임베디드시스템 기초(#514115)

#7. Timer B

한림대학교
전자공학과 이선우

한림대학교
HALLYM UNIVERSITY

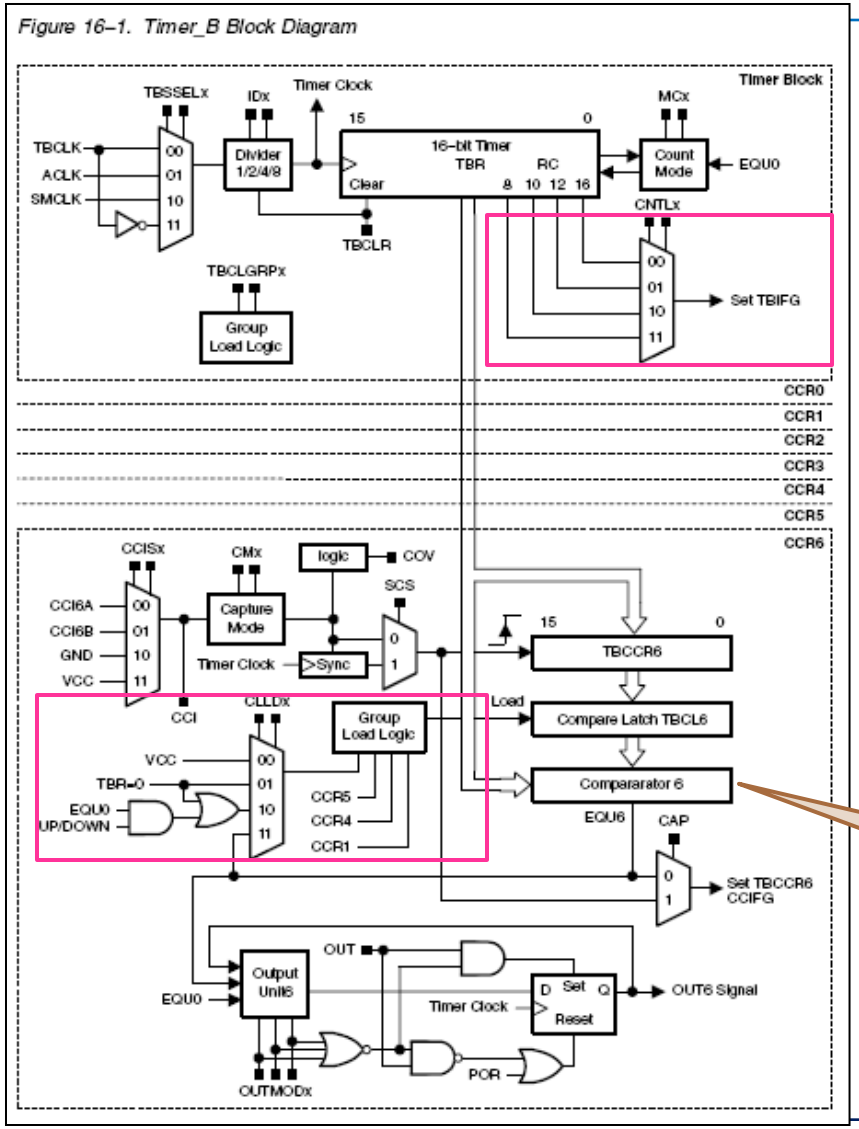
Contents

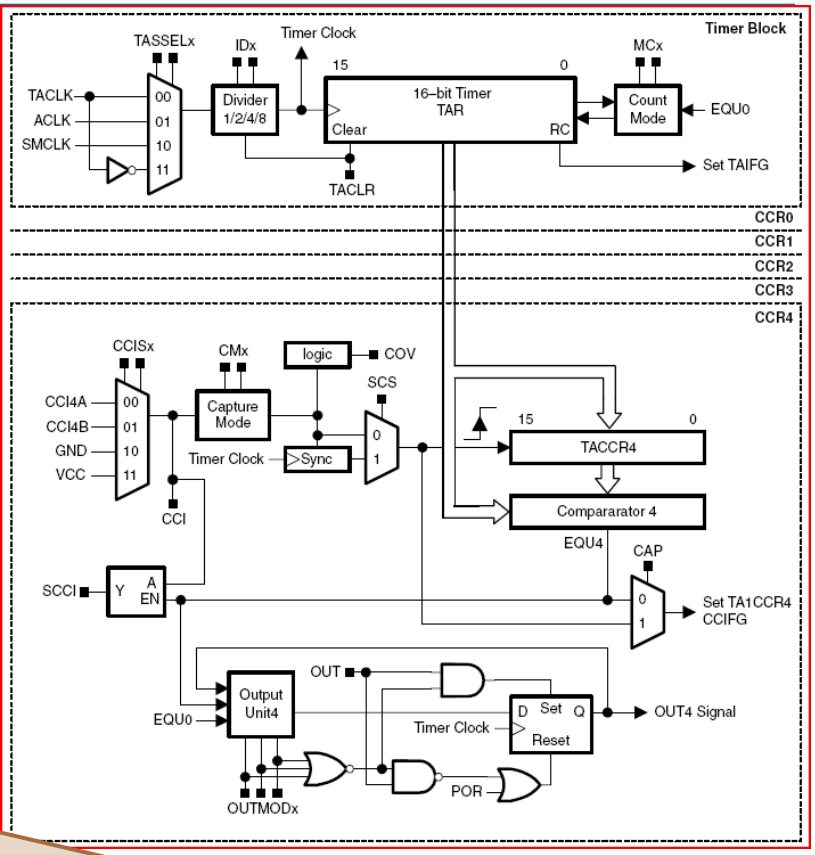
- Timer B5
  - Overview
  - Differences TB from TA
  - Up/continuous mode
  - Output unit
  - Capture operation

Timer_B7 overview

- Timer_B는 Timer_A와 아래 사항을 빼고 동일한 주변장치임
- Timer_A와 다른점
  - Timer_B의 길이(length)가 8/10/12/16 중 하나로 프로그램 통해 변경 가능
  - 모든 TBx 출력 핀들은 high-impedance state(0,1도 아닌 상태) 제공 가능
  - TBCCRx reg. 그룹화되어 있고 double-buffer를 가짐.
- FG461x series
  - Timer_B7: 7개의 CC block을 가짐(CCR0~CCR6)

