DESIGN AND DEVELOPMENT OF A PROTECTION SYSTEM FOR RESCUE TEAM

A PROJECT REPORT

Submitted by

ATHULYA A S	**(721219106007)**
MITHUN KUMAR N	**(721219106027)**
SUNDAR RAJ A	**(721219106052)**
YASSHIKAA J	**(721219106062)**

In partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

KARPAGAM INSTITUTE OF TECHNOLOGY,

COIMBATORE-641105

ANNA UNIVERSITY: CHENNAI-600025

MAY 2023

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report "**DESIGN AND DEVELOPMENT OF A PROTECTION SYSTEM FOR RESCUE TEAM" is the Bonafide work of "ATHULYA A S (721219106007), MITHUN KUMAR N (721219106027), SUNDAR RAJ A (721219106052), YASSHIKAA J (721219106062)**" who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr. S. Gopinath,M.E,Ph.D,	Mr. S. Mukunthan,M.Tech,
HEAD OF THE DEPARTMENT,	**SUPERVISOR,**
Department of Electronics and	**ASSISTANT PROFESSOR**,
Communication Engineering,	Department of Electronics and
Karpagam Institute of Technology,	Communication Engineering,
Coimbatore-641105.	Karpagam Institute of Technology,
	Coimbatore-641105.

Submitted for the project viva-voce examination held on ……………………

Internal Examiner	**External Examiner**

ACKNOWLEDGEMENT

With genuine humility, we are obediently thankful to God Almighty without him, this work would have never been a reality.

We express our profound gratitude to our respected chairman **Dr.R.Vasantha Kumar**, for giving this opportunity to pursue this course. At this pleasing moment of having successfully completed the project work, we wish to acknowledge our sincere gratitude and heartfelt thanks to our respected Principal **Dr.P.Manimaran**,**Ph.D.,** our respected vice-principal **Dr.D.Banu,Ph.D.,** for having given us the adequate support and opportunity for completing this project work successfully.

We express our deep sense of gratitude and sincere thanks to our beloved Head of the Department, **Prof.Dr.S.Gopinath**, who has been a spark for enlightening our knowledge. Our profound gratitude goes to project coordinators **Dr.S.Syed Jamaesha** and **Dr.N.Suma** ,review members and all the faculty members of Department of ECE for the invaluable knowledge they imparted on us. We express our deep sense of gratitude and sincere thanks to our project guide **Mr.S.Mukunthan**. Our humble gratitude and heartiest thanks go to our family members and friends for their encouragement and support throughout the course of this project.

ATHULYA A S (721219106007)

MITHUN KUMAR N (721219106027)

SUNDAR RAJ A (721219106052)

YASSHIKAA J (721219106062)

ABSTRACT

Safety is a major problem today, especially for the rescue crew. The release of harmful chemicals, a shortage of oxygen, and natural calamities are the main threats to their health. In this project, we created a system for continuous environmental monitoring that keeps track of things like oxygen levels and harmful gases like methane, carbon dioxide, and carbon monoxide. We create intelligent clothing for the rescue crew here. A tool that can be utilized to ensure the safety of rescue personnel is the smart suit. The tool can be used to keep an eye on the safety requirements of those workers' working environments. This gadget measures vibration, humidity, hazardous gas concentration, and temperature. The control room receives the data, processes it, and chooses what to do with it based on the sent parameters. In this study, data is sent and received using **LoRa** communication technology, and processed using an Arduino. **HX710B, MQ9, DHT11,** and **MAX30102** sensors can be used to obtain the parameters. The location is also monitored using the installed GPS module. The major goal of employing LoRa communication is to ensure that the connection is robust even at greater distances than Zigbee can cover. Zigbee has a range of up to 300 meters, however LoRa's range can reach up to 5 km, depending on the local traffic. The information from the sensors is processed by the Arduino microcontroller before being sent over GSM and LoRa. The control room can act appropriately, evacuating the employees if necessary, or improving the situation. Further the data is uploaded to the cloud where it can be monitored.

CHAPTER NO	TABLE OF CONTENTS			PAGE NO

CHAPTER NO	TABLE OF CONTENTS		PAGE NO

CHAPTER NO	TABLE OF CONTENTS			PAGE NO

LIST OF FIGURES

LIST OF ABBREVIATIONS

LoRa	-	Long Range
LCD	-	Liquid Crystal Display
GPS	-	Global Positioning System
SOS	-	Save Our Souls
SSC	-	Synchronous Serial Communication
GSM	-	Global System for Mobile Communication
ESSI	-	Enhanced Synchronous Serial Interface
DSP	-	Digital Signal Processing
Wi-fi	-	Wireless Fidelity
IOT	-	Internet of Things
MCU	-	Microcontroller Unit
CSS	-	Chip Spread Spectrum
FHSS	-	Frequency Hopping Spread Spectrum
FEC	-	Forward Error Correction

CHAPTER-1

1. INTRODUCTION

In today's era where safety and security are the top most priority in various critical processes, we see various scenario's where still human involves themselves providing helping hands for the needy. While studying the latest facts it has been reported that the people who work under these rescue teams are also undergoing many hazards sometimes even death. These incidents happen due to emission of poisonous gases, sudden change in health of the person, collision, unexpected flooding etc. This issue is unsolved without any optimum solution. The suggested system can never be implemented inside disaster area as it is non predictable but if managed on a wireless sensor network and then implemented accordingly as proposed in this paper as a suit the lives which are wasted inside various accident areas can be saved at right time. Moreover, the number of efficient sensors proposed in this paper make this solution one of its kinds. There are several ways out which can be implemented on same issue and various are implemented previously as well but they have one major issue of monitoring when the accident or hazard has occurred due to any reason. In such scenario the disaster management authority starts digging the complete site which is time consuming. It happens most of the time when rescue team cannot save maximum lives just because the lives those have proper pulse rate cannot be located properly. This system will not only locate the exact depth and GPS location of miner but it will continuously be updating the pulse rate of the person. This system in this way acts a lifesaving suit for the rescue team.

1.1 EXISTING SYSTEM

The Guardian XO is a full-body exoskeleton that enhances the wearer's strength and endurance. The potential cons of the Guardian XO include its high cost, which can limit its availability to some rescue teams, as well as its weight and size, which can make it difficult to transport and maneuver in certain rescue situations.

C- Thru is a smart helmet that uses augmented reality to provide situational awareness to rescue workers. The potential cons of C-Thru include its limited field of view, which can be a drawback in some rescue situations where a wider field of view is necessary, as well as the need for a high-speed network connection to transmit data to the helmet.

Argus is a smart visor that provides real-time biometric data to firefighters. The potential cons of Argus include the visor's high cost, which can limit its availability to some fire departments, as well as the visor's bulkiness, which can be a drawback in situations where firefighters need to move quickly and with agility. Rescue unit is a smart suit that includes sensors for monitoring vital signs and providing situational awareness to rescue workers. The potential cons of Requite include the need for regular calibration and maintenance of the suit's sensors, as well as the need for a reliable network connection to transmit data to a central command center.

Overall, while existing smart suit systems for rescue teams have the potential to enhance the safety, efficiency, and effectiveness of rescue operations, they also have some potential drawbacks, such as high costs, limited field of view, bulkiness, and the need for regular calibration and maintenance.

1.2 STATEMENT OF THE PROBLEM

Problem Statement: The current standard rescue gear and equipment used by rescue teams lack advanced technological integration, limiting the safety, efficiency, and effectiveness of rescue operations in hazardous and challenging environments. There is a need for a smart suit designed specifically for rescue teams that addresses these limitations and enhances their capabilities.

During Design the following issues need to be faced:

1. Inadequate Protection: The existing rescue protection system may not provide sufficient protection against extreme temperatures, fire, chemical exposure, and physical impact, putting the rescue team members at risk of injury or harm during operations.

2. Limited Communication and Coordination: Communication between team members and with the command center is crucial for effective coordination during rescue operations. However, the current system lacks integrated communication devices, hindering real-time communication and coordination efforts.

3. Lack of Vital Sign Monitoring: Monitoring the vital signs of rescue team members is essential to ensure their well-being in high-stress and physically demanding environments. The absence of integrated sensors in the system prevents real-time monitoring of vital signs, compromising the safety and health of the rescue personnel.

4. Inefficient Navigation and Orientation: Navigating complex and unfamiliar environments poses challenges to rescue teams. The absence of integrated navigation systems and augmented reality displays makes it difficult for team members to accurately determine their location, direction, and assess the surrounding conditions, leading to increased response times and reduced situational awareness.

5. Limited Environmental Hazard Detection: Identifying and responding to potential hazards, such as toxic gases, radiation, or unstable structures, is critical for the safety of the rescue team.

6. Lack of Durability and Ergonomics: The current gear may not be durable enough to withstand rugged environments and physical exertion, leading to wear and tear or discomfort for the rescue team members during extended operations.

1.3 OBJECTIVE

A smart suit for a rescue team has the potential to enhance the safety, efficiency, and effectiveness of rescue operations. The suit can be equipped with sensors, communication devices, and other features that can provide real-time data and insights to rescue workers. Smart suits can monitor the vital signs of rescue workers, such as heart rate, blood pressure, and body temperature. This information can help identify early signs of heat exhaustion, dehydration, or other medical issues, allowing rescue workers to take necessary breaks or receive medical attention. Smart suits can incorporate communication devices that enable rescue workers to communicate with each other and with the outside world.

This can help improve coordination, share information, and request assistance as needed. Smart suits can have sensors that provide situational awareness, such as detecting gas leaks, radiation levels, or other hazardous conditions. This information can help rescue workers make informed decisions about how to proceed with their operations. Smart suits can incorporate exoskeletons or other devices that improve mobility and physical strength, allowing rescue workers to perform tasks that might otherwise be difficult or impossible. Smart suits can have features such as impact- resistant materials, built-in airbags, or other safety features that protect rescue workers from injury or harm.

