

Generarea semnalelor PWM

Obiectiv

Familiarizarea cu cele doua module generatoare de semnale PWM din cadrul dsPIC-ului si generarea de semnale PWM independente sau complementare.

1. Introducere

Microcontrolerele din familia dsPIC33F prezinta doua module de generare a semnalelor modulate in latimea impulsurilor (semnale PWM), denumite MCPWM1 si MCPWM2. Primul modul PWM este folosit pentru a genera pana la 6 semnale PWM sincronizate, grupate in 3 perechi (PWM1H, PWM1L,etc). Cel de-al doilea modul PWM genereaza doua semnale complementare.

Modulul 1 poate fi folosit pentru urmatoarele aplicatii:

- Controlul motoarelor de curent alternativ trifazat;
- Controlul motoarelor cu reluctanta inversata;
- Controlul motoarelor de curent continuu fara perii;

Modulul 2 poate fi folosit pentru urmatoarele aplicatii:

- Controlul cuptoarelor cu inductie;

2. Registrii de control

- **PxTCON: PWM Time Base Control Register** – este folosit pentru configurarea modului de numarare, a prescalerului si a postscalerului, cat si pentru a porni numararea;

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
PTEN	—	PTSIDL	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTOPS<3:0>				PTCKPS<1:0>		PTMOD<1:0>	
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **PTEN: PWM Time Base Timer Enable bit**

- 1 = PWM time base is on
- 0 = PWM time base is off

bit 14 **Unimplemented: Read as '0'**

bit 13 **PTSIDL: PWM Time Base Stop in Idle Mode bit**

- 1 = PWM time base halts in CPU Idle mode
- 0 = PWM time base runs in CPU Idle mode

bit 12-8 **Unimplemented: Read as '0'**

bit 7-4 **PTOPS<3:0>: PWM Time Base Output Postscale Select bits**

1111 = 1:16 postscale

-
-
-

0001 = 1:2 postscale

0000 = 1:1 postscale

bit 3-2 **PTCKPS<1:0>: PWM Time Base Input Clock Prescale Select bits**

- 11 = PWM time base input clock period is 64 TCY (1:64 prescale)
- 10 = PWM time base input clock period is 16 TCY (1:16 prescale)
- 01 = PWM time base input clock period is 4 TCY (1:4 prescale)
- 00 = PWM time base input clock period is TCY (1:1 prescale)

bit 1-0 **PTMOD<1:0>: PWM Time Base Mode Select bits**

- 11 = PWM time base operates in a Continuous Up/Down mode with interrupts for double PWM updates
- 10 = PWM time base operates in a Continuous Up/Down Counting mode
- 01 = PWM time base operates in Single Event mode
- 00 = PWM time base operates in Free Running mode

- **PxTMR: PWM Time Base Register** – din acest registru pot fi setate/citite valoarea curenta a timerului si sensul de numarare (crescator/descrescator);

R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTDIR	PTMR<14:8>						
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTMR<7:0>							
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15 **PTDIR:** PWM Time Base Count Direction Status bit (read-only)

1 = PWM time base is counting down

0 = PWM time base is counting up

bit 14-0 **PTMR<14:0>:** PWM Time Base Register Count Value bits

- **PxTPER: PWM Time Base Period Register** – in acest registru se scrie perioada de timp a modularii, ceea ce da frecventa PWM-ului;

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	PTPER <14:8>						
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTPER <7:0>							
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15 **Unimplemented:** Read as '0'

bit 14-0 **PTPER<14:0>:** PWM Time Base Period Value bits

- **PWMxCON1: PWM Control Register 1** – se selecteaza modul independent sau complementar pentru fiecare pereche de pini de I/O

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	PMOD4	PMOD3	PMOD2	PMOD1
bit 15				bit 8			

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
PEN4H	PEN3H	PEN2H	PEN1H	PEN4L	PEN3L	PEN2L	PEN1L
bit 7				bit 0			

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15-12 **Unimplemented:** Read as '0'

bit 11-8 **PMOD<4:1>:** PWM I/O Pair Mode bits

1 = PWM I/O pin pair is in the Independent Output mode

0 = PWM I/O pin pair is in the Complementary Output mode

bit 7 **PEN4H:** PWMxH4 I/O Enable bits^(1,2)