Block diagram





Timer_A와 다른점: Compare Latch reg. (TBCLx)와 이 레지스터의 동작을 제어하는 회로가 존재 ➔ TBR/TBCLx 값이 비교됨 (TA의 경우 TACCRx와 비교)

Timer_B7 Length control

- Timer_B의 경우 SW를 통해 셀 수 있는 최대 범위를 조정할 수 있다.
  - CNTLx bits: TBCTL[12:11] bits
  - 00: 16-bit, TBR(max) = 0xFFFF
  - 01: 12-bit, TBR(max) = 0xFFF
  - 10: 10-bit, TBR(max) = 0x3FF
  - 11: 8-bit, TBR(max) = 0xFF
- Up & Continuous modes

Figure 16–2. Up Mode

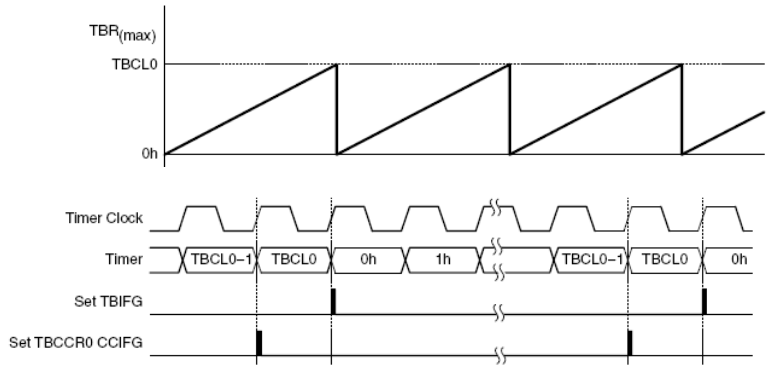
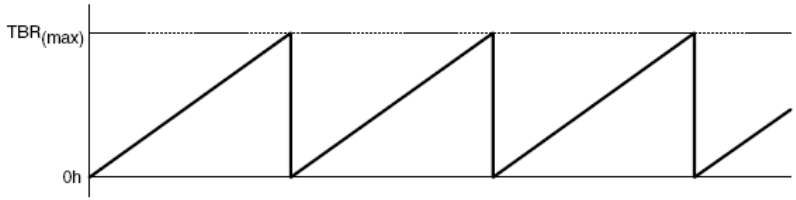
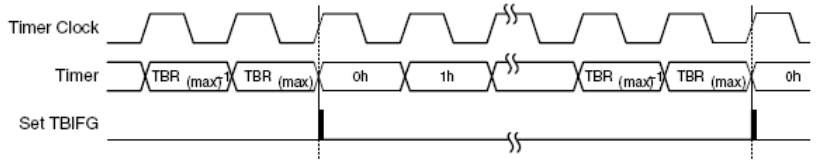


