



รายงาน

กลุ่มที่ 3

หัวข้อ MCP23008 I/O Expander with I2C (4LED,3switches)+I2C LCD

วิชา 030513243 COMPUTER PRACTICE I

จัดทำโดย

นาย ภาณุพงศ์ พร้าวงษ์ 6403051613059

นางสาว อัจฉรา พาทอน 6403051613041

นางสาว บุญยวีร์ ศรีทอง 6403051623127

## MCP23008 I/O Expander with I2C (4LED,3switches)+I2C LCD

MCP23008 เป็น I/O Expander ที่สามารถเพิ่มช่อง Input/Output (I/O) ในการเชื่อมต่อกับไมโครคอนโทรลเลอร์ โดยใช้ I2C (Inter-Integrated Circuit) เป็นส่วนการสื่อสาร.

4 LED: สามารถเชื่อมต่อ LED 4 ตัวไปยัง 4 ขาของ MCP23008 และควบคุมสถานะของ LED ได้โดยส่งข้อมูลผ่าน I2C เพื่อเปิด/ปิด LED

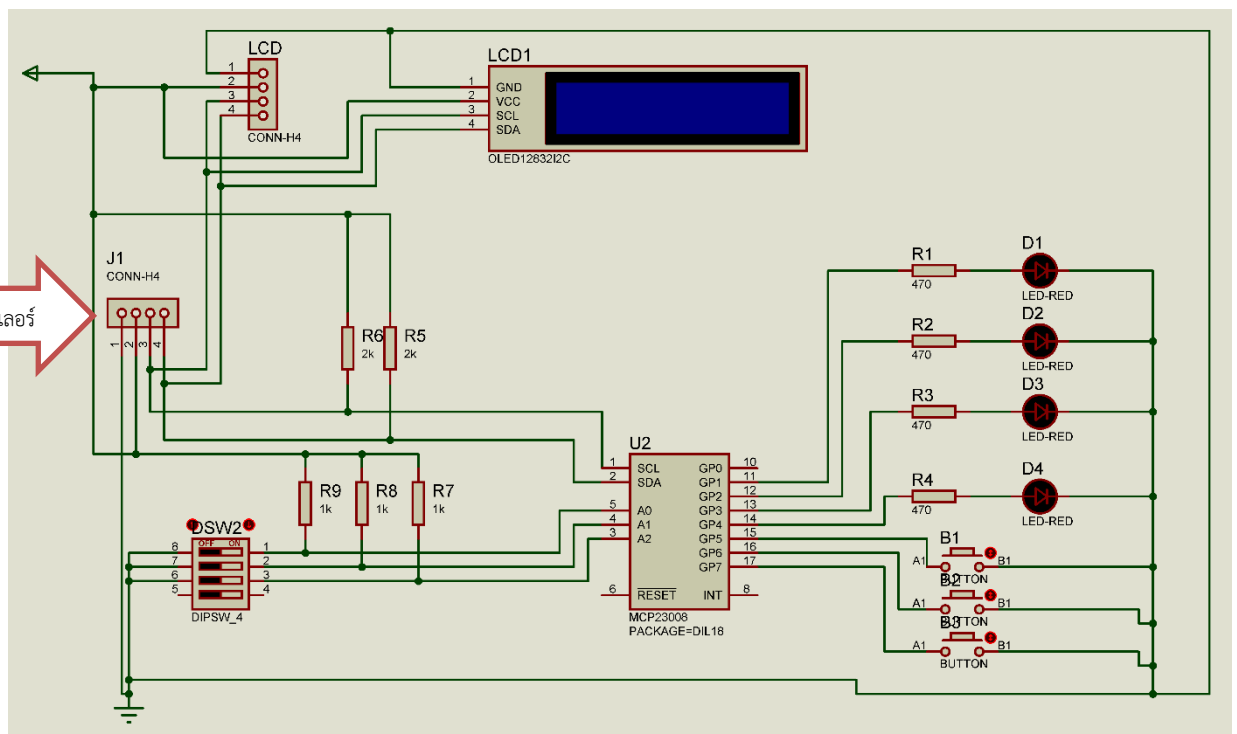
3 สวิตช์: สามารถเชื่อมต่อสวิตช์ 3 ตัวไปยัง 3 ขาของ MCP23008 และใช้ MCP23008 เพื่ออ่านสถานะของสวิตช์ที่เชื่อมต่อ ให้ตรวจับการกดปล่อยของสวิตช์และประมวลผลข้อมูล

I2C LCD: เชื่อมต่อหน้าจอ LCD ที่รองรับ I2C กับ MCP23008 ใช้ MCP23008 เพื่อควบคุมการแสดงผลบนหน้าจอ

