

FT6XF0XX

Application note

目录

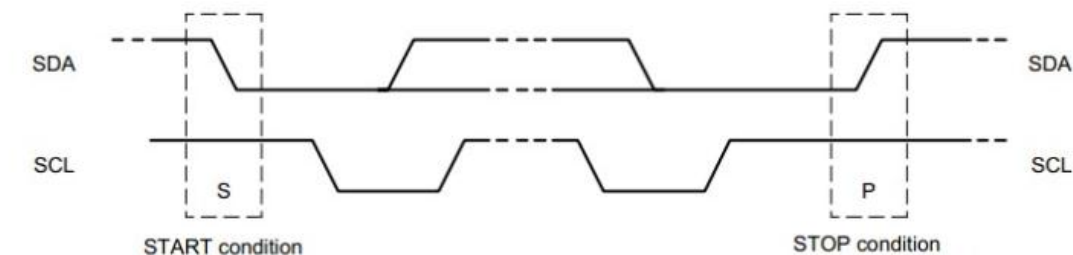
1	IIC 应用说明	3
2	应用范例	3

FT6XF0XX IIC 应用

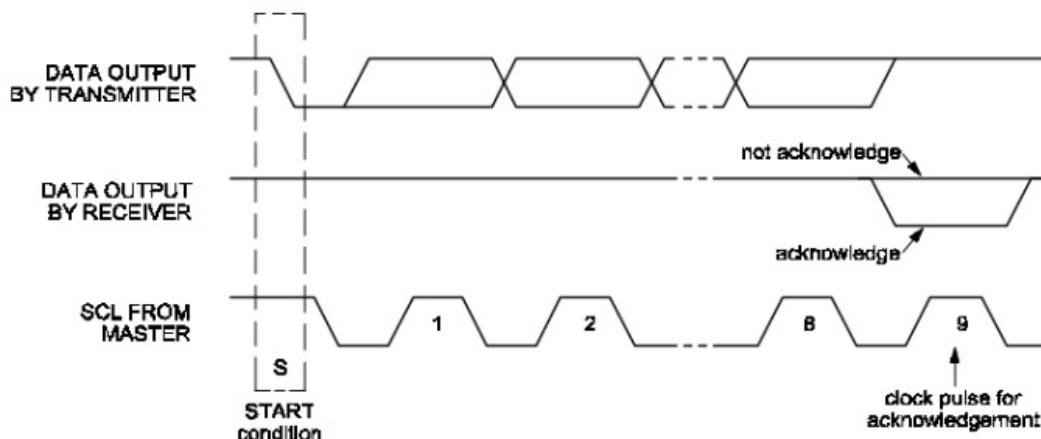
1 IIC 应用说明

I2C总线是一种串行数据总线，共二根信号线：双向的数据线SDA和时钟线SCL。

IIC协议：两条线可以挂多个设备。IIC设备里有个固化的地址，只有在两条线上传输的值等于（IIC设备）的地址时，才作出响应。



起始和停止条件



I²C 总线的响应

本说明以IC FT60F011A与存储芯片24C02为示范。

本程序中数据线SDA与SCL所对应的IO引脚：

```
#define IIC_SCL    RA4
```

```
#define IIC_SDA    RA2
```

2 应用范例

```

/*****
/* 文件名：  TEST_60F01x_IIC.c
/* 功能：    FT60F01x_IIC 功能演示
/* IC：     FT60F011A SOP8，存储芯片 24C02
/* 晶振：   16M/4T
/* 说明：   此演示程序为 60F01x_IIC 的演示程序.

```

* 该程序读取(24C02) 0x12 地址的值,取反后存入 0x13 地址

*

* FT60F011A SOP8

*

* VDD----- |1(VDD) (GND)8|-----GND

* IIC_SDA-----|2(PA2) (PA4)7|-----IIC_SCL

* NC-----|3(PA1) (PA5)6|-----NC

* NC-----|4(PA3) (PA0)5|-----NC

*

*/

//*****

#include "SYSCFG.h"

//*****

#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 WDT_256K 0X80

#define WDT_32K 0X00

//*****

//*****宏定义*****

#define unchar unsigned char

#define uint unsigned int

#define ulong unsigned long

#define IIC_SCL RA4

#define IIC_SDA RA2

#define SDA_OUT TRISA2 =0

#define SDA_IN TRISA2 =1

unchar IICReadData;

/*-----

* 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 = WDT_32K|OSC_16M|0X01; //INROSC
//  OSCCON = 0B01110001;           //WDT 32KHZ IRCF=111=16MHZ/4=4MHZ,0.25US/T
//                                //Bit0=1,系统时钟为内部振荡器(60F01x 保留位)
//                                //Bit0=0,时钟源 FOSC<2: 0>决定即编译选项时选择
//                                //暂禁止所有中断

    INTCON = 0;
    PORTA = 0B00000000;
    TRISA = 0B11101011;           //PA 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
//WPUA2 = 0;                     //关 PA2 上拉

    OPTION = 0B00001000;         //Bit3=1 WDT MODE,PS=000=1:1 WDT RATE
//                                //Bit7(PAPU)=0 ENABLED PULL UP PA

    MSCKCON = 0B00000000;        //Bit6->0,禁止 PA4, PC5 稳压输出(60F01x 保留位)
//                                //Bit5->0,TIMER2 时钟为 Fosc(60F01x 保留位)
//                                //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 版之后)
}
/*-----
* 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*2 Us
* 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: IIC_Start
* 功能:   产生 IIC 起始信号
* 输入:   无
* 输出:   无
```

