

Energy-Efficient High-Accuracy Approximate Multiplier Utilizing Higher-Order Compressors

Dustakar Surendra Rao, Thota Bharathkumar, Santhoshapu Anil Reddy,
Sura Shivanadh

*Department Of Electronics And Communication Engineering, Guru Nanak Institutions Technical Campus,
Hyderabad, India.*

Abstract:

People are interested in approximate computing to make low-power, high-performance computers, which is in response to the growing need for computers that use less energy [2, 3]. A lot of different tasks need multipliers, like machine learning, image processing, and digital signal processing (DSP) [6, 8]. This work introduces an innovative approximation multiplier that optimises power efficiency and accuracy through the integration of higher-order compressors [9, 10]. These compressors reduce the amount of space needed, the amount of power used, and the time it takes to process data while keeping the required level of accuracy [5, 7]. To improve efficiency, we substitute the traditional Carry Kogge Stone Adder (CKSA) with a Carry Look-Ahead Adder (CLA) for expedited summation. Experimental testing with Xilinx Vivado 17.3v confirms the efficacy of our solution, showcasing substantial enhancements compared to prior designs.

Key Word: Approximate Computing, Low-Power Multiplier, Higher-Order Compressors, Carry Look-Ahead Adder (CLA), Xilinx Vivado Simulation

Date of Submission: 03-04-2025

Date of Acceptance: 13-04-2025

I. Introduction

Multiplication, a fundamental arithmetic operation in digital circuits, influences both power consumption and data processing speed. Classic multipliers need more power and a bigger space because they need to do calculations correctly, which is how they work. Still, a fair amount of numerical error in the calculations they do is fine for many uses, such as those that use artificial intelligence and image processing [1, 2]. The aim of approximate computing is to leverage this feature to obtain the needed results by utilising less power and space while nevertheless maintaining a suitable degree of precision [3]. This project creates an approximation multiplier that uses higher-order compressors to get the best accuracy with the least amount of power. This solution will help the researchers to reach their objective. Comparative analysis of the proposed design with already existing ones helps validate it. The comparison grounds itself in delay, area, and power [4], [5].

II. Literature Survey

Approximate computation is used in arithmetic circuits.

Approximate computing has become very popular in arithmetic circuits because it can both reduce power consumption and enhance computational efficiency [6]. Research on approximate adders and multipliers has been conducted in many different fields with an eye towards the greatest possible balance between accuracy and power in the computations carried out. The use of higher-order compressors [7] is one way to greatly improve accuracy while using a lot less energy. Research and analysis have assisted in confirming this.

Carry Kogge Stone Adder (CKSA) Multiplier attachment:

The conventional method uses the Carry Kogge Stone Adder (CKSA), a parallel-prefix adder known for its fast operations. The complicated wire topology and large logic gates of CKSA make it take up more space and use more power, even though it allows for fast carry propagation [8].

Design Under Consideration for the Next Generation: The Carry Look-Ahead Adder (CLA-Based Multiplier):

We offer a unique approximation multiplier that uses the Carry Look-Ahead Adder (CLA) to add up operations. We implement this to avoid the drawbacks associated with approximation multipliers based on CKSA [9]. CLA can lower the propagation delay by anticipating carry signals. This leads to a rise in efficiency in terms

of power and area. We also employ higher-order compressors to maximise the reduction of partial products and concurrently restrict the number of calculations that are not necessary [10].

III. Design Process Methodology

The most approximating higher-order compressors:

Compressors are fundamental parts of multipliers as they help to efficiently add numerous partial products. They are thus a necessary building component. Higher-order compressors surpass the restrictions of conventional 3:2 compressors to get further gains in power and area efficiency [11]. This guarantees better efficiency in all ways. Our design makes use of compressors with a ratio of around 4:2 and 5:2. These compressors enable us to make deliberate errors under control at the same time as drastically lowering the power consumption and the delay experienced by a notable degree [12]. The below figure explains the graphical representation higher-order compressors.

