

# หน่วยการเรียนรู้ที่ 1 อุปกรณ์และเซนเซอร์

## จอ LCD 1602



คำว่า LCD ย่อมาจากคำว่า Liquid Crystal Display ซึ่งเป็นจอที่ทำมาจากผลึกคริสตัลเหลว หลักการคือด้านหลังจอจะมีไฟส่องสว่าง หรือที่เรียกว่า Backlight อยู่ เมื่อมีการปล่อยกระแสไฟฟ้าเข้าไปกระตุ้นที่ผลึกก็จะทำให้ผลึกโปร่งแสง ทำให้แสงที่มาจากไฟ Backlight แสดงขึ้นมาบนหน้าจอ ส่วนอื่นที่โดนผลึกปิดกั้นไว้จะมีสีที่แตกต่างกันตามสีของผลึกคริสตัล เช่น สีเขียว หรือ สีฟ้า ทำให้เมื่อมองไปที่จอก็จะพบกับตัวหนังสือสีขาว แล้วพบกับพื้นหลังสีต่าง ๆ กัน

จอ LCD จะแบ่งเป็น 2 แบบใหญ่ๆตามลักษณะการแสดงผลดังนี้

**1. Character LCD** เป็นจอที่แสดงผลเป็นตัวอักษรตามช่องแบบตายตัว

จอ LCD ที่ใช้งานร่วมกับโมดูล LCD I2C นี้มีอยู่ 2 รุ่น คือ

1. รุ่น LCD 16x2 หรือรุ่น LCD 1602 สามารถแสดงข้อความได้ 16 ตัวอักษร 2 บรรทัด
2. รุ่น LCD 20x4 หรือรุ่น LCD 2004 สามารถแสดงข้อความได้ 20 ตัวอักษร 4 บรรทัด

**2. Graphic LCD** เป็นจอที่สามารถกำหนดได้ว่าจะให้แต่ละจุดบนหน้าจอขึ้นแสง หรือปล่อยแสงออกไป ทำให้อจอสามารถสร้างรูปขึ้นมาบนหน้าจอได้ การระบุขนาดจะระบุในลักษณะของจำนวนจุด (Pixels) ในแต่ละแนว เช่น 128x64 หมายถึงจอที่มีจำนวนจุดตามแนวนอน 128 จุด และมีจุดตามแนวตั้ง 64 จุด

ในบทนี้จะกล่าวถึง Character LCD เพียงอย่างเดียว เนื่องจากใช้งานได้ง่าย และนิยมใช้งานในโปรเจกต์ต่างๆ ไปมากกว่า

## การเชื่อมต่อกับจอ Character LCD

การเชื่อมต่อจะมีด้วยกัน 2 แบบ คือ

- การเชื่อมต่อแบบขนาน - เป็นการเชื่อมต่อจอ LCD เข้ากับบอร์ด Arduino โดยตรง โดยจะแบ่งเป็นการเชื่อมต่อแบบ 4 บิต และการเชื่อมต่อแบบ 8 บิต ใน Arduino จะนิยมเชื่อมต่อแบบ 4 บิต เนื่องจากใช้สายในการเชื่อมต่อน้อยกว่า
- การเชื่อมต่อแบบอนุกรม - เป็นการเชื่อมต่อกับจอ LCD ผ่าน โมดูลแปลงรูปแบบการเชื่อมต่อกับจอ LCD จากแบบขนาน มาเป็นการเชื่อมต่อแบบอื่นที่ใช้สายน้อยกว่า เช่น การใช้โมดูล I2C Serial Interface จะเป็นการนำโมดูลเชื่อมเข้ากับตัวจอ LCD แล้วใช้บอร์ด Arduino เชื่อมต่อกับบอร์ดโมดูลผ่านโปรโตคอล I2C ทำให้ใช้สายเพียง 4 เส้น ก็ทำให้หน้าจอแสดงผลข้อความต่างๆ ออกมาได้

