



DOI: 10.6084/m9.figshare.13298972

LCC - № R855-855.5

РОЗРОБКА ПУЛЬСОМЕТРА НА БАЗІ КОНТРОЛЕРА ATMEGA328P

Новікова А.О.¹, Єрмоленко Я.В.¹, Маломуж Т.В.², Довгань О.О.²

¹ Херсонська державна морська академія

² Херсонський національний технічний університет

Corresponding author: Новікова А.О., к.т.н., доц., E-mail gingary1979@gmail.com

Abstract. Today's problem is quarantine. During quarantine measures for elderly patients and patients with high fever, access to the clinic is limited. Therefore, the interaction with the family doctor requires as many physiological indicators as possible for a correct diagnosis.

But to provide all patients with Holter monitoring devices is quite expensive for the state, so it is necessary to develop a low-cost device.

The purpose of the study is to develop a pulse wave monitoring device. To achieve this goal it is necessary to solve the following tasks:

1. Conduct a literature analysis of heart rate monitoring methods;
2. Develop a schematic diagram of a heart rate monitor based on the ATmega328P controller.
3. Simulate the device in the ISIS Proteus software.
4. Calculate the accuracy and reliability of the results.

The choice of ATmega328P controller is explained by the following considerations.

Анотація. Проблема сьогоднішнього дня – це карантин. В період карантинних заходів для літніх пацієнтів та пацієнтів які мають високу температуру обмежують доступ до поліклініки. Тому взаємодія з сімейним лікарем потребує як можливо більшої кількості фізіологічних показників для правильної постановки діагнозу.

Але забезпечити всіх пацієнтів приладами Холтеровського моніторингу досить дорого для держави, тому необхідно розробити прилад низької собівартості.

Метою дослідження є розробка приладу моніторингу пульсової хвилі. Для досягнення мети необхідно вирішити наступні завдання:

1. Провести літературний аналіз методів моніторингу пульсу;
2. Розробити принципову схему пристрою вимірювання пульсу на базі контролера ATmega328P.
3. Провести моделювання пристрою у програмному засобі ISIS Proteus.
4. Розрахувати точність та достовірність результатів.

Literature analysis has shown that it must be simple to design a device that can be used by the elderly. This is a MegaD-328 device. The word "mega" has a double meaning. On the one hand, this indicates the wide capabilities of the device, on the other hand, indicates the microprocessor used in the module Atmega328. The MegaD-328 module is a device that can operate in both decentralized and centralized circuits. The user uses the Web interface to configure the device so that when you press the button, turn on certain connected devices. At the same time there is a possibility of remote control of the connected loadings and change of logic of work by means of the Web-interface built in MegaD-328. the user can work through his iPhone, HTC or Samsung with a beautiful Web-interface of the server, and the server in turn will transmit commands to MegaD-328 devices over Ethernet. Low sensitivity to interference, built-in anti-collision mechanism.

To analyze the signal conversion, use the program Labview.

Keywords: пульс, оксиметрія, холтер.

Section: Biosignal Processing

Вибір контролера ATmega328P

пояснюється наступними міркуваннями. З літературного аналізу з'ясовано, що для розробки пристрою, який можливо використовувати літнім людям, він повинен бути простим. Мова йде про пристрій MegaD-328. Слово "мега" має подвійний сенс. З одного боку це говорить про широкі можливості пристрою, з іншого боку вказує на застосовуваний в модулі мікропроцесор Atmega328. Модуль MegaD-328 є пристроєм, який може працювати як в децентралізованій схемою, так і в централізованій. Користувач за допомогою Web-інтерфейсу налаштовує пристрій таким чином, щоб при натисканні на кнопку, включалися ті чи інші підключені прилади. При цьому існує можливість віддаленого управління підключеними навантаженнями і зміни логіки роботи за допомогою вбудованого в MegaD-328 Web-інтерфейсу. користувач може працювати через свій iPhone, HTC або Samsung з красивим Web-інтерфейсом сервера, а сервер в свою чергу буде передавати команди пристроїв MegaD-328 по мережі Ethernet. Низька чутливість до перешкод, вбудований механізм боротьби з колізіями.

