



CANTUS-CAN

- WATCHDOG -

32bits EISC Microprocessor *CANTUS*

Ver 1.1
April 24, 2013

Advanced Digital Chips Inc.

History

2013-02-19	Released	
2013-04-24	Modified	CANTUS-CAN

CANTUS-CAN Application Note : #0006B WATCHDOG

©Advanced Digital Chips Inc.

All right reserved.

No part of this document may be reproduced in any form without written permission from Advanced Digital Chips Inc.

Advanced Digital Chips Inc. reserves the right to change in its products or product specification to improve function or design at any time, without notice.

Office

(Gwanyang-dong, Keumkang Pentarium IT Tower) 22F, A-Tower,
282, Hagui-ro, Dongan-gu, Anyang-si, Gyeonggi-do, SEOUL 431-810 Rep. of KOREA

Tel : +82-31-463-7500

Fax : +82-31-463-7588

URL : <http://www.adc.co.kr>

— Table of Contents —

1 SUMMARY.....6

2 WATCHDOG TIMER RESET MODE.....7

2.1 REGISTER SET7

2.2 FUNCTION SET.....12

— List of Figures —

그림 2-1 Watchdog Timer Reset Mode Register Set 7

— List of Tables —

<i>Æ 2-1 Interrupt Enable Register (INTEN)</i>	8
<i>Æ 2-2 Watchdog Timer Counter Value Register (WDTCNT)</i>	9
<i>Æ 2-3 Watchdog Timer Control Register (WDTCTRL)</i>	9
<i>Æ 2-4 Interrupt Pending Register (INTPEND)</i>	10
<i>Æ 2-5 Interrupt Pending Register (PENDCLR)</i>	11

1 Summary

이 문서는 CANTUS의 Watchdog Timer에 대한 Application Note이다.

CANTUS는 32bit Watchdog Timer를 갖고 있다.

WDCNT를 '1'씩 감소 시켜 '0'가 되면, Watchdog Reset을 발생하여 System을 Reset 한다. Watchdog Reset으로 인해 System이 Reset되지 않기 위해서는 WDCNT가 '0'가 되기 전에 WDCNT를 다시 써야 한다.

이 문서는 CANTUS의 Watchdog Timer를 사용하는 방법을 기술 한다.

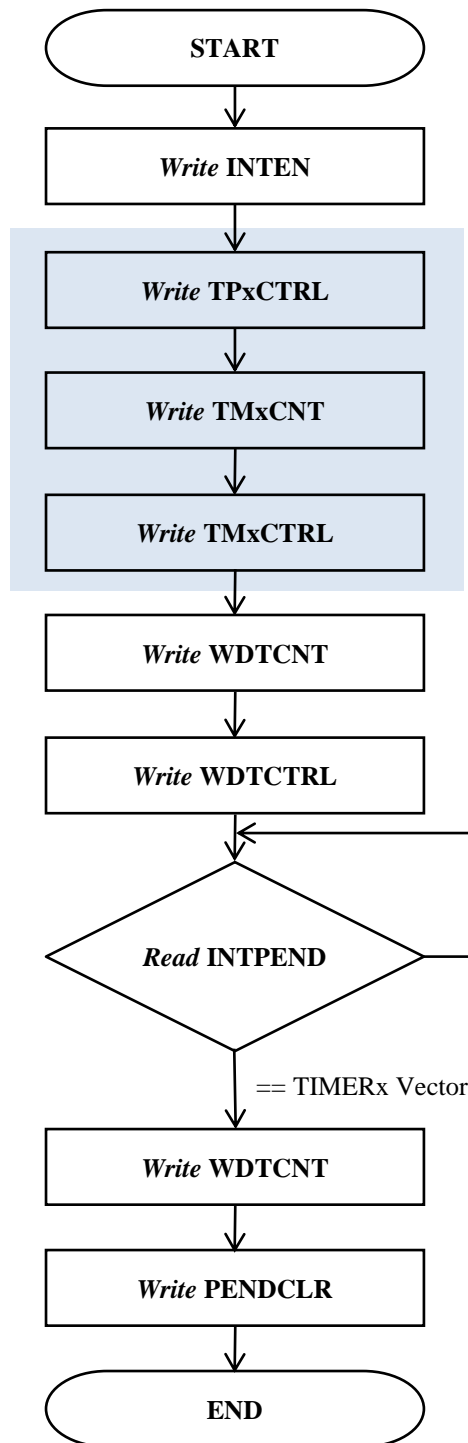
CANTUS의 Watchdog Timer는 CANTUS Datasheet '7 WATCHDOG TIMER'를 참조 하라.

