

FPGA Performance Characterization of an 8-Bit Ordinary Truncated Multiplier under Progressive Timing Constraints

Kuldeep Singh¹, Vinod Kumar¹, Ajay Kumar¹, Meenakshi¹, Rajesh Singh²

¹Department of Electrical and Electronics Engineering, Guru Jambheshwar University of Science and Technology, Hisar 125001, India

²Department of Electronics, RKSD College, Kaithal (Haryana) -136027, India

kuldeepsingh@giust.org; Vinodspec@yahoo.co.in; ajaypunia506@gmail.com; meenakshi203020@gmail.com; rsdeshwal@gmail.com

Abstract— This paper assesses a post-route FPGA performance interpretation of an 8-bit unsigned ordinary truncated multiplier executed on a Xilinx Zynq-7000 XC7Z020-CLG484-1 device utilization Vivado 2019.2. The architecture omits all elementary partial products in field $i+j < 6$ and maintains a complete 16-bit output, thereby establishes an intentiona an uncompensated baseline for evaluating future compensated techniques. Evaluated the layout's performance by conducting seventeen post-route runs were conducted using target clock frequencies ranging from 50 to 225 MHz. All implementation runs achieve positive worst negative slack (WNS), Involving +0.413 ns at the 225-MHz frequency constraint. As timing constraints are made more stringent, the resulting critical-path delay diminishes from 7.130 ns to 4.031 ns, whereas the frequency estimated from the available slack rises from 140.25 to 248.08 MHz. This gain is associated with implementation cost: LUT requirement increases from 54 to 72 and dynamic power increasing from 5 to 23 mW. The resulting area-delay product is reduced from 385.02 to 290.23 LUT·ns. The decrease in delay is sufficient to offset the increase in LUT Consumption. An exhaustive analysis of all 65,536 possible unsigned input pairs results value in a mean error distance of 80.25, worst-case error of 321, error rate of 93.75%, and mean signed error range -80.25.

Index Terms—approximate computing, FPGA, ordinary truncation, partial-product multiplier, timing closure, WNS, power-delay product, area-delay product.

I. INTRODUCTION

In modern computing systems, Multiplication play a crucial role in arithmetic operation, including machine-learning accelerators, embedded computing platforms, digital signal processing, image and video processing, and communication systems. A conventional Exact multiplication retains all partial product thereby ensuring arithmetic correctness of the output, nevertheless, the reduction and addition circuit required integrated process these partial products can demand significant logic resources resulting in additional hardware overhead and potentially affecting

performance. Approximate computing intentionally permits a controlled degree of inaccuracy when the target application can accommodate bounded numerical errors, enabling a trade-off between accuracy and hardware cost, energy consumption, or processing latency.

Partial-product truncation is one of the most straightforward approach of approximate multiplication. Rather than altering compressor truth tables or adopting logarithmic approximations, the designer eliminates low-significance terms from the partial-product matrix. Petra et al. demonstrated that omitted during partial products can be combined with correction functions to regulate mean-square error, In related work Garofalo et al. developed analytical characterization of maximum error for a family of truncated multipliers. Furthermore, architectures including TOSAM and partial-product perforation methods, demonstrate that truncation continues to be an effective mechanism reduce the computational cost of arithmetic operations [1]–[4].

The scope of the present study is intentionally limited to ordinary truncation without any compensation mechanism. The work does not intend to claim truncation in isolation as a new multiplier concept, but to formulate a controlled FPGA baseline under a broad range of implementation timing constraints. This characterization is necessary before introducing fixed-bias or data-dependent correction because the additional correction logic must ultimately be judged against the hardware and numerical behavior of uncompensated truncation.

The principal contribution is a 17-point post-route timing sweep spanning 50 to 225 MHz on the XC7Z020-CLG484-1 FPGA. The reported implementation metrics include WNS, realized critical-path delay, slack-derived frequency estimate, LUT and FF utilization, total and dynamic power, PDP, and ADP. In parallel, the arithmetic approximation is exhaustively characterized over all 65,536 unsigned 8-bit input pairs. The resulting data separate two effects that are frequently conflated: the fixed numerical error introduced by removing low-

order partial products and the variations in physical implementation resulting from progressively tighter timing constraints during place-and-route.

