# FPGA-based Real-time ECG Classification System using Quantized Inception-ResNeXt Neural Network and CWT Approximation

Tiancheng Cao, *Member, IEEE*, Wei Soon Ng, *Student Member, IEEE*, Wang Ling Goh, *Senior Member, IEEE*, Yuan Gao, *Member, IEEE*, Hen-Wei Huang, *Member, IEEE*

***Abstract*—This paper presents a software–hardware co-designed Field Programmable Gate Array (FPGA)-based real-time ECG classification system that combines methodological and practical innovations to achieve state-of-the-art performance with an ultra-compact model. On the software side, we introduce a hardware-adaptive, configurable quantization-aware training (QAT) framework that enables layer-wise precision assignment and flexible quantization, ensuring the trained model is highly accurate and hardware-friendly even at ultra-low bit-widths. On the hardware side, we propose a resource-efficient FPGA accelerator featuring a streaming architecture and a cosine-approximated CWT module, optimized for low-power and real-time inference. Implemented in FPGA, We demonstrate that a 6-layer Inception-ResNeXt network can achieve 99.5% inference accuracy on the MIT-BIH ECG dataset with 200mW dynamic power and 0.0767mJ/inference energy efficiency.**

***Index Terms* — *ECG, low-power Edge-AI, FPGA, Hardware-software co-design, streaming architecture, fixed-point quantization.***

## I. Introduction

Deep neural networks (DNNs) have demonstrated remarkable performance in a wide range of classification tasks in the areas of image recognition [1, 2] and natural language processing [3], and recently shown great potentials biomedical signal processing, particularly in electrocardiogram (ECG) analysis for cardiovascular disease (CVD) monitoring [4]. The emergence of wearable ECG devices, which enables continuous monitoring of cardiovascular activities, greatly benefits personalized medicine but also raises concerns of personal information security. The capacity of continuous monitoring and real-time analyzing cardiac data is essential for early arrhythmia detection and timely clinical intervention [5]. However, implementing DNNs on wearable devices is faceing substantial design challenges due to the limitations in memory size and computational capacity as well as power hungry computation [6]. Field Programmable Gate Array (FPGA) is a promising alternative which offers a combination of power efficiency and flexibility for system upgrades due to their reconfigurable nature. On the other hand, network weight quantization has emerged as an effective method to reduce memory usage while maintaining competitive classification accuracy [9,10]. The main challenge lies in balancing model complexity, accuracy, computational demands, and power consumption [11,12]. Among various DNNs, the Inception architecture stands out for its use of variable filter sizes and combined convolutions, enabling efficient feature extraction and making it well-suited for resource-constrained edge devices [13]. Given the trend toward personalized medicine and data security, there is a critical need for an end-to-end, edge-deployed ECG monitoring solution that leverages hardware–software co-design for optimal performance and robustness in limited environments. Recent advances in edge-based ECG classification highlight the necessity of integrating algorithmic innovation with hardware implementation [14–21]. While developments in deep learning and adaptive quantization have improved performance [16,17], real-time processing, energy efficiency, and hardware compatibility remain essential [20,21]. Thus, jointly optimizing neural network design and hardware realization is vital for next-generation wearable and edge medical devices.

Many methods have been explored to implement ECG monitoring algorithm and hardware-software co-design for ECG ventricular ectopic beat classification have been reported [14-28]. For instance, prior studies have introduced various 1D CNN-based approaches for ECG signal classification [22, 23]. However, using 1D ECG signals for model training, which often contain noise such as baseline wandering effects, necessitates extensive preprocessing to filter and extract features, including frequency domain characteristics [24]. Recent research focused on the mapping of ECG signals to

Manuscript received MMM-DD, YYYY, revised MMM-DD, YYYY; accepted MMM-DD, YYYY. This work was supported by the Nanyang Assistant Professorship Start-up Grant and the Eric and Wendy Schmidt AI in Science Postdoctoral Fellowship, a Schmidt Futures program. *(Tiancheng Cao and Wei Soon Ng contributed equally to this work.) (Corresponding Author: Yuan Gao and Hen-Wei Huang)*

T. Cao and W.S. Ng are with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 and they are also with Institute of Microelectronics (IME), A*STAR, Singapore 138634. (e-mail: tiancheng.cao@ntu.edu.sg, weisoon001@e.ntu.edu.sg).

Y. Gao is with the Institute of Microelectronics (IME), A*STAR, Singapore 138634. (e-mail: gaoy@ime.a-star.edu.sg).

W.L. Goh is with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798. (e-mail: ewlgoh@ntu.edu.sg).

Hen-Wei Huang is with the School of Electrical and Electronic Engineering and Lee Kong Chian School of Medicine, Nanyang Technological University, Singapore, 639798 (e-mail: henwei.huang@ntu.edu.sg).

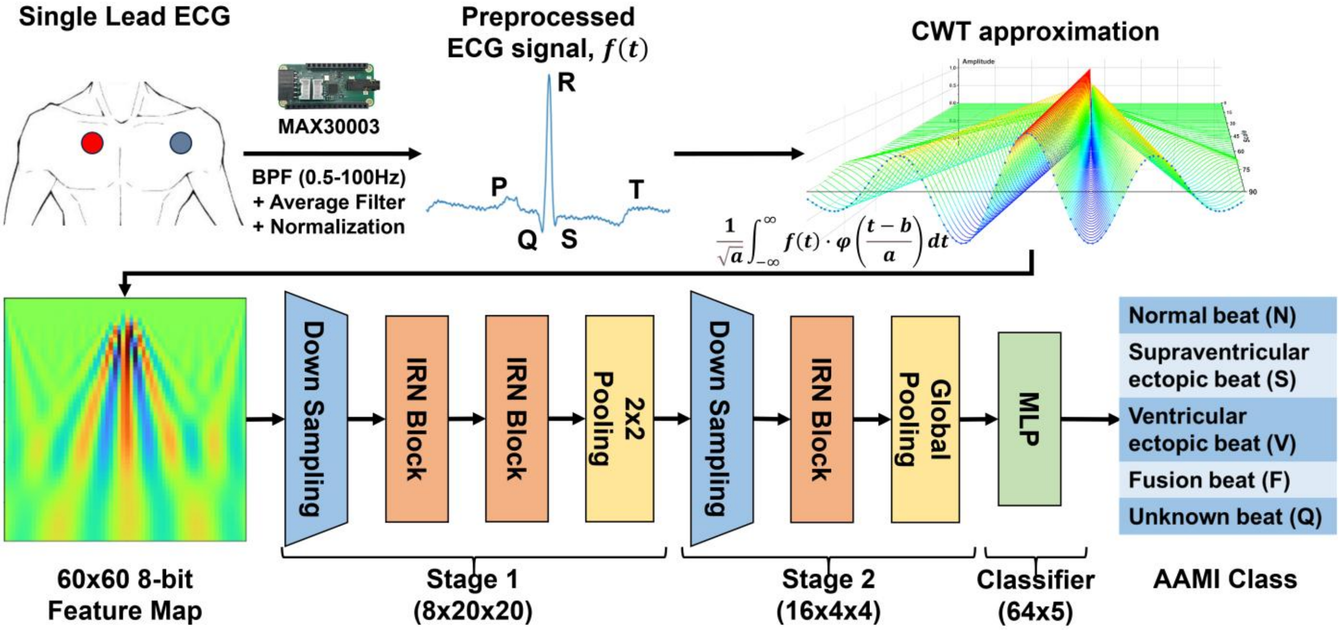


Fig. 1. Block diagram of the proposed real-time ECG signal classification system. Association for the Advancement of Medical Instrumentation (AAMI) recommends grouping ECG heartbeats into the five categories

frequency domain and processing them with DNNs like LSTM and 2D CNNs [25, 26]. Nonetheless, the huge amount of data involved in these approaches pose challenges for real-time implementation on edge devices. More recently, hardware solutions have been proposed and simulated on FPGA and ASIC platforms [27, 28] but achieving applications beyond binary classification remains a challenge. Moreover, the development of circuits for transforming ECG signals into 2D spectrogram representations still require enhancements tailored for edge devices [29, 30]. Furthermore, additional endeavors have been made to provide FPGA implementations [31-33]. However, addressing the critical need for hardware-software co-design on FPGA platforms remains a pressing concern.

