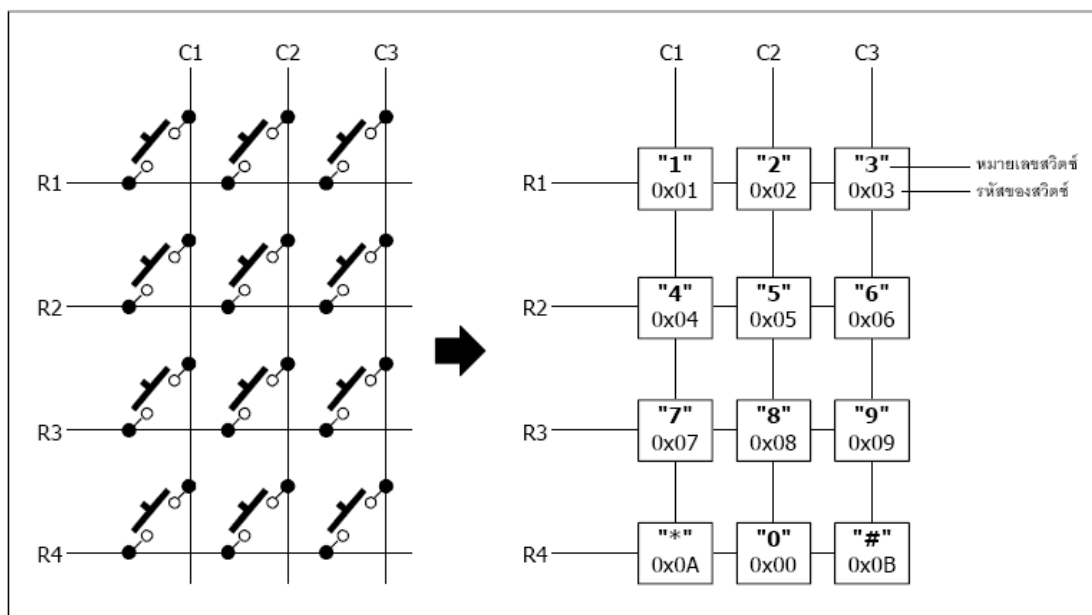


## ใบงานที่ 5 การอ่านค่าสวิตช์เมตริกซ์ 4x3 แสดงผลด้วย 7-Segment

### เนื้อหา

สวิตช์เมตริกซ์(matrix switch) หรือคีย์แพด(keypad) ถูกนำมาใช้เป็นอุปกรณ์ในการป้อนข้อมูลให้กับงานทางด้านไมโครคอนโทรลเลอร์ นอกเหนือจากสวิตช์แบบ กดติดปล่อยดับแบบธรรมดา (push button switch) โดยเฉพาะกับงานที่ต้องมีการป้อนข้อมูลทั้งตัวอักษรและตัวเลขเป็นจำนวนมาก จะพบว่าสวิตช์เมตริกซ์ถูกเลือกมาใช้งานเสมอ ที่เห็นได้ในชีวิตประจำวัน เช่น คีย์กดตัวเลขของระบบโทรศัพท์ เตลอบไมโครเวฟ เป็นต้น

การต่อใช้งานสวิตช์แบบเมตริกซ์ เป็นการนำสวิตช์ธรรมดามาต่อกันในแบบเมตริกซ์ คือ ขาด้านหนึ่งจะต่อในแนวหลัก(column) และขาด้านหนึ่งจะต่ออยู่ในแนวแถว(row) แสดงดังรูปที่ 5.1