II. RELATED WORK

Truncated multiplication has a well-established role in both fixed-width and approximate arithmetic. Hardware reduction is achieved by removing selected low-weight partial products from the multiplication process, while additional correction logic can be used when error reduction is required. Petra et al. investigated variable-correction functions for truncated binary multipliers and showed the trade-off among the discarded partial products, the complexity of the correction circuitry, and the resulting error performance [1]. Garofalo et al. later provided a closed-form analytical calculation of maximum error for a family of truncated multipliers designed for low mean-square error [2]. These works provide the theoretical foundation for treating discarded-column information and compensation as separable design choices.

Approximate multiplication studies have increasingly addressed scalable, configurable, and application-oriented architectures. Zervakis et al. investigated the use of partial-product perforation to achieve efficient approximate multiplication designs [3]. Vahdat et al. introduced TOSAM, which integrates to reduce area and energy consumption significantly while preserving application-level quality [4]. In contrast to simple truncation, these approaches employ supplementary approximation or correction operations.

For FPGA implementation analysis, the mapping process represents an important experimental variable. Increasing the strictness of timing constraint can cause logic restructuring, replication, modified placement, and alternative routing, so resource count and path delay need not vary monotonically with the requested clock period. Thus, relying on one implementation point provides an incomplete assessment of a multiplier architecture. The present work addresses this limitation through a progressive post-route timing sweep and adopts ordinary truncation as a reproducible

reference architecture for further compensation studies.

III. ORDINARY TRUNCATED MULTIPLIER ARCHITECTURE

A. Exact Multiplication Reference

For unsigned N-bit operands A and B, the exact product is computed from the complete partial-product matrix:

$$P_EX = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} A_i B_j 2^{(i+j)}.$$

For N=8, the complete matrix consists of 64 elementary bit products. The exact result occupies 16 bits.

B. Ordinary Column Truncation

A truncation boundary CUT is established and each terms satisfying $i+j < \text{CUT}$ are omitted. The retained product is

$$P_OT = \sum_{(i+j \geq \text{CUT})} A_i B_j 2^{(i+j)}$$

The present study employs CUT=6. Hence columns 0 through 5 are discarded and columns 6 through 14 are retained. The discarded contribution can be expressed as

$$D_CUT = \sum_{(i+j < \text{CUT})} A_i B_j 2^{(i+j)}$$

and, since no compensation is applied,

$$P_OT = P_EX - D_CUT, \quad e = P_OT - P_EX = -D_CUT$$

For CUT=6, the discarded region contains $1+2+3+4+5+6=21$ elementary partial products, corresponding to 32.81% of the 64 bit-level terms. This percentage indicates the amount of structural removal prior synthesis; but the resulting LUT reduction is not necessarily proportional because FPGA mapping modifies the organization of Boolean logic and carry structures.

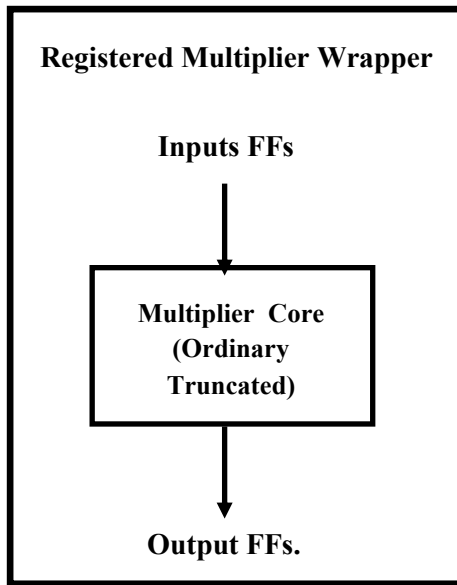


Fig. 1. Functional organization of the registered multiplier wrapper experimental model used for simulations and implementation.

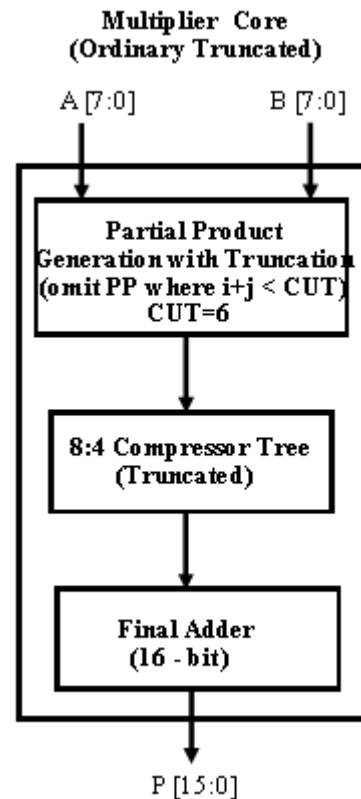


Fig. 2. Functional block diagram of the ordinary truncated multiplier.