This work presents an FPGA-based real-time ECG classification system. The key innovations in this work are summarized as follows. Firstly, we introduce a hardware-driven, configurable quantization-aware training (QAT) pipeline that jointly optimizes network training and quantization for the deployment constraints of FPGA-based edge devices. This adaptive framework enables layer-specific bit-width assignment, dynamic quantization resolution based on kernel statistics, and seamless integration of hardware limitations into the training process, yielding a highly compact model with state-of-the-art accuracy. Secondly, the system-level software–hardware co-design methodology is adopted, wherein the Inception-ResNeXt neural network architecture, cosine-approximated CWT feature extraction, and the FPGA hardware architecture are holistically optimized for real-time edge inference. Thirdly, we demonstrate the effectiveness of this integrated approach through a rigorous FPGA implementation, achieving 99.5% accuracy, low inference energy, and compact model size, thus establishing a new benchmark for low-power, high-performance edge-AI biomedical systems. Implementing a pipelined streaming architecture with layer wise customized dataflow and precision to achieve ultra-low latency inference while maintaining low resource consumption on FPGA. A 6-layer Inception-ResNeXt is designed with 8 bit / 4 bit resolution. MIT-BIH ECG dataset [34] with five distinct classes is used for evaluating our inference accuracy.

The rest of this paper is organized as follows, Section II introduces the overall system architecture and the Inception-ResNeXt network. Section III presents the band pass filter (BPF)-based approximated CWT signal processing block. Section IV presents the FPGA implementation and Section V shows the measurement results. Finally, Section VI concludes the paper.

## II. CVD RECOGNITION SYSTEM WITH INCEPTION-RESNEXT

Fig. 1 shows the block diagram of the proposed real-time ECG classification system. The raw ECG signal acquired from the sensor is first preprocessed in hardware by the MAX30003 AFE chip, which performs band-pass filtering, baseline stabilization, and real-time segmentation. The denoised ECG windows are then sent to the FPGA, where a CWT approximation extracts the feature map. The quantized feature map will be processed by a 2-stage Inception-ResNeXt network and a MLP classifier for classification.

### *A. Data Preprocessing*

MIT-BIH arrhythmia database [34] is used for neural network training. This dataset comprises 48 half-hour segments of ambulatory ECG recordings, captured from 47 individuals. The data sampling frequency is 360 Hz per lead with 11-bit resolution.

Each ECG beat is segmented with a focus on the R-wave peak time, as shown in Fig. 2. Specifically, the classification of

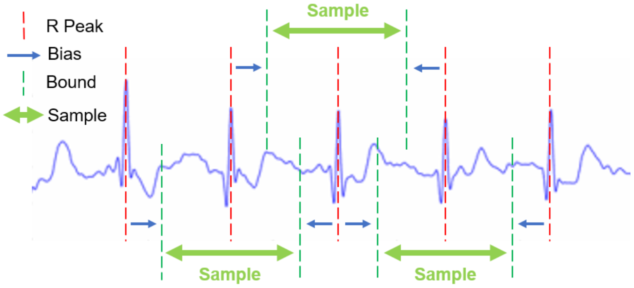


Fig. 2. Signal segmentation based on R-wave peak times. The method used to segment ECG signals focuses on the R-wave peaks to isolate individual heartbeats for precise arrhythmia classification.

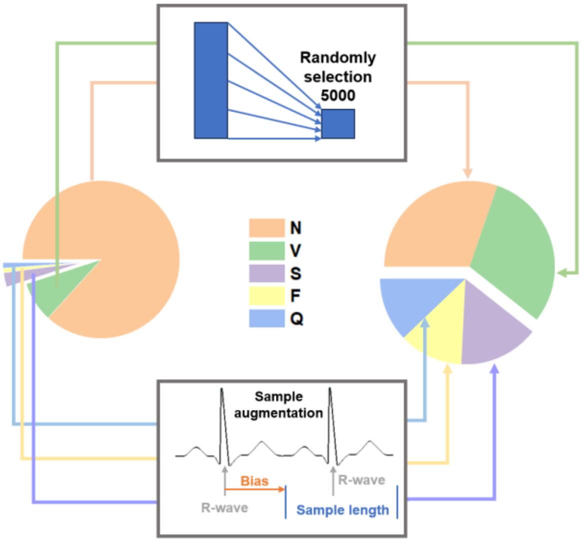
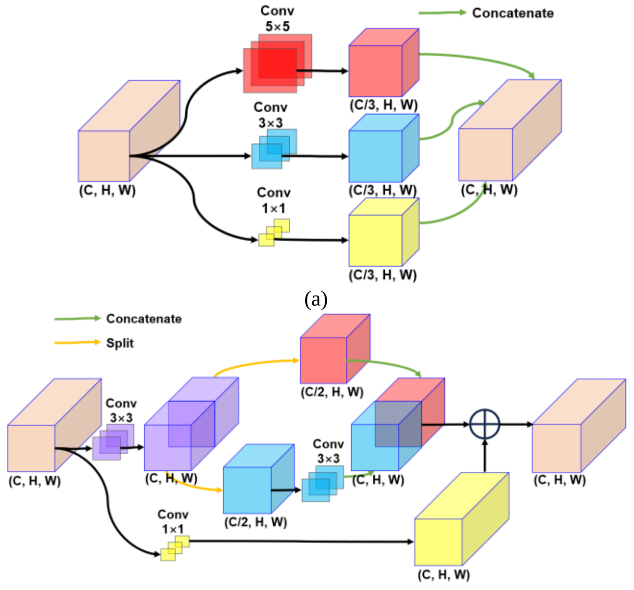


Fig. 3. Data preprocessing workflow for ECG signals.

arrhythmia types is anchored at the R-wave peak of every ECG beat. The R-wave peak is used as the center point to generate segmented ECG signal. The boundaries of each beat signal are defined by the preceding and subsequent R-peaks, with an additional bias applied to refine the segmentation. This method allow us to isolate individual ECG beats precisely yet consistantly. The range of a single beat is determined by:

$$T(Rpeak(k-1)+b) \leq T\big(Rpeak(k)\big) \leq T(Rpeak(k+1)-(120-b)) \tag{1}$$

where $T(Rpeak(k))$ is the R-wave peak time of annotation $k$ and $b$ is the beat range bias. However, it is important to note imbalanced sample distribution in this dataset, recognized by [35]. To mitigate the impact of imbalanced sample distribution, two strategies have been introduced during the preprocessing phase as shown in Fig. 3. In the training process, for common classes like Normal beats (N) and Ventricular ectopic beats (V), a random selection of 5000 samples from each class is made for every epoch. Conversely, for rarer classes such as Supraventricular ectopic Beats (S), Fusion beats (F) and Unclassifiable beats (Q), a dynamic approach is utilized to augment the data with a variable beat range bias, randomly chosen between 40 and 80, according to the equation (1). This strategy facilitates the generation of 2000 samples for each rare class. This method does not increase computational complexity nor requires extensive hyperparameter tuning. It serves as an implicit regularization technique, effectively balancing the dataset without imposing additional computation burdens. In addition, all dataset was performed with systematic random noise injection and shifting augmentation, ensuring the evaluation reflects both diversity and robustness.



(a)

(b)

| | ResNet Block | Inception Block | IRN Block |
|---|---|---|---|
| # Params (k) | 8.4 | 5.8 | 6.3 |
| MACs (M) | 1.05 | 0.66 | 0.73 |
| Acc (%) | 99.37 | 99.28 | 99.36 |

(c)

Fig. 4. (a) The architecture of the Inception network and (b) the proposed Inception-ResNeXt block (IRN block). (c) Ablation study comparing the proposed Inception-ResNeXt (IRN) block with conventional ResNet and Inception blocks.

Subsequently, all samples are standardized to a uniform size of 360 data points. For samples exceeding this length, truncation is applied to reduce the size to 360. Conversely, shorter samples are padded with zeros to extend their length to the standard size. This standardization ensures uniformity in the dataset, facilitating consistent processing and analysis.

### B. Inception-ResNeXt Architecture

Inception excels at capturing multi-scale features via multi-path convolutions, though it can be computationally intensive. Conversely, ResNeXt, featuring grouped convolutions and residual connections, offers efficiency and stable training but may lack detailed multi-scale feature extraction. Combining these architectures leverages their complementary strengths, providing a balanced solution for robust ECG signal processing. Although the CWT already extracts primary time-frequency information, the feature extractor within the Inception-ResNeXt model is by no means redundant. Instead, it is specifically designed to work on the resulting image-like data, utilizing

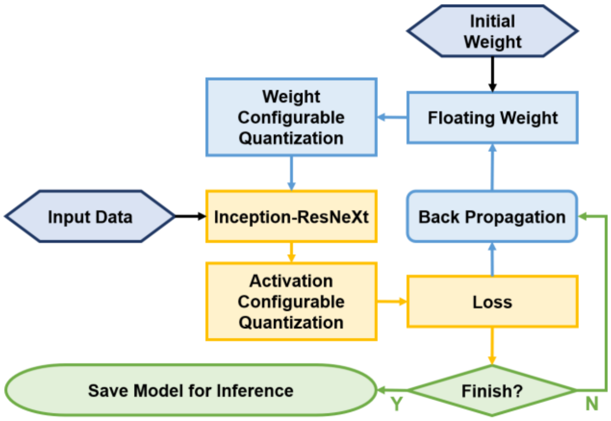


Fig. 5. The flow of the configurable quantization-aware training process.

multi-path convolutions to further refine features and capture spatial patterns at different scales. This complementary approach enhances the discriminative capability of the network, ensuring that the model can effectively learn subtle variations in the ECG signals.

