International Journal of Engineering & Technology, 8 (1) (2019) 1-7

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET
doi: 15803
Research paper

Low power multiply accumulate unit based on modified hybrid Han Carlson adder

Rakesh S [1, 2] *, K. S. Vijula Grace [1]

[1] *Department of ECE, Noorul Islam Centre for Higher Education, Thuckalay, Tamil Nadu, India*
[2] *Department of ECE, Mangalam College of Engineering, Ettumanoor, Kerala, India*
**Corresponding author E-mail: raksarian@yahoo.com*

Abstract

The aim of this work is to propose a modified architecture for implementing Multiply Accumulate (MAC) unit which can be used to build a low power Digital Signal Processor (DSP). The method improves the power consumption as well as the power-delay product. The proposed method uses hybrid parallel prefix adder in the multiplier stage and in the adder stage. In the architecture, a Hybrid Han Carlson adder with modified pre-processing and post-processing stages (HHCA_MPPS) is used. The units are designed using Verilog Hardware Description Language (HDL) and simulated and synthesized for Artix-7 series Field Programmable Gate Array (FPGA) using Xilinx Vivado Design Suite 2015.2. The analysis showed that the proposed design has significant improvement in the power consumption and the figure of merit (power-delay product). The MAC unit employing modified hybrid Han Carlson adder gave a power saving of 11.57% and improvement of 9.21% in the power-delay product.

Keywords: *Digital Signal Processor; Field Programmable Gate Array; Hybrid Han Carlson Adder; Modified Inequality Detector; Multiply Accumulate Unit.*

1. Introduction

Real time digital signal processing (DSP) has a great demand for wide range of powerful applications. The computational and hardware complexity of real time digital signal processors largely rely on the efficiency of the multiplier units and adder units which are embedded within them. A typical multiplier block comprises of a series of AND gates to generate the partial products and an adder block to sum up the terms. The power consumption of these units has always been a crucial design concern for engineers. Many researchers have been working over the years to optimize the performance of multiplier and adder topologies.

The multiply-accumulate (MAC) unit is the main digital block used commonly in digital signal processors. It is an important component in many signal processing applications that require Fast Fourier Transform computations or Inverse Fast Fourier Transform computations. In this paper we propose hybrid Han Carlson adder with modified pre-processing and post-processing (MPPS) stages. The basic MAC architecture consists of a multiplier, an adder and an accumulator register organized as in Fig. 1. The inputs are given to the multiplier, and the result is added with the previous result stored in the accumulator register, leading to the accumulation process. Multipliers are typically a combination of a partial-product generation unit and a carry propagate adder to add the partial products to find the final sum.

Copyright © 2019 Rakesh S, K. S. Vijula Grace. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

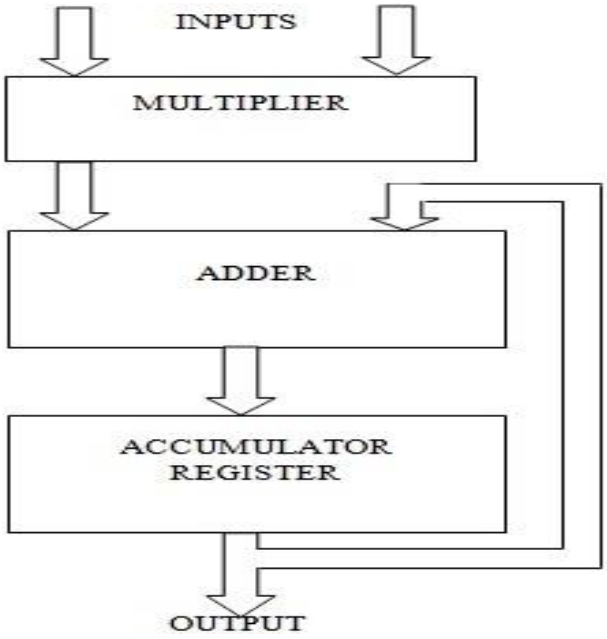


Fig. 1: General Architecture of MAC Unit.

The remaining of the paper is organized as follows. In Section 2, the literature review has been discussed. Section 3 describes the methodology used in the proposed system. The simulation results are discussed in Section 4. The conclusion is given in Section 5.

2. Review of literature

There has been extensive research going on in the area of low power digital signal processing during the past two decades. Mohamed et. al. introduced a different floating point MAC structure compared to the conventional one. He combined the multiplier and adder block and formed a new block called multiplier-accumulator block and used Wallace tree for multiply accumulation process [1]. Tung et. al. proposed a two's complement MAC architecture which used a carry save adder as the final adder. The carry save adder contained 3:2 counters and it used shorter interconnects [2].

