

# **FT61F13X**

## **IO Application note**

## 目录

1. I/O 端口 .....	3
1.1. I/O 端口相关寄存器汇总 .....	5
1.2. I/O 配置 .....	6
2. 应用范例 .....	8
联系信息 .....	11

## FT61F13x IO 应用

### 1. I/O 端口

根据封装类型的不同, FT61F13x 系列芯片最多有 18 个 I/O 引脚可用, 共分为 3 组: PORTA (8)、PORTB (8) 和 PORTC (2)。表 1-1 列出了所有 I/O 引脚的功能。

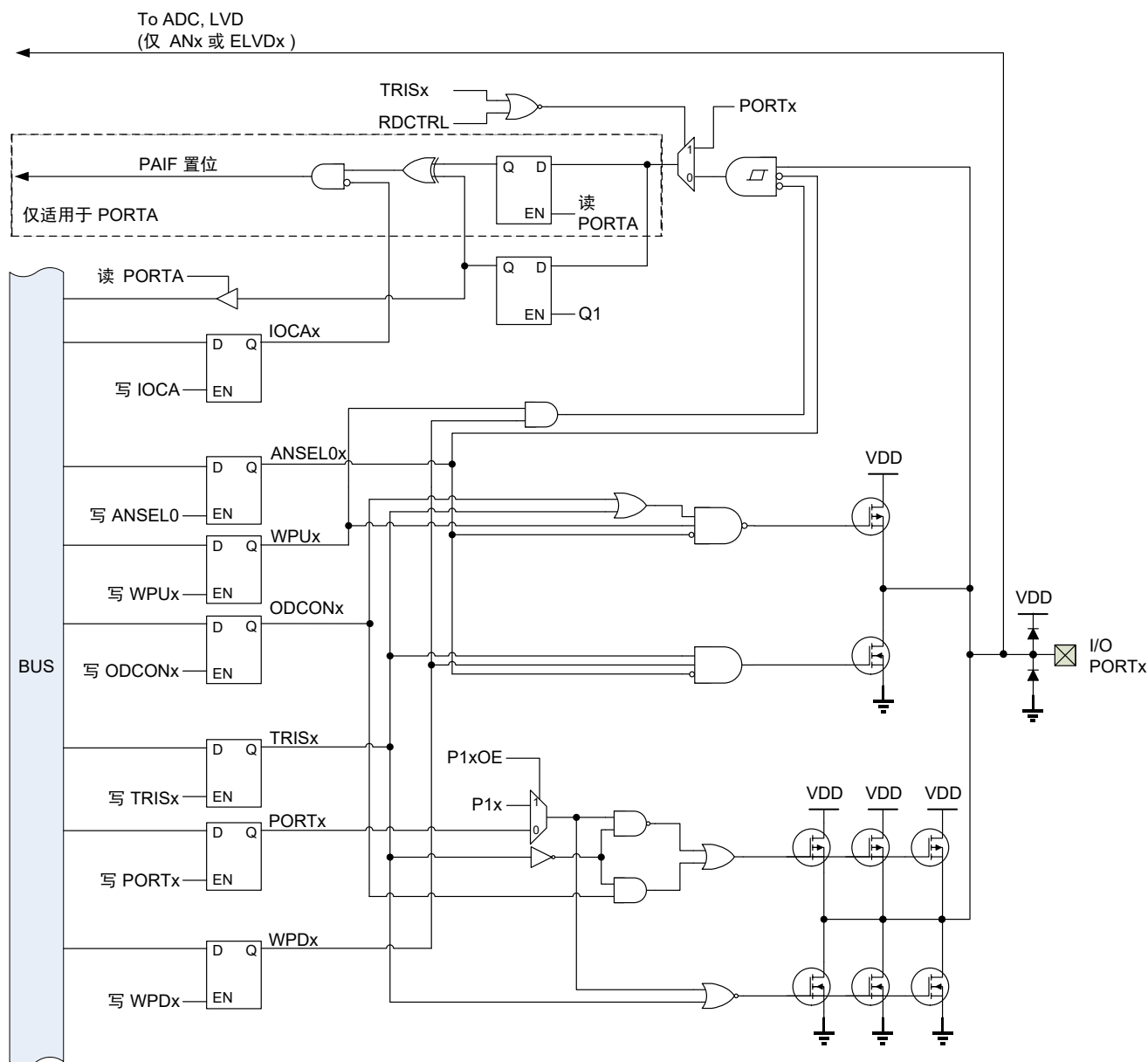


图 1-1 PORT 端口结构框图

所有 I/O 引脚均具有以下功能 (表 1-3, 表 1-4):