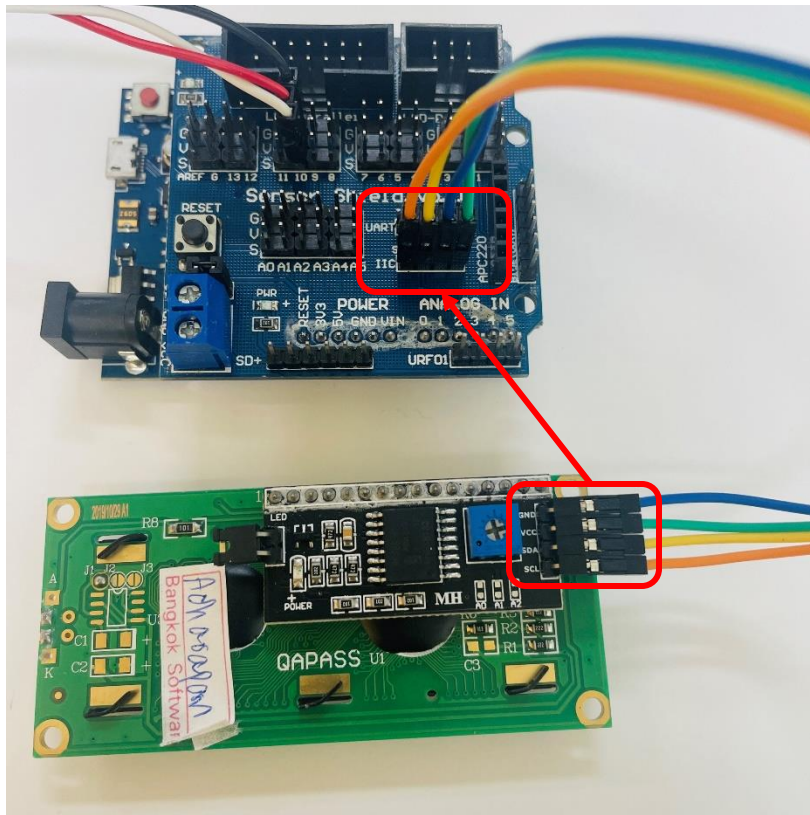
### ในบทนี้เราจะพูดถึงการเชื่อมต่อแบบอนุกรม

การเชื่อมต่อแบบอนุกรม จะใช้งาน โมดูล I2C Serial Interface Board Module มาเชื่อมต่อระหว่าง Arduino กับจอ LCD

### การต่ออุปกรณ์

ใช้สายแพ ตัวเมีย ตัวเมีย 4 เส้น

LCD		บอร์ด Arduino
GND	ต่อกับ	-
VCC	ต่อกับ	+
SDA	ต่อกับ	SDA
SCL	ต่อกับ	SCL



## รายละเอียดคำสั่งในการสั่งงานระหว่าง Arduino กับ จอ LCD

คำสั่งในการควบคุมจอ LCD ของ Arduino นั้น ทาง Arduino.cc เขียนเป็น Library มาให้เพื่อสะดวกในการนำไปใช้งาน หลังจากต่อสายเสร็จเรียบร้อยแล้ว ขั้นตอนแรกในการเริ่มเขียนโปรแกรมคือการเรียกใช้ Library ของ LCD จากไฟล์ชื่อ LiquidCrystal.h หลังจากนั้นมาดูกันว่า มีฟังก์ชันที่สำคัญอะไรบ้างที่ใช้สั่งงานให้จอ LCD

**ฟังก์ชัน LiquidCrystal():** ใช้ประกาศที่ต้องการส่งข้อมูลไปยังจอ LCD รูปแบบในการสั่งงานคือ  
LiquidCrystal lcd(rs, enable, d4, d5, d6, d7) <<<<<<< ในกรณีใช้งานแบบ 4 บิต

LiquidCrystal lcd(rs, enable, d0, d1, d2, d3, d4, d5, d6, d7) <<<<<<< ในกรณีใช้งานแบบ 8 บิต

ในบทความนี้ใช้แบบ 4 บิต คือ LiquidCrystal lcd(12, 11, 4, 5, 6, 7); ก็หมายถึงการเชื่อมต่อ rs ที่ขา 12 , Enable ที่ขา 11 , และ DB4-DB7 ที่ขา 4-7 ของ Arduino ตามลำดับ

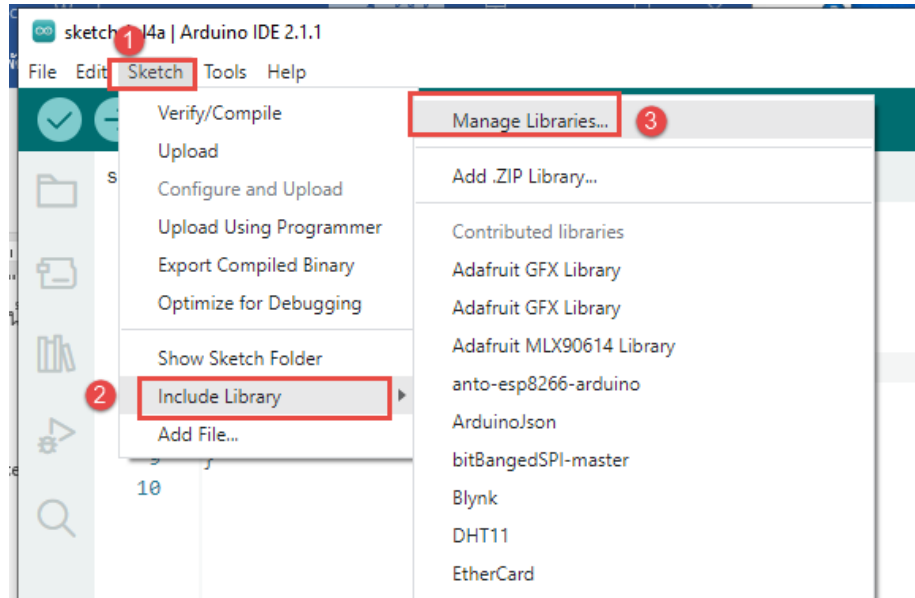
**ฟังก์ชัน begin():** ใช้กำหนดขนาดของจอ ในบทความนี้เราใช้ขนาด 16 ตัวอักษร 2 บรรทัด จึงประกาศเป็น lcd.begin(16, 2);

