TI DSP,MCU 및 Xilinux Zynq FPGA

프로그래밍 전문가 과정

이름	문지희
학생 이메일	mjh8127@naver.com
날짜	2018/5/11
수업일수	52 일차
담당강사	Innova Lee(이상훈)
강사 이메일	gcccompil3r@gmail.com

목차

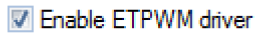
PINMUX Based Enhanced PWM 제어 (LED 밝기 제어)

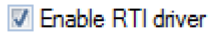
[CCS]

RTIETPWM으로 프로젝트 생성

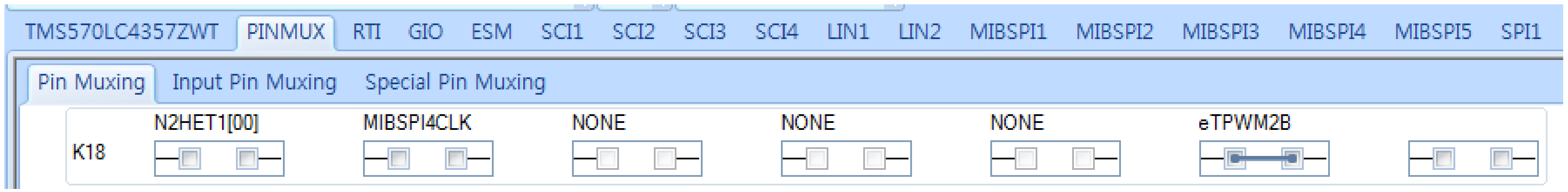
include option설정

[HALcogen]


☑ Enable ETPWM driver

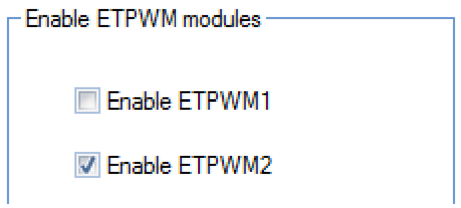

☑ Enable RTI driver

PINMUX-Pin Muxing의 K18에서 eTPWM2B연결



K18: N2HET1[00] | MIBSPI4CLK | NONE | NONE | NONE | eTPWM2B ✓

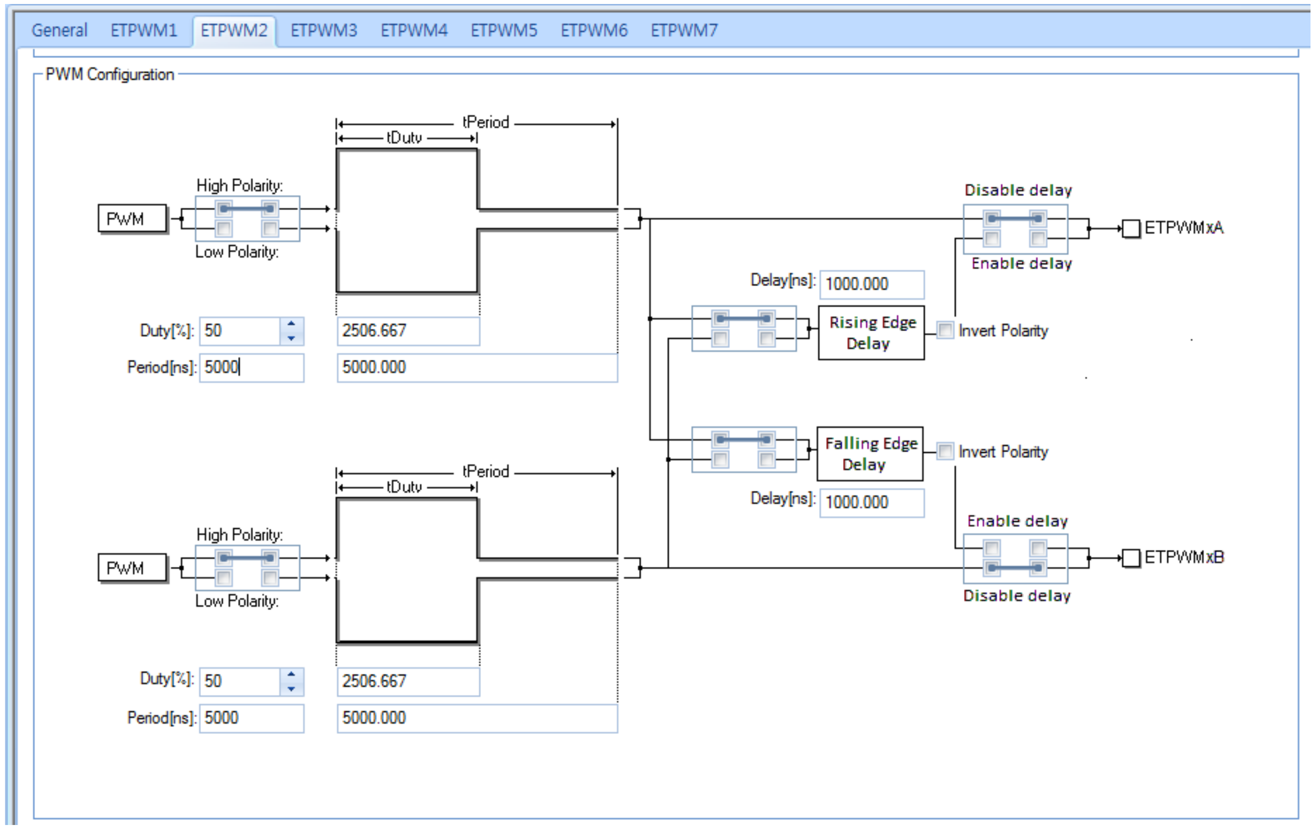
ETPWM - General - Enable ETPWM modules에서 Enalbe ETPWM2만 선택


Enable ETPWM modules

☐ Enable ETPWM1

☑ Enable ETPWM2

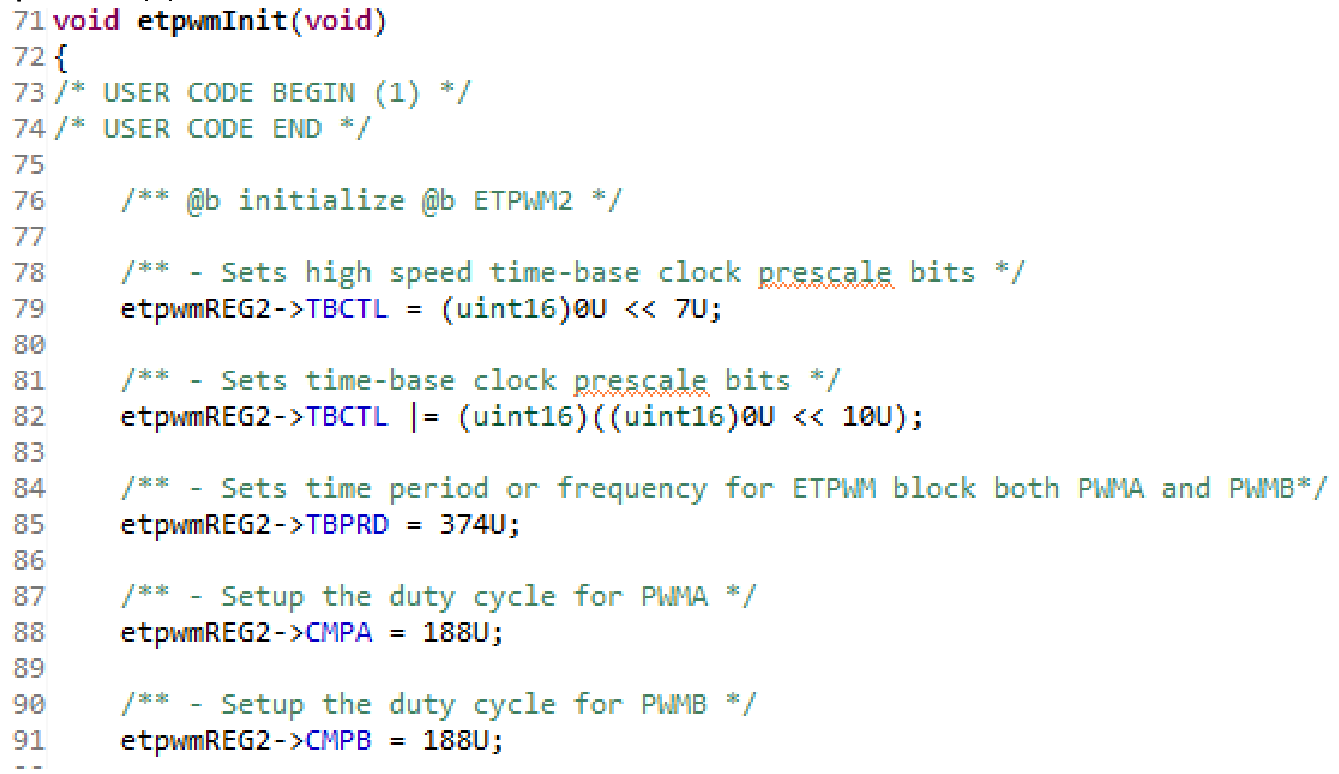
ETPWM - ETPWM2 - PWM Configuration에서 Period를 5000[ns]로 설정



Duty[%]: 50 | 2506.667
Period[ns]: 5000 | 5000.000
Delay[ns]: 1000.000
Delay[ns]: 1000.000
Duty[%]: 50 | 2506.667
Period[ns]: 5000 | 5000.000

```
#include "HL_sys_common.h"
#include "HL_etpwm.h"

int main(void)
{

    int i;
    unsigned short val = 0;
    etpwmInit();
    for(;;)
    {
        etpwmStartTBCLK();

        for(i=0; i<100000; i++)
            ;

        etpwmSetCmpB(etpwmREG2, val);

        val++;

        etpwmStopTBCLK();

        if(val == 500)
            val = 0;
    }

    return 0;
}
```

etpwmInit(1)



```
71 void etpwmInit(void)
72 {
73 /* USER CODE BEGIN (1) */
74 /* USER CODE END */
75
76     /** @b initialize @b ETPWM2 */
77
78     /** - Sets high speed time-base clock prescale bits */
79     etpwmREG2->TBCTL = (uint16)0U << 7U;
80
81     /** - Sets time-base clock prescale bits */
82     etpwmREG2->TBCTL |= (uint16)((uint16)0U << 10U);
83
84     /** - Sets time period or frequency for ETPWM block both PWMA and PWMB*/
85     etpwmREG2->TBPRD = 374U;
86
87     /** - Setup the duty cycle for PWMA */
88     etpwmREG2->CMPA = 188U;
89
90     /** - Setup the duty cycle for PWMB */
91     etpwmREG2->CMPB = 188U;
```

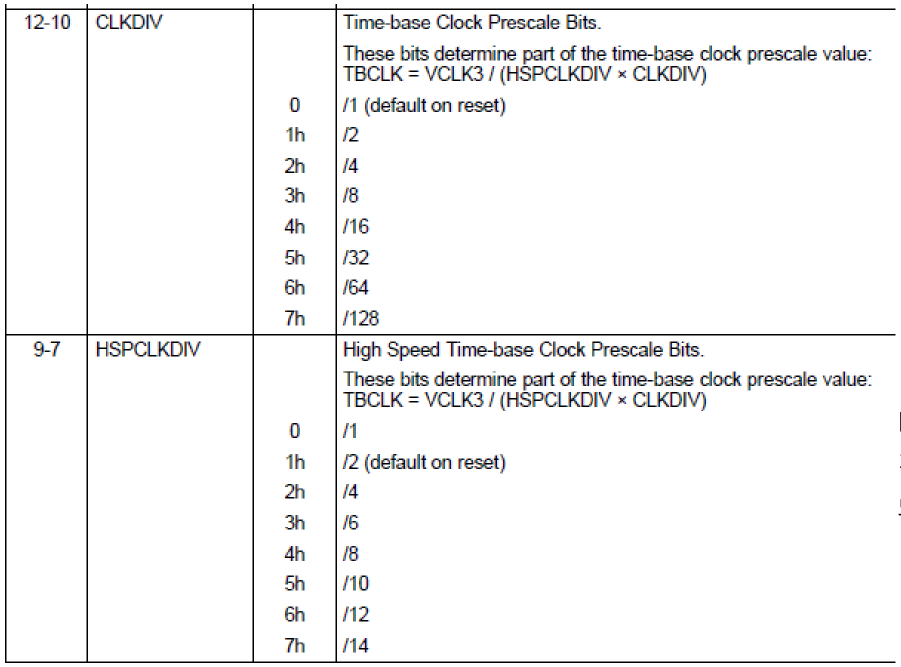
TBCTL

:79 etpwmREG2->TBCTL = (uint16)0U << 7U;

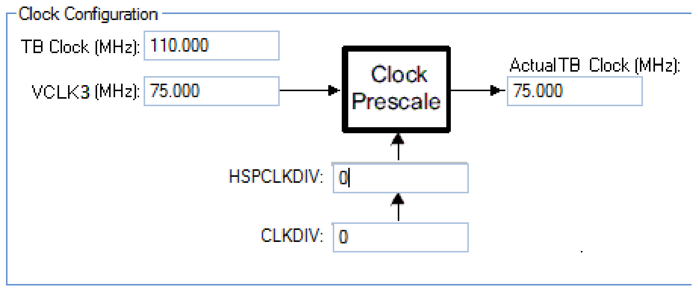
7번 비트를 0으로 셋 → 분주 x

:82 etpwmREG2->TBCTL |= (uint16)((uint16)0U << 10U);

10번 비트를 0으로 셋 → 분주 x



12-10	CLKDIV		Time-base Clock Prescale Bits.
			These bits determine part of the time-base clock prescale value: TBCLK = VCLK3 / (HSPCLKDIV × CLKDIV)
		0	/1 (default on reset)
		1h	/2
		2h	/4
		3h	/8
		4h	/16
		5h	/32
		6h	/64
		7h	/128
9-7	HSPCLKDIV		High Speed Time-base Clock Prescale Bits.
			These bits determine part of the time-base clock prescale value: TBCLK = VCLK3 / (HSPCLKDIV × CLKDIV)
		0	/1
		1h	/2 (default on reset)
		2h	/4
		3h	/6
		4h	/8
		5h	/10
		6h	/12
		7h	/14



Clock Configuration

TB Clock (MHz): 110.000

VCLK3 (MHz): 75.000

Actual TB Clock (MHz): 75.000

HSPCLKDIV: 0

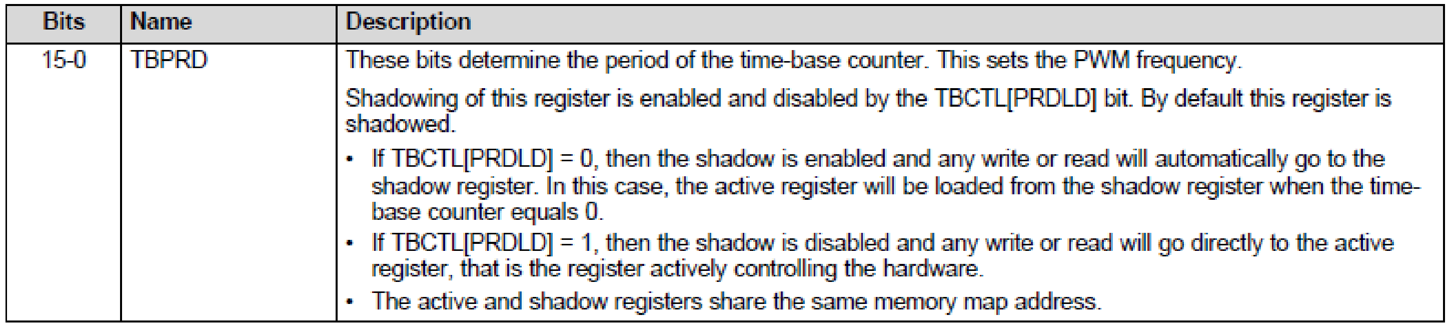
CLKDIV: 0

HALcogen에서 clock configuration을 살펴보면 HSPCLKDIV와 CLKDIV가 0인 것을 확인 가능. 그래서 VCLK3가 75MHz로 들어와 TB clock도 동일하게 분주되지 않고 75MHz가 되는 것.

