A Wireless Biosignal Measurement System using a SoC FPGA and Bluetooth Low Energy

Ricardo Joaquinito
Electronic System Design and Automation
INESC-ID/IST/ULisbon
Lisbon, Portugal
ricardo.joaquinito@tecnico.ulisboa.pt

Helena Sarmento
Electronic System Design and Automation
INESC-ID/IST/ULisbon
Lisbon, Portugal
helena.sarmento@tecnico.ulisboa.pt

***Abstract*—This paper presents the development of a prototype for a wireless biosignal measurement system, which makes use of a System-on-Chip FPGA, with an embedded ARM processor, and the Bluetooth Low Energy standard for wireless data transmission to a smartphone. The body temperature and heart rate are monitored using a steel-head thermistor and an ECG acquisition module, respectively. The ARM processor runs a Real-Time Operating System. A part of the algorithm that extracts the heart rate runs on custom hardware implemented in the FPGA fabric. The prototype successfully extracts the vital signs and sends updated values to a smartphone every second.**

Keywords—biosignals; ECG; SoC FPGA; Zynq; Bluetooth Low Energy

I. Introduction

Human being's physiological parameters are important indicators of health. Recently, the measurement of these parameters has gained notoriety in the consumer electronics market, primarily through wearable devices like fitness bands and smart-watches. Medical-grade wearables that can be worn by patients at home are now in demand, but, in order to obtain the medical grade, such devices must go through a rigorous regulatory process which can take from 6 months to 2 years [1].

The ease of reconfiguration and versatility of Systems-on-Chip with Field Programmable Gate Array (SoC FPGA) have been noted by the medical industry's developers as a good solution for this kind of equipment [1]. They are ideal for developing embedded systems if processing with the aid of custom hardware and regular system upgrades are desired.

The electrocardiographic (ECG) signal is one of the physiological signals that can be monitored by wearable devices [2]. It can be used to keep track of the person's heart rate and detect cardiac problems and other health issues. A highly reliable algorithm for these tasks can be computationally demanding and has real-time constraints, so hardware solutions in FPGAs have been proposed [3][4].

Regarding wearable devices, consideration must be taken on their energy consumption. New low energy wireless standards have emerged in recent years aimed at this kind of devices. Bluetooth Low Energy (BLE), an energy efficient variation of the Bluetooth technology, is now the most widely used open standard in that category, with predictions stating that, by 2018, 96% of all smartphones will support BLE [5].

This work explores the use of a SoC FPGA device with a BLE connection for the development of a wireless biosignal measurement system, which can be the basis for a wearable health-monitoring device. The heart rate is calculated from the person's ECG signal and the body temperature is also measured. The heart rate detection algorithm is partially running on custom hardware implemented in the FPGA fabric.

II. System Overview

Since the proposed system can be achieved with different combinations of development boards and sensors in the market, the general architecture of this system is introduced first with a neutral view of the components, followed by a listing and short description of the components used in the implemented prototype.

A. General architecture

The general architecture of the implemented system is presented on Figure 1. Everything represented besides the smartphone composes the sensing device.

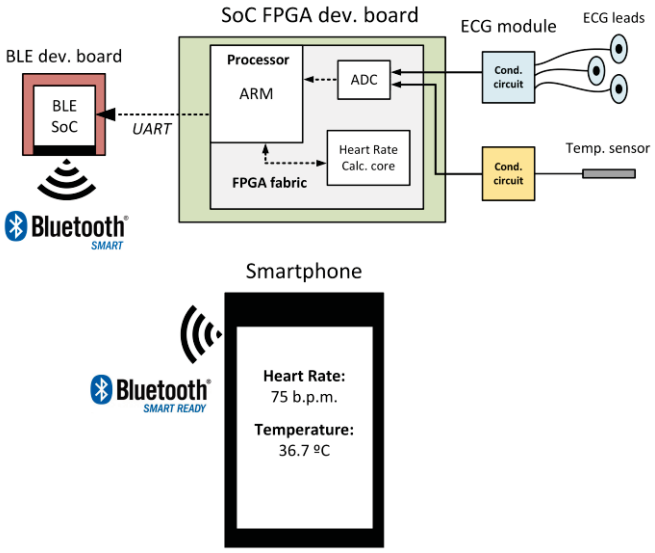

Fig. 1. General architecture of the proposed wireless biosignal measurement system.

