

Lab 3HB05 AVR Microcontroller

242-302 ADVANCED COMPUTER
ENGINEERING LABORATORY II

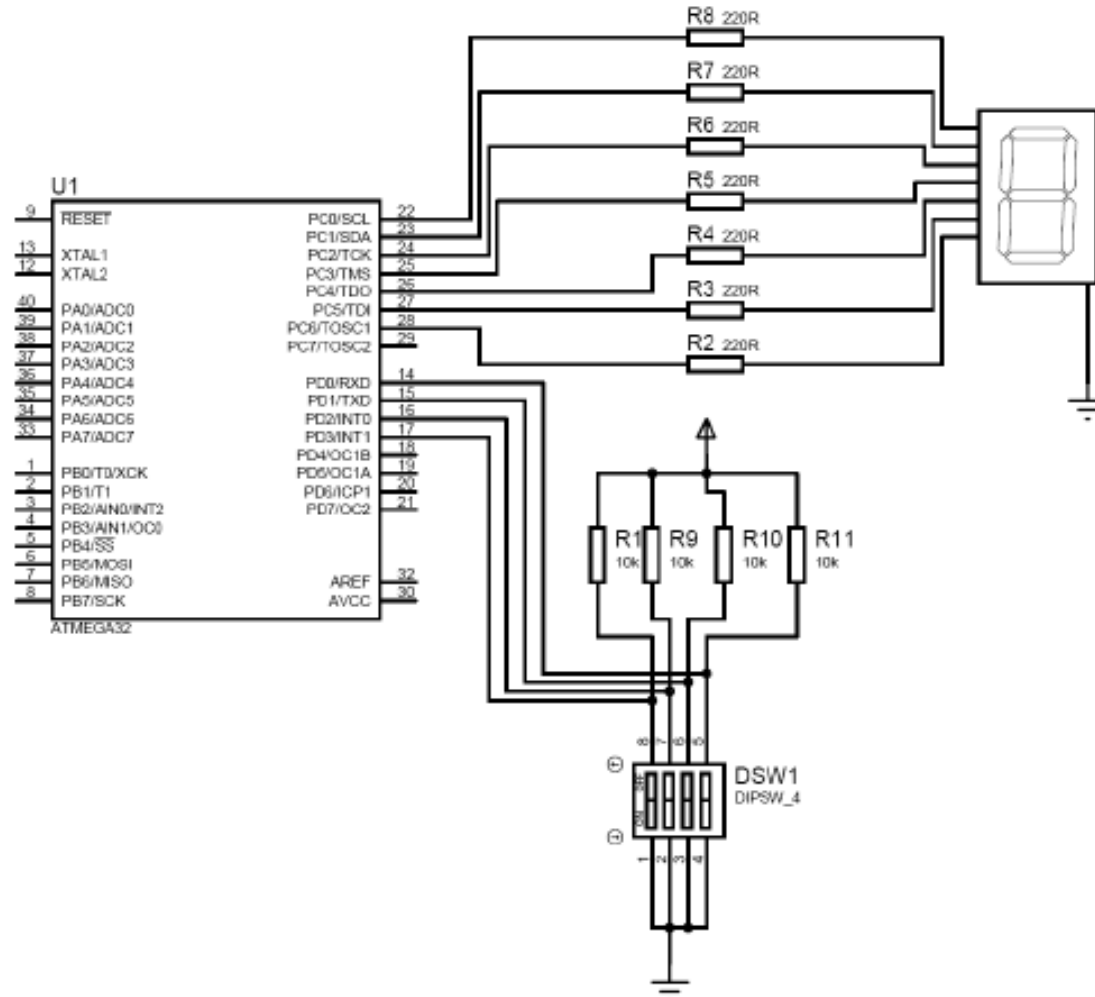
วัตถุประสงค์

- เพื่อให้นักศึกษาได้เรียนรู้วิธีการเขียนโปรแกรมประยุกต์ใช้งาน ไมโครคอนโทรลเลอร์ AVR ด้วยภาษา Assembly และ C
- เพื่อให้นักศึกษาได้เรียนรู้เทคนิคการดีบั๊กโปรแกรมภาษา Assembly และ C ก่อนที่จะนำโปรแกรมลงทดสอบบนฮาร์ดแวร์จริง
- เพื่อให้นักศึกษาได้เรียนรู้เทคนิคการออกแบบฮาร์ดแวร์และซอฟต์แวร์ร่วมกัน เพื่อให้ง่ายในการทดสอบความถูกต้อง
- เพื่อให้นักศึกษาได้ฝึกการเขียนซอฟต์แวร์สำหรับการทดสอบความถูกต้องของฮาร์ดแวร์

เป้าหมาย

- นักศึกษาสามารถใช้โปรแกรม AVRStudio ในการดีบั๊ก โปรแกรมภาษา Assembly และ C ของชิพยี่ห้อ AVR ได้อย่างมีประสิทธิภาพ
- นักศึกษาสามารถใช้โปรแกรม Proteus ในการทดสอบการทำงานร่วมกันของฮาร์ดแวร์และซอฟต์แวร์ได้อย่างมีประสิทธิภาพ ก่อนที่จะนำซอฟต์แวร์ไปทดสอบการทำงานบนฮาร์ดแวร์จริง

การทดลองที่ 1 เขียนโปรแกรมเพื่ออ่านค่าจากสวิตช์ และแสดงผลทาง 7-segment LED



รูปที่ 1.9 จากซีทแลบ 3HB05 หน้า 14

Assembly

```
.INCLUDE    "m32def.inc"
```

```
.EQU        ALL_PIN_OUT = 0xFF
```

```
.EQU        ALL_PIN_IN = 0x00
```

```
.DEF        VAR_A = R16
```

```
.DEF        TMP = R17
```

```
.CSEG
```

```
.ORG        0x0000
```

Assembly

```