TBPRD

```
:85        etpwmREG2->TBPRD = 374U;
```

Table 35-26. Time-Base Period Register (TBPRD) Field Descriptions



Bits	Name	Description
15-0	TBPRD	These bits determine the period of the time-base counter. This sets the PWM frequency. Shadowing of this register is enabled and disabled by the TBCTL[PRDLD] bit. By default this register is shadowed. • If TBCTL[PRDLD] = 0, then the shadow is enabled and any write or read will automatically go to the shadow register. In this case, the active register will be loaded from the shadow register when the time-base counter equals 0. • If TBCTL[PRDLD] = 1, then the shadow is disabled and any write or read will go directly to the active register, that is the register actively controlling the hardware. • The active and shadow registers share the same memory map address.

주기를 결정하는 레지스터

PWM주파수를 세팅한다.

374번 펄스가 뛴다는 의미.

Period를 10000으로 바꾸면 etpwmREG2->TBPRD = 749U; 로 코드 바뀜

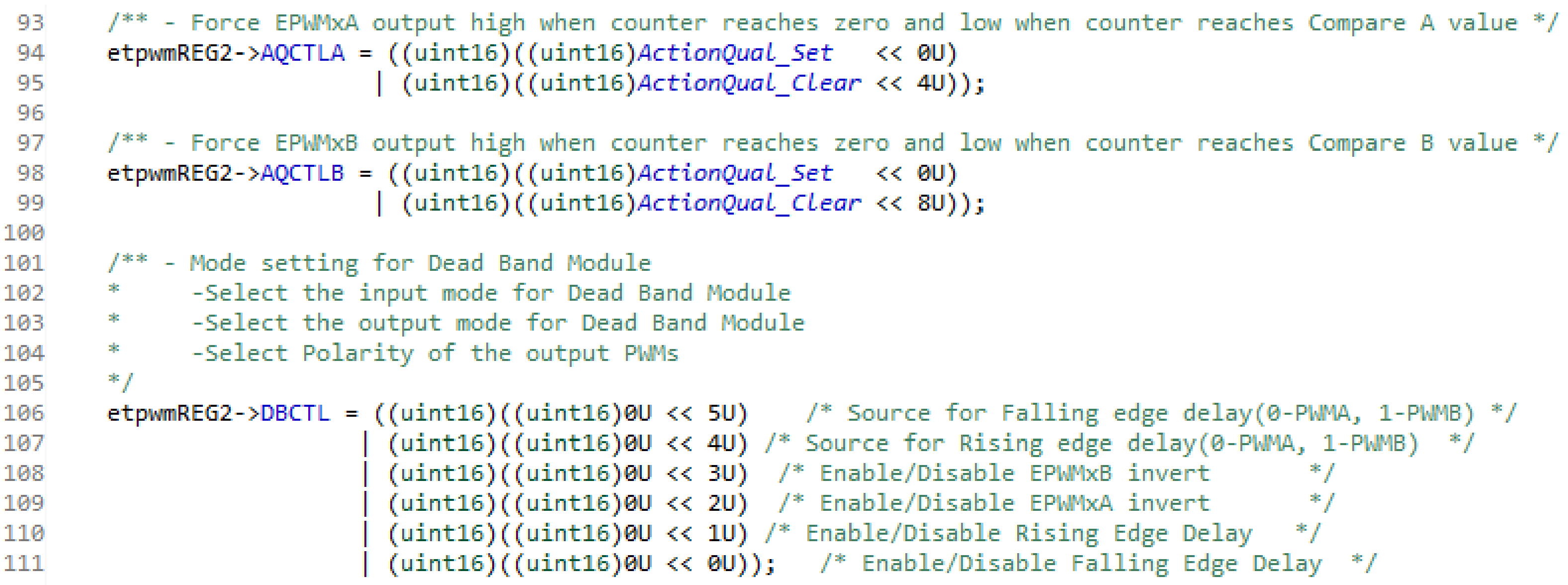
CMPA

:88 etpwmREG2->CMPA = 188U;

CMPA레지스터의 값은 TBPRD의 값과 연속적으로 비교한다. 두 레지스터의 값이 같아질 때 "time-base counter equal to counter compare A"이벤트(인터럽트)를 발생시킴.

188번째의 펄스 374/2 = 187이므로 1인 경우에서 0이 되었을 때 인터럽트를 발생시키는 것 같음.

etpwmInit(2)



```
/** - Force EPWMxA output high when counter reaches zero and low when counter reaches Compare A value */
etpwmREG2->AQCTLA = ((uint16)((uint16)ActionQual_Set   << 0U)
                   | (uint16)((uint16)ActionQual_Clear << 4U));

/** - Force EPWMxB output high when counter reaches zero and low when counter reaches Compare B value */
etpwmREG2->AQCTLB = ((uint16)((uint16)ActionQual_Set   << 0U)
                   | (uint16)((uint16)ActionQual_Clear << 8U));

/** - Mode setting for Dead Band Module
*      -Select the input mode for Dead Band Module
*      -Select the output mode for Dead Band Module
*      -Select Polarity of the output PWMs
*/
etpwmREG2->DBCTL = ((uint16)((uint16)0U << 5U)    /* Source for Falling edge delay(0-PWMA, 1-PWMB) */
                  | (uint16)((uint16)0U << 4U) /* Source for Rising edge delay(0-PWMA, 1-PWMB)  */
                  | (uint16)((uint16)0U << 3U)  /* Enable/Disable EPWMxB invert       */
                  | (uint16)((uint16)0U << 2U)  /* Enable/Disable EPWMxA invert       */
                  | (uint16)((uint16)0U << 1U) /* Enable/Disable Rising Edge Delay   */
                  | (uint16)((uint16)0U << 0U));   /* Enable/Disable Falling Edge Delay  */
```

A

ActionQual_set : 2

ActionQual_Clear : 1

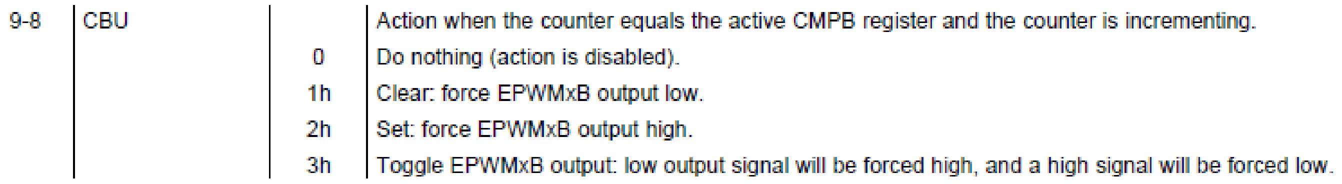
```
:94         etpwmREG2 ->AQCTLA = ((uint16)((uint16)ActionQual_Set    << 0U)
                        | (uint16)((uint16)ActionQual_Clear << 4U));
```

비트 1, 비트 4를 1로 세팅한다.

halcogen에서 B를 지정했기 때문에 A는 사용하지 않음.

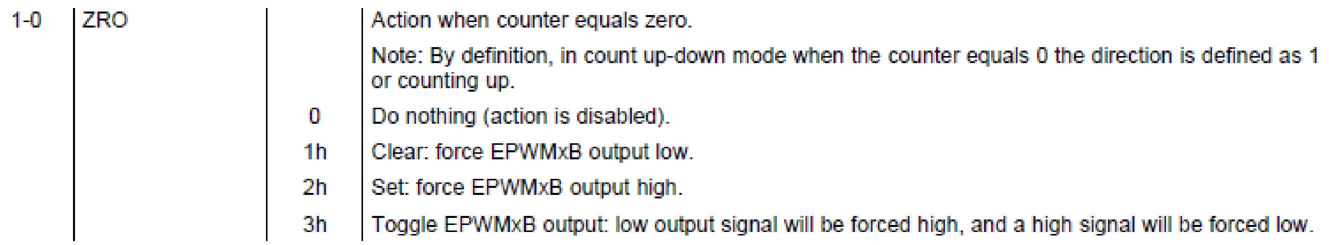
AQCTLB

```
:98         etpwmREG2->AQCTLB = ((uint16)((uint16)ActionQual_Set    << 0U)
                        | (uint16)((uint16)ActionQual_Clear << 8U));
```



9-8	CBU		Action when the counter equals the active CMPB register and the counter is incrementing.
		0	Do nothing (action is disabled).
		1h	Clear: force EPWMxB output low.
		2h	Set: force EPWMxB output high.
		3h	Toggle EPWMxB output: low output signal will be forced high, and a high signal will be forced low.

카운터가 cmpB레지스터랑 같을 때 EPWMxB의 output low 함.



1-0	ZRO		Action when counter equals zero. Note: By definition, in count up-down mode when the counter equals 0 the direction is defined as 1 or counting up.
		0	Do nothing (action is disabled).
		1h	Clear: force EPWMxB output low.
		2h	Set: force EPWMxB output high.
		3h	Toggle EPWMxB output: low output signal will be forced high, and a high signal will be forced low.

