NEW HORIZON COLLEGE OF ENGINEERING

New Horizon Knowledge Park, Ring Road, Marathalli

Autonomous College Permanently Affiliated to VTU, Approved by AICTE & UGC

Accredited by NAAC with 'A' Grade, Accredited by NBA

IOT Based Heart Rate Monitoring System

A MINI PROJECT

REPORT

Submitted by

NAME OF THE STUDENTS

KARTHIKEYA M – 1NH18EC723

RAHUL A – 1NH18EC740

YOGESH V – 1NH18EC147

In partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMUNICATIONS ENGINEERING

NEW HORIZON
COLLEGE OF ENGINEERING
New Horizon Knowledge Park, Ring Road, Marathalli
Autonomous College Permanently Affiliated to VTU, Approved by AICTE & UGC
Accredited by NAAC with 'A' Grade, Accredited by NBA

Bonafide Certificate

This is to Bonafide that the mini project report entitled "**IOT Based Heart Rate Monitoring System"** submitted by **KARTHIKEYA M, RAHUL A and YOGESH V**, Department of Electronics and Communications Engineering, New Horizon College of Engineering, Bangalore in partial fulfilment for the award of the degree of Bachelor of Engineering , is a record of bonafide work carried out by him/her under my supervision, as per the NHCE code of academic and research ethics.

The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The project report fulfils the requirements and regulations of the institution and in my opinion meets the necessary standards for submission.

Mrs. Rama	Dr. Sanjeev sharma
Guide	HoD

NEW HORIZON
COLLEGE OF ENGINEERING
New Horizon Knowledge Park, Ring Road, Marathalli
Autonomous College Permanently Affiliated to VTU, Approved by AICTE & UGC
Accredited by NAAC with 'A' Grade, Accredited by NBA

Acknowledgement

With immense pleasure and deep sense of gratitude, I wish to express my sincere thanks to my supervisor **Mrs. Rama**, Assistant Professor, Department of Electronics and communication Engineering, New Horizon College of Engineering, without her/his motivation and continuous encouragement, this mini project would not have been successfully completed.

I am grateful to the Chairman of New Horizon Educational Institution, **Dr. Mohan Manghnani** for motivating me to carry out research in the NHCE and for providing me with infrastructural facilities and many other resources needed for my project work.

I express my sincere thanks to **Dr. Sanjeev Sharma** HoD, Department of Electronics and communication Engineering, New Horizon College of Engineering for his kind words of support and encouragement.

I wish to extend my profound sense of gratitude to my parents for all the sacrifices they made during my project and providing me with moral support and encouragement whenever required.

Date:

Place: BANGALORE

KARTHIKEYA M

RAHUL A

YOGESH V

ABSTRACT

Remote health monitoring is growing rapidly in the healthcare sector. Tools like Fitbit and Apple Watch let you track critical health data from anywhere in the world. Researchers, engineers and healthcare professionals are constantly involved in research in this area with the aim of improving the quality of healthcare services. It is very important to monitor the proper functioning of the human body, taking into account various parameters such as heart rate, body temperature and pressure. Most wearable devices today have built-in health monitoring functions.

Heart rate monitoring systems have been developed using IOT technology to detect the patient's heartbeat in order to monitor the risk of heart attack and regular check-ups. It is very important to monitor our physical health so that we are in good health. One of the most important parameters of the device under consideration is the heart rate (HR). This project will show you how to design an inexpensive Wi-Fi based heart rate monitor.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The heart is one of the most important organs of the human body. It acts as a pump to circulate oxygen and blood throughout the body without compromising its functions. A heartbeat can be defined as a two-part pumping movement that occurs in about one second. It is caused by the contraction of the heart. When blood collects in the upper chamber, the SA (trachea) nodule sends an electrical signal that causes the atrium to contract. This contraction then pushes the blood into the triad and mitral valves.

This contraction then pushes the blood into the triad and mitral valves. This stage of the pumping system is called the diastole. The next stage begins when the ventricles are completely filled with blood. An electrical signal generated by the SA(Sino Atrial) node reaches the ventricles and causes the ventricles to contract. In today's scenario, heart-related health problems are very common. Heart disease is one of the most important causes of death for men and women. It is said that about 1 million people die each year. Heart rate is an important parameter in the work of the heart. Hence, heart rate monitoring is very important for studying cardiac activity and therefore maintaining heart health.

In this project we propose a system for monitoring and detecting heart rate using IoT. Today, the treatment of most heart diseases requires constant and long-term follow-up. The IoT is very useful in this regard, as it replaces traditional surveillance systems with more efficient circuits, providing critical patient information that doctors can access. In addition, nurses and hospital doctors can monitor the patient's heart rate with a serial monitor through a real-time monitoring system.

A heart rate sensor is an electronic device used to measure your pulse rate or heart rate. Tracking body temperature, heart rate and blood pressure is the main thing we do to stay healthy. we use a thermometer and blood pressure monitor to measure your body temperature. Heart rate can be checked in two ways. One is to manually check your heart rate on your wrist or neck, and the other is to use a heart rate monitor. In this project, pulse sensor will be used to measure the heartbeat using the fingers.

When the heart expands (expansion phase), the volume of blood in the fingertips increases, and when the heart contracts (contraction phase), the volume of blood in the fingertips decreases. The resulting pulsation of blood volume at the fingertips is directly proportional to the heart rate and, if the pulse rate per minute can be calculated, it is expressed as the heart rate per minute (bpm). For this, the pair of IR transmitter / receiver (LED) is in close contact with your fingertips.

When a heartbeat occurs, the number of blood cells under the sensor increases, which reflects more infrared waves on the sensor, and when there is no heartbeat, the intensity of the reflected beam decreases. The pulsating reflex is converted by the sensor into a corresponding current or voltage pulse. The output signal of the sensor is processed by suitable electronic circuitry to obtain a visible display (digital display).

CHAPTER 2

LITERATURE SURVEY

1. Microcomputer-based Automatic Heart Rate Counting System Mamun AL, Ahmed N., AL Qahtani (JATIT) Journal of Theory and Applied Technology ISSN 1992-8645: In this research paper, the heart rate signal is transmitted by the finger or ear using the IRTX-RX. Collected from. (Pair of infrared transmitter and receiver) Amplified modules to convert them to observable scale. A low-pass filter was used to filter out the intrinsic noise. These signals were counted by the microcontroller module (ATmega8L) and displayed on the LCD.

The microcontroller is programmed with an algorithm that starts the proposed heart rate counting system. The results obtained through this process were found to be satisfactory compared to the results obtained from manual tests involving heart rate counting. The proposed system is applicable for family, hospital, community medicine, sports health care and other medical purposes. It is also suitable for adults and children. However, this method in the developed systems requires further research, requires more functionality, and may be useful when considering future research results.

