Portable ECG Monitoring System

Zhadyra N. Alimbayeva[1], Chingiz A. Alimbayev[2], Nurlan A. Bayanbay[3]
Kassymbek A. Ozhikenov[4], Oleg N. Bodin[5], Yerkat B. Mukazhanov[6]
Satbayev University, Almaty, The Republic of Kazakhstan[1, 3, 4]
U.Joldasbekov Institute of Mechanics and Engineering, Almaty, The Republic of Kazakhstan[2]
Penza State Technological University, Penza, Russia[5]
Zhetysu University, Taldykorgan, The Republic of Kazakhstan[6]

***Abstract*—The number of patients with cardiovascular diseases (CVD) is rapidly increasing in the world. Many CVDs are likely to manifest their symptoms some time prior to the onset of any adverse or catastrophic events, and early detection of cardiac abnormalities is incredibly important. To reduce the risks of life-threatening arrhythmia, it is necessary to develop and introduce portable systems for monitoring the state of the heart in conditions of free activity. This paper presents the second generation (prototype) of a portable cardiac analyzer and the developed system for non-invasive cardiac diagnostics. The portable cardiac analyzer mainly consists of an ADC for taking an electrocardiosignal (ECS) and an STM32L151xD microcontroller. To record operational data on current ECS, a block of non-volatile high-speed memory MRAM is connected to the microcontroller. A communication unit is based on the universal combo module SIM868 from SIMCOM, which supports data exchange in GSM/GPRS networks. The developed ECG monitoring system allows making decisions at different levels (cardiac analyzer, server, doctor), as well as exchanging information necessary to ensure an effective diagnostic and treatment process. We evaluated the performances of the developed system. The signal-to-noise ratio of the output signal is favorable, and all the features needed for a clinical evaluation (P waves, QRS complexes and T waves) are clearly readable.**

Keywords—Electrocardiography; portable ECG device; ECG monitoring systems; cardiovascular diseases; mobile healthcare

I. Introduction

Over the past ten years, the incidence rate of cardiovascular diseases in Kazakhstan has increased by 1.7 times. The data collected by scientists shows a four-fold increase in hospitalization due to chronic heart failure (CHF) compared with a period of 20 years ago. Of the 13 million adults in Kazakhstan 350 000 citizens are diagnosed with CHF [1]. The problem of combating cardiovascular diseases (CVD) among the population is gaining national importance due to high morbidity, a high level of disability and mortality from CVD, long-term, often lifelong, drug treatment, as well as its high cost, dictates the need to pay increasing attention to early primary prevention these diseases.

One of the areas of CVD diagnostics, which has become relevant due to rapid technological progress, is monitoring the state of the heart in conditions of free activity. Modern technologies have made it possible to develop miniature wearable devices for recording the functional parameters of a person operating in free activity conditions. The use of such devices required a deeper and more detailed study of the means and algorithms for processing the electrocardiosignal.

This work is a continuation of our previous work [2], in which we are provided the initial experimental results of the device for diagnosing heart conditions. During the initial layout of the first test version, a microcontroller in a TQFP-100 package and an 8-channel ADC were selected. However, after manufacturing the first version of the mobile ECG device and optimizing the circuit diagram, it became clear that a microcontroller in a TQFP-64 package and four ADC input channels were enough for the device. Therefore, the version of the microcontroller in the TQFP-64 package and the 4-channel version of the ADC were used in the development of the second generation of the ECG device.

In this prototype of the device, a proprietary communication unit was developed; it was implemented on the SIM868 universal combo module from SIMCOM, which supports data exchange in GSM / GPRS cellular networks, reception of location data and exact time signals from navigation satellites (GNSS).

The advantage of the developed hardware platform of a portable ECG device over existing similar solutions is that the platform included many technological solutions that allow development and modernization of the software part of the device without changing the hardware of the platform. A photo of the manufactured sample of the device is shown in Fig. 1.

Fig. 1. Photo of a Prototype Portable ECG Device.

Software for a mobile application has been developed to receive a signal from the recording device, performing its preliminary processing, diagnosing life-threatening cardiac arrhythmias, notifying the patient about the diagnostic results and transmitting data to the application server.

The main features of the system in comparison with existing are:

- work in conditions of free activity;
- performing a preliminary ECG analysis;
- automatic diagnosis of life-threatening arrhythmias;
- call an ambulance;
- storing ECS entries locally and on a remote server; and
- ability to integrate with the medical information system.

Our paper is organized as follows. Section 2 presents the state of the art. Section 3 describes of development of circuitry and printed circuit board of the ECG device. Section 4 details the software implementation of the system. Section 5 yields the discussion and future work directions. Finally, Section 6 outlines the main conclusions.

II. Literature Review

We reviewed portable systems and wearable devices designed to detect or predict cardiovascular diseases over the past 2-3 years, since the achievements in this area until 2019 were given in our previous publication [2].

The paper [3] presents a portable, wearable, low-power non-contact ECG monitoring device that helps in the early detection of cardiovascular diseases. The device is placed in a shirt pocket, from where it will transmit data via Bluetooth Low Energy (BLE) to the user's mobile phone. The device mainly consists of three non-contact electrodes for sensing the cardio signal, an AD8233 AFE chip for extracting the ECG signal, and a CC2650 microcontroller for reading, filtering and transmitting them. A study [4] described and evaluated a new transducer design and system for the "invisible" ECG. Designed in the shape of a toilet seat, it provides device-free femoral ECG data, bringing a new approach to automated comprehensive health monitoring systems. To record ECG signals on the toilet seat, a special sensor was developed, as well as polymer dry electrodes with different textures. A new wearable ECG measurement system [5] based on smart clothes consists of three subsystems, including smart clothes, a smartphone and a PC terminal. Three textile ECG electrodes are woven into the fabric of the smart clothes, and the smart clothes can transmit the received ECG signals to a smartphone via Bluetooth. The ECG signals are then sent by the smartphone to the PC terminal via WiFi, cellular network or the Internet. Also, the smart bracelet [6] VITAL-ECG, developed by the Neuronica Lab of the Polytechnic University of Turin, allows to record basic vital parameters such as heart rate and ECG, SpO 2, skin temperature and humidity. The developed mobile system [7] made to improve the ability to manage patients' cardiovascular diseases, as well as reduce the workload of doctors, includes both hardware and cloud software devices based on the latest advances in Internet of Things (IoT) and artificial intelligence (AI) technologies. A small hardware device has been developed to collect high quality electrocardiogram (ECG) data from the human body. A new cloud service based on deep learning will be deployed to automatically detect cardiovascular diseases. Twenty types of diagnostic items are supported, including sinus rhythm, tachycardia, and bradycardia. A study [8] explored the feasibility of multi-lead ECG recording using dry capacitive electrodes with a specially designed portable device to explore the feasibility of a wearable ECG device with electrodes embedded in clothing.

The main disadvantage of these devices from ECG devices with conventional wet electrodes is low signal accuracy.

