Design and Analysis of a novel 8X8 bit signed/unsigned synchronous MAC architecture using clock gating scheme for fixed-point arithmetic

Rajkumar Sarma
VLSI Design, SEEE,
Lovely Professional University
Phagwara, Punjab, India
rajkumar.sarma86@gmail.com

Sandeep Dhariwal
VLSI Design, SEEE,
Lovely Professional University
Phagwara, Punjab, India
dhariwal.vlsi@gmail.com

Shruti Jain
ECE Department,
JUIT, Waknaghat, Solan
Himachal Pradesh, India
jain.shruti15@gmail.com

***Abstract*— Nowadays low power designs play a crucial role in any of the electronic devices. Many of the researchers are involved in the design of low power designs that play a crucial role in the digital signal systems. Out of these, Multiply-Accumulate unit is the basic block in digital signal processing systems. In this paper various MAC architectures along with its various sub blocks such as adder and multiplier are reviewed thoroughly. Novel MAC architectures for Unsigned MAC, Unsigned synchronized MAC and Signed MAC are proposed and implemented using CMOS 90nm technology using Cadence Virtuoso. A detailed comparison with respect to power and area is also done for the proposed architectures.**

Keywords—Signed MAC; Unsigned MAC; Block Enable Technique;Virtuoso; Cadence; CMOS 90nm.

I. INTRODUCTION

The DSP systems are used in various applications such as image processing, speech encoding, audio mixing etc. In the above applications MAC unit plays a critical role of operations as the input signals need to be multiplied and then added with the previous result. The basic MAC unit consists of multiplier, adder and accumulator register. The arithmetic operations of MAC can be done for two different number systems, a) fixed point and b) floating point. In the fixed-point representation, there are signed and unsigned numbers that are to be multiplied and then added; but at the same time the fixed-point number system is not efficient enough for arithmetic operations of larger numbers. So, there is a requirement for the floating-point number system. The floating-point number system is the combination of Mantissa term and the exponent terms. So, in general the real numbers in floating point number system is represented as $N=M*B^E$ where M is mantissa, B is base, and E is exponent. So, while designing a MAC unit, all these design constraints of fixed as well as floating point numbers must be taken care of.

On the other hand, the function of the MAC unit is termed as:

$$Z = \sum_{i=0}^{n-1} a_i b_i$$

The above equation represents that a MAC unit performs multiplication of two numbers and add the result with the previously stored values. As discussed, the basic building block for the MAC unit are multiplier and adder. Therefore, to make the MAC block efficient, the multiplier and adder blocks of the MAC unit must be efficient in terms of power, speed and area. The basic building block of the MAC unit is represented in the figure 1 [13]. Two n-bit inputs are fetched and multiplied by the multiplier block which produces the 2N bit output which goes as input to the resister block. The register stores the data momentarily and sends the data as an input to the to the adder. The adder adds the output from the register block with the previously accumulated value that is stored in the accumulator register. Thus, the overall output is taken from the output of the accumulator register which is stored in the feedback register for next consecutive steps. Hence the architecture of the MAC unit consists of an N-bit multiplier, 2N bit register, (2N+1) bit adder and two (2N+1)-bit accumulator register (one for storing the output value and the other one is for reading the previous output).

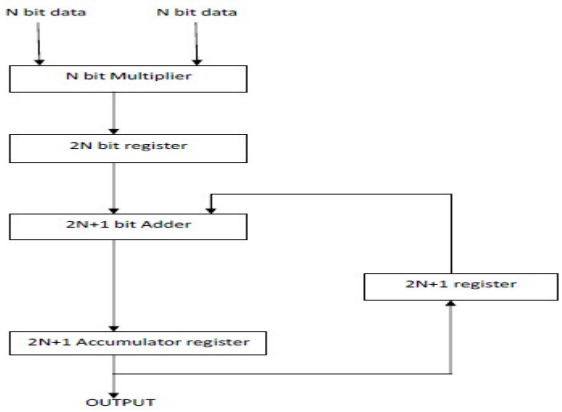

Figure1. Basic MAC Unit [13]

The remaining part of this paper is divided into the following sections: Section II describes the basic architecture of MAC unit. Section III describes the detailed review of various essential blocks individual blocks of MAC unit. Section IV describes the detailed review of various MAC units available in the literature and its comparative analysis.