**ฟังก์ชัน setCursor():** ใช้กำหนดตำแหน่งและบรรทัดของ Cursor เช่น lcd.setCursor(0, 1); คือ ให้เคอร์เซอร์ไปที่ตำแหน่งที่ 0 บรรทัดที่ 1 การนับตำแหน่งเริ่มจาก 0 ดังนั้น LCD 16x2 มีตำแหน่ง 0 – 15 บรรทัด คือ 0 กับ 1

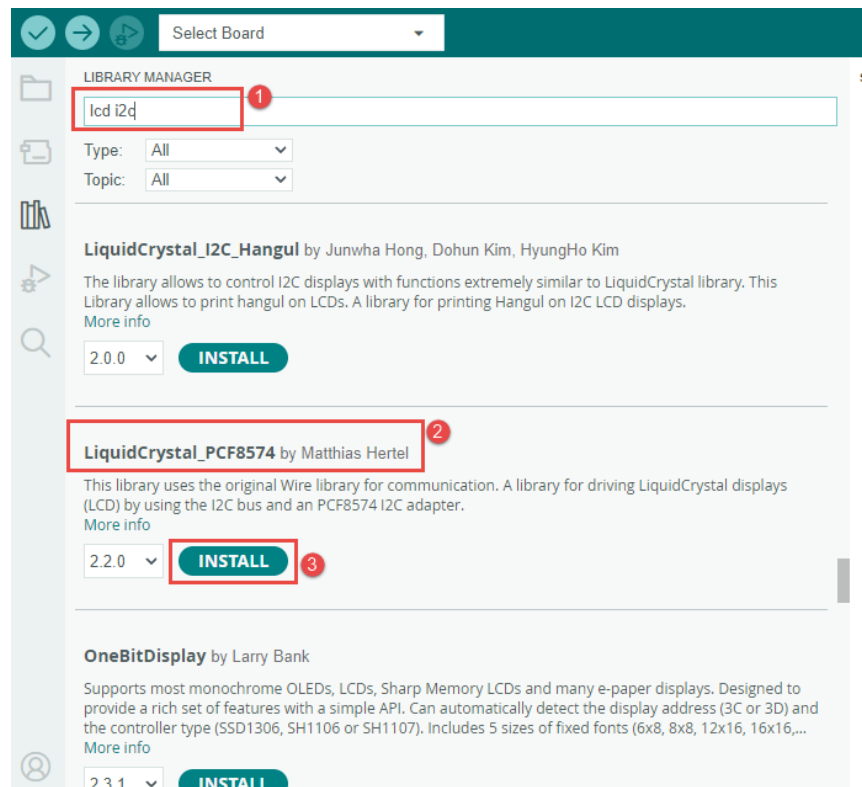
**ฟังก์ชัน print():** ใช้กำหนดข้อความที่ต้องการแสดง เช่น lcd.print("168Education"); คือ ให้แสดงข้อความ "168Education" ออกทางหน้าจอ LCD

