Industrial Engineering Journal
ISSN: 0970-2555
Volume : 48, Issue 1, No. 1, January 2019

DESIGN OF THE VLSI MULTIPLIER WITH REDUNDANT COMPRESSORS

Pamisetty Chandana[1], N. Yerriswami[2], S.Shabbir Ali[3]
[1]Assistant Professor, [2]Assistant Professor, [3]Assistant Professor, ECE Department, Anantha Lakshmi Institute of Technology and Sciences, Ananthapuramu, Andhra Pradesh, India.

ABSTRACT:

As technology scaling is reaching its limits,new approaches have been proposed for computational efficiency. Approximate computing is a promising technique for high performance and low power circuits as used in error-tolerant applications. Approximate arithmetic designs have attracted significant research interest among approximate circuits. In this paper, we have designed approximate redundant binary multiplier. In the existing method two approximate Booth encoders and two RB 4:2 compressors based on RB (full and half) adders are proposed for the RB multipliers. The approximate design of the RB-Normal Binary (NB) converter in the RB multiplier is also studied by considering the error characteristics of both the approximate Booth encoders and the RB compressors. The existing approximate RB multipliers are compared with previous approximate Booth multipliers. In the proposed method to reduce the circuit complexity and area we have used a new approximate redundant binary compressor used in approximate redundant binary multiplier. New Approximate RB multipliers are better than approximate NB Booth multipliers especially when the word size is large.

Index Terms— approximate computing, redundant binary (RB) multiplier, modified Booth encoder, RB compressor, RB-NB converter, partial product array, low power.

INTRODUCTION

Multiplication is one of the basic functions in all digital circuits. It is widely used in digital signal processing (DSP) applications. Hardware resourcesthan addition and subtraction. There are two different kinds of multiplication algorithms known as, serial multiplication algorithms and parallel multiplication algorithm. Serial multiplication schemes are widely used in sequential circuits, it contain feedback loop. Parallel multiplication algorithms often used in combinational circuits, it does not contain feedback structures. This paper presents various multiplier architectures. Multiplier architectures generally classified into two categories, one is "tree" multipliers and another one is "array" multipliers. Treemultipliers add as many partial products in parallel as possible and therefore, are very high performance architectures.

utilization and processing time required by it are more

Approximation of operands: Multiplication using approximate operands was first proposed by Mitchell with the concept of a logarithmic multiplier (LM). LM performs multiplication using only shifting and addition by converting the operands to approximate logarithmic numbers. Although the complexity of LM is significantly reduced compared with a conventional multiplier, it results in large errors. Recent designs of LMs aim to improve accuracy using fine piecewise linear approximation or iterative techniques. The use of approximate operands is further developed by an error-tolerant multiplier (ETM) and a dynamic range unbiased multiplier DRUM). Logarithmic Multiplier performs multiplication using only shifting and addition by converting the operands to approximate logarithmic numbers. Recent designs of LMs aim to improve accuracy using fine piecewise linear approximation or iterative techniques. The use of approximate operands is further developed by an error-tolerant multiplier (ETM) and a

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 48, Issue 1, No. 1, January 2019

Dynamic range unbiased multiplier (DRUM).

Approximation of PP tree: An inexact array multiplier has been proposed by ignoring some of the least significant columns of the PPs as a constant. A truncated multiplier has been proposed with a correction constant that is selected according to both the reduction and the rounding errors. However, this truncated multiplier has a large error if the PPs in the least significant columns are all ones or all zeros. Therefore, a truncated multiplier with variable correction has been proposed. Recently some error compensation strategies have been proposed to further improve the accuracy of fixed-width Booth multipliers. The error is compensated with the outputs of Booth encoders. The error compensation circuit proposed mainly uses a simplified sorting network.

Approximation of compressors: An inexact 4:2 counter has been used to design an approximate 44 Wallace multiplier that is further used to build larger size multipliers. Approximate 4:2 compressors have been proposed and used in a Dadda tree of 88 array multipliers. An 88 multiplier using approximate adders that ignore the carry propagation between PPs, has been proposed. Four multipliers are designed based on the approximate 4:2 compressors. Improved approximate 4:2 compressors have been proposed.