Figure 16–4. Continuous Mode

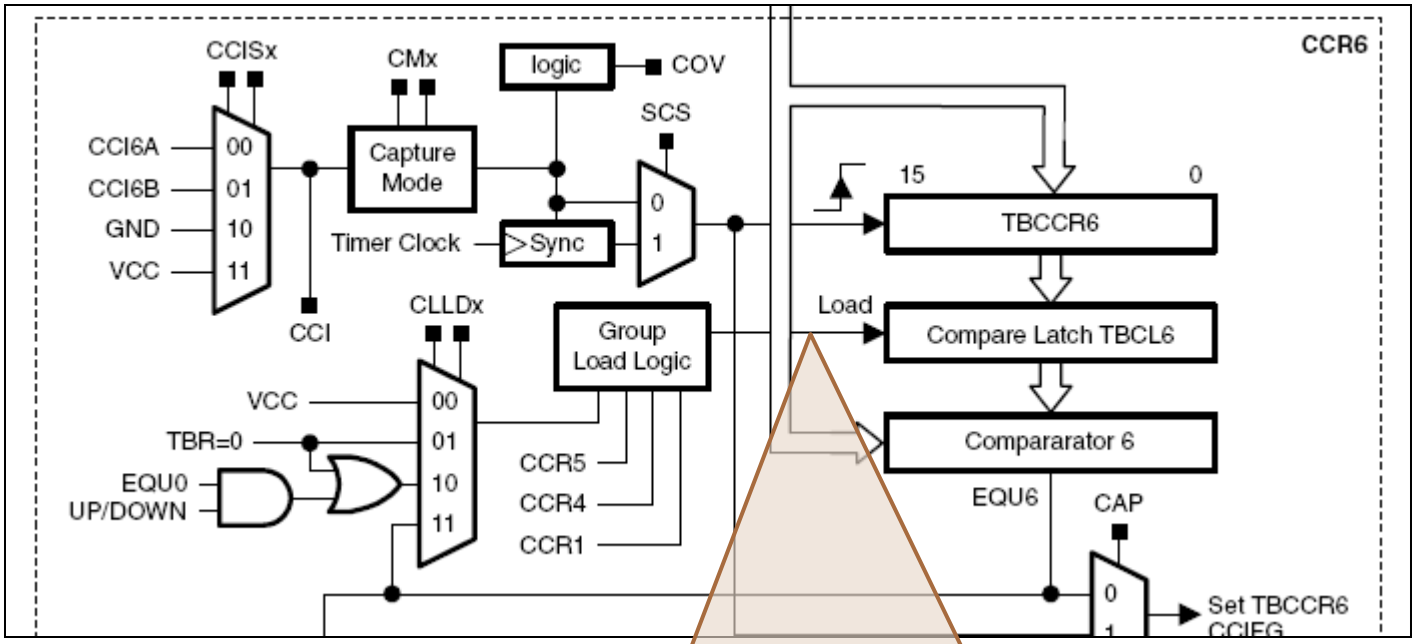


The TBIFG interrupt flag is set when the timer *counts* from $TBR_{(max)}$ to zero. Figure 16–5 shows the flag set cycle.

