



(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 14, Issue 6, June 2025

⊕ www.ijirset.com 🖂 ijirset@gmail.com 🖄 +91-9940572462 🕓 +91 63819 07438





www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 6, June 2025

|DOI: 10.15680/IJIRSET.2025.1406053|

Low Error Efficient Approximate Adders with Error Detection and Correction Mechanism for FPGAs

Dasarineerajkumar¹, Jeedi Chandra Shekar², Kothinti Manoj Reddy³, Narsampalli Akash⁴,

Dr.K Satish Reddy⁵

Department of Electronics and Communication Engineering Hyderabad Institute of Technology and Management,

Hyderabad, India^{1,2,3,4}

Professor, Department of ECE, Hyderabad Institute of Technology and Management, Hyderabad, India⁵

ABSTRACT: Approximate computing trades power, area, and speed for accuracy, making it ideal for error- tolerant applications like video encoding and signal processing. This project focuses on developing LEADx and APEx approximate adders for improved FPGA performance while maintaining acceptable accuracy. To address reliability, BCH and Reed-Solomon codes detect and correct bit-flip errors, particularly in the Least Significant Part (LSP) of computations. BCH handles single-bit errors, while Reed-Solomon corrects multi-bit errors, enhancing reliability and durability in error-prone applications.

I. INTRODUCTION

Approximate computing improves power efficiency, area, and speed by allowing controlled errors, making it suitable for applications like video encoding, image processing, and real-time signal processing. Traditional adders are inefficient in resource-constrained environments due to high power consumption and large die areas. This project presents two FPGA-based approximate adders: **LEADx** (Low-Error and Area-Efficient) for low Mean Square Error (MSE) applications, and **APEx** (Area and Power-Efficient) for energy-constrained applications.

Both use a hybrid design with accurate and approximate sections to optimize power and area. To ensure reliability, **BCH** and **Reed-Solomon** error correction codes are integrated to detect and correct errors, enhancing robustness in FPGA-based systems. The project also utilizes FPGA resources like ALMs, LUTs, BRAM, and DSP slices to maintain performance. Simulation results show reduced power consumption, increased speed, and controlled error rates, positioning these viable solutions for next-generation, energy-efficient computing.

II. LITERATURE SURVEY

High-Performance SAD Hardware for FPGA

In [1], Ahmad and Hamzaoglu proposed a SAD high checkout design geared towards FPGAs. Their design minimizes the average and maximum error through the use of all input LUTs and achieves a marked improvement in area and power consumption. Using physical area in comparison to earlier designs methods, they make use of generally 20% fewer LUTs and 38% less power, while maintaining very good accuracy..

Hoaaned approximate adder

P. Balasubramanian and co-authors have designed "hoaaned", a hardware-optimized approximate adder with strong results related to nearly normal error distributions. In practical terms, they shown that through image reconstruction experiments as part of a digital image processing workload on Xilinx Artix-7 FPGAs and ASIC environments. [2]

Video Encoding with Approximate Adders

Mittal et al. introduced an approximate arithmetic design for FPGAs targeted at the design of inexpensive approximate adders for the purposes of video encoding. This work obtained about 40% energy savings with excellent quality in real-time





www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 6, June 2025

|DOI: 10.15680/IJIRSET.2025.1406053|

Error-Resilient Approximate Adders

Designed an adder with error detection capabilities based on Reed-Solomon codes. Reduced power consumption by 28% while maintaining reliable performance for real-time applications showing the potential for efficiency on both exact and approximate operations.

III. METHODOLOGY

EXISTING METHODOLOGY :

The current version of the LOA (Lower-part OR Adder) is viewed as an approximate adder in our design methodology. It essentially consists of an exact and an inexact section. The exact section is anormal section that uses standard precise adders, i.e. Ripley's carried adders (RCA), or carry-lookahead adder (CLA), that calculate the MSBs (Most Significant Bits) with no error. By assumption, the LSBs (Least Significant Bits) are calculated as an approximate FA (Full Adder) which is powered by logical OR and this memory will produce some level of imprecision. Therefore, the logical AND for the inputs for the MSB is used as a carry input for the exact section

Limitations of LOA:

- 1. The hardware occupies a large area.
- 2. Energy consumption is significantly high, and accuracy is low.
- 3. Computation delay increases due to prolonged output calculation times

PROPOSED METHODOLOGY

The proposed methodology provides improvements of performance of approximate adders on FPGAs, a major change potentially by wasting unused inputs of LUTs approximate adders