Overall, smart suits for rescue teams have the potential to improve the safety, efficiency, and effectiveness of rescue operations, which can ultimately save lives and prevent injuries

CHAPTER – 2

2. LITERATURE SURVEY

Title: "A Smart Wearable device for securing the life of Coal Miners,"
Author: DigvijayGuleria1,GoliDheeraj2, Gokani Sriram3, Komal Chadha5

Description: The authors propose a smart wearable device for securing the lives of coal miners. The device includes gas sensors to detect methane and carbon monoxide, which are common hazards in coal mines, and sends an alert to the miner's supervisor if dangerous levels are detected. The device also includes a GPS system to track the location of miners in the event of an emergency and a heart rate monitor to detect signs of physical distress. Additionally, the device includes an emergency button that can be pressed by miners to signal for help. The authors highlight the importance of implementing safety measures for coal miners, as mining is a high-risk occupation with a high rate of fatalities and injuries. The smart wearable device can provide real- time data on the health and location of miners, which can help prevent accidents and save lives. The authors conclude that the device has the potential to be an effective tool for improving the safety of coal miners and should be further developed and tested for use in coal mines.

Title: "Smart wearables for fall detection"

Author: Xiaoqing chai, Renjie wu, Mathew pike, Hangchao Jin

Description: The authors highlight the challenges in developing accurate fall detection algorithms, as falls can occur in different ways and in different environments. They discuss the importance of testing and refining these algorithms to improve the accuracy of fall detection. The authors also note the potential privacy concerns associated with wearable devices that collect sensitive health data, and suggest the need for transparent and secure data collection and storage practices.

Title: “Smart monitoring using LoRa”

Author: T.Porselvi, Sai ganesh C.S. , Janaki.B, Priyadhaarshini.K, Sajitha begam

Description: The authors discuss the development of a smart monitoring system that uses LoRa technology to transmit data from various sensors to a central server. The system includes sensors for monitoring parameters such as temperature, humidity, and air quality, as well as sensors for detecting motion and vibration. The data collected by these sensors is transmitted to the server using LoRa technology, and can be analyzed in real-time to detect anomalies and trends.

Title: “Intelligent helmet system”

Author: Akshunya Mishra, Saksham Malhotra, Ruchira, Pallavi choudekhar

Description: The authors propose an intelligent helmet system that uses various sensors and technologies to improve the safety and performance of motorcycle riders. The system includes a helmet with integrated sensors for monitoring parameters such as heart rate, temperature, and humidity, as well as sensors for detecting the rider's head position and speed.

2.1 PROPOSED SYSTEM

The proposed smart suit for rescue teams aims to improve the safety and performance of rescue workers in hazardous environments. The system includes various sensors for monitoring parameters such as gas levels, temperature, humidity, and heart rate. These sensors are connected to a microcontroller, specifically an Atmega 328p, which processes the data and transmits it wirelessly using LoRa technology.

The transmitter side of the system includes a LoRa module, the sensors such as HX710B, MQ9, DHT11, MAX30102. A SOS system is also present that could possibly transmit message to the control monitor.

The LoRa module allows for long- range communication, making it ideal for use in environments where traditional communication methods may be unreliable.

The receiver side of the system includes another LoRa module and a Node MCU, which receives and processes the data transmitted by the transmitter side. It also contains an LCD display for viewing the data. The proposed smart suit system has numerous potential applications, such as in search and rescue operations, firefighting, and hazardous waste disposal.

The system could provide real-time monitoring of environmental conditions and the health of rescue workers, enabling them to take appropriate safety measures and respond to emergencies more effectively. The importance of addressing issues such as power consumption, sensor calibration, and data security in the development of such a system is also considered. They also suggest that future research and development could focus on integrating additional sensors and technologies to further improve the functionality of the smart suit system.

CHAPTER – 3

3. BLOCK DIAGRAM

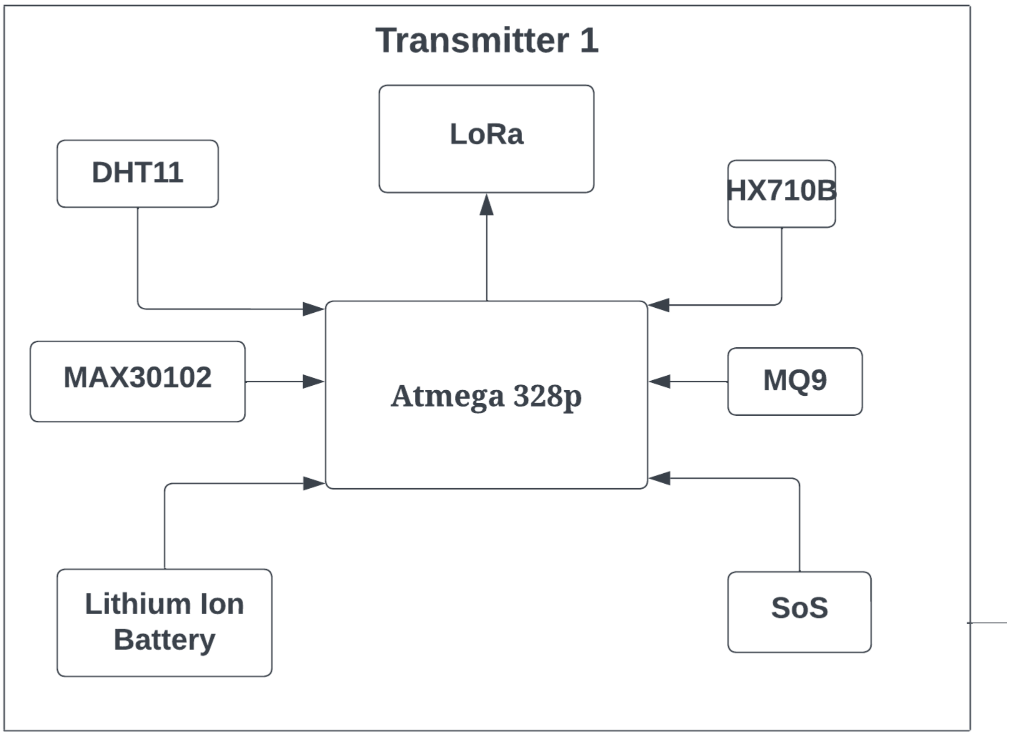

Figure 3.1: Transmitter Block Diagram

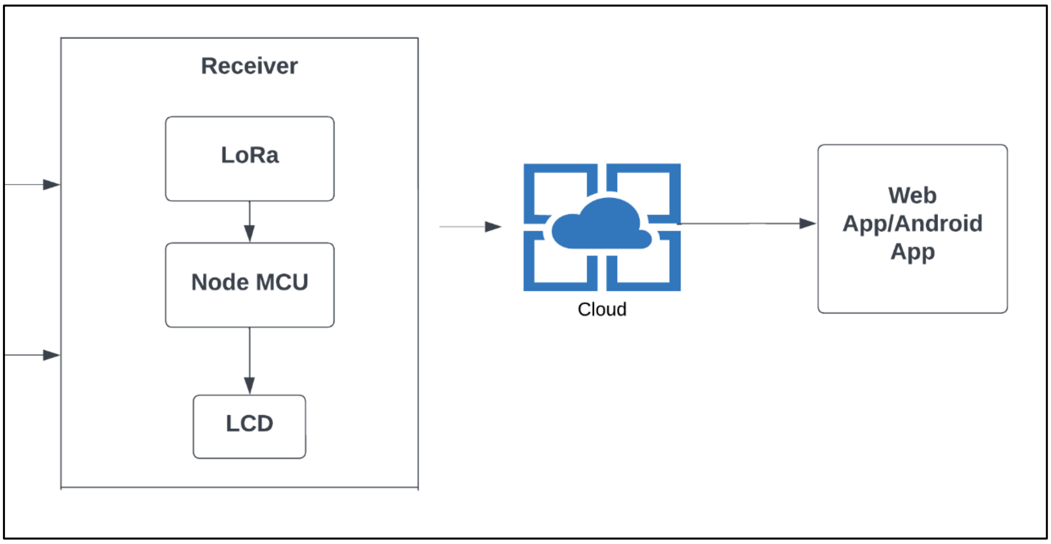

Figure 3.2: Receiver Block Diagram

3.1 EXPLANATION

The block diagram consists of two main parts: the transmitter side and the receiver side. The transmitter side includes the power supply, a microcontroller Atmega 328p, SOS (Switched-On Safety) button, sensors like HX710B, MQ9, MAX30102, DHT11, and a LoRa module. The receiver side includes another LoRa module, Node MCU, and an LCD display. At the core of the transmitter side is the Atmega 328p microcontroller. It receives data from various sensors such as HX710B (gas sensor), MQ9 (CO and combustible gas sensor), MAX30102 (pulse oximeter and heart rate sensor), andDHT11 (barometric pressure and temperature sensor). The SOS button is also connected to the Atmega 328p, allowing the rescue team member to signal for help in case of an emergency.

The Atmega 328p processes the data received from the sensors and sends it wirelessly using LoRa technology. The LoRa module allows for long-range communication, making it ideal for use in remote and hazardous environments. The data transmitted includes gas levels, temperature, humidity, heart rate, and SOS signal. On the receiver side, another LoRa module is used to receive the data transmitted by the transmitter side.

The Node MCU is used to process the data and display it on the LCD display. The display shows the data collected by the sensors, including gas levels, temperature, humidity, and heart rate. The SOS signal is also displayed prominently on the LCD display, alerting rescue team members to a potential emergency. Overall, the smart suit for rescue teams using Atmega 328p, SOS, sensors like HX710B, MQ9, MAX30102, DHT11, and LoRa technology provides real-time monitoring of the environment and the health of rescue team members. This enables rescue teams to take appropriate safety measures and respond to emergencies more effectively.

