ENHANCING PERFORMANCE AND FLEXIBILITY IN APPROXIMATE MULTIPLIERS THROUGH PIPELINING AND ADJUSTABLE PRECISION

Billa.Srilakshmi, Department of Electronics and Communication Engineering, DVR & Dr.HS MIC College of Technology, Kanchikacherla , Andhra Pradesh, Email billasrilakshmi312@gmail.com
Kota.Srinivasa Rao, Assistant Professor, Department of Electronics and Communication Engineering, DVR & Dr.HS MIC College of Technology, Kanchikacherla , Andhra Pradesh. Email Srinuk.449@gmail.com
Boddu.Pushpalatha, Department of Electronics and Communication Engineering, DVR & Dr.HS MIC College of Technology, Kanchikacherla , Andhra Pradesh, Email bpushpalatha2003@gmail.com
Visali.dasari, Department of Electronics and Communication Engineering, DVR & Dr.HS MIC College of Technology, Kanchikacherla , Andhra Pradesh, Email dasarivisali9@gmail.com
SEERAPU SUMALATHA, Department of Electronics and Communication Engineering, DVR & Dr.HS MIC College of Technology, Kanchikacherla , Andhra Pradesh, Email **suma.seerapu@gmail.com**

Abstract

In this research endeavor, we present a novel approach to designing a pipelined approximate multiplier, building upon prior work on high-accuracy approximate compressors. Multiplication operations are fundamental in numerous applications, often constituting a significant portion of power consumption. By integrating approximate computing techniques, we aim to mitigate this power overhead while maintaining acceptable levels of accuracy. Our proposed methodology involves the application of pipelining to enhance the performance of the approximate multiplier. Additionally, we introduce a novel 4-2 compressor design with heightened accuracy, further contributing to the efficiency of the overall system. Notably, our design encompasses an adjustable approximate multiplier capable of dynamically truncating partial products to accommodate varying accuracy requirements, providing flexibility to users. Moreover, we propose a straightforward error compensation circuit to minimize error distances, thereby enhancing the reliability of computations. Our adjustable approximate multiplier empowers users to tailor accuracy and power consumption to suit specific runtime demands. Experimental validation demonstrates significant improvements over existing accurate Wallace multiplier implementations, with notable reductions in parameter values. Furthermore, our pipelined approach outperforms non-pipelined designs, showcasing enhanced performance metrics. Synthesis and simulation of our proposed designs are executable using Xilinx Vivado 2018.3, ensuring practical feasibility and compatibility within existing hardware environments.

Keywords:

Approximate computing, Pipelining, Approximate multiplier, Deep learning, High precision, Re-configurable approximate design.

1 Introduction

In the pursuit of enhanced computing capabilities within battery-powered mobile devices, there has been a notable shift in design priorities from traditional focus on delay and area optimization towards minimizing power dissipation while upholding desired performance levels. A prevalent strategy for achieving energy efficiency in CMOS circuits involves reducing the supply voltage. However, this approach presents two primary challenges. Firstly, lowering the supply voltage inevitably leads to