2 WATCHDOG TIMER Reset Mode

2.1 Register Set

CANTUS의 Watchdog Timer를 Reset Mode로 사용하기 위해 다음과 같은 순서로 Register를 설정한다. 일반적으로 Watchdog Reset을 발생하지 않기 위해 TIMER(Timer/Counter Mode)를 사용하여 일정 주기로 WDTCNT를 다시 쓴다.

그림 2-1 Watchdog Timer Reset Mode Register Set



INTEN (Interrupt Enable Register)

이 Register를 설정하면 Interrupt Event를 Interrupt Controller에서 INTPEND에 기록한다. TIMERx Vector Number의 bit에 '1b'을 씌우므로써 TIMERx Interrupt를 설정한다.

표 2-1 Interrupt Enable Register (INTEN)

<i>Bit</i>	<i>R/W</i>	<i>Description</i>	<i>Default Value</i>
31	R/W	Vector No. 0x3F Interrupt Enable bit	0
30	R/W	Vector No. 0x3E Interrupt Enable bit	0
29	R/W	Vector No. 0x3D Interrupt Enable bit	0
28	R/W	Vector No. 0x3C Interrupt Enable bit	0
27	R/W	Vector No. 0x3B Interrupt Enable bit	0
26	R/W	Vector No. 0x3A Interrupt Enable bit	0
25	R/W	Vector No. 0x39 Interrupt Enable bit	0
24	R/W	Vector No. 0x38 Interrupt Enable bit	0
23	R/W	Vector No. 0x37 Interrupt Enable bit	0
22	R/W	Vector No. 0x36 Interrupt Enable bit	0
21	R/W	Vector No. 0x35 Interrupt Enable bit	0
20	R/W	Vector No. 0x34 Interrupt Enable bit	0
19	R/W	Vector No. 0x33 Interrupt Enable bit	0
18	R/W	Vector No. 0x32 Interrupt Enable bit	0
17	R/W	Vector No. 0x31 Interrupt Enable bit	0
16	R/W	Vector No. 0x30 Interrupt Enable bit	0
15	R/W	Vector No. 0x2F Interrupt Enable bit	0
14	R/W	Vector No. 0x2E Interrupt Enable bit	0
13	R/W	Vector No. 0x2D Interrupt Enable bit	0
12	R/W	Vector No. 0x2C Interrupt Enable bit	0
11	R/W	Vector No. 0x2B Interrupt Enable bit	0
10	R/W	Vector No. 0x2A Interrupt Enable bit	0
9	R/W	Vector No. 0x29 Interrupt Enable bit	0
8	R/W	Vector No. 0x28 Interrupt Enable bit	0
7	R/W	Vector No. 0x27 Interrupt Enable bit	0
6	R/W	Vector No. 0x26 Interrupt Enable bit	0
5	R/W	Vector No. 0x25 Interrupt Enable bit	0
4	R/W	Vector No. 0x24 Interrupt Enable bit	0
3	R/W	Vector No. 0x23 Interrupt Enable bit	0
2	R/W	Vector No. 0x22 Interrupt Enable bit	0
1	R/W	Vector No. 0x21 Interrupt Enable bit	0
0	R/W	Vector No. 0x20 Interrupt Enable bit	0

*** Interrupt Enable bit

0 : Interrupt Disable

1 : Interrupt Enable

WDTCNT (Watchdog Timer Counter Value Register)

Watchdog Timer는 APB Clock(PCLK)을 사용한다. WDTCNT는 이 Clock에 동기 되어 1씩 감소 된다. '0'가 되면 Watchdog Reset을 발생하여 System을 Reset하므로, WDTCNT가 '0'가 되기 전에 다시 써야 한다.

