iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 6 **Issue:** VII **Month of publication:** July 2018

DOI: http://doi.org/10.22214/ijraset.2018.7103

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 6 Issue VII, July 2018- Available at www.ijraset.com

Design of Multiplierless Multiple Constant Multiplication for Convolution Circuit

Goparaju Kolusu[1], Siva Reddy. V[2]
[1, 2]*Velagapudi Ramakrishna Siddhartha Engineering College*

Absrtact: ***A radix-3 partitioning scheme can provide the pre-multiplication factors for natural numbers, which they engaged to construct a convolution circuit i.e. used for multimedia and filtering applications. In proposed method, the partitioned unsigned integer input is multiplied with 32-bit floating point filter coefficient. Ancient architectures like distributed arithmetic, shifters, recoding circuitry and other multiplier circuits are substituted with ROMs and floating point adders used for improving the efficiency. In floating point inputs, for mantissa addition the carry save adder can be substituted by parallel prefix adder. Subsequently, derived from the multiplied result it led to less area occupation. The obtained results are simulated by Xilinx ISE and compared for betterment of area, power and delay with the former approaches.***
Index Terms: Convolution, Floating Point Adder, Gaussian filter, Parallel Prefix Adder.

I. INTRODUCTION

These days in the elaboration of great media substance have advanced a serious research action for the change of separating administrators, whose hardware (HW) multifaceted nature is a fundamental worry in applications intended to unadulterated speed, for instance picture and video elaboration[1], [2]. Such many-sided quality, in reality, by and large backslides in the designation of a major number of math administrators and a subsequent loosening of the aggregate circuit. The ongoing writing demonstrates that the previously mentioned issue is normally overseen either by repeating to the full/fractional serialization of the filters [3], [4] and collapsing strategies [5] or by meddling on inward multifaceted nature of intertwined duplicate adders and multiply accumulators (MAC). Since the prior strategy by and large causes an extensive diminishment of the filter exhibitions [6], the propelled approach remains the most precise technique to accomplish a decent power, performance, and area. In such manner, the whole expulsion of the multiplier hardware is by a long shot the favored alternative of a few creators [6], [7], who rehash to speedy adders and shifters rather than multipliers, according to the coding of the operands, canonical signed digit(CSD), and modified booth (MB), for the most part [8], [9]. The disentanglement of separating circuits turns out to be particularly successful when one of the operands is lessened to a limited arrangement of pre-figured qualities, as by virtue of predefined channel portions. In such cases, the Distributed Arithmetic (DA) technique [10] can be effectively connected to parcel augmentations in more straightforward movements and increments. By utilizing recollections to store pre-ascertained incomplete entireties, whose number can be decreased with the help of multiple-constant multiplication (MCM) strategies [11], [12]. DA can be, on a fundamental level, advantageously utilized rather than MB and CSD [13]. Nonetheless, genuine displays of DA result from a watchful tradeoff between its "characteristic" piece serial activity and the parallelism by which the fractional totals are figured, which can manual for the unreasonable increment of mapped physical resources[10].

In this brief, another dividing technique is proposed, in view of radix-3 terms, acquired with the arrangement of the weight issue [14], with the motivation to advance the shows of DA with the exhibits of regular reason multipliers in the particular settings of MCM. Despite the fact that it depends on the comparative activity standard of DA, we will exhibit that the proposed procedure is constantly beneficial in states of mapped physical assets and elaboration speed, for duplicating 32-bit Floating-Point (FP32) channel coefficients by the parceled whole number information sources. Along these lines, shifters and recoding hardware, run of the mill of other understood multiplier plans, is totally supplanted by floating point (FP) adders, whose HW many-sided quality can be rearranged to that of settled point adders, without undermining the precision. The determined usage is satisfactory for various consistent separating applications, working with FP precalculated portion coefficients and information amounts incorporated into a scope of number qualities good with mixed media elaboration. The execution of the proposed multiplier on a Xilinx ISE instrument, restores an aggregate defer way of 2.254 ns to produce an IEEE-754 FP32 [15] result, beginning from a 8-b unsigned whole number info. In this FP adder, for mantisa addition the carry save adder can be substituted with parallel prefix adder named kogge stone adder. Subsequently the former approach led to large area occupation, it has to be reduced by using the latter one.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 6 Issue VII, July 2018- Available at www.ijraset.com