CHAPTER-4

4. HARDWARE DESCRIPTION

- Power Supply Unit
- ATMEGA328P
- SOS
- Node MCU
- LCD Display
- Sensors
    1. DHT11
    2. MAX30102
    3. HX710B
    4. MQ9
- LoRa
- GPS

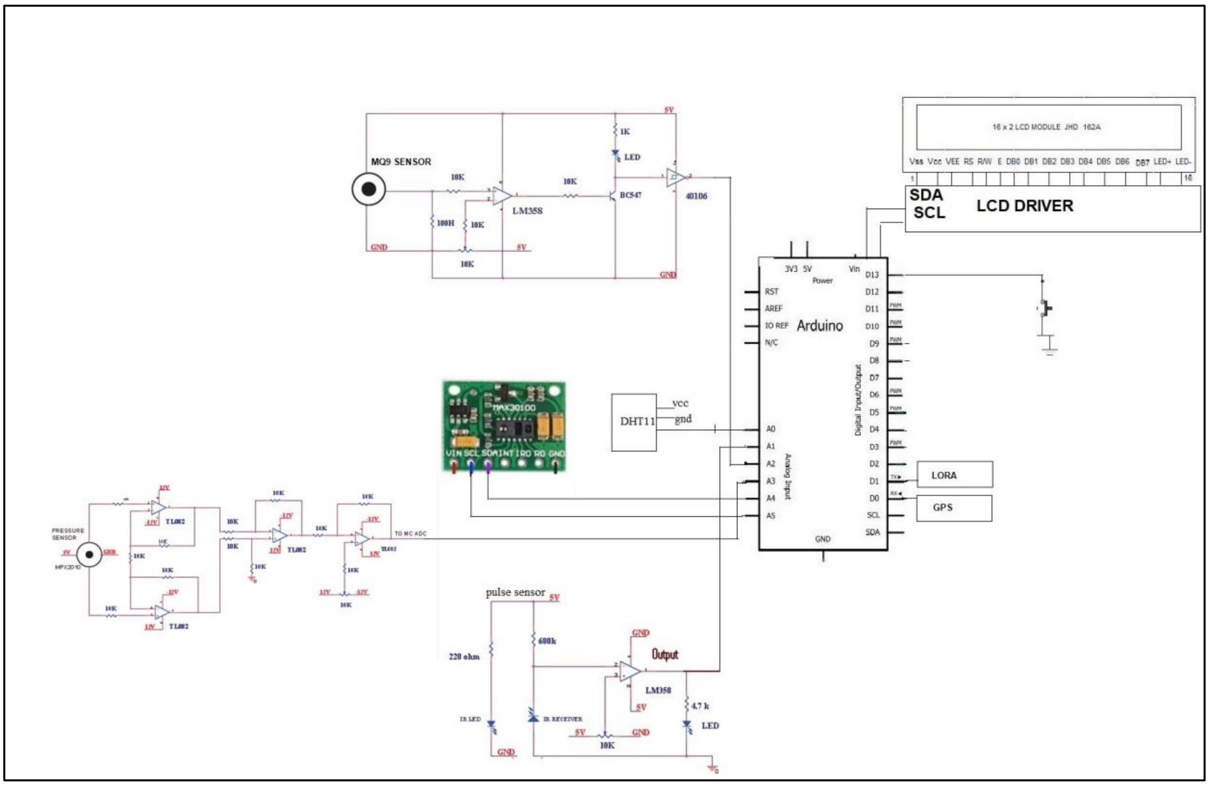

Figure 4.1: Hardware circuit connections of the system

4.1 LITHIUM-ION BATTERIES

Lithium-ion batteries are a type of rechargeable battery commonly used in various electronic devices, including smart suits for rescue teams. These batteries are preferred due to their high energy density, low self-discharge rate, and ability to retain their charge for long periods of time. In smart suits for rescue teams, lithium- ion batteries are used to power the sensors and microcontroller that are integrated into the suit. These sensors continuously collect data on various parameters, such as temperature, gas levels, and heart rate. The microcontroller processes this data and sends it wirelessly using a communication technology such as LoRa to a receiver. Lithium-ion batteries used in smart suits for rescue teams are often designed to be compact and lightweight to minimize the additional weight added to the rescue worker. The batteries also need to be durable enough to withstand the harsh conditions of the rescue operation. However, it is important to note that lithium-ion batteries have some safety concerns that need to be addressed. Therefore, proper safety measures and precautions should be taken during the design and use of smart suits to ensure the safe use of lithium-ion batteries.

4.2. ATMEGA 328 MICROCONTROLLER

LPC2148 microcontroller board based on a 16-bit/32-bit ARM7TDMI-S CPU with real- time emulation and embedded trace support, that combine microcontrollers with embedded high-speed flash memory ranging from 32kB to 512kB. A 128-bit wide memory interface and unique accelerator architecture enable 32-bit code execution at the maximum clock rate. For critical code size applications, the alternative 16-bit Thumb mode reduces code by more than 30% with a minimal performance penalty. The meaning of LPC is Low Power Low- Cost microcontroller. This is 32-bitmicrocontroller manufactured by Philips Semiconductors (NXP).

4.2.1 FEATURES OF ATMEGA328 MICROCONTROLLER

- High Performance, Low Power AVR
- 8-Bit Microcontroller Family
- AdvancedRISCArchitecture–131
- Powerful Instructions–Most Single Clock Cycle Execution–32x8
- General Purpose Working Registers
- Fully Static Operation
- Up to 20 MIPS Throughput at20MHz22
- On-chip 2-cycleMultiplier
- High Endurance Non-volatile Memory Segments
- 4/8/16/32KBytes of In-System Self-Programmable Flash program memory
- 256/512/512/1K Bytes EEPROM

4.2.2 PERIPHERAL FEATURE

Two 8-bit Timer/Counters with Separate Pre scaler and Compare Mode One 16-bit Timer/Counter with Separate Pre scaler, Compare Mode, and Capture Mode.

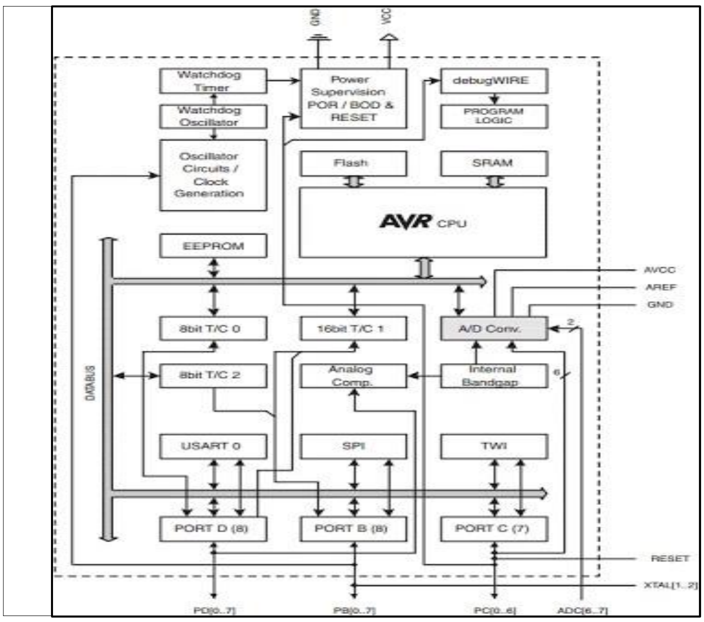

Figure 4.2: Microcontroller Architecture

4.2.3 ARCHITECTURAL OVERVIEW

The AVR core architecture in general. The main function of the CPU core is to ensure correct program execution. The CPU must, therefore, be able to access memories, perform calculations.

AVR uses a Harvard architecture with separate memories and buses for program and data. Instructions in the program memory are executed with a single level pipelining. While one instruction is being executed, the next instruction is prefetched from the program memory. This concept enables instructions to be executed in every clock cycle. The program memory is in- system reprogrammable flash memory. Status and Control Interrupt Unit 32x8 General Purpose Registers ALU Data Bus 8-bit Data SRAM SPI Unit Instruction Register Instruction Decoder Watchdog Timer Analog Comparator EEPROM I/O Lines I/O Module n Control Lines Direct Addressing Indirect Addressing I/O Module 2 I/O Module1Program Counter Flash Program. The fast-access register file contains 32 8- bit general purpose working registers with a single clock cycle access time.

This allows a single-cycle arithmetic logic unit (ALU) operation. In a typical ALU operation, two operands are output from the register file, the operation is executed, and the result is stored back in the register file – in one clock cycle. Six of the 32 registers can be used as three 16-bit indirect address register pointers for data space addressing – enabling efficient address calculations. After an arithmetic operation, the status register is updated to reflect information about the result of the operation. Program flow is provided by conditional and unconditional jump and call instructions, able to directly address the whole address space. Most AVR instructions have a single 16-bit word format. Every program memory address contains a 16- or 32-bit instruction. Program flash memory space is divided into two sections, the boot program section, and the application program section. Both sections have dedicated lock bits for write and read/write protection. The SPM instruction that writes into the application flash memory section must reside in the boot program section. During interrupts and subroutine calls, the return address program counter (PC) is stored on the stack. The stack pointer (SP) is read/write accessible in the I/O space.

The data SRAM can easily be accessed through the five different addressing modes supported in the AVR architecture. The memory spaces in the AVR architecture are all linear and regular memory maps. A flexible interrupt module has its control registers in the I/O space with an additional global interrupt enable bit in the status register. All interrupts have a separate interrupt vector in the interrupt vector table. The interrupts have priority in accordance with their interrupt vector position.

The lower the interrupt vector address, the higher the priority. The I/O memory space contains 64 addresses for CPU peripheral functions as control registers, SPI, and other I/O functions. The I/O memory can be accessed directly, or as the data space locations following those of the register file, 0x20 - 0x5F. In addition, the Atmega328P has extended I/O space from 0x60 - 0xFF in SRAM where only the ST/STS/STD and LD/LDS/LDD instructions can be used. 6.2ALU–Arithmetic Logic Unit The high-performance AVR ALU operates in direct connection with all the 32 general purpose working registers.

The stack pointer (SP) is read/write accessible in the I/O space. The data SRAM can easily be accessed through the five different addressing modes supported in the AVR architecture.

The memory spaces in the AVR architecture are all linear and regular memory maps. A flexible interrupt module has its control registers in the I/O space with an additional global interrupt enable bit in the status register.

The lower the interrupt vector address, the higher the priority. The I/O memory space contains 64 addresses for CPU peripheral functions as control registers, SPI, and other I/O functions. The I/O memory can be accessed directly, or as the data space locations following those of the register file, 0x20 - 0x5F.

In addition, the Atmega328P has extended I/O space from 0x60 - 0xFF in SRAM where only the ST/STS/STD and LD/LDS/LDD instructions can be used.

4.2.4 STATUS REGISTER

The status register contains information about the result of the most recently executed arithmetic instruction. This information can be used for altering program flow in order to perform conditional operations. Note that the status register is updated after all ALU operations, as specified in the instruction set reference. This will in many cases remove the need for using the dedicated compare instructions, resulting in faster and more compact code. The status register is not automatically stored when entering an interrupt routine and restored when returning from an interrupt.

