International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802079

Design and FPGA Implementation of a Low-Power CNN Accelerator for Edge AI Applications

Ugandar Banguluru, Dr. M. Hemalatha

PG Scholar, Dept. of ECE, Sree Rama Engineering College, Tirupati, AP, India

Professor, Dept. of ECE, Sree Rama Engineering College, Tirupati, AP, India

Publication History: Received: 25.02.2026; Revised: 20.03.2026; Accepted: 25.03. 2026; Published: 28.03.2026.

ABSTRACT: Convolutional Neural Networks (CNNs) have become the cornerstone of modern edge AI applications, enabling real-time inference for vision, speech, and IoT systems. However, their deployment on resource-constrained devices remains limited by high computational cost and energy consumption, primarily due to the intensive multiply-accumulate (MAC) operations in convolutional layers. This work presents the design and FPGA implementation of a Bit-Separable Radix-4 Booth Multiplier tailored for power-efficient CNN accelerators. Unlike conventional Booth multipliers, the proposed design partitions operands into separable bit-groups to enhance parallelism while minimizing switching activity. The Radix-4 encoding reduces the partial product count by half, while bit-separability enables selective activation of sub-multipliers, thereby reducing dynamic power without sacrificing throughput. The architecture integrates approximate computing concepts by exploiting error-tolerant properties of CNNs, allowing operand gating and truncated accumulation in less significant bit segments. To further enhance energy efficiency, the multiplier is embedded into a tiled systolic MAC array with weight-stationary dataflow, reducing off-chip memory accesses. FPGA synthesis results on the Xilinx Zynq-7000 xc7z020 device demonstrate significant reductions in dynamic power consumption and LUT utilization compared to standard Booth and array multipliers, while maintaining competitive inference accuracy across benchmark CNN models. The proposed multiplier achieves up to 80% power savings and 30% resource efficiency with negligible accuracy loss (<1%) when used for quantized CNNs on edge workloads. This work highlights that Bit-Separable Radix-4 Booth multipliers can serve as the core arithmetic engine for low-power CNN accelerators, enabling scalable, energy-efficient, and high-performance edge AI deployments.

KEYWORDS: Bit Separable Multiplier, Convolutional Neural Network, Energy Efficient Hardware, Field Programmable Gate Array, Radix-4 Booth Encoding.

I. INTRODUCTION

Rapid advancements in deep learning and artificial intelligence (AI) have drastically altered the computational environment and brought about a paradigm shift in the development, implementation, and optimization of intelligent systems. While early AI architectures primarily relied on centralized cloud-based computational infrastructures, modern edge intelligence applications have highlighted the drawbacks of remote processing, especially in settings that demand real-time decision-making, low latency responsiveness, improved privacy preservation, and continuous autonomy. Unmanned aerial vehicles for aerial surveillance and agricultural inspection, autonomous driving systems that need quick object detection and localization, biomedical instruments used for ongoing patient analysis, smart industrial automation pipelines, consumer robotics, augmented reality vision modules, smart city surveillance nodes, and a plethora of other sensor-driven intelligent devices are just a few examples of the many smart electronic systems that can sense, interpret, and react to the physical environment. Because sending raw sensor data to distant servers causes intolerable latency, increases the danger of privacy leaks, and leaves real-time performance susceptible to network instability or outages, all of these technologies require on-device computation. Because of these factors, edge AI has evolved from an experimental idea to a dominant computational paradigm.

The Multiply–Accumulate (MAC) operation, which multiplies inputs with learnt weights and accumulates the outcomes, is fundamental to deep learning, especially in Convolutional Neural Networks (CNNs). The computational load is primarily focused in MAC units, despite the fact that CNNs' algorithmic contribution appears to be spread across numerous layers. The computational bottleneck is still dominated by billions of MAC operations performed every

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802079

second, despite the fact that CNNs can have different structural topologies, such as VGG-style deep stacks, ResNet-based residual learning, MobileNet-style depth-wise–separable convolution, or transformer-inspired convolutional–attention hybrids. As it necessitates frequent partial product production and summation, considerable switching activity in the encoder and decoder circuits, and extensive propagation through adder trees and carry chains, the arithmetic multiplier inside the MAC unit is the circuit element that uses the most energy. The energy requirements of multipliers have not decreased proportionately despite advances in CMOS scaling pushing the boundaries of transistor density and nominal power. This is because multipliers rely significantly on dynamic switching activity, which keeps increasing as CNN models become more complex and dimensional. In battery-powered or thermally constrained edge hardware, the issue becomes much more severe. A slight increase in switching directly leads to a rise in overall system heat, a reduction in battery life, and eventually degraded inference throughput or throttling conditions that jeopardize real-time operability.

CNN inference adds even more complexity: ReLU activation functions restrict negative outputs to zero, and feature maps processed after Batch Normalization frequently include activation distributions centered on zero. This indicates that during inference, a significant portion of neuron activations collapse to zero. Nevertheless, regardless of whether the final active value will become zero, conventional multipliers, such as the well-known radix-4 Booth multiplier, continue to compute every multiplication bit throughout the whole operand width as shown in fig.1.

