**A Project Report on**

# REAL TIME WIRELESS EMBEDDED ELECTRONICS FOR SOILDER SECQURITY

**Submitted in partial fulfillment of the requirements for the award of degree of**

**BACHELOR OF TECHNOLOGY**

**IN**

**ELECTRONICS AND COMMUNICATION ENGINEERING**

**J.N.T. UNIVERSITY, KAKINADA**

**Submitted by**

## P.RAJESH(20F01A04I8)

**G.JAYANTH(20F01A04G5) A.REVANTH SIVAKUMAR(20F01A04E4)**

**M.MADHUSUDAN REDDY(20F01A04H6) P.SHALEM REDDY(20F01A04I5)**

Under the Esteemed Guidance of

# G. KOTESWARA RAO, M. Tech.,

Associate professor



## DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

**ST. ANN’S COLLEGE OF ENGINEERING & TECHNOLOGY, CHIRALA AUTONOMOUS**

(**Approved by AICTE New Delhi, Permanently Affiliated to JNTUK, Kakinada) (Accredited by NAAC with ‘A’ Grade, NBA & IE (I))**

# 2020-2024

## ST.ANN’S COLLEGE OF ENGINEERING & TECHNOLOGY, CHIRALA

**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**

(Approved by AICTE, New Delhi & Affiliated to JNTUK, Kakinada) (Accredited by NBA, IE(I) & NAAC with ‘A’ Grade)



**CERTIFICATE**

This is to certify that the project report entitled **“REALTIME WIRELESS EMBEDDED ELECTRONICS FOR SOLIDER SECURITY”** is submitted by P.RAJESH(20F01A04I8), G.JAYANTH(20F01A04G5), A.REVANTH SIVA KUMAR(20F01A04E4), M.MADHUSUDAN REDDY(20F01A04H6), P.SHALEM REDDY(20F01A04I5) **who carried out under my guidance and supervision, as a partial fulfillment for the award of BACHELOR OF TECHNOLOGY in ELECTRONICS AND COMMUNICATION ENGINEERING of J.N.T.U.K,** **Kakinada during the academic year 2020-2024**.

**Viva Voce Date:** .

**INTERNAL GUIDE HEAD OF THE DEPARTMENT**

**(G.KOTESWARA RAO) (Dr. D.RAJENDRA PRASAD)**

**EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

The successful completion of any task is not possible without proper suggestion, guidance and environment. Combination of these three factors act like backbone to my project titled **“REAL TIME WIRELESS EMBEDDED ELECTRONICS FOR SOILDER SECURITY”.** It is with immense pleasure that we would like to express our indebted gratitude to our guide **G. KOTESWARA RAO , M. Tech,** who guided and encouraged us in every step of the project work. His valuable moral support and guidance throughout the project helped us to a greater extent. We are thankful to his suggestions and discussions during this project.

We feel elated to thank our Head of Department of Electronics and Communication Engineering **Dr. D.RAJENDRA PRASAD, M. Tech., Ph. D,** for inspiring us all the way and arranging all the facilities and resources needed for our project.

I would like to thank our principal, **Dr. K. JAGADEESH BABU, M.Tech.,MIEEE,Ph.D,** for providing support and simulating environment. I would like to express my gratitude to the Management of **St. Ann’s College of Engineering & Technology** for providing a pleasant environment and excellent laboratory facilities.

I am thankful to all teaching and non-teaching staff of the Department of Electronics and Communication Engineering for the cooperation given for the successful completion of my project.

**PROJECT ASSOCIATES**

P.RAJESH (20F01A04I8)

G.JAYANTH (20F01A04G5)

A.REVANTH SIVA KUMAR (20F01A04E4)

M.MADHUSUDAN REDDY (20F01A04H6)

P.SHALEM REDDY (20F01A04I5)

# DECLARATION

This is to certify that the thesis titled **“REAL TIME WIRELESS EMBEDDED ELECTRONICS FOR SOLDIER SECURITY”** under the guidance of

**Mr. G. KOTEWARA RAO, M. Tech** is a bonafide work done by me, in partial fulfilment of the requirements for the award of the degree B. Tech. (Electronics and Communication Engineering) and submitted to the Department of Electronics & Communication Engineering, **St. Ann’s College of Engineering & Technology, Chirala.**

I also declare that this project is a result of my own effort and that has not been copied from anyone and I have taken only citations from the sources which are mentioned in the references.

This work was not submitted earlier at any other University or institute for the award of any degree.

**Place: SACET, Chirala. P. RAJESH (20F01A04I8)**

**Date: G. JAYANTH (20F01A04G5)**

**A. REVANTH SIVA KUMAR (20F01A0E4)**

**M. MADHUSUDAN REDDY (20F01A04H6)**

**P. SHALEM REDDY (20F01A04I5)**

|  |  |  |
| --- | --- | --- |
|  | **I**  **LIST OF FIGURES** |  |
| **FIGURE No** | **NAME OF THE FIGURE** | **PAGE No** |
| Fig 2.3 | Block diagram of Soldier position tracking | 6 |
| Fig 2.4 | Block diagram of Sensor Networks for Soldier Health Monitoring | 7 |
| Fig 3.1 | Block diagram of existing system | 9 |
| Fig 3.2 | Arduino Nano | 10 |
| Fig 4.1 | Block diagram of proposed system | 12 |
| Fig 4.2 | Real time wireless embedded electronics for soldier security circuit diagram | 13 |
| Fig 5.1 | Raspberry pi Zero 2 W | 15 |
| Fig 5.1.1 | Raspberry pi with advance features | 17 |
| Fig 5.2 | GPS Modem | 18 |
| Fig 5.2.1 | GPS Interfacing with Raspberry pi | 19 |
| Fig 5.3.1 | GSM Modem | 21 |
| Fig 5.3.1 | GSM Interfacing with Raspberry Pi | 21 |

|  |  |  |
| --- | --- | --- |
| Fig 5.3.2 | DDS18B20 Temperature sensor | 22 |
| Fig 6.1 | DDS18B20 Temperature sensor Interfacing with Raspberry Pi | 23 |
| Fig 6.1.2 | Heart Beat sensor | 24 |
| Fig 6.2 | Heart Beat sensor Interfacing with Raspberry Pi | 26 |
| Fig 6.2.2 | DHT11 Humidity sensor | 27 |
| Fig 6.3 | DH11 sensor Interfacing with Raspberry Pi | 28 |
| Fig 6.3.2 | MQ2 sensor | 29 |
| Fig 6.4 | MQ2 sensor Interfacing with Raspberry Pi | 30 |
| Fig 6.4.2 | MAX30102 Blood oxygen sensor | 31 |
| Fig 6.5 | MAX30102 Blood oxygen sensor Interfacing with Raspberry Pi | 32 |
| Fig 6.5.2 | Breath sensor | 34 |
| Fig 6.6.2  Fig 7.1.3  Fig 7.2  Fig 7.2.1 | Breath sensor Interfacing with Raspberry Pi  Installation of Raspberry Pi  Setup the Raspberry Pi  Plugin the Raspberry Pi | 36  39  39  40 |
| Fig 8 | Output of Soldier security system on Raspberry Pi | 36 |

**II**

# LIST OF ABBREVATIONS

|  |  |  |
| --- | --- | --- |
| IOT | :: | Internet of things |
| GPS | :: | Global Positioning System |
| GSM | :: | Global System for Mobile Communication |
| WSN | :: | Wireless Sensor Network |
| LCD | :: | Liquid Crystal Display |

**III**

**ABSTRACT**

One of the important and vital roles in a country’s defense is played by the army soldiers. Every year Soldiers get strayed or injured and it is time consuming to do search and rescue operations. In this paper, we present a WSN-based environmental and health monitoring approach in which sensor data is processed using robust and stable algorithm implemented in controller. These processed data are then sent to the base station via low-cost, low-power and secure communication links provided by a LoRa network infrastructure instead of cellular networks, since, they are either absent or doesn’t allow data transmission in warzone or remote areas.

We focus on monitoring environmental factors such as temperature, humidity, air pressure, air quality; physical factors such as motion, position, geographic location and health parameters like ECG (electro cardiograph), blood oxygen level, body temperature. Moreover, camera and microphone are used to monitor any undesirable situation of soldier. The aim of the system is to reduce the response time for any emergencies with the use of embedded system and WBASN, while being power efficient.

Key words: WSN, NMEA, LoRa, Raspberry Pi, WBASN

# TABLE OF CONTENTS Page

|  |  |
| --- | --- |
| **LIST OF FIGURES**  **LIST OF ABBREVATIONS ABSTRACT** | **I II**  **III** |
| **CHAPTER-1 INTRODUCTION** | **1** |
| 1.1 [Introduction to Wireless Communication](#_bookmark0) | [1](#_bookmark0) |
| 1.2 [Advantages of Wireless Communication](#_bookmark1) | [1](#_bookmark1) |
| 1.3 Applications of Wireless Communication | 2 |
| 1.4 Introduction to Project | 2-3 |
| **CHAPTER-2 LITERATURE SURVEY** | **4** |
| 2.1 Introduction to Literature Survey | 4-5 |
| 2.2 Literature Review | 5-6 |
| 2.3 Soldier Position Tracking | 6 |
| 2.4 Sensor Networks for Soldier Health Monitoring | 7 |
| **CHAPTER-3 SOLDIER SECURITY USING ARDUINO** | **8** |
| 3.1 Introduction to Existing Methodology | 8-9 |
| 3.2 Arduino Nano | 9-10 |
| 3.3 Disadvantages of Arduino Nano | 10 |
| **CHAPTER-4 SOLDIER SECURITY USING RASPBERRY PI** | **11** |
| 4.1 Introduction to Proposed Methodology | 11-12 |
| 4.2 Circuit Diagram | 13 |
| 4.3 Working Principle | 14 |

CHAPTER-5 HARDWARE DESCRIPTION 15

* 1. Raspberry Pi 15
     1. [Features 16-17](#_TOC_250010)
  2. [GPS Modem 17-18](#_TOC_250009)
     1. [GPS Interfacing with Raspberry Pi 18-19](#_TOC_250008)
  3. [GSM Modem 20](#_TOC_250007)
     1. GSM Description 20
     2. GSM Interfacing with Raspberry Pi 21

CHAPTER-6 SENSORS 22

* 1. [DS18B20 Temperature Sensor 22](#_TOC_250006)
     1. [Working Principle 22-23](#_TOC_250005)
     2. Temperature Sensor Interfacing with Raspberry Pi 23
  2. Heart Beat Sensor 24
     1. Working Principle 25
     2. Heart Beat Sensor Interfacing with Raspberry Pi 25-26
  3. DHT11 Humidity Sensor 26-27
     1. Working Principle 28
     2. DHT11 Sensor Interfacing with Raspberry Pi 29
  4. MQ2 Sensor 30
     1. Working Principle 30-31
     2. MQ2 Sensor Interfacing with Raspberry Pi 31
  5. MAX30102 Blood Oxygen Sensor 32
     1. Working Principle 33
     2. MAX30102 Blood Oxygen Sensor Interfacing with Raspberry Pi 33-34
  6. Breath Sensor 34
     1. Working Principle 35
     2. Breath Sensor Interfacing with Raspberry Pi 35-36

Chapter-7 SOFTWARE DESCRIPTION 37

* 1. Raspberry Pi OS 37
     1. Operation System 37
     2. Raspberry Pi Operation System 37-38

|  |  |
| --- | --- |
| 7.1.3 How to Install Operation System to Raspberry Pi | 38 |
| 7.2 Setup your Raspberry Pi | 39 |
| 7.3 Code | 40-42 |
| 7.3.1 Code Execution | 42 |
| 7.4 Raspberry Pi IDE | 43 |
| **CHAPTER-8 RESULT AND ANALYSIS** | **45-45** |
| **CHAPTER-9 CONCLUSION AND FUTURE SCOPE** | **45-46** |
| 9.1 Conclusion | 46 |
| 9.2 Future Scope | 47-48 |
| **CHAPTER-10 REFERENCES** | **49-50** |

CHAPTER 1

# INTRODUCTION

## INTRODUCTION TO WIRELESS COMMUNICATION

Wireless Communication is a method of transmitting information from one point to other, without using any connection like wires, cables or any physical medium. Wireless Communication is the fastest growing and most vibrant technological areas in the communication field. Generally, in a communication system information is transmitted from transmitter to receiver that are placed over a limited distance. With the help of Wireless Communication, the transmitter and receiver can be placed anywhere between few meters (like a T.V. Remote Control) to few thousand kilometers (Satellite Communication).