Fig. 4 shows the block digaram of the conventional Inception block, Fig. 4(a), and the proposed Inception-ResNeXt model, Fig. 4(b). The input feature map firstly goes through a 1×1 convolution to reduce channel dimensions and prepare features for subsequent operations. Next, the feature map is processed by a 3×3 convolution to extract spatial information. The resultant feature map is then split into two equal segments along the channel dimension as ResNeXt, each with half number of channels. The first segment of the split channels undergoes another 3×3 convolution to further refine its representation to achieve 5×5 effective receptive field, while the second segment retains its original processed features. These two segments are subsequently recombined through channel-wise concatenation, allowing multi-scale spatial patterns and complementary representations to merge effectively, similar to the parallel operations in Inception approach. Finally, the concatenated output is subjected to an element-wise addition with the original feature map from the 1×1 convolution, forming a residual connection.

Channel splitting reduces the computation load while retaining feature extraction capabilities. Hardware-specific optimizations, such as mapping the 1×1 and 3×3 convolutions to dedicated hardware units, and efficient channel concatenation, further enhance performance by minimizing data movement and latency. Moreover, it has been proven that consecutive 3×3 convolutions is equivalent to a 5×5 convolution [13]. This finding can be used to increase computational efficiency while preserving model capacity. The residual connections not only facilitate better gradient flow during the training but also promote data reuse, reducing redundant computations and boosting energy efficiency. These features make the IRN block a robust solution for real-time image processing and embedded AI tasks, where efficiency and high performance are critical.

To further validate the effectiveness of the proposed IRN block, we conducted an ablation study comparing it with conventional ResNet and Inception blocks under identical training and evaluation protocols. As summarized in Fig. 4(c), the IRN block achieves competitive accuracy with reduced computational complexity and parameter count compared to the ResNet block, and outperforms the Inception block in accuracy with only a slight increase in MACs. This demonstrates the balanced trade-off between efficiency and accuracy brought by our design.

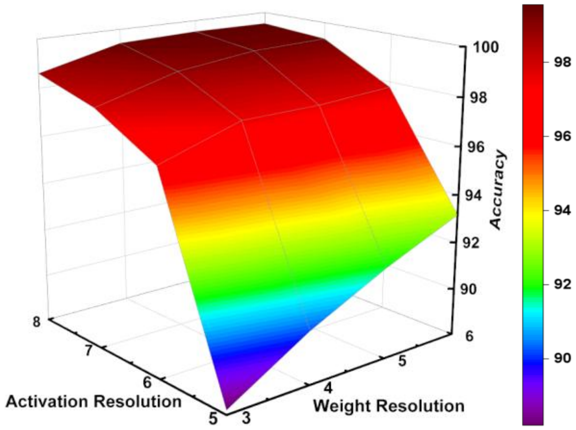


Fig. 6. System accuracy is evaluated across various weight and activation resolutions, demonstrating the trade-offs between quantization levels and computational efficiency.

## C. Configurable Quantization Aware Training

During the training phase of the Inception-ResNeXt, the configurable quantization method is applied for weight quantization and activation quantization, respectively. Quantization aware training (QAT) ensures that the model is optimized for quantized deployment [36], leading to reduced model size and lower computation complexity. The flowchart of the training process is shown in Fig. 5. For weight quantization in this system, the process is intricately intertwined with the backpropagation algorithm. Conversely, for activation quantization, the procedure is integrated into the forward pass of the network. The system accuracy was systematically evaluated across a range of weight and activation resolutions, as illustrated in Fig. 6. The results demonstrated a clear trend, indicating the tradeoffs in selecting different resolutions. Specifically, the model weights are quantized to a signed 4-bit representation with 8-bit activation resolution.

In the proposed quantization process, a 2D convolution kernel of dimensions (C, H, W) is utilized as an example. The initial step involves determining the maximum absolute value within the kernel.

$$k_{ma} = \max\left(|K_{c,h,w}|\right) \quad (2)$$

where, $K_{c,h,w}$ represents the value of the kernel at the $c^{th}$ channel, $h^{th}$ row, and $w^{th}$ column. The function max $(\cdot)$ operates over all the kernel dimensions, and $|\cdot|$ denotes the absolute value. Subsequently, the Most Significant Bit (MSB) resolution in the fixed-point representation is defined as follows:

$$r_{MSB} = r, if \begin{cases} k_{ma} > 2^{r} \\ k_{ma} < 2^{r+1} \end{cases} \quad (3)$$

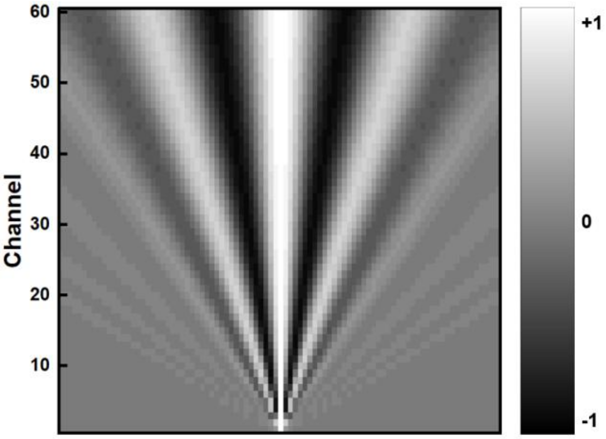


Fig. 7. The matrix of wavelets generated for different scales with the proposed discrete CWT approximation implemented on FPGA.

The n-bit signed fixed-point number has 1 sign bit and $(n-1)$ binary bits. The resolution of the Most Significant Bit (MSB) is $2^{r_{MSB}}$, decreasing exponentially to the resolution of the Least Significant Bit (LSB), which is $2^{r_{MSB}+2-n}$.

This approach ensures that the quantization process is dynamically adaptable, enhancing the precision of the weights while maintaining a balance between model size and performance. Post-linear and nonlinear operations, the resultant features are quantized to a signed 8-bit format, also utilizing a configurable fixed-point approach. This method of quantization for activations allows for a more nuanced representation of the feature space, catering to the intricate variations in the data while ensuring computational efficiency.

By quantization of weights and activations, the model size is reduced significantly, becoming more suitable for deployment on hardware with limited computational resources. Furthermore, the configurable nature of the fixed-point representation in both weight and activation quantization allows for a flexible adjustment of the trade-off between accuracy and computational demand. This adaptability is crucial in ensuring that the proposed system remains both efficient and effective in real-world applications.

## III. BPF-based CWT Approximation

### A. Theory Derivation

Continuous Wavelet Transform (CWT) represents signal frequency content at different scales and time intervals simultaneously. ECG signals are inherently non-stationary, with rapid transitions that can be critical for identifying arrhythmic events. CWT provides a powerful tool to analyze such signals in both time and frequency domains simultaneously. This dual-domain representation enhances the visibility of transient features that are often masked in purely time-domain analysis, making it particularly suitable for detecting subtle pathological changes. The original CWT formula is written as:

$$CWT(a,b) = \frac{1}{\sqrt{a}}\int_{-\infty}^{\infty} f(t)\cdot\varphi(\frac{t-b}{a})dt \qquad (4)$$

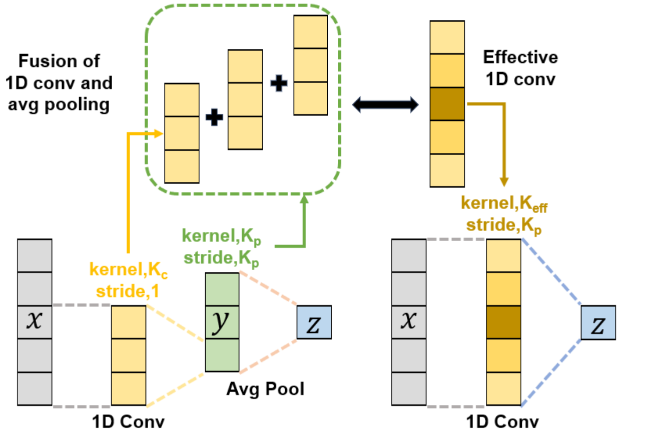


Fig. 8. Fusion of 1D convolution and average pooling into an effective 1D convolution operation for efficient hardware implementation.

where $f(t)$ represents the input time domain signal, $a$ and $b$ are the scale and translation of the transform, respectively and $\varphi$ is the wavelet function. In this design, Morlet wavelet, which is a multiplication of a complex exponential and a Gaussian window is chosen as the wavelet function. In contrast to Haar or Daubechies wavelets, the complex Morlet wavelet was selected for its superior time-frequency localization and phase sensitivity, which are critical for accurately capturing the subtle morphological features of ECG signals. The representation of Morlet is shown as

$$\varphi(t) = e^{i\omega t}e^{-\frac{t^2}{2}} \qquad (5)$$

In practical applications, the complex exponential function in the $e^{i\omega t}$ Morlet wavelet can be approximated with cosine function $\cos(\omega t)$ with small $\omega t$.

