

FT62F21X

Application note

目录

1	PORTA IO 相关寄存器的设置	3
2	应用范例	4

FT62F21x IO 应用

1 IO 相关寄存器的设置

本芯片共包含 6 个 GPIO。这些 IO 除了作为普通输入/输出端口以外还通常具备一些与内核周边电路通讯的功能。PORTA 是一个 6 位双向端口。与其相应的进出方向寄存器就是 TRISA 寄存器。

相关寄存器的各个位定义如下：

1) PORTA 寄存器

Bit	7	6	5	4	3	2	1	0
Name	-	-	PA5	PA4	PA3	PA2	PA1	PA0
Reset	-	-	x	x	x	x	x	x

Bit7~Bit6: -

Bit5: PORTA5 数据

Bit4: PORTA4 数据

Bit3: PORTA3 数据

Bit2: PORTA2 数据

Bit1: PORTA1 数据

Bit0: PORTA0 数据

2) TRISA 寄存器

Bit	7	6	5	4	3	2	1	0
Name	-	-	TRISA[5]	TRISA[4]	TRISA[3]	TRISA[2]	TRISA[1]	TRISA[0]
Reset	-	-	1	1	1	1	1	1

Bit7~ Bit6: N/A,读 0

Bit5~ Bit0: PORTA<5:0>输入/输出状态控制寄存器

1: 端口为输入状态

0: 端口为输出状态

3) WPUA 寄存器

Bit	7	6	5	4	3	2	1	0
Name	-	-	WPUA5	WPUA4	WPUA3	WPUA2	WPUA1	WPUA0
Reset	-	-	1	1	-	1	1	1

Bit7~Bit6、Bit3: N/A,读 0

Bit5~Bit0: Port A 弱上拉使能

1: 使能PORTA 端口弱上拉

0: 关闭PORTA 端口弱上拉

4) 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

2 应用范例

//*****

/* 文件名: Test_62F21X_io.c

* 功能: FT62F21X-IO 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 50Hz 占空比 50%的波形

* 当 DemoPortIn 接地时,DemoPortOut 输出高电平

* 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 DemoPortOut RA4
#define DemoPortIn RA2

//=====
//
// 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

//=====
//变量定义
//=====

/*-----
* 函数名: nterrupt ISR
* 功能: 中断服务函数
* 输入: 无
* 输出: 无
-----*/
void interrupt ISR(void)
{

}

/*-----
* 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: 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    PA2 为输入模式 PA4 输出模式
    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 PA4 输出低电平
    WPUA   = 1<<2;             //1: 使能 PA 口上拉  0:关闭 PA 口上拉    PA2 上拉
}
```

/*-----

* 函数名: main

* 功能: 主函数

* 输入: 无

* 输出: 无

-----*/

void main()

```
{
    POWER_INITIAL();           //系统初始化

    while(1)
    {
        DemoPortOut = 1;
        DelayMs(10);           //10ms
        if(DemoPortIn == 1)    //判断输入是否为高电平
        {
            DemoPortOut = 0;
        }
        DelayMs(10);
    }
}
```

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.