Wireless Communication does not require any physical medium to transfer the signals, the signals are propagated through the free space. The transmission and reception of signals is accomplished with the help of antennas. Antennas are electrical devices that transforms the electrical signals to radio signals in the form of Electromagnetic (EM) Waves and vice versa. Hence both the transmitter and receiver consists of an antenna. Wireless Communication consists of three elements: the transmitter, the channel and the Receiver. The transmitter converts the audio signal into a radio signal and broadcasts it as a radio wave via an antenna. The channel acts as a path for this radio waves to reach its destination receiver, where the receiver receives the radio waves and decodes into our required messages.

## Advantages of wireless communication

* + - Cost
    - Mobility
    - Ease of installation
    - Reliability
    - Disaster recovery

## Applications of wireless communication

* Wireless Communication is used to make calls to the people located in different parts of the world without any hindrances.
* This type of wireless communication can be included in security systems where it provides security and also provides us with required information.
* Wi-Fi which is very much into the market, also a major application of wireless.
* Wireless power transfer is used to transfer the power from one location to another location with the help of some connected devices.
* Bluetooth which is used to communicate, transfer the data is a wireless communication device.

## Introduction to project

The security of a nation is safeguarded by its armed forces, comprising the army, navy, and air force. However, ensuring the safety of soldiers, particularly in battlefield conditions, remains a significant concern. Often, soldiers lose their lives due to a lack of connectivity, hindering the ability of army base stations to monitor their locations and health statuses effectively. Continuous monitoring of soldiers facing harsh conditions is crucial to prevent life-threatening situations.

Wireless Sensor Networks (WSNs) play a vital role in health monitoring by enabling the collection of soldiers’ health and environmental data. Major research initiatives, such as the TALOS Exoskeleton project, involve collaboration between corporations, government agencies, universities, and national laboratories to develop wearable embedded devices capable of monitoring soldiers’ physical and environmental factors.

In-depth analyses by researchers like Scataglini et al. highlight the importance of smart wearable clothing in the military, while comprehensive surveys by Islam et al. delve into the impact of IoT on e-health monitoring.

Existing IoT-based health monitoring systems face several challenges, including high communication costs, data privacy issues, and limited analysis of monitored parameters. To address these issues, we propose an IoT- based health monitoring approach that leverages.

Body Sensor Networks (BSNs) comprising biomedical and physiological sensors. Instead of costly communication methods like GSM or Wi-Fi, we utilize low-cost, low- power, long-range communication links provided by IoT networks, such as IOTWAN.

Our solution overcomes the processing limitations of previous controllers like Arduino by using Raspberry Pi to process collected data. An advanced algorithm analyzes the data to detect, predict, and prevent crisis situations, triggering timely interventions. The base module receives images and audio recordings whenever emergency situations are detected, aiding in creating informed and efficient strategies to overcome challenges.

Through this project, we aim to enhance soldier safety by providing real-time monitoring and proactive crisis management capabilities, thereby contributing to the effectiveness of national security efforts.

CHAPTER 2

# LITERATURE SURVEY

## INTRODUCTION TO LITERATURE SURVEY

The literature survey serves as a critical foundation for understanding the landscape of embedded systems and their applications in ensuring soldier safety and security. Embedded systems, characterized by their specialized functionality and integration into various devices, have garnered increasing importance across multiple industries, including defense. As these systems control a myriad of devices we encounter daily, their optimization for specific tasks by design engineers has led to advancements in size, cost reduction, and performance enhancement.

In the realm of defense services, the adoption of emerging technologies, including embedded systems, plays a pivotal role in providing safety systems to soldiers. The literature survey explores the multifaceted parameters through which defense services ensure soldier safety, from monitoring vital signs to tracking locations in real time. This exploration delves into the utilization of wireless body area sensor networks (WBASNs) for monitoring physiological parameters like heart rate and body temperature, coupled with GPS tracking for precise location determination.

Central to the literature survey is an examination of the challenges encountered in military operations, particularly concerning soldier communication and navigation in adverse conditions. The inability to communicate with control rooms or fellow soldiers poses significant obstacles in locating lost troops during combat. Thus, there arises a critical need for advanced, portable, and robust systems designed to provide safety measures to soldiers, underscoring the importance of ongoing research and development in this domain.

By synthesizing existing research findings and technological advancements, the literature survey aims to identify gaps, trends, and opportunities for innovation in embedded systems tailored for soldier security and position tracking. Furthermore, it provides a contextual understanding of the state-of-the-art methodologies, technologies, and challenges faced in this field, laying the groundwork for subsequent research endeavors aimed at enhancing soldier safety and operational effectiveness.

## Literature review

The incorporation of robotic systems in military operations presents a paradigm shift in combat strategies, offering enhanced surveillance and protection capabilities in challenging environments. Military 2020 Spying Robot, as proposed by Sarmad Hameed, Muhammad Hamza Khan, and Naqi Jafri, addresses the inherent risks faced by soldiers in border regions by leveraging robotic technology for observation and reconnaissance. By deploying robots equipped with night vision cameras and RF transmitters, soldiers can remotely monitor suspicious activities without risking direct exposure to potential threats, thereby ensuring both safety and situational awareness in hostile environments.

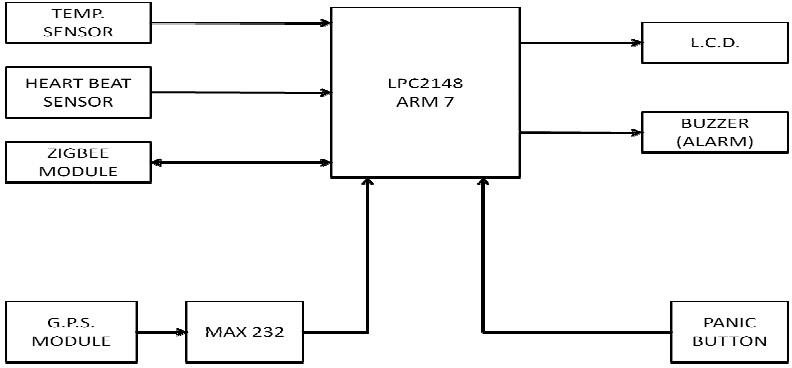
Furthermore, the integration of real-time surveillance and data collection capabilities, as demonstrated in projects such as Spying Robot With Night Vision Camera by Aaruni Jha, Apoorva Singh, and Ravinder Turna, enhances military intelligence gathering and response mechanisms. These robots, equipped with advanced sensors and communication modules, enable the seamless transmission of real-time footage and environmental data to command centers, facilitating informed decision-making and preemptive actions against potential threats. Additionally, initiatives led by researchers like Priyanka Yadav and Swati Gawhale underscore the strategic significance of robotic surveillance in gathering crucial intelligence during combat scenarios, thereby empowering military personnel to anticipate and counter enemy tactics effectively.

The navigation and obstacle avoidance capabilities of robotic systems play a crucial role in ensuring their operational effectiveness and safety. By employing techniques such as potential field methods and histogram in-motion mapping (HIMM), robotic platforms can navigate complex environments autonomously while avoiding collisions with obstacles. The integration of ultrasonic sensors and real-time obstacle avoidance algorithms, as highlighted in the research findings, enables robots to adapt dynamically to changing terrain and environmental conditions, thus enhancing their reliability and agility in mission-critical scenarios.

Moreover, the convergence of robotics and IoT technologies holds immense potential for revolutionizing military operations and enhancing strategic capabilities. By leveraging IoT-based healthcare systems and intelligent surveillance platforms, military organizations can optimize resource allocation, improve situational awareness, and mitigate operational risks. The seamless integration of robotic systems with IoT-enabled networks enables real-time data exchange and decision support, thereby enabling more effective command and control strategies and enhancing overall operational efficiency. In conclusion, the integration of robotic systems in military operations represents a transformative shift in modern warfare, offering unprecedented levels of surveillance, intelligence gathering, and tactical response capabilities. By harnessing the advancements in robotics, IoT, and artificial intelligence, military organizations can bolster their readiness, resilience, and effectiveness in addressing complex security challenges and safeguarding national interests in an ever-evolving threat landscape.

## Soldier Position tracking:

This system was proposed by P.S. Kurhe and S.S. Agrawalin. The system architecture is given by

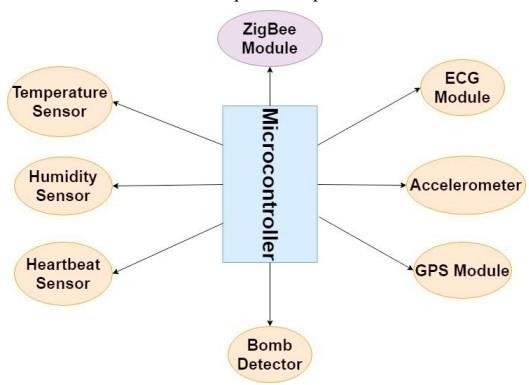


**Fig. 2.3: Block diagram of Soldier position tracking**

P.S. Kurhe and S.S. Agrawal introduce a system for tracking soldiers in real-time, enabling them to communicate their GPS coordinates with the control room during emergencies. Soldier position tracking has been crucial since World War II, facilitating navigation, positioning, target acquisition, and fleet management. The system uses body sensor networks to continuously measure and transmit soldier parameters wirelessly via GSM, allowing the control room to monitor vital signs such as heart rate and temperature.

## Sensor Networks for Soldier Health Monitoring

This system was proposed by Hock Beng Lim, Di Ma, Bang Wang, Ravishankar K Iyer, Kenneth L. Watkin. The system architecture is given by



**Fig. 2.4: Block diagram of Sensor Networks for Soldier Health Monitoring**

In the research by Hock Beng Lim, Di Ma, Bang Wang, Ravishankar K Iyer, Kenneth L. Watkin, the focus is on developing lightweight and compact sensors capable of monitoring human physiological parameters. The Body Sensor Network (BSN) comprises various biomedical and physiological sensors such as pulse rate sensor, electrocardiogram (ECG) sensor, and Electrodermal Activity (EDA) sensor. These sensors are placed on the human body to enable real-time health monitoring. The paper outlines the initial design of individual sensor nodes and presents a basic prototype of the system for data collection. Future plans include developing a fused data management system and a web portal for easy data access.

CHAPTER 3

# SOLDIER SECURITY USING ARDUINO

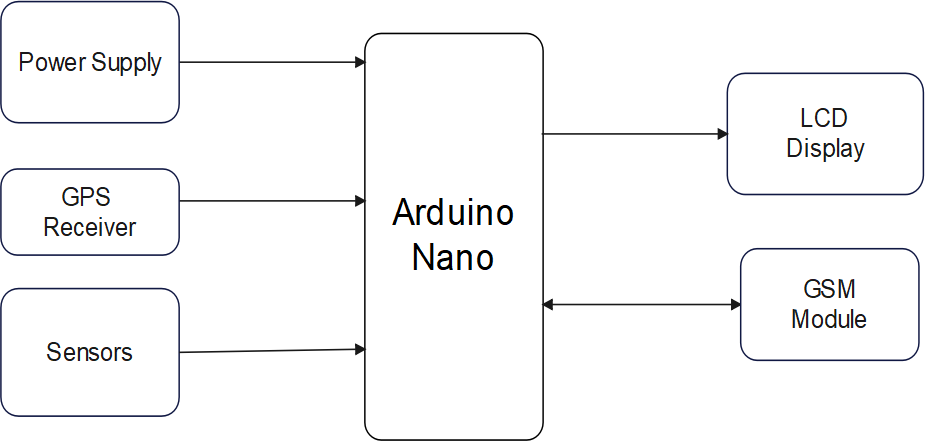
## INTRODUCTION TO EXISTING METHODOLOGY

In the current landscape of military operations, ensuring the safety and well-being of soldiers in battlegrounds is of paramount importance. The existing system, “Soldier’s Health Monitoring System using GSM,” represents a crucial step towards addressing these concerns by providing real-time monitoring of soldiers’ health statuses and ammunition levels. By mounting various instruments on soldiers, such as bio-sensor systems comprising physiological sensors, transmission modules, and processing capabilities, this system enables low-cost, wearable solutions for health monitoring. These sensors serve as vital tools in continuously assessing the health and readiness of soldiers during combat scenarios.

Moreover, the integration of GPS technology into soldier equipment plays a pivotal role in enhancing situational awareness and facilitating efficient coordination. By logging the longitude and latitude coordinates, GPS enables commanders to track soldiers’ movements in real time, thereby improving command and control capabilities on the battlefield. This technology is particularly valuable for field commanders, as it allows for precise location tracking of soldiers, even in dynamic and challenging environments.

