Human-AI Collaborative Design of ReActNet-XGBoost Hardware Accelerator for Personalized Wearable ECG Monitoring

Ching-Han Chen
Department of Computer Science and Information Engineering
National Central University
Taoyuan City, Taiwan
pierre@g.ncu.edu.tw

Ke-Shan Lin
Department of Computer Science and Information Engineering
National Central University
Taoyuan City, Taiwan
kslin.miat@gmail.com

Ju-Shan Hsiao
Department of Computer Science and Information Engineering
National Central University
Taoyuan City, Taiwan
jushan.miat@gmail.com

***Abstract*—The rapid expansion of personalized healthcare has positioned wearable electrocardiogram (ECG) monitoring systems as a pivotal solution for continuous cardiac assessment. However, achieving real-time signal classification under strict power and resource constraints remains a core challenge for edge-based deployment. While convolutional neural networks (CNNs) offer excellent feature extraction capabilities, their intensive computational and memory demands impede efficient implementation on lightweight hardware platforms. To overcome these limitations, we propose a hybrid hardware accelerator integrating a binarized ReActNet backbone for low-overhead feature extraction with an XGBoost decision tree classifier that provides interpretable and hardware-efficient inference. The ReActNet module leverages binary convolution operations to reduce arithmetic complexity, while the XGBoost component replaces dense layers, enhancing model transparency and reducing logic resource usage. To further streamline deployment, we introduce a generative hardware synthesis methodology based on large language models (LLMs), enabling automatic translation of trained XGBoost models into synthesizable VHDL code from their JSON representation. This LLM-assisted workflow significantly reduces development time, eliminates manual RTL coding, and promotes modular, reusable classifier generation adaptable to evolving clinical models. The proposed system is evaluated through RTL-level simulation and synthesis targeting low-power, resource-constrained FPGA platforms, demonstrating its architectural feasibility for real-time ECG classification at the edge. These findings highlight a unified and scalable framework that fuses binary neural networks, interpretable classifiers, and generative AI techniques to accelerate hardware–software co-design and broaden the deployment of edge AI in wearable health applications.**

Keywords—wearable ECG monitoring, edge computing, hardware accelerator, binary neural networks, tree-based classifiers, large language model (LLM), generative HDL synthesis

I. INTRODUCTION

The integration of artificial intelligence (AI) into wearable health monitoring devices has revolutionized personalized healthcare, enabling continuous and real-time physiological data analysis [1]. Among various physiological signals, electrocardiogram (ECG) monitoring plays a pivotal role in the early detection of cardiac anomalies and the ongoing assessment of cardiovascular health [2], [3]. Continuous ECG monitoring facilitates proactive cardiac care, allowing early intervention before the onset of critical events such as arrhythmias, ischemia, or myocardial infarction.

Driven by the aging global population and growing awareness of preventive healthcare, the demand for wearable ECG monitoring systems has surged [1], [2]. Recent market analyses predict that the wearable medical device sector will surpass tens of billions of U.S. dollars within the next few years, with cardiac monitoring comprising a significant share. This trend has accelerated the transition from bulky Holter monitors to lightweight, wireless, and long-term ambulatory monitoring devices. Modern wearable ECG devices integrate flexible sensors and wireless communication modules, offering improved patient comfort, extended monitoring durations, and continuous real-world data collection to enhance diagnostic accuracy and therapeutic outcomes.

As wearable devices proliferate, the need for real-time and efficient biosignal processing has become a crucial technical challenge. Reliance on cloud-based infrastructures introduces bottlenecks such as high latency, increased energy consumption, data privacy risks [4], and dependency on network availability [5]. To address these issues, edge computing has emerged as an indispensable paradigm, enabling localized signal processing and decision-making directly on resource-constrained devices. Studies have demonstrated that real-time ECG anomaly detection at the edge significantly enhances system responsiveness and preserves sensitive patient data.

2375-0197/25/$31.00 ©2025 IEEE
DOI 10.1109/ICTAI66417.2025.00215

Nevertheless, deploying sophisticated AI models on edge platforms remains challenging. Traditional convolutional neural networks (CNNs) [6], despite their excellent feature extraction capabilities, impose heavy computational and memory demands, making them unsuitable for low-power embedded systems. To address these limitations, a series of lightweight CNN architectures have been developed. MobileNet [7] pioneered the use of depthwise separable convolutions to reduce computational costs while maintaining accuracy. ShuffleNet [8] further optimized resource efficiency through channel shuffling and grouped convolutions. ReActNet [9] introduced binary convolution operations, replacing costly multipliers with XNOR and popcount computations, significantly lowering arithmetic complexity and memory access overhead—making it a strong candidate for FPGA or ASIC-based acceleration in wearable devices.

Despite these advances, many lightweight CNNs still rely heavily on fully connected (FC) layers, which dominate memory consumption and reduce model interpretability. In healthcare applications, interpretability is critical for ensuring clinical trust, regulatory compliance, and user acceptance. Gradient-boosted decision trees (GBDTs) [10], particularly XGBoost [11], [12], have emerged as promising alternatives, offering high classification accuracy and natural interpretability by enabling transparent tracing of decision paths and feature contributions.

Thus, hybrid designs combining lightweight feature extraction with interpretable classification are seen as an ideal direction for wearable ECG systems. However, implementing such hybrid AI models on FPGA platforms traditionally involves labor-intensive workflows. Engineers must manually translate trained models into register-transfer level (RTL) representations through architecture parsing, functional decomposition, and hardware description language (HDL) coding—processes that are time-consuming, error-prone, and inhibit rapid prototyping and adaptation.