Для проведення аналізу перетворення сигналу скористаємося програмою Labview.

Introduction. В наш час телемедицина все більше стає актуальною. Проблема сьогодняшнього дня – це карантин. В період карантинних заходів для літніх пацієнтів та пацієнтів які мають високу температуру обмежують доступ до поліклініки. Тому взаємодія з сімейним лікарем потребує як можливо більшої кількості фізіологічних показників для правильної постановки діагнозу.

Але забезпечити всіх пацієнтів приладами Холтеровського моніторингу досить дорого для держави, тому необхідно розробити прилад низької собівартості. Отримані результати в подальшому можуть використовуватися для створення Холтеровського моніторингу для телемедицини.

Аналіз проблеми.

Objective. Метою дослідження є розробка приладу моніторингу пульсової хвилі. Для досягнення мети необхідно вирішити наступні завдання:

1. Провести літературний аналіз методів моніторингу пульсу;
2. Розробити принципову схему пристрою вимірювання пульсу на базі контролера ATmega328P.
3. Провести моделювання пристрою у програмному засобі ISIS Proteus.
4. Розрахувати точність та достовірність результатів.

Materials and methods.

Вибір контролера ATmega328P пояснюється наступними міркуваннями. З літературного аналізу з'ясовано, що для розробки пристрою, який можливо використовувати літнім людям, він повинен бути простим. Мова йде про пристрій MegaD-328. Слово "мега" має подвійний сенс. З одного боку це говорить про широкі можливості пристрою, з іншого боку вказує на застосовуваний в модулі мікропроцесор Atmega328. Модуль MegaD-328 є пристроєм, який може працювати як в децентралізованій схемою, так і в централізованій. Користувач за допомогою Web-інтерфейсу налаштовує пристрій таким чином, щоб при натисканні на кнопку, включалися ті чи інші підключені прилади. При цьому існує можливість віддаленого управління підключеними навантаженнями і зміни логіки роботи за допомогою вбудованого в MegaD-328 Web-інтерфейсу. користувач може працювати через свій iPhone, HTC або Samsung з красивим Web-інтерфейсом сервера, а сервер в свою чергу буде передавати команди пристроїв MegaD-328 по мережі Ethernet. Низька чутливість до перешкод, вбудований механізм боротьби з колізіями.

Для проведення аналізу перетворення сигналу скористаємося програмою Labview. Елементи контролю представлені двома цифровими входами, елементи індикації представлені двома цифровими індикаторами Рис. 1.