Furthermore, advancements in soldier equipment extend beyond health monitoring to encompass communication and tactical capabilities. The incorporation of GSM and GPS modules into soldier gear enables high-speed, short-range wireless communications between soldiers, facilitating the relay of critical information related to situational awareness, tactical instructions, and covert surveillance data. This capability is essential for special operations reconnaissance missions and other tactical operations where seamless communication and coordination are imperative for mission success.

By leveraging these technologies, such as GSM, GPS, and bio-sensor systems, military organizations can implement basic yet effective lifeguarding systems for soldiers at a low cost and with high reliability. This approach not only enhances the safety and security of soldiers in the field but also empowers commanders with enhanced situational awareness and decision-making capabilities. As military operations continue to evolve, the integration of advanced technologies into soldier equipment will remain essential for maintaining operational effectiveness and ensuring the well-being of personnel in challenging environments.



**Fig. 3.1: Block diagram of existing system**

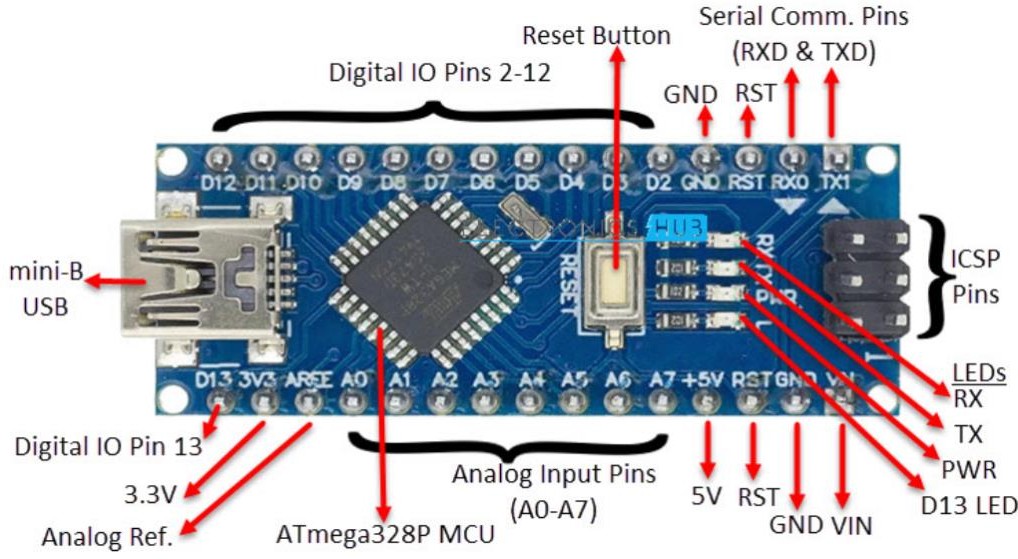
## Arduino Nano

The Arduino Nano, a compact microcontroller board based on the ATmega328P chip, stands as a versatile tool in the realm of electronics. Its diminutive size belies its capabilities, offering a plethora of features suited for a variety of projects. Despite its small footprint, the Nano boasts an array of digital and analog input/output pins, UART, SPI, and I2C interfaces, making it adaptable to a broad spectrum of applications.

Equipped with a built-in USB interface, the Nano simplifies programming and serial communication with computers, enhancing accessibility for users. Its compatibility with the Arduino IDE ensures a seamless development experience, allowing for the creation and deployment of code effortlessly. Additionally, the Nano’s onboard voltage regulator enables flexible power options, accommodating both USB and external power sources.

From robotics to wearable electronics and IoT devices, the Arduino Nano finds utility across

diverse domains. Its compact design, combined with its extensive connectivity options and ease of use, positions it as a favored choice among hobbyists, educators, and professionals alike. Whether used for prototyping, educational endeavors, or practical applications, the Nano remains an indispensable tool in the hands of electronic enthusiasts.



**Fig. 3.2: Arduino Nano**

* 1. **Disadvantages of Arduino Nano**
     + Limited Processing Power
     + Lack of Operating System
     + Limited Connectivity Options
     + Limited Memory and Storage

CHAPTER 4

# SOLDIER SECURITY USING RASPBERRY PI

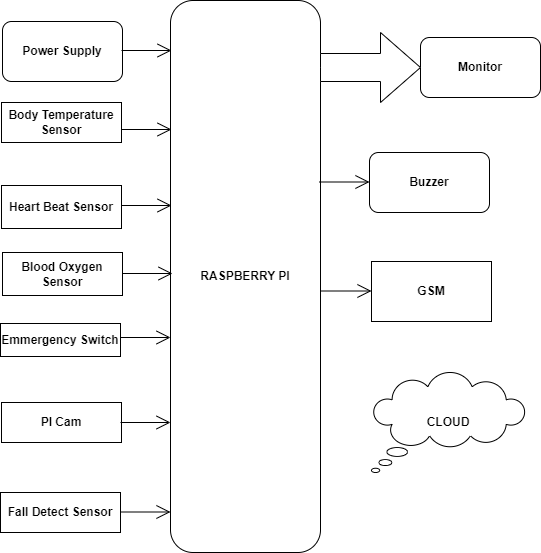
## INTRODUCTION TO PROPOSED METHODOLOGY

The proposed system is designed with the primary objective of monitoring both vital signs and environmental parameters using a network of sensors connected to a Raspberry Pi. By integrating sensors capable of measuring vital signs such as heart rate, blood oxygen levels, and body temperature, as well as environmental parameters like temperature, humidity, and air quality, the system provides a comprehensive approach to health and safety monitoring in various contexts.

In response to abnormalities detected in sensor readings, the system is programmed to initiate specific actions aimed at addressing potential risks or concerns. For instance, anomalies in vital signs or environmental conditions can trigger actions such as video capture using a Raspberry Pi camera to visually assess the situation. Additionally, location tracking via GPS enables real-time monitoring of individuals’ movements, providing crucial information for response and intervention strategies.

To ensure timely notifications and alerts, the system incorporates SMS alerts functionality facilitated by a GSM module. In the event of critical sensor readings or abnormal conditions, SMS alerts can be sent to designated recipients, enabling swift response and intervention. This feature enhances the system’s effectiveness in addressing emergent situations and mitigating potential risks to individuals’ health and safety.

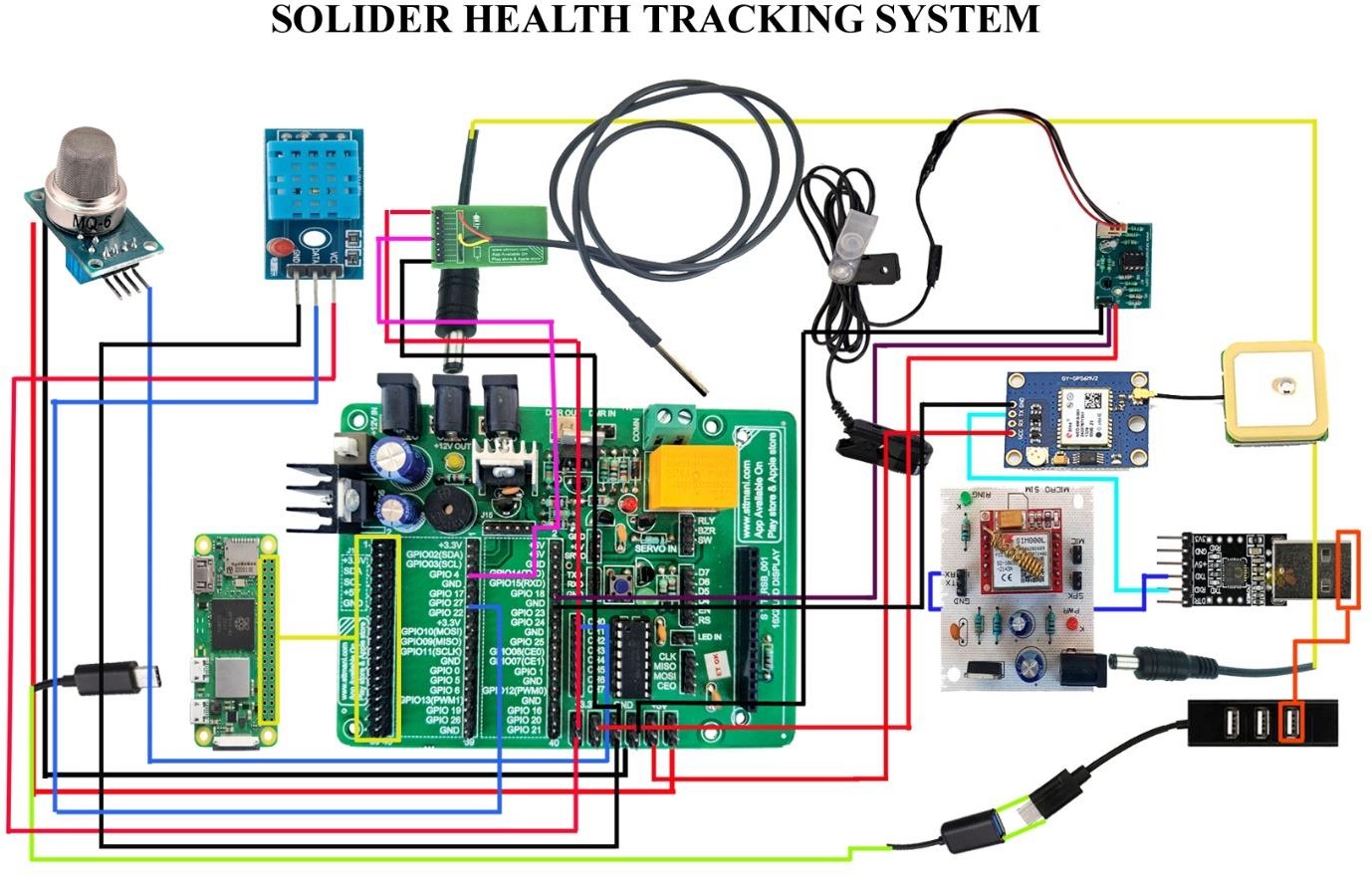
Furthermore, the system is designed to facilitate data storage and analysis both locally on the Raspberry Pi and remotely via an IoT cloud platform. By storing data locally, the system ensures continuous monitoring and logging of sensor readings, even in the absence of internet connectivity. Simultaneously, uploading data to an IoT cloud platform enables further analysis, visualization, and long-term monitoring, enhancing the system’s scalability and potential for insights generation over time.



**Fig. 4.1: Block diagram of proposed system**

* + - A Raspberry Pi, a small computer, is connected to various sensors that monitor health data like heart rate, blood oxygen, and body temperature.
    - An emergency switch can be used to signal for help.
    - The Raspberry Pi transmits this health data to a cloud service, likely for storage, analysis, and potential medical intervention.
    - A fall detection sensor can also send a signal to the cloud in case of a fall.

## Circuit diagram



**Fig. 4.2: Real time wireless embedded electronics for soldier security circuit diagram**

* + - **Upgraded system**: The proposed system, “Realtime Wireless Embedded Electronics for Soldier Security using IoT,” addresses flaws in the existing system by replacing Arduino by Raspberry pi for enhanced performance and flexibility.
    - Integration of IoT cloud: The proposed system incorporates IoT cloud for efficient storage and management of data, ensuring accessibility and reliability in real-time monitoring and analysis.
    - **Sensor Data Analysis**: Sensor data from the heartbeat sensor, SpO2 sensor, body temperature sensor, and 3-axis sensor are analyzed to detect anomalies such as irregular heartbeats, low oxygen levels, high body temperature, and falls.

## Working principle

The soldier security system operates by collecting sensor data, including vital signs and environmental parameters, through integrated sensors placed on the soldier and in their surroundings. Real-time analysis of this data enables the system to detect abnormal values, triggering predefined actions for emergency response. Additionally, GPS tracking continuously monitors the soldier’s location, facilitating rapid deployment of assistance when needed. In emergencies, a GSM module sends SMS alerts to designated contacts, while a buzzer provides immediate audible alerts to the soldier, ensuring swift intervention and enhancing overall safety and security in challenging operational environments.

## 

CHAPTER 5

# HARDWARE DESCRIPTION

**5.1 RASPBERRY PI**

Raspberry Pi is a series of small, single-board computers designed to be affordable, accessible, and versatile. These compact devices were created by the Raspberry Pi Foundation, a UK-based nonprofit organization, with the primary goal of promoting computer science education and enabling individuals to learn, experiment, and tinker with computing and electronics. Since its inception in 2012, the Raspberry Pi has gained widespread popularity, becoming a cornerstone in the world of DIY electronics, programming, and embedded systems.

