A MINI PROJECT REPORT ON

BUILDING AN ARDUINO-POWERED ECG MONITOR AND ALERTING SYSTEM

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF THE DEGREE OF

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

SUBMITTED BY

BOORA SREEJA **217R1A0407**

UNDER THE GUIDANCE OF

Dr. SUDHA ARVIND

ASSOCIATE PROFESSOR, ECE

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CMR TECHNICAL CAMPUS

Academic Year 2024-2025

DEPT. OF ELECTRONICS & COMMUNICATION ENGINEERINGCMRTECHNICALCAMPUS
Accredited by NBA &NAAC with A Grade
Kandlakoya (V), MedchalRoad, Hyderabad-501401

CERTIFICATE

This is to certify that the dissertation work entitled **"BUILDING AN ARDUINO-POWERED ECG MONITOR AND ALERTING SYSTEM"** being submitted by **BOORA SREEJA (217R1A0407)** in partial fulfillment of the requirement for the award of the degree of **Bachelor of Technology in Electronics and Communication Engineering of JNTU**, **Hyderabad** during the academic year 2024–2025.

Certified further, to the best of our knowledge that the work reported is not a part of any other project on the basis of which a degree or an award has been given on an earlier occasion to any other candidate. The results have been verified and found to be satisfactory.

Internal Guide | ***Head of the Department***

Dr. SUDHA ARVIND
Associative Professor, ECE Dept.

Mr.S.MALLESH
PROFESSOR, ECE Dept.

External Examiner

Director
Dr A. RAJI REDDY

ICASMMT 2024

MLRIT
Raising Engineers

CERTIFICATE OF PRESENTATION

This certificate is presented to

Gopagoni Deepika

in recognition and appreciation of contribution as an author of the paper entitled

Building an Arduino Powered ECG Monitor and Alerting System

at the first International Conference on Advancements in Sustainable Energy, Materials & Manufacturing Technology (**ICASMMT 2024**) during **13th & 14th September, 2024** organized by the Department of Mechanical Engineering, MLR Institute of Technology, Hyderabad, Telangana, India.

Dr. Balram Yelamasetti
Convener,
ICASMMT-2024

Dr. K. Srinivas Rao
Conference Chair
ICASMMT-2024

AIP Publishing AIP CONFERENCE PROCEEDINGS | edp sciences MATEC WEB OF CONFERENCES
MATERIALS SCIENCE FORUM | KEY ENGINEERING MATERIALS

Clarivate Web of Science

Scopus

ICASMMT 2024

MLRIT
Raising Engineers

CERTIFICATE OF PRESENTATION

This certificate is presented to

Thadkapally Charan Kumar

in recognition and appreciation of contribution as an author of the paper entitled

Building an Arduino Powered ECG Monitor and Alerting System

at the first International Conference on Advancements in Sustainable Energy, Materials & Manufacturing Technology (**ICASMMT 2024**) during **13th & 14th September, 2024** organized by the Department of Mechanical Engineering, MLR Institute of Technology, Hyderabad, Telangana, India.

Dr. Balram Yelamasetti
Convener,
ICASMMT-2024

Dr. K. Srinivas Rao
Conference Chair
ICASMMT-2024

AIP Publishing AIP CONFERENCE PROCEEDINGS | edp sciences MATEC WEB OF CONFERENCES
MATERIALS SCIENCE FORUM | KEY ENGINEERING MATERIALS

Clarivate Web of Science

Scopus

ICASMMT 2024

MLRIT
Raising Engineers

CERTIFICATE OF PRESENTATION

This certificate is presented to

Boora Sreeja

in recognition and appreciation of contribution as an author of the paper entitled

Building an Arduino Powered ECG Monitor and Alerting System

at the first International Conference on Advancements in Sustainable Energy, Materials & Manufacturing Technology (**ICASMMT 2024**) during **13th & 14th September, 2024** organized by the Department of Mechanical Engineering, MLR Institute of Technology, Hyderabad, Telangana, India.

Dr. Balram Yelamasetti
Convener, ICASMMT-2024

Dr. K. Srinivas Rao
Conference Chair ICASMMT-2024

AIP Publishing AIP CONFERENCE PROCEEDINGS | edp sciences MATEC WEB OF CONFERENCES
MATERIALS SCIENCE FORUM | KEY ENGINEERING MATERIALS

Clarivate Web of Science

Scopus

ICASMMT 2024

MLRIT
Raising Engineers

CERTIFICATE OF PRESENTATION

This certificate is presented to

Garlapati Karthikeya Reddy

in recognition and appreciation of contribution as an author of the paper entitled

Building an Arduino Powered ECG Monitor and Alerting System

at the first International Conference on Advancements in Sustainable Energy, Materials & Manufacturing Technology (**ICASMMT 2024**) during **13th & 14th September, 2024** organized by the Department of Mechanical Engineering, MLR Institute of Technology, Hyderabad, Telangana, India.

Dr. Balram Yelamasetti
Convener,
ICASMMT-2024

Dr. K. Srinivas Rao
Conference Chair
ICASMMT-2024

AIP Publishing AIP CONFERENCE PROCEEDINGS | edp sciences MATEC WEB OF CONFERENCES
MATERIALS SCIENCE FORUM | KEY ENGINEERING MATERIALS

Clarivate Web of Science

Scopus

CMR TECHNICAL CAMPUS
EXPLORE TO INVENT

NATIONAL INSTITUTIONAL RANKING FRAMEWORK

UGC AUTONOMOUS

ACCREDITED WITH GRADE A NAAC

INSTITUTION'S INNOVATION COUNCIL (Ministry of Education Initiative)

NATIONAL BOARD of ACCREDITATION

SIH SMART INDIA HACKATHON 2024

CERTIFICATE

OF APPRECIATION

TECHNOVANZA 4.0

This certificate is proudly presented to

GOPAGONI DEEPIKA

of Team **SMART SQUARE** for being selected in " Technovanza-4.0" Internal Hackathon for Smart India Hackathon, 2024 held at CMR Technical Campus Hyderabad on 6th September 2024..

Dr. K MURALI SPOC SIH , CMRTC	**Dr. K MAHESWARI** Programme Convener, CMRTC	**Dr. SUDHA ARVIND** Convener IIC ,CMRTC	Areddy **Dr. A. RAJI REDDY** Director, CMRTC

CMR TECHNICAL CAMPUS
EXPLORE TO INVENT

NATIONAL INSTITUTIONAL RANKING FRAMEWORK

UGC AUTONOMOUS

ACCREDITED WITH GRADE A NAAC

INSTITUTION'S INNOVATION COUNCIL (Ministry of Education Initiative)

NATIONAL BOARD OF ACCREDITATION

SIH SMART INDIA HACKATHON 2024

CERTIFICATE

OF APPRECIATION

TECHNOVANZA 4.0

This certificate is proudly presented to

THADKAPALLY CHARAN KUMAR

of Team **SMART SQUARE** for being selected in " Technovanza-4.0" Internal Hackathon for Smart India Hackathon, 2024 held at CMR Technical Campus Hyderabad on 6th September 2024..

Dr. K MURALI SPOC SIH , CMRTC	**Dr. K MAHESWARI** Programme Convener, CMRTC	**Dr. SUDHA ARVIND** Convener IIC ,CMRTC	Arreddy **Dr. A. RAJI REDDY** Director, CMRTC

CMR TECHNICAL CAMPUS
EXPLORE TO INVENT

NATIONAL INSTITUTIONAL RANKING FRAMEWORK

UGC AUTONOMOUS

ACCREDITED WITH GRADE A NAAC

INSTITUTION'S INNOVATION COUNCIL (Ministry of Education Initiative)

NATIONAL BOARD OF ACCREDITATION

SIH SMART INDIA HACKATHON 2024

CERTIFICATE
OF APPRECIATION

TECHNOVANZA 4.0

This certificate is proudly presented to

BOORA SREEJA

of Team **SMART SQUARE** for being selected in " Technovanza-4.0" Internal Hackathon for Smart India Hackathon, 2024 held at CMR Technical Campus Hyderabad on 6th September 2024..

Dr. K MURALI SPOC SIH , CMRTC	**Dr. K MAHESWARI** Programme Convener, CMRTC	**Dr. SUDHA ARVIND** Convener IIC ,CMRTC	**Dr. A. RAJI REDDY** Director, CMRTC

CMR TECHNICAL CAMPUS
EXPLORE TO INVENT

SMART INDIA HACKATHON 2024

CERTIFICATE
OF APPRECIATION

TECHNOVANZA 4.0

This certificate is proudly presented to

GARLAPATI KARTHIKEYA REDDY

of Team **SMART SQUARE** for being selected in " Technovanza-4.0" Internal Hackathon for Smart India Hackathon, 2024 held at CMR Technical Campus Hyderabad on 6th September 2024..

Dr. K MURALI	Dr. K MAHESWARI	Dr. SUDHA ARVIND	Dr. A. RAJI REDDY
SPOC SIH , CMRTC	Programme Convener, CMRTC	Convener IIC ,CMRTC	Director, CMRTC

ACKNOWLEDGEMENT

I thank **Dr. SUDHA ARVIND**, Associate Professor, Department of ECE for constant guidance over the past six months, and for providing the required guidance as an internal guide for the result- oriented implementation of ideas relevant to my project.

I thank **Dr. SUDHA ARVIND**, Mini Project Coordinator for her constant support in carrying out the project activities and reviews.

I thank **Mr.S.MALLESH**, HOD of Electronics and Communication Engineering for providing seamless support and knowledge over the past year and for providing the right suggestions at every phase for the successful completion of my project.

I express my whole-hearted gratitude to **Dr. A. RAJI REDDY**, Director, CMR Technical campus for providing us with a conducive environment for carrying out my academic schedules and projects with ease. There is a need to thank all staff members, friends and parents whose supported the project would have been deferred.

217R1A0407 BOORA SREEJA

ABSTRACT

Introduction:-

ECG, which stands for Electrocardiogram, is a medical test that measures the electrical activity of the heart over a specific period. It is anon-invasive and painless procedure that involves attaching electrodes tothe skin to record the heart's electrical signals.

Importance of this project:-

ECG, which provides a comprehensive overview of cardiac rhythms and conduction. In addition, ECG interpretation software is built into manymodern ECG systems and contains algorithms.

Implementation Mode:- Hardware and software.

Expected Outcome:-

ECG provides a visual representation of these signals, creating a graph known as an electrocardiogram. Each spike and wave on the graph corresponds to a specific event in the cardiac cycle.

Applications:-

- Fitness and activity heart rate monitors, Gaming peripherals.
- Portable ECG Remote health monitors, Biopotential signal acquisition.

Conclusion:-

From the ECG parameters, analysis a Heart Rate Variability (HRV) of the ECG signal can be implemented. According to the parameters changes comparing to the normal parameters above, the system may predict which disease that the user potentially suffering from.

CONTENTS   Page No

LIST OF FIGURES Page No

LIST OF FIGURES Page No

p

LIST OF TABLES

Page No

LIST OF ABBREVIATIONS

ECG	-	Electrocardiography
IOT	-	Internet of Things
GSM	-	Global system for mobile communication
FPGA	-	Field programmable gate array
API	-	Application programming Interface
LED	-	Light Emitting Diode
IDE	-	Integrated development environment
CPU	-	Central processing unit

CHAPTER 1
INTRODUCTION

1.1 OBJECTIVE

Heart diseases are becoming a big issue for the last few decades and many people die because of certain health problems. Therefore, heart disease cannot be taken lightly. By analyzing or monitoring the ECG signal at the initial stage this disease can be prevented. So we present this project, i.e. ECG Monitoring with AD8232 ECG Sensor & Arduino with ECG Graph.