2. Heartbeat Detection and Heart Attack Detection with the Internet of Things: IOT Aboobacker sidheeque, Arith Kumar, K. Satish, (IJESCE) International Journal of Engineering and Computing, April 2007: This study demonstrates the use of the Internet to implement heart rate monitoring systems and detecting heart attacks. Recently, the number of heart diseases and heart attacks has increased. The sensor is connected to a microcontroller that allows you to check your heart rate measurements and send them over the Internet.

The user can set the heart rate limitation level. Once these limits are set, the system starts monitoring and the system sends an alert to the controller as soon as the patient's heart rate exceeds certain limits. The controller sends this over the Internet to alert doctors and associated users. The system also alerts you when your heart rate is low. Each time the user logs in to monitor, the system also displays the patient's heart rate. In this way, the patients involved can not only track their heart rate, but also receive immediate heart attack alerts to their patients from anywhere, allowing them to save on time

3. SMS 2009 Alert for Heart Rate Monitoring at the IEEE Industrial Electronics and Applications Symposium, October 4-6, 2009, Kuala Lumpur, Malaysia. Warsuzarina Mat Jubadi, Siti Faridatul Aisyah Mohd Sahak University of Electronics TunHusseinOnn Malaysia BatuPahat, Johor, Malaysia: In this research paper, we show that heart rate can be measured by tracking heart rate with special medical equipment such as: I. As an electrocardiograph (ECG), a portable wristwatch, or other commercially available heart rate monitor. Despite its accuracy, it is somehow expensive, requires many clinical conditions, and the patient must be supervised by a healthcare professional for constant observation. Patients who have already been diagnosed with fatal heart disease need to constantly monitor their heart rate. In this article, we propose an alarm system that can track the heart rate status of patients. The heart rate is determined using photoprethysmograph (PPG) technology. This signal is processed by the PIC16F87 microcontroller to determine the heart rate per minute. Then send SMS notification to the mobile phone of the healthcare professional, patient's family members or their relatives using SMS. He also warns the family to respond quickly to the patient.

CHAPTER 3

EXISTISING SYSTEM AND PROBLEM STATEMENT

heart rate monitoring itself is not a technology that needs to be developed from scratch; many products available now use Electrocardiography, Photoplethysmography, Oscillometry (Blood pressure monitor method). For example, there are companies such as "tricog" and "alivecor" who's products focuses entirely on using Electrocardiography to measure and monitor the heart rate. the smart watches such as the apple i series and Samsung smart watches uses photoplethysmography to measure the heart rate. then there are products from OMRON which uses Oscillometry to measure and monitor heart rate.

so there are a lot of products in the market which uses different methodologies and principles that can measure and monitor heart rate and alert you in case of an heart attack

It has been shown that heart rate can be measured by tracking the pulse using special medical equipment such as an electrocardiograph (ECG), portable wristwatch, or other commercially available heart rate meters. Despite its accuracy, it is somehow expensive, requires many clinical conditions, and the patient must be supervised by a healthcare professional for constant observation. Patients who have already been diagnosed with fatal heart disease need to constantly monitor their heart rate.

The company Symposium on Industrial Electronics and Applications had proposed an alert system that can track a patient's heart rate status. The heart rate is determined using photoprethysmograph (PPG) technology. This signal is processed by the PIC16F87 microcontroller to determine the heart rate per minute. Then send SMS notification to the mobile phone of the healthcare professional, patient's family members or their relatives using SMS. He also warns the family to respond quickly to the patient.

PROBLEM STATEMENT:

As you know, Heart rate monitor is essential for monitoring patients, especially in the intensive care unit (ICU). Therefore, although the demand for Heart rate monitors is high, various problems arise such as a lack of hospital space, and costly maintenance is required for wiring and installation. The problems can be solved using wireless network to ensure the patients can be monitor continuously by doctors, nurses or caregivers anywhere and anytime even though the patients stay at home. Besides, the costs for wiring and installation might be reducing as well.

CHAPTER 4

4.1. PROPOSED METHODOLOGY

Our method uses a pulse sensor, Arduino board and a Wi-Fi module. After setting up the system, the pulse sensor will start sensing heart rate readings and with the use of Wi-Fi module it will transmit the data over internet.