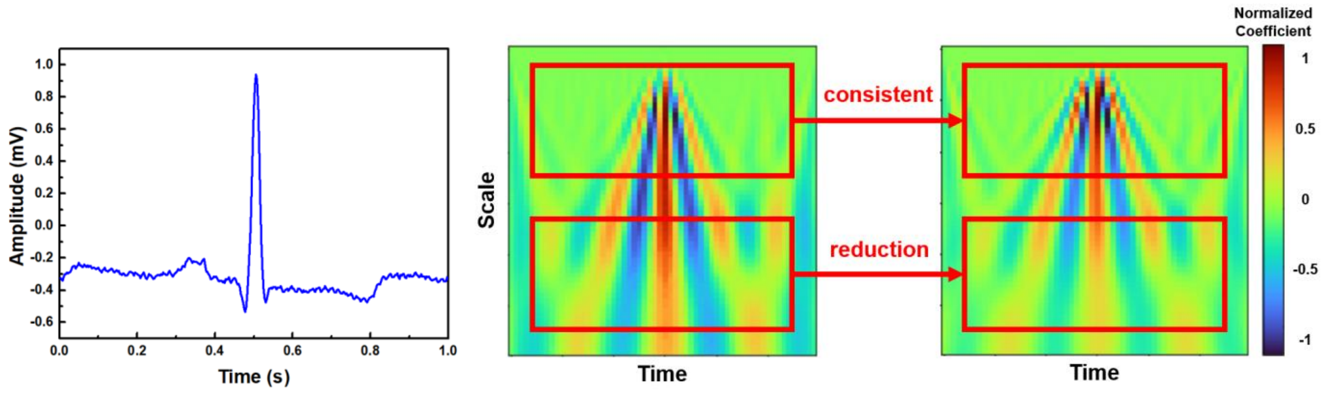
$$\varphi'(t) = \cos(\omega t)\cdot e^{-\frac{t^2}{2}} \qquad (6)$$

The cosine approximation for the Morlet wavelet is particularly effective for ECG signals, because of its low frequency characteristic. By using cosine instead of the full complex format, CWT hardware implementation in FPGA has reduced resource requirements, leading to faster computations and improved efficiency. This simplification also lowers power consumption, which is crucial for low-power medical devices. Despite the simplification, the accuracy for high-frequency feature extraction remains largely unaffected, making it an efficient approach for edge computing scenarios. Replace the wavelet function in CWT, we have

$$CWT'(a,b) = \frac{1}{\sqrt{a}}\int_{-\infty}^{\infty} f(t)\cdot\cos(\omega\cdot\frac{t-b}{a})\cdot e^{-\frac{(\frac{t-b}{a})^2}{2}}dt \quad (7)$$

To implement the CWT on FPGA, (7) is rewritten in a discrete format, where integration is replaced with summation and the continuous parameters are discretized.

$$CWT'(j,k) = \sum_n f[n]\cdot\varphi'[n-2k]$$



(a) (b) (c)

Fig. 9. An example of ECG signal, (a) original signal wave, is processed with (b) software CWT on computer and (c) proposed simplified half precision CWT on FPGA. All FPGA input signals were provided in IEEE 754 half-precision (float16) format.

$$= \frac{1}{\sqrt{j}} \sum_n f[n] \cdot \cos\left(\omega \cdot \frac{n-2k}{j}\right) \cdot e^{-\frac{\left(\frac{n-2k}{j}\right)^2}{2}} \qquad (8)$$

where $f[n]$ represents the discrete signal at sample $n$, $j$ and $k$ are the scale and translation of the transform, respectively.

## IV. FPGA Implementation and Acceleration

### A. Implementation of CWT Approximation

Following (8), the simplified discrete CWT can be segmented into two main components: wavelet generation and 1D convolution, followed by average pooling. After the scale and translation parameters of the Morlet wavelet are determined, the wavelet kernels across scales from 1 to 90 are precomputed offline and stored as a wavelet matrix on the FPGA, as illustrated in Fig. 7. This matrix contains the discrete wavelet coefficients for each scale and is independent of the specific input data, allowing for rapid access and reuse during inference. For each new input signal segment, the FPGA retrieves the appropriate pre-stored wavelet kernel for each scale and performs a 1D convolution with the input signal to obtain the set of CWT coefficients. This design enables real-time, resource-efficient processing without the need for runtime wavelet generation.

After obtaining the CWT coefficients for all scales, an average pooling operation is applied along the temporal translation dimension. While this pooling step is not part of the classical CWT mathematical definition, it is incorporated in our hardware design to reduce the size of the feature map and minimize resource usage in the subsequent neural network layers. By aggregating adjacent CWT coefficients, average pooling compresses the feature representation, achieving a favorable balance between information preservation and hardware efficiency.

To further reduce computational overhead, the wavelet convolution and average pooling operations are fused into a single wavelet matrix, as illustrated in Fig. 8. This integration removes the need for separate hardware dedicated to average pooling, thereby lowering hardware complexity. The fused CWT–pooling layer is realized as a one-dimensional convolutional operation with 60 output channels, implemented using the same hardware architecture as standard convolutional layers. The hardware implementation of discrete CWT apporximation is compared with software CWT in Fig. 9. It can be observed that results at smaller scales (high frequencies) are very close to the original, while at larger scales (low frequencies), a slight reduction occurs. The reason is that at smaller scales, wavelet transforms focus on the local details of the signal, which are dominated by high-frequency components. The cosine function effectively approximates these local structures, and since the amplitude of high-frequency components in ECG signals is typically low, the omission of the imaginary part (sine component) results in negligible information loss, thereby maintaining accuracy. The effect on ECG signal classification was minimal after QAT training, and the approximation significantly reduced FPGA resource consumption by eliminating the need for complex arithmetic units, thereby enhancing computation speed and efficiency, which is particularly beneficial for edge computing devices.

### B. Neural Network Hardware Architecture

After the quantization-aware training process is completed, the finalized model is exported and fully mapped onto the FPGA hardware. The FPGA operates only as an inference accelerator, using the fixed weights and quantization parameters determined during offline training. No further software process is required for quantization or model adjustment at runtime. The FPGA-based implementation of the neural network (NN) hardware architecture is illustrated in Fig. 10. As shown in Fig. 10(a), the design adopts a layer-by-layer streaming architecture employing dedicated hardware modules, which is particularly advantageous for compact neural networks in tinyML applications. This architecture offers high adaptability to support network compression techniques, such as pruning and quantization, that introduce structural irregularities.

The streaming architecture on FPGA allows for customized dataflow in all hardware blocks, which can be tailored to minimize control logics and memory size. This distinct advantage of FPGA implementation is leveraged to enable more efficient Inception-ResNeXt acceleration. For instance, the IRN block shown in Fig. 4(b) is implemented as 2 IRN

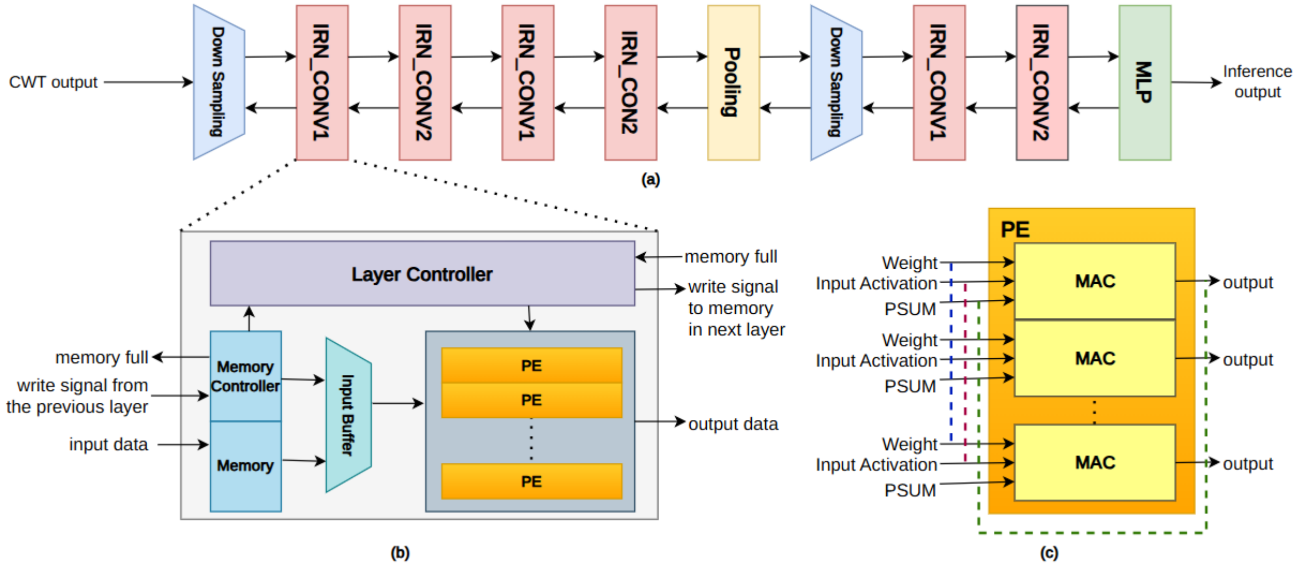


Fig. 10. Implementation of (a) streaming architecture with different hardware blocks catered to different layers, (b) architecture of each hardware blocks shown in 10(a), (c) architecture of each PE shown in 10(b). The PEs can be configured to achieve any type of input sharing. Additional output buffer may be added to suit different dataflows.