increased gate delay, impacting overall performance. To mitigate this issue, one common tactic is the scaling down of the threshold voltage. However, this introduces another concern – the degradation of noise immunity within the circuits. The quest for decreased power dissipation holds significant importance in digital circuit design. One established method for achieving this is through supply voltage reduction. Nonetheless, for CMOS circuits, the trade-off for lower supply voltage is often diminished performance. Although scaling the threshold voltage can alleviate some of this performance loss, it typically results in elevated static power dissipation. Studies such as those by Burretal, titled "Cryogenic ultra-low power CMOS" and "Ultra-low power CMOS technology," have demonstrated that optimizing for minimum energy consumption favors operation in the sub-threshold region. However, it's crucial to acknowledge that solutions aimed at minimizing energy consumption often correspond to lower performance outcomes. In our investigation, we focus on both energy and delay considerations during optimization, utilizing the energy-delay product as a metric to gauge circuit efficiency. By scrutinizing the effects of reduced supply and threshold voltages on CMOS circuit energy efficiency, we aim to elucidate strategies for achieving optimal performance within the constraints of power consumption. In traditional digital VLSI (Very Large Scale Integration) design, there's often an underlying assumption that circuits and systems must consistently deliver precise and accurate results. However, this rigid adherence to perfection may not always align with the realities of our non-digital experiences. In many real-world scenarios, what's often more critical is the attainment of results that are "good enough" rather than strictly accurate. Analog computation, which embraces this philosophy of achieving satisfactory outcomes rather than absolute precision, is widely accepted in various domains. This approach acknowledges that minor inaccuracies or deviations from ideal behavior are tolerable and sometimes even expected. Moreover, in numerous digital systems, the data being processed may already contain inherent errors or uncertainties. For instance, in communication systems, analog signals from the external environment are typically sampled and converted into digital data. These digital signals then undergo processing and transmission through potentially noisy channels before eventually being converted back into analog form. Throughout this complex journey, errors can manifest at various stages, further emphasizing the importance of resilience to imperfections. The ongoing advancements in transistor size scaling, driven by Moore's Law and similar trends, have introduced additional complexities. Factors such as noise and process variations have become increasingly prominent considerations in circuit design. As transistor dimensions shrink, the impact of noise and variations in manufacturing processes becomes more pronounced, potentially leading to greater uncertainty in circuit behavior. In essence, acknowledging and accommodating imperfections has become an integral aspect of modern VLSI design. Embracing the principles of analog computation, which prioritize pragmatic and robust solutions over absolute accuracy, can lead to more resilient and adaptable digital systems capable of navigating the complexities of real-world environments. In the realm of digital VLSI design, novel concepts and techniques have emerged to address the evolving demands of modern computing. Among these, two notable advancements are the concept of error tolerance (ET) and PCMOS (Probabilistic CMOS) technology. While the notion of imperfection may initially seem unappealing, the practical necessity for error-tolerant circuits was recognized as early as the 2003 International Technology Roadmap for Semiconductors (ITRS). Error tolerance in a circuit is defined by its ability to accommodate defects that can lead to internal errors, possibly resulting in external errors as well. Despite the seemingly negative connotation, an error-tolerant circuit is deemed successful if the system it's integrated into still produces acceptable results. In response to the challenge of error tolerance, various approaches have been explored. Truncated adders and multipliers, for instance, have been investigated, but often fall short in terms of speed, power efficiency, area utilization, or accuracy. For example, flagged prefixed adders have shown promise with a modest 1.3% speed enhancement over their non-flagged counterparts, albeit at the expense of a 2% increase in silicon area. Similarly, low-error area-efficient fixed-width multipliers have demonstrated significant area improvement, but with average errors as high as 12.4%. It's worth noting that the applicability of error-tolerant concepts varies across different digital systems. In critical systems like control systems, where the correctness of output signals is paramount, the use of error-tolerant circuits may be untenable. However, in digital signal processing (DSP) systems, particularly those handling sensory data like images and speech, error-tolerant circuits find practical applications. Systems involved in image processing, speech recognition, and other human sensory-related tasks

often prioritize overall performance and user experience over absolute precision. In these contexts, the ability to tolerate minor errors can be advantageous, allowing for more efficient utilization of resources and potentially enhancing system throughput and responsiveness. In summary, while the pursuit of error tolerance may seem counterintuitive in traditional digital design, its adoption opens up new avenues for optimizing performance and efficiency in a wide range of applications, particularly those involving human interaction and perception. Approximate computing represents a burgeoning paradigm in digital design, offering a departure from the strict requirement of exact computation to achieve significant improvements in power efficiency, speed, and area utilization. This approach holds particular relevance for embedded and mobile systems, which face stringent constraints on energy consumption and processing speed. The application of approximate computing finds fertile ground in numerous error-resilient domains, including multimedia processing, data mining, recognition tasks, and machine learning. Among the critical components where approximate computing can yield substantial benefits are multipliers. Multipliers serve as fundamental building blocks in microprocessors, digital signal processors, and embedded systems, supporting a wide array of applications from filtering operations to conventional neural networks. However, multipliers are notorious for their complex logic design and significant energy consumption, making them prime candidates for approximate design methodologies. In recent years, the exploration of approximate multiplier designs has emerged as a pivotal area of research. A typical multiplier comprises several essential blocks, including partial products generation, partial products reduction, and carry-propagate addition. Each of these blocks presents opportunities for introducing approximations to optimize performance and energy efficiency. For instance, one established approximation technique involves truncating partial products, wherein certain partial products are omitted from the computation process. To mitigate the impact of truncation errors, suitable correction functions are employed to refine the final result. By strategically implementing such approximations, designers can achieve notable reductions in energy consumption and area overhead while maintaining acceptable levels of accuracy for the intended application. Overall, the exploration of approximate computing techniques, particularly in the design of multipliers, represents a promising avenue for addressing the pressing challenges faced by modern digital systems operating under stringent energy and speed constraints. Through innovative approximation strategies, designers can unlock significant performance gains and pave the way for more efficient and sustainable computing solutions in embedded and mobile environments.

