

FT62F08X

WDT Application note

目录

1. 看门狗定时器	3
1.1. 看门狗时钟源.....	3
1.2. WDT 相关寄存器汇总.....	4
2. 应用范例.....	5
联系信息	9

FT62F08x WDT 应用

1. 看门狗定时器

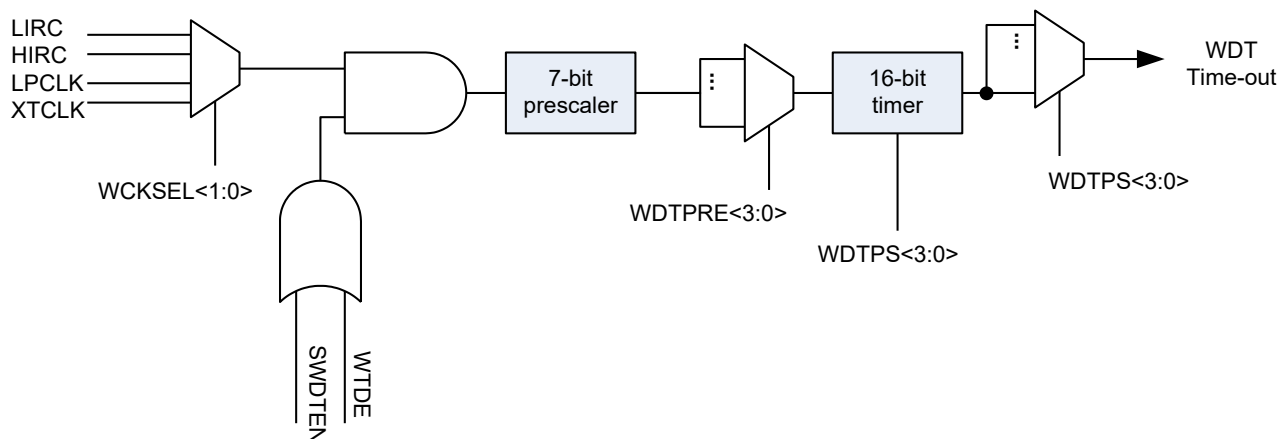


图 1-1 看门狗结构框图

看门狗是一个带 7 位预分频的 16 位计数器，其中预分频和周期可编程，分别由 WDTPRE 和 WDTPS 设置。

WDT 的硬件使能位位于配置寄存器 UCFG0 的第 3 位，WDTEN，软件使能位位于 WDTCON 寄存器的第 0 位，为 1 时表示使能看门狗，为 0 时禁止。

指令 CLRWDT、SLEEP 会清除看门狗计数器。

在使能了看门狗的情况下，处于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 WDT 则是作为一个复位源。

条件	看门狗状态
WDTEN 和 SWDTEN 同时为 0	清零
CLRWDT 指令	
进入 SLEEP、退出 SLEEP 时刻	
写 WDTCON	
写 WCKSEL	

注意：

- 如果内部慢时钟从 32k 切换到 256k 模式 (或反之从 256k 切换到 32k 模式，由 LFMOD 位控制)，都不影响看门狗计时。因为 WDT 时钟源选择 LIRC 时，固定使用 32k 时钟源，见[错误!未找到引用源。](#)的时钟框图；
- PWRT 和 OST 复用了 WDT 定时器，故 PWRT 或 OST 工作时，看门狗的复位功能是暂时屏蔽的；

1.1. 看门狗时钟源

WDT 有 4 种时钟源可选，由寄存器 MISC0 的 WCKSEL 位设置。在 WDT 使能的情况下，所选择的时钟源被自动使能，并在 SLEEP 模式下保持。

1.2. WDT 相关寄存器汇总

名称	状态	寄存器	地址	复位值
WCKSEL	<u>WDT 时钟源</u> 00 = LIRC 01 = HIRC 10 = LP (仅当 FOSC 为 LP 或 INTOSCIO 模式*) 11 = XT (仅当 FOSC 为 XT 或 INTOSCIO 模式*) *否则配置错误, 无 WDT 时钟源	MISC0[1:0]	0x11C	RW-00
WDTPRE	<u>WDT 预分频器</u> 000 = 1 100 = 16 001 = 2 101 = 32 010 = 4 110 = 64 011 = 8 111 = 128 (默认)	WDTCN[7:5]	0x97	RW-111
WDTPS	<u>WDT 周期</u> 0000 = 32 0111 = 4,096 0001 = 64 1000 = 8,192 0010 = 128 1001 = 16,384 0011 = 256 1010 = 32,768 0100 = 512 (默认) 1011 = 65,536 0101 = 1,024 11xx = 65,536 0110 = 2,048	WDTCN[4:1]		RW-0100
SWDTEN	1 = WDT 使能 0 = <u>WDT 关闭</u> (当 WDTE 选择由 SWDTEN 控制时)	WDTCN[0]		RW-0