II. PARTITIONING METHOD

The issue of working up minimal number of whole numbers and their qualities with the end goal that every one of the numbers in a restricted range is communicated as a blend of them has been looked since the 17^{th} century since it finds an extensive range of uses, including the decrease of the hardware dedicated to separating apparatus[16]. About this contention, a few specific shows have been distributed [17], [18]. where it has been absolutely exhibited that, having characterized the set W_r := {3^0, 3^1, 3^2,..., 3^{n-1}, R} with R = r − (3^0 + 3^1 + 3^2 + ⋯ + 3^{n-1}), each whole number in the range [0; r] is accomplished by a superposition of the terms in 'W_r' increased for a coefficient $C_i \in \{-1, 0, 1\}$. More as a rule, characterizing the arranged arrangement of positive whole numbers that total to r = $\lambda_0 + \lambda_1 + \cdots + \lambda_n$ with $\lambda_0 < \lambda_1 < \cdots < \lambda_n$as the segment of a positive number 'r' and the set $\{\lambda_0,...\lambda_{n-1}, \lambda_n\}$ as the parts of the parcel, the accompanying is illustrated.

Each whole number 0 ≤ q ≤ r is composed as

$$q = \sum_{i=0}^{n} C_i \succ_i \qquad (1)$$

There does not exist another parcel of 'r' fulfilling 1 with less parts than n + 1.

Table I Application Of The Partition Method

Input	Partition						
	3^0	3^1	3^2	3^3	3^4	...	λ_n
0	0	0	0	0	0	...	0
1	+1	0	0	0	0	...	0
2	-1	+1	0	0	0	...	0
3	0	+1	0	0	0	...	0
4	+1	+1	0	0	0	...	0
5	-1	-1	+1	0	0	...	0
...	...	...	...	...	...	...	...
q	C_0	C_1	C_2	C_3	C_4	...	C_n
...	...	...	...	...	...	...	...
R	+1	+1	+1	+1	+1	...	+1

A vital conclusion of the above properties demonstrates that each segment of 'r' is made by precisely n +1=$\lfloor log_3(2r) \rfloor$+ 1 sections. Table I exhibit the use of the parceling technique. For example, for a 8-b input, the parts are {1, 3, 9, 27, 81, 134}; the info 23 is modified as 23 = (−1)1 + (−1)3 + (0)9 + (+1)27 + (0)81 + (0)134, specifically, the arrangement of qualities from Table I will be {−1, −1, 0, +1, 0, 0}. The above outcomes is connected to MAC activities between a non specific vector of coefficients 'A_k' and a vector of sources of info 'x_k'

$$y = \sum_{k=0}^{K-1} A_k x_k \qquad (2)$$

By utilizing (1) in (2), it brings about

$$y = \sum_{k=0}^{K-1} \sum_{i=0}^{n} A_k C_{ki} \succ_i \qquad (3)$$

which is changed as far as the parts in 'W_r' as

$$y = \sum_{k=0}^{K-1} \left[\sum_{i=0}^{n-1} A_k C_{ki} 3^i + A_k C_{kn} R \right] \qquad (4)$$

Thinking about that 'n' is little in a few instances of intrigue and that the above segment is a direct superposition of parts, the past outcomes propose that the augmentations engaged with a specific channel can be redesigned as the parcel of pre-multiplied terms, acquired by computing from the earlier the item between the parts and''A_k'. This apportioning is extremely particular from DA that requires that x_k is Separated as $x_k = \sum_{i=0}^{s-1} b_{ki}2^i$, where $b_{ki} \in \{0, 1\}$ proposes the sign digit. That is, (2) can be DA divided as

$$y = \sum_{k=0}^{K-1}\sum_{i=0}^{s-1} A_k b_{ki} 2^i \qquad (5)$$

Though the utilization of 'b_{ki}' in position of 'C_{ki}' adds to lessen some "paste" rationale to actualize (5), unmistakable estimations of 's' in (5) are extensively higher than 'n' in (4). In this way, the quantity of administrators to execute the internal items in (4) is emphatically decreased.

