

# **FT62F21X**

## **Application note**

## 目录

1	TIMER2 相关寄存器的设置 .....	3
2	定时时间长度设置 .....	5
3	应用范例 .....	5

## FT60F01x Timer2 的应用

### 1 Timer2 相关寄存器的设置

定时器 2 为 16 位，其时钟源可选系统时钟或者内部 32MHz 时钟或 LIRC，可以作为计数器和定时器使用，当 TMR2 值等于 PR2 时会产生中断，Timer2 具有预分频器和后分频器，预分频比为 1: 1、1: 4 和 1: 16，后分频比为 1: 1~1: 16。

相关寄存器的各个位定义如下：

#### 1) T2CON0 寄存器

Bit	7	6	5	4	3	2	1	0
Name	PR2U	TOUTPS[3-0]				TMR2ON	T2CKPS[1:0]	
Reset	NA	0000				0	00	

Bit7: PR2、P1xDTy 寄存器的软件更新控制位，只写

写 1: 把 PR2/P1xDTy 缓冲值分别更新到 PR2 寄存器和 P1xDTy\_ACT

写 0: 无意义

Bit6~Bit3: 定时器 2 输出后分频比选择

TOUTPS<3:0>	后分频比	TOUTPS<3:0>	后分频比
0000	1:1	1000	1:9
0001	1:2	1001	1:10
0010	1:3	1010	1:11
0011	1:4	1011	1:12
0100	1:5	1100	1:13
0101	1:6	1101	1:14
0110	1:7	1110	1:15
0111	1:8	1111	1:16

Bit2: 打开定时器 2 位

1: 打开 Timer2

0: 关闭 Timer2

Bit1~Bit0: 定时器 2 驱动时钟预分频比选择

00 = 预分频比是 1: 1

01 = 预分频比是 1: 4

1x = 预分频比是 1: 16

#### 2) TMR2L 寄存器

Bit	7	6	5	4	3	2	1	0
Name	TMR2L[7:0]							
Reset	0000 0000							

Bit7~Bit0: Timer 2 计数结果寄存器 低 8 位 (TMR2[7:0])

#### 3) TMR2H 寄存器

Bit	7	6	5	4	3	2	1	0
Name	TMR2H[7:0]							
Reset	0000 0000							

Bit7~Bit0: Timer 2 计数结果寄存器 高 8 位 (TMR2[15:8])

## 4) PR2寄存器

Bit	7	6	5	4	3	2	1	0
Name	PR2[7:0]							
Reset	1	1	1	1	1	1	1	1
Type	RW	RW	RW	RW	RW	RW	RW	RW

Bit7~Bit0: Timer2比较寄存器

## 5) INTCON 寄存器

Bit	7	6	5	4	3	2	1	0
Name	GIE	PEIE	TOIE	INTE	PAIE	TOIF	INTF	PAIF
Reset	0	0	0	0	0	0	0	0
Type	RW	RW	RW	RW	RW	RW	RW	RW

Bit7: 全局中断使能

- 1: 使能所有未屏蔽中断
- 0: 禁止所有中断

Bit6: 外设中断使能

- 1: 使能所有未屏蔽中断
- 0: 禁止所有外设中断

Bit5: 定时器 0 溢出中断使能

- 1: 使能定时器 0 中断
- 0: 禁止定时器 0 中断

Bit4: 外部中断使能

- 1: 使能 PA2/INT 管脚外部中断
- 0: 禁止 PA2/INT 管脚外部中断

Bit3: PORTA 端口变化中断

- 1: 使能 PORTA 端口变化中断
- 0: 禁止 PORTA 端口变化中断

Bit2: 定时器 0 溢出中断标志位

- 1: Timer0 寄存器溢出 (必须软件清零)
- 0: Timer0 寄存器未溢出

Bit1: PA2/INT 管脚外部中断标志位

- 1: PA2/INT 管脚外部中断已发生 (必须软件清零)
- 0: PA2/INT 管脚外部中断未发生

Bit0: PORTA 端口变化中断标志位

- 1: PORTA<5:0>至少有一个端口状态发生了改变 (必须软件清零)
- 0: PORTA<5:0>没有一个端口发生状态改变

## 6) PIE1 寄存器

Bit	7	6	5	4	3	2	1	0
Name	EEIE	CKMIE	LVDIE	-	-	OSFIE	TMR2IE	-
Reset	0	0	0	-	-	0	0	-

