A Low Power and High Speed Array Multiplier Using On-The-Fly Conversion

S Aruna, S Venkatesh, K.Srinivasa Naik

Abstract:A low power and high speed On-The-Fly Conversion (OTFC) array multiplier is proposed with optimum design resulting in reduced delay, low power intake and dwindled silicon area. In the multiplier design (single precision truncated) recommended earlier, the product of 2N-bits produces 2N but partial products, excluding this 2N bit partial products, are going to be divided into 2N-(N/2) bits and N/2 bits. As a result finally, 2N bits are created by the adding of above bits using ripple carry adder. The array multiplier outlined in this paper is designed and implemented with no truncation or addition technique, instead, it is executed using a typical array multiplier scheme. The proposed array multiplier in this paper produces the high order bit (MSB) of the final product. The multiplier design outlined in this paper leverages the On the Fly Conversion converter that is implemented at the tail end of the multiplier. This is to achieve the expedited carry propagation in the last leg of the multiplication. To highlight and contrast the benefits of the proposed array multiplier we have considered the previous designs proposed for different bits (8, 16 and 32) for features and critical parameters like silicon area, delay and power. As part of the implementation, we are able to attain remarkable results with low power consumption, minimum delay, smaller area and less energy.

Keywords-Array multiplier, Truncation, OTF Conversion, Ripple Carry Adder

I. INTRODUCTION

Binary multipliers in the digital components widely used in the microprocessors, microcontrollers play a critical role. Historically multipliers have been complex and taxing on the efficiency. The process as such entails multiplication and addition and is slow as it involves multiple intermediary steps. Complex operations lead to greater power consumption and decreased speeds. Multipliers being core components of processors, optimizing designs and layouts for low power and rapid multipliers is of paramount importance. Low electricity consumption is likewise a vital difficulty in multiplier layout.To lessen giant electricity intake it is ideal to lessen the variety of operations, thereby decreasing dynamic power which contributes to a greater part of overall power consumption.

To tackle these challenges, a high speed and optimal consuming array multipliers is the need of the hour. The performance and accuracy of any system depend on the robustness of the critical components, that's a multiplier in most of those sorts of applications.

When a couple of n bit numbers are multiplied using a multiplier it yields a 2n bit result. To retain the full output and precision of the operation, Digital Signal Processor design would have to accommodate an ever increasing bit width which is unfeasible. To overcome the challenge of reducing the bit width Truncated multipliers are generally employed [1], [2].

The advantage of these multipliers is that some cells could be switched off when the result is known. Multiplier with truncation scheme produces (n + k) significant columns (MSB) instead of 2n. To compensate for the truncation, the error correction method is used and can be contained by using different schemes [3]. The truncation error when under the acceptable tolerance (< Ulp/2) is still applicable in the majority of DSP implementations. The combination of the multiplier and the truncation block along with the ripple carry addition component make up for the error in approximation during the addition step [3],[4]. These implementations with partial product reduction come with a shortcoming of limited applicability in DSP applications. The major concern here is the delay that is introduced during the addition operation. To tackle this challenge a full precision multiplier with On-The-Fly Conversion is being proposed and implemented which could be applied for multiplication operations [5]-[7].

Design and features of High Precision Multipliers along have been contrasted in this paper. We have evaluated our proposal pertinent to latency and magnitude of the multipliers. Following which, the design was gauged for constraints, power and area. CMOS 45 nanometer (45 nm) process has been leveraged in the RTL Compiler for logic synthesis using Cadence Tool. Physical design has been implemented via Encounter tool.

This paper has been segmented to detail the below work:

1) Multiplier design has been optimized for less Delay Area, lower consumption of Power and Energy in comparison to Other Multipliers.

2) Constant and consistent delay generation and faster carry propagation summation has been achieved by means of The OTF (On The Fly) Conversion Logic and irrespective Of the number of input bits [6].

3) Comparative analysis of size and delay of the proposed design versus current designs have been laid out [9].

