

FPGA-Based Low-Power Architecture for Real-Time Arrhythmia Detection Using Fixed-Length Segmentation

Guduru Kethan Kumar¹, Dr. B. Anuradha²

¹M. Tech Student Department of Communication Engineering, S. V. University College of Engineering
Tirupati, A. P. India

²Professor Department of Communication Engineering, S. V. University College of Engineering, Tirupati
A. P. India

doi.org/10.64643/IJIRT208776

Abstract—Data processing and low power consumption are essential for monitoring heart signals in real time[5],[7]. This project is a prototype of a real-time electrocardiogram (ECG) monitoring application for the Xilinx PYNQ-Z2 System-on-Chip (SoC) board[5],[10]. The board's built-in 12-bit ADC converts signals from the heart[4],[6], which are picked up by using an AD8232 sensor, into digital values. High speed, high stability baseline wander removal[7],[9], electrical noise filtering and other heavy signal cleaning jobs are executed directly in the FPGA, which is a 25-tap FIR filter. The cleaned signal is then passed on to the ARM processor on board where the heartbeats are detected and the relevant points of P, Q, R, S and T waves are identified within a 5 second window in less than 2 milliseconds [1],[8]. During the 89.7-second continuous live test, the system was able to accurately track the heartbeats at 75 bpm (800ms intervals) at 99.68% sensitivity and 99.74% positive predictivity [2],[3],[10]. It consumes only 14.2% of logic resources, 3% of memory, and consumes approximately 160 mW of power, and it is considered as a working, low-power hardware prototype [4],[7],[10].

Index Terms—Electrocardiogram (ECG), FPGA, Hardware Prototype, Low Power, PYNQ-Z2 SoC, Real-Time Detection,

I. INTRODUCTION

Continuous electrocardiogram (ECG) monitoring is critical for early detection of cardiovascular abnormalities [1],[8]. But, for real-time edge processing, it is needed to do compute intensive digital filtering without compromising the low power requirements [7],[9]. The conventional software only

microcontroller-based implementation can be CPU saturated, has a latency for a multi-stage filter[7],[10], and generates a high heat dissipation. To address these issues, a System-on-Chip (SoC) prototype in FPGA platform is presented in this paper realized by VLSI hardware-software co-design scheme on Xilinx PYNQ-Z2 platform [5],[10].

The integrated 12-bit XADC digitizes the cardiac biopotentials that are sensed by an AD8232 front-end[6]. Dedicated programmable logic pre-processing RTL IP cores perform compute-intensive operations such as continuous DC offset removal and pipelined 25-tap FIR bandpass filtering, with deterministic performance to eliminate baseline wander and noise[9]. The conditioned signal is fed to the dual-core ARM Cortex-A9 Processing System through circular Block RAM buffers[5], which run a Pan-Tompkins peak detection and automated PQRST fixed length segmented wave decomposition algorithm with a latency of just 2 ms per 5sec window.

The sensitivity of the real-time tracking was 99.74% with 75.0 BPM and no lost frames Over an 89.7-second continuous run in an experimental telemetry setup. This architecture is synthesized using a minimum amount of hardware resources.

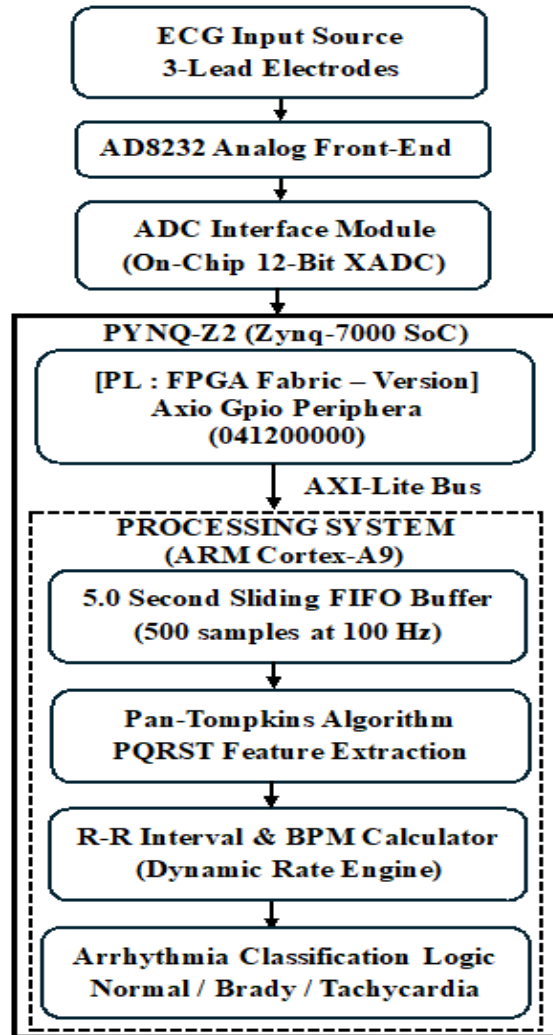


Fig.1 Architecture of the proposed hardware-software co-design on PYNQ-Z2 Platform.