The Raspberry Pi’s hardware, featuring components like CPU, RAM, USB ports, HDMI output, and GPIO pins, embodies affordability and accessibility, appealing to students, hobbyists, and professionals. Its flexibility shines through the ability to run various operating systems, notably the Linux-based Raspberry Pi OS, empowering users to delve into diverse applications, from basic tasks to intricate projects like home automation and robotics. Moreover, the GPIO pins facilitate interaction with the physical world, cementing its status as a go-to platform for electronics enthusiasts. Through democratizing technology and fostering community collaboration, Raspberry Pi continues to inspire innovation and learning across computing and electronics realms, serving as a cornerstone for creative exploration and problem-solving.



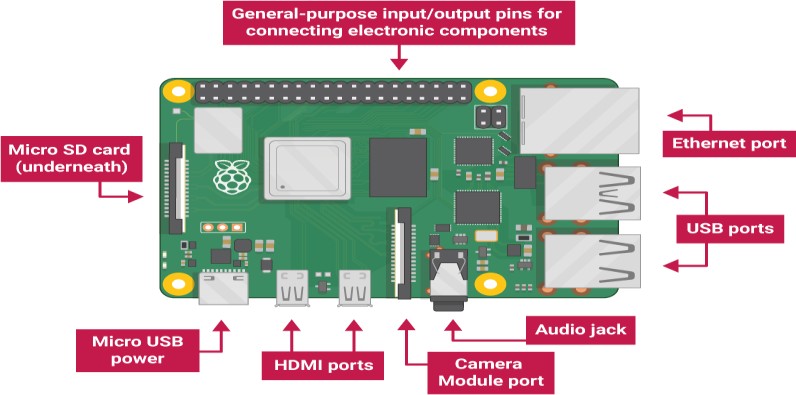
**Fig. 5.1: Raspberry pi Zero 2 W**

## Features

Raspberry Pi devices are known for their versatile features, making them suitable for a wide range of applications. Here are some of the key features of Raspberry Pi:

* + - * **Affordable:** Raspberry Pi boards are budget-friendly, making them accessible to a broad audience, including students, hobbyists, and professionals.
      * **Compact Size:** These credit card-sized computers are highly portable, allowing for easy deployment in various scenarios.
      * **Broad Hardware Options:** Raspberry Pi offers a variety of models with different hardware specifications to suit different needs, from entry-level to more powerful options.
      * **GPIO Pins:** Each Raspberry Pi has a set of GPIO pins that allow users to interact with and control external hardware, such as sensors, motors, and LEDs, making it ideal for electronics projects.
      * **Operating System Support:** Raspberry Pi supports a wide range of operating systems, with Raspberry Pi OS (formerly Raspbian) being the most popular. Users can also install various Linux distributions, Windows 10 IOT Core, and more.
      * **Connectivity:** Raspberry Pi boards typically feature USB ports, Ethernet, HDMI output, audio output, and built-in wireless options (Wi-Fi and Bluetooth), providing multiple options for connecting peripherals and devices.
      * **Software Ecosystem:** A rich software ecosystem is available for Raspberry Pi, including programming languages like Python, Java, and C/C++, as well as a vast library of pre-installed and downloadable software applications.
      * **Community and Support:** Raspberry Pi has a large and active community of enthusiasts and developers who offer support, tutorials, and resources to help users get started and troubleshoots issues.
      * **Educational Tool:** Raspberry Pi was originally designed for educational purposes, and it continues to serve as a valuable tool for learning computer science, programming, and electronics.
      * **Versatile Applications:** Raspberry Pi can be used for a wide range of projects, including web servers, media centers, home automation systems, retro gaming consoles, robotics, IOT (Internet of Things) devices, and more.
      * **Customization:** Users can modify and expand their Raspberry Pi setups through hardware add-ons called HATs (Hardware Attached on Top) and software customization, tailoring the device to their specific needs.
      * **Low Power Consumption:** Raspberry Pi devices are energy-efficient, making them suitable for applications where power consumption is a concern.

These features, along with the continuously growing community and support, contribute to the enduring popularity and versatility of Raspberry Pi for various creative and technical projects.



**Fig. 5.1.1: Raspberry pi with advance features**

## GPS Modem

Global Positioning System (GPS) technology is changing the way we work and play. You can use GPS technology when you are driving, flying, fishing, sailing, hiking, running, biking, working, or exploring. With a GPS receiver, you have an amazing amount of information at your fingertips. Here are just a few examples of how you can use GPS technology.



**Fig. 5.2: GPS Modem**

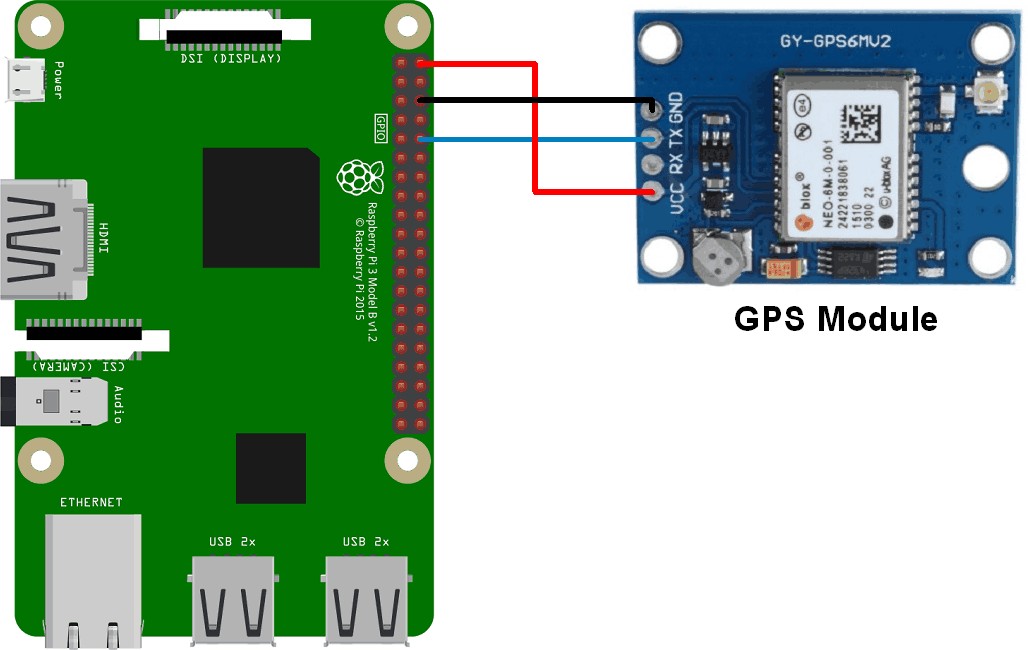
The Global Positioning System (GPS) is a satellite-based navigation system that sends and receives radio signals. A GPS receiver acquires these signals and provides you with information. Using GPS technology, you can determine location, velocity, and time, 24 hours a day, in any weather conditions anywhere in the world— for free.

GPS, formally known as the NAVSTAR (Navigation Satellite Timing and Ranging). Global Positioning System originally was developed for the military. Because of its popular navigation capabilities and because you can access GPS technology using small, inexpensive equipment, the government made the system available for civilian use. The USA owns GPS technology and the Department of Defense maintains it.

## GPS Interfacing with Raspberry Pi

GPS (Global Positioning System) interfacing with the Raspberry Pi has become increasingly popular for a wide range of applications, from navigation and location-based services to tracking and geospatial data analysis. The Raspberry Pi, with its GPIO pins and the capability to run Python scripts, is an ideal platform for connecting to GPS modules. By integrating a GPS module such as the popular NEO-6M, Raspberry Pi users can access accurate real-time geographic data, including latitude, longitude, altitude, and precise time information. This data can be harnessed for diverse projects, like creating interactive maps, developing vehicle tracking systems, or even implementing geofencing for IOT devices. The versatility of Raspberry Pi, combined with the accuracy and reliability of GPS, opens up a world of possibilities for hobbyists and professionals looking to incorporate location-based intelligence into their projects.

GPS interfacing with Raspberry Pi offers an exceptional foundation for location-based applications. The Raspberry Pi’s compatibility with a variety of GPS modules, such as the Ublox NEO-6M or Adafruit Ultimate GPS, allows developers to harness the power of global positioning technology. This integration is particularly valuable in fields like drone navigation, marine navigation, precision agriculture, and environmental monitoring. By accessing the NMEA data stream from the GPS module through the Raspberry Pi’s GPIO pins, users can not only determine their precise location but also access critical information like speed, heading, and the number of visible satellites. The combination of Python programming and the extensive community support for Raspberry Pi makes it relatively easy to create custom applications that use this GPS data, from creating GPS-guided robots to setting up real-time location-based weather stations. Moreover, the Raspberry Pi’s compact form factor, low power consumption, and affordability make it an attractive choice for embedded GPS solutions that require continuous, autonomous operation.



## 

**Fig. 5.2.1: GPS Interfacing with Raspberry pi**

* 1. **GSM Modem**

Global system for mobile communications (GSM)is the most popular standard for mobile phones in the world. GSM differs from its predecessors in that both signaling and speech channels are digital call quality, and thus is considered a second generation (2G) mobile phone system implementing GSM. GSM also pioneered a low-cost alternative to voice calls, the short Message service (SMS, also called “text messaging”), which is now supported on other mobile standards as well. One of the key features of GSM is the Subscriber Identity Module (SIM), commonly known as a SIM card.

The SIM is a detachable smart card containing the user’s subscription information and phonebook. This allows the user to retain his or her information after switching handsets alternatively the user can also change operators while retaining the handset simply by changing the SIM. Some operators will block this by allowing the phone to use only a single SIM, or only a SIM issued by them. This practice is known as SIM locking, and is illegal in some countries.

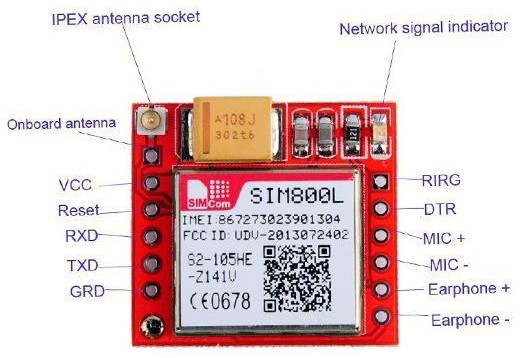
## GSM Description

GSM (Global System for Mobile communication) is a digital mobile telephony system that is widely used in Europe and other parts of the world. GSM uses a variation of time division multiple access (TDMA) and is the most widely used of the three digital wireless telephony technologies (TDMA, GSM, and CDMA).

GSM is a mobile communication modem it is stands for global system for mobile communication (GSM). The idea of GSM was developed at Bell Laboratories in 1970 .it is widely used mobile communication system in the world. GSM is an open and digital cellular technology used for transmitting mobile voice and data service operates at the 850MHZ, 900MHZ, 1800MHZ and 1900MHZ frequency bands.

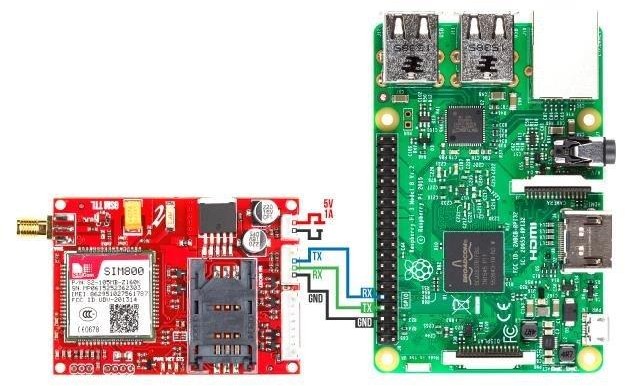
Rhydo LABZ’s GSM/GPRS Modem-RS232 is built with Quad Band GSM/GPRS engine–SIM800, works on frequencies 850/900/1800/1900 MHZ. The Modem is coming with RS232interface, which allows you connect PC as well as microcontroller with RS232 Chip (MAX232). The band rate is configurable from 9600-115200 (default baud rate is 9600) through AT command.

The GSM/GPRS Modem is having internal TCP/IP stack to enable to connect with internet via GPRS. It is suitable for SMS, Voice as well as DATA transfer application in M2M interface.



