

HEALTH CARE MONITORING SYSTEM FOR HOME QUARANTINE PEOPLE

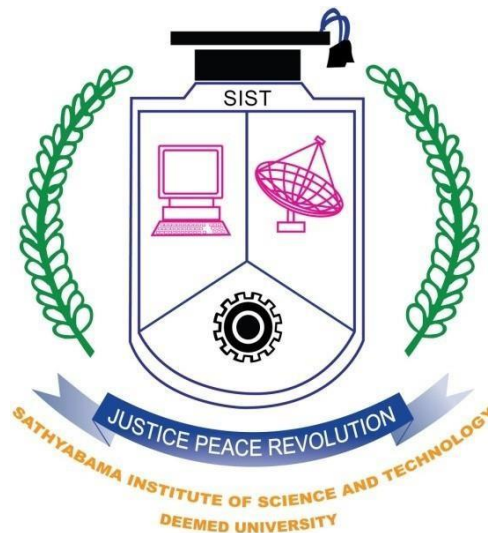
Submitted in partial fulfillment of the requirements for the award of

Bachelor of Engineering Degree in
Electrical and Electronics Engineering

by

A. BALAJI (Register No. 39140006)

J. DHINESH PANDI (Register No. 39140012)



**DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING
SCHOOL OF ELECTRICAL AND ELECTRONICS**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600

119

APRIL 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

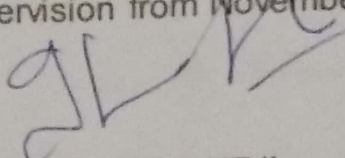
Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

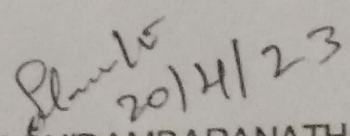
www.sathyabama.ac.in

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

BONAFIDE CERTIFICATE

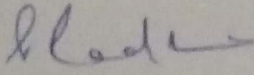
This is to certify that this Project Report is the bonafide work of **A.BALAJI** (Reg. No. 39140006) and **J.DHINESH PANDI** (Reg. No. 39140012) who carried out the project entitled "**HEALTH CARE MONITORING SYSTEM FOR HOME QUARANTINE PEOPLE**" under our supervision from November 2022 to April 2023.


Dr. M.L.BHARATHI
Internal Guide

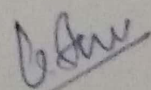

Dr. V. SNACHIDAMBARANATHAN
Head of the Department

Submitted for Viva voce Examination held on 26/04/2023

Name: Dr. S. RAGHIN

Signature: 

Internal Examiner


External Examiner

DECLARATION

We **A.BALAJI** (Reg. No. 39140006) and **J.DHINESH PANDI** (Reg. No. 39140012) hereby declare that the Project Report entitled "**HEALTH CARE MONITORING SYSTEM FOR HOME QUARANTINE PEOPLE**" done by us under the guidance of **Dr.M.L.BHARATHI** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Electrical and Electronics Engineering.

1. *A. Balaji*

2. *J. Dhinesh Pandi*

DATE: 20/04/2023

PLACE: Chennai

SIGNATURE OF THE CANDIDATES

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to Board of Management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr.N.M.Nandhita., Dean, School of Electrical and Electronics Engineering** and **Dr. V. Sivachidambaranathan, Head of the Department, Dept. of Electrical and Electronics Engineering** for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our Project Guide **Dr.M.L.Bharathi** for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

We wish to express our thanks to all Teaching and Non-teaching staff members of the Department of Electrical and Electronics Engineering who were helpful in many ways for the completion of the project.

ABSTRACT

The proposed project aims to develop a healthcare monitoring system for individuals undergoing home quarantine using the Internet of Things (IoT). The system will utilize various IoT devices such as wearables, sensors, and cameras to collect and transmit real-time health data to a central monitoring system. This data will include vital signs, activity levels, and other relevant health parameters. The central monitoring system will use machine learning algorithms to analyze the data and provide insights and alerts to healthcare professionals in case of any deviations from normal health parameters. The system aims to improve patient outcomes by providing timely interventions, reducing the burden on healthcare providers, and increasing patient satisfaction by allowing them to receive care in the comfort of their homes

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	v
	LIST OF FIGURES	ix
	LIST OF TABLES	xi
	LIST OF SYMBOLS AND ABBREVIATIONS	xii
1	INTRODUCTION	
1.1	GENERAL INTRODUCTION	1
1.2	DESIGN OF EMBEDDED SYSTEM	2
1.3	CLASSIFICATION	5
2	LITERATURE SURVEY	
2.1	MAJOR FINDINGS FROM LITERATURE SURVEY	6
2.2	PROBLEM IDENTIFICATION	9
2.3	SCOPE OF THE PROJECT	10
2.4	METHODOLOGY	10
2.5	STANDARD	11
	2.4.1 TERMINOLOGY	12
2.6	TRADE – OFF	15

3	PROJECT DESCRIPTION	17
3.1	EXISTING SYSTEM	17
3.2	PROPOSED SYSTEM	17
3.3	ARDUINO UNO AND ITS PROGRAMMING	
3.3.1	OVERVIEW	22
3.3.2	SUMMARY	24
3.3.3	PIN CONFIGURATION	25
3.4	ARDUINO SOFTWARE	29
4	HARDWARE DISCRPTION	
4.1	HARDWARE COMPONENTS	40
4.1.1	Heartbeat sensor	40
4.1.2	Temperature sensor	41
4.1.3	Micro electro-mechanical system	42
4.1.4	Bluetooth sensor	43
4.2	HARDWARE KIT	45
4.2.1	Benefits	46
5	SUMMARY AND CONCLUSION	48
5.1	CONCLUSION	48
5.2	SCOPE FOR FUTURE WORK	48

LIST OF FIGURES

FIGURE No.	TITLE	PAGE No.
1.1	Embedded product Development lfe cycle	4
3.1	Architecture block diagram	18
3.2	Monitoring section	18
3.3	Circuit diagram of regulator	21
3.4	Arduino uno	22
3.5	Thermocouple diagram	35
3.6	Temperature sensor	37
4.1	Heartbeat sensor	40
4.2	Temperature sensor	41
4.3	Micro electro-mechanical system	42
4.4	Bluetooth sensor	43

CHAPTER-1

INTRODUCTION

1.1 GENERAL

A system is something that maintains its existence and functions as a whole through the interaction of its parts. E.g., Body, Mankind, Access Control, etc. A system is a part of the world that a person or group of persons during some time interval and for some purpose choose to regard as a whole, consisting of interrelated components, each component characterized by properties that are selected as being relevant to the purpose.

- Embedded System is a combination of hardware and software used to
- achieve a single specific task.
- Embedded systems are computer systems that monitor, respond to, or control an external environment.
- Environment connected to systems through sensors, actuators and other I/O interfaces.
- Embedded system must meet timing & other constraints imposed on it by environment.

An embedded system is a microcontroller-based, software driven, reliable, real-time control system, autonomous, or human or network interactive, operating on diverse physical variables and in diverse environments and sold into a competitive and cost-conscious market.

An embedded system is not a computer system that is used primarily for devices in our everyday life like Washing Machine, Microwave Ovens, where they are embedded in. processing, not a software system on PC or UNIX, not a traditional business or scientific application. High-end embedded & lower end embedded

systems. High-end embedded system - Generally 32, 64 Bit Controllers used with OS. Examples Personal Digital Assistant and Mobile phones etc. Lower end embedded systems - Generally 8,16 Bit Controllers used with a minimal operating systems and hardware layout designed for the specific purpose. Examples Small controllers.

1.2 DESIGN OF EMBEDDED SYSTEM

Designing an embedded system for a healthcare monitoring system using IoT involves several steps, including hardware and software design, connectivity, data processing, and security. Here's an overview of the process:

1. Define the requirements: Define the scope and requirements of the healthcare monitoring system. Identify the types of healthcare data to be collected, monitored, and processed. Determine the number of sensors, data transmission frequency, power requirements, and data processing requirements.
2. Select the hardware platform: Choose an appropriate hardware platform, such as a microcontroller or a system-on-chip (SoC) with built-in connectivity and processing capabilities. Consider factors such as power consumption, processing power, and available peripherals.
3. Choose the sensors: Select the sensors required for the healthcare monitoring system. Sensors such as temperature, heart rate, blood pressure, and glucose level can be used. The selection of sensors will depend on the requirements of the project.
4. Develop the software: Develop the software required for the system, including the embedded software that runs on the hardware platform and the software for data processing and visualization. The software should be developed using appropriate programming languages and tools.
5. Establish connectivity: Establish connectivity to the Internet or a local network. Depending on the application, connectivity can be through Wi-Fi, Bluetooth, or other wireless protocols.
6. Process the data: The collected data should be processed, analyzed, and stored. The data can be stored locally on the embedded system or uploaded to a cloud-based platform for further analysis.

7. Ensure security: Ensure that the healthcare monitoring system is secure by using appropriate security measures such as encryption, authentication, and access control.
8. Test and deploy: Test the healthcare monitoring system to ensure that it meets the requirements and performs as expected. Once testing is complete, deploy the system in the target environment.
9. In this summary, designing an embedded system for a healthcare monitoring system using IoT involves careful planning, hardware and software development, connectivity, data processing, and security.

