TI DSP, MCU, Xilinx Zynq FPGA 프로그래밍 전문가 과정

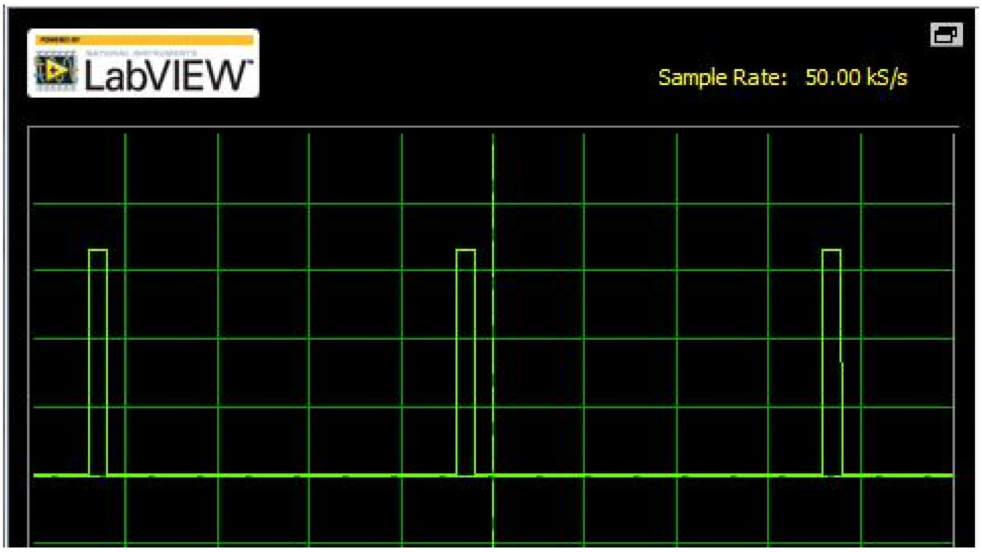
강사 – Innova Lee (이상훈)
gcccompil3r@gmail.com
학생 – 김형주
mihaelkel@naver.com