blocks, IRN_CONV1 and IRN_CONV2 as shown in Fig. 10(a). IRN_CONV1 includes the first 3×3 convolutional layer and the 1×1 convolutional layer, which share the same control logic, weight memory, and input activation memory. The output activations require special handling because all activations from the 1×1 convolutional layer and half of the activations from the first 3×3 convolutional layer do not pass through the second 3×3 convolutional layer. Therefore, half of the activations from the 1×1 convolutional layer and the 3×3 convolutional layer are added together before being sent to the IRN_CONV2 block, reducing the activation memory required in IRN_CONV2. The memory controller of the IRN_CONV2 block is then customized to read only half of the input feature maps for the PEs to perform the second 3×3 convolution, while the other half of the feature maps are bypassed and directly forwarded as output activations of the IRN_CONV2 block. This special memory handling can only be achieved by customizing the memory controller shown in Fig. 10(b) of the IRN_CONV2 block to efficiently store and read these activations for processing. Similarly, for downsampling layers, pooling layers, and the MLP layer, the logics are customized to achieve the ideal balance between resource usage and accelerator performance.

Furthermore, because different layers may require distinct quantization levels as determined by the configurable QAT, our hardware design integrates dedicated hardwired logic within each layer's processing block to natively support these layer-specific quantization levels. This eliminates the need for additional shifters or memory buffers and demonstrates one of the key advantages of adopting a streaming architecture.

### C. Dataflow for Efficient Throughput Balancing

One significant drawback of using a streaming architecture with dedicated layer modules is the low utilization of processing elements (PEs). PEs start to process only when the input data is ready, leading to considerable idle states while waiting for inputs. The ideal scenario is to have the same throughput for all layers, enabling maximum pipeline utilization. However, achieving this is challenging due to the large variety of NN architectures and structures.

The most promising approach for throughput balancing is through row or column parallelism [37]. Using this input stationary-output stationary (ISOS) dataflow, the input consumption and output production of each layer are balanced. However, achieving a highly balanced pipeline using ISOS would require a huge number of PEs, which is not practical for edge applications.

Therefore, instead of balancing the latency processing one row or one column for each layer, we balance the average latency to process all output activation pixels for each layer. This method reduces PE utilization and increases memory footprint by a small margin compared to the ISOS dataflow, but it significantly reduces the number of PEs required.The throughput of each layer can be estimated as shown in equation below.

$$Throughput \approx \frac{Unroll\ Factor}{\begin{array}{c}Number\ of\\ input\ pixels\\ per\ output\ pixel\end{array} \times \begin{array}{c}Time\ needed\\ to\ compute\ 1\\ input\ pixel\end{array} \times \begin{array}{c}Number\\ output\ pixel\end{array}} \qquad (9)$$

The throughput of each layer can be adjusted by tuning the unroll factor and the time needed to compute one input pixel. For layers that require a larger number of input pixels to compute one output pixel, we can increase the unroll factor by introducing more PEs or introduce faster MAC units for quicker computation, or both. Conversely, for layers that require a smaller number of input pixels, we can introduce slower MAC units such as bit-serial MAC units to reduce resource usage without sacrificing performance. Moreover, the PE architecture shown in Fig. 10(c) can be configured to cater to different dataflows suitable for different NN layers, adding another

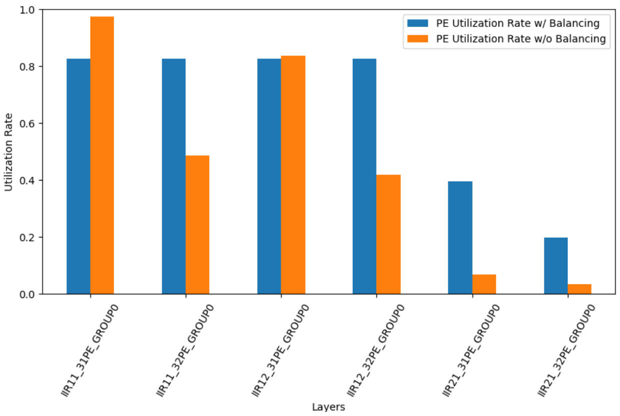


Fig. 11. PE utilization rate w/ & w/o throughput balancing.

degree of freedom for throughput tuning in each layer block. When the throughput of all layer blocks roughly matches the input consumption rate of the subsequent blocks, PE utilization is more balanced, and higher accelerator performance can be achieved while maintaining low resource usage. As shown in Fig. 11, PE utilization across different layers are better balanced with the throughput balancing technique introduced.

## V. Measurement Results

As the preprocessing by MAX30003 is fully hardware-based and incurs negligible latency and resource cost compared to FPGA inference, it is not separately measured in our reported results. The pre-processed data were converted to half-precision format (IEEE 754 binary16 [38]). This conversion was performed to optimize computational efficiency and resource utilization during the processing phase. The half-precision ECG signals were then processed utilizing the proposed Continuous Wavelet Transform (CWT) approximation method. This approximation technique was selected for its ability to effectively analyze the frequency components of the ECG signals, which is crucial for identifying arrhythmic events within the data.

QAT training was implemented in an environment equipped with an Intel 13600KF CPU, 32GB of RAM, and an NVIDIA RTX 3060Ti GPU. The model was optimized using stochastic gradient descent with a momentum of 0.8 and an initial learning rate of 0.01. Training was conducted over 300 epoch with a batch size of 200, and dropout (p=0.5) was applied in the fully connected layer to mitigate overfitting. Additionally, quantization-aware training was integrated using custom modules that perform 4-bit weight and 8-bit activation quantization to balance accuracy and computational efficiency. The proposed configurable quantization-aware training process was designed to enhance the model ability to maintain accuracy when deployed in environments with limited computational resources by incorporating quantization considerations directly into the training process, enabling the network to learn compensation strategies for the quantization-induced noise and thereby preserve critical feature representations despite reduced bit precision. Furthermore, by adaptively adjusting quantization precision across different layers, this approach minimizes the typical accuracy degradation associated with quantization while reducing memory and computational requirements. Fig. 12 shows the results of the configurable quantization-aware training process. The training phase achieved a remarkable accuracy from 93.28% for the model trained without configurable quantization-aware training to 99.5% under 4-bit weight quantization and 8-bit activation quantization, which also showcased the effectiveness of the proposed model.

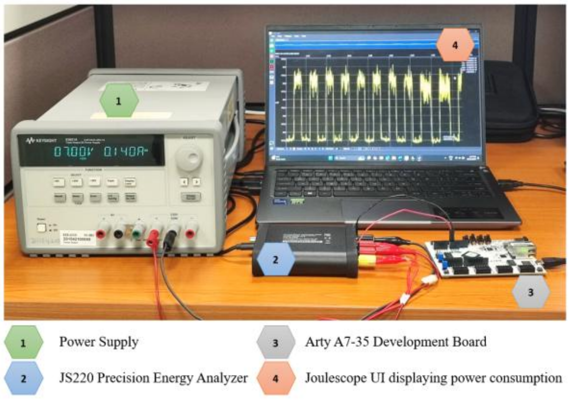

1
2
3
4
Power Supply
Arty A7-35 Development Board
JS220 Precision Energy Analyzer
Joulescope UI displaying power consumption

Fig. 13. Power consumption measurement setup with Power JS220 Precision Energy Analyzer. The higher power levels shown on the screen correspond to periods when the board is actively performing inference.

