

FT60F01X

WDT Application note

目录

1. WDT 相关寄存器的设置	3
2. 看门狗周期设置	4
3. 应用范例	5
联系信息	8

FT60F01x WDT 应用

1. WDT 相关寄存器的设置

看门狗的时钟源为内部慢时钟（32kHz），它是一个 16 位的计数器，和定时器 0 共用一个 8 位的预分频器，使能位位于配置寄存器 UCFG0 的第 3 位，WDTEN，为 1 时表示使能看门狗，为 0 时禁止，由外部串口写入控制。在使能了看门狗的情况下，MCU 睡眠时看门狗溢出事件可以作为一个唤醒源，而 MCU 正常工作时作为一个复位源。

相关寄存器的各个位定义如下：

1) WDTCON 寄存器

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN
Reset	-	-	-	0	1	0	0	0

Bit7~Bit5：保留位，读零

Bit4~Bit1 看门狗定时器周期选择位

WDTPS<3:0>	预分频比值	WDTPS<3:0>	预分频比值
0000	1:32	1000	1:8192
0001	1:64	1001	1:16384
0010	1:128	1010	1:32768
0011	1:256	1011	1:65536
0100	1:512(复位值)	1100	1:65536
0101	1:1024	1101	1:65536
0110	1:2048	1110	1:65536
0111	1:4096	1111	1:65536

Bit0：软件打开/关闭看门狗定时器位

1：打开

0：关闭

2.) OPTION 寄存器

Bit	7	6	5	4	3	2	1	0
Name	/PAPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
Reset	1	1	1	1	1	1	1	1

Bit7：PORTA 口上拉使能位

1：上拉功能被禁止

0：上拉功能使能

Bit6：触发中断边沿选择位

1：PA2/INT 上升沿触发中断

0：PA2/INT 下降沿触发中断

Bit5：Timer0 时钟选择位

1：PA2/T0CKI 管脚输入时钟

0：内部指令周期Fosc/4

Bit4: Timer0 时钟边沿选择位

- 1: PA2/T0CKI管脚由高到底变化时计数增加
- 0: PA2/T0CKI管脚由低到高变化时计数增加

Bit3: 预分频分配位

- 1: 预分频器分配给WDT
- 0: 预分频器分配给Timer0

Bit2~Bit0: 预分频大小设置位

Bit2: Bit0	Timer0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

3) UCFG0 寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
NA	CPB	MCLRE	PWRTEB	WDTE	NA	NA	FOSC0

Bit7: 保留位

Bit6: Flash 内容保护使能位

- 1: PROM 内容不保护
- 0: 启动 PROM 内容保护, MCU 能读, 串口不能读

Bit5: MCLRE 管脚功能选择位

- 1: PA3/MCLR 脚执行 MCLR 功能, 是复位脚
- 0: PA3/MCLR 脚执行 PA3 功能, 是数字输入引脚

Bit4: PWRTEB 上电延时使能位

- 1: PWRT 禁止
- 0: PWRT 使能

Bit3: WDTE 看门狗使能位

- 1: WDT 使能, 程序不能禁止
- 0: WDT 禁止, 但程序可通过设置 WDTCON 的 SWDTEN 位将 WDT 使能

Bit2~Bit1: 保留位, 读 0

Bit0: FOSC 时钟源选择位

- 1: INTOSC 模式, PA4 输出系统时钟的 4 分频, PA5 为 IO 引脚
- 0: INTOSCIO 模式, PA4 为 IO 引脚, PA5 为 IO 引脚

2. 看门狗周期设置

定时时长计算公式如下:

$$\text{看门狗周期} = (1/32000) * 16 \text{ 位预分频值} * 8 \text{ 位预分频值}$$

例如 WDTPS<3:0>=0000, PSA<2:0>=010, 看门狗周期=(1/32000)*32*4=4ms

3. 应用范例

```
//*****
/* 文件名: TEST_60F01x_WDT.c
* 功能:    FT60F01x-WDT 功能演示
* IC:      FT60F011A SOP8
* 晶振:    16M/4T
* 说明:    程序中开启看门狗并将看门狗时间设置为 32ms
*          a. 主函数先在 DemoPortOut 脚输出一个高 3ms 低 3ms 的信号,
*          b. 然后循环输出高 1ms 低 1ms, 500Hz 的信号。
*          如果不在主程序中清看门狗, 则每隔 32ms 单片机复位后, 会重复输出 a+b 信号;
*          如果在主函数中清看门狗, 则一直输出 b 信号
*
*          FT60F011A  SOP8
*          -----
*  VDD-----|1(VDD)   (GND)8|-----GND
*  NC-----|2(PA2)   (PA4)7|--DemoPortOut
*  NC-----|3(PA1)   (PA5)6|-----NC
*  NC-----|4(PA3)   (PA0)5|-----NC
*          -----
*/
//*****
#include "SYSCFG.h"
//*****宏定义*****
#define DemoPortOut PA4
/*-----
* 函数名: POWER_INITIAL
* 功能:   上电系统初始化
* 输入:   无
* 输出:   无
*-----*/
void POWER_INITIAL(void)
{
    OSCCON = 0B01110000;    //IRCF=111=16MHz/4T=4MHz,0.25us
    INTCON = 0;             //暂禁止所有中断

    PORTA = 0B00000000;
    TRISA = 0B00000000;    //PA 输入输出 0-输出 1-输入
                          //PA4->输出
    WPUA = 0B00000000;     //PA 端口上拉控制 1-开上拉 0-关上拉

    OPTION = 0B00001000;    //Bit3=1 WDT MODE,PS=000=1:1 WDT RATE
    MSCKCON = 0B00000000;
    //Bit4=0,禁止 LVR(60F01x O 版之前)
    //Bit4=0,LVREN 使能时,开启 LVR(60F01x O 版及 O 版之后)
```

//Bit4=1,LVREN 使能时,工作时开启 LVR,睡眠时自动关闭 LVR(60F01x O 版及 O 版后)

```

}
/*-----
* 函数名称: DelayUs
* 功能:      短延时函数 --16M-4T--大概快 1%左右.
* 输入参数: Time 延时时间长度 延时时长 Time*2Us
* 返回参数: 无
-----*/

```

```
void DelayUs(unsigned char Time)
```

```

{
    unsigned char a;
    for(a=0;a<Time;a++)
    {
        NOP();
    }
}
/*-----
* 函数名称: DelayMs
* 功能:      短延时函数
* 输入参数: Time 延时时间长度 延时时长 Time ms
* 返回参数: 无
-----*/

```

```
void DelayMs(unsigned char Time)
```

```

{
    unsigned char a,b;
    for(a=0;a<Time;a++)
    {
        for(b=0;b<5;b++)
        {
            DelayUs(98); //快 1%
        }
    }
}
/*-----
* 函数名称: DelayS
* 功能:      短延时函数
* 输入参数: Time 延时时间长度 延时时长 Time S
* 返回参数: 无
-----*/

```

```
void DelayS(unsigned char Time)
```

```

{
    unsigned char a,b;
    for(a=0;a<Time;a++)
    {

```

```

        for(b=0;b<10;b++)
        {
            DelayMs(100);
        }
    }
}

/*-----*/
* 函数名: WDT_INITIAL
* 功能:   初始化设置看门狗 1S 时间复位
* 设置看门狗定时时长 = 1/32000*16 位预分频值*8 位预分频值
*              = 1/32000*1024*1 = 32ms
*-----*/

void WDT_INITIAL (void)
{
    CLRWDT();           //清看门狗
    PSA=1;              //时钟分频分给 WDT
    WDTCON = 0B00001011; //WDTPS=0101=1:1024,PS=000=1:1
}

/*-----*/
* 函数名: main
* 功能:   主函数
* 输入:   无
* 输出:   无
*-----*/

void main()
{
    POWER_INITIAL();    //系统初始化
    WDT_INITIAL();
    DemoPortOut = 1;
    DelayMs(3);          //3ms
    DemoPortOut = 0;
    DelayMs(3);          //3ms
    while(1)
    {
        //CLRWDT();      //清看门狗
        DemoPortOut = 1;
        DelayMs(1);       //1ms
        DemoPortOut = 0;
        DelayMs(1);       //1ms
    }
}

```

联系信息

Fremont Micro Devices Corporation

#5-8, 10/F, Changhong Building
Ke-Ji Nan 12 Road, Nanshan District,
Shenzhen, Guangdong, PRC 518057

Tel: (+86 755) 8611 7811

Fax: (+86 755) 8611 7810

Fremont Micro Devices (HK) Limited

#16, 16/F, Block B, Veristrong Industrial Centre,
34-36 Au Pui Wan Street, Fotan, Shatin, Hong Kong SAR

Tel: (+852) 2781 1186

Fax: (+852) 2781 1144

<http://www.fremontmicro.com>

* Information furnished is believed to be accurate and reliable. However, Fremont Micro Devices Corporation assumes no responsibility for the consequences of use of such information or for any infringement of patents of other rights of third parties, which may result from its use. No license is granted by implication or otherwise under any patent rights of Fremont Micro Devices Corporation. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. Fremont Micro Devices Corporation products are not authorized for use as critical components in life support devices or systems without express written approval of Fremont Micro Devices Corporation. The FMD logo is a registered trademark of Fremont Micro Devices Corporation. All other names are the property of their respective owners.