รูปที่ 5.1 วงจร สวิตช์เมตริกซ์ ขนาด 4x3

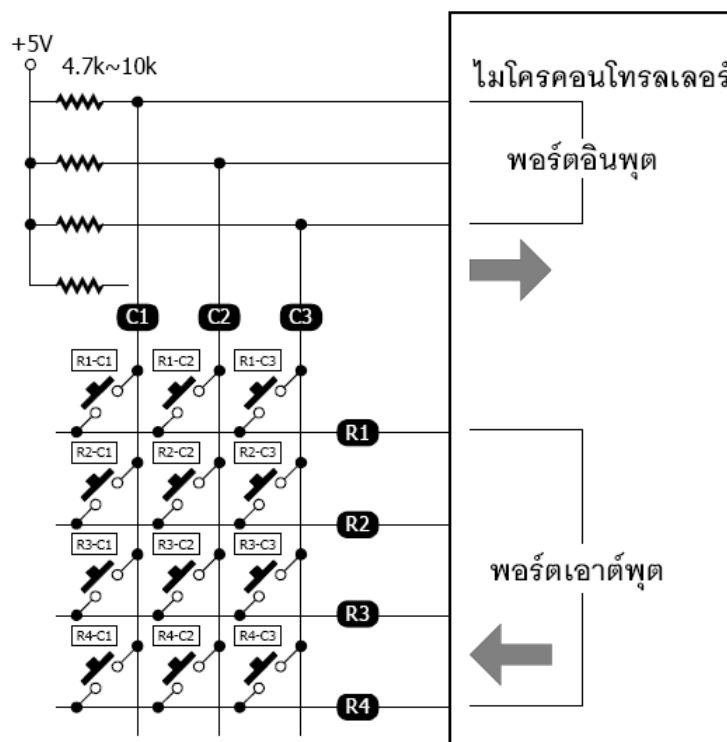
หลักในการอ่านค่าจากคีย์แพดนี้คือ จะต้องกำหนดรหัสประจำตำแหน่งของสวิตช์แต่ละตัวไว้ไม่ให้ซ้ำกัน ดังนั้นเมื่อสวิตช์ตัวใดถูกกดก็จะได้ค่ารหัสของสวิตช์ตัวดังกล่าวออกมา ซึ่งจะกำหนดค่าลงไปให้กับสวิตช์แต่ละตัวไว้ดังนี้

สวิตช์ตำแหน่ง R1 (row 1 หรือแถวที่ 1), C1 (column 1 หรือหลักที่ 1) มีค่า 0x01

สวิตช์ตำแหน่ง R1 (row 1 หรือแถวที่ 1), C2 (column 2 หรือหลักที่ 2) มีค่า 0x02

สวิตช์ตำแหน่ง R1 (row 1 หรือแถวที่ 1), C3 (column 3 หรือหลักที่ 3) มีค่า 0x03

สวิตช์ตำแหน่ง R2 (row 2 หรือแถวที่ 2), C1 (column 1 หรือหลักที่ 1) มีค่า 0x04  
 สวิตช์ตำแหน่ง R2 (row 2 หรือแถวที่ 2), C2 (column 2 หรือหลักที่ 2) มีค่า 0x05  
 สวิตช์ตำแหน่ง R2 (row 2 หรือแถวที่ 2), C3 (column 2 หรือหลักที่ 3) มีค่า 0x06  
 สวิตช์ตำแหน่ง R3 (row 3 หรือแถวที่ 3), C1 (column 1 หรือหลักที่ 1) มีค่า 0x07  
 สวิตช์ตำแหน่ง R3 (row 3 หรือแถวที่ 3), C2 (column 2 หรือหลักที่ 2) มีค่า 0x08  
 สวิตช์ตำแหน่ง R3 (row 3 หรือแถวที่ 3), C3 (column 2 หรือหลักที่ 3) มีค่า 0x09  
 สวิตช์ตำแหน่ง R4 (row 4 หรือแถวที่ 4), C1 (column 1 หรือหลักที่ 1) มีค่า 0x10  
 สวิตช์ตำแหน่ง R4 (row 4 หรือแถวที่ 4), C2 (column 2 หรือหลักที่ 2) มีค่า 0x00  
 สวิตช์ตำแหน่ง R4 (row 4 หรือแถวที่ 4), C3 (column 2 หรือหลักที่ 3) มีค่า 0x11