Case studies with R4ARBM applied to FIR filtering and high dynamic range (HDR) image processing are also provided. This paper has been extended significantly from its previous conference version [37]. The main differences are summarized as follows:

_ A new approximate Booth encoder with six errors in the K-map is proposed;

_ A new approximate RB 4:2 compressor at a smaller complexity is proposed;

_ For small approximate factors, rather than achieving a regular PP array by ignoring the correction term, an exact regular PP array is designed by combining the correction termsinto the PPs using logic optimization for more accurate results;

_ New approximate RB-NB converters are proposed;

_ Case studies are provided with applications to FIR filters, k-mean clustering and HDR image processing.

LITERATURE REVIEW

[1] J. Han and M. Orshansky et.al approximate computing has recently emerged as a promising approach to energy-efficient design of digital systems. Approximate computing relies on the ability of many systems and applications to tolerate some loss of quality or optimality in the computed result. By relaxing the need for fully precise or completely deterministic operations, approximate computing techniques allow substantially improved energy efficiency. This paper reviews recent progress in the area, including design of approximate arithmetic blocks, pertinent error and quality measures, and algorithm- level techniques for approximate computing.

[2] S. Venkataramani, S. Chakra Dhār et.al Diminishing benefits from technology scaling have pushed designers to look for new sources of computing efficiency. Multicores and heterogeneous accelerator- based architectures are a by-product of this quest to obtain improvements in the performance of computing platforms at similar or lower power budgets. In light of the need for new innovations to sustain these improvements, we discuss approximate computing, a field that has attracted considerable interest over the last decade. While the core principles of approximate computing — computing efficiency by producing results that are good enough or of sufficient quality — are not new and

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 48, Issue 1, No. 1, January 2019

are shared by many fields from algorithm design to networks and distributed systems, recent efforts have seen a percolation of these principles to all layers of the computing stack, including circuits, architecture, and software.

Approximate computing techniques have also evolved from ad hoc and application specific to more broadly applicable, supported by systematic design methodologies.

[4] H. R. Mahdiani, A. Ahmadi et.al The conventional digital hardware computational blocks with different structures are designed to compute the precise results of the assigned calculations. The main contribution of our proposed Bio- inspired Imprecise Computational blocks (BICs) is that they are designed to provide an applicable estimation of the result instead of its precise value at a lower cost. These novel structures are more efficient in terms of area, speed, and power consumption with respect to their precise rivals. Complete descriptions of sample BIC adder and multiplier structures as well as their error behaviors and synthesis results are introduced in this paper. It is then shown that these BIC structures can be exploited to efficiently implement a three-layer face recognition neural network and the hardware de-fuzzification block of a fuzzy processor.

[5] X. Cui,W. Liu et.al Due to its high modularity and carry-free addition, a redundant binary (RB) representation can be used when designing high performance multipliers. The conventional RB multiplier requires an additional RB partial product (RBPP) row, because an error- correcting word (ECW) is generated by both the radix-4 Modified Booth encoding (MBE) and the RB encoding. This incurs in an additional RBPP accumulation stage for the MBE multiplier. In this paper, a new RB modified partial product generator (RBMPPG) is proposed; it removes the extra ECW and hence, it saves one RBPP accumulation stage. Therefore, the proposed RBMPPG generates fewer partial product rows than a conventional RB MBE multiplier. Simulation results show that the proposed RBMPPG based designs significantly improve the area and power consumption when the word length of each operand in the multiplier is at least 32 bits; these reductions over previous NB multiplier designs incur in a modest delay increase (approximately 5%).

EXISTING METHOD

Four approximate RB multipliers are designed in this section based on two approximate Booth encoders, two approximate RB 4:2 compressors, and an approximate RB-NB converter. Both exact and approximate regular PP arrays are used to meet the trade-off between accuracy and complexity.

The Approximate Booth Encoders:

Two approximate Booth encoders are designed based on the conventional modified Booth encoding method and the new modified Booth encoding method, respectively.