1 = PWMxH4 pin is enabled for PWM output

0 = PWMxH4 pin disabled; I/O pin becomes general purpose I/O

- **PxDCy: PWM Duty Cycle Register y** – in acesti registri se introduc valorile pe 16 biti ale factorilor de umplere pentru perechile de iesiri y;

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PxDC1<15:8>							
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PxDC1<7:0>							
bit 7				bit 0			

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 15-0 **PxDC1<15:0>:** PWM Duty Cycle 1 Value bits

- **PWMxCON2: PWM Control Register 2 –**

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	SEVOPS<3:0>			
bit 15				bit 8			

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	IUE	OSYNC	UDIS
bit 7				bit 0			

Legend:							
R = Readable bit		W = Writable bit		U = Unimplemented, read as '0'			
-n = Value at POR		'1' = Bit is set		'0' = Bit is cleared		x = Bit is unknown	

bit 15-12 **Unimplemented:** Read as '0'

bit 11-8 **SEVOPS<3:0>**: PWM Special Event Trigger Output Postscale Select bits

1111 = 1:16 postscale

•
•
•

0001 = 1:2 postscale

0000 = 1:1 postscale

bit 7-3 **Unimplemented:** Read as '0'

bit 2 **IUE**: Immediate Update Enable bit

1 = Updates to the active PxDCy^(1,2) registers are immediate

0 = Updates to the active PxDCy^(1,2) registers are synchronized to the PWM time base

bit 1 **OSYNC**: Output Override Synchronization bit

1 = Output overrides via the PxOVDCON⁽¹⁾ register are synchronized to the PWM time base

0 = Output overrides via the PxOVDCON⁽¹⁾ register occur on the next TCY boundary

bit 0 **UDIS**: PWM Update Disable bit

1 = Updates from duty cycle and period buffer registers are disabled

0 = Updates from duty cycle and period buffer registers are enabled

Note 1: The letter "x" refers to the MCPWM module number.

2: The letter "y" refers to the MCPWM Duty Cycle register number.

3. Modurile de lucru

Modulele PWM pot lucra in patru moduri de lucru distincte, selectia facandu-se prin configurarea bitilor 0 si 1 (PTMOD) din registrul **PxTCON**:

- a) **Free running mode** (PTMOD<1:0> = 0b00) – in acest mod, registrul de numarare **PxTMR** numara crescator pana cand valoarea lui este egala cu perioada de baza (valoarea din registrul **PxTPER**). Dupa producerea acestui eveniment valoarea din registrul **PxTMR** este resetata la aparitia urmatorului front al semnalului de tact. Numararea si resetarea se repeata atata timp cat bitul PTEN (bitul 15 din registrul **PxTCON**) este setat.
- b) **Single event mode** (PTMOD<1:0> = 0b01) – la setarea bitului PTEN din registrul **PxTCON** incepe numararea. Dupa egalarea valorii din **PxTPER**, la urmatorul front al semnalului de tact registrul **PxTMR** este resetat, iar bitul PTEN este trecut in 0, fortandu-se astfel oprirea timerului.
- c) **Continuous up/down count mode** (PTMOD<1:0> = 0b10) – in acest mod timerul numara crescator pana cand se egaleaza valoarea din **PxTPER**. Dupa ce aceasta valoare a fost egalata, la aparitia urmatorului front al semnalului de tact, timerul incepe sa numere descrescator pana cand valoarea din el devine 0. Bitul 15 din registrul **PxTMR** indica sensul de numarare. Acesta se seteaza cand timerul incepe numararea descrescatoare.
- d) **Continuous up/down count mode with interrupts for double update of duty cycle** (PTMOD<1:0> = 0b11) – mod similar cu c), cu exceptia ca se genereaza cate o intrerupere atat atunci cand **PxTMR** este 0, cat si cand **PxTMR** egaleaza valoarea din **PxTPER**.