C. Registered Multiplier Wrapper

The combinational ordinary truncated core is positioned between common input and output registers. At a rising clock edge, A [7:0] and B[7:0] are captured by input registers; the registered operands drive the truncated core; and the 16-bit result is captured by the output register. The evaluated path is therefore a controlled register-to-register datapath expressed in Fig. 1.

This wrapper ensures that post-route timing measurements comparable across timing constraints and later across alternative multiplier cores. In fabric-only comparisons of structurally implemented approximate multipliers, the use of dedicated DSP resources should be prevented by disabling DSP inference.

TABLE I. POST-ROUTE FPGA RESULTS FOR THE 8-BIT ORDINARY TRUNCATED MULTIPLIER

Target (MHz)	Period (ns)	WNS (ns)	Tcrit (ns)	Fmax,est (MHz)	Ptot (W)	Pdyn (W)	PDP (nJ)	LUT	ADP (LUT·ns)	FF
50	20.000	12.870	7.130	140.25	0.110	0.005	0.0357	54	385.02	26
62.5	16.000	9.093	6.907	144.78	0.111	0.006	0.0414	54	372.98	26
75.09	13.330	6.287	7.043	141.98	0.112	0.008	0.0563	54	380.32	26
87.5	11.430	4.541	6.889	145.16	0.113	0.009	0.0620	54	372.01	26
100	10.000	3.092	6.908	144.76	0.115	0.010	0.0691	54	373.03	26

IV. RESULTS AND DISCUSSION

A. FPGA Platform and Timing Sweep

The proposed design was implemented in Xilinx Vivado 2019.2 for the Zynq-7000 XC7Z020-CLG484-1 device used on the ZedBoard. Seventeen implementation runs were performed for the selected target clock frequencies of 50, 62.5, 75.09, 87.5, 100, 112.5, 125, 133, 150, 162.6, 175.131, 180.995, 190.5, 200, 211.5, 220, and 225 MHz. The associated clock constraints range from 20.000 to 4.444 ns.

B. Post-Route Timing and Resource Results

112.5	8.890	2.492	6.398	156.30	0.116	0.011	0.0704	54	345.49	26
125	8.000	1.311	6.689	149.50	0.117	0.013	0.0870	54	361.21	26
133	7.519	0.923	6.596	151.61	0.118	0.014	0.0923	54	356.18	26
150	6.670	0.204	6.466	154.66	0.120	0.015	0.0970	54	349.16	26
162.6	6.150	0.104	6.046	165.40	0.121	0.017	0.1028	55	332.53	26
175.131	5.710	0.321	5.389	185.56	0.123	0.018	0.0970	59	317.95	24
180.995	5.525	0.323	5.202	192.23	0.123	0.019	0.0988	63	327.73	26
190.5	5.263	0.170	5.093	196.35	0.124	0.020	0.1019	66	336.14	26
200	5.000	0.036	4.964	201.45	0.125	0.020	0.0993	67	332.59	27
211.5	4.726	0.376	4.350	229.89	0.126	0.022	0.0957	69	300.15	31
220	4.545	0.420	4.125	242.42	0.128	0.023	0.0949	72	297.00	37
225	4.444	0.413	4.031	248.08	0.128	0.023	0.0927	72	290.23	34

All 17 existing implementations have positive WNS. At the relaxed 50-MHz target, WNS is +12.870 ns and the observed critical delay is 7.130 ns. At the maximum evaluated 225-MHz target, WNS is maintained +0.413 ns while the realized critical delay

declines below to 4.031 ns. Thus, the measured sweep does not yet encounter setup-timing failure. The slack-derived frequency estimate at the 225-MHz implementation is 248.08 MHz, but an additional sweep above 225 MHz is necessary before determining the maximum timing-closed operating frequency

