selects the carry.

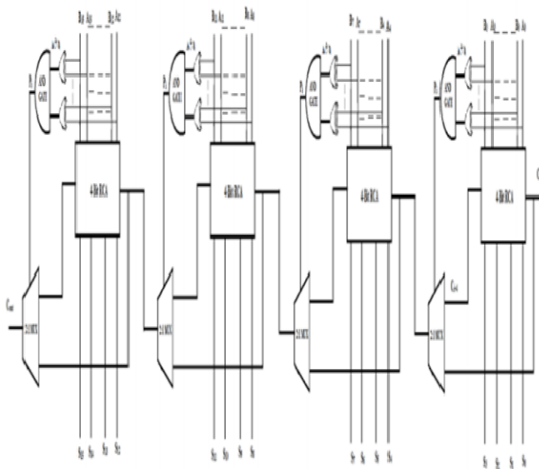


Figure 3: Conventional 16-bit CSKA

The figure 3 shows the basic architecture of conventional 16-bit CSKA. The conventional structure of the CSKA consists of stages containing chain of full adders (FAs) (RCA block) and 2:1 multiplexer (carry skip logic). The RCA blocks are connected to each other through 2:1 multiplexers, which can be placed into one or more level structures. The CSKA configuration (i.e., the number of the FAs per stage) has a great impact on the speed of this type of adder. Many methods have been suggested for finding the optimum number of the FAs. The techniques presented in make use of VSSs to minimize the delay of adders based on a single level carry skip logic.

c. Ripple Carry Adder using Efficient Adders

A Ripple Carry adder is a digital circuit that produces the arithmetic sum of two numbers. In, RCA the sum resulting at each stage need to wait for the incoming carry signal to perform the sum operation. The carry propagation can be speed-up in two ways. The first and most obvious way is to use a faster logic circuit technology. The second way is to generate carries by means of forecasting logic that does not

rely on the carry signal being rippled from stage to stage of the adder.

d. Carry Skip Adder using Efficient Adders

A Carry Skip Adder consists of a simple ripple carry adder with speed up carry chain called a skip chain. The chain defines the distribution of ripple carry blocks, which compose the skip carry blocks, which compose the skip adder. Skip Carry adder are divided into blocks, where a special circuit detects quickly if all the bits to be added are different ($P_i = 1$ in the entire block). The carry skip adder provides a compromise between a ripple carry adder and a CLA adder.

e. Hybrid Variable Latency CSKA Structure

To provide the variable latency feature for the VSS CSKA structure, we replace some of the middle stages in our proposed structure with a PPA modified in this project. It should be noted that since the Conv-CSKA structure has a lower speed than that of the proposed one, in this section, we do not consider the conventional structure.

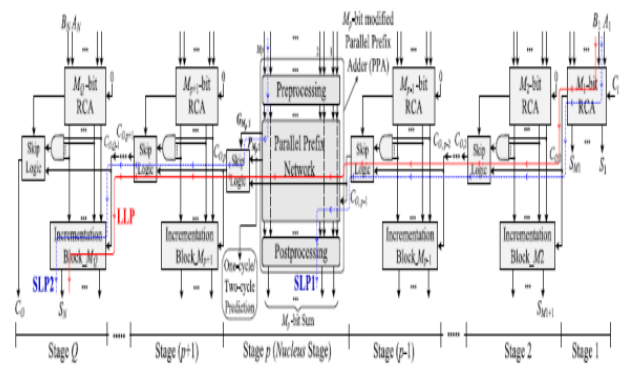


Figure 4 Hybrid Variable Latency CSKA Structure

In the figure4 shows the hybrid structure, the prefix network of the Brent–Kung adder is used for constructing the nucleus stage. One the advantages of the this adder compared with other prefix adders is that in this structure, using forward paths, the longest carry is calculated sooner compared with the intermediate carries, which are computed by backward paths. In addition, the fan-out of adder is lesser than other parallel adders, while the length of its wiring is smaller.

D ACCUMULATOR

The accumulator is designed to store cumulative addition of MAC unit, it is a group of registers which are designed for this. It has a reset pin which is used for resetting. When the reset value is high the content of the accumulator becomes zero and when reset is not equal to zero, the accumulator starts accumulating the summation.

The inputs to the accumulator are output from the array multiplier and the previous content of the accumulator.

