Development Report

Real-Time ECG and PPG Signal Processing and Monitoring System with STM32 and LabVIEW

Developer: Mohsen Anvari
Date: 28 August 2025

Introduction

Biomedical signal acquisition and processing play a crucial role in modern healthcare, enabling the monitoring of vital parameters such as heart activity and blood oxygenation. In particular, the simultaneous recording of ECG (Electrocardiogram) and PPG (Photoplethysmogram) signals are widely used in advanced applications such as blood pressure estimation from Pulse Transit Time (PTT), heart rate variability analysis, and cardiovascular health monitoring.

This project focuses on the development of a dual-channel system for real-time acquisition and processing of ECG and PPG signals using an STM32 microcontroller. The processed data is transmitted over UART and visualized on a PC through a custom LabVIEW interface.

The work combines embedded systems design, biomedical front-end integration, digital signal processing, and software development into a unified platform, demonstrating both theoretical knowledge and practical implementation.

Progress Steps

1. Task Understanding

Objective: Acquire ECG and PPG signals using biomedical front-end ICs, process them in real time with STM32, and visualize them on PC software.

2. Research & Hardware Setup

- Studied AD8232 ECG front-end: analog output, electrode-based input.
- Verified ECG hardware with oscilloscope.
- Used STM32F446RE Nucleo board as the main controller.
- Integrated MAX30102 Pulse Oximeter for PPG acquisition via I^2C.

3. Repository Setup

- Created a dedicated GitHub repository for code version control and documentation.

4. STM32 Configuration & Data Acquisition

- Configured ADC + Timer + USART (DMA base) for ECG sampling.
- Initial streaming to Tera Term showed delays.

- Solution for delay problem: Optimized acquisition by using Timer-triggered ADC without DMA (remove buffering), improving real-time response.

5. ECG Signal Acquisition & Validation

- Collected ECG using electrodes → observed high noise levels.
- Exported raw data to Python, applied digital filtering, clear ECG waveform obtained (Figure 1).
- Verified performance using an ECG signal generator, confirming correct AD8232 and STM32 operation.

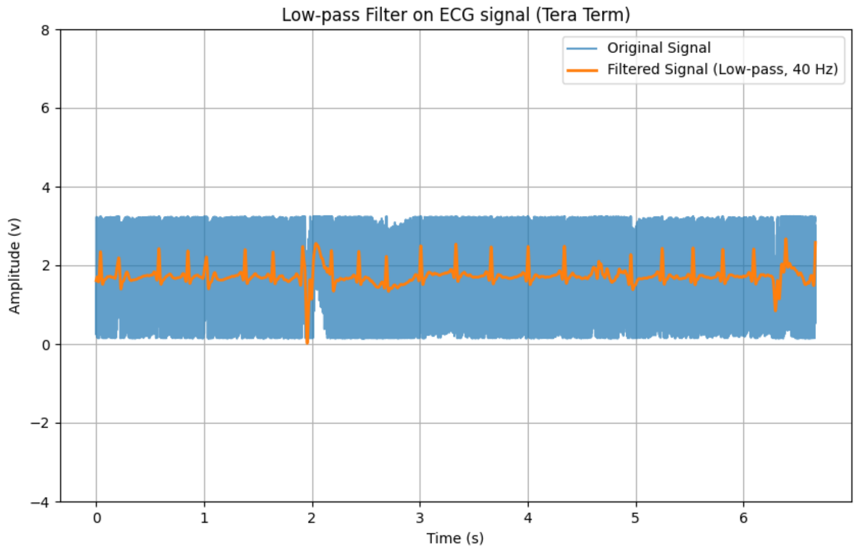

Figure 1: Raw ECG signal acquired via STM32 ADC and UART (Tera Term), exported to Python and digitally filtered to obtain a clear waveform, confirming correct data acquisition despite initial noise.

6. PC Software Development (LabVIEW)

- Developed a LabVIEW GUI to receive UART data and plot signals.
- Challenges resolved:
  - String-to-numeric data conversion.
  - Correct frame length detection.
  - Decimal formatting inconsistencies (comma vs. dot).
- Achieved stable real-time plotting of ECG at fs = 1600 Hz.

7. Embedded Signal Processing

- Implemented real time Moving Average Filter (LPF) on STM32.
- Window length = 40 → cutoff frequency ≈ 1600/40 = 40 Hz.