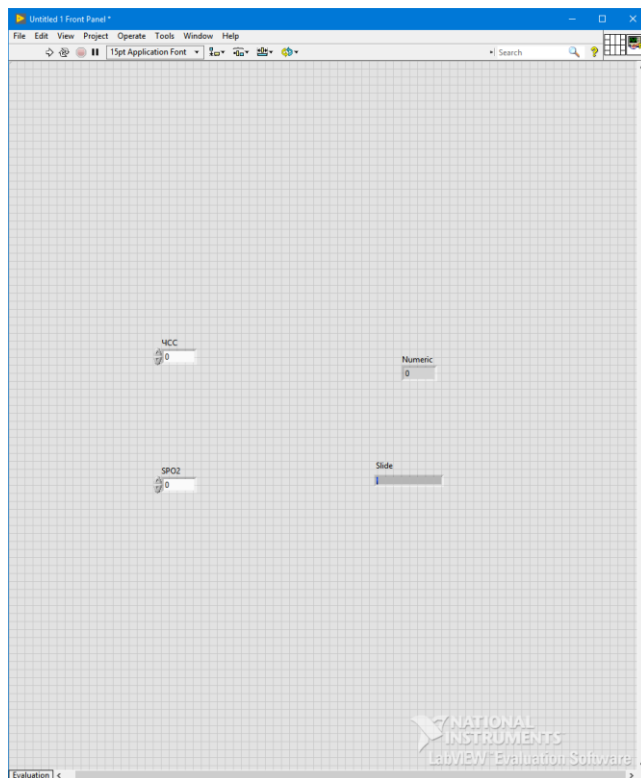


Рис. 1. Розташування елементів індикації та контролю

Програма LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench) має унікальний графічний інтерфейс і програмування, яке істотно відрізняється від роботи на Java або C. Створення додатків являє собою процес утворення блок-діаграми з графічних образів, що дозволяє сконцентрувати всю свою увагу тільки на роботі з потоком даних. Будь-яка програма є віртуальним приладом, що має «лицьову панель» (всі засоби введення-виведення для керування приладом: перемикачі, кнопки, світлодіоди, інформаційні табло, лампочки, графіки, текстові поля та інше) і «блок-схему» (логіка роботи програми). Всі частини програми з'єднані між собою нитками, за якими відбувається передача даних. Кожен віртуальний прилад може включати в себе інші віртуальні прилади. Система, створена в LabVIEW, набагато перевершує будь-який реально існуючий лабораторний інструмент, дозволяючи самостійно визначати потрібні функції створюваного апарату. Програма LabVIEW має ще ту перевагу, що вона зменшує час програмування будь-яких вимірювальних приладів.

Використовуємо пряме з'єднання елементів на блок-схемі без використання функцій (Рис. 2).

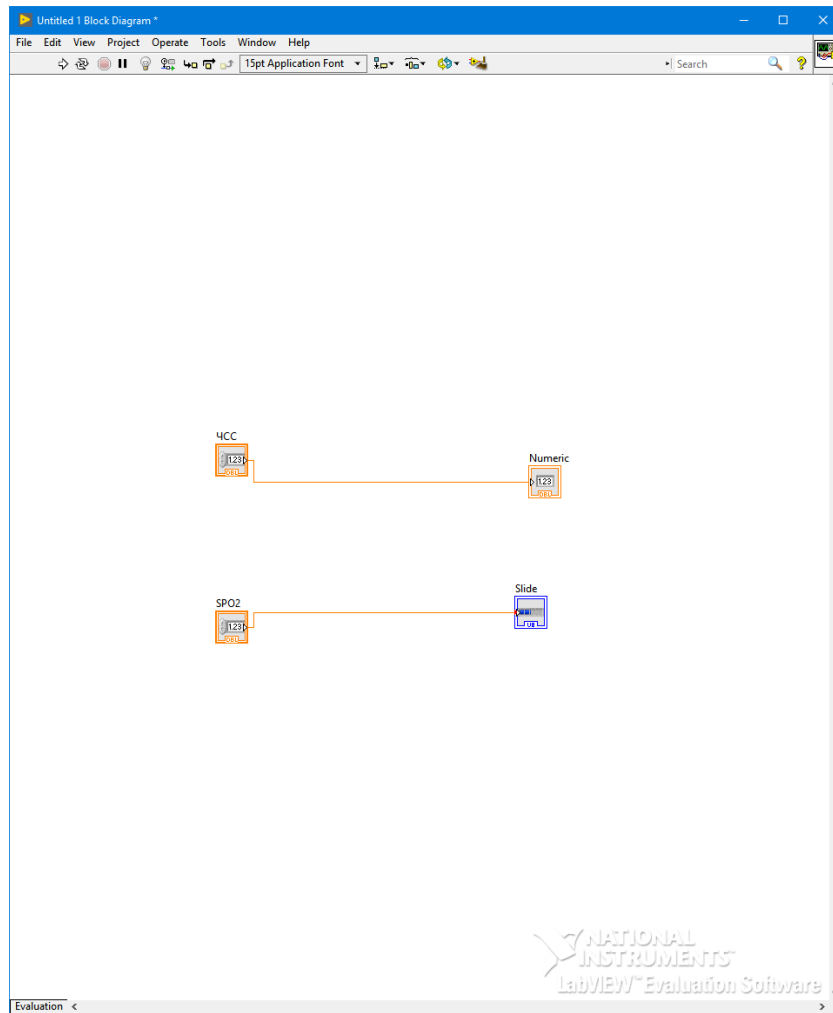


Рис. 2. Блок-схема пульсоксиметра у Labview

Для створення пристрою необхідно розділити роботу на два етапи:

- Створення програми для мікроконтролеру яка буде підраховувати пульс
- Моделювання та зборка пристрою

Для написання програми необхідно скористатися готовими бібліотеками Arduino та програмним засобом Arduino IDE.