(n – m)-bit Accurate Part	m-bit Approximate Part			m-bit Approximate Part			,
B _{m1} A _{m1} , , , B _{m41} A _{m1} B _m A _m	1 Am-1 Am-2 Bm-1 Bm-2	1 A _{0r3} A _{0r4} B _{1r3} B _{1r4}	• • •	1 A3 A2 I	B ₃ B ₂	1 A ₁ A ₀ B	B ₀ C _n
		I ₅ I ₄ I ₃ I ₂ I ₁ I ₀	_	I ₅ I ₄ I ₃	2 I ₁ I ₀	I ₅ I ₄ I ₃ I	2 I ₁ I ₀
LUT6 LUT6 LUT6	Proposed 2-bit AAd1	Proposed 2-bit AAd2		Propose AA	1 2-bit 12	Proposed AAd	2-bit
Dedicated Carry-chain	9LDT 06 05	0 ₆ 0 ₅		06	05	06 O	05
Cout Sn-1			•••	-+- S3	S ₂	- + - S ₁	S ₀





Fig 2: APEX (Area and Power-efficient Approximate Adder)





www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 6, June 2025

|DOI: 10.15680/IJIRSET.2025.1406053|

The new approximate adders are designed for error-tolerant applications like video encoding, providing sufficient quality even when input conditions are suboptimal. While they may not match **LEADx** for Mean Square Error (MSE), they offer better performance than previous approximate adders for video applications, focusing on minimizing size, power dissipation, and capacity.





Fig 4: Example of 16-bit APEx with 8-bit approximation.

Error Detection and Correction (EDC) mechanisms

- Bose-Chaudhuri-Hocquenghem (BCH)
- Reed-Solomon Codes, which are used to detect and correct errors

Reed-Solomon (RS) Code:

Reed-Solomon (RS) is an error correction method that adds redundant symbols to data for detecting and fixing multi-bit errors. It works by encoding data, transmitting it, detecting errors using syndromes, and correcting them with mathematical formulas. Used in CDs, QR codes, and satellites



Fig 5: Reed-Solomon Code block diagram

Bose-Chaudhuri-Hocquenghem (BCH) code

BCH code is an error correction technique that adds extra bits to detect and fix multi-bit errors, balancing power and reliability. It encodes data, transmits it, then detects and corrects errors during decoding. Used in flash memory, satellites, and digital broadcasting.





www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 6, June 2025

|DOI: 10.15680/IJIRSET.2025.1406053|



Fig 6: Bose-Chaudhuri-Hocquenghem (BCH) Codes block diagram

DESIGN



Fig 7: low Error and area efficient approximate adder(LEADx)





www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 6, June 2025

|DOI: 10.15680/IJIRSET.2025.1406053|



Fig 8: Area and power efficient approximate adder(APEx)



Fig 9: LEADX approximate adder with reed solomon and bch error detection and correction

Encoding	Decoding		
Message Input	Received Codeword		
-	1		
Generator Polynomial	Error Locator Polynomial		
1	l		
Parity Generation	Error Magnitude Calculation		
Ļ			
Codeword Output	Error Correcttion		
	Corrected		

Fig 10: APEX approximate adder with reed Solomon And bch error detection correction technique

IV. RESULT

SIMULATED OUTPUTS

Vivado's behavioral simulation lets developers verify RTL module functionality early using testbench. It shows output as waveforms without needing synthesis or implementation steps likeplace-and-route.

IJIRSET©2025





www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 6, June 2025

|DOI: 10.15680/IJIRSET.2025.1406053|

Value	10 000 000		
	10,333,338 ps	11,000,000 ps	
dabc	dabc		
adca	adca		
8789	8789		
87	87		
89	89		
1			
00000010	00000010		
00000008	00000008		
	adica adica 8789 87 89 1 00000010 00000000	addca addca a789 8789 87 87 89 89 1 00000010 00000010 00000010 000000008 000000008	

Fig 12: simulated output of leadx

Name	Value	10,999,998 ps
🖪 📲 A[15:0]	dabc	dabc
🖻 🌃 B[15:0]	adca	adca
🖬 📲 Sum[15:0]	87c7	87c7
🗉 📲 AccurateSum[7:0]	87	87
🖬 📲 ApproxSum[7:0]	dc	dc
🖬 📲 CorrectedSum[7:0]	c7	c7
14 CarryOut	1	
🗈 📲 N[31:0]	00000010	00000010
🖬 📲 M[31:0]	0000008	0000008

Fig 13: Leadx output with error correction mechanism using reed Solomon code

Name	Value	10,999,998 ps		
🖬 📲 A[15:0]	dabc	dabc		
🖬 📑 B[15:0]	adca	adca		
🖬 📲 Sum[15:0]	87eb	87eb		
🖬 📲 AccurateSum[7:0]	87	87		
🖬 📲 ApproxSum[7:0]	fe	fe		
🖸 📲 CorrectedSum[7:0]	eb	eb		
🕼 CarryOut	1			
0 - 1 N[31:0]	00000010	00000010		
🖬 🔜 M[31:0]	0000008	00000008		