The AVR status register – SREG – is defined as:

- **Bit7–I:** Global Interrupt Enable The global interrupt to enable bit must be set for the interrupts to be enabled. The individual interrupt enable control is then performed in separate control registers. If the global interrupt enables register is cleared, none of the interrupts are enabled independently of the individual interrupt enable settings.
- The I- bit is cleared by hardware after an interrupt has occurred, and is set by the RETI instruction to enable subsequent interrupts. The I-bit can also be set and cleared by the application with the SEI.
- **Bit 6 – T:** Bit Copy Storage The bit copy instructions BLD (bit LoaD) and BST (Bit Store) use the T-bit as source or destination for the

operated bit. A bit from a register in the register file can be copied into T by the BST instruction, and a bit in T can be copied into a bit in a register in the register file by the BLD instruction.

- **Bit 5 – H:** Half Carry Flag The half carry flag H indicates a half carry in some arithmetic operations. Half carry Is useful in BCD arithmetic.
- **Bit4–S:** Sign Bit, S=N V The S-bit is always an exclusive or between the negative flag N and the two's complement overflow flag V.
- **Bit3–V:** Two's Complement Overflow Flag The two's complement over flow flag V supports two's complement arithmetic's.
- **Bit 2 – N:** Negative Flag The negative flag N indicates a negative result in an arithmetic or logic operation.
- **Bit 1 – Z:** Zero Flag The zero flag Z indicates a zero result in an arithmetic or logic operation.
- **Bit 0 – C:** Carry Flag The carry flag C indicates a carry in an arithmetic or logic operation.

4.2.5 GENERAL PURPOSE REGISTER FILE

The register file is optimized for the AVR® enhanced RISC instruction set. In order to achieve the required performance and flexibility, the following input/output schemes are supported by the register file:

- One 8-bit output operand and one 8-bit result input
- Two 8-bit output operands and one 8-bit result input
- Two 8-bit output operands and one 16-bit result input
- One 16-bit output operand and one 16-bit result input Figure 6-2 shows the structure of the 32 general purpose working registers in the CPU.

4.2.6 STACK POINTER

The stack is mainly used for storing temporary data, for storing local variables and for storing return addresses after interrupts and subroutine calls. Note that the stack is implemented as growing from higher to lower memory locations. The stack pointer register always points to the top of the stack. The stack pointer points to the data SRAM stack area where the subroutine and interrupt stacks are located. A stack PUSH command will decrease the stack pointer. The stack in the data SRAM must be defined by the program before any subroutine calls are executed or interrupt are enabled. Initial stack pointer value equals the last address of the internal SRAM and the stack pointer must be set to point above the start of the SRAM.

4.2.7 RESET AND INTERRUPT HANDLING

AVR® provides several different interrupt sources. These interrupts and the separate reset vector each have a separate program vector in the program memory space. All interrupts are assigned individual enable bits which must be written logic one together with the global interrupt enable bit in the status register in order to enable the interrupt. Depending on the program counter value, interrupts may be automatically disabled when boot lock bits BLB02 or BLB12 are programmed. This feature improves software security. The lowest addresses in the program memory space are by default defined as the reset and interrupt vectors. The list also determines the priority levels of the different interrupts. The lower the address the higher is the priority level. RESET has the highest priority, and next is INT0 – the external interrupt request 0.

The interrupt vectors can be moved to the start of the boot flash section by setting the IVSEL bit in the MCU control register. When an interrupt occurs, the global interrupt enables I-bit is cleared and all interrupts are disabled. The user software can write logic one to the I- bit to enable nested interrupts. All enabled interrupts can then interrupt the current interrupt routine.

4.3 SOS SYSTEM

An SOS system is an important feature in a smart suit for rescue teams. The SOS system provides a quick and easy way for rescue workers to call for help in an emergency situation. Typically, an SOS button or switch is located on the smart suit, which can be easily accessed by the rescue worker.

When the SOS button is pressed, it sends a signal to the microcontroller in the smart suit. The microcontroller then initiates the transmission of an SOS message wirelessly using a communication technology such as LoRa to a receiver located outside the hazardous area. The SOS message typically includes information such as the location of the rescue worker and a brief description of the emergency.

The receiver then alerts other rescue team members, who can quickly respond to the emergency. The SOS system in a smart suit is an important safety feature that can help to save lives in emergency situations. It allows rescue workers to quickly and easily call for help when they are in distress, enabling them to receive timely assistance from their team members.

4.4 LCD DISPLAY

4.4.1 INTRODUCTION

Liquid crystal cell displays (LCDs) are used in similar applications where LEDs are used. These applications are display of display of numeric and alphanumeric characters in dot matrix and segmental displays.

LCDs are of two types:

- Dynamic scattering type
- Field effect type

4.4.2 WORKING

When sufficient voltage is applied to the electrodes the liquid crystal molecules would be aligned in a specific direction. The light rays passing through the LCD would be rotated by the polarizer, which would result in activating/highlighting the desired characters. The power supply should be of +5v, with maximum allowable transients of 10mv. To achieve a better/suitable contrast for the display the voltage (VL) at pin 3 should be adjusted properly. A module should not be removed from a live circuit.

The ground terminal of the power supply must be isolated properly so that voltage is induced in it. The module should be isolated properly so that stray voltages are not induced, which could cause a flicking display. LCD is lightweight with only a few, millimeters thickness since the LCD consumes less power, they are compatible with low power electronic circuits, and can be powered for long durations. LCD does not generate light and so light is needed to read the display. By using backlighting, reading is possible in the dark. LCDs have long life and a wide operating temperature range. Before LCD is used for displaying proper initialization should be done.

LCDs with a small number of segments, such as those used in digital watches and pocket calculators, have individual electrical contacts for each segment. An external dedicated circuit supplies an electric charge to control each segment. This display structure is unwieldy for more than a few display elements. Small monochrome displays such as those found in personal organizers, or older laptop screens have a passive-matrix structure employing super-twisted nematic (STN) or double-layer STN (DSTN) technology the latter of which addresses a color- shifting problem with the former and color-STN (CSTN) wherein color is added by using an internal filter. Each row or column of the display has a single electrical circuit.

The pixels are addressed one at a time by row and column addresses. This type of display is called passive-matrix addressed because the pixel must retain its state between refreshes without the benefit of a steady electrical charge. High-resolution color displays such as modern LCD computer monitors and televisions use an active-matrix structure. A matrix of thin-film transistors (TFTs) is added to the polarizing and color filters.

Each pixel has its own dedicated transistor, allowing each column line to access one pixel. When a row line is activated, all of the column lines are connected to a row of pixels and the correct voltage is driven onto all of the column lines. The row line is then deactivated and the next row line is activated

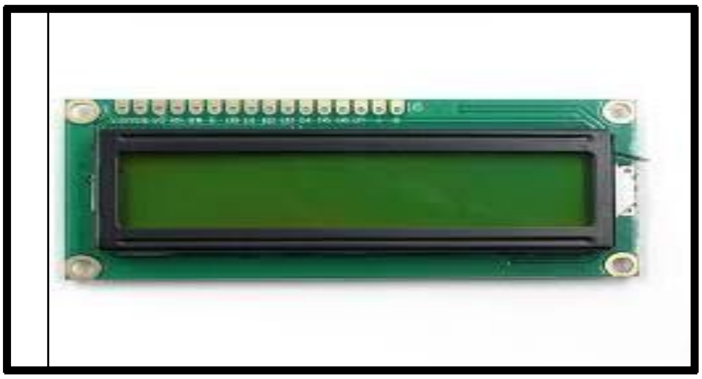

Figure 4.3: LCD Display

4.4.3 LCD PIN DESCRIPTIONS

The function of each pin of LCD is described below VCC, VSS and VEE while v and v provide +5v and ground, respectively, v is used for controlling LCD contrast.

There are two very important registers inside the LCD. The RS pin is used for their selection as follows.

If RS=0, the instruction code register is selected, allowing the user to send a command such as clear display, cursor at home, etc.

If RS=1 the data register is selected, allowing the user to send data to be displayed on the LCD.

R/W, read/write

R/W input allows the user to write information to the LCD or read information from it. R/W=1 when reading; R/W=0 when writing.

E, enable

The enable pin is used by the LCD to latch information presented on its data pins. When data is supplied to data pins, a high to low pulse must be applied to this pin in order for the LCD to latch in the data present at the data pins.

D0 - D7

The 8-bit data pins, D0 – D7, are used to send information to the LCD or read contents of the LCD'S internal registers. There are also instruction codes that can be sent to the LCD to clear the display or force the cursor to the home position or blink the cursor.

4.4.4 LCD INTERFACING WITH MICROCONTROLLER:

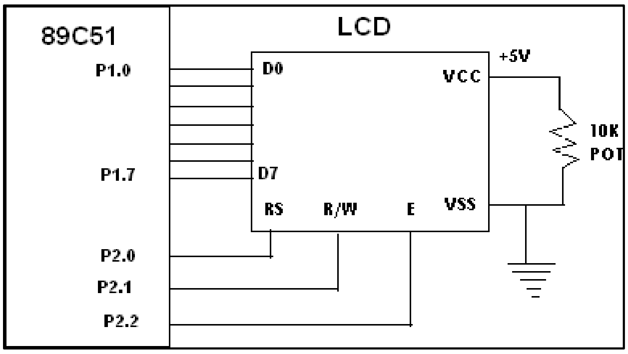

Figure 4.4: Interfacing LCD with Microcontroller

4.4.5 ADVANTAGES

- Consume much lesser energy (i.e., low power) when compared to LEDs.
- Utilizes the light available outside and no generation of light.
- It is more suitable to act as display elements.

- Since reflectivity is highly sensitive to temperature, used as temperature measuring sensor.
- Very cheap.

DISADVANTAGES

- Angle of viewing is very limited.
- External light is a must for display.
- Since not generating its own light and makes use of external light for display, contrast is poor.
- Cannot be used under wide range of temperature.

4.4.6 APPLICATIONS

- Watches
- Fax & Copy machines & Calculators.

4.5 NODE MCU

4.5.1. INTRODUCTION

- Your ESP8266 is an impressive, low-cost Wi-Fi module suitable for adding Wi-Fi functionality to an existing microcontroller project via a UART serial connection.
- The module can even be reprogrammed to act as a standalone Wi-Fi connected device–just add power!