In the below section details about the Multiplier, low power consuming, leveraging the algorithm that leveraging the algorithm that executes using the left to right method, in turn, resulting in reducing the overlay of fractional products in the last carry and save/total generating step. Section 3 in this paper explains the current state of the multipliers.

Revised Manuscript Received on February 11, 2019.

Dr. S Aruna, ECE Department, Andhra University college of Engineering (A), Visakhapatnam, India (Email: aruna9490564519@gamil.com)

Mr. S Venkatesh, ECE Department, Andhra University college of Engineering (A), Visakhapatnam, India (Email: venkatesh949388@gmail.com)

Dr.K.Srinivasa Naik, ECE Department, Vignan's Institute of Information Technology (A), Visakhapatnam, India. (Email: nivas97033205@gmail.com)

Design and execution of the proposed Multiplier are covered in section 4. To conclude, the results of the analysis are laid out in section 5. Section 6 summarizes the proposal and findings.

II. BASIC ARRAY MULTIPLIER

Consider the number of 2 unsigned n-bits multiplier, where $X=X_{(n-1)}$, $X_{(n-2)},\ldots X_0$ is that the number And $Y=Y_{n-1}$, $Y_{n-2},\ldots..Y0$ is that the number. the product $P= P_{2n-1}$, $P_{2n-2}\ldots P0$ will be Written As Follows [10]-[12].

$P=\sum_{(I=0,\ N-1)} \sum_{(J=0,N-1)} (X_{i,}\ Y_{j})\ 2^{i+J}$

An example of an 8-bit array multiplier is shown in Figure 2.1 below.

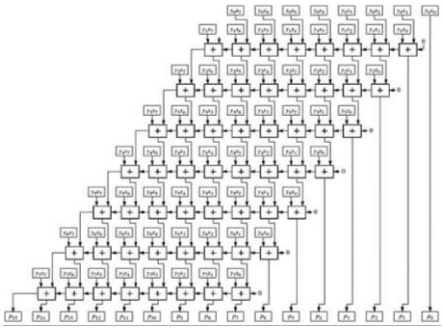

Figure 2.1:Architecture of 8-bit Array multiplier

To attain higher speeds of number crunching and data processing, a Systolic array multiplier network is proposed and employed. Each node in the array computes the partial result and passes it downstream [13].

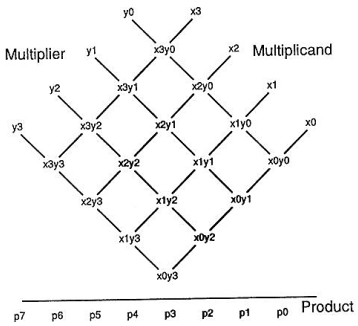

Figure 2.2: Methodology of systolic multiplication

Of the available two types of rounding in high-efficiency multipliers, the correctly rounded and faithfully rounded, faithfully rounded method has been considered for the Filter design as in Figure 2.2. With this type of rounding is generally tolerable in areas where the error is less than one unit (ULP) [2]. And the benefit is that the silicon area to produce such multipliers is noticeably lesser than the typical versions.

Multipliers with result truncated generally yield to a smaller footprint of the area required which leads to lower energy consumption and cost savings. When a multiplication operation has conducted this results in a width that is double that of the original bit width. Larger the input bit length wider the result. This typically calls for truncation of the result to the acceptable precision which is within the accepted range of less than 1Ulp. Additional correction circuits have been used to decrease the error caused by truncation.

Multipliers used in multifaceted DSP designs would have an advantage from the faithful rounding by truncating a slice of the partial product to reduce the power consumed.

III. CURRENT STATE

When a couple of N bit numbers are multiplied together, using the truncated Multiplier method, the resulting output is a 2N bit array. The value stored in the Least Significant Position of the result (LSP) is removed to achieve the Truncation in the multiplying operation. An Efficient and effective hardware multiplication of The Two Inputs bits A and B is then preserved in Column K as LSP Minor.