8. Extension to PPG Signal & Updating UART

- Configured I^2C peripheral for PPG acquisition from MAX30102.
- Designed multiplexed UART streaming protocol:

- E,0.00 → P,0.00 → E,0.00 → P,0.00 …
- Updated LabVIEW GUI to parse and plot ECG & PPG separately.
- Achieved a final synchronized sampling rate of 600 Hz for dual-channel acquisition, ensuring stable data flow without delay.

9. Dual-Channel Filtering Challenges

- Initial bug: same buffer shared between ECG & PPG → overlapping signals.
- Fixed by allocating independent buffers for each channel.
- Redesigned filter to allow user-defined cutoff frequencies.

10. Advanced Filtering

- Implemented High-Pass Filter (HPF) for PPG baseline drift removal.
- Realized PPG morphology issues, identified the cause, and rechecked and redesigned the HPF to resolve them (Figure 2).
- Tuned cutoff frequency ≈ 1 Hz to preserve signal morphology.

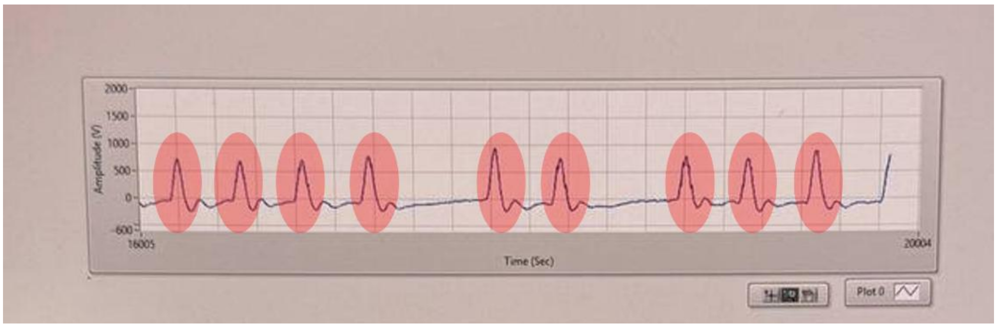

Figure 2: PPG signal after initial high-pass filtering shows morphological distortions.

Overall Workflow of ECG & PPG Acquisition System

The system integrates biomedical front-end sensors, an STM32 microcontroller, and PC software into a unified platform for real-time biosignal monitoring.

1. **Signal Acquisition**
   - **ECG:** Acquired from electrodes using the AD8232 analog front-end, providing a conditioned analog waveform.
   - **PPG:** Captured with the MAX30102 Pulse Oximeter via the I^2C interface.
2. **Microcontroller Processing (STM32F446RE Nucleo)**
   - **ECG Channel:** Sampled using the ADC with a timer trigger for accurate sampling.
   - **PPG Channel:** Read digitally from MAX30102 via I^2C.
   - **Filtering:**

   - **Moving Average (LPF)** removes high-frequency noise.
   - **High-Pass Filter (HPF)** removes baseline drift, especially in PPG.
  - **Data Packaging:** Both signals are combined into a structured UART protocol (E,0.00 → P,0.00 → ...) at 600 Hz synchronized sampling rate.

3. **Data Transmission**
  - The STM32 streams processed signals over UART to the PC without delay.
4. **PC Software (LabVIEW GUI)**
  - **UART Interface:** Receives and parses incoming frames.
  - **Real-Time Visualization:** Displays both raw and filtered ECG & PPG signals.
  - **Applications:** Enables synchronized analysis of ECG and PPG for features such as Pulse Transit Time (PTT), which can be used for blood pressure estimation.

This workflow ensures that noisy biomedical signals acquired from the body are digitally cleaned in real time and then visualized on the PC, providing a reliable platform for biomedical research and health monitoring applications.

Block Diagram of the ECG & PPG Acquisition System

This diagram shows the complete signal flow: ECG is acquired using the AD8232 analog front-end and PPG using the MAX30102 Pulse Oximeter. Both signals are processed by the STM32F446RE Nucleo board, where real-time digital filtering (LPF & HPF) is applied. The data is then transmitted via UART to the PC, where the LabVIEW GUI separates and plots raw and filtered ECG & PPG signals in real time, enabling advanced applications such as Pulse Transit Time (PTT)-based blood pressure estimation (Figure 3).