The ECG signal is extracted from a 2 or 3-electrode configuration and is preprocessed before being converted to digital format. The conditioning circuit filters the signal in order to eliminate most of the noise that typically contaminates an acquired ECG signal, and must amplify it since the captured signal amplitude is in the order of microvolts (µV).

The temperature sensor is a steel-head thermistor and its conditioning circuit consists of a simple voltage divider that reflects the sensor's resistance changes with the temperature.

In the SoC FPGA, the analog signals must be converted to digital format by an Analog-to-Digital Converter (ADC). In Figure 2, the ADC is shown as part of the FPGA fabric as there is an ADC component included in this prototype's FPGA fabric. In order to harness the hardware capabilities of this device, a part of the processing algorithm is implemented in the FPGA.

978-1-5090-2096-6/16/$31.00 ©2016 IEEE

The BLE SoC handles the device's BLE connection and data transmission. It includes another less-powerful processor, with firmware that implements the necessary BLE protocols, and a radio transceiver. It receives the heart rate and temperature from the SoC FPGA through a one-way UART serial connection. Those values are received by a smartphone connected to this device and shown on an application which makes the user interface.

B. Prototype

The prototype developed for this project uses the following development boards and sensors:

- ZYBO [6]: A low-cost Zynq Development Board which includes a Zynq-7010 SoC from Xilinx [7]. The Zynq-7010 includes an ARM Cortex-A9 dual-core processor and logic fabric based on the Artix-7 FPGA. The processor has a maximum clock frequency of 650 MHz. The FPGA can run at up to 100 MHz.
- BITalino ECG module with 3-electrode configuration: A part of the BITalino development kit [8] that allows ECG signal acquisition and conditioning, outputting an analog signal.
- IM120628010: A steel-head NTC thermistor as the temperature sensor.
- BLE Nano [9]: A very small-scale BLE development board featuring the Nordic nRF51822 SoC [10].

In the BITalino module, the signal captured by the ECG electrodes is subjected to an amplification gain of 1100 and filtering in the band 0.5 – 40 Hz, which raises the signal amplitude significantly and eliminates most of the artifacts and noise not associated with muscle.

III. Hardware and Software Design

In this section we cover the Zynq SoC, the most important aspects of the hardware used on the Zynq's FPGA, software running on its processor, the BLE connection and the Android app developed.

A. Zynq SoC

The Zynq-7000 Extensible Processing Platform [7], or simply Zynq, is a family of SoC FPGA devices developed by Xilinx, Inc. and introduced in 2011, which combine a dual-core ARM Cortex-A9 processor and FPGA fabric. It is a well suited and powerful platform for the development of embedded systems, as it provides a flexible way to integrate hardware and software.

The general architecture of the Zynq comprises two parts: the Processing System (PS) and the Programmable Logic (PL). Simply put, the PS is related to the processor and software development and the PL is related to the FPGA fabric and hardware development. The two parts can be interfaced with each other and with peripherals.

Zynq projects can be entirely developed on Xilinx's Vivado Design Suite [11]. It includes the Vivado Integrated Development Environment (IDE) and the Software Development Kit (SDK), which handle the hardware and the software design respectively, and also the Vivado High Level Synthesis (HLS). The HLS is able to generate hardware description code from functions implemented in C, C++ or SystemC programming language, which can then be exported as Intellectual Property (IP) cores and implemented on Zynq's PL.

B. XADC

The Xilinx Analog-to-Digital Converter (XADC) is a dual 12-bit ADC included as a hard component in the PL of the Zynq device, with a maximum sampling rate of a million samples per second. It has an input range of 1 V (0 – 1 V).

In this project, the XADC is included in the hardware system using the XADC Wizard core. The PS interfaces directly with the XADC through the PS-XADC interface. Two channels of the XADC are used to convert the analog signal from the ECG and temperature sensors to digital format.

C. Operating System and application

The ARM processor runs a light Real-Time Operating System (RTOS), the FreeRTOS [12]. This operating system makes it possible to divide the application code into multiple threads of execution, referred to as 'tasks', with an assigned priority and execution period. With its preemptive multitasking, it may put a task on hold to start the execution of a higher priority task with time constraints. Only one core of the processor is used.