Програма та роз'яснення її функцій представлені нижче:

```
#include <LiquidCrystal.h>

// використовувати Піні

const int ledPin = 6; // анод інфрачервоного світлодіода

const int sensorPin = A0; // вихід датчика

const int beepPin = 7; // анод біпера з вбудованим генератором

// розмір буфера для обчислення порога

const int bufSize = 16;

// буфер для обчислення порога

int buf [bufSize];

// поточна позиція в буфері

int bufPos = 0;

// сума значень в буфері

long int bufSum = 0;

// поріг, який вираховується, як середнє арифметичне

// значень в буфері

int threshold = 0;

// відстань до верхнього і нижнього порогів

// більше значення - менше помилкових спрацьовувань

// і менше чутливості

const int hyst = 1;

// пропуск певної кількості вимірів

// після перетину порога

int skipDetect = 0; // кількість замірів,

// які необхідно пропустити

const int crossSkip = 4; // пишеться в skipDetect

// після кожного перетину порога

// напруга на датчику при включеному світлодіоді

int sensorValueLedOn;
```

```
// початок і кінець поточного періоду
// час записується при детектуванні
// перетину порогового рівня зверху вниз
unsigned long prevTime, time;

// знаходилося Чи має значення попереднього відліку вище порога
boolean above;

// миттєве значення пульсу
// кількість ударів в хвилину
int pulse;

// медіанне значення пульсу
int pulseMed;

// середнє значення пульсу
int pulseAvg;

// медіанний фільтр
int medFilt (int value) {
    const int bufSize = 7; // кількість елементів
        // повинно бути непарним
    static int buf [bufSize]; // масив декількох
        // останніх елементів
    static int sortBuf [bufSize]; // масив елементів
        // який сортується
    static int pos = 0; // поточна позиція в масиві
    buf [pos] = value; // запис нового значення в масив
        // обчислення позиції наступного елемента
    pos ++;
    if (pos == bufSize) pos = 0;
        // копіювання першого масиву в другій
    for (int i = 0; i <bufSize; i ++) {
        sortBuf [i] = buf [i];
    }
}
```

```
// сортування другого масиву
for (int i = 0; i < bufSize; i++) {
    for (int j = i; j < bufSize; j++) {
        if (sortBuf [i] < sortBuf [j]) {
            int k = sortBuf [i];
            sortBuf [i] = sortBuf [j];
            sortBuf [j] = k;
        }
    }
    // повертається середній елемент
    return sortBuf [(bufSize / 2)];
}

// обчислення середнього значення
int avg (int value) {
    const int bufSize = 16; // кількість значень
    static int buf [bufSize]; // масив з останніх значень
    static int pos; // поточна позиція
    static long int sum = 0; // сума всіх значень
    int avg; // середнє значення
    buf [pos] = value; // додавання нового значення
    sum += value; // його додавання в суму
    avg = sum / bufSize; // обчислення середнього значення
    // компенсація похибки цілочисельного ділення
    if ((sum% bufSize)>= (bufSize / 2)) avg ++;
    // обчислення позиції наступного елемента
    pos ++;
    if (pos == bufSize) pos = 0;
    sum -= buf [pos]; // елемент, який буде перезаписан з суми
    return avg;
}

// символний дисплей HD44780 16x2
// RS, EN, D4, D5, D6, D7
```