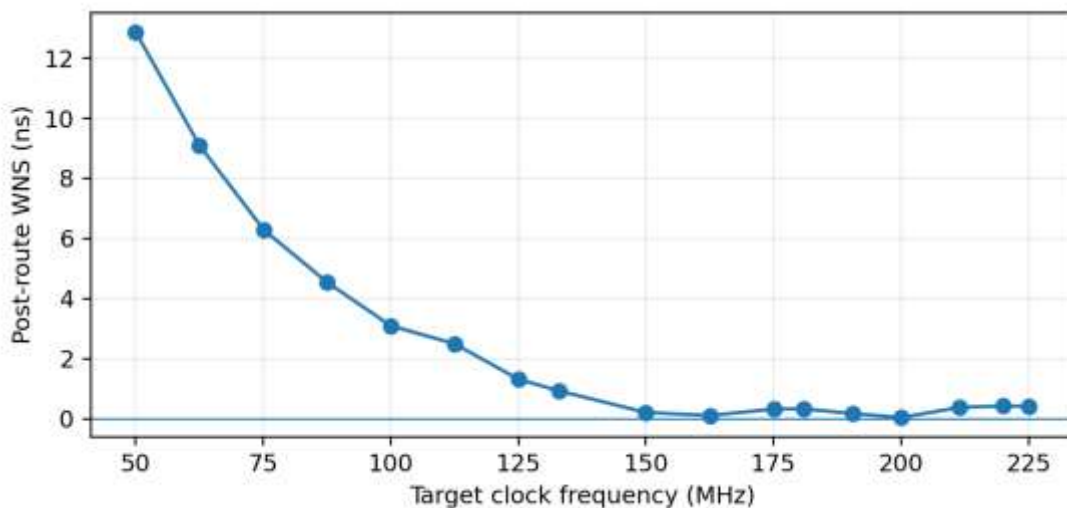


Fig. 3. Post-route WNS versus target clock frequency.

C. Constraint-Driven Critical-Path Behavior

The critical-path delay is not uniform across runs because each timing constraint invokes a new optimization problem. From 50 to 150 MHz, the routed delay remains broadly in the 6.4-7.1 ns range. Beyond approximately 162.6 MHz, tighter constraints produce progressively shorter implementations: 6.046 ns at 162.6 MHz, 5.389 ns at 175.131 MHz, 4.964 ns at 200 MHz, and 4.031 ns at 225 MHz. The corresponding slack-derived frequency estimate rises to 248.08 MHz.

The non-monotonic behavior observed under relaxed constraints is also informative. For example, the 75.09-MHz implementation has a 7.043-ns critical delay, whereas the 62.5-MHz run has 6.907 ns. Such differences are consistent with heuristic synthesis, placement, and routing decisions and demonstrate why Fmax should not be inferred from a single relaxed implementation.

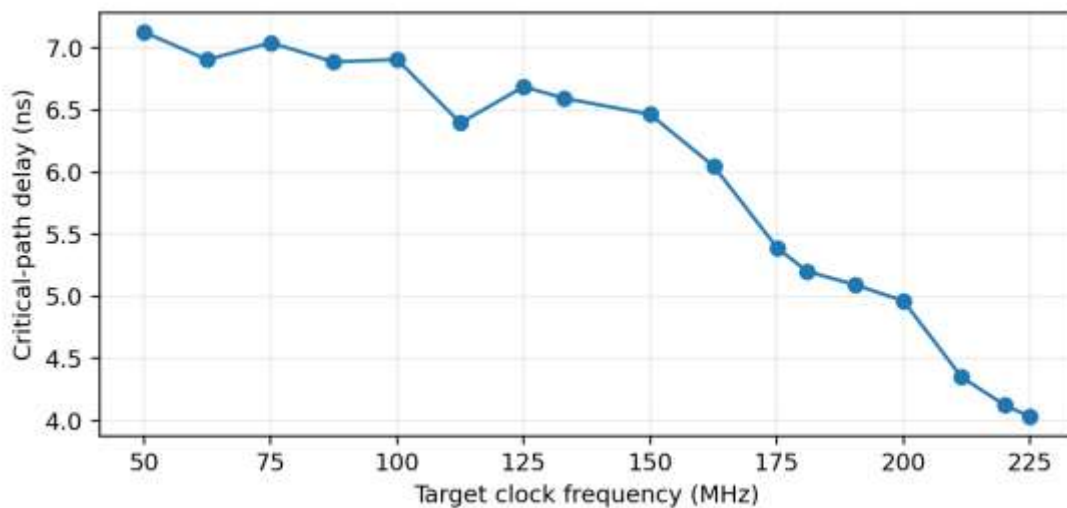


Fig. 4. Realized critical-path delay under progressively tightened timing constraints.

D. Power and PDP

Dynamic power rises from 0.005 W at 50 MHz to 0.023 W at 220-225 MHz, while total power increases from 0.110 to 0.128 W. The inferred static component remains 0.105 W throughout the dataset. The predominance of the static component demonstrates why total power shows considerably less variation than dynamic power across the frequency sweep.