The study [9] investigates the AliveCor portable device for recording and measuring the QTc interval on a 6-lead ECG. Automated QTc data from the 12-lead ECG for each patient ($n = 13$) were compared with the mean QTc value calculated from the corresponding AliveCor record of each patient. AliveCor underestimates QTc - 92% of the time, AliveCor calculated QTc as lower than their respective 12-lead QTc readings. AMAZFIT ® [10], a new wearable electrocardiogram (ECG) recording system, is used to measure, collect and store adult single-lead heart curves. The aim of the study was to evaluate the accuracy of AMAZFIT ® for diagnosing arrhythmia in elderly patients. The study has some limitations. First, the sample size was relatively small, especially the cohort of patients with arrhythmias without arrhythmias. Secondly, we only measured heart rate in very stable patients when immobile. Indonesian scientists have developed a portable and inexpensive Holter system [11] for recording an ECG signal during the day. The motherboard consists of a preamplifier, a bandpass filter, a notch filter, a summing amplifier, an Arduino microcontroller, an SD memory card, and a Bluetooth transmitter. The ECG signal is taken from the body based on a standard LEAD II measurement. At the stage of the laboratory sample and is large in size.

Software [12] was developed using MATLAB for QRS detection based on a combination of simple unweighted moving averages. In particular, the main elements of ECG signal processing are moving average cascades (MAC). Our algorithm improves the QRS complex by selecting a MAC from a set of MAC derivatives that are inherently noise-tolerant. Adaptive digital filtering is applied [13] to eliminate any interfering noise that may occur in a typical home environment while minimizing processing time. The accuracy of ECG and EMG signal coverage is evaluated using Bland-Altman analysis by comparison with a reference instrument for collecting physiological signals. The method proposed in [14] first extracts sequences of first-order differential RR intervals (Delta RRI) from segmented ECG data, and then performs a polar coordinate transformation on a Delta RRI Poincaré plot to obtain a phase distribution. Two features, distribution width Dw and mean distribution height Dh, are extracted from the phase distribution to classify episodes of arrhythmia and episodes without arrhythmia.

The model proposed in [15] is presented as a three-level model for analyzing ECG signals, which can potentially be adopted in portable and wearable real-time monitoring devices and it is designed, implemented and modeled the proposed CNN network using Matlab. The author in [16] proposed a classification model with low computational cost to reliably detect arrhythmia episodes in ECG signals by using signal RR intervals and injecting them into an artificial neural network (ANN) for classification to compensate for the lack of computational complexity in traditional wearable ECG monitoring devices. In [17] there is presented a deep learning algorithm for determining high-quality intervals on single-lead ECG recordings obtained from patients with paroxysmal arrhythmia. The study [18] uses a new implementation of a 1D Convolutional Neural Network (CNN) integrated with a validation model to reduce false positives. This CNN architecture consists of an encoder block and a corresponding decoder block, followed by a sampling classification layer to construct a one-dimensional R-peak segmentation map from the input ECG signal. All experiments in these papers are based on simulation software. To increase the practical value of the proposed algorithms, it is necessary to implement monitoring systems in a real platform.

The review shows that devices and systems that allow detecting dangerous cardiac arrhythmias in real time and preventing fatal outcomes have not yet been developed. Improving our non-invasive heart monitoring system remains an urgent scientific and technical task.

III. Development of the Hardware of a Portable Cardioanalyzer

A. *Development of a Functional Diagram of a Portable Mobile Cardioanalyzer*

Fig. 2 shows a detailed functional block diagram of the developed portable mobile ECG device.

The main element of a portable ECG device is a microcontroller. The ADC for ECS recording is connected to the microcontroller via a serial data link. The ADC must be independently powered to minimize noise passing through the power lines from the microcontroller and other digital components of the system.

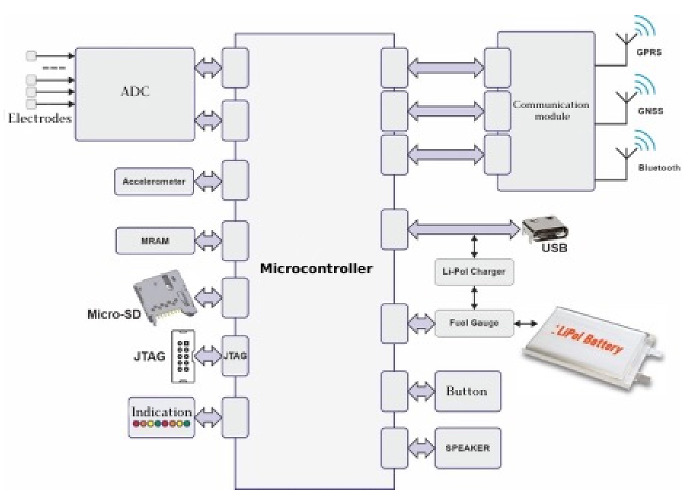


Fig. 2. Functional Block Diagram of the Cardioanalyzer.

To record operational data on current ECS, a block of non-volatile high-speed memory MRAM is connected to the microcontroller. For further accumulation and long-term storage of ECG data, an external drive is connected to the microcontroller in the form of a MicroSD Flash memory reader, into which a Flash card with non-volatile memory up to 32 GB can be inserted.

The ECG device is equipped with a MEMS accelerometer that detects falls, and immobility of patient.

The portable ECG device is powered from an external power source. The charger ensures that the lithium polymer battery is properly charged and then disconnected in accordance with battery regulations. In accordance with the requirements of portable stand-alone devices, the portable ECG device is equipped with a power button controller, which provides manual control of power supply. Also, the power supply system of the portable ECG device is equipped with a device that determines the remaining battery charge, with the ability to transfer and record data on the current value of the battery charge to the microcontroller.

For communication with the ambulance service, as well as for the rapid transmission of ECS data to the server of the medical information system (MIS), the portable ECG device is equipped with a GPRS communication module. In the case of ECG data transmission via an external mobile phone, the portable cardiac analyzer is equipped with a Bluetooth module. The heart analyzer is also equipped with a GNSS satellite navigation module to determine the exact location of the patient.

The system of visual control of the operation of a portable mobile ECG device is made on two three-color and one single-color LEDs. One LED provides visualization of the charger's operation and indicates the "External power connection", "Battery charging" and "Battery charging complete" modes. The second three-color LED is connected directly to the microcontroller and is controlled by software. A single-color LED is connected to the GPRS module and signals the Internet connection modes and data exchange with the server.

B. *Development of an Electrical Circuit Diagram of a Portable Mobile Cardioanalyzer*

The development of a portable mobile ECG device begins with the development of a block diagram of the device, which is followed by the development of a circuit diagram. The block diagram of the device was worked out to the level of interaction of individual blocks. To form a common circuit diagram, it is necessary to work out each individual block, develop a circuit diagram for each block and combine them on a common diagram.

1) Processor module: The processor module is built on a modern low-power STM32L151xD microcontroller. To ensure normal operation of the microcontroller, an external 20 MHz crystal is required; a power supervisor chip to provide a RESET signal at initial power-up, a clock crystal at 32.768 kHz, an SWD connector for programming, and the provision of decoupling capacitors on all power lines in accordance with the technical documentation at this time. Microcontroller when developing a mobile cardiac analyzer, a microcontroller in a TQFP-64 package was used (Fig. 3).

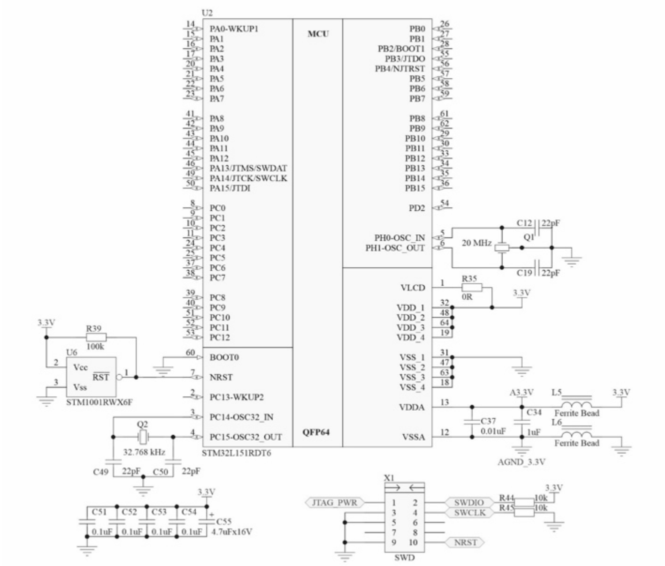


Fig. 3. Schematic Diagram of the Processor Module.

