Application Note

32-bit Cortex™-M0 MCU
NuMicro® Family

How to use WDT?

Application Note

Table of Contents-

1   INTRODUCTION

The purpose of Watchdog Timer is to perform a system reset after the software running into a problem. This prevents system from hanging for an indefinite period of time. The watchdog timer includes a 19-bit free running counter with programmable time-out intervals. The details of feature and structure are as follows.

1.1   Feature

• 19-bit WDT counter

• Six selections of Watch dog timer interval

• Support WDT interrupt and WDT reset

1.2   Structure

Setting WDTCR.WTE[7] enables the watchdog timer and the WDT counter starts counting up. When the counter reaches the selected time-out interval, Watchdog timer interrupt flag WTIF will be set immediately to request a WDT interrupt if the watchdog timer interrupt enable bit WTIE is set, in the meanwhile a specified delay time follows the time-out event. User must set WDTCR.WTR[0] (Watchdog timer reset) high to reset the 19-bit WDT counter to avoid CPU from Watchdog timer reset before the delay time expires. WTR bit is auto cleared by hardware after WDT counter is reset. There are eight time-out intervals with specific delay time which are selected by Watchdog timer interval select bits (WDTCR.WTIS[10:8]). If the WDT counter has not been cleared after the specific delay time expires, the watchdog timer will set Watchdog Timer Reset Flag (WTRF) high and reset CPU. This reset will last 64 WDT clocks then CPU restarts executing program from reset vector (0x0000 0000). WTRF will not be cleared by Watchdog reset. User may poll WTFR by software to recognize the reset source.

Table 1 Watchdog Timeout Interval Selection

| WTIS	Interrupt Timeout	Watchdog Reset Timeout	WTR Timeout Interval (WDT_CLK=12MHz)	WTR Timeout Interval (WDT_CLK=32KHz)
000	2^4 WDT_CLK	$(2^4 + 1024)$ WDT_CLK	86.67 us	32.5 ms
001	2^6 WDT_CLK	$(2^6 + 1024)$ WDT_CLK	90.67 us	34 ms
010	2^8 WDT_CLK	$(2^8 + 1024)$ WDT_CLK	106.67 us	40 ms
011	2^{10} WDT_CLK	$(2^{10} + 1024)$ WDT_CLK	170.67 us	64 ms
100	2^{12} WDT_CLK	$(2^{12} + 1024)$ WDT_CLK	426.67 us	160 ms
101	2^{14} WDT_CLK	$(2^{14} + 1024)$ WDT_CLK	1.45 ms	544 ms
110	2^{16} WDT_CLK	$(2^{16} + 1024)$ WDT_CLK	5.55 ms	2080 ms
111	2^{18} WDT_CLK	$(2^{18} + 1024)$ WDT_CLK	21.93 ms	8224 ms
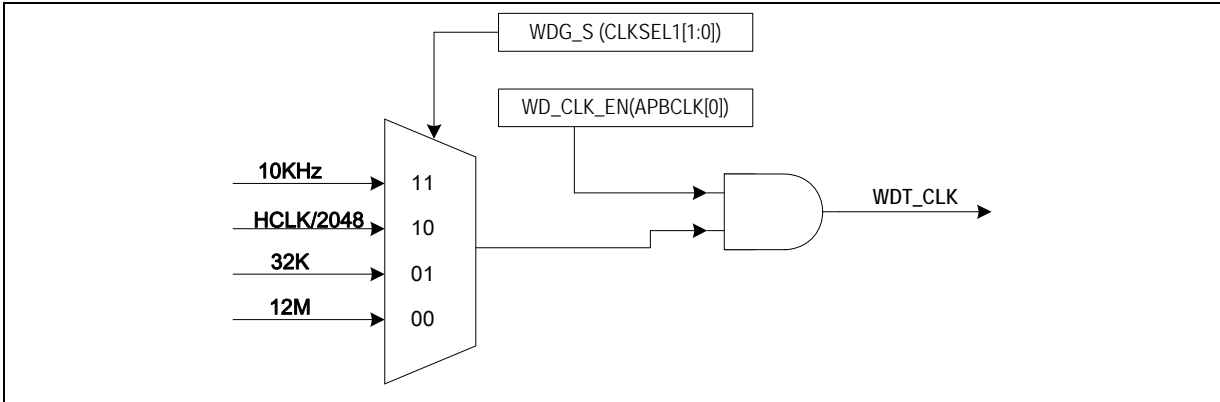
Figure 1 Watchdog Timer Clock Control