This project makes use of the AD8232 ECG module to monitor the ECG signal of the patient. APR33A3 voice module is used for voice alerts. ESP8266 WI-FI module is used for monitoring the ECG signal into the thingspeak cloud along withdate and time. The AD8232 is a neat little chip used to measure the electrical activity of the heart. This electrical activity can be charted as an ECG or Electrocardiogram. Electrocardiography is used to help diagnose various heart conditions. The processeddata is then uploaded to the ThingSpeak cloud platform for storage and analysis. In case of abnormal readings or emergencies, the system utilizes the GSM module to send alerts or notifications to caregivers or healthcare providers via SMS. To achievethis task microcontroller loaded programs written in embedded C language. This integrated system enables real-time monitoring of cardiac health remotely, providingtimely interventions when necessary.

The main objective of this project is:

- Design a patient health monitoring system using AD8232 ECG sensor.
- Wireless monitoring using IOT thing speak technology.
- Alert in abnormal conditions using GSM and voice module.
- It is a non-invasive and painless procedure that involves attaching electrodes to the skin to record the heart’s electrical signals.
- This allows them to early detect and promptly treat patients’ conditions, especially in the critical care ward.

1.2 Literature Survey:

While working on our system, it's essential to have knowledge about the existing workor published papers related to our system's design. These resources serve as valuable references for study, helping us analyse productive ways to build and incorporate components that we might not be familiar with additionally . We also need to explore the suitable

Integrated Development Environment (IDE) required for programming microcontroller.

Sohit Agarwal 1, Devashish Dasaya 2 the authors of “IoT-Based ECG Monitoring System for Health Care Applications” which describes about the Internet of Things(IOT) –enchanced ECG monitoring system that can either send data to a server in real time and can be seen in using Smartphone. It was proposed in the year Dec/20022 in Mathematical Statistician and Engineering Applications. Thelimitations for this is GSM module will be integrated into the system&allowing for phone calls and SMS messaging to be made.

Mamoona Humayun (2020) the author of IOT-Based Secure& Energy efficient Scheme for E-health Applications. Which describes about a standardised method oftransmitting patient data across the Internet of Things in E-health contexts within hospitals. It was proposed in the year Aug/2020 in Indian Journal of Science and Technology. The limitations for this is Insecure and call for the implementation of specialized, lightweight security measures due to limitations in available resources.

S. SheebaRani, JafarA.Alzubi,S.K.Lakshamanaprabu,Deepak Gupta,Ramachandran Manikandan the author of Optimal users secure data transmission on the internet of healthcare things (IOHT) with lightweight block ciphers.It was proposed in the yearMay/ 2019 in Multimedia Tools and Applications (Springer) and it describes about research into the challenges of data gathering in IoT-based healthcare apps has led tothe development of a novel healthcare data secure method that ensures the privacy and security of the patient records.

Sarada Prasad Gochhayat, Chhagan Lal, Lokesh Sharma, D. P. Sharma, Deepak Gupta, Jose Antonio Marmolejo Saucedo, Utku Kose the author of Reliable and secure data transfer in IoT networks .It describes about A healthcare CPS that is secure.SLA compliant, and energy efficient to ensure the confidentiality of allcommunications between patients and healthcare providers and the timely fulfillmentof all user demands.

1.2Tabular Form of Literature Survey

Name of the Author	Paper Title	International Journal/conference	Month/year	Description	Limitations
Sohit Agarwal, Devashish Dasaya	Iot-Based ECG Montoring System for Health Care Applications	Mathematical Statistician and Engineering Applications	Dec/20022	Internet of Things(IOT) – enchanced ECG monitoring system that can either send data to a server in real time and can be seen in using Smartphone	GSM module will be integrated into the system&allowing for phone calls and SMS messaging to be made
Mamoona Humayun (2020)	IOT-Based Secure& Energy efficient Scheme for E-health Applications.	Indian Journal of Science and Technology	Aug/2020	A standardised method of transmitting patient data across the Internet of Things in E-health contexts within hospitals	Insecure and call for the implementation of specialized, lightweight security measures due to limitations in available resources.
S. SheebaRani, JafarA.Alzubi,S.K.Lakshamana prabu,Deepak Gupta,Ramachandran Manikandan	Optimal users secure data transmission on the internet of healthcare things(IOHT) with lightweight block ciphers	Multimedia Tools and Applications (Springer)	May/ 2019	Research into the challenges of data gathering in IoT -based healthcare apps has led to the development of a novel healthcare data secure method that ensures the privacy&security of patient records.	Its execution time is far high than that of other cypher algorithms and the model utilized to create shares in the system
Sarada Prasad Gochhayat, Chhagan Lal, Lokesh Sharma, D. P. Sharma, Deepak Gupta, Jose Antonio Marmolejo Saucedo, Utku Kose	Reliable and secure data transfer in IoT networks	Wireless Networks(Springer)	Nov/ 2019	A healthcare CPS that is secure.SLA compliant, and energy efficient to ensure the confidentiality of all communications between patients and healthcare providers and the timely fulfillment of all user demands	Data size will be reduced to make efficient use of network resources, and future research

1.3 Thesis Outline:

The thesis explains the implementation of “Arduino powered ECG Monitor” using an Arduino uno microcontroller. The organization of the thesis is explained here with:

Chapter 1 Presents introduction which includes objective of the project, literature survey (Brief explanation and also in Tabular form) and thesis outline

Chapter 2 Second chapter includes theoretical background, existing and proposed methods

Chapter 3 Third chapter is implementation of the proposed method including Algorithm, flowchart

Chapter 4 Fourth chapter deals with software and hardware description of the project

Chapter 5 Fifth chapter deals with the results and, output waveform discussion.

Chapter 6 Sixth chapter deals with conclusion and future scope.

CHAPTER 2
THEORETICAL BACKGROUND, EXISTING & PROPOSED METHODS

2.1 Embedded Systems:

An embedded system is a computer system designed to perform one or a few dedicated functions often with real-time computing constraints. It is embedded as part of a completedevice often including hardware and mechanical parts. By contrast, a general-purpose computer, such as a personal computer (PC), is designed to be flexible and to meet a widerange of end-user needs. Embedded systems control many devices in common use today.

Embedded systems are controlled by one or more main processing cores that are typicallyeither microcontrollers or digital signal processors (DSP). The key characteristic, however, is being dedicated to handle a particular task, which may require very powerful processors. For example, air traffic control systems may usefully be viewed as embedded, even though they involve mainframe computers and dedicated regional and national networks between airports and radar sites. (Each radar probably includes one or more embedded systems of its own.)

Since the embedded system is dedicated to specific tasks, design engineers can optimize it to reduce the size and cost of the product and increase the reliability and performance.Some embedded systems are mass-produced, benefiting from economies of scale.

Physically embedded systems range from portable devices such as digital watches and MP3 players, to large stationary installations like traffic lights, factory controllers, or thesystems controlling nuclear power plants. Complexity varies from low, with a single microcontroller chip, to very high with multiple units, peripherals and networks mounted inside a large chassis or enclosure. such as the operating systems and microprocessors which power them, but they allow different applications to be loaded and peripherals to be connected. Moreover, even systems which don't expose programmability as a primary feature generally need to support software updates. On a continuum from "general purpose" to "embedded", large application systems will have subcomponents at most points even if the system as a whole is "designed to perform one or a few dedicated functions", and is thus appropriate to call "embedded".

Fig 2.1: Example of embedded system

Figure 2.1 shows the example of embedded system and also the chips present on it like ATMEGA.

Labeled parts include microprocessor (4), RAM (6), flash memory (7).Embedded systems programming is not like normal PC programming. In many ways, programmingfor an embedded system is like programming a PC 15 years ago. The hardware for the system is usually chosen to make the device as cheap as possible. Spending an extra dollara unit in order to make things easier to program can cost millions. Hiring a programmer for an extra month is cheap in comparison. This means the programmer must make do with slow processors and low memory, while at the same time battling a need for efficiency not seen in most PC applications. Below is a list of issues specific to the embedded field.

One of the first recognizably modern embedded systems was the Apollo Guidance Computer, developed by Charles Stark Draper at the MIT Instrumentation Laboratory. At the project's inception, the Apollo guidance computer was considered the riskiest itemin the Apollo project as it employed the then newly developed monolithic integrated circuits to reduce the size and weight. An early mass-produced embedded system was theAutonetics D-17 guidance computer for the Minuteman missile, released in 1961. It wasbuilt from transistor logic and had a hard disk for main memory. When the Minuteman II went into production in 1966, the D-17 was replaced with a new computer that was thefirst high-volume use of integrated circuits.

A. Tools:

Embedded development makes up a small fraction of total programming.There's also a large number of embedded architectures, unlike the PC world where 1 instruction set rules, and the UNIX world where there's only 3 or 4 major ones. This means that the tools are more expensive. It also means that they're lowering features, and less developed. On a major embedded project, at some point you will almost alwaysfind a compiler bug of some sort.

Debugging tools are another issue. Since you can't always run general programs on yourembedded processor, you can't always run a debugger on it. This makes fixing your program difficult. Special hardware such as JTAG ports can overcome this issue in part. However, if you stop on a breakpoint when your system is controlling real world hardware (such as a motor), permanent equipment damage can occur. As a result, people doing embedded programming quickly become masters at using serial IO channels anderror message style debugging.

B. Resources:

To save costs, embedded systems frequently have the cheapest processorsthat can do the job. This means your programs need to be written as efficiently as possible. When dealing with large data sets, issues like memory cache misses that nevermatter in PC programming can hurt you. Luckily, this won't happen too often- use reasonably efficient algorithms to start, and optimize only when necessary of course, normal profilers won't work well, due to the same reason debuggers don't work well

Memory is also an issue. For the same cost savings reasons, embedded systems usually have the least memory they can get away with. That means their algorithms must be memory efficient (unlike in PC programs, you will frequently sacrifice processor time for memory, rather than the reverse). It also means you can't afford to leak memory. Embedded applications generally use deterministic memory techniques and avoid the default "new" and "malloc" functions, so that leaks can be found and eliminated more easily. Other resources programmers expect may not even exist. For example, most embedded processors do not have hardware FPUs (Floating- Point Processing Unit).

Real Time Issues:

Embedded systems frequently control hardware, and must be able to respond to them inreal time. Failure to do so could cause inaccuracy in measurements, or even damage hardware such as motors. This is made even more difficult by the lack of resources available. Almost all embedded systems need to be able to prioritize some tasks over others, and to be able to put off/skip low priority tasks such as UI in favor of high prioritytasks like hardware control.

2.2 Electrocardiography (ECG) Fundamentals

Heart's Electrical Activity: Understanding how the heart generates electrical signals thatcan be detected on the skin. These signals are due to depolarization and repolarization of heart muscle cells

ECG Waveform: Familiarity with the P wave, QRS complex, and T wave,which represent different stages of the cardiac cycle.

Lead Placement: Knowledge of the standard placement of electrodes (e.g., Lead I, II,III, and augmented leads) to capture accurate ECG signals.

A. Signal Acquisition and Processing

Electrodes: Using conductive materials (like Ag/AgCl electrodes) to detect electrical signals from the skin.

Amplification: Since the electrical signals from the heart are very small (microvolts to millivolts), they need to be amplified using an instrumentation amplifier.

Filtering: Removing noise and interference from the signal using filters (e.g., low-pass, high-pass, band-pass, and notch filters) to obtain a clean ECG waveform.

Analog-to-Digital Conversion (ADC): Converting the analog ECG signal to a digital form using the Arduino's ADC for further processing and analysis.

B. Arduino Microcontroller

Hardware: Understanding the specifications and capabilities ofthe Arduino microcontroller, including its ADC resolution, sampling rate, and available memory.

Programming: Writing code in the Arduino IDE to read the ECG signal, process it, and implement the alerting mechanism.

Libraries and Shields: Utilizing libraries (such as the AD8232 library for ECG sensors) and shields (additional hardware modules) that simplify the development process.

C. Data Processing and Analysis

Signal Processing: Implementing algorithms to detect key features of the ECG waveform (e.g., R-peaks) and calculate parameters like heart rate.

Pattern Recognition: Using methods to identify abnormal patterns or arrhythmias in the ECG signal that could indicate cardiac issues.

Real-Time Processing: Ensuring that the system can process data in real-time to provide timely alerts.

D. Alerting Mechanism

Threshold Setting: Defining thresholds for parameters like heart rate or detected arrhythmias that will trigger an alert.