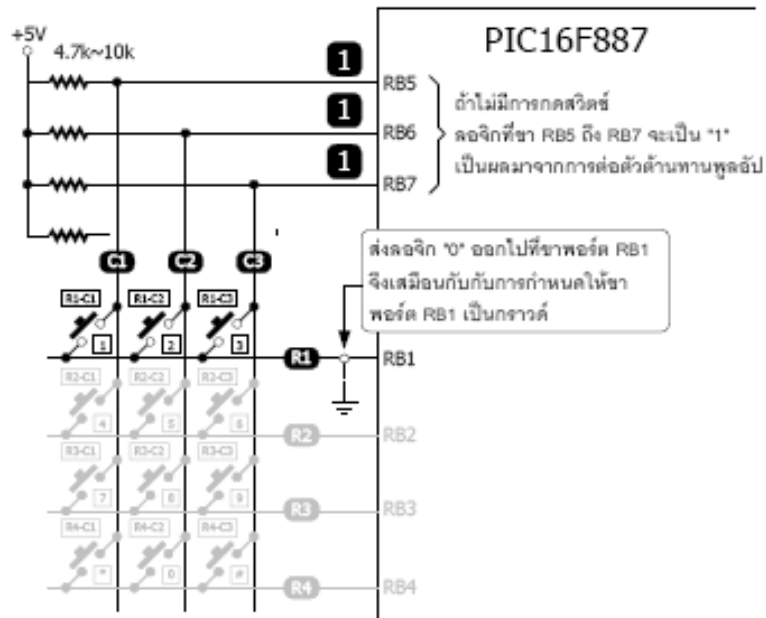


รูปที่ 5.2 การต่อวงจรสวิตช์เมตริกซ์ เข้ากับ ไมโครคอนโทรลเลอร์

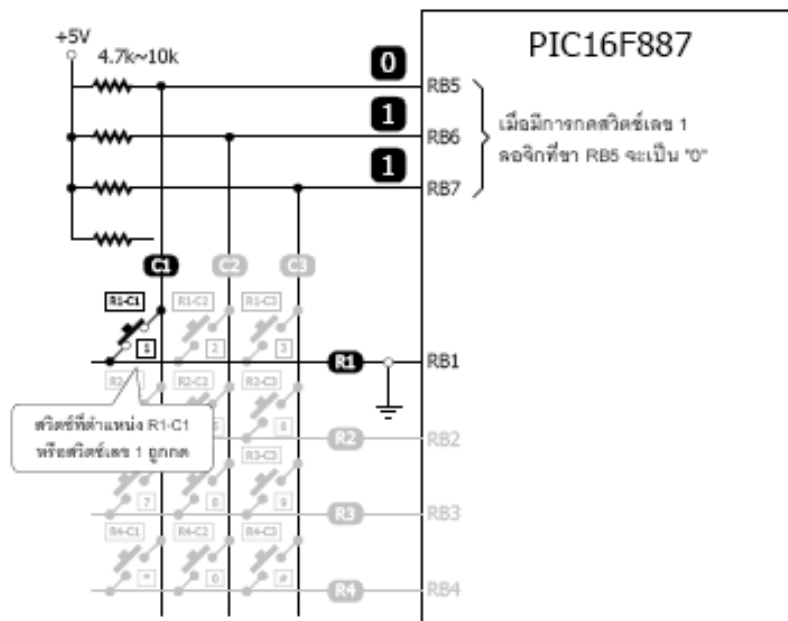
ต้องการอ่านค่าสวิตช์เมตริกซ์ในตำแหน่ง 0 ถึง 9 ออกมาแสดงผลที่ LED ตัวเลข 7 ส่วน วิธีการอ่านค่าจากสวิตช์สรุปได้ดังนี้