- 数字输出
- 数字输入
- 开漏
- 弱上拉
- 弱下拉

此外，部分 I/O 具有以下特殊功能：

1. 烧录调试引脚 (ISP-Data, ISP-CLK, ISP-Data1, ISP-CLK1)，硬件内部连接，不需设置。
2. 通过 IDE 界面配置，且在芯片初始化配置时加载的功能 (表 1-2)：
  - 外部时钟/晶振输入 (OSC1, OSC2)
  - 系统外部复位 (/MCLR B)
  - 内部时钟输出
3. 通过指令对相应 I/O 引脚进行配置的其他功能，可分为 3 类：
  - a. 数字输出
    - PWM
  - b. 数字输入
    - PWM 故障刹车
    - 外部边沿中断
    - GPIO 端口变化中断
    - ADC 触发 (ADC\_ETR)
    - Timer0 时钟输入
  - c. 模拟输入
    - LVD / BOR
    - ADC
    - $V_{REF+}$
    - $V_{REF-}$

引脚名	ISP 调试	时钟	ADC	中断	LVD	PWM	数字 I/O 上拉/下拉 开漏	源电流 (mA)	灌电流 (mA)
PA0			AN0	√	ELVD3	PWM 1N	√	24	55
PA1			AN1	√	ELVD2	PWM 1	√	24	55
PA2		输出	AN2	√	ELVD4	PWM 4	√	18	53
PA3				√		PWM 4	√	24	55
PA4			AN3 ( $V_{REF+}$ )	√ + INT		(BKIN)	√	24	55
PA5			( $V_{REF-}$ )	√			√	24	55
PA6	CLK1		AN4	√		PWM 1	√	24	55
PA7	DATA1		AN5	√		PWM 2	√	24	55
PB0		输出	AN6			PWM 3	√	24	55
PB1						PWM 1N	√	24	55
PB2						PWM 3	√	3, 6, 18	35, 53
PB3						PWM 4	√	3, 6, 18	35, 53
PB4							√	3, 6, 18	35, 53
PB5			AN7		ELVD1	PWM 1	√	3, 6, 18	35, 53
PB6					ELVD0	PWM 1N	√	3, 6, 18	35, 53
PB7				/MCLR B		PWM 2	√	3, 6, 18	35, 53
PC0	CLK	OSC-					√	3, 6, 18	35, 53
PC1	DATA	OSC+					√	3, 6, 18	35, 53
注		T0CKI = PA2	Trigger = PA4					$V_{DD}=5, V_{DS}=0.5$	

表 1-1 I/O 端口功能

注： PC0-1 和 PB2-7 支持 3 档可配置源电流驱动能力 (参阅 “PSRCBx” 和 “PSRCC”，表 1-4)，和 2 档可配置灌电流驱动能力 (参阅 “PSINKx”，表 1-4)。

## 1.1. I/O 端口相关寄存器汇总

名称	功能	默认
RDCTRL	当 TRISx = 0 (输出使能) 时, 读 PORTx 寄存器的返回值 • 输入锁存器 • 输出锁存器	输出锁存器
MCLRE	外部 I/O 复位	关闭
FOSC	• LP: PC1 (+) 和 PC0 (-) 接外部低速晶振 • XT: PC1 (+) 和 PC0 (-) 接外部高速晶振 • EC: PC1 (+) 接外部时钟输入, PC0 为 I/O • INTOSC: PB0 或 PA2 输出“指令时钟”, PC0 和 PC1 为 I/O • INTOSCIO: PC0 和 PC1 为 I/O	INTOSCIO

表 1-2 I/O 相关初始化配置寄存器

名称	地址	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位值
ANSEL0	0x11E	ANSEL0[7:0]								0000 0000
TRISA	0x85	TRISA[7:0], PORTA 方向控制								1111 1111
TRISB	0x86	TRISB[7:0], PORTB 方向控制								1111 1111
TRISC	0x87	-						PORTC 方向控制		---- --11
PORTA	0x05	PORTA 输出寄存器								xxxx xxxx
PORTB	0x06	PORTB 输出寄存器								xxxx xxxx
PORTC	0x07	-						PORTC 输出寄存器		---- --xx
WPUA	0x95	PORTA 弱上拉								1111 1111
WPUB	0x10D	PORTB 弱上拉								0000 0000
WPUC	0x93	-						PORTC 弱上拉		---- --00
WPDA	0x89	PORTA 弱下拉								0000 0000
WPDB	0x10E	PORTB 弱下拉								0000 0000
WPDC	0x8D	-						PORTC 弱下拉		---- --00
ODCONA	0x105	PORTA 开漏								0000 0000
ODCONB	0x106	PORTB 开漏								0000 0000
ODCONC	0x107	-						PORTC 开漏		---- --00
PSRCB1	0x88	PB5, PB4, PB3, PB2 源电流设置								1111 1111
PSRCB2	0x10C	-				PB7, PB6 源电流设置				---- 1111
PSRCC	0x94	-				PC1, PC0 源电流设置				---- 1111
PSINKB	0x10F	PORTB 灌电流设置						-		0000 0000
PSINKC	0x9F	-						PORTC 灌电流设置		---- --00
IOCA	0x96	IOCA[7:0]: PORTA 端口变化中断设置								0000 0000
OPTION	0x81	/PAPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111

