Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 5, 2021, Pages. 139-149
Received 15 April 2021; Accepted 05 May 2021.

Energy Efficient and High Throughput Multiply-Accumulate (MAC) Architecture

R.P. MeenaakshiSundhari[1], P. Kathiravan[2], VR. Shankar ganesh[3]
Keerthana[4]
[1]Professor, Department of ECE, P.A. College of Engineering and Technology Pollachi, rpmeenaakshi@gmail.com
[2]HoD, Department of ECE & P. A. polytechnic college, Pollachi
[3]HoD, Department of EEE & P. A. Polytechnic college, Pollachi
[4]UG Scholar , Department of ECE & P.A. College of Engineering and Technology Pollachi.

Abstract
The Multiply-ACCUMULATE (MAC) device is a familiar digital block that is widely used for numerous data-intensive applications in microprocessors and in the digital signal processors. In many filters, MAC units can effectively accelerate orthogonal frequency-division multiplexing algorithms. A two-cycle multiply-accumulated (MAC) high speed and an energy-efficient design is proposed that requires two supplementary numbers, bits of accumulator guard and saturated circuitry. The first stage consistsofonly part-product generation circuits and a reduction tree, whereas the second stage integrates all other functionality with a single sign extension approach. The proposed design is extendedto produce a double-throughput MAC (DT-MAC) device that either executes or accumulates efficient multiplying operations. A simpler method of combining the two binary numbers is achieved using adders to reduce processing time. The proposed adder is constructed with a KoggeStone Adder (KSA) and Brent-Kung (BK) parallel to the carry-look style adder in the MAC accumulator. It performs in a minimum period of time andit measures the fastest addition and is commonly used in industry for the achievement of highly efficient arithmetic circuits. In the KSA the carriers are computed in parallel. Chip area is reduced in BK. The proposed MAC with parallel prefix adder results in power reduction and high throughput.

Index Terms: Multiply-Accumulate unit, parallel prefix adder, Wallace Tree, partial product reduction.

INTRODUCTION

The power optimizedDigitalSignal Processing (DSP) system in wireless sensor networks is becoming more relevant. Thereforeconsidering the quickly developing environment of mobile devices and the extreme constraint of battery life power-conscious architecture needs to take account of variables. A power-aware DSP module can change energy consumption by reducing energy resources of systems or adjusting performance standards. For this purpose a flexible power scalability DSP module such as the variable bit accuracy and the variable memory space is useful which is utilized for a wide range of scenarios and for adapting any sensor node's operating circumstance. The DSP function uses the MAC method which makes it as the most successful process. The power-efficient MAC system is therefore essential for the power-aware DSP. MAC units like these are usually programmed for a preset operand size. For example, the 16-bit inputin practical sensor network implementations, each input often contains a low range while data-path hardware is intended to support the best possible accuracy. As an example the 8-bit multiplication onto a 16-bits multiplier will result in extreme power failure because signal switches are not required. Thus the power-aware architecture solution was proposed in previous works. The MAC is a regular digital block which gets utilized in microprocessors and in the digital signal processors mostly for data-intense applications.

For an example, MAC units usually speed up efficiently by multiplexing algorithms for orthogonal frequency division. A vital MAC architecture is a multiplier and a build-up as set out in Fig. 1.

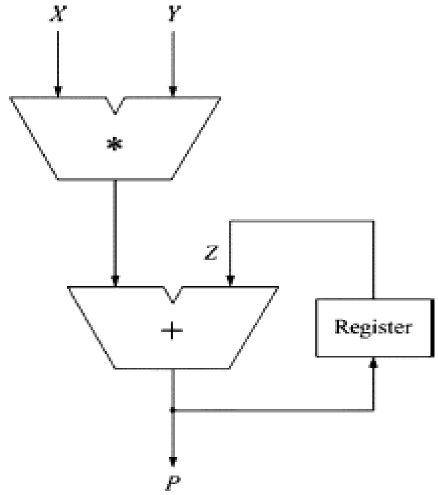

Fig.1. Architecture of general MAC

The inputs are provided to the multiplier and the final input summarizes the following objects. In addition, the multipliers are a component product unit (i.e.,PP unit) and a compact adder (final adder). In order to optimize MAC output, the crucial pause must be minimized by adding a separate pipeline record both within the proposed PP unit in between the final adder and PP unit. This provides a 3-cycle MAC architecture (As shown in Fig. 1), but boosts the bandwidth, electricity and the whole region. Many previous experiments have been based on architectural approaches to minimize multiplier latency in the PP device or in the final adder. It can be used in the PP unit with the algorithm changed from the stand or its succeeding devices. The partial tree reduction result of the PP system may be designed with high speed or speed compressors. Mathew et al, suggest a distribute forward look-adder to easily combine PP unit outputs. Liu et al, use the hybrid add-on to minimize the time delay as a design that needs equivalent time for each adder. Two separate transport propagations are wasted in the same MAC circuit, as transport takes time. The trick for the current method is to expand the commodity signal; the sign-extending circuit and the saturation device accumulate in the second phase. In the second phase of the pipeline, the product feedback would be searched.