Ahish et. al. developed a MAC unit with a partial product reduction block and the multiplier was implemented using Brent Kung adder. The design achieved better area, delay and power performance compared to the conventional multiplier employing Booth algorithm [3]. Nithish et. al. incorporated a modified carry select adder at the multiplier stage of the MAC unit to improve the area. A digital filter is then designed using the MAC unit which achieved significant reduction in area and power [4]. Maroju et. al. introduced four different MAC architectures which employed carry save adder at the adder stage and four different multipliers. The four different multipliers used in the design were Array Multiplier, Ripple Carry Multiplier with row bypass technique, Wallace Tree Multiplier and Dadda Multiplier [5].

Narendra et. al. developed a partial product reduction stage using 29 compressors. Compressor architectures were also utilized in the carry propagation adder stage and accumulation stage [6]. Abdelgawad designed a merged architecture which combined the accumulation stage with the multiplier circuit. The speed and throughput of the MAC unit were increased and the area was decreased [7]. Suryasnata et. al. introduced a power efficient Vedic multiplier and realized using 45nm CMOS technology. The conventional 6T CMOS based AND gate design is replaced with 5T design. The design was based on transmission gate logic and pass transistor logic. The delay is considerably reduced in the proposed system as the number of transistors in the critical path is decreased [8]. Rakesh et. al. presented a comparison of different MAC architectures [9].

3. Proposed system

In the proposed system a modified hybrid Han Carlson adder (HHCA) is used for adding the partial products in the vedic multiplier and also at the adder stage. HHCA belongs to the category of hybrid parallel prefix adder. The modification in this adder is done at the pre-processing and the post-processing stages. The pre-processing and post-processing stages consist of an inequality detector which performs exclusive OR operation. The inequality detector is designed here using four transistor logic. Switch level modeling is used to design the logic. The logic used is similar to the Gate Diffusion Input (GDI) logic.

Hybrid Parallel Prefix adder is a combination of Kogge Stone adder (KSA) and Brent Kung adder (BKA). It takes the best feature out of both these adders ie high speed of KSA and low area of BKA. HHCA uses Brent Kung stage in all the stages except the middle level. In Han Carlson Adder (HCA) there are more number of Kogge Stone stages than the Brent Kung stages compared to HHCA topology. So the speed is more for HCA due to the large number of Kogge Stone stages used in the carry computation but HHCA has better power consumption and area due to less number of gates or nodes [10-11].

Hybrid Parallel Prefix adders consist of a Propagate bit and Generate bit block (BGP) in the pre-processing stage, Group Generate and Propagate block (GGP) and Group Generate (GG) block in the carry computation stage and finally an inequality detector in the post processing stage for sum calculation. BGP block generates the propagate bit

$$P_i = A_i \text{ xor } B_i \quad (1)$$

Where

P_i = i^{th} Propagate bit

A_i = i^{th} A input

B_i = i^{th} B input

xor = ‘xor’ operation in digital

And generate bit

$G_i = A_i$ and B_i (2)

Where

G_i = i^{th} Generate bit

A_i = i^{th} A input

B_i = i^{th} B input

And = ‘and’ operation in digital

Figure 2 shows that GGP block generates

$G = G_i + P_i.G_{previous}$ (3)

$P = P_i \cdot P_{previous}$ (4)

Where

G = Group Generate bit

P = Group Propagate bit

$G_{previous}$ = Generate bit from previous stage

$P_{previous}$ = Propagate bit from previous stage

+ = ‘or’ operation in digital

. = ‘and’ operation in digital

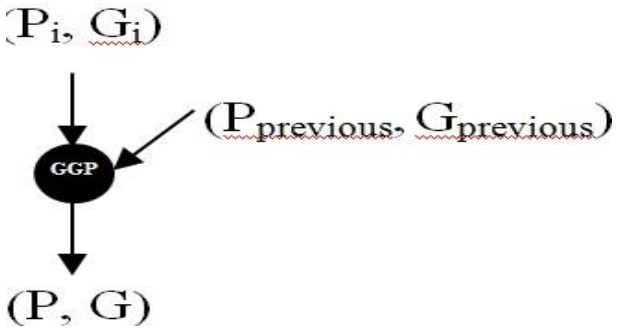


Fig. 2: Generation of Group Generate and Propagate Bits.