표 2-2 Watchdog Timer Counter Value Register (WDTCNT)

<i>Bit</i>	<i>R/W</i>	<i>Description</i>	<i>Default Value</i>
31 : 0	R/W	Watchdog timer counter 32-bit value. Down-counter	0xFFFF_FFFF

WDTCTRL (Watchdog Timer Control Register)

WDTMOD에 '0b'를 씌으로써 Reset Mode로 설정하고, WDTEN에 '1b'를 쓰면 Watchdog Timer가 동작한다.

표 2-3 Watchdog Timer Control Register (WDTCTRL)

<i>Bit</i>	<i>R/W</i>	<i>Description</i>	<i>Default Value</i>
31 : 1	R	Reserved	-
0	R/W	WDTEN : Watchdog timer enable bit 0 : Disable 1 : Enable	0

INTPEND (Interrupt Pending Register)

INTEN에 설정된 Interrupt Vector의 Interrupt Event를 기록한다.

표 2-4 Interrupt Pending Register (INTPEND)

Bit	R/W	Description	Default Value
31	R	Vector No. 0x3F Interrupt Pending bit	-
30	R	Vector No. 0x3E Interrupt Pending bit	-
29	R	Vector No. 0x3D Interrupt Pending bit	-
28	R	Vector No. 0x3C Interrupt Pending bit	-
27	R	Vector No. 0x3B Interrupt Pending bit	-
26	R	Vector No. 0x3A Interrupt Pending bit	-
25	R	Vector No. 0x39 Interrupt Pending bit	-
24	R	Vector No. 0x38 Interrupt Pending bit	-
23	R	Vector No. 0x37 Interrupt Pending bit	-
22	R	Vector No. 0x36 Interrupt Pending bit	-
21	R	Vector No. 0x35 Interrupt Pending bit	-
20	R	Vector No. 0x34 Interrupt Pending bit	-
19	R	Vector No. 0x33 Interrupt Pending bit	-
18	R	Vector No. 0x32 Interrupt Pending bit	-
17	R	Vector No. 0x31 Interrupt Pending bit	-
16	R	Vector No. 0x30 Interrupt Pending bit	-
15	R	Vector No. 0x2F Interrupt Pending bit	-
14	R	Vector No. 0x2E Interrupt Pending bit	-
13	R	Vector No. 0x2D Interrupt Pending bit	-
12	R	Vector No. 0x2C Interrupt Pending bit	-
11	R	Vector No. 0x2B Interrupt Pending bit	-
10	R	Vector No. 0x2A Interrupt Pending bit	-
9	R	Vector No. 0x29 Interrupt Pending bit	-
8	R	Vector No. 0x28 Interrupt Pending bit	-
7	R	Vector No. 0x27 Interrupt Pending bit	-
6	R	Vector No. 0x26 Interrupt Pending bit	-
5	R	Vector No. 0x25 Interrupt Pending bit	-
4	R	Vector No. 0x24 Interrupt Pending bit	-
3	R	Vector No. 0x23 Interrupt Pending bit	-
2	R	Vector No. 0x22 Interrupt Pending bit	-
1	R	Vector No. 0x21 Interrupt Pending bit	-
0	R	Vector No. 0x20 Interrupt Pending bit	-

*** Interrupt Pending Register의 각 비트의 값은 해당 인터럽트가 발생하였음을 나타낸다.

Interrupt Pending Register의 값은 Interrupt Pending Clear 레지스터에 의해 Clear된다.

일반적으로 해당 Interrupt가 끝날 때 Clear한다.

PENDCLR (Interrupt Pending Clear Register)

Interrupt Vector Number값으로 씬으로써 INTPEND의 Event를 Clear한다. INTPEND를 Clear하지 않으면 더 이상 Core에 Interrupt Request를 발생하지 않는다.