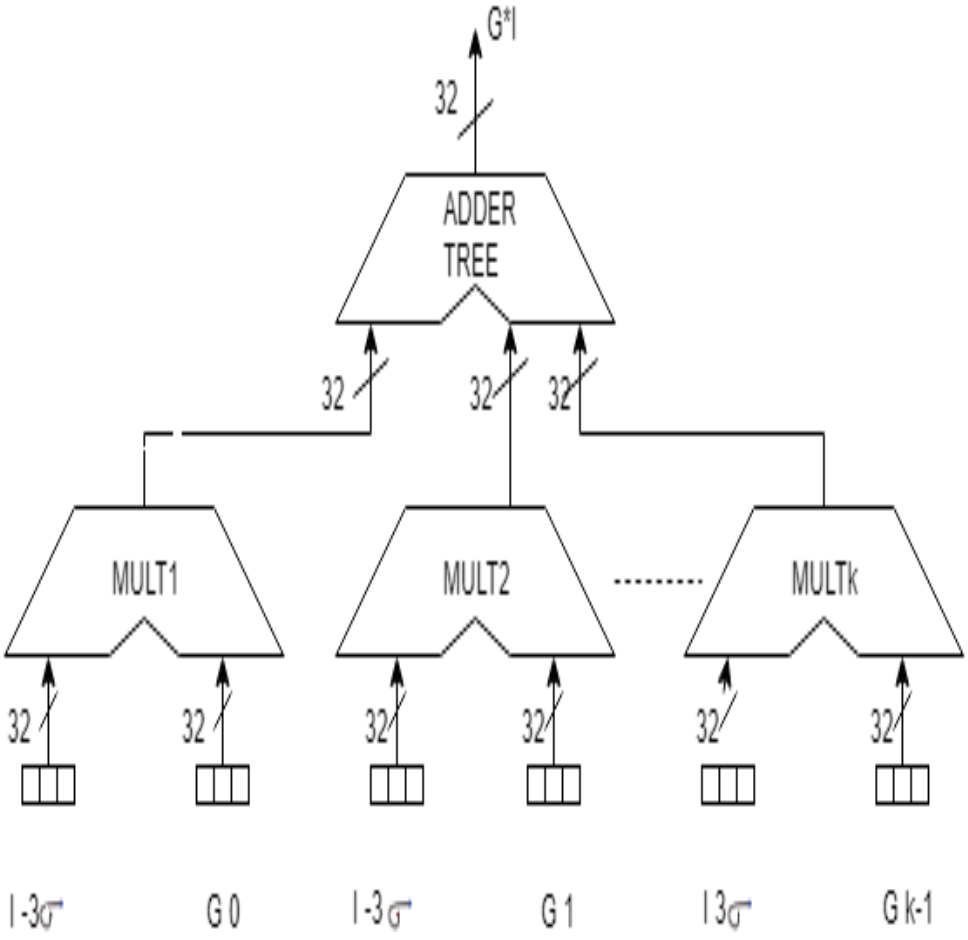


Fig. 1.outline of the convolution circuit

III. ARCHITCTURE DESIGN

The proposed method has been used for executing the plan show in Fig. 1 that registers the convolution G∗I between the FP32 part vector 'G' and the info vector 'I' of the unsigned whole number coded by 'm' bit (Uint-m). The proposed strategy is used to advance the development of the 'k' MACs that ascertain the item between the bland coefficient of the portion and the vector of information esteems. The subsequent structure is masterminded in Fig. 2. Whenever 'G' has been characterized, each component of 'I' can be separated in n + 1 sections, as per Table I, and spared in a same number of double port read-just recollections (ROMs), after they have been pre-increased by the segments of 'G'. A further $(n+1)2^{m+1}$ bit ROM spares the 'C_i' (2-b for every one) used to pick the indication of the operands.

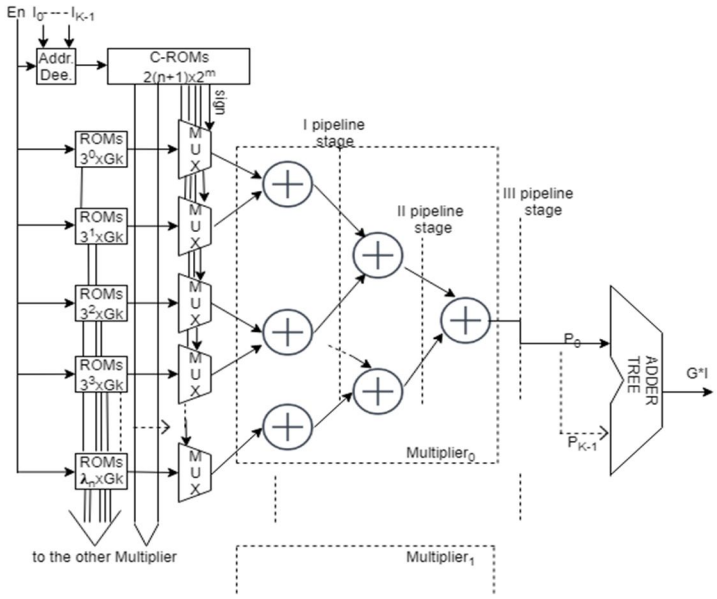


Fig.2. Scheme of the multiplier, deployed in a Gaussian convolution.