카운터가 0으로 같을 때 EPWMxB의 출력을 low로 함

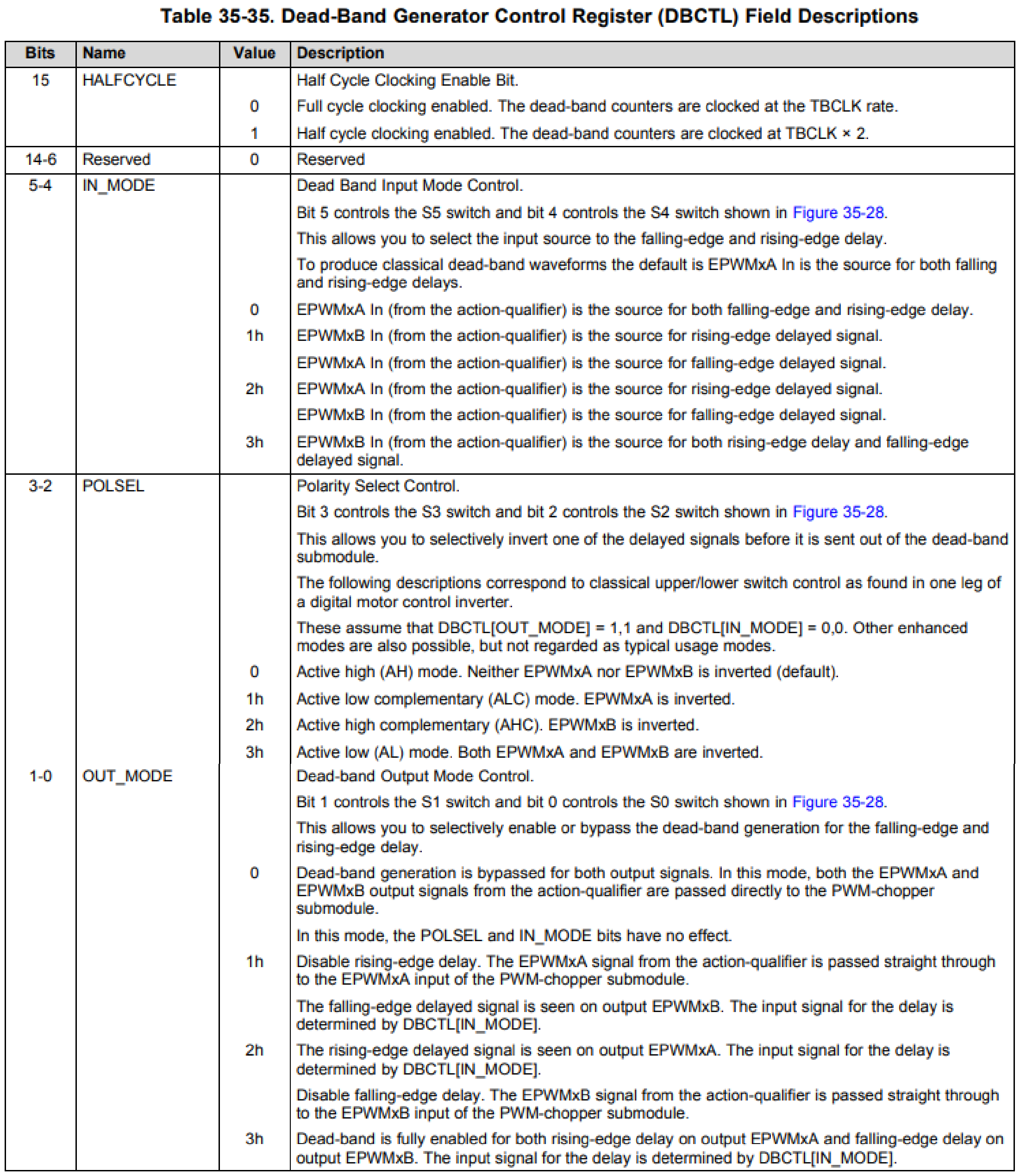


Table 35-35. Dead-Band Generator Control Register (DBCTL) Field Descriptions

Bits	Name	Value	Description
15	HALFCYCLE		Half Cycle Clocking Enable Bit.
		0	Full cycle clocking enabled. The dead-band counters are clocked at the TBCLK rate.
		1	Half cycle clocking enabled. The dead-band counters are clocked at TBCLK × 2.
14-6	Reserved	0	Reserved
5-4	IN_MODE		Dead Band Input Mode Control.
			Bit 5 controls the S5 switch and bit 4 controls the S4 switch shown in Figure 35-28.
			This allows you to select the input source to the falling-edge and rising-edge delay.
			To produce classical dead-band waveforms the default is EPWMxA In is the source for both falling and rising-edge delays.
		0	EPWMxA In (from the action-qualifier) is the source for both falling-edge and rising-edge delay.
		1h	EPWMxB In (from the action-qualifier) is the source for rising-edge delayed signal.
			EPWMxA In (from the action-qualifier) is the source for falling-edge delayed signal.
		2h	EPWMxA In (from the action-qualifier) is the source for rising-edge delayed signal.
			EPWMxB In (from the action-qualifier) is the source for falling-edge delayed signal.
		3h	EPWMxB In (from the action-qualifier) is the source for both rising-edge delay and falling-edge delayed signal.
3-2	POLSEL		Polarity Select Control.
			Bit 3 controls the S3 switch and bit 2 controls the S2 switch shown in Figure 35-28.
			This allows you to selectively invert one of the delayed signals before it is sent out of the dead-band submodule.
			The following descriptions correspond to classical upper/lower switch control as found in one leg of a digital motor control inverter.
			These assume that DBCTL[OUT_MODE] = 1,1 and DBCTL[IN_MODE] = 0,0. Other enhanced modes are also possible, but not regarded as typical usage modes.
		0	Active high (AH) mode. Neither EPWMxA nor EPWMxB is inverted (default).
		1h	Active low complementary (ALC) mode. EPWMxA is inverted.
		2h	Active high complementary (AHC). EPWMxB is inverted.
		3h	Active low (AL) mode. Both EPWMxA and EPWMxB are inverted.
1-0	OUT_MODE		Dead-band Output Mode Control.
			Bit 1 controls the S1 switch and bit 0 controls the S0 switch shown in Figure 35-28.
			This allows you to selectively enable or bypass the dead-band generation for the falling-edge and rising-edge delay.
		0	Dead-band generation is bypassed for both output signals. In this mode, both the EPWMxA and EPWMxB output signals from the action-qualifier are passed directly to the PWM-chopper submodule.
			In this mode, the POLSEL and IN_MODE bits have no effect.
		1h	Disable rising-edge delay. The EPWMxA signal from the action-qualifier is passed straight through to the EPWMxA input of the PWM-chopper submodule.
			The falling-edge delayed signal is seen on output EPWMxB. The input signal for the delay is determined by DBCTL[IN_MODE].
		2h	The rising-edge delayed signal is seen on output EPWMxA. The input signal for the delay is determined by DBCTL[IN_MODE].
			Disable falling-edge delay. The EPWMxB signal from the action-qualifier is passed straight through to the EPWMxB input of the PWM-chopper submodule.
		3h	Dead-band is fully enabled for both rising-edge delay on output EPWMxA and falling-edge delay on output EPWMxB. The input signal for the delay is determined by DBCTL[IN_MODE].

:106 etpwmREG2->DBCTL

비트 5~ 비트0 전부 0으로 지정.

Dead-band : 제어계에서 입력이 변화해도 출력이 생기지 않는 입력의 범위, 허용 할 수 있는 전압의 한계시점을 넘어간 것. 어셈에서의 포화연산

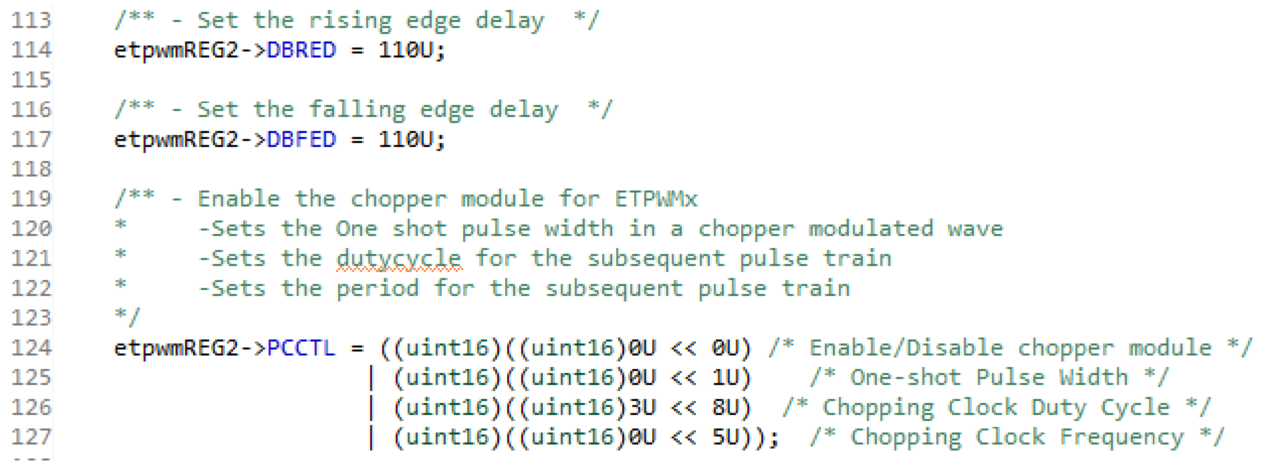
5, 4 : Dead Band의 입력모드를 조정한다. falling-edge와 rising-edge 두가지를 지연시키기 위한 소스가 EPWMxA에 존재

3, 2 : 극성 선택제어를 한다. 비트 3, 2 모두 0이므로 HIGH일 때 동작하고 EPWMxA 나 EPWMxB가 반전되지 않음.

1, 0 : Dead-band의 출력모드를 조정

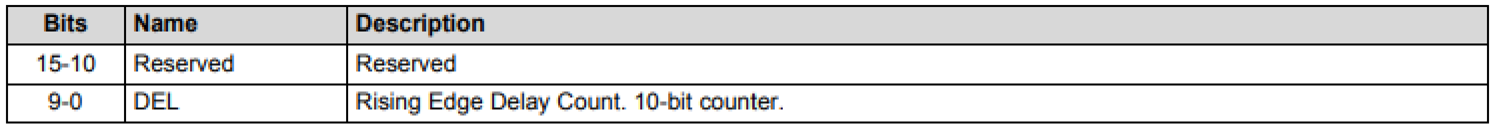
Dead-band의 생성은 두개의 출력신호에 대해 바이패스 된다. 이 모드에서 EPWMxA와 EPWMxB의 두개의 출력신호는 PWM-chopper로 직접 전달된다.

Chopper - 신호를 성형시킴



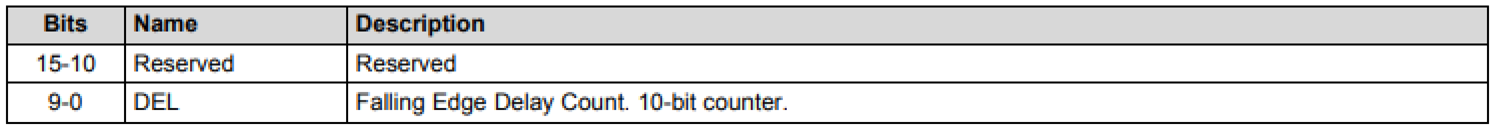
```
113     /** - Set the rising edge delay  */
114     etpwmREG2->DBRED = 110U;
115
116     /** - Set the falling edge delay  */
117     etpwmREG2->DBFED = 110U;
118
119     /** - Enable the chopper module for ETPWMx
120     *     -Sets the One shot pulse width in a chopper modulated wave
121     *     -Sets the dutycycle for the subsequent pulse train
122     *     -Sets the period for the subsequent pulse train
123     */
124     etpwmREG2->PCCTL = ((uint16)((uint16)0U << 0U) /* Enable/Disable chopper module */
125                       | (uint16)((uint16)0U << 1U)    /* One-shot Pulse Width */
126                       | (uint16)((uint16)3U << 8U)  /* Chopping Clock Duty Cycle */
127                       | (uint16)((uint16)0U << 5U));  /* Chopping Clock Frequency */
```

Table 35-37. Dead-Band Generator Rising Edge Delay Register (DBRED) Field Descriptions



Bits	Name	Description
15-10	Reserved	Reserved
9-0	DEL	Rising Edge Delay Count. 10-bit counter.

Table 35-36. Dead-Band Generator Falling Edge Delay Register (DBFED) Field Descriptions



Bits	Name	Description
15-10	Reserved	Reserved
9-0	DEL	Falling Edge Delay Count. 10-bit counter.

:114 etpwmREG2->DBRED = 110U;

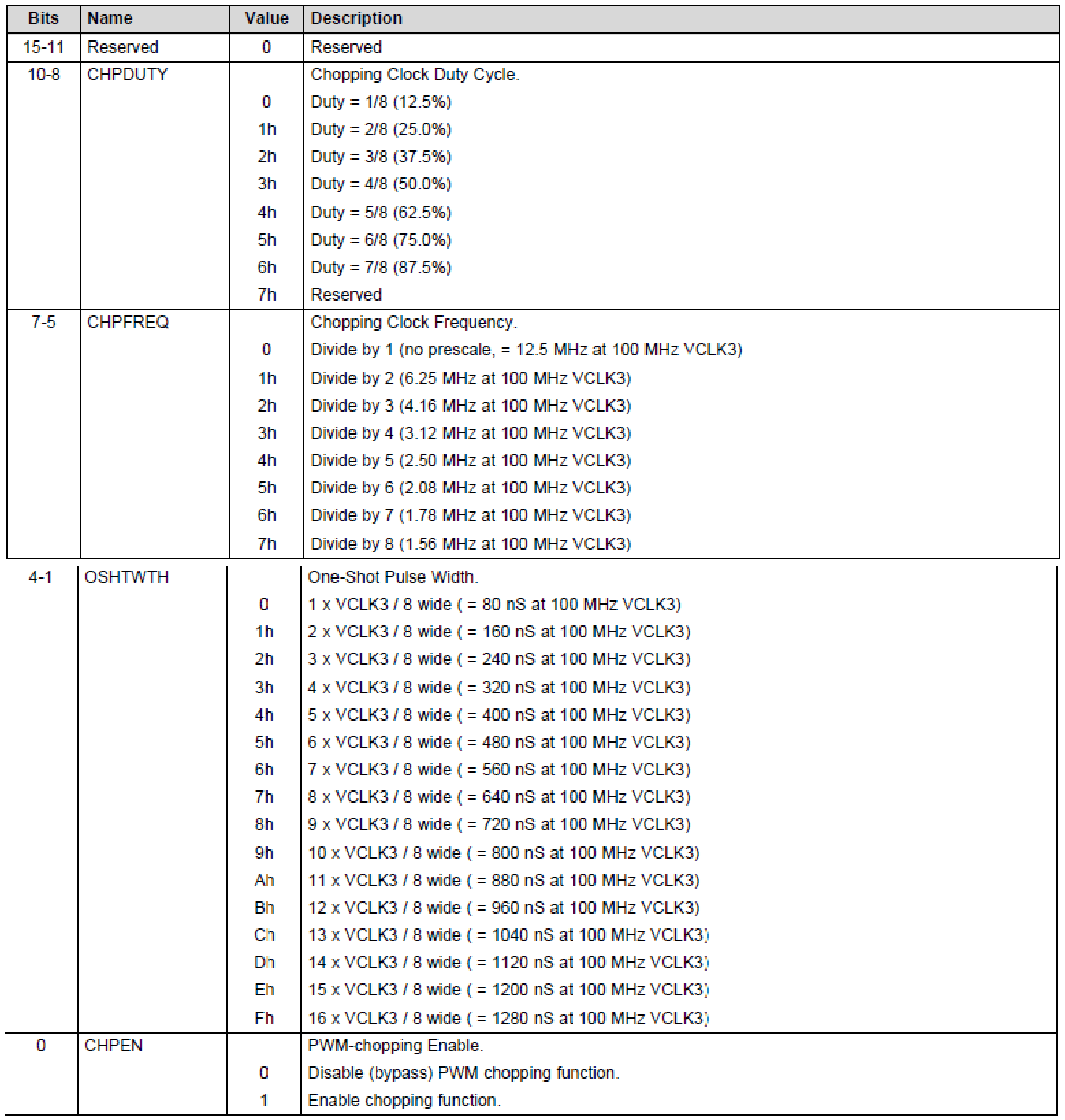
:117 etpwmREG2->DBFED = 110U;

rising edge와 falling edge를 110만큼 지연시킨다.

실제로는 sin파 신호인데 딜레이를 주어 위의 둥근 부분의 신호를 깎음 그래서 펄스파처럼 됨.??

:124 etpwmREG2->PCCTL

Table 35-50. PWM-Chopper Control Register (PCCTL) Bit Descriptions



Bits	Name	Value	Description
15-11	Reserved	0	Reserved
10-8	CHPDUTY		Chopping Clock Duty Cycle.
		0	Duty = 1/8 (12.5%)
		1h	Duty = 2/8 (25.0%)
		2h	Duty = 3/8 (37.5%)
		3h	Duty = 4/8 (50.0%)
		4h	Duty = 5/8 (62.5%)
		5h	Duty = 6/8 (75.0%)
		6h	Duty = 7/8 (87.5%)
		7h	Reserved
7-5	CHPFREQ		Chopping Clock Frequency.
		0	Divide by 1 (no prescale, = 12.5 MHz at 100 MHz VCLK3)
		1h	Divide by 2 (6.25 MHz at 100 MHz VCLK3)
		2h	Divide by 3 (4.16 MHz at 100 MHz VCLK3)
		3h	Divide by 4 (3.12 MHz at 100 MHz VCLK3)
		4h	Divide by 5 (2.50 MHz at 100 MHz VCLK3)
		5h	Divide by 6 (2.08 MHz at 100 MHz VCLK3)
		6h	Divide by 7 (1.78 MHz at 100 MHz VCLK3)
		7h	Divide by 8 (1.56 MHz at 100 MHz VCLK3)
4-1	OSHTWTH		One-Shot Pulse Width.
		0	1 x VCLK3 / 8 wide (= 80 nS at 100 MHz VCLK3)
		1h	2 x VCLK3 / 8 wide (= 160 nS at 100 MHz VCLK3)
		2h	3 x VCLK3 / 8 wide (= 240 nS at 100 MHz VCLK3)
		3h	4 x VCLK3 / 8 wide (= 320 nS at 100 MHz VCLK3)
		4h	5 x VCLK3 / 8 wide (= 400 nS at 100 MHz VCLK3)
		5h	6 x VCLK3 / 8 wide (= 480 nS at 100 MHz VCLK3)
		6h	7 x VCLK3 / 8 wide (= 560 nS at 100 MHz VCLK3)
		7h	8 x VCLK3 / 8 wide (= 640 nS at 100 MHz VCLK3)
		8h	9 x VCLK3 / 8 wide (= 720 nS at 100 MHz VCLK3)
		9h	10 x VCLK3 / 8 wide (= 800 nS at 100 MHz VCLK3)
		Ah	11 x VCLK3 / 8 wide (= 880 nS at 100 MHz VCLK3)
		Bh	12 x VCLK3 / 8 wide (= 960 nS at 100 MHz VCLK3)
		Ch	13 x VCLK3 / 8 wide (= 1040 nS at 100 MHz VCLK3)
		Dh	14 x VCLK3 / 8 wide (= 1120 nS at 100 MHz VCLK3)
		Eh	15 x VCLK3 / 8 wide (= 1200 nS at 100 MHz VCLK3)
		Fh	16 x VCLK3 / 8 wide (= 1280 nS at 100 MHz VCLK3)
0	CHPEN		PWM-chopping Enable.
		0	Disable (bypass) PWM chopping function.
		1	Enable chopping function.