Figure 3 explains that GG block generates

$G = G_i + P_i. G_{previous}$ (5)

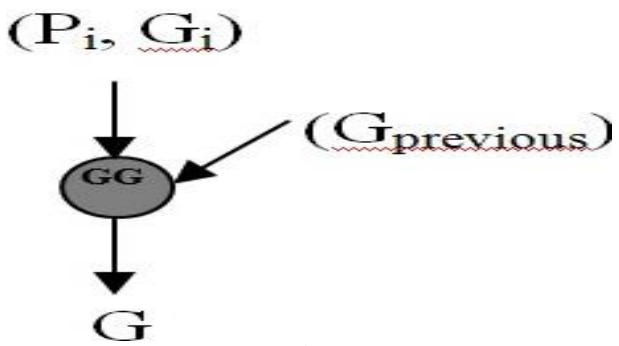


Fig. 3: Generation of Group Generate Bit.

Figure 4 clearly shows how the GGP block and GG blocks are used to compute the carry at different stages of HHCA. The expressions for the generation of carry C_0 to C_7 are also given.

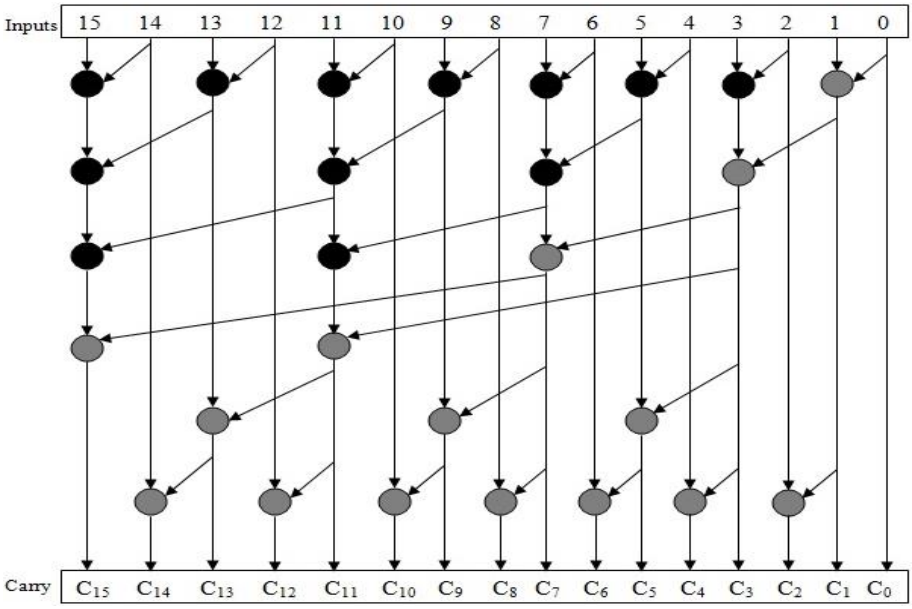


Fig. 4: Structure of a 16 Bit Hybrid Han Carlson Adder.

$C_0 = G_0 + P_0C_{in}$ (if $C_{in} = 1$, otherwise $C_0 = G_0$) (6)

$C_1 = G_1 + P_1\, C_0$ (7)

$C_2 = G_2 + P_2\, C_1$ (8)

$C_3 = G_3 + P_3\, G_2 + P_3\, P_2\, C_1 = G_{3:2} + P_{3:2}\, C_1$ (9)

$C_4 = G_4 + P_4\, C_3$ (10)

$C_5 = G_5 + P_5\, G_4 + P_5\, P_4\, C_3$ (11)

$C_5 = G_{5:4} + P_{5:4}C_3$ (12)

$C_6 = G_6 + P_6\, C_5$ (13)

$C_7 = G_7 + P_7\, G_6 + P_7\, P_6\, G_5 + P_7\, P_6\, P_5G_4 + P_7\, P_6\, P_5\, P_4\, C_3$ (14)

$C_7 = G_7 + P_7\, G_6 + P_7\, P_6\, (G_5 + P_5G_4) + P_7\, P_6\, P_5\, P_4\, C_3$ (15)

$C_7 = G_{7:6} + P_{7:6}\, G_{5:4} + P_{7:6}\, P_{5:4}\, C_3$ (16)

$C_7 = G_{7:4} + P_{7:4}\, C_3$ (17)

The carry bits are used to compute the final sum using the expression,

$Si = P_i$ xor C_{i-1} (18)