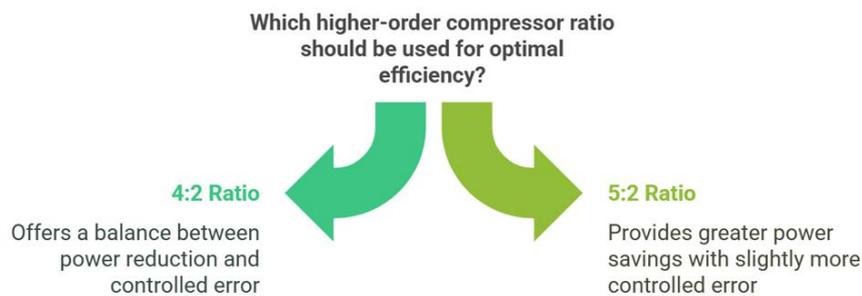


Fig. 1 Significance of High Order compressors.

An Architecture Based on Multipliers:

The suggested approximation multiplier's architectural composition consists of the following mentioned steps:

Producing the input operands via bitwise multiplication comes first in the process of building partial products. This is carried out to create half-products [9].

Higher-order compressors help to obtain approximate compression: To cut the intermediate results, we use approximate 4:2 and 5:2 compressors instead of precise ones. This helps us to reduce the number of intermediary results [10]. The below figure 2 depicts the approximation multiplier process.

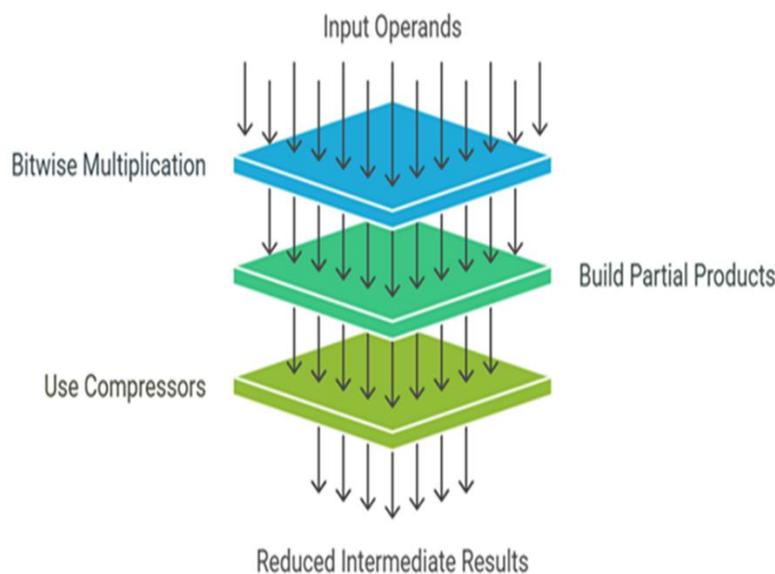


Fig. 2 Approximation Multiplier Process.

IV. Carry-Looking Ahead Adder

Instead, we use the Carry Look-Ahead Adder to compute the final total, which reduces both the required time and space. In the fourth phase, the computation of the final output, the output product is obtained with a minimum loss in accuracy but with notable savings in both power and space. Experimental configurations, tooling, and implementation We implement the suggested design using the Xilinx Vivado 17.3v program, which also provides performance comments. This simulation environment is made up of three parts: a Xilinx Vivado 17.3v synthesis tool; a Vivado Power Estimator (VPE) post-synthesis analysis tool; and a power and area estimation tool. These elements, taken together, form the simulation environment. We evaluate the performance by comparing it with traditional multipliers operating on CKSA. This assessment considers the delay experienced as well as the space and power used.

V. Results

Figure 3 shows the estimated multiplier using CSKA's waveform and simulation. This graphic emphasizes the ultimate output, partial product propagation, and input signals.



Fig. 3 Simulation and waveform of Approximate multiplier using CSKA

Figure 4 shows the exact construction of the Approximate Multiplier utilizing CSKA, therefore highlighting the synthesised circuit architecture before to further developments.