As the Wallace tree is formed into parallel artefacts, its time of operation is proportionate to the count of inputs. So the total count of outputs is reduced by number 1 of the inputs. Several counters (3:2) or (7:3) are used to reduce output volume at each pipeline stage during the actual implementation. The MAC combines multiplication and combination calculations with current operations and proposes that the hybrid CSA structure decreases the critical path and improves the production frequency. A CLA is introduced into the CSA tree to decrease the count of sections in the last adder. In addition, combined intermediate measurement effects in place of final additive outputs are added to maximize the production rate by increasing the pipeline space.

Section II covers two cycles of MAC architecture and three cycles. Section III explains the partial decline of Wallace tree materials. In Section IV the parallel prefix adder is briefed. The findings are displayed in Section V andthe article finishes in Section VI.

PIPELINED MAC ARCHITECTURE

A single MAC design is a multiplier and an integrated adder, as seen in Figure 2. The feedback is given to the multiplier and summarizes certain items in the accumulated adder. In addition, the multipliers are a component product unit (PP unit) and a compact adder (final adder). The crucial pause needs to be minimized by adding a separate recording pipeline both in the PP unit and among the PP unit and ultimate adder to improve the MAC output. This provides a 3-cycle MAC design (Fig. 2), but raises overhead for latency, energy and area.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 5, 2021, Pages. 139-149
Received 15 April 2021; Accepted 05 May 2021.

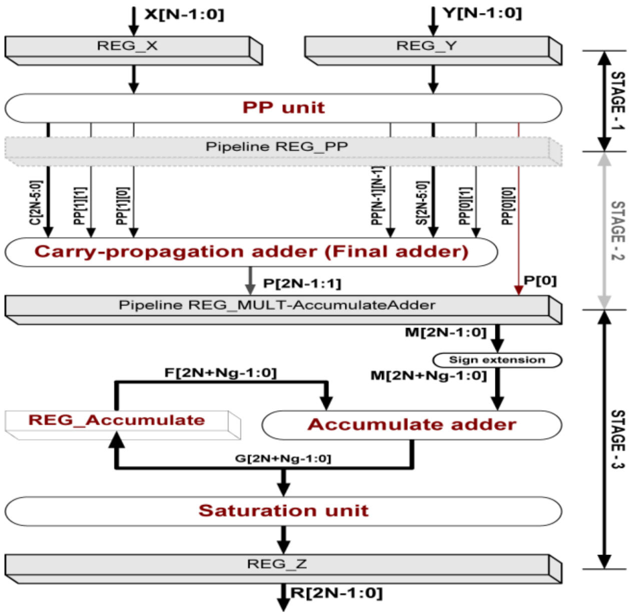

Fig.2 The block diagram of the MAC architecture with 2 and 3 stages

The MAC nature of the two supplements suggested is shown in figure 3. In the first stage evaluates to the simple architecture in Fig. 2, the actual design changes the final adder with a carry-save adder in the second. So the significant latency of the MAC architecture depends still on the PP, but now the delays of both phases are the same. The second stage is quicker particularly for larger operand sizes andby which it can accumulate more guard pieces.