Thus the pre-processing stage uses an inequality detector to compute the "Propagate bit, P_i" from the input bits A_i and B_i. The post processing block also uses an inequality detector to generate the final "sum bit, S_i". Inequality detector performs exclusive OR operation. It

gives a high output when the inputs are unequal and hence the name inequality detector. The modified inequality detector is designed using the switch level modeling in Verilog HDL code. It is implemented using 4 MOS transistors out of which there are 2 PMOS transistors and 2 NMOS transistors. The conventional transistor implementation of the inequality detector needs 12 MOS transistors out of which 6 are NMOS transistors and 6 are PMOS transistors. Figure 5 shows the circuit diagram of the modified inequality detector. Thus there is a reduction in the number of transistors in the circuit which reduces the power consumption appreciably and improves the figure of merit (power - delay product) also.

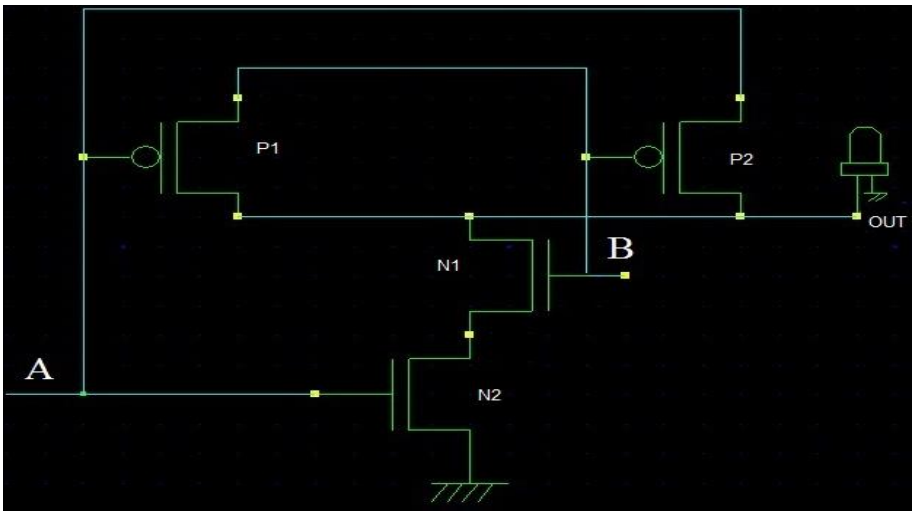


Fig. 5: Proposed Inequality Detector Design.

The working of the proposed inequality detector is explained hereafter. When both A and B are zero, both pmos will be on and both nmos will be off. Hence the input of the pmos is transferred to the output. So output is pulled to zero. When AB=01, pmos P1 will be on and P2 will be off. Similarly nmos N1 will be on and N2 will be off. Hence the source voltage of P1 which is logic 1, is moved to the output. When AB=10, pmos P1 will be off and P2 will be on. Similarly nmos N1 will be off and N2 will be on. Here the source voltage of P2 will reach the output and hence the output becomes 1. When both the inputs are high AB=11, both pmos will be off and both nmos will be on. This will pull down the output to ground potential. Low swing can be caused at the output when A=0 and B=0. In that case we expect the output to be zero but due to the poor high to low transition characteristics of the pmos, the output will be V_{Tp}[12].

The proposed architecture of the multiply accumulate unit that can be used in a Digital Signal Processor is given in Figure 6. The architectures employ a modified Vedic multiplier which uses a modified hybrid parallel prefix adder to sum up the partial products. At the adder stage also the modified hybrid parallel prefix adder is incorporated to add the result of multiplication and the previous result stored in the accumulator register. The accumulation process is actually taking place at the adder stage. The final result is again stored in the register. The accumulator register is basically a parallel in parallel out shift register.

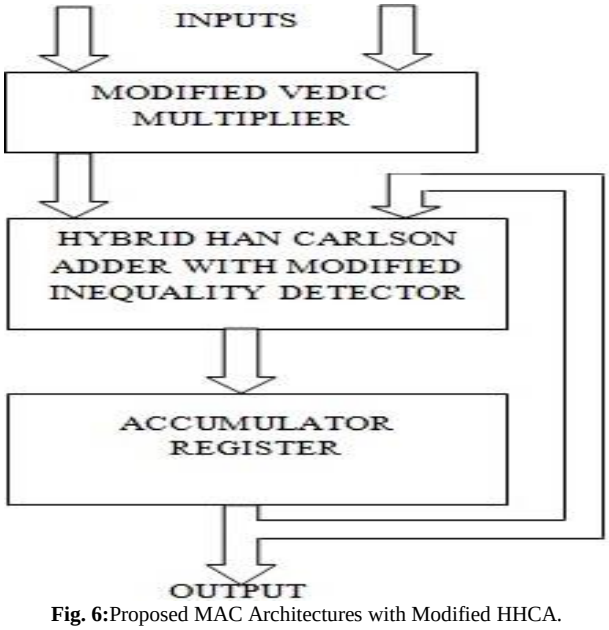


Fig. 6: Proposed MAC Architectures with Modified HHCA.