4.2. BLOCK DIAGRAM

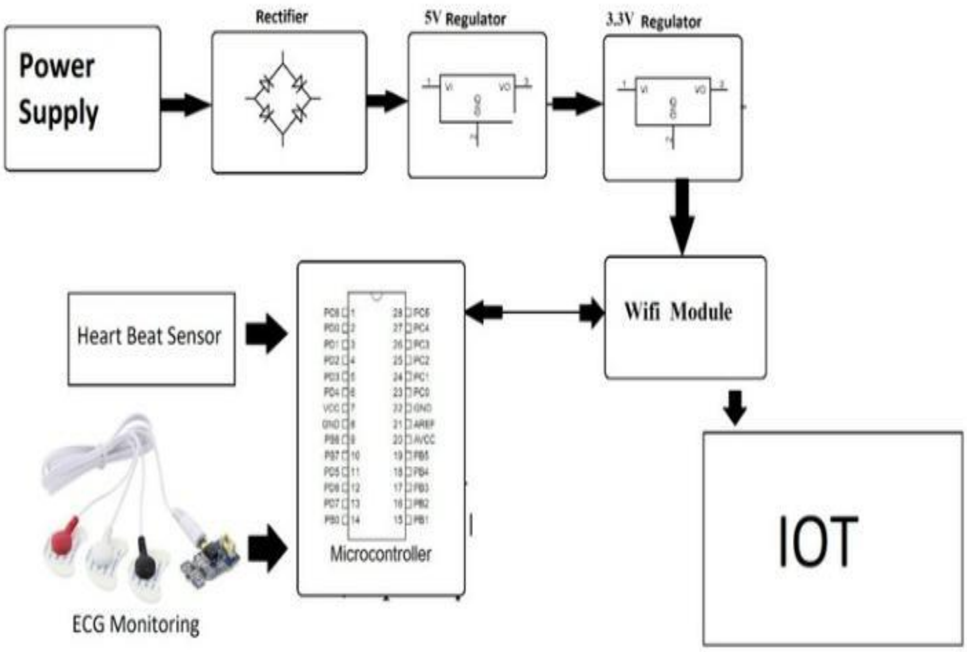


Figure 1

4.3. CIRCUIT DIAGRAM

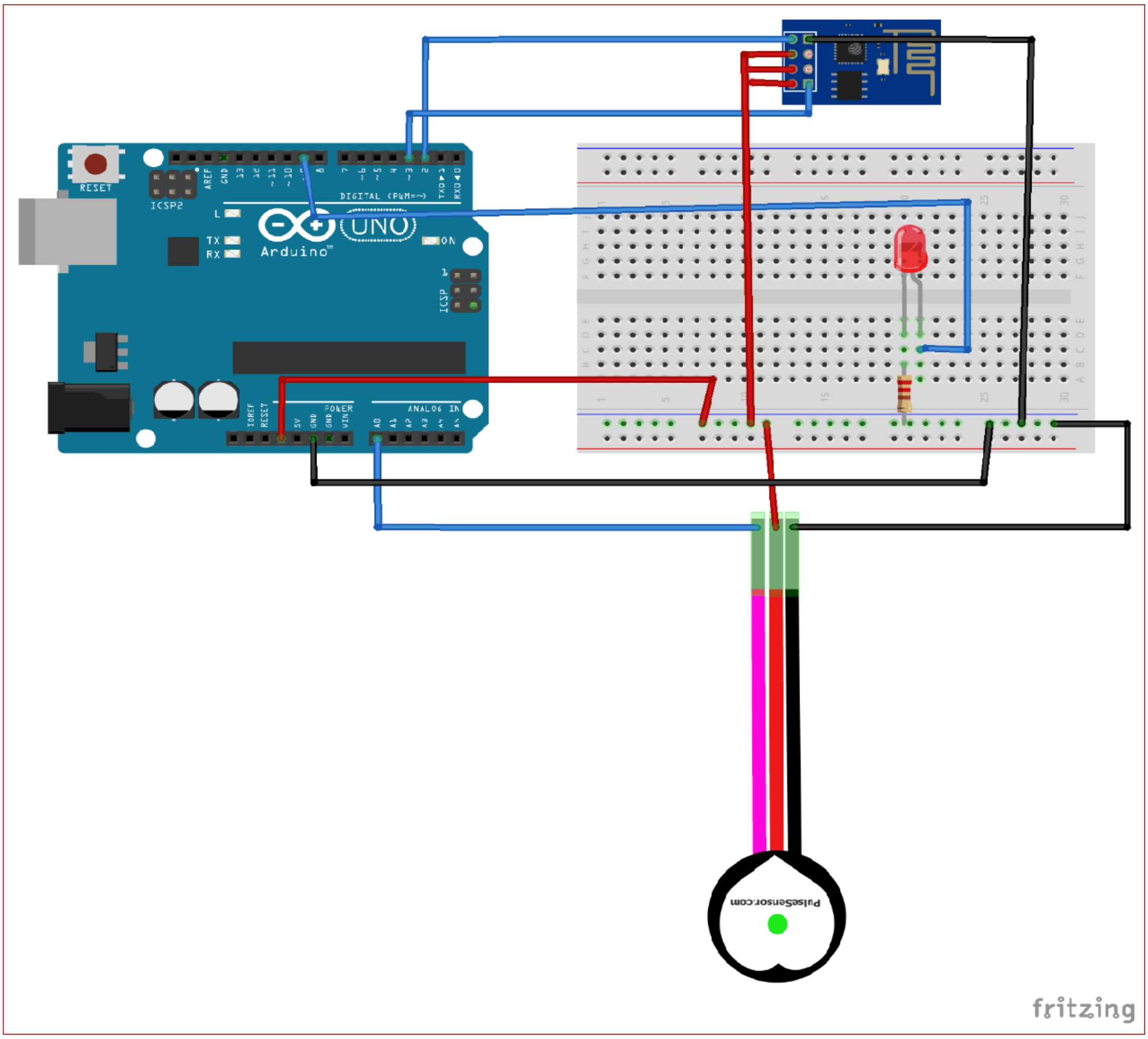


Figure 2

Sl no	Pin Name	Arduino Pin
1	ESP8266 VCC	3.3V
2	ESP8266 RST	3.3V
3	ESP8266 CH-PD	3.3V
4	ESP8266 RX	TX
5	ESP8266 TX	RX
6	ESP8266 GND	GND
7	Pulse Sensor VCC	3.3V
8	Pulse Sensor Signal	A0
9	Pulse Sensor GND	GND
10	Led +ve Pin	9V
11	Led -ve Pin	GND

4.4. WORKING

Upload the code to Arduino UNO and Power on the system. The Arduino asks us to place our finger in the sensor and press the switch.

Place any finger (except the Thumb) in the sensor clip and push the switch (button). Based on the data from the sensor, Arduino calculates the heart rate and displays the heartbeat in bpm. While the sensor is collecting the data, sit down and relax and do not shake the wire as it might result in a faulty values. After the result is displayed on the LCD, if you want to perform another test, just push the rest button on the Arduino and start the procedure once again.

The sensor collects heart rate data and pulses from the human body, the LED blinks according to the pulses. Then the ESP32 sends the data to the ThingSpeak cloud. In order to do that, the board must be connected to a Wi-Fi network. The ThingSpeak platform gives us the flexibility to manage and store data. It also has smart tools and widgets to visualize and analyze data using graphs. It can be used to monitor real-time data and the result can be viewed remotely via a monitor, mobile phone, or laptop.

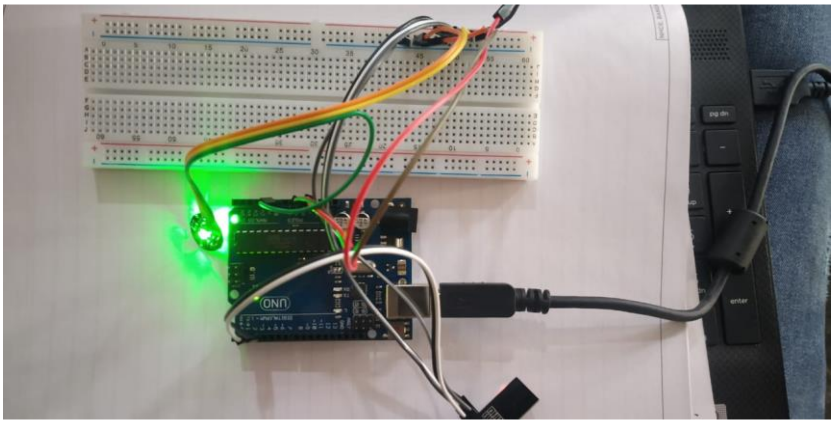


Figure 3

There is three cases in which the heart rate is displayed:

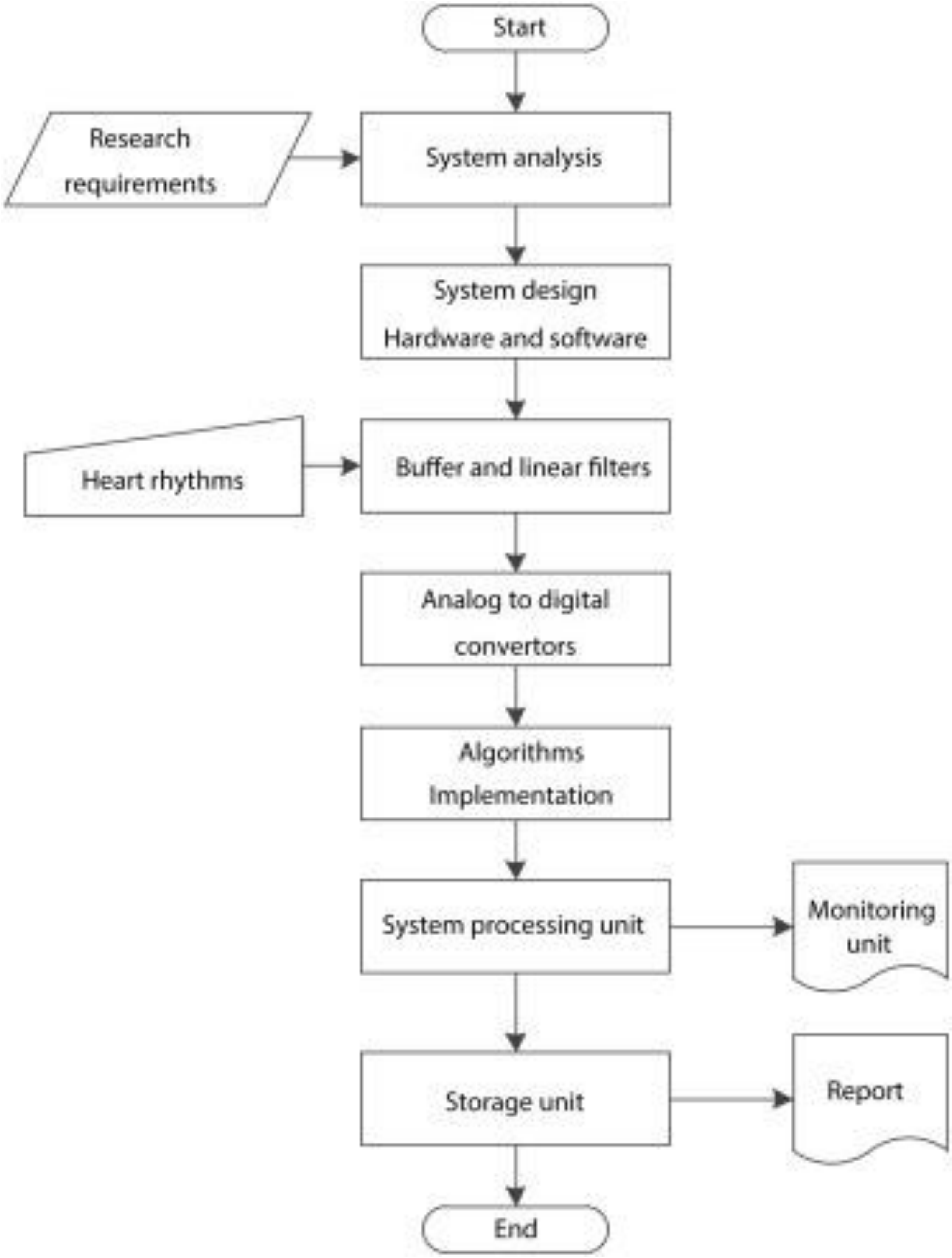
1. Low Pulse Rate: The low pulse rate is displayed when the heart rate per BPM(Beats per minute) is >40 and <60. The low pulse rate may lead to medical complications this indicates that the patient needs the doctors help(ex: Low BP)

2. Normal Pulse Rate: The normal pulse rate ranges is between >60 and <100 which indicates that the patient has the normal range pulse rate with no complication.

3. High Pulse Rate: The high pulse rate is between

   >100 and <150 which indicates the patient has the high pulse range that could result in the heart related diseases(ex: High Blood Pressure)

4.5. FLOW CHART



CHAPTER 5

PROJECT DESCRIPTION

5.1. HARDWARE DESCRIPTION:

1) ARDUINO UNO

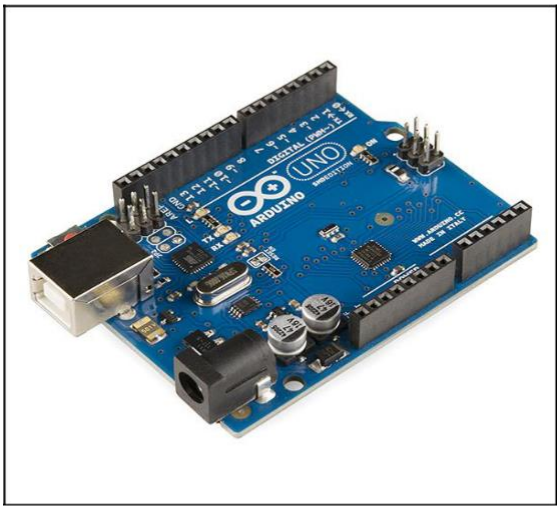


Figure 4

The development panel contains 14 digital I / O ports (six can output PWM), and 6 analog I / O ports, and can be programmed through Arduino IDE (Integrated Development Environment) through a USB Type-B cable, or via a USB battery cable External 9 volts, although it can accept voltages between 7 and 20 volts. It is some were similar to Arduino Nano and also Leonardo.

Material reference design is distributed under an ingenious Commons Attribution Share-Alike 2.5 license and might be found on the Arduino website. Planning and production documents may also be used for specific versions of the fabric.

The word "uno" means "one" in Italian and was chosen because the original version of the Arduino program. Is Uno the primary in a very series of USB-based Arduino boards?] Arduino IDE 1.0 is that the reference version of Arduino and has now evolved into a more modern version. ATmega328 is pre-programmed on the board with a boot loader that may send new code thereto without the utilization of external hardware developers.

The Arduino project started in Ivrea (IDII), Interaction Design Institute, Ivrea, Italy. At the time, students used the BASIC Stamp microcontroller, which was a large fee for many students. In 2003, Hernando Barajan created the wire development platform for the IDII graduate thesis project, which was implemented under the supervision of Massimo Banzi and Casey Rias, and they are known for their editing languages. The purpose of this project is to create simple, low-cost tools for non-engineers to create digital project.

But instead of continuing with the wiring work, they commissioned the project and renamed it Arduino. The first Arduino cards used FTDI USB for Serial Driver chips and ATmega168. Uno is different from all previous motherboards, it has ATmega328P controller and ATmega16U2 (Atmega8U2 to R2) USB serial adapter.