Communication: Implementing communication methods (e.g., Bluetooth, Wi-Fi, GSM) to send alerts to a caregiver or medical professional.

User Interface: Creating a simple interface (e.g., LCD display, smartphone app) to display real-time ECG data and alerts.

E. Power Management

Power Supply: Designing a reliable power supply for the system, which may include battery management if the system is portable.

Energy Efficiency: Optimizing the system for low power consumption, which is crucial for portable and long-term monitoring applications.

F. Prototype Development and Testing

Prototyping: Building and testing the initial prototype to ensure it functions correctly. **Validation**: Comparing the system's performance against standard ECG machines to validate its accuracy and reliability.

Iterative Improvement: Continuously refining the design based on testing and feedback. Combining these theoretical concepts with practical implementation will result in an effective and reliable Arduino-powered ECG monitoring and alerting system.

2.3 Explanation of Embedded Systems:

Software Architecture:

There are several different types of software architecture in common use.

- **Simple Control Loop:**

In this design, the software simply has a loop. The loop calls subroutines, each of which manages a part of the hardware or software.

- **Interrupt Controlled System:**

Some embedded systems are predominantly interrupt controlled. This means that tasks performed by the system are triggered by different kinds of events.

An interrupt could be generated for example by a timer in a predefined frequency, or by a serial port controller receiving a byte.

These kinds of systems are used if event handlers need low latency and the event handlers are short and simple.

Usually these kinds of systems run a simple task in a main loop also, but this task is not very sensitive to unexpected delays. Sometimes the interrupt handler will add longer tasks to a queue structure.

Later, after the interrupt handler has finished, these tasks are executed by the main loop. This method brings the system close to a multitasking kernel with discrete processes

- **Cooperative Multitasking:**

A non-preemptive multitasking system is very similar to the simple control loop scheme, except that the loop is hidden in an API.

The programmer defines a series of tasks, and each task gets own environment to "run" in. When a task is idle, it calls an idle routine, usually called "pause", "wait", "yield", "nop" (stands for no operation), etc.

The advantages and disadvantages are very similar to the control loop, except that adding new software is easier, by simply writing a new task.

A. Stand Alone Embedded System:

These systems take the input in the form of electrical signals from transducers or commands from human beings such as pressing of a button etc.., process them and produce desired output.

This entire process of taking input, processing it and giving output is done in standalone mode. Such embedded systems comes under stand alone embedded systems

Eg: microwave oven, air conditioner etc..

B. Real-time embedded systems:

Embedded systems which are used to perform a specific task or operation in a specific time period are called real-time embedded systems. There are two types of real-time embedded systems.

Hard Real-time embedded systems:

These embedded systems follow an absolute deadline time period i.e.., if the tasking is not done in a particular time period then there is a cause of damage to the entire equipment.

Eg: consider a system in which we have to open a valve within 30 milliseconds. If thisvalve is not opened in 30 ms this may cause damage to the entire equipment. So in such cases we use embedded systems for doing automatic operations.

2.4 Existing Method:

Fig 2.4 Communication System

Figure 2.4 shows the communication system by using firebase, mobile app, ESP8266, Things and also web application

The process is automated with the help of our suggested solution. The node MCU's ESP8266 microcontroller collects data from a number of sensors, including those that measure temperature, humidity, and vital signs. Read both temperature and humidity with the DHT11 Temperature and Humidity Sensor. Accurate environmental monitoring is possible thanks to the use of a capacitive humidity sensor and a thermistor-based temperature sensor. Data (I/O) - Digital serial Data Output, Ground (VCC), and Power (GND) are its three pins. Microcontroller: As the microcontroller for the node, theESP8266 is used. A platform for creating Internet of Things applications that use the free and open-source Lua programming language.

The Wi-Fi SOC it uses is the ESP8266, andit comes preloaded with firmware that communicates with the Thing Speak cloud and makes data accessible via mobile app. The "Power" pins are the first of various varieties. A USB port, a 3.3V power input pin, a ground pin, and an external power input pin labelledVin are all included on the Micro-USB Node MCU. Second, pressing this button will resetthe microcontroller (control pins EN, RST).

The pulse sensor, to round things off, makes use of Analog pin A0's capacity to measure Analog voltage. Sensor data is continuously collected by a microcontroller and then transmitted to the cloud and various mobile applications. When the data is collected, the Node MCU microcontroller with built-in WIFIwill send it to the Thing Speak cloud. Users using Android and IOS devices can use the Mobile App that provides this data.

2.5 Proposed Method:

- The ECG monitoring system presented combine electrocardiogram (ECG) sensing technology with a voice module, ESP8266 microcontroller, ThingSpeak cloud platform, and GSM communication capability.
- The ECG sensor captures heart activity, which is processed by the ESP8266 microcontroller and transmitted to the ThingSpeak cloud platform for storage and analysis.
- If there is any Abnormalities in ECG readings trigger alerts via the GSM module, allowing for immediate intervention.
- Additionally, the system in corporates a voice module to facilitate two-way communication between patients and healthcare providers

Global System for Mobile Communication (GSM) Definition:

GSM, which stands for Global System for Mobile communications, reigns (important) asthe world’s most widely used cell phone technology. Global system for mobile communication (GSM) is a globally accepted standard for digital cellular communication.GSM is the name of a standardization group established in 1982 to create a common European mobile telephone standard that would formulate specifications fora pan-European mobile cellular radio system operating at 900 MHz. It is estimated that many countries outside of Europe will join the GSM partnership.When the system detectsany abnormal condition, it will send the alert SMS to the predefine mobile number via GSM.

Fig 2.6(a) GSM Module

Figure 2.6(a) shows GSM module typically includes a SIM card slot, antenna connectors and various input & output pins for integration

APR33A3 Voice Module:

Total 11 minutes of recording time each channel (M0toM7) having 1.3 minutes ofrecording time.

Single chip, high quality voice recording and play back solution,low cost

When the system detects any abnormal condition, it will give the voice alert through speaker.

APR33A3 voice module is a type of audio recording and playback module used in various electronics projects. It can store and play back audio messages, making it useful for applications such as greeting cards, talking toys, and voice prompts.

Key features of the APR33A3 voice module include:

Multiple messages: It can store multiple audio messages, with the exact number andlength depending on the memory capacity.

Easy interface: Typically uses simple buttons or external triggers for recording and playback.

Playback modes: Various playback options such as looped playback or one-timeplayback.

Fig 2.6(b) APR33A3 Voice Module

Figure-2.6(b) shows APR33A3 voice module, mainly used for audio recording and playback in various projects.

CHAPTER 3

IMPLEMENTATION OF PROPOSED METHOD

3.1Implementation:

3.1(a) ECG:

Electrocardiogram refers to the recording of electrical changes that occur in the heart during a cardiac cycle. It may be abbreviated as ECG or EKG.

Fig 3.1: Electrical Signals of heart in the form of graph

Figure 3.1 shows the electrical signals of heart in the graph in the lcd or in any cloud Components and Waveforms

The ECG is composed of several key components:

P wave: Represents atrial depolarization, which leads to the contraction of the atria.**QRS complex:** Represents ventricular depolarization and consists of three parts:

Q wave: The first negative deflection after the P wave.R wave: The first positive deflection after the Q wave.S wave: The first negative deflection after the R wave.

T wave: Represents ventricular repolarization, which is the recovery phase when the ventricles are preparing for the next contraction.

U wave: Sometimes seen following the T wave, though its exact origin is not entirely understood.

Intervals and Segments

PR interval: The time from the onset of the P wave to the start of the QRS complex. It reflects the time the electrical impulse takes to travel from the atria to the ventricles.

QT interval: The time from the start of the QRS complex to the end of the T wave. It represents the total time for ventricular depolarization and repolarization.

ST segment: The flat, isoelectric section of the ECG between the end of the S wave and the start of the T wave. This segment can indicate myocardial ischemia or infarction if it is elevated or depressed.

Leads and Views

ECGs are recorded from multiple leads, each providing a different view of the heart's electrical activity.

The standard 12-lead ECG includes:

Limb leads: I, II, III, a VR, a VL, a VF

Precordial (chest) leads: V1, V2, V3, V4, V5, V6

Each lead offers a unique perspective, allowing for comprehensive analysis of the heart's electrical activity from different angles. Interpretation ECG interpretation requires careful analysis of the waveforms, intervals, and segments in the context of the clinical presentation. It involves looking for patterns, assessing the regularity of the heart rhythm, measuring intervals, and identifying abnormalities. Interpretation can be complex and often requires expertise in cardiology.

A. Connect the ECG Sensor:

RA (Right Arm) electrode to A0: Connect the output pin of the AD8232 to the analogpin A0 on the Arduino.

LA (Left Arm) and RL (Right Leg) electrodes: Attach the electrodes to the appropriate places on the body.

GND and 3.3V: Connect the sensor's ground (GND) to the Arduino's GND and the sensor’s power (3.3V) to the Arduino’s 3.3V.

B. Interfacing the ESP8266 Wi-Fi Module:

ESP8266 TX to Arduino RX: Connect the TX pin of the ESP8266 to the RX pin of the Arduino.

ESP8266 RX to Arduino TX: Connect the RX pin of the ESP8266 to the TX pin of the Arduino.

ESP8266 VCC to Arduino 3.3V: Power the ESP8266 with 3.3V from the Arduino. ESP8266 GND to Arduino GND: Connect the ground of the ESP8266 to the Arduino's ground.

Software Configuration:

Install the ESP8266WiFi library in your Arduino IDE.

Configure the Wi-Fi credentials and connect to ThingSpeak.

C. Integrate the GSM Module:

GSM TX to Arduino RX: Connect the TX pin of the GSM module to the RX pin of the Arduino.

GSM RX to Arduino TX: Connect the RX pin of the GSM module to the TX pin of the Arduino.

GSM VCC to Arduino 5V: Power the GSM module with 5V from the Arduino.

GSM GND to Arduino GND: Connect the ground of the GSM module to the Arduino's ground.

Software Configuration:

Install the Software Serial library to communicate with the GSM module.

Write functions to send SMS alerts using AT commands.

D. Connect the APR33A3 Voice Module:

Voice Module Control Pins to Arduino Digital Pins: Follow the specific wiring diagramfor your voice module.

Voice Module VCC to Arduino 5V: Power the voice module with 5V from the Arduino. Voice Module GND to Arduino GND: Connect the ground of the voice module to the Arduino's ground.

Write functions to play pre-recorded voice messages when an abnormal heart rate is detected.

E. Power the System:

Use a reliable power source: Either a battery pack or a USB connection to power the Arduino and the connected modules.

Software

Arduino software:

- Arduino is a prototype platform (open-source) based on an easy-to-use hardware and software.
- It consists of a circuit board, which can be programed and a ready-made software called Arduino IDE (Integrated Development Environment), which is used to write and upload the computer code to the physical board

3.1(b) ARDUINO IDE INSTALLATION

Step 1: First we must have our Arduino board and a USB cable.

Step 2: Download Arduino IDE Software.We can get different versions of Arduino IDE from the Download page on the Arduino Official website.We must select our software, which is compatible with your operating system (Windows, IOS, or Linux). After file download is complete, unzip the file.

Step3: Power up your board Connect the Arduino board to your computer using the USB cable. The green power LED (labeled PWR) should glow.

Step 4: Launch Arduino IDE

After our Arduino IDE software is downloaded, we need to unzip the folder. Inside the folder, we find the application icon with an infinity label (application.exe). double-click the icon to start the IDE.

Step5:Open your project once the software starts,you have two options

1.Create a new project

2.Open an existing pro

Step 6: Select your Arduino board.

Go to Tools -> Board and select our board.

Step 7: Select serial port and upload the program.

3.2 ALGORITHM

Start: Begin the process.

Initialize System: Set up all components.

Power Supply: Distribute power to the system.

Collect and Send ECG Data: Gather ECG data and send it to Arduino Uno.

Process ECG Data: Analyze the ECG data

Abnormal ECG?:

Check if ECG is abnormal. **No**:

Continue collecting data. **Yes**:

Proceed to next steps.