8,9번비트 1로 지정, 0, 1, 5 번 비트 0으로 지정

CHPDUTY

3 이므로 duty = 4/8 (50%)

나머지는 비트 0

CHPFREQ

1이므로 Divide by 1(1분주)

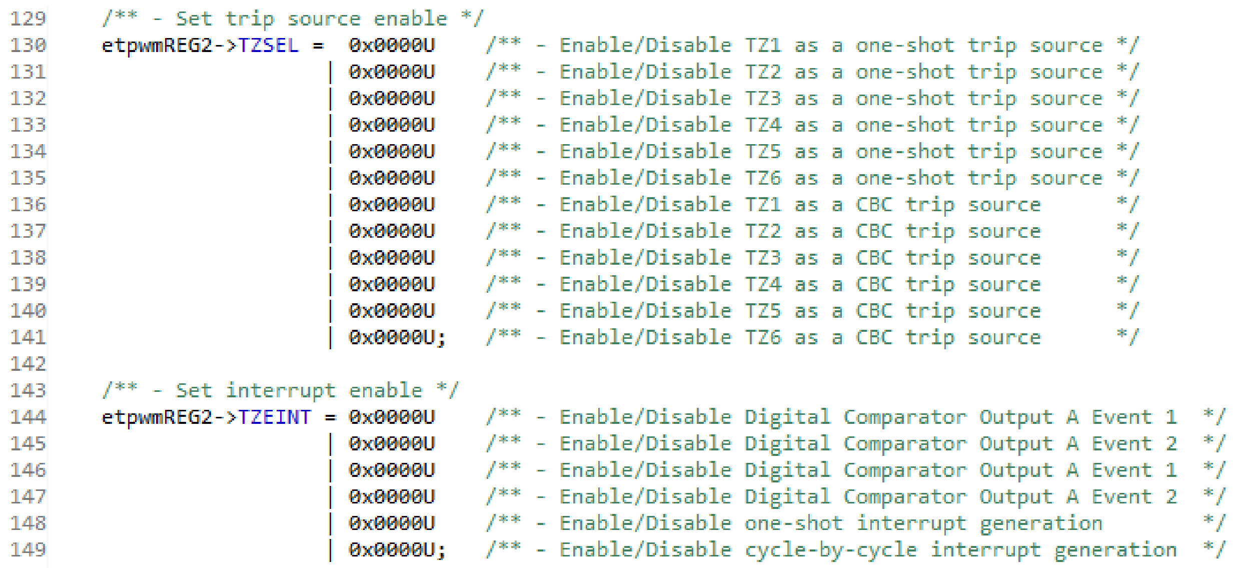
OSHTWTH

1이므로 1*VCLK3 / 8 wide

CHPEN

PWM-chopping 설정

PWM chopping function을 사용하지 않음.



```
129     /** - Set trip source enable */
130     etpwmREG2->TZSEL =  0x0000U    /** - Enable/Disable TZ1 as a one-shot trip source */
131                       | 0x0000U    /** - Enable/Disable TZ2 as a one-shot trip source */
132                       | 0x0000U    /** - Enable/Disable TZ3 as a one-shot trip source */
133                       | 0x0000U    /** - Enable/Disable TZ4 as a one-shot trip source */
134                       | 0x0000U    /** - Enable/Disable TZ5 as a one-shot trip source */
135                       | 0x0000U    /** - Enable/Disable TZ6 as a one-shot trip source */
136                       | 0x0000U    /** - Enable/Disable TZ1 as a CBC trip source      */
137                       | 0x0000U    /** - Enable/Disable TZ2 as a CBC trip source      */
138                       | 0x0000U    /** - Enable/Disable TZ3 as a CBC trip source      */
139                       | 0x0000U    /** - Enable/Disable TZ4 as a CBC trip source      */
140                       | 0x0000U    /** - Enable/Disable TZ5 as a CBC trip source      */
141                       | 0x0000U;   /** - Enable/Disable TZ6 as a CBC trip source      */
142
143     /** - Set interrupt enable */
144     etpwmREG2->TZEINT = 0x0000U    /** - Enable/Disable Digital Comparator Output A Event 1  */
145                       | 0x0000U    /** - Enable/Disable Digital Comparator Output A Event 2  */
146                       | 0x0000U    /** - Enable/Disable Digital Comparator Output A Event 1  */
147                       | 0x0000U    /** - Enable/Disable Digital Comparator Output A Event 2  */
148                       | 0x0000U    /** - Enable/Disable one-shot interrupt generation        */
149                       | 0x0000U;   /** - Enable/Disable cycle-by-cycle interrupt generation  */
```

:130 etpwmREG2->TZSEL

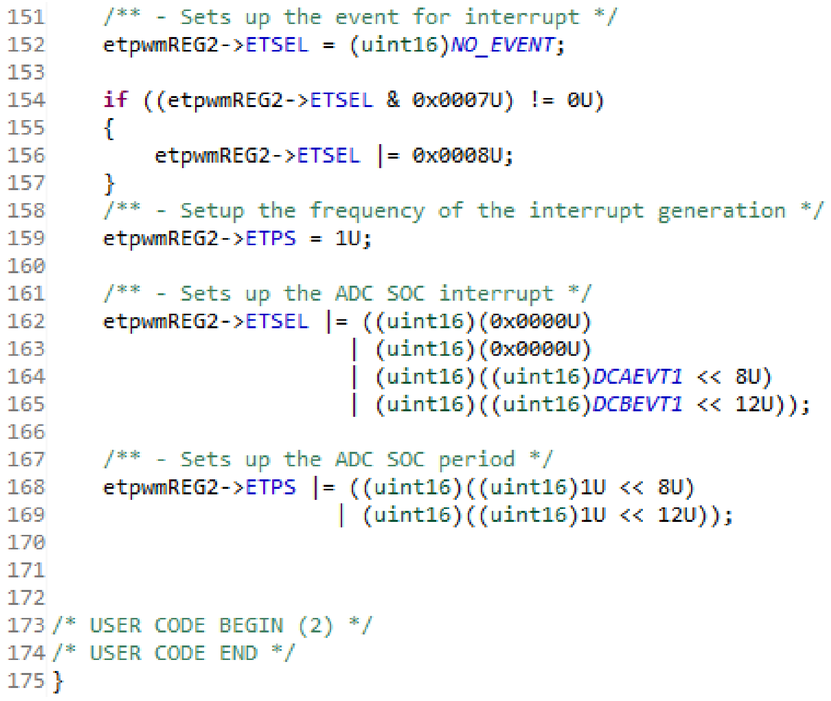
전부 0

인터럽트 전부 disable

:144 etpwmREG2->TZEINT

전부 0

다 disable. 인터럽트 막음



```
151     /** - Sets up the event for interrupt */
152     etpwmREG2->ETSEL = (uint16)NO_EVENT;
153
154     if ((etpwmREG2->ETSEL & 0x0007U) != 0U)
155     {
156         etpwmREG2->ETSEL |= 0x0008U;
157     }
158     /** - Setup the frequency of the interrupt generation */
159     etpwmREG2->ETPS = 1U;
160
161     /** - Sets up the ADC SOC interrupt */
162     etpwmREG2->ETSEL |= ((uint16)(0x0000U)
163                        | (uint16)(0x0000U)
164                        | (uint16)((uint16)DCAEVT1 << 8U)
165                        | (uint16)((uint16)DCBEVT1 << 12U));
166
167     /** - Sets up the ADC SOC period */
168     etpwmREG2->ETPS |= ((uint16)((uint16)1U << 8U)
169                       | (uint16)((uint16)1U << 12U));
170
171
172
173 /* USER CODE BEGIN (2) */
174 /* USER CODE END */
175 }
```

:152 etpwmREG2->ETSEL = (uint16)NO_EVENT;

NO_EVENT = 0

:154 ETSEL이 0이기 때문에 if문 해당되지 않아 제낀다.

:159 etpwmREG2->ETPS = 1U;

:162 ETSEL 8,12번 비트 0으로 고정

DCAEVT1 : 0, DCBEVT2 : 0

:168 ETPS 8, 12번 1로 고정

Table 35-45. Event-Trigger Selection Register (ETSEL) Field Descriptions

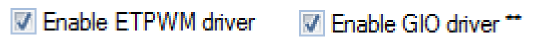
Bits	Name	Value	Description
15	SOCBEN		Enable the ADC Start of Conversion B (EPWMxSOCB) Pulse.
		0	Disable EPWMxSOCB.
		1	Enable EPWMxSOCB pulse.
14-12	SOCBSEL		EPWMxSOCB Selection Options.
			These bits determine when a EPWMxSOCB pulse will be generated.
		0	Enable DCBEVT1.soc event.
		1h	Enable event time-base counter equal to zero. (TBCTR = 0x0000).
		2h	Enable event time-base counter equal to period (TBCTR = TBPRD).
		3h	Enable event time-base counter equal to zero or period (TBCTR = 0x0000 or TBCTR = TBPRD). This mode is useful in up-down count mode.
		4h	Enable event time-base counter equal to CMPA when the timer is incrementing.
		5h	Enable event time-base counter equal to CMPA when the timer is decrementing.
		6h	Enable event: time-base counter equal to CMPB when the timer is incrementing.
		7h	Enable event: time-base counter equal to CMPB when the timer is decrementing.
11	SOCAEN		Enable the ADC Start of Conversion A (EPWMxSOCA) Pulse.
		0	Disable EPWMxSOCA.
		1	Enable EPWMxSOCA pulse.
10-8	SOCASEL		EPWMxSOCA Selection Options.
			These bits determine when a EPWMxSOCA pulse will be generated.
		0	Enable DCAEVT1.soc event.
		1h	Enable event time-base counter equal to zero. (TBCTR = 0x0000).
		2h	Enable event time-base counter equal to period (TBCTR = TBPRD).
		3h	Enable event time-base counter equal to zero or period (TBCTR = 0x0000 or TBCTR = TBPRD). This mode is useful in up-down count mode.
		4h	Enable event time-base counter equal to CMPA when the timer is incrementing.
		5h	Enable event time-base counter equal to CMPA when the timer is decrementing.
		6h	Enable event: time-base counter equal to CMPB when the timer is incrementing.
		7h	Enable event: time-base counter equal to CMPB when the timer is decrementing.
7-4	Reserved	0	Reserved
3	INTEN		Enable ePWM Interrupt (EPWMx_INT) Generation.
		0	Disable EPWMx_INT generation.
		1	Enable EPWMx_INT generation.

Table 35-47. Event-Trigger Prescale Register (ETPS) Field Descriptions

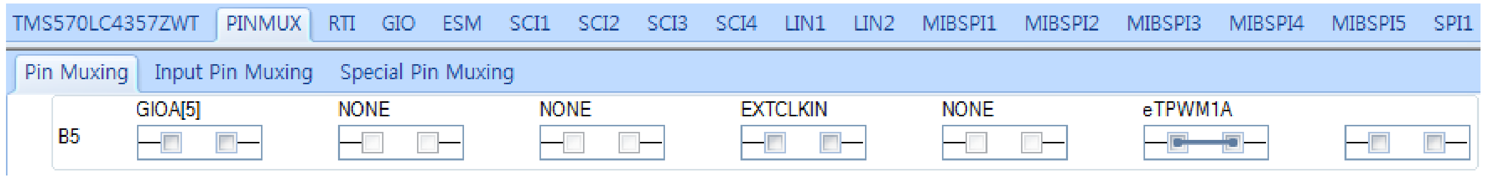
Bits	Name	Value	Description
15-14	SOCBCNT		ePWM ADC Start-of-Conversion B Event (EPWMxSOCB) Counter Register.
			These bits indicate how many selected ETSEL[SOCBSEL] events have occurred.
		0	No events have occurred.
		1h	1 event has occurred.
		2h	2 events have occurred.
		3h	3 events have occurred.
13-12	SOCBPRD		ePWM ADC Start-of-Conversion B Event (EPWMxSOCB) Period Select.
			These bits determine how many selected ETSEL[SOCBSEL] events need to occur before an EPWMxSOCB pulse is generated. To be generated, the pulse must be enabled (ETSEL[SOCBEN] = 1). The SOCB pulse will be generated even if the status flag is set from a previous start of conversion (ETFLG[SOCB] = 1). Once the SOCB pulse is generated, the ETPS[SOCBCNT] bits will automatically be cleared.
		0	Disable the SOCB event counter. No EPWMxSOCB pulse will be generated.
		1h	Generate the EPWMxSOCB pulse on the first event: ETPS[SOCBCNT] = 0,1.
		2h	Generate the EPWMxSOCB pulse on the second event: ETPS[SOCBCNT] = 1,0.
		3h	Generate the EPWMxSOCB pulse on the third event: ETPS[SOCBCNT] = 1,1.
11-10	SOCACNT		ePWM ADC Start-of-Conversion A Event (EPWMxSOCA) Counter Register.
			These bits indicate how many selected ETSEL[SOCASEL] events have occurred.
		0	No events have occurred.
		1h	1 event has occurred.
		2h	2 events have occurred.
		3h	3 events have occurred.
9-8	SOCAPRD		ePWM ADC Start-of-Conversion A Event (EPWMxSOCA) Period Select.
			These bits determine how many selected ETSEL[SOCASEL] events need to occur before an EPWMxSOCA pulse is generated. To be generated, the pulse must be enabled (ETSEL[SOCAEN] = 1). The SOCA pulse will be generated even if the status flag is set from a previous start of conversion (ETFLG[SOCA] = 1). Once the SOCA pulse is generated, the ETPS[SOCACNT] bits will automatically be cleared.
		0	Disable the SOCA event counter. No EPWMxSOCA pulse will be generated.
		1h	Generate the EPWMxSOCA pulse on the first event: ETPS[SOCACNT] = 0,1.
		2h	Generate the EPWMxSOCA pulse on the second event: ETPS[SOCACNT] = 1,0.
		3h	Generate the EPWMxSOCA pulse on the third event: ETPS[SOCACNT] = 1,1.
7-4	Reserved	0	Reserved
1-0	INTPRD		ePWM Interrupt (EPWMx_INT) Period Select.
			These bits determine how many selected ETSEL[INTSEL] events need to occur before an interrupt is generated. To be generated, the interrupt must be enabled (ETSEL[INT] = 1). If the interrupt status flag is set from a previous interrupt (ETFLG[INT] = 1) then no interrupt will be generated until the flag is cleared via the ETCLR[INT] bit. This allows for one interrupt to be pending while another is still being serviced. Once the interrupt is generated, the ETPS[INTCNT] bits will automatically be cleared.
			Writing a INTPRD value that is the same as the current counter value will trigger an interrupt if it is enabled and the status flag is clear.
			Writing a INTPRD value that is less than the current counter value will result in an undefined state.
			If a counter event occurs at the same instant as a new zero or non-zero INTPRD value is written, the counter is incremented.
		0	Disable the interrupt event counter. No interrupt will be generated and ETFRC[INT] is ignored.
		1h	Generate an interrupt on the first event INTCNT = 01 (first event).
		2h	Generate interrupt on ETPS[INTCNT] = 1,0 (second event).
		3h	Generate interrupt on ETPS[INTCNT] = 1,1 (third event).