The produced carry is then totaled, LSP minor column bits are ignored, the final result is the approximate value of the actual product by leveraging the truncation black attached in the last leg of the multiplier scheme is denoted in below Figure 3.1.

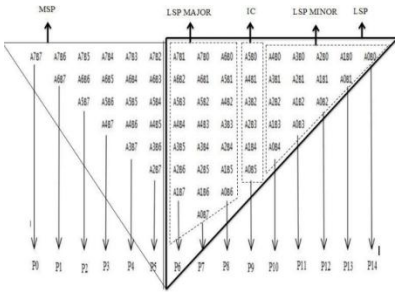

Figure 3.1: Reduction array of truncated multiplier

This technique is used to remove the initial (n + k) columns of the partial multiplication array. Here (k), which is lower than n, is determined by the value of the truncation parameter used in the last stage of the multiplier. Gates and full adders have been employed in the Left to right multiplier scheme [14]. N0bit adder and Full adders are used in conjunction to produce the MSBs of the partial products in 2n bits at the very last stage of the multiplier. On the Fly, conversion technique is used to parallelize the operation and to ignore the final adder.

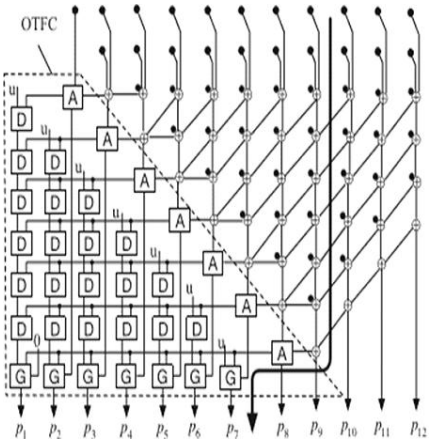

Figure 3.2: single precision truncated multiplier

Figure 3.2 show left to right truncated multiplier [14], [15].In every, and each stage of and gates, the produces the

product so the total adder produces the add and carry throughout the partial product reduction method. The left module within the last stage of the multiplication within the dotted line box below represented in the final step of the array multiplication is the OTFC converter [16]. The sum of X & y the performed by A cell and to produce the sum Zi,0 and carry Zi,1 [17]. As shown in Figure 3.3.

A-cell produces

$Z_{i,0}$=X Y

$Z_{i,1}$=X.Y

The decision to transmit the carry from the LSP (low bit position) is performed by the D cell as shown in Figure 3.4. Dpi is set for propagating and Dgi is set as the generation signal. The index in the array truncation is "i" if Dp is set to 1 carry propagated in enabled and disabled when dpi anddgi are set to zero [20]-[23].