Figure 16–5. Continuous Mode Flag Setting



TBCLx와 TBCCRx의 관계



프로그램에서는 TBCLx reg.에 access할 수 없다.
따라서 비교하고자 하는 값을 변경하려면 TBCCRx의 값을
바꾸고 Load 신호를 줘서 이 값을 TBCLx로 load시켜야 한다.
Load 방법: CLLDx=TBCCTLx[10:0] bits 이용 설정
-00:TBCCRx에 데이터를 쓸 때 바로
-01: TBR=0 될 때
-10: TBR=TBCL0 or 0 될 때
-11: TBR=TBCLx가 될 때

MSP430FG461x series의 Timer_B7

Timer_B7의 Input/Output Pins

Timer_B7 Signal Connections					
Input Pin Number PZ/ZQW	Device Input Signal	Module Input Name	Module Block	Module Output Signal	Output Pin Number PZ/ZQW
83/B8 - P1.4	TBCLK	TBCLK	Timer	NA	
	ACLK	ACLK			
	SMCLK	SMCLK			
83/B8 - P1.4	$\overline{\text{TBCLK}}$	INCLK			
78/D8 - P2.1	TB0	CCI0A	CCR0	TB0	78/D8 - P2.1
78/D8 - P2.1	TB0	CCI0B			ADC12 (internal)
	DV_{SS}	GND			
	DV_{CC}	V_{CC}			
77/E8 - P2.2	TB1	CCI1A	CCR1	TB1	77/E8 - P2.2
77/E8 - P2.2	TB1	CCI1B			ADC12 (internal)
	DV_{SS}	GND			
	DV_{CC}	V_{CC}			
76/A11 - P2.3	TB2	CCI2A	CCR2	TB2	76/A11 - P2.3
76/A11 - P2.3	TB2	CCI2B			
	DV_{SS}	GND			
	DV_{CC}	V_{CC}			
67/E12 - P3.4	TB3	CCI3A	CCR3	TB3	67/E12 - P3.4
67/E12 - P3.4	TB3	CCI3B			
	DV_{SS}	GND			
	DV_{CC}	V_{CC}			
66/G9 - P3.5	TB4	CCI4A	CCR4	TB4	66/G9 - P3.5
66/G9 - P3.5	TB4	CCI4B			
	DV_{SS}	GND			
	DV_{CC}	V_{CC}			
65/F11 - P3.6	TB5	CCI5A	CCR5	TB5	65/F11 - P3.6
65/F11 - P3.6	TB5	CCI5B			
	DV_{SS}	GND			
	DV_{CC}	V_{CC}			
64/F12 - P3.7	TB6	CCI6A	CCR6	TB6	64/F12 - P3.7
	ACLK (internal)	CCI6B			
	DV_{SS}	GND			
	DV_{CC}	V_{CC}			

TBIV의 Interrupt Vector 값

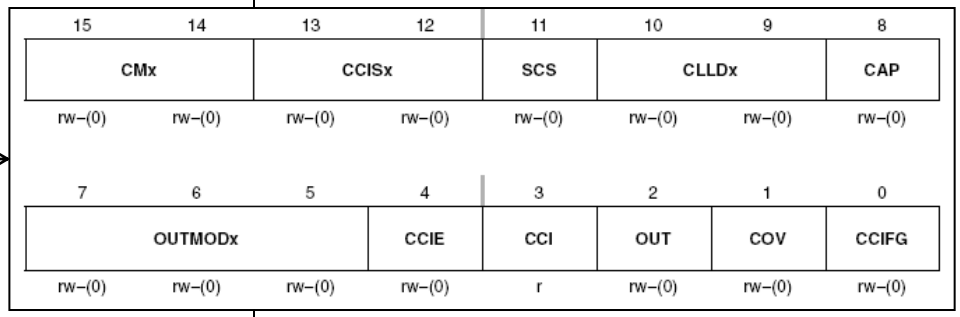
TBIV Contents	Interrupt Source	Interrupt Flag	Interrupt Priority
00h	No interrupt pending	–	
02h	Capture/compare 1	TBCCR1 CCIFG	Highest
04h	Capture/compare 2	TBCCR2 CCIFG	
06h	Capture/compare 3†	TBCCR3 CCIFG	
08h	Capture/compare 4†	TBCCR4 CCIFG	
0Ah	Capture/compare 5†	TBCCR5 CCIFG	
0Ch	Capture/compare 6†	TBCCR6 CCIFG	
0Eh	Timer overflow	TBIFG	Lowest