II. BASIC BUILDING BLOCKS OF MAC

A. Multiplier

As we know that the processing elements mainly involve the multiplication of two numbers. So, there is a need of multiplier in such type of processing systems. Various fast and efficient multipliers are described in the literature. Array multiplier is a basic multiplier which follows the principle of product generation and addition. But when the total number of addition levels increases, this architecture becomes bulkier with higher PDP. Solution for this problem can be Wallace tree multiplier based on Wallace tree structure. in [11], the Wallace tree multiplier is designed which generates the product of two numbers

978-1-5386-6483-4/18/$31.00 ©2018 IEEE
DOI 10.1109/ICICS.2018.00092

using purely combinational logic, i.e., in one gating step. A rapid square-root process is also outlined in [11]. The figure for the same is shown in figure 2. However, in Wallace tree multiplier every partial product is added in a single direction from top to bottom, so the number of adder increases. To overcome this problem a rectangular styled Wallace tree multiplier [1] is proposed in which the partial products are divided into two groups and added in the opposite direction. The partial products in the first group are added downward, and the partial products in the second group are added upward. On the other hand, in the literature a phase mode parallel multiplier [10] is also proposed. The proposed multiplier has a Wallace-tree structure comprising trees of carry save adders for the addition of partial products. This structure has a regular layout; hence it is suitable for a pipeline scheme.

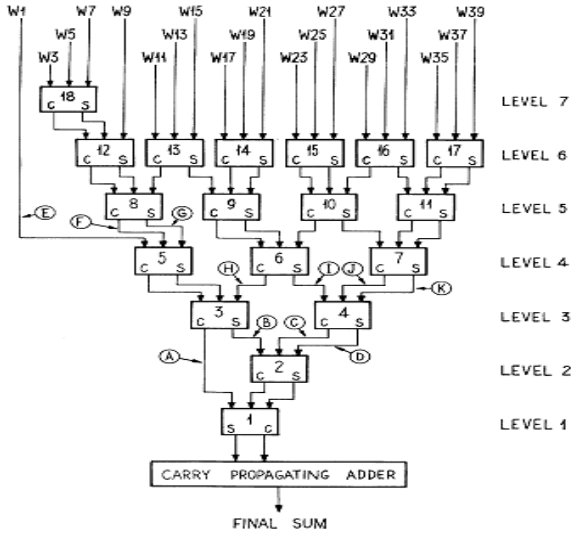

Figure 2: Wallace tree multiplier (addition of partial product)

A few architectures in the literature focused on the optimization of adder cell also. As adder is a basic cell in multiplier or divider, the optimization is mainly focused on adder part. A Carry-Select-Adder optimization technique [9] is proposed in the literature in which a carry-select-adder partitioning algorithm is used for high-performance Booth-encoded Wallace-tree multiplier. By taking into various data arrival times, a branch-and-bound algorithm is proposed and a generalized technique to partition an n-bit carry-select adder into a number of adder blocks is proposed such that the overall delay of the design can be minimized. In a separate approach an improved algorithm [17] for designing efficient modulo (2n + 1) multipliers had been proposed. By manipulating the Booth tables and by applying a simple correction term, the proposed multiplier is the most efficient among all the known modulo (2n + 1) multipliers and is almost as efficient as those for ordinary integer multiplication. On the other hand, in [12] a comparative analysis is done for designing multiplier using complementary MOS (CMOS) logic style, complementary pass-transistor (CPL) logic style and double-pass transistor (DPL) logic style. A single precision reversible floating-point multiplier is proposed in [19]. A 24-bit multiplier is proposed in this work by decomposing the whole 24 bits in three portions of 8 bit each.

B. Adder

An adder (also referred to as summer) is a logic circuit which adds two or more variables. Adders are unavoidable part of logic circuits as they are not only used for addition but also to multiply, calculate the addresses, increment operations, table indices etc. Most common adders operate on binary numbers although they can be constructed for BCD, excess -3 formats etc. In the literature there are various full adder architectures are proposed. In [20] a novel low power hybrid full adder is proposed using MOSIS 90nm Technology, which consumes very low power. The proposed design is compared with its conventional full adder which consists of 28 transistors. In a different approach, a hybrid 1-bit full adder is proposed which uses CMOS as well as TG logic styles [8]. The entire design was implemented in 90nm as well as 180nm technologies. At 1.8V supply voltage, the proposed design, offers very less power and moderately low delay. The proposed adder in [8] is shown in figure 3.

