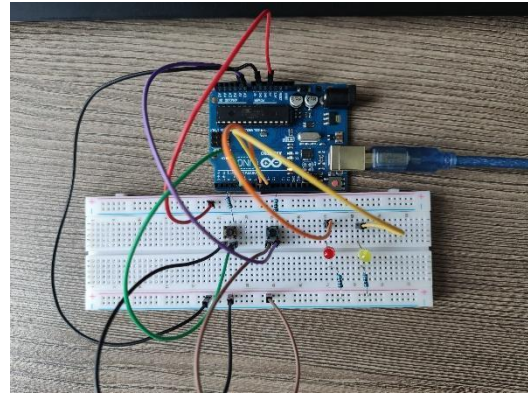
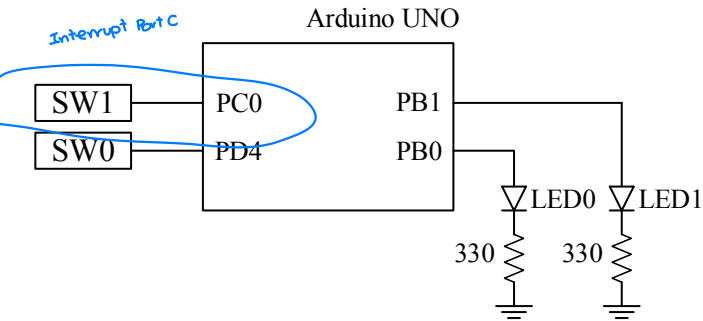


Exercise 3: (Using the pin change interrupt). We have connected a circuit as shown below.



ကနဦးပိတ်
မလုပ်ပိတ်

ကနဦးပိတ်
မလုပ်ပိတ်

ကနဦးပိတ်
မလုပ်ပိတ်

We would like to write an Assembly program to do the following tasks.

- Task1 – The microcontroller monitors the status of the switch SW0: if we press the switch SW0, LED0 is on; if we do not press the switch SW0, LED0 is off.
- Task2 – The microcontroller monitors the status of the switch SW1: if we **press and release** the switch SW1, the LED1 will be toggled (on $\leftrightarrow$ off or off $\leftrightarrow$ on).

As a result, the microcontroller will do Task1 in the main program and do Task2 by using the pin change interrupt (via the switch SW1 connected to the pin PC0). Answer the following questions.

- a) To use the pin change interrupt as specified above, what should be the values of the following registers?

PCICR	=	0x02
PCMSK0	=	0x00
PCMSK1	=	0x01
PCMSK2	=	0x00

- b) Complete the following Assembly program.

```

1  .ORG 0x0
2  JMP MAIN
3  .ORG 0x08
4  JMP PCINT8_ISR
5

```

```

6 ;Start the main program
7 .ORG 0x100
8 MAIN: LDI R16, HIGH(RAMEND)
9 OUT SPH, R16
10 LDI R16, LOW(RAMEND)
11 OUT SPL, R16
12
13 CALL PIN_SETUP ;Set up pin modes
14 LDI R16, (1<<PCIE1)
15 STS PCICR, R16 ;Enable pin change interrupt PORTC
16 LDI R16, 0x01
17 STS PCMSK1, R16 ;Enable interrupt from pin PC0
18 SEI ;Enable the global interrupt
19 LDI R20, 2 ;A dummy variable used in ISR
20
21 LOOP: SBIC PIND, 4 ;Main task
22 RJMP L_OFF
23 RJMP L_ON
24 L_OFF: CBI PORTB, 0 ;Turn LED0 off
25 RJMP LOOP
26 L_ON: SBI PORTB, 0 ;Turn LED0 on
27 RJMP LOOP
28 ;End of the main program
29

```

```

30 PIN_SETUP: ;Subroutine to set up pin modes
31 SBI DDRB, 0 ;Set the pin PB0 as the output -> LED0
32 CBI PORTB, 0 ;PB0=0
33 SBI DDRB, 1 ;Set the pin PB1 as the output -> LED1
34 CBI PORTB, 1 ;PB1=0
35 CBI DDRD, 4 ;Set the pin PD4 as the input
36 SBI PORTD, 4 ;Set pull-up resistor on PD4
37 SBI PORTC, 0 ;Set pull-up resistor on PC0
38 RET ;Return to the main program
39

```

```

40 .ORG 0x200
41 PCINT8_ISR: ;ISR of the PORTC Pin Change Interrupt
42 DEC R20
43 BRNE HERE
44 IN R17, PORTB ;Read PORTB
45 LDI R18, (1<<1) ;R18=0b00000010
46 EOR R17, R18
47 OUT PORTB, R17 ;Toggle PB1
48 LDI R20, 2
49 HERE: RETI ;Return to the main program