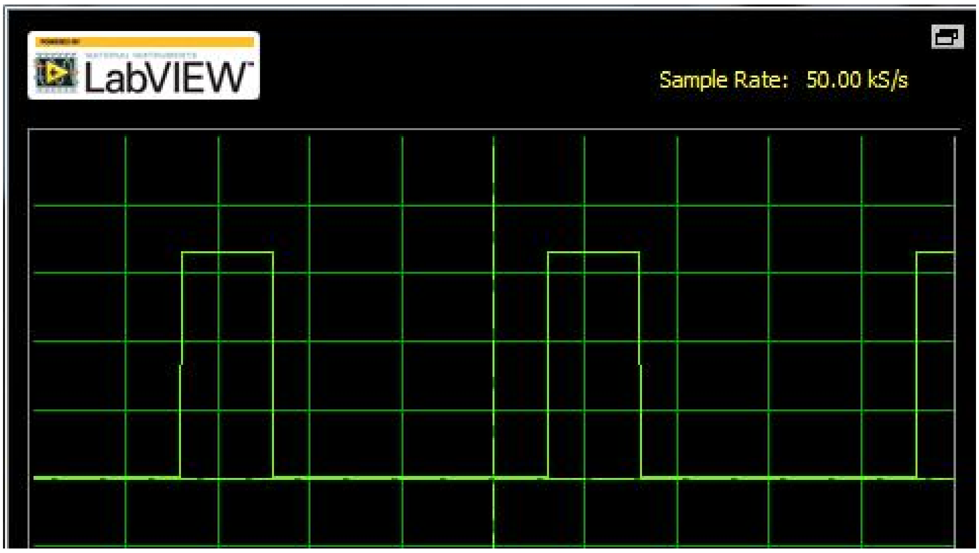
etPWM with Timer

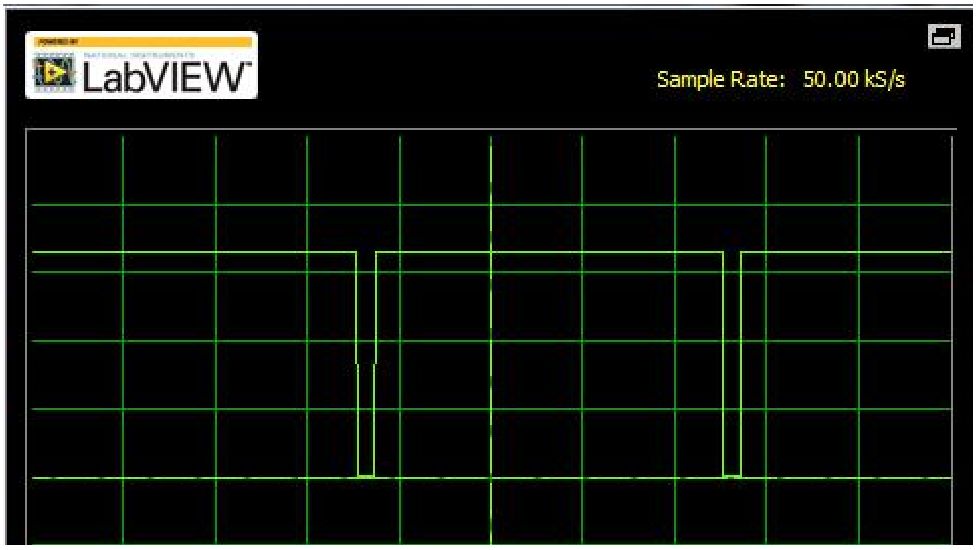
코드

```


#include "HL_sys_core.h"
#include "HL_mibspi.h"
#include "HL_esm.h"
#include "HL_rti.h"
#include "HL_sys_common.h"
#include "HL_system.h"
#include "HL_etpwm.h"
int idx = 0;
uint32 value = 0;
uint32 duty_arr[19] = {1000, 3000, 5000, 7000, 9000,
                      11000, 13000, 15000, 17000, 19000,
                      17000, 15000, 13000, 11000, 9000,
                      7000, 5000, 3000, 1000};
void pwmSet(void);
void delay(uint32);
int main(void)
{
    rtiInit();
    rtiEnableNotification(rtiREG1, rtiNOTIFICATION_COMPARE0);
    etpwmInit();
    etpwmStartTBCLK();
    delay(10000);
    _enable_IRQ_interrupt_();
    rtiStartCounter(rtiREG1, rtiCOUNTER_BLOCK0  );
    for(;;);
    return 0;
}
void pwmSet(void)
{
    value = duty_arr[idx % 19];
    etpwmSetCmpA(etpwmREG1, value);
    idx++;
}
void delay(uint32 delay)
{
    int i;
    for(i=0;i<delay;i++)
        ;
}
void rtiNotification(rtiBASE_t *rtiREG, uint32 notification){
    pwmSet();
}