ldi    VAR_A, ALL_PIN_OUT  
out    DDRC, VAR_A
```

```
ldi    VAR_A, ALL_PIN_IN  
out    DDRD, VAR_A
```

```
ldi    TMP, 0x00
```

MAIN:

```
in     VAR_A, PIND  
andi   VAR_A, 0x0F
```

Assembly

```
ldi    ZL, low(TB_7SEGMENT*2)
ldi    ZH, high(TB_7SEGMENT*2)

add     ZL, VAR_A
adc     ZH, TMP

lpm

out     PORTC, R0
rjmp    MAIN
```

Assembly

--a--
f b
--g--
e c
--d--

; hgfedcba hgfedcba

TB_7SEGMENT:

.DB	0b00111111,	0b00000110	; 0 and 1
.DB	0b01011011,	0b01001111	; 2 and 3
.DB	0b01100110,	0b01101101	; 4 and 5
.DB	0b01111101,	0b00000111	; 6 and 7
.DB	0b01111111,	0b01101111	; 8 and 9
.DB	0b01110111,	0b01111100	; A and B
.DB	0b00111001,	0b01011110	; C and D
.DB	0b01111001,	0b01110001	; E and F

.DSEG

C Programming

```
#include <avr/io.h>
```

```
int main(void)
```

```
{
```

```
    unsigned char TB7SEG[] = {  
        0b00111111, 0b00000110, 0b01011011, 0b01001111,  
        0b01100110, 0b01101101, 0b01111101, 0b00000111,  
        0b01111111, 0b01101111, 0b01110111, 0b01111100,  
        0b00111001, 0b01011110, 0b01111001, 0b01110001};
```

```
    unsigned char DISPLY;
```

```
    unsigned char SWITCH;
```

```
    DDRC = 0xFF;
```

```
    DDRD = 0x00;
```