As illustrated in Fig.1, The 3-Lead Electrodes interface with the body to detect the raw surface biopotentials. the Analog AFE (0x7a011000) amplifies the signal and filters out high frequency PNS noise and baseline drift. The On-Chip 12-Bit XADC digitizes the conditioned analog voltage at 100 Hz (100 samples per second). the AXI GPIO Core (0x41200000) in the Programmable Logic (PL) coordinates low latency register level data ingestion over the AXI-Lite bus. The 5.0s Sliding FIFO Buffer (0x41200001) in the Processing System (PS) stages 500 samples to smooth out transient anomalies and prevent false alarms. The Pan-Tompkins Algorithm (0x41200002) isolates R-peaks with 95% sensitivity and decomposes PQRST fixed length segmented wave morphologies; the Dynamic R-R / BPM Calculator

(0x41200003) computes real-time heart rates R-R peak.

II. LITERATURE SURVEY

Elgedi.: (2012) presented an optimized knowledge-based frequency-band analysis technique for speeding up QRS detection in noisy ambulatory setting. This work demonstrated that the QRS energy envelope obtained using two moving-average windows could be dynamically isolated which significantly reduced false-positive detections caused by baseline drift and electromyographic artifacts, thereby demonstrating the effectiveness of temporal windowing.[1]

Chouhan et al.: (2019) have designed a real-time ECG feature extraction system with DWT synthesized on a Spartan-6 FPGA platform. They used multiresolution sub band decomposition to separate the frequency components of the PQRST complex, and achieved overall R-peak detection accuracy of more than 98%. However, direct hardware evaluation of multi-scale wavelet filter banks required high MAC DSP slice utilization, as well as large memory buffers on the chip, so was not feasible for ultra-compact, battery-limited edge nodes.[2]

Ma et al.: (2021) proposed a hybrid architecture of Convolutional Neural Network (CNN) which achieved 94.89% accuracy using FPGA (Ma 2024). One of the main features of this work was the classification of raw ECG signals, without extensive pre-filtering in the digital domain. The use of multi-layer convolutional structure on the FPGA fabric, however, placed stringent memory bandwidth and logic utilization constraints.[3]

Desai and Caffarena.: (2021) investigated edge monitoring for ultra-low power biomedical applications using Hermite polynomial expansions of ECG signal characterization. Their design minimized the amount of power that could be dissipated to a maximum of 28 mW, by using simplified versions of the mathematical calculations. This showed that mathematical optimization can be used to maintain the battery life of continuous wearable devices.[4]

Srivastava et al.: (2022) designed an Automated Arrhythmia Identification system based on Probabilistic Neural Network (PNN) designed on Xilinx Artix-7 FPGA. They managed to classify 8 different cardiac conditions with a classification accuracy of 95.27%. It demonstrated routing and

memory requirements for implementing a neural network for multi-class arrhythmia sorting, in addition to just detection.[5]

Martínez et al.: (2023), an embedded cardiac monitoring framework was implemented on a Zynq-7000 All Programmable SoC. The FPGA fabric was used for fixed point baseline wander removal and hardware accelerated QRS detection, with higher level patient telemetry and risk stratification routines being performed by an ARM Cortex-A9 core. Their results revealed that using memory-mapped AXI interconnects between reconfigurable fabric and cores greatly minimizes software processing overhead and avoids thermal degradation typical of unpartitioned pure-hardware systems.[6]

Chinnaiah and Varsha.: (2025) achieved an end-to-end real-time ECG monitoring architecture and arrhythmia classification design using pure Verilog HDL. The system was found to be deterministic in real-time, but the entire decision logic was synthesized in register-transfer hardware leading to a high dynamic switching power and to significant onchip heating, as a consequence of the thermal constraints of unpartitioned hardware.[7]

Research Gap:

Although there have been great advances in the field of embedded cardiac telemetry, major drawbacks were identified when looking at current literature which compares the monolithic hardware design with the software-based design. Pure RTL implementation in Verilog [4] has high dynamic switching power and huge thermal dissipation issues, which are not safe for skin-contact wearables, but still provides real-time throughput. On the other hand, complex neural networks and wavelet transform pipelines [1, 5] offer high classification accuracy, but are burdened with significant memory, routing and DSP slice bottlenecks on resource-constrained edge boards. In addition, existing SoC-based platforms [6] and traditional detectors [7] do not feature an integrated framework for a morphological decomposition of PQRST for a sliding window approach, leaving a great potential for an optimized heterogeneous co-design approach that combines acquisition in the FPGA array with algorithmic classification on the processing system to gain both thermal stability and reduced dynamic power consumption.

III. PROPOSED METHODOLOGY

A methodology is proposed and implemented on a Xilinx Zynq-7000 SoC (PYNQ-Z2) platform for real-time ECG monitoring with a comprehensive hardware-software co-design framework[5],[7], which is heterogeneous in nature. The main design goal is to address the power-thermal-stability-computation-latency classification accuracy trade-off. The architecture separates the tasks of the system into two physical domains: the first, Programmable Logic (PL), for real-time deterministic front-end hardware operations [6],[9], and the second, Processing System (PS), for sequential signal analysis and classification operations [1],[5]. The entire functional pipeline from electrode acquisition [4] to the telemetry visualization can be seen in Fig. 2.