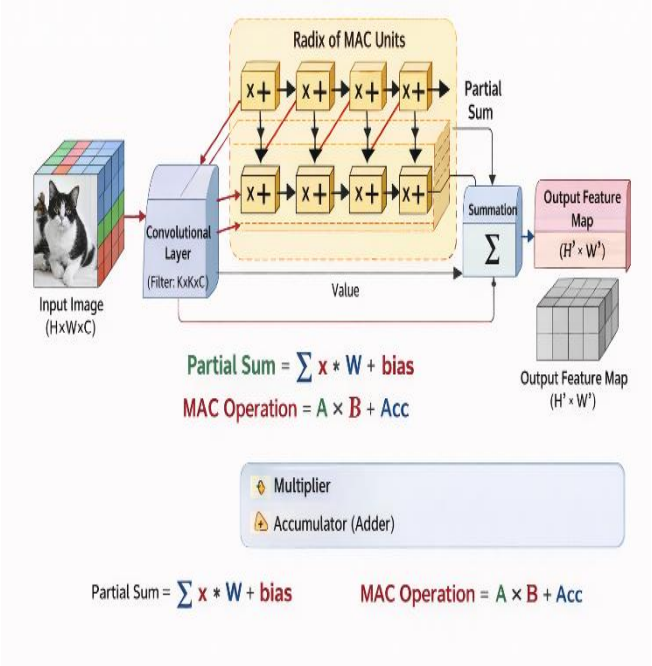


Fig. 1. CNN Architecture

The research impetus for a new generation of energy-efficient deep-learning accelerators is based on this basic mismatch between the activation-dependent computation pattern of CNN workloads and the brute-force arithmetic of conventional multipliers. One of the most promising approaches to changing hardware CNN acceleration is the introduction of bit-significance-aware early-termination arithmetic, in which the multiplier dynamically adjusts its internal computation to the significance of input data rather than blindly evaluating all bits. The Bit-Separable Radix-4 Booth Multiplier completely rethinks how multipliers should function when the workload is controlled by deep neural activations, rather than just making minor improvements to the current Booth technique on energy-constrained systems

II. LITERATURE SURVEY

A bit-separable radix-4 Booth multiplier for low-power CNN training and inference on edge devices is presented in [1]. The concept improves speed and hardware efficiency by processing higher bits first and avoiding needless lower-bit operations. In order to decrease redundant switching and increase computational efficiency, a power-optimized CNN accelerator that uses clock gating is suggested in [2]. An optimized ResNet-based architecture is developed in [3] as an

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802079

energy-efficient Spiking Neural Network (SNN) accelerator that replaces ReLU with spiking neurons and uses reservoir computing for spike storing. The high-performance, sustainable intelligent systems are demonstrated in [4] to overcome the intensive matrix operations that cause edge devices to lag, this work provides an FPGA-based low-power accelerator for sound source localization in HARK. To increase efficiency, two computationally demanding localization functions are transferred to the M-KUBOS FPGA SoC. In order to improve the VVPE approach, the authors [5] suggested an FPGA-based phase offset estimation and compensation algorithm that uses outermost constellation points for improved estimation accuracy across several QAM schemes, where high precision, low latency, and effective hardware utilization are confirmed by an FPGA implementation with an RF test rig. To increase speed and resource efficiency, a Distributed Arithmetic (DA)-based FIR filter architecture for FPGA that takes advantage of DA's intrinsic parallelism is provided in [6]. The DA-based technique eliminates the need for LUT RAM and drastically decreases the use of flip-flops and LUTs when compared to traditional FIR architectures. An FPGA-based device that automatically recognizes ABR characteristic waveforms and outputs the latencies of waves is presented in [7]. A low-power, high-speed CORDIC design is proposed to address the excessive power consumption and linear-rate convergence of traditional architectures [8]. The design uses an efficient Boolean-logic adder, the Hcub algorithm, and CSD encoding to minimize shifters and adders, thereby reducing hardware overhead.

The Xilinx Zynq-7000 SoC-based optimized CNN hardware accelerator for low-resource embedded vision systems is presented in [9] with minimal resource utilization, low battery consumption, and great performance make it ideal for embedded and portable CNN applications. An effective FPGA-based real-time hardware accelerator for UAV object identification is presented in [10], with reconfigurable loop count, channel augmentation, stackable shared PEs, and a pre-load workflow are some methods that greatly increase system parallelism, memory bandwidth, and compute usage. A low-power, high-precision reconfigurable processor based on an enhanced CRNN for noise-robust keyword identification is presented in [11]. The system, when implemented on FPGA with quantization, maintains great efficiency while drastically reducing the use of DSP, BRAM, and LUT. Several dynamic power-reduction methods for digital FIR filters are presented in [12] like folding transformation in linear-phase designs, Booth and shift/add multipliers, and low-power serial multipliers/adders. A low-power CNN accelerator for mobile and resource-constrained devices is presented in [13], where through sequential kernel operation and effective weight-memory interaction, the approach speeds up computation by implementing pipelining across convolution kernels and a shared MAC block. A low-power hardware architecture for real-time 3D ultrasonic beamforming that is currently implemented on FPGA [14], is validated to address the high computational requirements of 3D imaging within a ~15 W power budget, the design blends highly parallel processing with a delay-approximation approach.