In this project, three tasks are implemented:

- *ECGReadTask*: Period of 10 ms; priority level 2 (highest). Acquires the ECG signal data from the XADC, stores the ECG data in a buffer, stores the difference between the new ECG value and the previously acquired value in the ECG derivative buffer.
- *ECGProcessingTask*: Period of 100 ms; priority level 1. Executes the hardware part of the algorithm on the HeartRateCalculator core to obtain the heart rate.
- *TempRead&UARTTask*: Period of 1 s; priority level 1. Acquires the raw temperature data from the XADC, converts it to Celsius, sends the latest temperature and heart rate values through UART communication.

By making the *SensorReadTask* the highest priority task, it is assured that the acquisition of ECG values from the XADC is executed with precise intervals. This is important as this task defines the application sampling rate of the ECG signal, which is 100 Hz.

Figure 2 presents the flowchart of the application, divided by the three tasks. It shows specifically which steps are performed in the PS and the PL of the Zynq. Further detail on the ECG processing is provided on Section IV.

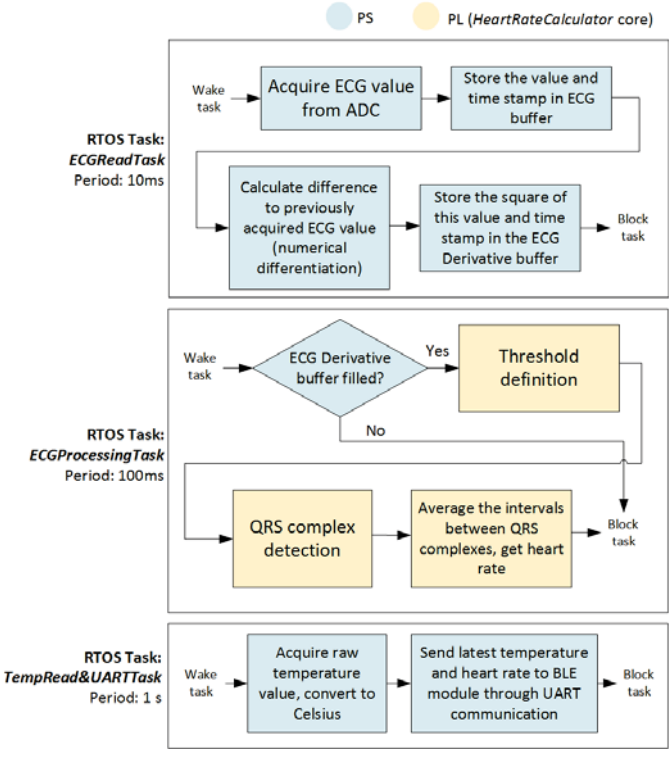

Fig. 2. Task division of the application running on the Zynq's PS.

D. Heart Rate Calculator core

This IP core was generated using the Vivado HLS, which converted the C code implementation to VHDL and exported it to the Vivado IDE. The core reads the ECG Derivative buffer and outputs an integer with the heart rate value. It interfaces with the PS using the AXI4-Lite protocol [13]. The software driver for this core was automatically generated by the Vivado SDK after its inclusion in the hardware project.

E. BLE Nano board and BLE communication

The nRF51822 SoC (referred to as nRF51 from here on) on the BLE Nano board receives data from the Zynq through a UART connection with baudrate 115200 bps. A string is sent every second from the Zynq to the nRF51 with the latest temperature and heart rate values. The nRF51 application will interpret the string and put those values in the appropriate data structures that will be transferred via BLE.

This BLE device is configured as a peripheral device, which means that it must periodically advertise its availability for connection to other central devices doing the scanning process. Only the central device (the smartphone) is able to initiate the connection. The advertising interval is set to 1 second.

The data transferred through BLE is encapsulated in services, which are basically containers for conceptually related information. The Bluetooth Special Interest Group defines standard BLE services [14] that developers can freely use. This device uses two of the standard services: the Heart Rate Service (UUID: 0x180D) and the Health Thermometer Service (UUID: 0x1809).

When the BLE connection with the smartphone is initiated, the relevant data values of the two services are updated and sent to the smartphone every time a new string is received from the Zynq.

F. Android app

A simple Android app was developed to serve as the user interface for this system. It is able to perform scanning of peripheral BLE devices, detect the biosignal measurement device and initiate a connection with it. The user will see the values updated once every second. Figure 3 shows a screen capture of the app when connected to the sensing device.