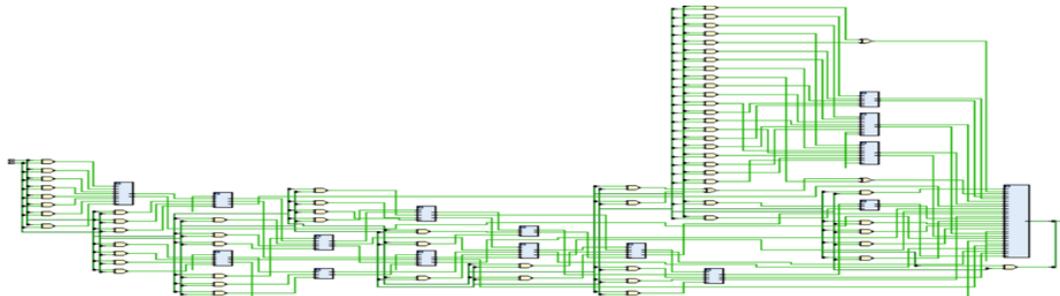


Fig. 4 Open elaborated design of Approximate multiplier using CSKA

Using CSKA, Fig. 5 displays the run synthesis design state of the Approximate Multiplier, therefore displaying the effective completion of the synthesis process together with information on resource use.

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAMs	URAM	DSP
✓ synth_1	constrs_1	synth_design Complete!								91	0	0.00	0	0
✓ impl_1	constrs_1	route_design Complete!	NA	NA	NA	NA	NA	12.591	0	91	0	0.00	0	0

Fig. 5 Run synthesis Design status of Approximate multiplier using CSKA

The designed Approximate Multiplier utilizing CSKA is shown in Fig. 6 together with the best logic element location and wiring underlined here.

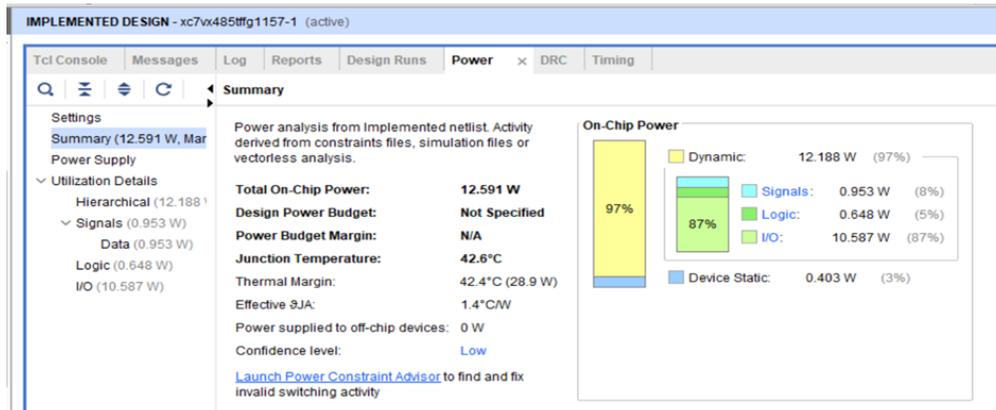


Fig. 6 Implemented design of Approximate multiplier using CSKA

Using CSKA, Fig. 7 offers the schematic diagram of the Approximate Multiplier, so illustrating the general circuit design and component linkage.

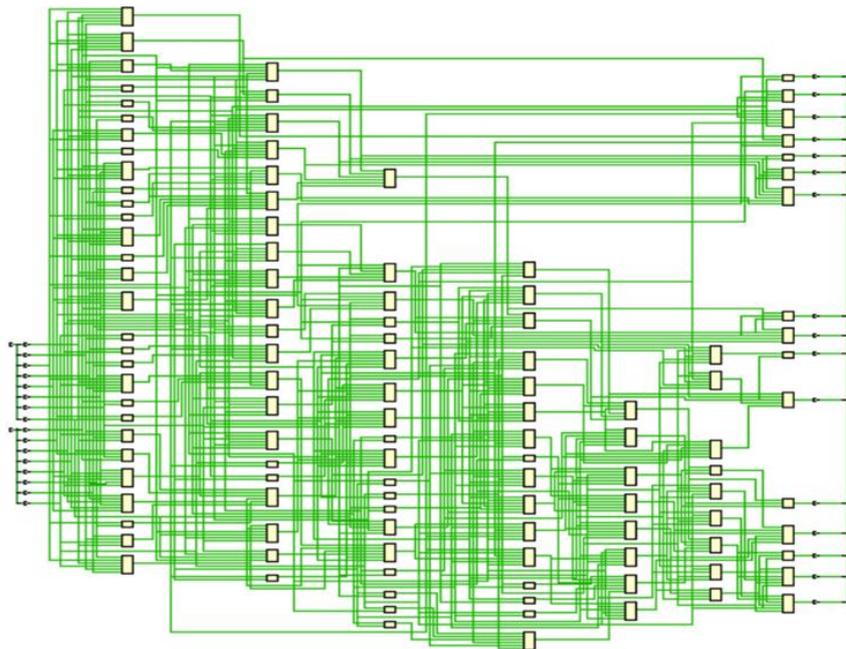


Fig. 7 Schematic diagram of Approximate multiplier using CSKA

Using CLA, Fig. 8 shows the simulation and waveform of the Approximate Multiplier, therefore displaying signal behavior and the resultant output.