2 literature Survey

Narayana Moorthy et.al [1] The demand for supporting diverse digital signal processing (DSP) and classification applications on energy-constrained devices has been steadily increasing. These applications frequently involve intensive matrix multiplications using fixed-point arithmetic, often tolerating computational errors to some extent. Therefore, enhancing the energy efficiency of these multiplications is paramount. In this concise study, we introduce novel multiplier architectures designed to optimize computational accuracy and energy consumption trade-offs during the design phase. Compared to conventional precise multipliers, our proposed architectures offer significant energy savings per operation, while maintaining an average computational error of approximately 1%. Furthermore, we demonstrate through rigorous experimentation that such a marginal computational error has negligible effects on the quality of DSP outputs and the accuracy of classification applications. This underscores the practical viability and effectiveness of our proposed multiplier architectures in real-world scenarios, where energy efficiency is paramount without compromising computational integrity.

Zervakis et.al [2] Approximate computing stands as a promising paradigm for designing hardware architectures in applications inherently resilient to errors, prioritizing power savings over absolute accuracy. In this study, we delve into the realm of multi-level approximation, spanning from algorithmic down to circuit-level optimizations, to craft low-power approximate arithmetic architectures tailored for hardware multipliers. Recognizing the limited power savings achievable through singular approximation techniques in isolation, we embark on exploring hybrid methodologies. These methodologies leverage the simultaneous application of multiple techniques spanning different layers of abstraction. We introduce the innovative concept of "perforation" in

approximate arithmetic circuit design, which enables us to navigate the intricate landscape of hybrid designs effectively.

Momeni et.al [3] Inexact computing, also known as approximate computing, emerges as a compelling paradigm, especially in the context of digital processing at nanometric scales. This paper focuses on the analysis and design of two novel approximate 4-2 compressors tailored for integration within a multiplier architecture. These compressors leverage distinct compression features, strategically balancing computational imprecision, quantified through error rate and normalized error distance metrics, with key circuit-centric design parameters such as transistor count, delay, and power consumption. Four distinct methodologies for integrating the proposed approximate compressors within a Dadda multiplier are proposed and comprehensively analyzed. Extensive simulation results are presented, offering insights into the performance characteristics of each design scheme. Moreover, the practical applicability of the approximate multipliers is demonstrated through an illustrative example in image processing, showcasing their efficacy in real-world scenarios. By meticulously exploring the trade-offs between computational accuracy and circuit-level metrics, this study advances our understanding of approximate computing techniques and their potential implications for future digital processing systems operating at nanoscale resolutions.

Chang et.al [4] Approximate computing presents a compelling avenue for applications tolerant to computational errors, offering the potential for meaningful results with reduced power consumption. In this study, the authors introduce a novel imprecise 4-2 compressor tailored for integration within multipliers for image processing applications. Beyond considering solely output values, the authors also analyze pattern distributions to inform the synthesis of the imprecise 4-2 compressor. Comparative analysis against precise counterparts reveals significant advantages offered by the proposed imprecise 4-2 compressor. Specifically, it demonstrates a remarkable 56% reduction in power consumption and a 39% decrease in delay. When integrated into multipliers, the benefits extend further: simulation results indicate a notable 33% improvement in power consumption and a 30% reduction in delay compared to multipliers employing precise components.

Esposito et.al [5] The advent of approximate computing marks a significant trend in digital design, offering a compelling trade-off between exact computation requirements and enhanced speed and power performance. In this paper, we put forth novel approximate compressors along with an algorithm tailored to leverage their capabilities in crafting efficient approximate multipliers. Through the application of our proposed methodology, we have successfully synthesized approximate multipliers catering to various operand lengths utilizing a 40-nanometer library. Comparative analysis against previously introduced approximate multipliers underscores the superiority of the circuits we propose. Specifically, our circuits exhibit superior power efficiency or speed for a given target precision. This advancement represents a significant stride in the realm of approximate computing, as it enables designers to achieve optimized performance characteristics tailored to specific application requirements. The results of our study highlight the efficacy of leveraging approximate computing techniques in the design of digital systems, particularly in scenarios where exact precision is not paramount. By offering improved power or speed performance without compromising on essential precision thresholds, our approach contributes to the ongoing evolution of digital design methodologies, paving the way for more efficient and agile computing systems.