Activate Alerts: Enable voice and SMS alerts.

ESP8266 &GSM:Use Wi-Fi and GSM for communication.

Send Data to Thingspeak: Transmit data to IoT platform.

Alert Condition Met?:

Check if critical condition is met.

No: Continue monitoring.

Yes: Send alerts.

Send Alerts:

Send voice and SMS alerts to predefined numbers.

End:

End the process

3.3 FLOW CHART

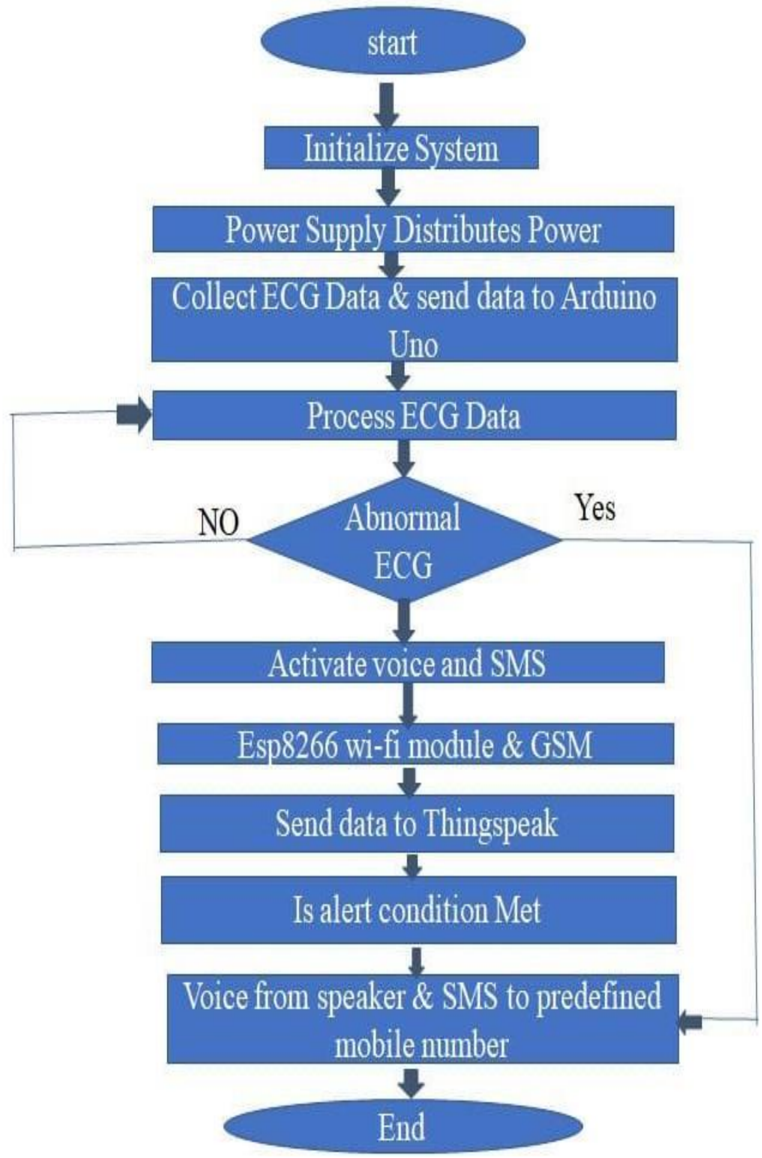


Fig: 3.3 Flowchart

Figure 3.3 shows the flowchart of the project, a flowchart is a Graphical representation of a process, showing the sequence of steps used for this project.

Initialization:

Initialize the Arduino, display, and communication modules.

Set up the necessary pin modes (e.g., for the ECG sensor, buzzer, and communication module).

Initialize any libraries required for the display and communication.

Data Acquisition:

Continuously read the analog signal from the ECG sensor.

Signal Processing:

Filter the ECG signal to remove noise (e.g., using a simple moving average filter).

Detect peaks in the ECG signal to calculate the heart rate.

Calculate the time interval between peaks to determine the heart rate in BPM.

Display Data:

Update the OLED or LCD display with the current heart rate and ECG waveform.

Alert Mechanism:

Set thresholds for normal and abnormal heart rates.

Trigger an alert if the heart rate falls outside the normal range (e.g., sound the buzzer and send a notification via the communication module).

CHAPTER 4

HARDWARE & SOFTWARE DESCRIPTION

4.1 Introduction:

In this chapter the block diagram of the project and design aspect of independent modules are considered.

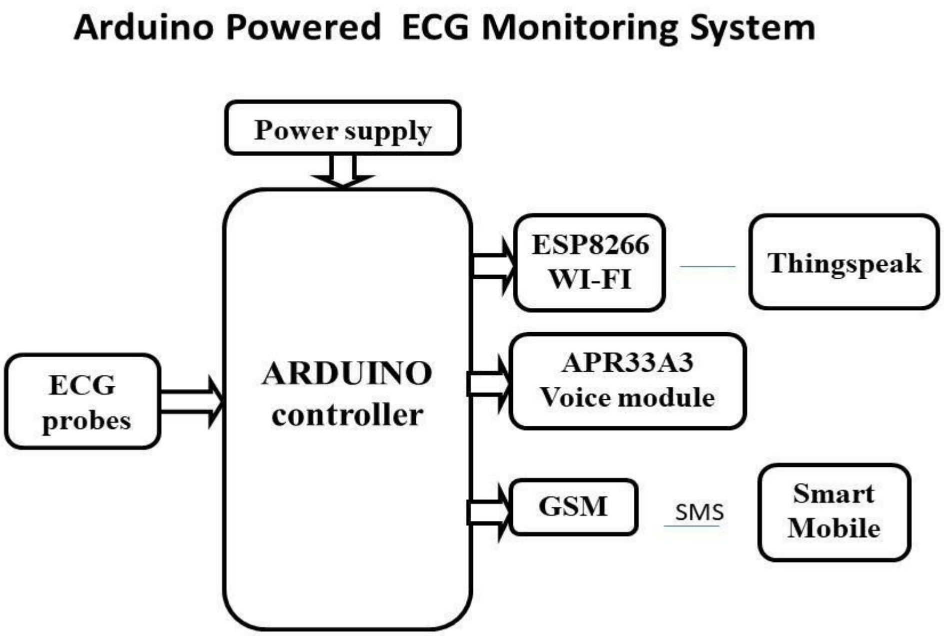


Fig 4.1 Block diagram of Arduino powered ECG monitor

Figure-4.1shows about the block diagram of the project, it represent different parts of the system and lines or arrows that illustrate the relationships and flow of information or materials between these parts.

The major building blocks of this project are:

- Power supply.
- Arduino UNO.
- AD8232 ECG module.
- APR33A3 Voice module.
- GSM.
- ESP8266 WI-FI Module.

4.2Microcontroller

Fig: 4.2 Arduino uno atmega328p Microcontroller

Figure-4.2 shows the Arduino Board which is having pins to connect with the other hardware components and to dump the code.

The Arduino Uno is a microcontroller board which has ATmega328 from the AVR family. There are 14 digital input/output pins, 6 Analog pins and 16MHz ceramic resonator.

USB connection, power jack and also a reset button is used. Its software is supported bya number of libraries that makes the programming easier.

ATMEGA328:

Features

- **High Performance, Low Power AVR® 8-Bit Microcontroller**
- **Advanced RISC Architecture**
  - 131 Powerful Instructions – Most Single Clock Cycle Execution
  - 32 x 8 General Purpose Working Registers
  - Fully Static Operation
  - Up to 20 MIPS Throughput at 20 MHz
  - On-chip 2-cycle Multiplier

- **High Endurance Non-volatile Memory Segments**

– 4/8/16/32K Bytes of In-System Self-Programmable Flash

program memory (ATmega48PA/88PA/168PA/328P)

– 256/512/512/1K Bytes EEPROM (ATmega48PA/88PA/168PA/328P)

– 512/1K/1K/2K Bytes Internal SRAM (ATmega48PA/88PA/168PA/328P)

– Write/Erase Cycles: 10,000 Flash/100,000 EEPROM

– Data retention: 20 years at 85°C/100 years at 25°C(1)

– Optional Boot Code Section with

_Independent Lock Bits In-System

_Programming by On-chip Boot Program

_True Read-While-Write Operation

– Programming Lock for Software Security

- **Peripheral Features**

– Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode

– One 16-bit Timer/Counter with Separate Prescaler, Compare

Mode.

– Byte-oriented 2-wire Serial Interface (Philips I2C compatible)

– Programmable Watchdog Timer with Separate On-chip Oscillator

– On-chip Analog Comparator

– Interrupt and Wake-up on Pin Change

- **Special Microcontroller Features**

  – Power-on Reset and Programmable Brown-out Detection

  – Internal Calibrated Oscillator

  – External and Internal Interrupt Sources

  – Six Sleep Modes: Idle, ADC Noise Reduction, Power-save,

  Power-down, Standby, and Extended Standby

- **I/O and Packages**

  – 23 Programmable I/O Lines

  – 28-pin PDIP, 32-lead TQFP, 28-pad QFN/MLF and 32-pad QFN/MLF

- **Operating Voltage:**

  – 1.8 - 5.5V for ATmega48PA/88PA/168PA/328P

- **Temperature Range:**

  – -40°C to 85°C

- **Speed Grade:**

  – 0 - 20 MHz @ 1.8 - 5.5V

- **Low Power Consumption at 1 MHz, 1.8V, 25°C for ATmega48PA/88PA/168PA/328P:**

  – Active Mode: 0.2 mA

  – Power-down Mode: 0.1 µA Power-save Mode: 0.75 µA (Including 32 kHz RTC)

PDIP

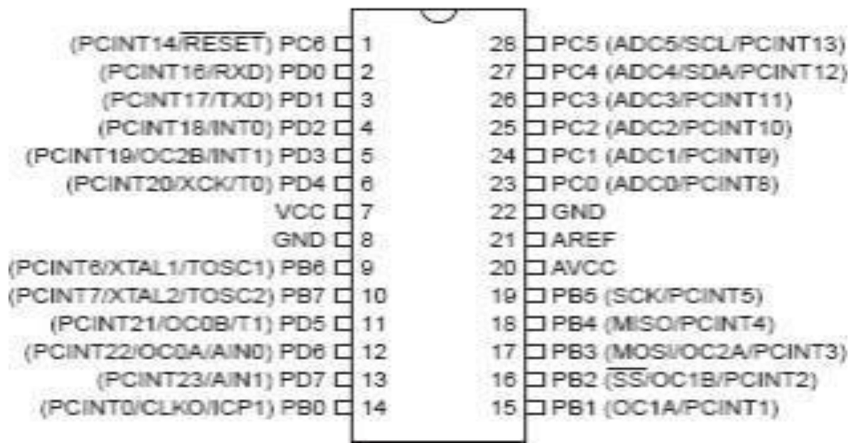


Fig: 4.2(a) PDIP

Figure 4.2(a) shows ATmega328 is a 28-pin AVR Microcontroller, manufactured by Microchip,follows RISC Architecture and has a flash-type Program memory of 32KB

4.1Pin Descriptions

4.1.1VCC Digital supply voltage.

4.1.2GND Ground.

4.1.3Port B (PB7:0) XTAL1/XTAL2/TOSC1/TOSC2

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for eachit). The Port B output buffers have symmetrical drive characteristics with both high sinkand source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running. Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit. Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier.

If the Internal Calibrated RC Oscillator is used as chip clock source, PB7..6 is used as TOSC2..1 input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set. The various special features of Port B are elaborated in ”Alternate Functions of Port B” on page 82 and ”System Clock and Clock Options” on page 26.

4.1.4 Port C (PC5:0)

Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each it). The PC5..0 output buffers have symmetrical drive characteristics with both high sink and source capability.
As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

4.1.5 PC6/RESET

If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C. If the RSTDISBL Fuse is un programmed, PC6 is used as a Reset input.
A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running.

The minimum pulse length is given in Table 28-3 on page 318. Shorter pulses arenot guaranteed to generate a Reset. The various special features of Port C are elaboratedin ”Alternate Functions of Port C” on page 85.