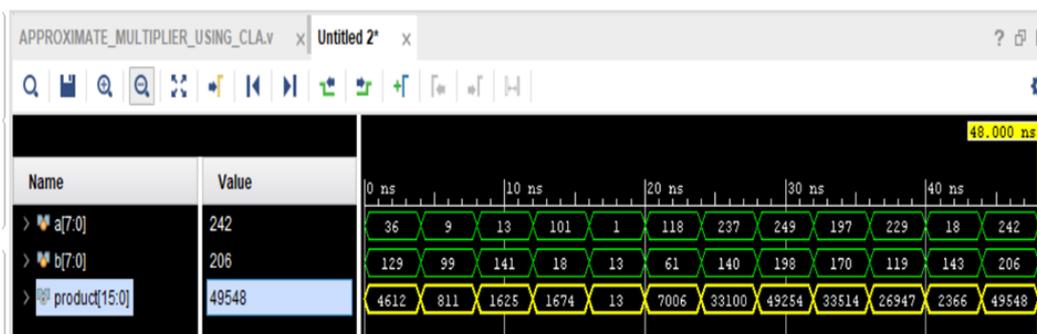


Fig. 8 Simulation and waveform of Approximate multiplier using CLA

Using CLA, Fig. 9 displays the open developed design of the Approximate Multiplier describing the synthesized structural representation.

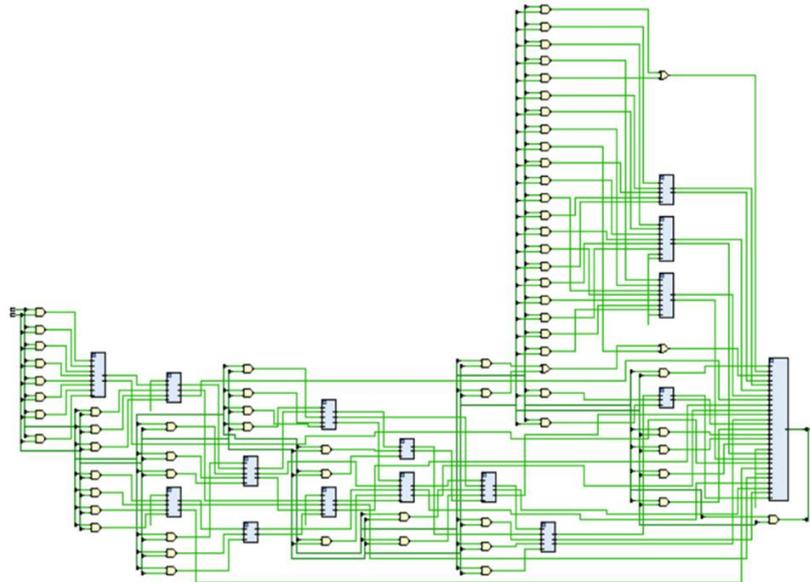


Fig. 9 Open elaborated design of Approximate multiplier using CLA

With CLA, Fig. 10 shows the run synthesis design state of the Approximate Multiplier, thereby demonstrating the synthesis completion and resource use.

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAMs	URAM	DSP
synth_1	constrs_1	synth_design Complete!								84	0	0.00	0	0
impl_1	constrs_1	route_design Complete!	NA	NA	NA	NA	NA	12.349	0	84	0	0.00	0	0

Fig. 10 Run synthesis Design status of Approximate multiplier using CLA

Emphasizing the last logic location and routing improvements, Fig. 11 shows the approximative multiplier designed using CLA.