In response to these challenges, generative AI—particularly large language models (LLMs)—has shown transformative potential. Trained LLMs can interpret structured model descriptions, such as JSON-formatted decision trees, and automatically generate syntactically and functionally correct Verilog or VHDL hardware code [13]. Benchmarking frameworks such as ResBench have demonstrated that while different LLMs vary in hardware resource optimization [14], their functional correctness and scalability are promising. Building upon these foundations, combining lightweight, hardware-friendly models like ReActNet and XGBoost with generative synthesis techniques offers a new pathway toward fast, automated, and scalable hardware accelerator design.

Motivated by these insights, this work proposes a hybrid hardware accelerator for real-time ECG classification, integrating a binarized ReActNet feature extractor [9] with an XGBoost classifier [12]. Additionally, an LLM-assisted hardware generation methodology is introduced to automatically translate trained XGBoost models into synthesizable VHDL code. The proposed system is validated through RTL-level simulation, confirming signal correctness and module functionality, and demonstrating its deployment potential on resource-constrained FPGA platforms in wearable ECG applications. The main contributions of this work are summarized as follows:

1. We propose a hybrid hardware accelerator integrating ReActNet and XGBoost, optimized for one-dimensional ECG signal classification on resource-constrained edge devices.

2. We introduce a generative hardware design methodology using a large language model to automatically convert trained XGBoost structures into synthesizable VHDL code.

3. We implement the proposed architecture targeting FPGA platforms and validate its functionality and resource efficiency through RTL simulation and synthesis analysis, using the MIT-BIH Arrhythmia dataset for model training and verification.

4. We demonstrate the practical feasibility of applying generative AI in hardware design, paving the way for automated, interpretable, and efficient AI acceleration in wearable healthcare applications.

The remainder of this paper is organized as follows. Section II reviews related work on binary neural networks (BNNs), tree-based classifiers such as XGBoost, and LLM-driven HDL generation. Section III describes the proposed hardware architecture and the generative synthesis flow. Section IV presents the experimental setup and evaluation results based on simulation. Finally, Section V concludes the paper and outlines directions for future work.

II. Related Work

The growing demand for real-time physiological signal monitoring has accelerated the development of efficient, interpretable, and hardware-friendly machine learning models for wearable edge devices. In particular, electrocardiogram (ECG) monitoring presents a unique set of requirements, including low-latency inference, energy-efficient operation, and clinical interpretability. Recent research has explored several key directions toward addressing these challenges, including binary neural networks, tree-based classification models, and generative methodologies for hardware synthesis. This section reviews these technological advances and highlights their relevance to the proposed hybrid system.

A. Binary Neural Networks for Efficient Edge Inference

The deployment of conventional convolutional neural networks (CNNs) on edge platforms is hindered by their high computational and memory complexity. In response, binary neural networks (BNNs) have emerged as a promising alternative, replacing real-valued weights and activations with binary representations to minimize resource requirements. Early works such as BinaryConnect [15] and BinaryNet demonstrated that binarization enables inference using simple XNOR and popcount operations rather than multipliers, drastically reducing model size and computational energy.

Building on these foundations, XNOR-Net [16] introduced additional refinements through the incorporation of scaling factors, significantly improving inference accuracy while maintaining bitwise efficiency. However, initial BNNs often suffered substantial accuracy degradation, limiting their adoption in precision-sensitive applications such as healthcare. To address these shortcomings, ReActNet proposed a carefully

re-engineered binary network architecture [9], utilizing full convolutions and novel activation functions like RSign and RPReLU to enhance gradient flow and model expressiveness. These improvements enabled BNNs to approach full-precision accuracy in specific application scenarios while retaining ultra-low computational complexity, making them highly attractive for edge-based ECG analysis.

Despite their theoretical advantages, BNNs remain relatively underexplored in FPGA-based deployments for real-time physiological signal processing [17]. Existing applications have largely focused on image classification benchmarks such as CIFAR and ImageNet, which differ significantly from ECG signals in terms of temporal dynamics and noise resilience. As a result, further investigation is needed to evaluate the suitability of BNNs for biosignal-centric inference tasks.

B. *Tree-Based Classifiers for Transparent and Reliable Inference*

Interpretability is a critical requirement in medical applications, where model outputs must be understandable to clinicians and auditable for regulatory compliance. Gradient-boosted decision trees (GBDTs), and in particular XGBoost [12], have gained prominence for their ability to provide high classification accuracy with explicit, traceable decision paths. XGBoost has been successfully applied in various biomedical classification tasks, including arrhythmia detection, heart disease prediction, and ECG signal analysis.

Unlike deep neural networks, tree-based models naturally support deterministic execution and low-latency inference, which is crucial for edge deployment. Furthermore, their relatively compact memory footprint makes them attractive candidates for FPGA implementation. Recent efforts to accelerate GBDTs on hardware platforms have explored pipelined architectures, pruning strategies to reduce tree depth, and lookup-table-based evaluation methods to minimize decision latency [18]–[20]. However, most existing hardware implementations rely on handcrafted RTL design, limiting scalability and adaptability when retraining or reconfiguring models for personalized healthcare applications.

The synergy between efficient binary feature extraction and interpretable tree-based classification presents an opportunity to design hybrid architectures that leverage the strengths of both paradigms for wearable ECG monitoring.