表 1-3 I/O 相关用户寄存器的地址和复位值

名称	状态		寄存器	地址	复位值
TRISA	PORTA	PORT端口数字输出 (方向控制) 1 = 关闭 0 = 使能 (关闭上拉/下拉)	TRISA[7:0]	0x85	RW-1111 1111
TRISB	PORTB		TRISB[7:0]	0x86	RW-1111 1111
TRISC	PORTC		TRISC[1:0]	0x87	RW-11
ANSEL0	1 = 关闭上拉/下拉, 及数字输入 (仅适用于8个ADC通道) 0 = (无动作)		ANSEL0[7:0]	0x11E	RW-0000 0000
/PAPU	1 = 关闭所有 PORTA 上拉功能 0 = 上拉由 WPUA 控制		OPTION[7]	0x81	RW-1
WPUA	PORTA	弱上拉 1 = 使能 (PORTA 默认值) 0 = 关闭 (PORTB, C 默认值)	WPUA[7:0]	0x95	RW-1111 1111
WPUB	PORTB		WPUB[7:0]	0x10D	RW-0000 0000
WPUC	PORTC		WPUC[1:0]	0x93	RW-00
WPDA	PORTA	弱下拉 1 = 使能 0 = 关闭	WPDA[7:0]	0x89	RW-0000 0000
WPDB	PORTB		WPDB[7:0]	0x10E	RW-0000 0000
WPDC	PORTC		WPDC[1:0]	0x8D	RW-00
ODCONA	PORTA	开漏 1 = 使能 0 = 关闭	ODCONA[7:0]	0x105	RW-0000 0000
ODCONB	PORTB		ODCONB[7:0]	0x106	RW-0000 0000
ODCONC	PORTC		ODCONC[1:0]	0x107	RW-00
PORTA	PORTA	数据输出寄存器	PORTA[7:0]	0x05	RW-xxxx xxxx
PORTB	PORTB		PORTB[7:0]	0x06	RW-xxxx xxxx
PORTC	PORTC		PORTC[1:0]	0x07	RW-xx
PSINKB	PB7-PB2	灌电流 (mA) 1 = 53                      0 = <u>35</u>	PSINKB[7:2]	0x10F	RW-0000 00
PSINKC	PC1, PC0		PSINKC[1:0]	0x9F	RW-00
PSRCB1[7:6]	PB5	源电流 (mA) (00) = 3 (01) = 6 / (10) = 6 (11) = <u>18</u>	PSRCB1[7:6]	0x88	RW-11
PSRCB1[5:4]	PB4		PSRCB1[5:4]	0x88	RW-11
PSRCB1[3:2]	PB3		PSRCB1[3:2]	0x88	RW-11
PSRCB1[1:0]	PB2		PSRCB1[1:0]	0x88	RW-11
PSRCB2[3:2]	PB7		PSRCB2[3:2]	0x10C	RW-11
PSRCB2[1:0]	PB6		PSRCB2[1:0]	0x10C	RW-11
PSRCC[3:2]	PC1		PSRCC[3:2]	0x94	RW-11
PSRCC[1:0]	PC0		PSRCC[1:0]	0x94	RW-11

表 1-4 I/O 相关用户寄存器

## 1.2. I/O 配置

每个 PORT 端口, 均需根据其相应功能配置以下 4 个模块(表 1-5):

- 弱上拉
- 弱下拉
- 数字输入
- 数字输出

功能	数字输入	上拉/下拉	数字输出	设置
ISP-DATA	On	Off	On	(硬件内置, 忽略指令)
ISP-CLK	On	Off	Off	(硬件内置, 忽略指令)
/MCLRB	On	上拉	Off	(初始化配置, 忽略指令)
时钟输出	(忽略)	Off	On	(初始化配置, 忽略指令)
OSC+ (EC)	On	(可选)	Off	(初始化配置, 忽略指令)
OSC+ / OSC- (LP, XT)	Off	Off	Off	(初始化配置, 忽略指令)
ADC	Off	Off	Off	TRISx = 1; ANSEL0x = 1
LVD	Off <sup>(5)</sup>	Off	Off	TRISx = 1; ANSEL0x = 1 (除 PB6 外)
V <sub>REF+</sub> / V <sub>REF-</sub>	Off	Off	Off	TRISx = 1
Timer0 时钟	On	(可选)	Off	TRISx = 1
ADC 触发	On	(可选)	Off	TRISx = 1
端口变化中断	On	(可选)	Off	TRISx = 1
PA4-INT	On	(可选)	Off	TRISx = 1
BKIN	On	(可选)	Off	TRISx = 1
数字输入	On	(可选)	Off	TRISx = 1
PWM	On	Off	On	TRISx = 0
数字输出	On	Off	On	TRISx = 0