4.1.6 Port D (PD7:0)

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for eachbit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running. The various special features of Port D are elaborated in ”Alternate Functions of Port D” on page 88**.**

4.1.7 AVCC

AVCC is the supply voltage pin for the A/D Converter, PC3:0, and ADC7:6. It should beexternally connected to VCC, even if the ADC is not used. If the ADC is used, it should be connected to VCC through a low-pass filter. Note that PC6..4 use digital supply voltage, VCC.

4.1.8 AREF

AREF is the analog reference pin for the A/D Converter.

4.1.9 ADC7:6 (TQFP and QFN/MLF Package Only)

In the TQFP and QFN/MLF package, ADC7:6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

Overview

The ATmega48PA/88PA/168PA/328P is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega48PA/88PA/168PA/328P achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing spe

Comparison between ATmega48PA, ATmega88PA, ATmega168PA and ATmega328P

4.2 Memory Size:

Device	Flash	EEPROM	RAM	Interrupt Vector Size
ATmega48PA	4K Bytes	256 Bytes	512 Bytes	1 instruction word/vector
ATmega88PA	8K Bytes	512 Bytes	1K Bytes	1 instruction word/vector
ATmega168PA	16K Bytes	512 Bytes	1K Bytes	2 instruction words/vector
ATmega328P	32K Bytes	1K Bytes	2K Bytes	2 instruction words/vector

The ATmega48PA, ATmega88PA, ATmega168PA and ATmega328P differ onlyin memory sizes, boot loader support, and interrupt vector sizes. Table 2-1 summarizes the different memory and interrupt vector sizes for the three devices.

ATmega88PA, ATmega168PA and ATmega328P support a real Read-While-Write Self- Programming mechanism. There is a separate Boot Loader Section, and the SPM instruction can only execute from there.

In ATmega48PA, there is no Read-While-Write support and no separate Boot Loader Section. The SPM instruction can execute from the entire Flash.

4.3 ADAPTER SUPPLY:

The AC adapter, AC/DC adapter or AC/DC converter is a type of external power supply,often enclosed in a case similar to an AC plug. Other names include plug pack, plug-in adapter, adapter block, domestic mains adapter, line power adapter, wall wart, or power adapter. AC adapters are used with electrical devices that require power but do not contain internal components to derive the required voltage and power from mains power. The internal circuitry of an external power supply is very similar to the design that wouldbe used for a built-in or internal supply External power supplies are used both with equipment with no other source of power and with battery-powered equipment, where the supply, when plugged in, can sometimes charge the battery in addition to powering the equipment.

Fig 4.3:Adapter

Figure-4.3 shows Adapter, An adapter typically refers to a device or component that allows different incompatible interfaces or devices to work together. which is mainly used to connect the devices.

4.4 Light Emitting Diode :

A light-emitting diode (LED) is a semiconductor light source. LED’s are used as indicatorlamps in many devices, and are increasingly used for lighting. Introduced as a practical electronic component in 1962, early LED’s emitted low- intensity red light, but modern versions are available across the visible, ultraviolet and infrared wavelengths, with very high brightness.

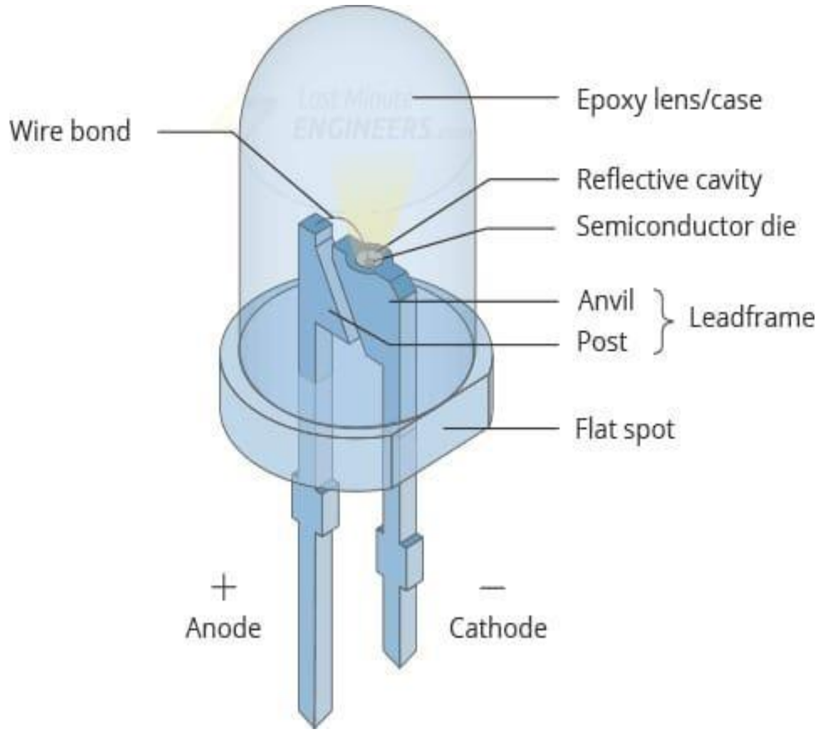


Fig 4.4: LED PARTS

Figure 4.4 shows the internal parts of the LED

Working:

The structure of the LED light is completely different from that of the light bulb. Amazingly, the LED has a simple and strong structure. The light-emitting semiconductor material is what determines the LED's color.

When a diode is forward biased (switched on), electrons are able to recombine with holeswithin the device, releasing energy in the form of photons. This effect is called electroluminescence and the color of the light (corresponding to the energy of the photon)is determined by the energy gap of the semiconductor. An LED is usually small in area (less than 1 mm^2), and integrated optical components are used to shape its radiation pattern and assist in reflection. LED’s present many advantages over incandescent light sources including lower energy consumption, longer lifetime, improved robustness, smaller size, faster switching, and greater durability and reliability.

LED lights have a variety of advantages over other light sources:

- High-levels of brightness and intensity
- High-efficiency
- Low-voltage and current requirements
- Low radiated heat
- High reliability (resistant to shock and vibration)
- No UV Rays
- Long source life
- Can be easily controlled and programmed

Applications of LED fall into three major categories:

- Visual signal application where the light goes more or less directly from the LED tothe human eye, to convey a message or meaning.
- Illumination where LED light is reflected from objects to give visual response tothese objects.
- Generate light for measuring and interacting with processes that do not involve the human visual system.

4.5 Electrocardiography

Electrocardiogram (ECG): working principle

Electrocardiogram refers to the recording of electrical changes that occur in the heart during a cardiac cycle. It may be abbreviated as ECG or EKG.

Working principle of electrocardiograph:

It works on the principle that a contracting muscle generates a small electric current thatcan be detected and measured through electrodes suitably placed on the body.

For a resting electrocardiogram, a person is made to lie in the resting position and electrodes are placed on arms, legs and at six places on the chest over the area of the heart.

The electrodes are attached to the person's skin with the help of a special jelly.

The electrode picks up the current and transmits them to an amplifier inside the electrocardiograph. Then an electrocardiograph amplifies the current and records them on a paper as a wavy line.

In an electrocardiograph, a sensitive lever traces the changes in current on a moving sheetof paper.

A modern electrocardiograph may also be connected to an oscilloscope, an instrument that display the current on a screen.

An ECG is a paper or digital recording of the electrical signals in the heart. It is also called an electrocardiogram or an EKG. The ECG is used to determine heart rate, heart rhythm, and other information regarding the heart's condition. ECGs are used to help diagnose heart arrhythmias, heart attacks, pacemaker function, and heart failure.

ECG can be analyzed by studying components of the waveform. These waveform components indicate cardiac electrical activity. The first upward of the ECG tracing is the P wave. It indicates atrial contraction.

The QRS complex begins with Q, a small downward deflection, followed by a larger upwards deflection, a peak (R); and then a downwards S wave. This QRS complex indicates ventricular depolarization and contraction.

Finally, the T wave, which is normally a smaller upwards waveform, representing ventricular es a clue about the condition of heart muscle and is used to diagnose heart disorders

It helps the doctors to determine whether the heart is normal, enlarged or if its certain repolarization.

Application of ECG:

- It indicates the rate and rhythm or pattern of contraction of heart
- It give regions are damaged

- It can also reveal irregularities in heart's rhythm known as 'arrhythmia'
- It is used by doctors to diagnose heart damage in conditions like high blood pressure, rheumatic fever and birth defects
- An ECG also helps to determine the location and amount of injury caused by heart attack and later helps to assess the extent of recovery

4.6 AD8232 ECG Sensor:

This sensor is a cost-effective board used to measure the electrical activity of the heart. This electrical activity can be charted as an ECG or Electrocardiogram and output as an analog reading. ECGs can be extremely noisy, the AD8232 Single Lead Heart Rate Monitor acts as an op-amp to help obtain a clear signal from the PR and QT Intervals easily.

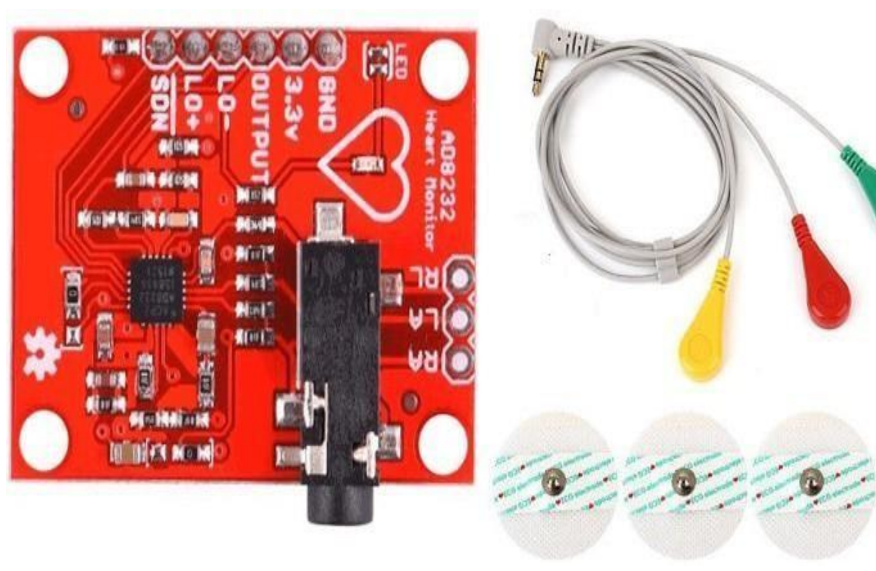


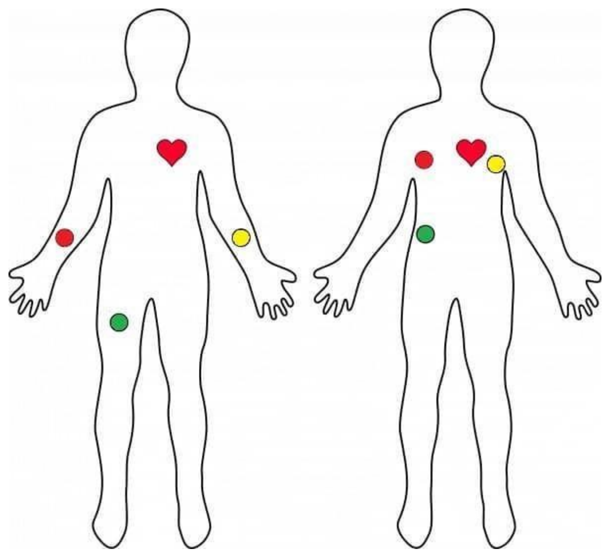
Fig 4.6 :AD8232 ECG Sensor

Figure- 4.6 shows ADE8232 ECG sensor, it mainly has there sensors one is forleft and two are for left .It should attach to be hands.

The AD8232 is an integrated signal conditioning block for ECG and other biopotential measurement applications. It is designed to extract, amplify, and filter small biopotential signals in the presence of noisy conditions, such as those created by motion or remote electrode placement.