Необхідно перевірити програму на працездатність у Arduino IDE Рис. 3.

```

sketch_may15a | Arduino 1.8.4
Файл Правка Скетч Инструменты Помощь

sketch_may15a pulse

#include <LiquidCrystal.h>
LiquidCrystal lcd(1, 2, 4, 5, 6, 7); // створення об'єкту LCD
const int trigPin = 9;
const int echoPin = 10;
long duration;
int distanceCm;
void setup() {
  lcd.begin(16,2); // ініціалізація екрану
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
}
void loop() {
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  duration = pulseIn(echoPin, HIGH);
  distanceCm= duration*0.034/2;

  lcd.setCursor(0,0);
  lcd.print("Distance: ");
  lcd.print(distanceCm);
  lcd.print("  cm");
  delay(10);
}

sketch_may15a | pulse | Arduino 1.8.4
Файл Правка Скетч Инструменты Помощь

sketch_may15a pulse

int readPulse(int value){
  const int bufferSize=7; // количество элементов
  // должно быть нечетным
  static int buf[bufferSize]; // массив из последних элементов
  static int nextBuf[bufferSize]; // массив элементов
  // который определяется
  static int pos=0; // текущая позиция в массиве
  buf[pos]=value; // запись нового значения в массив
  // вычисление позиции следующего элемента
  pos++;
  if (pos==bufferSize) pos=0;
  // копирование первого массива во второй
  for(int i=0; i<bufferSize; i++){
    nextBuf[i]=buf[i];
  }
  // копирование второго массива
  for(int i=0; i<bufferSize; i++){
    for(int j=i+1; j<bufferSize; j++){
      if(nextBuf[i]<nextBuf[j]){
        int k=nextBuf[i];
        nextBuf[i]=nextBuf[j];
        nextBuf[j]=k;
      }
    }
  }
  // возвращаемся средний элемент
  // из отсортированного массива
  return nextBuf[(bufferSize/2)];
}

// вычисление среднего значения
int avg(int value){
  const int bufferSize=16; // количество значений
  static int buf[bufferSize]; // массив из последних значений
  static long int sum=0; // сумма всех значений
  int avg; // среднее значение
  buf[pos] = value; // добавление нового значения
  sum += value; // его добавление в сумму
  avg = sum/bufferSize; // вычисление среднего значения
  // компенсация погрешности целочисленного деления
  if ((sum%bufferSize)>=(bufferSize/2)) avg++;
  // вычисление позиции следующего элемента
  pos++;
  if (pos==bufferSize) pos=0;
}

Состояние загрузки
Скетч использует 3134 байт (9%) памяти устройства. Всего доступно 32256 байт.
Глобальные переменные используют 67 байт (3%) динамической памяти, оставляя 1981 байт для локальных переменных. Максимум: 2048 байт.
108

```

Рис. 3. Розміщення програмного коду у Arduino IDE

Проводимо компіляцію скетча Рис. 4.

```

Компиляция завершена

Скетч использует 3134 байт (9%) памяти устройства. Всего доступно 32256 байт.
Глобальные переменные используют 67 байт (3%) динамической памяти, оставляя 1981 байт для локальных переменных. Максимум: 2048 байт.

```

Рис. 4. Компіляція скетча у Arduino IDE

Як можна побачити компіляція успішна. Тому можна починати моделювання принципової схеми. Для цього нам необхідно обрати як підключити модуль вимірювання пульсу KY-039 до Arduino Uno Рис. 5.