4. Prescalerul si postscalerul

Durata impulsurilor PWM este data de un timer pe 16 biti cu prescaler si postscaler. Exista patru posibilitati de configurare a prescalerului, 1:1, 1:4, 1:16, 1:64, alegerea scarii dorite facandu-se prin setarea bitilor 2 si 3 din registrul **PxTCON**. Postscalerul este folositor atunci cand factorul de umplere nu trebuie modificat la fiecare ciclu PWM. Exista mai multe valori pentru postscalare cuprinse intre 1:1 si 1:16. **Postscalerul este sters atunci cand se scrie in unul dintre registrele PxTMR sau PxTCON.**

5. Perioada de timp a PWM-ului

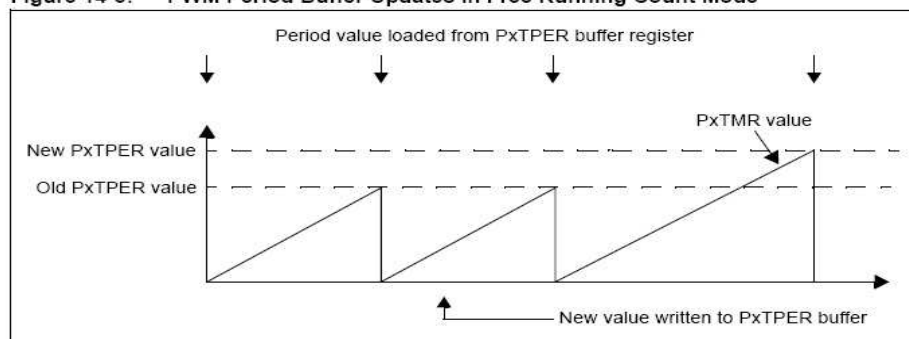
Valoarea incarcata in **PxTPER** determina perioada de numarare a registrului **PxTMR**. Registrul **PxTPER** este de fapt un buffer al registrului real, acesta din urma nefiind accesibil

utilizatorilor. Valoarea din bufferul **PxTPER** este copiata in registrul real la aparitia unuia dintre urmatoarele evenimente:

- La functionarea in unul din modurile free running sau single event: atunci cand continutul registrului **PxTMR** este resetat (dupa ce valoarea acestuia a egalat continutul registrului **PxTPER**);
- In modurile up/down counting: atunci cand registrul **PxTMR** este zero;

Valoarea din bufferul **PxTPER** este automat copiata in registrul fizic atunci cand bitul PTEN din registrul **PxTCON** este pe 0 logic.

Figure 14-3: PWM Period Buffer Updates in Free Running Count Mode



Equation 14-1: PWM Period Calculation for Free Running Count Mode (PTMOD = 10 or 11)

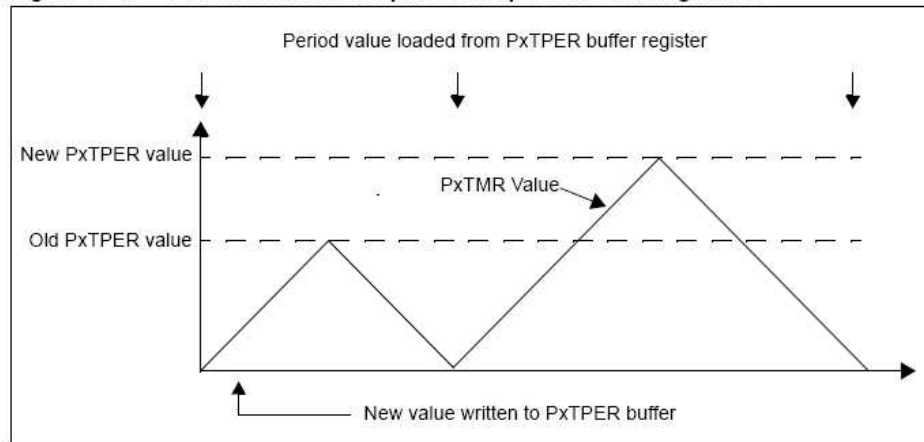
$$PxTPER = \frac{FCY}{FPWM \times (PxTMR \text{ Prescaler})} - 1$$

Example:

$$\begin{aligned} FCY &= 20 \text{ MHz} \\ FPWM &= 20,000 \text{ Hz} \\ PxTMR \text{ Prescaler} &= 1:1 \end{aligned}$$

$$\begin{aligned} PxTPER &= \frac{20,000,000}{20,000 \times 1} - 1 \\ &= 1000 - 1 \\ &= 999 \end{aligned}$$