AD8232 ECG Sensor Placement on Body

It is recommended to snap the sensor pads on the leads before application to the body. The closer to the heart the pads are, the better the measurement. The cables are color-coded to help identify proper placement.



Red: RA (Right Arm)

Yellow: LA (Left Arm)

Green: RL (Right Leg)

4.7(a) GSM:

Global System for Mobile Communication (GSM) Definition:

GSM, which stands for Global System for Mobile communications, reigns (important) asthe world’s most widely used cell phone technology. Cell phones use a cell phone service carrier’s GSM network by searching for cell phone towers in the nearby area. Global system for mobile communication (GSM) is a globally accepted standard for digital cellular communication.

GSM is the name of a standardization group established in 1982 to create a common European mobile telephone standard that would formulate specifications for a pan-European mobile cellular radio system operating at 900 MHz. It is estimated that many countries outside of Europe will join the GSM partnership.

Fig 4.7(a): GSM Module

Figure-4.7(a) shows GSM module typically includes a SIM card slot, antennaconnectors and various input & output pins for integration

It means engineers cut their teeth on the technology, creating an unconscious preference.

4.7(b) APR33A3 Voice Module

Features:

- Total 11 minutes of recording time each channel(M0 to M7) having 1.3 minutes of recording time.
- Single chip, high quality voice recording and playback solution.
- User friendly and easy to use operation.
- Non-Volatile flash memory technology, no battery backup required.

- Audio output to drive a speaker or audio out for public address system.
- Can record voice with the help of on-board microphone

How to Record your Voice:

- We can use 8 channels (M0 TO M7) each channel having 1.3 minutes recording length.
- Onboard MIC will automatically be used for recording.
- Supply voltage: 12v AC/DC.
- Switch on the board power LED (LD1) will on.
- Put the jumper in the board JP1 (REC) Section.
- While in record mode select J5 (M0-M7) to select a channel to record the message.
- Let us assume we want to record a message in channel M0, Connect M0 to GND (IN Board J3-VCC, GND).

Now whatever we speak will be captured by MIC and recorded, and the status LED (LD2) will be on in record mode indicating that chip is currently recording. Once duration is full the LED (LD2) will off means that segment is full. Now you can disconnect the GND Connection from M0, if before the duration is this connection is removed, then that many seconds are recorded and rest duration is kept empty.

How to use a Microcontroller?

- Better Do Voice Recording can be done Manually
- To play back connect Controller I/Os to M0 to 7

4.8 ESP8266 WI-FI Module:

ESP 8266 Description:

The ESP8266 Wi-Fi Module is a self-contained SOC with integrated TCP/IP protocol stack that can give any microcontroller access to your Wi-Fi network.

The ESP8266 is capable of either hosting an application or offloading all Wi-Finetworking functions from another application processor.

Each ESP8266 module comes pre-programmed with an AT command set firmware, meaning, you can simply hook this up to your Arduino device and get about as much Wi-Fi-ability as a Wi-Fi Shield offers (and that's just out of the box)! The ESP8266 moduleis an extremely cost- effective board.

This module has a powerful enough on-board processing and storage capability that allows it to be integrated with the sensors and other application specific devices throughits GPIOs with minimal development up-front and minimal loading during runtime.

Its high degree of on-chip integration allows for minimal external circuitry, including thefront-end module, which is designed to occupy minimal PCB area.

The ESP8266 supports APSD for VoIP applications and Bluetooth coexistence interfaces; it contains a self-calibrated RF allowing it to work under all operating conditions, and requires no external RF parts.

There is an almost limitless fountain of information available for the ESP8266, all of which has been provided by amazing community support.

In the Documents section below you will find many resources to aid you in using the ESP8266, even instructions on how to transform this module into an IoT (Internet of Things) solution!

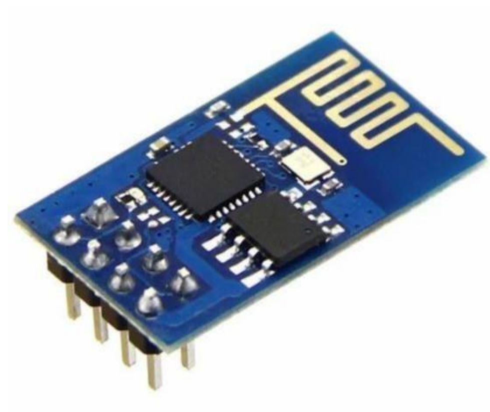


Fig4.8: ESP8266 Wi-Fi Module

Figure-4.8 shows the ESP8266 wi-fi module mainly used in various IOT for connecting the devices

4.9 Thingspeak:

ThingSpeak

Introduction:

The Internet of Things(IoT) is a system of 'connected things'. The things generally consist of an embedded operating system and an ability to communicate with the internet or with the neighboring things.

One of the key elements of a generic IoT system that bridges the various 'things' is an IoT service. An interesting implication from the 'things' comprising the IoT systems is that the things by themselves cannot do anything.

At a bare minimum, they should have an ability to connect to other 'things'. But the real power of IoT is harnessed when the things connect to a 'service' either directly or via other 'things'.

In such systems, the service plays the role of an invisible manager by providing capabilities ranging from simple data collection and monitoring to complex data analytics. The below diagram illustrates where an IoT service fits in an IoT ecosystem:

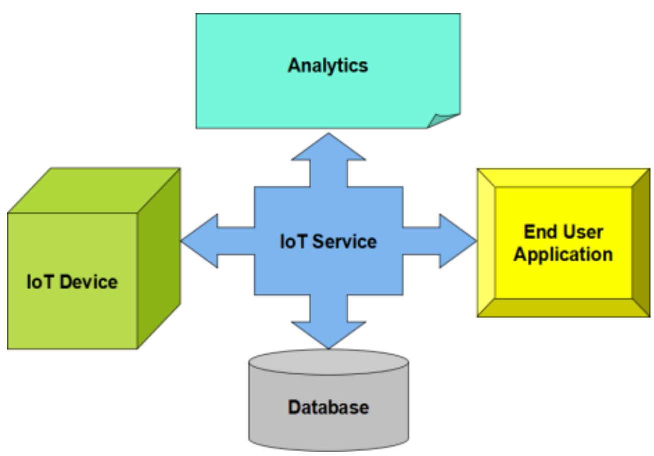


Fig 4.9: Connected system of IOT service

Figure 4.9 shows the connected system of the IOT services and how they are connected to each other.

What is ThingSpeak

ThingSpeak is a platform providing various services exclusively targeted for building IoT applications. It offers the capabilities of real-time data collection, visualizing the collected data in the form of charts, ability to create plugins and apps for collaborating with web services, social network and other APIs. We will consider each of these features in detail below.The core element of ThingSpeak is a 'ThingSpeak Channel'. Achannel stores the data that we send to ThingSpeak and comprises of the below elements:

- 8 fields for storing data of any type - These can be used to store the data froma sensor or from an embedded device.
- 3 location fields - Can be used to store the latitude, longitude and the elevation.These are very useful for tracking a moving device.
- 1 status field - A short message to describe the data stored in the channel.

To use ThingSpeak, we need to sign up and create a channel. Once we have a channel, we can send the data, allow ThingSpeak to process it and also retrieve thesame. Let us start exploring ThingSpeak by signing up and setting up a channel.

This project is implemented using following software's:

- Express PCB – for designing circuit
- Arduino IDE Studio compiler - for compilation part

4.9(a)Arduino IDE Compiler:

This instructable adds to any of the Arduino on a Breadboard instructables.

1. We need a microcontroller with a preloaded Bootloader, or must load your own

2. Not all ATmega328's are equal

(A bootloader, very simply, is a program that sits on the chip and manages theupload of your sketches onto the chip)

4.9(b)Procedural steps for compilation, simulation and dumping:

Compilation and simulation steps:

Step 1: Parts

1 x Arduino on a Breadboard 1 x Arduino UNO Connecting Wires
Arduino IDE installed on your PC Step

Step 2: The Approach

We use the Arduino UNO to bootload the ATmega328 that is sitting on theArduino-on-a- Breadboard. This is fairly straightforward having an ATmega328P-PU, but needs an extra step for an ATmega328-PU.

Step 3: Program your Arduino UNO as an ISP

We need to program the Arduino UNO to act as an ISP (In-System Programmer), sothat it can burn the bootloader onto the Breadboard chip.

1. Open the Arduino IDE
2. Open the ArduinoISP sketch (under File, Examples)
3. If you're using version 1.0 of the IDE:

Search for void heartbeat and change the line that reads:
delay(40); to delay(20);

Connect your UNO to the PC, making sure it's not connected to the Arduino onBreadboard.

Ensure your UNO is selected under the Boards menu option, and upload the sketch.

Step 4: Connect your ATmega328

- **UNO 5v ---> ATmega pin 7 (VCC)**
- **UNO GND ---> ATmega pin 8 (GND)**

- **UNO pin 10 ---> ATmega pin 1 (RESET)**
- **UNO pin 11 ---> ATmega pin 17 (MOSI)**
- **UNO pin 12 ---> ATmega pin 18 (MISO)**
- **UNO pin 13 ---> ATmega pin 19 (SCK)**

Step 5: Which ATmega328 are you using?

The two variants that are of interest to us are the ATmega328-PU and the ATmega328P-PU.

The -PU suffix means that the chips are in a PDIP package, the format we need for our breadboard. The 328P is a Pico Power processor, designed for low power consumption, and is usedon Arduino boards. Given low power consumption this is the first choice.

The 328 does not have Pico Power technology, and is not used on the Arduino boards –and is not explicitly supported by the Arduino IDE.

Unfortunately the websites that sell these chips don't always differentiate between themand forums are filled with people struggling to use the ATmega328-PU.
Luckily there is a workaround - take a look at my Crash Bang website.

Step 6: ATmega328-PU workaround

Each microprocessor has a signature – a unique code that identifies its model. When you bootload a chip (or even upload a sketch) the Arduino IDE checks that the chip selected matches the type it’s connected to. Even though the ATmega328- PU in essence functions in the same way as the ATmega328P-PU, it has a different signature,and one that isn’t recognised by the Arduino IDE.

Step 7: Bootload the ATmega328

In the Arduino IDE, from the Tools menu:

- under the Board option choose Arduino UNO
- under the Serial Port option ensure the correct port is selected
- under the Programmer option choose Arduino as ISP

To burn the Bootloader, choose Burn Bootloader from the Tools menu
You should see a message “Burning bootloader to I/O Board (this may take a minute)" Once the bootloader has been burned, a message confirming the success gets displayed. ”Congratulations: You're now ready to load sketches onto your Arduino on abreadboard!”

CHAPTER 5

RESULTS AND DISCUSSION

Heart Rate Monitoring: The system should be able to measure and display the heart ratein beats per minute (BPM) accurately. This is the most basic function of an ECGmonitoring system.

ECG Signal Display: The system should be capable of capturing and displaying the ECG waveform on a screen, such as an OLED display or on a computer via serial communication. This waveform represents the electrical activity of the heart.

Real-Time Data Processing: The system should process the ECG data in real-time to detect any abnormalities in heart rhythms, such as arrhythmias.

Alert System: If the system detects an abnormal heart rate or rhythm, it should trigger an alert. This could be a visual alert on the display, an audible alert via a buzzer, or a notification sent to a smart phone or other device.

Data Logging: The system could store the ECG data for later analysis. This might involve saving data to an SD card or transmitting it to a remote server.