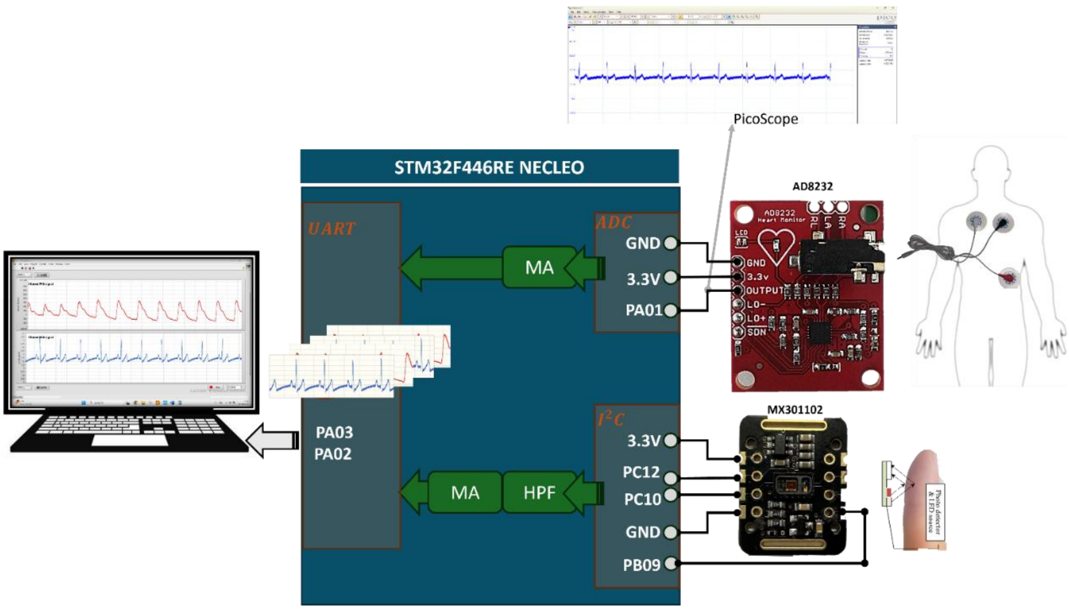

Figure 3: ECG & PPG Acquisition System Block Diagram

Real-Time Display of Raw and Filtered ECG & PPG Signals

The LabVIEW-based PC software receives data over UART, separates ECG and PPG channels, and displays both raw (unfiltered) and digitally filtered signals in real time (Figure 4 and Figure 5). This allows direct comparison of noise-prone input signals with the improved outputs after STM32-based filtering (LPF & HPF), clearly demonstrating the effectiveness of the embedded signal processing.