4. Results and discussion

The simulation results of the modified inequality detector and the proposed MAC unit architecture are presented in Figure 7 and 8 respectively. Also the power reports generated are shown in Figure 9 and 10. A comparison between the existing MAC design and the proposed design is given in Table 1. The comparison table clearly reveals that the proposed 16 bit MAC design using modified hybrid Han Carlson adder gave a power saving of 11.57% and improvement of 9.21% in the power-delay product.

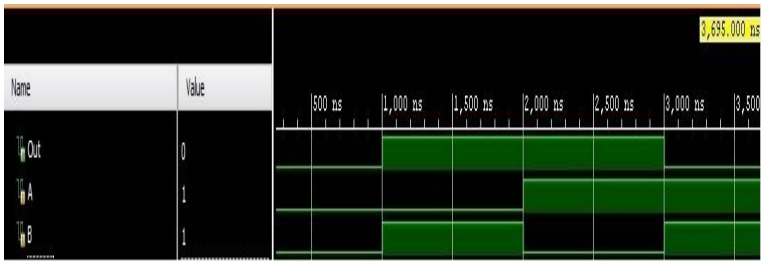


Fig. 7: Simulation Result of Modified Inequality Detector.

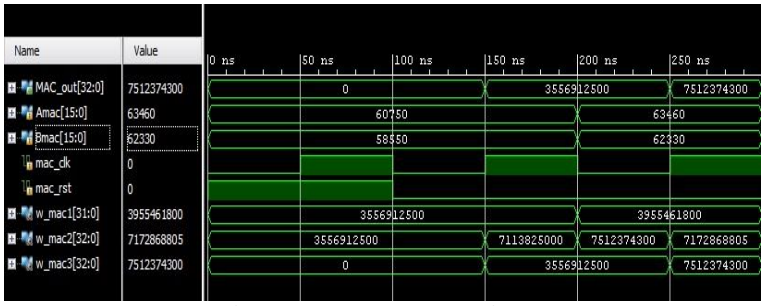


Fig. 8: Simulation Result of 16 Bit MAC Unit with Modified Hybrid Han Carlson Adder.

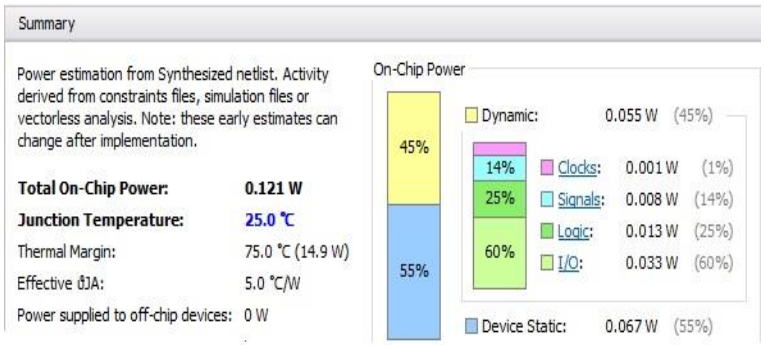


Fig. 9: Power Report of 16 Bit MAC Unit with Hybrid Han Carlson Adder.

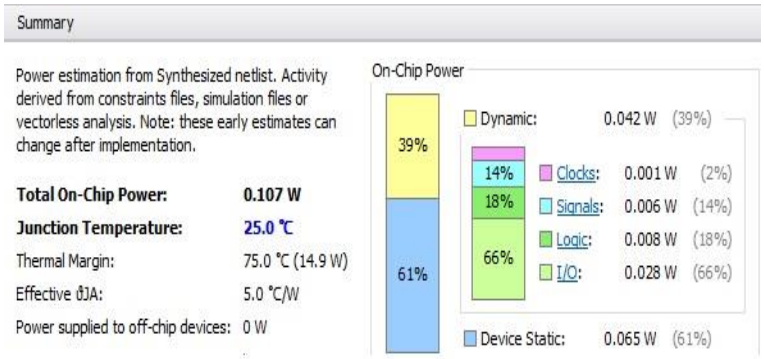


Fig. 10: Power Report of 16 Bit MAC Unit with Modified Hybrid Han Carlson Adder.

Table 1: Comparison of MAC Architectures

Architecture	On Chip Power (W)	Power Delay Product (nJ)
16 bit MAC using Hybrid Han Carlson Adder	0.121	1.770
Proposed 16 bit MAC using Modified Hybrid Han Carlson Adder(HHCA_MPPS)	0.107	1.607

5. Conclusion

The proposed design of MAC unit for a Digital Signal Processor is found to be an optimized design in terms of power consumption and power-delay product. It is evident from the performance analysis that the 16 bit MAC design with modified Hybrid Han Carlson adder showed an improvement of 11.57% in the power consumption and 9.21% in the figure of merit. The power efficient designs can be utilized for many data intensive digital signal processing applications. In future work more focus may be applied to the high speed design part of the architecture thereby reducing the delay further and improving speed.

References

[1] M. Mohamed AsanBasiri, Sk. Noor Mahammad, "Configurable Folded IIR Filter Design", *IEEE Transactions on Circuits and Systems - II: Express Briefs*, Vol. 62, No.12, (2015), pp. 1144 – 1148. https://doi.org/10.1109/TCSII.2015.2468917.

[2] Tung Thanh Hoang, Magnus Själander, Per 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, (2010), pp. 3073 – 3081.https://doi.org/10.1109/TCSI.2010.2091191.

[3] S. Ahish, Y.B.N. Kumar, Dheeraj Sharma, M.H. Vasantha, "Design of High Performance Multiply-Accumulate Computation Unit", *Proc. of IEEE International Advance Computing Conference (IACC)*, (2015), pp. 915-918.

[4] V. Nithish Kumar, Koteswara Rao Nalluri, G. Lakshminarayanan, "Design of Area and Power Efficient Digital FIR Filter Using Modified MAC Unit", *Proc. of IEEE International Conference on Electronics and Communication Systems*, (2015), pp. 884-887.

[5] MarojuSaiKumar, D. Ashok Kumar, Dr. P. Samundiswary, "Design and Performance Analysis of Multiply-Accumulate (MAC) Unit", *Proc. of IEEE International Conference on Circuit, Power and Computing Technologies (ICCPCT)*, (2014), pp. 1084-1089.

[6] C.P. Narendra, Dr. K.M. Ravi Kumar, "Low Power MAC Architecture for DSP Applications", *Proc. of IEEE International Conference on Circuits, Communication, Control and Computing (I4C)*, (2014), pp. 404 – 407.

[7] A. Abdelgawad, "Low Power Multiply Accumulate Unit (MAC) for Future Wireless Sensor Networks", *Proc. of IEEE Sensors Applications Symposium (SAS)*, (2013), pp. 129-132.https://doi.org/10.1109/SAS.2013.6493571.

[8] SuryasnataTripathy, L.B. Omprakash, K. Sushanta Mandal, B.S. Patro, "Low Power Multiplier Architectures Using Vedic Mathematics in 45nm Technology for High Speed Computing", *Proc. of IEEE International Conference on Communication, Information & Computing Technology (ICCICT)*, (2015).

[9] S. Rakesh, K.S. Vijula Grace, "A Survey on the Design and Performance of various MAC Unit Architectures", *Proc. of IEEE International Conference on Circuits and Systems (ICCS)*, (2017), pp. 312 – 315.https://doi.org/10.1109/ICCS1.2017.8326011.

[10] Tackdon Han, David A. Carlson, "Fast Area - Efficient VLSI Adders", *Proc. of IEEE 8th Symposium on Computer Arithmetic (ARITH)*, (1987), pp 49 – 56. https://doi.org/10.1109/ARITH.1987.6158699.

[11] SreenivaasMuthyalaSudhakar, Kumar P. Chidambaram, Earl E. Swartzlander Jr., "Hybrid Han-Carlson Adder", *IEEE*, (2012), pp 818 – 821.

[12] I. S. Abu-Khater, A. Bellaouar, M.I. Elmastry, "Circuit Techniques for CMOS Low-Power High-Performance Multipliers", *IEEE Journal of Solid-state Circuits*, Vol. 31, No.10, (1996), pp. 1535-1546.https://doi.org/10.1109/4.540066.