- (1) กำหนดให้ ขาพอร์ต RB5 ถึง RB7 เป็นพอร์ตอินพุตดิจิตอล (ดูรูปที่ 5.3ประกอบ)
- (2) กำหนดให้ ขาพอร์ต RB1 ถึง RB4 เป็นพอร์ตเอาต์พุตดิจิตอล (ดูรูปที่ 5.3ประกอบ)
- (3) ส่งลอจิก “1” ออกไปยังขาพอร์ต RB1 ถึง RB4
- (4) จากนั้นเริ่มต้นการตรวจสอบแถวที่ 1 ด้วยการทำให้ ขา RB1 เป็น “0” แล้วอ่านค่าจากขาพอร์ต RB5 ถึงRB7 ว่าขาใดเป็นลอจิก “0” หรือไม่ หากขาใดเป็นลอจิก “0” นั่นคือเกิดการกด

สวิตช์ที่ขานั้น หากขา RB5 เป็น “0” นั่นคือเกิดการกดสวิตช์ที่ตำแหน่ง R1, C1 ซึ่งก็คือ สวิตช์เลข 1 โปรแกรมจะคืนค่าเป็น 0x01 กลับมา การวนรอ่านค่านี้ จะใช้เวลา 600 มิลลิวินาที หากไม่มีการกดสวิตช์เลยจะเปลี่ยนการตรวจสอบไปยังแถวที่ 2



(ก) สถานะของขาพอร์ตเมื่อเริ่มต้นการตรวจสอบการกดสวิตช์เมตริกซ์แถวที่ 1



(ข) สถานะของขาพอร์ตเมื่อมีการกดสวิตช์เมตริกซ์ตำแหน่ง C1-R1

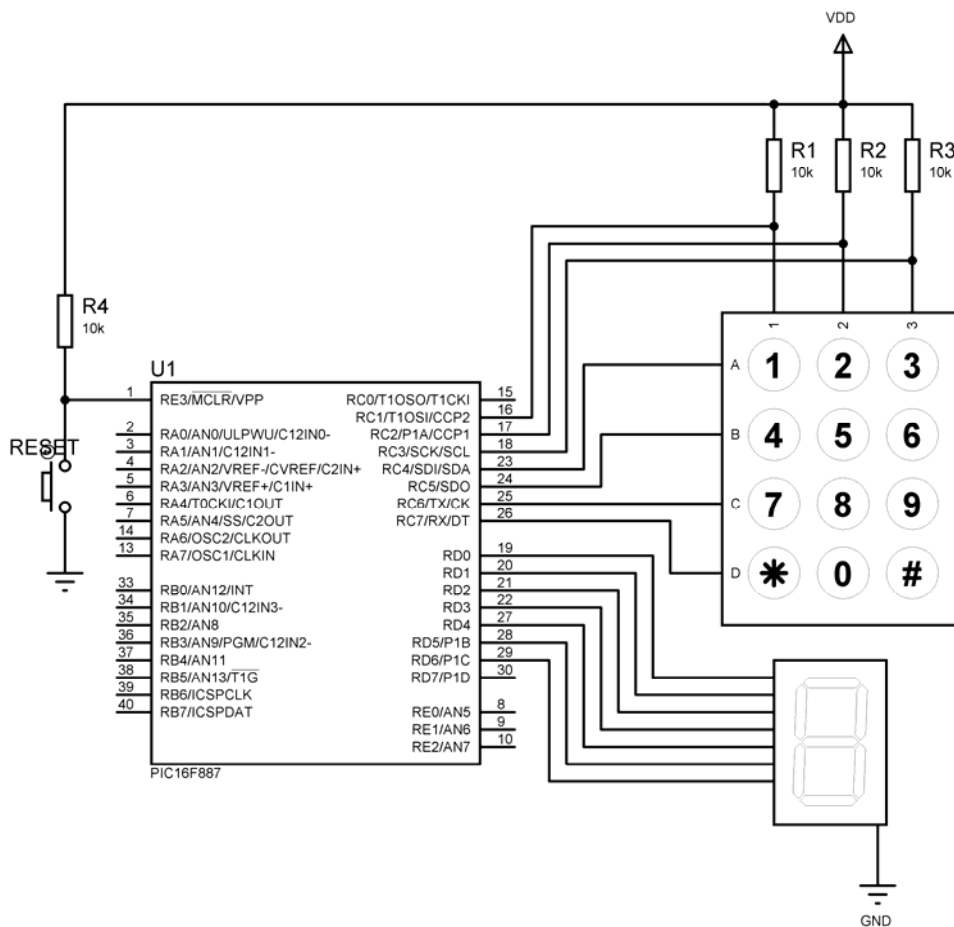
รูปที่ 5.3 การตรวจสอบสถานะการทำงานของสวิตช์เมตริกซ์