Embedded Product Development Life Cycle

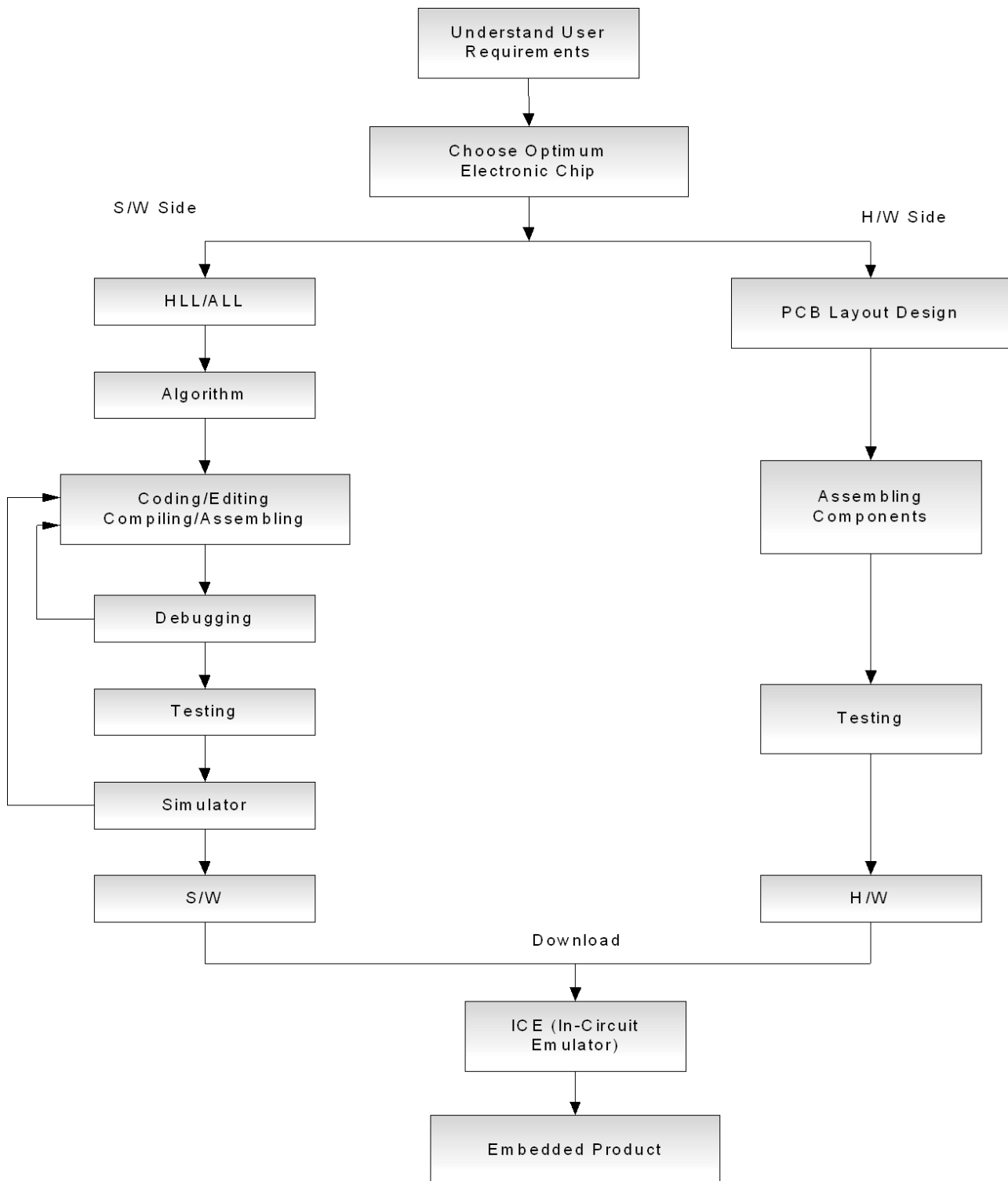


Fig 1.1 Embedded product development life cycle

From the initial state of the project to the final fabrication the design considerations will be taken like the software consideration and the hardware components, sensor, input and output. The electronics usually uses either a microprocessor or a microcontroller. Some large or old systems use general-purpose mainframe computers or minicomputers.

1.3 CLASSIFICATION:

- Real Time Systems.
- RTS is one which has to respond to events within a specified deadline.
- A right answer after the dead line is a wrong answer
- RTS classification
 - Hard Real Time Systems
 - Soft Real Time System

Hard Real Time Systems

- "Hard" real-time systems have very narrow response time.
- Example: Nuclear power system, Cardiac pacemaker.

Soft Real Time System

- "Soft" real-time systems have reduced constraints on "lateness" but still must operate very quickly and repeatably.
- Example: Railway reservation system - takes a few extra seconds the data remains valid.

CHAPTER-2

LITERATURE SURVEY

2.1 MAJOR FINDINGS FROM LITERATURE SURVEY

Secure health monitoring using medical wireless sensor networks

In this paper, we are addressing security requirements of health monitoring using medical sensor networks and proposed a secure framework called “SHM” using wireless sensor networks. SHM provides security services such as confidentiality, authenticity and integrity to patient data at low-cost. To provide confidentiality we have used PingPong-128 stream cipher. Authenticity and data integrity is achieved by Ping-Pong-MAC (message authentication code). We have designed a real-time prototype with ECG and tri-axial accelerometer sensors on Telos-mote for evaluating the proposed secure framework and our results confirm their feasibility.

A Novel and Lightweight System to Secure Wireless Medical Sensor Networks.

IEEE Journal of Biomedical and Health Informatics

Wireless medical sensor networks (MSNs) are a key enabling technology in e-healthcare that allows the data of a patient's vital body parameters to be collected by the wearable or implantable biosensors. However, the security and privacy protection of the collected data is a major unsolved issue, with challenges coming from the stringent resource constraints of MSN devices, and the high demand for both security/privacy and practicality. In this paper, we propose a lightweight and secure system for MSNs. The system employs hash-chain based key updating mechanism

and proxy-protected signature technique to achieve efficient secure transmission and fine-grained data access control. Furthermore, we extend the system to provide backward secrecy and privacy preservation. Our system only requires symmetric-key encryption/decryption and hash operations and is thus suitable for the low-power sensor nodes. This paper also reports the experimental results of the proposed system in a network of resource-limited motes and laptop PCs, which show its efficiency in practice. To the best of our knowledge, this is the first secure data transmission and access control system for MSNs until now.

Effective Movement Classification for Context Awareness in Medical Applications Networking

Today new medical applications evolve from large stationary devices to small and smart mobile systems that will enable e.g., more efficient post operative health care. These mobile systems that benefit from ongoing miniaturization and energy savings in hardware will allow continuous monitoring of patients accompanying and supporting therapy and detect emergency situations. Additionally, to vital data, the physiological load or the context of patients are important to analyze and understand recorded data of mobile patients. Current approaches for movement classification aim to detect very specific movement patterns and are dependent on precise sensor placements and are thus not suited for everyday usage. Therefore, we developed a movement detection and classification algorithm that can be easily integrated in existing embedded devices. Using data from a single accelerometer embedded into the device, the algorithm can classify between different movement patterns - the "context" - of the monitored person. We will describe the hardware and the algorithm and will provide first evaluation results demonstrating the effectiveness of this approach for providing context awareness in mobile medical applications in real-time.

Healthcare WSN: Cluster Elections and Selective Forwarding Defense

Wireless sensor networks are an array of different components. Upgrading a single component may not be enough. It is necessary to take each component and apply

the best possible techniques to upgrade the system. This paper looks at the configurations necessary for a medical based WSN with particular focus on clustering and routing. It includes an implementation of a WSN architecture in Contiki operating system using Tmote Sky and open mote technologies. The underlying power consumption related to this architecture is analyzed and real-world measurements are presented. Finally, the case of a modified secure routing algorithm is introduced with simulation-based results showing single and collaborative selective forwarding detection/correction, and network latency. Additionally, a solution to the problem of malicious nodes dropping control messages has been provided. The uniqueness of the paper resides in the implementation of monitoring mechanism without any watchdog nodes. Moreover, to the best of the authors' knowledge, this mechanism has not been implemented on the Contiki OS.

Data security and privacy in wireless body area networks

The wireless body area network has emerged as a new technology for e-healthcare that allows the data of a patient's vital body parameters and movements to be collected by small wearable or implantable sensors and communicated using short-range wireless communication techniques. WBAN has shown great potential in improving healthcare quality, and thus has found a wide range of applications from ubiquitous health monitoring and computer assisted rehabilitation to emergency medical response systems. The security and privacy protection of the data collected from a WBAN, either while stored inside the WBAN or during their transmission outside of the WBAN, is a major unsolved concern, with challenges coming from stringent resource constraints of WBAN devices, and the high demand for both security/privacy and practicality/usability. In this article we look into two important data security issues: secure and dependable distributed data storage, and fine-grained distributed data access control for sensitive and private patient medical data. We discuss various practical issues that need to be taken into account while fulfilling the security and privacy requirements. Relevant solutions in sensor networks and WBANs are surveyed, and their applicability is analyzed.

2.2 PROBLEM IDENTIFICATION

IoT-based health monitoring systems have several benefits. These systems can remotely Bluetooth. The cloud server processes the data and provides real-time health status updates to healthcare providers or family members. The system can also send alerts in case of any abnormality in the patient's health. The problem to be addressed in an IoT-based healthcare monitoring system is the need for continuous and remote monitoring of patient health parameters. Traditional healthcare systems are typically limited to in-person visits, which can be inconvenient and time-consuming for both patients and healthcare providers. Additionally, these systems may not provide real-time monitoring, leading to delayed interventions in case of any abnormality in the patient's health.

The problem to be addressed in an IoT-based healthcare monitoring system is the need for continuous and remote monitoring of patient health parameters. Traditional healthcare systems are typically limited to in-person visits, which can be inconvenient and time-consuming for both patients and healthcare providers. Additionally, these systems may not provide real-time monitoring, leading to delayed interventions in case of any abnormality in the patient's health. Furthermore, the existing healthcare monitoring systems can be expensive and require frequent maintenance, which may not be feasible for all patients, especially those from low-income backgrounds. Also, data security and privacy can be a concern with the storage and transfer of sensitive health data over the internet. Therefore, an IoT-based healthcare monitoring system can address these issues by providing a cost-effective and convenient solution for continuous and remote monitoring of patient health parameters, such as heart rate, blood pressure, and glucose level. It can also provide real-time notifications to healthcare providers and caregivers in case of any abnormality, enabling timely interventions and potentially preventing severe health consequences. However, the system must ensure adequate security measures to protect patient data from unauthorized access and misuse.

2.3 SCOPE OF THE PROJECT

- The system would need to be designed to collect data from various sources such as wearable devices, medical equipment, and sensors. This data would include vital signs such as heart rate, blood pressure, and temperature, as well as other health-related data such as physical activity, sleep patterns, and medication adherence.
- Data analysis and processing: The system would need to be able to process and analyze the data collected from various sources to identify any abnormal trends or patterns that could indicate a potential health issue.
- Communication: The system would need to be able to communicate with healthcare providers and/or caregivers to notify them of any potential health issues that have been detected. This could be done through a mobile app, web portal, or other communication channels.
- Security and privacy: The system would need to ensure the security and privacy of patient data by implementing measures such as encryption and access controls.
- User interface: The system would need to have an intuitive and user-friendly interface that allows patients and healthcare providers to access and interpret the data collected by the system.