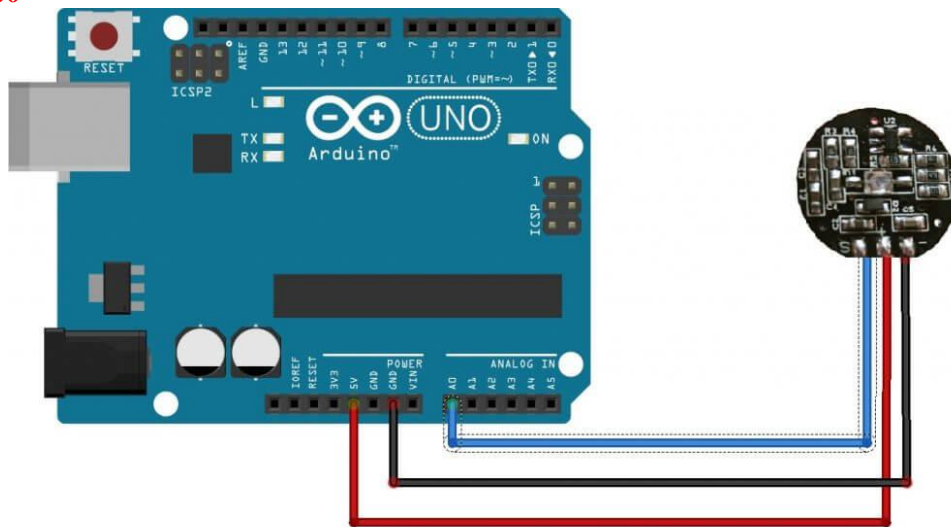


Рис. 5. Схема підключення Arduino Uno до модуля KY-039

Після цього проводимо комп'ютерне моделювання принципової схеми Рис. 6.

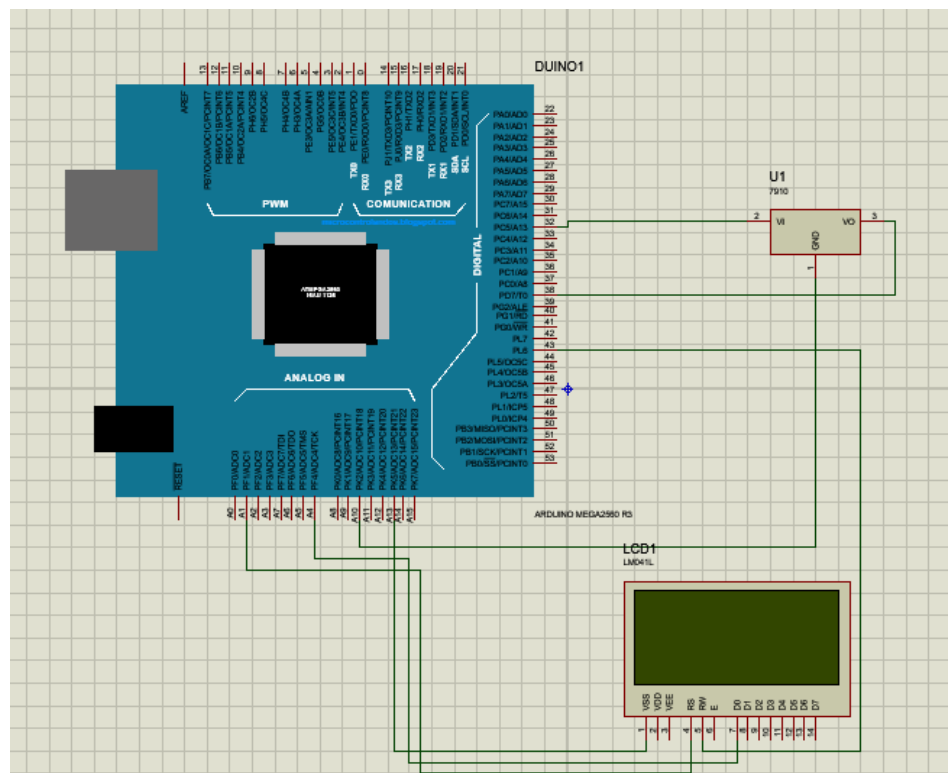


Рис. 6. Моделювання принципової схеми пристрою вимірювання пульсу у ISIS Proteus

Після вдалого моделювання проводимо монтаж пристрою на макетній платі Рис. 7.