C Programming

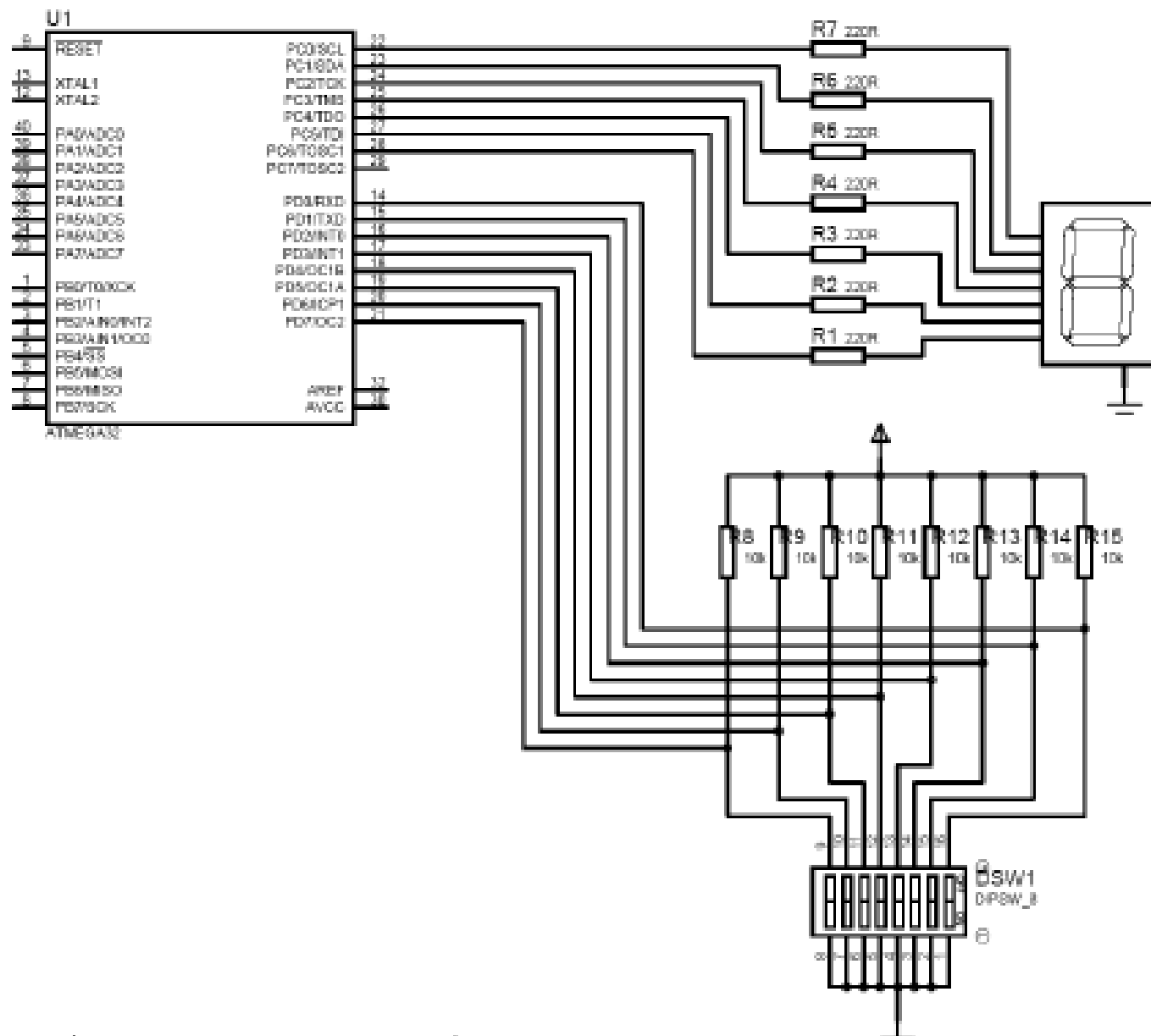
```
while(1)
{
    SWITCH = PIND;
    SWITCH &= 0x0F;
    DISPLY = TB7SEG[SWITCH];
    PORTC = ~DISPLY;
}
}
```