Radix-4 Approximate MBE:

The K-map of the radix-4 approximate modified Booth encoder (R4AMBE6), i.e., appij61, with 6 errors in the K-map is shown in Table 4, where 0 denotes an entry in which a '1' is replaced by a '0' and 1 denotes a '0' entry that has been replaced by a '1'. Therefore, three modifications change a '1' to a '0' and three modifications change a '0' to a '1' in the K-map. The output of R4AMBE6 is given as follows

$$app_{ij6-1} = (b_{2i} + b_{2i-1})(b_{2i+1} \oplus a_i)$$
$$E_i = (b_{2i+1}\overline{b_{2i}}) + (b_{2i+1}\overline{b_{2i-1}})$$

Compared with the exact MBE, R4AMBE6 can significantly reduce both the complexity and the critical path delay of Booth encoding. The gate level structure of R4AMBE6 is shown in Fig. 5. The conventional design of MBE (Fig. 2) consists of four XNOR-2 gates, one XOR- 2 gate, one OR-3 gate, one OR-2 gate and one NAND-2 gate. The R4AMBE6 design only requires

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 48, Issue 1, No. 1, January 2019

one XOR-2 gate, one AND-2 gate and one OR-2 gate.

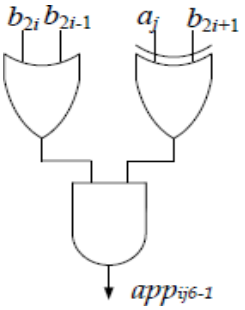

Figure 1: The gate-level circuit of the proposed R4AMBE6.

Radix-4 Approximate NMBE:
In this approximate design, there are more entries changed from '0' to '1' than those changed from '1' to '0'. Therefore, the approximate results produced by R4ANMB6 will be usually larger than its exact counterpart.

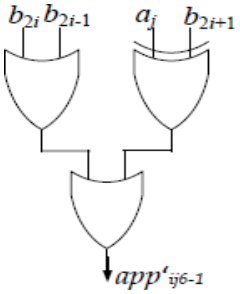

Figure 2: The gate-level circuit of the proposed R4ANMBE6.

Approximate RB 4:2 Compressors:
An efficient designs must ensure that the error between the approximate RB compressor and its exact counterpart remains as small as possible. Therefore, the following four types of compression results are equivalent: (0, 0) = (0, 0), (0, 1) = (1, 0), (1, 0) = (0, 1) and (1, 1) = (1, 1).
Approximate RB 4:2 Compressor 1:
Sk+ can be simplified by ignoring the asymmetric part of the exact RB compressor (ERBC) in Eq. (14). The first approximate RB compressor (ARBC-1) is given by the following expressions:

$$S_{k1}^{-} = x_k^{-} \oplus x_k^{+} \oplus y_k^{-} \oplus y_k^{+} \oplus ((x_{k-1}^{-} + x_{k-1}^{+})(y_{k-1}^{-} +$$

$$S_{k1}^{+} = x_{k-1}^{-} x_{k-1}^{+} + y_{k-1}^{-} y_{k-1}^{+}$$

The gate level circuit of the approximate RB compressor is given in Fig. 7. Theapproximate Sk1+ has only 3 gates, while the exact Sk1+ requires 12 gates.

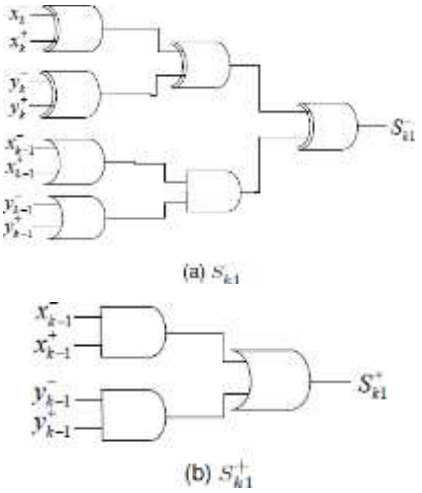

Figure 3: The gate level circuit of ARBC-1.

Approximate RB 4:2 Compressor 2:
Sk- and Sk+ can be further simplified as an approximate RB compressor. The second approximate RB 4:2 compressor (ARBC-2) is given by:

$$S_{k2}^{-} = x_k^{-} \oplus x_k^{+} \oplus y_k^{-} \oplus y_k^{+} \oplus (y_{k-1}^{-} + y_{k-1}^{+})$$

$$S_{k2}^{+} = y_{k-1}^{-} y_{k-1}^{+}$$

Also ARBC-2 generates results that are larger than its exact counterpart. The gate level design of the approximate RB compressor is given in Fig. 8. ARBC-2 further reduces the gate count of Sk2 from7 to 5. Therefore, ARBC-2 reduces the gate count from 19 to 6, which is significantly simpler than ERBC.