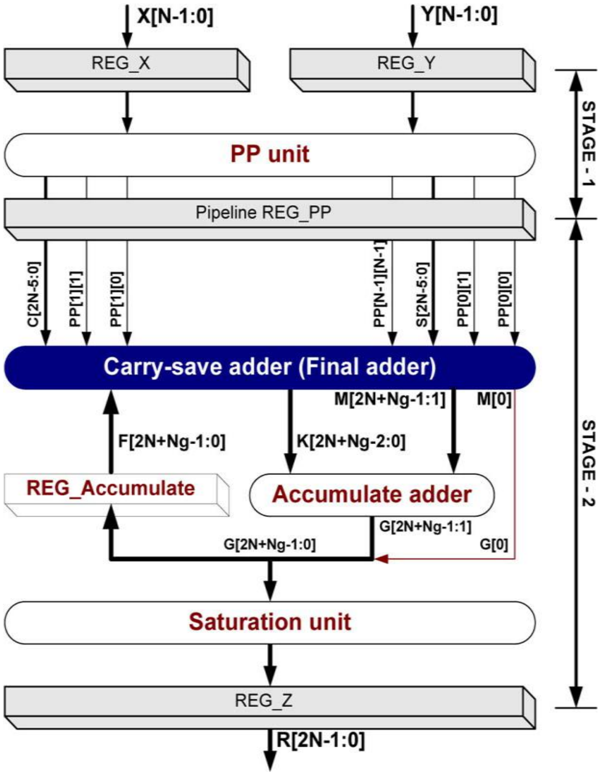

Fig.3 The block diagram of a proposed MAC-2C new architecture.

Three architectures the final adder and the engineered adder are the same PP unit configuration.

- MAC-2C is a two-cycle MAC with a path from PP to final adder (in Fig. 2).
- MAC-3C is a 3-cycle in MAC but key route is entirely inside the PP unit (Fig. 2).
- MAC-NEWsignifies our 2-cycle MAC which also covers the PP machine fully; (Fig. 3).

The new MAC system intrinsically provides a faster pace than the two-cycle foundation architecture. We therefore evaluate whether the time constraints available can be used to minimize energyandpower. Below we just lower the door with the present timing slack, so this is a practical way of preserving energy. The voltage reduction is certainly an option, but it needs a specific generation and overhead. MAC-2C and MAC-NEW are now compared with the same time restrictions. For example, Step 1 of the MAC-NEW with a time limit corresponding to the critical pause in phase 1 for MAC-2C is added. Once the last adder in MAC-NEW is replaced, the PP system meets the time limit for low-speed gates. In the other hand as the existing carrying adder precedes the accumulation adder of the MAC-NEW, the gates must be slightly extended to meet the stage 2 of the MAC-2C.

THE WALLACE TREE MULTIPLIER

It is an efficient digital circuit hardware program multiplying two integral components. It has three steps:

- Multiply every bit of one story, every bit of a different statement.
- Minimize the sum of partial products to the 2 layers of half and full adders.
- Group and attach 2 numbers of wires to the traditional adder.

The disadvantage of the wallace tree is the faster rate than the naïve insertion of partial products with normal suppliers. They haveO(log n)-reduction layers, but for each and every layer only O(n) propagation lag is present. The naive inclusion of partial objects takes O(log2n) time. Since the partial products are generated as the final addition O(1) and O(log n), the total multiplication is O (log n). A complex theoretical perspective is used by the Wallace tree algorithm to multiply class NC1. In comparison to the naive inclusion of partial products, the downside of the Wallace tree is the far larger door count. These estimates just take account of gate delays and do not fix wire delays that are also very important. The wallace tree is often seen with an additional tree of 3/2 or 4/2.

It's a fast way to multiply. The routine of the Wallace Tree multiplier is swifter for large operands. As seen in Fig. 4, the partial product matrix is restructered to make a tree-like framework for a screen multiplier. The amount of supplements and the essential path is decreased. The key functional block of every processor unit is a multiplier. There are many multiplication algorithms that can be built using the type of the multiplier. The Wallace tree algorithm is beneficial for speed of operation among several multiplication algorithms. Technology advances expand the high-speed and low power consumption. The Wallace tree multiplier's operation is the same in the first cycle of multiplication and generates partial products. The Wallace tree multiplier introduces partial products in the first three rows in the second round. The quantity and the product are then added to the next set of partial parts. This method continues until the finished product is manufactured. For this row-wise add-on process, half and full adders are used. In the processing of finished product conditions additional products therefore play a very important function. The rate of addition will affect the process of multiplication.