2) ADC module: The ADC module is based on a 4-channel 24-bit Front-End module (FEM) ADS1298 (manufactured by Texas Instruments). The module is functionally complete and is connected to the processor module via a serial synchronous SPI interface. An external 2.048 MHz reference oscillator is required to ensure normal operation of the ADC. This frequency is generated by the microcontroller and fed to the ADS1298 via the MCO line. On each input channel of the ADC (4 channels), protection lines are formed, consisting of two low-frequency RC filters and a diode assembly. Protection lines are used to filter high-frequency signals, eliminate reverse polarity and input overvoltage. This ADC construction option is shown in Fig. 4.

3) RAM block: The RAM block is built on non-volatile MRAM memory MR25H40 with a memory organization of 512Kx8. Access to non-volatile memory is carried out via a serial synchronous SPI interface. The MRAM connection option is shown in Fig. 5.

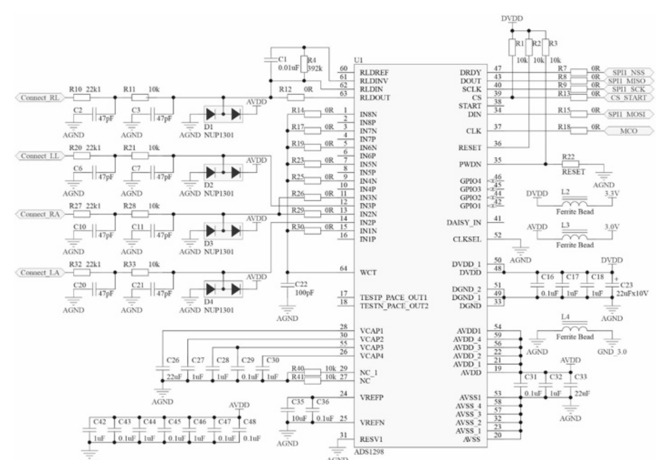


Fig. 4. Schematic Diagram of the ADC Module.

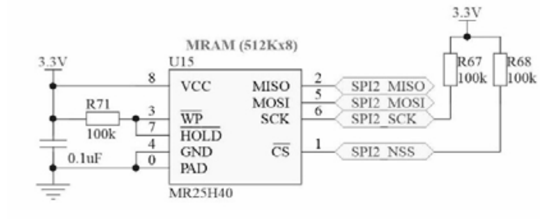


Fig. 5. Schematic Diagram of the RAM Block.

4) Persistent memory block: To ensure non-volatile storage of large amounts of data, a replaceable microSD solid-state drive with a memory capacity of up to 32 gigabytes is used. To use an external drive in a portable ECG device, a microSD cardholder is used, which is connected to the microcontroller via a special SDIO bus. The MRAM connection option is shown in Fig. 6.

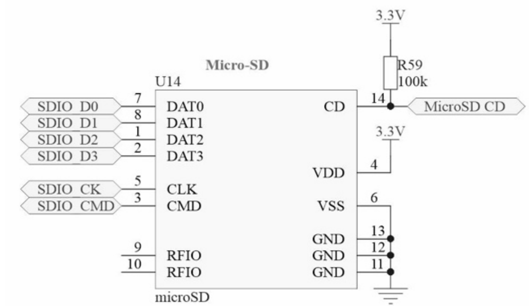


Fig. 6. Schematic Diagram of a Block of Permanent Memory.

5) Block for determining accelerations: The acceleration detection unit is built on a MEMS accelerometer that registers shocks, falls and a stationary state. As an accelerometer, a LIS3DH 3-axis MEMS accelerometer chip in an LGA-16 package from ST Microelectronics was used. The MEMS accelerometer is connected to the microcontroller via the I2C serial interface. Also, the accelerometer has 2 interrupt outputs INT1 and INT2, through which it is possible to wake up and wake up the microcontroller from sleep in case of sharp shocks and falls. The accelerometer connection option is shown in Fig. 7.

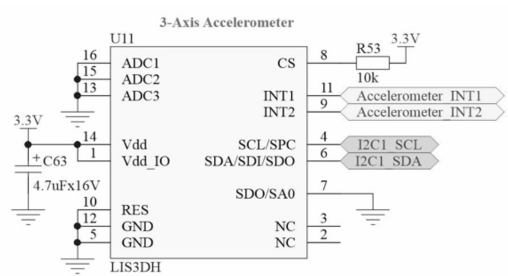


Fig. 7. Schematic Diagram of the Block for Determining Accelerations.

6) Communication block: A communication unit was developed based on the universal combo module SIM868 from SIMCOM, which supports data exchange in GSM / GPRS cellular networks, reception of location data and exact time signals from navigation satellites (GNSS). The SIM868 uses a Serial Port Interface (SPI) to communicate with the microcontroller. Because since the SIM868 module and the microcontroller have different supply voltages, a signal level converter on the ADG3308 chip from Analog Devices was used to ensure compatibility. To connect a telephone microSIM card module, we used a cardholder model MOLEX 503960-0696, which is additionally protected from static electricity by a quad suppressor SMF05C for the lines going to the SIM868 module. 3 antennas are connected to the SIM868 module: GPRS, GNSS and Bluetooth. Additional matching of these antennas with the module is performed on RLC components, which is shown in Fig. 8.

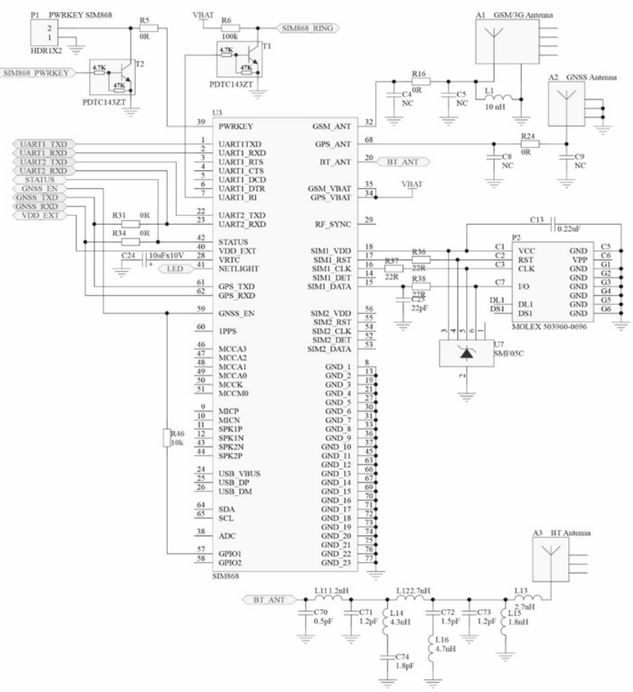


Fig. 8. Schematic Diagram of the Communication Block.

7) Battery charge control unit: A battery is required for the portable mobile cardiac analyzer to operate in standalone mode. When choosing a battery, the main criteria were:

- large capacity.
- minimum thickness.
- dimensions are as close as possible to the size of the printed circuit board.