Enhanced PWM Peripheral 및 동작 과정 설명(모터제어)

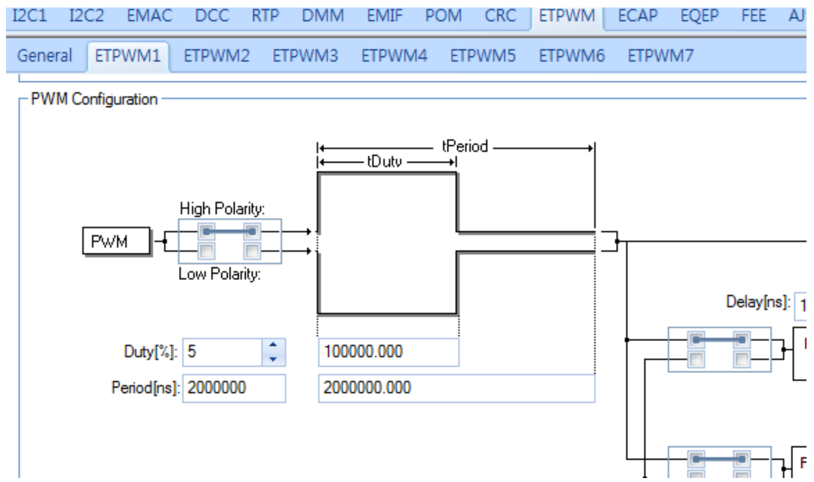
[HALcogen]



☑ Enable ETPWM driver ☑ Enable GIO driver **

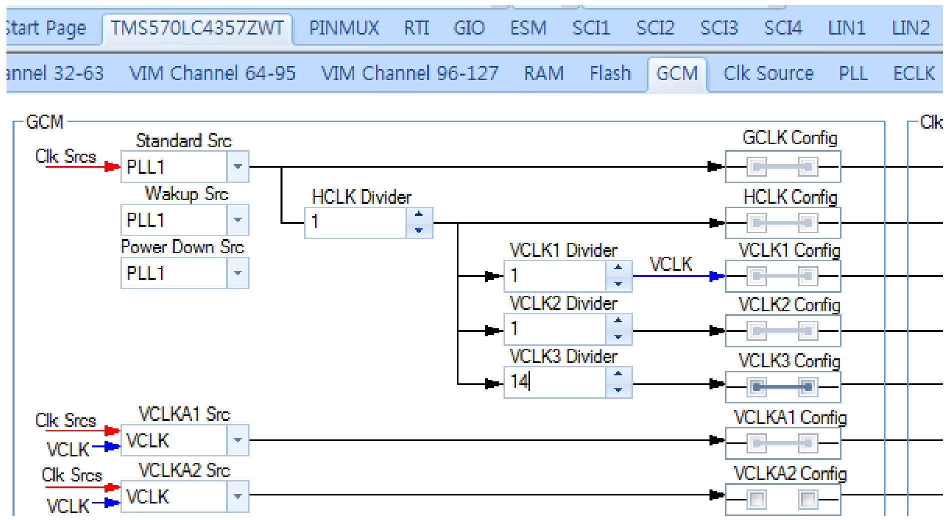


B5: GIOA[5] | NONE | NONE | EXTCLKIN | NONE | eTPWM1A (selected)



Duty[%]: 5 — 100000.000

Period[ns]: 2000000 — 2000000.000



Standard Src: PLL1

Wakup Src: PLL1

Power Down Src: PLL1

HCLK Divider: 1

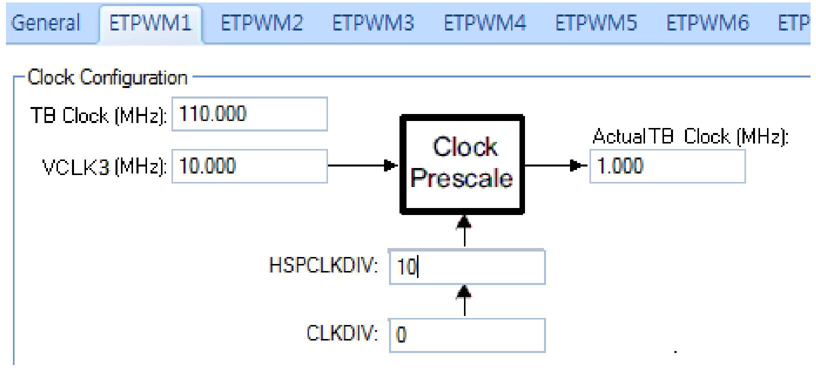
VCLK1 Divider: 1

VCLK2 Divider: 1

VCLK3 Divider: 14

VCLKA1 Src: VCLK

VCLKA2 Src: VCLK



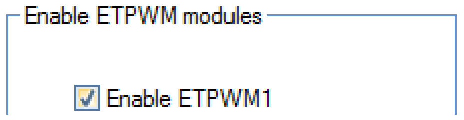
TB Clock (MHz): 110.000

VCLK3 (MHz): 10.000

ActualTB Clock (MHz): 1.000

HSPCLKDIV: 10

CLKDIV: 0

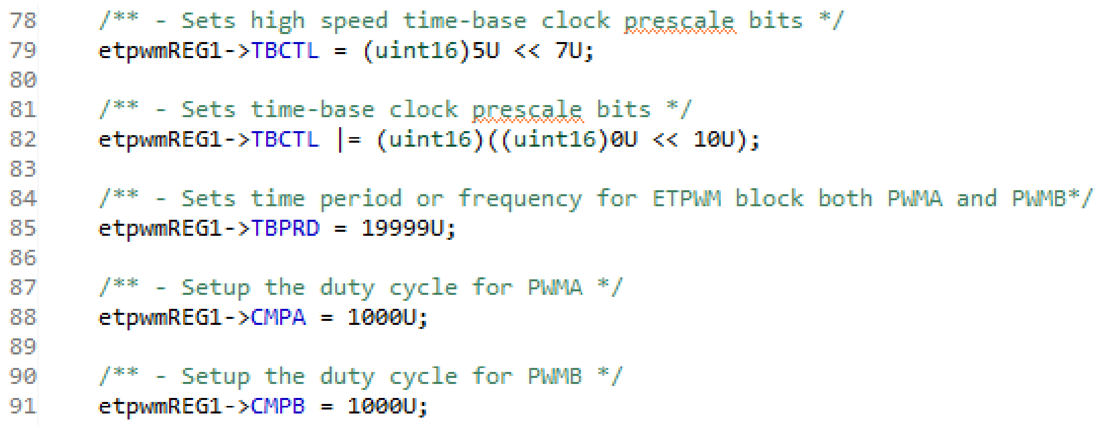


Enable ETPWM modules

☑ Enable ETPWM1

```
#include "HL_sys_common.h"
#include "HL_system.h"
#include "HL_etpwm.h"

int idx = 0;
uint32 value = 0;
uint32 duty_arr[20] = {1000, 1100, 1200, 1300, 1400,
                       1500, 1600, 1700, 1800, 1900,
                       2000, 1900, 1800, 1700, 1600,
                       1500, 1400, 1300, 1200, 1100};

void pwmSet(void);
void delay(uint32);

int main(void)
{
    etpwmInit();
    etpwmStartTBCLK();

    delay(10000);

    for(;;)
    {
        pwmSet();
        delay(5000000);
    }

    return 0;
}

void delay(uint32 delay)
{
    int i;

    for(i = 0; i < delay; i++)
        ;
}

void pwmSet(void)
{
    value = duty_arr[idx % 20];
    etpwmSetCmpA(etpwmREG1, value);
    idx++;
}
```

etpwmInit



```
78    /** - Sets high speed time-base clock prescale bits */
79    etpwmREG1->TBCTL = (uint16)5U << 7U;
80
81    /** - Sets time-base clock prescale bits */
82    etpwmREG1->TBCTL |= (uint16)((uint16)0U << 10U);
83
84    /** - Sets time period or frequency for ETPWM block both PWMA and PWMB*/
85    etpwmREG1->TBPRD = 19999U;
86
87    /** - Setup the duty cycle for PWMA */
88    etpwmREG1->CMPA = 1000U;
89
90    /** - Setup the duty cycle for PWMB */
91    etpwmREG1->CMPB = 1000U;
```

:79 7,9번 비트 1로 셋

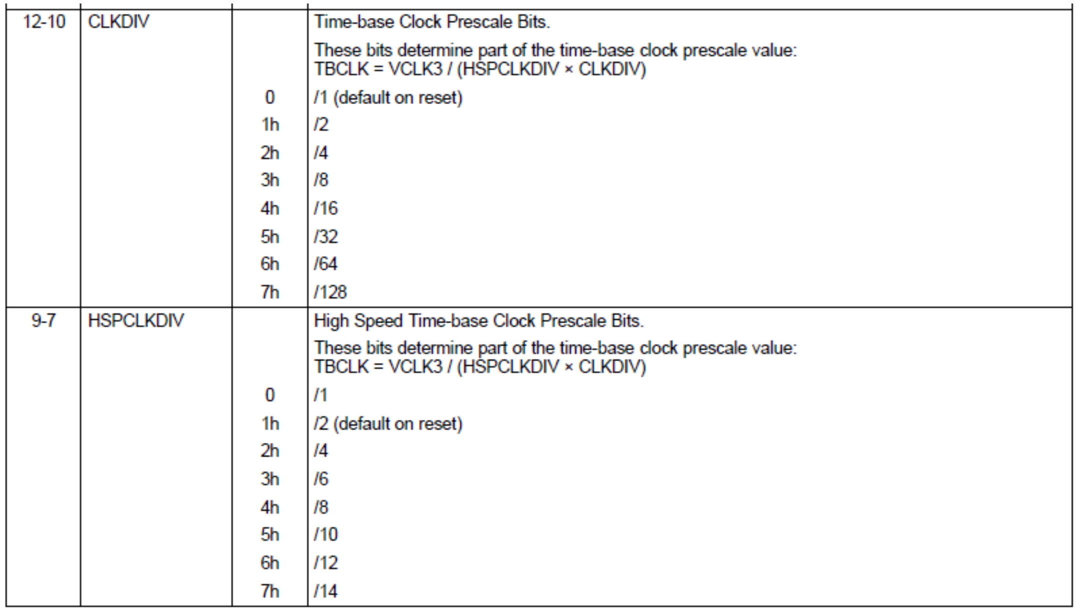
:82 10번 비트 0으로 고정

:85 TBPRD

time-base 카운터의 주기를 19999로 정함.

:88 CMPA, :91 CMPB

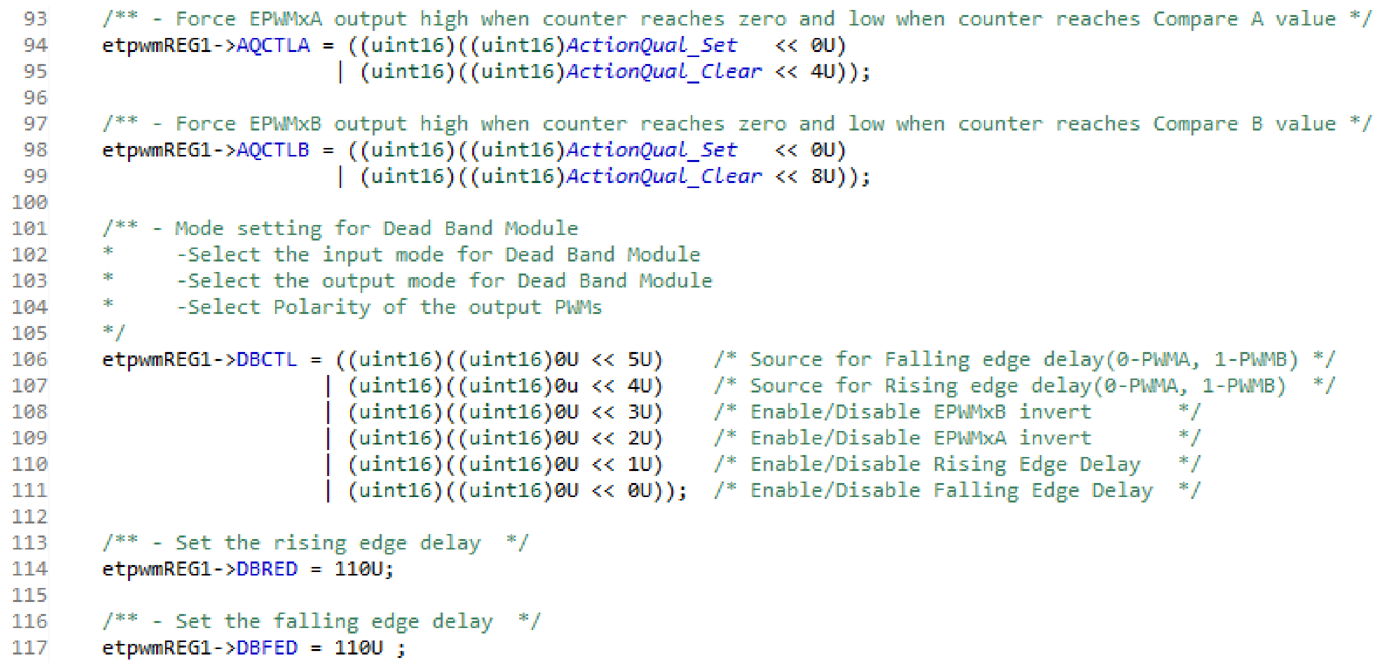
TBCTR의 값과 계속 비교하여 1000이 되어 같아지면 인터럽트 발생



12-10	CLKDIV		Time-base Clock Prescale Bits. These bits determine part of the time-base clock prescale value: TBCLK = VCLK3 / (HSPCLKDIV × CLKDIV)
		0	/1 (default on reset)
		1h	/2
		2h	/4
		3h	/8
		4h	/16
		5h	/32
		6h	/64
		7h	/128
9-7	HSPCLKDIV		High Speed Time-base Clock Prescale Bits. These bits determine part of the time-base clock prescale value: TBCLK = VCLK3 / (HSPCLKDIV × CLKDIV)
		0	/1
		1h	/2 (default on reset)
		2h	/4
		3h	/6
		4h	/8
		5h	/10
		6h	/12
		7h	/14

