

Low Power Pipelined Multiply-Accumulate Unit using Optimized Adders and Multipliers

Sujatha Cyril, Dharmistan K. Varugheese

Date of Submission: 15-09-2020

Date of Acceptance: 29-09-2020

ABSTRACT: In the majority of the Digital signal processing (DSP) applications, the critical operations usually involve many multiplications and /or accumulations. So for real time signal processing applications, high through put multiplier –accumulator (MAC) is always a key element to achieve a high-performance digital signal processing. In the last few years the main consideration of MAC design is to enhance its speed. This is because speed and through put rate are always the concerns of digital signal processing systems. However due to the increase of portable electronic products, low power designs also become another major consideration. This is because the limited battery energy of these portable products restricts the power consumption of the system. Therefore the main motivation of this work is to investigate various pipelined MAC architectures and circuit and the design techniques which are suitable for the implementation of high through put signal processing algorithms. The goal of this project was to design and VLSI implementation of pipelined MAC for high-speed DSP applications at 65nm technology. For designing the pipelined MAC, various architecture of multipliers and one bit full adders are considered. The static and dynamic one bit full adder will be implemented as the basic block. For checking the functionality of the whole system spice code is written using the Hspice by defining all the blocks in the circuit as the sub circuits. Then a schematic capture is done using schematic composer from virtuoso starting from bottom level to top level. Finally the layout for the complete MAC is done using virtuoso.

Key words: Multipliers, adders, low power, high speed, VLSI

I. INTRODUCTION

Digital signal processing (DSP) sounds more familiar if considering its numerous applications in telephony, mobile radio, satellite communications, speech processing, video and image processing, biomedical applications, radar,

and sonar. The objects that digital processing methods deal with are digital signals. It is different from the ones in nature that are continuous. For a digital signal processing system, it is always composed of a) a sample and hold unit to make the continuous signals from some signals sources to be discrete, b) an analog to digital converter converting the discrete signals presented in real numbers to numbers presented by certain bits of '0' and '1', c) a digital processing unit or processor executing different digital signal methods such as FIR filters, d) a digital to analog converter to generate desired control signals to complete the interactions with the environment. A digital signal processor (DSP) is a specialized microprocessor designed specifically for the digital signal processing [1]. It can be the processor used in the embedded system introduced in the above section. KihakShim, IK Kyun Oh [2] proposed a lower power 8×8 bit MAC is designed minimizing the power consumption at each of design levels, and introduced a new method where the transistor count is reduced by 40%. A new booth selector circuit using NMOS PTL (pass transistor logic) which has excellent power delay product is also proposed to reduce the number of transistors for the registers. The architecture of the MAC consists of three typical functional blocks, booths decoder, Wallace tree, ELM adder. To increase the operating frequency a two-stage pipeline scheme is adopted. The opcodes and two 8-bit operands are fed into the input registers. The operands are transferred to the Booth decoder block that consists of Booth encoders and Booth selectors. The Booth decoder converts the operands into 40-bit recoded partial products. Summation of these partial products is performed in Wallace tree block. To balance the delays of each pipeline stage the pipeline registers are inserted between the Wallace tree blocks. The delay introduced by Wallace tree is a dominant component of the delay in MAC. Thus the reduction of the delay in Wallace tree is important for reducing total operation time. The total number of transistor used for design MAC are 4440

transistors [2]. Ichiro Kuroda [3] presented a parallel MAC architecture designed for the DSP applications. The data path architecture of the processor is designed to realize parallel execution of a data transfer and SIMD parallel arithmetic operations. SIMD parallel 16 bit MAC instructions are introduced with the symmetric rounding scheme, which maximizes the accuracy of the 16-bit accumulation. This 16 bit MAC instruction on a 64-bit data path is shown to be efficiently utilized for DSP applications such as the convolution in the multimedia RISC processor. The architecture is implemented using the convolution function and IDCT architectures [3]. Jae Sung Lee, Young SeopJeon, and MyungH.Sunwoo [4] present a new DSP instructions and their hardware architecture for high speed FFT. The instructions perform new operations flows, which are different from the MAC operation on which existing DSP chips heavily dependent. He proposed the DPU (data processing unit) supporting the instructions and shows two times faster than the existing DSP chips for FFT. The architecture has been modelled by the Verilog HDL and logic synthesis has been performed using .35 μ m standard cell library. Dusan Suvakovic and C. Andre T. Salama [5] proposed a pipelined 15 \times 15 bit multiply accumulate unit, optimized for energy recovery systems is presented. The applied architectural and circuit level optimizations are aimed at minimizing the total non-recoverable energy per computation. For that purpose the number of pipeline stages is minimized, utilizing the logic gates based on high fan in differential NMOS trees. The MAC includes three pipeline stages operated from two-phase non-overlapping power clock and process one multiply accumulate operation per clock cycle. The multiply-accumulate (MAC) unit described in this paper features an unconventional architecture, which is more appropriate to adiabatic circuits than the conventional architectures for parallel multiplication. The architectural design strategy was formulated assuming the availability of a latch whose non-adiabatic dissipation and sensing ability is independent of the size of the associated circuit and the load capacitance. Given the latch properties, the total non-adiabatic dissipation in the arithmetic data path depends only on the number of latches and it is minimized for an architecture that requires the Minimum number of latches. Since each adiabatic gate has to include one latch, this effectively means that the data path architecture can be based on a relatively small number of complex logic gates. To enable successful utilization of the new architecture, a Differential latch with such properties is proposed. The

designed MAC data path is based on complex, high fan-in logic gates. Logic functions of these gates are performed by differential output sensing adiabatic latches at their outputs NMOS logic trees with differential, output sensing adiabatic latches at their outputs [5]. Shyh-jye Jou and Chang-Yu Chen [6] a new full-adder circuit for pipeline architecture is proposed. Compared with other full adder circuits, it has high operational speed, smallest transistor number and the lowest power speed ratio. The new full-adder cell is then used to design a pipelined 8 \times 8-b multiplier-accumulator. In the MAC, a special pipelined structure is designed to reduce the latency. Survey on the existing architectures of MAC is carried out and it was concluded that MAC with static and dynamic full adder is very fast when compared with the other architecture. As the selected architecture with static and dynamic full adder consists of both static and dynamic blocks the delay to get the output than other dynamic architectures is reduced. As the full adder is basic block of the whole design, the reduction in the delay of the adder reduces the delay of the whole design. The selected architecture uses less transistor count than other architectures. The selected architecture is best suitable for pipelined architectures where the total latency is equal to the number of clock cycles and best suitable for any DSP applications [7-10]. The power /speed ratio of the design is less compared with the other architectures.