**Fig. 5.3.1: GSM Modem**

## 5.3.2 GSM Interfacing with Raspberry Pi

Interfacing a GSM module with a Raspberry Pi enables communication via SMS, voice calls, and data transmission, expanding the device's capabilities for remote monitoring, alerting, and control applications. Typically, this involves connecting the GSM module to the Raspberry Pi's UART pins or USB port for serial communication. Through appropriate software libraries and configurations, the Raspberry Pi can send AT commands to the GSM module, instructing it to perform various tasks such as sending SMS messages, making calls, and accessing network data. Additionally, the Raspberry Pi's GPIO pins can be utilized to trigger GSM module functionalities based on predefined events or conditions, enhancing automation and responsiveness in deployed systems. Overall, GSM interfacing with Raspberry Pi offers a versatile platform for integrating cellular communication into IoT, security, and remote monitoring projects, enabling connectivity and control even in areas without

|  |
| --- |
|  |
|  |  |

**Fig. 5.3.2: GSM Interfacing with Raspberry pi**

CHAPTER 6

# SENSORS

## DS18B20 Temperature sensor

The DS18B20 sensor, developed by Maxim Integrated, is prized for its accuracy and simplicity. Its single data line operation simplifies integration into microcontroller projects, supporting multiple sensors for concurrent temperature readings. Embraced by electronics and maker communities, its compatibility with Arduino and Raspberry Pi enhances its appeal. With a compact design, low power usage, and extensive temperature range, it serves diverse applications, spanning from home thermostats to industrial automation.

The DS18B20, a "1-Wire temperature sensor," offers a temperature range of -55°C to +125°C with typical accuracy of ±0.5°C. It’s TO-92 package enables easy embedding, while its low power consumption and parasitic power mode suit battery-powered projects. Widely used in weather stations, industrial control, and home automation, it provides precise temperature monitoring with straightforward interfacing and ample code resources. Affordable, reliable, and versatile, the DS18B20 is a cornerstone in electronics and IoT temperature sensing.



## 

**Fig. 6.1: DDS18B20 Temperature sensor**

* + 1. **Working principle**

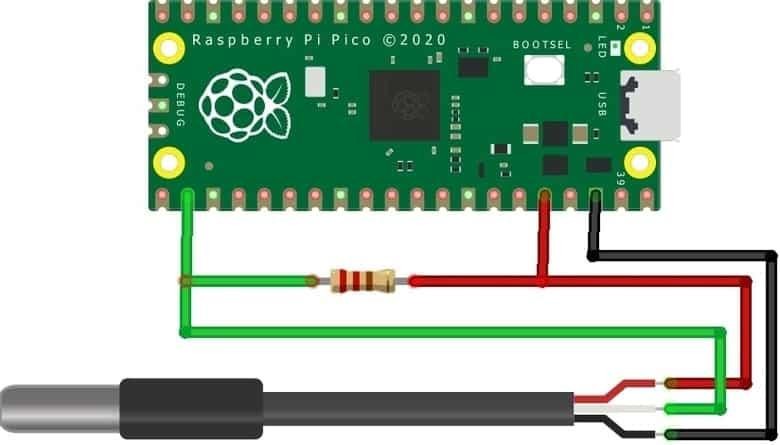
The DS18B20 sensor employs digital temperature sensing principles and the 1-Wire communication protocol for multiple sensor connectivity on a single microcontroller pin. Its internal temperature-sensitive element converts resistance changes to digital signals, allowing for precise temperature readings

. The sensor's 12-bit

resolution mode enhances accuracy, while onboard EEPROM stores unique serial codes and calibration data. Notably, it supports simultaneous monitoring of multiple temperatures and can generate alarms for critical thresholds. Widely used for its simplicity and versatility, it's a popular choice in temperature-sensing applications**.**

## 6.1.2 Temperature sensor Interfacing with Raspberry pi

Interfacing a DS18B20 temperature sensor with a Raspberry Pi enables precise temperature measurements using its digital one-wire interface, enhancing projects with Python libraries for data collection and conversion. This capability proves invaluable for applications like smart home automation and IoT environmental monitoring, leveraging the DS18B20's precision and compatibility with the Raspberry Pi. Integrating the DS18B20 with the Raspberry Pi offers developers and enthusiasts a fundamental skill set for incorporating accurate temperature monitoring and control into various projects, from climate control systems to environmental data logging.

The DS18B20's digital precision and one-wire communication protocol make it an ideal choice for Raspberry Pi projects, facilitating efficient temperature data collection and conversion with Python libraries. This integration opens doors to diverse applications, including climate control and IoT environmental monitoring, showcasing the DS18B20's versatility and wide temperature range. Mastering DS18B20 sensor interfacing with the Raspberry Pi equips individuals with a foundational skill set for enhancing their projects' functionality and value through accurate temperature sensing and control capabilities.

**Fig. 6.1.2: DDS18B20 Temperature sensor Interfacing with Raspberry Pi**

## Heart Beat sensor

A Heartbeat Sensor is an electronic device designed to measure heart rate, indicating the speed at which the heart beats. Alongside body temperature and blood pressure monitoring, tracking heart rate is fundamental to maintaining overall health. Traditionally, body temperature is measured using thermometers, while arterial or blood pressure is monitored using a sphygmomanometer. Heart rate, however, can be monitored manually by checking the pulse at the wrists or neck, or through the use of a Heartbeat Sensor.

This project entails the creation of a Heart Rate Monitor System utilizing Arduino and a Heartbeat Sensor. The system operates by detecting the pulse and translating it into heart rate readings. By understanding the principle and workings of the Heartbeat Sensor, alongside the integration of Arduino, individuals can develop a practical and effective method for monitoring heart rate in real-time. This project serves as an accessible and valuable resource for those seeking to track and manage their heart health with the aid of technology.



**Fig. 6.2: Heart Beat sensor**

## Working principle

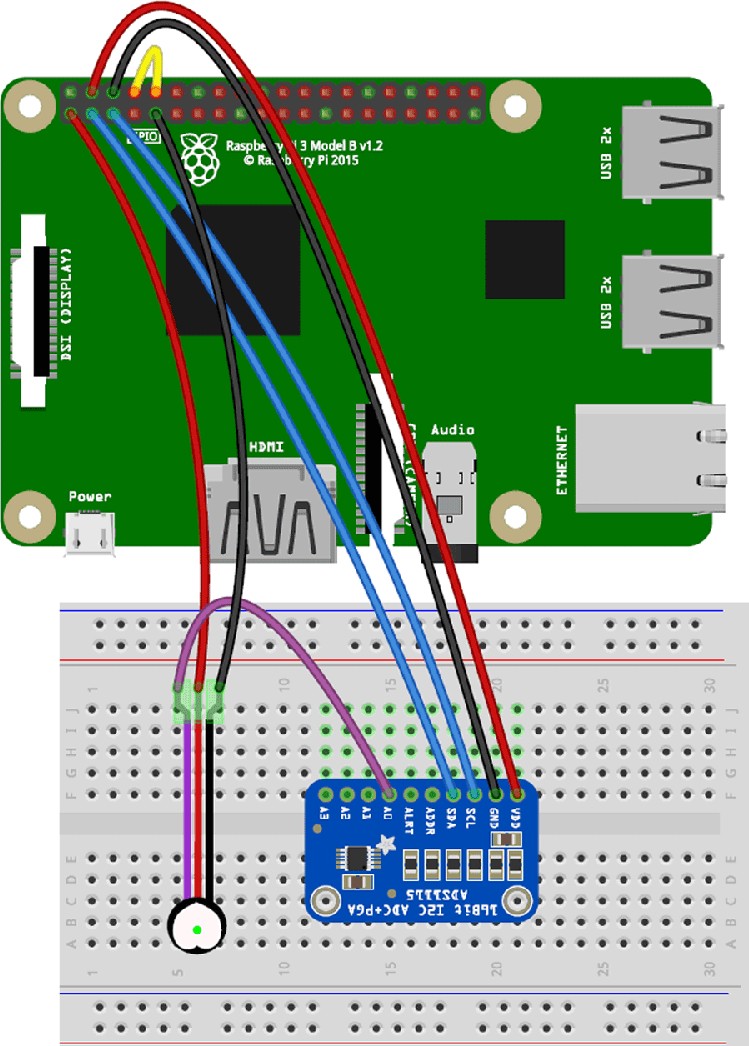
The heartbeat sensor operates based on the principle of photoplethysmography (PPG), which involves measuring the variations in light absorption caused by blood flow through the blood vessels. Typically, the sensor consists of an infrared (IR) light-emitting diode (LED) and a photodetector placed on opposite sides of a body part, such as a fingertip or earlobe. The IR LED emits light into the tissue, and the photodetector measures the amount of light that is transmitted or reflected back. As blood pulsates through the blood vessels with each heartbeat, the volume of blood in the tissue changes, altering the amount of light absorbed or reflected. This variation in light intensity is then converted into an electrical signal, from which the heart rate can be calculated.

The sensor's ability to detect subtle changes in light absorption allows it to accurately capture the pulsatile nature of blood flow, thereby providing real-time heart rate measurements. By interfacing the sensor with microcontroller platforms like Arduino, these electrical signals can be processed and analyzed to derive heart rate values, which can then be displayed or transmitted for monitoring purposes. Overall, the heartbeat sensor's working principle leverages the optical properties of blood to capture and interpret the body's pulsatile signals, offering a non-invasive and efficient method for heart rate monitoring.

## Heart Beat sensor Interfacing with Raspberry Pi

Interfacing a heartbeat sensor with the Raspberry Pi opens up a realm of possibilities for health monitoring, fitness tracking, and biometric applications. Heartbeat sensors, like pulse oximeters or ECG (Electrocardiogram) modules, can be connected to the Raspberry Pi's GPIO pins, allowing it to collect and process real-time data on a person's heart rate and related physiological parameters. This integration enables the development of innovative healthcare solutions, including remote patient monitoring, fitness wearables, and biofeedback systems. With the Raspberry Pi's computational power and Python programming capabilities, developers can process and analyse the data received from the sensor to detect irregularities, monitor trends, and generate alerts when necessary. This can be instrumental in providing timely healthcare interventions or enabling individuals to keep track of their well-being. Whether used in medical research, sports science, or personal wellness projects, the combination of a heartbeat sensor and the Raspberry Pi enhances the capacity to understand and respond to vital health information in an accessible and customizable manner.

Heartbeat sensor interfacing with the Raspberry Pi not only offers a wealth of health-related possibilities but also empowers individuals and healthcare professionals to take advantage of real-time, accurate heart rate monitoring. The Raspberry Pi's flexibility and connectivity options, coupled with a variety of heartbeat sensor choices, enable applications that range from telemedicine and stress management to sports performance optimization. Using sensors like the MAX30102 or AD8232, users can capture both pulse rate and heart rhythm data. The Raspberry Pi, equipped with its GPIO pins and robust computing capabilities, can then process this data, displaying it on a connected screen or even transmitting it over a network for remote monitoring.



**Fig. 6.2.2: Heart Beat sensor Interfacing with Raspberry Pi**

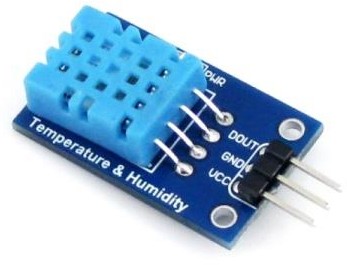
## DHT11 Humidity sensor

The DHT11 sensor is a compact and cost-effective digital temperature and humidity sensor widely used in various applications, particularly in the field of electronics and IoT. Developed by Aosong Electronics, this sensor is designed to provide accurate and reliable measurements of temperature and relative humidity.

The DHT11 sensor consists of a humidity sensing component, a temperature measuring component, and an integrated analog-to-digital converter and signal conditioning circuitry. It communicates with microcontrollers through a single-wire digital interface, making it easy to integrate into projects. With its simplicity, affordability, and ability to provide basic environmental data, the DHT11 sensor is a popular choice for weather stations, home automation systems, and other projects where monitoring temperature and humidity is essential.

One notable feature of the DHT11 sensor is its simplicity of use, requiring minimal external components for operation. It operates within a specified voltage range and provides digital output, simplifying interfacing with microcontrollers such as Arduino and Raspberry Pi. However, while cost-effective and easy to use, the DHT11 sensor does have limitations, including lower accuracy compared to more advanced sensors and a relatively slow response time. Despite these limitations, its affordability and ease of use make it a popular choice for applications where precise measurements are not critical, such as environmental monitoring in home automation projects.

Additionally, the DHT11 sensor is often utilized in projects where real-time monitoring of temperature and humidity is required. Its compact size and low power consumption make it suitable for embedded systems and IoT (Internet of Things) applications. Although it may not offer the precision of higher-end sensors, its affordability and simplicity make it an accessible option for educational purposes and prototyping. Moreover, the DHT11 sensor's robustness and wide operating range make it suitable for diverse environments, from indoor climate control to outdoor weather monitoring. Despite its limitations, the DHT11 sensor continues to be a popular choice for beginners and enthusiasts looking to explore temperature and humidity sensing in their projects.