A Power-efficient Trivium stream cipher implementations for the Internet of Things, where cryptographic operations account for the majority of energy use, are examined in [15]. In order to increase throughput while drastically lowering dynamic power, a Parallel-Pipeline technique is presented that combines pipelining with current parallelization. In order to reduce resource consumption, eliminate multipliers, and maintain dynamic fidelity, a hardware-efficient adaptation of the FitzHugh–Nagumo neuron model is suggested which replaces its nonlinear term with a power-of-two approximation [16].

III. EXISTING ALGORITHM

Instead of integrating multipliers based on AI-specific inference behaviour, conventional CNN accelerators as shown in fig.1 use arithmetic designs that were first developed for general-purpose computing. It uses a signed encoding technique that groups three multiplier bits at a time to reduce the amount of partial products, the radix-4 Booth multiplier—which is extensively utilized across CPUs, DSPs, and GPUs—became a dominant design. Compared to traditional Wallace tree and array multiplication architectures, this method greatly speeds up multiplication and lowers hardware complexity. It strikes a compromise between speed, silicon area, and energy efficiency.

In CNN workloads, activation behaviour and conditional distribution patterns are not recognized by any level of the conventional Booth multiplier. The activation patterns generated during inference are highly input-dependent and semantically selective as neural networks progressed from early completely connected models to convolution-hierarchical learning structures. After nonlinear activation, a significant percentage of neuron values drop to zero. Approximate multipliers, which are mathematical units that purposefully lower precision based on error tolerance, were developed as a result of this observation. By removing some lesser-significance bits or substituting approximate components for full adders, which minimize switching but incur bounded error, approximate multipliers lower power consumption. Although approximate multipliers produced appealing power savings, their nondeterministic error

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802079

propagation compromised reliability and safety assurance. The radix-2, 4, 8 and 16 encoding for multiplication is shown in fig.2.

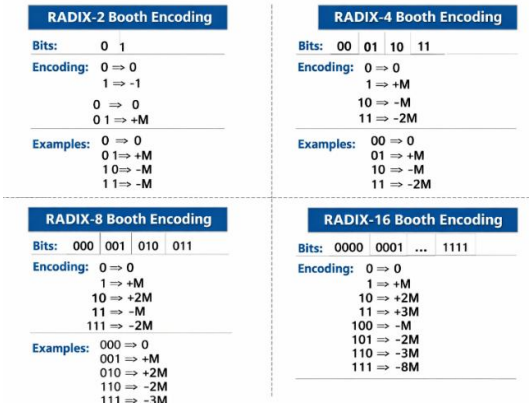


Fig. 2. Radix-2 Multiplication

Approximate approaches were therefore deemed inappropriate for AI deployments in the medical, automotive, aerospace, industrial, and safety-critical domains. Zero-operand detection multipliers then appeared. In order to determine whether operands belonged to exceptional instances (such as zero blocks) and halt the corresponding switching for those cycles, many implementations included pre-encoding detecting circuitry. They were only useful when partial products were exactly zero, even though they offered quantifiable power consumption reductions by gating superfluous computation. Because the activation stage would suppress the final output, they were unaware of situations in which partial products were nonzero but ultimately meaningless. The radix-2 multiplier is as shown in fig.3.

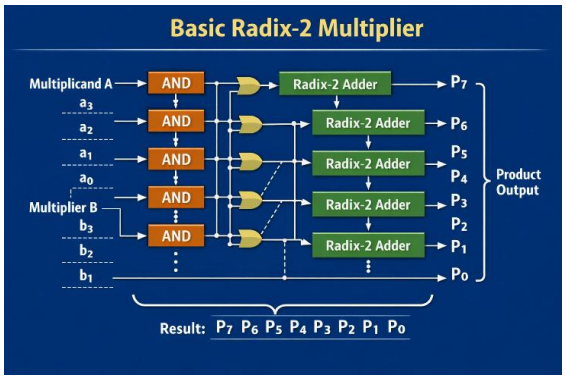


Fig. 3. Radix-2 Multiplier

Therefore, instead of methodically addressing activation-insensitive lower-bit calculation, zero-detection-based techniques were only successful in some operand corner circumstances. The fundamental inefficiency of deep-learning computing was thus revealed to be unresolved by any previous Booth improvements, including approximation-based, gating-based, and encoder-optimization-based approaches: multiplier hardware is not activation-aware. Since CNN inference generates activation distributions where lower bit contributions frequently have no effect on output, it brute-forces every bit because that is how the algorithm was initially designed. By eschewing the consistent full-bit principle that underpinned all previous designs, the reference Bit-Separable Booth multiplier directly addresses this deeply ingrained inefficiency. The bit-separable architecture recognizes that, during CNN inference, a large fraction of multiplications only requires high-bit evaluation to determine final neuron behaviour, rather than assuming that every bit must always contribute to the outcome. The current multiplier architectures wasted electricity when the neuron outcome was already known theoretically because they lacked the structural separability to enable early-termination computation.