```

결과


LabVIEW
Sample Rate: 50.00 kS/s

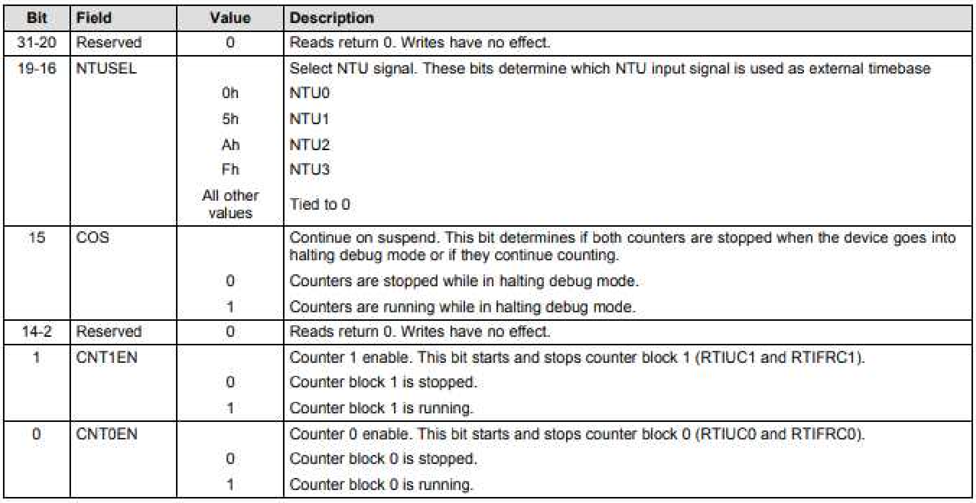

LabVIEW
Sample Rate: 50.00 kS/s


LabVIEW
Sample Rate: 50.00 kS/s

분석

```
rtiREG1->GCTRL = (uint32)((uint32)0x5U << 16U) | 0x00000000U;
```

Table 17-2. RTI Global Control Register (RTIGCTRL) Field Descriptions

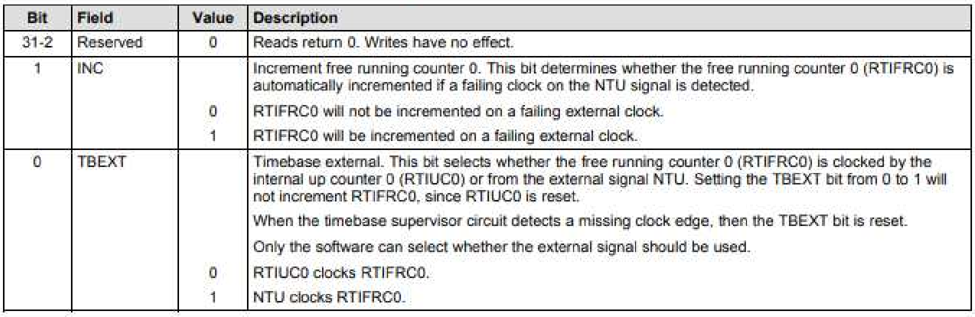


Bit	Field	Value	Description
31-20	Reserved	0	Reads return 0. Writes have no effect.
19-16	NTUSEL		Select NTU signal. These bits determine which NTU input signal is used as external timebase
		0h	NTU0
		5h	NTU1
		Ah	NTU2
		Fh	NTU3
		All other values	Tied to 0
15	COS		Continue on suspend. This bit determines if both counters are stopped when the device goes into halting debug mode or if they continue counting.
		0	Counters are stopped while in halting debug mode.
		1	Counters are running while in halting debug mode.
14-2	Reserved	0	Reads return 0. Writes have no effect.
1	CNT1EN		Counter 1 enable. This bit starts and stops counter block 1 (RTIUC1 and RTIFRC1).
		0	Counter block 1 is stopped.
		1	Counter block 1 is running.
0	CNT0EN		Counter 0 enable. This bit starts and stops counter block 0 (RTIUC0 and RTIFRC0).
		0	Counter block 0 is stopped.
		1	Counter block 0 is running.

19-16 bit : 5 h 이므로, NTU1을 timer의 input signal로 사용한다.

```
rtiREG1->TBCTRL = 0x00000000U;
```

Table 17-3. RTI Timebase Control Register (RTITBCTRL) Field Descriptions

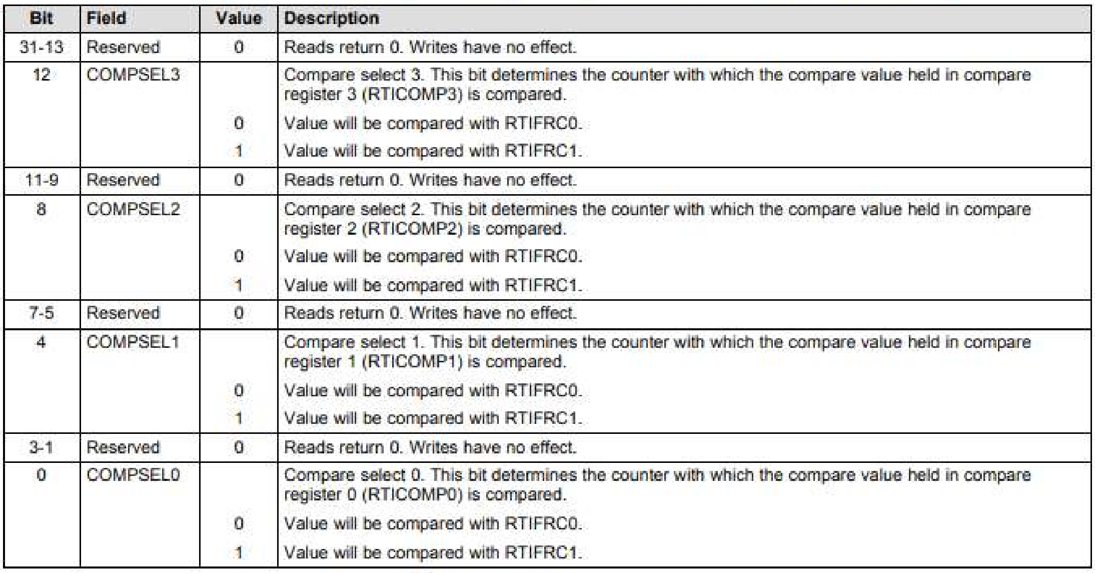


Bit	Field	Value	Description
31-2	Reserved	0	Reads return 0. Writes have no effect.
1	INC		Increment free running counter 0. This bit determines whether the free running counter 0 (RTIFRC0) is automatically incremented if a failing clock on the NTU signal is detected.
		0	RTIFRC0 will not be incremented on a failing external clock.
		1	RTIFRC0 will be incremented on a failing external clock.
0	TBEXT		Timebase external. This bit selects whether the free running counter 0 (RTIFRC0) is clocked by the internal up counter 0 (RTIUC0) or from the external signal NTU. Setting the TBEXT bit from 0 to 1 will not increment RTIFRC0, since RTIUC0 is reset. When the timebase supervisor circuit detects a missing clock edge, then the TBEXT bit is reset. Only the software can select whether the external signal should be used.
		0	RTIUC0 clocks RTIFRC0.
		1	NTU clocks RTIFRC0.