## วิธีการโหลด Library มาใช้งาน

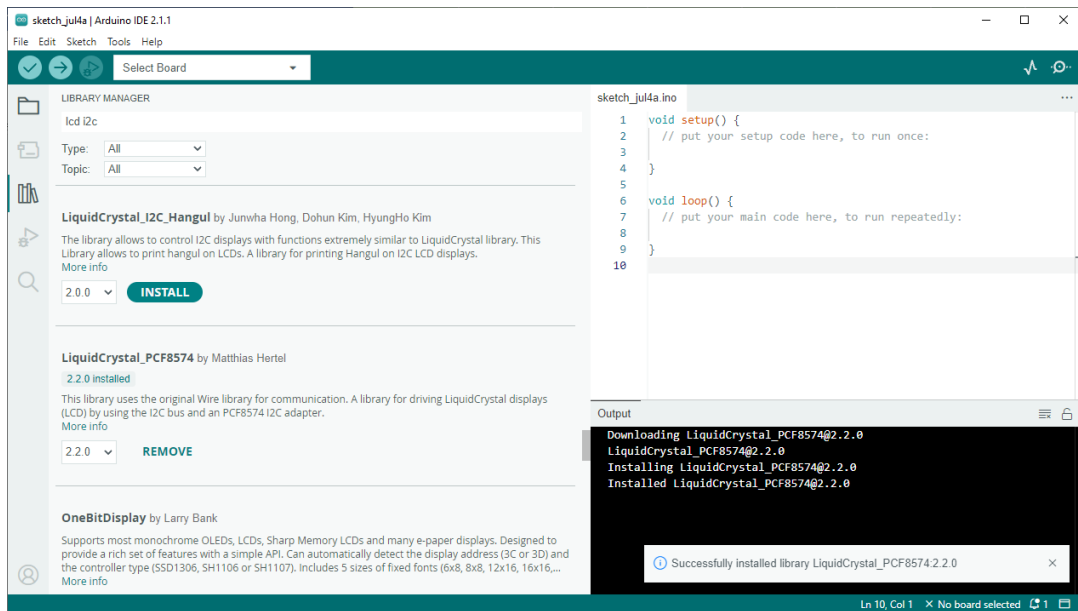
1. เปิดโปรแกรม Arduino IDE
2. ไปที่คำสั่ง Sketch > Include Library คลิกเลือกคำสั่ง Manage Libraries



3. จะปรากฏหน้าต่าง Library Manager ในช่องค้นหาให้พิมพ์ว่า LCD I2C จากนั้นหาไลบรารีที่ชื่อว่า **LiquidCrystal\_PCF8574** แล้วให้คลิกที่ปุ่ม **Install**

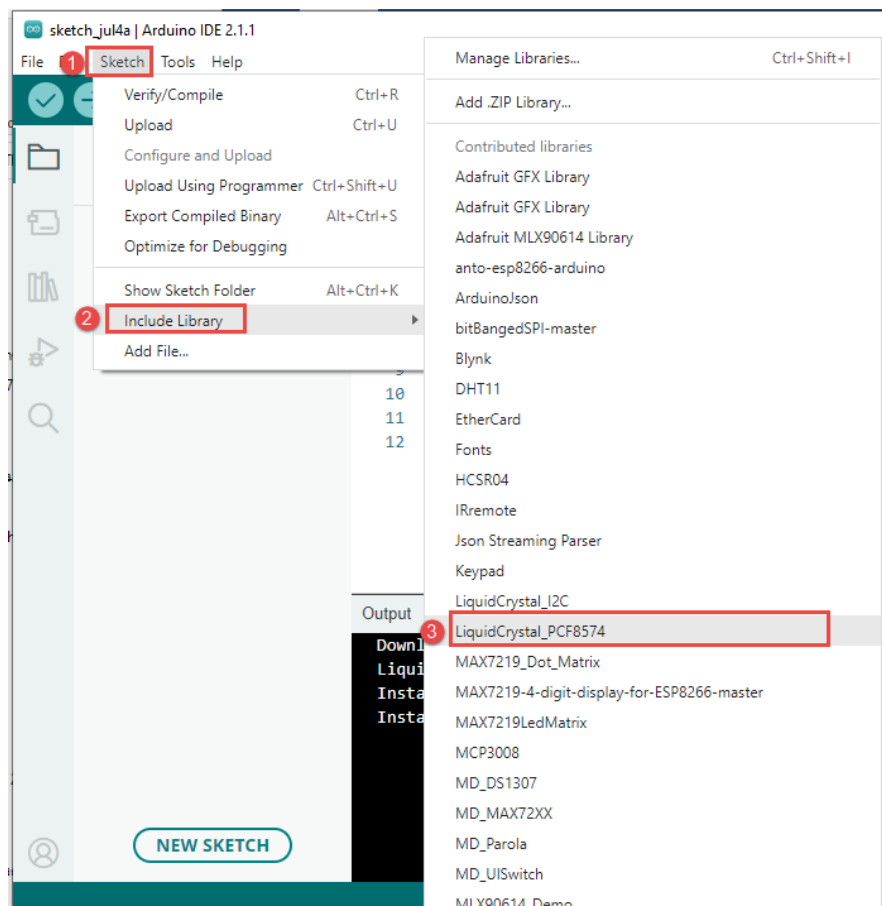


#### 4. รอจนกว่าจะทำการติดตั้งเสร็จ



#### การเรียกใช้ไลบรารี

1. ไปที่โปรแกรม Arduino IDE
2. เลือกที่คำสั่ง Sketch > Include Library คลิกที่คำสั่ง LiquidCrystal\_PCF8574



