# DSP LAB FINAL EXAM

04 December 2021 11:58

Q1. Write a dsPIC executable program to generate 100 kHz PWM on channel one and two (using PWM-channels of the processor) with different duty ratios. The duty ratio count (minimum count in channel-1 while maximum count in channel-2) needs to be load from the given series (Assume the series as: 200, 079, 450, 721, 231, 367, 108, 917, 289, 333).

step 1 → sort the series

```
.equ __30F6010, 1

        .include "p30f6010.inc"
        .global __reset
;-----------------------------------------------------------------------
;Configuration bits:
;-----------------------------------------------------------------------


        config __FWDT, WDT_OFF              ;Turn off Watchdog Timer
        .text
    __reset:
    ;   series: 200, 079, 450, 721, 231, 367, 108, 917, 289, 333

    ;   this series is stored from 0x1000 to 0x1012

    MOV #200, W0
    MOV W0, 0x1000
    MOV #79, W0
    MOV W0, 0x1002
    MOV #450, W0
    MOV W0, 0x1004
    MOV #721, W0
    MOV W0, 0x1006
    MOV #231, W0
    MOV W0, 0x1008
    MOV #367, W0,
    MOV W0, 0x100A
    MOV #108, W0
    MOV W0, 0x100C
    MOV #917, W0
    MOV W0, 0x100E
    MOV #289, W0
    MOV W0, 0x1010
    MOV #333, W0
    MOV W0, 0x1012
```

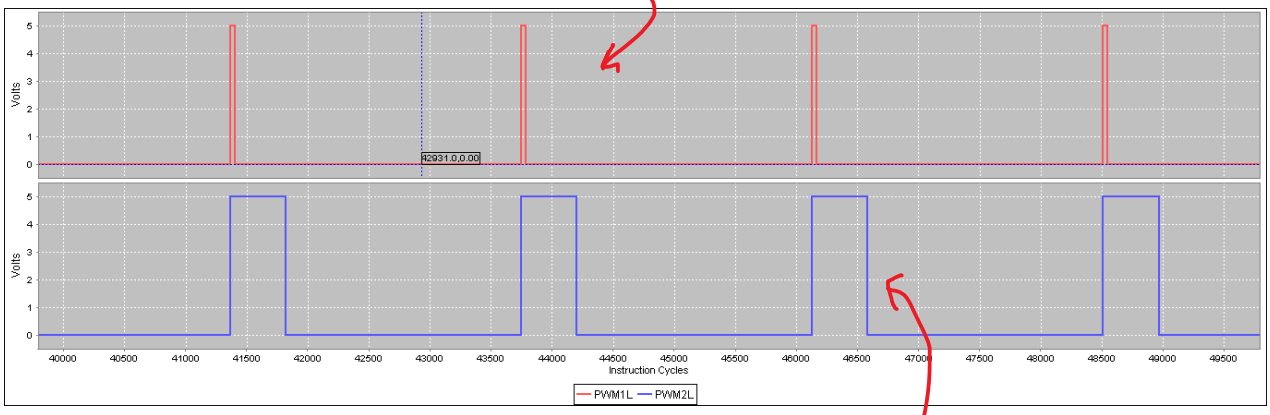
```

    ; bubble sort starts

    MOV #0x0000, W0          ; PROGRAM COUNTER FOR INNER LOOP
    MOV #0x0000, W5          ; PROGRAM COUNTER FOR OUTER LOOP

    LOOP4: DO #8, END4       ; OUTER LOOP FOR BUBBLE SORT
    MOV #0x1000, W1          ; STARTING ADDRESS TO W1
    MOV #0x1002, W2          ; STARTING ADDRESS TO W2
    LOOP1:  DO #8, END1      ; INEER LOOP FOR IMMEDIATE SORT
        MOV [W1], W3         ; TEMP STORAGE OF VALUE 1 FOR CP
        MOV [W2], W4         ; TEMP STORAGE OF VALUE 2 FOR CP
        CP W3, W4            ; CP SETS CARRY IF W3>W4
        BRA C, LOOP2         ; BRANCH TO LOOP2 IF CARRY IS SET
        BRA LOOP3            ; BRANCH TO LOOP3 IS IT ISNT

        LOOP2:  MOV W3, [W2]     ; SWAPPING NUMBERS IF W3>W4
        MOV W4, [W1]             ; SWAPPING NUMBERS IF W3>W4
        BRA LOOP3                ; BRANCH TO LOOP3 (UNCONDITIONAL)

        LOOP3: INC2 W1, W1       ; INCREMENT ADDRESS 1 FOR NEXT SWEEP
        INC2 W2, W2              ; INCREMENT ADDRESS 2 FOR NEXT SWEEP
    END1: INC W0,W0              ; INNER LOOP PC INCREMENT
    END4: INC W5, W5             ; OUTER LOOP PC INCREMENT

    ; end of sorting
    ; now the minimum value is stored in 0x1000 where the
    ; maximum value is stored in 0x1012

    ; PWM1 needs minimum value (i.e. 0x1000)
    ; PWM2 needs maximum value (i.e. 0x1012)

    ; PWM Code starts from here