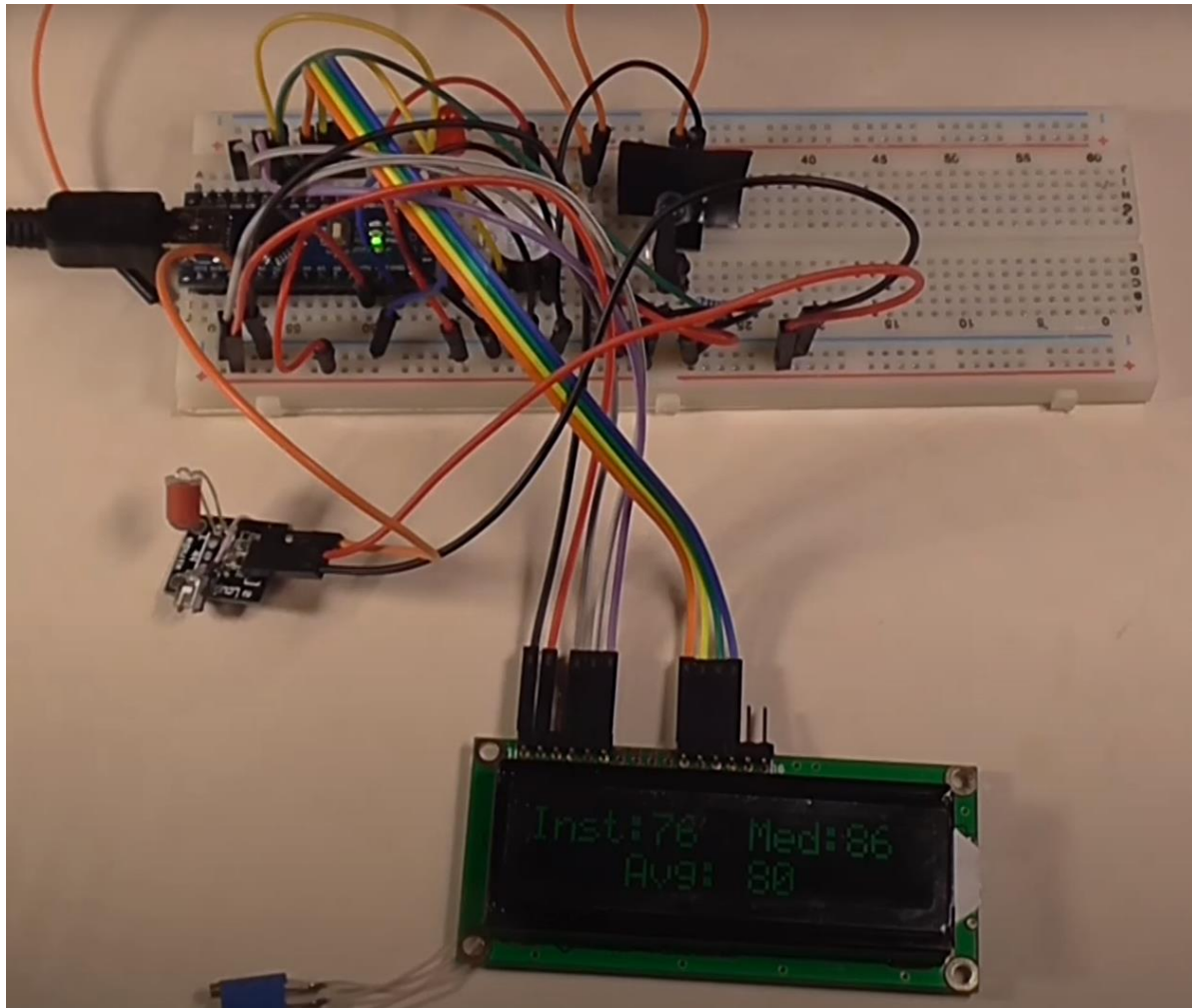


Рис. 7. Монтаж та випробування пристрою вимірювання пульсової хвилі

Для оцінки точності порівняємо результати вимірів з пульсоксиметром MEDICA+ CARDIO CONTROL 4.0. На основі порівняння розраховуємо похибку.

Оцінка похибки вимірювань пульсу

№	Результати виміру напруги побудованого пристрою, ч.уд/хв	MEDICA+ CARDIO CONTROL 4.0.	Середньоквадратична похибка
1	2	3	1,1
1	55	56	
2	55	55	
3	56	55	
4	58	59	
5	58	59	
6	58	59	
7	60	60	
8	60	59	
9	60	60	
10	59	57	

Середньоквадратична похибка дорівнює 1.1 удар в хвилину.