The feature list is impressive and includes:

- 802.11 b/g/n protocol
- Wi-Fi Direct (P2P), soft-AP
- Integrated TCP/IP protocol stack

The hardware connections required to connect to the ESP8266 module are fairly straight-forward but there are a couple of important items to note related to power: The ESP8266 requires 3.3V power–do not power it with 5 volts!

The ESP8266 needs to communicate via serial at 3.3V and does not have 5V tolerant inputs, so you need level conversion to communicate with a 5V microcontroller like most Arduinos use.

4.5.2 CONNECTIONS AVAILABLE ON THE ESP8266 WIFI MODULE

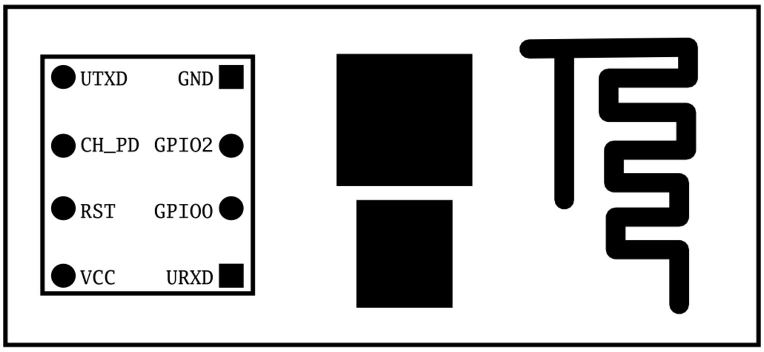

Figure 4.5: ESP8266 Top View

When power is applied to the module you should see the red power light turn on and the blue serial indicator light flicker briefly.

With FTDI 3.3V Board (Legit)

If you have a 3.3V FTDI Serial to USB board you can get started without fear of destroying your newESP8266 Wi-Fi module. Do note that many FTDI boards have a solder jumper to convert from 3.3V to5V operation so ensure it is set to enable 3.3V operation.

Using Arduino IDE Serial Monitor

If you already have the Arduino IDE installed the easiest way to get started is to use the built-in Serial Monitor:

- Plug in the Wi-Fi module.
- Choose the correct serial port from the Tools > Serial Port menu.
- Open the Serial Monitor via the Tools menu or "magnifying glass" icon on the editor window.
- For the default firmware version (00160901), ensure Carriage return is selected in the line ending pop-up menu at the bottom of the serial monitor.
- For the default firmware version, ensure the communication speed is set to 115200 baud.

Using GNU Screen

It's possible to use GNU Screen out of the box with the default version of the firmware (00160901) which expects Carriage-Return-only line endings, e.g. (on OS X): screen /dev/tty.usbserial-AB12345 115200

Unfortunately, the updated firmware versions require Carriage-Return-and- New-Line line endings and there appears to be no way to configure screen to send both with one key press. Instead, you need to press <enter> or Ctrl-M then follow that with Ctrl-J. You might have more success with something like minicom or picoohm with firmware versions.

4.6 SENSORS

4.6.1 HX710B

The HX710B Air Pressure Sensor Module monitors the air's pressure in the range of 0 to 40 kilopascals (kPa) and turns it into an electrical impulse which microcontrollers or other electrical gadgets can interpret and read.

It is a common and frequently utilized sensor in a wide range of applications that demand precise pressure sensing, including weather monitoring, industrial control systems, and medical equipment.

The HX710B device is a piezoresistive pressure sensor which determines pressure by detecting variations in electrical resistance produced by pressure-induced displacement of a sensing element.

The module is made up of a pressure sensing element, a signal conditioning circuit, and an output amplifier. The sensor chip is a tiny, thin-film device having a diaphragm and piezoresistive components.

The signal conditioning circuit boosts and purifies the electrical signal from the sensing element, and the output amplifier generates a voltage or current proportionate to the pressure detected by the sensor.

Figure 4.6: HX710B sensor

4.6.2 MQ9

The MQ9 sensor is another type of gas sensor that is commonly used in smart suits for rescue teams. This sensor is capable of detecting various types of gases, including carbon monoxide, methane, and propane. The MQ9 sensor works on a similar principle to the HX710B sensor.

When the gas being detected comes into contact with the sensing material in the sensor, it causes a change in the resistance of the sensor. This change in resistance is then measured by the microcontroller in the smart suit, which can determine the concentration of the gas. In a smart suit for rescue teams, the MQ9 sensor can be used to detect the presence of toxic gases and alert the rescue worker in case of an emergency.

For example, if the sensor detects a high concentration of carbon monoxide, it can trigger an alarm to alert the rescue worker to evacuate the area immediately.

The MQ9 sensor is a low-cost and effective solution for gas detection in smart suits for rescue teams. It is commonly used due to its high sensitivity and accuracy in detecting a range of different gases.

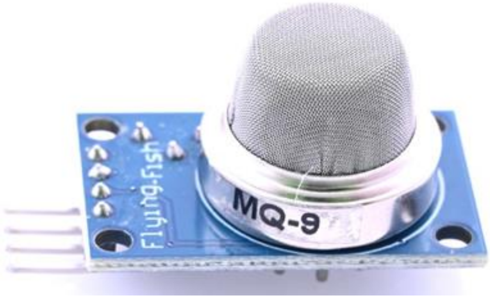

Figure 4.7: MQ9 Sensor

4.6.3 MAX30102

The MAX30102 is an integrated optical sensor module specifically designed for heart rate and blood oxygen level monitoring.

It utilizes photoplethysmography (PPG) technology to measure heart rate. It emits red and infrared light onto the skin and measures the reflected light to detect changes in blood volume. The sensor calculates the heart rate based on these measurements.

It also provides an estimation of blood oxygen saturation (SpO2) levels. It uses the absorption of red and infrared light by oxygenated and deoxygenated blood to estimate the oxygen saturation level.

This helps improve the accuracy and reliability of heart rate and SpO2 measurements, even in environments with varying lighting conditions.

It features high-sensitivity photodetectors that enable accurate measurement of even low amplitude blood flow signals. This enhances the performance of the sensor in detecting heart rate and SpO2 levels.

It is designed to be energy-efficient, allowing for longer battery life in portable and wearable applications. It offers low-power modes to minimize power consumption when not actively measuring.

The sensor module has a compact size, making it suitable for applications with space constraints or wearable devices where size and weight are important considerations.

Figure 4.8: MAX30102 Sensor

4.6.4 DHT11

The DHT11 sensor is a type of sensor that is commonly used in smart suits for rescue teams for measuring atmospheric pressure and temperature. It uses a piezoresistive pressure sensor and a temperature sensor to measure these values. The DHT11 sensor measures atmospheric pressure by sensing the deflection of a silicon diaphragm.

The sensor can also measure temperature using an on-chip temperature sensor. In a smart suit for rescue teams, the DHT11 sensor can be used to measure atmospheric pressure and temperature inside the hazardous area. This allows other rescue team members to monitor the atmospheric conditions inside the hazardous area and take appropriate action in case of an emergency.

The DHT11 sensor is a low-cost and effective solution for measuring atmospheric pressure and temperature. It is commonly used in smart suits for rescue teams due to its accuracy and reliability in measuring these values.

Figure 4.9: DHT11 SENSOR

4.7 LORA MODULE

4.7.1 INTRODUCTION

A LoRa module is a type of wireless communication module that uses a long- range, low-power modulation technique known as LoRa (short for "Long Range").

LoRa technology enables long-range communication with low power consumption, making it suitable for use in smart suits for rescue teams, Internet of Things (IoT) devices, and other applications that require low-power, long-range wireless communication.

Lora modules consist of a LoRa transceiver chip, typically manufactured by Semtech Corporation, and a microcontroller unit (MCU) that controls the LoRa transceiver. LoRa modules can transmit and receive data over distances of several kilometers, depending on the environment and the power output of the module.

Lora modules use a frequency-hopping spread spectrum (FHSS) technique to avoid interference with other wireless signals in the same frequency band. This allows LoRa modules to coexist with other wireless devices without interfering with their operation.

Lora modules are commonly used in smart suits for rescue teams for transmitting sensor data, such as vital signs and environmental conditions, from the suit to a remote receiver. The low power consumption and long-range capabilities of LoRa technology make it an ideal choice for these types of applications.

Figure 4.10: LoRa module

4.7.2 FEATURES

4.7.2.1 Long Range: LoRa modules are designed to provide long-range wireless communication, typically up to several kilometers in open areas. Low Power.

4.7.2.2 Consumption: LoRa modules are designed to operate on low power, making them ideal for battery-powered devices and IoT applications.

4.7.2.3 High Sensitivity: LoRa modules have high sensitivity to detect weak signals, which enables them to communicate over long distances while maintaining a low power.

4.7.2.4 Consumption Secure: LoRa modules use AES encryption for secure communication, ensuring that data is transmitted and received securely.

4.7.2.5 Frequency Hopping Spread Spectrum (FHSS): LoRa modules use FHSS to avoid interference with other wireless signals in the same frequency band, allowing them to coexist with other wireless devices.

4.7.2.6 Easy to Use: LoRa modules are designed to be easy to use, with simple interfaces and programming tools available for developers.

4.7.2.7 Low Cost: LoRa modules are typically low-cost compared to other wireless communication technologies, making them an affordable choice for IoT and smart suit applications.

4.7.3. WORKING

4.7.3.1 TRANSMISSION

- The data to be transmitted is fed into the LoRa module through the microcontroller unit (MCU).
- The MCU generates a waveform based on the LoRa modulation technique.
- The LoRa transceiver chip amplifies the waveform and transmits it over the air using an antenna.
- The LoRa transceiver chip also uses frequency hopping spread spectrum (FHSS) to avoid interference from other wireless signals in the same frequency band.
- The transmitted data is received by another LoRa module located within the range of the transmitter

4.7.3.2 RECEPTION

- The LoRa transceiver chip in the receiver module detects the incoming signal from the transmitter.
- The signal is amplified and demodulated by the LoRa transceiver chip to extract the original data.
- The data is then passed on to the MCU for processing and analysis.

LoRa modules use a unique modulation technique that enables long-range communication with low power consumption.

The LoRa modulation technique uses chirp spread spectrum (CSS) modulation, which spreads the signal across a wider frequency range than traditional modulation techniques. This results in a signal that can travel long distances while using low power. In addition to CSS modulation, LoRa modules use forward error correction (FEC) and adaptive data rate (ADR) techniques to ensure reliable data transmission over long distances.

FEC adds redundancy to the transmitted data to correct any errors that may occur during transmission, while ADR adjusts the data rate dynamically based on the signal strength and noise level of the transmission environment.

4.8 GPS

4.8.1 FEATURES

- **Accuracy:** GPS systems can be accurate to within a few meters, but this level of precision may not be sufficient for some applications.
- **Power consumption:** GPS systems can consume a significant amount of power, which could be a challenge for a smart suit that is designed to be lightweight and portable. It may be necessary to optimize the design of the GPS system to minimize power consumption and maximize battery life.
- **User interface:** The user interface for a GPS system in a smart suit should be designed to be intuitive and easy to use. This may involve integrating the GPS functionality with other features of the suit, such as a heads-up display or haptic feedback.

Figure 4.11. GPS

- **Data storage:** The GPS system will need to store data on the wearer's location, which could be used for navigation, tracking, or other purposes. The amount of data that needs to be stored will depend on the application, but it may be necessary to include a data storage device, such as a solid-state drive, in the smart suit.
- **Connectivity:** Depending on the application, it may be necessary to provide connectivity for the GPS system, such as Wi-Fi or cellular data.

CHAPTER-5

5. SOFTWARE REQUIREMENTS

5.1 ARDUINO IDE

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and Genuine hardware to upload programs and communicate with them.

In this chapter, we will learn about the different components on the Arduino board. We will study the Arduino UNO board because it is the most popular board in the Arduino board family. In addition, it is the best board to get started with electronics and coding. Some boards look a bit different from the one given below, but most Arduinos have a majority of these components in common.

Various kinds of Arduino boards are available depending on different microcontrollers used. However, all Arduino boards have one thing in common: they are programmed through the Arduino IDE. The differences are based on the number of inputs and outputs (the number of sensors, LEDs, and buttons you can use on a single board), speed, operating voltage, form factor, etc. Some boards are designed to be embedded and have no programming interface (hardware), which you would need to buy separately. Some can run directly from a 3.7V battery, others need at least5V. Arduino is a prototype platform (open-source) based on easy-to-use hardware and software. It consists of a circuit board, which can be programmed (referred to as a microcontroller) and a ready-made software called Arduino.

The key features are:

Arduino boards are able to read Analog or Digital input signals from different sensors and turn it into an output such as activating a motor, turning LED on/off, connect to the cloud and many other actions.

You can control your board functions by sending a set of instructions to the microcontroller on the board via Arduino IDE (referred to as uploading software).

Unlike most previous programmable circuit boards, Arduino does not need an extra piece of hardware (called a programmer) in order to load a new code onto the board. You can simply use a USB cable.

Additionally, the Arduino IDE uses a simplified version of C++, making it easier to learn to program.

In this chapter, we will study in depth, the Arduino program structure and we will learn more new terminologies used in the Arduino world. The Arduino software is open- source. The source code for the Java environment is released under the GPL and the C/C++ microcontroller libraries are under theLGPL.

Sketch − The first new terminology is the Arduino program called “sketch”. Structure Arduino programs can be divided into three main parts: **Values** (variables and constants), and Functions. In this tutorial, we will learn about the Arduino software program, step by step, and how we can write the program without any syntax or compilation error.

Let us start with the Structure. Software structure consists of two main functions

- Setup( ) function
- Loop( ) function

Purpose − After creating a setup () function, which initializes and sets the initial values, the loop () function does precisely what its name suggests, and loops consecutively, allowing your program to change and respond.

Use it to actively control the Arduino board.

- INPUT
- OUTPUT
- RETURN

Decision making structures require that the programmer specify one or more conditions to be evaluated or tested by the program. It should be along with a statement or statements to be executed if the condition is determined to be true, and optionally, other statements to be executed if the condition is determined to be false. Following is the general form of a typical decision-making structure found in most of the programming languages.

5.2 EMBEDDED C

An embedded system is a computer system designed to perform one or a few dedicated functions often with real-time computing constraints. It is embedded as part of a complete device often including hardware and mechanical parts.

By contrast, a general-purpose computer, such as a personal computer (PC), is designed to be flexible and to meet a wide range of end-user needs. Embedded systems control many devices in common use today.

Embedded systems are controlled by one or more main processing cores that are typically either microcontrollers or digital signal processors (DSP). The key characteristic, however, is being dedicated to handle a particular task, which may require very powerful processors.

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

Embedded Systems talk with the outside world via peripherals, such as:

● Serial Communication Interfaces (SCI): RS-232, RS-422, RS-485 etc.

● Synchronous Serial Communication Interface: I2C, SPI, SSC and ESSI (Enhanced Synchronous Serial Interface).

Advantages

- Reliability
- Simple control loop
- Interrupt controlled system
- Cooperative multitasking
- Preemptive multitasking or multi-threading

5.2.1 SOFTWARE CODING

Transmitter

```
#include <Wire.h>
#include "MAX30105.h"
#include <LiquidCrystal_I2C.h>
#include <LiquidCrystal.h>
#include "HX710B.h"
#include <DHT.h>
#define DHTPIN A1
#define Emergency_button  8
#define DOUT  3
#define CLK  2
int annalogpin[] = {A0,A2};
DHT dht(DHTPIN, DHT11);
HX710B scale;
float calibration_factor = 1000;
LiquidCrystal_I2C lcd(0x27, 16, 2);
```

```
MAX30105 particleSensor;
int sentT;
int
pre_sec,sec,bpm,pressure,highPres,lowPres,presSEC,h,temp,gasSensor,gas
SensorP,gasSensorF;
bool bpstart,prmf,buttonStat;
void setup()
{
  Serial.begin(9600);
  if (!particleSensor.begin(Wire, I2C_SPEED_FAST))
  {
    while (1);
  }
  particleSensor.setup();
  particleSensor.setPulseAmplitudeRed(0x0A);
  particleSensor.setPulseAmplitudeGreen(0);
  lcd.init();
  lcd.backlight();
  lcd.setCursor(0, 0);
  lcd.print("  Smart suit  ");
  lcd.setCursor(0, 1);
  lcd.print("                ");
  delay(1000);
  lcd.clear();
  scale.begin(DOUT, CLK);
  scale.set_scale();
  scale.tare();
  long zero_factor = scale.read_average();
  scale.set_scale(calibration_factor);
```

```
  dht.begin();
  pinMode(Emergency_button,INPUT_PULLUP);
  gasSensorP=parameterRead(0, 1000);
  if(gasSensorP>200)
  {
    gasSensorP-200;
  }
}
void loop()
{
  scale.set_scale(calibration_factor);
  pressure=scale.get_units();
  h = dht.readHumidity();
  temp=dht.readTemperature();
  if(pressure<100)
  {
    prmf=0;
  }
  if(pressure>100)
  {
    presSEC++;
    if(presSEC>30)
    {
      prmf=1;
      presSEC=0;
    }
    if(prmf==0)
    {
      highPres=pressure/8;
```

```
    lowPres=(highPres*10)/15;
   }
  }
  long irValue = particleSensor.getIR();
  if(irValue>50000)
  {
   pre_sec++;
   if(pre_sec>=3)
   {
    bpstart=1;
   }
  }
  if(bpstart==1)
  {
   sec++;
  }
  if(sec>59)
  {
   sec=0;
   if(irValue>50000)
   {
    bpm=(irValue/1000)-4;
   }
   else if(irValue<5000)
   {
bpm=(irValue/1000)+((irValue%1000)/100)+(((irValue%1000)%100)/10)+
(((irValue%1000)%100)%10);
   }
   if(bpm>99)
```

```
  {
    bpm=99;
  }
  bpstart=0;
}
buttonStat=digitalRead(Emergency_button);
gasSensor=(parameterRead(0, 10));
gasSensorF= gasSensor;
lcd.setCursor(0, 0);
WriteIntToLCD2digit(sec);
lcd.print(" ");
WriteIntToLCD2digit(bpm);
lcd.print(" ");
WriteIntToLCD3digit(highPres);
lcd.print("/");
WriteIntToLCD2digit(lowPres);
lcd.print("mmHg");
lcd.setCursor(0, 1);
WriteIntToLCD2digit(h);
lcd.print("% ");
WriteIntToLCD2digit(temp);
lcd.print(char(0xDF));
lcd.print("C ");//9
if(gasSensorF<200)
{
  lcd.print("G-0");
}
else if(gasSensorF>=200)
{
```

```
    lcd.print("G-1");
  }
  lcd.print(" ");
  lcd.print(buttonStat);
  delay(500);
  sentT++;
  if(sentT>5)
  {
    sentT=0;
    Serial.write('*');
    Send2IntToSerial(bpm);
    Send3IntToSerial(highPres);
    Send2IntToSerial(lowPres);
    Send2IntToSerial(h);
    Send2IntToSerial(temp);
    Send3IntToSerial(gasSensorF);
    Serial.print(buttonStat);
    Serial.write('#');
  }
}
void WriteIntToLCD4digit(unsigned int dat)
{
  lcd.print(dat/1000);
  lcd.print((dat%1000)/100);
  lcd.print(((dat%1000)%100)/10);
  lcd.print(((dat%1000)%100)%10);
}
void WriteIntToLCD3digit(unsigned int dat)
{
```

```
  lcd.print(dat/100);
  lcd.print((dat%100)/10);
  lcd.print((dat%100)%10);
}
void WriteIntToLCD2digit(unsigned int dat)
{
  lcd.print(dat/10);
  lcd.print(dat%10);
}
void Send4IntToSerial(unsigned int dat)
{
  Serial.print(dat/1000);
  Serial.print((dat%1000)/100);
  Serial.print(((dat%1000)%100)/10);
  Serial.print(((dat%1000)%100)%10);
}
void Send3IntToSerial(unsigned int dat)
{
  Serial.print(dat/100);
  Serial.print((dat%100)/10);
  Serial.print((dat%100)%10);
}
void Send2IntToSerial(unsigned int dat)
{
  Serial.print(dat/10);
  Serial.print(dat%10);
}
int parameterRead(unsigned char ad, int sample)
{
```

```
  int i, Iadc, I_v;
   Iadc = 2; I_v = 0;
  for (i = 0; i < sample; i++)
  {
   Iadc = analogRead(annalogpin[ad]);
   if (Iadc > I_v)
   {
     I_v = Iadc;
   }
   delay(1);
  }
  return I_v;
}
```