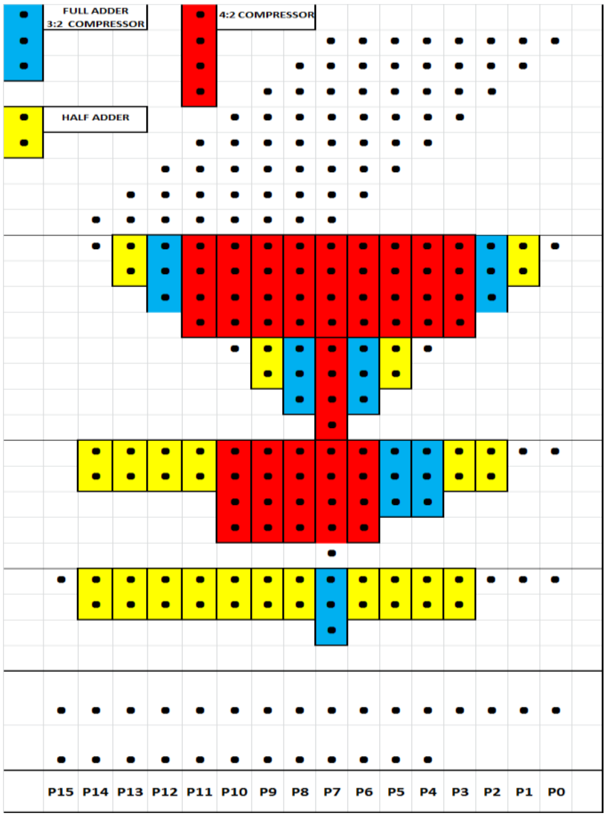

Fig. 4. An 8X8 Wallace Tree Multiplier with 4:2 Compressors

In developing wallace trees multipliers, the adder architecture plays an essential role in enhancing the efficiency of the multiplication process. This article introduces a new system for wallace propagators in which PPAs will connect the final line of partial products to the total generated in the previous stage and generate the final conditions for the commodity. The PPAs are designs derived in the first place from the concept of generating and propagating parts.

A compressor (4:2) takes four bits of a partial column matrix, creates a column-to-one column contribution of two bits, and the same column produces one bit. Thus (4:2) a compressor is fitted with four inputs I1, I2, I3 and I4 and gives two Sum S and C outputs with the COUT CIN. The compressor shows (4:2) with two absolute adders in Figure 5.

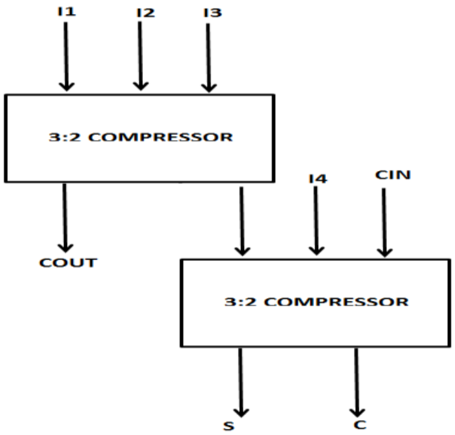

Fig. 5. 4:2 Compressors

PARALLEL PREFIX ADDER

The Kogge–Stone adder

The Kogge–Stone Adder (KSA or KS) is a parallel prefix form of carry look ahead adder. The Kogge–Stone adder has more fields than the Brent–Kung adder, but on each stage it has a lower fan which improves the output for normal CMOS method nodes. For Kogge–Stone adders, however, cable congestion remains a concern. As seen, a "propagate" and a "generate" bit occurs per vertical point. In the end (vertically) the climax bits are formed when the initial spread bits after the input are XOR'd (red boxes). For example, the propagated bit in the far-right red box (one "1") and a carrying bit (the "0"), producing "1" are determined by the first (less significant) sum bit. For the second case to the right (a "0"), the second bit of the XORing is determined by a C0 (a "0") producing a "0"
The adder sparsity is the sum of bitsthat the truck creates in the so-called sparse Kogge–Stone adder (SKA). Each bit is called Sparsity-1, every bit is called Sparsity-2, with Sparsity-4 being the fourth bit. The resulting transportation is then used as inputs for much shorter carries or other designs which produce the final quantities of the components. Increased economies minimize the overall estimate needed to mitigate congestion.

Adder is a factor in the design and processing of automatic circuits. Adder thus constitutes the principal field of research for the success of digital structures in the VLSI architecture. The foundation for power consumption and delay is efficiency. Parallel VLSI technology prefix adders have proved reliable. Adders are not only used for arithmetical processes but also for address and index calculations. Logic gate variations are used by adders to combine the number with the binary values. The producers are divided and able to merge the figures. The simultaneous integration of Prefix Adders into microprocessors, DSPs, handheld computers and other high speed applications is carried out. The decreased complexity and latency of the Parallel Prefix Adders improves productivity with factors such as delay and strength. The parallel prefix supplements are essential for high-speed arithmetic circuits.