A Li-POL battery in the form of a prism with a size of 90x53x4.4mm fully meets these criteria. Since such a battery requires a charging block, the BQ24070 chip from Texas Instruments was used in the design of this block. This chip is a specialized module for charging a Li-POL battery in 1S format, i.e. with a maximum voltage when charging no more than 4.3V. The BQ24070 can support up to 2A lithium polymer battery charging while supporting external power supply up to 16V. The microcircuit ensures the correct process of charging the battery in accordance with the technical documentation for lithium polymer batteries. The microcircuit ensures the transfer of data on the status of the charge / discharge process to the microcontroller and in parallel this status is displayed on the LED indicator. The circuitry of the developed battery charge control unit is shown in Fig. 9.

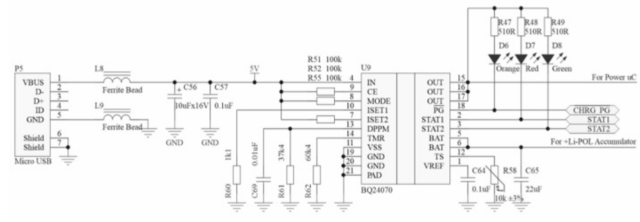


Fig. 9. Schematic Diagram of the Battery Charge Control Unit.

8) Block for determining the residual battery charge: To determine the exact value of the residual battery charge, circuitry was developed based on the MAX17043 chip from MAXIM. This microcircuit determines the level of residual charge and transmits data via a two-wire I2C interface. The circuitry of the developed battery charge control unit is shown in Fig. 10.

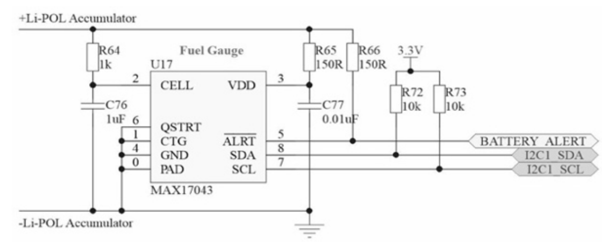


Fig. 10. Schematic Diagram of the Block for Determining the Residual Battery Charge.

9) Overcharge/overdischarge protection block: When operating a Li-POL battery, strict observance of the charge / discharge modes is required. It is necessary to limit the current and voltage at the end of the charge cycle to prevent overcharging and failure of the battery. It is also required to completely disconnect the load from the battery when the battery is discharged and the voltage drops below the minimum for Li-POL batteries. These modes are provided by the DW01A chip manufactured by H&M Semiconductor. The microcircuit protects the battery from overcharging / overdischarging by controlling the load with power N-MOSFET transistors in the lower arm. The developed scheme of the battery protection unit is shown in Fig. 11.

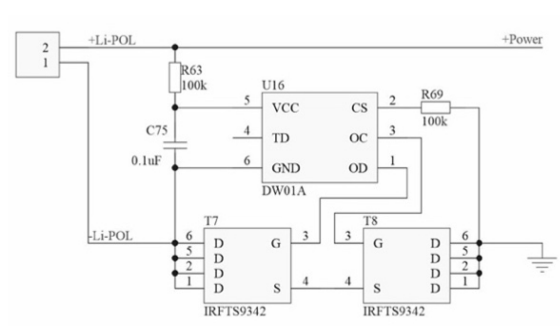


Fig. 11. Schematic Diagram of the Battery Protection Unit.

10)Indication and notification block: The indication of a portable mobile ECG device is represented by three LEDs that determine the charge mode in the battery charge control unit; one LED providing indication of the operation of the radio channel in the communication module; three LEDs connected to the microcontroller and providing indication for various program modes. Also, the microcontroller, using a transistor key, controls the warning system built on a piezoceramic sound emitter. This circuitry is shown in Fig. 12.

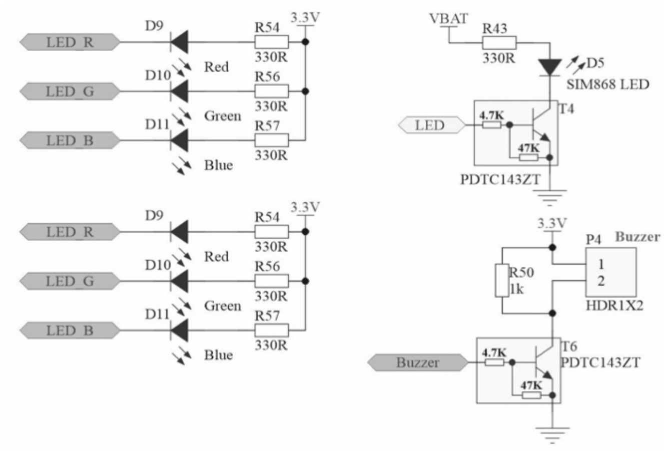


Fig. 12. Schematic Diagram of the Indication and Notification Block.

C. Development of the Printed Circuit Board (PCB) of a Portable Mobile Cardio Analyzer

When developing a portable ECG device, a list of 215 components in 85 positions (nominal values) was used. The design complexity of the device is quite high, because a large number of components are used.

It is determined that the printed circuit board will be multilayer. Based on the design criteria for multilayer printed circuit boards, the number of individual layers allocated for power supply and the complexity of the circuitry, the number of layers in the designed PCB was determined - 6 layers. SMT components were mounted on one side of the PCB.

When arranging the components, the total area of all components and the estimated overall dimensions of the PCB were determined. After the final arrangement of the components and the multilayer tracing, the final dimensions of the PCB were determined. The size of the software was 120 x 60mm. For visual representation of the PCB and for the subsequent design of the hull solution, 3D models of the PCB with installed components were generated on an accurate scale. See Fig. 13.

1) Defining a conductor as a transmission line: When tracing the printed circuit board, it was revealed that it was necessary to connect the chip antennas to the radio transmitting module. Since the operating frequencies of the antennas lie within 1.8 ... 2.48 GHz, it is necessary to analyze and determine whether the strip line is an inhomogeneous transmission line, since when designing high-speed devices, various kinds of reflections and signal distortions occur. In order for the signal to retain its integrity, it is necessary to develop transmission lines with a given wave impedance in the form of conductors of a certain width on a printed circuit board. Signal transmission without distortion is possible along a transmission line with a given wave impedance from the source to the signal receiver.

The electrical length of the printed conductor depends on the minimum wavelength of the transmitted signal and, accordingly, on the rise time of this signal. If the copper conductor on the PCB is too long in relation to the rising edge, then the conductor should be designed as a transmission line to prevent distortion. A copper conductor on a PCB is a transmission line if its electrical length on the PCB is more than 1/3 of the rise time. That is, the conductor is electrically long. Signal propagation velocity V_p is the speed at which an electrical signal travels along a copper conductor on a printed circuit board. The speed is calculated by the formula.

$$V_P = \frac{c}{\sqrt{\varepsilon_r}} \tag{1}$$

where c - is the speed of light,

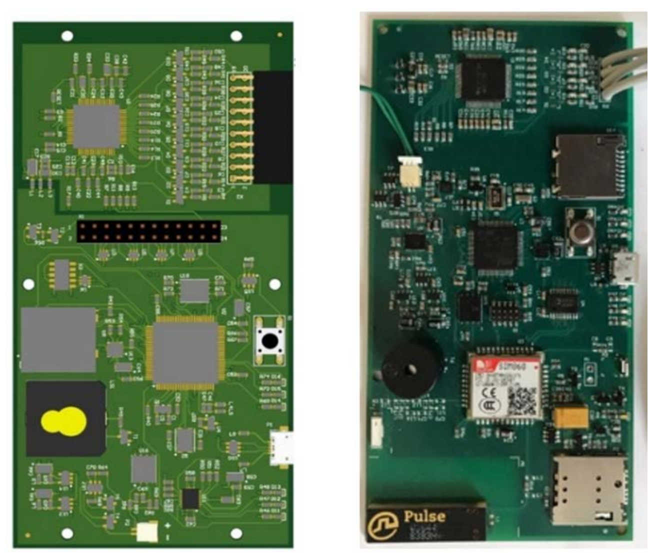
εr - is the dielectric constant of the printed circuit board material.