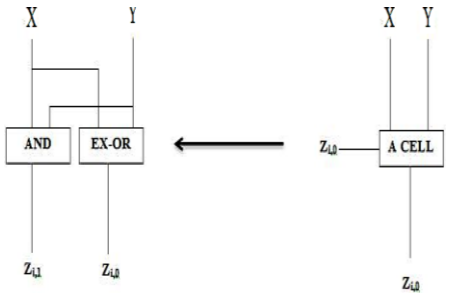

Figure 3.3: A-cell

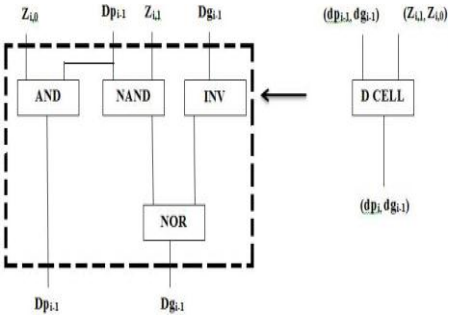

Figure 3.4: D-cell

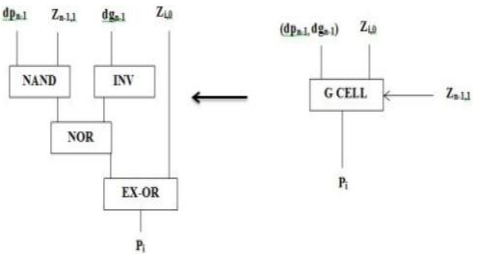

Figure 3.5: G-cell

As dgiis set to 1, D-cell produces the carryout. The output ($Z_{i,0}$, $Z_{i,1}$),from the A-cell, and (Dp_{i-1},Dg_{i-1}), from D – cell are used as inputs for the next stage. (1, 0) is produced as the very first values of Dp_0 and Dg_0, this signifies the carry propagation to On the Fly conversion converter and the value of the result is unknown. D-cell is represented as

$$Dp_i = Dp_i.Z_{i,0}$$

$$Dg_i=\overline{Dp_{i-1}.Z_{i,1}}+\overline{Dg_{i-1}}$$

The propagation and generate signals from the previous steps are used to generate the output of D-cell. The final D row cells produce the Dp_{n-1}& Dg_{n-1} bits. The MSB (Most Significant Part) of the product of the input bits is produced by the G-cells. As shown in Figure 3.5.

$$P_i=z_{i,0}\ (dp_{n-1}.z_{n-1}+dg_{n-1})$$

A-cell, D-cell and G-cell execute to provide the ultimate bits generated by the products. The n+1 bit array multiplication factor entails (n-1) A-cells, (n2-3n+2) / 2D-cells [24] and (n-1)G-cells [25]. The projected array multiplier factor is explained in section 4 that focuses on reducing the dimensions of the on-the-fly-conversion.

IV. ARRAYMULTIPLIER WITH PROPOSED OTFC

Given the ever-growing need for high speed microdevices and processors with less an optimal combination of delay, area and power consumption, the proposed digital multiplier takes partial and resultant products into consideration.

This proposal details about one of the possible ways to strike a balance between the critical features, accuracy, efficiency, performance and cost. Constant and consistent delay generation and faster carry propagation summation for the array multiplier has been achieved by means of The OTF (On The Fly) Conversion Logic and irrespective of the number of Input Bits. As shown in Figure 4.1.

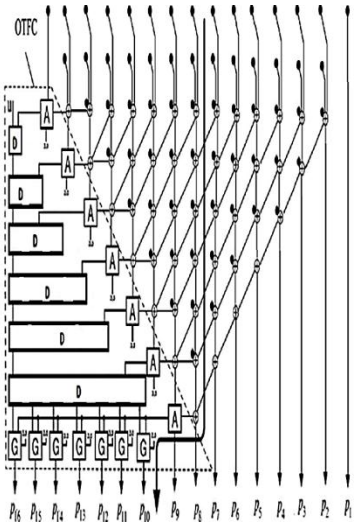

Figure 4.1: Array multiplier with OTFC

The sum of X & Y the designed A cell and to produce the sum $Z_{i,0}$ and carry $Z_{i,1}$. As shown in Figure 4.2.

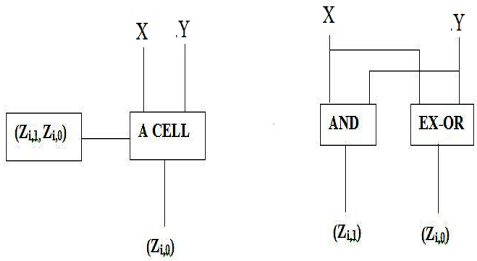

Figure 4.2: designed A-cell

Here Z(i,0)=X Y.

Z(i,1)=X.Y .

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

The decision to transmit the carry from the LBP (Lower Bit Position) is performed by the D cell. Dpi is set for propagating and dgi is set as the generation signal. The index in the array truncation is 'i'. if dp is set to 1 carry propagation is enabled and disabled when dpi and dgi are set to zero. As shown in the below Figure 4.3.