Fig 14: Leadx output with error correction mechanism using bch code

Name	Value	10,999,998 ps	
🖪 🌃 A[15:0]	dabc	dabc	
🗉 🌃 B[15:0]	adca	adca	Ę
🖬 🃲 Sum[15:0]	87f7	87£7	
AccurateSum[7:0]	87	87	
🖬 📲 ApproxSum[7:0]	dc	dc	
E 📲 CorrectedSum[7:0]	f7	£7	
🖟 CarryOut	1		
🖬 📲 N[31:0]	00000010	00000010	
🖬 😻 M[31:0]	00000008	0000000	
	0000000	0000000	

Fig 16: APEX output error correction mechanism using reed Solomon code





www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 6, June 2025

Name	Value	10,999,998 ps
🖬 – 📲 A[15:0]	dabc	dabc
🖬 📲 B[15:0]	adca	adca
🖬 📲 Sum[15:0]	87d5	87d5
AccurateSum[7:0]	87	87
🖬 📲 ApproxSum[7:0]	fe	fe
🖬 📲 CorrectedSum[7:0]	d5	d5
1 CarryOut	1	
🖬 📲 N[31:0]	00000010	0100000
🖬 📲 M[31:0]	0000008	00000008

|DOI: 10.15680/IJIRSET.2025.1406053|

V. POWER REPORT

Report Power function analyzes detailed power usage by predicting power per LUT based on switching activity and pin capacitance. It's netlist-aware, supports multiple design blocks, and uses hardware data for the selected device/package. Accurate predictions require predefined operating conditions (process, voltage, temperature) and specific switching and clock constraints, typically

Fig 18: Power consumed by the LEAD

Power analysis from Implemented n derived from constraints files, simuli vectorless analysis.	etlist. Activity ation files or	On-Chip Pov	Dynami	c:	5.495 W (9	8%)
Total On-Chip Power:	5.618 W	-		Signals:	0.225 W	(4%)
Junction Temperature:	50.6 °C	98%	95%	Logic:	0.037 W	(1%)
Thermal Margin:	49.4 °C (10.6 W)			I/O:	5.234 W	(95%)
Effective dJA:	4.6 °C/W					
Power supplied to off-chip devices:	o w	1	Device	Static:	0.122 W	(2%)
Confidence level:	Low					

Fig 19: Power consumed by the apex

Fig 20: Power consumption by accurate adder

Т

www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 6, June 2025

|DOI: 10.15680/IJIRSET.2025.1406053|

VI. CONCLUSION

We designed and built low-error efficient approximate adders (LEADx and APEX) for FPGA applications. The proposed adders are well-suited for applications tolerating small errors due to their compromises with accuracy, power, and area efficiency. LEADx's strength lies in its area and delay performance, while APEX sacrifices accuracy for a smaller power footprint. To ensure the reliability of approximate adders, error detection and correction (EDC) functions with Reed-Solomon (RS) and BCH encoding are added. We showed the accuracy is improved through EDC, with reasonable area and power costs. Performance evaluations demonstrated that LEADx (and APEX) approximate adders with EDC strikes a balance between accuracy and hardware efficiency. Therefore, the addition of EDC to the approximate adder architectures is a promising approach for energy efficient, high- performance FPGA platforms, making them appropriate for use in real-time applications such as image processing, machine learning accelerators, resilient IoT devices, and digital signal processing systems

REFERENCES

- 1. G. A. Gillani, M. A. Hanif, B. Verstoep, S. H. Gerez, M. Sha que, and A. B. J. Kokkeler, MACISH: Designing approximate MAC accelerators with internal-self-healing, IEEE Access, vol. 7, pp. 7714277160, 2019.
- 2. E. Kalali and I. Hamzaoglu, An approximate HEVC intra angular prediction hardware, IEEE Access, vol. 8, pp. 25992607, 2020.
- 3. T. Ayhan and M. Altun, Circuit aware approximate system design with case studies in image processing and neural networks, IEEE Access, vol. 7, pp. 47264734, 2019.
- 4. W. Ahmad and I. Hamzaoglu, An efficient approximate sum of absolute differences hardware for FPGAs, in Proc. IEEE Int. Conf. Consume. Electron. (ICCE), Las Vegas, NV, USA, Jan. 2021, pp. 15
- 5. P. Balasubramanian, R. Nayar, D. L. Maskell, and N. E. Mastorakis, "An approximate adder with a near-normal error distribution: Design, error analysis and practical application," IEEE Access, vol. 9, pp. 4518–4530, 2021.
- 6. N. Van Toan and J.-G. Lee, "FPGA-based multi-level approximate multipliers for high performance error-resilient applications," IEEE Access, vol. 8, pp. 25481–25497, 2020.
- 7. G. A. Gillani, M. A. Hanif, B. Verstoep, S. H. Gerez, M. Sha que, and A. B. J. Kokkeler, MACISH: Designing approximate MAC accelerators with internal-self- healing, IEEE Access, vol. 7, pp. 7714277160, 2019

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

www.ijirset.com