INC field와 TBEXT field를 0으로 clear한다.

```
rtiREG1->COMPCTRL = 0x00001000U | 0x00000100U | 0x00000000U | 0x00000000U;
```

Table 17-5. RTI Compare Control Register (RTICOMPCTRL) Field Descriptions

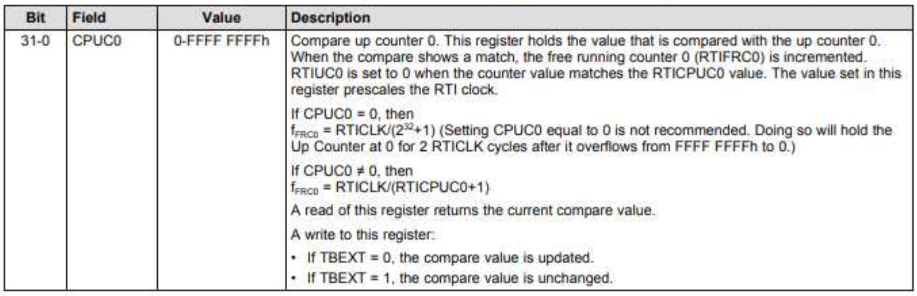


Bit	Field	Value	Description
31-13	Reserved	0	Reads return 0. Writes have no effect.
12	COMPSEL3		Compare select 3. This bit determines the counter with which the compare value held in compare register 3 (RTICOMP3) is compared.
		0	Value will be compared with RTIFRC0.
		1	Value will be compared with RTIFRC1.
11-9	Reserved	0	Reads return 0. Writes have no effect.
8	COMPSEL2		Compare select 2. This bit determines the counter with which the compare value held in compare register 2 (RTICOMP2) is compared.
		0	Value will be compared with RTIFRC0.
		1	Value will be compared with RTIFRC1.
7-5	Reserved	0	Reads return 0. Writes have no effect.
4	COMPSEL1		Compare select 1. This bit determines the counter with which the compare value held in compare register 1 (RTICOMP1) is compared.
		0	Value will be compared with RTIFRC0.
		1	Value will be compared with RTIFRC1.
3-1	Reserved	0	Reads return 0. Writes have no effect.
0	COMPSEL0		Compare select 0. This bit determines the counter with which the compare value held in compare register 0 (RTICOMP0) is compared.
		0	Value will be compared with RTIFRC0.
		1	Value will be compared with RTIFRC1.

COMP0와 COMP1은 RTIFRC0를, COMP2와 COMP3는 RTIFRC1을 선택한다.

rtiREG1->CNT[0U].CPUCx = 7U;

Table 17-8. RTI Compare Up Counter 0 Register (RTICPUC0) Field Descriptions

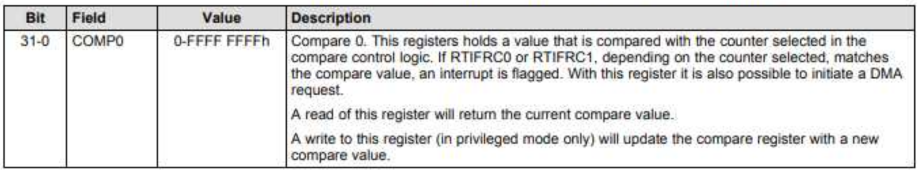


Bit	Field	Value	Description
31-0	CPUC0	0-FFFF FFFFh	Compare up counter 0. This register holds the value that is compared with the up counter 0. When the compare shows a match, the free running counter 0 (RTIFRC0) is incremented. RTIUC0 is set to 0 when the counter value matches the RTICPUC0 value. The value set in this register prescales the RTI clock. If CPUC0 = 0, then f_{FRC0} = RTICLK/(2^{32}+1) (Setting CPUC0 equal to 0 is not recommended. Doing so will hold the Up Counter at 0 for 2 RTICLK cycles after it overflows from FFFF FFFFh to 0.) If CPUC0 ≠ 0, then f_{FRC0} = RTICLK/(RTICPUC0+1) A read of this register returns the current compare value. A write to this register: • If TBEXT = 0, the compare value is updated. • If TBEXT = 1, the compare value is unchanged.

f_FRC0 = RTICLK / 8

rtiREG1->CMP[0U].COMPx = 9375000U;

Table 17-16. RTI Compare 0 Register (RTICOMP0) Field Descriptions

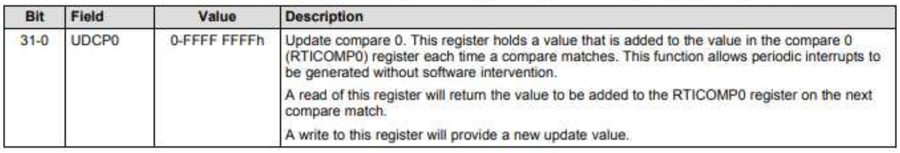


Bit	Field	Value	Description
31-0	COMP0	0-FFFF FFFFh	Compare 0. This registers holds a value that is compared with the counter selected in the compare control logic. If RTIFRC0 or RTIFRC1, depending on the counter selected, matches the compare value, an interrupt is flagged. With this register it is also possible to initiate a DMA request. A read of this register will return the current compare value. A write to this register (in privileged mode only) will update the compare register with a new compare value.