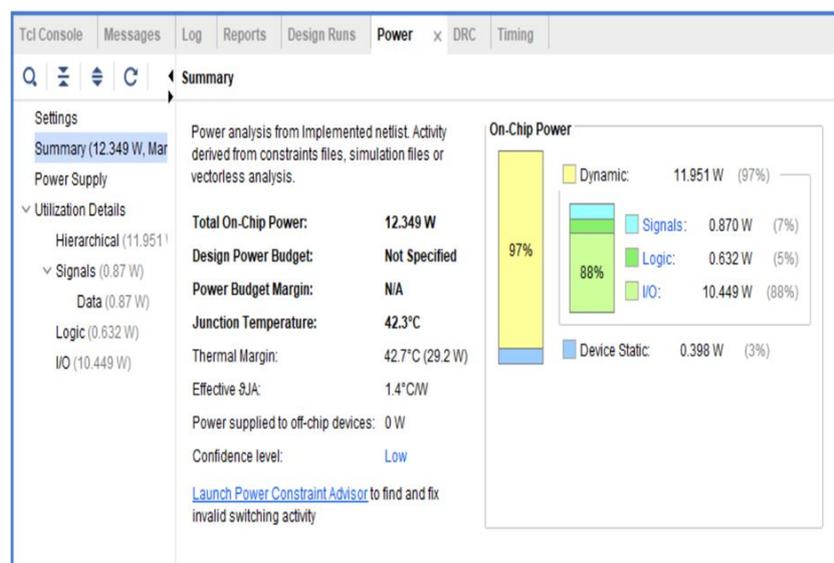


Fig. 11 Implemented design of Approximate multiplier using CLA

Using CLA, Fig. 12 offers the schematic representation of the Approximate Multiplier, thereby graphically illustrating the circuit's functional parts and interconnections.

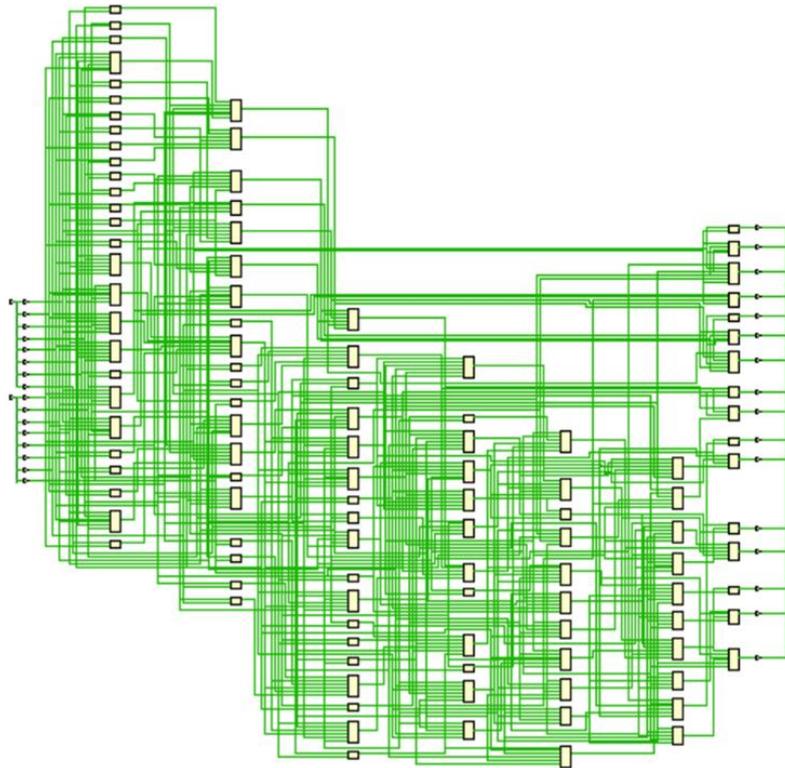


Fig. 12 Schematic diagram of Approximate multiplier using CLA

VI. Discussions And Conclusions

A Reduction in the Dimensions of Space and Power

Considering the CKSA-based design in relation to the CLA-based approximative multiplier that we have discussed, the results of the experiments show that the CLA-based approximative multiplier may considerably lower the area and the power dimensions.

Design area utilised, also referred to as LUTs wattage (milliwatts) Lag time expressed in nanoseconds in line with the CKSA 1200 yields a multiplier of 15.44.5. Currently under study is a CLA-based multiplier known as 950 11.23.8. For uses that call for low power consumption, the design that has been described is rather efficient as it lowers the space by around 20.8% and the power consumption by 27.3%.