The wire platform consists of an electrical printed circuit (PCB) with ATmega168 controller, the IDE relies on editing and library functions, and will program the microcontroller easily.

General pin functions:

• LED indicator: There is a built-in LED indicator removed from the digital end 13. When the pin is high, the LED indicator lights up; when the pin is low, it turns off.

• VIN: When using an external power source, the input voltage of the Arduino / Genuino development board (as opposed to 5 volts from a USB connection or other adjustable power source). You can save voltage through this pin, or if you supply the voltage through a power outlet, you can access it through this pin.

• 5 V: This adjustable 5-pin pin extracts from the regulator on the board. The board can be powered by DC socket (7-20V), USB socket (5V) or VIN pin (7-20V) for the board. The voltages supplied via the 5V or 3.3V pin will exceed the regulator and the circuit board may be damaged.

• 3V3: 3.3V power generated by the built-in voltage regulator. The maximum current consumption is 50 mA.

• GND: Ground Pin.

• IOREF: This Arduino / Genuino pin provides reference voltage for the microcontroller operation. The correct shielding layer design can read the voltage at the IOREF pin and select the appropriate power source, or allow the voltage transformer at the output to operate at 5V or 3.3V. Reset: Usually used to add the back button to protect things on the board.

• RESET: Usually used to add the back button to protect things on the board.

Special pin functions:

Uno 14 digital and 6 analog pins is used for input or output under program control (using pin function), digital recording () & and digital reading () working at 5 volts. Each pin can provide or receive 20 mAh as the recommended working condition and has an internal fastening resistance of 20-50 km (by default, it is divided). Any I / O station should not exceed 40 mA max to avoid permanent damage to the microcontroller. Uno has 6 analog inputs called A0 to A5. Each provides 10-bit precision (i.e. 1024 different values). By default, actually the measurement range is from ground to 5 volts, but the upper bound of the range can be definitely changed using the AREF pin and the Analog Reference () function.

In addition to this, some pins also have special functions:

• Serial / UART: Used to receive (RX) and send TTL serial data. These pins are attached to the pins corresponding to the ATmega8U2 USB-to-TTL serial chip.

• External interrupt: pins 2 and 3. These pins can be configured to create interruptions when they are low, up, down, or change in value.
• PWM: pins 3, 6, 5, 10, 9, and 11. PWM output can provide 8-bit with AnalogWrite () function.

• SPI (Serial Terminal Interface): pins 10 (SS), 11 (MOSI), 12 (MISO), and 13 (SCK). These pins support a SPI connection using the SPI library.

• TWI / I²C: SDA (A4) Pin and SCL Pin (A5). Support for TWI connections using the Wire Library.

• AREF (analog reference): reference voltage of the analog input.

Communications:

Arduino Uno contains many tools for communicating with the computer, another Arduino / Genuino development board or another microcontroller. ATmega328 provides UART TTL (5V) serial communication, which can be used in 0 (RX) and 1 (TX) digital pins. ATmega16U2 on the board sends this serial connection via USB and displays it as the default software port on the computer. However, in Windows, an .inf file is required. Arduino (IDE) has a serial display that allows you to send simple text data between forms. When transferring data via USB to the serial chip and USB connection to the computer, the RX and TX indicators on the board will flash (but not for serial connections over the years 0 and 1). The serial programs library allows serialization.'

2)PULSE SENSOR

Pulse Sensor is a well-designed plug-and-play heart-rate sensor for Arduino. It can be used by students, artists, athletes, makers, and game & mobile developers who want to easily incorporate live heartrate data into their projects. The sensor clips onto a fingertip or earlobe and plugs right into Arduino with some jumper cables. It also includes an open-source monitoring app that graphs your pulse in real time.

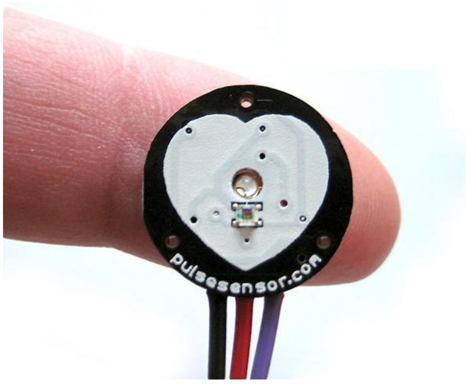


Figure 5

3)NodeMCU

NodeMCU is an open source firmware which used for IoT and uses CPU-ESP8266. Which has memory of 128kB and Storage 4MB, it 32-bit, CPU operating voltage is 3.3V, it has one analog input and 16 general purpose input-output pins, it contains SPI, UART and I2C communication.

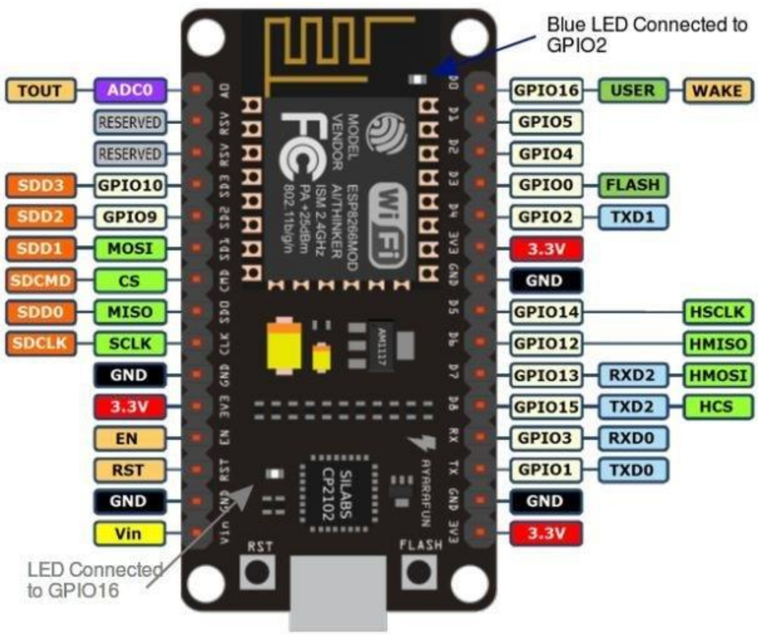


Figure 6

4) LED

LEDs are lights that fit effectively into an electrical circuit and efficient . Be that as it may, in contrast to conventional radiant bulbs, they don't have a fiber that will wear out, and they don't get particularly hot. They are lit up exclusively by the development of electrons in a semiconductor material, and they keep going similarly up to a standard transistor. The life expectancy of a LED outperforms the short existence of a glowing bulb by a huge number of hours.