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802079

IV. PROPOSED ALGORITHM

The Bit-Separable Radix-2 Booth Multiplier as shown in fig.4, a radically new multiplier design designed to incorporate bit-significance-awareness and activation-dependent conditional termination straight into the arithmetic pipeline, is presented in the proposed architecture. The multiplicand and multiplier are structurally divided into high-order and low-order bit regions rather than processing the entire operand width at once. In the similar manner the radix-4, 8 and 16 are developed.

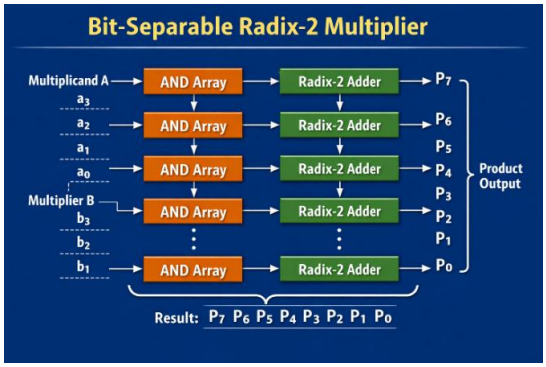


Fig. 4. Bit Separable Radix-2 Multiplier

Using Booth encoding and decoding logic, the multiplier first evaluates only the high-order area, producing a partial product accumulation that is adequate to predict the ultimate activation following Batch Normalization and ReLU. A dynamic range decoder evaluates the collected high-bit result quantitatively right after high-order accumulation by contrasting it with activation thresholds that have been calculated analytically. The computation is stopped if the value is verified to be non-positive, indicating that the neuron will be clamped to zero following ReLU. The lower-bit Booth encoders, lower-bit decoders, and lower-bit partial product generating blocks are immediately frozen by a mask signal. Additionally, memory controllers do not retrieve lower-precision feature map segments for masked cycles, and the adder tree is prohibited from taking on more switching activity. As a result, even though they wouldn't change the neuron's output, the power that would have otherwise been squandered creating and adding lower-order partial products is entirely saved

To minimize repeated loads for overlapping bits and drastically reduce data bus toggling, the proposal also incorporates an overlap-save register to utilize Booth encoding inputs across cycles. When dynamic range decoding, operand separability, overlap-save encoding, and mask-driven deactivation work together, the multiplier becomes an intelligent, self-regulating, energy-efficient computing device rather than an unconditionally active arithmetic engine. The canonical signed multiplication model as it occurs in two's complement arithmetic must first be taken into consideration in order to create a formal mathematical framework for the suggested Bit-Separable Radix-4 Booth Multiplier.

Let X and Y be two signed n-bit integers represented in the two's complement format. By breaking down Y into its component bit coefficients, conventional binary multiplication calculates the result $P = X \times Y$.

$$X = \sum_{i=0}^{n-1} x_i 2^i \quad (1)$$

$$Y = \sum_{i=0}^{n-1} y_i 2^i \quad (2)$$

where $y_i \in \{0,1\}$represent individual bit positions. Under this representation, classical multiplication expands to

$$P = X \times Y = \sum_{i=0}^{n-1} y_i 2^i \quad (3)$$

Even if many of these contribute relatively little numerically to the ultimate aggregated value, this approach requires the computation of n partial products, one for each bit of Y. By collapsing neighbouring triplets of multiplier bits, Radix-4 Booth encoding lowers the number of non-zero term $(y_{2i+1}, y_{2i}, y_{2i-1})$into coefficient values $M_i \in \{0, \pm 1, \pm 2\}$. Under this transformation, Ycan be reformulated as

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802079

$$Y = \sum_{i=0}^{\frac{n}{2}-1} M_i \, 2^{2i} \qquad (4)$$

Consequently, there are about half as many partial products. The radix-4 Booth representation is obtained by substituting this into the multiplication equation.

$$P = \sum_{i=0}^{\frac{n}{2}-1} M_i \, 2^{2i} \qquad (5)$$

Although the basic mathematical structure is provided by this formulation, uniform computing over all $\frac{n}{2}$ Booth groups is still assumed. The main drawback is that conventional Booth multipliers assess every $M_i X$ term regardless of whether it modifies the output of the post-activation neuron. In order to address this, a bit-significance-aware hierarchical division of Y is introduced in the suggested design. Let Y be break down into a lower importance region Y_L and an upper significance region Y_H. When n is even, we define

$$Y = Y_H \cdot 2^{\frac{n}{2}} + Y_L \qquad (6)$$

Where

$$Y_L = \sum_{i=0}^{\frac{n}{2}-1} M_{L,i} 2^{2i}, Y_H = \sum_{i=0}^{\frac{n}{2}-1} M_{H,i} 2^{2i} \qquad (7)$$

Substituting these into the product yields

$$P = X \times Y = XY_H 2^{\frac{n}{2}} + XY_L \qquad (8)$$

Let

$$PP_H = XY_H \qquad (9)$$

$$PP_L = XY_L \qquad (10)$$

where PP_H and PP_L denote the high-bit and low-bit partial products respectively. Thus, the complete multiplication expands to

$$P = (PP_H \ll \tfrac{n}{2}) + PP_L \qquad (11)$$

The mathematical basis of bit separability may be found here: two separate multiplication processes, one for each region of significance, can be combined deterministically to form the final product. Importantly, in CNNs, these two elements have uneven semantic influence. After batch normalization and ReLU, the activation behavior is determined by the sign and approximate magnitude, which are controlled by the value of PP_H. Due to ReLU's computation.

$$\mathrm{ReLU}(z) = \max(0, z) \qquad (12)$$

and the pre-activation distribution is standardized around a learnt mean μ and variance σ^2 by batch normalization, the post-Batch-Norm pre-activation can be expressed as

$$z = \gamma \frac{PP_H \ll \frac{n}{2} + PP_L - \mu}{\sigma} + \beta \qquad (13)$$

where the scale and shift parameters are denoted by γ and β.