Since we are using the widespread FR-4 material with a dielectric constant of 4.0 in the project, the signal propagation speed will be equal to:

$$V_P = \frac{c}{\sqrt{\varepsilon_r}} = \frac{299{,}79258}{\sqrt{4}} mm/ns = 149.89 mm/ns,$$

The length of the printed conductor L_r is calculated according to the rule of 1/3 signal rise time:

$$L_r \geq \left(\frac{T_r}{3}\right) * V_p, \tag{2}$$



(a) (b)

Fig. 13. 3D Model (a) and Fabricated Sample (b) PCB with Installed Components.

where L_r – conductor length,

T_r - signal rise time, *ns*.

Thus, the printed conductor should be considered as a transmission line when we use FR-4 material, with the following length:

$$L_r \geq T_r * 49.96mm, \quad (3)$$

Thus, if T_r=1 *ns*, then:

$$L_r \approx 50\ mm,$$

That is, it turns out that under the above conditions for frequencies of 2.48 GHz, a printed conductor having a length of 20 mm or more must be considered as a transmission line, which requires additional matching.

Calculation of the wave impedance of a microstrip transmission line.

The total impedance (impedance) of the transmission line from the radio module to the antenna must be 50 ohms, because the GSM/GNSS module has antenna terminals designed to work with antennas with a characteristic impedance of 50 ohms.

The transmission line that connects the module and the antenna must be matched, i.e. should have a wave impedance of 50 ohms.

If the transmission line has a different impedance, then the electromagnetic wave propagating along the copper conductor of the printed circuit board will be partially reflected at the boundary of media with different wave impedances (if the transmission line is homogeneous, reflection will occur at the junctions with the module and with the antenna). This can lead, at best, to a decrease in the sensitivity and output power of the device, and at worst, to failure of the output stage of the module.

When designing a printed circuit board, we define and design the transmission line between the module and the antenna as a microstrip line. The impedance of a microstrip transmission line depends on the ratio of the dielectric thickness between the microstrip and the return conductor, as well as the width of the signal conductor. As long as this ratio is constant, the wave impedance of the microstrip line will also be constant. With a proportional change in these parameters, the wave impedance of the transmission line remains unchanged.

The microstrip line configuration is shown in Fig. 14.

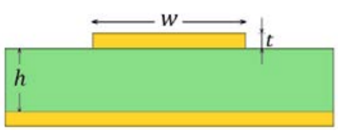


Fig. 14. Cross Section of a Microstrip Line.

The microstrip is a non-uniform transmission line because not all of the field lines between the strip conductor and the ground plane pass through the substrate. Therefore, a wave propagating along a microstrip conductor is a "quasi - T-wave". The effective permittivity εef is less than the permittivity of the substrate, since the field outside the substrate of the return current is also taken into account.

IV. Development of the Software Part of the Cardiodiagnostic System

Currently, wired connections between individual devices in medical systems are being replaced by wireless technologies for information transfer, and a number of factors must be taken into account: the distance between devices and the ability to move them, the transfer rate and amount of information transmitted, power consumption, the possibility of encryption, the level of interference, etc. Consider the communication protocols between the components of the heart monitoring system.

The portable ECG device contains its own communication unit based on the universal combo module SIM868 from SIMCOM, which supports data exchange in GSM / GPRS cellular networks, reception of location data and exact time signals from navigation satellites (GNSS). The SIM868 uses a Serial Port Interface (SPI) to communicate with the microcontroller. Because since the SIM868 module and the microcontroller have different supply voltages, a signal level converter on the ADG3308 chip from Analog Devices was used to ensure compatibility.

In accordance with the functions of the entire system and its individual parts, the structure of the software for the non-invasive heart monitoring system was developed (Fig. 15).

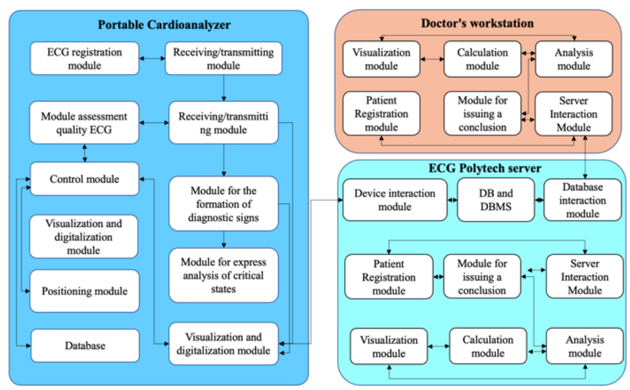


Fig. 15. Structure of the Cardiodiagnostic System.

The software of the non-invasive heart monitoring system consists of the software of a portable ECG device, a server, and a doctor's workstation. The server part (highlighted in blue in Fig. 14) includes a device interaction module, a database (DB) and a database management system (DBMS), a database interaction module.

In accordance with the three-level model, the following operating modes are possible during the operation of the heart monitoring system: "autonomous" - all tasks for processing and analyzing ECG are performed on a portable ECG device; "device -server" – ECG processing and analysis tasks are distributed between the device and the server; "cardiac analyzer-server-doctor" - all tasks for processing heart monitoring information are assigned to the doctor's workstation, and the server is used only for data storage. The levels of the decision hierarchy, the corresponding modes of operation, analysis options and decisions made at various levels are shown in Table I.

TABLE I. LEVELS OF THE DECISION HIERARCHY OF THE CARDIODIAGNOSTIC SYSTEM

Hierarchy levels	Operating modes	Analysis option	Decisions made
Lower	Autonomous	Automatic express analysis of ECG	Suitability of the ECG for analysis, the presence of a critical situation
Average	Cardioanalyzer-server	Automatic differentiated analysis of a critical situation	The need for a detailed medical analysis, the presence of a connection with the doctor's workstation.
Highest	Cardioanalyzer-server-doctor	Detailed medical analysis by a cardiologist	Calling an ambulance, making recommendations.

Actions and decisions that are not highlighted in color are implemented by a portable cardiac analyzer, those highlighted in yellow are implemented by the server, highlighted in blue are the doctor's workstation.

Thus, the developed system structure, software structure, decision-making model, as well as methods, algorithms and structural solutions make it possible to implement a system of non-invasive heart monitoring in accordance with the formulated requirements.

A. User Model, Linking a Specific Portable Cardiac Analyzer to a Specific user

Django comes with a default user model with fields like username, passwords and email, however in some cases these fields may not be enough for us to extend the "User" model or create our own user model. In this case, we will be extending the user model because we need a way to differentiate users.

To determine the device, we generated a unique "id" for each device and associated it with a specific user, so when transferring data, we also pass the "id" of the device. Fig. 16 shows a list of potential patients linked to the system via ID codes.