วิธีการทดลอง

1. ทำการ compile และ debug โปรแกรมทั้งภาษา Assembly และ ภาษา C
2. ทดลองอัดโปรแกรม และสังเกตผลการทำงาน
3. บันทึกผลการทดลอง ตาราง Checkpoint หน้า 16

การทดลองที่ 2 เขียนโปรแกรม เพื่อนับลอจิกต่ำ จากดิปสวิทช์

โปรแกรมจะทำการอ่านค่าจากดิปสวิทช์จำนวน 8 ตัว ที่
ต่อกับพอร์ต D จากนั้น ทำการตรวจสอบบิตที่มาค่า
ลอจิกต่ำ และแสดงผลออก 7-segment ที่ต่อที่พอร์ต C



รูปที่ 1.12 จากซีทแลบ 3HB05 หน้า 17

C Programming

```
#include <avr/io.h>
```

```
int main(void)
```

```
{
```

```
    unsigned char TB7SEG[] = {  
        0b00111111, 0b00000110, 0b01011011, 0b01001111,  
        0b01100110, 0b01101101, 0b01111101, 0b00000111,  
        0b01111111, 0b01101111, 0b01110111, 0b01111100,  
        0b00111001, 0b01011110, 0b01111001, 0b01110001};
```

```
    unsigned char DISPLY;
```

```
    unsigned char SWITCH;
```

```
    unsigned char mask, i, temp, count;
```

```
    DDRC = 0xFF;
```

```
    DDRD = 0x00;
```

C Programming

```
while(1)
{
    SWITCH = PIND;
    mask = 0b00000001;
    count = 0;
    for(i=0;i<8;i++)
    {
        temp = SWITCH & (mask<<i);
        if(temp)
            count++;
    }
    count = 8 - count;
    DISPLY = TB7SEG[count];
    PORTC = ~DISPLY;
}
}
```