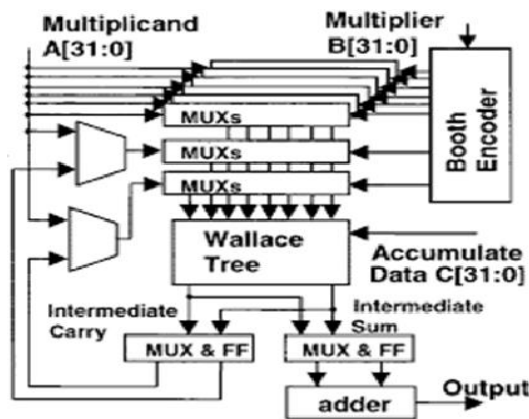


Figure5: Accumulator structure

IV. RESULTS AND DISCUSSION

The output of proposed adder is as shown in the figure. The output obtained from the adder which uses proposed algorithm is synthesized using Xilinx software and also the various parameters like CPU time, delay and total memory used are noted. This timing report shows the total timing required for executing the design and also the total memory used. When the memory requirement is low it means the method is area efficient.

A. OUTPUT OF CARRY SKIP ADDER



Figure 6: Output of carry skip adder

Figure 6 shows the output of carry skip adder. From the Xilinx output of carry skip adder, inputs DataA (15:0) and DataB (15:0) represents input numbers of size 16. The signal Dataout selects whether addition operation is going to be carried out. The output signal carryout represents final added output format.

B. OUTPUT OF 16 BIT MAC UNIT

Figure 7 shows the MAC unit output. From the Xilinx output of carry skip adder, inputs DataA (15:0) and DataB (15:0) represents input numbers of size 16. The signal Dataout selects whether addition operation is going

to be carried out. The output signal carryout represents final added output format.

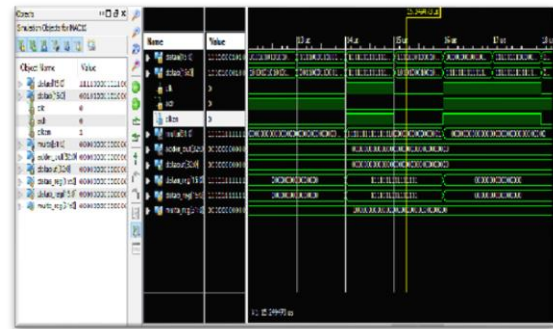


Figure 7 Output of MAC unit

C.PERFORMANCE ANALYSIS

Area efficient means the number of resources used must be low. Area dependent parameters are number of slices, number of I/O LUT, Number of bonded IOBs and total memory usage. Every slice contains four logic function generators, eight storage elements, wide function multiplexers and carry logic. A Look Up Table (LUT) is a collection of logic gates hard-wired on the FPGA. LUTs store a predefined list of outputs for every combination of inputs and provide a fast way to retrieve the output of a logic operation.

a. Synthesis report

The output obtained from carry skip adder which uses proposed algorithm is synthesized using Xilinx software and also the various parameters like CPU time, delay and total memory used are noted. This timing report shows the total timing required for executing the design and also the total memory used.

b. Device utilization summary

Bonded IOBs is the total amount of input output buffers used in the program. The number of slices, number of four input Look Up Tables and the Input Output Blocks are listed in the following table. The devices utilized for single precision carry skip adder.

Device Utilization Summary		
Slice Logic Utilization	Used	Available
Number of Slice Registers	97	93,120
Number used as Flip Flops	97	
Number used as Latches	0	
Number used as Latch-thrus	0	
Number used as AND/OR logics	0	
Number of Slice LUTs	37	46,560
Number used as logic	32	46,560
Number using O6 output only	32	
Number using O5 output only	0	
Number using O5 and O6	0	
Number used as ROM	0	
Number used as Memory	0	16,720
Number used exclusively as route-thrus	5	
Number with same-slice register load	4	
Number with same-slice carry load	1	
Number with other load	0	
Number of occupied Slices	24	11,640
Number of LUT Flip Flop pairs used	93	
Number with an unused Flip Flop	0	93
Number with an unused LUT	5	93

Figure 8 Device Utilization Summary

c. Power report

The power gives the summary of total power utilized by the carry skip adder. It also shows the power utilized by various on chip component such as clocks, IOs etc. It also gives detail about the area used in the chip, the available space and the utilization. Figure 9 says the power report of MAC unit.

```

-----
2. Summary
2.1. On-Chip Power Summary
-----
|                                     |
|                                     |
|-----|-----|-----|-----|-----|
| On-Chip | Power (mW) | Used | Available | Utilization (%) | |
|---|---|---|---|---|---|
| Clocks | | 0.00 | 1 | --- | --- |
| Logic | | 0.00 | 37 | 46560 | 0 |
| Signals | | 0.00 | 172 | --- | --- |
| IOs | | 0.00 | 68 | 240 | 28 |
| DSPs | | 0.00 | 1 | 288 | 0 |
| Static Power | | 1292.75 | | | |
| Total | | 1292.75 | | | |
|-----|-----|-----|-----|-----|
2.2. Thermal Summary
-----
| Thermal Summary |
|-----|-----|
| Effective TjA (C/W) | 2.7 |
| Max Ambient (C) | 81.5 |
| Junction Temp (C) | 53.5 |
-----

```

Figure 9 Power Report Summary

IV. CONCLUSION

A design of high performance 16-bit Multiplier-and-Accumulator (MAC) is implemented here and a static CMOS CSKA structure called CI-CSKA was proposed with MAC, which exhibits a higher speed and lower energy consumption compared with those of the conventional one.