5.1 ADVANTAGES:

- Cost-effective, making it accessible to a wide range of users.
- Remote Monitoring
- Real-time Alert
- Continous Monitoring

- Data Analysis
- Integration of Voice Communication

5.2 APPLICATIONS:

- Homes
- Old age homes
- Hospital
- Rural areas

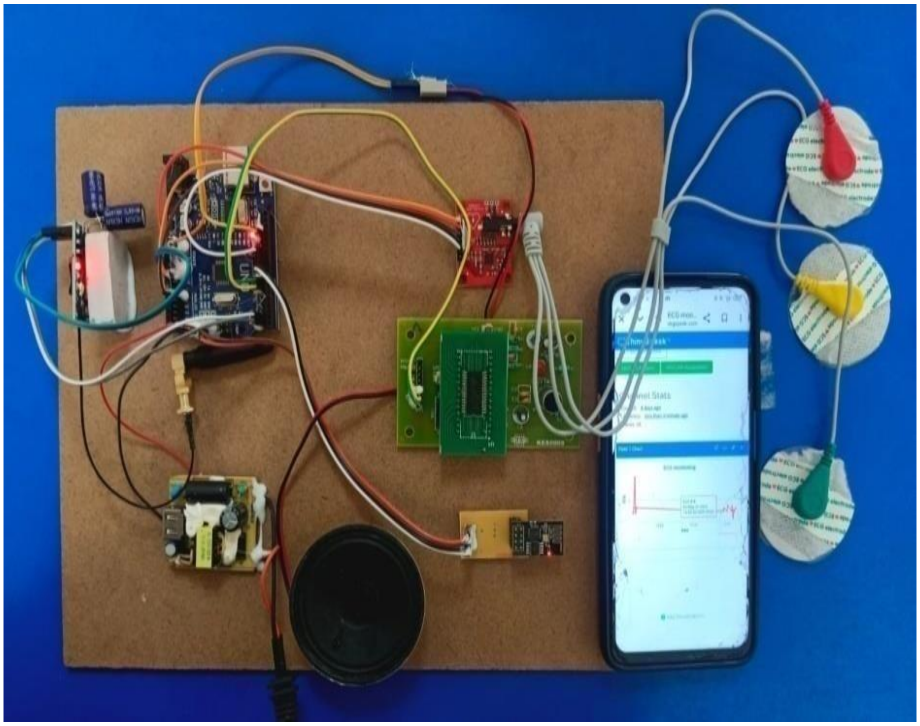


Fig 5.1: Overall Hardware Kit

The Figure 5.1 shows the overall kit of the Building an Arduino Powered ECG MonitorAnd Alerting System and it includes all the components such as ECG Sensors, Voice Module, Wi-Fi Module, Arduino Uno, GSM.

Fig 5.2:Graph in thingspeak with date and time

The Figure 5.2 shows the electrical signals of the heart rate in thingspeak in the form of graph with date and time

Fig 5.3: Speaker

Figure 5.3 shows the speaker for alerting by using voice module and that can give alert message via speaker

The APR33A3 is a voice recorder and playback IC that can be used with a speaker to produce amplified sound

How it works:

The APR33A3 is a voice recording and playback IC that includes a microphone input, audio processing, storage memory, and playback circuitry. It can record and play back audio on up to eight channels. To play back, you can connect the Controller I/Os to M0 to M7.

Using a speaker:

The APR33A3 comes with a small test speaker, but it may not produce loud sound. To get louder sound, you can connect an amplifier to the APR33A3 and use a larger speaker.

Speaker fundamentals:

Speakers convert electrical energy into mechanical energy, which compresses air and creates sound. A speaker's driver has a wire voice coil that's suspended in a magnetic field. When an electrical signal is sent to the coil, it causes the cone to move back and forth, creating sound waves.

Fig 5.4: Sms Alert

Figure 5.4 shows the alert SMS by using GSM Module which is a Alert for Abnormal Condition of a patient in this way it shows the SMS

SMS stands for Short Message Service. It is a text messaging service that allows the exchange of short text messages between mobile devices. SMS messages typically have a maximum length of 160 characters and can be sent and received on various mobile networks.

Patient 1:

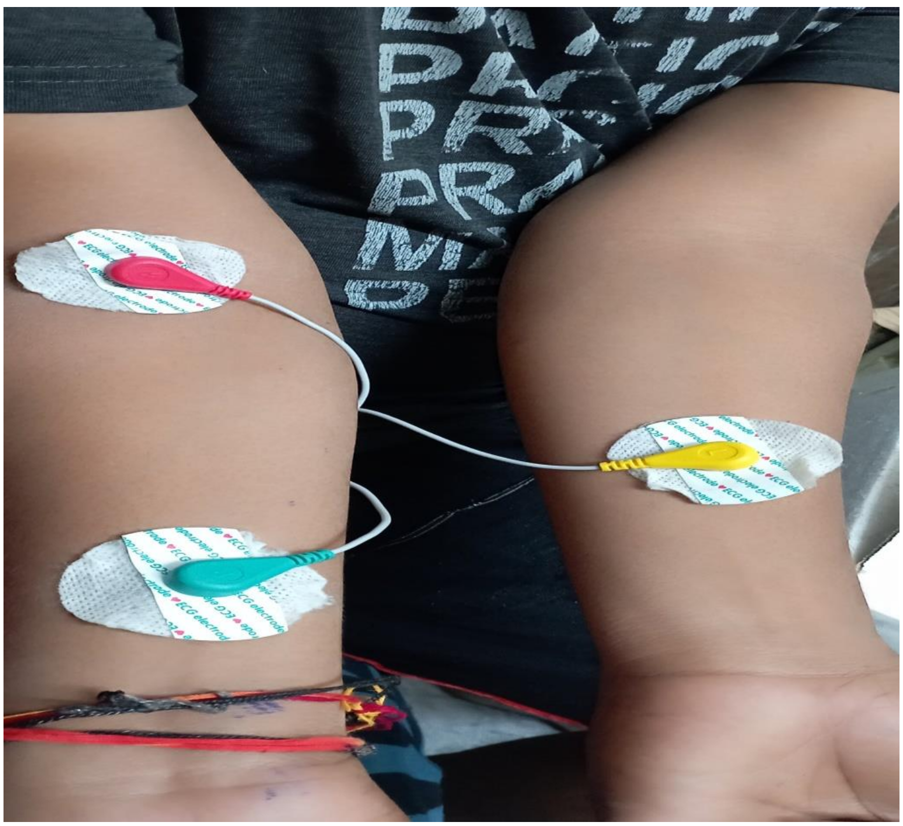


Fig 5.5 Patient 1 having Normal ECG

The Figure 5.5 shows the Patient 1 having a normal ECG, a normal electrocardiogram (ECG) shows a regular heart rhythm at a rate of 60–100 beats per minute (bpm).

Many different heart conditions can show up on an ECG, including a fast, slow, or abnormal heart rhythm, a heart defect, coronary artery disease, heart valve disease, or an enlarged heart.

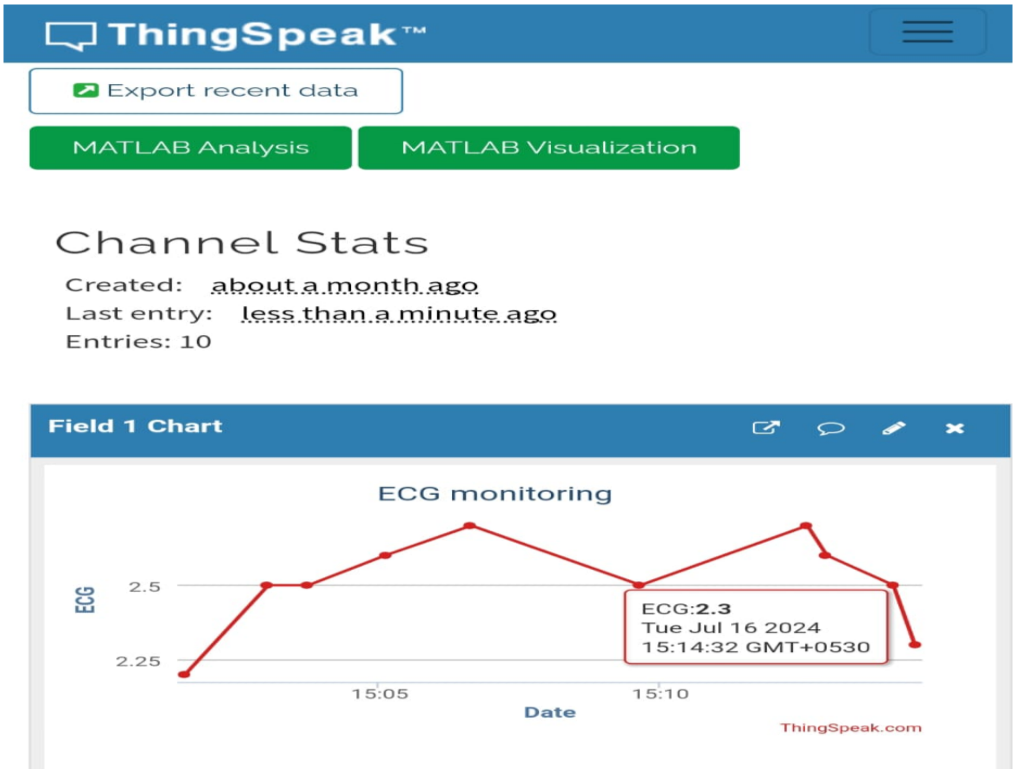


Fig 5.6 Graph of Patient 1 having Normal ECG

The Figure5.6 shows the Graph of a patient 1 in Thingspeak cloud with date and time ECG (Electrocardiogram) monitoring graph from the IoT platform ThingSpeak. ThingSpeak is commonly used for storing and visualizing sensor data in real-time, and here it displays ECG values over time.

If this data comes from a real-time monitoring system, it could be useful for analyzing heart activity patterns. This could relate to health monitoring systems, IoT in healthcare, or sensor data processing. Let me know if you need help with anything specific, like setting up the ThingSpeak channel, analyzing data,

Patient 2:

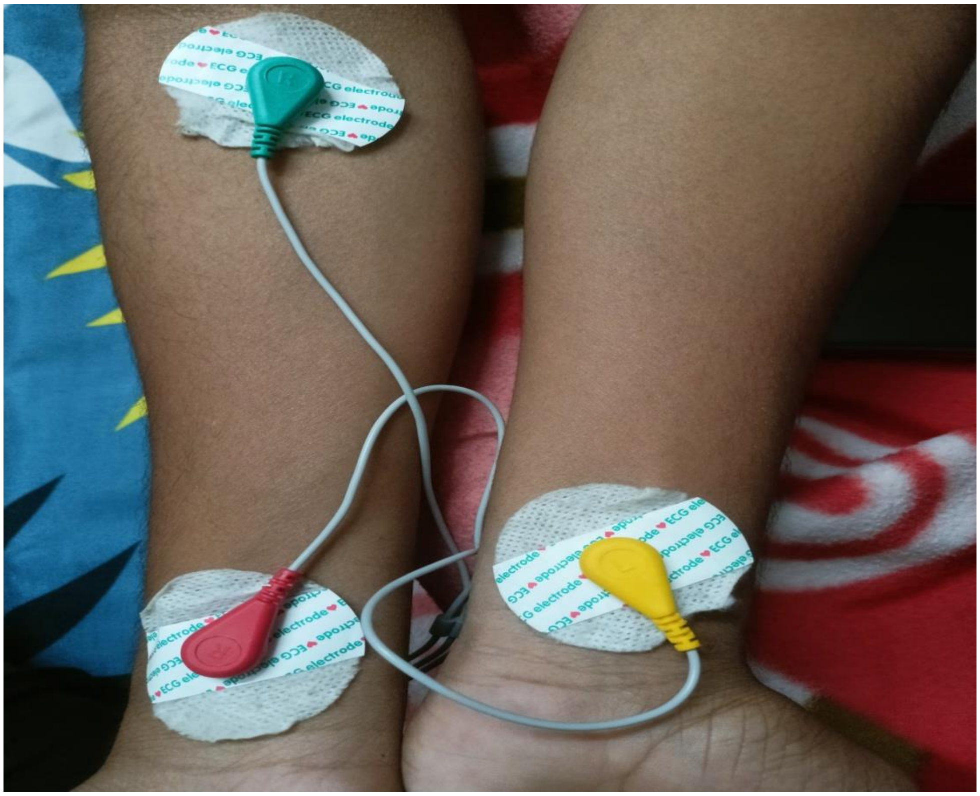


Fig 5.7 Patient 2 having Abnormal ECG

The Figure5.7shows the Patient 2 having a Abnormal ECG, An abnormal ECG (electrocardiogram) can indicate various heart issues or irregularities.