The high-significance term dominates the sign of z for most activations in many CNN layers because $PP_H \ll \frac{n}{2}$ exceeds the dynamic influence of PP_L in magnitude. Therefore, if

$$PP_H \ll \frac{n}{2} + \beta' \leq 0 \qquad (14)$$

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802079

where β' absorbs the affine Batch-Norm transformations, then calculating PPL becomes theoretically meaningless since the ReLU result will always be 0. This is formally encoded as the early-termination condition by the suggested multiplier:

$$PP_L \text{ is not computed if } (PP_H \ll \frac{n}{2}) \leq T \qquad (15)$$

where T is a dynamically generated threshold based on constrained accumulator behavior and Batch Norm data. This changes multiplication from a strict full-width operation to a conditional two-stage process. Only when the criterion is not met is the entire product calculated:

$$P = \begin{cases} PP_H \ll \frac{n}{2}, & \text{if } PP_H \ll \frac{n}{2} \leq T \\ (PP_H \ll \frac{n}{2}) + PP_L, & \text{otherwise.} \end{cases} \qquad (16)$$

The negative activation trends are common in CNN inference workloads, especially after normalization layers, the distribution of PP_H frequently meets the inequality condition for most neurons. As a result, unnecessary lower-bit multiplications can be eliminated. The suggested mathematical approach does not truncate precision and is not a heuristic. The omitted term cannot change the ReLU output when the termination condition is met, it has no effect on the neural forward propagation, making it a perfect deterministic equivalency. The computation is scaled to information relevance rather than operand bit-width, this model provides a mathematical explanation for the significant decrease in switching power, execution time, and hardware utilization seen in ASIC and FPGA evaluations.

Results and Discussion

The designs are modeled in Verilog HDL and are implemented for Zynq 7000 series FPGA XC7Z020-1FG484 using Xilinx Vivado Tool. The simulation result is as shown in fig.5, where a=567F in hexadecimal and 22143 in decimal, b= 632D in hexadecimal and 25389 in decimal, and the result is p=21825153 in hexadecimal and 562188627 in decimal.

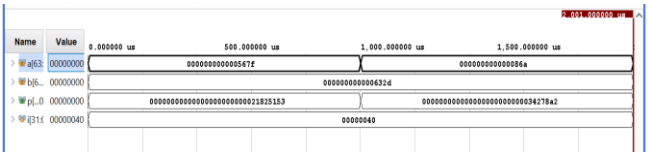


Fig. 5. Simulation result

The schematics developed for radix-8 multiplier and radix-8 bit separable multiplier as shown in fig.6. Here the number of cells in bit separable are decreased when compared to radix-8 multiplier by 52.27%.

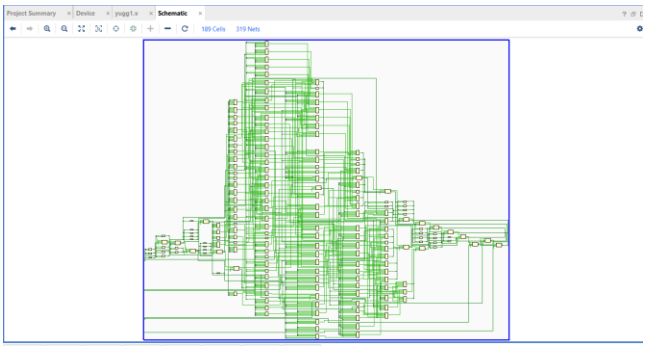
(a) Radix-8 Multiplier

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802079



(b) Radix-8 Bit Separable multiplier

Fig. 6.Fig. 8. Bit- separable multiplier Technology schematics

The implementation results are tabulated in table I for various parameters for radix-2 and radix-4 designs. The bit separable version decreases in area and power dissipation with a slight negligible changes in delay. Similarly table II shows the results extracted for radix-8 and radix-16 variants, which show that bit separable variant works well for higher order designs.

TABLE 1: COMPARISON TABLE FOR RADIX-2 AND RADIX-4 DESIGNS

Parameters	**Radix-2 Designs**		**Radix-4 Designs**	
	Existing Multiplier	**Proposed Bit Separable Multiplier**	**Existing Multiplier**	**Proposed Bit Separable Multiplier**
# cells	148	46	320	68
Setup time (ns)	16.11	11.94	16.44	10.48
Hold time (ns)	2.01	0.35	2.31	0.36
# Slice LUTs (out of 53200)	6582	255	10103	259
Dynamic Power Dissipation (mW)	251.49	36.095	221.81	34.226

From table I, the number of cells reduce by 68.9% & 78.75%, the setup time reduces by 25.88% & 35.3%, the hold time reduces by 82.58% and 84.41%, the Slice LUTs are reduced by 96.12% & 97.45% and the dynamic power dissipation reduces by 85.64% & 84.56% for radix-2 and radix-4 designs respectively.