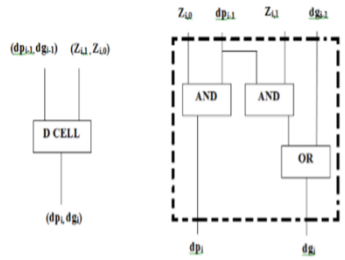

Figure 4.3: Designed D-cell

Here

$Dp_i = Z_{(i,0)}.Dp_{(i-1)}$.

$Dg_i = Z_{(i,1)}.Dp_{(i-1)} + Dg_{(i-1)}$.

The interim outputs of all the D - cells are passed on to the next stages of the D - cells as inputs. n-1 the D - cell result is passed into the G - cell as inputs. As shown in the below Figure 4.4.

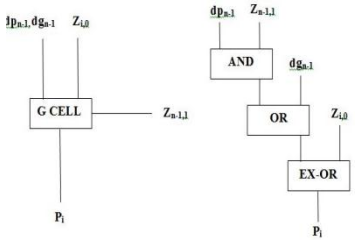

Figure 4.4: Designed D cell

The G-Cell Produces the ultimate Partial Product Of OTFC.

$Pi=(Z_{(1,n-1)}, Dp_{(n-1)}+Dg_{(n-1)}) Z_{(i,0)}$.

V. SYNTHESIS RESULT

The proposed high speed array multiplier has been developed in Cadence tool using the CMOS 45nm technology. The existing versions of the multipliers listed above have been developed likewise to contrast the results. Results have been collected for the crucial features area, energy consumption (power times the delay) and delay for 8,16 and 32-bit inputs and are detailed in the below tables (1,2,3) for observation.

TABLE 1

Comparison of synthesis results for 8-bit proposed array multiplier with On-The-Fly Conversion

Types	Gate Count	Area (μM^2)	Power (Mw)	Delay (Ns)	Energy (PJ)
Proposed	**256**	**1252**	**3.259**	**3.14**	**10.233**
Left-to-right	**273**	**1398**	**3.410**	**3.14**	**10.707**
Array	**349**	**1728**	**3.488**	**3.245**	**11.318**
Wallace	**309**	**1295**	**3.374**	**3.245**	**10.948**
Vedic	**276**	**1474**	**3.310**	**3.145**	**10.409**

Table

TABLE 2

Comparison of synthesis results for 16-bit proposed array multiplier with On-The-Fly Conversion

Types	Gate Count	Area (μm^2)	Power (mW)	Delay (ns)	Energy (PJ)
Proposed	**1118**	**5349**	**9.893**	**3.14**	**31.064**
Left-to right	**1244**	**6305**	**10.732**	**3.14**	**33.693**
Array	**1487**	**7321**	**11.431**	**3.245**	**37.093**
Wallace	**1327**	**5739**	**10.540**	**3.245**	**34.202**
Vedic	**1174**	**6256**	**10.075**	**3.145**	**31.685**

Table

TABLE 3

Comparison of synthesis results for 32-bit proposed array multiplier with On-The-Fly Conversion

Types	Gate Count	Area (μm^2)	Power (mW)	Delay (ns)	Energy (PJ)
Proposed	**4942**	**23459**	**17.426**	**3.14**	**54.717**
Left-to right	**5389**	**26726**	**32.754**	**3.14**	**102.847**
Array	**6352**	**31257**	**36.121**	**3..245**	**117.212**
Wallace	**5906**	**25158**	**20.549**	**3.245**	**66.668**
Vedic	**5319**	**26891**	**17.840**	**3.145**	**56.106**

The physical implementation of the proposed design has been achieved through the Encounter tool. As mentioned earlier, our proposed design of the array multiplier features less delay, smaller footprint in the area, low power consumption leading to less energy usage as well. After the front-end design, verification of the multiplier was done for the physical design optimized outputs from the previous steps are passed into. I the verification process effort and iterations for attaining the desired specifications was done to get the transformed result. This result is passed into the Physical design, encounter tool, as the starting point. Floor planning, Power plan, and Placement design considerations are done in each of these stages to meet the eventual optimization proposed. As part of this process, multiple iterations have been performed to fine tune each of the steps and obtain the poise between the paralleled features.