2.4 METHODOLOGY

- Identifying the healthcare needs: The first step is to identify the healthcare needs of the target population. This includes determining the type of health conditions that require monitoring and the specific parameters that need to be tracked.
- Selecting the appropriate sensors: Based on the identified healthcare needs, appropriate sensors should be selected. For example, for monitoring blood glucose levels, a glucose meter can be used, while for tracking heart rate and blood pressure, a wearable device with sensors can be used.
- Developing the IoT platform: An IoT platform needs to be developed to collect, store, and analyze the data generated by the sensors. This platform should be secure and scalable to accommodate the data generated by a large number of devices.

- **Connecting the sensors:** The sensors need to be connected to the IoT platform. This can be done through a wireless network such as Wi-Fi, Bluetooth, or cellular networks.
- **Data collection and analysis:** The data collected from the sensors is then processed and analyzed to provide insights into the patient's health status. Machine learning algorithms can be used to analyze the data and provide real-time alerts and notifications to healthcare providers.
- **Implementation and deployment:** The IoT platform should be implemented and deployed in the healthcare environment. The platform should be user-friendly and easy to use for both patients and healthcare providers.
- **Maintenance and support:** Once the system is deployed, it should be maintained and supported to ensure its continuous operation. This includes updating the software, replacing faulty sensors, and providing technical support to users.

2.5 STANDARD

The IEEE standard adapted for this project are IEEE 11073-10406-2018 , IEEE 2410-2020 , IEEE 802.11, IEEE 2413-2019 and IEEE P2933.1.1-2021.

IEEE 11073-10406-2018: This standard defines a communication protocol for personal health devices that allows for interoperability and integration with other healthcare systems.

IEEE 2410-2020: This standard provides guidelines for the design of connected health devices and systems, including those that incorporate IoT technologies.

IEEE 802.11: This standard defines wireless LAN technologies that could be used for connecting IoT devices in a home quarantine monitoring system.

IEEE 2413-2019: This standard provides a framework for the development and implementation of IoT-based healthcare systems. It covers aspects such as data security, interoperability, and system architecture.

IEEE P2933.1.1-2021: This standard provides guidelines for an IoT-enabled remote health management system that can be used for monitoring individuals in home quarantine. It outlines a model architecture that includes sensors, communication protocols, data storage and analysis, and user interfaces.

2.4.1 TERMINOLOGY

IEEE 11073-10406-2018 is part of the IEEE 11073 family of standards for medical device communication, and it specifically defines a communication protocol for personal health devices. The standard provides a common framework for data exchange between personal health devices and other healthcare systems, such as electronic health records and hospital information systems. The communication protocol defined in IEEE 11073-10406-2018 enables interoperability and integration of personal health devices with other healthcare systems, which can improve the efficiency and accuracy of healthcare delivery. It also supports the collection of standardized data from personal health devices, which can be used for clinical research and analysis. Overall, IEEE 11073-10406-2018 is an important standard for the development of personal health devices and the integration of these devices with other healthcare systems. Its adoption can help to improve the quality and accessibility of healthcare services for patients.

IEEE 2410-2020 is a standard that provides guidelines for the design and development of connected health devices and systems, including those that incorporate IoT technologies. The standard aims to promote the safe and effective use of connected health devices by providing guidance on their design, implementation, and use.

The standard covers a range of topics related to connected health devices and systems, including:

- Design considerations, such as user needs, safety, security, and privacy
- Device and system requirements, such as interoperability, data exchange, and data quality.
- Implementation guidelines, such as software and hardware design, testing, and validation.
- User interface design, such as usability and accessibility.
- System maintenance, such as software updates and device management.

By providing guidelines for the design and development of connected health devices and systems, IEEE 2410-2020 can help to ensure that these devices are safe, effective, and user-friendly. It can also help to facilitate the integration of connected health devices and systems into existing healthcare workflows, ultimately improving patient outcomes and healthcare delivery.

IEEE 802.15.4 standard defines a low-power wireless personal area network (WPAN) that operates in the 2.4 GHz, 915 MHz, and 868 MHz frequency bands. It is designed for low-cost, low-power consumption, and low-data-rate wireless connectivity among devices, making it ideal for IoT applications such as home quarantine monitoring systems.

The standard supports various network topologies, including star, tree, and mesh, allowing for flexible deployment of devices. It also includes features such as channel hopping, acknowledgments, and retransmissions to ensure reliable communication even in noisy environments.

By using IEEE 802.15.4 for the communication between IoT devices in a home quarantine monitoring system, it is possible to create a network that is efficient, reliable, and secure, while consuming minimal power. This can help to ensure that the devices can operate for extended periods without requiring frequent battery replacements or charging, while also providing timely and accurate monitoring of patients in quarantine.

IEEE 2413-2019 provides a comprehensive framework for the development and implementation of IoT-based healthcare systems. This standard covers a range of aspects related to the design, implementation, and operation of such systems, including:

System architecture: This standard provides guidance on the overall architecture of an IoT-based healthcare system, including the various components and subsystems that are involved.

Interoperability: IEEE 2413-2019 emphasizes the importance of interoperability in IoT-based healthcare systems, and provides guidelines for achieving interoperability between different components and subsystems.

Data security and privacy: This standard provides guidance on the various security and privacy concerns that must be addressed in IoT-based healthcare systems. This includes guidelines for data encryption, authentication, and access control.

Device integration: IEEE 2413-2019 provides guidance on the integration of various IoT devices and sensors into a healthcare system, including guidelines for device selection, deployment, and management.

Data analytics: This standard covers the use of data analytics in IoT-based healthcare systems, including guidelines for data analysis and visualization.

IEEE P2933.1.1-2021 is a standard that provides guidelines for an IoT-enabled remote health management system that can be used for monitoring individuals in home quarantine. The standard outlines a model architecture that includes sensors, communication protocols, data storage and analysis, and user interfaces.

Some of the key features of this standard include:

Sensor deployment and data collection: The standard provides guidance on the deployment of sensors for monitoring vital signs and other health-related data, as well as the collection of this data using various communication protocols.

Data storage and analysis: IEEE P2933.1.1-2021 provides guidelines for the storage and analysis of health-related data, including the use of cloud-based platforms for data management and analysis.

User interface design: The standard includes guidelines for the design of user interfaces, including mobile applications and web-based portals, that allow users to access their health-related data and communicate with healthcare providers.

Security and privacy: IEEE P2933.1.1-2021 includes guidelines for ensuring the security and privacy of health-related data in IoT-based remote health management systems.

2.6 TRADE-OFFS

- **Privacy and Security:** IoT devices generate a large amount of data, and this data needs to be transmitted and stored securely. There is always a risk of data breaches or cyber-attacks, which can compromise the privacy of patients' health information. Healthcare organizations need to invest in robust security measures to ensure that patient data is protected at all times.

- **Cost:** The cost of developing and deploying an IoT-based healthcare monitoring system can be high. The cost of sensors, connectivity devices, and data storage can add up quickly. This can make it difficult for smaller healthcare organizations or individuals to implement such a system.
- **Reliability:** IoT devices can be prone to failures and malfunctions, which can impact the reliability of the healthcare monitoring system. It is important to ensure that the devices are regularly maintained and updated to ensure their continued functionality.
- **Compatibility:** There are many different types of IoT devices and platforms available, and not all of them may be compatible with each other. This can make it difficult to integrate different devices and platforms into a single healthcare monitoring system.
- **User Experience:** Patients may find it difficult to use IoT devices and platforms, especially if they are not familiar with technology. Healthcare providers need to ensure that the devices and platforms are user-friendly and easy to use to ensure patient engagement and compliance.

CHAPTER 3

PROJECT DESCRIPTION

3.1 EXISTING SYSTEM:

Based on visiting hospitals, it was approved of the effectiveness of this project and its ability to facilitate between the patient and his doctor. Utilizing the available services of GSM and GPS technologies to build a smart Based on visiting hospitals, it was approved of the effectiveness of this project and its ability to facilitate communication between the patient and his doctor. Utilizing the available services of GSM and GPS technologies to build a smart.

3.2 PROPOSED SYSTEM:

The patient's Medical Data has ability to transmit real-time scanned data to the medical center server. The doctor can access patient medical data directly from the medical center server through Internet or a personal computer connected to the internet

Mainly, the proposed system aims to cover an end-to-end smart health application that can be build up from two functional blocks. However, the main function of the first building is to gather all sensory data that are related to the monitored persons, whereas the second block function are to store, process and present the resulted information of this stage the doctors and nursery staff that are fallowing the case of monitored person.