The multipliers are supplanted with the 'n' adders in the dotted box of Fig. 2, shared along a $\lceil log_2(n+1) \rceil$ profundity tree. In crucial, the adders ought to have a FP engineering, yet it is conceivable to use a custom coding for incomplete outcomes, keeping in mind the end goal to decrease their intricacy without changing the exactness of the augmentation. While the plan is locked in to incorporate with a few sorts of pieces, as a contextual investigation it has been used to play out the Gaussian channel by the part $G(x, \sigma) = Ae^{-(x^2/2\sigma^2)}$ for its huge dissemination in picture and video elaboration stream, where they for the most part work with Uint-8 input. Because of its detachability property, for which a 2-D filter can be isolated in two back to back 1-D items [3], [16], the channel execution in the space– time area is regularly wanted to the recurrence change. Results from Section II permit dividing 'I' by utilizing the essential $n + 1 = \lfloor log_3 2(2^8 - 1) \rfloor + 1 = 6$ sections from Table I, and we compose the convolution as

$$G(x,\sigma) * I(x) = \sum_{j=0}^{K-1} G_j I_{j-\frac{K-1}{2}} = \sum_{j=0}^{K-1} G_j \sum_{i=0}^{n} C_i \succ_i$$
$$= \sum_{j=0}^{K-1} \sum_{i=0}^{n} (G_j \succ_i) C_i = \sum_{j=0}^{K-1} P_j \qquad (6)$$

Specifically, all the 'n' inward items '$G_j \succ_i$' are precomputed for each estimation of 'I' in the range [0;255], for every piece coefficient, Since 'G_j' remains constant once 'σ' and 'K' have been characterized. Given that the portion length is choose with K = 6σ + 1 focuses with great precision [3] and that, for the Gaussian symmetry, it is plausible to store just 3σ + 1 esteems, (6) can be executed by the path in Fig. 2. The information 'I' is used to get to the ROMs that, on a basic level, are sharable by every one of the multipliers. Yields from the C-ROMs give the signs to choose the contributions to the principal viper's line of multipliers. The extra ROMs, having profundity (3σ + 1), require the $G_j \succ_i$ coefficients that must be included toward the last outcome. The advantages of the proposed result end up applicable when a major number of multipliers are required. In the satisfaction detailed in the following segment, with m = 8 and σ = 4, 25 MACs are required for a full-parallel circuit; hence, utilizing the radix-3 strategy, it is likely to spare 50 adders in regards to the radix-2 DA.

A. IEEE 754 standard floating point Addition Algorithm

Floating point expansion is excessively mind boggling than multiplication, little blueprint of this have been clarified underneath

A3 = A1 + A2

A3 = (M1 x 2E1) +/ - (M2 x 2E2)

1) A1 and A2 must be added if the examples are equivalent i.e E1=E2.

2) Suppose A1 has the greater estimation of the 2 numbers. Outright estimation of A1 must be greater than supreme estimation of A2, else trade the qualities with the end goal that abs(A1) is bigger than abs(A2). abs(A1)>abs(A2).
3) Beginning estimation of the exponent must be the greater of the 2 numbers, since we perceive the exponent of A1 will be more noteworthy , so essential exponent result E3 = E1.
4) Compute the distinction of exponents i.e. exp diff = (E1-E2).
5) Change the decimal point of Mantissa on the left (M2) through the exponent distinction. Presently the exponents of the two A1 and A2 are same.
6) Calculate the sum/distinction of the mantissas relying upon the sign piece S1 and S2. On the off chance that indications of A1 and A2 are same (S1 == S2) at that point add the mantissas On the off chance that indications of A1 and A2 are not same (S1 != S2) at that point subtract the mantissas
7) Normalize the subsequent mantissa (M3) if required. (1.M3 arrangement) and the primer exponent result E3=E1 required to be balanced by the normalization of mantissa.
8) If both of the operands can be ∞ or if (E3>Emax) , overflow has happened ,the yield must be set to ∞. If(E3<Emin) at that point it's a underflow and the yield must be set to 0.
9) Not a numbers are not upheld.