The delay to incorporate Carry-Look Ahead can be solved with the parallel prefix extension system. This concept is to measure small intermediate prefixes until all bits are measured and find large prefix community. The estimation of the parallel prefix takes three crucial steps:
1) Quantify input bits no produced and also the propagated signals.
2) Calculate the whole parallel chain holding estimate called prefix.
3) Calculate the final input figure.
Pre-processing step: We measure, generate and transmit signals on each pair of inputs A and B at this point. These signals are seen in the logic equations 1&2.
Pi = Ai x-or Bi ...……….(1)
Gi = Ai and Bi ……...….(2)
The Carry Generation Network: We quantify each bit at this point. The execution of these activities is performed in tandem. Transmission and generation are seen as intermediate signals. Below are the logical equations for propagation and generation.
Pi:j = Pi:kand Pk-1:j …………(3)
Gi:j = Gi:k or (Pi:kand Gk-1:j) ..(4)
As shown in fig 6, Black/gray cells enforce the two equations provided, normally used in the following prefix trees debate.
Stage of Post Processing: The logic equations used to calculate the total bits for the input bits are given below.
Ci = (Pi and Cin) or Gi ……(5)
Si = Pi xor Ci-1 ……………(6)

Received 15 April 2021; Accepted 05 May 2021.

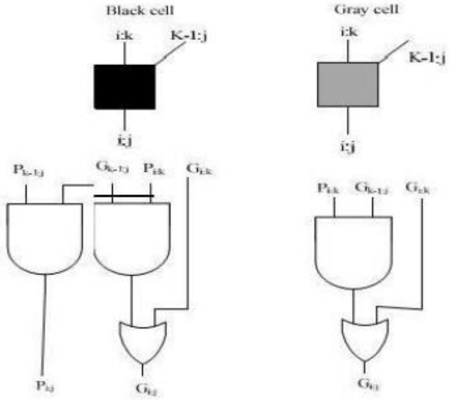

Fig.6. TheBlack cell and Gray cell

It is a parallel prefix formation for Carry Look-ahead Adder. Fig. 7 is seen as a parallel prefix adder composed of operator nodes. It is the fastest adder based on the time setup.

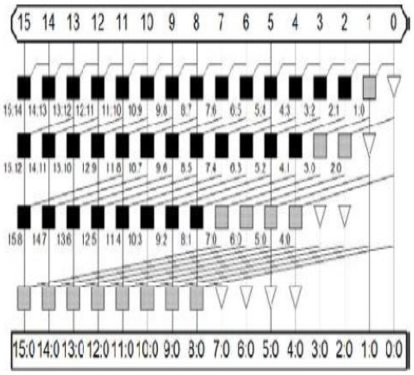

Fig.7.16-bit ks adder

The Brent Kung Adder

The Brent–Kung adder for the wagon lookout is the Prefix Adder (PPA) parallel (CLA). The adder construction has reduced cable congestion and decreased the regularity, which improves stability and less surface requirements for the Kogge-Stone adder (KSA). It's much smoother than belt adders (RCA). The first multi-bit supplements were developed in the early days and called after the transport impact, which spread from right to left. The time required was directly proportional to the inserted bit duration. The carrier is tested in parallel at Brent–Kung adders, thereby minimizing the additional period. That's the other way around. Further testing has been carried out to reduce energy consumption and flash range and to speed up Brent-Kung add-ons and other concurrent add-ons to make them ideal for low strength.

A Brent-Kung adder is a traditional parallel adder that minimizes the chip area and encourages its growth.

The Network of Brent-Kung Prefix for the Nuclear Process is part of the hybrid system (Fig. 8). One of the benefits of the adder being that the longer carriage is identical to the intermediate carriages that the scheme is described by forward roads. Compared to some parallel additives, the fan out the adder is weaker, yet the duration of the cable is smaller.

