

**EEEN 3449**  
**HW # 4**

Name: \_\_\_\_\_

1. Write an instruction sequence to generate a 3 KHz waveform with 60% duty cycle from the PWM1 pin (PP1) and use **left-aligned** mode. Assume that the E clock frequency is 24 MHz, and the prescale factor is 32.
- Set PWM1 output to start with **low** level.
  - Set PWM1 output to start with **high** level.
- Note: PWMPER1 & PWMDTY1 can hold any value 0~255.

E clock is @ 24 MHz, and assume the prescaler = 32  $\Rightarrow$  Clock A is @ 750 KHz

Waveform frequency = 3 KHz, then the period =  $\frac{1}{3000}$  s

$$\# \text{ of counts for clock A in } \frac{1}{3000} \text{ s} = \frac{\frac{1}{3000}}{\frac{1}{750 \times 10^3}} = \frac{1}{3000} \times 750 \times 10^3 = 250$$

[PWMPER1]=250 ; [PWMDTY1]=250 - 150 = 100

*Solution for a): Set PWM1 output to start with low level.*

```

movb    #0 , PWMCLK           ; select clock A as the clock source for PWM1
movb    #5 , PWMPRCLK        ; set clock A prescaler to 32
movb    #0 , PWMPOL          ; channel 1 output low at the start of the period
movb    #0 , PWMCAE          ; select left-aligned mode
movb    #$0C , PWMCTL        ; 8-bit mode, stop PWM in wait and freeze mode
movb    #250 , PWMPER1       ; set period value in decimal
movb    #100 , PWMDTY1       ; set duty value in decimal
movb    #0 , PWMCNT1         ; reset the PWM1 counter
bset    PWME , $02          ; enable PWM channel 1

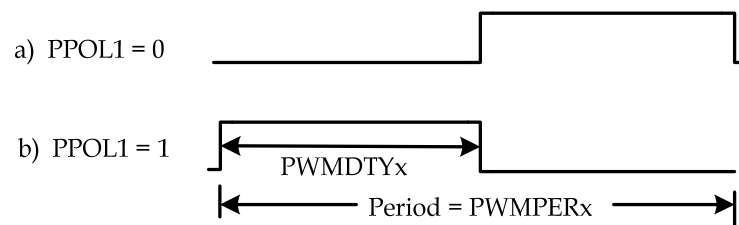
```

*Solution for b): Set PWM1 output to start with high level.*

```

movb    #0 , PWMCLK           ; select clock A as the clock source for PWM1
movb    #5 , PWMPRCLK        ; set clock A prescaler to 32
movb    #2 , PWMPOL          ; channel 1 output high at the start of the period
movb    #0 , PWMCAE          ; select left-aligned mode
movb    #$0C , PWMCTL        ; 8-bit mode, stop PWM in wait and freeze mode
movb    #250 , PWMPER1       ; set period value in decimal
movb    #150 , PWMDTY1       ; set duty value in decimal
movb    #0 , PWMCNT1         ; reset the PWM1 counter
bset    PWME , $02          ; enable PWM channel 1

```



PWM left-aligned output waveform

2. Write an instruction sequence to generate a square wave with 300  $\mu$ s period and 60% duty cycle from PWM0 and use **center-aligned** mode. Assume that the E clock frequency is 24 MHz, and the prescale factor is 16.

- Set PWM0 output to start with low level.
- Set PWM0 output to start with high level.