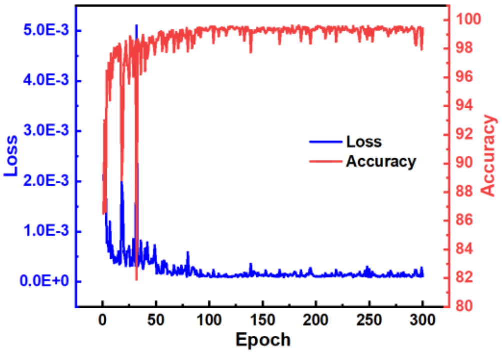


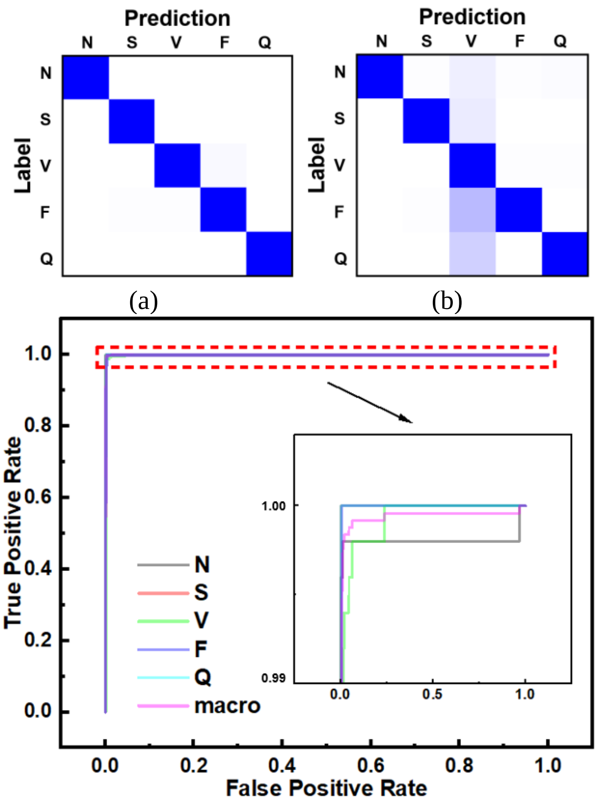
Fig. 12. The training process of the Inception-ResNeXt model

The proposed Inception-ResNeXt model is deployed on an Arty A7-35 development board with the architecture shown in Fig. 10. To ensure an efficient and reliable implementation, the proposed architecture was described in HDL and developed using the standard Xilinx Vivado [44] 2022.1 toolflow for synthesis, implementation and bitstream generation. The generated bitstream was then deployed onto the Arty A7-100T FPGA development board. For performance evaluation, batches of preprocessed ECG signals were transmitted from a host laptop to the FPGA board via a USB interface. The FPGA performed real-time inference, and the resulting classifications were collected on the host for accuracy evaluation. Average inference latency was measured directly on the hardware using timestamping at the input and output of each batch. Power consumption was measured following the setup shown in Fig. 13. The FPGA board was powered through a JS220 precision power analyzer, with the power consumption monitored via the analyzer's user interface, in line with the manufacturer's guidelines [45]. The implementation of model on an FPGA

TABLE I. COMPARISONS WITH STATE-OF-THE-ART FPGA IMPLEMENTATIONS ON 5-CLASS-MIT-BIH DATASET

| | [39] | [40] | [41] | [42] | [43] | **This Work** (w/o throughput balancing) | **This Work** (w/ throughput balancing) |
|---|---|---|---|---|---|---|---|
| Feature Extraction | LC-CTDA | N.R. | LC | BernoulliRBM | N.R. | CWT | CWT |
| Network Architecture | ANN | CNN | SNN | CNN | CNN | Inception-ResNeXt | Inception-ResNeXt |
| Model Size (Kbit) | 59.5 | 177 | 75.6 | 51200 | 6.55 | 25.4 | 25.4 |
| Hardware Platform | Pynq-Z2 | ZC706 | XC7A100T | PYNQ-Z1 | PYNQ-Z2 | XC7A35T | XC7A35T |
| LUTs | 6293 | 2510 | 659 | 17579 | 10877 | 8414 | 11364 |
| DSPs | 0 | 96 | 0 | 85 | 53 | 0 | 0 |
| FFs | 1331 | N.R. | 783 | 20060 | 1949 | 15971 | 21118 |
| BRAMs | 0 | N.R. | 17 | 39.5 | 2 | 28 | 28 |
| Inference Latency (ms) | 2.52 | 38.6 | 0.504 | 14 | 0.233 | 1.78 | 0.350 |
| Power | N.R. | N.R. | N.R. | 1.53* | 0.131 | 1.00*/0.0893 | 1.02*/0.219 |
| Energy/ Inference (mJ) | N.R. | N.R. | N.R. | 21.42* | 0.0305 | 1.78*/0.158 | 0.357*/0.0767 |
| Accuracy (%) | 99.2 | 98.9 | 98.2 | 99.1 | 96.5 | 99.5 | 99.5 |

[a] * Measurement based on the entire development board rather than just the FPGA core. N.R.: Not reported.



(a) (b)

(c)

Fig. 14. The confusion matrix for the final model's ECG signal classification is displayed, providing a detailed breakdown of classification performance across different arrhythmia types, with a comparison between models trained (a) with and (b) without quantization-aware training. (c) ROC curves of all ECG classes and macro-average, showing high discriminative power of the proposed model.

platform was evaluated to assess its real-world applicability and performance. The confusion matrix, a pivotal tool for visualizing the performance of classification models is shown in Fig. 14(a). This matrix provides a detailed breakdown of the model predictive capabilities across different classes, offering insights into its precision and reliability in classifying ECG signals. To underscore the impact of the configurable quantization-aware training on the model performance, the confusion matrix of the model trained without the incorporation of configurable quantization-aware training is presented in Fig. 14(b). Notably, it was observed that the accuracy of the FPGA implementation achieves significantly 99.5% for this software-hardware co-design model benefiting from this innovative configurable quantization aware training approach. To provide a more comprehensive evaluation of the class-wise discrimination, Fig. 14(c) further presents the receiver operating characteristic (ROC) curves and corresponding macro-average for all five ECG classes. All classes achieve high AUC values, and the inset highlights the near-perfect separability in the high-AUC region, demonstrating the excellent performance and robustness of the proposed approach in multi-class ECG classification.

Table. I summarized the measurement results and compared with other recent state-of-the-art designs. It can be observed that this design achieved the highest accuracy among all implementations, demonstrating the performance of the proposed network architecture. Additionally, as depicted in Fig. 15, this design achieved the highest accuracy with the smallest model size, providing a good balance between model size and accuracy. Furthermore, this design achieved low energy consumption per inference, which is a significant advantage for edge devices. This is attributed to the implemented streaming architecture, where the neural network is directly mapped onto the FPGA, resulting in an optimized and efficient hardware implementation. These results also demonstrate that FPGAs can serve as an ideal platform for software–hardware co-design, particularly in the context of neural network implementation.

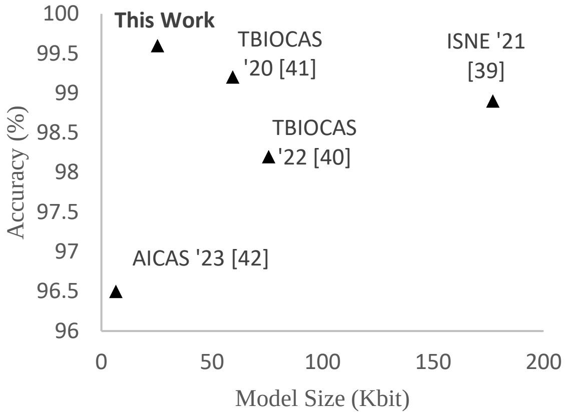


Fig. 15. The relationship between model accuracy and model size is depicted. Higher accuracy is preferable, while a smaller model size is more desirable.

Although lower energy per inference is reported in [32], its accuracy is not comparable to our implementation. This highlights that SNNs still lag behind CNNs in applications where high precision is critical—such as those requiring medical reliability. We prioritize accuracy above all the other metrices because accuracy is the most important metric for medical reliability. The results with and without the throughput balancing method is compared in section IV.B. With balanced throughput among all hardware blocks, the inference latency is reduced by 5 times with 35.1% increase in LUT utilization. This result highlights the importance of throughput balancing in a pipelined streaming architecture.

## VI. Discussion

This study introduces an FPGA-based, real-time ECG classification system that integrates an innovative Inception-ResNeXt architecture with configurable quantization-aware training and a cosine-approximated Continuous Wavelet Transform, achieving 99.5% inference accuracy on the MIT-BIH dataset. By embedding the quantization process into the training phase, the model effectively learns to counteract quantization-induced errors, a challenge that has traditionally led to accuracy degradation in similar resource-constrained applications. This approach, in contrast to earlier methods, not only preserves critical signal features but also enhances overall performance, thereby setting a new benchmark in the domain of low-power biomedical signal processing.