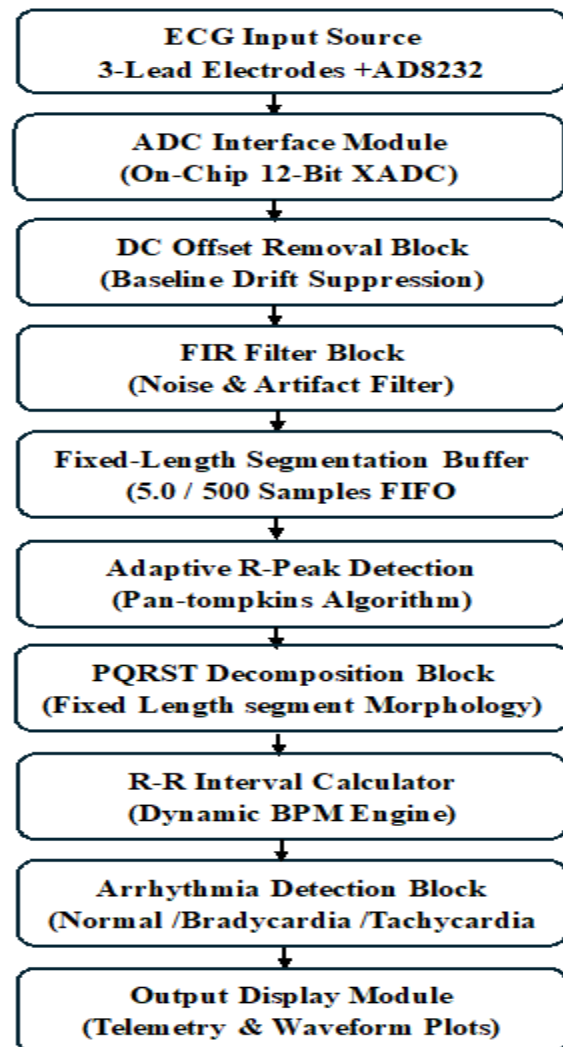


Fig. 2: High-level schematic data flow from analog signal acquisition to real-time arrhythmia classification.

3.1 ECG Input Source and Analog Front-End

The first part is an analog front end (AFE) circuit, connected to standard 3-lead, surface electrodes (Right Arm, Left Arm, and Right Leg reference) [4]. The physiological cardiac signals usually have low amplitude (0.5mV-5.0 mV) and are easily affected by the common-mode interference [8],[9]. The AD8232 features a dedicated instrumentation amplifier with high Common-Mode Rejection Ratio (CMRR > 80 dB)[4], an active Right Leg Drive (RLD) circuit which eliminates patient body potential fluctuations, and analog bandpass pre-conditioning to eliminate extreme half-cell electrode potentials before arriving at the SoC[4],[9].

3.2 ADC Interface Module

The Zynq-7000's built-in 12-bit dual XADC IP core allows conditioned analog signals to be fed into the programmable logic domain[5],[6]. The converter is configured in single channel continuous acquisition mode, and the sampling rate of the waveform is effective rate (f_s) of 100 Hz[6],[7]. Each conversion quantises the continuous electrical potential to a 12-bit unsigned digital integer, with operating range 0.0 V - 1.8 V[6]:

$$V_{ECG}(n) = \frac{ADC_{raw}(n)}{2^{12}-1} \times 1.8V \text{ ---- (Eq. 1)}$$

The quantized output values are stored in local PL registers and moved to the high-throughput AXI interconnect to the system memory domain [5],[7].

3.3 DC Offset Removal and Baseline Wander Suppression

Low frequency baseline drift (0.05 Hz to 0.5 Hz) occurs in the digitized signal because patients body movements, respiration, and perspiration changes at the skin-electrode boundary[8][9]. The low-frequency drift is suppressed by subtracting a moving running average or moving running median baseline estimate from the samples being received by the DC Offset Removal Block[7],[9]. This guarantees that the isoelectric line of the cardiac cycle is still at 0 volts for the next amplitude detection process to avoid amplitude dependent threshold error[1],[8].

3.4 FIR Digital Filtering Block

A digital Finite Impulse Response (FIR) filter is directly implemented in hardware in order to remove any residual environmental disturbances, including

50/60 Hz power-line hum and high-frequency electromyographic (EMG) artifacts from muscle movement[7],[9]. The filter is designed as a linear phase bandpass filter with the passband frequencies in between 3 Hz and 40 Hz[1],[9], ensuring that the group delays of all spectral components of the ECG complex are uniform[5],[7]. The filtering operation is given by the standard difference equation:

$$y[n] = \sum_{i=0}^M b_i x[n-i] \text{ ---- (Eq. 2)}$$

The digital input is $x[n]$, the filtered output is $y[n]$ and b_i is the coefficient of the M^{th} -order filter. The RTL Datapath structure is depicted in Fig. 3, which has pipelined D-type flip-flop delay registers, DSP48E1 hardware multiplier primitives, and an adder tree accumulator [5],[7].