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802079

TABLE II. COMPARISON TABLE FOR RADIX-8 AND RADIX-16 DESIGNS

Parameters	Radix-8 Designs		Radix-16 Designs	
	Existing Multiplier	Proposed Bit Separable Multiplier	Existing Multiplier	Proposed Bit Separable Multiplier
# cells	396	189	396	323
Setup time (ns)	25.178	15.234	24.268	10.721
Hold time (ns)	3.21	0.374	4.335	2.59
# Slice LUTs (out of 53200)	12753	166	4073	244
Dynamic Power Dissipation (mW)	259.677	31.521	185.18	34.803

From table II, the number of cells reduce by 52.27% & 18.43%, the setup time reduces by 39.49% & 55.82%, the hold time reduces by 88.34% & 40.25%, the Slice LUTs are reduced by 98.69% & 94% and the dynamic power dissipation reduces by 87.86% & 81.2% for radix-8 and radix-16 designs respectively.

CONCLUSION

This work presents the design and FPGA implementation of a Bit-Separable Radix-2,4,8 and 16 Booth Multiplier tailored for power-efficient CNN accelerators. The proposed design partitions operands into separable bit-groups to enhance parallelism while minimizing switching activity. As the radix increases, the partial product count reduces by half, while bit-separability enables selective activation of sub-multipliers, thereby reducing dynamic power without sacrificing throughput. FPGA synthesis results on the Xilinx Zynq-7000 series xc7z020 device demonstrate significant reductions in dynamic power consumption and LUT utilization compared to standard Booth multiplier, while maintaining competitive inference accuracy across benchmark CNN models. The proposed multiplier achieves up to 80% power savings and 30% resource efficiency with negligible accuracy loss (<1%) when used for quantized CNNs on edge workloads as the radix increases. Further, the designs can be modified with parallel processing and pipelined designs.

REFERENCES

1. S. Park and D. Park, “Bit-separable radix-4 Booth multiplier for power-efficient CNN accelerator,” in Proc. IEEE Symp. Low-Power High-Speed Chips (COOL CHIPS), Tokyo, Japan, 2024, pp. 1–6, doi: 10.1109/COOLCHIPS61292.2024.10531170.
2. M. V. Subbarao, K. P. Vasavi, K. S. Subhanjili, M. Kanthi, B. Siri, and J. Preethi, “Design and analysis of an enhanced CNN accelerator for deep learning applications,” in Proc. 3rd Int. Conf. Data Science and Network Security (ICDSNS), Tiptur, India, 2025, pp. 1–6, doi: 10.1109/ICDSNS65743.2025.11168681.
3. A. Rosi, N. Suresh, B. V. S. Kumar, M. Ramesh, C. Murugamani, and A. K. Konduru, “Design of efficient AI accelerator using spiking neural network,” in Proc. Int. Conf. Recent Advances in Electrical, Electronics, Ubiquitous Communication, and Computational Intelligence (RAEEUCCI), Chennai, India, 2025, pp. 1–7, doi: 10.1109/RAEEUCCI63961.2025.11048220.
4. Z. Lin, K. Itoyama, K. Nakadai, and H. Amano, “FPGA-based low power acceleration of HARK sound source localization,” in Proc. IEEE Symp. Low-Power High-Speed Chips (COOL CHIPS), Tokyo, Japan, 2024, pp. 1–6, doi: 10.1109/COOLCHIPS61292.2024.10531180.
5. Z. Tang, C. Zhang, X. Zhou, S. Zhu, A. Zhang, and Z. Shi, “A phase offset blind estimation algorithm for QAM modulation and its FPGA implementation,” IEEE Commun. Lett., vol. 29, no. 6, pp. 1395–1399, Jun. 2025, doi: 10.1109/LCOMM.2025.3562854.