**Fig. 6.3: DHT11 Humidity sensor**

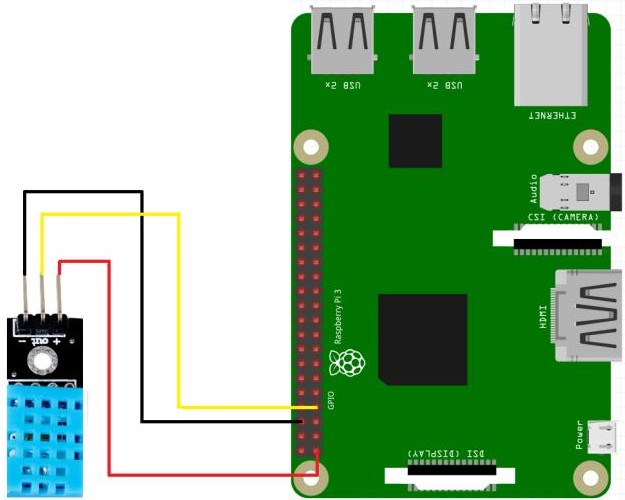
## Working principle

The DHT11 sensor operates on the principle of capacitive humidity sensing combined with a thermistor for temperature measurement. It contains a humidity sensing element and a thermistor, both of which change their electrical properties in response to changes in humidity and temperature, respectively. The humidity sensing element is made of a moisture-sensitive polymer that absorbs or releases water vapor from the surrounding air, causing its electrical capacitance to vary with humidity levels. Similarly, the thermistor's resistance changes with temperature variations.

When the DHT11 sensor is powered, it measures the relative humidity and temperature by sensing the capacitance of the humidity sensing element and the resistance of the thermistor, respectively. These electrical variations are converted into digital signals by an internal analog-to-digital converter (ADC) and then transmitted to the microcontroller through a single-wire digital communication protocol. The microcontroller interprets these signals and processes them to obtain humidity and temperature readings, which can be further utilized for various applications such as climate control systems, weather stations, and environmental monitoring devices. Overall, the DHT11 sensor's working principle relies on the capacitive and resistive properties of its sensing elements to accurately measure humidity and temperature levels in the surrounding environment.

## DHT11 sensor Interfacing with Raspberry Pi

Interfacing a DHT11 sensor with a Raspberry Pi allows for the utilization of real-time environmental data, expanding project capabilities to new heights. The DHT11 sensor, specifically designed for precise temperature and humidity measurements, seamlessly integrates with the Raspberry Pi's GPIO pins, offering a straightforward setup process. With the extensive support of Python libraries, you can effortlessly read and process the sensor's output, ensuring accurate and up-to-date temperature and humidity readings. This integration is particularly valuable for a wide array of applications, ranging from creating sophisticated climate control systems to developing IoT-powered weather stations for continuous monitoring of local weather conditions. By harnessing the computational power of the Raspberry Pi alongside the precision of the DHT11 sensor, makers, developers, and enthusiasts can embark on innovative projects that revolutionize environmental monitoring and data-driven applications. This collaborative synergy not only empowers individuals to gather essential climate information but also provides them with the tools to leverage this data for a multitude of creative and impactful applications, making it an essential skillset for those engaged in the realms of IoT, home automation, and environmental analysis.



**Fig. 6.3.2: DH11 sensor Interfacing with Raspberry Pi**

## MQ2 sensor

The MQ-2 sensor is a gas sensor that is commonly used to detect a variety of gases, including LPG, propane, hydrogen, methane, and other flammable gases. It is often employed in applications such as gas leakage detection, industrial safety, and air quality monitoring.

A gas detector is a device that detects the presence of gases in an area, often as part of a safety system. A gas detector can sound an alarm to operators in the area where the leak is occurring, giving them the opportunity to leave.



**Fig. 6.4: MQ2 sensor**

## Working principle

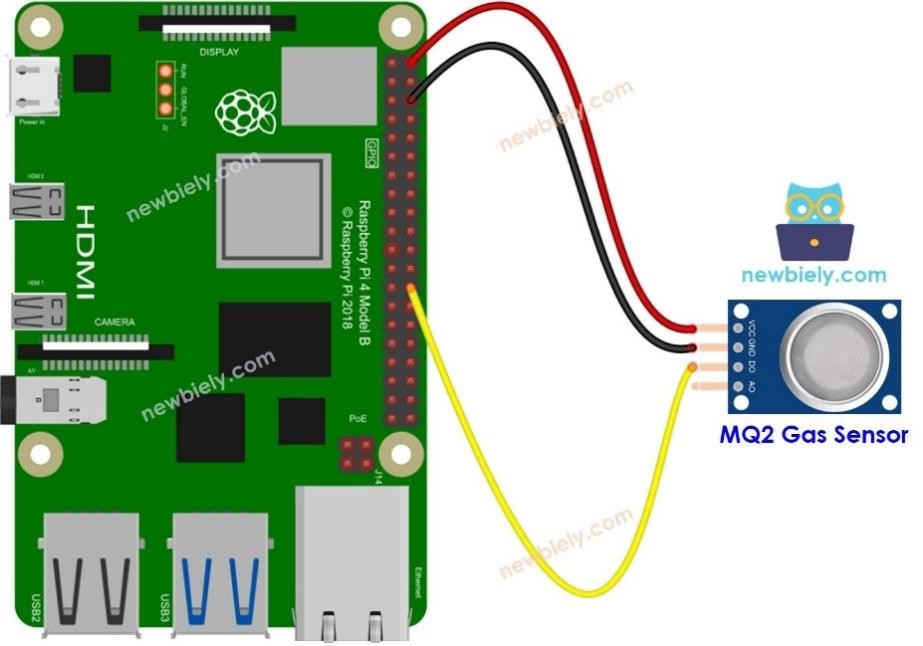
The MQ-2 sensor functions through a combination of heating and sensing elements designed to detect the presence of gases in the surrounding environment. Its heating element, typically crafted from ceramic material, maintains a constant temperature through electrical current, ensuring a stable baseline resistance for the sensor while heightening its reactivity to gases. Adjacent to this element lies the sensing element, often composed of tin dioxide (SnO2), which exhibits changes in electrical conductivity when exposed to specific gases.

Upon exposure to the ambient air, gas molecules interact with the surface of the sensing element, altering its electrical conductivity. This interaction results in a change in the resistance of the sensing element, directly proportional to the concentration of the detected gas. The sensor then translates this resistance variation into an analog voltage signal, where higher gas concentrations correlate with increased output voltage. Additionally,

calibration may be necessary to maintain precise gas detection, achieved by exposing the sensor to known gas concentrations and adjusting its sensitivity accordingly. This ensures the sensor's reliability and accuracy in providing gas concentration measurements over time.

## MQ2 sensor Interfacing with Raspberry Pi

To interface an MQ-2 gas sensor with a Raspberry Pi, you'll require the following hardware components. Firstly, you'll need a Raspberry Pi board, which serves as the core computing platform. Ensure your Raspberry Pi is powered and configured correctly. Additionally, obtain an MQ-2 gas sensor module equipped with pins for power supply and signal output. Jumper wires are essential for establishing connections between the MQ-2 sensor and the Raspberry Pi's GPIO pins. Optionally, a breadboard can simplify prototyping. Finally, ensure stable power sources for both the Raspberry Pi and the MQ-2 sensor, considering a 5V power supply for the sensor and the appropriate power source for the Raspberry Pi model you're using.

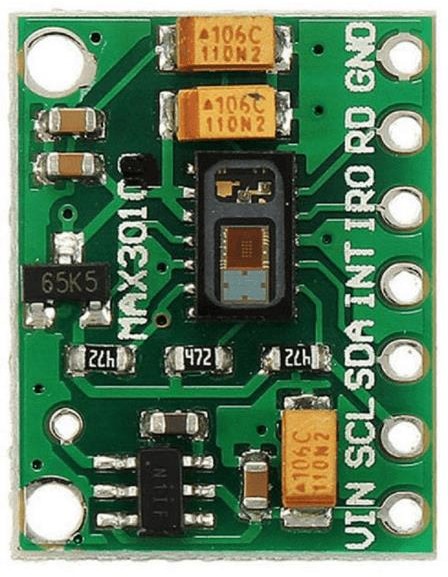
Once you have gathered the necessary components, proceed with the hardware setup. Connect the MQ-2 sensor to the Raspberry Pi using jumper wires, ensuring proper alignment of pins. Pay close attention to power connections, grounding, and signal lines. Optionally, employ a breadboard for easier prototyping. Confirm the stability of power sources to avoid issues during operation. With the hardware setup complete, you can proceed to software configuration and programming to enable communication between the MQ-2 sensor and the Raspberry Pi for gas detection and monitoring applications.

**Fig. 6.4.2: MQ2 sensor Interfacing with Raspberry Pi**

## MAX30102 Blood oxygen sensor

The MAX30102 stands as a ground breaking advancement in health monitoring technology, offering a compact and integrated solution for tracking vital physiological parameters. Its ingenious design integrates red and infrared LEDs alongside a highly sensitive photodetector, enabling precise measurement of light absorption within blood vessels. Operating on a reflective principle, the sensor emits light into the skin, where it interacts with haemoglobin and reflects back to the detector. Through sophisticated analysis of these absorption patterns, the MAX30102 delivers real-time readings of blood oxygen saturation (SpO2) levels and heart rate with exceptional accuracy.

Renowned for its versatility and reliability, the MAX30102 finds widespread application in various domains, from wearable health and fitness devices to sophisticated medical monitoring equipment. Its compact form factor, coupled with low power consumption, makes it particularly suitable for continuous health monitoring applications, ensuring seamless integration into everyday life. Whether incorporated into fitness trackers for monitoring workout intensity or deployed in medical settings for patient monitoring, the MAX30102 continues to revolutionize the landscape of health monitoring technology, empowering individuals and healthcare professionals with actionable insights into vital signs and overall well-being.



**Fig. 6.5: MAX30102 Blood oxygen sensor**

## Working Principle

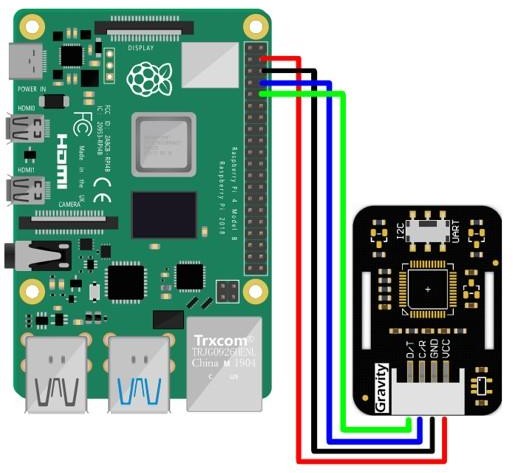
The MAX30102 blood oxygen sensor utilizes photoplethysmography (PPG) to measure variations in light absorption caused by blood volume changes in tissue. It integrates red and infrared LEDs with a photodetector to emit light into the skin and detect the amount of light transmitted or reflected back. These wavelengths are selected because they interact differently with oxygenated and deoxygenated haemoglobin, enabling the sensor to discern blood oxygen saturation (SpO2) levels and heart rate.

By analysing the amplitude and frequency of the detected signals, the MAX30102 accurately determines SpO2 levels and heart rate. Its operation revolves around analysing light absorption patterns in the skin, making it a vital tool for health monitoring applications. The sensor's compact design and reliable performance contribute to its widespread use in various settings, from wearable devices to medical equipment, empowering users with real-time insights into their physiological well-being.

## MAX30102 Blood oxygen sensor Interfacing with Raspberry Pi

Integrating the MAX30102 blood oxygen sensor with a Raspberry Pi elevates health monitoring capabilities to new heights. With the Raspberry Pi's computational prowess and connectivity options, users can capture precise data on blood oxygen saturation levels and heart rate, enabling comprehensive health tracking applications. This synergy empowers developers to implement sophisticated algorithms for real-time data processing, trend analysis, and visualization, fostering innovation in health monitoring and analysis.