PDP does not exhibit a monotonic trend because the increase in dynamic power is partially offset by the reduction in critical-path delay. It rises from 0.0357 nJ at 50 MHz to a local maximum of 0.1028 nJ at 162.6 MHz, then remains near 0.093-0.102 nJ at the highest-frequency points. At 225 MHz the reported PDP is 0.0927 nJ. Consequently, The implementation subjected to the highest timing pressure offers improved speed but does not achieve the lowest energy consumption under the reported power assumptions.

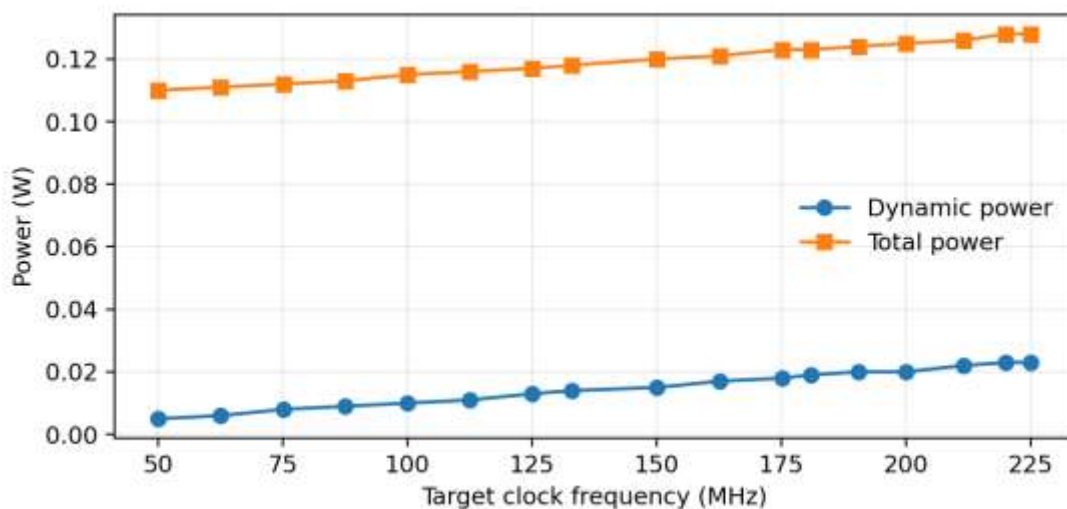


Fig. 5. Reported total and dynamic power versus target clock frequency.

E. LUT/FF Cost and ADP

LUT utilization remains constant at 54 LUTs from 50 throughout 150 MHz, then increases as the timing constraints become more stringent. 55 LUTs at 162.6

MHz, 59 at 175.131 MHz, 67 at 200 MHz, and 72 at 220-225 MHz. FF utilization is 26 remains low for most of the lower-frequency cases and increases when the tightest constraints are applied, peaking at 37 FFs at 220 MHz and 34 FFs at 225 MHz. These variations suggest that the implemented netlist undergoes timing-driven restructuring and/or logic replication.

Although LUT utilization increases, the ADP shows a substantial improvement, primarily because the decrease in critical delay outweighs the area increase. ADP decreases from 385.02 LUT·ns at 50 MHz to 290.23 LUT·ns at 225 MHz, a decrease of

approximately 24.6%. The minimum reported ADP occurs at range 225 MHz. This result illustrates an important implementation trade-off: A design with faster routing may use additional logic but can still result in improved area-delay performance.