compare value를 9375000으로 갱신한다.

rtiREG1->CMP[0U].UDCPx = 9375000U;

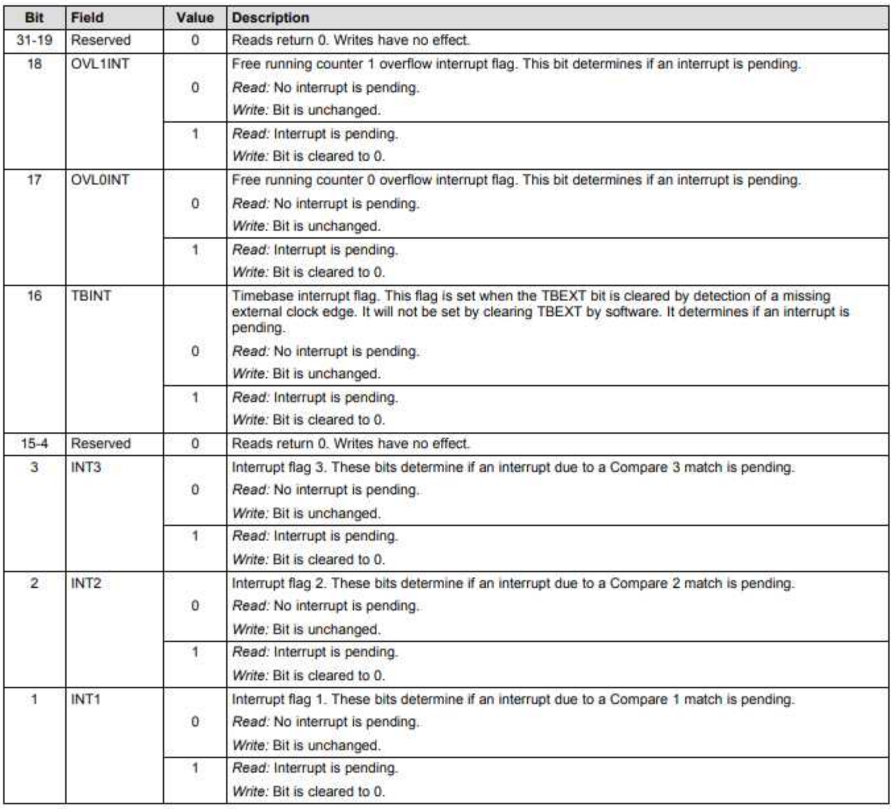
Table 17-17. RTI Update Compare 0 Register (RTIUDCP0) Field Descriptions



Bit	Field	Value	Description
31-0	UDCP0	0-FFFF FFFFh	Update compare 0. This register holds a value that is added to the value in the compare 0 (RTICOMP0) register each time a compare matches. This function allows periodic interrupts to be generated without software intervention. A read of this register will return the value to be added to the RTICOMP0 register on the next compare match. A write to this register will provide a new update value.

counter0의 초기값을 9375000으로 갱신한다.

rtiREG1->INTFLAG = 0x0007000FU;



Bit	Field	Value	Description
31-19	Reserved	0	Reads return 0. Writes have no effect.
18	OVL1INT		Free running counter 1 overflow interrupt flag. This bit determines if an interrupt is pending.
		0	*Read:* No interrupt is pending. *Write:* Bit is unchanged.
		1	*Read:* Interrupt is pending. *Write:* Bit is cleared to 0.
17	OVL0INT		Free running counter 0 overflow interrupt flag. This bit determines if an interrupt is pending.
		0	*Read:* No interrupt is pending. *Write:* Bit is unchanged.
		1	*Read:* Interrupt is pending. *Write:* Bit is cleared to 0.
16	TBINT		Timebase interrupt flag. This flag is set when the TBEXT bit is cleared by detection of a missing external clock edge. It will not be set by clearing TBEXT by software. It determines if an interrupt is pending.
		0	*Read:* No interrupt is pending. *Write:* Bit is unchanged.
		1	*Read:* Interrupt is pending. *Write:* Bit is cleared to 0.
15-4	Reserved	0	Reads return 0. Writes have no effect.
3	INT3		Interrupt flag 3. These bits determine if an interrupt due to a Compare 3 match is pending.
		0	*Read:* No interrupt is pending. *Write:* Bit is unchanged.
		1	*Read:* Interrupt is pending. *Write:* Bit is cleared to 0.
2	INT2		Interrupt flag 2. These bits determine if an interrupt due to a Compare 2 match is pending.
		0	*Read:* No interrupt is pending. *Write:* Bit is unchanged.
		1	*Read:* Interrupt is pending. *Write:* Bit is cleared to 0.
1	INT1		Interrupt flag 1. These bits determine if an interrupt due to a Compare 1 match is pending.
		0	*Read:* No interrupt is pending. *Write:* Bit is unchanged.
		1	*Read:* Interrupt is pending. *Write:* Bit is cleared to 0.

