FLOOD MONITORING AND EARLY WARNING

PHASE-3

BY:

Team leader : S.Shivani

Team member: S.Thara

Team member:S.Saranya

Team member:D.sree Sanjana

THE COMPONENTS USED FOR THIS PYTHON SCRIPT CIRCUIT :

1.LCD 16*2

- Stands for "Liquid Crystal Display." LCD is a flat panel display technology commonly used in TVs and computer monitors. It is also used in screens for mobile devices, such as laptops, tablets, and smartphones. LCD displays don't just look different than bulky CRT monitors, the way they operate is significantly different as well.

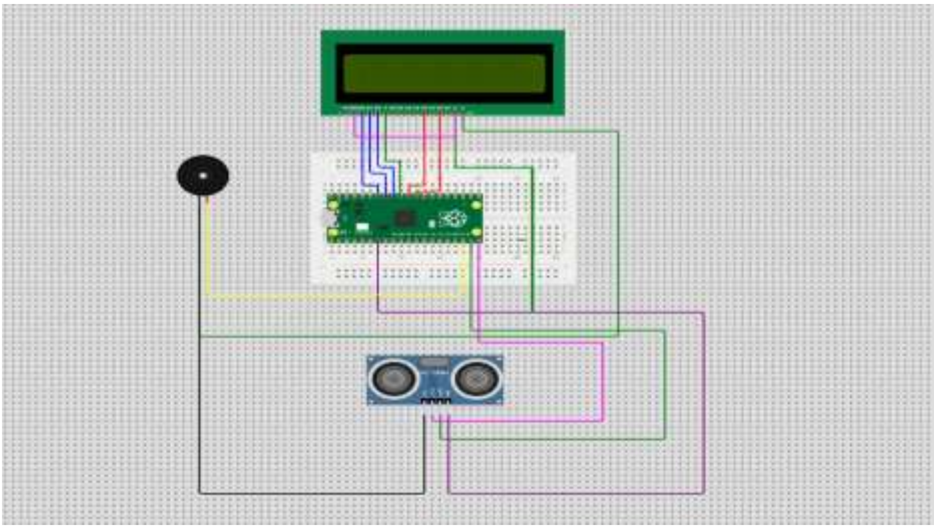
2.ULTRASONIC SENSOR

- Stands for "Liquid Crystal Display." LCD is a flat panel display technology commonly used in TVs and computer monitors. It is also used in screens for mobile devices, such as laptops, tablets, and smartphones. LCD displays don't just look different than bulky CRT monitors, the way they operate is significantly different as well.

3.BUZZER

- A buzzer is an electrical device that is used to make a buzzing sound for example, to attract someone's attention . She rang a buzzer at the information desk. Collins COBUILD Advanced Learner’s Dictionary. Copyright © HarperCollins Publishers 1. 2. 3. New Zealand Collins English Dictionary. Copyright © HarperCollins Publishers

THE CIRCUITS USING THE ABOVE COMPONENTS BY RASHBERRY



THE PYTHON SCRIPT:

```
# Import necessary libraries
import RPi.GPIO as GPIO
import time
import Adafruit_CharLCD as LCD  # You'll need to install the Adafruit_CharLCD library

# Set up Ultrasonic Sensor pins
TRIG = 23
ECHO = 24

# Set up Buzzer pin
BUZZER = 18

# Set up LCD pins
lcd_rs = 25
lcd_en = 24
lcd_d4 = 23
lcd_d5 = 17
lcd_d6 = 21
lcd_d7 = 22
lcd_columns = 16
lcd_rows = 2

# Initialize the LCD
```

```
lcd = LCD.Adafruit_CharLCD(lcd_rs, lcd_en, lcd_d4, lcd_d5, lcd_d6, lcd_d7, lcd_columns, lcd_rows)

# Initialize GPIO settings
GPIO.setmode(GPIO.BCM)
GPIO.setup(TRIG, GPIO.OUT)
GPIO.setup(ECHO, GPIO.IN)
GPIO.setup(BUZZER, GPIO.OUT)
GPIO.output(BUZZER, GPIO.LOW)  # Make sure the buzzer is initially turned off

def distance():
    # Send a pulse on the TRIG pin
    GPIO.output(TRIG, True)
    time.sleep(0.00001)
    GPIO.output(TRIG, False)

    # Measure the time it takes for the ECHO pin to go high
    while GPIO.input(ECHO) == 0:
        pulse_start = time.time()

    while GPIO.input(ECHO) == 1:
        pulse_end = time.time()

    # Calculate the distance
    pulse_duration = pulse_end - pulse_start
    distance_cm = pulse_duration * 17150

    return distance_cm


try:

    while True:

        # Read distance from Ultrasonic Sensor

        dist = distance()


        # Display the distance on the LCD

        lcd.clear()

        lcd.message("Distance: {:.1f} cm".format(dist))


        # Check if water level is above a threshold (adjust as needed)

        if dist < 10:  # Example threshold, adjust accordingly

            GPIO.output(BUZZER, GPIO.HIGH)  # Turn on the buzzer

        else:

            GPIO.output(BUZZER, GPIO.LOW)  # Turn off the buzzer


        time.sleep(1)


except KeyboardInterrupt:

    GPIO.cleanup()
```