Figure 14-4: PWM Period Buffer Updates in Up/Down Counting Modes



**Equation 14-2: PWM Period Calculation in Up/Down Counting Modes
(PTMOD = 00 or 01)**

$$PxTPER = \frac{FCY}{FPWM \times (PxTMR \text{ Prescaler}) \times 2} - 1$$

Example:

$FCY = 20 \text{ MHz}$

$FPWM = 20,000 \text{ Hz}$

$PxTMR \text{ Prescaler} = 1:1$

$$\begin{aligned} PxTPER &= \frac{20,000,000}{20,000 \times 1 \times 2} - 1 \\ &= 500 - 1 \\ &= 499 \end{aligned}$$

6. Intreruperi

Intreruperile sunt generate diferit, in functie de modul de lucru selectat din registrul PxTCON, cat si de valorile setate pentru prescaler si postscaler.

- In modul free running se genereaza cate o intrerupere atunci cand valoarea din registrul PxTMR este resetata la 0. Folosind postscalerul se poate reduce frecventa generarii de intreruperi.
- In modul single event, ca si in precedentul mod, se genereaza o intrerupere atunci cand PxTMR este resetat, ca urmare a egalarii valorii din PxTPER. Bitul PTEN din registrul PxTCON este de asemenea resetat pentru a opri functionarea timerului. Postscalerul nu are nicio influenta asupra momentului generarii intreruperii.

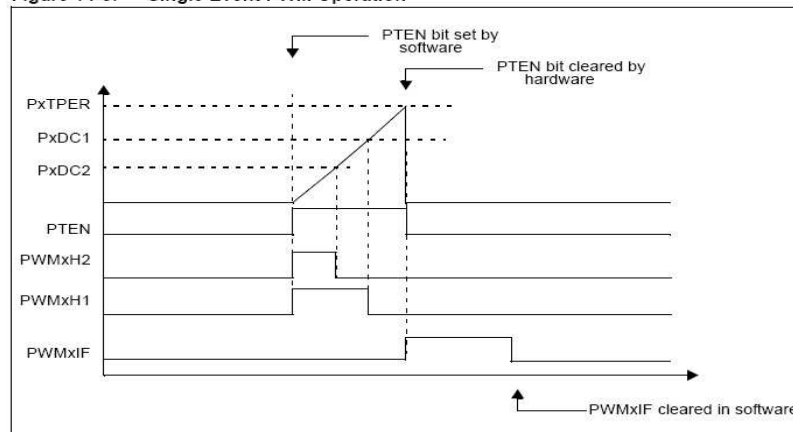
- In modul up/down counting se genereaza cate o intrerupere de fiecare data cand timerul ajunge in 0 si incepe numararea crescatoare. Pentru a se genera mai putine intreruperi se poate utiliza postscalerul.
- In modul up/down counting with double update of duty cycle se genereaza cate o intrerupere atunci cand valoarea din PxTMR este egala cu 0 sau cu continutul registrului PxTPER. Postscalerul nu are nici o influenta asupra frecventei de generare a intreruperilor.

7. Modurile de iesire PWM

Polaritatea iesirilor PWM poate fi setata din registrul **FPOR** folosind bitii HPOL si LPOL. HPOL seteaza polaritatea iesirilor PWMxH1:PWMxH4, iar LPOL pe cea a iesirilor PWMxL1:PWMxL4. Daca bitul de configurare este pe 1 logic, atunci iesirile corespunzatoare vor fi active pe nivel inalt. Din registrul **PWMxCON1** se seteaza care pini sunt folositi pentru PWM. In cazul in care un pin de I/O este setat ca iesire a unui modul PWM, atunci registrele **TRIS** si **PORT** ce controleaza acel pin sunt dezactivate.