Bit7: EE 写中断使能位

- 1: 使能 EE 写操作完成中断
- 0: 关闭 EE 写操作完成中断

Bit6: 快时钟测量慢时钟操作完成中断使能位

- 1: 使能快时钟测量慢时钟操作完成中断

0: 关闭快时钟测量慢时钟操作完成中断

Bit5: 按键中断使能位

1 = 使能按键中断

0 = 禁止按键中断

Bit4~Bit3: 保留位

Bit2: 振荡器故障中断允许位

1 = 允许振荡器故障中断

0 = 禁止振荡器故障中断

Bit1: Timer2 与 PR2 比较相等中断使能位

1: 使能 timer2 的值等于 PR2 中断

0: 关闭使能 timer2 的值等于 PR2 中断

Bit0: 保留位

### 7) PIR1

Bit	7	6	5	4	3	2	1	0
Name	EEIF	CKMIF	LVDIF	-	-	-	TMR2IF	-
Reset	0	0	0	-	-	-	0	-

Bit7: EE 写中断标志位

1: EE 写操作完成 (必须软件清零)

0: EE 写操作未完成

Bit6: 快时钟测量慢时钟操作完成中断标志位

1: 快时钟测量慢时钟操作完成 (必须软件清零)

0: 快时钟测量慢时钟未完成

Bit5: LVD 中断标志位

1 = LVD 检测电压低于所设置阈值

0 = LVD 检测电压高于所设置阈值, 或已经由软件清 0

Bit4~Bit2: 保留位

Bit1: Timer2 与 PR2 比较相等中断标志位

1: timer2 的值等于 PR2 (必须软件清零)

0: timer2 的值不等于 PR2

Bit0: 保留位

## 2 定时时间长度设置

在系统时钟和 4T 模式下, 定时时长计算公式如下:

$$\text{定时时长} = \frac{1}{\text{系统时钟频率}} * 4 * \text{预分频值} * \text{后分频值} * PR2$$

## 3 应用范例

```

//*****
/* 文件名: Test_62F21X_Timer2.c
* 功能: FT62F21X_Timer2 功能演示
* IC: FT62F21X SOP8
* 晶振: 16M/4T

```

```

* 说明: 当 DemoPortIn 悬空或者高电平时,
*       DemoPortOut 输出 2.5KHz 占空比 50%的波形-Tm2 实现
*       当 DemoPortIn 接地时,DemoPortOut 输出高电平.关定时器
* Memory: Flash 1KX14b, EEPROM 128X8b, SRAM 64X8b
*       FT62F21X SOP8
*
*       -----
* DemoPortOut ----- |1(PA4)           (PA3)8 |-----NC
* NC-----|2(TKCAP)       (PA0)7 |-----NC
* NC-----|3(VDD)         (PA1)6 |-----NC
* NC-----|4(VSS)         (PA2)5 |-----DemoPortIn
*
*       -----
*/
//=====
//=====
#include "SYSCFG.h";
#include "FT62F21X.h";

#define unchar    unsigned char
#define uint      unsigned int
#define ulong     unsigned long

//=====
//
// 系统时钟
//=====
#define OSC_16M   0X70
#define OSC_8M    0X60
#define OSC_4M    0X50
#define OSC_2M    0X40
#define OSC_1M    0X30
#define OSC_500K  0X20
#define OSC_250K  0X10
#define OSC_32K   0X00

#define DemoPortOut RA4
#define DemoPortIn  RA2
//=====
//变量定义
//=====

//=====
//Funtion name: interrupt ISR
//parameters: 无
//returned value: 无