```



 1 2

1 2 3 4 5 6 7 8

1 2 3 4 5 6 7 8

- c) Upload your assembly program to your Arduino UNO board. Take a video to demonstrate your result. Name it “Ex3” and submit to the Google Classroom.

https://drive.google.com/drive/folders/1u1haDM-cXLsXZIC9xzIBD9-gBf_8yR9h?usp=share_link

```

1  .ORG 0x0      ; Set the origin at address 0x0000 (Reset Vector)
2  JMP MAIN      ; Jump to MAIN routine (bypass interrupt vector area)
3
4  .ORG 0x08      ; Set the interrupt vector for PCINT8 (Pin Change Interrupt on PC0)
5  JMP PCINT8_ISR ; Jump to PCINT8_ISR when the interrupt occurs
6
7  .ORG 0x100     ; Start the main program at address 0x0100
8  MAIN:
9      ; Initialize Stack Pointer
10     LDI R16, HIGH(RAMEND) ; Load high byte of RAMEND (top of stack)
11     OUT SPH, R16          ; Set stack pointer high byte
12     LDI R16, LOW(RAMEND)  ; Load low byte of RAMEND
13     OUT SPL, R16          ; Set stack pointer low byte
14
15     ; Call the subroutine to configure input/output pins
16     CALL PIN_SETUP
17
18     ; Enable Pin Change Interrupts on PCINT8 (PC0)
19     LDI R16, (1<<PCIE1) ; Enable Pin Change Interrupts for PCINT[14:8]
20     STS PCICR, R16        ; Store in Pin Change Interrupt Control Register
21
22     ; Enable PCINT8 (bit 0 of PCMSK1)
23     LDI R16, 0x01        ; Enable only PCINT8 (bit 0 of PCMSK1)
24     STS PCMSK1, R16      ; Store in Pin Change Mask Register 1
25
26     SEI                  ; Enable Global Interrupts ✓
27     LDI R20, 2           ; Load debounce counter with 2
28
29     ; Main Loop - Monitors SW0 (PD4) to control LED0 (PB0)
30     LOOP:
31         SBIC PIND, 4      ; Skip next instruction if SW0 (PD4) is pressed (logic LOW)
32         RJMP L_OFF        ; If not pressed, turn off LED0
33         RJMP L_ON         ; If pressed, turn on LED0
34
35     L_OFF:
36         CBI PORTB, 0      ; Clear bit 0 of PORTB (Turn LED0 OFF)
37         RJMP LOOP         ; Repeat loop
38
39     L_ON:
40         SBI PORTB, 0      ; Set bit 0 of PORTB (Turn LED0 ON)
41         RJMP LOOP         ; Repeat loop
42
43     ; Subroutine to Configure I/O Pins
44     PIN_SETUP:
45         SBI DDRB, 0        ; Set PB0 (LED0) as an OUTPUT
46         CBI PORTB, 0       ; Ensure LED0 is OFF initially
47
48         SBI DDRB, 1        ; Set PB1 (LED1) as an OUTPUT
49         CBI PORTB, 1       ; Ensure LED1 is OFF initially
50
51         CBI DDRD, 4        ; Set PD4 (SW0) as an INPUT
52         SBI PORTD, 4       ; Enable pull-up resistor for SW0
53
54         SBI PORTC, 0       ; Enable pull-up resistor for SW1 (PC0)
55         RET                ; Return from subroutine
56
57     .ORG 0x200 ; Pin Change Interrupt Service Routine (ISR) at address 0x0200
58     PCINT8_ISR:
59         DEC R20            ; Decrease debounce counter
60         BRNE HERE          ; If not zero, skip LED toggle and exit ISR
61
62         IN R17, PORTB      ; Read PORTB (LED states)
63         LDI R18, (1<<1)    ; Load mask for PB1 (LED1)
64         EOR R17, R18       ; Toggle PB1 (LED1)
65         OUT PORTB, R17     ; Store new LED state back to PORTB
66
67         LDI R20, 2         ; Reset debounce counter
68     HERE:
69         RETI              ; Return from Interrupt

```

} enable PC1

} enable PCINT8

สามารถตั้งค่าได้เป็นค่าอื่นได้ (debouncing)

Monitor the switch (SW0)

pressed → LED ON
not pressed → LED OFF

} toggle

Triggered when SW1 (PC0) changes state (press or release)

เช่น exercise 2 ของเรา: trigger ก็คือ edge falling edge (เช่นกดค้างไว้แล้วปล่อย...)