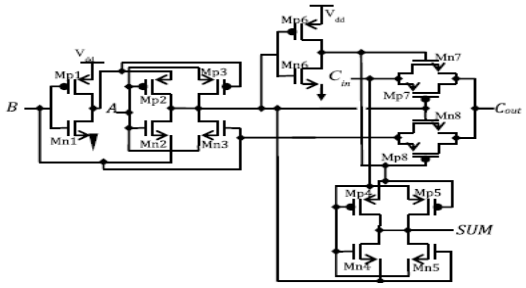

Figure 3: Full adder design in [8] which consumes very less power

III. PREVIOUS WORKS ON MAC UNIT

				a X	b w	c x	d y	z
0	az	az	cx&by	dw	cy	dy		dz
0	0	ax	cw^ay	az	dx	cz		0
0	0	cw&ay	by	cx^by	bz	0		0
P_7	P_6	P_5	P_4	P_3	P_2	P_1		P_0

Figure 4: Partial product addition matrix in [3]

In [2], Abdelgawad A. et.al, proposes 8-bit, 16-bit and 32-bit MAC and implemented on Xilinx ISE and on FPGA board. The design shows improvement in area and power. 4:2 compressor circuits are used to make the multiplier circuit faster. An 8-bit MAC unit is proposed using Cadence Virtuoso 180nm Technology in [14]. Various adder/multiplier circuits are also compared and implemented for the said MAC architecture. Deepak S. et. al in [3] had proposed a multiplier, where the terms are rearranged to reduce the "total number of partial product

by 25%", is shown in the figure 4. The proposed multiplier is further used to propose a MAC architecture. Cadence NC Sim and RTL compiler are used for doing all these analyses.

In a different approach, Jagadeesh P. et.al [4] had proposed a multiplier circuit which uses modified Wallace tree multiplier and carry save adder. Further a MAC unit is designed for 64-bit input which operates at 217 MHz and consumes a total power dissipation of 177.732 mW. In [15], Abdelgawad A., had proposed an ASIC implementation of the 32-bit MAC.The proposed architecture had reduced the complexity of the hardware, therefore reduces the power consumption and decreases the delay which decrease the area by 5.5%, power by 9% and delay by 13% compared to the conventional MAC architecture. The block diagram of the proposed design is shown in figure 5. The simulation is done 180nm technology using HDL.

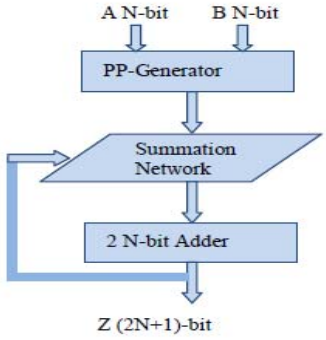

Figure 5: Proposed design in [15]

Francis, Teffi et.al [5] had proposed a novel MAC architecture based on modified Braun Multiplier with bypassing technique. Designs are implemented in 130nm CMOS technology. Transmission Gate, Double Pass Transistor Logic etc. are used to design the full adders in the circuit. Warrier, Rakesh et.al [6] had proposed a pipelined MAC architecture that consists of a 16-bit x16-bit multiplier. The implemented multiplier is based on Baugh-Wooley algorithm. The proposed architecture is found to be more efficient in terms of power consumption which is 30% to 80% less than the conventional MAC architectures. The implementation is done in 65nm CMOS using HDL in TSMC library. In Anitha R. et.al [16], Vedic multiplier and reversible logics are implemented. Using these finally a 32-bit MAC architecture has been designed as shown in figure 6. The implementations are done using Verilog HDL in Cadence RTL.

Moreover, as per Suvakovic, Dusan et. al [18], as adiabatic architecture offers less energy dissipation, a MAC unit based of adiabatic logic is proposed. The non-adiabatic dissipation is optimized by using less number of complex logic gates. This paper also discussed the comparative analysis of the proposed MAC with the existing designs. On the other hand, in Bing-jie XIA et.al. [7], a novel design is implemented in ModelSim in TSMC 90nm CMOS technology. Here 4-pipelined high-performance split multiply-accumulator (MAC) architecture was proposed. A partial product compression circuit based on interleaved adders and a hybrid "partial product reduction tree (PPRT)" are proposed to increase the speed of operation of the architecture. The advantage of the proposed MAC is that it can perform 1-way 32-bit or 4-way 16-bit signed/unsigned multiply or MAC operations and 2-way parallel multiply add operations. The figure 7 shows the proposed architecture.

