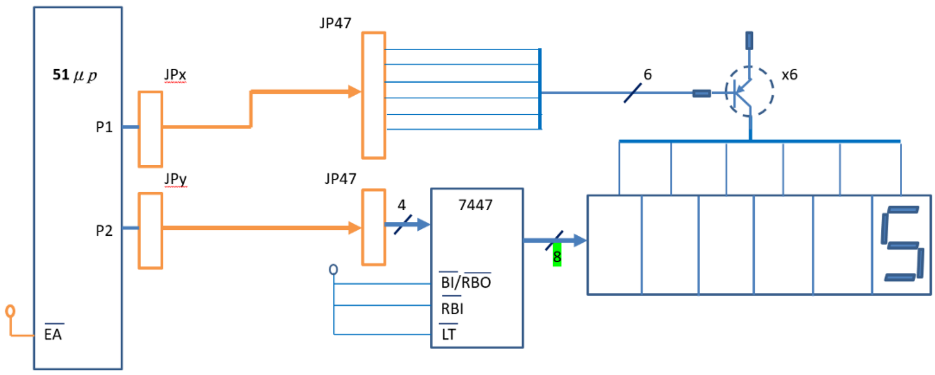
Microprocessor Lab 2.2 Report

0416106 彭敬樺
0416109 周才錢

Subject and Goal:

This lab is about using μ-Vision 51IDE residing on MegaWin82G516 to:

- The access of six 7-segment LED for ON/OFF and pattern control in the 7-segment LED module.
- Organize display patterns in static or dynamic form can be achieved as required.



Preparations:

- Power cable and required connection from the output to the led input is established. The on/off of led is controlled by the output of port P1 and P2 that is connected to 7-segement LED module.
- Check the correctness and check if there are any defective on the board by activating all the 7 LED using static/dynamic pattern display.
- Knowing the operational limit of the six 7-segment LED module.

Operating Procedure:

- Jumper-wiring for circuit setup
- Check the six 7-segment LED module to see if it's working or not by manual wiring to the circuitry.
- Code preparation
- Task execution:
  - Start IDE51 emulation,
  - Start execution and troubleshooting if necessary.

Code Preparation:

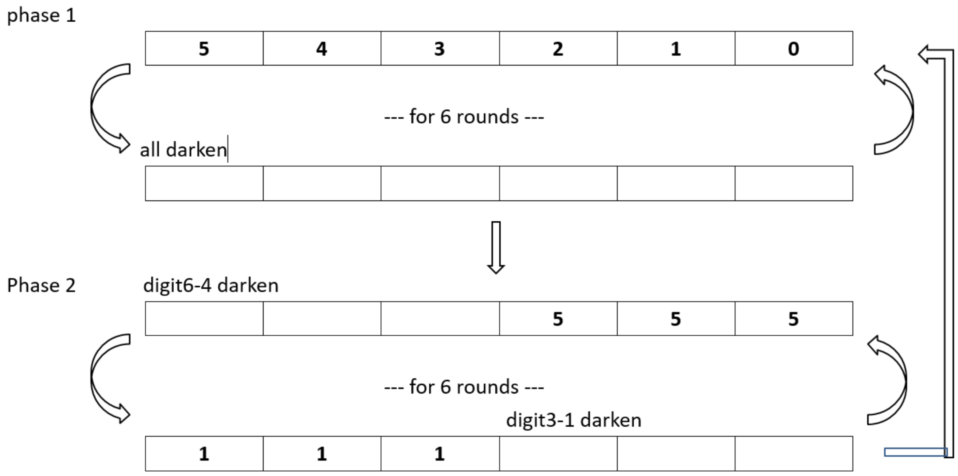
```
                org     0
        mov     SP, #50H
start:  mov     R7, #6
next1:
        mov     R5, #250
```

```
next11:
        mov     R6, #6
        mov     R1, #0FEH
        mov     R2, #0
next12:
    mov    A, R1
    mov    P1, A
    RL     A
    mov    R1, A
    mov    A, R2
    inc    R2
    mov    P2, A
    call   delay1     ; ===KKK===
    djnz   R6, next12
    djnz   R5, next11
    mov    P1, #0FFH
    call   delay2     ; ===LLL===
    djnz   R7, next1
    mov    R7, #6
next2:
    mov    P1, #0F8H
    mov    P2, #5
    call   delay2     ; ===III===
    mov    P1, #0C7H
    mov    P2, #1
    call   delay2     ; ===JJJ===
    djnz   R7, next2
```

```
        jmp    start
      delay1:
               push   1
               mov    R1, #200
               djnz   R1, $
               pop    1
               ret
    delay2:
               push   1
               push   2
               push   3
               mov    R1, #100
         dd22: mov    R2, #250
         dd21: mov    R3, #10
               djnz   R3, $
               djnz   R2, dd21
               djnz   R1, dd22
               pop    3
               pop    2
               pop    1
               ret
               end
```

Observation:

– The initial code provided result in more than 6 rounds of phase two due to the inclusion of two rounds in each loop. Therefore, by reducing the loop counter from 6 to 3, we can achieve a more accurate dynamic display sequence wanted.