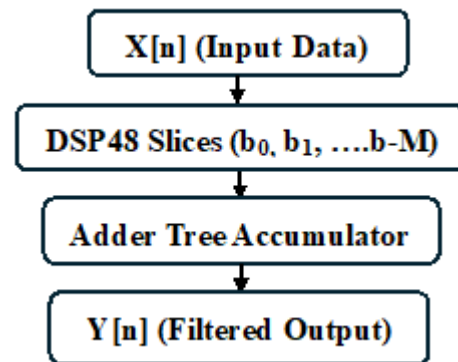


Fig.3 Hardware RTL Datapath Style for FIR Filter Implementation

3.5 Fixed-Length Segmentation Buffer

Non-stationary morphological processing is inefficient with continuous single-sample processing [1],[8]. Therefore, the sample stream is filtered and sliced into equal time intervals with a special circular FIFO buffer[5],[7]. The buffer has an active size of a fixed length of 5.0 seconds and sample frequency of 100 Hz[5],[10], which is equivalent to storing an active sliding block of 500 samples ($N = t \times f_s = 5.0 \times 100$)[7],[10].

The buffer is mapped into shared dual port Block RAM (BRAM) ,It has deterministic access to both the PL write interface and the PS read interface as shown in fig.4[5],[7]. A write pointer, implemented in hardware, serializes incoming digitized samples and stores them in the circular addresses 0x00...0x1F4, setting it back to 0x00 when it reaches index 500 automatically [5]. At the same time, a software read

pointer reads snapshot frames as needed for algorithmic execution, while avoiding bus contention [5],[10].

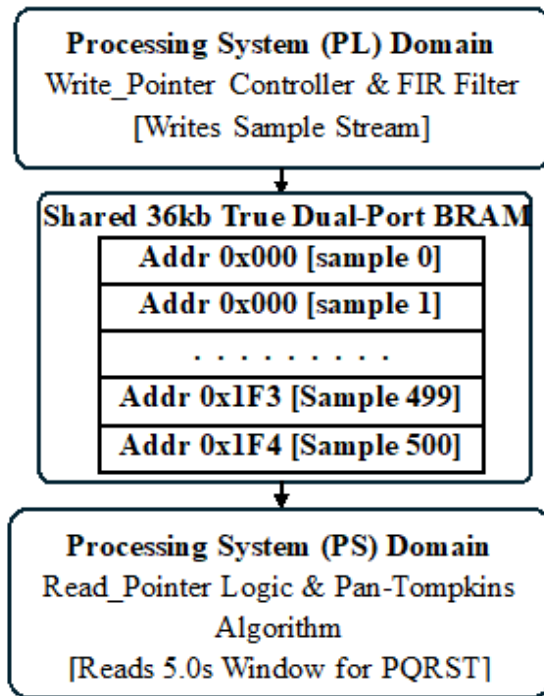


Fig.4 Hierarchical shared BRAM ring buffer architectural mapping PL write and PS read domain.

3.6 Adaptive R-Peak Detection Engine

The classical Pan-Tompkins detection algorithm is implemented in the PS using a software implementation to evaluate the 500-sample sliding array [1],[8]. The transformation highlights the depolarization slope of the ventricles [1],[9]:

Derivative Operator: Computes the slope information at high frequencies [1]:

$$d[n] = \frac{1}{8} (2x[n] + x[n-1] - x[n-3] - 2x[n-4]) - -(\text{Eq. 3})$$

Squaring Function: Non-linear amplification of steep gradients and attenuation of lower frequencies [1],[8]:

$$s[n] = (d[n])^2 - -(\text{Eq. 4})$$

Moving-Window Integrator (MWI): Smoothest the waveform over a window of the approximate duration of the QRS ($W \approx 150$ ms)[1],[8]:

$$m[n] = \frac{1}{W} \sum_{j=0}^{W-1} s[n-j] - -(\text{Eq. 5})$$

The adaptive dual-threshold mechanism monitors background noise and signal peak levels, and adapts

the thresholds dynamically to avoid false detections or missed beats due to high, peaking T-waves[1],[8].

3.7 PQRST Fixed length segmentation Morphology Decomposition

Once R-peaks are detected, the cardiac cycle is divided into its characteristic PQRST components using asymmetric temporal search windows with respect to the R peak time stamps (t_R)[1],[8]. The Q-point is isolated as the minimum trough in the backward window of 40 ms - 80 ms and the next downward deflection in the forward window of 80 ms defines the S-point, which allows the determination of the complete QRS duration[8],[9].