†MSP430x4xx devices only

예제 코드:Up mode 이용 클럭신호 발생

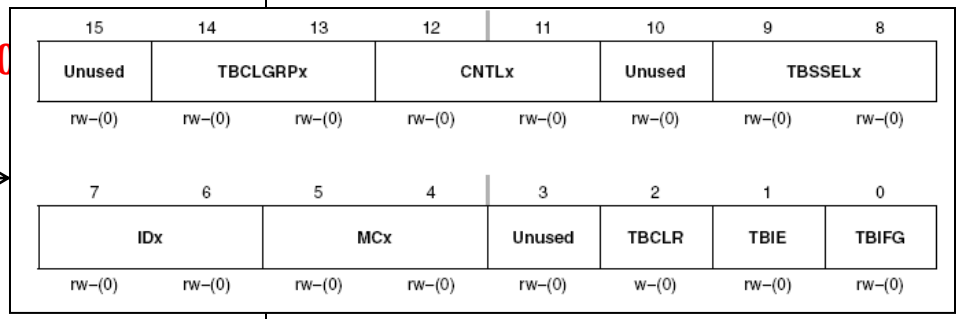
요구사항: 10msec 의 인터벌마다 P1.0 출력 toggle. Timer_B, up mode, ISR 사용

```
#pragma vector=TIMERB0_VECTOR
__interrupt void tb7_isr(void)
{
  P1OUT ^= 0x01;  //toggle P1.0
}

void main(void)
{
  P1DIR |= 0x01;

  //setup Timer_B
  TBCCTL0 = 0x0010;  // 0000 0000 0001 0000
  TBCCR0 = 10485; // 10msec
  TBCTL = 0x0210  ; //0000 0010 0001 000
  __enable_interrupt();
}
```



15	14	13	12	11	10	9	8
CMx		CCISx		SCS	CLLDx		CAP
rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)

7	6	5	4	3	2	1	0
OUTMODx			CCIE	CCI	OUT	COV	CCIFG
rw-(0)	rw-(0)	rw-(0)	rw-(0)	r	rw-(0)	rw-(0)	rw-(0)



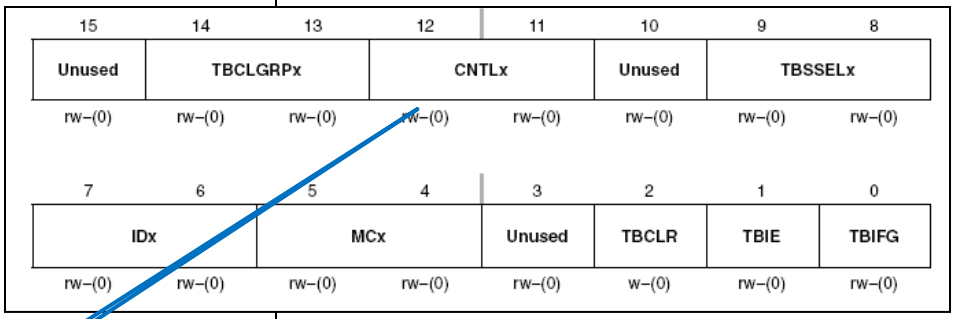
15	14	13	12	11	10	9	8
Unused	TBCLGRPx		CNTLx		Unused	TBSSELx	
rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)

7	6	5	4	3	2	1	0
IDx		MCx		Unused	TBCLR	TBIE	TBIFG
rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	w-(0)	rw-(0)	rw-(0)