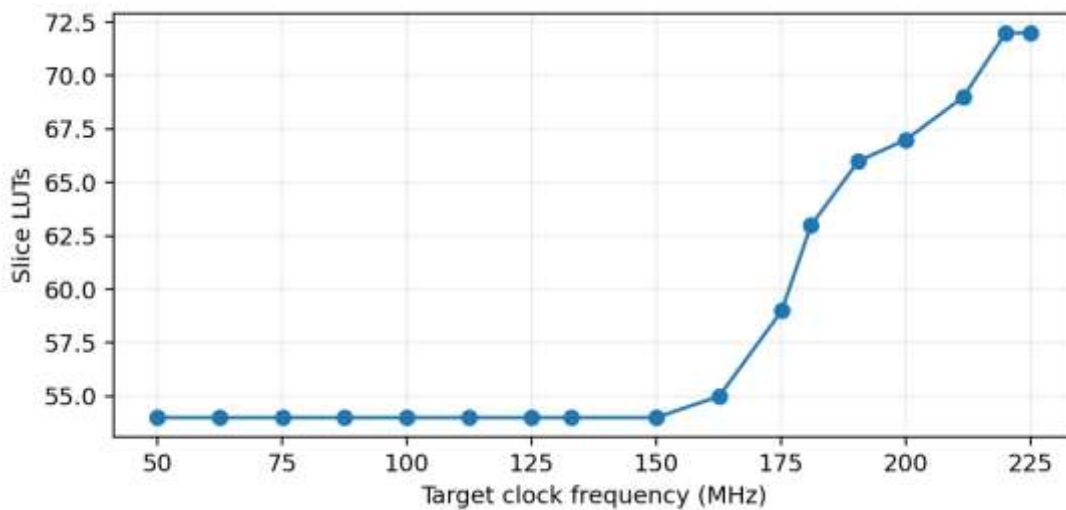


Fig. 6. LUT utilization as the target timing constraint is tightened.

F. Complexity-Accuracy Interpretation

The simulated results illustrate two independent trade-offs. Initially, ordinary truncation eliminates 21 of 64 elementary partial products and thereby reducing the complexity of the arithmetic structure. This simplification generates a systematic negative error, as the contribution eliminated by truncation is not compensated. Second, increasingly aggressive FPGA timing constraints reduce the realized critical delay but eventually require additional LUTs and registers. The latter reflects an implementation-level trade-off rather than the arithmetic approximation itself.

This distinction makes the conventional truncated multiplier a useful reference for evaluating subsequent compensated architectures. The performance of a fixed-bias design can be evaluated by examining how much of the 80.25 MED and -80.25 mean bias it removes for its added hardware cost. Correspondingly, a BWC-3 design can be determined by whether its data-dependent correction produces a larger accuracy recovery than fixed bias while maintaining competitive LUT, power, PDP, and ADP. Therefore, this work establishes a baseline at the uncompensated end of the complexity-accuracy design space.

V. CONCLUSION

This work presents an 8-bit ordinary truncated multiplier has been characterized on a Zynq-7000 XC7Z020 FPGA over 17 post-route timing targets from 50 to 225 MHz. With CUT=6, 21 of the 64

elementary partial products are eliminated and no correction term is introduced. All measured runs maintain positive setup WNS; at the 225-MHz target the routed design reports +0.413 ns WNS, 4.031 ns critical delay, a slack-derived frequency estimate of 248.08 MHz, 72 LUTs, 34 FFs, 23 mW dynamic power, 0.0927 nJ PDP, and 290.23 LUT·ns ADP. Relative to the 50-MHz implementation, timing-driven optimization reduces critical delay by approximately 43.5% while LUT count increases by 33.3%. ADP nevertheless improves by approximately 24.6%. Exhaustive arithmetic evaluation shows all possible input combinations MED=80.25, WCE=321, error rate=93.75%, and mean signed error=-80.25, verifying the expected one-sided bias of uncompensated truncation. This study consequently establishes a controlled FPGA and numerical reference point for the next phase, in which fixed-bias and boundary-weighted compensation methods can be compared according to the additional hardware needed to recover accuracy.

REFERENCES

- [1] N. Petra, D. De Caro, V. Garofalo, E. Napoli, and A. 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, 2010, doi: 10.1109/TCSL.2009.2033536.
- [2] V. 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, 2011, doi: 10.1109/TC.2010.236.

[3] G. Zervakis, K. Tsoumanis, S. Xydis, D. Soudris, and K. Pekmestzi, "Design-Efficient Approximate Multiplication Circuits Through Partial Product Perforation," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 24, no. 10, pp. 3105–3117, Oct. 2016, doi: 10.1109/TVLSI.2016.2535398.

[4] S. Vahdat, M. Kamal, A. Afzali-Kusha, and M. Pedram, "TOSAM: An Energy-Efficient Truncation- and Rounding-Based Scalable Approximate Multiplier," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 27, no. 5, pp. 1161–1173, May 2019, doi: 10.1109/TVLSI.2018.2890712.

[5] P. Kulkarni, P. Gupta, and M. Ercegovic, "Trading Accuracy for Power With an Underdesigned Multiplier Architecture," in Proc. 24th Int. Conf. VLSI Design, 2011, pp. 346–351, doi: 10.1109/VLSID.2011.51.

[6] R. Zendegani, M. Kamal, M. Bahadori, A. Afzali-Kusha, and M. Pedram, "RoBA Multiplier: A Rounding-Based Approximate Multiplier for High-Speed Yet Energy-Efficient Digital Signal Processing," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 25, no. 2, pp. 393–401, Feb. 2017, doi: 10.1109/TVLSI.2016.2587696.