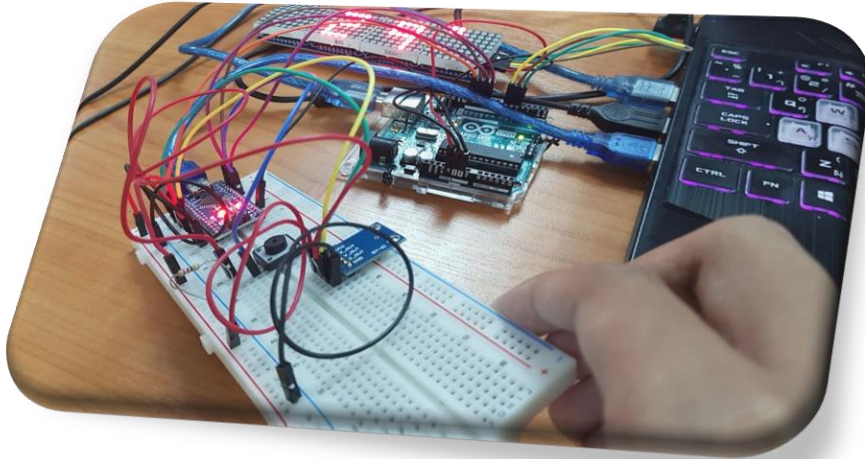


# Level Meter

## #Assignment 5

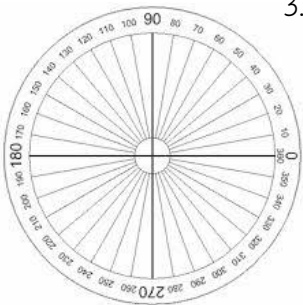


### แนวคิดและโครงสร้างของโปรแกรม

1. สร้างฟังก์ชันเพื่อเปลี่ยนค่าจาก accelerator ให้มีความนิ่งมากขึ้น
2. สร้างปุ่ม calibrate เพื่อรับค่าแกน X ที่ 0 องศา เมื่อมีการกด

```
//----- Calibrate -----
int c = digitalRead(8);
if (c == HIGH && millis() - Time > 250 || millis() < 1000 ) {
    Serial.println(" Calibrate !!");
    Calibrate = get_x();
    Time = millis();
}
```

3. นำค่าจากแกน X ที่ได้มาเข้าฟังก์ชัน map เพื่อแปลงข้อมูลให้ได้เป็นค่ามุมที่ต้องการ เมื่อมุมมีค่าเกิน 90 องศา จะตรวจสอบค่าจากแกน Z ในการเช็ค



```
//----- Map-Func -----
if (get_z() < 363 && get_x() >= 359)
    angle = map(get_x(), Calibrate, Calibrate + 70, 0, 90);
else if (get_z() >= 363)
    angle = map(get_x(), Calibrate + 70, Calibrate, 90, 180);
else if (get_x() < 359)
    angle = map(get_x(), Calibrate, Calibrate - 70, 359, 270);
```

4. นำค่าองศาที่ได้มาส่งข้อมูลข้ามบอร์ด โดยใช้ไลบรารี #include<SPI.h>

```
//----- Sent-Data -----
int sent = (int)angle;
Serial.print(" Sent = ");
Serial.println(sent);
digitalWrite(SS, LOW);
SPI.transfer(sent / 100 + '0');
SPI.transfer((sent / 10) % 10 + '0');
SPI.transfer(sent % 10 + '0');
digitalWrite(SS, HIGH);
SPI.endTransaction();
```

5. นำค่าที่รับมาโดยใช้ไลบรารี #include<SPI.h>

```
f = buffer[2] - '0';
m = buffer[1] - '0';
l = buffer[0] - '0';
```

แสดงบน dot matrix 8\*32 โดยใช้ไลบรารี #include <LedControl.h>

```
if ((buffer[0] == l && buffer[1] == m && buffer[2] == f)
&& (buffer[0] != buffer[1] && buffer[1] != buffer[2]))
{
    Serial.println(" Work !!! ");
} else {
    if (m == 0 && l == 0) { //check digit of value
        showNum(1, num[f]);
        lc.clearDisplay(2);
        lc.clearDisplay(3);
    } else if (l == 0) {
        showNum(1, num[f]);
        showNum(2, num[m]);
        lc.clearDisplay(3);
    } else {
        showNum(1, num[f]);
        showNum(2, num[m]);
        showNum(3, num[l]);
    }
}
```