Architecture block diagram

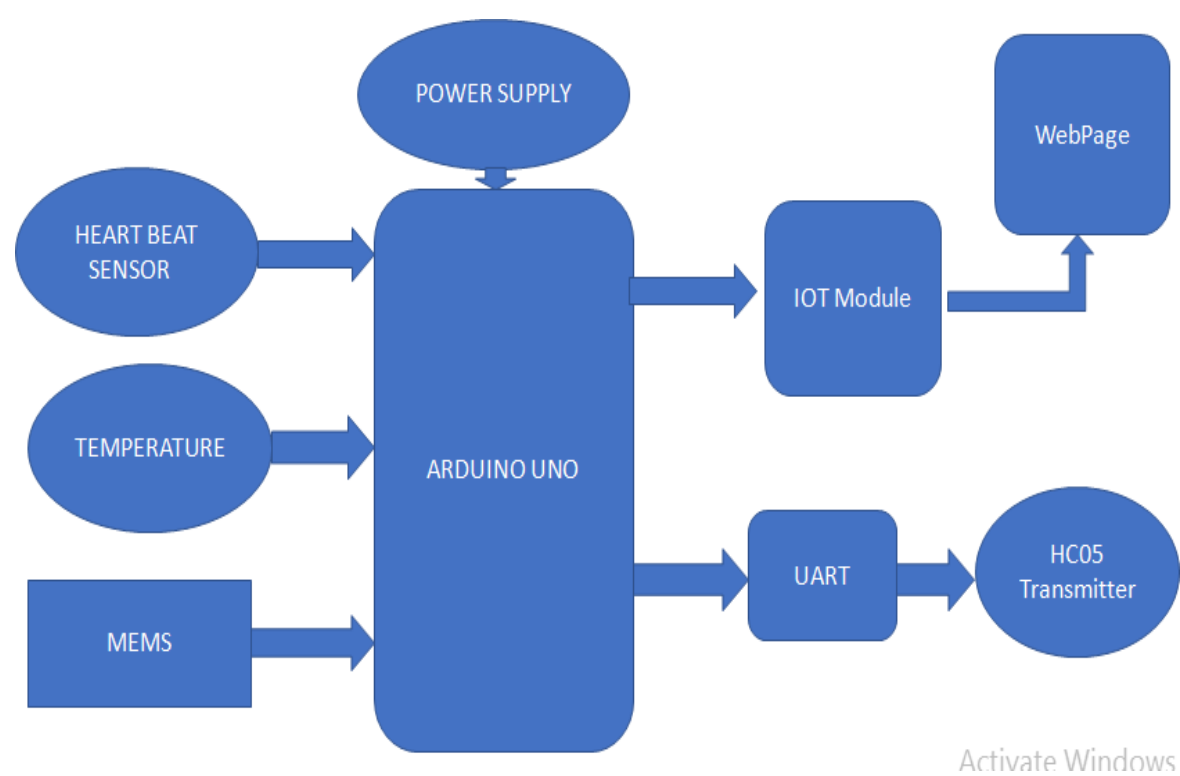


Fig 3.1 Architecture block diagram

Monitoring diagram

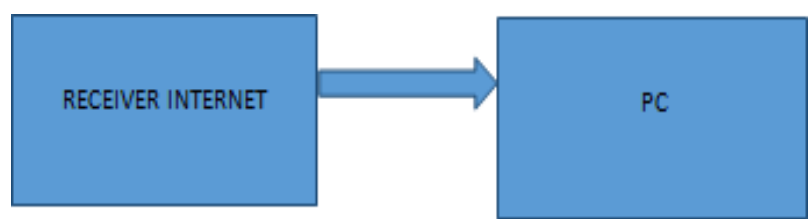


Fig 3.2 Monitoring section

POWER SUPPLY

All electronic circuits work only in low DC voltage, so we need a power supply unit to provide the appropriate voltage supply for their proper functioning. This unit consists of transformer, rectifier, filter & regulator. AC voltage of typically 230volts rms is connected to a transformer voltage down to the level to the desired ac voltage. A diode rectifier that provides the full wave rectified voltage that is initially filtered by a simple capacitor filter to produce a dc voltage. This resulting dc voltage usually has some ripple or ac voltage variation. A regulator circuit can use this dc input to provide dc voltage that not only has much less ripple voltage but also remains the same dc value even the dc voltage varies somewhat, or the load connected to the output dc voltages change.

TRANSFORMER:

A transformer is a static piece of which electric power in one circuit is transformed into electric power of same frequency in another circuit. It can raise or lower the voltage in the circuit, but with a corresponding decrease or increase in current. It works with the principle of mutual induction. In our project we are using a step-down transformer to providing a necessary supply for the electronic circuits. Here we step down a 230volts ac into 12volts ac.

RECTIFIER:

A dc level obtained from a sinusoidal input can be improved 100% using a process called full wave rectification. Here in our project for full wave rectification we use bridge rectifier. From the basic bridge configuration, we see that two diodes (say D2 & D3) are conducting while the other two diodes (D1 & D4) are in off state during the period $t = 0$ to $T/2$. Accordingly for the negative cycle of the input the conducting diodes are D1 & D4. Thus, the polarity across the load is the same.

In the bridge rectifier the diodes may be of variable types like 1N4001, 1N4003, 1N4004, 1N4005, 1N4007 etc.... can be used. But here we use 1N4007, because it can withstand up to 1000v.

FILTERS:

In order to obtain a dc voltage of 0 Hz, we have to use a low pass filter. So that a capacitive filter circuit is used where a capacitor is connected at the rectifier output & a dc is obtained across it. The filtered waveform is essentially a dc voltage with negligible ripples & it is ultimately fed to the load.

REGULATORS:

The output voltage from the capacitor is more filtered & finally regulated. The voltage regulator is a device, which maintains the output voltage constant irrespective of the change in supply variations, load variations & temperature changes. Here we use fixed voltage regulator namely LM7805. The IC LM7805 is a +5v regulator which is used for microcontroller.

FEATURES AND DESCRIPTION OF REGULATORS

- Output Current up to 1A
- Output Voltages of 5, 6, 8, 9, 10, 12, 15, 18, 24V
- Thermal Overload Protection
- Short Circuit Protection
- Output Transistor Safe Operating Area Protection

CIRCUIT DIAGRAM:

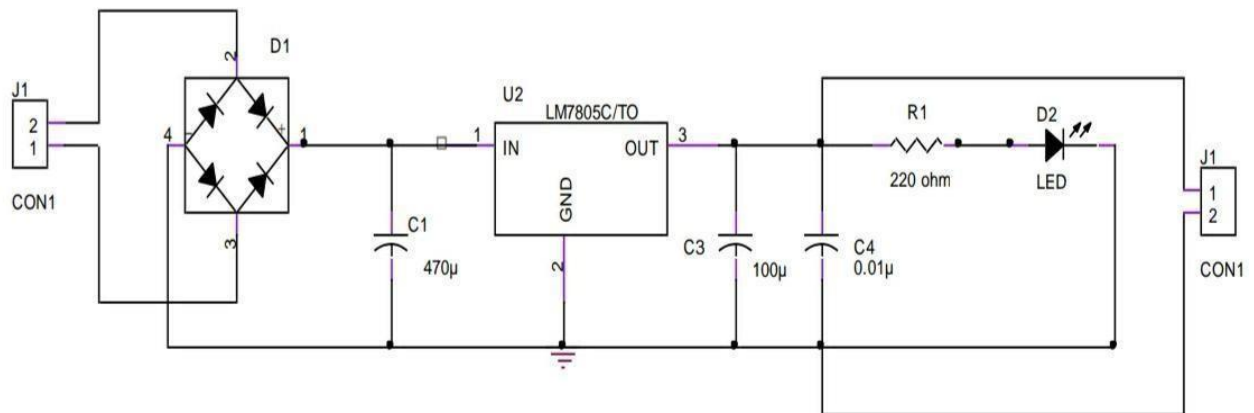


Fig 3.3 Circuit Diagram of Regulator

3.3 ARDUINO UNO AND ITS PROGRAMMING

Arduino is a tool for making computers that can sense and control more of the physical world than your desktop computer. It's an open-source physical computing platform based on a simple microcontroller board, and a development environment for writing software for the board.

Arduino can be used to develop interactive objects, taking inputs from a variety of switches or sensors, and controlling a variety of lights, motors, and other physical outputs. Arduino projects can be stand-alone, or they can communicate with software running on your computer. The boards can be assembled by hand or purchased preassembled; the open-source IDE can be downloaded for free.

The Arduino programming language is an implementation of Wiring, a similar physical computing platform, which is based on the Processing multimedia programming environment.

ARDUINO UNO



Fig 3.4 ARDUINO UNO

3.3.1 Overview

The Arduino microcontroller is an easy to use yet powerful single board computer that has gained considerable traction in the hobby and professional market. The Arduino is open-source, which means hardware is reasonably priced and development software is free. This guide is for students in ME 2011, or students anywhere who are confronting the Arduino for the first time. For advanced Arduino users, prowl the web; there are lots of resources.

This guide covers the Arduino Uno board (Spark fun DEV-09950, \$29.95), a good choice for students and educators. With the Arduino board, you can write programs and create interface circuits to read switches and other sensors, and to control

motors and lights with very little effort. Many of the pictures and drawings in this guide were taken from the documentation.

The Duemilanove board features an Atmel ATmega328 microcontroller operating at 5 V with 2 kb of RAM, 32 kb of flash memory for storing programs and 1 kb of EEPROM for storing parameters. The clock speed is 16 MHz, which translates to about executing about 300,000 lines of C source code per second. The board has 14 digital I/O pins and 6 analog input pins. There is a USB connector for talking to the host computer and a DC power jack for connecting an external 6-20 V power source, for example a 9 V battery, when running a program while not connected to the host computer. Headers are provided for interfacing to the I/O pins using 22 g solid wire or header connectors.

The Arduino programming language is a simplified version of C/C++. If you know C, programming the Arduino will be familiar. If you do not know C, no need to worry as only a few commands are needed to perform useful functions. An important feature of the Arduino is that you can create a control program on the host PC, download it to the Arduino and it will run automatically. Remove the USB cable connection to the PC, and the program will still run from the top each time you push the reset button. Remove the battery and put the Arduino board in a closet for six months. When you reconnect the battery, the last program you stored will run. This means that you connect the board to the host PC to develop and debug your program, but once that is done, you no longer need the PC to run the program.

The Arduino Uno is a microcontroller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

3.3.2 Summary:

- Microcontroller ATmega328
- Operating Voltage 5V
- Input Voltage (recommended) 7-12V
- Input Voltage (limits) 6-20V
- Digital I/O Pins 14 (of which 6 provide PWM output)
- Analog Input Pins 6
- DC Current per I/O Pin 40 mA
- DC Current for 3.3V Pin 50 mA
- Flash Memory 32 KB (ATmega328) of which 0.5 KB used by bootloader
- SRAM 2 KB (ATmega328)
- EEPROM 1 KB (ATmega328)
- Clock Speed 16 MHz

3.3.3 Pin Configuration

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall- wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts

The power pins are as follows:

- VIN. The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- 5V. this pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board.
- 3V3. A 3.3-volt supply generated by the on-board regulator. Maximum current draw is 50 mA.

IOREF GND. Ground pins

This pin on the Arduino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs for working with the 5V or 3.3V.

Memory

The AT mega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2KB of SRAM and 1 KB of EEPROM.

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using pin Mode(), digital Write(), and digital Read() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull- up resistor (disconnected by default) of 20-50 ohms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and Trans mit (TX) TTL serial data.

These pins are connected to the corresponding pins of the ATmega8U2 USB-to- TTL Serial chip.

External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attach Interrupt () function for details

PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analog Write () function.

SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.

LED: 13. There is a built- in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e., 1024 different values). By default, they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analog Reference () function. Additionally, some pins have specialized functionality:

- **TWI: A4 or SDA pin and A5 or SCL pin.** Support TWI communication using the Wire library.

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with analog Reference ().

Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board

Repositories that extend core programming functionality. Libraries let you re-use code without having to physically copy and paste it into all your programs. The standard Arduino software installation comes with several libraries you may use, and you can download others from the Arduino support pages and from third-party websites that publish Arduino library code. A good example of a library you'll use with the Robot and likely many other robot projects is

Servo. This library allows you to connect one or more hobby R/C servos to the Arduino's digital I/O pins. The Servo library comes with the standard Arduino installation package Library->Servo. This adds the line

```
#include <Servo.h>
```

Which tells the Arduino IDE that you wish to include the Servo library in your sketch. With the functionality of the library now available to you, you can use its various functions to control one or more servos. For example, you can use the write function to rotate a servo to a specific position, from 0 to 180 degrees. The following code `myServo.write(90);`

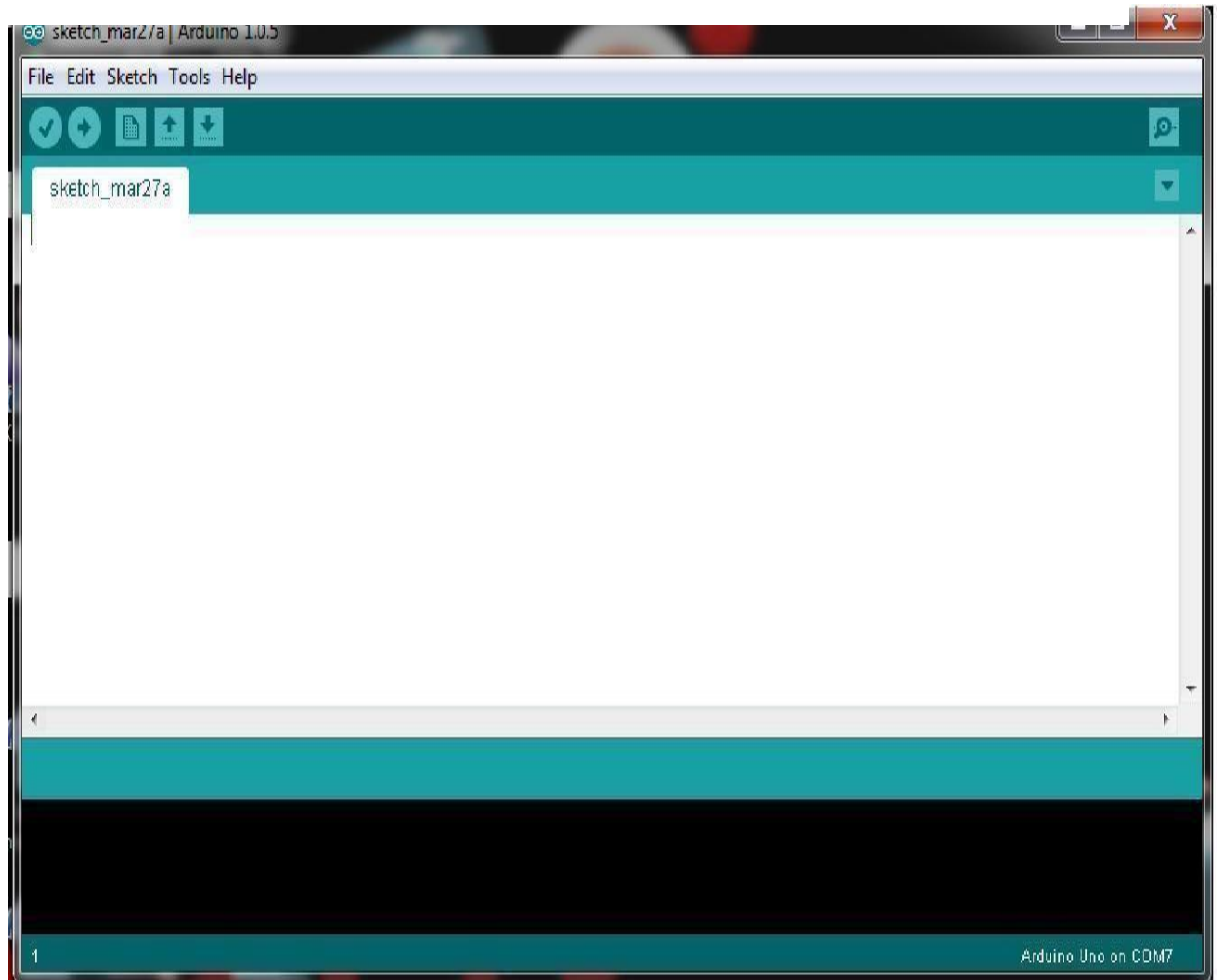
Moves a servo to its midpoint, or 90 degree position. Structurally, Arduino sketches are very straightforward and are pretty easy to read and understand. The Arduino program contains two main parts: `setup ()` and `loop ()`. These are programming functions that do what their names suggest: `setup ()` sets up the Arduino hardware, such as specifying which I/O lines you plan to use, and whether

They are inputs or outputs. The `loop ()` function is repeated endlessly when the Arduino is operating.

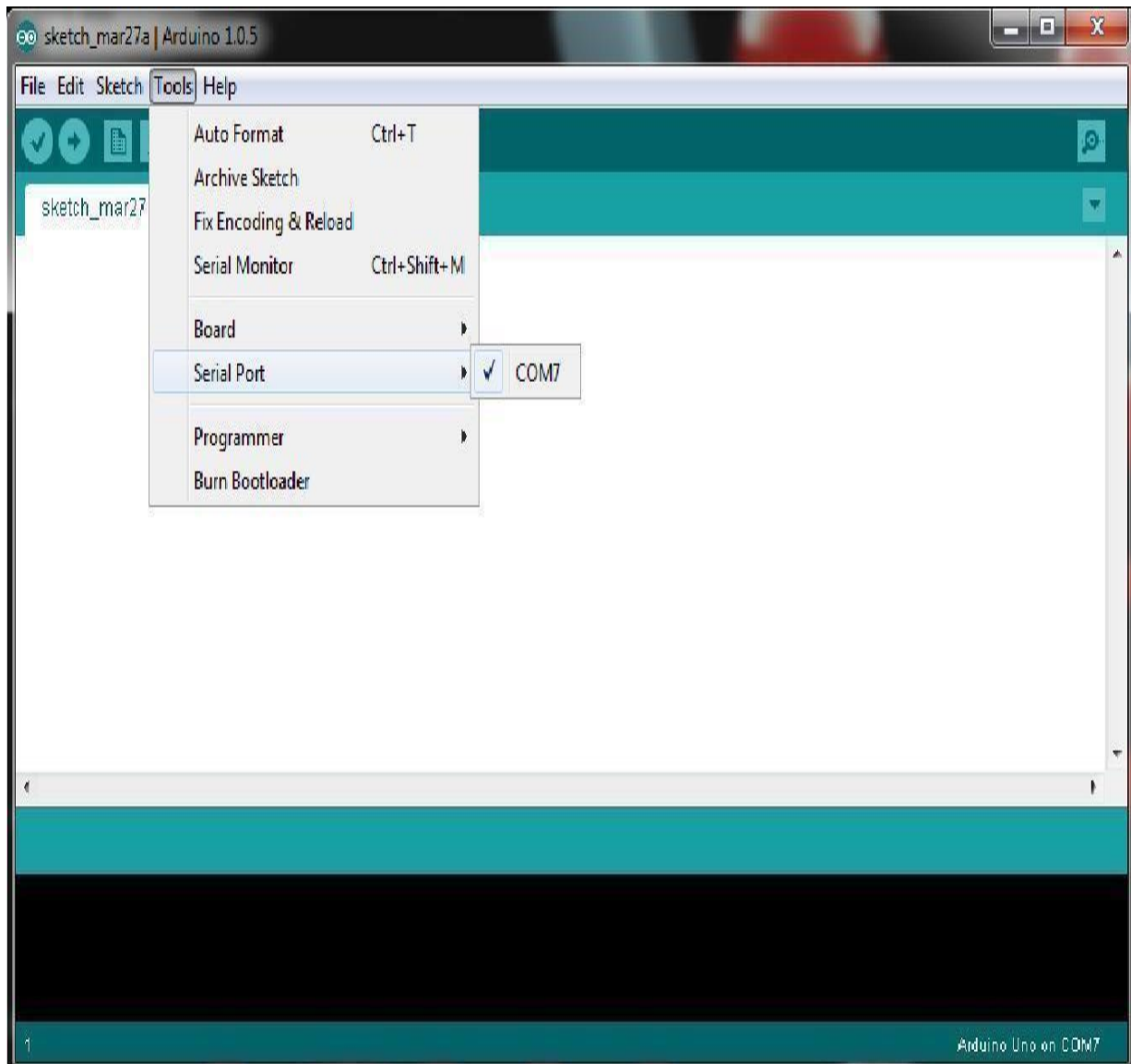
Arduino IDE (Integrated development environment) is used to write the program and dump into the Arduino board

ARDUINO SOFTWARE:

1. Open Arduino IDE as shown below

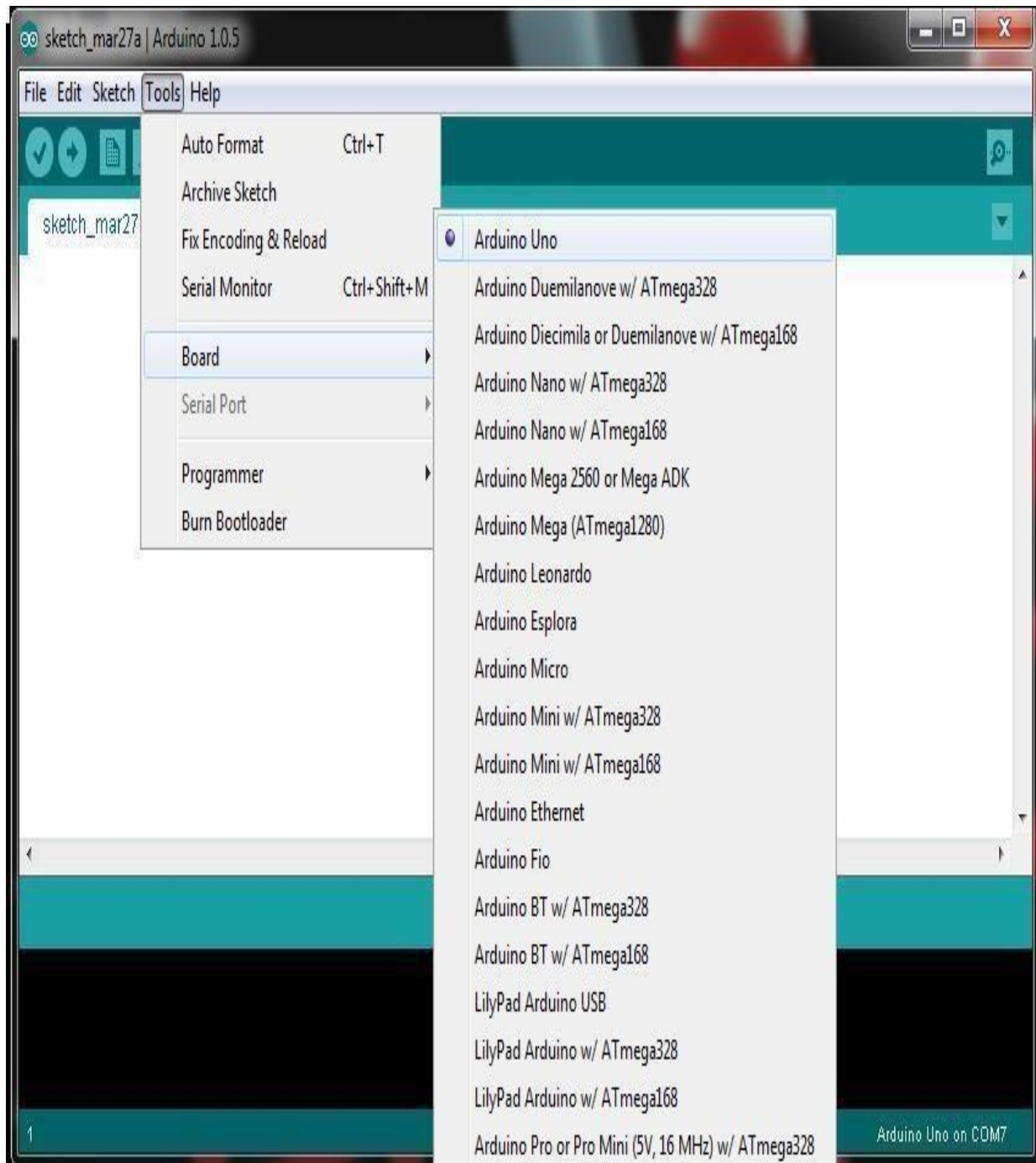


2. Select the COM Port from tool



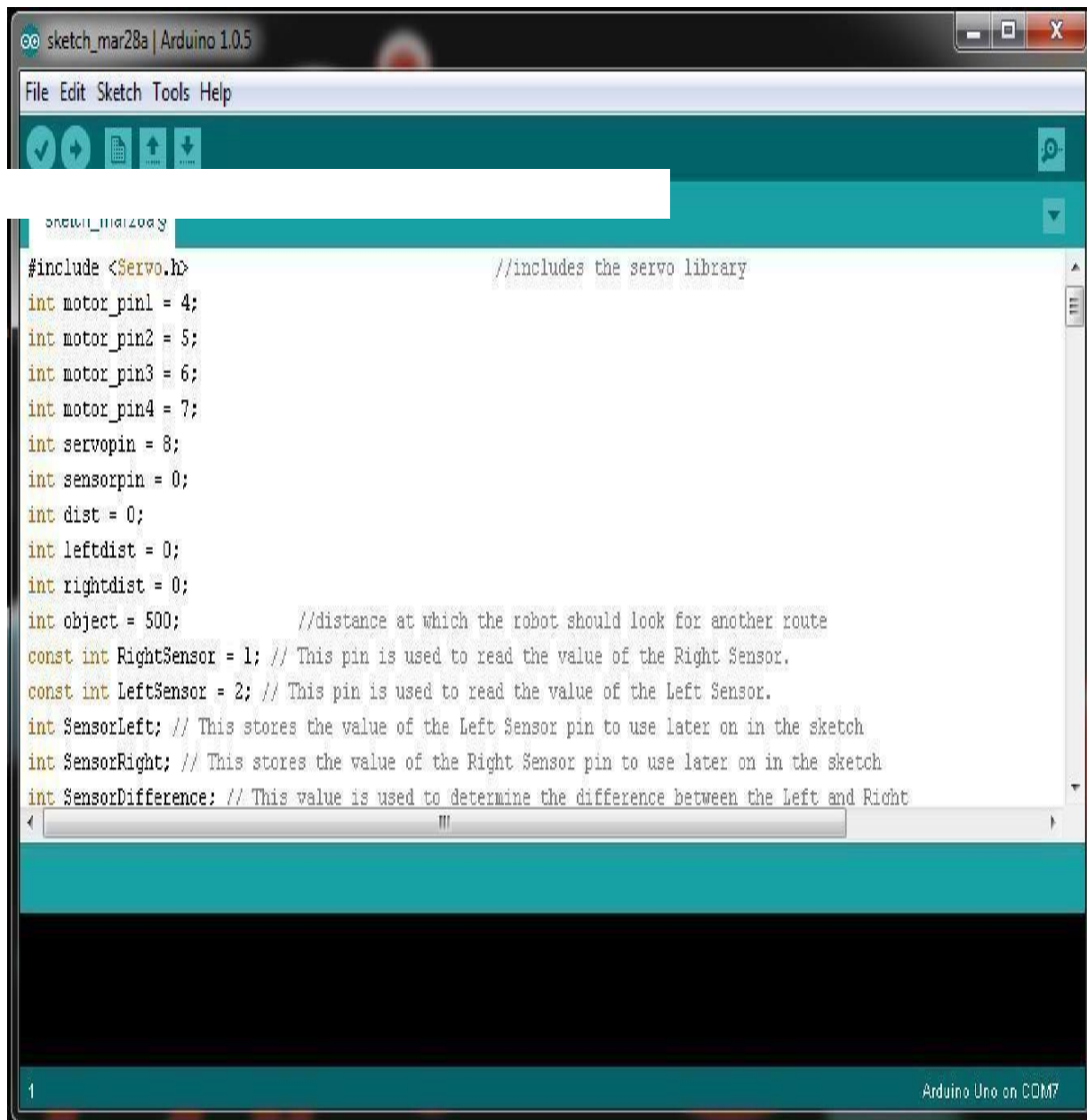
3. Select the required Arduino board from Tools as shown below

Select the required Arduino board



4. Write the sketch in Arduino IDE

Sketch in Arduino IDE



```
sketch_mar28a | Arduino 1.0.5
File Edit Sketch Tools Help

#include <Servo.h> //includes the servo library
int motor_pin1 = 4;
int motor_pin2 = 5;
int motor_pin3 = 6;
int motor_pin4 = 7;
int servopin = 8;
int sensorpin = 0;
int dist = 0;
int leftdist = 0;
int rightdist = 0;
int object = 500; //distance at which the robot should look for another route
const int RightSensor = 1; // This pin is used to read the value of the Right Sensor.
const int LeftSensor = 2; // This pin is used to read the value of the Left Sensor.
int SensorLeft; // This stores the value of the Left Sensor pin to use later on in the sketch
int SensorRight; // This stores the value of the Right Sensor pin to use later on in the sketch
int SensorDifference; // This value is used to determine the difference between the Left and Right

1 Arduino Uno on COM7
```

5. Compile and upload the Sketch to Arduino board
upload the Sketch to Arduino board



TEMPERATURE SENSOR

Temperature is the most often-measured environmental quantity. This might be expected since most physical, electronic, chemical, mechanical, and biological systems are affected by temperature. Certain chemical reactions, biological processes, and even electronic circuits perform best within limited temperature ranges. Temperature is one of the most commonly measured variables and it is therefore not surprising that there are many ways of sensing it. Temperature sensing can be done either through direct contact with the heating source, or remotely, without direct contact with the source using radiated energy instead. There are a wide variety of temperature sensors on the market today, including Thermocouples, Resistance Temperature Detectors (RTDs), Thermistors, Infrared, and Semiconductor Sensors.

THERMOCOUPLE:

It is a type of temperature sensor, which is made by joining two dissimilar metals at one end. The joined end is referred to as the HOT JUNCTION. The other end of these dissimilar metals is referred to as the COLD END or COLD JUNCTION. The cold junction is actually formed at the last point of thermocouple material. If there is a difference in temperature between the hot junction and cold junction, a small voltage is created. This voltage is referred to as an EMF (electro-motive force) and can be measured and in turn used to indicate temperature.

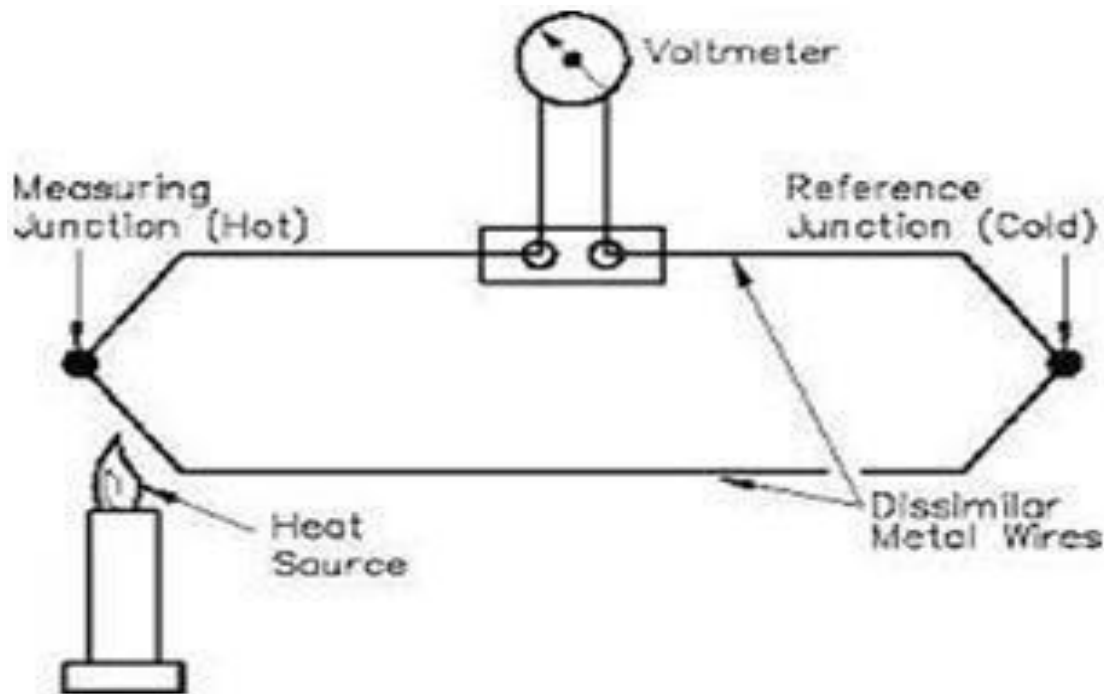


Fig 3.5 Thermocouple diagram

The RTD is a temperature sensing device whose resistance changes with temperature. Typically built from platinum, though devices made from nickel or copper are not uncommon, RTDs can take many different shapes like wire wound, thin film. To measure the resistance across an RTD, apply a constant current, measure the resulting voltage, and determine the RTD resistance. RTDs exhibit fairly linear resistance to temperature curves over their operating regions, and any nonlinearity are highly predictable and repeatable. The PT100 RTD evaluation board uses surface mount RTD to measure temperature. An external 2, 3 or 4-wire PT100 can also be associated with measure temperature in remote areas. The RTDs are biased using a constant current source. So as to reduce self-heat due to power dissipation, the current magnitude is moderately low. The circuit shown in figure is the constant current source uses a reference voltage, one amplifier, and a PNP transistor.