วิธีการทดลอง

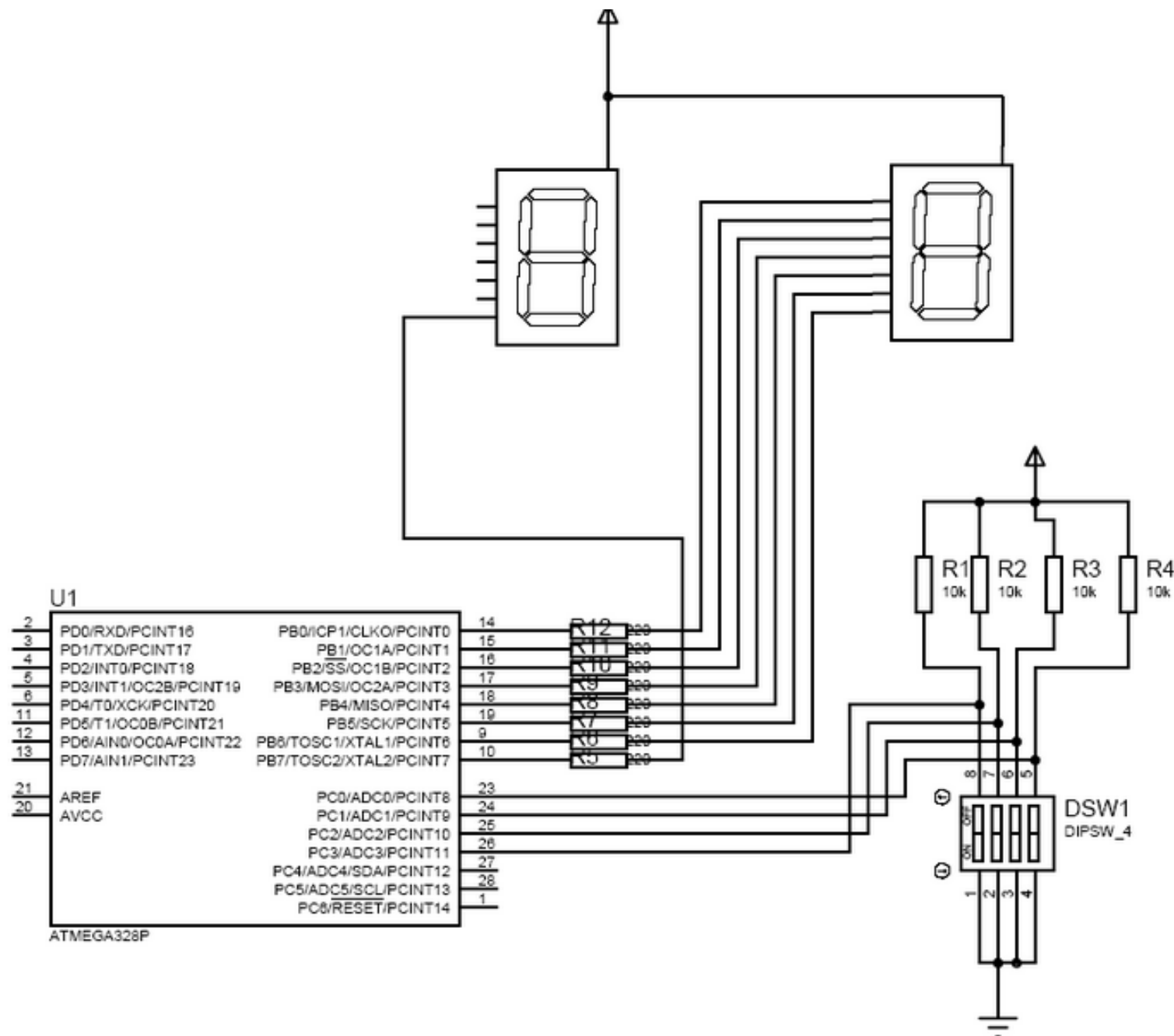
1. ทดลองอัดโปรแกรม และสังเกตผลการทำงาน
2. บันทึกผลการทดลอง ตาราง Checkpoint หน้า 19

การทดลองที่ 3 เขียนโปรแกรม เพื่ออ่านค่าจาก สวิตช์ โดยข้อมูลเป็นเลข signed number

จงเขียนโปรแกรมเพื่อแปลงค่าจากสวิตช์เป็นเลขฐานสิบแบบ
มีเครื่องหมาย ซึ่งมีค่าอยู่ระหว่าง $-8 \dots +7$ ออกแสดงผลทาง
แอลอีดี แบบ 7 เซกเมนต์ 2 ตัว

