IoT-Based Flood Monitoring and Alert System Using ESP32

Team Name: KARROT

Team Member Details:

- **Manjunath D**
  - Contact: 8884563215
  - Branch: ECE
  - Year: 2029
  - College: UVCE (University of Visvesvaraya College of Engineering)
- **Abhishek**
  - Contact: 9481509353
  - Branch: ECE
  - Year: 2029
  - College: UVCE (University of Visvesvaraya College of Engineering)

1. Aim

Floods are one of the most destructive natural disasters, causing significant loss of life and property every year. Traditional flood monitoring methods often lack real-time updates and early warning systems. To overcome these limitations, this project presents an IoT-based Flood Monitoring and Alert System designed using an ESP32 microcontroller, which continuously monitors water levels and provides instant alerts in case of flood risk.

The system uses an ultrasonic sensor to measure water level and indicates the safety status through LEDs, an LCD display, and a buzzer alarm. It also integrates a servo motor for potential automatic control of floodgates. This project offers a cost-effective, scalable, and reliable solution for early flood detection and prevention.

Objectives

- To design and implement an automated system for real-time flood level monitoring.
- To alert nearby residents or authorities about potential flood risks using visual and audio indicators.
- To utilize IoT capabilities for possible remote monitoring and data logging.
- To build a low-cost, energy-efficient, and easy-to-deploy system suitable for rural and urban flood-prone areas.

2. Procedure

2.1. Working Principle / Theory

The ultrasonic sensor continuously measures the distance between itself and the water surface. The ESP32 processes this distance data to determine the water level.

- When the water level is below a safe limit, the green LED glows, and the LCD displays "Status: SAFE – No Flood Risk."
- When the level rises to a warning range, the yellow LED glows, and the LCD shows "Status: WARNING – Water Rising."
- When the level crosses the danger limit, the red LED glows, the buzzer sounds an alarm, and the LCD displays "Status: DANGER – Flood Risk!"
- The servo motor can be programmed to close a barrier or open a drainage channel automatically during the danger condition.

Optionally, the ESP32's Wi-Fi capability allows the system to upload data to an IoT platform or send SMS/email alerts for remote monitoring.

2.2. Components Used

Component	Description
ESP32 Microcontroller	The main control unit that processes data from the ultrasonic sensor and controls all output devices.
Ultrasonic Sensor (HC-SR04)	Measures the distance between the sensor and the water surface to determine the water level.
16x2 LCD Display	Displays real-time water level and safety status (e.g., “SAFE,” “WARNING,” “DANGER”).
LED Indicators	Three LEDs represent different safety levels – Green (Safe), Yellow (Warning), Red (Danger).
Buzzer	Emits an alarm sound when the water level exceeds the danger threshold.
Servo Motor	Simulates the automatic operation of a floodgate or barrier system.
Resistors and Jumper Wires	Used for current limiting and circuit connections.

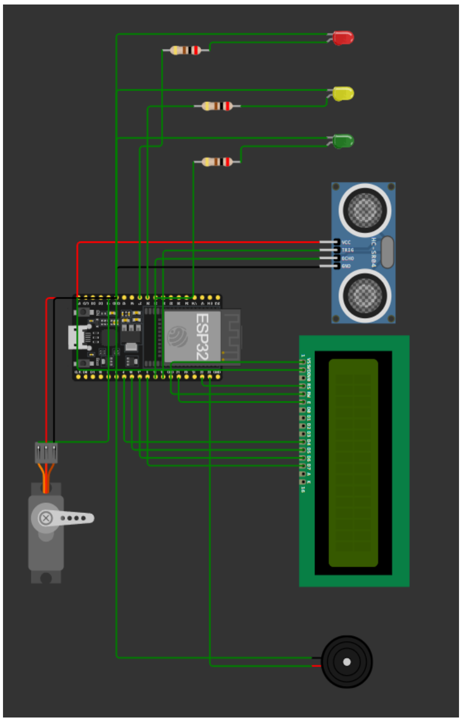
2.3. Implementation

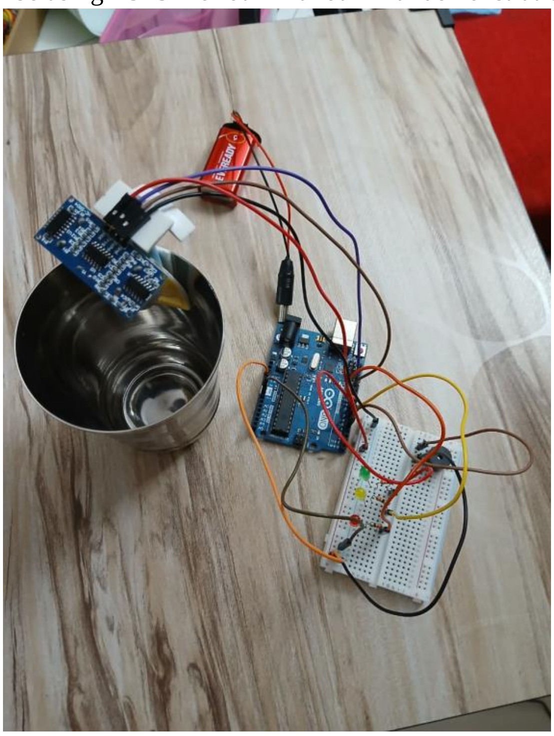
The implementation involved circuit design, software coding, and hardware testing.

- **Circuit Design:** All components were interfaced with the ESP32. The HC-SR04 sensor’s Trig and Echo pins are connected to digital I/O pins for distance measurement. The LCD display is interfaced in 4-bit mode to show real-time data. Three LEDs are connected to GPIO pins through resistors, and the buzzer is triggered by the ESP32 when the danger level is detected. The servo motor is controlled using PWM signals from the ESP32. Power is supplied via USB or a regulated 5V source.
- **Software & Testing:** The system is programmed using the Arduino IDE with the ESP32 board support package. The code includes modules for ultrasonic distance measurement, LCD display output, servo control, and conditional logic to determine the safety level and trigger corresponding outputs. Simulation and testing were done on Tinkercad and later verified on actual hardware.

2.4. Picture of the Model

(We will be using ESP32 for our final our final demonstration on 8th)





2.5. Result

The implemented system was successfully tested on hardware. The device accurately monitored the water level in real-time.

- In the **"Safe"** state, the green LED was illuminated, and the LCD displayed "Status: SAFE".
- As the water level rose to the **"Warning"** threshold, the system correctly switched, illuminating the yellow LED and updating the LCD to "Status: WARNING".
- When the water level crossed the **"Danger"** limit, the red LED and the buzzer were activated, and the LCD displayed "Status: DANGER – Flood Risk!". All visual and audio alerts functioned as designed, and the servo motor responded appropriately to simulate gate operation during the danger condition.

2.6. Future Scope

- IoT cloud integration for real-time mobile alerts.
- Solar power support for standalone operation.
- Machine learning algorithms for predictive flood analysis.
- GPS tracking for mapping and location-based alerts.

2.7. Limitations

- The accuracy of the ultrasonic sensor can be affected by external factors like heavy fog, turbulence, or floating debris.
- The system's IoT functionality relies on a stable Wi-Fi connection, which might be unavailable in remote areas or during a natural disaster.
- The current prototype is powered via USB, making it dependent on a nearby power source. For field deployment, an independent power solution (like solar) would be required.
- The servo motor in the model only simulates a floodgate; a real-world application would require interfacing with much larger, industrial-grade mechanical systems.

3. Referral

- Tinkercad, Wokwi for online simulations of the circuit
- Arduino platforms (https://docs.arduino.cc/libraries/esp32servo/)