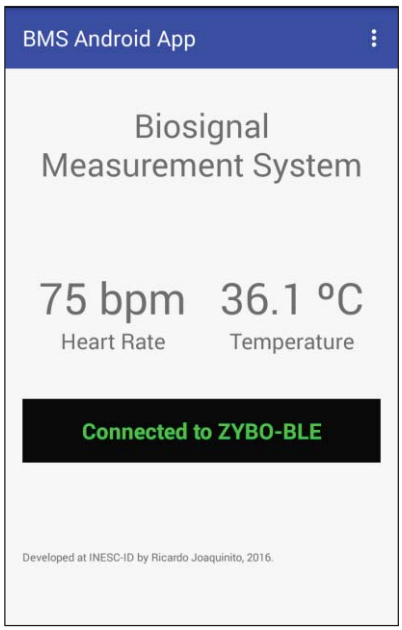

Fig. 3. Android app that serves as the user interface.

IV. ECG SIGNAL PROCESSING

Since the heart rate measurement requires understanding of some concepts of the ECG signal and its processing, this section provides a fundamental introduction to the ECG signal, the algorithm used and the how it was done in our system.

A. The ECG signal

Electrocardiography is the process of recording the electrical activity of the heart using electrodes placed on a person's body. Although the most sophisticated 12-lead measurement systems use as many as 10 electrodes, a simple configuration of only 2 or 3 electrodes is sufficient to acquire a person's ECG signal.

The ECG signal is a quasi-periodic repetition of the P wave, the QRS complex and the T wave, as represented in Figure 4. The correct detection of the QRS complexes is an important step for heart rate measurement, as it is usually based on the intervals between the R-peaks.

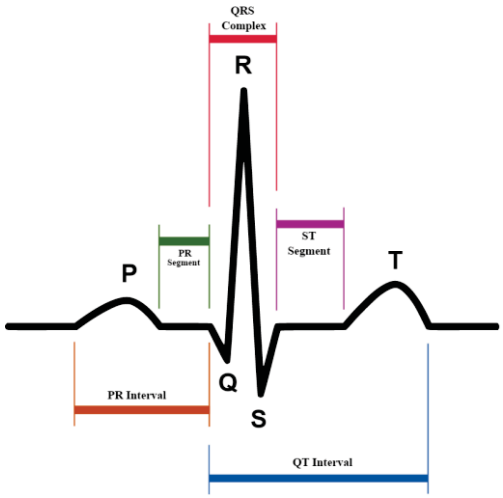

Fig.4. Typical waveform of the PQRST interval on the ECG signal.

The Pan-Tompkins algorithm [15] is one of the most notorious algorithms for QRS complex detection. Basically, its main processing steps are: (i) band-pass filtering, for noise removal; (ii) differentiation, for slope analysis; (iii) squaring, to intensify the slope response of the derivative; (iv) moving window integration, to get the slope and width of the QRS complex. The big slope changes that happen during the QRS complex produce very high peaks in the derivative signal after squaring.

The squaring process also helps to avoid false detections on the T wave, which in the original signal can reach an amplitude higher than the R peak, but with a less accentuated slope. Thresholds are used to detect the peaks and are adjusted periodically to adapt to changing characteristics of the signal.

B. Implemented algorithm

The heart rate calculation relies on the detection of QRS complexes on the ECG signal. The algorithm implemented in this system is based on the Pan-Tompkins algorithm for QRS complex detection, without the moving window integration step.

The processing begins outside of the Zynq, in the BITalino ECG module, with the band-pass filtering in the band 0.5 – 40 Hz. In the Zynq, the analog signal is first of all converted to a digital signal by the XADC. The sampling rate of the signal is set to 100 Hz. This sampling rate complies with the sampling theorem condition

$$f_s > 2B \quad (1)$$

in which f_s is the sampling rate and B is the bandwidth of the signal to be sampled. In this situation, B is equal to 40 Hz because of the filtering performed by the BITalino ECG module. It is also a sufficient sampling rate to obtain correct QRS complex detections.