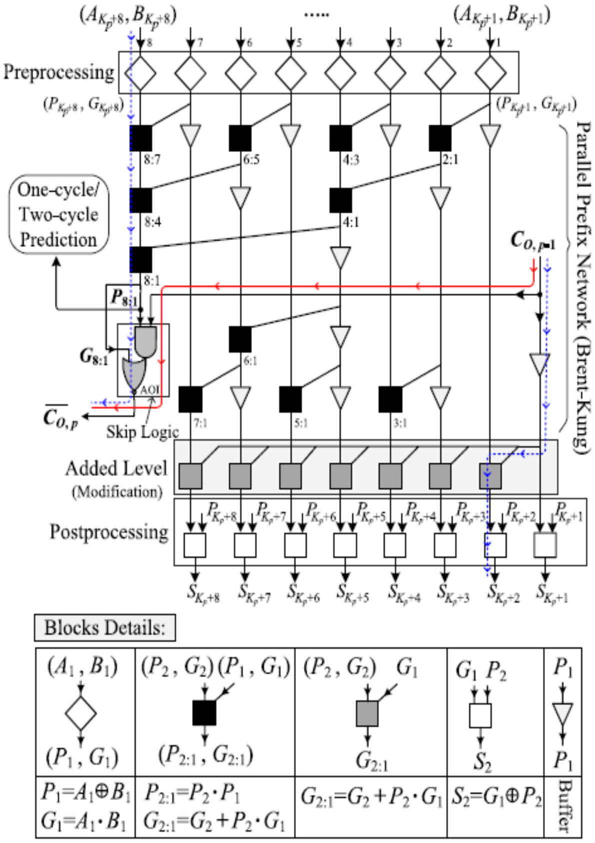
Preprocessing
One-cycle/ Two-cycle Prediction
Parallel Prefix Network (Brent-Kung)
Skip Logic
AOI
Added Level (Modification)
Postprocessing
Blocks Details:
$P_1 = A_1 \oplus B_1$
$G_1 = A_1 \cdot B_1$
$P_{2:1} = P_2 \cdot P_1$
$G_{2:1} = G_2 + P_2 \cdot G_1$
$G_{2:1} = G_2 + P_2 \cdot G_1$
$S_2 = G_1 \oplus P_2$
Buffer

Fig.8.TheBrent Kung adder

Finally, the style is plain and conventional. Internal phase p configuration, including the revised PPA and the skip justification, is seen in Fig 8. Notice that the PPA is determined to be 8 (i.e. Mp = 8) for this figure.

At the pre-processing point, the propagation signals (Pi) and signals (Gi) are determined, as seen in the diagram. In the next step, the longer propagation (i.e. G 8:1) of the prefix network and P8:1 is determined using the concurrent prefix network Brent–Kung which is generated faster than all intermediate signals in the network by all propagated input signals. Signal P8:1 in the skip logic would be used to decide if the outcome of execution from the preceding stage (i.e. CO,p−1) should be skipped. This signal is often used as a prediction signal in the latency adder vector. Both these activities should be described in parallel with other methods. If P8:1 is one, CO,p−1 can miss the stage predicting those essential routes. In contrast, CO,p is identical to G8:1 and P8:1 is zero. In this scenario, no critical route would be permitted.

Intermediate carriers are determined by the parallel network prefix functions CO, P−1 and intermediate signals (Fig. 8). Finally, the product quantity is determined at the post-processing stage.

RESULTS AND DISCUSSIONS

The simulation outcome of the proposed MAC using wallace tree reduction architecture and the parallel prefix adder is given in Fig.9. The proposed design has been implemented in VHDL and simulated using Modelsim. RTL design is synthesized using Xilinx ISE and its parameters are shown in Table I.

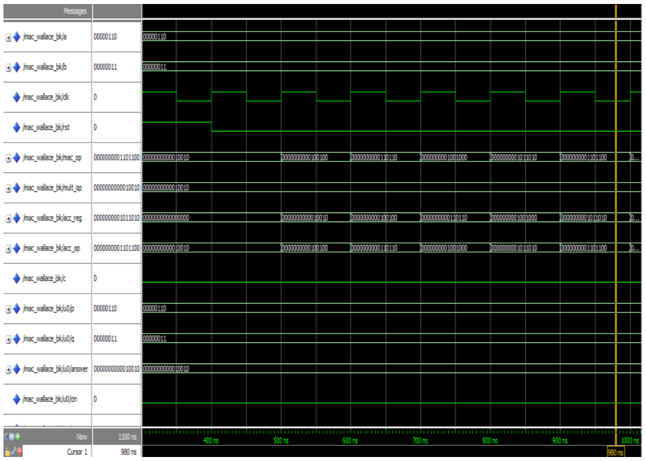
Messages
/mac_wallace_bk/a 00000110 00000110
/mac_wallace_bk/b 00000011 00000011
/mac_wallace_bk/clk 0
/mac_wallace_bk/rst 0
/mac_wallace_bk/mac_op 0000000001101100 0000000000010010 0000000000100100 0000000000110110 0000000001001000 0000000001011010 0000000001101100
/mac_wallace_bk/mult_op 0000000000010010 0000000000010010
/mac_wallace_bk/acc_reg 0000000001011010 0000000000000000 0000000000010010 0000000000100100 0000000000110110 0000000001001000 0000000001011010
/mac_wallace_bk/acc_op 0000000001101100 0000000000010010 0000000000100100 0000000000110110 0000000001001000 0000000001011010 0000000001101100
/mac_wallace_bk/c 0
/mac_wallace_bk/u0/p 00000110 00000110
/mac_wallace_bk/u0/q 00000011 00000011
/mac_wallace_bk/u0/answer 0000000000010010 0000000000010010
/mac_wallace_bk/u0/cin 0
Now 1100 ns
Cursor 1 980 ns
400 ns
500 ns
600 ns
700 ns
800 ns
900 ns
1000 ns
980 ns