- The corrected code below will be the one used in the demonstration and has fulfilled the goals of the requires six 7-segment LED dynamic display sequence.

- The changed code below has no more bug or warning message popup. Therefore, lab 2.2 has been completed and the process has not been a harsh path due to sufficient preparation from reading material prepared before the start of the lab.

- When **===III===** is omitted from the revised code, the six 7-segment LED will only display “1” in digit 6 to 4 and digit 3 to 1 will darken for three rounds. In other words, the state where digit 6 to 4 darken and digit 3 to 1 display “5” will be skipped. The reason is because the code in line **===III===** controls the delay for the skipped state and due to the omission, the output will be immediately replaced by the next state.

- When **===JJJ===** is omitted from the revised code, the six 7-segment LED will only display “5” in digit 3 to 1 and digit 6 to 4 will darken for three rounds. In other words, the state where digit 3 to 1 darken and digit 6 to 4 display “1” will be skipped. The reason is because the code in line **===JJJ===** controls the delay for the skipped state and due to the omission, the output will be immediately replaced by the next state.

- When **===LLL===** is omitted from the revised code, display sequence in phase 1 will darken all the time. The reason is due to the insufficient time for the LED segments to have an appropriate execution time to display the output.

- The delay provided by **delay1** have an appropriate execution time balance between the 4 phases.

- The deletion of R1-push and R1-pop from the revised code in **delay1** will result in inappropriate changes in the six 7-segment LED power switch because the used register is not restored.

- The omitting of push-pop instructions from the revised code in **delay2** will result in nothing in particular due to the re-initialization of the used register after function **delay2** is called.

revised code:

```
            org     0
    mov     SP, #50H
start:  mov     R7, #6
next1:
    mov     R5, #250
next11:
    mov     R6, #6
    mov     R1, #0FEH
    mov     R2, #0
next12:
    mov     A, R1
    mov     P1, A
    RL      A
    mov     R1, A
    mov     A, R2
    inc     R2
```

```
        mov    P2, A
        call   delay1     ; ===KKK===
        djnz   R6, next12
        djnz   R5, next11
        mov    P1, #0FFH
        call   delay2     ; ===LLL===
        djnz   R7, next1
        mov    R7, #3
    next2:
        mov    P1, #0F8H
        mov    P2, #5
        call   delay2     ; ===III===
        mov    P1, #0C7H
        mov    P2, #1
        call   delay2     ; ===JJJ===
        djnz   R7, next2
        jmp    start
    delay1:
               push   1
```

```
               mov    R1, #200
               djnz   R1, $
               pop    1
               ret
    delay2:
               push   1
               push   2
               push   3
               mov    R1, #100
        dd22:  mov    R2, #250
        dd21:  mov    R3, #10
               djnz   R3, $
               djnz   R2, dd21
               djnz   R1, dd22
               pop    3
               pop    2
               pop    1
               ret
               end
```

Comprehensive evaluation:

- When **===KKK===** is omitted from the revised code, display sequence in phase 1 will darken all the time. The reason is due to the insufficient time for the six 7-segment LED to have an appropriate execution time to display the output. If the delay offered by **delay1** is made 1000 times of the original value, then we will see the state of phase 1 display the pattern “0”, ”1”, ”2”, “3”, “4”, “5” sequentially and repeated for 6 rounds. When on number is displayed, the other LED segments will darken.
- The code line marked by **===KKK===** will solve the problem above as long as the delay is not too short.

Designated Assignment:

K = mod(19,6) == 1

- Connecting ground in J54 to the power switch in J47 of pin “C1, C2, C3, C4, C5, C6”. This will ensure all six 7-segment LED to be powered.
- Connecting the J45 of pin “S1, S2, S3, S4” and J47 of pin “SE1, SE2, SE3, SE4”. By changing the value of each bit in the switch will result in the wanted digit appearing on the six 7-segment LED. In this case it is “0001” using right part as the lower bit.