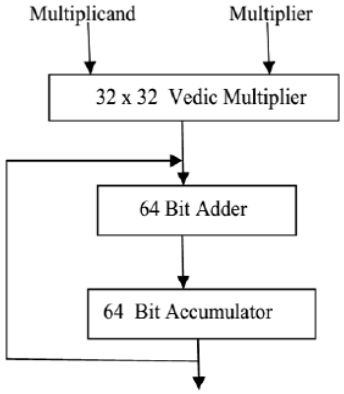

Figure 6: Proposed design in [16]

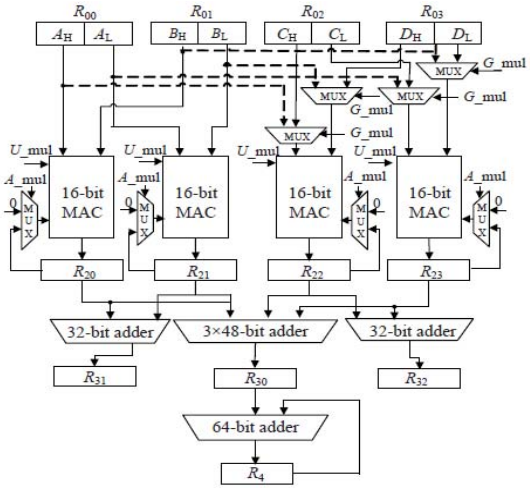

Figure 7: Proposed design in [7]

In [13], an 8-bit MAC unit is proposed using Cadence Virtuoso 180nm Technology. Various adder/multiplier circuits are compared and implemented for the MAC. Clock signal is not provided hence there is a synchronization issue. In the same year, Amaricai, Alexandru et.al [21] proposes a Floating Point multiply-add unit for low-precision formats. The multiplication and addition/subtraction operations are merged to achieve this architecture which is required in the mantissa data processing into a single operation. The architecture is implemented on FPGA board. The paper by Kataeva, Irina et.al [23] explains about RSFQ DS Processor, mainly used for removal of interferences from any signal. The author proposed MAC unit for floating point Multiplication-Addition. The MAC unit consists of three units i.e. parallel multiplier, combiner and accumulator as shown in the figure 8. The combiner performs summation

of sums and carries from M-MSB bits of the multiplier. The simulation is verified in VHDL.

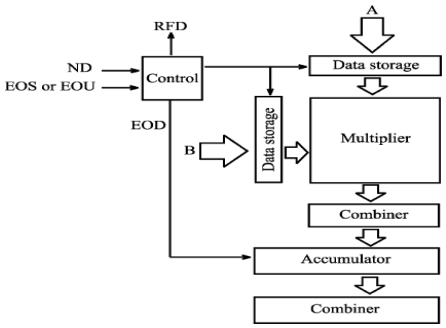

Figure 8: Explains the architecture proposed in [23]

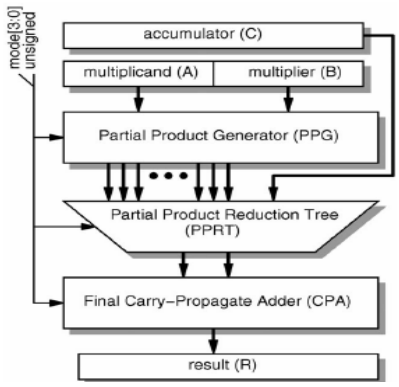

Figure 9: Architecture of Danysh, Albert et.al [26]

In [24], Bunyk, Paul I. et.al, describes a MAC unit specific for programmable Band pass filtering. This MAC unit can be clocked at a frequency of 20 GHz and can perform 2.5 billion MAC operations/second for 7-bit data. The simulation is verified in VHDL. Whereas in [25] Cardoso, Jaime S. et.al, minimization of Accumulator unit in MAC for block matching motion estimation is proposed. The FPGA implementation and mathematical models are discussed in this paper. A 64-bit fixed-point vector MAC architecture is proposed in Danysh, Albert et.al [26], which supports multiple precisions. The vector MAC can perform "one 64X64", "two 32X32", "four 16X16", or "eight 8X8" bit signed or unsigned multiplication and accumulation using the same hardware as the scalar 64-bit MAC architecture. The proposed design, as shown in the figure 9, is implemented using Verilog HDL in Synopsys tool. A power-delay efficient two-cycle MAC architecture is proposed by Hoang, Tung Thanh et.al [27]. The proposed architecture includes accumulation guard bits and saturation circuitry as well as it supports two's complement numbers. The implementation is done on VHDL and designed in 65nm 1.1V cell library.

IV. Proposed MAC Architectures

The proposed MAC architecture mainly focuses on the signed architecture based on the synchronized block enabling with effective pipelining technique. The block enabling is a power saver technique which activate a circuit momentarily and for rest of the time the circuit is inactive. Because of this simple phenomenon most of the energy/power can be saved. Secondly, the main reason for introducing synchronization is to avoid unnecessary loss of data. Due to non-availability of proper synchronization the data which are under process in the preceding block may be lost while transferring the same to the next block. Thirdly and most importantly the pipeline processing is the utmost requirement of the digital system as it increases the efficiency of the system tremendously.