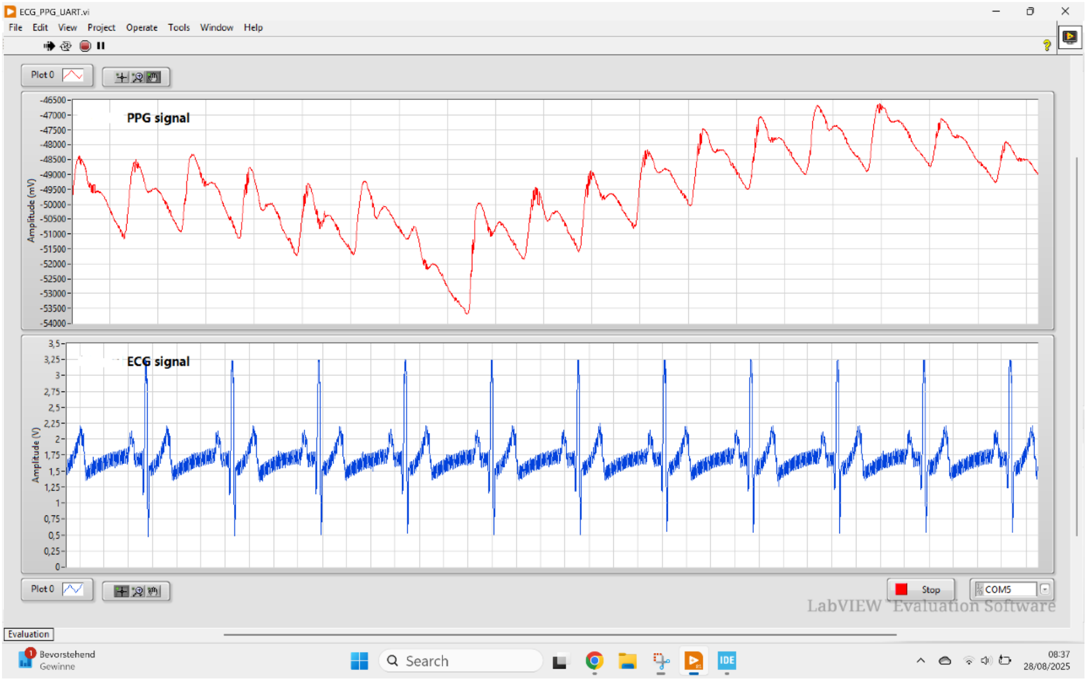

Figure 4: raw (unfiltered) PPG and ECG signal

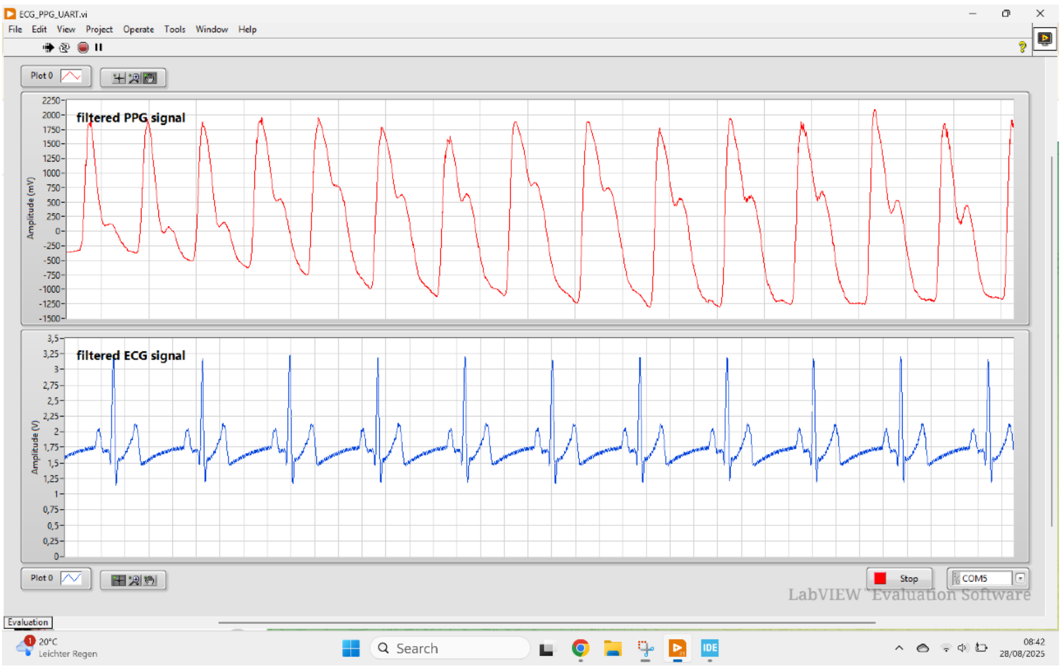

Figure 5: filtered PPG and ECG signal

Results

The developed system successfully achieved real-time acquisition, processing, and visualization of ECG and PPG signals.

- **ECG Results:** Clear P-QRS-T complexes obtained after filtering.
- **PPG Results:** Stable morphology from MAX30102 with reduced baseline drift.
- **Performance:** Final sampling rate of 600 Hz achieved with no delay, ensuring synchronized ECG & PPG signals — an essential requirement for applications such as blood pressure estimation from pulse transit time (PTT).
- **Software Outcome:** LabVIEW GUI provided reliable real-time monitoring with independent display of raw and filtered signals.

Skills demonstrated

- Embedded C programming (STM32 HAL, peripherals).
- Digital signal processing (MA, LPF, HPF).
- Biomedical instrumentation (ECG/PPG acquisition, noise reduction).
- Debugging with oscilloscope, Python, LabVIEW.
- Software engineering (GitHub repo, modular firmware design).

Future Work

To further extend the capabilities of the developed system, the following improvements are planned:

1. PC Software Enhancement (LabVIEW)

- Update the LabVIEW-based GUI to include a signal recording feature, allowing ECG and PPG data to be stored for offline analysis and long-term monitoring.

2. Frequency-Domain Analysis in Python

- Implement offline analysis of recorded signals in the frequency domain to evaluate the impact of digital filters.
- Verify that the chosen cutoff frequencies (LPF & HPF) effectively remove noise and baseline drift without losing essential physiological components of ECG and PPG signals.

3. Feature Extraction and Integration with Python

- Develop Python scripts to automatically detect R-waves in the ECG and identify systolic peaks in the PPG signal for future Pulse Transit Time (PTT) analysis.
- Integrate the LabVIEW-based acquisition software with Python processing modules

4. Enhanced Signal Visualization on LCD

- Integrate real-time display of ECG, PPG, and other biomedical signals on an ILI9341 LCD