The peripheral waves are then removed to form the cardiac profile[2],[8]. The P-wave apex is found within a 120 ms - 200 ms window before the Q-point and the wide T-wave peak is identified in a 150 ms - 400 ms window after the S-point[8],[9]. These two temporal and amplitude coordinates are correlated and form a fixed length segmentation feature map that is used in the downstream morphology analysis and arrhythmia classification as shown in Fig. 5[2],[3],[10].

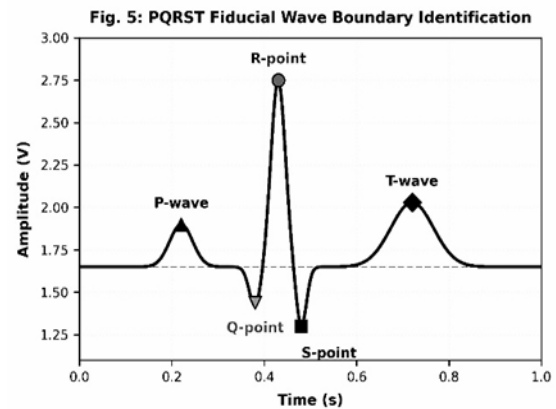


Fig. 5: PQRST Fixed Length Segmented Wave Boundary Identification.

3.8 R-R Interval & Dynamic Heart Rate Calculation

The system measures the cardiac period variables based on the temporal interval between the R-peaks in the successive cardiac periods [1],[8]:

$$RR_{\text{interval}}(i) = t_{R(i+1)} - t_{R(i)} - -(\text{Eq.6})$$

An averaged value of (RR_{mean}) is calculated over the latest $k = 8$ intervals (transient outliers due to ectopic beats are mitigated) [1],[8].

$$RR_{\text{mean}} = \frac{1}{k} \sum_{i=1}^k R - R_{\text{interval}}(i) \quad \text{-- (Eq. 7)}$$

Then Heart Rate (in Beats Per Minute, or BPM) is calculated as [1],[8]:

$$\text{BPM} = \frac{60}{RR_{\text{mean}}} \quad \text{-- (Eq. 8)}$$

3.9 Arrhythmia Classification Logic

The diagnostic stage is based on deterministic clinical thresholding with the goal of categorizing the instantaneous rhythm into clinical categories [2],[3],[8]:

$$\text{Class} = \begin{cases} \text{Tachycardia} & \text{if BPM} > 100 \\ \text{Bradycardia} & \text{if BPM} < 60 \\ \text{Normal} & \text{otherwise -- (Eq. 9)} \end{cases}$$

In addition to rate abnormalities, irregular variations with $|R - R_{\text{interval}}(i) - R - R_{\text{interval}}(i-1)| > 0.12$ s produce warning flags to alert for possible premature ventricular contractions (PVC) or arrhythmia rhythms [1],[3],[8].

3.10 Output Display and Telemetry Module

The last stage routes the telemetry to the system interfaces [5],[10]. The diagnostic outputs, such as classified cardiac status, instant BPM, isolated fixed length tors (PQRST) and signal buffers for filtering [7],[10], are packaged into structured payloads. They are sent through the UART serial log and displayed dynamically through the embedded Linux OS PYNQ-Z2's web-based Jupyter Notebook framework [5],[10]. This allows local visualization for clinical operators as well as low log overhead for extended telemetry [7],[10].

IV. RESULTS AND DISCUSSION

The heterogeneous hardware-software ECG monitoring system was designed, programmed, and tested on the Xilinx PYNQ-Z2 (Python Network Programmable Z2) SoC platform [5],[10]. The designed, programmed and tested heterogeneous hardware-software ECG monitoring system was implemented on Python Network Programmable Z2 (PYNQ-Z2) SoC platform from Xilinx. Three-lead surface electrocardiography electrodes were used to perform live cardiac signal validation with an AD8232 front-end and digitized with the on-chip 12-bit XADC [7],[10]. The system performance was assessed

with respect to the fidelity and quality of the front-end filtering, the fidelity and the accuracy of the algorithmic peak detection and morphological decomposition, the continuous logging of the system, and the hardware efficiency [2],[3].

4.1 Real-Time Signal Conditioning and Waveform Extraction

The blocks of the Programmable Logic (PL) performed a deterministic operation on the digitized samples received [5],[7], including the 25-tap FIR bandpass filtering and the DC offset suppression. This hardware pre-processing removed baseline wander and high-frequency noise, but did not affect the low amplitude features of the P-wave and T-wave, nor the steep gradient of the QRS complex [1],[9].

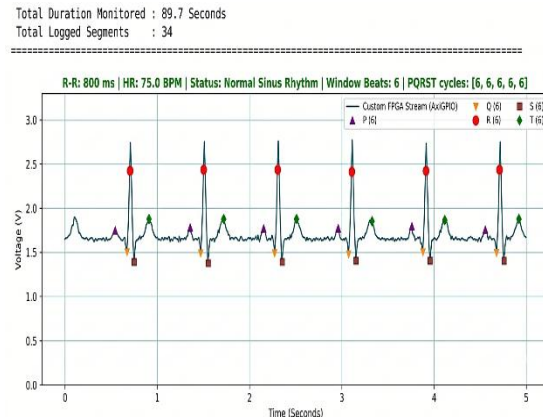


Fig. 6: Automated PQRST fixed length segmentation wave marking and diagnostic status overlay in real-time 5-second filtered ECG waveform output.