The implications of these findings are significant for the development of edge-AI devices, especially wearable health monitors that demand real-time performance and low power consumption. Implementing a multiclass classification system for cardiac irregularities directly on FPGA hardware enables rapid, edge-based detection of arrhythmias and other cardiac events without reliance on cloud processing. This local processing is essential for time-sensitive applications, such as pre-hospital emergency management and continuous patient monitoring where immediate, accurate feedback can be lifesaving. Furthermore, our hardware-software co-design strategy and optimized dataflow facilitate advanced neural network operations on FPGA platforms, delivering an energy-efficient and reliable solution for continuous cardiac monitoring. This work contributes to the broader research discourse by demonstrating that adaptive quantization strategies can reconcile the trade-off between computational efficiency and model accuracy, thereby broadening the scope of practical applications in personalized healthcare.

Nevertheless, the current system presents some limitations. Its validation is restricted to the MIT-BIH dataset, which may not fully represent the variability and noise encountered in diverse clinical environments. Comprehensive evaluation on independent and more heterogeneous ECG collections remains an important direction for future research, and we will continue to expand our validation as access to broader datasets and hardware resources becomes available. Moreover, while the fixed-point quantization method reduces memory usage and computational load, its performance under different signal conditions and with other types of biomedical data remains to be verified. Future research should focus on expanding the evaluation to more heterogeneous datasets, developing adaptive quantization techniques that dynamically respond to signal variations, and further refining the hardware architecture to reduce latency and enhance throughput.

## VII. Conclusion

This work presents a hardware-adaptive, configurable quantization-aware training (QAT) framework as the foundation of an efficient FPGA-based real-time ECG classification system. By embedding layer-wise, flexible quantization into the training loop and co-designing both the Inception-ResNeXt neural network and cosine-approximated CWT pipeline for hardware deployment, the proposed approach achieves state-of-the-art accuracy and energy efficiency at ultra-low bit-widths. Comprehensive FPGA implementation and measurements on the Arty A7-35 platform demonstrate 99.5% inference accuracy on the MIT-BIH ECG dataset, with a compact 6-layer network achieving an average inference latency of 0.35 ms, dynamic power of 200 mW, and energy efficiency of 0.0767 mJ per inference. These results not only surpass existing FPGA-based solutions in both accuracy and model size but also validate the effectiveness of our hardware-driven QAT methodology for reliable, low-power edge-AI biomedical applications. This work thus advances both the theoretical and practical frontiers of software–hardware co-design for edge intelligence in healthcare.

## References

[1] Y. LeCun, Y. Bengio and G. Hinton, “Deep learning,” *Nature*, vol. 521, pp. 436-444, 2015.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.

[3] X. He, K. Wang, H. Huang, T. Miyazaki, Y. Wang, and S. Guo, “Green resource allocation based on deep reinforcement learning in content-centric IoT,” *IEEE Trans. Emerg. Topics Comput.*, Feb. 13, 2018.

[4] S. S. Virani et al., “Heart disease and stroke statistics—2021 update,” *Circulation*, vol. 143, no. 8, pp. 1–6, Feb. 2021.

[5] T. Yang, L. Yu, Q. Jin, L. Wu, and B. He, "Localization of origins of premature ventricular contraction by means of convolutional neural network from 12-Lead ECG," *IEEE Trans. Biomed. Eng.*, vol. 65, no. 7, pp. 1662–1671, Jul. 2018.

[6] X. Sun, R. Liu, X. Peng and S. Yu, "Computing-in-Memory with SRAM and RRAM for Binary Neural Networks," *2018 14th IEEE International Conference on Solid-State and Integrated Circuit Technology (ICSICT)*, 2018.

[7] M. Courbariaux, et al., "Binarized neural network: Training deep neural networks with weights and activations constrained to+ 1 or-1," arXiv: 1602.02830, 2016.

[8] T. Cao et al., "A non-idealities aware software–hardware co-design framework for edge-AI deep neural network implemented on memristive crossbar," *IEEE J. Emerging Sel. Top. Circuits Syst.*, vol. 12, no. 4, pp. 934-943, Dec. 2022.

[9] M. Rastegari, et al., "XNOR-net: ImageNet classification using binary convolutional neural networks," arXiv: 1603.05279, 2016.

[10] T. Cao et al., "RRAM-PoolFormer: a resistive memristor-based PoolFormer modeling and training framework for edge-AI applications," *2023 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2023.

[11] Y. Zhao, Z. Shang, and Y. Lian, "A 13.34 μw event-driven patient-specific ANN cardiac arrhythmia classifier for wearable ECG sensors," *IEEE Trans. Biomed. Circuits Syst.*, vol. 14, no. 2, pp. 186–197, Apr. 2020.

[12] M. Saeed et al., "Evaluation of level-crossing ADCs for event-driven ECG classification," *IEEE Trans. Biomed. Circuits Syst.*, vol. 15, no. 6, pp. 1129–1139, Dec. 2021.

[13] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2818-2826.

[14] J. Loh and T. Gemmeke, "Stream processing architectures for continuous ECG monitoring using subsampling-based classifiers," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 32, no. 1, pp. 68–78, Jan. 2024, doi: 10.1109/TVLSI.2023.3329360.

[15] M. Janveja, A. K. Sharma, A. Bhardwaj, J. Pidanic, and G. Trivedi, "An optimized low-power VLSI architecture for ECG/VCG data compression for IoHT wearable device application," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 31, no. 12, pp. 2008–2015, Dec. 2023, doi: 10.1109/TVLSI.2023.3314611.

[16] M. Bahrami and M. Forouzanfar, "Sleep apnea detection from single-lead ECG: A comprehensive analysis of machine learning and deep learning algorithms," *IEEE Trans. Instrum. Meas.*, vol. 71, Art. no. 4003011, pp. 1–11, 2022, doi: 10.1109/TIM.2022.3151947.

[17] Z. Zhang, Y. Guan, and W. Ye, "An energy-efficient ECG processor with ultra-low-parameter multistage neural network and optimized power-of-two quantization," *IEEE Trans. Biomed. Circuits Syst.*, vol. 18, no. 6, pp. 1296–1307, Dec. 2024, doi: 10.1109/TBCAS.2024.3385993.

[18] S. Liu, Y. Liang, Z. Zhang, and P. Wan, "FPGA implementation of staged projection refining multiple orthogonal matching pursuit algorithm for compressed sensing," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 33, no. 5, pp. 1334–1347, May 2025, doi: 10.1109/TVLSI.2025.3529954.

[19] D. L. T. Wong, Y. Li, D. John, W. K. Ho, and C.-H. Heng, "Low complexity binarized 2D-CNN classifier for wearable edge AI devices," *IEEE Trans. Biomed. Circuits Syst.*, vol. 16, no. 5, pp. 822–831, Oct. 2022, doi: 10.1109/TBCAS.2022.3196165.

[20] J. Lai, H. Tan, J. Wang, *et al.*, "Practical intelligent diagnostic algorithm for wearable 12-lead ECG via self-supervised learning on large-scale dataset," *Nat. Commun.*, vol. 14, Art. no. 3741, June 2023, doi: 10.1038/s41467-023-39472-8.

[21] A. Khunte, V. Sangha, E. K. Oikonomou, *et al.*, "Detection of left ventricular systolic dysfunction from single-lead electrocardiography adapted for portable and wearable devices," *NPJ Digit. Med.*, vol. 6, Art. no. 124, July 2023, doi: 10.1038/s41746-023-00869-w.

[22] D. Li, J. Zhang, Q. Zhang and X. Wei, "Classification of ECG signals based on 1D convolution neural network," 2017 IEEE 19th International Conference on e-Health Networking, Applications and Services (Healthcom), Dalian, China, 2017, pp. 1-6.

[23] L. Wei, D. Liu, J. Lu, L. Zhu and X. Cheng, "A low-cost hardware architecture of convolutional neural network for ECG classification," *2021 9th International Symposium on Next Generation Electronics (ISNE)*, Changsha, China, 2021, pp. 1-4.

[24] S.S. Kulkarni; S.O. Rajankar, "Preprocessing techniques of electrocardiogram," *Int. J. Eng. Comput. Sci.*, vol. 5, pp. 16746-16748, 2016.

[25] P. Madan; V. Singh; D.P. Singh; M. Diwakar; B. Pant; A. Kishor, "A hybrid deep learning approach for ECG-based arrhythmia classification," *Bioengineering*, vol. 9, pp. 152, 2022.

[26] A. Ullah, S.M. Anwar, M. Bilal, and R.M. Mehmood, "Classification of arrhythmia by using deep learning with 2-D ECG spectral image representation," *Remote Sens.*, vol. 12, pp. 1685, 2020.

[27] D. L. T. Wong, Y. Li, D. John, W. K. Ho and C. -H. Heng, "An energy efficient ECG ventricular ectopic beat classifier using binarized CNN for edge AI devices," *IEEE Trans. Biomed. Circuits Syst.*, vol. 16, no. 2, pp. 222-232, April 2022.