3 Methodology
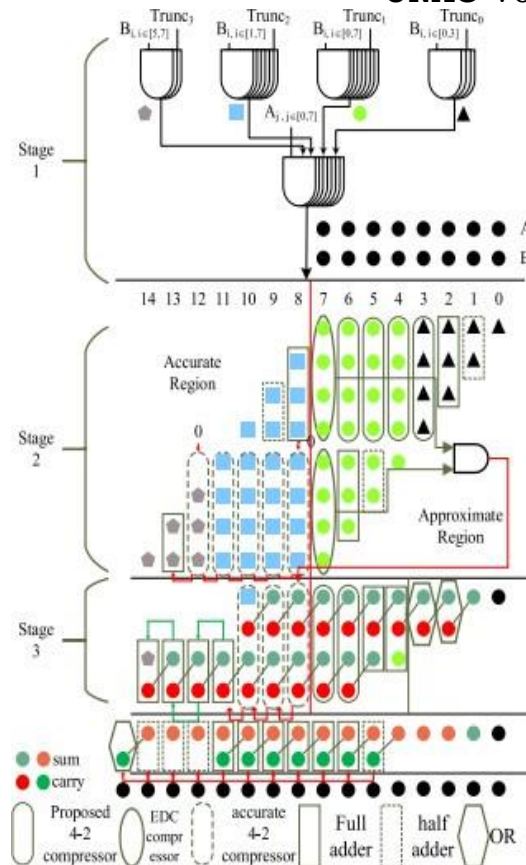
Fig 1 proposed approximate multiplier

The diagram represents showcases an approximate multiplier design, which prioritizes a balance between low power consumption and high accuracy during multiplication operations in a computer system. Here's a detailed breakdown of the different stages within the multiplier circuit:

Input Numbers:

- The diagram likely shows two binary numbers (A and B) represented by a series of bits (0s and 1s). These bits signify the values being multiplied.

Accurate Region:

- This stage focuses on multiplying the most significant bits (MSBs) of numbers A and B. MSBs hold the greatest weight or value within the binary numbers.
- The specific design of this accurate multiplication circuit depends on the implementation, but it likely involves traditional multiplication techniques like full or partial product generation, followed by addition stages, to ensure precise multiplication of these crucial bits.

Approximate Region:

- This section multiplies the least significant bits (LSBs) of A and B. LSBs contribute less to the overall value of the binary number.
- Unlike the accurate region, this stage utilizes an approximate multiplication circuit. This circuit prioritizes lower power consumption and may employ techniques that sacrifice some precision in the product of these LSBs. Examples of approximation techniques include using smaller adders, employing truncation (cutting off some LSBs), or leveraging stochastic computing (using probabilistic methods).

Truncation Control:

- This section acts as a control knob for the approximate multiplication stage. It allows you to define the number of least significant bits used in the approximate multiplication. By adjusting this control, you can strike a balance between:
  o **Accuracy:** Using more LSBs in the approximate multiplication leads to a more accurate final result, but consumes more power.
  o **Power Consumption:** Utilizing fewer LSBs reduces power consumption but may introduce more error in the final product.

Output:

- The final stage combines the results from both the accurate and approximate regions. This combined product retains the high precision from the MSBs and leverages the power-efficient approach from the LSBs.

Overall Significance:

This approximate multiplier design offers a compelling solution for computer systems that require a balance between power efficiency and multiplication accuracy. By strategically applying approximation techniques to less crucial bits, the design achieves significant power savings while maintaining a high degree of accuracy in the final result.

Additional Notes:

- The trade-off between accuracy and power consumption depends on the specific application. Error-tolerant applications might tolerate a slight decrease in accuracy for substantial power savings.
- Research in approximate computing is ongoing, with advancements in techniques and hardware design to further optimize this balance between accuracy and power efficiency.