### การออกแบบวงจร

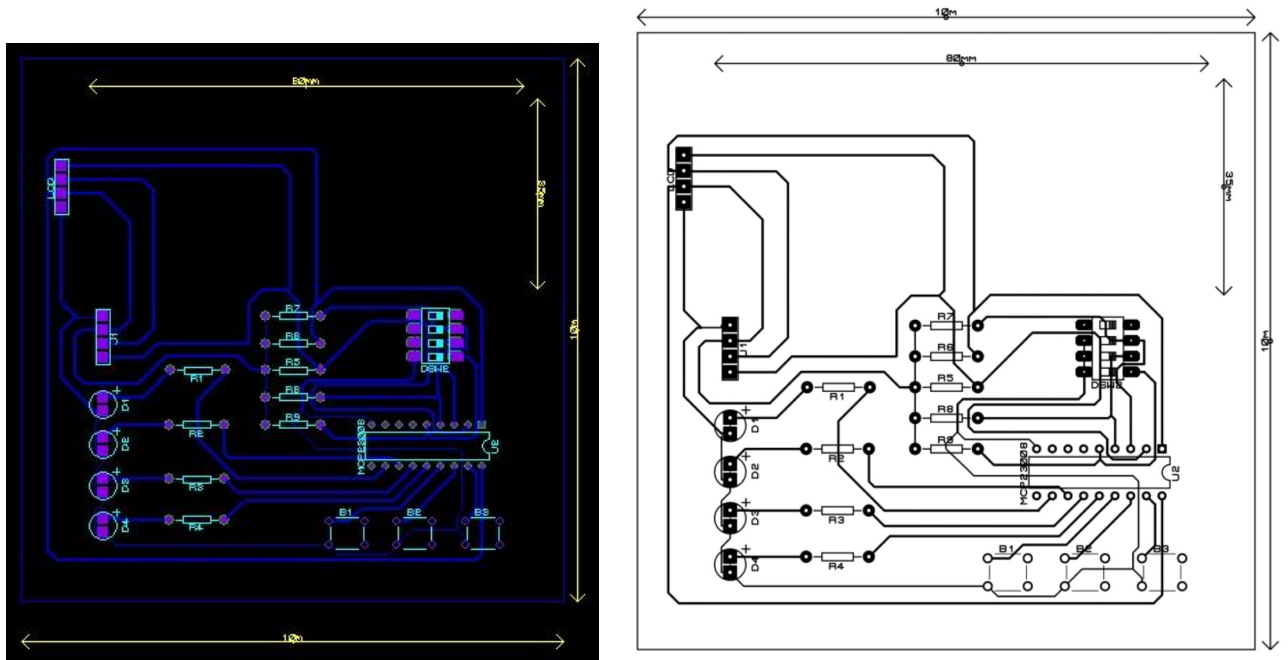
- LED1-4 : เชื่อมต่อกับขา GP0-GP4 ของ MCP23008
- Switch1-3 : เชื่อมต่อกับขา GP5-GP7 ของ MCP23008
- A0-A2 เข้า dip switch เพื่อกำหนดแอดเดรส ของ MCP23008
- การสื่อสารอนุกรมแบบ I2C รับส่งข้อมูลระหว่างไมโครคอนโทรลเลอร์กับจอLCD ใช้สายสัญญาณ SCL , SDA ต่อตัวต้านทานแบบ pull-up

### การออกแบบวงจร schematic



รูปที่ 1 วงจร schematic MCP23008 I/O Expander with I2C (4LED,3switches)+I2C LCD

## การออกแบบวงจร PCB



รูปที่ 2 วงจร PCB MCP23008 I/O Expander with I2C (4LED,3switches)+I2C LCD

## ตัวอย่างโปรแกรมควบคุม hardware

### การทำงานโปรแกรมควบคุมการทำงาน

- ในตอนแรก หน้าจอLCD แสดงผล LCD ON Press to start  
จำเป็นต้องกดสวิตช์เพื่อเริ่มทำงาน
- สวิตช์ที่1 เป็นการกดวิ่งไฟเรียงLEDทั้ง4ตัว พร้อมแสดงผลหน้าจอว่า LED :) RUNNING!  
กดอีกครั้งเพื่อเป็นการหยุดไฟที่ตัวนั้นๆ พร้อมแสดงผลหน้าจอว่า LED :( STOP  
สามารถกดอีกครั้งหนึ่งเพื่อให้ไฟวิ่งต่อได้
- สวิตช์ที่2 เป็นการกดเพื่อแสดงไฟกระพริบ พร้อมแสดงผลหน้าจอว่า LED :) BLINK!  
กดอีกครั้งเพื่อหยุดการกระพริบได้ จะแสดงผลหน้าจอ LED :( STOP
- สวิตช์ที่3 กดเพื่อเป็นการRESET ค่าให้กลายเป็น0 คือจะแสดงผลตอนแรกที่ยังไม่ได้กดสวิตช์นั่นเอง  
ในระหว่างที่กดสวิตช์ที่1 หรือ2 สามารถกดสวิตช์นี้(3) เพื่อ RESET ค่ากลับไปเป็นตอนแรกได้เลย