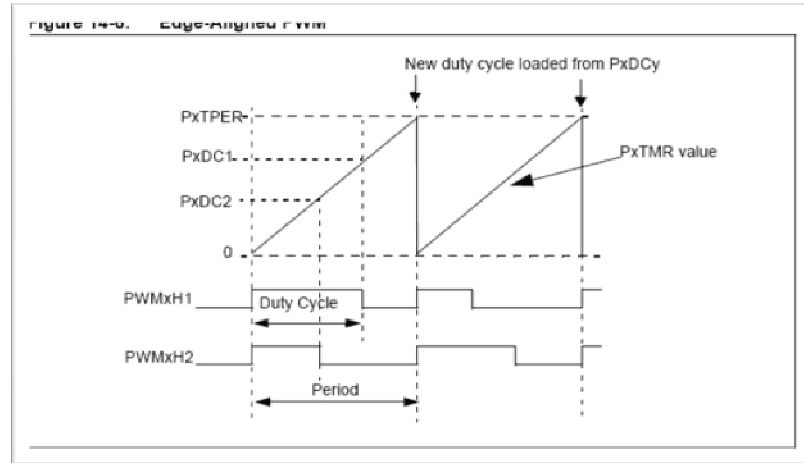
- a) **Single event PWM operation** (PTMOD <1:0>=01)- in acest mod pinii PWM I/O sunt trecuti in stare activa atunci cand se seteaza bitul PTEN. La egalarea valorii dintr-un registru **PxDCy** (ce da factorul de umplere), pin-ul corespunzator de I/O este trecut in stare inactiva. Cand se egaleaza valoare din **PxTPER**, valoarea din **PxTMR** este resetata, bitul PTEN este trecut in 0 si se genereaza o intrerupere. Generearea de semnale PWM se opreste pana cand bitul PTEN este din nou setat prin software.

Figure 14-5: Single Event PWM Operation

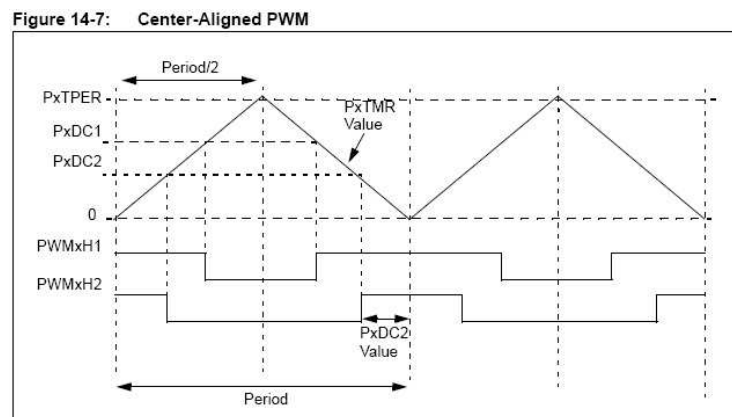


- b) **Edge-aligned PWM** – semnalele de acest tip sunt produse atunci cand modulele PWM functioneaza in modul free running. Perioada specifica a semnalului PWM este retinuta in registrul **PxTPER**, iar factorii de umplere sunt specificati de registrii **PxDCy**.

Presupunem ca factorul de umplere este mai mare decat 0 si nu sunt activate updateurile imediate (IUE=0). In acest caz iesirile PWM sunt activate la inceputul fiecărei perioade (**PxTMR=0**). Fiecare iesire va fi dezactivata de fiecare data cand valoarea din **PxTMR** egaleaza valoarea din registrul ce memoreaza valoarea factorului de umplere corespunzator. Daca valoarea din **PxDCy** este 0, atunci iesirea este mentinuta inactiva pe toata perioada, iar in cazul in care valoarea din **PxDCy** este mai mare decat cea retinuta in **PxTPER**, atunci iesirea corespunzatoare este mentinuta activa pe intreaga perioada de baza.



- c) **Center-aligned PWM** – aceste semnale sunt produse atunci cand modulul functioneaza in unul dintre cele doua moduri up/down counting. Una din iesirile PWM este trecuta in stare activa atunci cand valoarea din PxTMR o egaleaza pe cea din **PxDCy** si timerul numara descrescator (PTDIR=1). Atunci cand PxTMR are aceeasi valoare ca si **PxDCy**, dar timerul numara crescator, iesirea devine inactiva. Daca valoarea din **PxDCy** este 0, atunci iesirea este inactiva pe intreaga perioada, iar daca **PxDCy** este mai mare decat **PxTPER**, atunci iesirea este activa.



- d) **Complementary PWM output** – in acest mod, iesirile unei perechi nu pot fi simultan active. Modul complementar se activeaza prin resetarea bitilor corespunzatori din registrul **PWMxCON1**. Acest mod este modul predefinit la resetarea dispozitivului.
- e) **Independent PWM output** – in acest mod iesirile unei perechi pot fi simultan active. Activarea se face prin plasarea pe 1 a bitilor corespunzatori din registrul **PWMxCON1**.