Results
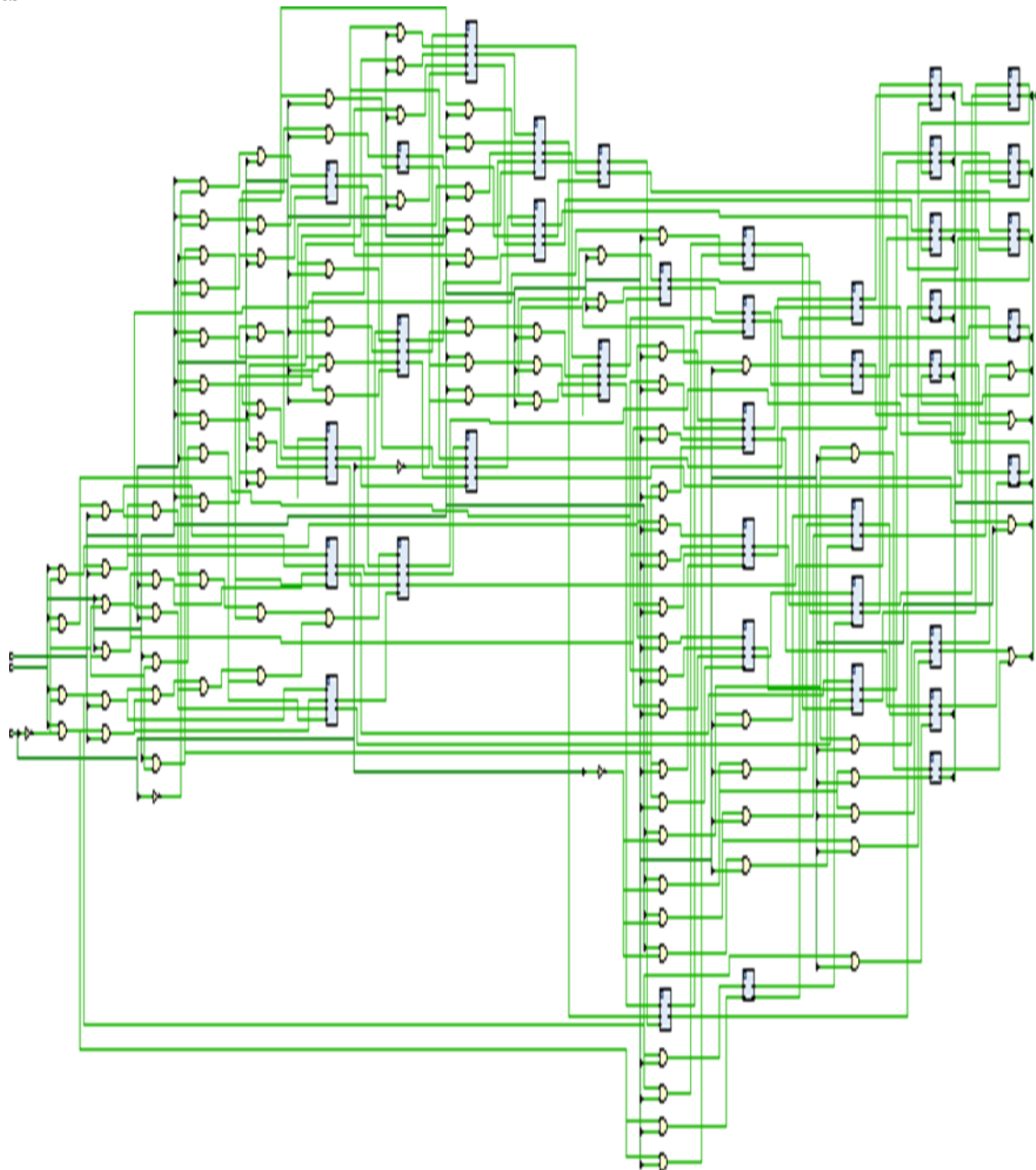
Fig 2 RTL Schematic
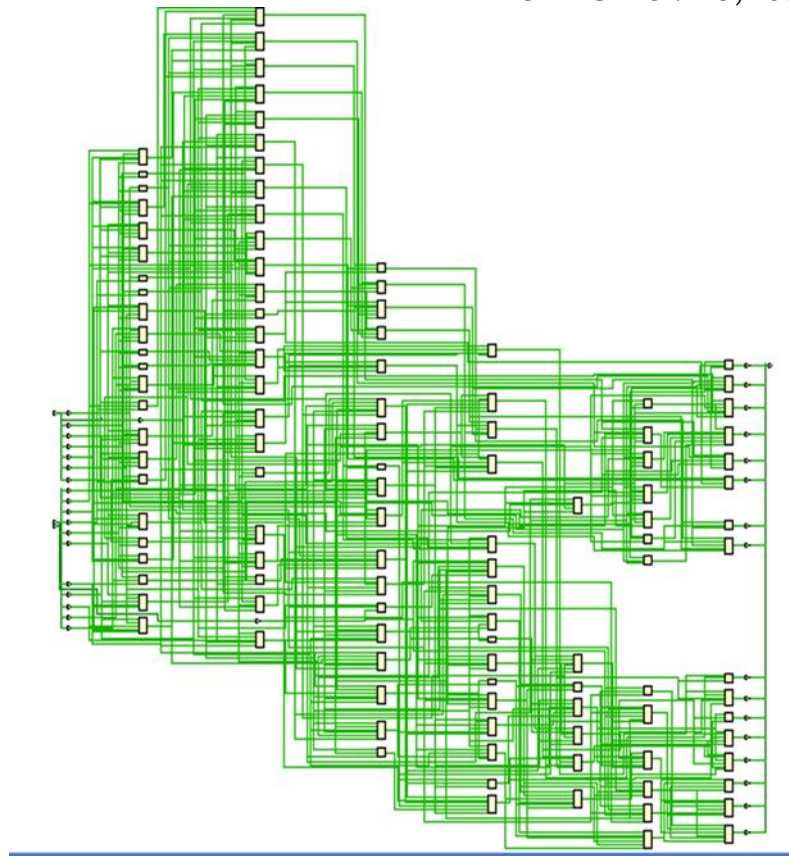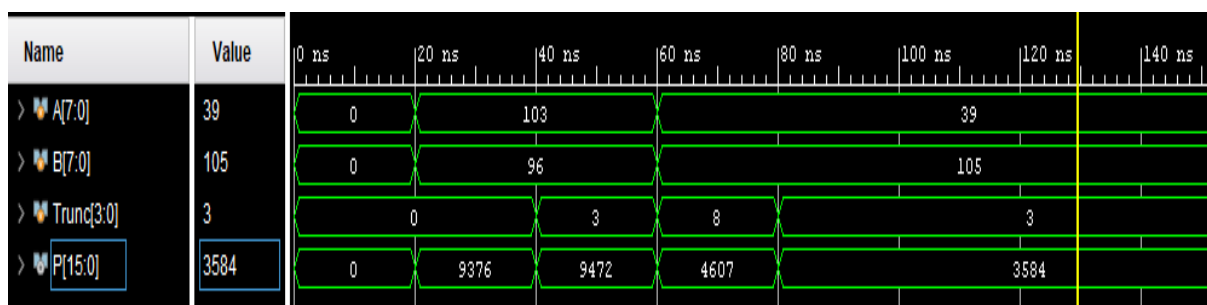
Fig 3: Technology schematic