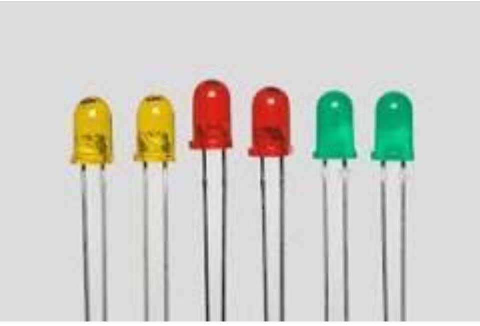


Figure 7

5)Resistor:

Resistors are electronic parts which have a particular, failing to change electrical opposition. The resistor's opposition restricts the progression of electrons through a circuit.

They are aloof segments, which means they just consume power (and can't create it). Resistors are generally added to circuits where they supplement dynamic segments like operation amps, microcontrollers, and other incorporated circuits. Ordinarily resistors are utilized to constrain current, partition voltages, and draw up I/O lines.

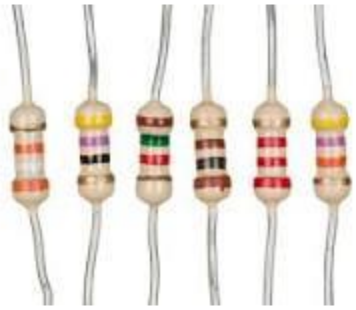


Figure 8

6) Bread Board:

A breadboard is a rectangular board with many mounting gaps. They are utilized for making electrical associations between electronic parts. The associations aren't perpetual and they can be evacuated and set once more. Indeed, you can even supplant segments to redo your venture or work on a totally extraordinary one, utilizing a similar breadboard. The positive rails are shown by red lines, while the negative rails are demonstrated by dark ones.

Figure 9

7) CONNECTING WIRES:

The wire is a single, usually cylindrical, flexible thread or metal rod. Wires are used to carry mechanical loads or electrical and telecommunications signals. The wire is usually formed by drawing the metal through the hole in the die or draw plate. The wire gauges come in various standard sizes as expressed in terms of gauge number. Like "multistrand wire", the term 'wire' is more loosely used to refer to a bundle of such threads, more properly called a wire rope or an electrical cable in mechanics.

8)POWER SOURCE

Choosing an appropriate power source is crucial to ensuring the reliability of the mapping device. In order to fit the needs of this device, the power source must be portable and able to provide output for long periods of time without fully discharging. Different types of batteries were compared and their pros and cons weighed against one another. The power source will evolve with the progression of this project. For the proof-of-concept stage, the device does not necessarily need to be portable. In this stage, the device will be powered through a wired connection to a computer, where the data from the sensors will be sent. In later stages where the communication is wireless, the device will be battery-powered. Single-use alkaline batteries 27 are the optimal choice for this project due to their reliability versus rechargeable batteries. It was decided that alkaline batteries would be used due to their performance in the value analysis. Without having to worry about hydrogen leaks from charging and with the proper heat protection, the threat of an explosion occurring is diminished. The Arduino inside the device requires little power to operate, relative to the stepper motor. The chosen power source for the device is a set of alkaline AA batteries. These batteries provide 12 volts, which is enough to power the servo motor, as well as the other components. The Arduino and the motor shield can easily be powered from this source, so the device only requires a single power source.

SOFTWARE DESCRIPTION:

1)ARDUINO IDE

Figure 10

What is Arduino IDE?

Arduino is basically an open source platform with easy-to-use programs and programs. The Arduino board can actually read the sensor input light - converting it to engine output playback, LED playback and content posting online. By sending a set of commands to the microcontroller on the motherboard, you will tell the motherboard what to do. For many years, from everyday objects to aristocratic scientific instruments, Arduino has been the mastermind of thousands of works. This open source platform brings together a global community of manufacturers, students, hobbies, artists, developers, and professionals. Their contributions add a lot of information and can greatly help novices and experts.

Arduino was born at the Ivrea Interaction Design School, a fast and easy-to-use original tool, especially suitable for college students without an electronic background and programming. After entering the wider community, the Arduino development team started making changes to adapt to new needs and challenges, and expanded its product range from simple 8-bit tables to

3D IoT applications for portable and compact environmental products. All Arduino development boards are completely open source, allowing users to work independently, and ultimately customize them according to their specific needs. The program is also open source and is developed through contributions from users all over the world.

CHAPTER 6

ADVANTAGES AND APPLICATION

6.1. ADVANTAGES

1. Using a heart rate monitor for all activities, what level of effort is required for an individual to complete a specific task and in what situations (e.g. weather, indoor or outdoor workouts, equipment) Or you get clear data on whether it is free flow) ... Fatigue, exposure to drugs and caffeine, sea level, time of day, etc. In other words, what is your body doing and what should it do to achieve the physical feat of the day of your choice?

2. A heart rate monitor is a fantastic tool giving you clear indication and evaluation of the condition of your cardiovascular system during physical activity. Again, awareness is power! This is great information to share when you see your doctor at your yearly physical.

3. Using a heart rate monitor can be your personal trainer. Your heart rate and your "perceived exercise rate" should indicate to you that you need to increase your strength, retreat, or you are in your ditch. I'll tell you if. This will help you narrow down the issues you want to achieve and ensure the best results in the time you spend on your fitness plan to make your workouts safer.

4.Using a heart rate monitor for any activity gives you definite data.

5.Cost effective

6.It gives you clear indication and evaluation of cardio vascular system.

7. Indicates your heart's ability to "recover" from a given exercise and or interval within a workout once again giving you more info on the condition of your cardiovascular system. Faster recovery rate indicates enhanced cardiovascular capacity.

6.2. APPLICATIONS

Heart rate monitors have two major applications. They are used in hospitals to monitor the Heartrate of the patients and they are used in fitness industry.

Medical industry

the heart rate monitors has a vital role in the medical industry as they are the main source of monitoring the pulse rate and heart rate of the patients. it is extremely useful for monitoring the patients' cardiovascular diseases and immediately alert the doctors in case of a stroke or a heart attack. the early alert can help save the life of the patients.
it is also helpful in monitoring the heart rate of the elderly people who have high BP or low BP.

fitness or sports industry

in fitness or sports industry, they are important tools that provide feedback specific to your body. This information can help you design, start and follow a workout routine that is personalized to your body and level of fitness. Measuring your heart rate is the most accurate and objective way to determine the benefit you get from your workout.

In the process of achieving your improved fitness goal, the direct outcome is increased efficiency of your heart. By improving your heart efficiency **you will be able to exercise for longer and faster** with a lower heart rate. And training will feel easier because your heart 'doesn’t have to work as hard

CHAPTER 7

7.1. RESULT AND DISCUSSION

RESULT:

The prototype of the project titled " IoT based heart rate monitor " does give the desired result and output. the data taken from the pulse sensor is processed and the exchange of data between the wifi module and the web application works efficiently over a given interval of time so that the status is regularly updated.