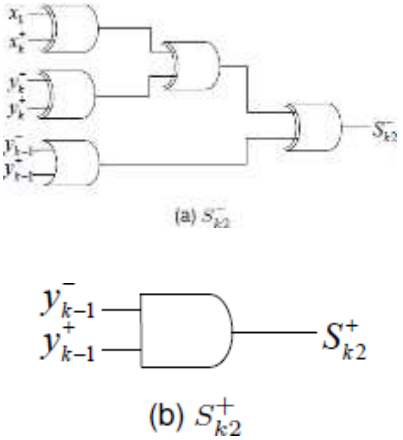

Figure 4: The gate level circuit of ARBC-2

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 48, Issue 1, No. 1, January 2019

PROPOSED METHOD

The Approximate Booth Encoders:

The K-map for the radix-4 approximate modified Booth encoder [1], here 0(circled) denotes the modification on which 1 is made 0 and 1 denotes that a circled 0 is replaced by 1.So that here 6 of the values are modifies to make the Booth encoder simple with making it approximate. The truth table is symmetrical is the main principle which brought the approximate design. So now the 0 to 1 and 1 to 0 replacements, 3 of each gives the output.

Radix-4 Approximate MBE:

The K-map of the radix-4 approximate modified Booth encoder (R4AMBE6), i.e., appij61, with 6 errors in the K-map is shown in Table 4, where 0 denotes an entry in which a '1' is replaced by a '0' and 1 denotes a '0' entry that has been replaced by a '1'. Therefore, three modifications change a '1' to a '0' and three modifications change a '0' to a '1' in the K-map. The output of R4AMBE6 is given as follows

$$app_{ij6-1} = (b_{2i} + b_{2i-1})(b_{2i+1} \oplus a_i)$$

$$E_i = (b_{2i+1}\overline{b_{2i}}) + (b_{2i+1}\overline{b_{2i-1}})$$

Compared with the exact MBE, R4AMBE6 can significantly reduce both the complexity and the critical path delay of Booth encoding. The gate level structure of R4AMBE6 is shown in Fig. 5. The conventional design of MBE (Fig. 2) consists of four XNOR-2 gates, one XOR-2 gate, one OR-3 gate, one OR-2 gate and one NAND-2 gate. The R4AMBE6 design only requires one XOR-2 gate, one AND-2 gate and one OR-2 gate.

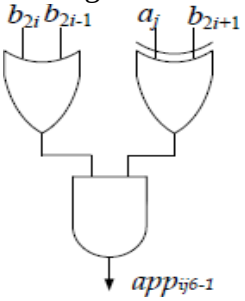

Figure 5: The gate-level circuit of the proposed R4AMBE6.

Radix-4 Approximate NMBE:

In this approximate design, there are more entries changed from '0' to '1' than those changed from '1' to '0'. Therefore, the approximate results produced by R4ANMB6 will be usually larger than its exact counterpart.

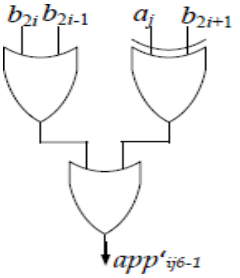

Figure 6: The gate-level circuit of the proposed R4ANMBE6.

New Proposed approximate redundant binary encoder:

In the proposed method a new approximate redundant binary multiplier is designed. As we are having several stages in existing method here too we have some stages like Approximation of operands, Approximation of PP generation, Approximation of PP tree, Approximation of compressors. At the first stage approximate redundant modified booth encoders are used to generate the partial products, in the second stage we have used the new approximate 4:2 compressors instead of approximate redundant binary 4:2 compressors. It consumes with less area because compared to existing compressors this compressors gate count is low. And in the last stage approximate RB-NB converter which is a simple NOR gate is used to convert from RB to NB, which accumulates the product.