Fig 4 simulation

Resource	Utilization	Available	Utilization %
LUT	86	134600	0.06
IO	36	400	9.00

Fig 5 Area

```
Max Delay Paths
--------------------------------------------------------------------
Slack:                 inf
  Source:              B[0]
                         (input port)
  Destination:         P[13]
                         (output port)
  Path Group:          (none)
  Path Type:           Max at Slow Process Corner
  Data Path Delay:     8.563ns  (logic 4.031ns (47.072%)  route 4.532ns (52.928%))
  Logic Levels:        9  (IBUF=1 LUT2=1 LUT4=2 LUT5=2 LUT6=2 OBUF=1)
```

Fig 6 Delay

Power estimation from Synthesized netlist. Activity derived from constraints files, simulation files or vectorless analysis. Note: these early estimates can change after implementation.

Total On-Chip Power: 10.437 W

Design Power Budget: **Not Specified**

Fig 7 Power

	Area (in LUT's)	Delay (in ns)	Power (in Watts)
Proposed	86	8.563	10.437
Extension method	86	5.537	10.657

Table 1 Evaluation table for Area, Delay, power

Conclusion

In conclusion, our research presents a comprehensive exploration of pipelined approximate multiplier design, leveraging advancements in approximate computing techniques. We address the critical challenge of power consumption associated with multiplication operations by introducing innovative methodologies that balance accuracy, power efficiency, and performance. Through the integration of pipelining and novel 4-2 compressor designs, we enhance the efficiency of our approximate multiplier architecture. The adjustable nature of our multiplier allows users to dynamically adjust accuracy levels, catering to diverse application requirements without compromising on performance. Additionally, we introduce error compensation mechanisms to bolster the reliability of computations, further enhancing the robustness of our design. Experimental validation demonstrates the tangible benefits of our approach, showcasing significant improvements over traditional accurate Wallace multiplier implementations. Notably, our pipelined approach exhibits superior performance metrics compared to non-pipelined designs, underscoring the efficacy of our proposed methodologies. The practical feasibility of our designs is underscored by their compatibility with Xilinx Vivado 2018.3, ensuring seamless integration within existing hardware environments. Overall, our research contributes to the advancement of approximate computing techniques, offering practical solutions for power-efficient and high-performance multiplier design in various applications.

Feature Scope

The feature scope of this research entails the development and integration of advanced techniques in approximate computing to address power consumption challenges while maintaining acceptable levels of accuracy in multiplier designs. Key features include the conceptualization and implementation of a pipelined approximate multiplier, alongside the introduction of a novel 4-2 compressor design with heightened accuracy. Additionally, the research proposes an adjustable approximate multiplier capable of dynamically adapting accuracy levels to suit specific runtime demands, complemented by a straightforward error compensation mechanism to enhance computational reliability. Experimental validation through simulations and synthesis using tools like Xilinx Vivado 2018.3 serves to assess

performance metrics such as power consumption, delay, and parameter values, thereby demonstrating the efficacy of the proposed methodologies.

References

[1] A. Wang, B. H. Calhoun, and A. P. Chandrakasan, Sub-Threshold Design for Ultra Low-Power Systems, vol. 95. New York, NY, USA: Springer, 2006.

[2] Q. Xu, T. Mytkowicz, and N. S. Kim, "Approximate Computing: A Survey," IEEE Design & Test, vol. 33, no. 1, pp. 8-22, Feb. 2016.

[3] V. Leon, G. Zervakis, D. Soudris, and K. Pekmestzi, "Approximate Hybrid High Radix Encoding for Energy-Efficient Inexact Multipliers," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 26, no. 3, pp. 421– 430, Mar. 2018.

[4] C.–H. Chang, J. Gu, and M. Zhang, "Ultra low-voltage low-power CMOS 4-2 and 5-2 compressors for fast arithmetic circuits," IEEE Trans. Circuits and Syst. I: Reg. Papers, vol. 51, no. 10, pp. 1985-1997, Oct. 2004.

[5] A. Momeni, J. Han, P. Montuschi, and F. Lombardi. "Design and Analysis of Approximate Compressors for Multiplication," IEEE Trans. Comput., vol. 64, no. 4, pp. 984-994, Apr. 2015.

[6] O. Akbari, M. Kamal, A. Afzali-Kusha, and M. Pedram. "Dual-Quality 4:2 Compressors for Utilizing in Dynamic Accuracy Configurable Multipliers," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol.25, no. 4, pp. 1352-1361, Apr. 2017.

[7] Z. Yang, J. Han, and F. Lombardi. "Approximate compressor for error resilient multiplier design," IEEE Int. Symp. on Defect and Fault Tolerance in VLSI and Nano. Syst. (DFTS), Amherst, MA, 2015, pp. 183- 186.

[8] M. Ha and S. Lee. "Multipliers with Approximate 4-2 Compressors and Error Recovery Modules," IEEE Embedded Systems Letters, vol. 10, no. 1, pp. 6-9, Mar. 2018.

[9] L. Qian, C. Wang, W. Liu, F. Lombardi, and J. Han, "Design and evaluation of an approximate Wallace-Booth multiplier," 2016 IEEE Int. Symp. Circuits and Syst. (ISCAS), Montreal, QC, 2016, pp. 1974-1977.

[10] X. Yi, H. Pei, Z. Zhang, H.Zhou, and Y. He. "Design of an EnergyEfficient Approximate Compressor for Error-Resilient Multiplications," 2019 IEEE Int. Symp. Circuits and Syst. (ISCAS). Sapporo, Japan, 2019, pp. 1-5.