(5) การตรวจสอบแฉวที่ 2 จะเกิดขึ้นเมื่อทำให้ขา RB1 กลับมาเป็น “1” แล้วทำให้ขา RB2 เป็น “0” จากนั้นรอ่านค่าจากขาพอร์ต RB5 ถึง RB7 เช่นเดิม หากในคราวนี้ ที่ ขา RB5 เป็น “0” นั้น หมายถึงเกิดการกดสวิตซ์ที่ตำแหน่ง R2, C1 ซึ่งก็คือสวิตซ์ เลข 4 โปรแกรมจะคืนค่าเป็น 0x04 กลับมา หากไม่มีการกดสวิตซ์ เลข จะเปลี่ยนการตรวจสอบไปยังแฉวที่ 3 และ 4 แล้ววนกลับมา ยังแฉวที่ 1 อีกครั้ง

(6) เมื่อการตรวจสอบมาถึงแฉวที่ 4 จะพบว่า ต้องมีการตรวจสอบสวิตซ์ 3 ตัวคือ \*, 0 และ # ดังนั้นค่าที่ได้ จากการตรวจสอบการกดสวิตซ์ จะเกิดได้ 3 ค่าคือ 10 หรือ 0x0A, 0 หรือ 0x00 และ 11 หรือ 0x0B เราจะสนใจเพียงค่า 0x00 เท่านั้น จึงต้องมี การเขียนโปรแกรมเพื่อ อกรองเอา เฉพาะข้อมูลเลข 0 ถึง 9 อีกชั้นหนึ่ง

(7) เมื่อได้ค่าของสวิตซ์แล้ว จึงนำค่าดังกล่าวไปใช้งานต่อได้ อาทิ นำข้อมูลไปใช้ ขับ LED ธรรมดาหรือ LED ตัวเลข 7 ส่วน หรือนำไปแสดงที่โมดูล LCD นำไปใช้ ประมวลผลด้าน เกี่ยวกับข้อมูลรหัสตัวเลขก็ได้

1. จงต่อวงจรตามรูปข้างล่างนี้ เพื่อใช้ในการทดลอง



รูปที่ 5.4 วงจรอ่านค่าสวิตช์เมตริกซ์

2. ให้นักศึกษาเขียนโปรแกรมตามตัวอย่างข้างล่างนี้
  - 2.1 ให้นักศึกษาสร้างไฟล์โปรเจกต์ ชื่อ “lab\_51\_รหัส3ตัวหลัง” บันทึกชื่อไฟล์เป็น “lab\_51\_รหัส3ตัวหลัง.c”

```
#include <pic.h>

#define _XTAL_FREQ 2000000

__CONFIG(HS & LVPDIS & WDTDIS);

const unsigned char number[10]=
{0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F};

void Delay_ms(unsigned int tick)
{
    while(tick--)
```

```

    {
        __delay_ms(1);
    }
}
void unsigned char keypad()
{
    unsigned char kp;
    TRISC1=1;
    TRISC2=1;
    TRISC3=1;
    TRISC4=0;
    TRISC5=0;
    TRISC6=0;
    TRISC7=0;
    RC4=1;
    RC5=1;
    RC6=1;
    RC7=1;

    // scan ROW1
    RC4=0;
    if(RC1==0)
    {
        kp=1;
        Delay_ms(200);
    }
    else if(RC2==0)
    {
        kp=2;
        Delay_ms(200);
    }
    else if(RC3==0)