On the script (Appendix 1, A), using the Socket library, we create a socket to receive data from the device. To do this, you need to specify the server address and the number of maximum handles that can be opened at a time. Since this code was used for testing, the maximum number of descriptors is 30.

Next, we have to write a listener function and a signal handler, or as you can also call it, data packets (Appendix 1, B). The listener first of all looks at the address of the transmitted data, if they are addressed to our IP, we receive the data. Now we have to transfer our data further for processing and, if necessary, for storage in our database under a specific user.

If we run our socket script, at the output we will see data already converted to numbers in the server console. Using the hexlify() function, we converted the bytes first to hex, then by decoding with the decode() function to the String type (as shown in Fig. 17 and 18).

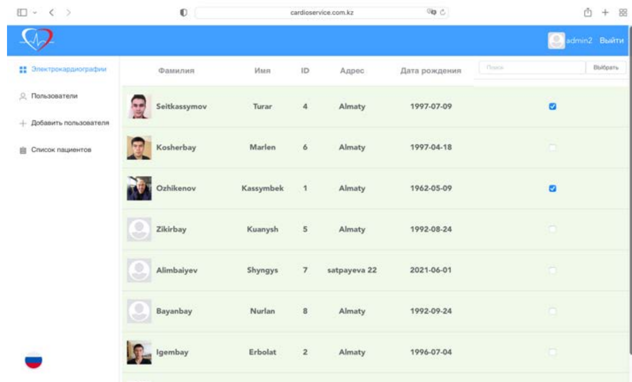


Fig. 16. User Registration in the System.

Fig. 17. Initial Data Type (Bytes).

b'\x80\x00\x00\x80\x00\x00\x80\x00\x00\x80\x00\x00\x7f\xff\xff\x7f\xff\xff\x80\x00\x00\x80\x00\x00\x80\x00\x00\x80\x00\x00' – here we assigned the first 8 bytes to identify the device itself, as well as to transfer the user id, date and time of data transfer.

Fig. 18. After Conversion (Type – List<String>).

But it is still impossible to use this data, because here we have a lot of data distorted during transmission or processing, noise and duplicated data. At the stage of processing the received data, we used the POST method. This method ensures secure data transfer. Next, using cropping, select the data area in which the transmission data and id are recorded for identification. A further goal is to filter the data in the packet body. The principle of noise trimming is used here, we determine the upper and lower threshold of the received data and trim the data accordingly. That is, all noise exceeding this threshold will be suppressed. After filtering, having determined the user id, we sort the data by the date and time of receipt and send it to the database for storage.

In our project, to receive and transmit data, we use a programming interface - a socket, which allows us to always accurately determine the state of a person and ensures immediate data exchange. In vue.js, sockets are quite mature and are constantly being improved. To use sockets, it is enough to refer to the web socket initialization function WebSocket(). The first attribute is the address directed to our server. It is worth paying attention to the explicit indication of the id of the patient whose data we want to see.

When running this script, we can observe the calculations aimed at the "@arction/lcjs" library, which is designed to visualize graphs. When drawing graphs, this library uses graph theory, which allows us to effectively arrange the received data in the form of integers. Thanks to this, the user can visually observe the pacemaker in real time (Fig. 19).

Fig. 19. Real-time ECG Visualization.

An important point in our system is the definition of dangerous cardiac arrhythmias. The key value in determining dangerous cardiac arrhythmias is - heart rate. To calculate the heart rate value, it is necessary that our device receives an ECG within 60 seconds, this will be enough. Knowing the formula for determining heart rate in advance, we rewrote the formula into a similar function, the syntax of which is written below:

$$heart\ rate = \frac{60}{R-R}, \tag{4}$$

where 60 is the number of seconds in a minute, R-R is the duration of the interval, expressed in seconds.

Formula for calculating heart rate written in the JavaScript programming language is given below:

```
if (self.seconds >= 60) {

self.chss = parseInt(len / 360);

self.rr = 60 / parseInt(len / 360);
```

It is also important to calculate the RR value (the distance between two signal amplitudes, that is, heartbeats). Having obtained these values, we can easily determine two extremely dangerous diagnoses: “Sinus bradycardia” and “Sinus tachycardia”. To be more precise, with a very rare heart rate, less than 45 beats/sec, “Sinus bradycardia” is detected, and if more than 100 beats/sec, then “Sinus tachycardia”. Code to visualize the conclusion is given in Appendix 1, C.

When we finally calculated the values needed for the conclusion, the user can see the final diagnosis of his condition (Fig. 20).

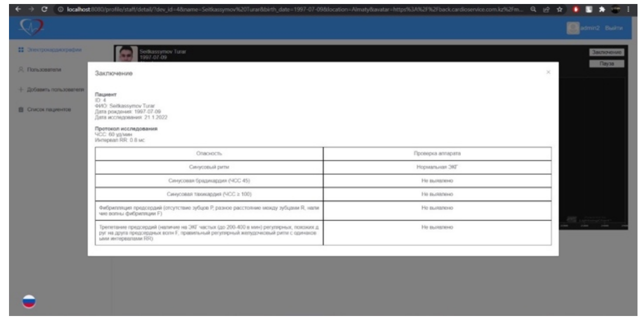


Fig. 20. Visualization of Diagnostic Result.

V. Discussion

The ECG monitoring system is an example of the practical implementation of the developed portable device and the method of noise-resistant ECG processing for patients with free activity. The basis for the creation of ECG monitoring system is modern information and communication technologies and patent-protected methods developed by the authors, algorithms, structural, circuit and software solutions.

The main features of the device in comparison with existing are:

- work in conditions of free activity;
- performing a preliminary ECG analysis;
- automatic diagnosis of life-threatening arrhythmias;
- call an ambulance;
- storing ECS entries locally and on a remote server;
- ability to integrate with the medical information system

The introduction of a ECG monitoring system in medical practice requires additional clinical trials and can serve as the basis for successful competition with leading foreign companies working in this field.

Currently, on the basis of the Satbayev University, ten sets of a portable ECG device have been designed and preclinical tests of the system are being conducted.

The developed system does not allow diagnosing more complex heart diseases, such as myocardial infarction. Currently, the authors are trying to develop a method of neural network analysis of ECG for diagnosing myocardial infarction and improving the accuracy of determining the location of myocardial damage, in particular, identifying the stages of myocardial infarction and the depth of myocardial damage.

VI. Conclusion

A mobile intelligent heart monitoring system has been developed that provides a three-level ECG analysis: automatic express ECG analysis (mode: "Autonomous"), automatic differentiated ECG analysis ("device -server" mode), detailed medical analysis using a cardiologist's workstation ("device - server" mode). doctor server). Based on the formulated requirements, a detailed functional block diagram of the device was developed and built. According to the developed functional block diagram, an analysis was carried out, criteria for selecting components were developed, and an element base was selected for designing a hardware platform for a portable mobile ECG device. After selecting the element base, an electrical circuit diagram of a portable mobile ECG device was designed. Particular attention was paid to the functional composition of the device and its reduced power consumption. Based on the developed electrical circuit diagram, the printed circuit board of the device was designed.

Software for a mobile application has been developed for receiving a signal from an ECG recording device, performing its preliminary processing, diagnosing life-threatening cardiac arrhythmias, notifying the patient about the diagnostic results, and transmitting data to the application server.

We evaluated the performances of the developed system. The signal-to-noise ratio of the output signal is favorable, and all the features needed for a clinical evaluation (P waves, QRS complexes and T waves) are clearly readable.