As multiplier and the adder are the two core blocks, a detailed analysis has been done while choosing the appropriate circuits. While selecting the adder, the key importance is given to the power efficient design. To address the issue of power dissipation, the circuit proposed by [8] is used in the proposed MAC design. On the other way around, as per discussion on the Section I, it is found that even though the array multiplier is found to be the simplest multiplier algorithm, it produces very high delay in comparison with Wallace tree multiplier. On the other hand, rectangular styled Wallace tree multiplier is better option as it divides the partial products into two groups and hence it is faster than the conventional Wallace tree multiplier [1] (a detailed comparison for 4-bit operation is shown in Table I as per [22]). Therefore, Wallace tree multiplier is chosen for the proposed design.

The proposed MAC architecture is designed for 8-bit operation and divided into following three architectural designs:

A. Proposed Unsigned MAC architecture (UMAC)

The unsigned architecture is nothing but the traditional architecture of MAC. As discussed earlier, for n-bit inputs, it consists of a multiplier of size n-bit, a (2n+1)-bit adder and a register or accumulator of size n and (2n+1) bit. The figure 10 shows the detailed block diagram.

TABLE I. COMPARISON OF MULTIPLIER TYPE IN DIFFERENT LOGIC STYLES

Multiplier Type	Logic Style	Delay (ns)	Power (µW)	PDP (fJ)	No. of Transistor
Array	CMOS	8.3	10.73	89.06	384
	CPL	4.337	24.7	131.82	368
	DPL	4.667	19.72	92.03	448
Tree	CMOS	4.247	10.68	45.35	384
	CPL	4.105	23.61	125.25	368
	DPL	4.526	19.87	89.93	448

The key disadvantage of this architecture is non-availability of synchronized mechanism because of which the appropriateness of this architecture is doubtful. It creates the accurate results, but it is very difficult to verify. Moreover, the static power consumption for the UMAC architecture is very high because of the non-availability of block enabling technique. As the architecture doesn't require any synchronized mechanism, only the registers are designed with clock pulses making the multiplier and adder blocks active throughout the period of simulation. Hence the rise in static power consumption. The output waveform of the UMAC architecture is shown in figure 11.

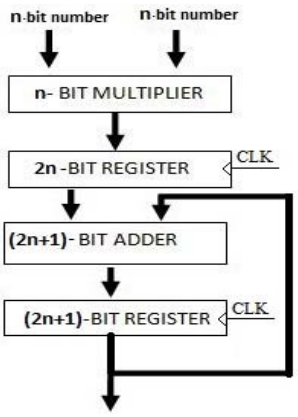

Figure 10: Proposed architecture of UMAC

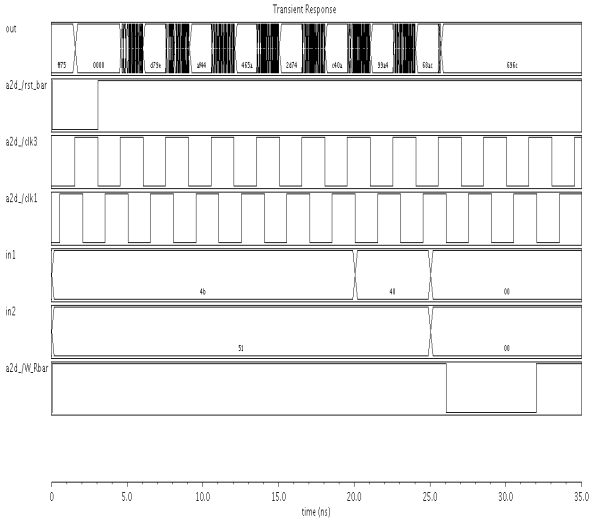

Figure 11: output waveform of the proposed UMAC architecture

B. *Proposed Unsigned synchronized MAC architecture (USMAC)*

The addon to the USMAC architecture is that individual blocks of the design are connected to the clock pulse with a Pulse Edge Detection (PED) block and a Latch block to detect the edges of the clock and to temporarily store the processed data respectively (as shown in the figure 12). Because of the small modification in the overall architecture, it is now possible to read and validate all the results of this architecture. Moreover, the static power consumption of USMAC is also very low in comparison to the UMAC architecture as each block of the USMAC architecture is synchronized with proper clock pulse. On the other hand, pipelining mechanism is also introduced for proper latching of data. For regulating pipeline mechanism, the clock signal to the individual block is provided with proper delay as an improper latching of data may develop undesired result at the final output. Therefore, a gap of 500 Pico second is kept between the 1st and the 2nd block and so on. The block diagram and the output waveform are shown the figure 13 and 14 respectively.