II. ADDERS AND MULTIPLIERS FOR MAC

The most typical feature that differentiates a DSP from any GPP is the Multiply and Accumulate unit. All DSP Algorithms would require some form of the Multiplication and Accumulation Operation of the form. This is the most important block of the complete DSP. It is composed of an adder, multiplier and the accumulator. Usually adders implemented in DSPs are Ripple Carry Adders, Carry-Select or Carry-Save adders as speed is of utmost importance in a DSP. Basically the multiplier will multiply the inputs and give the results to the adder, which will add the multiplier results to the previously accumulated results. This operation eases the computation of the most important formula i.e. $b(n) \times (n-k)$ which is needed in filters, Fourier analyzers, etc. The inputs for the MAC are supposed to be fetched from some memory location and fed to the multiplier block of the MAC, which will perform multiplication and give the result to adder which will accumulate the result and then if needed will also store the result into a memory

location. This entire process is to be achieved in a single clock cycle.

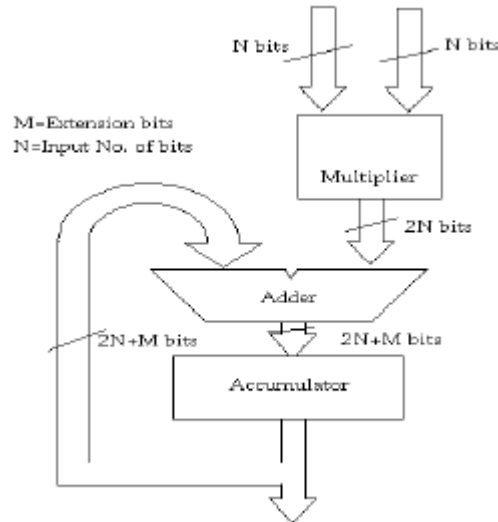
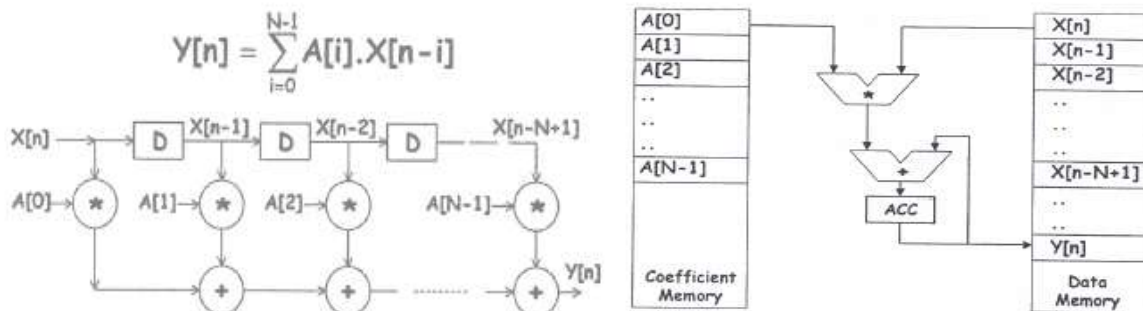


Figure 1 Simple Block Diagram of MAC [1]

Figure 1 shows the block diagram of MAC unit which consists of multiplier, adder and accumulator. First the inputs are applied to multiplier, the multiplied outputs are given as input to the adder, and other input is applied to adder. The result is stored in accumulator. In most of the DSP applications MAC is used for many manipulations, now let us consider how the design

of MAC is used in the design of FIR filter. Figure 2 shows the MAC based FIR filter design; the FIR filter design involves many additions, and multiplications. These manipulations can be implemented by using MAC. In the pipeline stage the inputs are given for the each MAC and final output can be obtained at the last stage of MAC.



(a) Data flow graph (b) Multiply and Accumulate Logic

Figure 2 MAC based implementation of FIR Filter design [1]

2.1 Summary of adders

From the study carried out the ripple carry adder is having less area, less power and more delay. Carry select adder is having less delay among four adders, but it has very high power and very large area. Bit serial adder and carry look adder is having delay in between ripple carry adder and carry select adder, but the area and power more than the ripple carry adder. As the work mainly deals with the layout of the design, the area is main criteria, and power also comes in to picture. For the

above constraints ripple carry adder is best suitable, but it has more delay. The area and power are very high for other adders and delay variation is not that much concern, so in most of the DSP applications ripple carry adder preferred than any other adders. From the Figure 3 we can observe that bit serial adder has more area when compared with the other adders, and ripple carry adder has least area among the four adders, carry look adder and carry select adder stand in between them. For lower number of bits the area among the four adders differentiate

much, as the number of bits increases the area variation is more.