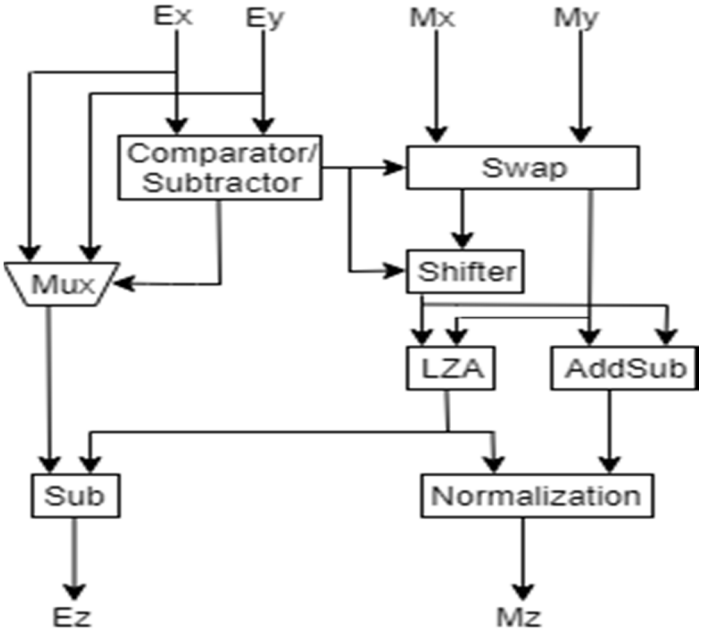


Fig.3. Architecture of a floating point addition/ subtraction

IV. PROPOSED METHOD

A. *Kogge Stone Adder*

These adders include the execution of an activity in parallel. This is finished by division the activity in littler parts which are ascertained in parallel. The result of the activity relies upon the fundamental data sources. Parallel Prefix Adder (PPA) can be indistinguishable to carry look forward snake (CLA). A CLA is a class of adder used in advanced rationale. CLA is developed to overcome the inactivity presented by repulsing result of carry bits in RCA. A CLA raises speed by diminishing convey bits. It processes at least one carry bits by the aggregate, which diminishes the hold up time to figure the result of greater piece esteem. CLA uses the idea about generating 'G' and propagating 'P' carries. The 2 vary in the strategy their convey age square is executed. Conditions used to create carry in CLA is known as:

$$C_i = G_{i-1}\ OR\ (P_{i-1}\ AND\ C_{i-1}) \quad (1)$$

The primary advantage of PPA is the convey diminishes the quantity of rationale levels by fundamentally creating the conveys in parallel. PPA speediest viper by focus on configuration time and is the alternative of incredible execution snake in industry.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 6 Issue VII, July 2018- Available at www.ijraset.com

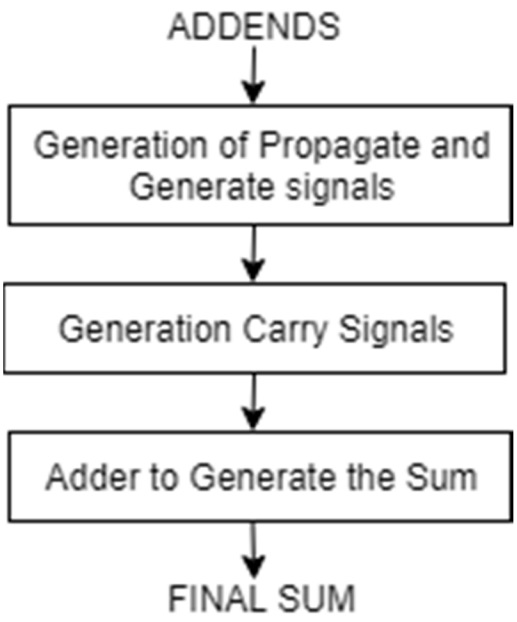


Fig 4 Structure of Parallel Prefix Adder

Fig.4. demonstrates the arranging of PPA. Kogge Stone Adder (KSA) is a PPA. It is considered as quickest and is expansively utilized as a part of industry for awesome execution number-crunching circuits. KSA handle the 3-stage configuration of the CLA adder, the improvement is in the carry delivering stage which is the most serious one. In KSA carries are ascertained quick by computing the carries in parallel. This is by and large attractive to utilize a adder with adequate planning, area and productivity exchange off. The carry computation technique experiences to accelerate the aggregate activity fundamentally. This decreases the area and increment the speed, the structure of KSA contain three preparing stages. KSA is a parallel prefix kind of CLA. This produces carry in O (log n) time and is generally regarded as the speediest viper and is typically used in the business for vast execution number-crunching circuits. In this KSA, carries are figured quick by ascertaining them in parallel at the cost of expanded area. It has three creating stages for figuring the sum bits. Functioning of KSA is:

1) *Pre-processing:* In this progression the calculation of propagate and generate signals related too every combine of bits in 'An' and 'B'. Those signs are appeared with the rationale conditions given underneath:

$$P_i = A_i \; xor \; B_i \qquad (3)$$
$$G_i = A_i and \; B_i \qquad (4)$$

2) Carry look ahead system: This block isolates KSA from different adders and is the principle control behind its awesome execution. This progression includes estimation of conveys resulting to each piece. It uses assemble proliferate and produce as centre signals which are appeared by the rationale conditions beneath:

Black Cell. The black cell holding 2 sets of engender and produce signals (G_i, P_i) and (G_j, P_j) as information and figures a couple of proliferate and create signals (G,P) as yield

$$G = G_i \text{ OR } (P_i \text{ AND } G_j) \qquad (5)$$
$$P = P_i \text{ AND } P_j \qquad (6)$$

Gray Cell. The gray cell holding 2 sets of engender and produce signals (G_i,P_i) and (G_j, P_j) as information sources and Calculates a create flag 'G' accordingly

$$G = G_i \text{ OR } (P_i \text{ AND } G_j) \qquad (7)$$

3) Post processing: This is the last advance and is regular to all adders of this family. It includes figuring of aggregate bits. Total bits are computed by the rationale demonstrated as follows:

$$S_i = P_i \text{ xor } C_{i-1}.$$

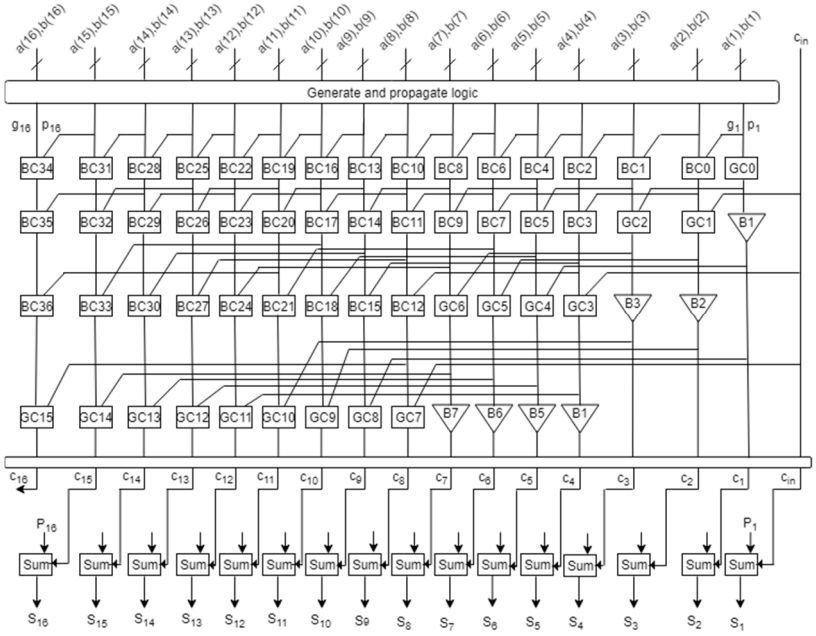


Figure 5: Schematic of 16-bit Kogge Stone Adder

V. EXPERIM ENTAL RESULTS

Design	No. of Slices	No .of FFs	No. of 4i/p LUTs	Delay (ns)	Power (mW)	Frequency (M Hz)
Existing[19]	4392	857	8222	17.597	197	56.829
Proposed	4214	857	7868	17.155	196	58.292

Table.1.comparision of results

The proposed work is designed in verilog HDL and synthesis is done using Xilinx ISE tool. The results of existing and proposed architecture is demonstrates in Table 1. In Xilinx ISE tool synthesis was implemented for Vertix-4. The above Table shows the comparison of no. of LUTs, no. of flip flops, no. of slices, delay, power and frequency.