表 1-1 WDT 相关用户寄存器

名称	功能	默认
WDTE	<u>WDT</u> <ul style="list-style-type: none"> 使能 (指令不能禁止) <u>通过指令控制 (SWDTEN)</u> 	SWDTEN 控制

表 1-2 WDT 选项初始化配置寄存器

2. 应用范例

```
//*****
/* 文件名: TEST_62F08x_WDT.c
* 功能:    FT62F08x-WDT 功能演示
* IC:      FT62F088 LQFP32
* 内部:    16M/2T
* empno: 500
* 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
*
* 参考原理图 TEST_62F08x_sch.pdf
*/
//*****
#include "SYSCFG.h"
//*****宏定义*****
#define unchar unsigned char
#define uint unsigned int
#define unlong unsigned long

#define DemoPortOut PB3
/*-----
* 函数名: interrupt ISR
* 功能:   中断处理, 包括定时器 0 中断和外部中断
* 输入:   无
* 输出:   无
-----*/
void interrupt ISR(void)
{
}
/*-----
* 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 = 0B01110001; // IRCF=111=16MHz
    INTCON = 0;          //暂禁止所有中断
```

```

PORTA = 0B00000000;
TRISA = 0B00000000;      //PA 输入输出 0-输出 1-输入
PORTB = 0B00000000;
TRISB = 0B00000000;      //PB 输入输出 0-输出 1-输入  PB3-OUT
PORTC = 0B00000000;
TRISC = 0B00000000;      //PC 输入输出 0-输出 1-输入
PORTD = 0B00000000;
TRISD = 0B00000000;      //PD 输入输出 0-输出 1-输入

WPUA = 0B00000000;      //PA 端口上拉控制 1-开上拉 0-关上拉
WPUB = 0B00000000;      //PB 端口上拉控制 1-开上拉 0-关上拉
WPUC = 0B00000000;      //PC 端口上拉控制 1-开上拉 0-关上拉
WPUD = 0B00000000;      //PD 端口上拉控制 1-开上拉 0-关上拉

WPDA = 0B00000000;      //PA 端口下拉控制 1-开下拉 0-关下拉
WPDB = 0B00000000;      //PB 端口下拉控制 1-开下拉 0-关下拉
WPDC = 0B00000000;      //PC 端口下拉控制 1-开下拉 0-关下拉
WPDD = 0B00000000;      //PD 端口下拉控制 1-开下拉 0-关下拉

PSRC0 = 0B11111111;      //PORTA,PORTB 源电流设置最大
PSRC1 = 0B11111111;      //PORTC,PORTD 源电流设置最大

PSINK0 = 0B11111111;     //PORTA 灌电流设置最大 0:最小, 1:最大
PSINK1 = 0B11111111;     //PORTB 灌电流设置最大 0:最小, 1:最大
PSINK2 = 0B11111111;     //PORTC 灌电流设置最大 0:最小, 1:最大
PSINK3 = 0B11111111;     //PORTD 灌电流设置最大 0:最小, 1:最大

ANSELA = 0B00000000;     //全为数字管脚
}
/*-----
* 函数名: DelayUs
* 功能:   短延时函数 --16M-2T--大概快 1%左右.
* 输入:   Time 延时时间长度 延时时长 Time μs
* 输出:   无
-----*/
void DelayUs(unsigned char Time)
{
    unsigned char a;
    for(a=0;a<Time;a++)
    {
        NOP();
    }
}
/*-----

```

```

* 函数名: DelayMs
* 功能:   短延时函数
* 输入:   Time 延时时间长度 延时时长 Time ms
* 输出:   无

```

```

-----*/
void DelayMs(unsigned char Time)
{
    unsigned char a,b;
    for(a=0;a<Time;a++)
    {
        for(b=0;b<5;b++)
        {
            DelayUs(197);    //快 1%
        }
    }
}
/*-----

```

```

* 函数名: WDT_INITIAL
* 功能:   初始化设置看门狗
* 设置看门狗复位时间=系统时钟周期*预分频值*定时周期
*
    =1/32000*1*1024=23ms

```

```

-----*/
void WDT_INITIAL (void)
{
    CLRWDT();                //清看门狗
    MISC0 = 0B00000000;      //看门狗时钟 32k
    WDTCON = 0B00001011;     //WDTPS=1010=1:1024,预分频 1:1
}
/*-----

```

```

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

```

```

-----*/
void main(void)
{
    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 or 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.