예제 코드: Continuous mode 이용

요구사항: 1024Hz 의 인터벌마다 P1.0 출력 toggle. Timer_B, continuous mode,

```
#pragma vector=TIMERB1_VECTOR
__interrupt void tb7_isr(void)
{
  if(TBIV == 0x0e)
       P1OUT ^= 0x01;  //toggle P1.0
}

void main(void)
{
  P1DIR |= 0x01;

  //setup Timer_B
  // contin. mode, SMCLK, CNTLx=10bit
 TBCTL = 0x1222  ; //0001 0010 0010 0010
  __enable_interrupt();
}
```



복수 인터벌 만들기 (연속모드 이용법)

```
#include <msp430xG46x.h>
#define INTVAL1  1000
#define INTVAL2  3000
#define INTVAL3  5000

 #pragma vector=TIMERB1_VECTOR
__interrupt void ccr_handler(void)
{
switch (TAIV)
  {
     case 8:  //use TBCCR4
       P1OUT ^= 0x01;  //toggle P1.0
      TBCCR4 += INTVAL1;
       break;

     case 10: //use TBCCR5
       P1OUT ^= 0x02;  //toggle P1.1
      TBCCR5 += INTVAL2;
       break;

    case 12: //use TBCCR6
       P1OUT ^= 0x04;  //toggle P1.2
      TBCCR6 += INTVAL3;
  }
}
```

요구사항: 3개 CC block이용하여 3개 다른 주기의 클럭 펄스 생생

```
void main(void)
{

  P1DIR |= 0x07;

  //setup Timer_B
  TBCCTL4 = 0x0010;
  TBCCR4 = INTVAL1;
  TBCCTL5 = 0x0010;
  TBCCR5= INTVAL2;
  TBCCTL6= 0x0010;
  TBCCR6= INTVAL3;
  TBCTL = 0x0120  ; //SSEL=ACLK, MCx=continuous
  __enable_interrupt();
}
```

Timer_B7 input/output pin spec.

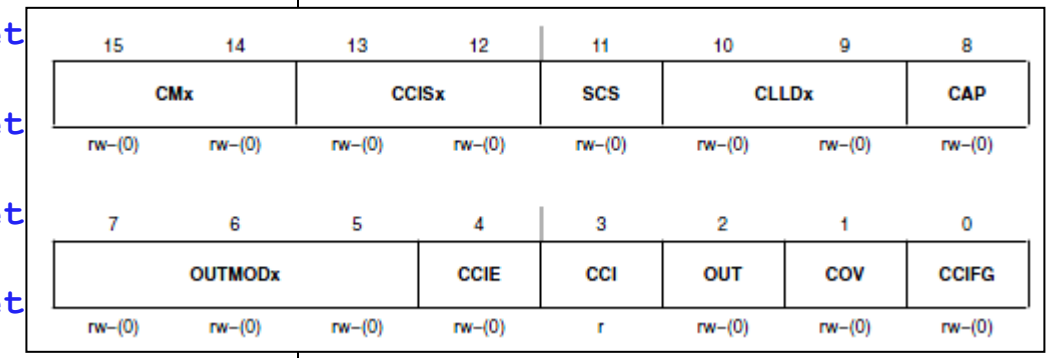
Timer_B7 Signal Connections					
Input Pin Number PZ/ZQW	Device Input Signal	Module Input Name	Module Block	Module Output Signal	Output Pin Number PZ/ZQW
83/B8 - P1.4	TBCLK	TBCLK	Timer	NA	
	ACLK	ACLK			
	SMCLK	SMCLK			
83/B8 - P1.4	TBCLK	INCLK			
78/D8 - P2.1	TB0	CCI0A	CCR0	TB0	78/D8 - P2.1
78/D8 - P2.1	TB0	CCI0B			ADC12 (internal)
	DV_{SS}	GND			
	DV_{CC}	V_{CC}			
77/E8 - P2.2	TB1	CCI1A	CCR1	TB1	77/E8 - P2.2
77/E8 - P2.2	TB1	CCI1B			ADC12 (internal)
	DV_{SS}	GND			
	DV_{CC}	V_{CC}			
76/A11 - P2.3	TB2	CCI2A	CCR2	TB2	76/A11 - P2.3
76/A11 - P2.3	TB2	CCI2B			
	DV_{SS}	GND			
	DV_{CC}	V_{CC}			
67/E12 - P3.4	TB3	CCI3A	CCR3	TB3	67/E12 - P3.4
67/E12 - P3.4	TB3	CCI3B			
	DV_{SS}	GND			
	DV_{CC}	V_{CC}			
66/G9 - P3.5	TB4	CCI4A	CCR4	TB4	66/G9 - P3.5
66/G9 - P3.5	TB4	CCI4B			
	DV_{SS}	GND			
	DV_{CC}	V_{CC}			
65/F11 - P3.6	TB5	CCI5A	CCR5	TB5	65/F11 - P3.6
65/F11 - P3.6	TB5	CCI5B			
	DV_{SS}	GND			
	DV_{CC}	V_{CC}			
64/F12 - P3.7	TB6	CCI6A	CCR6	TB6	64/F12 - P3.7
	ACLK (internal)	CCI6B			
	DV_{SS}	GND			
	DV_{CC}	V_{CC}			