Fig.9. Simulation result of proposed MAC

Table.I Performance Comparison between existing method and proposed method

MAC Types	Area(Gate Count)	Power(mW)	Delay(ns)
MAC-2C	2088	167.88	26.301
MAC-3C	2272	199.12	25.893
MAC-WALLACE-KS	1544	132.79	8.863
MAC-WALLACE-BK	1406	130.97	7.715

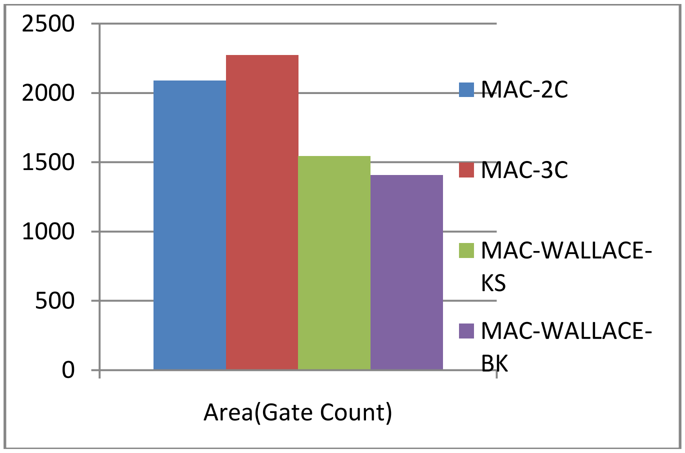

Fig.10. Comparison chart of Gate Count for Different MAC designs

Received 15 April 2021; Accepted 05 May 2021.

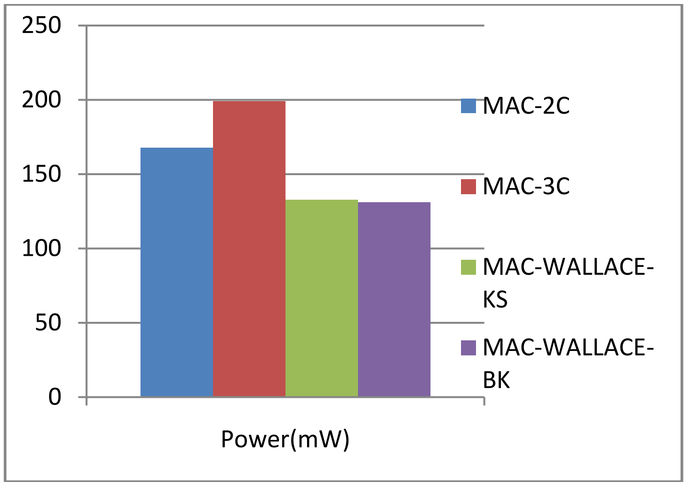

Fig.11. Comparison chart of Power for Different MAC designs

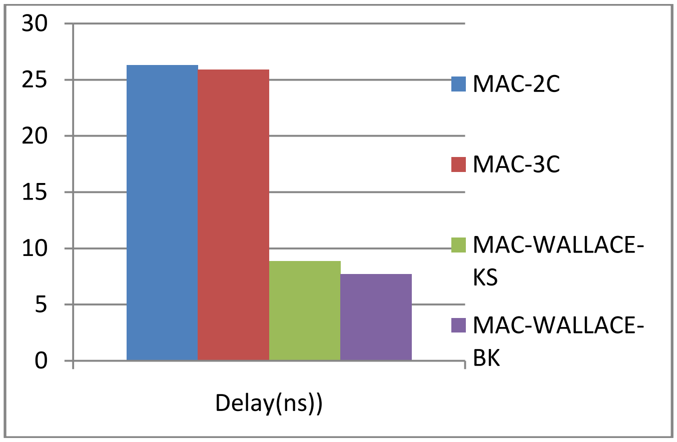

Fig.12. Comparison chart of Delay for Different MAC designs

CONCLUSION

The modern two-cycle multiply-accumulate (MAC) architercture, which is a high-speed, energy-efficient design is proposed. The design of the wallace tree reduction results in a partial product reduction. Replacing the multiplier accumulator adder with a parallel prefix adder makes the two-cycle MAC architecture quicker and more energy effective and region efficient than the basic two-cycle MAC architecture. The latest system is calculated at 31 percent quicker and decreases operating resources by an average of 32 percent relative to traditional double and triple-cycle MAC designs.