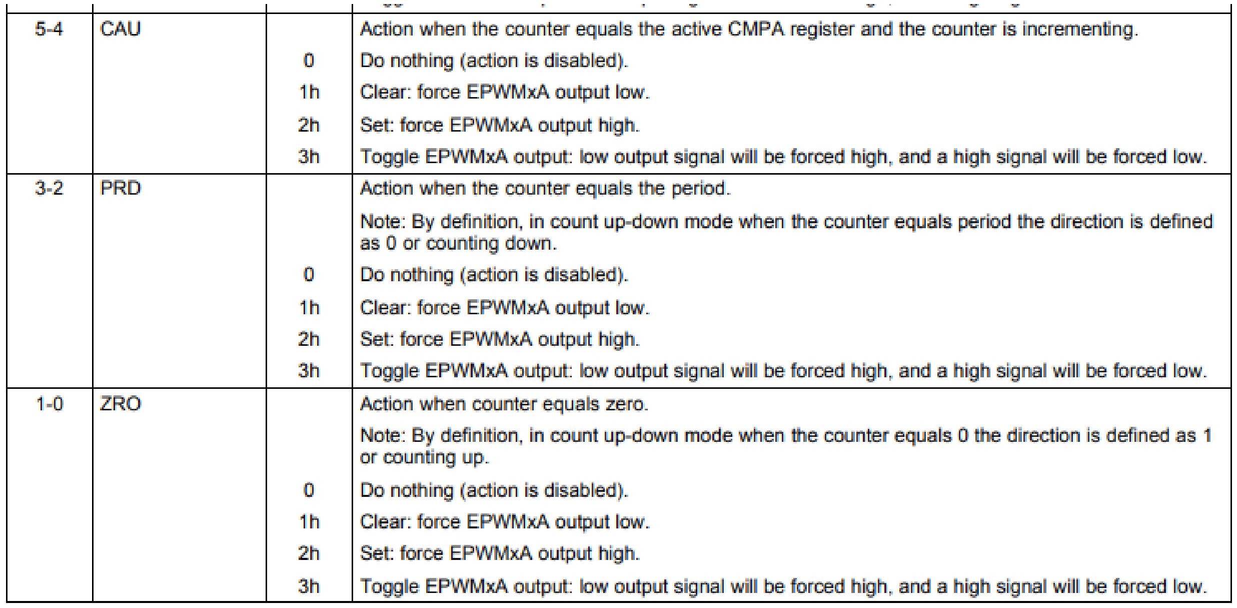
:79 HSPCLKDIV 10분주함

:82 CLKDIV는 분주 없음.



```
93      /** - Force EPWMxA output high when counter reaches zero and low when counter reaches Compare A value */
94      etpwmREG1->AQCTLA = ((uint16)((uint16)ActionQual_Set   << 0U)
95                         | (uint16)((uint16)ActionQual_Clear << 4U));
96
97      /** - Force EPWMxB output high when counter reaches zero and low when counter reaches Compare B value */
98      etpwmREG1->AQCTLB = ((uint16)((uint16)ActionQual_Set   << 0U)
99                         | (uint16)((uint16)ActionQual_Clear << 8U));
100
101     /** - Mode setting for Dead Band Module
102     *     -Select the input mode for Dead Band Module
103     *     -Select the output mode for Dead Band Module
104     *     -Select Polarity of the output PWMs
105     */
106     etpwmREG1->DBCTL = ((uint16)((uint16)0U << 5U)    /* Source for Falling edge delay(0-PWMA, 1-PWMB) */
107                       | (uint16)((uint16)0u << 4U)    /* Source for Rising edge delay(0-PWMA, 1-PWMB)  */
108                       | (uint16)((uint16)0U << 3U)    /* Enable/Disable EPWMxB invert       */
109                       | (uint16)((uint16)0U << 2U)    /* Enable/Disable EPWMxA invert       */
110                       | (uint16)((uint16)0U << 1U)    /* Enable/Disable Rising Edge Delay   */
111                       | (uint16)((uint16)0U << 0U));  /* Enable/Disable Falling Edge Delay  */
112
113     /** - Set the rising edge delay  */
114     etpwmREG1->DBRED = 110U;
115
116     /** - Set the falling edge delay  */
117     etpwmREG1->DBFED = 110U ;
```

:94	0, 4번 비트 1로 고정
:98	AQCTLB를 사용하지 않음
:106	0~5번 비트 0으로 고정
:114	DBRED
:117	DBFED



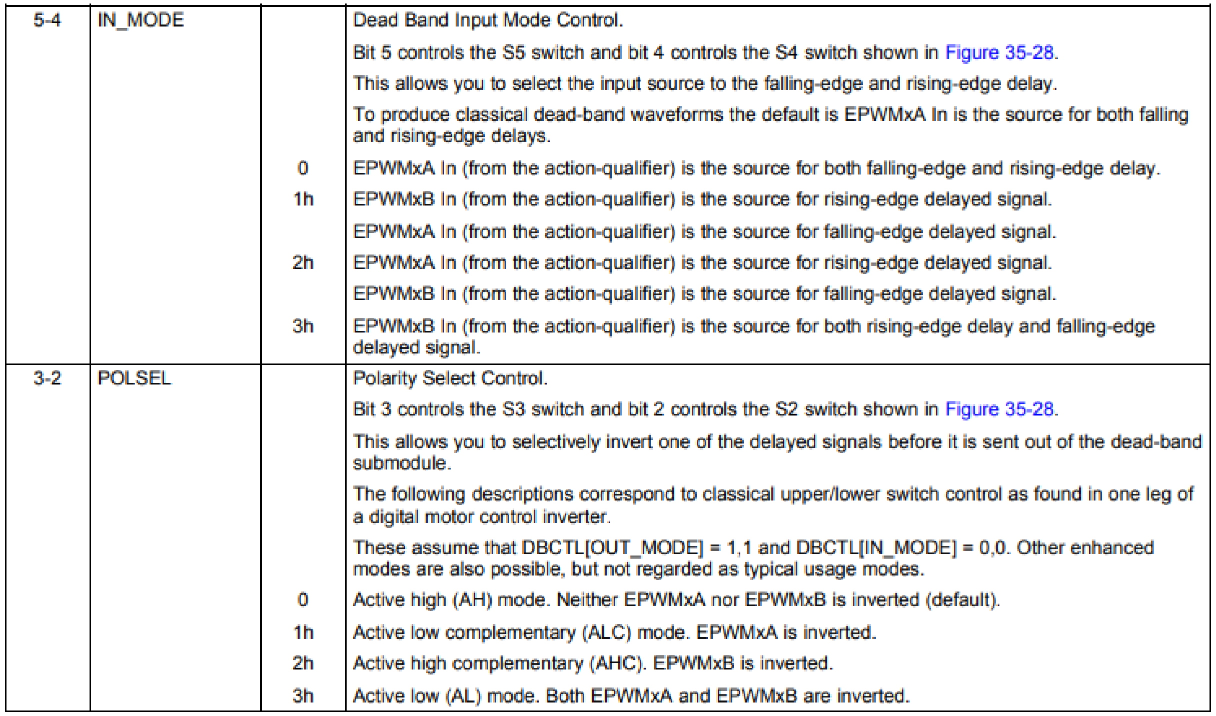
5-4	CAU		Action when the counter equals the active CMPA register and the counter is incrementing.
		0	Do nothing (action is disabled).
		1h	Clear: force EPWMxA output low.
		2h	Set: force EPWMxA output high.
		3h	Toggle EPWMxA output: low output signal will be forced high, and a high signal will be forced low.
3-2	PRD		Action when the counter equals the period.
			Note: By definition, in count up-down mode when the counter equals period the direction is defined as 0 or counting down.
		0	Do nothing (action is disabled).
		1h	Clear: force EPWMxA output low.
		2h	Set: force EPWMxA output high.
		3h	Toggle EPWMxA output: low output signal will be forced high, and a high signal will be forced low.
1-0	ZRO		Action when counter equals zero.
			Note: By definition, in count up-down mode when the counter equals 0 the direction is defined as 1 or counting up.
		0	Do nothing (action is disabled).
		1h	Clear: force EPWMxA output low.
		2h	Set: force EPWMxA output high.
		3h	Toggle EPWMxA output: low output signal will be forced high, and a high signal will be forced low.

AQCTLA

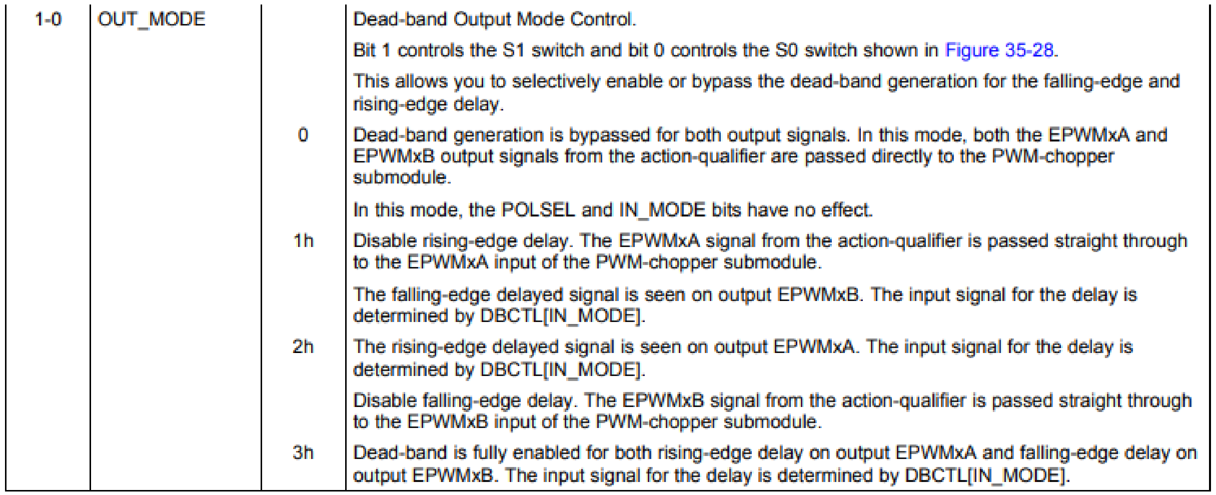
ZRO : 카운터가 0이될 때 ERPWMxA의 출력이 1이 된다.

CAU : CMPA레지스터와 카운터가 같은 값이 될 때 활성화됨. EPWMxA가 0으로 된다.

DBCTL



5-4	IN_MODE		Dead Band Input Mode Control.
			Bit 5 controls the S5 switch and bit 4 controls the S4 switch shown in Figure 35-28.
			This allows you to select the input source to the falling-edge and rising-edge delay.
			To produce classical dead-band waveforms the default is EPWMxA In is the source for both falling and rising-edge delays.
		0	EPWMxA In (from the action-qualifier) is the source for both falling-edge and rising-edge delay.
		1h	EPWMxB In (from the action-qualifier) is the source for rising-edge delayed signal.
			EPWMxA In (from the action-qualifier) is the source for falling-edge delayed signal.
		2h	EPWMxA In (from the action-qualifier) is the source for rising-edge delayed signal.
			EPWMxB In (from the action-qualifier) is the source for falling-edge delayed signal.
		3h	EPWMxB In (from the action-qualifier) is the source for both rising-edge delay and falling-edge delayed signal.
3-2	POLSEL		Polarity Select Control.
			Bit 3 controls the S3 switch and bit 2 controls the S2 switch shown in Figure 35-28.
			This allows you to selectively invert one of the delayed signals before it is sent out of the dead-band submodule.
			The following descriptions correspond to classical upper/lower switch control as found in one leg of a digital motor control inverter.
			These assume that DBCTL[OUT_MODE] = 1,1 and DBCTL[IN_MODE] = 0,0. Other enhanced modes are also possible, but not regarded as typical usage modes.
		0	Active high (AH) mode. Neither EPWMxA nor EPWMxB is inverted (default).
		1h	Active low complementary (ALC) mode. EPWMxA is inverted.
		2h	Active high complementary (AHC). EPWMxB is inverted.
		3h	Active low (AL) mode. Both EPWMxA and EPWMxB are inverted.



1-0	OUT_MODE		Dead-band Output Mode Control.
			Bit 1 controls the S1 switch and bit 0 controls the S0 switch shown in Figure 35-28.
			This allows you to selectively enable or bypass the dead-band generation for the falling-edge and rising-edge delay.
		0	Dead-band generation is bypassed for both output signals. In this mode, both the EPWMxA and EPWMxB output signals from the action-qualifier are passed directly to the PWM-chopper submodule.
			In this mode, the POLSEL and IN_MODE bits have no effect.
		1h	Disable rising-edge delay. The EPWMxA signal from the action-qualifier is passed straight through to the EPWMxA input of the PWM-chopper submodule.
			The falling-edge delayed signal is seen on output EPWMxB. The input signal for the delay is determined by DBCTL[IN_MODE].
		2h	The rising-edge delayed signal is seen on output EPWMxA. The input signal for the delay is determined by DBCTL[IN_MODE].
			Disable falling-edge delay. The EPWMxB signal from the action-qualifier is passed straight through to the EPWMxB input of the PWM-chopper submodule.
		3h	Dead-band is fully enabled for both rising-edge delay on output EPWMxA and falling-edge delay on output EPWMxB. The input signal for the delay is determined by DBCTL[IN_MODE].

IN_MODE : Dead Band 입력모드 조종

EPWMxA 내에 falling-edge와 rising-edge를 지연시킴.

POLSEL : 극성 선택 조정.

1일 때 동작하는 모드이다. EPWMxA나 EPWMxB 가 반전되지 않음.

OUT_MODE : Dead Band 출력모드 조종

D ead-band의 생성은 두개의 출력신호에 대해 바이패스 된다. 이 모드에서 EPWMxA와 EPWMxB의 두개의 출력신호는 PWM-chopper로 직접 전달된다.