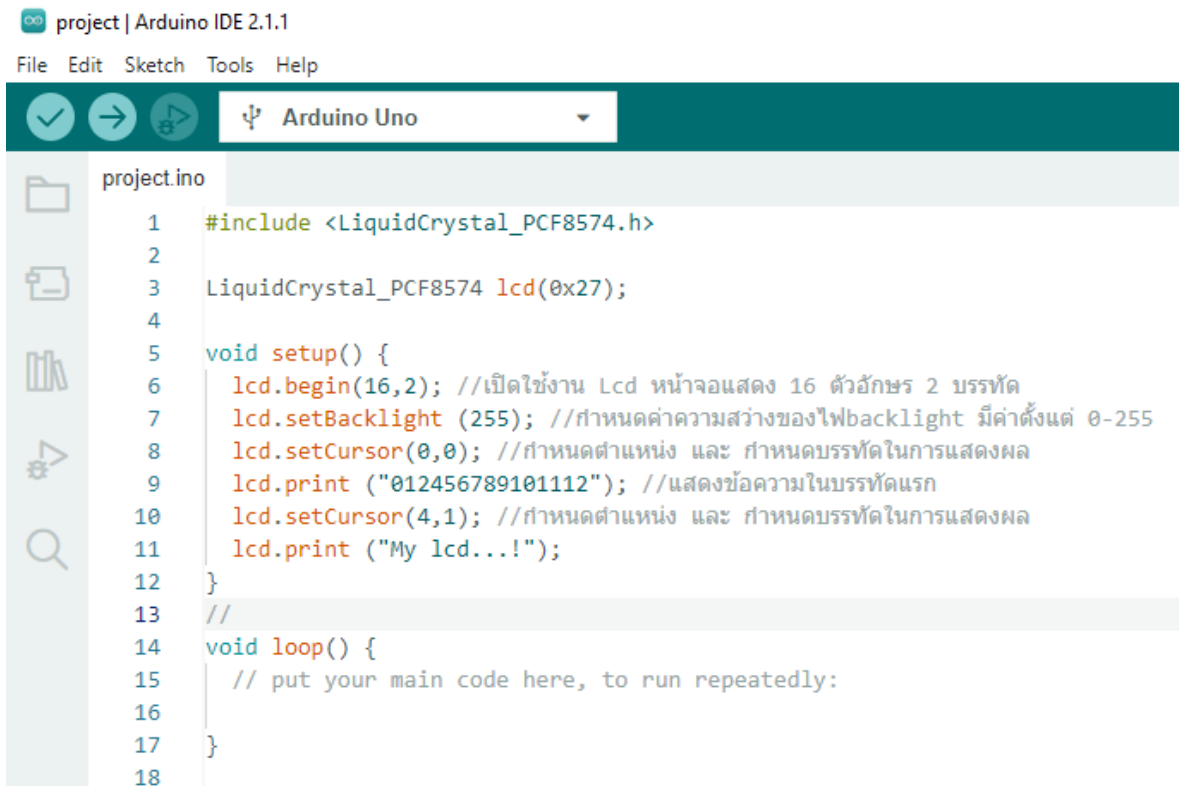
จะปรากฏโค้ดตามภาพ

```
sketch_jul4a.ino
1  #include <LiquidCrystal_PCF8574.h>
2
3  void setup() {
4    // put your setup code here, to run once:
5
6  }
7
8  void loop() {
9    // put your main code here, to run repeatedly:
10
11 }
12
```

บรรทัดที่ 3 เราจะพิมพ์โค้ด `LiquidCrystal_PCF8574 lcd(0x27);`

```
sketch_jul4a.ino
1  #include <LiquidCrystal_PCF8574.h>
2
3  LiquidCrystal_PCF8574 lcd(0x27);
4
5  void setup() {
6    // put your setup code here, to run once:
7
8  }
9
```

## ทดลองเขียนคำสั่งให้อจอ lcd แสดงผล



The screenshot shows the Arduino IDE 2.1.1 interface. The menu bar includes File, Edit, Sketch, Tools, and Help. The toolbar contains icons for saving, running, and uploading. The board is set to 'Arduino Uno'. The sketch editor shows the following code:

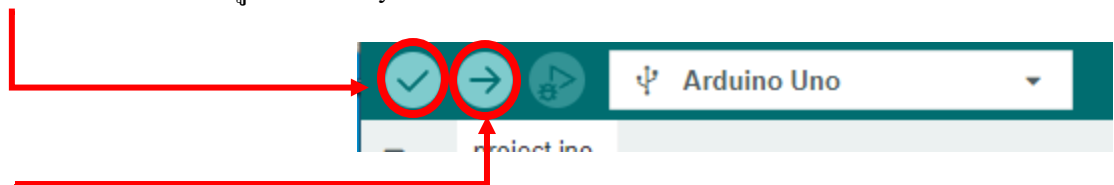
```
1 #include <LiquidCrystal_PCF8574.h>
2
3 LiquidCrystal_PCF8574 lcd(0x27);
4
5 void setup() {
6   lcd.begin(16,2); //เปิดใช้งาน Lcd หน้าจอแสดง 16 ตัวอักษร 2 บรรทัด
7   lcd.setBacklight (255); //กำหนดค่าความสว่างของไฟbacklight มีค่าตั้งแต่ 0-255
8   lcd.setCursor(0,0); //กำหนดตำแหน่ง และ กำหนดบรรทัดในการแสดงผล
9   lcd.print ("012456789101112"); //แสดงข้อความในบรรทัดแรก
10  lcd.setCursor(4,1); //กำหนดตำแหน่ง และ กำหนดบรรทัดในการแสดงผล
11  lcd.print ("My lcd...!");
12 }
13 //
14 void loop() {
15   // put your main code here, to run repeatedly:
16 }
17 }
18
```

### อธิบายโค้ด

เริ่มต้นโค้ดกำหนดลักษณะการแสดงผลของจอ lcd คือการแสดงผลภาพ 16 ตัวอักษร 2 แถว ตั้งค่าความสว่าง มีค่าตั้งแต่ 0-255 กำหนดให้ตัวอักษรที่ 0 แถวที่ 0 แสดงข้อความ “012356789101112” กำหนดให้ตัวอักษรที่ 4 แถวที่ 1 แสดงข้อความ “My lcd...!”

จากนั้นทำการบันทึกไฟล์ก่อน แล้วเช็คความถูกต้องของ syntax ถ้าถูกต้อง ให้ทำการ Upload โค้ดคำสั่งลงบอร์ด Arduino

กดที่เครื่องหมาย เพื่อเช็คที่โค้ดถูกต้องตาม syntax ใหม่



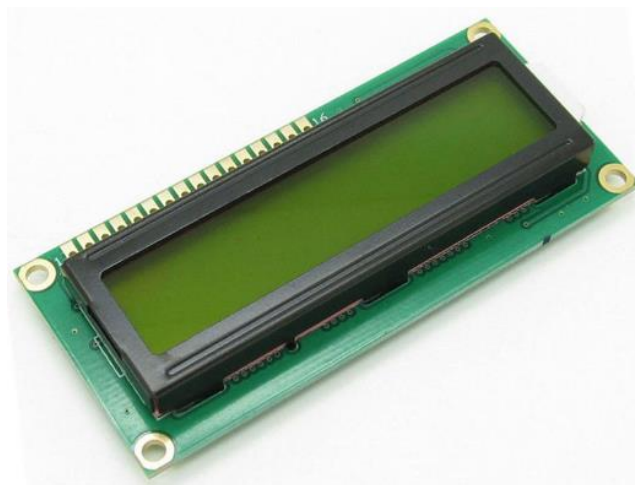
กดที่เครื่องหมาย เพื่อ upload โปรแกรมเข้าบอร์ด

ผลลัพธ์ที่ได้



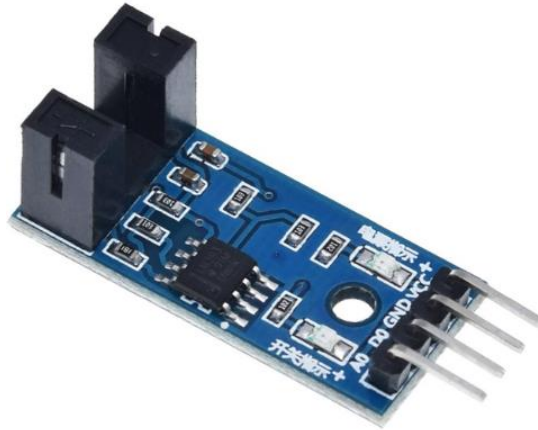
หมายเหตุ กรณีที่จอ lcd เป็นสีเหลืองให้เขียนโค้ด

บรรทัดที่3 LiquidCrystal\_PCF8574 lcd(0x3F);





## เซนเซอร์นับเหรียญ นับรอบ นับจำนวน ตรวจจับวัตถุผ่าน ก้ามปู



เป็นโมดูลที่ใช้สำหรับวัดความเร็วการหมุนของมอเตอร์การทำงานเซนเซอร์นับเหรียญจะส่งสัญญาณ digital ออกมาเป็น 1 กับ 0 (HIGH,LOW) และ analog ออกมาเป็น 1024 , 0 (HIGH,LOW) ทำให้สามารถต่อใช้งานกับ Arduino ได้ง่าย