REFERENCES

1. A.Prasanth, S.Jayachitra, 'A Novel Multi-Objective Optimization Strategy for Enhancing Quality of Service in IoT enabled WSN Applications', Peer-to-Peer Networking and Applications, Vol.13, 2020, pp.1905–1920.
2. S. Yoshizawa and Y. Miyanaga, “Use of a variable wordlength technique in an OFDM receiver to reduce energy dissipation,” *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 55, no. 9, pp. 2848–2859, Oct. 2008. [20] R. K. Kolagotla, J. Fridman, B. C. Aldrich, M. M. Hoffman, W. C. Anderson, M. S. Allen, D. B. Witt, R. R. Dunton, and L. A. Booth, “High performance dual-MAC DSP architecture,” *IEEE Signal Process. Mag.*, vol. 19, no. 4, pp. 42–53, Jul. 2002.
3. S. Hong and S.-S. Chin, “Reconfigurable embedded MAC core design for low-power coarse-grain FPGA,” *Electron. Lett.*, vol. 39, no. 7, pp. 606–608, Apr. 2003.
4. M. Själander, H. Eriksson, and P. Larsson-Edefors, “An efficient twinprecision multiplier,” in *Proc. IEEE Int. Conf. Comput. Des. (ICCD)*, Oct. 2004, pp. 30–33.

5. S. K. Mathew, M. A. Anders, B. Bloechel, T. Nguyen, R. K. Krishnamurthy, and S. Borkar, "A 4-GHz 300-mW 64-bit integer execution ALU with dual supply voltages in 90-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 40, no. 1, pp. 44–51, Jan. 2005.
6. H. Eriksson, P. Larsson-Edefors, M. Sheeran, M. Själander, D. Johansson, and M. Schölin, "Multiplier reduction tree with logarithmic logic depth and regular connectivity," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2006, pp. 4–8.
7. S. KanagaSubaRaja, S. UshaKiruthika, 'An Energy Efficient Method for Secure and Reliable Data Transmission in Wireless Body Area Networks Using RelAODV', International Journal of Wireless Personal Communications, ISSN 0929-6212, Volume 83, 2015
8. T. T. Hoang, M. Själander, and P. Larsson-Edefors, "Double throughput multiply-accumulate unit for FlexCore processor enhancements," presented at the IEEE Int. Symp. Parallel Distrib. Process. (IPDPS), Reconfigurable Archit. Workshop (RAW), Rome, Italy, May 2009.
9. S.-R. Kuang and J.-P. Wang, "Design of power-efficient configurable booth multiplier," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 3, pp. 568–580, Mar. 2010.
10. D. Brooks and M. Martonosi, "Dynamically exploiting narrow width operands to improve processor power and performance," in *Proc. Int. Symp. High-Perform. Comput. Archit.*, 1999, pp. 13–22.
11. M. Hatamian and G. L. Cash, "A 70-MHz 8-bit 8-bit parallel pipelined multiplier in 2.5- m CMOS," *IEEE J. Solid-State Circuits*, vol. JSSC-21, no. 4, pp. 505–513, 1986.
12. P. M. Kogge and H. S. Stone, "A parallel algorithm for the efficient solution of a general class of recurrence equations," *IEEE Trans. Comput.*, vol. C-22, no. 8, pp. 786–193, Aug. 1973.
13. J. Sklansky, "Conditional-sum addition logic," *IRE Trans. Electronic Comput.*, vol. EC-9, pp. 226–231, 1960.
14. Murugan, S., Jeyalaksshmi, S., Mahalakshmi, B., Suseendran, G., Jabeen, T. N., & Manikandan, R. (2020). Comparison of ACO and PSO algorithm using energy consumption and load balancing in emerging MANET and VANET infrastructure. *Journal of Critical Reviews*, *7*(9), 2020.