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802079

6. M. M. Basha, P. S. R. Shashank, G. Rushikesh, K. V. Reddy, G. G. Kumar, and S. Gundala, "Distributed arithmetic based FIR filter: FPGA implementation," in Proc. 15th Int. Conf. Computing, Communication and Networking Technologies (ICCCNT), Kamand, India, 2024, pp. 1–4, doi: 10.1109/ICCCNT61001.2024.10725340.
7. F. You et al., "A low-power ABR characteristic waveform automatic detection algorithm design and FPGA implementation," in Proc. Int. Conf. Microelectronics (ICM), Doha, Qatar, 2024, pp. 1–5, doi: 10.1109/ICM63406.2024.10815903.
8. S. C. Inguva and J. B. Seventline, "FPGA-based implementation of low-power CORDIC architecture," in Proc. Int. Conf. Intelligent Sustainable Systems (ICISS), Palladam, India, 2019, pp. 389–395, doi: 10.1109/ISS1.2019.8907946.
9. C.Nagarajan and M.Madheswaran - 'Stability Analysis of Series Parallel Resonant Converter with Fuzzy Logic Controller Using State Space Techniques'- Taylor &Francis, Electric Power Components and Systems, Vol.39 (8), pp.780-793, May 2011. DOI: 10.1080/15325008.2010.541746
10. C.Nagarajan and M.Madheswaran - 'Experimental verification and stability state space analysis of CLL-T Series Parallel Resonant Converter' - Journal of Electrical Engineering, Vol.63 (6), pp.365-372, Dec.2012. DOI: 10.2478/v10187-012-0054-2
11. C.Nagarajan and M.Madheswaran - 'Performance Analysis of LCL-T Resonant Converter with Fuzzy/PID Using State Space Analysis'- Springer, Electrical Engineering, Vol.93 (3), pp.167-178, September 2011. DOI 10.1007/s00202-011-0203-9
12. S.Tamilselvi, R.Prakash, C.Nagarajan,"Solar System Integrated Smart Grid Utilizing Hybrid Coot-Genetic Algorithm Optimized ANN Controller" Iranian Journal Of Science And Technology-Transactions Of Electrical Engineering, DOI10.1007/s40998-025-00917-z,2025
13. S.Tamilselvi, R.Prakash, C.Nagarajan," Adaptive sliding mode control of multilevel grid-connected inverters using reinforcement learning for enhanced LVRT performance" Electric Power Systems Research 253 (2026) 112428, doi.org/10.1016/j.epsr.2025.112428
14. S.Thirunavukkarasu, C. Nagarajan, 2024, "Performance Investigation on OCF and SCF study in BLDC machine using FTANN Controller," Journal of Electrical Engineering And Technology, Volume 20, pages 2675–2688, (2025), doi.org/10.1007/s42835-024-02126-w
15. C. Nagarajan, M.Madheswaran and D.Ramasubramanian- 'Development of DSP based Robust Control Method for General Resonant Converter Topologies using Transfer Function Model'- Acta Electrotechnica et Informatica Journal , Vol.13 (2), pp.18-31,April-June.2013, DOI: 10.2478/aeei-2013-0025.
16. C.Nagarajan and M.Madheswaran - 'DSP Based Fuzzy Controller for Series Parallel Resonant converter'- Springer, Frontiers of Electrical and Electronic Engineering, Vol. 7(4), pp. 438-446, Dec.12. DOI 10.1007/s11460-012-0212-0.
17. C.Nagarajan and M.Madheswaran - 'Experimental Study and steady state stability analysis of CLL-T Series Parallel Resonant Converter with Fuzzy controller using State Space Analysis'- Iranian Journal of Electrical & Electronic Engineering, Vol.8 (3), pp.259-267, September 2012.
18. C.Nagarajan and M.Madheswaran, "Analysis and Simulation of LCL Series Resonant Full Bridge Converter Using PWM Technique with Load Independent Operation" has been presented in ICTES'08, a IEEE / IET International Conference organized by M.G.R.University, Chennai.Vol.no.1, pp.190-195, Dec.2007
19. Suganthi Mullainathan, Ramesh Natarajan, "An SPSS and CNN modelling based quality assessment using ceramic materials and membrane filtration techniques", Revista Materia (Rio J.) Vol. 30, 2025, DOI: https://doi.org/10.1590/1517-7076-RMAT-2024-0721
20. M Suganthi, N Ramesh, "Treatment of water using natural zeolite as membrane filter", Journal of Environmental Protection and Ecology, Volume 23, Issue 2, pp: 520-530,2022
21. B. Khabbazan and S. Mirzakuchaki, "Design and implementation of a low-power embedded CNN accelerator on a low-end FPGA," in Proc. 22nd Euromicro Conf. Digital System Design (DSD), Kallithea, Greece, 2019, pp. 647–650, doi: 10.1109/DSD.2019.00102.
22. G. Li, J. Zhang, M. Zhang, and H. Corporaal, "An efficient FPGA implementation for real-time and low-power UAV object detection," in Proc. IEEE Int. Symp. Circuits and Systems (ISCAS), Austin, TX, USA, 2022, pp. 1387–1391, doi: 10.1109/ISCAS48785.2022.9937449.
23. L. Guo, P. Lin, L. Guo, and B. Liu, "Implementation of a CRNN-based low-power keyword recognition system on FPGA," in Proc. IEEE 14th Int. Conf. ASIC (ASICON), Kunming, China, 2021, pp. 1–4, doi: 10.1109/ASICON52560.2021.9620311.
24. B. Rashidi, B. Rashidi, and M. Pourormazd, "Design and implementation of low power digital FIR filter based on low power multipliers and adders on Xilinx FPGA," in Proc. 3rd Int. Conf. Electronics Computer Technology (ICECT), Kanyakumari, India, 2011, pp. 18–22, doi: 10.1109/ICECTECH.2011.5941647.

