

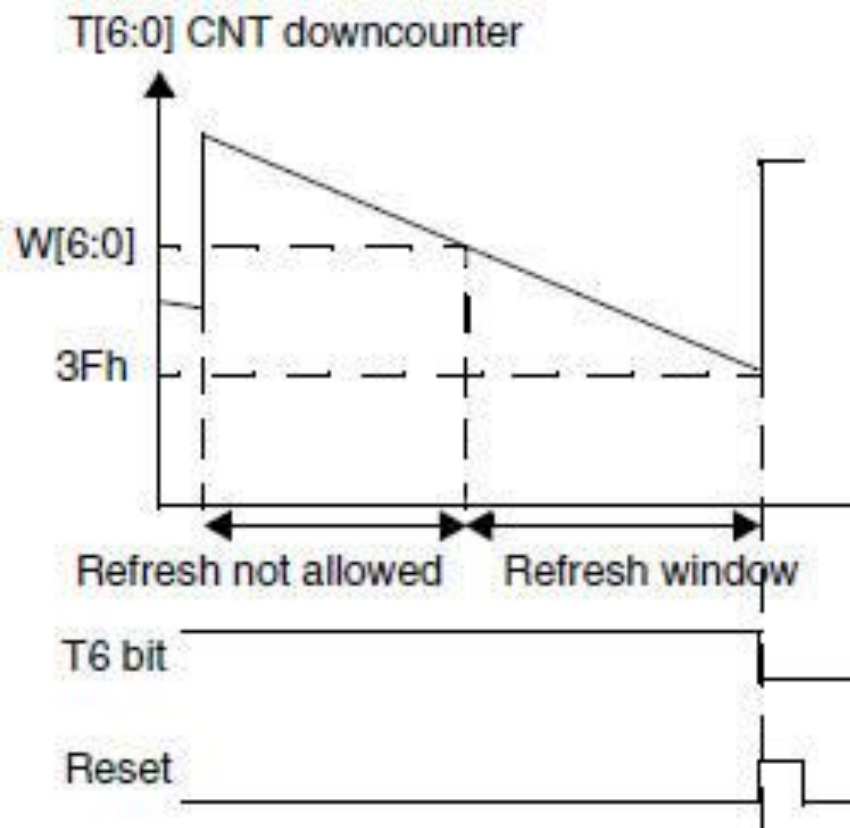
WWDG

2010.06.17

WWDG 개요

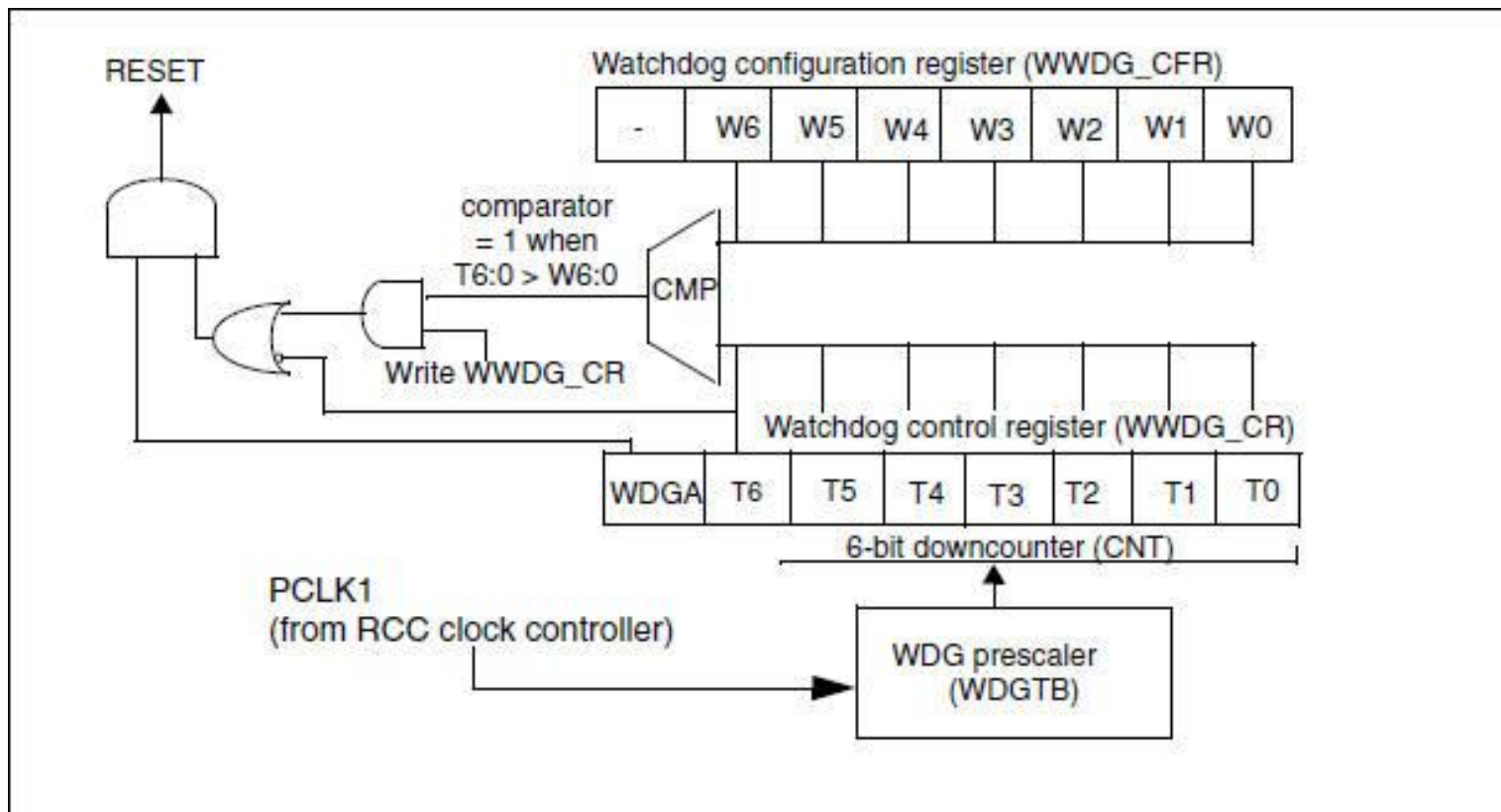
- window watchdog은 어떤 software 상의 오류를 검출해주는 용도로 사용이 가능하다
- software 상의 오류가 나타나는 원인은 어떤 외부적인 요인에 의한 것일 수도 있고, 내부 알고리즘 상의 예상치 못한 문제일 수도 있다.
- 이러한 문제가 발생했을 때 software의 수행 시간 등의 변화가 발생할 수 있고, 이러한 변화에 무척 민감한 제품의 경우 이를 검출해서 대처해야 한다.
- watchdog 회로는 프로그램 되어 있는 기간을 소모해 버리면 MCU reset을 발생시키게 된다. 이를 발생하지 못하게 하기 위해서 주기적으로 down counter의 내용을 재 설정 해주어야 한다.
 - T6 비트가 0이 되는 순간에 리셋이 발생
 - 이 상황이 오기 전에 down counter의 내용을 재 설정 해주어야 한다.
- 재 설정 하는 작업을 Refresh라고 부르는데 이것을 수행할 수 있는 영역이 제한되어 있다.

Refresh window



- Refresh가 가능한 부분과 Refresh를 할 수 없는 구간을 설정할 수 있다.
- Refresh가 허락되지 않은 구간에서 Refresh를 하는 행위는 또한 MCU reset을 일어나도록 만든다.
- T[6:0] 값이 시간이 진행됨에 따라서 감소하는 그림을 보여주고 있다.