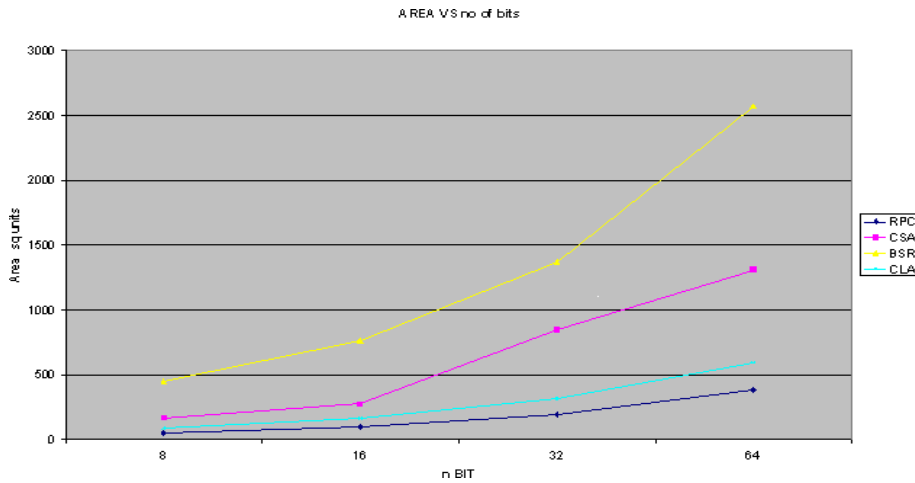


Figure 3 Area Comparison of Adders [14]

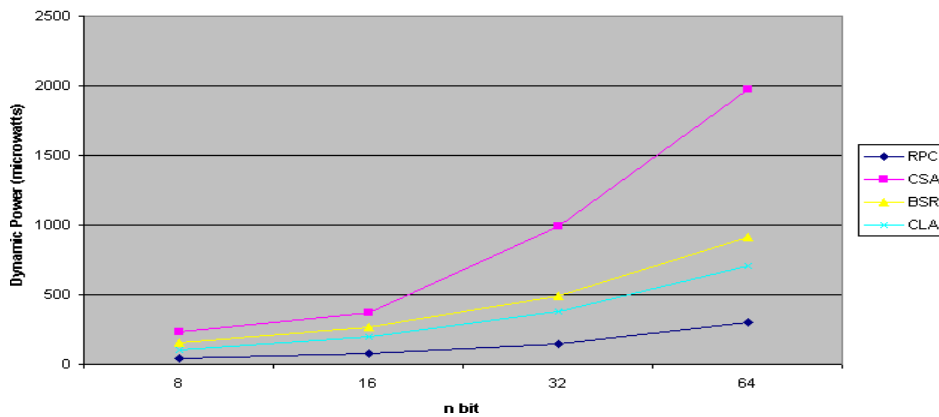


Figure 4 Power Comparisons of Adders [14]

The Figure 4 power variation is also same as the area variation, the power is small for ripple carry adder, and it is more for carry select adder. Similarly as the area the number of bits increases the power also increases. Delay is the main important consideration in many designs; especially in the DSP applications speed is the main criteria. Among the four adders carry select adder is having very less delay when compared to the other adders. RCA is having more delay as the output of present stage depends up on the output of the previous stage.

2.2 Summary of multipliers

As the work is proposed for high speed DSP applications, so every block of the design should perform the operation very fast. The multiplier is one of the major blocks of the design; the comparisons of several multiplier architecture are verified for best speed. From the comparison table of multiplier we can observe Hitachi and

Inoue multiplier are have high-speed performance when compared with the other multiplier. But the implementation of these two multipliers is very difficult as the complexity of their hardware structure is very high, as a result the cost of implementing the design is also high. The layout of these two multipliers is very irregular, and the power consumption is also very high. Next preferable multiplier with respect speed is Wallace tree multiplier and Dadda multiplier. The architecture of the two multiplier is almost same with little difference. For lower bits up to 16 bits the Dadda multiplier is slightly faster than Wallace tree multiplier and for higher bits greater than 16 bit both the multiplier has same delay. For the lower bits the Dadda multiplier is faster, but the power consumption is higher than the Wallace tree multiplier for the same number of bits. So in general Wallace tree multiplier is generally preferable for DSP applications as it can perform fast, have medium complexity.

Different types of adders and multipliers selecting the best of them is complicated as each of them as its own features, advantages and disadvantages. Now let us analyse and compare each of them with respect power, area and delay. The Table 1 compares multiplier

architectures with respect to area, power, speed, layout, cost and complexity of the multiplier. Selection of particular multiplier depends up on the application, as we now mainly deals with the speed and area Wallace tree multiplier can be chosen for the implementation of the MAC.

Table 1 Comparison of Multipliers

TYPE	SPEED	COMPLEXITY	LAYOUT	AREA	POWER	COST
Array	Low	Simple	Regular	Smallest	High	Low
Carry save	Medium	Simple	Regular	Smaller	High	Average
Booths	High	Medium	Irregular	Medium	Medium	Medium
Wallace Tree	Higher	Medium	Irregular	Medium	Medium	Average
Dadda	Higher	Medium	Irregular	Medium	High	Average
Hitachi	Highest	More complex	Medium	Largest	Very high	More
Inoue	Highest	More complex	Irregular	Largest	Very high	More

Selection of particular multiplier depends up on the application, as we now mainly deals with the speed and area Wallace tree multiplier can be chosen for the implementation of the MAC.

III. DESIGN OF MAC UNIT

In most of the DSP applications pipeline MAC is mostly used for many applications as its computation is very fast than normal parallel processing, the pipeline MAC consists of arrangement of MAC in a pipeline manner, the number of pipeline stages depends up on the application.