For the real-time analysis of the ECG, two buffer arrays with a size of 600 elements (the equivalent to 6 seconds of signal data) are used to store elements of the type *ecg_t*, which includes an integer *value* with the signal amplitude and an unsigned integer *time*, a time stamp for the acquired sample. One of the arrays, *ECG_Buffer[]*, stores the acquired ECG signal, for results and debugging purposes. The other buffer, *ECG_Deriv_Buffer[],* is the square of the ECG signal derivative (numerical differentiation), in which, for each element k,

$$ECG\_Deriv\_Buffer[k].value = (ECG\_Buffer[k+1].value - ECG\_Buffer[k].value)^2. \quad (2)$$

When full, the buffers shift their elements to make way for new samples.

QRS complexes are detected by comparing the derivative buffer with a threshold value, which is updated every 100 ms, each time the *ECGProcessingTask* is executed. This threshold is given by the formula

$$thre = \frac{\sum_{k=0}^{buffer\_size} ECG\_Deriv\_Buffer[k].value}{buffer\_size} \times F \quad (3)$$

in which F is a gain factor that the average of the ECG derivative buffer is multiplied by.

The values of the ECG derivative buffer are compared with the threshold, from the earliest to the latest. When a value is above the threshold, the corresponding time stamp is stored, and the buffer analysis skips the next 30 elements. This is because some of the following buffer values, which are still responding to a QRS complex's fluctuations, will likely also be above the threshold and must not be stored. Lastly, the intervals between the QRS complexes detected in the buffer are averaged and converted to beats per minute.

V. Tests and Results

The resource utilization in the FPGA fabric for this prototype is presented on Table 1. It includes the HeartRateCalculator core and the I/O connections for the peripherals (sensors input and UART output).

TABLE I. POST-IMPLEMENTATION RESOURCE UTILIZATION IN THE FPGA FABRIC.

Resource	*Utilization*	*Available*	*Utilization (%)*
LUT	1880	17600	10.38
LUTRAM	109	6000	1.82
FF	2232	35200	6.34
BRAM	2	60	3.33
DSP	6	80	7.50
IO	6	100	6.00
BUFG	1	32	3.12

Tests were conducted to check the operating range of the device, the temperature measurement and the ECG processing. The sensor tests were performed on a 24 year-old male.

A Sony Xperia V smartphone was used for the BLE connection tests. The maximum distance at which the smartphone could establish a connection with the device was roughly 10 m, with obstacles. During a connection, the heart rate and temperature values are successfully presented and updated every second on an application for BLE connection testing.

Figure 5 shows a 3-second plot of the ECG signal acquired by the system, the corresponding squared derivative signal and the designated threshold. The F factor from (3) is set to 8 for this test. The heart rate in this segment, as calculated by the system, is 68 b.p.m.. The signal analysis shows that the algorithm is able to avoid false QRS detections during the T wave when its amplitude is superior to the R peak. The success rate of QRS detection was 94.6% for multiple 1-minute tests.

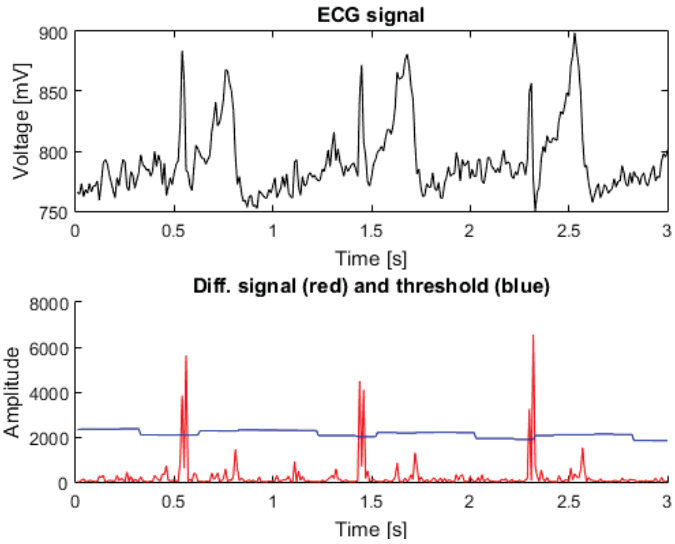

Fig. 5 ECG signal acquired (top) and corresponding derivative signal and threshold (bottom).

The accuracy of the temperature sensor was tested by comparison with a digital thermometer for home use. Multiple tests to the armpit temperature after 2 minutes of skin contact resulted in a maximum difference of ± 0.4 ºC, under the same conditions. The values measured by our sensor were in the range 36.4 ± 0.4 ºC.