สถานะถ้าไม่เจอวัตถุผ่านจะเป็น HIGH หรือค่าจะเป็น 1 ถ้ามีวัตถุผ่านจะเปลี่ยนสถานะเป็น LOW หรือ ค่าจะเป็น 0

ในโปรแกรมเราจะใช้ขา 2 ของ Arduino ในการอ่านข้อมูล เป็นขา interrupt เหตุผลสำหรับที่ต้องใช้ ขา interrupt ก็คือให้ขั้วจิ้งหหวะ โปรแกรมเมื่อ ขา 2 เปลี่ยนสถานะ ลอจิก 1 เป็น 0 ให้มางานในฟังก์ชัน interrupt 2 ก่อน เพื่อให้สามารถอ่านสัญญาณที่เปลี่ยนแปลงได้ทัน

ถ้าเราไม่ใช้ขา interrupt อ่าน บางครั้งโปรแกรมไปทำงานอย่างอื่นอยู่ เช่นติดคำสั่ง Delay อยู่ ทำให้วนกลับมาอ่านสถานะลอจิกที่เซนเซอร์นับเหรียญส่งมาไม่ทัน

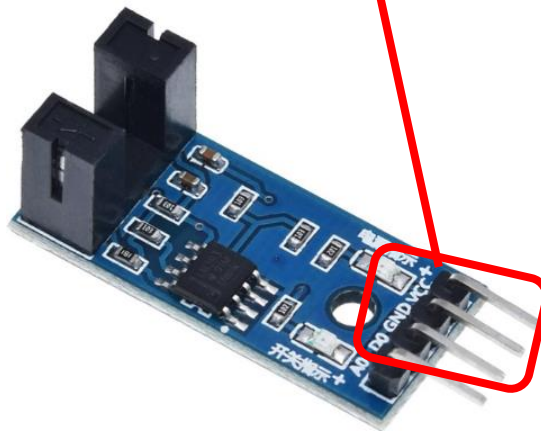
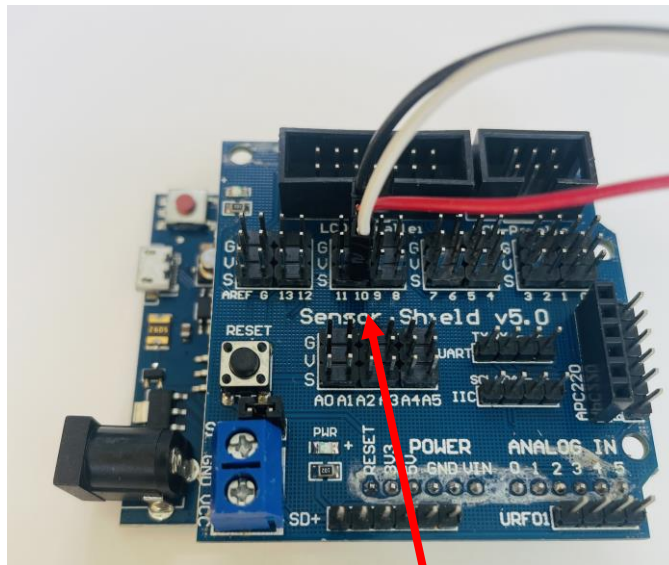
### วิธีการต่ออุปกรณ์ เซ็นเซอร์นับเหรียญ นับรอบ นับจำนวน ตรวจจับวัตถุผ่าน ก้ามปู

การต่อให้ใช้สายแพ ตัวเมีย ตัวเมีย ต่อเข้ากับบอร์ด Arduino uno r3 ไปต่อกับเซนเซอร์ ก้ามปู

S10 -> D0

V10 -> VCC

G10 -> GND



เริ่มต้นการเขียนคำสั่ง

### 1. ประกาศค่าตัวแปร

```
1 #include <LiquidCrystal_PCF8574.h>
2
3 LiquidCrystal_PCF8574 lcd(0x27);
4
5 int pin_coin_10 = 10; //กำหนดขาเซนเซอร์นับเหรียญ
6 int state_coin_10 = 0; //รับค่าจากเซนเซอร์
7
8 bool check_coin_10 = false; //เช็คสถานะเซนเซอร์
9
10 int count_coin_10 = 0; //เก็บค่าที่นับเหรียญ
11
```

