การเพิ่มจอ LCD I2C เข้าไปในโปรเจกต์ Arduino ของคุณจะช่วยให้คุณสามารถแสดงผลระยะทางที่วัดได้จากเซ็นเซอร์ Ultrasonic HC-SR04 ได้อย่างสะดวก นี่คือตัวอย่างโค้ดที่รวมการใช้งานเซ็นเซอร์ Ultrasonic HC-SR04 และจอ LCD I2C:

**อุปกรณ์ที่ต้องใช้:**

1. Arduino Uno
2. เซ็นเซอร์ Ultrasonic HC-SR04
3. จอ LCD I2C (เช่น 16x2)
4. สายจัมเปอร์
5. บอร์ดเบรดบอร์ด (ถ้ามี)

**การเชื่อมต่อ:**

1. เชื่อมต่อ VCC ของ HC-SR04 กับ 5V ของ Arduino
2. เชื่อมต่อ GND ของ HC-SR04 กับ GND ของ Arduino
3. เชื่อมต่อ Trig ของ HC-SR04 กับ Pin 9 ของ Arduino
4. เชื่อมต่อ Echo ของ HC-SR04 กับ Pin 10 ของ Arduino
5. เชื่อมต่อ GND ของ LCD I2C กับ GND ของ Arduino
6. เชื่อมต่อ VCC ของ LCD I2C กับ 5V ของ Arduino
7. เชื่อมต่อ SDA ของ LCD I2C กับ A4 ของ Arduino
8. เชื่อมต่อ SCL ของ LCD I2C กับ A5 ของ Arduino

**โค้ดตัวอย่าง:**

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

const int trigPin = 9;

const int echoPin = 10;

long duration;

int distance;

LiquidCrystal\_I2C lcd(0x27, 16, 2); // กำหนดที่อยู่ I2C ของจอ LCD

void setup() {

pinMode(trigPin, OUTPUT);

pinMode(echoPin, INPUT);

Serial.begin(9600);

lcd.init(); // เริ่มต้นการทำงานของจอ LCD

lcd.backlight(); // เปิดไฟหลังจอ LCD

}

void loop() {

// ส่งสัญญาณเสียง

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

// วัดเวลาที่สัญญาณเสียงใช้ในการเดินทางไปกลับ

duration = pulseIn(echoPin, HIGH);

// คำนวณระยะทาง

distance = duration \* 0.034 / 2;

// แสดงผลระยะทางบน Serial Monitor

Serial.print("Distance: ");

Serial.print(distance);

Serial.println(" cm");

// แสดงผลระยะทางบนจอ LCD

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Distance: ");

lcd.setCursor(0, 1);

lcd.print(distance);

lcd.print(" cm");

// หน่วงเวลา 500 มิลลิวินาที

delay(500);

}

**การทำงานของโค้ด:**

1. **การตั้งค่า**:
   * กำหนดขา trigPin และ echoPin
   * ตั้งค่า trigPin เป็นขาออก (OUTPUT) และ echoPin เป็นขาเข้า (INPUT)
   * เริ่มการสื่อสารแบบ Serial และการทำงานของจอ LCD
2. **การส่งสัญญาณเสียง**:
   * ส่งสัญญาณเสียงจาก trigPin โดยการตั้งค่า HIGH และ LOW ตามลำดับ
3. **การวัดเวลา**:
   * ใช้ฟังก์ชัน pulseIn เพื่อวัดเวลาที่สัญญาณเสียงใช้ในการเดินทางไปกลับ
4. **การคำนวณระยะทาง**:
   * คำนวณระยะทางจากเวลาที่วัดได้ โดยใช้สูตร: (\text{ระยะทาง} = \frac{\text{เวลา} \times 0.034}{2})
5. **การแสดงผล**:
   * แสดงผลระยะทางบน Serial Monitor และจอ LCD

ลองนำโค้ดนี้ไปใช้ดูนะครับ! ถ้ามีคำถามเพิ่มเติมหรือปัญหาอะไร สามารถถามได้เลยครับ 😊