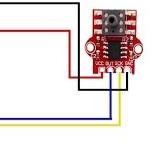
Furthermore, this integration facilitates remote monitoring and data sharing, enabling users to transmit health data to cloud-based platforms for further analysis or remote access by healthcare professionals. With network connectivity options like Wi-Fi or Ethernet, telemedicine applications benefit from remote patient monitoring, ensuring timely intervention when necessary. The seamless integration of the MAX30102 sensor and Raspberry Pi paves the way for wearable health devices that seamlessly integrate into daily life, whether as fitness trackers, smartwatches, or medical-grade wearables. The open-source nature of both platforms fosters a vibrant developer community, offering a wealth of resources and support for creating ground breaking health monitoring solutions.



**Fig. 6.5.2: MAX30102 Blood oxygen sensor Interfacing with Raspberry Pi**

## Breath sensor

Breath sensors, also called breathalyzers, play a critical role in analyzing exhaled breath. They detect specific substances like alcohol or biomarkers related to medical conditions using various technologies. In medicine, they aid in diagnosing conditions such as diabetes or detecting diseases like lung cancer. Additionally, breath sensors are vital for alcohol detection in law enforcement and personal use, offering a quick and non-invasive way to assess intoxication levels. Overall, these sensors provide efficient and versatile means to analyze breath composition for diagnostic, safety, and monitoring purposes.



**Fig. 6.6: Breath sensor**

## Working principle

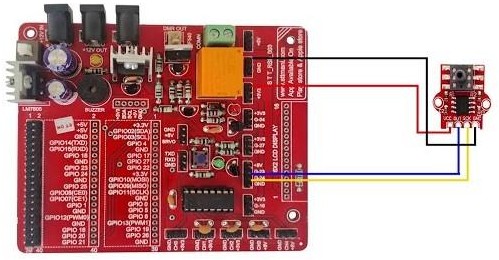
Breath sensor function based on the principle of detecting specific substances present in exhaled breath. They utilize various detection methods such as chemical reactions, optical sensors, or gas-sensing technologies to analyze breath composition. These sensors typically contain sensitive components capable of reacting with target substances, producing measurable signals indicative of their presence.

The working principle involves the interaction between the sensing element of the sensor and the target substances in the breath sample. When a person exhales into the sensor, the sensing element reacts with the target substances, causing a change in its properties or generating a measurable signal. This change in signal is then detected and quantified by the sensor, providing information about the concentration of the target substances in the breath sample. Overall, breath sensors enable non-invasive and rapid analysis of breath composition, making them invaluable tools for various applications in medicine, safety, and substance detection.

## Breath sensor Interfacing with Raspberry Pi

Interfacing a breath sensor with a Raspberry Pi involves establishing both hardware and software connections to enable communication between the sensor and the Pi. This typically entails physically connecting the sensor's output pins to the GPIO pins on the Raspberry Pi using jumper wires and ensuring proper alignment and secure connections. Additionally, software installation may be necessary to install specific Python libraries or device drivers required for the sensor to work with the Raspberry Pi. Once the hardware and software are set up, the GPIO pins on the Raspberry Pi can be configured to communicate with the breath sensor, initializing them in Python scripts and specifying whether they will be used for input or output.

Once the hardware and GPIO pins are configured, Python code can be written to read data from the breath sensor. This involves polling the sensor at regular intervals or setting up interrupt-based callbacks to capture breath data in real-time. The data collected from the sensor can then be processed as needed, which may include converting analog sensor readings to digital values, applying calibration factors, or performing data analysis to extract relevant information. Finally, the breath sensor data can be integrated into Raspberry Pi applications or projects, such as displaying sensor readings on an LCD screen, logging data to a file, sending alerts based on certain thresholds, or controlling external devices based on breath sensor inputs.



**Fig. 6.6.2: Breath sensor Interfacing with Raspberry Pi**

CHAPTER 7

# 

# SOFTWARE DESCRIPTION

* 1. **RASPBERRY PI OS**

## Operating System

An operating system (OS) is a fundamental software component that serves as the bridge between computer hardware and the applications and users who interact with it. It is the core system software that manages the computer's resources, such as the CPU, memory, storage devices, and peripherals, while also providing a user-friendly interface for human interaction. The primary functions of an OS include task management, memory allocation, file system management, and input/output control. It acts as a control center, ensuring that multiple processes can run concurrently, sharing resources efficiently and securely. The OS abstracts the hardware complexity, enabling software applications to run on a wide range of hardware configurations without needing to be tailored for each specific setup. Whether it's a personal computer, server, smartphone, or embedded device, an OS is essential for the effective operation and management of computing systems. It plays a crucial role in ensuring that users can interact with their devices, run applications, and perform tasks with ease.

## Raspberry pi Operating system

Raspberry Pi OS, formerly known as Raspbian, is an open-source operating system designed specifically for the Raspberry Pi single-board computer. The Raspberry Pi is a popular, low-cost, credit-card-sized computer that was created to promote computer science education and enable DIY electronics projects. Raspberry Pi OS is the recommended operating system for these devices, and it is optimized to provide a user-friendly computing experience on the Raspberry Pi hardware.

Raspberry Pi OS is built on the Debi an Linux distribution, and it benefits from the extensive software ecosystem and support of the Debi an community. It is available in different versions, with the most common being the "Raspberry Pi OS with Desktop" and the "Raspberry Pi OS Lite." The desktop version comes with a graphical user interface and various pre-installed applications, making it suitable for general-purpose computing tasks, while the lite version is a minimal, headless OS intended for more specialized applications.

The OS includes a wide range of software tools and programming languages, making it an excellent platform for educational purposes, hobbyist projects, and even small-scale server applications. It also provides access to the Raspberry Pi's GPIO (General-Purpose Input/Output) pins, allowing users to interact with external hardware and sensors. This capability makes Raspberry Pi OS particularly popular among tinkerers, makers, and those interested in learning about embedded systems and IOT (Internet of Things) development.

Raspberry Pi OS is continually updated and maintained by the Raspberry Pi Foundation, ensuring ongoing support and improvements for the Raspberry Pi community. It offers an accessible and versatile environment for both beginners and experienced users to explore computing, programming, and electronics on this versatile and affordable platform.

## How to Install Operating System to Raspberry pi

To use your Raspberry Pi, you’ll need an operating system. By default, Raspberry Pi check for an operating system on any SD card inserted in the SD card slot.

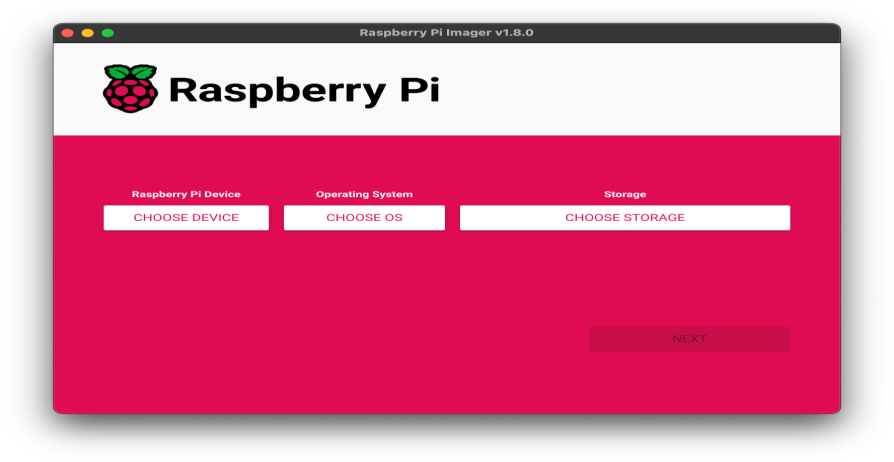
Depending on your Raspberry Pi model, you can also boot an operating system from other storage devices, including USB drives, storage connected via a HAT, and network storage.

To install an operating system on a storage device for your Raspberry Pi, you’ll need: A computer you can use to image the storage device into a boot device. A way to plug your storage device into that computer

Most Raspberry Pi users choose microSD cards as their boot device.

We recommend installing an operating system using Raspberry Pi Imager.

Raspberry Pi Imager is a tool that helps you download and write images on mac OS, Windows, and Linux. Imager includes many popular operating system images for Raspberry Pi. Imager also supports loading images downloaded directly from Raspberry Pi or third-party vendors such as Ubuntu. You can use Imager to preconfigure credentials and remote access settings for your Raspberry Pi.

 **Fig. 7.1.3: Raspberry Pi Installation**

## 7.2 Setup your Raspberry Pi

After installing an operating system image, connect your storage device to your Raspberry Pi.

First, unplug your Raspberry Pi’s power supply to ensure that the Raspberry Pi is powered down while you connect peripherals. If you installed the operating system on a micro SD card, you can plug it into your Raspberry Pi’s card slot now. If you installed the operating system on any other storage device, you can connect it to your Raspberry Pi now.



**Fig. 7.2: Setup the Raspberry Pi**

Then, plug in any other peripherals, such as your mouse, keyboard, and monitor.

Finally, connect the power supply to your Raspberry Pi. You should see the status LED light up when your Pi powers on. If your Pi is connected to a display, you should see the boot screen within minutes.



**Fig. 7.2.1:Plugin the Raspberry Pi**

**7.3 Code**

# Import necessary libraries

import time

import RPi.GPIO as GPIO

import Adafruit\_DHT

import smbus2

from gps import gps

from twilio.rest import Client

# Initialize GPIO

GPIO.setmode(GPIO.BCM)

# Initialize sensors and modules

# Initialize GSM module, GPS module, etc.

# Function to read data from temperature and humidity sensor

def read\_temperature\_humidity\_sensor():

# Read data from DHT sensor

sensor = Adafruit\_DHT.DHT22

pin = 4 # GPIO pin where the sensor is connected

humidity, temperature = Adafruit\_DHT.read\_retry(sensor, pin)

return {'temperature': temperature, 'humidity': humidity}

# Function to read data from smoke sensor

def read\_smoke\_sensor():

# Read data from smoke sensor

# Return smoke level

# Function to read data from blood oxygen sensor

def read\_blood\_oxygen\_sensor():

# Read data from blood oxygen sensor

# Return blood oxygen level

# Function to read data from heartbeat sensor

def read\_heartbeat\_sensor():

# Read data from heartbeat sensor

# Return heartbeat rate

# Function to read data from GPS module

def read\_gps\_data():

# Read GPS data

# Return latitude and longitude

# Function to read sensor data

def read\_sensors():

sensor\_data = {}

# Read data from each sensor

sensor\_data.update(read\_temperature\_humidity\_sensor())

sensor\_data['smoke'] = read\_smoke\_sensor()

sensor\_data['blood\_oxygen'] = read\_blood\_oxygen\_sensor()

sensor\_data['heartbeat'] = read\_heartbeat\_sensor()

# Add GPS data

# sensor\_data.update(read\_gps\_data())

return sensor\_data

# Function to send SMS alerts

def send\_sms\_alert(sensor, value):

# Use Twilio to send SMS alerts

# Initialize Twilio client with your credentials

client = Client("YOUR\_TWILIO\_ACCOUNT\_SID", "YOUR\_TWILIO\_AUTH\_TOKEN")

# Construct message

message = f"Alert! {sensor} reading is {value}."

# Send SMS

client.messages.create(to="RECIPIENT\_PHONE\_NUMBER", from\_="YOUR\_TWILIO\_PHONE\_NUMBER", body=message)

# Main function

def main():

while True:

# Read sensor data

sensor\_data = read\_sensors()

# Check sensor readings and compare with correct values

if sensor\_data['temperature'] > 30:

send\_sms\_alert("Temperature", sensor\_data['temperature'])

if sensor\_data['humidity'] > 70:

send\_sms\_alert("Humidity", sensor\_data['humidity'])

# Repeat for other sensors

# Wait for some time before reading sensors again

time.sleep(10) # Adjust the delay according to your requirements

# Run the main function

if \_name\_ == "\_main\_":

    main()

## 7.3.1 Code Execution

The provided Python script serves as a comprehensive framework for interfacing various sensors and modules with a Raspberry Pi to monitor environmental conditions and vital signs. The script initializes necessary libraries and GPIO settings, preparing the Raspberry Pi for sensor data acquisition. Import statements bring in libraries for sensor communication (e.g., Adafruit\_DHT for temperature and humidity sensor), GPS data retrieval, and SMS alerting via Twilio.

The script defines functions for reading sensor data, such as temperature, humidity, smoke level, blood oxygen, heartbeat rate, and GPS coordinates. Each function interacts with the corresponding sensor or module to retrieve real-time data. Additionally, there’s a function for sending SMS alerts using Twilio, which is invoked when sensor readings exceed predefined thresholds, indicating abnormal conditions.

In the main function, a continuous loop is established to periodically read sensor data, perform threshold checks, and trigger SMS alerts if necessary. This ensures ongoing monitoring of environmental parameters and vital signs, allowing for timely intervention in case of anomalies.