C. *Automated HDL Generation via Large Language Models*

Translating machine learning models into hardware description languages (HDLs) suitable for synthesis has traditionally involved significant manual effort. Conventional approaches require developers to parse trained model structures, design datapaths and control logic, and ensure correctness through extensive verification cycles. This manual workflow presents a major bottleneck in rapidly prototyping and deploying AI accelerators on resource-constrained edge devices.

Recent advances in generative AI, particularly large language models (LLMs) such as GPT-4 and Code Llama, have introduced new possibilities for automating HDL code generation. By interpreting structured model representations, such as decision trees in JSON format, LLMs can synthesize functionally correct Verilog or VHDL descriptions without human intervention. Studies such as ResBench [14] and VRank [21] have shown that LLM-generated HDL can achieve comparable correctness and synthesis performance to manually authored designs, while significantly improving development speed and portability.

Currently, automated HDL generation has been largely limited to simple digital logic or static decision trees, and has not yet been fully extended to complex AI inference pipelines. Nonetheless, this technology has demonstrated strong potential for integration into dynamic medical applications, particularly in scenarios that demand rapid deployment of personalized models—such as condition-specific arrhythmia detection in wearable ECG systems.

D. *Toward a Unified, Scalable Edge Inference Framework*

Although significant progress has been made individually in lightweight CNN design, interpretable machine learning, and automated HDL synthesis, few studies have explored their combined potential in a cohesive architecture optimized for edge-based healthcare applications. Most existing solutions focus on improving isolated components—either neural networks, decision trees, or HDL generation—without systematically addressing system-level co-design challenges such as resource balancing, dataflow synchronization, and modular reusability.

Recent efforts in real-time FPGA-based ECG monitoring [22] have demonstrated the feasibility of deploying lightweight CNNs such as MobileNet [7] and ShuffleNet [8] for edge biosignal analysis [23]. Nevertheless, these models still inherit the interpretability limitations of deep learning approaches, and their fully connected layers often dominate memory consumption on embedded platforms. Furthermore, traditional manual HDL workflows remain a bottleneck for deploying retrained or updated models in dynamic healthcare environments.

By combining binarized feature extraction (via ReActNet), interpretable classification (via XGBoost), and automated HDL synthesis (via LLMs), this work proposes a novel hybrid framework that directly addresses the computational, interpretability, and design automation requirements of wearable ECG monitoring systems. The unified architecture enables scalable, low-power, and clinically trustworthy AI deployment at the edge, advancing the state of the art in personalized healthcare technology.

III. Proposed System Framework

This section presents the detailed design of the ReActNet-XGBoost hardware accelerator, targeting real-time, low-power, and interpretable electrocardiogram (ECG) classification on resource-constrained edge platforms. The framework systematically integrates lightweight feature extraction, interpretable classification, and automated HDL generation, ensuring modularity, scalability, and verifiability. To address the increasing complexity of embedded AI systems, the architecture is designed through hierarchical functional decomposition using IDEF0 modeling and dynamic behavior specification using GRAFCET [24] diagrams. Furthermore, large language models (LLMs) are leveraged to automate

hardware description generation, significantly reducing development effort and enhancing adaptability.

A. Design Objectives and System Overview

Traditional embedded system design often follows a linear sequence from requirements analysis to deployment. However, for modern edge AI applications, where complexity, optimization, and rapid prototyping are critical, a modular and hierarchical methodology is indispensable. Compared to conventional workflows relying on handcrafted RTL or monolithic deep learning accelerators, the proposed system introduces a significant leap by offering a lightweight, interpretable, and highly automatable solution specifically tailored for biomedical signal processing. The proposed system addresses these demands with the following objectives:

1. Efficiency– Minimize computational load and memory usage through binarized operations and fixed-point arithmetic.

2. Interpretability– Employ decision-tree-based classifiers to provide transparent and clinically understandable inference results.

3. Automation– Utilize LLMs for automatic HDL generation, expediting hardware design and updates.

4. Scalability– Construct a modular architecture capable of extension to multimodal and multidimensional biomedical data.

The overall system workflow is depicted in Fig. 1, showing the signal processing and hardware implementation pipeline from ECG input to inference output.

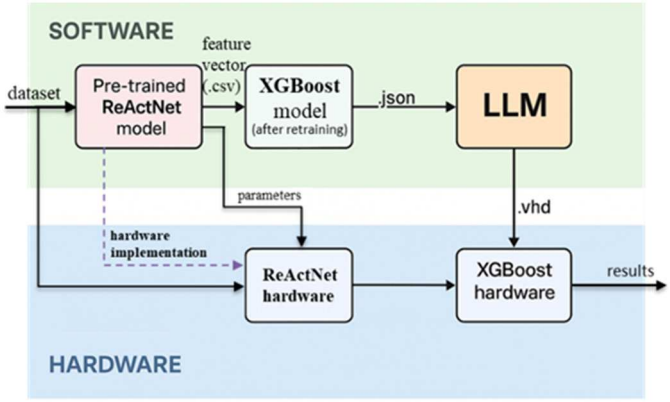


Fig. 1. End-to-end workflow of the proposed ReActNet-XGBoost system integrating LLM-based hardware generation.

B. Functional Decomposition and Modular Design via IDEF0

To systematically manage design complexity and ensure verifiability, the proposed hardware architecture is decomposed using IDEF0 functional modeling. This methodology enables clear specification of each module's input, output, control, and mechanism, facilitating parallel development and modular integration. As illustrated in Fig. 2, the top-level function (A0) is hierarchically partitioned into three major functional modules:

1. A1 Initialization– Handles signal preparation, module resets, and control triggering.

2. A2 ReActNet Feature Extraction– Executes binarized convolutional operations for extracting compact temporal ECG features.