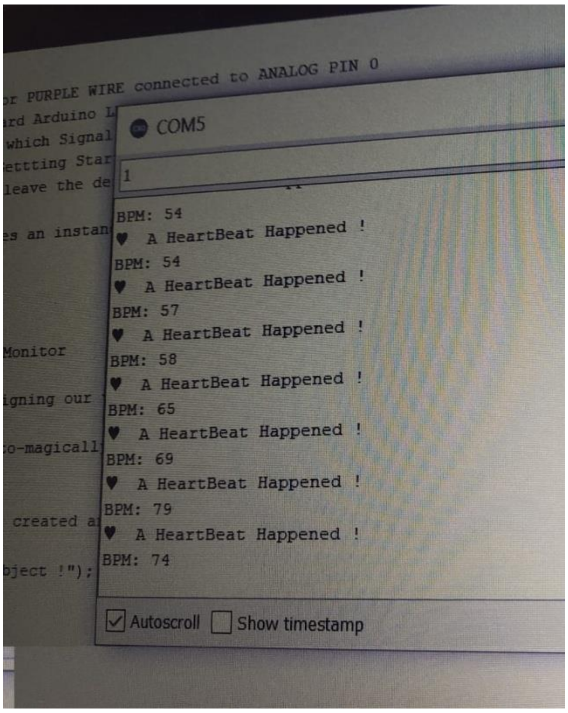


Figure 11

the three cases of results are:

1. Low Pulse Rate: The low pulse rate is displayed when the heart rate per BPM(Beats per minute) is >40 and <60. The low pulse rate may lead to medical complications this indicates that the patient needs the doctors help(ex: Low BP)

2. Normal Pulse Rate: The normal pulse rate ranges is between >60 and <100 which indicates that the patient has the normal range pulse rate with no complication.

3. High Pulse Rate: The high pulse rate is between

   >100 and <150 which indicates the patient has the high pulse range that could result in the heart related diseases(ex: High Blood Pressure)

DISCUSSION:

This prototype has a bi-directional data flow between the NodeMCU and the web application. The microcontroller is connected to the appliances as well as to the web application. it will analyze the pulse rate and send it to the web application. by this data the user can understand the pattern of the heart rate.

CHAPTER 8

FUTURE SCOPE

In future we can upgrade this project in many ways.

1. The current version of the processing application displays the near-time heart rate but does not record anything. Here is a lot of room for improvements. Logging heart rate with respect to time can be useful for analyzing the heart rate pattern throughout the day. by implementing this we can make the heart rate monitor more efficient.

2.The current heart rate monitor is not wearable or portable. it requires some space even if you are carrying it with you. this is a major drawback because you cannot carry it always.
we can make a wearable heart rate monitor such as a watch which is wearable and tracks and monitors the heart rate throughout the day. with this implementation we can also make it a fitness related.

3.another change is that we can make the heart rate monitor plot a graph of the heart rate(beats per minute versus time) this can also help analyzing the heart rate pattern which is extremely helpful to maintain good health.

4. we can also upgrade this project to send immediate messages to the healthcare facility in case of a heart attack using MQTT protocol, and implement location monitoring to this system.

CHAPTER 9

CONCLUSION

The heart beat sensor which is interfaced with microcontroller senses the heartbeat of person and transmits them over internet using Wi-Fi module. System allows setting limits of heart beat. After setting these limits person can start monitoring the heart beat and whenever the person’s heart beat goes above certain set point they can get an alert on high heart beat and also about chances of heart attack. Also the system alerts for lower heartbeat.

In this system a real time heart rate monitoring and heart attack detection system is realized by using IoT. The proposed design is advantageous to patients of different age groups by providing real time heart health monitoring. It also provides security and privacy to the datas of the patient.

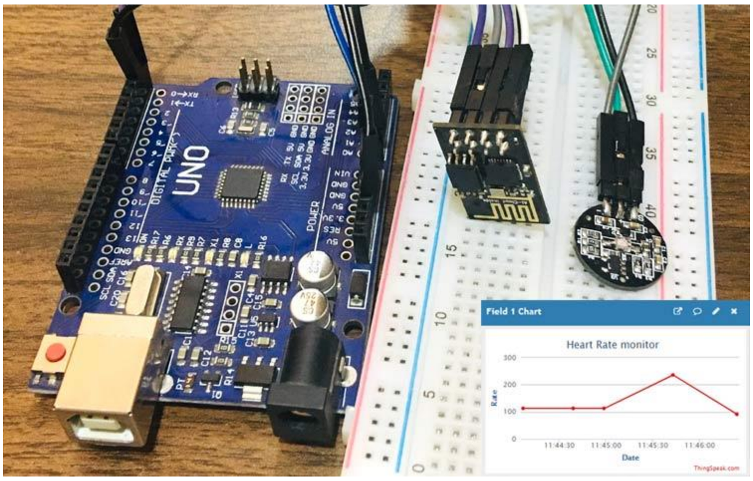


Figure 12

There are many ways that we can expand or improve in the future to achieve this project.

CHAPTER 9

REFRENCE

1. .A microcontroller based automatic heart rate counting system from fingertip Mamun AL, Ahmed N, ALQahtani (JATIT) Journal OF Theory and Applied technology ISSN 1992-8645.

2. Heartbeat and Temperature Monitoring System for remote patients using Arduino Vikram Singh, R. Parihar, Akash Y TangipahoaD Ganorkar (IJAERS), International Journal of Advanced Engineering and Science eissn2349-6495.

3. A GSM Enabled Real time simulated Heart Rate Monitoring and control system Sudhindra F, Anna Rao S.J, (IJRET) International Journal of Research In Engineering And Technology, e ISSN 2319-3163.

4. Heart beat Sensing and Heart Attack Detection Using internet of things: IOT Aboobacker sidheeque, Arith Kumar, K. Sathish,(IJESCE) International Journal Of Engineering Science and Computing, April 2007

5. A Heartbeat and Temperature Measuring System for Remote Health

6. Monitoring using Wireless Body Area Network Mohammad Wajih Alam , Tanin Sultana and Mohammad Sami Alam International Journal of Bio Science and Bio-Technology Vol.8, No.1 (2016)

7. Heartbeat Monitoring Alert via SMS 2009 IEEE Symposium on Industrial Electronics and Applications October 4-6, 2009, Kuala Lumpur, Malaysia. Warsuzarina Mat Jubadi, Siti Faridatul Aisyah

   Mohd Sahak Dept. of Electronics Engineering University Tun Hussein Onn Malaysia Batu Pahat, Johor, Malaysia.