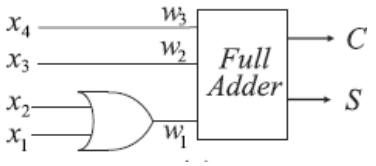

Figure 7: New approximate 4:2 compressors

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 48, Issue 1, No. 1, January 2019

Approximate RB-NB Converter:

As the approximate Booth encoders and approximate RB compressors generate results that are generally larger than the exact results, the biased approximate results can be compensated using ARNC with smaller values. The principle of compensation is to use an approximate adder that produces results that are smaller than its exact results. Therefore, the complexity of the RB-NB converter can be reduced, while the overall accuracy of the approximate RB multipliers is also increased. The truth table of a possible approximate RB-NB converter is given by Table 6, a simple NOR is gate used in the approximate RB-NB digit converter as follows:

$$S_k' = \overline{S_k^- + S_k^+}$$

The approximate RB multipliers using ARNC can reduce the error of an entire multiplier.

Results

Simulation results

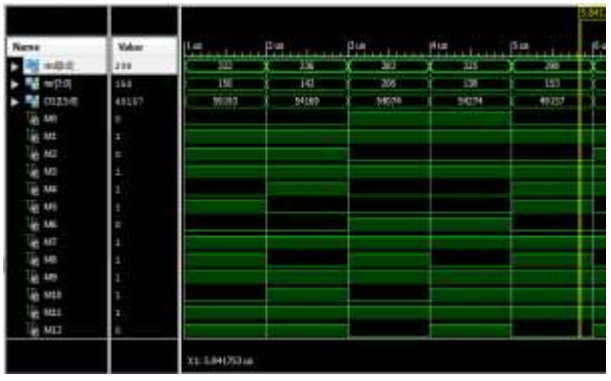

Figure 8: simulation wave forms of redundant binary multiplier

The above fig clearly explains about experimental outcomes of multiplier binary compressor. In this RB-4:2 compressor based booth encoder. Here all partial products and regular products are helps the approximation of RB multiplier design.

Base paper:

Figure 9: simulation results of RB booth multiplier

The above figure 9 clearly explains about experimental outcomes of RB Booth encoder design. This results suggest that it is useful for error-resilient applications.

Extension Results:

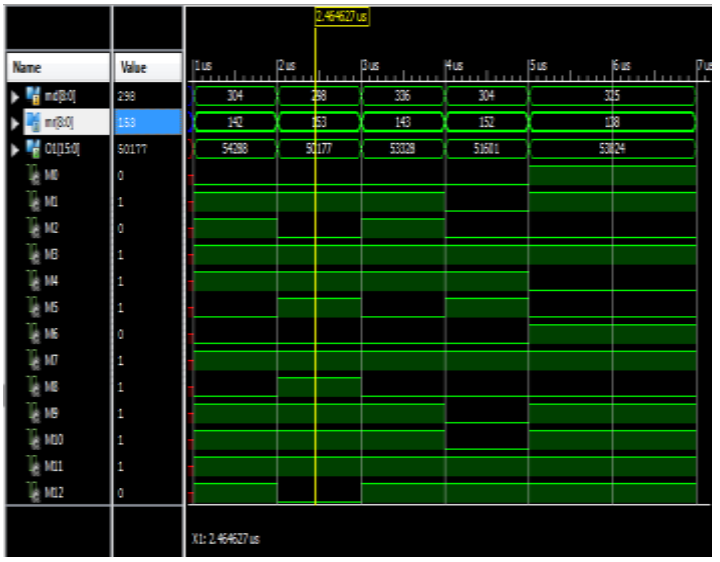

Figure 10: New approximate redundant binary compressor

Figure 10 clearly explains about a modern redundant approximate design, here all waveforms are clearly identified and getting improved results.

Table 1: performance analysis

	Area	**Delay**	**power**
ARB multiplier	40	5.947ns	0.321
Extension	29	5.900ns	0.321

The above table 1 clearly explains about, area, Delay and power analysis of proposed and existed models. Here Area 29%, delay 5.9 ns and power 32 mw can be attained. These outcome are more improved, and outperforms the methodology.

CONCLUSION

The proposed approximate booth encoders on conventional exact booth encoders (R4AMBE6) and new exact booth encoders (R4ANMBE6) have moderate error and energy consumption; also they achieve a very good tradeoff between area and performance. The proposed new approximate RB 4:2 compressors reduce the energy consumption and area by over compared with approximate RB 4:2 compressors.

The proposed approximate redundant binary multipliers are efficient for error-tolerant applications with high accuracy.

REFERENCES

[1] J. Han and M. Orshansky," Approximate computing: an emerging paradigm for energy-efficient design," in Proc. 18th IEEE European Test Symposium, 2013, pp.1-6.