## โปรแกรมควบคุม Hardware

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27,16,2);
void setup() {
    lcd.init();
    lcd.backlight();
    lcd.setCursor(5,0);
    lcd.print("LCD ON");
    lcd.setCursor(1,1);
    lcd.print("Press to start ");

    Wire.beginTransmission(0x21);
    Wire.write(0x06);
    Wire.write(0xE0);
    Wire.endTransmission();
    Serial.begin(115200);

    Wire.beginTransmission(0x21);
    Wire.write(0x00);
    Wire.write(0xE0);
    Wire.endTransmission();
    Serial.begin(115200);
    numwrite(0x00);
}

bool x =false;
bool y=false;
char num[]={0x02,0x04,0x08,0x10};
int i=0;
```

```
void loop() {
    if((readsw(0x21)&0x20)==0){
        while((readsw(0x21)&0x20)==0);
        if(x==false){
            lcd.init();
            lcd.backlight();
            lcd.setCursor(5,0);
            lcd.print("LED :)");
            lcd.setCursor(4,1);
            lcd.print("RUNNING!");
            x=true;
        }else{lcd.init();
            lcd.backlight();
            lcd.setCursor(5,0);
            lcd.print("LED :)");
            lcd.setCursor(6,1);
            lcd.print("STOP");
            x=false;}
    }

    if((readsw(0x21)&0x40)==0){
        while((readsw(0x21)&0x40)==0);
        if(y==false){
            lcd.init();
            lcd.backlight();
            lcd.setCursor(5,0);
            lcd.print("LED :)");
            lcd.setCursor(5,1);
            lcd.print("BLINK!");
            y=true;
        }else{
            lcd.init();
            lcd.backlight();
            lcd.setCursor(5,0);
```

```

    lcd.print("LED :(");
    lcd.setCursor(6,1);
    lcd.print("STOP");
    y=false;}
}
if((readsw(0x21)&0x80)==0){
while((readsw(0x21)&0x80)==0);
    lcd.init();
    lcd.backlight();
    lcd.setCursor(5,0);
    lcd.print("LED ON");
    lcd.setCursor(1,1);
    lcd.print("Press to start");
    numwrite(0x00);
    delay (100);
    x=false;
    y=false;
}
if(x==true){
    if(i<4){
        numwrite(num[i]);
        i++;
        delay(100);
    }else i=0;
}
if(y==true){
    numwrite(0x1E);
    delay (100);
    numwrite(0x00);
    delay (100);
}else{y=false;}
}

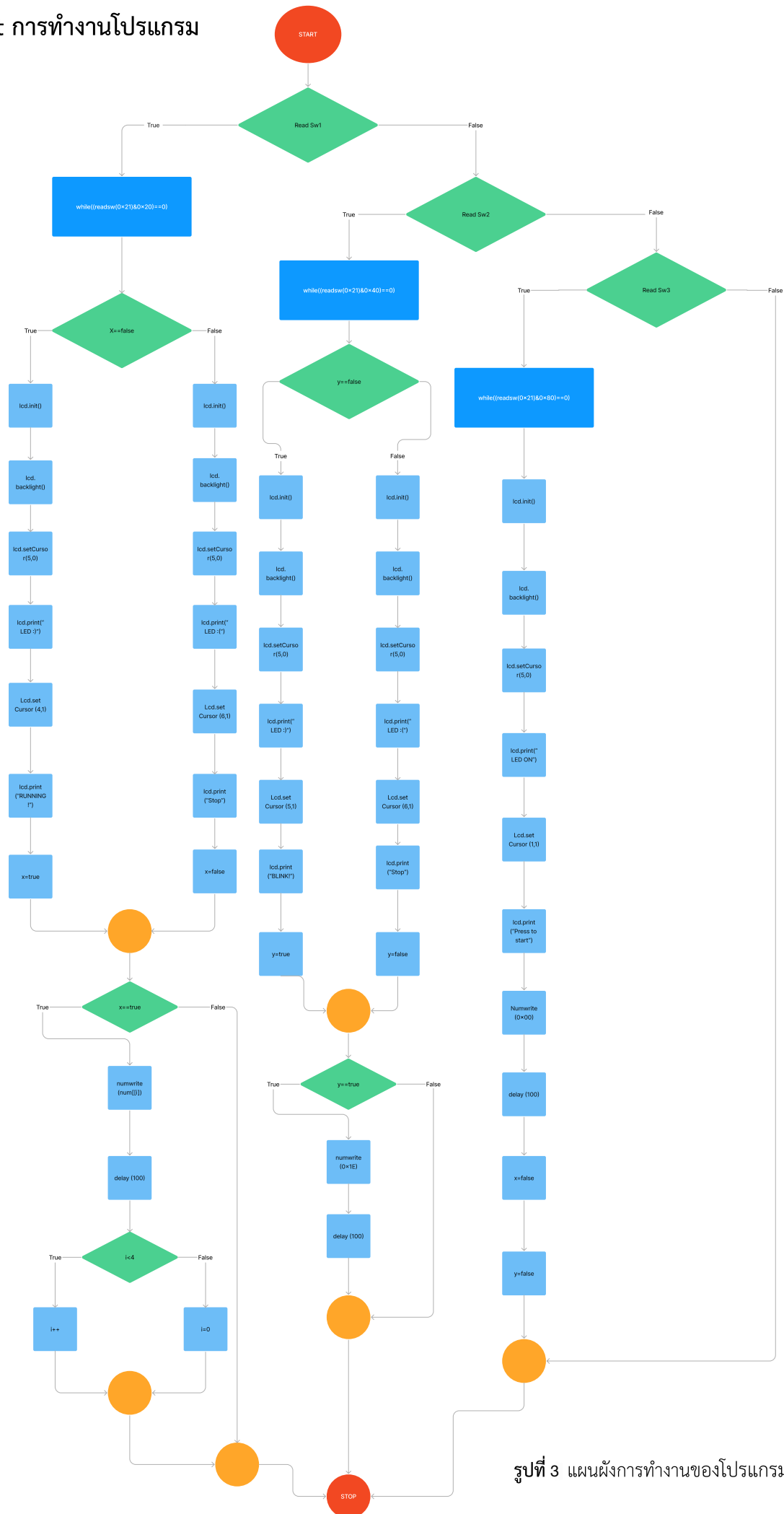
```

```

void numwrite(char num){
    Wire.beginTransmission(0x21);
    Wire.write(0x09);
    Wire.write(num);
    Wire.endTransmission();
}
unsigned char readsw(char address){
    Wire.beginTransmission(address);
    Wire.write(0x09);
    Wire.endTransmission();
    Wire.requestFrom(address,1);
    return(Wire.read());
}

```

## flowchart การทำงานโปรแกรม



รูปที่ 3 แผนผังการทำงานของโปรแกรม

## ผลการทดลอง Hardware ขณะทำงานและทดสอบ

- เชื่อมต่อโปรแกรมหน้าจอพร้อมแสดงผลหน้าจอว่า LCD ON Press to start



รูปที่ 4 ฮาร์ดแวร์ทดสอบระบบ ขณะเชื่อมต่อโปรแกรม

- กดสวิตช์ที่1 ไฟวิ่งเรียงLEDทั้ง4ตัว พร้อมแสดงผลหน้าจอว่า LED :) RUNNING! และเมื่อกดสวิตช์ที่ 1 อีกครั้ง ไฟledจะหยุดวิ่งที่ตัวนั้นๆ พร้อมแสดงผลหน้าจอว่า LED :( STOP และเมื่อกดอีกครั้งก็จะวิ่งต่อ



รูปที่ 5.1 ฮาร์ดแวร์ทดสอบระบบ ขณะกดสวิตซ์ที่ 1



รูปที่ 5.2 ฮาร์ดแวร์ทดสอบระบบ ขณะกดสวิตซ์ที่ 1 อีกครั้ง

- กดสวิตช์ที่2 ไฟกระพริบทั้ง4ตัว พร้อมแสดงผลหน้าจอว่า LED :) BLINK! และเมื่อกดสวิตซ์ที่2อีกครั้ง ไฟledจะหยุดกระพริบ พร้อมแสดงผลหน้าจอว่า LED :( STOP และเมื่อกดอีกครั้งก็จะกระพริบต่อ



รูปที่ 6.1 ฮาร์ดแวร์ทดสอบระบบ ขณะกดสวิตซ์ที่ 2



รูปที่ 6.2 ฮาร์ดแวร์ทดสอบระบบ ขณะกดสวิตซ์ที่ 2 อีกครั้ง

- สวิตซ์ที่3 กดเพื่อเป็นการRESET พร้อมแสดงผลหน้าจอว่า LCD ON Press to start ซึ่งเป็นค่าเริ่มต้น



รูปที่ 7 ฮาร์ดแวร์ทดสอบระบบ ขณะกดสวิตซ์ที่ 3

Address	Switch 1	Switch 2	Switch 3
0X20	on	on	on
0X21	off	on	on
0X22	on	off	on
0X23	off	off	on

Address	Switch 1	Switch 2	Switch 3
0X24	on	on	off
0X25	off	on	off
0X26	on	off	off
0X27	off	off	off