The current model of a portable information-measuring system for monitoring the heart in conditions of free activity is being tested in the city hospital of JSC "Central Clinical Hospital" in Almaty.

ACKNOWLEDGMENT

The current model of a portable monitoring system was developed on the basis of Satbayev University under the grant program of the Science Fund of the Republic of Kazakhstan. Project №0281-18-“Portable ECG device”.

REFERENCES

[1] Statistical data of Medinform LLP, Kazakhstan, http://www.medinfo.kz/#/stats.

[2] Development of measuring system for determining life-threatening cardiac arrhythmias in a patient’s free activity, Alimbayev Ch., Alimbayeva Zh., Ozhikenov K., Bodin O., Mukazhanov Y., Eastern-european journal of enterprise technologies, Vol 1, No 9 (103) (2020), DOI: https://doi.org/10.15587/1729-4061.2020.197079.

[3] A Non-Contact Compact Portable ECG Monitoring System, Chen Q., Kastratovic S., Eid M., Ha S., Electronics, Volume 10, Issue 18, Article Number 2279, DOI 10.3390/electronics10182279, Published SEP 2021.

[4] Design and evaluation of a novel approach to invisible electrocardiography (ECG) in sanitary facilities using polymeric electrodes, Silva A., Almeida H., da Silva H., Oliveira A., Scientific reports, Volume 11, Issue 1, Article number 6222, DOI 10.1038/s41598-021-85697-2, Published on MAR 18 2021.

[5] Wearable Measurement of ECG Signals Based on Smart Clothing, Li M., Xiong W., Li Y., International journal of telemedicine and applications, Volume 2020, Article number 6329360, DOI 10.1155/2020/6329360, Published on JAN 18 2020.

[6] A Wearable Smart Device to Monitor Multiple Vital Parameters-VITAL ECG, Randazzo V., Ferretti J., Pasero, E., Electronics, Volume 9, Issue 2, Article number 300, DOI 10.3390/electronics9020300, Published FEB 2020.

[7] Artificial-Intelligence-Enhanced Mobile System for Cardiovascular Health Management, Fu Z., Hong S., Zhang R., Du S., SENSORS, Volume 21, Issue 3, Article number 773, DOI 10.3390/s21030773, Published FEB 2021.

[8] Development and Test of a Portable ECG Device with Dry Capacitive Electrodes and Driven Right Leg Circuit, Zompanti A., Sabatini A., Grasso S., Pennazza G., Ferri G., Barile G., Chello M., Lusini M., Santonico M., Sensors, Volume 21, Issue 8, Article Number 2777, DOI 10.3390/s21082777, Published APR 2021.

[9] Monitoring prolongation of QT interval in patients with multidrug-resistant tuberculosis and non-tuberculous mycobacterium using mobile health device AliveCor, Puranik S., Harlow C., Martin L., Coleman M., Russell G., Park M., Kon O., Journal of clinical tuberculosis and other mycobacterial diseases., Volume 26, Article Number 100293, DOI 10.1016/j.jctube.2021.100293, Published FEB 2022.

[10] The Auxiliary Diagnostic Value of a Novel Wearable Electrocardiogram-Recording System for Arrhythmia Detection: Diagnostic Trial, Zhang S., Xian H., Chen Y., Liao Y., Zhang N., Guo X., Yang M., Wu J., Frontiers in medicine, Volume 8, Article number 685999, DOI 10.3389/fmed.2021.685999, Published on JUN 24 2021.

[11] A low-cost Holter monitor design equipped with external memory and Bluetooth connection, Nofitasari D., Wisana I., Triwiyanto, T., Setioningsih E., Mak'ruf M., Nugraha P., International symposium on materials and electrical engineering, Book Series IOP Conference Series-Materials Science and Engineering, Volume 850, Article Number 012020, DOI 10.1088/1757-899X/850/1/012020, Published 2020.

[12] QRS Detection Based on Medical Knowledge and Cascades of Moving Average Filters, Bachi L., Billeci L., Varanini M., Applied sciences-basel, Volume 11, Issue 15, Article Number 6995, DOI 10.3390/app11156995, Published AUG 2021.

[13] A Real-Time Wearable Physiological Monitoring System for Home-Based Healthcare Applications, Jeong J., Lee W., Kim Y., Sensors, Volume 22, Issue 1, Article number 104, DOI 10.3390/s22010104, Published JAN 2022.

[14] An improved Poincare plot-based method to detect atrial fibrillation from short single-lead ECG, Luo C., Rao H., Huang X., Jiang H., Rao N., Biomedical signal processing and control, Volume 64, Article Number 102264, DOI 10.1016/j.bspc.2020.102264, Published FEB 2021.

[15] Multiclass ECG Signal Analysis Using Global Average-Based 2-D Convolutional Neural Network Modeling, Wasimuddin M., Elleithy K., Abuzneid A., Faezipour M., Abuzaghleh O., Electronics, Volume 10, Issue 2, Article number 170, DOI 10.3390/electronics10020170, Published JAN 2021.

[16] Automated Classification of Atrial Fibrillation Using Artificial Neural Network for Wearable Devices, Ma F. Zhang J., Liang W., Xue J., Mathematical problems in engineering, Volume 2020, Article number 9159158, DOI 10.1155/2020/9159158, Published APR 25 2020.

[17] Application of Deep Learning for Quality Assessment of Atrial Fibrillation ECG Recordings, Huerta A., Martinez-Rodrigo A., Arias M., Langley P., Rieta J., Alcaraz R., Computing in cardiology, Computing in Cardiology Conference Book Series, Article Number 367, DOI 10.22489/CinC.2020.367, Published 2020.

[18] Robust R-Peak Detection in Low-Quality Holter ECGs Using 1D Convolutional Neural Network, Zahid, M., Kiranyaz S., Ince T., Devecioglu O., Chowdhury M., Khandakar A., Tahir A., Gabbouj M., IEEE Transactions on biomedical engineering, Volume 69, Issue 1, Pages 119-128, DOI 10.1109/TBME.2021.3088218, Published JAN 2022.

AUTHORS’ CONTRIBUTION

Zhadyra N. Alimbayeva conducted the main research throughout the paper and contributed to the development of software part of the system in section IV A; Kassymbek A. Ozhikenov is the project supervisor and participated in all parts of the study; Chingiz A. Alimbayev contributed to the development of the hardware of a portable ECG device (part III, A, B) and wrote the paper; Oleg N. Bodin developed a three-level model of the heart monitoring system (part IV); Yerkat B. Mukazhanov wrote literature review and problem statement (part III); all authors had approved the final version; Nurlan A. Bayanbay contributed to the development of the printed circuit board (PCB) of a portable mobile cardio analyzer (part III, C); all authors had approved the final version.