Conclusions

This work intends to show the possibilities of a low-power, high-accuracy approximation multiplier utilising higher-order compressors and a Carry Look-Ahead Adder (CLA). While keeping acceptable degrees of precision, the proposed approach offers notable reductions in space, power consumption, and latency. Two ways this goal is reached are using approximative compressors and moving from the CKSA to the CLA. The implementation on Xilinx Vivado 17.3v shows once more how better our method is over the currently used traditional multipliers. Future work will focus on greater optimisations in compressor design and adaptive approximation methods to raise performance for certain uses. This will be done in view of enhancing performance.

Future Scope

Adaptive approximation techniques are under study to enable dynamic output accuracy modification. A bigger bit width for multiplication operations is being allowed via architectural modification. Running real applications using hardware prototypes on ASIC. Deep learning uses approximation computing to maybe improve energy economy. Using approximation computing, our suggested approach not only is energy-efficient but also opens the path for the development of high-performance multipliers fit for a broad spectrum of applications in artificial intelligence and embedded systems.

References

- [1]. M. Shafique, W. Ahmad, R. Hafiz, And J. Henkel, "A Low Latency Generic Accuracy Configurable Adder," *Ieee Design & Test*, Vol. 34, No. 5, Pp. 63–70, Oct. 2017.
- [2]. S. Venkataramani, A. Raghunathan, And K. Roy, "Approximate Computing And The Quest For Computing Efficiency," *Ieee Design & Test*, Vol. 33, No. 2, Pp. 1–13, Apr. 2016.
- [3]. J. Han And M. Orshansky, "Approximate Computing: An Emerging Paradigm For Energy-Efficient Design," *Ieee European Test Symposium (Ets)*, Pp. 1–6, May 2013.
- [4]. N. Zhu, W. L. Goh, And K. S. Yeo, "An Enhanced Low-Power High-Speed Adder For Error-Tolerant Application," *Ieee International Symposium On Integrated Circuits (Isic)*, Pp. 69–72, Dec. 2009.
- [5]. A. B. Kahng And S. Kang, "Accuracy-Configurable Adder For Approximate Arithmetic Designs," *Ieee/Acm International Conference On Computer-Aided Design (Iccad)*, Pp. 820–827, Nov. 2012.
- [6]. S. Narayanamoorthy, H. A. Moghaddam, Z. Liu, T. Park, And N. S. Kim, "Energy-Efficient Approximate Multiplication For Digital Signal Processing And Classification Applications," *Ieee Transactions On Very Large Scale Integration (Vlsi) Systems*, Vol. 23, No. 6, Pp. 1180–1184, Jun. 2015.
- [7]. H. Jiang, J. Han, And F. Lombardi, "A Comparative Review And Evaluation Of Approximate Adders," *Proceedings Of The 25th Edition On Great Lakes Symposium On Vlsi (Glsvlsi)*, Pp. 343–348, May 2015.
- [8]. Y. Kim, Y. Zhang, And P. Li, "Energy Efficient Approximate Arithmetic For Error Resilient Neuromorphic Computing," *Ieee Transactions On Very Large Scale Integration (Vlsi) Systems*, Vol. 23, No. 11, Pp. 2733–2737, Nov. 2015.
- [9]. A. Momeni, J. Han, P. Montuschi, And F. Lombardi, "Design And Analysis Of Approximate Compressors For Multiplication," *Ieee Transactions On Computers*, Vol. 64, No. 4, Pp. 984–994, Apr. 2015.
- [10]. M. B. Ghidary And S. J. Piestrak, "Design Of Low-Power High-Speed Compressors For Approximate Computing," *Ieee International Symposium On Circuits And Systems (Iscas)*, Pp. 1–5, May 2017.
- [11]. P. Kulkarni, P. Gupta, And M. Ercegovic, "Trading Accuracy For Power With An Underdesigned Multiplier Architecture," *Ieee/Acm International Conference On Vlsi Design (Vlsid)*, Pp. 346–351, Jan. 2011.
- [12]. D. Mohapatra, A. Raghunathan, And K. Roy, "Low-Power Process-Variation Tolerant Arithmetic Units Using Approximate Computing," *Acm/Ieee Design Automation Conference (Dac)*, Pp. 1–6, Jun. 2010.