```

```
84
85          mov #0x0400, w0          ; PWM module is disabled, continue operation in
86          mov w0, PTCON            ; Idle mode, special event interrupt disabled,
87                                   ; immediate period updates enabled, no external
88                                   ; synchronization
89          ; Set the PWM Period
90          mov #0x094D, w0          ; Select period to be approximately 2.5?s
91          mov w0, PTPER            ; PLL Frequency is ~480MHz. This equates to a
92                                   ; clocke period of 2.1ns. The PWM period and
93                                   ; duty cycle registers are triggered on both +ve
94                                   ; and -ve edges of the PLL clock. Therefore,
95                                   ; one count of the PTPER and PDCx registers
96                                   ; equals 1.05ns.
97                                   ; So, to achieve a PWM period of 2.5?s, we
98                                   ; choose PTPER = 0x094D
99
100         ; Select individual Duty Cycle Control
101
102
103         mov #0xFFFF, w0          ; Fault interrupt disabled, Current Limit
104         mov w0, PWMCON1          ; interrupt disabled, trigger interrupt,
105                                  ; disabled, Primary time base provides timing,
106                                  ; DC1 provides duty cycle information, positive
107                                  ; dead time applied, no external PWM reset,
108                                  ; Enable immediate duty cycle updates
109
110         ; Duty Cycle Setting
111
112         ; PWM1 needs minimum value (i.e. 0x1000)
113         MOV #0x1000, W0          ; MINIMUM VALUE ADDRESS
114         MOV [W0], W1
115
```

```
109
110         ; Duty Cycle Setting
111
112         ; PWM1 needs minimum value (i.e. 0x1000)
113         MOV #0x1000, W0          ; MINIMUM VALUE ADDRESS
114         MOV [W0], W1
115
116                                  ; To achieve a duty cycle of 50%, we choose
117         MOV W1, PDC1             ; the PDC1 value = 0.5*(PWM Period)
118                                  ; The ON time for the PWM = 1.25?s
119                                  ; The Duty Cycle Register will provide
120                                  ; positive duty cycle to the PWMxH outputs
121                                  ; when output polarities are active high
122                                  ; (see IOCON1 register)
123
124         ; PWM2 needs maximum value (i.e. 0x1012)
125         MOV #0x1012, W0          ; MAXIMUM VALUE ADDRESS
126         MOV [W0], W2
127                                  ; To achieve a duty cycle of 50%, we choose
128         MOV W2, PDC2             ; the PDC1 value = 0.5*(PWM Period)
129                                  ; The ON time for the PWM = 1.25?s
130                                  ; The Duty Cycle Register will provide
131                                  ; positive duty cycle to the PWMxH outputs
132                                  ; when output polarities are active high
133                                  ; (see IOCON1 register)
134
135         bset PTCON, #15          ; turn ON PWM module
```

Sorting & PWM outputs are in next page

PWM1 → 4f (min)




0395 (Max)

Min

| | | | | |
|---|---|---|---|---|
| 0FFE | 0x0000 | 0 | 00000000 00000000 | '..' |
| 1000 | 0x004F | 79 | 00000000 01001111 | '.O' |
| 1002 | 0x006C | 108 | 00000000 01101100 | '.l' |
| 1004 | 0x00C8 | 200 | 00000000 11001000 | '.È' |
| 1006 | 0x00E7 | 231 | 00000000 11100111 | '.ç' |
| 1008 | 0x0121 | 289 | 00000001 00100001 | '.!' |
| 100A | 0x014D | 333 | 00000001 01001101 | '.M' |
| 100C | 0x016F | 367 | 00000001 01101111 | '.o' |
| 100E | 0x01C2 | 450 | 00000001 11000010 | '.Â' |
| 1010 | 0x02D1 | 721 | 00000010 11010001 | '.Ñ' |
| 1012 | 0x0395 | 917 | 00000011 10010101 | '.□' |
| 1014 | 0x0000 | 0 | 00000000 00000000 | '..' |
| 1016 | 0x0000 | 0 | 00000000 00000000 | '..' |
| 1018 | 0x0000 | 0 | 00000000 00000000 | '..' |
| 101A | 0x0000 | 0 | 00000000 00000000 | '..' |

Max value