3. A3 XGBoost Classification– Performs interpretable tree-based inference using hardware-mapped decision logic.

Among these, the A2 module plays a critical role in extracting lightweight yet discriminative features suitable for low-power edge deployment. The internal structure of A2 is further decomposed into three primary submodules, as shown in Fig. 3:

1. A21 1×3 Convolution Module– Captures fine-grained local patterns in ECG signals, such as P-waves, QRS complexes, and T-waves.

2. A22 Binary Convolution Layer Module– Replaces costly multipliers with efficient XNOR and popcount operations for binarized computation.

3. A23 Average Pooling Layer– Downsamples the feature map while preserving essential temporal features.

This hierarchical decomposition facilitates modular verification, accelerates prototyping, and supports future extensions such as multi-lead ECG or multimodal biosignal integration.

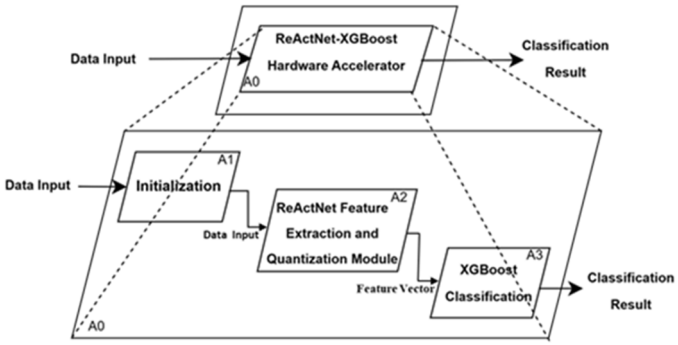


Fig. 2. IDEF0 top-level functional decomposition of the proposed system.

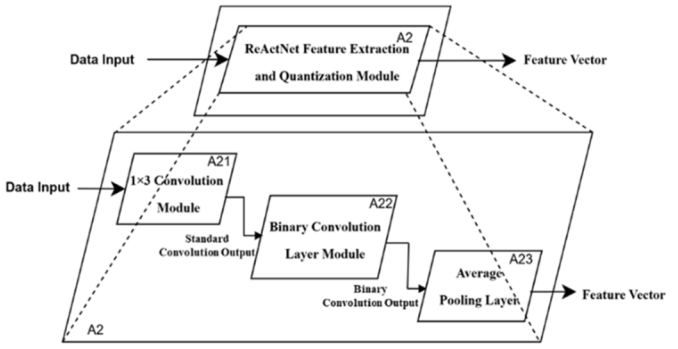


Fig. 3. Expanded IDEF0 submodule view of the A2 feature extraction module.

C. Dynamic Behavior Modeling via GRAFCET

To complement static functional decomposition, the proposed system employs GRAFCET [24] to model the sequential and event-driven behaviors across hardware modules. GRAFCET provides a formal mechanism to represent discrete state transitions and synchronization conditions, making it

particularly suitable for coordinating modular operations in edge AI hardware systems. Fig. 4 illustrates the GRAFCET model governing the runtime control logic of the proposed ReActNet-XGBoost accelerator. The control sequence begins in an idle state and proceeds through initialization, feature extraction, and classification modules, before returning to the idle state upon completion. The defined state transitions are as follows:

1. State 0 (Idle) – Waits for an external start signal to initiate processing.

2. State 1 (Initialization)– Resets control registers and prepares input signals.

3. State 2 (ReActNet Feature Extraction)– Executes 1D convolution, binarization, and pooling operations.

4. State 3 (XGBoost Classification)– Performs hardware-based decision tree inference using the extracted features.

5. State 4 (Finish)– Completes the classification task and transitions back to the idle state.

This finite-state control logic ensures that modules are activated in the correct sequence, prevents resource conflicts, and enables straightforward simulation-based verification. By explicitly modeling the system's dynamic behavior, GRAFCET supports modular testing and lays the foundation for scalable control extensions.

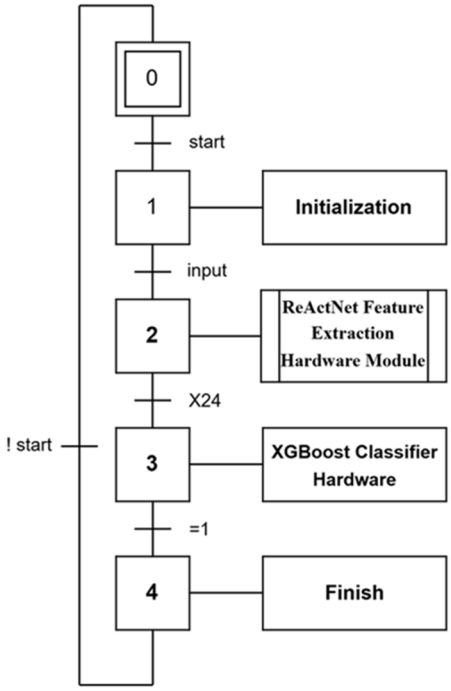


Fig. 4. GRAFCET control model for the proposed ReActNet-XGBoost accelerator.

D. Lightweight Feature Extraction: ReActNet Adaptation

Building upon the modular functional decomposition introduced in Section III-B, the internal architecture of the ReActNet feature extraction module is specifically optimized for one-dimensional ECG signal processing.