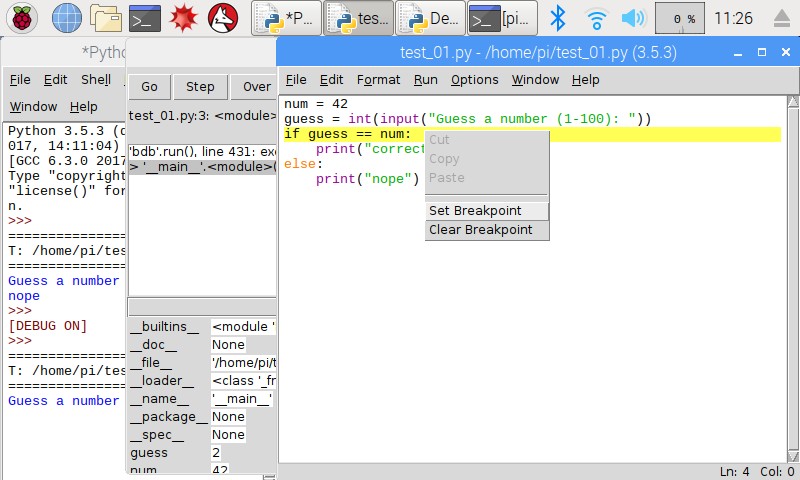
The execution of this script facilitates proactive surveillance and response mechanisms, enhancing situational awareness and safety in diverse applications, such as home automation, healthcare, and environmental monitoring. Adjustments to the delay between sensor readings can be made to suit specific monitoring requirements.

## 7.4 Raspberry pi IDE

Raspberry Pi IDE, commonly known as Thonny, is a user-friendly integrated development environment tailored for beginners and educators, specifically designed to support Python programming on the Raspberry Pi platform.

Thonny offers a simplified interface with intuitive features, making it an ideal choice for those new to programming or learning Python. Its lightweight design and straightforward layout streamline the coding process, enabling users to focus on writing and executing Python code without unnecessary complexities.

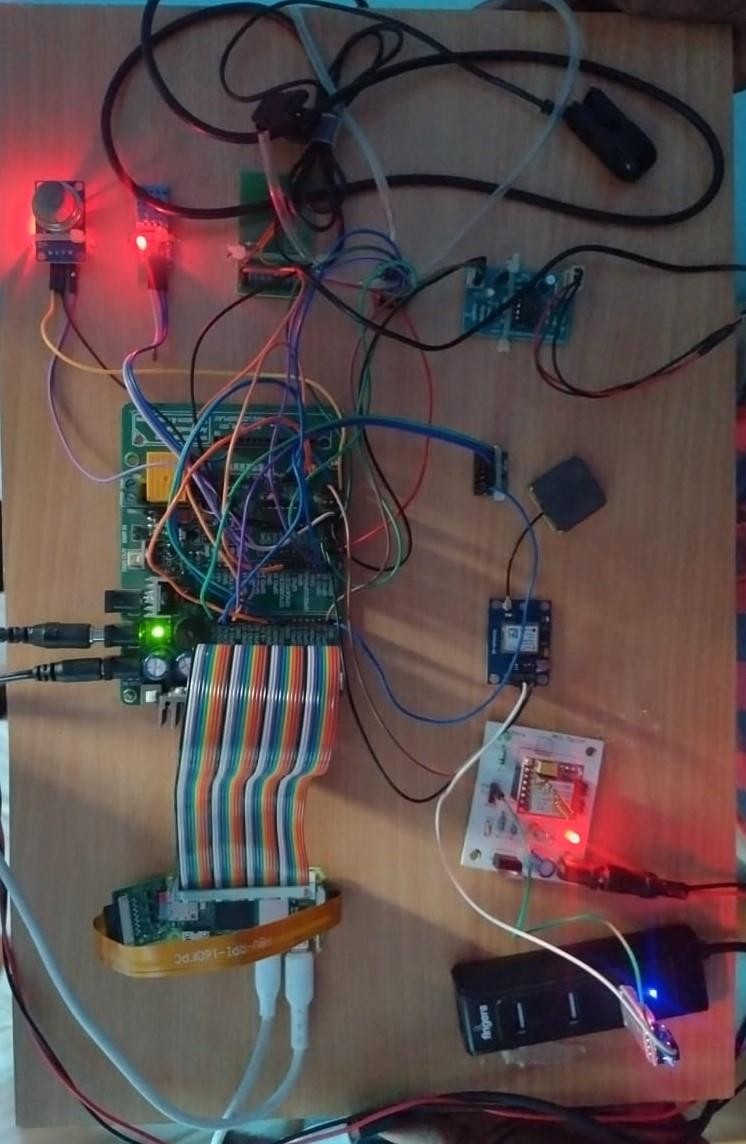
Thonny provides essential features for Python development, including syntax highlighting, code completion, and interactive debugging tools, enhancing the coding experience and facilitating learning. Its built-in debugger allows users to step through their code line by line, inspect variables, and identify errors in real-time, fostering a deeper understanding of programming concepts. Additionally, Thonny offers seamless integration with Raspberry Pi's GPIO pins, enabling users to control external hardware and explore physical computing projects directly from the IDE, making it a versatile and powerful tool for both educational and hobbyist purposes.



**Fig. 7.4: Raspberry Pi IDE Interface**

CHAPTER 8

# RESULT AND ANALYSIS



**Fig. 8: Output of Soldier security system on Raspberry Pi**

The solder security system project achieved its objectives through meticulous integration of various sensors, including heartbeat, blood oxygen, temperature, humidity, and air quality sensors, with the Raspberry Pi for comprehensive data acquisition. By harnessing the processing power of the Raspberry Pi, the project effectively implemented anomaly detection algorithms, enabling automated actions such as video capture, location tracking, and SMS alerts to be triggered promptly in response to abnormal events, thus enhancing soldier safety in challenging environments.

Furthermore, the project ensured seamless data integrity and scalability by incorporating local data storage on the Raspberry Pi while also facilitating integration with an IoT cloud platform. This combination not only enabled efficient real-time monitoring but also provided a robust foundation for future expansion and enhancement of the system. Additionally, the inclusion of a user interface allowed for user interaction and manual emergency response, enhancing the system's usability and versatility. Through comprehensive testing, validation, and documentation processes, the project demonstrated the functionality, reliability, and scalability of the solder security system on Raspberry Pi, showcasing its potential applicability in various domains such as healthcare, environmental monitoring, and security.

CHAPTER 9

# CONCLUSION AND FUTURE SCOPE

# 

# 9.1 CONCLUSION

# The proposed solder security system project utilizing Raspberry Pi showcases a comprehensive approach to monitoring vital signs, environmental parameters, and responding to abnormalities effectively. The system's functionality encompasses the integration of multiple sensors, including heartbeat, blood oxygen, body temperature, and environmental sensors, along with a Raspberry Pi camera for video capture. Abnormalities in sensor readings trigger automated actions such as video capture, location tracking, and SMS alerts, ensuring timely responses to potential threats or health issues.

# Hardware and software integration play a pivotal role in the project's success. The Raspberry Pi acts as the central processing unit, orchestrating sensor data acquisition, anomaly detection algorithms, and communication with external modules such as GSM and GPS. Python scripts facilitate seamless integration, enabling sensor data acquisition, anomaly detection, and system control, demonstrating a cohesive fusion of hardware and software components.

# Communication and alerting mechanisms are vital aspects of the system. The GSM module enables SMS alerting with location data, providing a reliable means of communication for emergency notifications to specified contacts. Additionally, GPS integration enhances the system's functionality by accurately tracking device location during abnormal events, bolstering overall responsiveness and effectiveness in emergency situations.

# Data management and cloud integration further enhance the system's capabilities. Local data storage on the Raspberry Pi ensures data integrity and accessibility for immediate response actions, while integration with an IoT cloud platform allows for centralized data storage, analysis, and real-time monitoring. This integration not only enhances system scalability but also facilitates long-term data management, contributing to the project's robustness and versatility.

# 9.2 Future Scope

# The proposed solder security system project utilizing Raspberry Pi lays a solid foundation for future advancements and expansions in several areas. Firstly, in terms of sensor technology, there is potential for integrating additional sensors to monitor a wider range of vital signs and environmental parameters. This could include sensors for detecting specific gases, pollutants, or even bio-markers indicative of certain health conditions, thereby enhancing the system's ability to provide comprehensive monitoring and early warning capabilities.

# Secondly, advancements in anomaly detection algorithms and machine learning techniques could further refine the system's ability to identify and respond to abnormalities. By continuously analyzing and learning from sensor data patterns, the system could become more adept at distinguishing between benign fluctuations and genuine threats, leading to more accurate and reliable automated responses.

# Thirdly, in terms of communication and alerting mechanisms, future developments could focus on enhancing the system's connectivity and interoperability. Integration with emerging communication technologies such as 5G networks or satellite communication systems could provide more robust and resilient communication channels, particularly in remote or challenging environments where traditional networks may be unreliable.

# Additionally, advancements in cloud computing and data analytics could unlock new possibilities for real-time monitoring, predictive analytics, and decision support. By leveraging cloud-based platforms for data storage, analysis, and visualization, the system could gain insights into long-term trends, patterns, and correlations, enabling proactive interventions and preventive measures.

# Moreover, the project's scalability and adaptability pave the way for its deployment in diverse domains beyond military applications. For instance, similar systems could be deployed in healthcare settings for remote patient monitoring, in industrial environments for safety and compliance monitoring, or in smart cities for environmental monitoring and disaster management.

# Overall, the future scope of the solder security system project is characterized by continuous innovation and expansion, driven by advancements in sensor technology, data analytics, communication networks, and interdisciplinary collaborations. As technology evolves and new challenges emerge, the project has the potential to evolve into a versatile tool for enhancing safety, security, and well-being in various contexts.

# 

CHAPTER 10

# REFERENCES

* + 1. Muhammad Hamza, M Atique-ur-Rehman, Hamza Shafqat, Subhan Bin Khalid, “CGI SCRIPT AND MJPG VIDEO STREAMER BASED SURVEILLANCE ROBOT USING RASPBERRY PI” in 16th International Bhurban Conference on Applied Sciences & Technology (IBCAST) Islamabad, Pakistan, 8th – 12th January, 2019, IEEE 2019.
    2. Juan G. Parada-Salado, Luis E. Ortega-García, Luis F. Ayala-Ramírez, Francisco J. Pérez Pinal, “A Low-Cost Land Wheeled Autonomous MiniRobot For In-Door Surveillance” In Ieee Latin America Transactions, Vol. 16, No. 5, May 2018.
    3. S M Ashish, Madhurya Manjunath, Ravindra L, Mohammed Nadeem, Neelaja K, “Automated Hybrid Surveillance Robot” In International Journal Of Innovations In Engineering Research And Technology [Ijiert], Issn: 2394-3696; Volume 5, Issue 5, May2018.
    4. T. Saravanakumar, D. Keerthana, D. Santhiya, J. Sneka, D. Sowmiya, “Surveillance Robot Using Raspberry Pi-IoT” in International Journal of Electronics, Electrical and Computational System (IJEECS);ISSN 2348-117X; Volume 7, Issue 3, March 2018.
    5. Harshitha R, Muhammad Hameem Safwat Hussain, “SURVEILLANCE ROBOT USING RASPBERRY PI AND IOT” in 2018 IEEE - DOI 10.1109/ICDI3C.2018.00018; International Conference on Design Innovations for 3Cs Compute Communicate Control.
    6. Gaurav S Bagul, Vikram C Udawant, Kalpana V Kapade and Jayesh M Zope, “IOT BASED SURVEILLANCE ROBOT” in International Journal of Trend in Research and Development, Mar - Apr 2018; Volume 5(2), ISSN: 2394-9333.
    7. Rutuja. R. Bachhav, Priyanka. S. Bhavsar, Devyani. K. Dambhare, “Spy Robot For Video Surveillance And Metal Detection” In International Journal For Research In Applied Science & Engineering Technology (Ijraset); Issn: 2321-9653; Volume 6 Issue V, May 2018.
    8. Neha Joisher, Sonaljain, Tejas Malviya, Vaishali Gaikwad (Mohite), “Surveillance System Using Robotic Car” In International Conference On Control, Automation And Systems, Ieee 2018.
    9. Fatma Salih, Mysoon S.A. Omer, “Raspberry Pi As A Video Server” In The International Conference On Computer, Control, Electrical, And Electronics Engineering (Iccceee) 2018.