Generally, ECG readings capture electrical activity in the heart, with normal readings showing regular patterns representing heartbeats. Abnormalities might show up as irregular patterns or unusual intervals between beats.

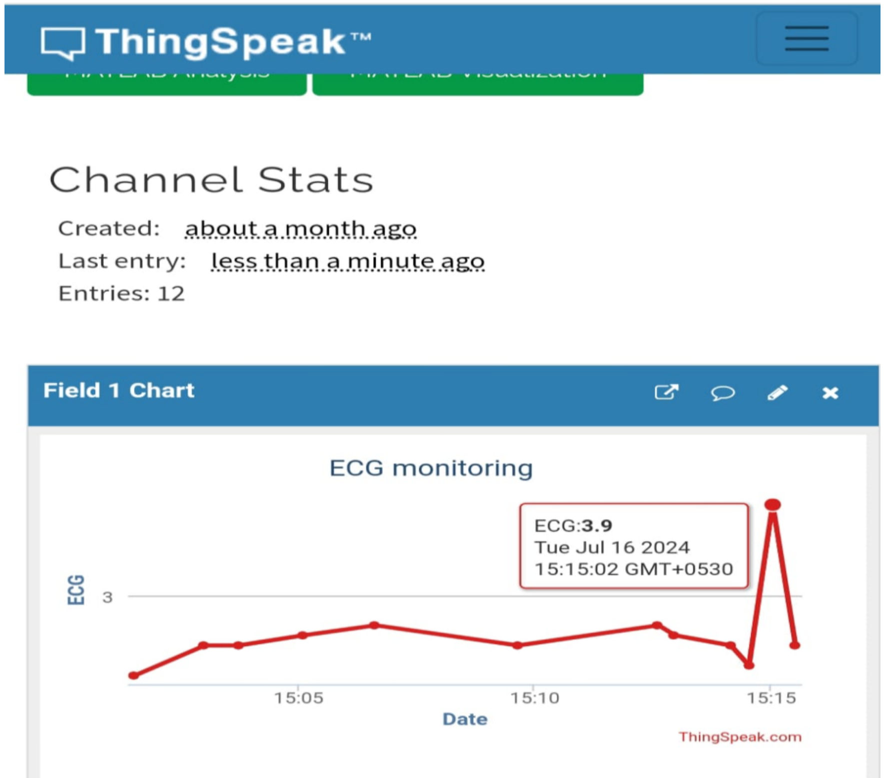


Fig 5.8 Graph of Patient 2 having Abnormal ECG

The Figure 5.8 shows the Graph of a patient 2 in Thingspeak cloud with date and time ECG (Electrocardiogram) monitoring graph from the IoT platform ThingSpeak

ECG monitoring graph on ThingSpeak, we can see an unusual spike in the ECG reading, reaching a value of 3.9. This sudden increase, followed by a drop, could indicate an anomaly or irregularity in the heart's activity. Here are some possibilities for this kind of spike.Above 3.8 it is considered as Abnormal ECG

CHAPTER 6

CONCLUSION & FUTURE SCOPE

6.1 CONCLUSION:

The ECG Monitor is designed using Arduino along with the alerting system. The patient’s heart rate is measured in electrical signals by ECG probes. All these electrical signals are fed to the ARDUINO. As a result in this project, Arduino continuously reads the data from the ECG sensor and it will be uploaded into the thingspeak cloud along withdate and time.

When the system detects any abnormal condition, it will send the alert SMS to the predefined mobile number via GSM and also it will give the voice alert through speaker and this project advantages are Cost-effective, making it accessible to a wide range of users, remote monitoring , real-time Alert, continuous monitoring, Data Analysis, Integration of Voice Communication. Applications of this project are Homes, Old age homes, Hospital, Rural areas.

It is a non-invasive and painless procedure that involves attaching electrodes to the skin to record the heart’s electrical signals. This allows them to early detect and promptly treat patients’ conditions, especially in the critical care ward. Real-time ECG monitoring can also be used for remote monitoring of patients.

Building an Arduino-powered ECG monitoring and alerting system offers a promising low-cost solution for continuous heart health monitoring. Such a system can empower individuals to track their heart health in real-time and alert them to potential issues, potentially saving lives by enabling early detection of heart conditions.

The project combines various fields, including electronics, programming, and biomedical engineering, providing a rich learning experience and numerous opportunities for innovation. While the current system can effectively monitor heart rate and detect basic abnormalities, future advancements in signal processing, wireless communication, IoT integration, and miniaturization could significantly enhance its capabilities and usability.

Ultimately, with further development, rigorous testing, and adherence to medical standards, this Arduino-powered ECG monitoring system could become a valuable tool for personal health monitoring and remote healthcare applications.

6.2 FUTURE SCOPE

The project can be extended by adding blood pressure measuring; sugar levels of the personetc.This project can also be extended by using GPRS and 3G technologies. We can also include python code for easy understanding and which is more faster technology. Instead of Arduino we can include Raspberry pi for better output. The trend towards wearable health monitoring devices is rapidly growing. Arduinopowered ECG systems can be integrated into wearable devices, enhancing their functionality and appeal. Integration withdata analytics and AI algorithms can provide predictive insights into cardiac health, helpingin early detection of abnormalities and proactive healthcare management

Advanced Signal Processing:

Implement more sophisticated signal processing algorithms to better filter out noise and artifacts from the ECG signals.

Use machine learning models to classify different types of arrhythmias and other heart conditions based on the ECG data.

Wireless Communication:

Incorporate wireless communication modules like Bluetooth, Wi-Fi, or GSM to transmit ECG data to remote servers or smartphones for real-time monitoring and alerts.

Develop a mobile application to display real-time ECG data and alerts, enabling remote monitoring by healthcare providers or caregivers.

Integration with IoT:

Connect the ECG monitoring system to the Internet of Things (IoT) platforms for continuous health monitoring and data analysis.

Utilize cloud-based services to store and analyze large volumes of ECG data, enabling long-term health trend analysis and predictive analytics.

Power Efficiency:

Improve power management to extend battery life, making the device more suitable for long-term use.

Explore energy harvesting techniques to power the device using body heat or movement.

Miniaturization:

Develop a smaller, more portable version of the system for easy use in various settings, including home, gym, or during travel.

Integrate all components into a single compact PCB (Printed Circuit Board) design.

REFERENCES

[1].Sohit Agarwal 1, DevashishDasaya "IoT-Based ECG Monitoring System for Health Care Applications" Mathematical Statistician and Engineering Applications 2022 https://www.philstat.org/index.php/MSEA/article/download/1894/1384/3330

[2].MamoonaHumayun, NZ Jhanji, Malak Z Alamri "IoT-based Secure and Energy Efficient scheme for Ehealth applications' 'Science and Technology2020. https://indjst.org/articles/iot-based-secure-and-energyefficient-scheme-for-e-health-applications

[3].Sarada Prasad Gochhayat, Chhagan Lal, Lokesh Sharma, D. P. Sharma, Deepak Gupta, Jose Antonio Marmolejo Saucedo, UtkuKose "Reliable and secure data transfer in IoT networks"Springer2019. [4].Ashutosh Sharma, GeetanjaliRathee, Rajiv Kumar, Hemraj Saini, VijayakumarVaradarajan, Yunyoung Nam and Naveen Chilamkurti "A Secure, Energy- and SLA-Efficient (SESE) E-Healthcare Framework for Quickest Data Transmission Using Cyber-Physical System"MDPI2019. https://doi.org/10.3390/s19092119

[5].Mohamed Elhosen , Gustavo Ramírez-González, Osama M. Abu-Al Nasar , Shihab A. Shawkat, Arunkumar N, Ahmed Farouk "Secure Medical Data Transmission Model for IoTbasedHealthcare Systems" IEEE2018. https://www.researchgate.net/publication/323925165_Secure_Medical_Data_Transmission_Model_for_IoT-Based_Healthcare_Systems

[6]. SohailSaif, Rajni Gupta and Suparna Biswas "Implementation of Cloud-Assisted Secure Data Transmission in WBAN for Healthcare Monitoring"Springer2018. https://link.springer.com/chapter/10.1007/978-981-10-8237-5_64

[7]. Hai Tao, MdZakirulAlamBhuiyan, Ahmed N. Abdalla, Mohammad Mehedi Hassan, Jasni Mohamad Zain, and ThaierHayajneh "Secured Data Collection with Hardwarebased Ciphers for IoT-based Healthcare"IEEE2018 https://scholar.google.com/citations?view_op=view_citation&hl=en&user=1u2fgf4AAAAJ&citation_for_vie w=1u2fgf4AAAAJ:MXK_kJrjxJIC [8]. Munish Bhatia, Sandeep K. Sood "A comprehensive health assessment framework to facilitate IoT-assisted smart workouts: A predictive healthcare perspective" Elsevier2017 https://dblp.org/rec/journals/cii/BhatiaS17.html

[9].Haiping Huang, Member, Tianhe Gong, Ning Ye, Ruchuan Wang and Yi Dou "Private and Secured Medical Data Transmission and Analysis for Wireless Sensing Healthcare System"2017. https://scholar.google.com/citations?view_op=view_citation&hl=en&user=7OTRDQcAAAAJ&citation_for_v iew=7OTRDQcAAAAJ:XiSMed-E-HIC

[10]. Mohamed Elhoseny ET. All (2017) "Secure Data Transmission in WSN" Springer2017.

APPENDIX:
SOURCE CODE

```
#include
#include
 #include
#include

SoftwareSerial mySerial(8,9);// RX,TX
intflag=0;
Stringhbeat="";
 inta=50;
 floathr=0;
floatsp=0;
charch;
 Stringdata;
Stringwdata;
 Stringdata1="";
charcnct[10]=",CONNECT";
StringapiKey="VGHY5JI0P5LUOBJJ";
 inttcount=0;
intcount=0;
inth=0;
intv1=7;
intv2=6;
voidsetup()
 {
                Serial.begin(9600);
                     pinMode(13,OUTPUT);
                     pinMode(10,INPUT);
                     pinMode(11,INPUT);
                     mySerial.begin(115200);
                     mySerial.print("AT\r\n");
                     delay(1000);
                      mySerial.println("AT+RST");
                      pinMode(11,INPUT);
                      pinMode(10,INPUT);
                     pinMode(v1,OUTPUT);
                      pinMode(v2,OUTPUT);
                     digitalWrite(v1,1);
                     digitalWrite(v2,1);
                      Serial.println("AT+CNMI=2,2,0,0,0");
                      digitalWrite(13,1);
                     delay(700);
```

```
            digitalWrite(13,0)
            delay(700);
             digitalWrite(13,1);
             delay(700);
            digitalWrite(13,0);
            digitalWrite(v1,0);
             delay(200);
             digitalWrite(v1,1);
             delay(2000);

      }
intscount=0;
voidloop()
 {
            digitalWrite(13,0);
             if((digitalRead(11)==1)||(digitalRead(12)==1))
 {
             tcount++;
}
if(count>12)
{
            count=0;
}
delay(1000);
if(tcount>10)
 {
             tcount=0;
            if(analogRead(A0)>3.8)
{
             send_alert();
 }
digitalWrite(v2,0);
 delay(200);
 digitalWrite(v2,1);
 delay(2000);

StringgetStr="GET/update?api_key=";
getStr+=apiKey;
 getStr+="&field1=";
getStr+=String(analogRead(A0));
getStr+="\r\n\r\n";

cmd+=String(getStr.length())
```

```
mySerial.println(cmd);
count++;
}
}
 voidinit_sms()
{
         Serial.println("AT+CMGF=1");
        delay(200);
        Serial.println("AT+CMGS=\"+917013909409\"");
        delay(200);
 }
 voidsend_data(Stringmessage)
{
        Serial.println(message);
        delay(200);
 }
voidsend_sms()
 {
             Serial.write(26);
}
voidsend_alert()
{
        init_sms();
        send_data("***ALERT***AbnormalECG");
         delay(200);
        send_sms();
        delay(2000);

}
```