[2] V. Chippa, S. Chakradhar, K. Roy and A. Raghunathan," Analysis and characterization of inherent application resilience for approximate computing,"in Proc. 50th Annual Design Automation Conference (DAC), 2013, Article 113, 9 pages.

[3] S. Venkataramani, S. Chakradhar, K. Roy and A. Raghunathan, "Approximate computing and the quest for computing efficiency," in Proc. 52nd Annual Design Automation Conference (DAC), 2015, Article 120, 6 pages.

[4] Q. Xu, N. S. Kim and T. Mytkowicz," Approximate computing: a survey," IEEE Design and Test, vol. 33, no.1, pp. 8-22, 2016.

[5] Y. Wang, H. Li, X. Li," Real-time meets approximate compureting: An elastic CNN inference accelerator with adaptive trade-off between QoS and QoR," in Proc. 54th Annual Design Automation Conference (DAC), 2017.

[6] H. Jiang, J. Han and F. Lombardi," A comparative review and evaluation of approximate adders," in Proc. 25th IEEE/ACM Great Lakes Symposium on VLSI, 2015, pp. 343-348.

[7] S.-L. Lu, Speeding up processing with approximation circuits, Computer, vol. 37, no. 3, pp. 67-73, 2004.

[8] N. Zhu, W. L. Goh, W. Zhang, K. S. Yeo and Z. H.'Kong, "Design of low-power high-speed truncation-error-tolerant adder and its application in digital signal processing," IEEE Trans. VLSI Syst., vol. 18, no. 8, pp.1225-1229, 2010.

[9] K. Du, P. Varian and K. Mohanram, "High-performance reliable variable latency carry select addition," in Proc. Design, Automation and Test in Europe Conference and Exhibition (DATE), 2012, pp. 1257-1262.

[10] H. R. Mahdiani, A. Ahmadi, S. M. Fakhraie and C. Lucas, "Bioinspired imprecise computational blocks for efficient VLSI implementation of soft-computing applications," IEEE Trans. Circuits Syst.: Part I Regular Papers, vol. 57, no. 4, pp. 850-862, 2010.

[11] M. Imani, D. Peroni, and T. Rosing," CFPU: Configurable floating point multiplier for energy-efficient computing," in Proc. 54th Annual Design Automation Conference (DAC), 2017, pp. 18-22.

[12] J. Mitchell, "Computer multiplication and division using binary logarithms," IRE Trans. Electronic Computers, vol. EC-11, no. 4, 512-517, 1962.
[13] J. Low and C. Jong, "Unified Mitchell-based approximation for efficient logarithmic conversion circuit," IEEE Trans. Computers, vol. 64, no. 6, pp. 1783-1797, 2015.
[14] M. Sullivan and E. E. Swartzlander, Jr., "Truncated error correction for flexible approximate multiplication," in Proc. Conf. Record the 46th Asilomar Conf. Signals, Systems and Computers, 2012, pp. 355-359.
[15] K. Y. Kyaw, W. L. Goh and K. S. Yeo, "Low-power high-speed multiplier for error-tolerant application," in Proc. Electron Devices and Solid-State Circuits, 2010, pp. 1-4.
[16] S. Hashemi, R. Bahar and S. Reda, "DRUM: A dynamic range unbiased multiplier for approximate applications," in Proc. IEEE/ACM International Conference on Computer Design, 2015, pp. 418-425.
[17] P. Kulkarni, P. Gupta, and M. Ercegovac, "Trading accuracy for power with an underdesigned multiplier architecture," in Proc. 24th Int. Conf. VLSI Design, 2011, pp. 346-351.
[18] K. Bhardwaj, P. Mane and J ¨org Henkel,"Power-and area-efficient approximate wallace tree multiplier for error-resilient systems," in Proc. 15th IEEE Int. Symp. Quality Electronic Design, 2014, pp. 263-269.
[19] H. Jiang, J. Han, F. Qiao, F. Lombardi,"Approximate radix-8 Booth multipliers for low-power and high performance operation," IEEE Trans. Computers, vol. 65, pp. 2638-2644, Aug. 2016.
[20] V. Leon, G. Zervakis, D. Soudris and K. Pekmestzi,"Approximate hybrid high radix encoding for energy-efficient inexact multipliers," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 26, no. 3, pp. 421-430, 2018.