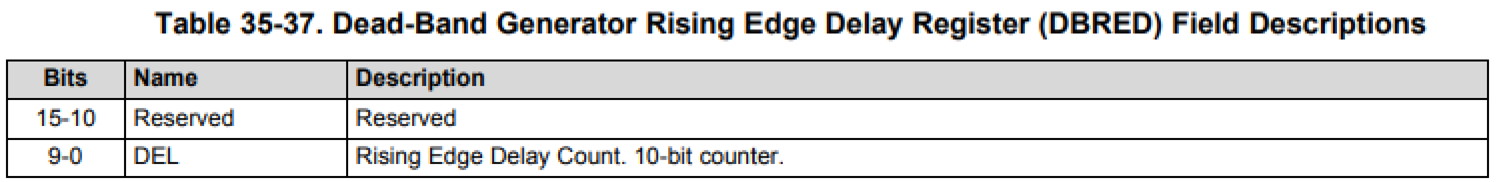


Table 35-37. Dead-Band Generator Rising Edge Delay Register (DBRED) Field Descriptions

Bits	Name	Description
15-10	Reserved	Reserved
9-0	DEL	Rising Edge Delay Count. 10-bit counter.

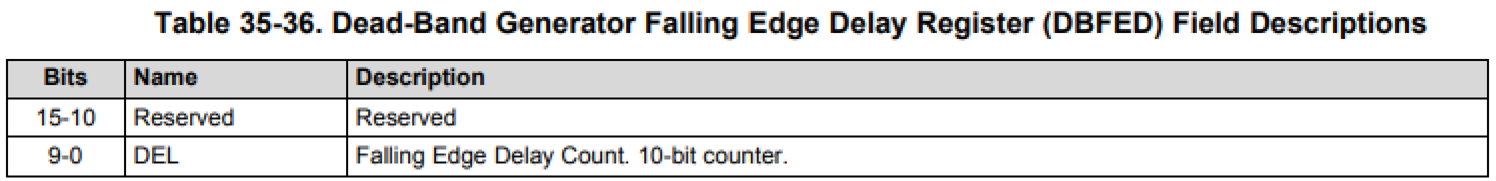


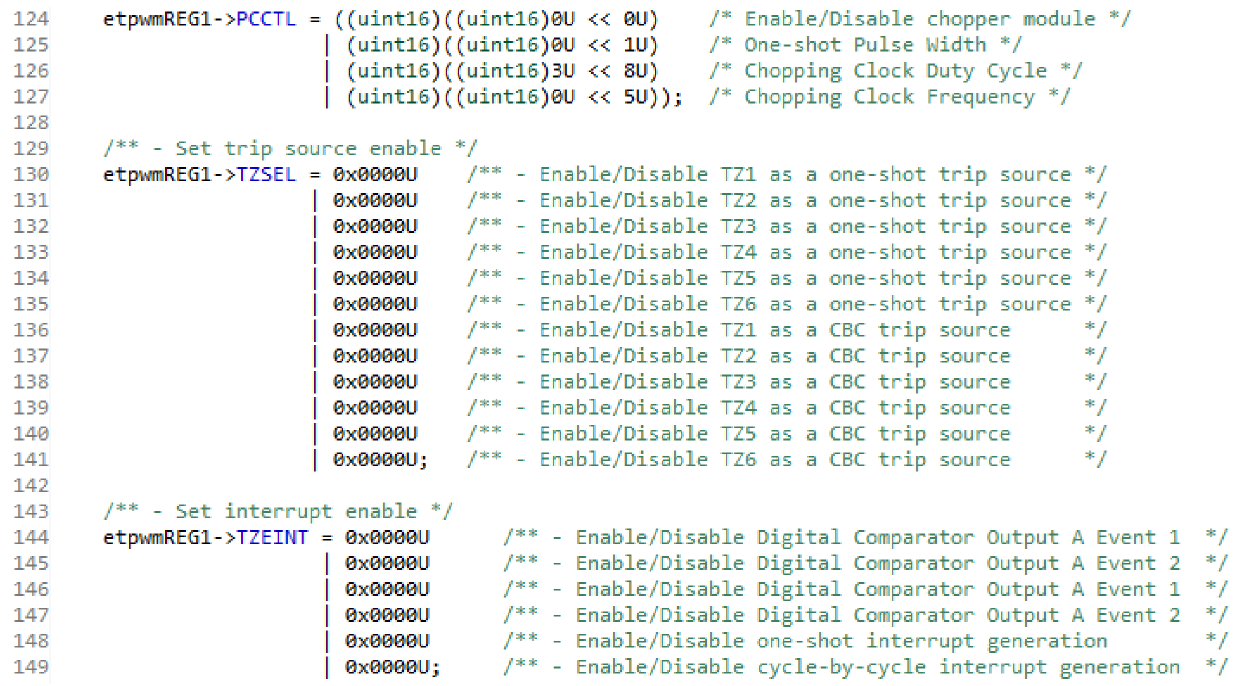
Table 35-36. Dead-Band Generator Falling Edge Delay Register (DBFED) Field Descriptions

Bits	Name	Description
15-10	Reserved	Reserved
9-0	DEL	Falling Edge Delay Count. 10-bit counter.

```
:114        etpwmREG2->DBRED = 110U;

:117        etpwmREG2->DBFED = 110U;
```

rising edge와 falling edge를 110만큼 지연시킨다.

실제로는 sin파 신호인데 딜레이를 주어 위의 둥근 부분의 신호를 깎음 그래서 펄스파처럼 됨.



```
124     etpwmREG1->PCCTL = ((uint16)((uint16)0U << 0U)    /* Enable/Disable chopper module */
125                       | (uint16)((uint16)0U << 1U)    /* One-shot Pulse Width */
126                       | (uint16)((uint16)3U << 8U)    /* Chopping Clock Duty Cycle */
127                       | (uint16)((uint16)0U << 5U));  /* Chopping Clock Frequency */
128 
129     /** - Set trip source enable */
130     etpwmREG1->TZSEL = 0x0000U    /** - Enable/Disable TZ1 as a one-shot trip source */
131                      | 0x0000U    /** - Enable/Disable TZ2 as a one-shot trip source */
132                      | 0x0000U    /** - Enable/Disable TZ3 as a one-shot trip source */
133                      | 0x0000U    /** - Enable/Disable TZ4 as a one-shot trip source */
134                      | 0x0000U    /** - Enable/Disable TZ5 as a one-shot trip source */
135                      | 0x0000U    /** - Enable/Disable TZ6 as a one-shot trip source */
136                      | 0x0000U    /** - Enable/Disable TZ1 as a CBC trip source      */
137                      | 0x0000U    /** - Enable/Disable TZ2 as a CBC trip source      */
138                      | 0x0000U    /** - Enable/Disable TZ3 as a CBC trip source      */
139                      | 0x0000U    /** - Enable/Disable TZ4 as a CBC trip source      */
140                      | 0x0000U    /** - Enable/Disable TZ5 as a CBC trip source      */
141                      | 0x0000U;   /** - Enable/Disable TZ6 as a CBC trip source      */
142 
143     /** - Set interrupt enable */
144     etpwmREG1->TZEINT = 0x0000U     /** - Enable/Disable Digital Comparator Output A Event 1  */
145                       | 0x0000U     /** - Enable/Disable Digital Comparator Output A Event 2  */
146                       | 0x0000U     /** - Enable/Disable Digital Comparator Output A Event 1  */
147                       | 0x0000U     /** - Enable/Disable Digital Comparator Output A Event 2  */
148                       | 0x0000U     /** - Enable/Disable one-shot interrupt generation        */
149                       | 0x0000U;    /** - Enable/Disable cycle-by-cycle interrupt generation  */
```

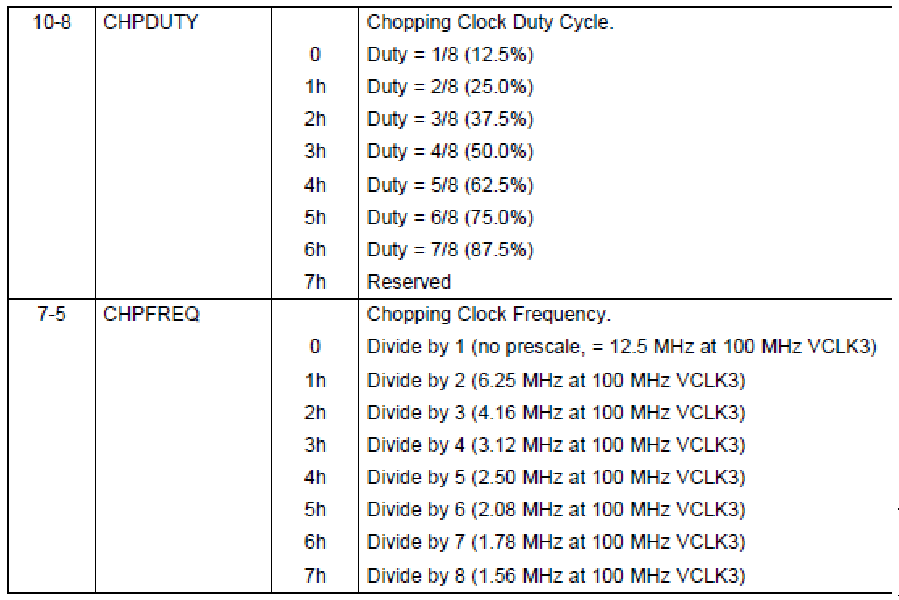
:124 8,9비트 1로 고정, 0,1,5비트 0으로 고정

CHPDUTY Duty = 50%

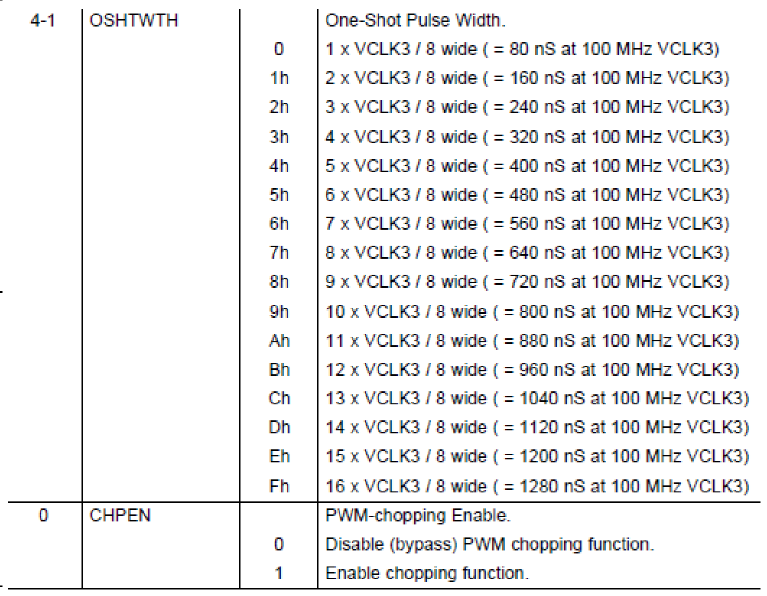
CHPFREQ 분주없음

OSHTWTH1 * VCLK / 8 wide

CHPEN PWM-chopping 비활성화



10-8	CHPDUTY		Chopping Clock Duty Cycle.
		0	Duty = 1/8 (12.5%)
		1h	Duty = 2/8 (25.0%)
		2h	Duty = 3/8 (37.5%)
		3h	Duty = 4/8 (50.0%)
		4h	Duty = 5/8 (62.5%)
		5h	Duty = 6/8 (75.0%)
		6h	Duty = 7/8 (87.5%)
		7h	Reserved
7-5	CHPFREQ		Chopping Clock Frequency.
		0	Divide by 1 (no prescale, = 12.5 MHz at 100 MHz VCLK3)
		1h	Divide by 2 (6.25 MHz at 100 MHz VCLK3)
		2h	Divide by 3 (4.16 MHz at 100 MHz VCLK3)
		3h	Divide by 4 (3.12 MHz at 100 MHz VCLK3)
		4h	Divide by 5 (2.50 MHz at 100 MHz VCLK3)
		5h	Divide by 6 (2.08 MHz at 100 MHz VCLK3)
		6h	Divide by 7 (1.78 MHz at 100 MHz VCLK3)
		7h	Divide by 8 (1.56 MHz at 100 MHz VCLK3)



4-1	OSHTWTH		One-Shot Pulse Width.
		0	1 x VCLK3 / 8 wide (= 80 nS at 100 MHz VCLK3)
		1h	2 x VCLK3 / 8 wide (= 160 nS at 100 MHz VCLK3)
		2h	3 x VCLK3 / 8 wide (= 240 nS at 100 MHz VCLK3)
		3h	4 x VCLK3 / 8 wide (= 320 nS at 100 MHz VCLK3)
		4h	5 x VCLK3 / 8 wide (= 400 nS at 100 MHz VCLK3)
		5h	6 x VCLK3 / 8 wide (= 480 nS at 100 MHz VCLK3)
		6h	7 x VCLK3 / 8 wide (= 560 nS at 100 MHz VCLK3)
		7h	8 x VCLK3 / 8 wide (= 640 nS at 100 MHz VCLK3)
		8h	9 x VCLK3 / 8 wide (= 720 nS at 100 MHz VCLK3)
		9h	10 x VCLK3 / 8 wide (= 800 nS at 100 MHz VCLK3)
		Ah	11 x VCLK3 / 8 wide (= 880 nS at 100 MHz VCLK3)
		Bh	12 x VCLK3 / 8 wide (= 960 nS at 100 MHz VCLK3)
		Ch	13 x VCLK3 / 8 wide (= 1040 nS at 100 MHz VCLK3)
		Dh	14 x VCLK3 / 8 wide (= 1120 nS at 100 MHz VCLK3)
		Eh	15 x VCLK3 / 8 wide (= 1200 nS at 100 MHz VCLK3)
		Fh	16 x VCLK3 / 8 wide (= 1280 nS at 100 MHz VCLK3)
0	CHPEN		PWM-chopping Enable.
		0	Disable (bypass) PWM chopping function.
		1	Enable chopping function.