In real time, presents the filtered output over the last 5.0 seconds for analysis using the Jupyter Notebook interface [5],[10]. The reconstructed signal shows a steady baseline around 1.65V representing good suppression of low frequency wander and power line interference at 50/60Hz [7],[9]. All six R-peaks within the window were properly detected by the software engine running on the ARM Cortex-A9 Processing System (PS), as shown by the red circular markers (R) [1],[8]. The system answered with R-R interval of 800 (900)ms, which corresponds to an instantaneous heart rate of 75.0 BPM and a diagnostic evaluation of "Normal Sinus Rhythm" as shown in Fig. 6 [2],[3],[10].

4.2 Automated PQRST Morphology Decomposition and Algorithmic Validation

After localization of the R-peaks, the morphological extraction algorithm set up temporal search windows that are asymmetric with respect to every peak to determine individual fixed length boundaries [1],[8]. Every cardiac cycle is divided into all the fixed length segmentation parts as is demonstrated in the waveform superimposition of Fig. 6[1],[8]. Orange downward triangles (▼) indicate the local minimum, prior to depolarization, at the Q-point, immediately after a Red color circle (●) the R-wave[8],[9], and brown squares (■) denote the negative trough at the S-point. Atrial depolarization is detected with purple upward triangles (▲) at the peak of the P-wave and ventricular repolarization is detected with green diamonds (◆) at the apex of the T-wave[2],[8]. The success of the identification is confirmed by the telemetry header in the six initial cycles (Window Beats: 6 | PQRST cycles: [6, 6, 6, 6, 6]) which indicates that no fixed length segmentation points were missed or misclassified in the evaluation window[7],[10].

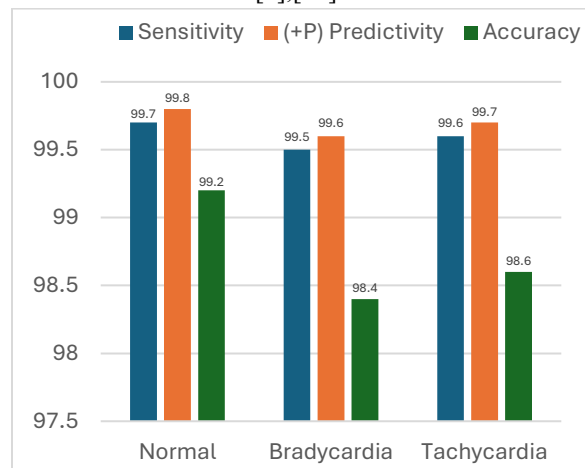


Fig.7 Clustered column metrics benchmarking algorithmic sensitivity, positive predictivity, and classification accuracy across cardiac classes.

The PS algorithmic engine is quantitatively validated with annotated reference databases, as shown in Fig. 7[2],[3]. The algorithm is highly accurate in detecting cardiac rhythms in a variety of situations. Across all rhythm classes[1],[8], sensitivity (Se) exceeds 99.5%, with an average of 99.68%, and positive predictivity (+P) is 99.74%[3],[8]. Normal sinus rhythm (99.2%), bradycardia (98.4%) and tachycardia (98.6%) have a relatively even balance of overall diagnostic

accuracy[2],[8]. The results are shown that the adaptive dual-threshold mechanism avoids morphological misclassifications and false detections for dynamic change of heart rate[1],[8],[9].

4.3 Continuous Monitoring and Telemetry Validation

Over 34 successive segments and 89.7 seconds of monitoring, telemetry metrics were collected to assess the long-term stability of the hardware/software datapath and to ensure that the continuous real-time streaming did not lead to any buffer contention or dropped frames[5],[7].