8. Duty cycle register buffering

Pentru fiecare registru **PxDCy** exista cate un buffer accesibil din program si un registru nemapat in memorie ce contine valoarea efectiva de comparare. Factorul de umplere este updatat la momente specifice de timp. In modurile free running si single event factorul de umplere este updatat cand valoarea **PxTMR** este egala cu ce a **PxTPER** si **PxTMR** este trecut in 0. In modul up/down counting updateul se face atunci cand **PxTMR** este 0, iar timerul incepe sa numere crescator. Modul up/down counting with double updates face update atat cand valoarea din **PxTMR** este 0, cat si atunci cand aceasta egaleaza valoarea din **PxTPER**.

Figure 14-12: Duty Cycle Update Times in Up/Down Counting Mode

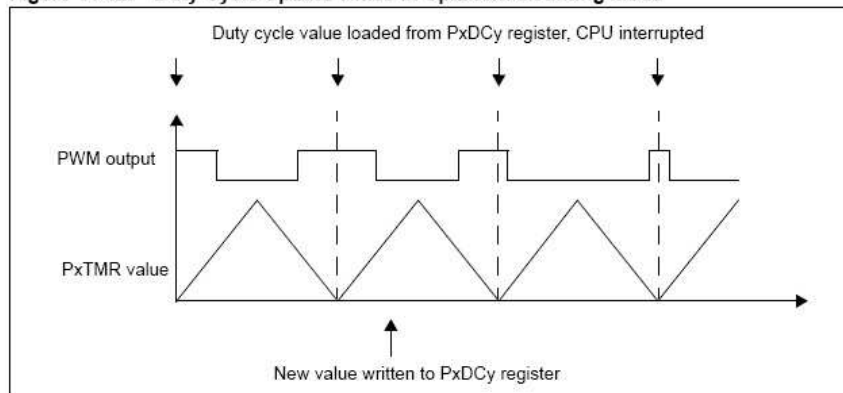
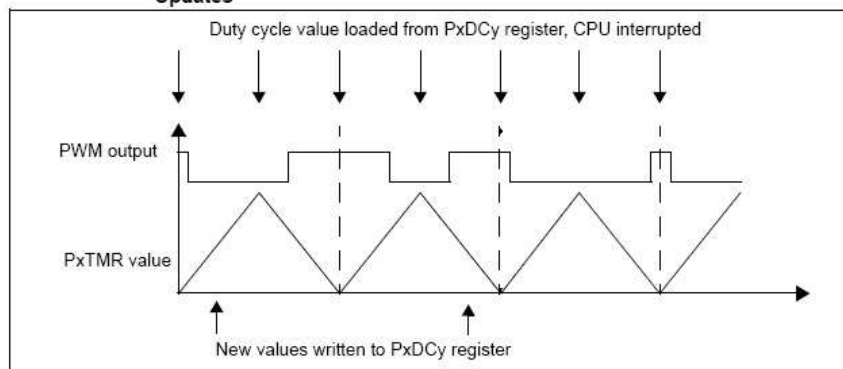


Figure 14-13: Duty Cycle Update Times in Up/Down Counting Mode with Double Updates