## 2. ในส่วนของ void setup() กำหนดการรับค่าของข้อมูล

```
14 void setup() {
15     lcd.begin(16,2); //เปิดใช้งาน Lcd หน้าจอแสดง 16 ตัวอักษร 2 บรรทัด
16     lcd.setBacklight (255); //กำหนดค่าความสว่างของไฟbacklight มีค่าตั้งแต่ 0-255
17     lcd.setCursor(0,0); //กำหนดตำแหน่ง และ กำหนดบรรทัดในการแสดงผล
18     lcd.print ("012456789101112"); //แสดงข้อความในบรรทัดแรก
19     lcd.setCursor(4,1); //กำหนดตำแหน่ง และ กำหนดบรรทัดในการแสดงผล
20     lcd.print ("My lcd...!");
21
22     pinMode (pin_coin_10, INPUT); // กำหนดรับค่าข้อมูล
23
24 }
```

### อธิบายโค้ด

**pinMode** คำสั่งที่มีไว้สำหรับกำหนดการทำงานของ pin ที่ต้องการใช้งาน ให้ทำงานเป็นแบบ OUTPUT หรือ INPUT ในที่นี้กำหนดให้ pin\_coin\_10 การทำงานเป็นแบบ input

## 3. สร้างฟังก์ชัน sensor\_coin

```
// ฟังก์ชัน sensor_coin
void sensor_coin(){
    state_coin_10 = digitalRead(pin_coin_10);

    if (check_coin_10 == false && state_coin_10 == 1){ //ถ้ามีเหรียญหยอดเข้ามาแล้วผ่านตัวเซนเซอร์จะมีค่าเป็น 1
        count_coin_10 += 1; // ให้ทำการบวกตัวแปรไป 1
        check_coin_10 = true; //เปลี่ยน Check_coin_10 เป็น true
    } else{
        if (state_coin_10 == 0){// เช็การหยอดเหรียญค้างไว้ เพื่อไม่ให้นับไปเรื่อยๆ
            check_coin_10 = false; // กำหนดสถานะ เป็น False
        }
    }
}
```

### อธิบายโค้ด

สร้างเงื่อนไข ถ้ามีการหยอดเหรียญผ่านเซนเซอร์ ทำการเพิ่มค่าตัวแปร count\_coin\_10 ไปทีละ 1 และเปลี่ยน check\_coin\_10 เป็นจริง

และ ถ้า state\_coin\_10 เท่ากับ 0 เพื่อทำการเช็คการหยอดเหรียญค้างไว้หรือไม่เพื่อไม่ให้ระบบทำการนับไปเรื่อยๆ check\_coin\_10 เป็นเท็จ

#### 4. เขียนคำสั่งใน void loop()

เช็คการหยอดเหรียญว่าหยอดครบไหม

```
}  
//  
void loop() { // การทำงานใน loop  
  if(count_coin_10 == 0){ //ถ้า count_coin_10 = 0 ให้ทำฟังก์ชัน sensor_coin  
    | sensor_coin();  
  }else if(count_coin_10 == 1){ //ถ้า count_coin_10 = 1 ให้ทำฟังก์ชัน sensor_coin  
    | sensor_coin();  
  }else{ //ถ้าหยอดเหรียญครบ 2 เหรียญแล้วให้ทำฟังก์ชัน select_menu  
    | select_menu();  
  }  
}
```

#### อธิบายโค้ด

ให้ void loop จะทำการเช็คว่ามีเหรียญหยอดไปกี่เหรียญ และได้ทำการหยอดครบ 2 เหรียญแล้วหรือยัง ถ้าครบแล้วก็ให้เลือกเมนู

#### ทดสอบการเขียนโปรแกรม

```
29 void loop() { // การทำงานใน loop  
30   if(count_coin_10 == 0){ //ถ้ายังไม่มีการหยอดเหรียญให้ทำฟังก์ชัน sensor_coin  
31     | lcd.clear();  
32     | lcd.setCursor(0,0);  
33     | lcd.print("Insert coin");  
34     | lcd.setCursor(0,1);  
35     | lcd.print("Get coin 10B");  
36     | sensor_coin();  
37   }else if(count_coin_10 == 1){ //ถ้า มีการหยอดไปแล้ว 1 เหรียญให้ทำฟังก์ชัน sensor_coin  
38     | lcd.clear();  
39     | lcd.setCursor(0,0);  
40     | lcd.print("Insert coin");  
41     | lcd.setCursor(0,1);  
42     | lcd.print("coin balance 1");  
43     | sensor_coin();  
44   }else{ //ถ้าหยอดเหรียญครบ 2 เหรียญแล้วให้ทำฟังก์ชัน select_menu  
45     | lcd.clear();  
46     | lcd.setCursor(0,0);  
47     | lcd.print("select menu");  
48     | //select_menu();  
49   }  
50 }
```

# ทดสอบรันโปรแกรมบนอุปกรณ์

