

Components:

Ultra sonic sensor

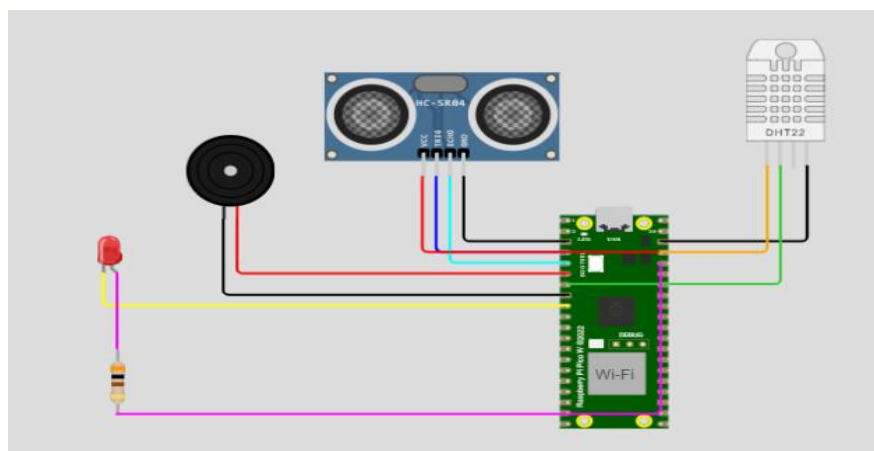
Buzzer

DHT22TEMP sensor

LED

NODE MCU

CIRCIUT DIAGRAM :



PROGRAM:

```
import time
import machine
import dht

# Define GPIO pins
TRIG_PIN = machine.Pin(2, machine.Pin.OUT)
ECHO_PIN = machine.Pin(3, machine.Pin.IN)
BUZZER_PIN = machine.Pin(4, machine.Pin.OUT)
DHT_PIN = machine.Pin(5)
LED_PIN = machine.Pin(6, machine.Pin.OUT)

def distance_measurement():
    # Trigger ultrasonic sensor
    TRIG_PIN.on()
    time.sleep_us(10)
    TRIG_PIN.off()

    # Wait for echo to be HIGH (start time)
    while not ECHO_PIN.value():
        pass
```

```

pulse_start = time.ticks_us()

# Wait for echo to be LOW (end time)
while ECHO_PIN.value():
    pass
pulse_end = time.ticks_us()

# Calculate distance
pulse_duration = time.ticks_diff(pulse_end, pulse_start)
distance = pulse_duration / 58 # Speed of sound (343 m/s) divided by 2

return distance

def read_dht_sensor():
    d = dht.DHT22(DHT_PIN)
    d.measure()
    return d.temperature(), d.humidity()

buzz_start_time = None # To track when the buzzer started

while True:
    dist = distance_measurement()
    temp, humidity = read_dht_sensor()

    # Check if the distance is less than a threshold (e.g., 50 cm)
    if dist < 50:
        # Turn on the buzzer and LED
        BUZZER_PIN.on()
        LED_PIN.on()
        status = "Flood Alert"
        buzz_start_time = time.ticks_ms()
    elif buzz_start_time is not None and time.ticks_diff(time.ticks_ms(),
buzz_start_time) >= 60000: # 1 minute
        # Turn off the buzzer and LED after 1 minute
        BUZZER_PIN.off()
        LED_PIN.off()
        status = "No Flood Alert"
    else:
        status = "No Flood Alert"

    print(f"Distance: {dist:.2f} cm")
    print(f"Temperature: {temp:.2f}°C, Humidity: {humidity:.2f}%")
    print("Status:", status)

    time.sleep(2)

```