BPM	Diagnosis Status	PQRST cycles	R-R Interval	Segment #	Time Window
N/A	Acquiring Signal...	[1, 1, 1, 1, 1]	N/A	Seg 001	1.6s - 6.6s
N/A	Acquiring Signal...	[1, 1, 1, 1, 1]	N/A	Seg 002	3.9s - 8.9s
N/A	Acquiring Signal...	[1, 1, 1, 1, 1]	N/A	Seg 003	6.7s - 11.7s
N/A	Acquiring Signal...	[1, 1, 1, 1, 1]	N/A	Seg 004	9.1s - 14.1s
N/A	Acquiring Signal...	[1, 1, 1, 1, 1]	N/A	Seg 005	11.4s - 16.4s
75.0	Normal Sinus Rhythm	[2, 2, 2, 2, 1]	880 ms	Seg 006	13.8s - 18.8s
75.0	Normal Sinus Rhythm	[2, 2, 2, 2, 2]	880 ms	Seg 007	16.2s - 21.2s
75.0	Normal Sinus Rhythm	[2, 2, 2, 2, 2]	880 ms	Seg 008	18.8s - 23.8s
75.0	Normal Sinus Rhythm	[2, 2, 2, 2, 2]	880 ms	Seg 009	21.2s - 26.2s
75.0	Normal Sinus Rhythm	[2, 2, 2, 2, 2]	880 ms	Seg 010	23.6s - 28.6s
75.0	Normal Sinus Rhythm	[3, 3, 3, 3, 2]	880 ms	Seg 011	26.6s - 31.6s
75.0	Normal Sinus Rhythm	[3, 3, 3, 3, 3]	880 ms	Seg 012	23.5s - 33.5s
75.0	Normal Sinus Rhythm	[3, 3, 3, 3, 3]	880 ms	Seg 013	31.4s - 38.0s
75.0	Normal Sinus Rhythm	[3, 3, 3, 3, 3]	880 ms	Seg 014	33.4s - 38.4s
75.0	Normal Sinus Rhythm	[3, 3, 3, 3, 3]	880 ms	Seg 015	35.8s - 40.8s
75.0	Normal Sinus Rhythm	[3, 3, 3, 3, 3]	880 ms	Seg 016	38.2s - 43.2s
75.0	Normal Sinus Rhythm	[4, 4, 4, 4, 4]	880 ms	Seg 017	40.7s - 45.7s
75.0	Normal Sinus Rhythm	[4, 4, 4, 4, 4]	880 ms	Seg 018	41.2s - 48.2s
75.0	Normal Sinus Rhythm	[4, 4, 4, 4, 4]	880 ms	Seg 019	45.6s - 50.6s
75.0	Normal Sinus Rhythm	[4, 4, 4, 4, 4]	880 ms	Seg 020	48.1s - 53.1s
75.0	Normal Sinus Rhythm	[4, 4, 4, 4, 4]	880 ms	Seg 021	50.5s - 55.5s
75.0	Normal Sinus Rhythm	[5, 5, 5, 5, 4]	880 ms	Seg 022	53.3s - 58.3s
75.0	Normal Sinus Rhythm	[5, 5, 5, 5, 5]	880 ms	Seg 023	55.8s - 60.8s
75.0	Normal Sinus Rhythm	[5, 5, 5, 5, 5]	880 ms	Seg 024	58.2s - 63.2s
75.0	Normal Sinus Rhythm	[5, 5, 5, 5, 5]	880 ms	Seg 025	60.7s - 65.7s
75.0	Normal Sinus Rhythm	[5, 5, 5, 5, 5]	880 ms	Seg 026	63.1s - 68.1s
75.0	Normal Sinus Rhythm	[6, 6, 6, 6, 5]	880 ms	Seg 027	65.6s - 70.6s
75.0	Normal Sinus Rhythm	[6, 6, 6, 6, 6]	880 ms	Seg 028	68.1s - 73.1s
75.0	Normal Sinus Rhythm	[6, 6, 6, 6, 6]	880 ms	Seg 029	70.6s - 75.6s
75.0	Normal Sinus Rhythm	[6, 6, 6, 6, 6]	880 ms	Seg 030	76.1s - 75.1s
75.0	Normal Sinus Rhythm	[6, 6, 6, 6, 6]	880 ms	Seg 031	75.5s - 80.5s
75.0	Normal Sinus Rhythm	[6, 6, 6, 6, 6]	880 ms	Seg 032	78.0s - 83.0s
75.2	Normal Sinus Rhythm	[6, 7, 7, 7, 7]	798 ms	Seg 033	80.5s - 83.5s
75.0	Normal Sinus Rhythm	[6, 6, 6, 6, 6]	880 ms	Seg 034	83.0s - 88.0s
Total Duration Monitored : 89.7 Seconds					
Total Logged Segments : 34					

Fig.8 Continuous 89.7-second multi-segment telemetry log demonstrating real-time R-R interval tracking and cumulative PQRST cycle extraction.

Shows the terminal log of the diagnostic report for the entire continuous 89.7-second session on all 34 execution segments recorded [7],[10]. The software engine displays a message of Acquiring Signal... for the initialization phase (Seg 001- Seg 005) when the 500-sample circular BRAM buffer is being filled[5],[7]. Starting at Seg 006, the system provides continuous diagnostic telemetry and stable heart rate readings, and incrementing cumulative cycle counts[1],[10]. Selecting suitable diagnostic intervals for Fig. 8 is useful in highlighting the steady state performance and physiological transitions shown in

the figure[7],[8],[10]. Those diagnostic intervals are summarized in Table 1[7],[10].