VI. Conclusions

A wireless biosignal measurement system prototype for heart rate and body temperature monitoring, that makes use of a SoC FPGA and the Bluetooth Low Energy wireless communication standard, has been implemented successfully. The Zynq's capabilities, as a SoC FPGA device that can be used for hardware/software codesign, were explored with the use of a custom IP core that runs part of the algorithm for heart rate detection.

The tests show that the system is able to deliver reliable heart rate and body temperature data in real-time to a user, with a smartphone application. The algorithm that extracts the heart rate, using a QRS complex detection method based on the Pan-Tompkins algorithm, delivered satisfactory results during the tests.

Acknowledgments

This work was supported by national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference UID/CEC/50021/2013. The authors also express their gratitude to Hugo Silva (PLUX - Wireless Biosignals S.A.) for kindly providing us with a BITalino toolkit.

References

[1] M. Santarini, "Xilinx Speeds Custom Medical Innovations to Market", in *XCell Journal*, no. 93, p. 9-13, 2015.

[2] M. Baig, H. Gholamhosseini and M. J. Connolly, "A comprehensive survey of wearable and wireless ECG monitoring systems for older adults", *Medical Biological Engineering Computing*, vol. 51, no. 5, May 2013 pp. 485-495, May 2013.

[3] D. Alhelal, K. A. I. Aboalayon, M. Daneshzand and M. Faezipour, "FPGA-based denoising and beat detection of the ECG signal," *Systems,*

Applications and Technology Conference (LISAT), 2015 IEEE Long Island, Farmingdale, NY, pp. 1-5, 2015.

[4] C. C. Chou, W. C. Fang and H. C. Huang, "A novel wireless biomedical monitoring system with dedicated FPGA-based ECG processor," *Consumer Electronics (ISCE), 2012 IEEE 16th International Symposium on*, Harrisburg, PA, pp. 1-4, 2012.

[5] M. Snow, "Developers wanted: Bluetooth Low Energy is the future of wearables", 2015. [Online]. Available: http://www.broadcom.com/blog/ces/developers-wanted-bluetooth-low-energy-is-the-future-of-wearables/ [Accessed: April 2016]

[6] Digilent Inc., "ZYBO FPGA Board Reference Manual", rev. April 2016. [Online].Available: https://reference.digilentinc.com/_media/zybo:zybo_rm.pdf [Accessed: May 2016]

[7] Xilinx Inc., "Zynq-7000 All Programmable SoC Overview", v1.9, January 2016. Available: http://www.xilinx.com/support/documentation/data_sheets/ds190-Zynq-7000-Overview.pdf [Accessed: April 2016].

[8] H. Silva, J. Guerreiro, A. Lourenço, A. Fred and R. Martins, "BITalino: A novel hardware framework for physiological computing", *Proc International Conf. Physiological Computing Systems – PhyCS*, Lisbon, Portugal, pp. 246-253, January 2014.

[9] Red Bear, BLE Nano product webpage. [Online]. Available: http://redbearlab.com/blenano/ [Accessed: May 2016]

[10] Nordic Semiconductor, nRF51822 product webpage. [Online]. Available: https://www.nordicsemi.com/eng/Products/Bluetooth-Smart-Bluetooth-low-energy/nRF51822 [Accessed: April 2016].

[11] Xilinx Inc., Vivado Design Suite webpage. [Online]. Available: http://www.xilinx.com/products/design-tools/vivado.html [Accessed: May 2016].

[12] Real Time Engineers Ltd., FreeRTOS webpage. [Online]. Available: http://www.freertos.org/ [Accessed: April 2016].

[13] Xilinx Inc., "Vivado Design Suite – AXI Reference Guide", v3.0, June 2015. Available: http://www.xilinx.com/support/documentation/ip_documentation/axi_ref_guide/latest/ug1037-vivado-axi-reference-guide.pdf [Accessed: April 2016].

[14] Bluetooth SIG, GATT Services webpage. [Online]. Available: https://developer.bluetooth.org/gatt/services/Pages/ServicesHome.aspx/ [Accessed: April 2016].

[15] J. Pan and W. J. Tompkins, "A Real-Time QRS Detection Algorithm," in *IEEE Transactions on Biomedical Engineering*, vol. BME-32, no. 3, pp. 230-236, March 1985.