The speed enhancement was achieved by modifying the structure through the concatenation and incrimination techniques. MAC unit performs important operation in many of the digital signal processing (DSP) applications. The multiplier is designed using array multiplier and the adder is done with carry skip adder. The efficiency of the proposed structure for both FSS (Frequency Response in element Shape) and VSS was studied by comparing its power and the total design is coded with verilog-HDL and the synthesis is done using Cadence RTL compiler using typical libraries of Xilinx 14.5 Version . It exploits a modified parallel adder structure at the middle stage for increasing the slack time, which provided us with the opportunity for lowering the energy consumption by reducing the supply voltage. The efficacy of this structure was compared versus those of the variable latency RCA, C2SLA, and hybrid C2SLA structures. Again, the suggested structure shows the lowest delay making itself as a better candidate for high -speed low-energy applications.

REFERENCES

1. Aamir A. Farooqie, and Vojin G. Oklobdzija(1998), 'General data path organization of MAC unit for VLSI implementation of DSP processors', IEEE Transaction on Circuits and System., pp. 260-263.
2. AapurvaKaul and Abhijeetkumar (2016), 'Simulation of 64-bit MAC unit using kogge stone adder and ancient indian mathematics', Int. Journal of Engineering Research and Application., june. ISSN: 2248-9622 vol. 6, pp. 01-05.
3. AkondiNarayanaKiran and G.VeeraPandu(2012), 'Hardware efficient VLSI architecture of parallel MAC for high speed signal processing application' Int. Journal of Engineering Research and Application., August ISSN: 2278-0181 vol. 1, issue 6, pp. 01-05.
4. Anusree T U and Bonifus P L (2016), 'Design and analysis of modified fast compressors for MAC unit,' Int. Journal of Computer Trends and Technology., june. vol. 36, pp. 213-218.
5. P.Asadee., (2009), 'A new MAC design using high speed partial product summation tree,' IEEE Transaction on Circuits and System., pp. 234-231.
6. AshishB.Kharateand P.R. Gumble (2009), 'VLSI Design and implementation of low power MAC for digital FIR filter,' Int. Journal of Electronics and communication and computer Engineering., july. vol. 4, pp. 58-61.
7. Avisek Sen., DebarshiDatta., and ParhaMitra (2013), 'Low power MAC unit using DSP processors,' International journal of Recent Technologyand Engineering., January. ISSN: 2277-3878, volume-6, pp. 93-95.

8. Belle W. Y. Wei., Wen- Jung Liu., and Xiaoping Huang (1994), 'A high-performance CMOS redundant binary Multiplication and accumulation(MAC) unit,' IEEE Transaction on Circuits and System. Fundamental Theory and Application, January. Vol 41, no. 1 pp. 33-39.
9. N. Chaumartin., G. Geort., A.Lorenzi., J.M. Troude., G. Vanneuville., and V. Verfaillie (1991), 'VLSI ASIC design for MAC video processing integration in SGS-THOMSON microelectronics chip,' IEEE Transaction on Circuits and System., pp. 131-134.
10. CyrilPrassana Raj., Kulkarni S.Y and Shanthala S (2009), 'Design and VLSI implementation of pipelined multiply accumulate unit,' IEEE Transaction On Second International Conference on Emerging Trends in Engineering and Technology., pp. 381-386.
11. DivyanshuRao., Naveen Khare., and Ravi Mohan (2016) , 'VLSI implementation of high speed MAC unit using karatsuba multiplication technique,' Journal of Network Communications and Emerging Technologies., January. Volume 6, issue 1, pp. 24-28.
12. DurgaBhavani. A, and NagarajKrisnhnanaik(2015), 'VLSI design of low power MAC unit,' International Journal of Current Engineering and Science Research., January. Volume 2, issue 6, pp. 113-118.
13. Ganeshbabu.C, and Santhiya (2016), 'Design and implementation of multiplier using Reversible logic in Multiple Accumulate unit,' International Journal of Advanced research in Electronics and Communication Engineering ., April. volume 5, Issue 4, pp. 1080-1086.
14. Gitika Bhatia., Karanbir Singh Bhatia., Pradeep Kumar., and ShashankSrivastava (2015) , 'Design and implementation of MAC unit based on vedic square and its application,' IEEE Transaction on Conference on Electrical computer and Electronics., pp. 998-1001.
15. Haoran Wang., JoknGlossner., Kai Chirca and Micheal Schulte (2010), 'A static low power high performance 32-bit carry skip adder,' IEEE Transaction on Conference on Electrical computer and Electronics., pp. 99-102.