As below shown in Figure 5.1. The PD (Physical Design) implementation. This is the end result of the different verification steps like the tuning of timing and routing. Leveraging the placement technique we placed all the cells on the Integrated Circuit.

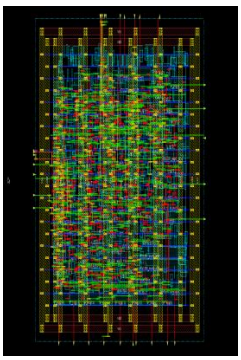

Figure 5.1: Physical Design of Array Multiplier

VI. CONCLUSION

This paper proposes an Array bit Multiplier that is highly efficient and optimized for less delay, lower silicon area, and low Power and energy consumption. To increase the efficiency of carrying propagation in the final step of the multiplier OTF conversion logic has been implemented which also provides constant delay irrespective of the input bits processed. As noted earlier in the paper, the proposed multiplier could be implemented in various other techniques, like CMOS 32nm and 28nm, 7nm etc. Advanced and still in research technology like FIN FET and CNT FET could also be considered.

VII.ACKNOWLEDGMENT

We would like to express our gratitude to the Department of Electronics and Communication Engineering, Andhra University, Visakhapatnam for their valuable guidance, advice, and support. We thank Dr.Sanath Kumar, Asst. Director, CITD (Central Institute of Tool Design) for providing training and access to design software Cadence to successfully complete this proposal.

REFERENCES

[1] A flexible low power DSP with a programmable truncated multiplier by (Manuel.D. la Guia Solaz, W.Han, and R. Conway), IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 59, no. 11, pp. 2555–2568, 2012.

[2] Hou - J. Ko And Shen- F. Hsiao, "Design And Application Of Faithfully Rounded And Truncated Multipliers With Combined Deletion, Reduction, Truncation, And Rounding," *IEEE Trans. Circuits Syst. II, Express Briefs*, Vol. 58, No. 5, Pp. 304–308, 2011.

[3] The o. A. Drane, T. M. Rose, and G. A. Constantinides, "On the systematic creation of faithfully rounded truncated multipliers and arrays," IEEE Trans. Comput., vol. 63, no. 10, pp. 2513–2525, 2014.

[4] Nicola. Petra, D. De Caro, V. Garofalo, E. Napoli, Anda. G. M. Strollo, "Truncated Binary Multipliers With Variable Correction And Minimum Mean Square Error," *IEEE Trans. Circuits Syst. I, Reg. Papers*, Vol. 57, No. 6, Pp. 1312–1325, Jun. 2010.

[5] Sang Man Moh and Suk Han Yoon "HIGH SPEED ARRAY MULTIPLIERS BASE on ON THE FLY CONVERSION" ETRL journal, volume 19,no. 4 dec 1997

[6] Milos. D. Ercegovac And Tomas Lang, "On-The-Fly Conversion Of Redundant Into Conventional Representation," *IEEE Trans. Comput.*, Vol. C-36, No. 7, Pp. 895–897, 1987.

[7] Mr. S Vasu Krishna and E Mahesh Kumar "A Novel Energy-Efficient Multiplier Using OTFC" IOSR Journal of VLSI and Signal Processing (IOSR-JVSP) Volume 8, Issue 2, Ver. I Mar.-Apr. 2018.

[8] Z. Shun, Oliver. A. Pfander, H. J. Pfleiderer, And A. Bermak, "A VLSI Architecture For A Run-Time Multi-Precision Reconfigurable Booth Multiplier," In *Proc. 14th IEEE Int. Conf. Circuits Syst.*, Pp. 975–978, 2007

[9] A. Vazquez And E. Antelo, "Area And Delay Evaluation Model For CMOS Circuits," Internal Report, University Santiago De Compostela (Spain), Jun. 2012.