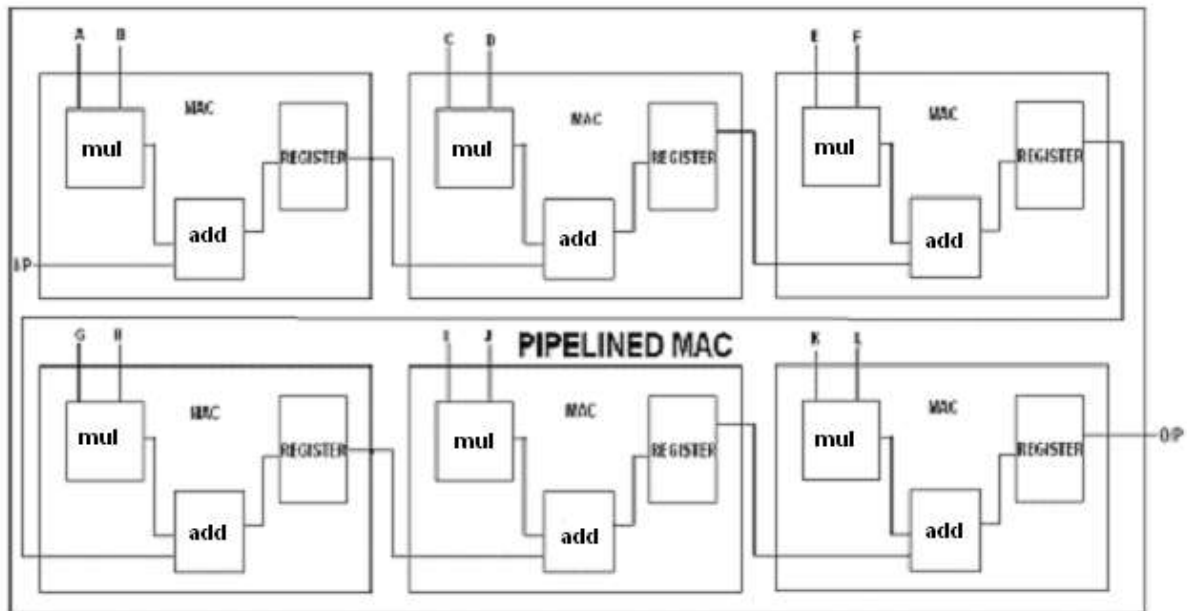


Figure 5 Simple Block Diagram of Pipelined MAC [6]

Figure 5 shows the block diagram of pipelined MAC, the main blocks pipeline MAC consists of Multiplier, Adder and Register. The

selection of multiplier and adder determines the speed, area power of the total design; we can say the multiplier is the main block which mainly

decides the speed of the design. Now let us go through the various architectures of multipliers, one bit full adders and adders.

IV. VLSI IMPLEMENTATION OF MAC UNIT

Figure 5 shows the schematic diagram for 8 x 8 Wallace tree multiplier. The multiplier totally consists of sixteen to give sixteen outputs. Each tree consists of full adder, half adder and gate as

the basic blocks. The carry of each full adder is given as the input next stage and sum is given as the input next full adder of same stage. The final sum of each tree gives us each output of the multiplier. The tree structure of the multiplier goes on increasing till eight stage and from eight stage it decreases, the first tree and the last tree has single and gate. In the schematic from Z0 to Z15 shows the outputs of each tree.

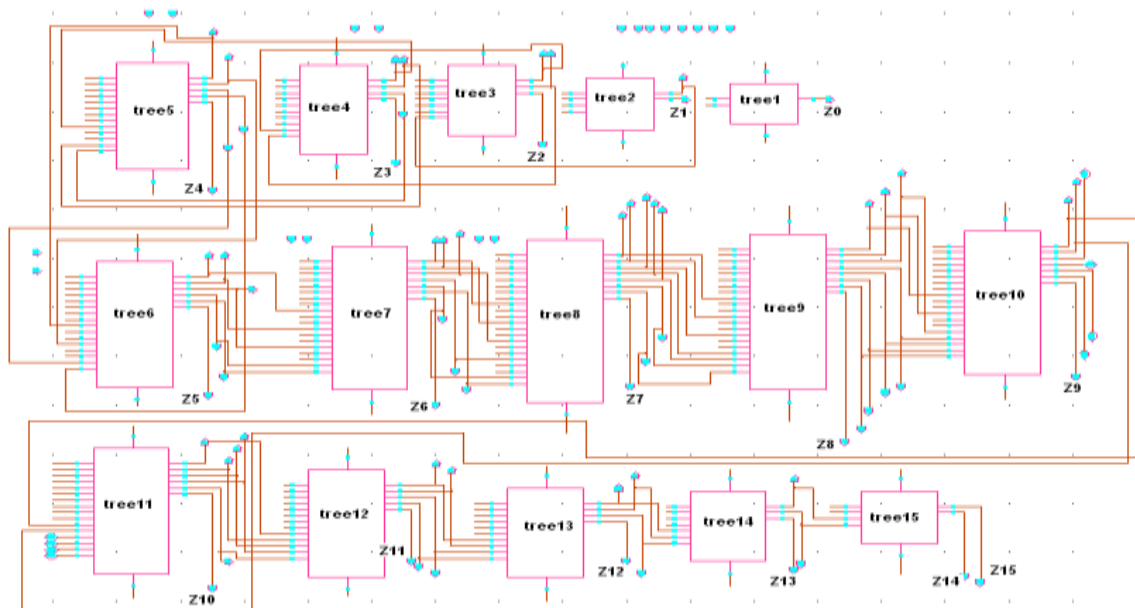


Figure 5 Schematic of Wallace Tree Multiplier

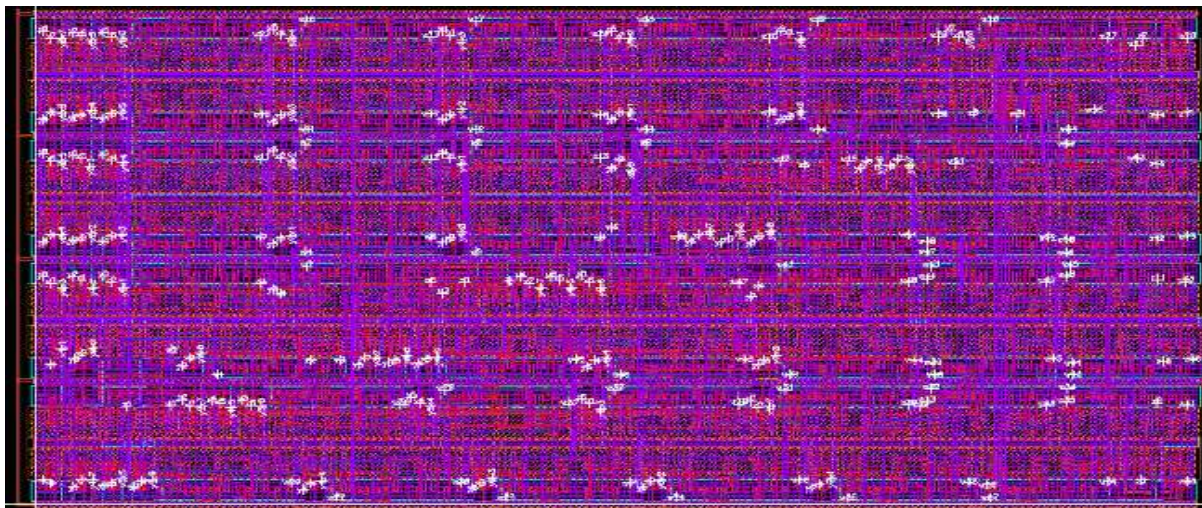


Figure 6 Layout of Wallace Tree Multiplier

Figure 6 shows the layout of the 8 x 8 Wallace tree multiplier. In drawing the layout of the multiplier, each tree of the multiplier is drawn separately and they are instantiated to form whole

multiplier. The arrangement of trees is very difficult task, as each multiplier has irregular structure they should be arranged in such way that the layout of the multiplier should be in perfect

shape. In process of arrangement of trees first tree , and then on the top of the first tree second tree and till seventh tree one above the other are arranged, and from the eight tree they arranged next to the placed trees which gives the perfect shape for the multiplier. In the arrangement, abutment of the trees is used to have common VDD and VSS. The

layout as continuous N-well which is more advantageous, three metal layers are used for the routing with metal one width .23, metal two width .28 and metal three width of .28. The layout is cleared from DRC and LVS.

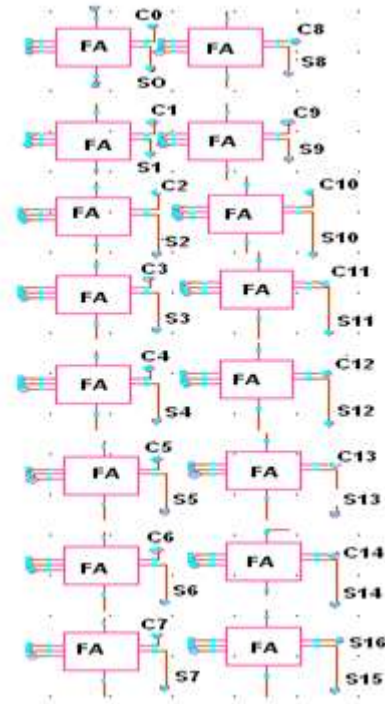
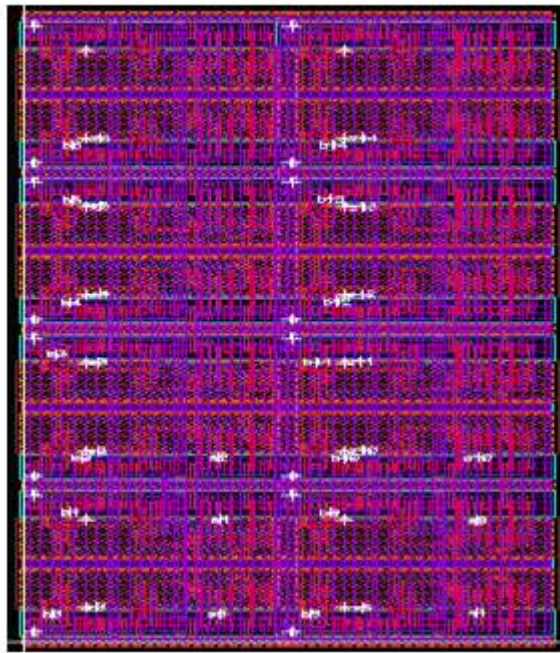


Figure 7 Layout and Schematic of 16 bit RCA Adder

Figure 7 shows the schematic and the layout of the RCA adder, the adder is for sixteen bit input, as the output of the multiplier is sixteen bit. The full adder are arranged in such way that the carry output of one adder is given as the input next adder., by this the delay may be increased, but the when compared to the other adder it has very less

area , regular structure, and less power dissipation. The delay of the RCA can be reduced by using static and dynamic full adder. Similarly in the layout of the adder the full adders are instantiated and where the output of one adder is given as input to the next by using metal two.

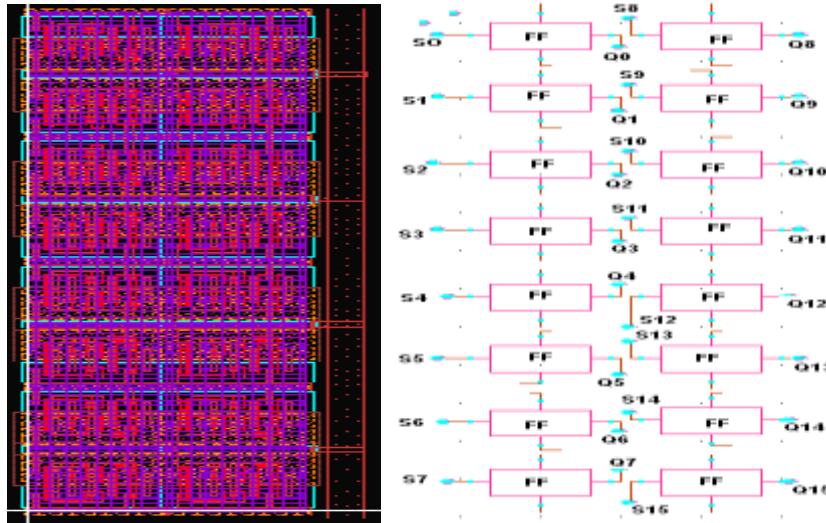


Figure 8 Layout and Schematic of 16 bit Register

Figure 8 shows the schematic and layout of sixteen bit register; Register is a group of flip flops, D flip-flops are used to form register. Sixteen flip flops are used one after the other. Similarly in

the layout D flip flops are used, two metal layers are used for routing.

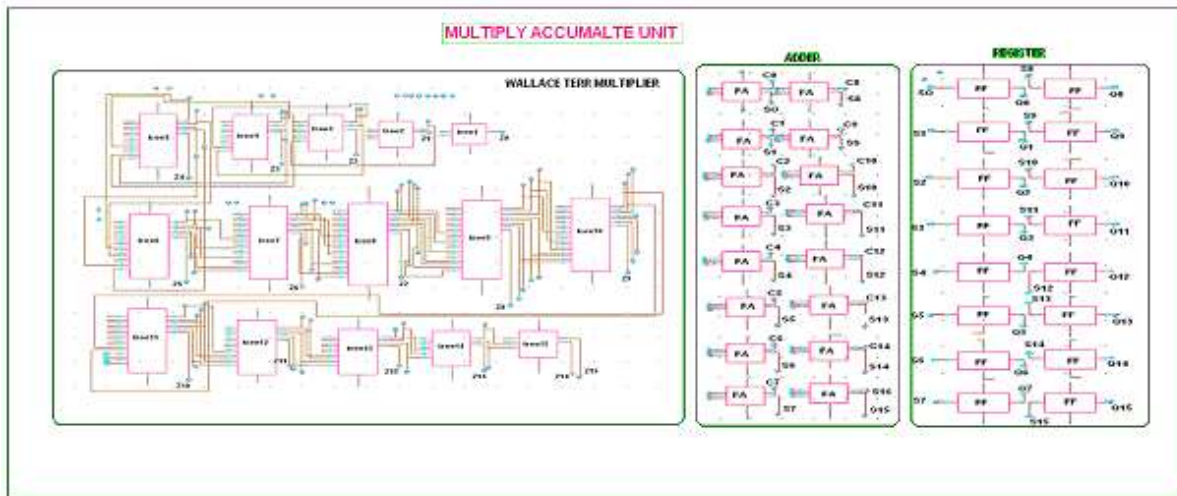


Figure 9 Schematic Of Multiply Accumulate Unit

Figure 9 shows the schematic arrangement of MAC unit which includes multiplier, adder and register. These three blocks are instantiated to form

MAC unit. The outputs of the multiplier are given as the inputs to the adder and the outputs from the adder are stored in the register.

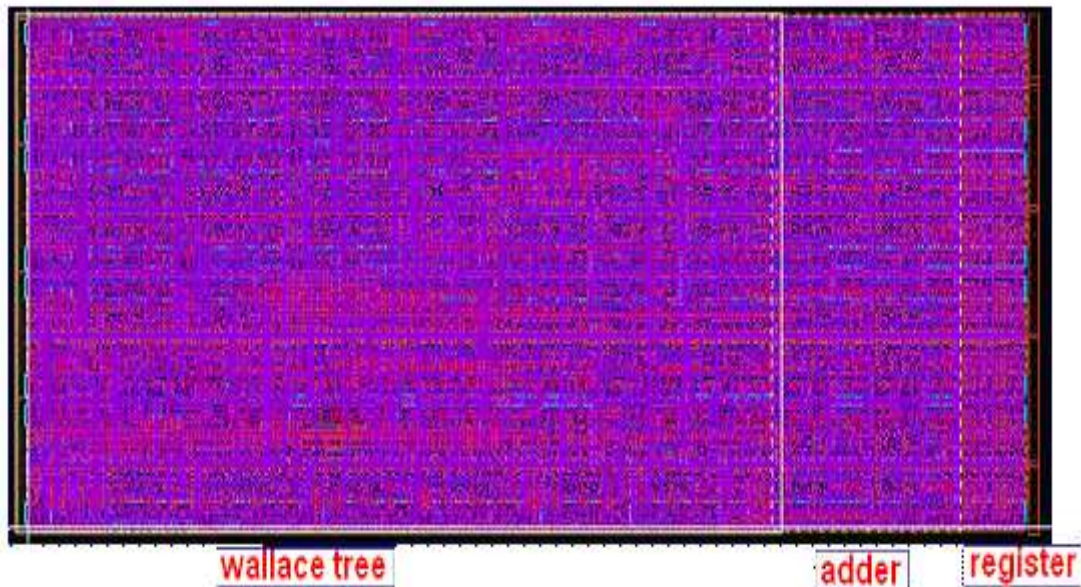


Figure 10 Layout Of Multiply Accumulate Unit

Figure 10 shows the layout of the multiply accumulate unit, similarly as the schematic the multiplier, adder and register are instantiated to form multiply accumulate unit. The outputs are routed as the inputs to the adder using metal three, similarly the adder outputs are routed to the accumulator using metal three. The blocks are not

flattened, only top level routing is used to connect the blocks. The area occupied by the MAC unit is $.52 \times 10 \text{mm}^2$. As the multiplier, adder and register have perfect shape, the MAC unit also comes in to perfect shape with VDD and VSS rails on the right and left of the block.

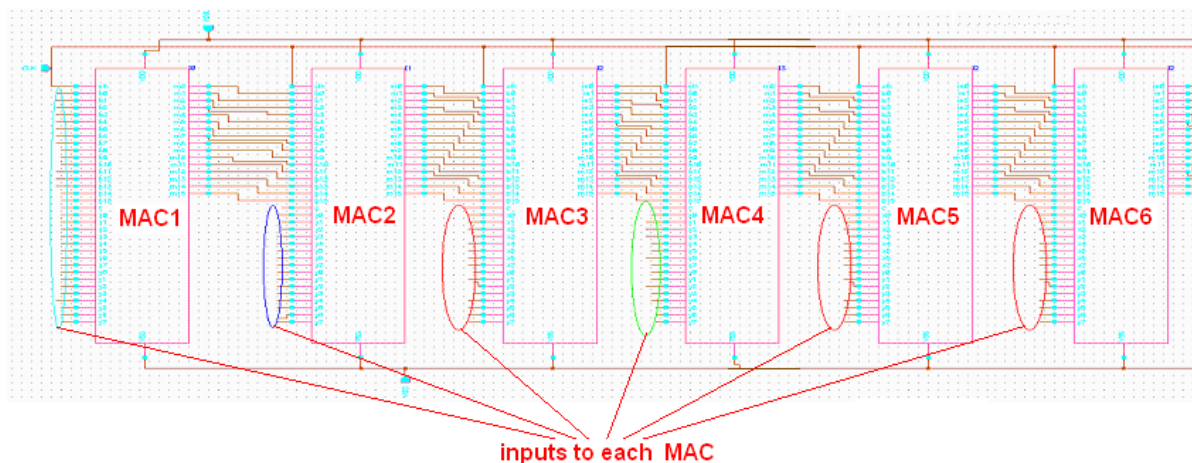


Figure 11 Schematic of Pipelined MAC

Figure 11 shows the schematic of the pipelined MAC unit, in the schematic six MAC are arranged one after the other where the output of MAC is given as the input next MAC adder. The inputs to each MAC are marked, the VDD and VSS are given at the top and bottom. Clock input to all the MAC is given as common. The final outputs

can be obtained from the sixth MAC unit. To find the performance at each stage of the MAC, each MAC is added in the schematic and layout is drawn and the area is found out. Pipeline MAC for six stages is shown in the schematic. Similarly the stages can be increased just instantiating by each MAC unit depending up on the application.

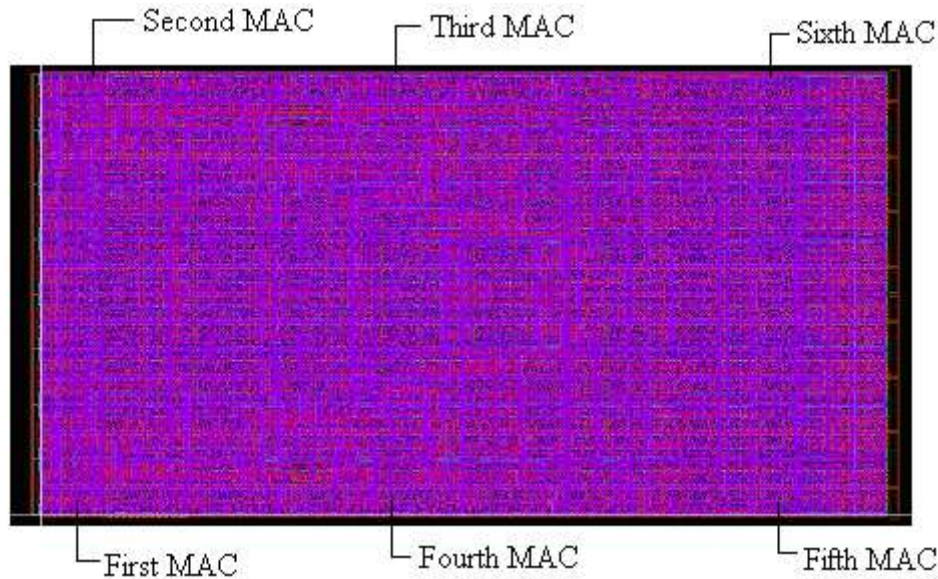


Figure 12 Layout of Pipelined MAC

Figure 12 shows the layout of the pipelined MAC for six stages, similarly as the schematic to find the performance at each stage, each MAC is instantiated. The arrangement of MAC unit to form pipeline MAC, the first MAC is arranged and the second MAC is arranged exactly at the top of the first MAC. The third MAC is arranged next to the second MAC and fourth MAC on the top of the second MAC unit, this was continued till six stages and it clearly marked in the

Figure 4.26. The routing from one MAC to other is very difficult; metal four has been used for routing. Much time has been taken for the routing of the pipelined stages. The area occupied by six stages of pipelined MAC is $1.05 \times 3.08 \text{ mm}^2$, the area occupied by fifth stage and sixth stages are same because as they are one above the other. Table 2 shows the performance summary of the whole design which specifies area, power and speed of the whole design at each stage.

Table 2 Performance summary of Pipelined MAC

TYPE	POWER DISSIPATION	DELAY	AREA
Full Adder	1.9 mw	800ps	$.3(y) * .72(x) \text{ mm}^2$
Wallace Tree Multiplier	5.66 mw	1.24ns	$.52 * .10 \text{ mm}^2$
Multiply Accumulate Unit	8.24 mw	2ns	$.52 * .148 \text{ mm}^2$
Pipelined MAC 2 ND Stage	16.38mw	4ns (2clocks)	$1.05 * 1.48 \text{ mm}^2$
Pipelined MAC 3 RD Stage	25.68mw	6ns (3clocks)	$1.05 * 2.56 \text{ mm}^2$
Pipelined MAC 4 th Stage	33.08mw	8ns (4clocks)	$1.05 * 2.56 \text{ mm}^2$
Pipelined MAC 5 th Stage	42.32mw	10ns (5clocks)	$1.05 * 3.08 \text{ mm}^2$
Pipelined MAC 6 th Stage	50.26 mw	12ns (6clocks)	$1.05 * 3.08 \text{ mm}^2$

The speed of the first MAC is 2ns which exactly one clock cycle, the speed of the second stage MAC is 4 ns which is two clock cycles, similarly the speed for six stage pipelines MAC is 12ns which means for clock cycle of 2ns we will get the output of the respective stage. The operating frequency of the whole design is 83.3MHz. The area of first MAC is $.52 * 1.48\text{mm}^2$ and the area of the second MAC is $1.05 * 1.48\text{mm}^2$. For the first and second MAC the area on the x axis is same as they are arranged one above the other. The area occupied by the fifth stage and sixth is same; finally the area occupied by the six stage of pipeline MAC is $1.05 * 3.08\text{mm}^2$. The power dissipation of the whole design is 50.26mw, as the number of stages increases the power dissipation also increases. The total number of transistor count for the design is about 3200.

V. CONCLUSION

This work presents design and VLSI implementation of pipelined MAC of high-speed DSP applications. DSP applications involve many additions and multiplications and finally accumulating them, the design, which can perform these operations, is MAC. MAC is a multiply accumulate unit which performs the addition and multiplication operation. Pipelined MAC is arrangement of MAC's in pipelined manner. The speed of the design depends up on the speed of the blocks inside the design. The main blocks in the design are adders and multipliers. The architecture of the pipeline MAC is explained, with all the internal blocks. The architecture consists of multiplier, adder as the basic blocks, first the architectures of several multipliers are considered and they are compared with respect to area, speed and power and complexity. The advantages, disadvantages and applications of each multiplier are clearly explained. Finally it was concluded the Wallace tree multiplier can be implemented for the design. As the one bit full adder is the block in the design of the MAC, different architectures are considered and they are explained with block diagrams. From the survey it was concluded that static and dynamic full adder has very less delay compared with the others. The advantaged and disadvantages of each adder are mentioned. The performance summary of the adders is tabulated which clearly gives that static and dynamic full adder has less delay. Architectures' of adders are taken and they explained with block diagrams, the area and power variation graphs are shown for the adders. Finally the performance of the design is tabulated with respect area, delay and power. The

power of the design is 50.26mw, the area is $1.05 * 3.08\text{mm}^2$ and the speed of the design is 12ns.

REFERENCES

- [1]. M.Tech. Credit Seminar Report, Electronics Systems Group, EE Dept, IIT Bombay, submitted November '02-2000, DSP Architectures for System Design", Vinay Savla (02307910), Supervisor: Prof A. N. Chandorkar.
- [2]. Kihak Shin, IkKyun Oh, Sang Min, BeomSeomRyu, Kie Young Lee and Tae Won Cho "A Multi-Level Approach to Low Power Mac Design" IEEE Trans. VLSI systems, vol 48, pp 361-763, 1999.
- [3]. Ichiro Kuroda, Eri Murata, KouheiNadehara, KazumasaSuzukiTomohisaAraitandAtsushi Okamuratt "A 16-bit Parallel Mac Architecture for a Multimedia RiscPirocessor"IEEE Trans. VLSI systems, vol. 83, no. 83, pp 103-112, 1995.
- [4]. Jae Sung Lee, Young SeopJeon, and Myung H. Sunwoo "Design of New Dsp instructions and Their Hardware Architecture for High-Speed FFT" IEEE Trans. VLSI systems, , pp 80-90, 2001.
- [5]. DusanSuvakovic, C. Andre, Salama "A Pipelined Multiply-Accumulate Unit Design for Energy Recovery DSP Systems" IEEE International Symposium on Circuits and Systems, May 28-31, 2000.
- [6]. Shyh-JyeJou, Chang-Yu Chen, En-Chung and Chau-Chin Su "A Pipeline Multiplier-Accumulator Using a High Speed Low-Power Static and Dynamic Full Adder" Journal of Solid State State Circuits, Vol 32, no- 1, January 2000.
- [7]. Sung-Mo Kang and Yusuf Leblebici, "CMOS Digital integrated circuits", Third Edition, Tata McGraw-Hill Publishing Company Limited, 2003.
- [8]. BerilSedaÇiftçi "Design and Realization of a High Speed 64 x 64 – bit Multiplier for Low Power Applications" Sabancı University Spring 2003.
- [9]. John Kim, Earl E. Swartzlander, "Improving the Recursive Multiplier" IEEE Trans. VLSI systems, vol-5, pp 2-5, 2000.
- [10]. G. Goto, et. Al., "A 54x54-b regularly structured tree multiplier", IEEE J. Solid-State Circuits, vol. 27, no. 9, Sept. 1992.
- [11]. S.Shah, A. J. AI-Khabb, D. AI-Khabb, "Comparison of 32-bit Multipliers for Various Performance Measures", The 12th International Conference on

- Microelectronics Tehran, Oct. 31- Nov.2, 2000
- [12]. Jan M.Rabaey, AnanthaChandrakasan and BorivojeNikolic, “Digital Integrated Circuits”, Second Edition, Prentice Hall Electronics and VLSI series, 2004.
- [13]. Pascal C.H. Meier, Rob A. Rutenbar and Richard carley, “Exploring multiplier architecture and layout for low power”, IEEE Custom Integrated circuits Conference, 1996.
- [14]. Gopivenugopal, Krishna Sumanth, Chagarlamudi Amit Kathuria, Advanced Adder Architectures, www.unix-ecs.umass.edu .