Peak Detection of PPG Signals Using Fixed-Point Digital Filters Implemented in VHDL

H.K.Y. Niroshana
Computer Science and Engineering
University of Moratuwa
Moratuwa, Sri Lanka
yasantha.21@cse.mrt.ac.lk

W.M. Wimalasiri
Computer Science and Engineering
University of Moratuwa
Moratuwa, Sri Lanka
wimalasiri.21@cse.mrt.ac.lk

C. Hettiarachchi
Computer Science and Engineering
University of Moratuwa
Moratuwa, Sri Lanka
chathuranga@cse.mrt.ac.lk

***Keywords*—Fixed-point arithmetic, FPGA, VHDL, Digital filters, Real-time signal processing**

I. Introduction

Photoplethysmography (PPG) is a non-invasive optical method widely employed in biomedical signal processing to estimate vital parameters such as heart rate and blood oxygen saturation. Accurate and real-time peak detection is essential for reliable feature extraction in PPG analysis. Field-Programmable Gate Arrays (FPGA), with their inherent parallelism, low latency, and energy efficiency, provide an effective hardware platform for implementing such algorithms in embedded healthcare systems. Integrating PPG signal processing with FPGA-based architectures thus enables high-performance, real-time physiological monitoring for next-generation wearable and portable medical devices.The VHDL implementation and simulation scripts are available at GitHub repository.

II. Literature Review

Numerous algorithms have been proposed for PPG peak detection, each balancing complexity, robustness, and suitability for hardware realization. Early approaches such as zero-crossing and local maxima/minima (LCM) detection offered low computational cost but were highly sensitive to motion artifacts and baseline drift [1]. Adaptive thresholding and morphological filtering methods improved noise robustness by adjusting dynamically to signal variations [2], [3].Wavelet-based techniques further enhanced performance in noisy conditions but introduced higher computational complexity [4].

FPGA-based implementations have gained increasing popularity due to their deterministic timing, inherent parallel processing capabilities, and high degree of reconfigurability. A range of approaches has been explored, including the detection of R-peaks in ECG signals [5], the processing of PPG signals, and FPGA-based heart rate estimation [6]. Despite this progress, there remains a notable lack of open-source VHDL designs dedicated specifically to PPG signal processing. To address this gap, this work presents a resource-efficient fixed-point VHDL implementation for PPG peak detection in real-time monitoring systems.

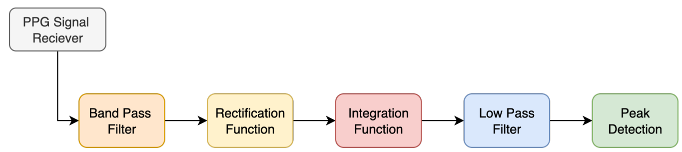

Fig. 1. Overall PPG Signal Processing Pipeline

III. Methods and Materials

A. Signal Acquisition and Simulation Dataset

The PPG signals used for simulation were obtained from the MIMIC PERform AF dataset [7], [8], recorded using bedside monitors at a sampling rate of 125 Hz.

B. Signal Preprocessing

The preprocessing pipeline (Fig. 1) enhances PPG signal quality by removing baseline drift, high-frequency noise, and motion artifacts. It comprises four main stages: bandpass filtering, rectification, integration, and low-pass filtering.The filter coefficients were derived from prior work [9].

1) Bandpass Filtering: A 4th-order Butterworth bandpass filter was employed to isolate the cardiac frequency range and suppress noise.

$$H(z) = \frac{0.582 - 1.165z^{-2} + 0.582z^{-4}}{1 - 0.687z^{-1} - 0.815z^{-2} + 0.193z^{-3} + 0.347z^{-4}} \tag{1}$$

2) Rectification (Absolute Value): The filtered signal was rectified to obtain positive amplitudes.

$$y(n) = |x(n)| \tag{2}$$

3) Integration (Moving Average): The filtered signal was passed through a moving-window integrator of 30 samples of width to smooth the waveform and preserve a single peak per cardiac cycle.

$$y(n) = \frac{1}{30}\sum_{k=1}^{30} x(n-k) \tag{3}$$

ERU Symposium 2025 - DOI 10.31705/ERU.2025.38

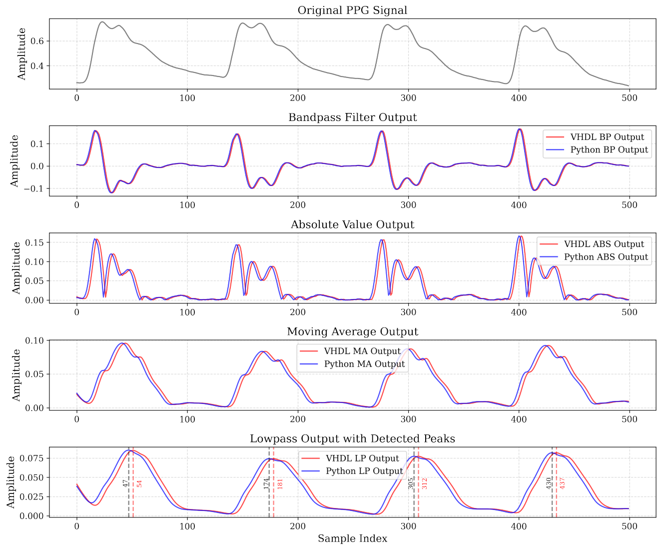

Fig. 2. Pipeline validation using PPG data

4) Low-Pass Filtering: A first order butterworth low-pass filter was applied for additional smoothing.

$$H(z) = \frac{0.059 + 0.059z^{-1}}{1 - 0.881z^{-1}} \tag{4}$$

5) Peak Detection: Peaks were detected by identifying local maxima where,

$$x(n) > x(n-1) \text{ and } x(n) > x(n+1) \tag{5}$$

Here, $x(n)$ is the input PPG signal, $y(n)$ the stage output, $H(z)$ the filter transfer function with coefficients b_i, a_i, and z^{-k} a unit delay of k samples.

