THE ROBOTICS CLUB

Integrating Knowledge...

CHAPTER 1

1.1 INTRODUCTION:

Floods can be very dangerous and often happen without much warning. Many people don't get enough time to escape, which can put their lives at risk. To solve this problem, the Flood Rescue Smart pod was designed. This pod is a smart safety shelter that helps protect people during floods. It uses sensors to watch for signs of flooding like rising water, heavy rain, or ground movement. When danger is detected, it automatically takes action to keep people inside safe — like turning on lights, sounding alarms, or sealing the shelter. The smart pod is strong, safe, and works on its own without needing someone to control it. It gives people a safe place to stay until the flood passes or help arrives. This system is a smart way to save lives and reduce damage during floods, especially in areas where flooding happens often.

1.2 PURPOSE OF THE PROJECT:

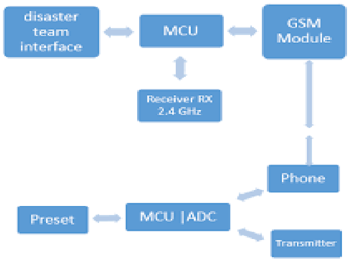
Floods can happen suddenly and cause a lot of danger. People may not always have time to run to safety. That's why the Flood Rescue Smart Bunker System was made to help keep people safe during floods. This pod is like a safe room or shelter that can protect people when water levels rise. It uses special sensors to check for danger like heavy rain or rising water. If it senses a flood is coming, it can automatically take action like closing doors, turning on lights or alarms, and making sure the people inside are safe. It doesn't need anyone to control it at the time it works on its own. The smart pod helps people stay safe until rescue teams come. It also helps rescuers find where people are during a flood. The main aim of this system is to save lives, reduce fear, and give people a safe place to wait during floods. It's a smart way to stay prepared and protected in areas where floods happen often.

1.3 EXISTING SYSTEMS:

STATIM Pod (USA) – Flood Survival Capsule

The STATIM Pod is an emergency flood survival shelter developed by Brahman Industries in the United States. STATIM stands for Storm, Tornado, And Tsunami Interconnected Module, and it was designed as a life-saving alternative when people are unable to evacuate from disasters like flash floods or tsunamis. The pod floats on water like a lifeboat, keeping occupants above rising floodwaters. If it tips over in water, it can automatically flip itself upright to maintain stability and safety. Built with marine-grade materials to withstand strong water currents, impacts, and severe weather.

1.4 BLOCK DIAGRAM OF EXISTING SYSTEM:



1.5 LIMITATIONS OF EXISTING SYSTEMS:

1. No Smart Technology

The STATIM Pod is largely mechanical and lacks automation or sensors (like water level, rain detection, or alerts).

It doesn't offer early warnings or communication with emergency teams.

2. No Internal Monitoring

It does not monitor air quality, temperature, or internal conditions — important for long stays.

3. High Cost & Size

Large and expensive to install, especially for small homes or public use in developing countries.

4. Limited Customization

Designed for general survival, but not specifically for flood monitoring, alerts, or adaptive response.

5. No Anchoring System

If water currents are strong, it may drift or get damaged if not properly secured.

1.6 PROJECT ARCHITECTURE:

The Flood Rescue Smart pod is System an emergency shelter equipped with sensors and smart technology to detect and respond to flood conditions. Key sensors like water level, rain, temperature, humidity, to monitor environment. Data is processed by a microcontroller (Arduino/Raspberry Pi), which triggers alarms, opens the bunker door. Once occupants enter, the system seals the door, turns on lights, ventilation, and provides access to emergency supplies. Power is backed by batteries ` to ensure continuous operation. The waterproof, insulated bunker keeps occupants safe and shares its location with rescue teams for quick response.

CHAPTER 2

LITERATURE REVIEW

2.1 LITERATURE SURVEY

Floods are among the most devastating natural disasters, causing significant loss of life, damage to infrastructure, and displacement of communities. Traditional flood management methods, such as early warning systems, evacuation plans, and manual rescue operations, often face limitations due to delayed response times, lack of real-time data, and heavy reliance on human intervention. To overcome these challenges, recent research has focused on integrating technology into flood detection and response systems. Various studies have proposed the use of sensors like water level detectors, combined with microcontrollers such as Arduino, to monitor rising water levels. These author systems automatically notify users via SMS when water reaches critical levels. However, while these models improve monitoring and alerting, they do not provide direct protection or shelter for individuals. The concept of a Flood Rescue Smart pod extends beyond monitoring, offering an innovative solution that combines detection, communication, and immediate physical protection. Unlike existing systems, the smart pod includes an automated shelter that activates upon sensing flood conditions, provides temporary refuge, and ensures occupant safety with life-support systems, power backup. This project addresses a critical gap in existing research and systems by providing not just warning, but actionable, self-contained safety during flood emergencies.

CHAPTER 3
PROPOSED SYSTEM

3.1 PROPOSED SYSTEM:

The Flood Rescue Smart pod is an advanced, technology-driven emergency solution designed to protect human lives during flood disasters. Unlike traditional systems that focus solely on issuing warnings, this smart pod combines real-time flood detection, automated protective mechanisms, and communication capabilities to ensure immediate safety and survival. It operates through a network of sensors—including water level, rain, temperature, humidity, and pressure sensor, gyroscope. which continuously monitor environmental conditions. These sensors are connected to a microcontroller (Arduino) that acts as the control unit.

The smart pod itself is designed using waterproof, insulated materials and is equipped with a motorized door that seals shut after entry to prevent water ingress. Inside, it contains essential survival components such as LED lighting, a ventilation or oxygen supply system, and emergency supplies like food, water, and medical kits. The system also includes a reliable power backup supported by rechargeable batteries to maintain operation during blackouts. This autonomous shelter requires minimal human intervention, making it especially effective in sudden or nighttime flooding events when evacuation may not be possible. By integrating detection, communication, and automated protection, the proposed system offers a comprehensive solution that not only alerts but actively saves lives during floods.

3.2 ADVANTAGES OF PROPOSED SYSTEM:

Flood Rescue Smart pod provides several advantages that make it a highly effective and reliable .The solution during flood emergencies. One of the main benefits is its ability to operate automatically without human intervention. The system continuously monitors environmental conditions such as rising water levels and heavy rainfall through sensors, and when it detects danger, it instantly activates alarms, opens the bunker, and sends alerts to emergency contacts. This immediate and automated response is crucial in sudden or nighttime flooding when people may not have time to evacuate. It is designed to be waterproof, insulated, and sealed, providing a secure space for individuals to take refuge during a flood. The interior includes ventilation, lighting, and emergency supplies such as food, water, and first-aid kits, ensuring basic survival needs are met and system includes a reliable power backup supported by rechargeable batteries , which allows it to function even during power outages a common problem during floods. The system is also compact and can be installed in homes, schools, or community centers in flood-prone areas. By combining real-time detection, communication, and life-saving shelter into a single system, the Flood Rescue Smart pod significantly increases safety, reduces panic, and enhances preparedness for flood disasters.

3.3 HARDWARE REQUIREMENTS:

3.3.1Bread Board:

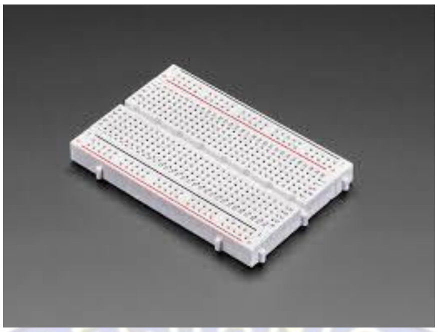


Fig. No. 3.3.1.1. Bread board

A breadboard, solderless breadboard, or protoboard is a construction base used to build semi-permanent prototypes of electronic circuits. Unlike a perfboard or stripboard, breadboards do not require soldering or destruction of tracks and are hence reusable. For this reason, breadboards are also popular with students and in technological education.

3.3.2 Lead Acid battery:

Fig.No.3.3.2.1. Lead Acid battery

The lead–acid battery is a type of rechargeable. It is the first type of rechargeable battery ever created. Compared to modern rechargeable batteries, lead–acid batteries have relatively low energy density. Despite this, they are able to supply high surge currents. These features, along with their low cost, make them attractive for use in motor vehicles to provide the high current required by starter motors. Lead–acid batteries suffer from relatively short cycle lifespan and overall lifespan (due to the double sulfation in the discharged state), as well as long charging times.

3.3.3 DHT11 Sensor

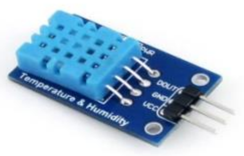


Fig.No.3.3.3.1 DHT11

DHT11 is a low-cost digital temperature and humidity sensor that comes equipped with a built-in capacitive humidity sensor, a temperature thermistor, and a microcontroller to digitize the readings.

Specifications:

Temperature Range: 0°C to 50°C

Humidity Range: 20% to 90% RH

Operating Voltage: 3.3V to 5V

Output: Digital signal (1 reading/second)

1. Humidity Measurement: A capacitive humidity sensing element is used in the sensor. It consists of two electrodes separated by a polymer layer. With a change in surrounding humidity, the dielectric constant of the polymer alters, which changes the capacitance. This variation is translated into a humidity value by the internal chip.

2. Temperature Measurement: A thermistor, which is a resistor that varies in resistance with temperature, is utilized in the sensor. Since the temperature rises or falls, the resistance also changes, and this is converted to a temperature value by the microcontroller within.

3.3.4 Pressure Sensor:

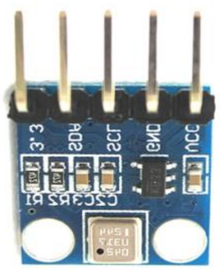


Fig. No. 3.3.4.1Pressure sensor

The BMP180 is a small electronic sensor used to measure air pressure, temperature, and even altitude. It is commonly used in weather stations, smartphones, and projects like drones and GPS systems. The sensor works by detecting changes in air pressure using a tiny chip inside it. These changes can help tell how high or low something is above sea level. The BMP180 connects to microcontrollers like Arduino using I2C wires (just two data wires), making it easy to use. It can measure air pressure very accurately and can even detect small changes in height, like going up or down stairs. It also has a temperature sensor to help give better readings. It runs on low power (just 3.3V) and is very small in size, which makes it perfect for portable or battery-powered devices. People use it in projects to track altitude, or understand the environment better.

3.3.5 Water level sensor:

Fig. No. 3.3.5.1. Water level sensor

The SEN18 sensor operates by detecting changes in resistance as water levels vary. It features a series of exposed parallel conductive traces. When water comes into contact with these traces, it alters the resistance, which is then converted into an analog voltage. This voltage can be read by an Arduino to determine the water level .

Pinout and Connections:

VCC: Connects to 3.3V or 5V on the Arduino

GND: Connects to ground.

S (Signal): Analog output pin connected to an analog input on the Arduino

3.3.6 R385 Diaphragm Pump:

Fig. No. 3.3.6.1 R385 Diaphragm Pump

A small DC water pump that can move water efficiently. Automatically activated during flood detection to pump water out of the bunker or affected area, reducing water accumulation. Since the pump draws more current than what the Arduino can supply, a relay module is used to switch it on/off safely based on sensor input.

3.3.7 Arduino:

Fig. No. 3.3.7.1. Arduino

Arduino UNO is a microcontroller board based on the **ATmega328P**. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

3.3.8 MPU6050:

Fig. No. 3.3.8.1 MPU6050

It is also known as gyroscope where it is used as an 6-axis motion tracking sensor. The gyroscope measures the angular velocity or rate of rotation around x, y, z axes and we use four main pins of those are VCC , GND ,SDA and SCL .where VCC and GND pins are used for power supply and SDA , SCL are I2C pins where SDA transfers the data and the SCL pin helps to send the values according to the given delay and the reason to integrate this in our project is to know the pod body is stable or tilted .

3.3. 9 Inflatable air bag:

Fig. No. 3.3.9.1 Inflatable air bag

An inflatable airbag is a flexible membrane that inflates using air or gas under pressure. It can serve as a safety mechanism, lifting aid, or cushioning system in robots, automation, or rescue equipment.

3.3.10 Adafruit PCA9548 I2C Multiplexer:

Fig. No. 3.3.10.1. Adafruit PCA9548 I2C Multiplexer

The Adafruit PCA9548 I2C Multiplexer Breakout is a handy board that solves a common I2C problem: connecting multiple identical I2C devices (which share the same address) to a single microcontroller. It uses the PCA9548A chip to act as an I2C switch, allowing you to select one of eight downstream I2C channels at a time. This means you can connect, for instance, eight of the same temperature sensors to your Arduino or Raspberry Pi, even though they all have the same I2C address. We communicate with the multiplexer itself via its own I2C address (typically 0x70). By writing a simple byte to it, you can enable the desired channel, then seamlessly communicate with the device connected to that specific channel. The board usually supports both 3.3V and 5V logic, making it versatile for various projects. It's an essential tool for expanding your I2C bus when address conflicts arise.

3.3.11 Relay:

Fig. No. 3.3.11.1. Relay

Relay module is an electromechanical switch used to control high-voltage devices using a low-voltage signal, like those from a microcontroller (e.g., Arduino, Raspberry Pi, ESP32, etc.). It acts as a bridge between the low-power control circuit and the high-power load.

3.3. 12 Buzzer:

Fig. No. 3.3.12.1. Buzzer

A buzzer is a device that makes a buzzing or beeping sound when electricity flows through it. It's like a tiny speaker that converts electrical signals into audible sounds. These sounds are often used to alert someone or signal something is happening, like in alarms, computers, or printers. A buzzer usually has a coil of wire that creates a magnetic field when electricity flows through it. This magnetic field then interacts with a metal diaphragm or a piezoelectric crystal, causing it to vibrate rapidly and produce sound.

3.3.13 LED:

Fig. No. 3.3.13.1 LED

A light-emitting diode (LED) is a semiconductor device that emits light when an electric current flows through it. It's a two-lead semiconductor light source that works by using a PN junction where electrons and holes recombine, releasing energy as light (photons). LEDs are energy-efficient and have a wide range of applications, from simple indicators to large-scale lighting. Led s are essentially a PN junction diode, where a layer of n-type semiconductor material (with excess electrons) is combined with a layer of p-type semiconductor material

3.3.14 DIODE :

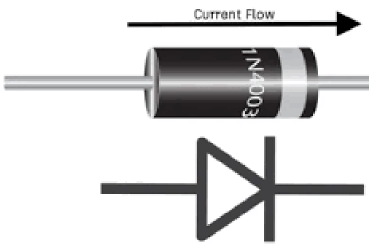


Fig. No. 3.3.14.1 DIODE

A diode is a two-terminal semiconductor device that primarily allows current to flow in one direction. It acts like a one-way switch, facilitating current flow in the forward direction while restricting it in the reverse direction. Diodes are widely used in electronic circuits for rectification (converting AC to DC), voltage regulation, and protection against overvoltage.

3.3.15 Resistor:

Fig. No. 3.3.15.1 Resistor

A resistor is an electrical component that limits or regulates the flow of electrical current in an electronic circuit. Resistors can also be used to provide a specific voltage for an active device such as a transistor All other factors being equal, in a direct-current (dc) circuit, the current through a resistor is inversely proportional to its resistance and directly proportional to the voltage across it. This is the well-known ohms law. In alternating-current (AC) circuits, this rule also applies as long as the resistor does not contain inductance or capacitance

3.3.16 Jumper wires:

Fig. No.3.3.16.1 Jumper wires

Jumper wires are small wires used to connect different parts of a circuit without soldering. They are commonly used in electronics projects with breadboards and microcontrollers like Arduino.

Types of Jumper Wires:

1. Male to Male – Pin on both ends (used with breadboards and Arduino).

2. Male to Female – Pin on one end, socket on the other.

3. Female to Female – Socket on both ends.

3.4 SOFTWARE REQIREMENTS

3.4.1 Arduino IDE:

The Arduino integrated development environment (IDE) is a cross-platform application (for Microsoft Windows, macOS, and Linux) that is based on Processing IDE which is written in Java. It uses the Wiring API as programming style and HAL. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and syntax highlighting, and provides simple one-click mechanisms to compile and upload programs to an Arduino board.

Fig. No. 3.4.1 Arduino IDE

3.4.2 Fusion 360:

Autodesk Fusion is a commercial computer-aided design (CAD), computer-aided manufacturing (CAM), computer-aided engineering (CAE) and printed circuit board (PCB) design software application, developed by Autodesk. It is available for Windows, macOS and web browsers, with simplified applications available for Android and iOS.

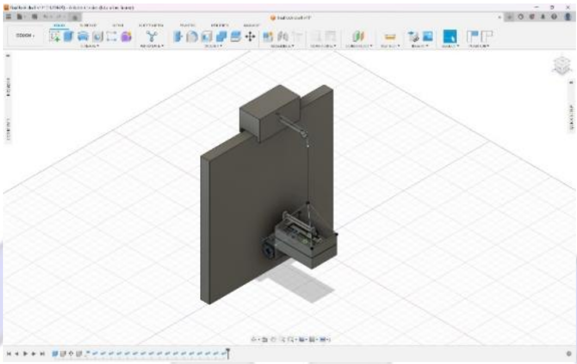


Fig.No.3.4.2 Fusion 360

3.4.3 Fritzing:

Fritzing is an open-source initiative to develop amateur or hobby CAD software for the design of electronics hardware, intended to allow designers and artists to build more permanent circuits from prototypes. It was developed at the University of Applied Sciences Potsdam.

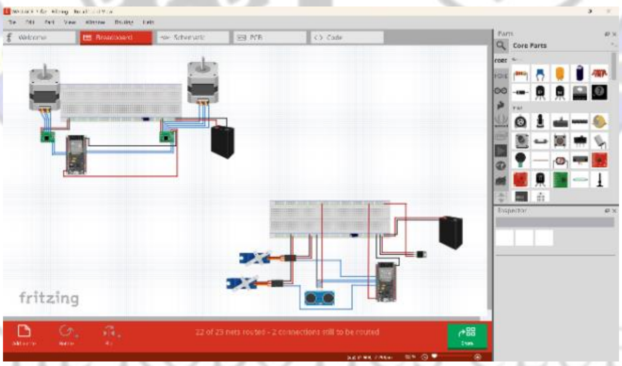


Fig.No.3.4.3 Fritzing

CHAPTER 4

WORKING OF PROPOSED SYSTEM

4.1 Block diagram

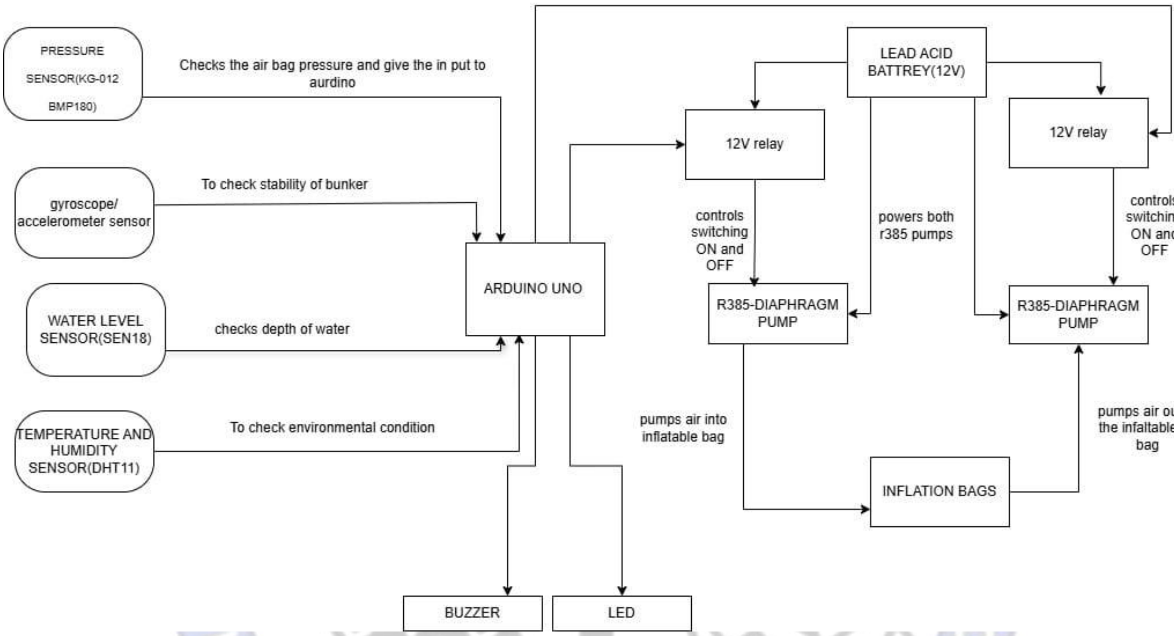


Fig.4.1.1 Block Diagram

4.2 Working

The flood rescue smart pod is a specially designed safety system created to protect people during floods. It works using various electronic components and sensors that detect dangerous conditions and respond automatically to ensure safety. The Arduino UNO microcontroller, It acts like the brain of the system. If water levels start rising, the water level sensor checks how high the water has reached. The pressure sensor (BMP180) monitors the air and water pressure to understand the severity of flooding ,the DHT11 temperature and humidity sensor monitors the climate inside and outside the pod to ensure it remains safe for people.. if the flooding becomes too severe, the system triggers the pump, which allows air to fill the inflatable airbag. This airbag helps lift the bunker above the water, keeping it afloat and safe. To help people locate the bunker during emergencies, the system uses LED lights and a buzzer for alerts. The MPU6050 gyroscope/accelerometer helps detect the movement or tilt of the Pod to maintain its balance. All the components are powered by a 12V lead-acid battery, which ensures the system works even during power outages.

The smart pod operates automatically, without needing manual control, and ensures that people inside remain safe and protected during floods.

4.3 CIRCUIT DIAGRAM

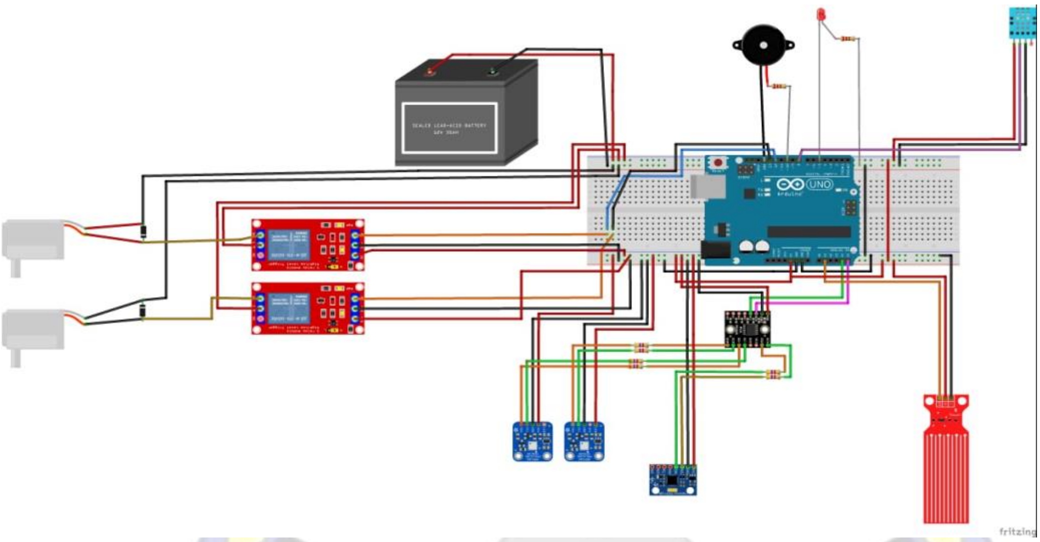


Fig. 4.3.1 Circuit diagram

4.4 CAD

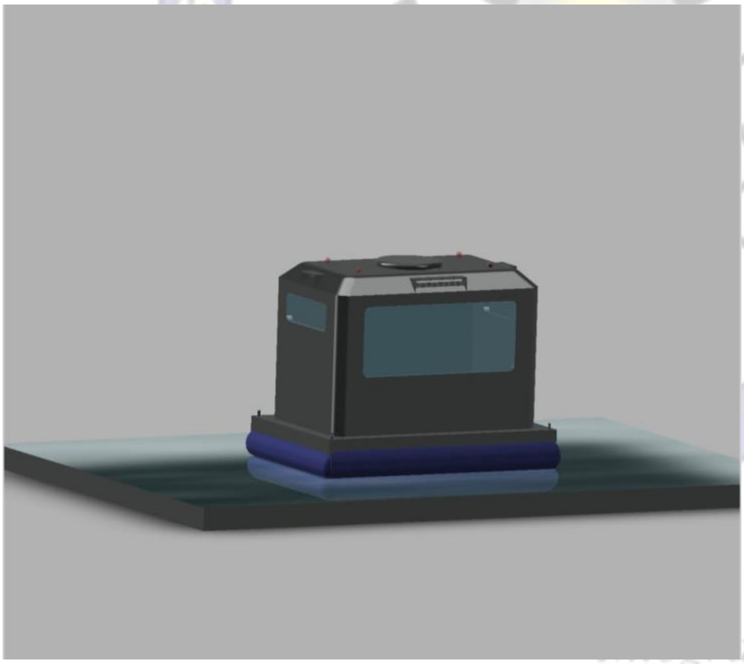


Fig.No.4.4.1 CAD

CHAPTER 5

RESULTS

The Flood Rescue Smart pod has proven to be an effective solution for enhancing safety during floods. It provides a secure and reliable shelter for individuals in flood-prone areas, helping to reduce the risk of injuries and fatalities. Equipped with sensors that detect rising water levels in real time, the system activates automatically to offer protection before the situation worsens. Some advanced models include features like automatic floating capabilities, and emergency communication systems, which enable users to alert rescue teams or share their location during disasters. Additionally, the pod is designed to operate on backup power or solar energy, ensuring continuous functionality even during electricity outages. Overall, the system significantly improves survival chances and supports timely rescue operations.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 CONCLUSION

Flood-Rescue Smart Pod System is a smart and practical solution that helps save lives during sudden urban floods. It uses modern technology like sensors, pumps, airbags, and a microcontroller Arduino UNO to detect danger early and respond automatically. When floodwaters rise, the system quickly inflates airbags to lift the bunker above the water, helping keep people safe and dry. It also has an anchor system to keep the pod stable and prevent it from being carried away by strong currents or wind. At the same time, lights and a buzzer turn on to alert rescue teams and guide them to the location. What makes this system special is that it works by itself, without needing anyone to operate it during an emergency.

6.2 FUTURE SCOPE

Flood Rescue Smart pod is vast and promising, especially as climate change increases the frequency and severity of floods worldwide. In the coming years, the system can be enhanced with more advanced technologies such as AI-based flood prediction, IoT integration for real-time remote monitoring, and automated alerts to authorities and family members. This could also be designed with modular structures to accommodate more people or to be relocated easily based on changing flood risk zones. Integration with smart city infrastructure and government disaster management systems can further boost its effectiveness in saving lives. With ongoing research and innovation, the Smart Pod could become a standard safety feature in flood-prone regions, playing a critical role in future disaster resilience and preparedness.

CHAPTER 7

REFERENCES

1. IoT Based Flood Monitoring and Alerting System: https://www.boltiot.com/projects/iot-based-flood-monitoring-and-alerting-system

2. AEM's monitoring systems: https://aem.eco/solution/flood-risk-management/

3. Flood Monitoring and Warning System: Het-Sens a Proposed Model from IEEE: https://ieeexplore.ieee.org/document/9257693

4. Nevon Projects: https://nevonprojects.com

5. SIH: https://sih.gov.in

SOURCE CODE

```
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_MPU6050.h>
#include <Adafruit_BMP085.h>
#include <DHT.h>


#define DHTPIN A2
#define DHTTYPE DHT11
#define MAX_P 102000        //max pressure
#define MIN_P 98000         //min pressure
#define WATER_SENSOR_PIN A1
#define WATER_LEVEL_THRESHOLD 250   //water threshold

// Relay Pins
#define RELAY_MOTOR_PUMP_1_PIN 3
#define RELAY_MOTOR_PUMP_2_PIN 4

// Buzzer Pin
#define BUZZER_PIN 5

// LED Pins
#define LED_PIN_1 6
#define LED_PIN_2 7
#define LED_PIN_3 8
#define LED_PIN_4 9
#define DANGER_GYRO_THRESHOLD 20   //gyro safety threshold

// I2C Multiplexer address
#define TCAADDR 0x70
```

```
DHT dht(DHTPIN, DHTTYPE);
Adafruit_MPU6050 mpu;
Adafruit_BMP085 bmp1;
Adafruit_BMP085 bmp2;

void tcaSelect(uint8_t channel) {
if (channel > 7) return;
  Wire.beginTransmission(TCAADDR);
  Wire.write(1 << channel);
  Wire.endTransmission();
}
void setup() {
 Serial.begin(9600);
 Wire.begin();
 pinMode(RELAY_MOTOR_PUMP_1_PIN, OUTPUT);
 digitalWrite(RELAY_MOTOR_PUMP_1_PIN, LOW);
 pinMode(RELAY_MOTOR_PUMP_2_PIN, OUTPUT);
 digitalWrite(RELAY_MOTOR_PUMP_2_PIN, LOW);
 pinMode(BUZZER_PIN, OUTPUT);
 digitalWrite(BUZZER_PIN, LOW);

 // LED setup
  pinMode(LED_PIN_1, OUTPUT);
  digitalWrite(LED_PIN_1, LOW);
  pinMode(LED_PIN_2, OUTPUT);
  digitalWrite(LED_PIN_2, LOW);
  pinMode(LED_PIN_3, OUTPUT);
  digitalWrite(LED_PIN_3, LOW);
  pinMode(LED_PIN_4, OUTPUT);
  digitalWrite(LED_PIN_4, LOW);
```

```
dht.begin();
tcaSelect(0);
  if (!bmp1.begin()) {
    Serial.println("Error: BMP180 (Channel 0) not found!");
  } else {
    Serial.println("BMP180 (Channel 0) found.");
  }

  tcaSelect(1);
  if (!mpu.begin()) {
    Serial.println("Error: MPU6050 (Channel 1) not found!");
  } else {
    Serial.println("MPU6050 (Channel 1) found.");
  }

  tcaSelect(2);
  if (!bmp2.begin())
 {
