

โครงการเครื่องเตือนอุณหภูมิและระยะห่างสำหรับคนทำอาหารใน Tinkercad โดยใช้ Arduino UNO + TMP Sensor + Ultrasonic + LCD I2C + Buzzer

1. ภาพรวมการทำงาน

อุปกรณ์นี้ใช้เพื่อ ตรวจสอบความปลอดภัยในครัว โดยใช้

- เซนเซอร์วัดอุณหภูมิ TMP → ตรวจสอบความร้อนบริเวณเตา
- เซนเซอร์วัดระยะ Ultrasonic → ตรวจสอบว่ามีคนอยู่ใกล้หรือไม่
- LCD I2C → แสดงค่าอุณหภูมิและระยะห่าง
- Buzzer → ส่งเสียงเตือนเมื่อเกิดเงื่อนไขอันตราย

2. ขั้นตอนการทำงาน

ขั้นตอนที่ 1: การวัดอุณหภูมิ

- TMP Sensor (เช่น TMP36) จะส่งสัญญาณแรงดันไฟฟ้าออกมาตามอุณหภูมิที่วัดได้
- Arduino อ่านค่าจากขา Analog A0 → แปลงเป็นแรงดัน (V) → คำนวณเป็น °C
- ตัวอย่าง:
 - 0.75V → 25°C
 - 1.0V → 50°C

- ค่าที่วัดได้จะใช้ในการตัดสินใจว่าร้อนเกินกว่ากำหนดหรือไม่ (เช่น > 40°C)

ขั้นตอนที่ 2: การวัดระยะ

- Ultrasonic ส่งคลื่นเสียงความถี่สูงออกมา (40 kHz) ผ่านขา Trig
- คลื่นเสียงกระทบวัตถุ (คน) และสะท้อนกลับมายังเซนเซอร์ (ขา Echo)
- Arduino จับเวลาที่เสียงเดินทางไป-กลับ แล้วคำนวณเป็นระยะทาง

ระยะทาง (cm) = (เวลาเดินทางของเสียง (μs) × 0.034) / 2

- ถ้าระยะที่วัดได้มากกว่า 50 cm → แปลว่าไม่มีคนอยู่ใกล้เตา

ขั้นตอนที่ 3: การแสดงผลบน LCD

- LCD I2C แสดงค่าทั้งสองอย่างพร้อมกัน
 - บรรทัดที่ 1: อุณหภูมิ (Temp: XX.X C)
 - บรรทัดที่ 2: ระยะห่าง (Dist: XX.X cm)
- อัปเดตค่าทุก 0.5 วินาที

ขั้นตอนที่ 4: เงื่อนไขการเตือน

- Arduino จะตรวจเงื่อนไข:
 - ถ้า อุณหภูมิ > 40°C และ ระยะห่าง > 50 cm
 - → เปิด Buzzer เพื่อเตือนว่ามีความร้อนสูงแต่ไม่มีคนอยู่ใกล้ → อาจลืมหันหลังไป
- ถ้าไม่ตรงเงื่อนไข → ปิดเสียง Buzzer

3. ลำดับเหตุการณ์เมื่อใช้งานจริง

1. เปิดเครื่อง → LCD แสดงค่าเริ่มต้นของอุณหภูมิและระยะ
2. TMP ตรวจวัดความร้อนที่เตาอย่างต่อเนื่อง
3. Ultrasonic ตรวจการมีอยู่ของคนในระยะไม่เกิน 50 cm
4. ถ้าเตาร้อนเกิน 40°C และไม่มีคน → Buzzer ดัง
5. เมื่อคนกลับเข้ามาใกล้เตา หรือปิดเตาทำให้อุณหภูมิลด → Buzzer หยุด

4. จุดเด่นของโครงการ

- เพิ่มความปลอดภัยในครัว ป้องกันการลืมหันหลังไป
- ใช้งานง่าย แค่เปิดเครื่องก็เริ่มตรวจจับทันที
- ข้อมูลชัดเจน ด้วย LCD I2C แสดงผลแบบ Real-time
- สามารถปรับค่าเกณฑ์อุณหภูมิและระยะให้เหมาะกับสภาพครัว

อุปกรณ์ที่ใช้

1. Arduino Uno R3
2. TMP36 (หรือสามารถใช้ TMP sensor อื่นก็ได้ใน Tinkercad)
3. Ultrasonic Sensor HC-SR04
4. LCD 16x2 พร้อมโมดูล I2C
5. Active Buzzer
6. สาย Jumper
7. โพรโทบอร์ด (Protoboard) หรือ เบรดบอร์ด (Breadboard)

การต่อวงจร

1. TMP Sensor (TMP36)

- ขา VCC → 5V
- ขา GND → GND
- ขา Vout → A0

2. Ultrasonic Sensor HC-SR04

- VCC → 5V
- GND → GND
- Trig → ขา 9

- Echo → ขา 10

3. LCD I2C

- VCC → 5V
- GND → GND
- SDA → SDA
- SCL → SCL

4. Buzzer

- ขา + → ขา 8
- ขา - → GND

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
// Pin กำหนด Ultrasonic
#define TRIG_PIN 9
#define ECHO_PIN 10
// Pin TMP
#define TMP_PIN A0
// Pin buzzer
#define BUZZER_PIN 8
void setup() {
    lcd.init();
    lcd.backlight();
    pinMode(TRIG_PIN, OUTPUT);
    pinMode(ECHO_PIN, INPUT);
    pinMode(BUZZER_PIN, OUTPUT);
}
void loop() {
    // อ่านค่าอุณหภูมิจาก TMP36
    int sensorValue = analogRead(TMP_PIN);
    float voltage = sensorValue * 5.0 / 1024.0;
    float temperatureC = (voltage - 0.5) * 100.0;
    // อ่านค่าระยะทางจาก Ultrasonic
    digitalWrite(TRIG_PIN, LOW);
```

```
    delayMicroseconds(2);
    digitalWrite(TRIG_PIN, HIGH);
    delayMicroseconds(10);
    digitalWrite(TRIG_PIN, LOW);
    long duration = pulseIn(ECHO_PIN, HIGH);
    float distance = duration * 0.034 / 2; // cm
    // แสดงผลบน LCD
    lcd.setCursor(0, 0);
    lcd.print("Temp: ");
    lcd.print(temperatureC, 1);
    lcd.print("C");
    lcd.setCursor(0, 1);
    lcd.print("Dist: ");
    lcd.print(distance, 1);
    lcd.print("cm ");
    // เงื่อนไขการเตือน
    if (temperatureC > 40 && distance > 50) {
        digitalWrite(BUZZER_PIN, HIGH);
    } else {
        digitalWrite(BUZZER_PIN, LOW);
    }
    delay(500);
}
```