8. J. Allen, Photoplthysmography and its application in clinical physiological measurement, Physiol. Meas, vol. 28, pp. R1 R39,

APPENDIX

Arduino code

```
#include <SoftwareSerial.h>
#include <stdlib.h>
#define DEBUG true
SoftwareSerial ser(2,13);
#define SSID "WiFi Name" // Enter Your WiFi Name Here
#define PASS "WiFi Password" // Enter Your WiFi Password Here
#define IP "api.thingspeak.com"// thingspeak.com ip

String msg = "GET /update?key=Your Api Key"; //Enter your API key

//Variables
int error;
int sensorPin = 0; // Connect Pulse Sensor Signal Pin to Analog Pin A0
int ledpin = 9; // Connect Led Positive Pin to Arduino Pin 9

volatile int BPM; // int that holds raw Analog in 0. updated every 2mS
volatile int Signal; // holds the incoming raw data
volatile int IBI = 600; // int that holds the time interval between beats
volatile boolean Pulse = false; // "True" when heartbeat is detected. "False" when not a
"live beat".
volatile boolean QS = false; // becomes true when Arduino finds a beat.

static boolean serialVisual = true; // Set to 'false' by Default. Re-set to 'true' to see Arduino
Serial Monitor ASCII Visual Pulse
volatile int rate[10]; // array to hold last ten IBI values
volatile unsigned long sampleCounter = 0; // used to determine pulse timing
volatile unsigned long lastBeatTime = 0; // used to find IBI
volatile int P =512; // used to find peak in pulse wave, seeded
volatile int T = 512; // used to find trough in pulse wave, seeded
volatile int thresh = 525; // used to find instant moment of heart beat, seeded
volatile int amp = 100; // used to hold amplitude of pulse waveform, seeded
volatile boolean firstBeat = true; // used to seed rate array so we startup with
reasonable BPM
volatile boolean secondBeat = false; // used to seed rate array so we startup with
reasonable BPM

void setup() {
    Serial.begin(115200); //or use default 115200.
ser.begin(115200);
Serial.println("AT");
ser.println("AT");
delay(3000);
if(ser.find("OK")) {
connectWiFi();
```

```
}
interruptSetup();
}

void loop() {
start: //label
error=0;
updatebeat();
//Resend if transmission is not completed
if (error==1) {
goto start;
}

delay(1000);
}

void updatebeat() {
String cmd = "AT+CIPSTART=\"TCP\",\"";
cmd += IP;
cmd += "\",80";
Serial.println(cmd);
ser.println(cmd);
delay(2000);
if(ser.find("Error")) {
return;
}
cmd = msg ;
cmd += "&field1=";
cmd += BPM;
cmd += "\r\n";
Serial.print("AT+CIPSEND=");
ser.print("AT+CIPSEND=");
Serial.println(cmd.length());
ser.println(cmd.length());
if(ser.find(">")) {
Serial.print(cmd);
ser.print(cmd);
}
else{
Serial.println("AT+CIPCLOSE");
ser.println("AT+CIPCLOSE");
//Resend...
error=1;
}
}

boolean connectWiFi() {
Serial.println("AT+CWMODE=1");
```

```
ser.println("AT+CWMODE=1");
delay(2000);
String cmd="AT+CWJAP=\"";
cmd+=SSID;
cmd+="\",\"";
cmd+=PASS;
cmd+="\"";
Serial.println(cmd);
ser.println(cmd);
delay(5000);
if(ser.find("OK")) {
Serial.println("OK");
return true;
}else {
return false;
}
}

void interruptSetup() {
    // MAKE SURE GLOBAL INTERRUPTS ARE ENABLED
}
ISR (TIMER2_COMPA_vect){                    // triggered when Timer2 counts to 124
  cli();                                    // disable interrupts while we do this
   Signal = analogRead(sensorPin);          // read the Pulse Sensor
  sampleCounter += 2;                       // keep track of the time in mS
 int N = sampleCounter - lastBeatTime;      // monitor the time since the last beat to avoid
noise
       // find the peak and trough of the pulse wave
 if(Signal < thresh && N > (IBI/5)*3){      // avoid dichrotic noise by waiting 3/5 of last IBI
     if (Signal < T){                       // T is the trough
     T = Signal;                            // keep track of lowest point in pulse wave
}
}
  if(Signal > thresh && Signal > P){        // thresh condition helps avoid noise
     P = Signal;                            // P is the peak
  }                                         // keep track of highest point in pulse wave


  if (N > 250){                             // avoid high frequency noise
    if ( (Signal > thresh) && (Pulse == false) && (N > (IBI/5)*3) ){
    Pulse = true;                           // set the Pulse flag when there is a pulse
        digitalWrite(ledpin,HIGH);          // turn on pin 13 LED
      IBI = sampleCounter - lastBeatTime;   // time between beats in mS
     lastBeatTime = sampleCounter;          // keep track of time for next pulse
       if(secondBeat){                      // if this is the second beat
          secondBeat = false;               // clear secondBeat flag
    for(int i=0; i<=9; i++){                // seed the running total to get a realistic BPM at startup
                rate[i] = IBI;
```

```
}
}
if(firstBeat){                                   // if it's the first time beat is found
firstBeat = false;                               // clear firstBeat flag
secondBeat = true;                               // set the second beat flag
sei();                                           // enable interrupts again
return;                                          // IBI value is unreliable so discard it
}
word runningTotal = 0;                           // clear the runningTotal variable
for(int i=0; i<=8; i++){                         // shift data in the rate array
rate[i] = rate[i+1];                             // and drop the oldest IBI value
runningTotal += rate[i];                         // add up the 9 oldest IBI values
}
rate[9] = IBI;                                   // add the latest IBI to the rate array
runningTotal += rate[9];                         // add the latest IBI to runningTotal
runningTotal /= 10;                              // average the last 10 IBI values
BPM = 60000/runningTotal;                        // how many beats can fit into a minute? that's
BPM!
QS = true;                                       // set Quantified Self flag

}
}
if (Signal < thresh && Pulse == true){   // when the values are going down, the beat is over
digitalWrite(ledpin,LOW);                        // turn off pin 13 LED
Pulse = false;                                   // reset the Pulse flag so we can do it again
amp = P - T;                                     // get amplitude of the pulse wave
thresh = amp/2 + T;                              // set thresh at 50% of the amplitude
P = thresh;                                      // reset these for next time
T = thresh;
}
if (N > 2500){                                   // if 2.5 seconds go by without a beat
thresh = 512;                                    // set thresh default
P = 512;                                         // set P default
T = 512;                                         // set T default
lastBeatTime = sampleCounter;                    // bring the lastBeatTime up to date
firstBeat = true;                                // set these to avoid noise
secondBeat = false;                              // when we get the heartbeat back
}
sei();

}
```