    Serial.println("Error: BMP180 (Channel 2) not found!");
  } else {
    Serial.println("BMP180 (Channel 2) found.");
  }
  Serial.println("--- Setup Complete ---");
}

void loop() {
  float temp = dht.readTemperature();
  float hum = dht.readHumidity();
 int waterValue = analogRead(WATER_SENSOR_PIN);
  bool waterDetected = waterValue > WATER_LEVEL_THRESHOLD;
```

```
 bool waterLow = waterValue <= WATER_LEVEL_THRESHOLD;
 tcaSelect(0);
 float pressure1 = bmp1.readPressure();
 float temperature1_bmp = bmp1.readTemperature();
 tcaSelect(2);
 float pressure2 = bmp2.readPressure();
 float temperature2_bmp = bmp2.readTemperature();
 tcaSelect(1);
 sensors_event_t a, g, temp_mpu;
 mpu.getEvent(&a, &g, &temp_mpu);

 // Control Motor Pump 1 (FILLING)
 if (waterDetected) {
   digitalWrite(RELAY_MOTOR_PUMP_1_PIN, HIGH);
   Serial.println("Water level HIGH. Motor Pump 1 ON (FILLING).");
 } else if (pressure1 >= MAX_P || pressure2 >= MAX_P) {
   digitalWrite(RELAY_MOTOR_PUMP_1_PIN, LOW);
   Serial.println("Air bag filled. Motor Pump 1 OFF.");
 }

// Control Motor Pump 2 (DRAINING)
 if (waterLow) {
   digitalWrite(RELAY_MOTOR_PUMP_2_PIN, HIGH);
   Serial.println("Water level LOW. Motor Pump 2 ON (DRAINING).");
 } else if (pressure1 <= MIN_P || pressure2 <= MIN_P) {
   digitalWrite(RELAY_MOTOR_PUMP_2_PIN, LOW);
   Serial.println("Deflation successful. Motor Pump 2 OFF.");
 }

if(digitalRead(RELAY_MOTOR_PUMP_1_PIN)==HIGH)||digitalRead(RELAY_MOTOR_P
UMP_2_PIN) == HIGH) {
```

```
    digitalWrite(BUZZER_PIN, HIGH);
    Serial.println("Buzzer ON: A motor pump is active.");
  }
else
{
    digitalWrite(BUZZER_PIN, LOW);
  }

 if (waterDetected) {
    digitalWrite(LED_PIN_1, HIGH);
    digitalWrite(LED_PIN_2, HIGH);
    digitalWrite(LED_PIN_3, HIGH);
    digitalWrite(LED_PIN_4, HIGH);
    Serial.println("LEDs ON: Water level is high.");
  }
else
{
    digitalWrite(LED_PIN_1, LOW);
    digitalWrite(LED_PIN_2, LOW);
    digitalWrite(LED_PIN_3, LOW);
    digitalWrite(LED_PIN_4, LOW);
    Serial.println("LEDs OFF: Water level is not high.");
  }
  Serial.print("Water Level: ");
  Serial.print(waterValue);
  Serial.print(" (Threshold: ");
  Serial.print(WATER_LEVEL_THRESHOLD);
  Serial.print(") - ");
  Serial.print(waterDetected ? "Water HIGH" : "Water LOW/OK");
  Serial.print(" - ");
```

```
 Serial.print("DHT Temp: ");
 Serial.print(temp);
 Serial.print(" C, DHT Humidity: ");
 Serial.print(hum);
 Serial.print(" % - ");
Serial.print("Gyro (rad/s) X: ");
 Serial.print(g.gyro.x);
 Serial.print(", Y: ");
 Serial.print(g.gyro.y);
 Serial.print(", Z: ");
 Serial.print(g.gyro.z);
 Serial.println();
if (abs(pressure1 - pressure2) >= 5) {
   Serial.println("Unequal pressure in airbags!");
  }
 else
{
   Serial.println("Pressure Balanced");
  }
if(abs(g.gyro.x)>DANGER_GYRO_THRESHOLD||abs(g.gyro.y)>DANGER_GYRO_THR
ESHOLD || abs(g.gyro.z) > DANGER_GYRO_THRESHOLD)
{
   Serial.println("UNSTABLE / DANGEROUS TILT");
  }
 Else
 {
   Serial.println("Tilt is Stable");
  }
 Serial.println("---");
 delay(5000);
}
```

THE ROBOTICS CLUB

Integrating Knowledge...

THE ROBOTICS CLUB

Integrating Knowledge...