C. Fixed-Point Quantization and Normalization

For FPGA implementation, all coefficients and data paths were represented in fixed-point format to achieve a balance between hardware efficiency and numerical precision. A real-valued coefficient b was quantized as

$$b_{Qn.m} = \text{round}(b \cdot 2^m) \tag{6}$$

During the multiply–accumulate (MAC) operation, intermediate values may exceed the target word length.Therefore, wider accumulators were used to prevent overflow. The final result was normalized to the desired format by right-shifting and truncation:

$$y_{Qn.m} = \text{truncate}\left(\frac{y_{\text{acc}}}{2^m}\right) \tag{7}$$

IV. Results and Discussions

The proposed VHDL-based PPG peak detection pipeline was synthesized for a Xilinx Basys 3 FPGA. The design occupies only 514 LUTs (2.47%) and 367 registers (0.88%), with no Block RAM usage and only 16 DSP48 slices (17.78%), demonstrating a compact and computation-efficient implementation suitable for low-power wearable and embedded systems.

TABLE I.
RMSE between VHDL (fixed-point) and Python (floating-point) implementations

Processing Stage	RMSE
Bandpass filter	0.015876
Absolute value	0.019651
Moving average	0.006447
Lowpass filter	0.005338

The fixed-point VHDL-based design was validated against its Python floating-point counterpart (Fig. 2). As shown in Table I, all processing stages exhibited RMSE values below 0.02, confirming minimal quantization error. The bandpass and low-pass filters effectively suppressed baseline wander and high-frequency noise, producing a stable waveform with a clear dominant peak per cardiac cycle (Fig. 2).A consistent offset of approximately 7 samples (0.056s), attributable to cumulative filter group delay, was observed but does not affect peak-to-peak intervals, thus preserving heart rate accuracy.

These results confirm that the synthesized hardware pipeline delivers numerically robust and stable peak detection while significantly reducing computational complexity compared to software based implementations.

V. Conclusion

This work presents an efficient FPGA-based implementation of a PPG peak detection pipeline for real-time heart rate estimation. The use of fixed-point digital filters enables accurate signal processing with minimal computational overhead, making the design well-suited for resource-constrained wearable biomedical applications.

References

[1] F. Scholkmann, J. Boss, and M. Wolf, "An Efficient Algorithm for Automatic Peak Detection in Noisy Periodic and Quasi-Periodic Signals," Algorithms, vol. 5, no. 4, pp. 588–603, Dec. 2012, doi: 10.3390/a5040588.

[2] M. Elgendi, "Fast QRS Detection with an Optimized Knowledge-Based Method: Evaluation on 11 Standard ECG Databases," PLoS One, vol. 8, no. 9, p. e73557, Sept. 2013, doi: 10.1371/journal.pone.0073557.

[3] P. H. Charlton et al., "Detecting beats in the photoplethysmogram: benchmarking open-source algorithms," Physiol. Meas., vol. 43, no. 8, p. 085007, Aug. 2022, doi: 10.1088/1361-6579/ac826d.

[4] Y. Yang, L. Xu, L. Zhao, Y. Chen, and M. Wang, "PPG signals processing using wavelet transform and adaptive filter," in Proceedings of the 32nd Chinese Control Conference, July 2013, pp. 3623–3627. Accessed: Oct. 13, 2025. [Online]. Available: https://ieeexplore.ieee.org/document/6640050

[5] A. M. Colak, T. Manabe, Y. Shibata, and F. Kurokawa, "Peak Detection Implementation for Real-Time Signal Analysis Based on FPGA," Circuits and Systems, vol. 9, no. 10, pp. 148–167, Oct. 2018, doi: 10.4236/cs.2018.910016.

[6] A. Chowdhury, D. Das, A. B. M. Eldaly, R. C. C. Cheung, and M. H. Chowdhury, "Photoplethysmogram-based heart rate and blood pressure estimation with hypertension classification," IPEM-Translation, vol. 9, p. 100024, Mar. 2024, doi: 10.1016/j.ipemt.2024.100024.

[7] Peter H Charlton, "MIMIC PERform Datasets." Zenodo, July 15, 2025. doi: 10.5281/ZENODO.6807402.

[8] A. Johnson, T. Pollard, and R. Mark, "MIMIC-III Clinical Database." PhysioNet, 2015. doi: 10.13026/C2XW26.

[9] K. Meddah, M. Kedir-Talha, and H. Zairi, "FPGA-based system for heart rate calculation based on PPG signal," in 2017 5th International Conference on Electrical Engineering - Boumerdes (ICEE-B), Oct. 2017, pp. 1–5. doi: 10.1109/ICEE-B.2017.8192157.