:130

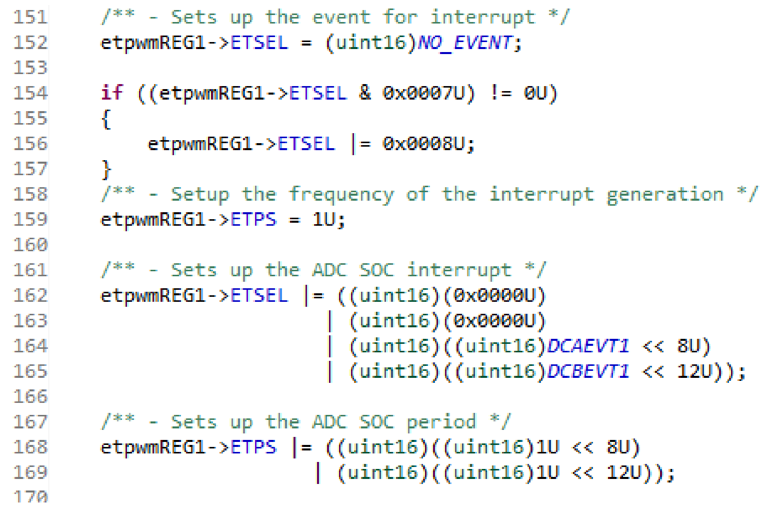
TZ1~6 one-shot trip source 비활성화

TZ1~6 CBC trip source 비활성화

:144

Digital Comparator Output A Event 1~4 인터럽트 비활성화

Trip-zone One-Shot 인터럽트 비활성화, Trip-zone Cycle-byCycle인터럽트 비활성화-



```
151     /** - Sets up the event for interrupt */
152     etpwmREG1->ETSEL = (uint16)NO_EVENT;
153
154     if ((etpwmREG1->ETSEL & 0x0007U) != 0U)
155     {
156         etpwmREG1->ETSEL |= 0x0008U;
157     }
158     /** - Setup the frequency of the interrupt generation */
159     etpwmREG1->ETPS = 1U;
160
161     /** - Sets up the ADC SOC interrupt */
162     etpwmREG1->ETSEL |= ((uint16)(0x0000U)
163                       | (uint16)(0x0000U)
164                       | (uint16)((uint16)DCAEVT1 << 8U)
165                       | (uint16)((uint16)DCBEVT1 << 12U));
166
167     /** - Sets up the ADC SOC period */
168     etpwmREG1->ETPS |= ((uint16)((uint16)1U << 8U)
169                      | (uint16)((uint16)1U << 12U));
170
```

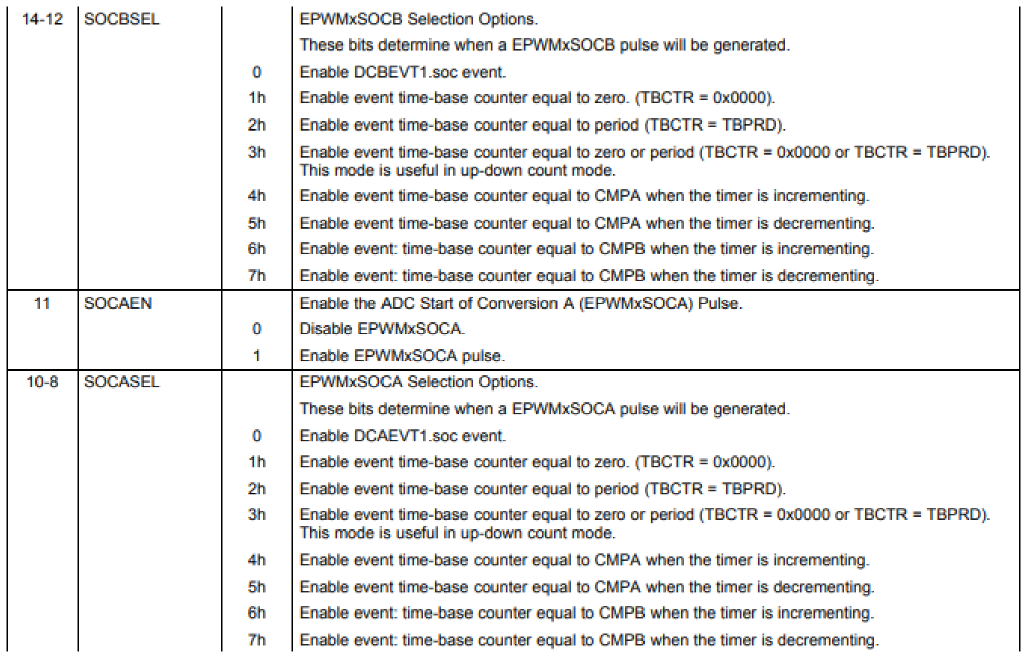
ETSEL이 0이므로 if문 만족하지 않음.

:159 1로 셋

:162 ETSEL 8,12번 비트 0으로 고정

DCAEVT1 : 0, DCBEVT2 : 0

:168 ETPS 8, 12번 1로 고정

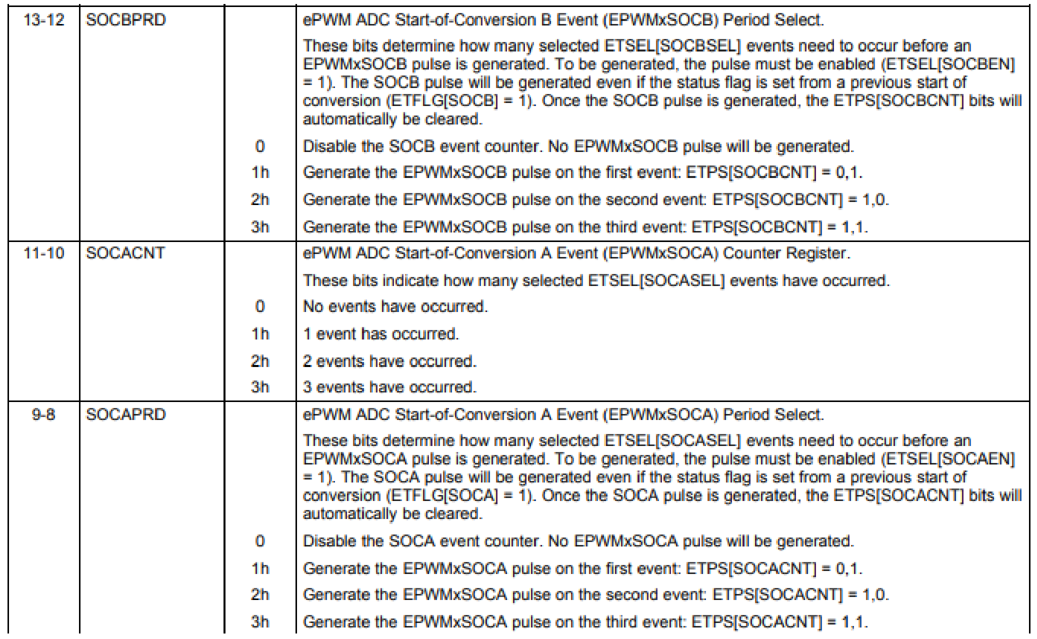


14-12	SOCBSEL		EPWMxSOCB Selection Options.
			These bits determine when a EPWMxSOCB pulse will be generated.
		0	Enable DCBEVT1.soc event.
		1h	Enable event time-base counter equal to zero. (TBCTR = 0x0000).
		2h	Enable event time-base counter equal to period (TBCTR = TBPRD).
		3h	Enable event time-base counter equal to zero or period (TBCTR = 0x0000 or TBCTR = TBPRD). This mode is useful in up-down count mode.
		4h	Enable event time-base counter equal to CMPA when the timer is incrementing.
		5h	Enable event time-base counter equal to CMPA when the timer is decrementing.
		6h	Enable event: time-base counter equal to CMPB when the timer is incrementing.
		7h	Enable event: time-base counter equal to CMPB when the timer is decrementing.
11	SOCAEN		Enable the ADC Start of Conversion A (EPWMxSOCA) Pulse.
		0	Disable EPWMxSOCA.
		1	Enable EPWMxSOCA pulse.
10-8	SOCASEL		EPWMxSOCA Selection Options.
			These bits determine when a EPWMxSOCA pulse will be generated.
		0	Enable DCAEVT1.soc event.
		1h	Enable event time-base counter equal to zero. (TBCTR = 0x0000).
		2h	Enable event time-base counter equal to period (TBCTR = TBPRD).
		3h	Enable event time-base counter equal to zero or period (TBCTR = 0x0000 or TBCTR = TBPRD). This mode is useful in up-down count mode.
		4h	Enable event time-base counter equal to CMPA when the timer is incrementing.
		5h	Enable event time-base counter equal to CMPA when the timer is decrementing.
		6h	Enable event: time-base counter equal to CMPB when the timer is incrementing.
		7h	Enable event: time-base counter equal to CMPB when the timer is decrementing.

ETSEL

DCAEV1.soc이벤트 활성화

DCBEVT1.soc이벤트 활성화



13-12	SOCBPRD		ePWM ADC Start-of-Conversion B Event (EPWMxSOCB) Period Select.
			These bits determine how many selected ETSEL[SOCBSEL] events need to occur before an EPWMxSOCB pulse is generated. To be generated, the pulse must be enabled (ETSEL[SOCBEN] = 1). The SOCB pulse will be generated even if the status flag is set from a previous start of conversion (ETFLG[SOCB] = 1). Once the SOCB pulse is generated, the ETPS[SOCBCNT] bits will automatically be cleared.
		0	Disable the SOCB event counter. No EPWMxSOCB pulse will be generated.
		1h	Generate the EPWMxSOCB pulse on the first event: ETPS[SOCBCNT] = 0,1.
		2h	Generate the EPWMxSOCB pulse on the second event: ETPS[SOCBCNT] = 1,0.
		3h	Generate the EPWMxSOCB pulse on the third event: ETPS[SOCBCNT] = 1,1.
11-10	SOCACNT		ePWM ADC Start-of-Conversion A Event (EPWMxSOCA) Counter Register.
			These bits indicate how many selected ETSEL[SOCASEL] events have occurred.
		0	No events have occurred.
		1h	1 event has occurred.
		2h	2 events have occurred.
		3h	3 events have occurred.
9-8	SOCAPRD		ePWM ADC Start-of-Conversion A Event (EPWMxSOCA) Period Select.
			These bits determine how many selected ETSEL[SOCASEL] events need to occur before an EPWMxSOCA pulse is generated. To be generated, the pulse must be enabled (ETSEL[SOCAEN] = 1). The SOCA pulse will be generated even if the status flag is set from a previous start of conversion (ETFLG[SOCA] = 1). Once the SOCA pulse is generated, the ETPS[SOCACNT] bits will automatically be cleared.
		0	Disable the SOCA event counter. No EPWMxSOCA pulse will be generated.
		1h	Generate the EPWMxSOCA pulse on the first event: ETPS[SOCACNT] = 0,1.
		2h	Generate the EPWMxSOCA pulse on the second event: ETPS[SOCACNT] = 1,0.
		3h	Generate the EPWMxSOCA pulse on the third event: ETPS[SOCACNT] = 1,1.

ETPS

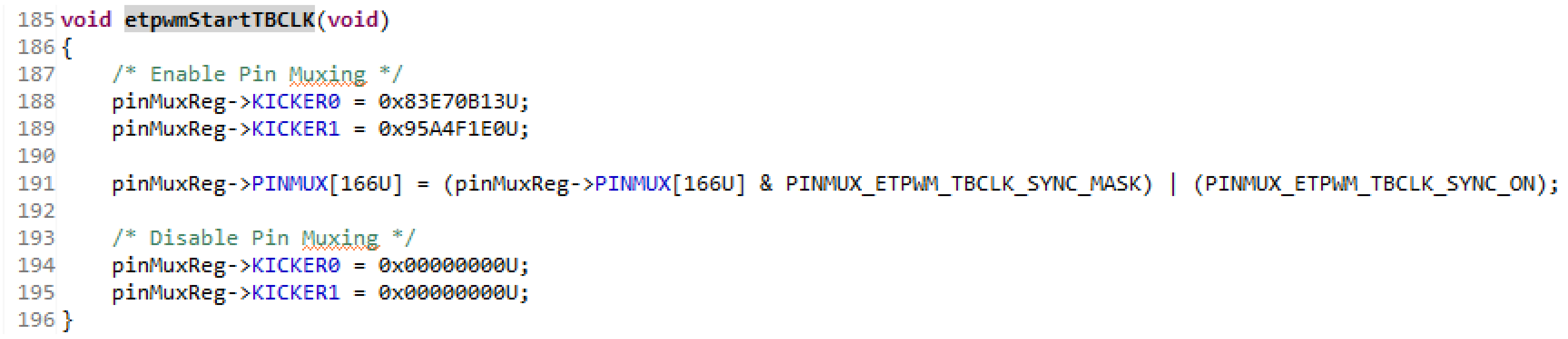
EPWMxSOCB펄스 생성.

EPWMxSOCB펄스의 첫번째 이벤트 ETPS[SOCBCNT] = 0.1

EPWMxSOCA펄스 생성.

EPWMxSOCA펄스의 첫번째 이벤트 ETPS[SOCACNT] = 0.1

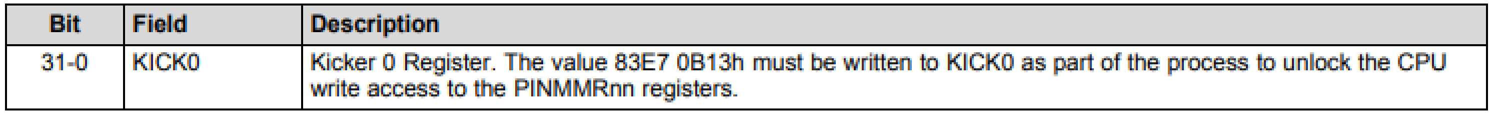
etpwmStartTBCLK();



```
void etpwmStartTBCLK(void)
{
    /* Enable Pin Muxing */
    pinMuxReg->KICKER0 = 0x83E70B13U;
    pinMuxReg->KICKER1 = 0x95A4F1E0U;

    pinMuxReg->PINMUX[166U] = (pinMuxReg->PINMUX[166U] & PINMUX_ETPWM_TBCLK_SYNC_MASK) | (PINMUX_ETPWM_TBCLK_SYNC_ON);

    /* Disable Pin Muxing */
    pinMuxReg->KICKER0 = 0x00000000U;
    pinMuxReg->KICKER1 = 0x00000000U;
}
```

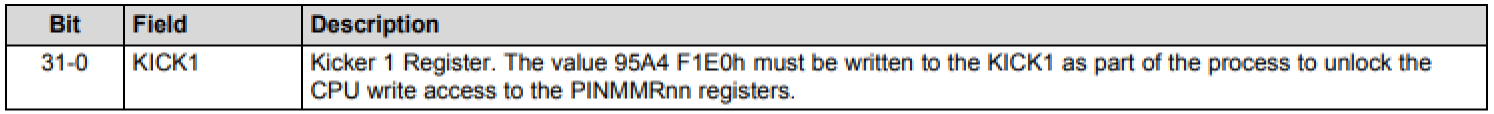
0xFFU | 0x2U

Table 6-15. Kicker Register 0 Field Descriptions



Bit	Field	Description
31-0	KICK0	Kicker 0 Register. The value 83E7 0B13h must be written to KICK0 as part of the process to unlock the CPU write access to the PINMMRnn registers.

Table 6-16. Kicker Register 1 Field Descriptions



Bit	Field	Description
31-0	KICK1	Kicker 1 Register. The value 95A4 F1E0h must be written to the KICK1 as part of the process to unlock the CPU write access to the PINMMRnn registers.