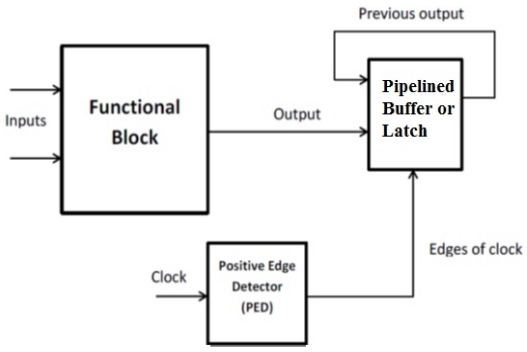

Figure 12: Block enabling architecture

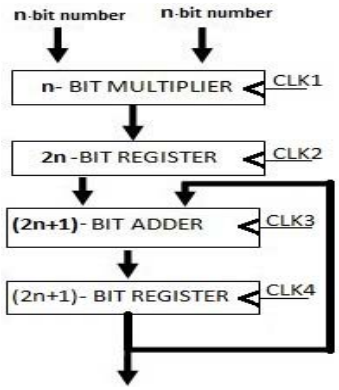

Figure 13: Block diagram of USMAC architecture

C. *Proposed Signed MAC architecture (SMAC)*

The MAC operation for positive as well as negative number is the taken care off in the SMAC architecture. For choosing the positive and negative number, multiplexers are used in this architecture. For negative number multiplication, the negative values are represented in 2's complement format. As mentioned in the previous sub-section, a gap of 500 Pico second is maintained between each block for proper latching of data. The block diagram and output waveform of the signed architecture are shown in the figure 15 and 16.

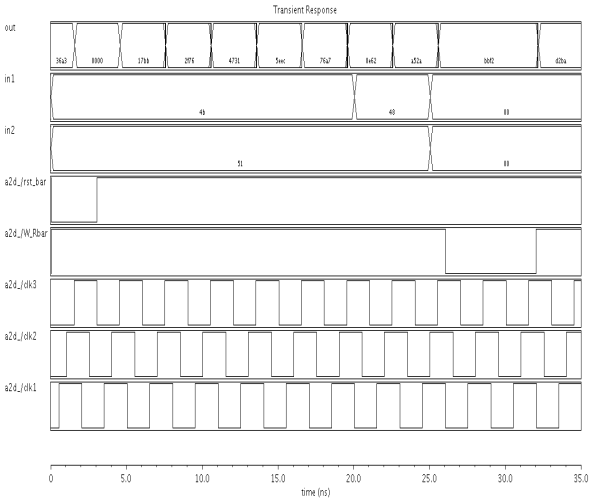

Figure 14: Output waveform of the USMAC architecture

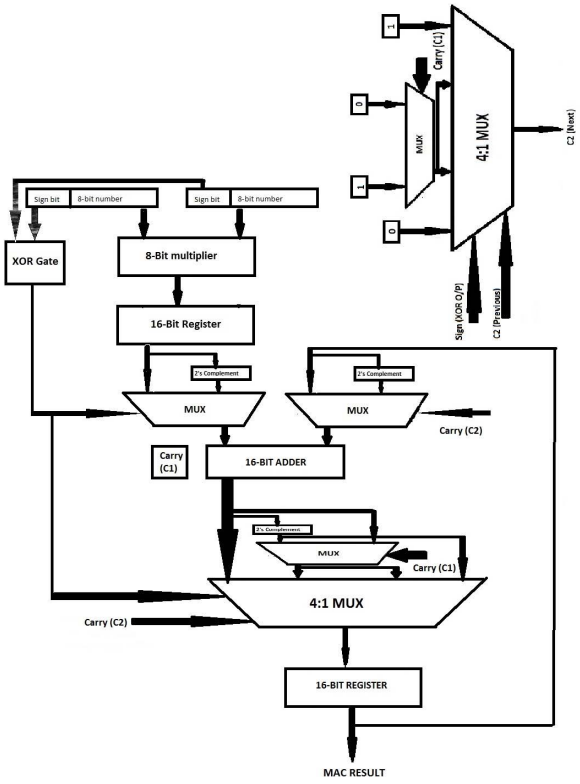

Figure 15: Block diagram of SMAC architecture

V. Result Discussion

The proposed MAC architectures are implemented at the cadence virtuoso 90nm technology. The power consumption of the implemented designs is calculated using Cadence Spectra Tool. Table II shows the detailed report of the static power, average power and area, whereas figure 17 shows the graphical analysis. The static power is evaluated for 2V supply voltage whereas the average power in measured for a simulation period of 20nS and at a frequency of 333.33 MHz. On the other hand, the Area is measured in terms of total number of transistors.