표 2-5 Interrupt Pending Register (PENDCLR)

<i>Bit</i>	<i>R/W</i>	<i>Description</i>	<i>Default Value</i>
31 : 8	R	Reserved	-
7 : 0	W	Interrupt Pending Register Clear Value (0x20 ~ 0x3F)	0xFF

*** Interrupt Pending Register를 Clear 하기 위해서는 Interrupt Vector No. 값으로 Write 한다.

2.2 Function Set

아래는 CANTUS의 Watchdog Timer를 1초로 설정하고, TIMER 2를 사용하여 500ms 주기로 WDCNT를 다시 쓰는 예이다. 여기서 APB Clock(PCLK)은 약 48MHz이다.

```
#define WATCHDOG_V 47619047
main()
{
    ...

    U32 tmcnt;
    U32 temp;
    U32 i = 0;

    tmcnt = 732;

    *R_INTEN |= (1<<INTNUM_TIMER2);

    *R_TP2CTRL = F_TPCTRL_CNTRST;
    *R_TP2CTRL &= ~(F_TPCTRL_CNTRST);

    *R_TM2CNT = tmcnt;
    *R_TM2CTRL = 0;

    *R_WDCNT = WATCHDOG_V;
    *R_WDTCTRL = F_WDTCTRL_WDTEN_EN;

    *R_TM2CTRL = F_TMCTRL_PFSEL_32768 | F_TMCTRL_TMEN;

    while(1)
    {
        temp = *R_INTPEND;
        if( (temp & (1<<INTNUM_TIMER2)) == (1<<INTNUM_TIMER2))
        {
            *R_WDCNT = WATCHDOG_V;
            *R_PENDCLR = (INTNUM_TIMER2+0x20);
            i++;

            debugprintf("i = %d\r\n", i);
        }
    }
    ...
}
```

*R_INTEN

Interrupt Enable Register는 SDK/Include/CANTUS/interrupt.h에

```
#define R_INTEN ((volatile unsigned int*)(0x80020810))
```

로 정의되어 있다.

▶ 여기서는 Interrupt Enable Register 0에 (1<<INTNUM_TIMER2)를 대입하여 TIMER 2 Interrupt Enable을 설정 한다.

***R_WDTCNT**

Watchdog Timer Counter Value Register 는 SDK/include/CANTUS/watchdog.h에

```
#define R_WDTCNT ((volatile unsigned int*)0x80020C04)
```

로 정의되어 있다.

▶ 여기서는 R_WDTCNT에 1초를 위한 값을 저장한다. 이 값의 시간은 CANTUS의 PCLK에 의해 결정되며 '0'가 되기 전에 다시 쓰지 않으면 Watchdog Reset을 발생하여 System을 Reset한다.

***R_WDTCTRL**

Watchdog Timer Control Register 는 SDK/include/CANTUS/watchdog.h에

```
#define R_WDTCTRL ((volatile unsigned int*)0x80020C00)
#define F_WDTCTRL_WDTEN_DIS      (0<<0)
#define F_WDTCTRL_WDTEN_ENABLE  (1<<0)
```

로 정의되어 있다.

▶ 여기서는 Watchdog Timer Control Register에 F_WDTCTRL_WDTEN_ENABLE을 대입하여 Watchdog Timer를 동작시킨다.

***R_INTPEND**

Interrupt Pending Register는 SDK/Include/CANTUS/interrupt.h에

```
#define R_INTPEND ((volatile unsigned int*)(0x8002080C))
```

로 정의되어 있다.

▶ 여기서는 Interrupt Pending Register 0를 읽어 (1<<INTNUM_TIMER2)와 비교한다. R_INTEN로 Enable된 TIMER 2는 Interrupt Event를 R_INTPEND에 기록한다.

***R_PENDCLR**

Interrupt Pending Clear Register는 SDK/include/CANTUS/interrupt.h에

```
#define R_PENDCLR ((volatile unsigned int*)0x80020800)
```

로 정의되어 있다.

▶ 여기서는 Interrupt Pending Clear Register에 (INTNUM_TIMER2 + 0x20)을 대입하여 Interrupt Pending Register를 Clear한다. Interrupt Pending Register를 Clear하지 않으면 더 이상 Core에 Interrupt Request를 발생하지 않는다.