Thermistors:

Similar to the RTD, the thermistor is a temperature sensing device whose resistance changes with temperature. Thermistors, however, are made from semiconductor materials. Resistance is determined in the same manner as the RTD, but thermistors exhibit a highly nonlinear resistance vs. temperature curve. Thus, in the thermistors operating range we can see a large resistance change for a very small temperature change. This makes for a highly sensitive device, ideal for set-point applications.

Semiconductor sensors:

They are classified into different types like Voltage output, Current output, Digital output, Resistance output silicon and Diode temperature sensors. Modern semiconductor temperature sensors offer high accuracy and high linearity over an operating range of about 55°C to +150°C. Internal amplifiers can scale the output to convenient values, such as 10mV/°C. They are also useful in cold-junction compensation circuits for wide temperature range thermocouples. A brief details about this type of temperature sensor are given below.

Sensor ICS:

There are a wide variety of temperature sensor ICs that are available to simplify the broadest possible range of temperature monitoring challenges. These silicon temperature sensors differ significantly from the above mentioned types in a couple of important ways. The first is operating temperature range. A temperature sensor IC can operate over the nominal IC temperature range of -55°C to +150°C. The second major difference is functionality.

A silicon temperature sensor is an integrated circuit, and can therefore include extensive signal processing circuitry within the same package as the sensor. There is no need to add compensation circuits for temperature sensor Ics. Some of these are analogue circuits with either voltage or current output. Others combine analogue-

sensing circuits with voltage comparators to provide alert functions. Some other sensor ICs combine analogue-sensing circuitry with digital input/output and control registers, making them an ideal solution for microprocessor-based systems.

Digital output sensor usually contains a temperature sensor, analog-to-digital converter (ADC), a two-wire digital interface and registers for controlling the IC's operation. Temperature is continuously measured and can be read at any time. If desired, the host processor can instruct the sensor to monitor temperature and take an output pin high (or low) if temperature exceeds a programmed limit. Lower threshold temperature can also be programmed and the host can be notified when temperature has dropped below this threshold. Thus, digital output sensor can be used for reliable temperature monitoring in microprocessor-based systems.

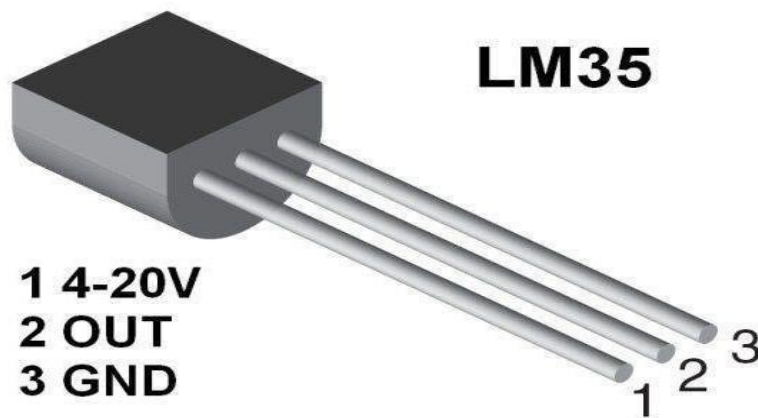


Fig 3.6 Temperature sensor

Above temperature sensor has three terminals and required Maximum of 5.5 V supply. This type of sensor consists of a material that performs the operation according to temperature to vary the resistance. This change of resistance is sensed by circuit and it calculates temperature. When the voltage increases then the temperature also rises.

We can see this operation by using a diode. Temperature sensors directly connected to microprocessor input and thus capable of direct and reliable communication with microprocessors. The sensor unit can communicate effectively with low-cost processors without the need of A/D converters.

An example for a temperature sensor is **LM35**. The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius temperature. The LM35 is operates at -55° to $+120^{\circ}\text{C}$. The basic centigrade temperature sensor ($+2^{\circ}\text{C}$ to $+150^{\circ}\text{C}$) is shown in figure below.

Features of LM35 Temperature Sensor:

- Calibrated directly in $^{\circ}$ Celsius (Centigrade)
- Rated for full -55° to $+150^{\circ}\text{C}$ range
- Suitable for remote applications
- Low cost due to wafer-level trimming
- Operates from 4 to 30 volts
- Low self-heating,
- $\pm 1/4^{\circ}\text{C}$ of typical nonlinearity

Operation of LM35:

- The LM35 can be connected easily in the same way as other integrated circuit temperature sensors. It can be stuck or

established to a surface and its temperature will be within around the range of $\pm 0.01^{\circ}\text{C}$ of the surface temperature.

- This presumes that the ambient air temperature is just about the same as the surface temperature; if the air temperature were much higher or lower than the surface temperature, the actual temperature of the LM35 die would be at an intermediate temperature between the surface temperature and the air temperature.
- The temperature sensors have well known applications in environmental and process control and also in test, measurement and communications. A digital temperature is a sensor, which provides 9-bit temperature readings. Digital temperature sensors offer excellent precise accuracy, these are designed to read from 0°C to 70°C and it is possible to achieve $\pm 0.5^{\circ}\text{C}$ accuracy. These sensors completely aligned with digital temperature readings in degree Celsius.

CHAPTER 4

HARDWARE DISCRIPTION

4.1 HARDWARE COMPONENTS

4.1.1 Heartbeat sensor



Fig 4.1 Heartbeat sensor

These devices use electrical detection to track your heart rate. They detect electrical activity through a band that wraps around your chest. For most of these devices to work as designed, the band must be wet, or you need to use a conductive gel where the sensors touch your skin. A heartbeat sensor is a device that detects and measures the heartbeat rate of a person. It is commonly used in medical settings to monitor the vital signs of patients, especially those who are critically ill. The sensor works by detecting the electrical impulses that are generated by the heart's contractions and then converting them into a readable signal.

There are different types of heartbeat sensors, including contact and non-contact sensors. Contact sensors are usually attached to the skin using adhesive pads, and they use electrodes to detect the electrical signals produced by the heart. Non-contact

sensors, on the other hand, use light or sound to detect the heartbeat rate without touching the skin.

Heartbeat sensors are also used in wearable technology, such as fitness trackers, to monitor the heart rate during physical activity. These devices usually use optical sensors that shine a light onto the skin and then measure the amount of light that is reflected back. The amount of reflected light changes as the blood flow in the blood vessels changes, which correlates with the heartbeat rate.

Overall, heartbeat sensors are a useful tool in monitoring and tracking heart health, as they provide important information about the heart's performance and can alert medical professionals to potential problems.

4.1.2 Temperature sensor



Fig 4.2 Temperature sensor

Temperature sensors work by providing readings via electrical signals. Sensors are composed of two metals that generate an electrical voltage or resistance when a

temperature change occurs by measuring the voltage across the diode terminals. When the voltage increases, the temperature also increases.

A temperature sensor is a device used to measure the temperature of an object or environment. There are many different types of temperature sensors, including thermocouples, resistance temperature detectors (RTDs), thermistors, and infrared sensors.

Thermocouples are made up of two different metals that are joined together at one end. When the temperature at the junction of the two metals changes, a voltage is produced that can be measured and used to calculate the temperature.

RTDs are made of a length of wire that has a resistance that changes with temperature. By measuring the resistance of the wire, the temperature can be calculated.

4.1.3 Micro electro-mechanical system



Fig 4.3 Micro electro-mechanical system

MEMS, or Micro Electro-Mechanical System, is a chip-based technology where sensors are composed of a suspended mass between a pair of capacitive plates. When the sensor is

tilted, a difference in electrical potential is created by this suspended mass. The created difference is then measured as a change in capacitance. They are small-scale devices that integrate electrical and mechanical components on a single chip. MEMS technology has enabled the development of a wide range of applications, from sensors and actuators to biomedical devices and micro-robots.

MEMS devices are typically fabricated using semiconductor manufacturing processes, such as lithography, etching, and deposition. They are often made of materials such as silicon, glass, or polymers. MEMS devices can be designed to be very small, with dimensions ranging from micrometers to millimeters.

Some common examples of MEMS devices include accelerometers, gyroscopes, pressure sensors, microphones, and inkjet print heads. These devices can be found in a variety of consumer electronics products, such as smartphones, tablets, and wearable devices.

4.1.4 Bluetooth sensor



Fig 4.4 Bluetooth sensor

A Bluetooth Vibration Sensor is an Accelerometer capable of recording vibration data and transfer it to a receiver using Bluetooth standard protocol. These sensors usually integrate an accelerometer and a Bluetooth transmitter to establish wireless connectivity.

A Bluetooth sensor is a device that uses Bluetooth technology to wirelessly transmit data to other devices, such as smartphones, tablets, or computers. These sensors can measure various types of data, including temperature, humidity, motion, and more. They are typically small, battery-powered, and designed to be easy to use and integrate with other devices.

Bluetooth sensors are often used in the Internet of Things (IoT) applications, where they can be used to collect data from a wide range of sensors and transmit it to other devices for analysis. They are also commonly used in wearable devices, such as fitness trackers and smart watches, to monitor various health and fitness metrics.

To use a Bluetooth sensor, you need a compatible device that can communicate with it via Bluetooth. Many Bluetooth sensors come with their own mobile apps that you can download to your smartphone or tablet to view and analyze the data they collect. Some sensors also support integration with third-party apps and platforms, making it easy to use them with other tools and services.