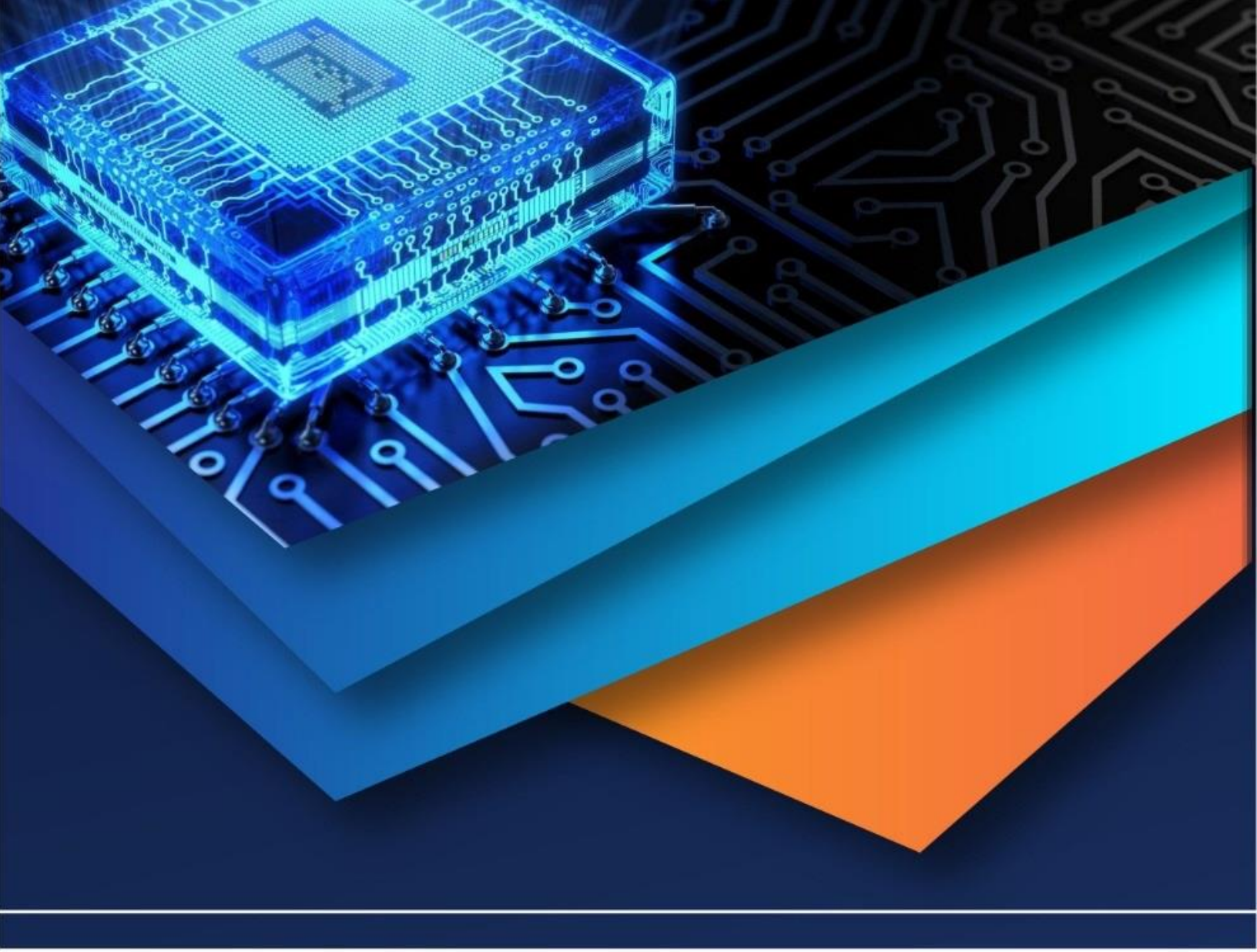
VI. CONCLUSION

In this succinct, an effective term-dividing plan has been appeared, that permits executing the hardware for convolution administrators, normally utilized in FILTERS, without multipliers, encoders. These are completely supplanted by streamlined adders and ROMs for putting away premultiplied coefficients. The proposed arrangement gets best in class exhibitions. The outcome is well reasonable for the utilization of multiconstant multiplication methods, with a specific end goal to additionally diminish the circuital topology.