จากรูป กำหนดให้

สวิตช์ต่อกับ PC0-PC3 และ 7-Segment ต่อกับ PD0-PD7



วิธีการทดลอง

1. เขียนโปรแกรม
2. ทดลองอัดโปรแกรม และสังเกตผลการทำงาน
3. Checkpoint

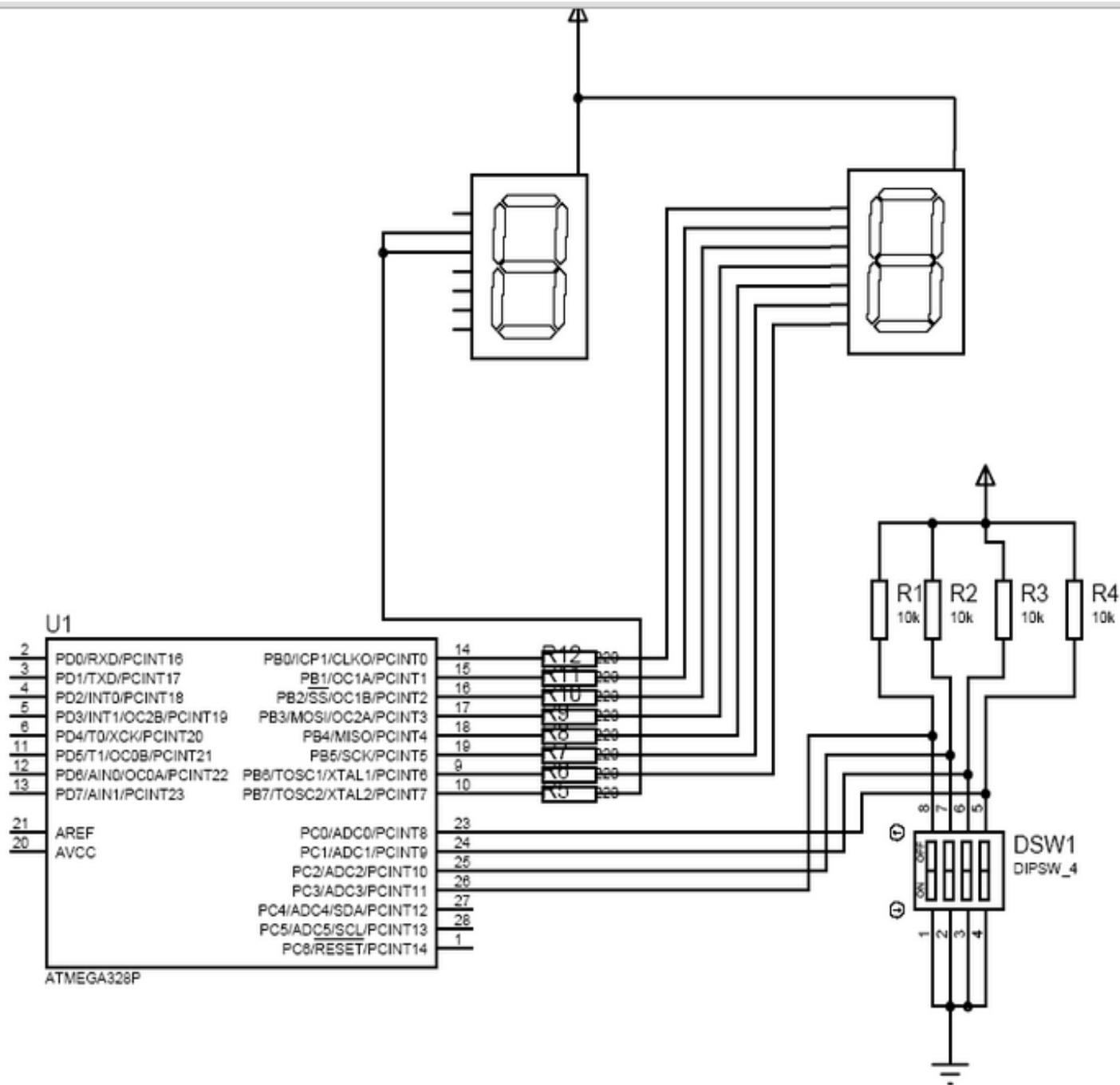
เขียนโปรแกรมโดยไม่แก้ไขตาราง 7-segment

การทดลองที่ 4 เขียนโปรแกรม เพื่อแสดงค่าจาก สวิตช์ โดยข้อมูลเป็นเลขฐานสิบ

จงเขียนโปรแกรมเพื่อแปลงค่าจากสวิตช์เป็นเลขฐานสิบแบบ
ไม่มีเครื่องหมาย ซึ่งมีค่าอยู่ระหว่าง 0-15 ออกแสดงผลทาง
แอลอีดี แบบ 7 เซกเมนต์ 2 ตัว

จากรูป กำหนดให้

สวิตช์ต่อกับ PC0-PC3 และ 7-Segment ต่อกับ PD0-PD7



วิธีการทดลอง

1. เขียนโปรแกรม
2. ทดลองอัดโปรแกรม และสังเกตผลการทำงาน
3. Checkpoint

เขียนโปรแกรมโดยไม่แก้ไขตาราง 7-segment