[10] M. L. Hsia and O. T. C. Chen, "Low-power multiplier optimized by partial-product summation and adder cells," in Proc. IEEE Int. Conf. Circuits Syst., pp. 3042–3045, 2009.

[11] Manchall. Ahuja and Sakshi, "Design and analysis of bypassing multiplier," in Proc. 5th Int. Conf. ART Com, pp. 241–246, 2013

[12] Ming. C. Wen, S. Jyan. Wang, And Y. N. Lin, "Low Power Parallel Multiplier With Column Bypassing," In *Proc. IEEE Int. Symp.Circuits Syst.*, Pp. 1638–1641, 2005

[13] Jia Jia. Chen, C. H. Chang, F. Feng W. Ding, and J. Ding, "Novel design algorithm for low-complexity programmable FIR filters based on an extendeddouble-base number system," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 62, no. 1, pp. 224–233, 2015.

[14] H. Bessalah, K. Messaoudi, M. Issad, N. Anane, and M. Anane, "Left-to-right serial multiplier for large numbers on FPGA," in Proc. IEEE ICM, pp. 1–6, 2009.

[15] Z. Huang and M. D. Ercegovac, "High-performance low-power left-to-right array multiplier design," IEEE Trans. Comput., vol. 54, no. 3, pp. 272–283, Mar. 2005.

[16] N. Takagi and T. Horiyama, "A high-speed reduced-size adder under left-to-right input arrival," IEEE Trans. Comput., vol. 48, no. 1, pp. 76–80, Jan. 1999.

[17] Valeria. Garofalo, N. Petra, And E. Napoli, "Analytical Calculation Of The Maximum Error For A Family Of Truncated Multipliers Providing Minimum Mean Square Error," *IEEE Trans. Comput.*, Vol. 60, No. 9, Pp. 1366–1371, Sep. 2011

[18] Sabya. Das And Sunil P. Khatri, "Generation Of The Optimal Bit-Width Topology Of The Fast Hybrid Adder In A Parallel Multiplier," In *Proc. IEEE Int. Conf. Integr. Circuit Des. Technol.*, Pp. 1–6, May 2007.

[19] Z. Shun, Oliver. A. Pfander, H. J. Pfleiderer, And A. Bermak, "A VLSI Architecture For A Run-Time Multi-Precision Reconfigurable Booth Multiplier," In *Proc. 14th IEEE Int. Conf. Circuits Syst.*, Pp. 975–978, 2007

[20] Valeria. Garofalo, N. Petra, And E. Napoli, "Analytical Calculation Of The Maximum Error For A Family Of Truncated Multipliers Providing Minimum Mean Square Error," *IEEE Trans. Comput.*, Vol. 60, No. 9, Pp. 1366–1371, Sep. 2011

[21] L. Ciminiera And Paolo. Montuschi, "Carry-Save Multiplication Schemes Without Final Addition," *IEEE Trans. Comput.*, Vol. 45, No. 9, Pp. 1050–1055, 1996.

[22] S. Rong. Kuang and J. P. Wang "Design of power-efficient configurable booth more multiple" IEEE Trans. Circuits Syst. I, Reg.Papers, vol. 57,no. 3, pp. 568–580, 2010.

[23] Z. Yu, L. Wasserman, and A. N. Willson, Jr., "A painless way to reduce power dissipation by over 18 Booth-encoded carry-save array multipliers for DSP," in Proc. IEEE Workshop Signal Process. Syst., pp. 571–580, 2000.

[24] Wen Yan, Milos D. Ercegovac, He Chen "An Energy Efficient Multiplier With Fully Overloaded Partial Products Reduction And Final Addition "*IEEE Trans. Circuits Syst. I, Regular Papers Express Briefs*, 2016.

[25] Mr. S Vasu Krishna and E Mahesh Kumar "A Novel Energy-Efficient Multiplier Using OTFC" IOSR Journal of VLSI and Signal Processing (IOSR-JVSP) Volume 8, Issue 2, Ver. I Mar.-Apr. 2018.