REFERENCES

[1] S. L. Chen, "VLSI implementation of an adaptive edge-enhanced image scalar for real-time multimedia applications," IEEE Trans. Circuits Syst. Video Technol., vol. 23, no. 9, pp. 1510–1522, Sep. 2013.
[2] F. C. Huang, S. Y. Huang, J. W. Ker, and Y. C. Chen, "High performance SIFT hardware accelerator for real-time image feature extraction," IEEE Trans. Circuit Syst. Video Technol., vol. 22, no. 3, pp. 340–351, Mar. 2012.
[3] G. D. Licciardo, A. D'Arienzo, and A. Rubino, "Stream processor for realtime inverse tone mapping of full-HD images," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 23, no. 11, pp. 2531–2539, Nov. 2015.
[4] M. Vigliar and G. D. Licciardo, "Hardware coprocessor for stripe-based interest point detection," US Patent 20 130 301 930, Nov. 14, 2013.
[5] K. K. Parhi, VLSI Signal Processing Systems: Design and Implementation. New York, NY, USA: Wiley, 2007.
[6] B. C. Paul, S. Fujita, and M. Okajima, "ROM-based logic (RBL) design: A low-power 16 bit multiplier," IEEE J. Solid—State Circuits, vol. 44, no. 11, pp. 2935–2942, Nov. 2009.
[7] S. Y. Park and P. K. Meher, "Low-power, high-throughput, and low-area adaptive FIR filter based on distributed arithmetic," IEEE Trans. Circuits Syst.—II, Exp. Briefs, vol. 60, no. 6, pp. 346–350, Jun. 2013.
[8] R. M. Hewlitt and E. S. Swartzlantler, 1"Canonical signed digit representation for FIR digital filters," in Proc. IEEE Workshop Signal Process. Syst., Lafayette, LA, USA, Oct. 2000, pp. 416–426.

[9] K. Tsoumanis, N. Axelos, N. Moshopoulos, G. Zervakis, and K. Pekmestzi, "Pre-encoded multipliers based on non-redundant radix- 4 signed-digit encoding," IEEE Trans. Comput., vol. 65, no. 2, pp. 670–676, Feb. 2016.
[10] A. Peled and B. Liu, "A new hardware realization of digital filters," IEEE Trans. Acoust., Speech, Signal Process., vol. ASSP-22, no. 6, pp. 456–462, Dec. 1974.
[11] Y. Voronenko and M. Püschel, "Multiplierless multiple constant multiplication," ACM Trans. Algorithms, vol. 3, no. 2, pp. 1–39, May 2007.
[12] L. Aksoy, P. Flores, and J. Monteiro, "Efficient design of FIR filters using hybrid multiple constant multiplication on FPGA," in Proc. IEEE 32nd ICCD, Oct. 2014, pp. 42–47.
[13] A. Berkeman, V. Owall, and M. Torkelson, "A low logic depth complex multiplier using distributed arithmetic," IEEE J. Solid-State Circuits, vol. 35, no. 4, pp. 656–659, Apr. 2000.
[14] E. O'Shea, "Bachet's problem: As few weights to weigh them all," ArXiv e-prints, Oct. 2010.
[15] IEEE Standard for Binary Floating—Point Arithmetic, Amer. Nat. Std. Inst. (ANSI), Washington, DC, USA, IEEE 754—1985, 1985.
[16] M. Vigliar and G. D. Licciardo, "Multiplierless coprocessor for Difference of Gaussian (DOG) calculation," US Patent 20 130 301 950, Nov. 14, 2013.
[17] G. H. Hardy and E. M. Wright, An Introduction to the Theory of Numbers (Sixth Edition). New York, NY, USA: Oxford Univ. Press, 2008.
[18] S. K. Park, "The r-complete partitions," Discrete Math., vol. 183, no. 1–3, pp. 293–297, Mar. 1998.
[19] Gian Domenico Licciardo; Carmine Cappetta; Luigi Di Benedetto; Mario Vigliar "Weighted Partitioning for Fast Multiplierless Multiple-Constant Convolution Circuit" IEEE Transactions on Circuits and Systems II: Express Briefs Year: 2017, Volume: 64, Issue: 1, Pages: 66 - 70
[20] Bujjibabu Penumutchi; Satyanarayana Vella ; Harichandraprasad Satti "Kogge Stone Adder with GDI technique in 130nm technology for high performance DSP applications" 2017 International Conference On Smart Technologies For Smart Nation (Smart Tech Con) Year: 2017 Pages: 5 - 10



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY