

FLOOD MONITORING SYSTEM

PHASE – 4 : DEVELOPMENT PART-2

TEAM MEMBERS : 1. PRIYADHARSHINI

2. KADHAMBARI

3. RAJESHWARI

4. KAVIYA

5. ARCHANA

6. SHALINI

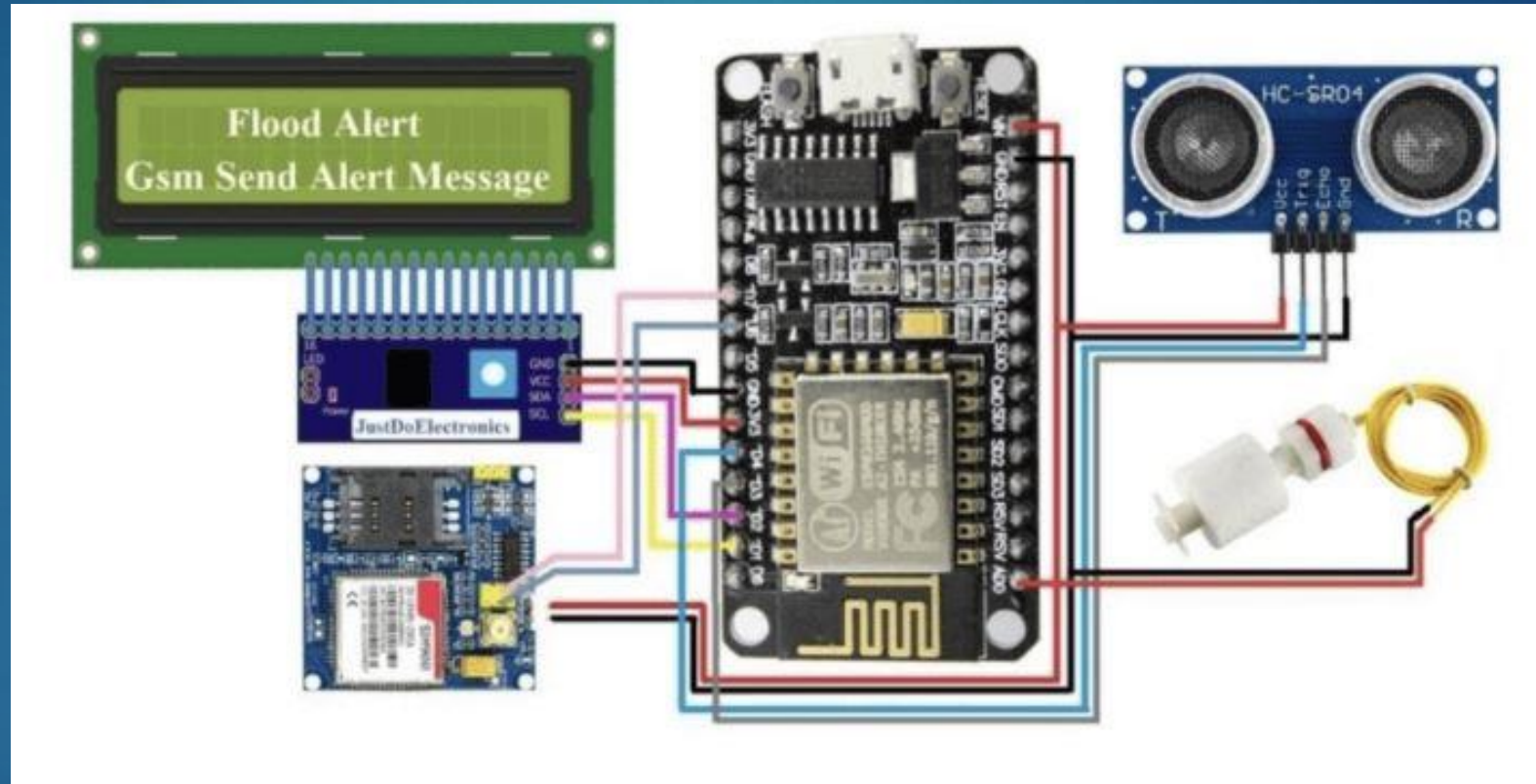
INTRODUCTION :

- ▶ we will explore the design and implementation of a Flood Monitoring System using various components such as a 16×2 LCD display with I2C, ultrasonic sensor, float sensor, and SIM800L module. This system aims to provide real-time flood level monitoring and alert notifications to receive Text Messages

COMPONENTS NEEDED:


S NO.	COMPONENTS	QUANTITY1.
1.	ESP8266	1
2.	GSM module (SIM800I)	1
3.	Ultrasonic Sensor	1
4.	Float Sensor	1
5.	16x2 LCD Display With I2C	1
6.	Zero PCB	1
7.	5v Power Supply	1

CIRCUIT DIAGRAM:




PROGRAM:

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <NewPing.h>
#include <SoftwareSerial.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
#define TRIGGER_PIN 12
#define ECHO_PIN 11
#define MAX_DISTANCE 200
NewPing sonar(TRIGGER_PIN, ECHO_PIN, MAX_DISTANCE);
#define FLOAT_SENSOR_PIN 10
SoftwareSerial gsmSerial(8, 9);
```




```
#define GSM_BAUDRATE 9600
#define FLOOD_THRESHOLD 50
String phoneNumbers[] = { "+9188305848xx", "+9188305848xx" };


void setup() {
  lcd.begin(16, 2);
  lcd.backlight();
  gsmSerial.begin(GSM_BAUDRATE);
  delay(2000);
  sendCommand("AT");
  sendCommand("AT+CMGF=1");
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Flood Monitoring");
  lcd.setCursor(0, 1);
  lcd.print("System");
  delay(3000); }
```

```
void loop() {  
    unsigned int distance = sonar.ping_cm();  
    int floatSensorValue = digitalRead(FLOAT_SENSOR_PIN);  
    int floodLevel = distance;  
    lcd.clear();  
    lcd.setCursor(0, 0);  
    lcd.print("Water Level: ");  
    lcd.print(floodLevel);  
    lcd.print("cm");  
    if (floodLevel > FLOOD_THRESHOLD && floatSensorValue == HIGH) {  
        sendAlertSMS(floodLevel);  
    }  
    delay(500);  
}
```



```
void sendAlertSMS(int floodLevel) {  
    String message = "Flood Alert! Water level is ";  
    message += floodLevel;  
    message += "cm. Take necessary actions.";  
    for (int i = 0; i < sizeof(phoneNumbers) / sizeof(phoneNumbers[0]); i++) {  
        sendCommand("AT+CMGS=\"\" + phoneNumbers[i] + \"\"");  
        delay(1000);  
        sendCommand(message);  
        delay(100);  
        sendCommand((String) char(26));  
        delay(1000);  
    }  
}
```

```
void sendCommand(String command) {  
    gsmSerial.println(command);  
    delay(1000);  
    while (gsmSerial.available()) {  
        gsmSerial.read();  
    }  
}
```

OUTPUT:



CONCLUSION:

- ▶ This system enables early detection of floods, allowing authorities to take immediate action and mitigate potential risks. By leveraging the capabilities of these components, we can build an effective flood monitoring system to enhance disaster management and safeguard lives and property.