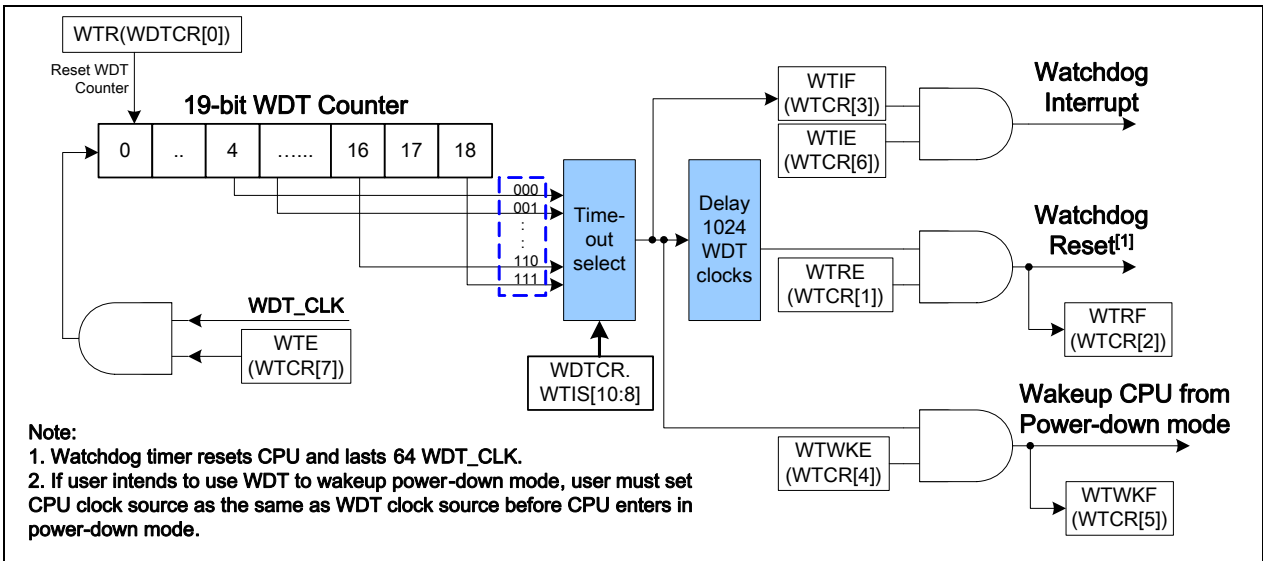Figure 2 Watchdog Timer Block Diagram

2   HOW TO PROGRAM WDT

2.1   PROGRAM FLOW OF WDT

1. Select **WDG_S** bit in **CLKSEL1** register and set the **WDG_EN** bit in **APBCLK** register to enable **WDG** clock source
2. Select **WTIS** bit in **WTCR** register for Timeout Interval
3. Set **WTRE** bit in **WTCR** register to enable Watchdog Timer Reset function
4. Set **WTIE** bit in **WTCR** register and **ISER** to enable interrupt.
5. Set **WTE** bit in **WTCR** register to enable Timer module

2.2   Sample code

```c
#include <stdio.h>

#include "NUC1xx.h"

/*-------------------------------------------------------------------------

  Define variable

 -------------------------------------------------------------------------*/

        static uint32_t WDTCOUN=0;

/*-------------------------------------------------------------------------

  Function subroutine

 -------------------------------------------------------------------------*/

void Delay(uint32_t delayCnt)

{

   while(delayCnt--)

   {

      __NOP();

      __NOP();

   }

}

/*-------------------------------------------------------------------------

  Interrupt subroutine

 -------------------------------------------------------------------------*/

void WDT_IRQHandler(void) // Timer0 interrupt subroutine

{

        UNLOCKREG();

        WDT->WTCR.WTIF =1;

        WDTCOUN++;
```

```
        if (WDTCOUN<10) WDT->WTCR.WTR=1;          //Wait 10 times


        LOCKREG();
        GPIOA->DOUT = 0x00; //Toggle GPIOA_0
        Delay(10000);
        GPIOA->DOUT = 0x01;
}
/*-------------------------------------------------------------------------
  MAIN function
  -------------------------------------------------------------------------*/
int32_t main (void)
{
        GPIOA->PMD.PMD0=1;
        Delay(1000);
        while(GPIOA->PIN&0x8000);                 //Wait low signal of GPIOA_15


        NVIC_DisableIRQ(WDT_IRQn);                //Disable WDT interrupt
        outpw(&WDT->WTCR ,0 );                    //Disable WDT
        UNLOCKREG();
        /* Step 1. Enable and Select WDT clock source */
        SYSCLK->CLKSEL1.WDG_S =3;                 //Select 10Khz for WDT clock source
        SYSCLK->APBCLK.WDG_EN =1;                 //Enable WDT clock source


        /* Step 2. Select Timeout Interval */
        WDT->WTCR.WTIS=4;                         //Select level 3 (11-bit mode)


        /* Step 3. Enable Watchdog Timer Reset function */
        WDT->WTCR.WTRE = 1;


        /* Step 4. Enable WDT interrupt */
        WDT->WTCR.WTIF =1;                        //Write 1 to clear for safety
        WDT->WTCR.WTIE = 1;
        NVIC_EnableIRQ(WDT_IRQn);                 //Enable WDT Interrupt


        /* Step 5. Enable WDT module */
```

```
        WDT->WTCR.WTE = 1;              //Enable WDT
    LOCKREG();
    while(1);
}
```

3   REVISION HISTORY

REV.	DATE	DESCRIPTION
1.00	March 1, 2010	1. Initially issued.
1.01	April 8, 2010	1. Remove register description

Important Notice

Nuvoton products are not designed, intended, authorized or warranted for use as components in systems or equipment intended for surgical implantation, atomic energy control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, combustion control instruments, or for other applications intended to support or sustain life. Further more, Nuvoton products are not intended for applications wherein failure of Nuvoton products could result or lead to a situation wherein personal injury, death or severe property or environmental damage could occur.

Nuvoton customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Nuvoton for any damages resulting from such improper use or sales.

Please note that all data and specifications are subject to change without notice. All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.