E clock is @ 24 MHz, and assume the prescaler = 16  $\Rightarrow$  Clock A is @ 1.5 MHz

$$\text{Period} = 1 \text{ ms, then \# of counts for clock A in } 300 \mu\text{s} = \frac{300 \times 10^{-6}}{1.5 \times 10^{-6}} = 300 \times 10^{-6} \times 1.5 \times 10^6 = 450$$

$$\text{Period} = 2 \times [\text{PWMPER0}] = 450 \Rightarrow [\text{PWMPER0}] = 225$$

$$[\text{PWMDTY0}] = [\text{PWMPER0}] \times 40\% = 90 \quad \text{Note: 40\% of a period is low \& 60\% is high}$$

**Solution for a): Set PWM0 output to start with low level.**

$$[\text{PWMPER0}] = 225; \quad [\text{PWMDTY0}] = 225 \times 40\% = 90$$

```

movb    #0, PWMCLK           ; select clock A as the clock source for PWM0
movb    #4, PWMPRCLK         ; set clock A prescaler to 16
movb    #0, PWMPOL           ; channel 0 output low at the start of the period
movb    #1, PWMCAE           ; select center-aligned mode
movb    #$0C, PWMCTL         ; 8-bit mode, stop PWM in wait and freeze mode
movb    #225, PWMPER0        ; set period value in decimal
movb    #90, PWMDTY0         ; set duty value in decimal
bset    PWME, $01           ; enable PWM channel 0

```

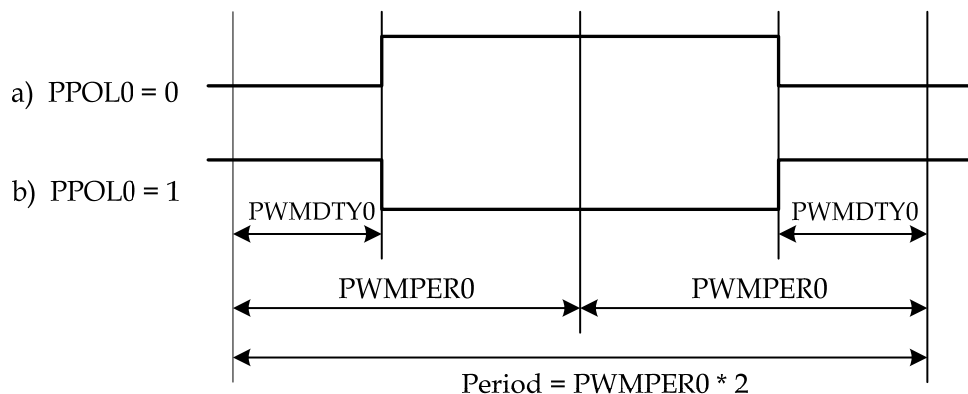
**Solution for b): Set PWM0 output to start with high level.**

$$[\text{PWMPER0}] = 225; \quad [\text{PWMDTY0}] = 225 \times 60\% = 135$$

```

movb    #0, PWMCLK           ; select clock A as the clock source for PWM0
movb    #4, PWMPRCLK         ; set clock A prescaler to 16
movb    #1, PWMPOL           ; channel 0 output high at the start of the period
movb    #1, PWMCAE           ; select center-aligned mode
movb    #$0C, PWMCTL         ; 8-bit mode, stop PWM in wait and freeze mode
movb    #225, PWMPER0        ; set period value in decimal
movb    #135, PWMDTY0        ; set duty value in decimal
bset    PWME, $01           ; enable PWM channel 0

```



PWM center aligned output waveform

3. Write an instruction sequence to generate a 60 Hz digital waveform with 60% duty cycle using the **16-bit mode** from the PWM1 output pin. Assume that the E clock frequency is 24 MHz, and the prescale factor is 8.
- Select **left aligned** mode and set PWM1 output to start with low level.
  - Select **left-aligned** mode and set PWM1 output to start with high level.
  - Select **center-aligned** mode and set PWM1 output to start with low level.
  - Select **center-aligned** mode and set PWM1 output to start with high level.