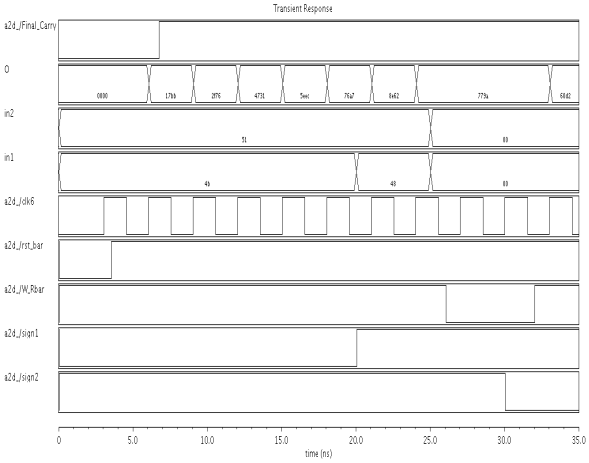

Figure 16: Output waveform of the SMAC architecture

TABLE II. COMPARISON OF PROPOSED MAC ARCHITECTURES

Architecture	Static Power in µW (For Vdd=2V)	Average Power in µW (For Vdd=2V and Simulation Period=20ns)	Area (Total number of Transistors)
UMAC	3072	2253	4556
USMAC	758	2905	5744
SMAC	1721	7317	10928

As discussed earlier, the static power consumption for UMAC architecture is the highest as block enabling is not used in this architecture. On the other hand, as the area of the SMAC architecture is approximately two times larger (in terms of number of transistors) than the USMAC architecture, the static and therefore the average power consumption of SMAC architecture is higher than the USMAC architecture.

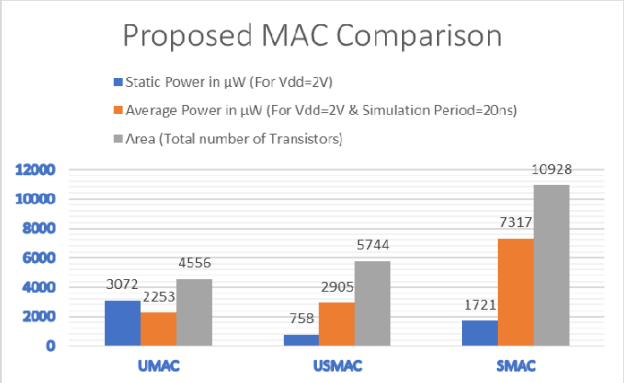

Figure 17: Graphical Analysis of the proposed MAC architectures

VI. Conclusion and Future Work

As mentioned in the previous section, designing MAC architecture with the help of synchronous as well as pipeline mechanism can help the circuit to produce the result appropriately with a very low loss of consumed power. The proposed work can be further extended for signed floating point operation along with pipeline mechanism.

Reference

[1] Niichi Itoh, Yuka Naemura, Hiroshi Makino, Yasunobu Nakase, Tsutomu Yoshihara, and Yasutaka Horiba, “A 600-MHz 54 54-bit Multiplier with Rectangular-Styled Wallace Tree”, IEEE Journal of Solid-State Circuits, vol. 36, no. 2, pp. 249-257, February 2001.

[2] A. Abdelgawad and M. Bayoumi, "High Speed and Area-Efficient Multiply Accumulate (MAC) Unit for Digital Signal Prossing Applications," in IEEE International Symposium on Circuits and Systems, New Orleans, LA, 2007.

[3] D. S and B. J. Kailath, "Optimized MAC Unit Design", IEEE Conference, 2012.

[4] P.Jagadees, S.Ravi and K. H. Mallikarjun, "Design ofHigh Performance 64 bit MAC Unit," in International Conference on Circuits, Power and Computing Technologies, 2013.

[5] T. Francis, T. Joseph and J. K. Antony, "Modified Mac Unit for Low Power High Speed DSP Application Using Multipler with Bypassing Technique and Optimized Adders," in 4th ICCCNT - 2013, Tiruchengode, India, 2013.

[6] R. Warrier, C. Vun and W. Zhang, "A Low-Power Pipelined MAC Architecture using Baugh-Wooley based Multiplier," in IEEE 3rd Global Conference on Consumer Electronics (GCCE), 2014.

[7] B.J. Xia, P. Liu and Q.D. Yao, "New method for high performance multiply-accumulator design," Journal of Zhejiang University Science, vol. 10, no. 7, pp. 1067-1074, 2009.

[8] P. Bhattacharyya, B. Kundu, S. Ghosh, V. Kumar and A. Dandapat, "Performance Analysis of a Low-Power High-Speed Hybrid 1-bit Full Adder Circuit," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, pp. 1-8, 2014.