**表 1-5 I/O 配置标志和用户寄存器**

注:

1. TRISx = 0: “数字输出” 使能, “上拉/下拉” 自动关闭 (忽略 WPDx, WPUx)。
2. TRISx = 1: “数字输出” 关闭。
3. ANSEL0x = 1: “上拉”、“下拉”、“数字输入” 自动关闭 (忽略 WPDx, WPUx)。
4. 可关闭 “数字输入” 的唯一指令为 “ANSEL0x = 1”。
5. 将 PORT 端口设置为 LVD 输入时, 其“数字输入”、“上拉”和“下拉”功能被自动关闭。当 LVD 输入需要在不同的通道之间切换使用时, 通过设置 “ANSEL0x = 1” 可关闭当前未被选择通道的“数字输入”, 但 PB6 无 ANSEL0x 控制, 无法关闭“数字输入”, 因此不应仅在部分时间作为 LVD 输入。
6. “/PAPU = 1” 关闭所有 PAx 端口的 “弱上拉” 功能。PBx 和 PCx 没有此类控制位。
7. /MCLR 使能: PB7 的弱上拉功能自动使能 (忽略 WPUB[7]); 读 PORTB[7] 的值为 “0”。
8. 对 PORTx 数据输出寄存器进行写操作, I/O 端口将输出相应的逻辑电平。每组多达 8 个 I/O 的数据寄存器共用相同的地址, 写操作实际执行 ‘读-修改-写’ 的过程, 即先读取该组 PORTx 端口锁存器值 (输出或输入), 然后修改, 再写回 PORTx 数据寄存器。
9. 数字输出和数字输入功能可以共存, 有些应用需要同时使能数字输出和数字输入。
10. 当 TRISx = 0 时, 通过 IDE 界面可选择读取 PORTx 输出或输入锁存器的值。

完全复位或系统复位时, PORTx 寄存器不会复位, 但 TRISx 将被重置为 “1”, 从而关闭输出。

## 2. 应用范例

```
// Project: test_61F13x_IO.c
// Device: FT61F13X
// Memory: Flash 3Kx14b, EEPROM 128x8b, SRAM 256x8b
// Company:
// Version:
/* 文件名: test_61F13x_IO.c
 * 功能:    FT61F13x-IO 功能演示
 * IC:      FT61F135 SOP20
 * 晶振:    16M/2T
 * 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
 * 注意:    因为 PA6 是烧录口的 CLK, 当用 PA6 做烧录口仿真时, 此程序必须断开烧录口
 *          测试验证, 断开后要断电 VCC 复位单片机。
 */
                FT61F135  SOP20
            -----
VDD-----|1(VDD)  (VSS)20|-----VSS
NC-----|2(PC1)   (PA0)19|-----NC
NC-----|3(PC0)   (PA1)18|-----NC
NC-----|4(PB7)   (PA2)17|-----NC
NC-----|5(PB6)   (PA3)16|-----NC
NC-----|6(PB5)   (PA4)15|---DemoPortOut
NC-----|7(PB4)   (PA5)14|-----NC
NC-----|8(PB3)   (PA6)13|----DemoPortIn
NC-----|9(PB2)   (PA7)12|-----NC
NC-----|10(PB1)  (PB0)11|-----NC
            -----
*/

//=====================================================
#include "SYSCFG.h";
//=====================================================
//Variable definition
//=====================================================
#define uchar        unsigned char
#define DemoPortOut  PA4
#define DemoPortIn   PA6
/*-----
 * 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/2T=8MHz, 0.125μs
    OPTION = 0B00001000;    //Bit3=1 WDT, Bit[2:0]=000=WDT RATE 1:1
    INTCON = 0;              //暂禁止所有中断

    PORTA = 0B00000000;
    TRISA = 0B01000000;     //PA 输入输出 0-输出 1-输入
                             //PA4-OUT PA6-IN

    PORTC = 0B00000000;
    TRISC = 0B00000000;     //PC 输入输出 0-输出 1-输入

    WPUA = 0B01000000;     //PA 端口上拉控制 1-开上拉 0-关上拉
                             //开 PA6 上拉
    WPUC = 0B00000000;     //PC 端口上拉控制 1-开上拉 0-关上拉
}

```

```

/*-----
* 函数名: DelayUs
* 功能:   短延时函数
* 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--16M-2T--大概快 1%左右
* 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:  main  
* 功能:   主函数  
* 输入:   无  
* 输出:   无  
-----*/  
void main(void)  
{  
    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);  
    }  
}
```

## 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.