Receiver:

```
#include "ThingSpeak.h"
#include <LiquidCrystal_I2C.h>
#include <ESP8266WiFi.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
#define relay  D5
unsigned long myChannelNumber = 720961;
const char * myWriteAPIKey = "4MOXA1THSYVM9UMX";
const char *ssid =  "electronics";
const char *pass =  "wifihotspot";
WiFiClient client;
char inChar;
char rx1[25];
char rx2[25];
char xy,y,cx;
```

```
bool stringComplete = false;
int  ult,ult2;
int bpm,pressure,highPres,lowPres,h,temp,gasSensorF;
int buttonStat;
void setup()
{
  lcd.init();
  lcd.backlight();
  Serial.begin(9600);
  Serial.print("Connecting...");
  Serial.println(ssid);
  pinMode(relay,OUTPUT);
  digitalWrite(relay,LOW);
  lcd.setCursor(0, 0);
  lcd.print("   smart suit  ");
  lcd.setCursor(0, 1);
  lcd.print("                ");
  Serial.print("Connect ing...");
  Serial.println(ssid);
  WiFi.begin(ssid, pass);
  while (WiFi.status() != WL_CONNECTED)
  {
    delay(500);
    Serial.println(",");
  }
  Serial.println("Connected ok...");
  ThingSpeak.begin(client);
  lcd.clear();
}
```

```
void loop()
{
  if (stringComplete)
  {
    cx++;
    if(cx>2)
    {
      bpm=((rx1[1]-48)*10)+(rx1[2]-48);
      highPres=((rx1[3]-48)*100)+((rx1[4]-48)*10)+(rx1[5]-48);
      lowPres=((rx1[6]-48)*10)+(rx1[7]-48);
      h=((rx1[8]-48)*10)+(rx1[9]-48);
      temp=((rx1[10]-48)*10)+(rx1[11]-48);
      gasSensorF=((rx1[12]-48)*100)+((rx1[13]-48)*10)+(rx1[14]-48);
      buttonStat=(rx1[15]-48);
      for(y=0;y<=17;y++)
      {
        rx1[y]=0;
      }
      stringComplete = false;
      inChar=0;
      xy=0;
      cx=0;
    }   }
  lcd.setCursor(0, 0);
  WriteIntToLCD2digit(bpm);
  lcd.print(" ");
  WriteIntToLCD3digit(highPres);
  lcd.print("/");
  WriteIntToLCD2digit(lowPres);
```

```
lcd.print("mmHg");
lcd.setCursor(0, 1);
WriteIntToLCD2digit(h);
lcd.print("% ");
WriteIntToLCD2digit(temp);
lcd.print(char(0xDF));
lcd.print("C ");//9
if(gasSensorF<200)
{
  lcd.print("G-0");
}
else if(gasSensorF>=200)
{
  lcd.print("G-1");
}
lcd.print(" ");
lcd.print(buttonStat);
ult2++;
if(ult2>15)
{
  ult2=0;
  ult++;
}
if (ult > 15)
{
  digitalWrite(relay,HIGH);
  delay(500);
  lcd.clear();
  lcd.setCursor(0, 0);
```

```
lcd.print("UPLOADING");
delay(2000);
ThingSpeak. setField(1, bpm);
ThingSpeak. setField(2, highPres);
ThingSpeak. setField(3, lowPres);
ThingSpeak. setField(4, h);
ThingSpeak. setField(5, temp);
ThingSpeak. setField(6, gasSensorF);
ThingSpeak. setField(7, buttonStat);
int x = ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);
if (x == 200)
{
  lcd.setCursor(0, 1);
  lcd.print("UPLOADED");
  ult = 0;
  delay(3000);
  digitalWrite(relay,LOW);
  delay(300);
}
else
{
  lcd.setCursor(0, 1);
  lcd.print("NW error");
  lcd.setCursor(10, 1);
  lcd.print(String(x));
  delay(1500);
  delay(1500);
}
lcd.clear();
```

```
  }}
int String_Compare(unsigned char src[], unsigned char dest[], unsigned
char v)
{
  int i;
  for (i = 0; i <= v && src[i] == dest[i]; i++);
  return src[i] - dest[i];
}
void serialEvent()
{
  if(ult<13)
  {
    while (Serial.available())
    {
      inChar = (char)Serial.read();
      rx1[xy]= inChar;
      if(rx1[0]=='*')
      {
        xy++;
      }
      else if(rx1[0]!='*')
      {
        xy=0;
      }
      if (inChar == '#')
      {
        stringComplete = true;
      }}}  }
void WriteIntToLCD4digit(unsigned int dat)
```

```
{
  lcd.print(dat / 1000);
  lcd.print((dat % 1000) / 100);
  lcd.print(((dat % 1000) % 100) / 10);
  lcd.print(((dat % 1000) % 100) % 10);
}
void WriteIntToLCD3digit(unsigned int dat)
{
  lcd.print(dat / 100);
  lcd.print((dat % 100) / 10);
  lcd.print((dat % 100) % 10);
}
void WriteIntToLCD2digit(unsigned int dat)
{
  lcd.print(dat / 10);
  lcd.print(dat % 10);
      }
```

CHAPTER-6

6. RESULTS

Smart suits with built-in sensors for monitoring vital signs can provide real-time data on the physical well-being of rescue team members. This information can be crucial in identifying signs of decreased pulse rate, change in pressure and temperature or medical emergencies, allowing prompt intervention and ensuring the safety and health of the team. Various parameter analysis is shown in table 6.1.

SENSOR NAME	MEASURED PARAMETERS	OBSERVED VALUES
HX710B	Pressure	089/59mmHg
MQ9	Inflammable gases	237
MAX30102	Pulse rate	75
DHT11	Humidity and temperature	75% 30’c

TABLE 6.1 : Output Analysis

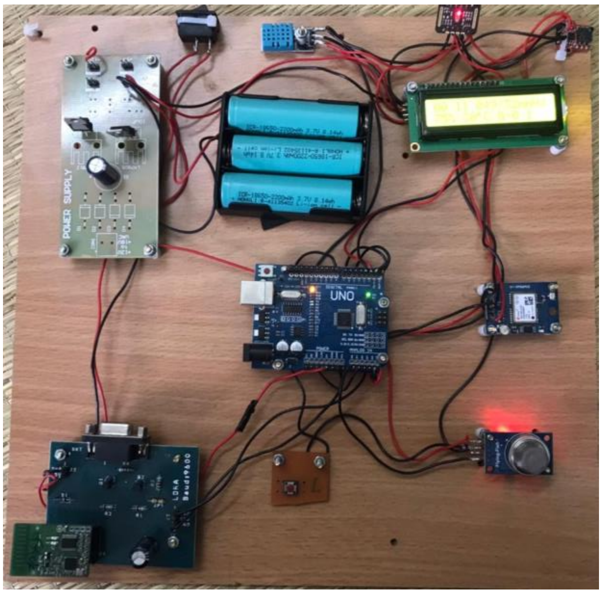

Figure 6.1 : Transmitter Hardware

The setup shown above will be embedded in a coat or clothing material that is resistant to various factors like fire and heat. The data is collected from the setup is send to the cloud and is viewed as shown below.

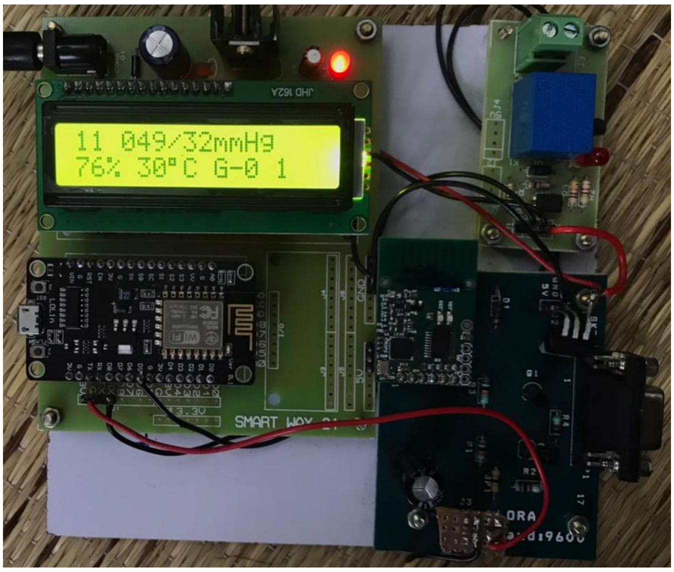

Figure 6.2: Receiver-Hardware

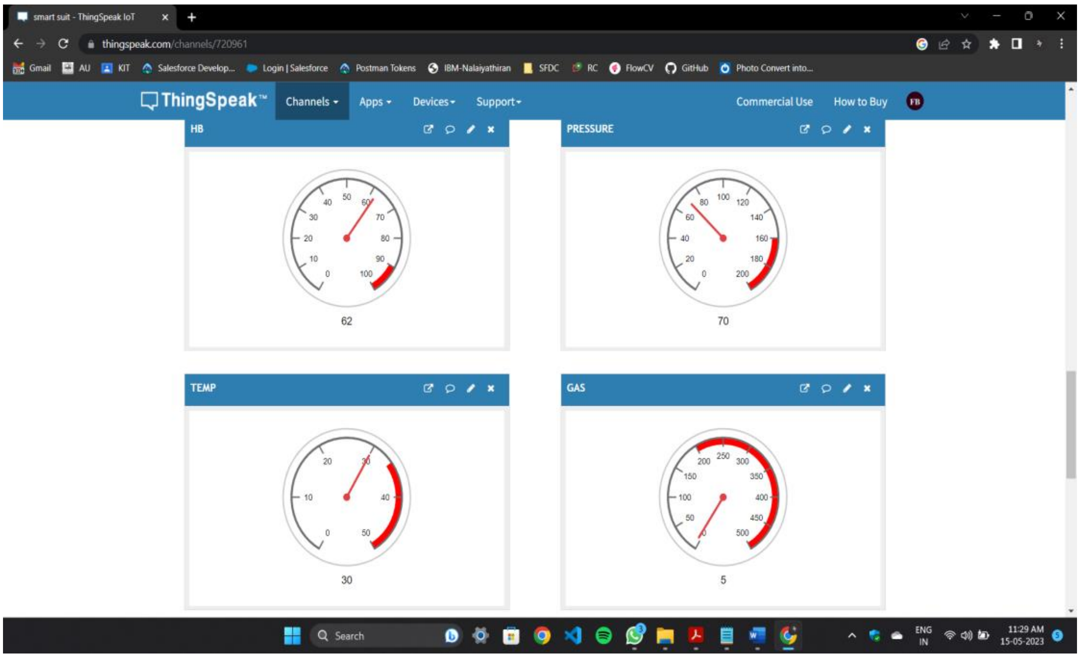

Figure 6.3: Output displayed in cloud

The data from the receiver side is uploaded and fetched in the cloud (ThingSpeak) as shown above.