4.2 Hardware kit

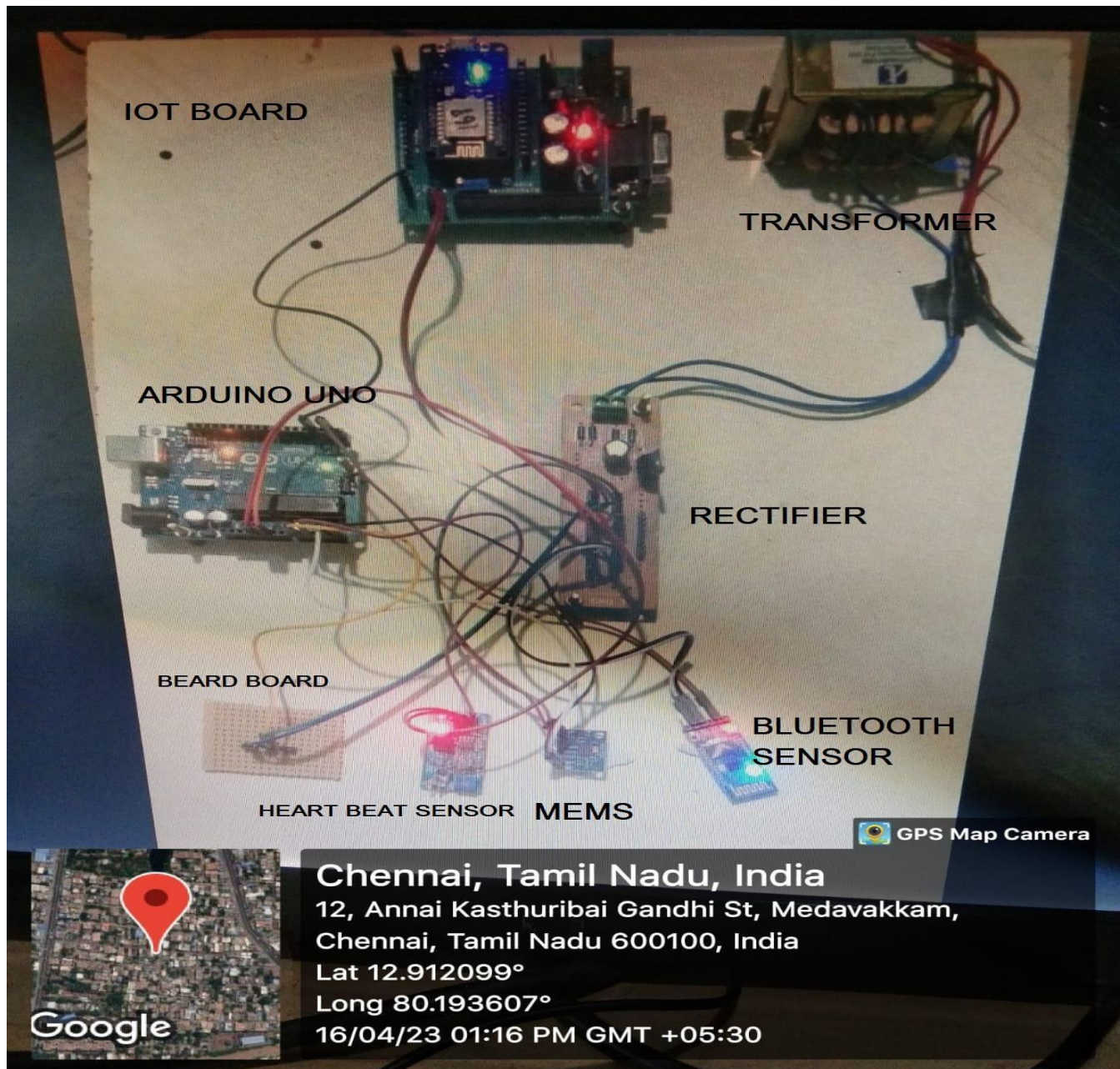


Fig 4.5 Hardware kit

An IoT-based healthcare monitoring system is designed to monitor the health of patients remotely. The system collects data from various sensors such as pulse sensors,

temperature sensors, and oxygen saturation sensors. The collected data is then transmitted to a cloud server using wireless communication protocols such as Wi-Fi or

4.2.1 Benefits

IoT-based health monitoring systems have several benefits. These systems can remotely Bluetooth. The cloud server processes the data and provides real-time health status updates to healthcare providers or family members. The system can also send alerts in case of any abnormality in the patient's health. The system is designed to improve patient care by providing continuous monitoring and early detection of health issues. Monitor patients health and provide real-time updates to healthcare providers or family members, which can improve the quality of people's lives. IoT devices and sensors can collect and analyze health conditions, lifestyle choices, and the environment to recommend preventative measures, which can reduce the occurrence of diseases and acute states. IoT-based health monitoring systems can reduce costly visits to doctors and hospital admissions and make testing more affordable. IoT-based health monitoring systems can reduce costly visits to doctors and hospital admissions and make testing more affordable. These systems can also be used where the patient and healthcare expert(s) are at different locations, allowing patients to continue their routine life while doctors monitor their health. IoT helps in data assortment, which can be used to manage and organize vast amounts of healthcare data.

There are several benefits of implementing a healthcare monitoring system using IoT (Internet of Things) technology. Some of them are:

- Remote Monitoring: IoT-based healthcare systems allow for remote monitoring of patients, which can significantly reduce the need for hospitalization and improve patient outcomes. With remote monitoring, doctors can monitor the patient's health in real-time and make necessary adjustments to the treatment plan.

- **Timely Intervention:** IoT-based healthcare systems can provide real-time data on a patient's condition, which can alert healthcare providers of any changes in their health. This can enable timely intervention, preventing adverse events and improving outcomes.
- **Cost Reduction:** By reducing the need for hospitalization, IoT-based healthcare systems can significantly reduce the cost of healthcare. Additionally, remote monitoring can help prevent complications, reducing the need for expensive interventions.
- **Improved Patient Experience:** IoT-based healthcare systems can improve the patient experience by reducing the need for hospitalization, allowing patients to recover in the comfort of their own homes. It can also improve patient engagement, as patients can take an active role in their care by tracking their own health data.
- **Data Analytics:** IoT-based healthcare systems generate a vast amount of data, which can be used for analytics and insights. These insights can help healthcare providers make data-driven decisions, improving patient outcomes and reducing costs.

Overall, IoT-based healthcare systems have the potential to revolutionize healthcare by providing remote monitoring, timely intervention, cost reduction, improved patient experience, and data analytics.

CHAPTER 5

SUMMARY AND CONCLUSION

5.1 CONCLUSION

In conclusion, the implementation of a healthcare monitoring system using IoT technology has significant potential to improve healthcare services by enabling real-time monitoring and analysis of patients' health conditions. The system can collect and analyze vital signs, track medication usage, and provide alerts to healthcare providers in case of emergencies. This can lead to better management of chronic conditions, earlier detection of health issues, and improved patient outcomes.

The healthcare monitoring system can also improve patient engagement and satisfaction by providing patients with greater control over their health and enabling them to receive personalized care. Furthermore, the system can reduce healthcare costs by reducing the need for hospitalizations and emergency department visits.

Overall, the healthcare monitoring system using IoT technology has the potential to transform the way healthcare services are provided, leading to more efficient and effective care delivery, improved patient outcomes, and reduced costs.

5.2 SCOPE FOR FUTURE WORK

1. **Data Analytics:** You can use machine learning algorithms to analyze the data collected from the healthcare monitoring system to identify patterns and trends. This can help healthcare professionals make more informed decisions and provide better care.
2. **Predictive Analytics:** By using predictive analytics, you can predict the likelihood of certain health conditions or diseases based on the data collected by the monitoring system. This can help doctors and patients take preventative measures to avoid potential health problems.

3. Remote Monitoring: You can expand the capabilities of your healthcare monitoring system to include remote monitoring of patients. This can be particularly useful for elderly or chronically ill patients who require continuous monitoring.
 4. Wearable Technology: Integrating wearable technology such as fitness trackers or smartwatches into the healthcare monitoring system can provide additional data points that can be used to improve patient care.
 5. Real-Time Alerts: You can develop an alert system that will notify healthcare professionals and/or patients when certain conditions or health parameters fall outside of acceptable ranges. This can help to ensure timely intervention and prevent serious health issues.
 6. Integration with Electronic Health Records (EHR): You can integrate your healthcare monitoring system with electronic health records to provide a more comprehensive view of patient health over time.
 7. Telemedicine: You can use your healthcare monitoring system to enable telemedicine consultations between patients and healthcare professionals. This can help to improve access to care, particularly in rural or underserved areas.
- Overall, the potential for future work in a healthcare monitoring system using IoT technology is vast, and there are many opportunities to improve patient care and outcomes.

REFERENCE PAPER:

- [1] International Organization for Standardization, Occupational health and safety ISO 45001.
- [2] X. Chen and P. Yu, "Research on hierarchical mobile wireless sensor network architecture with mobile sensor nodes," in International Conference on Biomedical Engineering and Informatics, IEEE, 2010.
- [3] P. Deshpande and M. Madankar, "Techniques improving throughput of wireless sensor network," in International Conference on Circuit, Power and Computing Technologies, IEEE, 2015.
- [4] "Network Topology and Extent," in Communication Networks: Principles and Practice, McGraw Hill Education, 2005 .
- [5] Y. Choi, Y. Jeon and S. Park, "A study on sensor nodes attestation protocol in a Wireless Sensor Network," in International Conference on Advanced Communication Technology, IEEE, 2010.
- [6] Y. Tan and K. Tseng, "Low-Voltage, DC Grid-Powered LED Lighting System with Smart Ambient Sensor Control for Energy Conservation in Green Building," in Smart Grid Infrastructure and Networking, McGraw-Hill, 2013.
- [7] L. Yan-Fang, W. Yu-Bin, S. Ying, Z. Yan-Jie, Z. Ting-Ting, W. Chang and L. Tong-Yu, "Fiber laser methane sensor and its application in coal mine safety," Procedia Engineering, vol. 26, pp. 1200-1204, 2011.
- [8] J. Dickens and R. Teleka, "Mine safety sensors: Test results in a simulated test stope," in 6th Robotics and Mechatronics Conference, 2013.
- [9] W. Bing, X. Zhengdong, Z. Yao and Y. Zhenjiang, "Study on coal mine safety management system based on "Hazard, Latent Danger and Emergency Responses"," Procedia Engineering, vol. 84, pp. 172-177, 2014.
- [10] R. Nutter and M. Aldridge, "Status of mine monitoring and communications," IEEE Transactions on Industry Applications, vol. 24, no. 5, pp. 820-826, 1988.

+ [11] Honeywell, HIH-4000 Series Humidity Sensor Datasheet, 2010.

[12] S. Pindado, J. Perez and S. Avila-Sanchez, "On Cup Anemometer Rotor Aerodynamics," *Sensors*, no. 12, pp. 6198-6217, 2012.

[13] I. Figaro USA, TGS 2611 - Methane Sensor Datasheet, 2005.

[14] Figaro, Technical Information for TGS2611, 2005.