- 3Fh라는 값은 T[6] 비트가 0이 되는 순간을 의미하는 것이다.
- W[6:0]은 window register 값을 의미하는 것이다.

WWDG Reset 조건



- WWDG 리셋이 발생할 수 있는 것은 2가지
 - 하나는 down counter 값이 0x40보다 아래로 떨어지는 경우이고,
 - 다른 하나는 down counter 값이 window의 범위를 벗어나서 설정되는 것 즉, $T[6:0]$ 값이 $W[6:0]$ 값보다 크게 되는 상황이 되면 리셋이 발생

Early wakeup interrupt (EWI)

- EWI가 enable되면 down counter의 값이 0x40이 되는 순간 발생
- 이 interrupt를 이용해서 WWDG_IRQHandler에서 down counter의 값을 재 설정하는 과정을 수행할 수 있다.
- WWDG_SR register의 EWIF bit를 0으로 쓰게 되면 이 EWI 인터럽트를 clear할 수 있다.
- WWDG_CR register의 WDGA bit를 1로 설정하면 watchdog 기능이 활성화 된다.
 - 이때부터 T[6:0] 비트를 7-bit downcounter로 해서 계속 값이 감소한다.
 - 0x40에서 0x3F가 되는 순간 즉, T6 비트가 0이 되는 순간 리셋이 된다.
 - EWI를 활성화시키면 이렇게 0x3F가 되어 리셋이 되기 전에 인터럽트를 발생시킬 수 있는 것이다.
 - 만약 counter 값이 window register에 저장된 값보다 큰 상황에서 counter 값을 재 설정하게 되면 이때도 역시 리셋이 된다.