인터럽트 감지 플래그를 모두 초기화한다.

```
rtiREG1->CLEARINTENA = 0x00070F0FU;
```

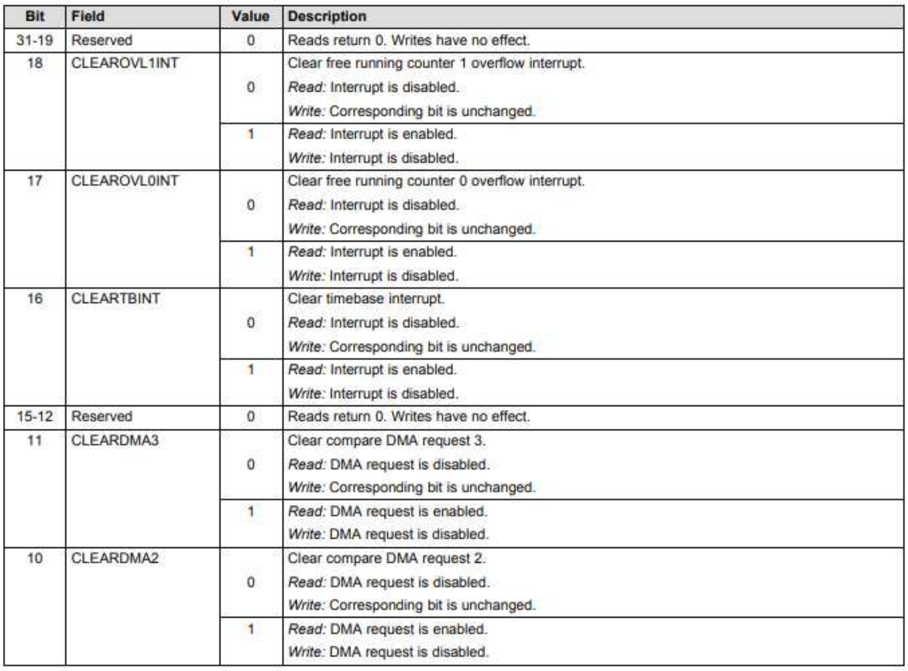


Table 17-27. RTI Clear Interrupt Control Register (RTICLEARINTENA) Field Descriptions

Bit	Field	Value	Description
31-19	Reserved	0	Reads return 0. Writes have no effect.
18	CLEAROVL1INT		Clear free running counter 1 overflow interrupt.
		0	*Read:* Interrupt is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read:* Interrupt is enabled. *Write:* Interrupt is disabled.
17	CLEAROVL0INT		Clear free running counter 0 overflow interrupt.
		0	*Read:* Interrupt is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read:* Interrupt is enabled. *Write:* Interrupt is disabled.
16	CLEARTBINT		Clear timebase interrupt.
		0	*Read:* Interrupt is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read:* Interrupt is enabled. *Write:* Interrupt is disabled.
15-12	Reserved	0	Reads return 0. Writes have no effect.
11	CLEARDMA3		Clear compare DMA request 3.
		0	*Read:* DMA request is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read:* DMA request is enabled. *Write:* DMA request is disabled.
10	CLEARDMA2		Clear compare DMA request 2.
		0	*Read:* DMA request is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read:* DMA request is enabled. *Write:* DMA request is disabled.