Key adaptations include the use of 1×3 convolution kernels to capture localized temporal patterns, the replacement of multipliers with XNOR and popcount operations [9] for binarized computation, and the removal of fully connected layers to minimize memory footprint and hardware complexity. RPReLU activation functions are employed to maintain sufficient model expressiveness despite the use of binary weights and activations.

This streamlined module design significantly reduces computational overhead while preserving classification performance. Although the module's HDL description is manually developed in this work, its highly modular structure supports future extensions through LLM-assisted hardware generation workflows.

E. Interpretable Classification: XGBoost Hardware Mapping

The classification stage employs XGBoost to achieve high accuracy and model transparency for ECG signal analysis. The trained model is serialized into a structured JSON format and retrained to align with hardware constraints, including fixed-point Q2.6 precision and tree depth limitations.

Parallel tree evaluation and majority voting aggregation are implemented to minimize inference latency. The classifier logic is automatically translated into synthesizable VHDL through a retraining and LLM-based generation process, allowing efficient hardware deployment and rapid model updates.

F. Automated HDL Generation Using LLMs

Manual development of synthesizable hardware description language (HDL) for tree-based classifiers is a labor-intensive and error-prone process. To address this limitation, this work adopts an automated HDL generation pipeline driven by large language models (LLMs) [13]. The workflow is structured into four key stages:

1. Parsing– Structural metadata, including node thresholds, feature splits, and leaf outputs, is extracted from the JSON-serialized XGBoost model.

2. Prompt Engineering– The extracted structural information is transformed into structured prompts designed to guide the LLM for accurate code generation.

3. Code Synthesis– The LLM generates synthesizable VHDL modules, implementing tree traversal, threshold comparisons, and leaf value outputs.

4. Integration– Individual decision tree modules are assembled into a complete ensemble classifier, incorporating majority voting logic.

This automated approach significantly reduces hardware design cycles, minimizes manual coding errors, and supports rapid retraining and redeployment. While minor syntactic corrections, such as port name adjustments or missing state handling, are sometimes required, prompt refinement techniques and hierarchical prompt strategies continue to improve code generation fidelity.

The LLM-driven HDL generation enables fast model iteration and hardware acceleration for personalized healthcare applications, where device-specific model updates are frequently necessary.

Fig. 5 illustrates the end-to-end pipeline for generating and deploying an LLM-generated classifier. The upper portion shows the software flow, where a JSON-formatted XGBoost model is parsed and translated into VHDL using an LLM. The lower portion presents the hardware logic block diagram of the synthesized classifier, including a fixed-point feature register, decision tree evaluator, and majority voting accumulator.

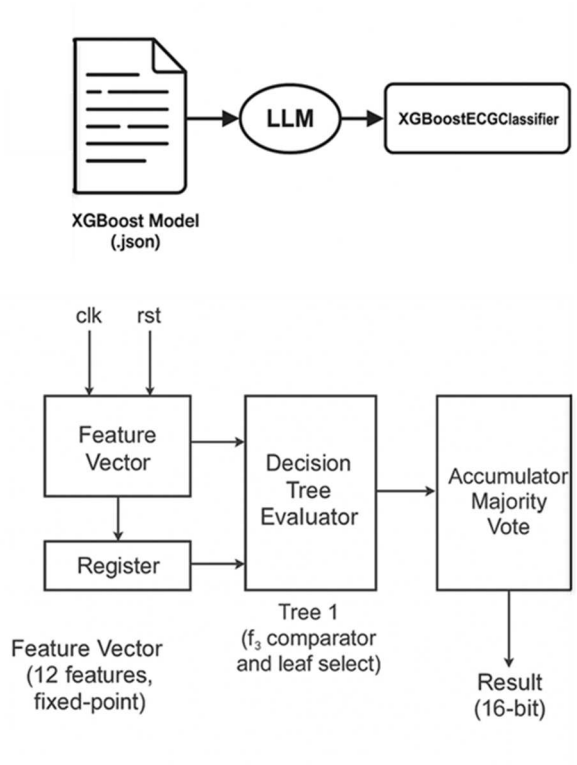


Fig. 5. HDL Generation and Deployment Pipeline for LLM-Based XGBoost Classifier

G. Summary

This section developed a hardware-accelerated ECG classification system by integrating lightweight binary feature extraction, interpretable decision tree classification, and LLM-assisted HDL generation. Through IDEF0-based modular decomposition and GRAFCET-based dynamic control modeling, the proposed design achieves systematic scalability, synchronized module operation, and efficient verification.

The system directly addresses the critical challenges in edge AI deployment—efficiency, interpretability, and automation—offering a low-latency, low-power, and transparent solution tailored for wearable biomedical applications. The combination of binary convolution, symbolic inference, and generative hardware synthesis demonstrates a promising direction for resource-constrained AIoT healthcare systems.

While the current work focuses on binary classification using single-lead ECG signals, the modular and generative architecture paves the way for future extensions, such as multi-lead ECG processing and multi-signal biosensor fusion, which are outlined in the conclusion and future work.

IV. Experimentation And Discussion

A. Experimental Procedure and Flow

The evaluation consisted of two major stages: (1) software-side model training and conversion, and (2) hardware-side inference simulation. The ReActNet model was used to extract 1D temporal features from ECG signals, which were then used to train an XGBoost classifier. Quantization-aware retraining was applied to align with the Q2.6 format. ReActNet modules were manually implemented in synthesizable VHDL, while the XGBoost classifier logic was automatically generated via LLMs using model metadata in JSON format.