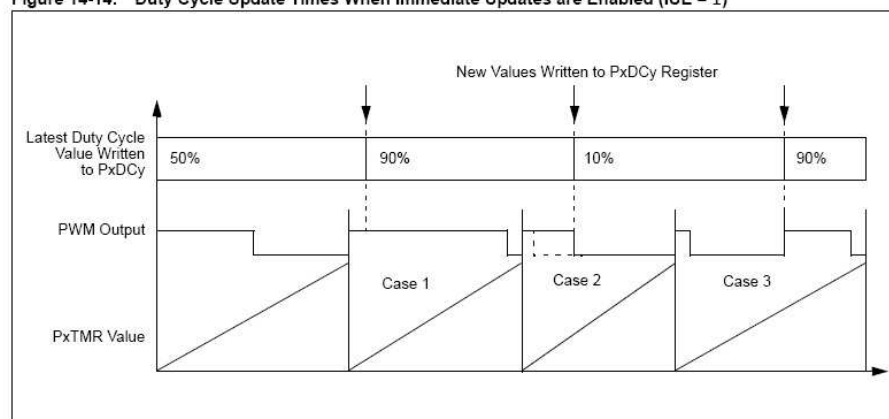


Update-ul imediat al factorului de umplere

Bitul 2 al registrului **PWMxCON2**, bitul IUE ofera optiunea ca updateul valorilor factorilor de umplere sa se faca imediat ce o valoare a fost scrisa in registrii **PxDCy**. Acest lucru permite updateul valorii factorului de umplere fara a mai astepta sfarsitul perioadei din **PxTPER**. Daca bitul IUE este setat atunci aceasta optiune este activata, in caz contrar updateul imediat este dezactivat. Se pot distinge urmatoarele posibilitati:

- ✓ Daca iesirea PWM este activa in momentul in care se scrie o noua valoare, iar noua valoare a factorului de umplere este mai mare decat cea veche, latimea impulsului este marita;
- ✓ Daca iesirea PWM este active in momentul in care se scrie o noua valoare, iar noua valoare a factorului de umplere este mai mica decat cea veche, latimea impulsului se micsoreaza;
- ✓ Daca iesirea PWM este inactiva in momentul in care se scrie o noua valoare, iar noua valoare este mai mare decat perioada de baza, iesirea PWM devine activa si ramane asa pana la scrierea unei noi valori pentru factorul de umplere.

Figure 14-14: Duty Cycle Update Times When Immediate Updates are Enabled (IUE = 1)



9. Exemplu

```
#include "p33FJ128MC802.h"
// Select Internal FRC at POR
_FOSCSEL(FNOSC_FRC);
// Enable Clock Switching and Configure
_FOSC(FCKSM_CSECMD & OSCIOFNC_OFF); // FRC + PLL
// _FOSC(FCKSM_CSECMD & OSCIOFNC_OFF & POSCMD_XT); // XT + PLL
_FWDT(FWDTEN_OFF); // Watchdog Timer Enabled/disabled by user software

void initPLL(void)
{
    // Configure PLL prescaler, PLL postscaler, PLL divisor
    PLLFBD = 41; // M = 43 FRC
    //PLLFBFBD = 30; // M = 32 XT
    CLKDIVbits.PLLPOST=0; // N1 = 2
    CLKDIVbits.PLLPRE=0; // N2 = 2

    // Initiate Clock Switch to Internal FRC with PLL (NOSC = 0b001)
    __builtin_write_OSCCONH(0x01); // FRC
    // __builtin_write_OSCCONH(0x03); // XT
    __builtin_write_OSCCONL(0x01);

    // Wait for Clock switch to occur
    while (OSCCONbits.COSC != 0b001); // FRC
    //while (OSCCONbits.COSC != 0b011); // XT

    // Wait for PLL to lock
    while(OSCCONbits.LOCK!=1) {};
}

void init_PWM1(void);
void init_PWM2(void);

int main()
{
    initPLL();
    init_PWM1();
    init_PWM2();
    while(1){} // Infinite Loop
}

void init_PWM1()
{
    P1TCONbits.PTOPS = 0; // Timer base output scale
    P1TCONbits.PTMOD = 0; // Free running

    P1TMRbits.PTDIR = 0; // Numara in sus pana cand timerul = perioada
    P1TMRbits.PTMR = 0; // Baza de timp

    P1DC1 = 0x2710;
    P1DC2 = 0x4E20;
    P1DC3 = 0x7530;

    P1TPER = 0x4E20;

    PWM1CON1bits.PMOD1 = 1; // Canalele PWM1H si PWM1L sunt independente
    PWM1CON1bits.PMOD2 = 1; // Canalele PWM2H si PWM2L sunt independente
    PWM1CON1bits.PMOD3 = 0; // Canalele PWM3H si PWM3L sunt complementare

    PWM1CON1bits.PEN1H = 1; // Pinul PWM1H setat pe iesire PWM
    PWM1CON1bits.PEN1L = 0; // Pinul PWM1L setat pe I/O general purpose

    PWM1CON1bits.PEN2H = 1; // Pinul PWM1H setat pe iesire PWM
    PWM1CON1bits.PEN2L = 0; // Pinul PWM1L setat pe I/O general purpose

    PWM1CON1bits.PEN3H = 1; // Pinul PWM1H setat pe iesire PWM
```

[illegible]