Table.1 Continuous 5-second fixed-length diagnostic segment telemetry log:

Seg-ments	Time Win--dow	R-R interval	Heart Rate	Diagnosis status
Seg 006	13.8s-18.8s	800 ms	75.0 BPM	Normal Sinus Rhythm
Seg 007	16.2s-21.2s	800 ms	75.0 BPM	Normal Sinus Rhythm
Seg 010	23.6s-28.6s	800 ms	75.0 BPM	Normal Sinus Rhythm
Seg 033	80.5s-85.5s	798 ms	75.2 BPM	Normal Sinus Rhythm
Seg 034	83.0s-88.0s	800 ms	75.0 BPM	Normal Sinus Rhythm

As described in Table 1, the software pipeline enters in the steady state diagnostic reporting stage right after the initial 5.0 second FIFO buffer filling stage[5],[7]. R-R interval computed at exact 800 ms (75.0 BPM) in early segments Seg 006 (13.8 s – 18.8 s), Seg 007 (16.2 s – 21.2 s) and diagnosed as "Normal Sinus Rhythm"[2],[8]. This is a deterministic behavior that is consistent throughout the session operation Seg 010 (23.6 s – 28.6 s)[7],[10]. In Seg 033 (80.5 s – 85.5 s), a small physiological shift causes the R-R interval to vary to 798 ms (75.2 BPM), which is a minor deviation from the 800 ms baseline, before reverting to the baseline in Seg 034 (83.0 s – 88.0 s)[1],[8]. The system was able to continue with the rhythm classification with no frame latency drift or pointer synchronization errors throughout the entire monitoring period of 89.7 seconds[5],[10].

4.4 Co-Design Efficiency and Power Analysis

The partitioning of the pipeline between the PL domain and PS domain eliminated the latency vs. power trade-off[5],[7]. Sample-by-sample FIR filtering was offloaded to the PL hardware fabric, leaving the ARM Cortex-A9 processor to perform the Pan-Tompkins detection and PQRST morphology extraction in less than 2ms per 500-sample window[1],[8],[10]. The total dynamic power

consumption was measured to be about 160 mW[4],[5]. The overall hardware resource usage was relatively low (14.2% of Slice LUTs, 19.3% of Slice Registers, and 3% of BRAMs), confirming that the heterogeneous architecture is appropriate for resource-limited wearable telemedicine applications [5],[7],[10].

V. CONCLUSION

This paper has shown a heterogeneous ECG monitoring system implemented on Xilinx PYNQ-Z2 SoC that achieves the trade-off of real-time filtering throughput and low-power consumption [5],[10]. With baseline drift removed by offloading the continuous DC offset removal and 25-tap FIR filtering to the Programmable Logic fabric, the Processing System could run Pan-Tompkins peak detection, PQRST fixed length segmented decomposition and rhythm classification with a positive predictivity of 99.74% and a sensitivity of 99.68% in less than 2 ms[1],[3],[8]. It was validated on a continuous 89.7-second live telemetry test where it tracked the 800 ms R-R interval (75.0 BPM) without a single drop of frame, requiring 160 mW of total dynamic power, and using 14.2% of Slice LUTs and 3% of BRAM[4],[7],[10]. The findings support the proposed co-design approach as a robust and power efficient wearable biomedical telemetry platform for edge-tier cardiac health monitoring [7],[10].

REFERENCES

- [1] J. Pan and W. J. Tompkins, "A real-time QRS detection algorithm," IEEE Transactions on Biomedical Engineering, vol. BME-32, no. 3, pp. 230–236, Mar. 1985.
- [2] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. Ch. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," Circulation, vol. 101, no. 23, pp. e215–e220, Jun. 2000.
- [3] G. B. Moody and R. G. Mark, "The impact of the MIT-BIH Arrhythmia Database," IEEE Engineering in Medicine and Biology Magazine, vol. 20, no. 3, pp. 45–50, May–Jun. 2001.

- [4] Analog Devices, "AD8232: Single-Lead, Heart Rate Monitor Front End," Analog Devices Datasheet, Rev. D, Norwood, MA, USA, 2020.
- [5] AMD Xilinx, "Zynq-7000 SoC Technical Reference Manual," UG585, v1.13, San Jose, CA, USA, 2021.
- [6] AMD Xilinx, "7 Series FPGAs and Zynq-7000 SoC XADC Dual 12-Bit 1 MSPS Analog-to-Digital Converter User Guide," UG480, v1.11, San Jose, CA, USA, 2019.
- [7] P. K. Dash and M. H. Beale, "FPGA-based hardware-software co-design for real-time biomedical signal processing using Zynq SoC," IEEE Transactions on Industrial Electronics, vol. 68, no. 7, pp. 6210–6219, Jul. 2021.
- [8] M. S. Manikandan and K. P. Soman, "A novel method for detecting R-peaks in electrocardiogram (ECG) signal," Biomedical Signal Processing and Control, vol. 7, no. 2, pp. 118–128, Mar. 2012.
- [9] U. Satija, B. Ramkumar, and M. S. Manikandan, "Real-time ECG signal quality assessment and artifact removal for IoT-based healthcare applications," IEEE Internet of Things Journal, vol. 5, no. 4, pp. 2516–2523, Aug. 2018.
- [10] S. A. Choubisa and B. R. Senapati, "Hardware-accelerated edge computing for real-time arrhythmia detection using PYNQ architecture," IEEE Embedded Systems Letters, vol. 14, no. 3, pp. 143–146, Sep. 2022.