P2.1, P2.2, P2.3
P3.4, P3.4, P3.6, P3.7

Example code #1: PWM 신호 발생

```
void main(void)
{
    //port setting
    P2SEL |= 0x0C; //0000 1110
    P2DIR |= 0x0C;
    P3SEL |= 0xF0; //1111 0000
    P3DIR |= 0xF0;
    //setup Timer_B
    TBCCR0 = 512-1;
    TBCCTL1 = 0x00e0; //reset/set mode
    TBCCR1 = 383;
    TBCCTL2 = 0x00e0; //reset/set mode
    TBCCR2 = 128;
    TBCCTL3 = 0x00e0; //reset/set
    TBCCR3 = 64;
    TBCCTL4 = 0x00e0; //reset/set
    TBCCR4 = 32;
    TBCCTL5 = 0x00e0; //reset/set
    TBCCR5 = 16;
    TBCCTL6 = 0x00e0; //reset/set
    TBCCR6 = 8;
    TBCTL = 0x0210; //SMCLK, up mode
}
```

• 요구사항: 2KHz 주파수를 가지며 TB1~TB6까지 다른 듀티 비의 신호 발생.
TB1=75%, TB2=25%, TB3=12.5%, TB4=6.26%, TB5=3.13%, TB6=1.566%



15	14	13	12	11	10	9	8
CMx		CCISx		SCS	CLLDx		CAP
rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)

7	6	5	4	3	2	1	0
OUTMODx			CCIE	CCI	OUT	COV	CCIFG
rw-(0)	rw-(0)	rw-(0)	rw-(0)	r	rw-(0)	rw-(0)	rw-(0)

Timer_B Capture mode

- Capture mode의 동작은 Timer_A와 동일.
- 즉, 외부 입력 신호(CCIxA/B)에 입력되는 클럭 신호에 의해 증가하는 TBR 값을 CCRx reg.에 저장함.
- FG461x MCU 의 경우 Timer_A에 비해 모두 7개의 CC block이 있으므로 최대 7개까지 값을 저장할 수 있다.
- 따라서 동시에 On-timer, Off-time, period 등을 측정할 수 있다.

Example code: 외부 신호 ON-time 측정

```
int start_t, end_t, status;
void main(void)
{
  int result;
  //port setting
  P2SEL |= 0x08; //enable CCI2A
  P2DIR &= ~0x08;
  //setup Timer_B
  //capture on rising, CCIxA,
Sync., CAP mode, irq. enable
  TBCCTL2 = 0x4910;
  TBCCR2 = 0;
  TBCTL = 0x0220; //SMCLK,
cont.mode
  __enable_interrupt();
  status = 0;
 while(1) {
  if(status==2) {
    status = 0;
    result = end_t - start_t;
    display_ontime();
    }
  }
}

#pragma vector = TIMERB1_VECTOR
__interrupt void tb_cap_isr(void)
{
  if(TBIV==4) {
    if(status==0) {
      start_t = TBCCR2;
      TBCCTL2 = 0x8910; //change
to falling edge
      status =1;
    }
    else if(status ==1) {
      end_t = TBCCR2;
      TBCCTL2 = 0x4910; //change
to rising edge
      status = 2;
    }
}
```