The data is also updated continuously to the mobile application as shown in the below figure.

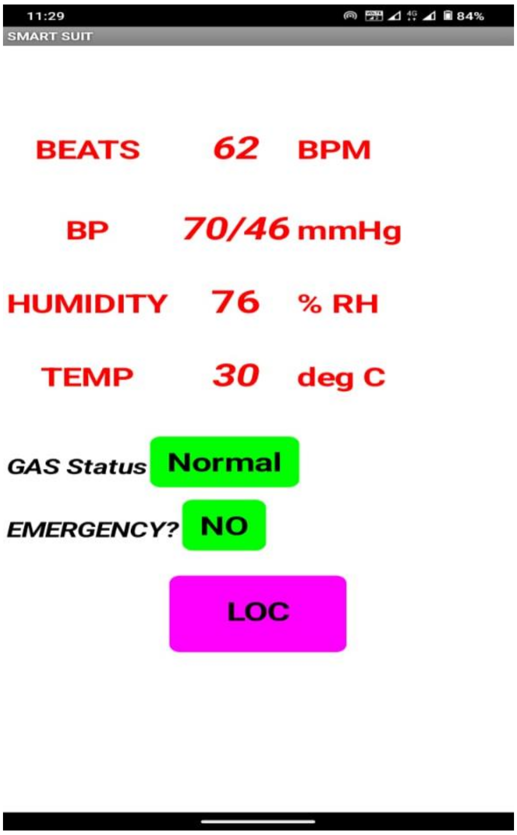

Figure 6.4: Output Obtained from Receiver

CHAPTER 7

7. CONCLUSION

According to the analysis, the existing methods are time and money consuming. The suggested solution not only resolves these issues but also increases accuracy. It is the greatest cost-effective method offered to improve the safety of the rescue team workers. It could be feasible to prevent the loss of money and valuable human lives. Also, it saves money and time.

In conclusion, this smart suits for rescue teams offer numerous benefits and advancements in enhancing the safety and effectiveness of rescue operations. These suits are equipped with advanced technologies and features that aid in communication, monitoring vital signs, providing protection, and improving situational awareness. They hold great promise in revolutionizing rescue operations by providing advanced capabilities, enhancing safety, and improving the overall efficiency and effectiveness of rescue teams.

7.1 FUTURE WORK

The future of smart suits for rescue team is likely to involve the development and integration of advanced technologies that can provide higher levels of accuracy, efficiency, and automation. These improve the safety and reliability. Continuous efforts should be made to improve the comfort, ergonomics, and usability of smart suits. This includes designing suits that fit a wide range of body types, considering user feedback in the development process, and conducting rigorous user testing to ensure the suits meet the specific needs and preferences of rescue team members. Exploring materials and technologies that enable self-repairing capabilities can extend the lifespan of smart suits and reduce maintenance needs

CHAPTER 8

REFERENCES

1. Lukowicz, P., Baker, M.G., Paradiso, and J.: Guest Editors' Introduction: Hostile Environments. IEEEPervasive Computing.9,13– 15 (2010).

2. Kwon, G.H., Smith-Jackson, T.L., Bostian, C.W.: Socio-cognitive aspects of inter-operability: Understanding communication task environments among different organizations. ACM Transactions on Computer-Human Interaction.18,1–21(2011).

3. Kaur, R., & Singh, R. (2017). Smart textiles and their applications. Journal of Textile Science & Engineering, 7(3), 1-9.

4. Sonmez, F., & Gok, T. (2017). Wearable smart textiles and their applications. Journal of Textile and Apparel Technology and Management, 10(4), 1-13.

5. Zhou, X., Li, Y., & Chou, S. Y. (2018). Smart clothing: Connecting human with clouds and big data for sustainable health monitoring. Sustainable Cities and Society, 38, 196-205.

6. Dunne, L. E., & Keogh, W. (2018). Smart clothing: the future or a fad? The Journal of The Textile Institute, 109(9), 1175-1187.

7. Wang, Y., Wu, Y., Li, C., & Yu, Y. (2018). Development of smart protective clothing based on Internet of things technology. Textile Research Journal, 88(10), 1157-1172.

8. Zhou, X., Li, Y., & Chou, S. Y. (2018). Smart clothing: Connecting human with clouds and big data for sustainable health monitoring. Sustainable Cities and Society, 38, 196-205.

9. Wu, S., Li, H., & Liu, Y. (2019). The development of smart clothing: A review. Journal of Fiber Bioengineering and Informatics, 12(2), 295-307.

10. Yang, S., Li, X., & Guo, X. (2019). Development of smart clothing for health monitoring: A review. Journal of Sensors, 2019, 1-17.

11. Siddiqui, M. A., & Song, W. Z. (2019). A review on smart textiles development and applications. Journal of Textile and Apparel, Technology and Management, 11(3)

12. Jafari, R., & Ebrahimi, M. (2019). A review on smart protective clothing for industrial applications. Journal of Intelligent Material Systems and Structures, 30(2), 153-178

13. Sarker, M. M. R., & Aziz, M. A. (2020). Smart clothing and wearable sensors in healthcare. Journal of Applied Science and Engineering Innovation, 7(1), 1-9.

14. Asaadi, M., & Montazer, M. (2020). A review on smart textiles in sportswear: Materials, technologies, and applications. Journal of Industrial Textiles, 50(1), 3-29.

15. Sarker, M. M. R., & Aziz, M. A. (2020). Smart clothing and wearable sensors in healthcare. Journal of Applied Science and Engineering Innovation, 7(1), 1-9.

16. Yordanova, N., & Jeong, J. S. (2021). A review of smart textiles and their applications in the automotive industry. International Journal of Automotive-Technology,22(3),1043-105

KARPAGAM
INSTITUTE OF TECHNOLOGY
Inspiring Innovation
Approved by AICTE and Affiliated to Anna University
Accredited by NBA (CSE, IT, ECE)
(An ISO 9001:2015 and ISO 14001:2015 Certified Institution)

9th International Conference on Latest Trends in Science, Engineering and Technology
May 05-06, 2023

CERTIFICATE OF PARTICIPATION

This is to certify that ~~Mr.~~/ Ms./ ~~Dr.~~ ATHULYA A S of KARPAGAM INSTITUTE OF TECHNOLOGY has participated and presented a paper titled DESIGN AND DEVELOPMENT OF A PROTECTION SYSTEM FOR RESCUE TEAM in the ISTE Sponsored 9th International Conference on Latest Trends in Science, Engineering and Technology (ICLTSET'23) organized by Karpagam Institute of Technology, Coimbatore held on May 05 - 06, 2023.

Dr. S. GOPINATH
Convener

Dr. D. BHANU
Vice Principal

Dr. P. MANIMARAN
Principal

KARPAGAM
INSTITUTE OF TECHNOLOGY
Inspiring Innovation
Approved by AICTE and Affiliated to Anna University
Accredited by NBA (CSE, IT, ECE)
(An ISO 9001:2015 and ISO 14001:2015 Certified Institution)

9th International Conference on Latest Trends in Science, Engineering and Technology
May 05-06, 2023

CERTIFICATE OF PARTICIPATION

This is to certify that Mr./ ~~Ms.~~/ ~~Dr.~~ MITHUN KUMAR N of KARPAGAM INSTITUTE OF TECHNOLOGY has participated and presented a paper titled DESIGN AND DEVELOPMENT OF A PROTECTION SYSTEM FOR RESCUE TEAM in the ISTE Sponsored 9th International Conference on Latest Trends in Science, Engineering and Technology (ICLTSET'23) organized by Karpagam Institute of Technology, Coimbatore held on May 05 - 06, 2023.

Dr. S. GOPINATH
Convener

Dr. D. BHANU
Vice Principal

Dr. P. MANIMARAN
Principal

KARPAGAM INSTITUTE OF TECHNOLOGY
Inspiring Innovation
Approved by AICTE and Affiliated to Anna University
Accredited by NBA (CSE, IT, ECE)
(An ISO 9001:2015 and ISO 14001:2015 Certified Institution)

9th International Conference on Latest Trends in Science, Engineering and Technology
May 05-06, 2023

CERTIFICATE OF PARTICIPATION

This is to certify that Mr./ ~~Ms.~~/ ~~Dr.~~ SUNDAR RAJ A of KARPAGAM INSTITUTE OF TECHNOLOGY has participated and presented a paper titled DESIGN AND DEVELOPMENT OF A PROTECTION SYSTEM FOR RESCUE TEAM in the ISTE Sponsored 9th International Conference on Latest Trends in Science, Engineering and Technology (ICLTSET'23) organized by Karpagam Institute of Technology, Coimbatore held on May 05 - 06, 2023.

Dr. S. GOPINATH	Dr. D. BHANU	Dr. P. MANIMARAN
Convener	Vice Principal	Principal

KARPAGAM INSTITUTE OF TECHNOLOGY
Inspiring Innovation
Approved by AICTE and Affiliated to Anna University
Accredited by NBA (CSE, IT, ECE)
(An ISO 9001:2015 and ISO 14001:2015 Certified Institution)

9th International Conference on Latest Trends in Science, Engineering and Technology
May 05-06, 2023

CERTIFICATE OF PARTICIPATION

This is to certify that ~~Mr.~~/ Ms./ ~~Dr.~~ YASSHIKAA J of KARPAGAM INSTITUTE OF TECHNOLOGY has participated and presented a paper titled DESIGN AND DEVELOPMENT OF A PROTECTION SYSTEM FOR RESCUE TEAM in the ISTE Sponsored 9th International Conference on Latest Trends in Science, Engineering and Technology (ICLTSET'23) organized by Karpagam Institute of Technology, Coimbatore held on May 05 - 06, 2023.

Dr. S. GOPINATH	Dr. D. BHANU	Dr. P. MANIMARAN
Convener	Vice Principal	Principal