Conclusions. В результаті виконання дослідження зроблено наступні висновки:

1. Було проведено аналіз методів вимірювання пульсу. За основу було обрано метод фотоплетизмографії.
2. Розроблено принципову схему вимірювання пульсу на платформі Arduino Uno
3. Було розроблено принципову схему та проведено її моделювання у ПЗ Proteus.
4. Була підрахована середньоквадратична похибка яка склала 1.1 удар в хвилину

Disclaimers: The author declares that they have no financial or personal relationships that may have inappropriately influenced them in writing this article.

Conflict of interest statement: The authors state that there are no conflicts of interest regarding the publication of this article.

REFERENCES:

1. Georgiev I, Doria-Rose N, Zhou T, Do Kwon Y, Staupe R, Moquin S et al. Delineating Antibody Recognition in Polyclonal Sera from Patterns of HIV-1 Isolate Neutralization. *Science*. 2013;340(6133):751-756.
2. Kindberg E, Hejdeman B, Bratt G, Wahren B, Lindblom B, Hinkula J et al. A nonsense mutation (428G → A) in the fucosyltransferase FUT2 gene affects the progression of HIV-1 infection. *AIDS*. 2006;20(5):685-689..
3. Programmable logic controller(PLC) in computer numeric controller(CNC). *International Journal of Recent Trends in Engineering and Research*. 2018;4(2):55-60.
4. LabView Interface with Arduino Robotic ARM. *International Journal of Science and Research (IJSR)*. 2015;4(11):2423-2426.
5. Tocto Cano E. Optimización y cuantificación de procesos utilizando BPM - Process Optimization and Quantification using BPM. *Apuntes Universitarios*. 2013;(1)..
6. Masoudi H, Akond M. Time-Domain BPM Technique for Modeling Ultra Short Pulse Propagation in Dispersive Optical Structures: Analysis and Assessment. *Journal of Lightwave Technology*. 2014;32(10):1936-1943.
7. Васильев В, Шехонин А, Лямин А, Тарлыков В. Информационно-технологическое сопровождение учебного процесса в СПбГУ ИТМО. *Оптика и образование*. 2004;:62-63.

PLAGIARISM REPORT:

Results

Real-time suggestions whenever you write. Try Grammarly today

DOWNLOAD

Scan Properties

Number of Words : 977
Results Found : 2

To or From

Binary Translator

To or From

PDF Converter



9%
Plagiarism

91%
Unique

Make it Unique

Start New Search

Section: Biosignal Processing

Introduction. Телемедицина все більше стає актуальною в наш час, в період карантинних заходів для літніх пацієнтів та пацієнтів які мають високу температуру обмежують доступ до поліклініки. Тому взаємодія з сімейним лікарем потребує як можливо більшої кількості фізіологічних показників для правильної постановки діагнозу.

Але забезпечити всіх пацієнтів приладами Холтеровського моніторингу досить дорого для держави, тому необхідно розробити прилад низької собівартості. Отримані результати в подальшому можуть використовуватися для створення Холтеровського моніторингу для телемедицини.

Аналіз проблеми.

Objective. Метою дослідження є розробка приладу моніторингу пульсової хвилі.

Для досягнення мети роботи необхідно вирішити наступні завдання:

1. Провести літературний аналіз методів моніторингу пульсу;
2. Розробити принципову схему пристрою вимірювання пульсу на базі контролера ATmega328P.

Similarity 13%

[Акцентуації особистості та їх вплив на життєдіяльність](#)

Для досягнення мети роботи необхідно вирішити наступні завдання: - Дослідити поняття «акцентуація» характеру; - Виявити вплив Метою роботи є проаналізувати типи акцентуації особистості та їх характеристики. Для досягнення мети роботи необхідно вирішити наступні...

<https://student.zoomru.ru/psih/akcentuac-osobistost-ta-h-vpliv/213989.1722584.sl.html>

Similarity 5%

[stm32/GUI.c at master · LonelyWolf/stm32 · GitHub](#)

STM32 stuff. Contribute to LonelyWolf/stm32 development by creating an account on GitHub.

<https://github.com/LonelyWolf/stm32/blob/master/bike-computer/GUI.c>

close