[28] D. L. T. Wong, Y. Li, D. John, W. K. Ho and C. -H. Heng, "Low complexity binarized 2D-CNN classifier for wearable edge AI devices," *IEEE Trans. Biomed. Circuits Syst.*, vol. 16, no. 5, pp. 822-831, Oct. 2022.

[29] T. Cao, Z. Zhang, W. L. Goh, C. Liu, Y. Zhu and Y. Gao, "A ternary weight mapping and charge-mode readout scheme for energy efficient FeRAM crossbar compute-in-memory system," *2023 IEEE 5th International Conference on Artificial Intelligence Circuits and Systems (AICAS)*, Hangzhou, China, 2023.

[30] T. Cao, W.S. Ng, W.L. Goh and Y. Gao, "DWT-PoolFormer: discrete wavelet transform-based quantized parallel PoolFormer network implemented in FPGA for wearable ECG monitoring," *2024 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, Xi'an, China, 2024.

[31] B. Dal and M. Aşkar, "Fixed-point FPGA implementation of ECG classification using artificial neural network," *2022 Medical Technologies Congress (TIPTEKNO)*, Antalya, Turkey, pp. 1-4, 2022.

[32] T. Cao, Z. Zhang, W. L. Goh, C. Liu, Y. Zhu and Y. Gao, "ECG Classification using Binary CNN on RRAM Crossbar with Nonidealities-Aware Training, Readout Compensation and CWT Preprocessing," *2023 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, Toronto, ON, Canada, 2023, pp. 1-5, doi: 10.1109/BioCAS58349.2023.10389002.

[33] C. Szegedy, V. Vanhoucke, S. Ioffe, et al., "Rethinking the inception architecture for computer vision," *IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 2818-2826, 2016.

[34] *MIT-BIH Arrhythmia Database*. Massachusetts Institute of Technology, 1975. [Online]. Available: https://www.physionet.org/content/mitdb/.

[35] S. Dalal and V. P. Vishwakarma, "Classification of ECG signals using multi-cumulants based evolutionary hybrid classifier," *Sci. Rep.*, vol. 11, p. 15092, 2021, doi: 10.1038/s41598-021-94363-6.

[36] T. Cao *et al.*, "Edge PoolFormer: modeling and training of PoolFormer network on RRAM crossbar for Edge-AI applications," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, doi: 10.1109/TVLSI.2024.3472270.

[37] Y. Yang, J. S. Emer and D. Sanchez, "ISOSceles: Accelerating Sparse CNNs through Inter-Layer Pipelining," *2023 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, Montreal, QC, Canada, 2023.

[38] IEEE Standard for Floating-Point Arithmetic, IEEE Std 754-2008 (Revision of IEEE Std 754-1985), pp. 1–70, Aug. 2008, doi: 10.1109/IEEESTD.2008.4610935.

[39] Y. Zhao, Z. Shang and Y. Lian, "A 13.34 μW Event-Driven Patient-Specific ANN Cardiac Arrhythmia Classifier for Wearable ECG Sensors," in *IEEE Transactions on Biomedical Circuits and Systems*, vol. 14, no. 2, pp. 186-197, April 2020.

[40] L. Wei, D. Liu, J. Lu, L. Zhu and X. Cheng, "A low-cost Hardware Architecture of Convolutional Neural Network for ECG Classification," *2021 9th International Symposium on Next Generation Electronics (ISNE)*, Changsha, China, 2021

[41] H. Chu *et al.*, "A Neuromorphic Processing System With Spike-Driven SNN Processor for Wearable ECG Classification," in *IEEE Transactions on Biomedical Circuits and Systems*, vol. 16, no. 4, pp. 511-523, Aug. 2022.

[42] K. Inadagbo, B. Arig, N. Alici and M. Isik, "Exploiting FPGA Capabilities for Accelerated Biomedical Computing," *2023 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, Poznan, Poland, 2023.

[43] M. -Y. Ku, T. -S. Zhong, Y. -T. Hsieh, S. -Y. Lee and J. -Y. Chen, "A High Performance Accelerating CNN Inference on FPGA with Arrhythmia Classification," *2023 IEEE 5th International Conference on Artificial Intelligence Circuits and Systems (AICAS)*, Hangzhou, China, 2023.

[44] AMD, "*Vivado Design Suite User Guide: Getting Started UG910 (v2025.1)*", May 2025.

[45] Jetperch LLC, "*Joulescope JS220 User's Guide*," January 2025.

**Tiancheng Cao** (Member, IEEE) received his B.Eng. (Highest Distinction) from Nanyang Technological University, Singapore (NTU) in 2021, supported by the NTU Science and Engineering Undergraduate Scholarship, and his Ph.D. in 2024 under the Nanyang President Graduate Scholarship with Best PhD Thesis Award. During his doctoral studies, he also served as a attached researcher at Institute of Microelectronics (IME) at the Agency for Science, Technology and Research (A*STAR), Singapore.

He is currently a Schmidt AI in Science Fellow supported by Schmidt Sciences at the Centre for System Intelligence and Efficiency (CSIE), NTU, Singapore. He has led and contributed to several real-time edge-AI systems for biomedical applications and has published extensively in top-tier journals and conferences in the fields of circuits and systems. His research interests span neuromorphic computing, edge computing, Internet of Medical Things (IoMT), and translational medicine.

**Wei Soon Ng** received the B.E. degree in Electrical and Electronic Engineering from Nanyang Technological University (NTU), Singapore, in 2021. He is currently pursuing the Ph.D. degree with the School of Electrical and Electronic Engineering, NTU. His research interests include efficient hardware acceleration for Tiny Machine Learning (TinyML) and Field-Programmable Gate Array (FPGA) implementation.

**Wang Ling Goh** (Senior Member, IEEE) received both her Bachelor of Engineering Degree in Electrical and Electronic Engineering and Doctor of Philosophy in Microelectronics from the Department of Electrical and Electronic Engineering at the Queen's University of Belfast in United Kingdom. She joined the School of Electrical and Electronic Engineering (EEE) at the Nanyang Technological University (NTU), Singapore as a lecturer and became an Associate Professor in 2004. Dr Goh had served in various academic positions such as Associate Dean (Academic) at the Graduate College, Deputy Director (Undergraduate) of the Renaissance Engineering Programme, Associate Dean (Outreach & External Relations) at the College of Engineering, as well as the Assistant Chair of Students and Assistant Head of Division, both at the School of EEE. She is currently the Programme Coordinator of the Undergraduate Programme and the Deputy Director for the Master of Science Programme (Electronics) at the School of EEE. She is also a Co-Chair of the NTU Teaching Excellent Academy.

Dr Goh had been a General Chair and advisory/technical committee member at various international conferences. Dr Goh's research interests include digital/mixed-signal Integrated Circuit (IC), biomedical circuits and neuromorphic IC. She has co-authored 1 international professional technical reference text, filed 16 patents, and published ~280 research papers in international journals and conferences. Dr Goh has trained more than 200 graduate students.

**Yuan Gao** (Member, IEEE) received the B.E and M.E degrees in electrical engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2000 and 2002, respectively, and the Ph.D. degree in electrical engineering from the National University of Singapore, Singapore, in 2008.

Since 2007, he has been with the Institute of Microelectronics (IME), Agency for Science, Technology and Research (A*STAR), Singapore. He is currently a principal investigator and principal scientist in the Integrated Circuit Design and Systems (ICDS) Department, where he is leading the next generation intelligent sensor interface IC development. He has authored or coauthored 3 book chapters, more than 120 peer-reviewed international journal and conference papers and has more than 10 US patents granted or filed. He has co-supervised 8 PhD students and he is an accredited A*STAR PhD Scholar supervisor. He received A*STAR Graduate Academy Star Mentor Award in 2023 and IEEE Solid State Circuit Society Outstanding Reviewer Award in 2023. His primary research areas include energy efficient analog and mixed-signal IC design in the emerging areas such as AI hardware, intelligent sensor interface, biomedical microsystem and energy harvesting.

Dr. Gao was TPC member of the IEEE International Solid-State Circuits Conference (ISSCC) between 2015 – 2020 and served as Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS−I: REGULAR PAPERS between 2020 – 2022. Currently he is an Associate Editor of the IEEE TRANSACTIONS ON BIOMEDICAL CIRCUITS AND SYSTEMS.

Hen-Wei Huang received the B.S. and M.S. degrees in mechanical engineering from the National Taiwan University, Taiwan, in 2011 and 2012, respectively, and the Ph.D. degree in robotics from ETH Zürich in 2018.

He is currently an Assistant Professor at the School of Electrical and Electronic Engineering as well as the LKC School of Medicine, Nanyang Technological University, Singapore. He is also the director of the DARE Lab. His research interests include in vivo wireless sensor networks, ingestible electronics, robotic-assisted drug delivery, and translational medicine.