```

```
{
kp=3;
Delay_ms(200)
}
RC4=1;

// scan ROW2
RC5=0;
if(RC1==0)
{
kp=4;
Delay_ms(200);
}
else if(RC2==0)
{
kp=5;
Delay_ms(200);
}
else if(RC3==0)
{
kp=6;
Delay_ms(200)
}
RC5=1;

// scan ROW3
RC6=0;
if(RC1==0)
{
kp=7;
Delay_ms(200);
}
```

```
else if(RC2==0)
{
kp=8;
Delay_ms(200);
}
else if(RC3==0)
{
kp=9;
Delay_ms(200)
}
RC6=1;

// scan ROW4
RC7=0;
if(RC1==0)
{
kp=10;
Delay_ms(200);
}
else if(RC2==0)
{
kp=11;
Delay_ms(200);
}
else if(RC3==0)
{
kp=12;
Delay_ms(200)
}
RC7=1;
return kp;
}
```



```

void main()
{
    TRISD=0x00;
    PORTD=0x00;
    unsigned char key;
    while(1)
    {
        key=keypad();
        PORTD=number[key];
    }
}

```

2.2 ให้นักศึกษาต่อวงจรทดลองตามรูปที่ 5.4 แล้วดาวน์โหลดโปรแกรมลงในหน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์

2.3 อธิบายการทำงานของวงจรและโปรแกรม ควรแก้ไขปรับปรุงในส่วนใดบ้าง

.....

.....

.....

.....

.....

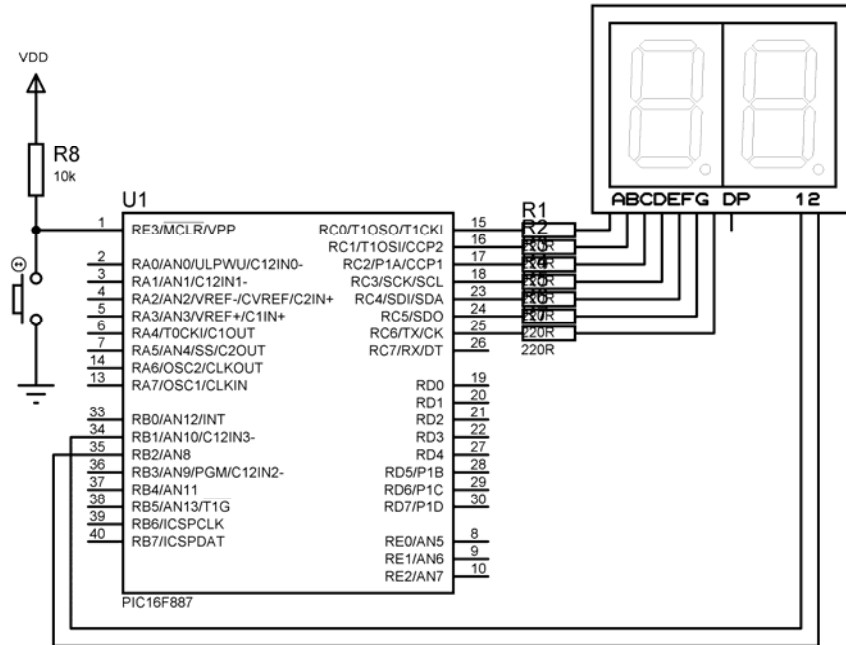
.....

.....

.....



4. จากรูปข้างล่างให้นักศึกษาออกแบบวงจรเพิ่มเติมโดยต่อกีย์แพด และเขียนโปรแกรมเพื่อควบคุมการทำงาน 7 segment 2 หลัก



.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Lined writing area consisting of 28 horizontal dotted lines.