Hardware modules were connected through fixed-point data paths and control signals. Each data handoff—from feature extraction to classification—was verified to ensure logic correctness and signal integrity. This tightly integrated pipeline enables low-latency operation without off-chip memory dependencies.

B. Dataset and Label Reconfiguration

The MIT-BIH Arrhythmia Dataset [25], retrieved from Kaggle, was used as the benchmark for system evaluation. All ECG data are publicly available and de-identified. No new human subjects were involved, and no additional informed consent was required. The original dataset comprises five heartbeat categories, with the sample distribution summarized in Table I.

TABLE I. Class Distribution in MIT-BIH Dataset

Class (Label)	Description	Samples (%)
0: N	Normal beat	90,587 (82.8%)
1: S	Supraventricular premature beat	2,779 (2.5%)
2: V	Premature ventricular contraction	7,236 (6.6%)
3: F	Fusion of ventricular and normal beat	803 (0.7%)
4: Q	Unclassifiable beat	8,039 (7.4%)

Due to the highly imbalanced distribution, a binary classification strategy was adopted: normal heartbeats (class 0) were retained as class 0, while all abnormal classes (1–4) were merged into class 1, as shown in Table II.

TABLE II. Binary Classification Mapping

Condition	Original Classes
Normal (0)	0: N
Abnormal (1)	1: S, 2: V, 3: F, 4: Q

To further illustrate the classification, Fig. 6 shows a one-second ECG segment sampled at 360 Hz from the MIT-BIH Arrhythmia Database [26], with the processed signal's rolling standard deviation and novelty detection results using XGBoost. The x-axis is in units of 1/360 seconds, while the y-axis in each section represents signal amplitude (mV), rolling STD, and Euclidean distance, respectively. The blue shaded areas

represent segments where no anomalies were detected, while the red lines indicate predicted abnormal heartbeats.

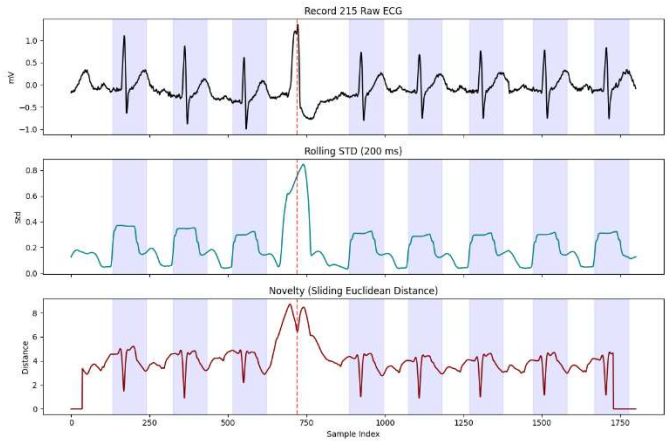


Fig. 6. ECG Signal Analysis with XGBoost-Based Rolling STD and Novelty Detection.

C. Hardware Implementation and Quantization Strategy

All hardware computations adopted a Q2.6 fixed-point format (two integer bits and six fractional bits), achieving a balanced trade-off between numerical precision and hardware resource efficiency [10]. Floating-point operations were entirely eliminated, thereby reducing logic utilization, power consumption, and critical path delay—making the system highly suitable for low-power edge applications.

Signal handoffs between modules were fully synchronized via clocked control signals. Waveform simulation was conducted to validate correct activation and data handshaking across all processing stages.

D. Waveform Simulation and Functional Verification

Functional correctness of the system was verified through RTL waveform simulation using ModelSim. As shown in Fig. 7, the simulation output exhibits stable signal transitions throughout all modules without any undefined values (X states) or control hazards. The feature extractor and classifier modules responded correctly to control triggers, with all handshaking signals and data paths operating in a synchronized manner.

To confirm classification effectiveness, a zoomed-in view of the final operation stage is provided in Fig. 8. The simulation recorded a total of 63,121 input samples and correctly classified 60,575, yielding a macro-average classification accuracy of 95.97%. The hit and hit_count signals demonstrate successful output confirmation and match the expected classification outcome, providing strong evidence of system-level functional reliability.

E. System Analysis and Discussion

The proposed ReActNet-XGBoost hardware accelerator achieves a favorable trade-off among model efficiency, interpretability, and hardware deployment feasibility for wearable ECG anomaly detection. By integrating a binary convolutional neural network with a tree-based classifier, the system mitigates the computational complexity and opacity inherent in conventional deep learning approaches. The adoption of a Q2.6 fixed-point quantization scheme significantly reduces resource utilization and dynamic power consumption, while preserving classification accuracy—thereby enhancing suitability for low-power FPGA-based implementations.

Furthermore, the incorporation of LLM-assisted HDL generation expedites hardware development by automating the conversion of trained models into synthesizable descriptions, facilitating rapid updates in scenarios requiring frequent personalization. The modular architecture further supports extensibility to additional physiological signals such as photoplethysmography (PPG) and blood pressure. These attributes collectively validate the accelerator's applicability to next-generation edge AI wearable health monitoring platforms.