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802079

25. K. Khalil, A. Kumar, and M. Bayoumi, "Low-power convolutional neural network accelerator on FPGA," in Proc. IEEE 5th Int. Conf. Artificial Intelligence Circuits and Systems (AICAS), Hangzhou, China, 2023, pp. 1–5, doi: 10.1109/AICAS57966.2023.10168646.
26. R. Sampson et al., "FPGA implementation of low-power 3D ultrasound beamformer," in Proc. IEEE Int. Ultrasonics Symp. (IUS), Taipei, Taiwan, 2015, pp. 1–4, doi: 10.1109/ULTSYM.2015.0514.
27. A. Meena, S. Shiyamala, S. V. Kumar, M. A. Saleemnawaz, and R. Uthirasamy, "Low power enhanced Trivium implementation using parallel-pipeline technique," in Proc. Int. Conf. Smart Technologies and Systems for Next Generation Computing (ICSTSN), Villupuram, India, 2022, pp. 1–4, doi: 10.1109/ICSTSN53084.2022.9761343.
28. R. Badiei, S. Timarchi, and A. Zakaleh, "Low-power resource-efficient FPGA implementation of modified FitzHugh–Nagumo neuron for spiking neural networks," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 72, no. 11, pp. 1780–1784, Nov. 2025, doi: 10.1109/TCSII.2025.3615935.
29. Sugumar, R. (2025). Designing Resilient and Scalable Cloud-Native Frameworks for Generative AI Content Production. *International Journal of Research Publications in Engineering, Technology and Management (IJRPETM), 8(6), 13268-13279.*
30. Soundappan, S. J. (2020). Big Data Analytics in Healthcare: Applications for Pandemic Forecastin. *International Journal of Advanced Research in Computer Science & Technology (IJARCST), 3(1), 2248-2253.*
31. Aarthi, K., Thirumoorthy, P., Tamizharasu, K., Manoja, R., Kalyanasundaram, P., & Rajasekar, M. (2025, September). Improved Network lifetime using Cluster based Power-Aware Balanced Routing Protocol for Device to Device Communication. In *2025 6th International Conference on Electronics and Sustainable Communication Systems (ICESC)* (pp. 1005-1010). IEEE.
32. Mathew, A. Cybersecurity 5.0: From Firewalls to Fully Autonomous Digital Protection.
33. Rengarajan, A. (2025). Cloud-Based AI-Driven Threat Detection Framework for Smart Grid Cybersecurity. *International Journal of Future Innovative Science and Technology (IJFIST), 8(6), 16065.*
34. Anbazhagan, K. (2025). Next-Generation Enterprise Cloud AI for Healthcare: Secure CNN Pipelines and Privacy Controls. *International Journal of Future Innovative Science and Technology (IJFIST), 8(6), 15980.*
35. Socrates, S., Shanmugapriya, M., Murugeshwari, B., & Angalaeswari, S. (2024). Efficient Design for Implantable Device Constant Current Induction Doubly Fed Generating Incorporating Grid Connectivity. In *Intelligent Solutions for Sustainable Power Grids* (pp. 382-392). IGI Global Scientific Publishing.
36. Sugumar, R. (2026). Performance Optimization Frameworks for Financial Web Platforms with Real-Time Transaction Processing. *International Journal of Engineering & Extended Technologies Research (IJEETR), 8(2), 600-611.*
37. Anbazhagan, K. (2025). AI Driven Zero Trust Security Model for Enterprise Data Protection and Intelligent Infrastructure Management. *International Journal of Technology, Management and Humanities, 11(03), 101-107.*
38. Prabha, P. S., & Rengarajan, A. (2025). ENHANCING CLOUD RESOURCE ALLOCATION WITH VISION TRANSFORMER, DEEP REINFORCEMENT LEARNING, AND IMPROVED SHRIKE OPTIMIZATION ALGORITHM. *Corrosion Management ISSN: 1355-5243, 35(2), 233-245.*
39. Vimal, V. R., & Banerjee, J. S. (2025). Integrating PSO, GA, and ACO for Optimized ECG Feature Selection and Classification of Cardiac Disorders. *SGS-Engineering & Sciences, 1(5).*
40. Gopinathan, V. R. (2023). Cloud-First AI Security Architecture for Protecting Enterprise Digital Ecosystems and Financial Networks. *International Journal of Research and Applied Innovations, 6(6), 10031-10039.*
41. Mathew, A. A Secure, Trustworthy, and Regulated Framework for AI Agents in Distributed Networks.
42. Anbazhagan, K. (2025). Secure AI Enabled Enterprise Ecosystems for Fraud Prevention Compliance Automation and Real Time Analytics. *International Journal of Multidisciplinary Research in Science, Engineering, Technology & Management, 1(4), 6-13.*
43. Soundappan, S. J. (2026). Building Trustworthy AI: Explainability and Security in Modern Cloud-Native Data-Driven Ecosystem Platforms. *International Journal of Engineering & Extended Technologies Research (IJEETR), 8(2), 570-579.*
44. Sugumar, R. (2025). Cyber-Secure Cloud Architecture Integrating Network and API Controls for Risk-Aware SAP Healthcare Data Platforms. *International Journal of Humanities and Information Technology, 7(4), 53-60.*
45. Vimal, V. R., & Banerjee, J. S. (2025). Integrating PSO, GA, and ACO for Optimized ECG Feature Selection and Classification of Cardiac Disorders. *SGS-Engineering & Sciences, 1(5).*