```

```
//=====
void interrupt ISR(void)
{
    //定时器 2 的中断处理*****
    if(TMR2IE && TMR2IF)                //200us 中断一次 = 2.5KHz
    {
        DemoPortOut = ~DemoPortOut;    //翻转电平
        TMR2IF = 0;
    }
}

/*-----
* 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 = OSC_16M;                  //bit7 Timer2 选择 LIRC 为时钟源时 LIRC 的频率选择
                                        //0:32KHz    1:256KHz
                                        //bit[6:4] 系统频率选择
                                        //bit[2]高速内部时钟状态    1:ready    0:not ready
                                        //bit[1]低速内部时钟状态    1:ready    0:not ready

    INTCON = 0;                        //暂禁止所有中断
    OPTION = 0;
    TRISA  = 1<<2;                    //1:输入 0:输出
    PSRCA  = 0;                        //00: 4mA 01/10:8mA    11:28mA
                                        //bit[3:2]控制 PA5 源电流 bit[1:0]控制 PA4 源电流
    PSINKA = 0;                        //bit[1:0] 控制 PA5 和 PA4    0:灌电流最小 1 灌电流最大
    PORTA  = 0;                        //1:PAx 输出高电平    0:PAx 输出低电平
    WPUA   = 1<<2;                    //1: 使能 PA 口上拉    0:关闭 PA 口上拉
}
/*-----
* 函数名称:   TIMER2_INITIAL
* 功能:       初始化设置定时器 1
* 相关寄存器: T2CON TMR2 PR2 TMR2IE TMR2IF PEIE GIE
-----*/
void TIMER2_INITIAL (void)
{
    T2CON = 0B00000001;                //Bit[1,0]=01,T2 时钟分频 1:4
                                        //Bit[6:3]=0000,T2 输出时钟分频 1:1
    T2CON1 = 0B00001000;                //Bit[2:0] 000: 指令周期 100:HIRC    Timer2 时钟源选择
}
```

```
TMR2H = 0;
TMR2L = 0;           //TMR2 赋初值
PR2H = 0;
PR2L = 200;          // 设置 TMR2 输出比较值定时
                      //200us=(1/16000000)*4*4*200(PR2)
                      //16M-4T-4 分频

TMR2IF = 0;          //清 TIMER2 中断标志
TMR2IE = 1;          //使能 TIMER2 的中断
TMR2ON = 1;          //使能 TIMER2 启动
PEIE=1;              //使能外设中断
GIE = 1;              //使能全局中断
}
/*-----
* 函数名: main
* 功能:   主函数
* 输入:   无
* 输出:   无
-----*/
void main()
{
    POWER_INITIAL();  //系统初始化

    TIMER2_INITIAL(); //初始化 T2

    while(1)
    {
        if(DemoPortIn == 1) //判断输入是否为高电平
        {
            TMR2IE = 1;      //开定时器 2
        }
        else
        {
            TMR2IE = 0;      //关定时器 2
            DemoPortOut = 1;
        }
    }
}
//=====
```



Fremont Micro Devices (SZ) Limited

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

Tel: (86 755) 86117811

Fax: (86 755) 86117810

Fremont Micro Devices (Hong Kong) Limited

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

Tel: (852) 27811186

Fax: (852) 27811144

Fremont Micro Devices (USA), Inc.

42982 Osgood Road Fremont, CA 94539

Tel: (1-510) 668-1321

Fax: (1-510) 226-9918

Web Site: <http://www.fremontmicro.com/>

\* Information furnished is believed to be accurate and reliable. However, Fremont Micro Devices, Incorporated (BVI) 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, Incorporated (BVI). Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. Fremont Micro Devices, Incorporated (BVI) products are not authorized for use as critical components in life support devices or systems without express written approval of Fremont Micro Devices, Incorporated (BVI). The FMD logo is a registered trademark of Fremont Micro Devices, Incorporated (BVI). All other names are the property of their respective own.