*Solution for a): Select **left aligned mode** and set PWM1 output to start with **low level**.*

E clock is @24 MHz and the prescaler = 8  $\Rightarrow$  Clock A is @ 3 MHz

Waveform frequency = 60 Hz, then the period =  $\frac{1}{60}$  sec.

$$\# \text{ of counts for the period} = \frac{\frac{1}{60}}{\frac{1}{3 \times 10^6}} = \frac{1}{60} \times 3 \times 10^6 = 50,000$$

$$[\text{PWMDTY1:PWMDTY0}] = 50,000 \times (1 - 60\%) = 20,000$$

```

movb    #0 , PWMCLK           ; select clock A as the clock source
movb    #0 , PWMPOL           ; set PWM0:PWM1 output to start with low level
movb    #3 , PWMPRCLK         ; set prescaler to 8
movb    #\$1C , PWMCTL        ; concatenate PWM0:PWM1, stop PWM in wait or freeze mode
movb    #0 , PWMCAE           ; select left align mode
movw    #50000 , PWMPER0      ; set period to 50000
movw    #20000 , PWMDTY0      ; set duty to 24000
bset    PWME , PWME1         ; enable PWM0:PWM1 (PWME0 is in don't care)

```

*Solution for b): Select **left aligned mode** and set PWM1 output to start with **high level**.*

$$[\text{PWMPER1:PWMPER0}] = 50000$$

$$[\text{PWMDTY1:PWMDTY0}] = 50000 \times 60\% = 30000$$

```

movb    #0 , PWMCLK           ; select clock A as the clock source
movb    #2 , PWMPOL           ; set PWM0:PWM1 output to start with high level
movb    #3 , PWMPRCLK         ; set prescaler to 8
movb    #\$1C , PWMCTL        ; concatenate PWM0:PWM1, stop PWM in wait or freeze mode
movb    #0 , PWMCAE           ; select left-aligned mode for PWM channel 1
movw    #50000 , PWMPER0      ; set period to 50000
movw    #30000 , PWMDTY0      ; set duty to 30000
bset    PWME , PWME1         ; enable PWM0:PWM1 (PWME0 is in don't care)

```

*Solution for c):* Select **center aligned mode** and set PWM1 output to start with **low level**.

$$2 \times [\text{PWMPER1} : \text{PWMPER0}] = 50000 \quad [\text{PWMPER1} : \text{PWMPER0}] = 25000$$
$$2 \times [\text{PWMDTY1} : \text{PWMDTY0}] = 50000 \times 40\% = 20000 \quad [\text{PWMDTY1} : \text{PWMDTY0}] = 10000$$

```
movb    #0, PWMCLK           ; select clock A as the clock source
movb    #0, PWMPOL           ; set PWM0:PWM1 output to start with low level
movb    #3, PWMPCRCLK        ; set prescaler to 8
movb    #0x1C, PWMCTL        ; concatenate PWM1:PWM0, stop PWM in wait or freeze mode
movb    #2, PWMCAE           ; select center-aligned mode for PWM channel 1
movw    #25000, PWMPER0      ; set period to 25000
movw    #10000, PWMDTY0     ; set duty to 10000
bset    PWME, PWME1         ; enable PWM1:PWM0 (PWME0 is in don't care)
```

*Solution for d):* Select **center aligned mode** and set PWM1 output to start with **high level**.

$$2 \times [\text{PWMPER1} : \text{PWMPER0}] = 50000 \quad [\text{PWMPER1} : \text{PWMPER0}] = 25000$$
$$2 \times [\text{PWMDTY1} : \text{PWMDTY0}] = 50000 \times 60\% = 30000 \quad [\text{PWMDTY1} : \text{PWMDTY0}] = 15000$$

```
movb    #0, PWMCLK           ; select clock A as the clock source
movb    #2, PWMPOL           ; set PWM0:PWM1 output to start with high level
movb    #3, PWMPCRCLK        ; set prescaler to 8
movb    #0x1C, PWMCTL        ; concatenate PWM1:PWM0, stop PWM in wait or freeze mode
movb    #2, PWMCAE           ; select center-aligned mode for PWM channel 1
movw    #25000, PWMPER0      ; set period to 25000
movw    #15000, PWMDTY0     ; set duty to 12000
bset    PWME, PWME1         ; enable PWM1:PWM0 (PWME0 is in don't care)
```