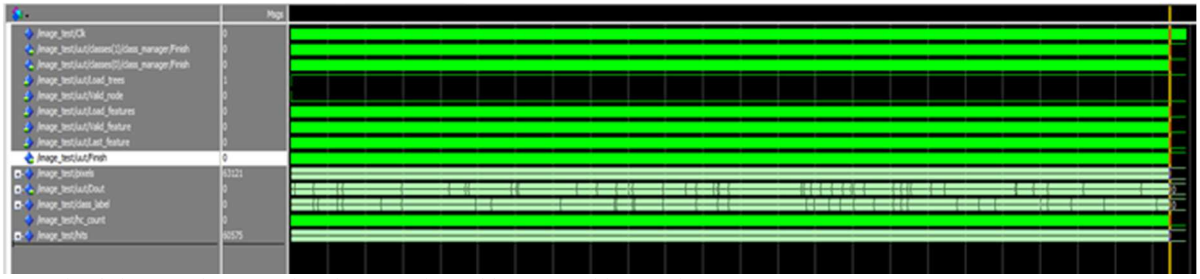


Fig. 7. RTL simulation waveform overview demonstrating stable system-wide operation.

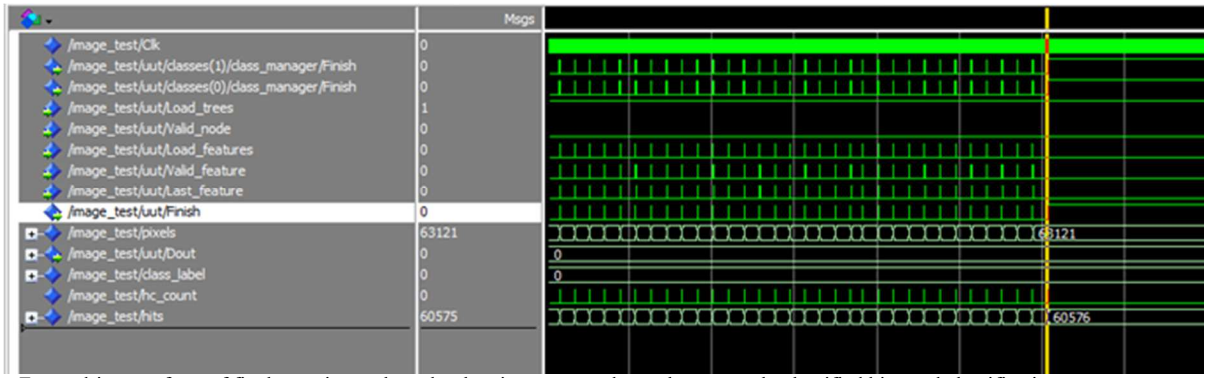


Fig. 8. Zoomed-in waveform of final experimental results showing processed samples, correctly classified hits, and classification accuracy.

V. Conclusion And Future Work

This work presents a hybrid hardware accelerator that integrates a binarized ReActNet feature extractor, an XGBoost-based decision tree classifier, and an LLM-assisted HDL generation pipeline for real-time electrocardiogram (ECG) anomaly detection on edge devices. The system achieved a macro-average classification accuracy of approximately 95.97%, while keeping minimal resource use and ultra-low power, making it highly suitable for wearable healthcare applications.

The core contributions include the development of a lightweight, interpretable ECG hardware accelerator optimized for edge deployment, and the pioneering integration of generative artificial intelligence into the hardware design flow. Specifically, the use of LLMs for automatic VHDL code generation enhances hardware design flexibility and automation, pushing the boundaries of embedded system development.

Future work will extend the generative synthesis framework to support more classifiers and multimodal biosignal processing, enabling new applications of generative AI in next-generation wearable healthcare systems.

Acknowledgment

The authors would like to thank the National Science and Technology Council, Taiwan, R.O.C., for their financial support (Grant Nos. NSTC 112-2221-E-008 -029 -MY3).

References

[1] A. Pantelopoulos and N. G. Bourbakis, "A survey on wearable sensor-based systems for health monitoring and prognosis," IEEE Trans. Syst., Man, Cybern. C, Appl. Rev., vol. 40, no. 1, pp. 1–12, Jan. 2010, doi: 10.1109/TSMCC.2009.2032660.

[2] M. M. Baig, H. GholamHosseini, and M. J. Connolly, "A comprehensive survey of wearable and wireless ECG monitoring systems for older adults," Med. Biol. Eng. Comput., vol. 51, no. 5, pp. 485–495, May 2013, doi: 10.1007/s11517-012-1021-6.

[3] S. Majumder, L. Chen, O. Marinov, C.-H. Chen, T. Mondal, and M. J. Deen, "Noncontact wearable wireless ECG systems for long-term monitoring," IEEE Rev. Biomed. Eng., vol. 11, pp. 306–321, May 2018, doi: 10.1109/RBME.2018.2840336.

[4] M. Hizem, L. Bousbia, Y. Ben Dhiab, M. Ould-Elhassen Aoueileyine, and R. Bouallegue, "Reliable ECG anomaly detection on edge devices for Internet of Medical Things applications," Sensors, vol. 25, no. 8, p. 2496, Apr. 2025, doi: 10.3390/s25082496.

[5] M. Chen, Y. Ma, Y. Li, D. Wu, Y. Zhang, and C.-H. Youn, "Wearable 2.0: Enabling human-cloud integration in next generation healthcare systems," IEEE Commun. Mag., vol. 55, no. 1, pp. 54–61, Jan. 2017, doi: 10.1109/MCOM.2017.1600410CM.

[6] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, May 2015, doi: 10.1038/nature14539.