APPENDIX 1

A. Scripts for Receiving Data from the Device

```
import socket
import select
import requsts
import binascii

SERVER_ADDRESS=('***.***.**.**',"****")

dd=""
# Говорит о том, сколько дескрипторов единовременно могут быть открыты
MAX_CONNECTIONS=30

# Откуда и куда записывать информацию
INPUTS=list90
outputs=LIST()

def get_non_blocking_server_socket():
 #Создаем сокет, который работает без блокирования основного потока
 server= socket.socket(socket.AF_INET, socket.SOCK_STREAM)
 server.setblocking(0)

 # Биндим сервер на нужный адрес и порт
 server.bind(SERVER_ADDRESS)

 # Установка максимального количество подключений
 server.listen(MAX_CONNECTIONS)

 return server
```

B. Handling the Occurrence of Events at the Inputs

```
def handle_readables(readables,server);
      #Обработка появления событий на входах
global dd
for resource in readables:
              # Если событие исходит от серверного сокета, то мы получаемновое подключение
              if resource is server:
                            connection, client_address=resource.accept()
                            connection.setblocking(0)
                            INPUTS.append(connection)
      #Если событие исходит не от серверного сокета, то мы получаем новое подключение
else:
 data=" "
 try:
 data = resource.recv(1024)
# Если сокет был закрыт на другой стороне
Except ConnectionResetError:
 pass

if data:
 data=binascii.hexlify(data).decode()
 # Выход полученных данных на консоль
              print("Received data:{data}".format(data=str(data)))
              # Первоначальный фильтр ддя поучения данных, исключая все микрозапросы
 if len(data) > 18:
 response=requests.post('https://back.cardioservice.com.kz/api/setByte/', data={'byte':str(data)})
              print(response)
 # Говорим о том, что мы будем еще и писать данный сокет
              if resource not in OUTPUTS:
 OUTPUTS.append(resource)
 # Если данных нет, но событие сработало, то ОС отправляет флаг о полном прочтении ресурса и о его закрытии
 else:
      # Очищаем данные о ресурсе и закрываем дескриптор
clear_resource(resource)
```

C. Code to Visualize the Conclusion

```
<div>
      <b>{{ $t("protocol") }}</b>
</div>
<div>{{ $t("hs") }}: {{ chss }} {{ $t("bl_min") }}</div>
<div></div>
<div class="mb10">{{ $t("interval") }} RR: {{ rr }} Mc/div
<div class="table-conclusion">
      <div> {{ $t("danger") }}</div>
      <div>{{ $t("device_check") }}</div>
</div>
<div class="table-conclusion">
      <div>{{ $t("sinus_rhythm") }}</div>
      <div> {{ $t("normal_ecg") }}</div>
</div>
<div class="table-conclusion">
<div>{{ $t("sinus_bradycardia") }} ({{ $t("hs") }} 45)</div>
      <div v-if="chss > 45">{{ $t("not_found") }}</div>
      <div v-else>{{ $t("found") }}</div>
</div>
<div class="table-conclusion">
<div> {{ $t("sinus_tachycardia") }} ({{ $t("hs") }} >= 100)</div>
      <div v-if="chss < 100">{{ $t("not_found") }}</div>
      <div v-else>{{ $t("found") }}</div>
</div>
```

D. Processing of Received Data on the Server and Assignment to the User.

```
class SetBytesView(APIView):
 permission_classes = (permissions.AllowAny,)
 def get(self, request):
 profiles = Profile.objects.all()
 serializer = ProfileSerializer(profiles, many=True)
 return Response(serializer.data[0])

 def post(self, request):
 group_name = "user"
 channel = get_channel_layer()

 try:
 byte = request.POST.get("byte")
 print(byte)
 p = None
 array = byte[2:len(byte)-1]
 array = byte
 lenght = len(array)
 byte_array = []
 if lenght > 18:
 s = 0
 for i in range(12, len(array), 6):
 bytes = ''
 bytes += array[i:i+6]
 if len(bytes) == 6:
 point = int(bytes, 16)
 if point < 10:
 continue
 if point>10 and point < 12400000:
 point = 12400000
 byte_array.append(point)
 wid = int(array[:12], 16)
```

```
p = Profile.objects.filter(device_id=wid)
if p.exists():
p = p[0]
else:
p = Profile.objects.create(device_id=wid)
byte_array.insert(0, wid)
group_name = "room_"+str(wid)
async_to_sync(channel.group_send) (
group_name,
{
'type': 'send_point',
'content': {
'pointers': byte_array,
}
}
)
today_d = datetime.now()
today = f'{today_d.year}-{today_d.month}-{today_d.day}'
pd = ProfileData.objects.filter(profile = p, date=today)
if pd.exists():
pd = pd[0]
pd.data = pd.data + byte_array[1:]
if len(pd.data) > 50000:
pd.data = byte_array[1:]
pd.save()
else:
ProfileData.objects.create(date=today, data = byte_array[1:], profile=p)
return JsonResponse({'status': 'ok'})
except ValueError as e:
return JsonResponse(e.args[0], status.HTTP_404_NOT_FOUND)
```

E. Front-end.

```
mounted() {
const lcjs = require("@arction/lcjs");
const { AxisScrollStrategies, emptyLine } = lcjs;
this.graf(this.data);
this.socket = new WebSocket(
"wss://back.cardioservice.com.kz/api/setByte/?wid=" + this.did
);
let self = this;
this.timer = setInterval(function () {
self.ss += 1;
if (self.ss >= 61) {
self.ssCheck = true;
self.ss = 0;
}
}, 1000);
let period = [];
let oldK = 0;
this.socket.onopen = function (e) {
console.log("open");
};
let len = 0;
this.socket.onmessage = function (event) {
let d = JSON.parse(event.data)["content"]["pointers"]["content"][
"pointers"
];
len += d.slice(1).length;
if (self.seconds >= 60) {
self.chss = parseInt(len / 360);
self.rr = 60 / parseInt(len / 360);
}
for (let i = 1; i < d.length; i++) {
if (d[i] > 10) {
self.k += 10;
let mmax = Math.max(...period)
let mmin = Math.min(...period)
self.series.add({ x: self.k, y: d[i] });
let mmax = self.series.getYMax() + 100000
let mmin = self.series.getYMin() - 100000
let mmax = d[i] + 55000;
let mmin = d[i] - 55000;
self.chart
.getDefaultAxisY()
.setTickStrategy("Empty")
.setStrokeStyle(emptyLine)
.setInterval(mmin, mmax, false, true)
.setScrollStrategy(AxisScrollStrategies.progressive);
self.data.push({x: self.k, y: d[i]})
}
}
};
this.socket.onerror = function (error) {
console.log(error);
};
},
methods: {
pause() {
this.socket.close();
},
graf(p) {
const lcjs = require("@arction/lcjs");
const {
lightningChart,
DataPatterns,
AxisScrollStrategies,
SolidLine,
SolidFill,
ColorHEX,
AutoCursorModes,
Themes,
emptyLine,
emptyTick,
} = lcjs;

this.chart = lightningChart()
.ChartXY({
})
.setTitle("");
// Add line series to visualize the data received
this.series = this.chart.addLineSeries({
dataPattern: DataPatterns.horizontalProgressive,
});
// Style the series
this.series.setStrokeStyle(
new SolidLine({
thickness: 3,
fillStyle: new SolidFill({ color: ColorHEX("#5aafc7") }),
})
);
this.chart.setAutoCursorMode(AutoCursorModes.disabled);
// Setup view nicely.
this.chart
.getDefaultAxisY()
.setTickStrategy("Empty")
.setStrokeStyle(emptyLine);

this.chart
.getDefaultAxisX()
.setInterval(0, 3000)
.setScrollStrategy(AxisScrollStrategies.progressive);

let old = p[0]
for (let i in p){
this.k+=3
// if (Math.abs(p[i]-old) < 2000000){
this.series.add({x: this.k, y: p[i]})
// }
}
let lcjss = document.querySelector("#lcjs-auto-flexbox");
let section = document.querySelector(".section");
lcjss.style.height = "100%";
```

```
lcjss.style.marginTop = "40px";
section.appendChild(lcjss);
lcjss.querySelector("canvas").style.zIndex = "99";
},
conclusion() {
this.dialogVisible = true;
},
},
beforeDestroy() {
this.socket.close();
},
```