Watchdog enable

- watchdog은 리셋이 되면 늘 disable되어 있는 상태가 된다.
- 이것을 활성화 시키는 것은 WWDG_CR register의 WDGA bit를 1로 설정하는 것이다.
- 하지만 한번 활성화를 시키면 그 이후에 이 값을 0으로 만드는 것은 그 비트에 0을 쓴다고 해도 불가능하다.
- 오직 리셋을 통해서 하드웨어적으로 초기화시키는 방법 밖에 없다.
- downcounter는 free-running
 - 이것이 의미하는 것은 downcounter의 값이 줄어드는 것은 자동
 - 심지어 watchdog이 disable되어도 계속 downcounter의 값이 줄어드는 동작을 하게 된다.
 - watchdog이 enable될 때 반드시 T6 비트는 1로 설정되어 있어야 한다.
 - 만약 그렇지 않고 0으로 되어 있으면 watchdog을 enable 시키자마자 리셋이 발생해버릴 것이다.

WWDG Time

WDGTB	Min timeout value	Max timeout value
0	113 μ s	7.28 ms
1	227 μ s	14.56 ms
2	455 μ s	29.12 ms
3	910 μ s	58.25 ms

- WWDG에 클럭으로 작용하는 것은 PCLK1 (36,000,000)
- 이 값을 4096으로 나눈 값을 기본으로 해서 WDTB[1:0]에 있는 값을 한번 더 나누어 준다.
- 예제 코드에서는 이 값을 11로 설정했다.
- 즉 2의 3승 8로 나누게 되는 것이다.
- 36,000,000을 4096으로 나누면 약 8,789가 나오고 이것을 다시 8로 나누면 약 1,099가 된다. 즉 1초에 1,099번 동작하기 때문에 한 값이 감소되는 시간은 약 910 usec가 되는 것이다.
- IWDG에 비해 매우 작은 간격으로 설정을 할 수 있다. 매우 정밀한 Watchdog을 구현할 수 있는 것이다.

예제 코드 시나리오

- Early Wakeup interrupt (EWI)를 이용 WWDG counter를 재 설정
- WWDG timeout은 최대 값이 58.25 ms가 되도록 설정
- refresh window는 65로 설정 - 0x41
- WWDG counter는 64 (0x40)가 되었을 때 EWI 인터럽트가 발생
- 이때 refresh window 값은 65이기 때문에 WWDG counter 보다 작고 downcounter의 값을 재 설정할 수 있다. IWDG에서와 마찬가지로 Yellow LED를 toggle 시키도록 설정
- 문제를 일으키는 방법
 - **KEY1 EXTI line priority를 WWDG 인터럽트보다 높게 설정**
 - EXTI 인터럽트 발생 시 EXTI line pending bit clear 작업을 하지 않는다.
 - EXTI 인터럽트가 계속 호출 되고 결국 WWDG 인터럽트는 수행 되지 못함
- WWDG 인터럽트가 수행되지 못함으로 인해 timeout 값이 다 끝나는 (WWDG counter의 6번 비트가 0이 됨) 약 58.25 msec가 지나고 나면 결국 WWDG reset이 발생
- WWDG reset 발생한 이후에 이 리셋이 WWDG reset으로 인해서 발생한 것인지 아니면 최초 Power on으로부터 발생한 것인지를 구분해서 빨간색 LED로 이것을 구분하도록 설정

Priority 설정

```
void NVIC_Configuration(void) {  
.....  
    NVIC_PriorityGroupConfig(NVIC_PriorityGroup_1);  
.....  
    /* Enable the EXTI1 Interrupt */  
    NVIC_InitStructure.NVIC_IRQChannel = EXTI1_IRQn;  
    NVIC_InitStructure.NVIC_IRQChannelPreemptionPriority = 0;  
    NVIC_InitStructure.NVIC_IRQChannelSubPriority = 1;  
    NVIC_InitStructure.NVIC_IRQChannelCmd = ENABLE;  
    NVIC_Init(&NVIC_InitStructure);  
    /* Enable the WWDG Interrupt */  
    NVIC_InitStructure.NVIC_IRQChannel = WWDG_IRQn;  
    NVIC_InitStructure.NVIC_IRQChannelPreemptionPriority = 1;  
    NVIC_InitStructure.NVIC_IRQChannelSubPriority = 1;  
    NVIC_InitStructure.NVIC_IRQChannelCmd = ENABLE;  
    NVIC_Init(&NVIC_InitStructure);  
}
```

- PriorityGroup을 0이 아닌 값으로 설정
- EXTI1_IRQn은 PreemptionPriority를 0으로 설정
- WWDG_IRQn은 1로 설정 - EXTI1_IRQn가 더 높게 만든다.

Early Wakeup interrupt (EWI) Handler

```
void WWDG_IRQHandler(void)
{
    /* Update WWDG counter */
    WWDG_SetCounter(0x7F);

    /* Clear EWI flag */
    WWDG_ClearFlag();

    /* Toggle Yellow */
    LED_Toggle_Yellow();
}
```

- WWDG_IRQHandler는 EWI 인터럽트가 발생했을 때 호출
- 카운터 값은 0x7F로 최대 값으로 설정한다.