[9] M. Liao, C. Su, C. Chang and A. C. Wu, "A Carry-Select-Adder Optimization Technique for High-Performance Booth-Encoded Wallace-Tree Multipliers," in IEEE International Symposium on Circuits and Systems, ISCAS 2002. , 2002.

[10] Takeshi Onomi, Kiyoshi Yanagisawa, Masashi Seki, and Koji Nakaj ima, “P hase-Mode P ipelined Parallel Multiplier”, IEEE Transactions on Applied Superconductivity, vol. 2, no. 1, pp. 541-544, March 2001.

[11] C. S. Wallace, "A Suggestion for a Fast Multiplier," IEEE Transactions on Electronic Computers, pp. 14-17, 1964.

[12] A. K. Singh, B. P. De and S. Maity, "Design and Comparison of Multipliers Using Different Logic Styles," International Journal of Soft Computing and Engineering (IJSCE), vol. 2, no. 2, pp. 374-379, 2012.

[13] S. S and S. Y. Kulkarni, "VLSI Design and Implementation of Low Power MAC Unit with Block Enabling Technique," European Journal of Scientific Research, vol. 30, no. 4, pp. 620-630, 2009.

[14] S. S, C. P. Raj and S.Y.Kulkarni, "Design and VLSI Implementation of Pipelined Multiply Accumulate Unit," in Second International Conference on Emerging Trends in Engineering and Technology, 2009.

[15] A. Abdelgawad, "Low Power Multiply Accumulate Unit (MAC) for Future Wireless Sensor Networks," in IEEE conference, 2013.

[16] R. Anitha, N. Deshmukh, S. K. Sahoo, S. Prabhakar and Karthikeyan, "A 32 BIT MAC Unit Design Using Vedic Multiplier and Reversible Logic Gate," in 2015 International Conference on Circuit, Power and Computing Technologies [ICCPCT], 2015.

[17] L.A. Sousa, “Algorithm for modulo (2n+1) multiplication”, Electronlcs Leters, Vol. 39, No. 9, pp. 752-754, 1st May 2003.

[18] D. Suvakovic and C. A. T. Salama, "Energy Efficient Adiabatic Multiplier-Accumulator Design," Journal of VLSI Signal Processing, vol. 33, pp. 83-103, 2003.

[19] Michael Nachtigal and Himanshu Thapliyal, “Design of a Reversible Single Precision Floating Point Multiplier Based on Operand Decomposition”, IEEE international conference on Nanotechnology joint symposium with Nano Korea, Korea, 2010.

[20] M.Jayaprakash, Dr.A.Shanmugam and M.Peer Mohamed, “Design And Analysis Of Low Power Hybrid Adder”, Journal of Theoretical and Applied Information Technology, Vol. 58 No.3, pp. 618-622, 2013.

[21] A. Amaricai, O. Boncalo and C.-E. Gavriliu, "Low-precision DSP-based floating-point multiply-add fused for Field Programmable Gate Arrays," The Institution of Engineering and Technology, vol. 8, no. 4, pp. 187-197, 2014.

[22] Amarveer Kaur Dhindsa and Rajkumar Sarma, “Pipelined and Clock Gated Mac Architecture Design And Implementation”, Far East Journal of Electronics and Communications, Special Issue IV, 2016

[23] I. Kataeva, H. Zhao, H. Engseth, E. Tolkacheva and A. Kidiyarova-Shevchenko, "RSFQ Digital Signal Processor for Interference Cancellation," IEEE Transactions on Applied Superconductivity, vol. 15, no. 2, pp. 405-410, 2005.

[24] P. I. Bunyk, Q. P. Herr and M. W. Johnson, "Demonstration of Multiply-Accumulate Unit for Programmable Band-Pass ADC," IEEE Transactions on Applied Superconductivity, vol. 15, no. 2, pp. 392-395, 2005.

[25] J. S. Cardoso and Luís Corte-Real, "Accumulator Size Minimization for a Fast Cumulant-Based Motion Estimator," IEEE Transactions on Circuits And Systems For Video Technology, vol. 15, no. 12, pp. 1660-1664, 2005.

[26] A. Danysh and D. Tan, "Architecture and Implementation of a Vector/SIMD Multiply-Accumulate Unit," IEEE Transactions on Computers, vol. 54, no. 3, pp. 284-293, 2005.

[27] T. T. Hoang, M. Själander and P. Larsson-Edefors, "A High-Speed, Energy-Efficient Two-Cycle Multiply-Accumulate (MAC) Architecture and Its Application to a Double-Throughput MAC Unit," IEEE Transactions on Circuits And Systems—I: Regular Papers, vol. 57, no. 12, pp. 3073-3081, 2010.