[7] A. G. Howard et al., "MobileNets: Efficient convolutional neural networks for mobile vision applications," arXiv preprint arXiv:1704.04861, Apr. 2017. [Online]. Available: https://arxiv.org/abs/1704.04861.

[8] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Salt Lake City, UT, USA, Jun. 2018, pp. 6848–6856, doi: 10.1109/CVPR.2018.00716.

[9] Z. Liu, Z. Shen, M. Savvides, and K.-T. Cheng, "ReActNet: Towards precise binary neural network with generalized activation functions," in Proc. Eur. Conf. Comput. Vis. (ECCV), Lecture Notes in Computer Science, vol. 12359. Cham, Switzerland: Springer, 2020, pp. 143–159, doi: 10.1007/978-3-030-58568-6_9.

[10] T. Tanaka, R. Kasahara, and D. Kobayashi, "Efficient logic architecture in training gradient boosting decision tree for high-performance and edge computing," arXiv preprint arXiv:1812.08295, Dec. 2018. [Online]. Available: https://doi.org/10.48550/arXiv.1812.08295.

[11] F. Doshi-Velez and B. Kim, "Towards a rigorous science of interpretable machine learning," arXiv preprint arXiv:1702.08608, Feb. 2017. [Online]. Available: https://doi.org/10.48550/arXiv.1702.08608.

[12] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. (KDD), San Francisco, CA, USA, Aug. 2016, pp. 785–794, doi: 10.1145/2939672.2939785.

[13] W. Sun, B. Li, G. L. Zhang, X. Yin, C. Zhuo, and U. Schlichtmann, "Paradigm-based automatic HDL code generation using LLMs," arXiv preprint arXiv:2501.12702, Jan. 2025. [Online]. Available: https://doi.org/10.48550/arXiv.2501.12702.

[14] C. Guo and T. Zhao, "ResBench: Benchmarking LLM-generated FPGA designs with resource awareness," arXiv preprint arXiv:2503.08823, Mar. 2025. [Online]. Available: https://doi.org/10.48550/arXiv.2503.08823.

[15] M. Courbariaux, Y. Bengio, and J.-P. David, "BinaryConnect: Training deep neural networks with binary weights during propagations," in Proc. 29th Int. Conf. Neural Inf. Process. Syst. (NIPS), Montreal, QC, Canada, Dec. 2015, vol. 2, pp. 3123–3131.

[16] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "XNOR-Net: ImageNet classification using binary convolutional neural networks," in Proc. Eur. Conf. Comput. Vis. (ECCV), Lecture Notes in Computer Science, vol. 9908. Cham, Switzerland: Springer, 2016, pp. 525–542, doi: 10.1007/978-3-319-46493-0_32.

[17] Y. Su, K. P. Seng, L. M. Ang, and J. Smith, "Binary neural networks in FPGAs: Architectures, tool flows and hardware comparisons," Sensors, vol. 23, no. 22, p. 9254, Nov. 2023, doi: 10.3390/s23229254.

[18] A. Alcolea and J. Resano, "FPGA accelerator for gradient boosting decision trees," Electronics, vol. 10, no. 3, Art. no. 314, Jan. 2021, doi: 10.3390/electronics10030314.

[19] R. Olmos and A. Otero, "Optimized XGBoost architecture on FPGA for high-performance and seamless deployment," presented at the 7th Workshop on Accelerated Machine Learning (AccML), 2025. [Online]. Available: https://accml.dcs.gla.ac.uk/papers/2025/7th_AccML_paper_5.pdf.

[20] A. Khataei and K. Bazargan, "TreeLUT: An efficient alternative to deep neural networks for inference acceleration using gradient boosted decision trees," in Proc. ACM/SIGDA Int. Symp. Field-Programmable Gate Arrays (FPGA), Monterey, CA, USA, Feb. 2025, pp. 14–24, doi: 10.1145/3706628.3708877.

[21] Z. Zhao, R. Qiu, I.-C. Lin, G. L. Zhang, B. Li, and U. Schlichtmann, "VRank: Enhancing Verilog code generation from large language models via self-consistency," arXiv preprint arXiv:2502.00028, Jan. 2025. [Online]. Available: https://doi.org/10.48550/arXiv.2502.00028.

[22] K. G. Shanthi, A. M. J. Kinol, S. Santhi, and K. Kannan, "Real-time FPGA integration for ECG monitoring: Bidirectional recurrent chimp search model," IETE J. Res., vol. 70, no. 8, pp. 6848–6863, Aug. 2024, doi: 10.1080/03772063.2024.2311744.

[23] R. Wu, B. Liu, P. Fu, and H. Chen, "An efficient lightweight CNN acceleration architecture for edge computing based on FPGA," Appl. Intell., vol. 53, no. 11, pp. 13867–13881, Nov. 2023, doi: 10.1007/s10489-022-04251-3.

[24] P. Baracos, *Grafcet Step-by-Step: Tutorial and Reference Guide to the Grafcet Language*. Montreal, QC, Canada: Famic Automation, 1992.

[25] S. Fazeli, "ECG Heartbeat Categorization Dataset," Kaggle, 2018. [Online]. Available: https://www.kaggle.com/datasets/shayanfazeli/heartbeat [Accessed: Jan. 2025].

[26] G. B. Moody and R. G. Mark, "MIT-BIH Arrhythmia Database," PhysioNet, Feb. 2005. [Online]. Available: https://physionet.org/content/mitdb/1.0.0/ [Accessed: Apr. 2025].