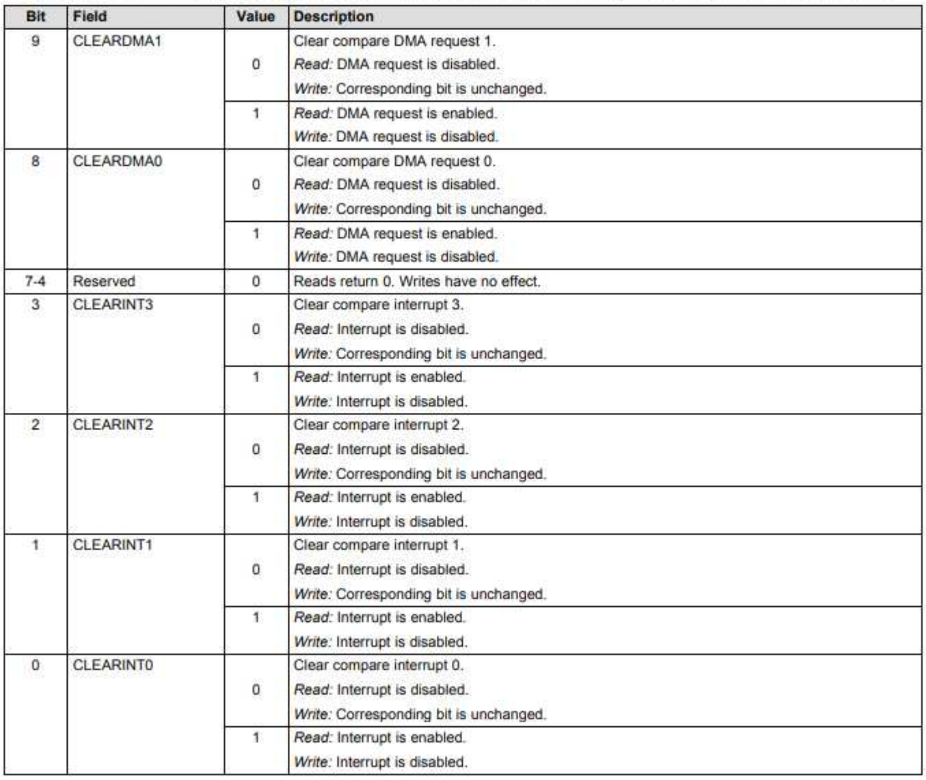


Table 17-27. RTI Clear Interrupt Control Register (RTICLEARINTENA) Field Descriptions (continued)

Bit	Field	Value	Description
9	CLEARDMA1		Clear compare DMA request 1.
		0	*Read:* DMA request is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read:* DMA request is enabled. *Write:* DMA request is disabled.
8	CLEARDMA0		Clear compare DMA request 0.
		0	*Read:* DMA request is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read:* DMA request is enabled. *Write:* DMA request is disabled.
7-4	Reserved	0	Reads return 0. Writes have no effect.
3	CLEARINT3		Clear compare interrupt 3.
		0	*Read:* Interrupt is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read:* Interrupt is enabled. *Write:* Interrupt is disabled.
2	CLEARINT2		Clear compare interrupt 2.
		0	*Read:* Interrupt is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read:* Interrupt is enabled. *Write:* Interrupt is disabled.
1	CLEARINT1		Clear compare interrupt 1.
		0	*Read:* Interrupt is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read:* Interrupt is enabled. *Write:* Interrupt is disabled.
0	CLEARINT0		Clear compare interrupt 0.
		0	*Read:* Interrupt is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read:* Interrupt is enabled. *Write:* Interrupt is disabled.

Interrupt와 DMA request를 모두 disable한다.

```
rtiREG->INTFLAG = notification
```

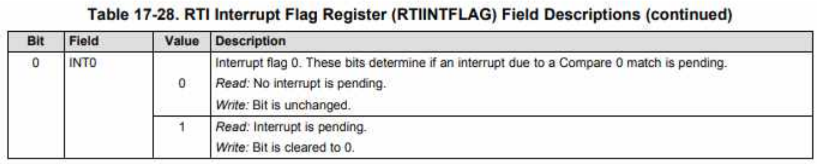
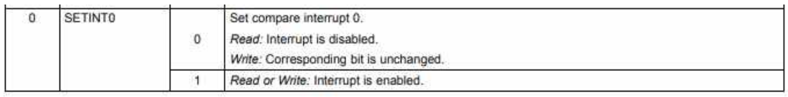


Table 17-28. RTI Interrupt Flag Register (RTIINTFLAG) Field Descriptions (continued)

Bit	Field	Value	Description
0	INT0		Interrupt flag 0. These bits determine if an interrupt due to a Compare 0 match is pending.
		0	*Read:* No interrupt is pending. *Write:* Bit is unchanged.
		1	*Read:* Interrupt is pending. *Write:* Bit is cleared to 0.

Compare 0 interrupt의 interrupt pending 여부 flag를 clear한다.(초기화 과정)

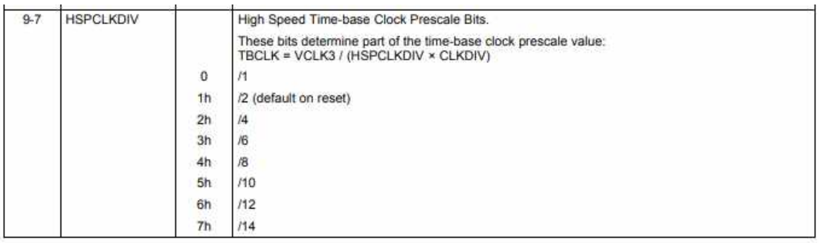
```
rtiREG->SETINTENA    = notification
```



0	SETINT0		Set compare interrupt 0.
		0	*Read:* Interrupt is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read or Write:* Interrupt is enabled.

Interrupt 0를 enable한다.

```
etpwmREG1->TBCTL = (uint16)5U << 7U;
```



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

TBCLK = VCLK3 / 10

```
etpwmREG1->TBPRD = 19999U;
```

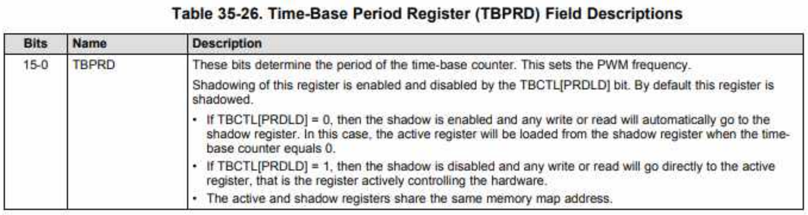


Table 35-26. Time-Base Period Register (TBPRD) Field Descriptions

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.

뭔소린지 잘 모르겠다..

counter의 주기를 19999으로 설정하는 듯 하다.(20ms)

```
etpwmREG1->CMPA = 2000U;
```

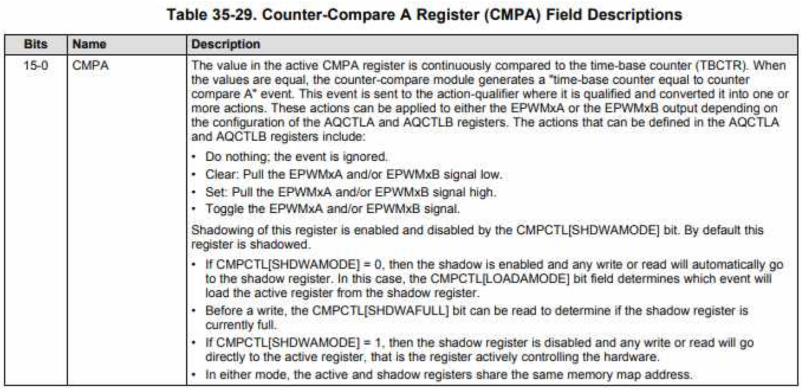
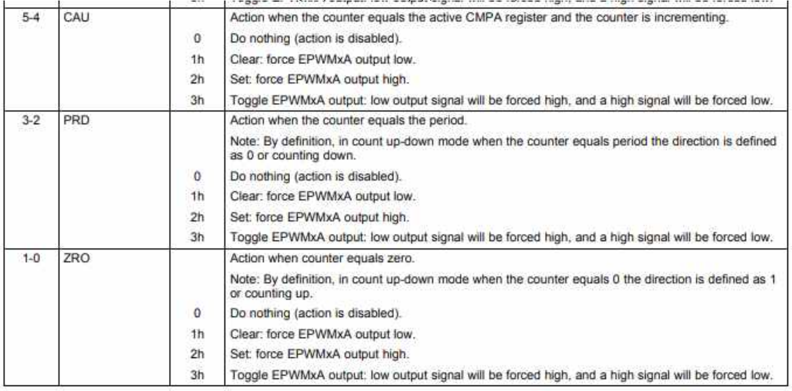


Table 35-29. Counter-Compare A Register (CMPA) Field Descriptions

Bits	Name	Description
15-0	CMPA	The value in the active CMPA register is continuously compared to the time-base counter (TBCTR). When the values are equal, the counter-compare module generates a "time-base counter equal to counter compare A" event. This event is sent to the action-qualifier where it is qualified and converted it into one or more actions. These actions can be applied to either the EPWMxA or the EPWMxB output depending on the configuration of the AQCTLA and AQCTLB registers. The actions that can be defined in the AQCTLA and AQCTLB registers include: • Do nothing; the event is ignored. • Clear: Pull the EPWMxA and/or EPWMxB signal low. • Set: Pull the EPWMxA and/or EPWMxB signal high. • Toggle the EPWMxA and/or EPWMxB signal. Shadowing of this register is enabled and disabled by the CMPCTL[SHDWAMODE] bit. By default this register is shadowed. • If CMPCTL[SHDWAMODE] = 0, then the shadow is enabled and any write or read will automatically go to the shadow register. In this case, the CMPCTL[LOADAMODE] bit field determines which event will load the active register from the shadow register. • Before a write, the CMPCTL[SHDWAFULL] bit can be read to determine if the shadow register is currently full. • If CMPCTL[SHDWAMODE] = 1, then the shadow register is disabled and any write or read will go directly to the active register, that is the register actively controlling the hardware. • In either mode, the active and shadow registers share the same memory map address.

이 또한 잘 모르겠지만, duty cycle의 주기를 2000으로 설정하는 듯 하다(2ms)

```
etpwmREG1->AQCTLA = ((uint16)((uint16)ActionQual_Set    << 0U)
                  | (uint16)((uint16)ActionQual_Clear << 4U));
```



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.

ZRO field를 2로, CAU field를 1로 set한다. 설명은 위와 같이.