```
-----*/
```

```
void IIC_Start(void)
```

```
{
    SDA_OUT;          //SDA 线输出
    IIC_SDA=1;
    IIC_SCL=1;
    DelayUs(10);
    IIC_SDA=0;        //START:when CLK is high,DATA change form high to low
    DelayUs(10);
    IIC_SCL=0;        //钳住 I2C 总线，准备发送或接收数据
    DelayUs(10);
}
```

```
}
/*-----
 * 函数名: IIC_Stop
 * 功能:   产生 IIC 停止信号
 * 输入:   无
 * 输出:   无
-----*/
void IIC_Stop(void)
{
    SDA_OUT;                //SDA 线输出
    IIC_SCL=0;
    IIC_SDA=0;              //STOP:when CLK is high DATA change form low to high
    DelayUs(10);
    IIC_SCL=1;
    DelayUs(10);
    IIC_SDA=1;              //发送 I2C 总线结束信号
    DelayUs(10);
}
/*-----
 * 函数名: IIC_Wait_Ack
 * 功能:   等待应答信号到来
 * 输入:   无
 * 输出:   返回值: 1, 接收应答失败
 *           0, 接收应答成功
-----*/
unsigned char IIC_Wait_Ack(void)
{
    unsigned char ucErrTime=0;
    SDA_IN;                //SDA 设置为输入
    IIC_SDA=1;
    DelayUs(5);
    IIC_SCL=1;
    DelayUs(5);
    while(IIC_SDA)
    {
        ucErrTime++;
        if(ucErrTime>250)    //等待超时
        {
            IIC_Stop();
            return 1;
        }
    }
    IIC_SCL=0;              //时钟输出 0
    return 0;
}
```

```
}

/*-----
 * 函数名: IIC_Ack
 * 功能:   产生 ACK 应答
 * 输入:   无
 * 输出:   无
-----*/
void IIC_Ack(void)
{
    IIC_SCL=0;
    SDA_OUT;                //SDA 线输出
    IIC_SDA=0;
    DelayUs(5);
    IIC_SCL=1;
    DelayUs(5);
    IIC_SCL=0;
}

/*-----
 * 函数名: IIC_NAck
 * 功能:   不产生 ACK 应答
 * 输入:   无
 * 输出:   无
-----*/
void IIC_NAck(void)
{
    IIC_SCL=0;
    SDA_OUT;                //SDA 线输出
    IIC_SDA=1;
    DelayUs(5);
    IIC_SCL=1;
    DelayUs(5);
    IIC_SCL=0;
}

/*-----
 * 函数名: IIC_Send_Byte
 * 功能:   IIC 发送一个字节
 * 输入:   写入要发送的一个字节数据 txd
 * 输出:   无
-----*/
void IIC_Send_Byte(unsigned char txd)
{
    unsigned char t;
    SDA_OUT;                //SDA 线输出
```



```
IIC_SCL=0; //拉低时钟开始数据传输
for(t=0;t<8;t++)
{
    if((txd&0x80)>7)
        IIC_SDA=1;
    else
        IIC_SDA=0;
    txd<<=1;
    DelayUs(5);
    IIC_SCL=1;
    DelayUs(5);
    IIC_SCL=0;
    DelayUs(5);
}
}
/*-----
* 函数名: IIC_Read_Byte
* 功能:   IIC 读一个字节
* 输入:   无
* 输出:   读出存储器里面的数据并返回 receive
-----*/
unsigned char IIC_Read_Byte(void)
{
    unsigned char i, receive=0;
    SDA_IN; //SDA 设置为输入
    for(i=0;i<8;i++)
    {
        IIC_SCL=0;
        DelayUs(5);
        IIC_SCL=1;
        receive<<=1;
        if(IIC_SDA)receive++;
        DelayUs(5);
    }
    IIC_NAck(); //发送 nACK

    return receive;
}

/*-----
* 函数名: IIC_READ
* 功能:   IIC 读出制定位置的数据
* 输入:   address
```

* 输出： 读出 address 存储器里面的数据 iicdata

-----*/

unsigned char IIC_READ(unsigned char address)

```
{
    unsigned char iicdata = 0;
    IIC_READ_Begin:
        IIC_Start();
        IIC_Send_Byte(0xa0);
        if(IIC_Wait_Ack())goto IIC_READ_Begin;
        IIC_Send_Byte(address);                //填要读的数据地址
        if(IIC_Wait_Ack())goto IIC_READ_Begin;
        IIC_Start();
        IIC_Send_Byte(0xa1);
        if(IIC_Wait_Ack())goto IIC_READ_Begin;
        iicdata=IIC_Read_Byte();
        IIC_Stop();
        return iicdata;
}
```

/*-----*/

* 函数名： IIC_WRITE

* 功能： IIC 把数据 data 写入制定的位置 address

* 输入： address, data

* 输出： 无

-----*/

void IIC_WRITE(unsigned char address,unsigned char data)

```
{
    IIC_WRITE_Begin:
        IIC_Start();
        IIC_Send_Byte(0xa0);
        if(IIC_Wait_Ack())goto IIC_WRITE_Begin;

        IIC_Send_Byte(address);
        if(IIC_Wait_Ack())goto IIC_WRITE_Begin;

        IIC_Send_Byte(data);
        if(IIC_Wait_Ack())goto IIC_WRITE_Begin;

        IIC_Stop();
}
```

/*-----*/

* 函数名： main

* 功能： 主函数

* 输入： 无

* 输出: 无

-----*/

void main()

```
{  
    POWER_INITIAL();           //系统初始化  
  
    IICReadData = IIC_READ(0x12); //读取 0x12 地址 EEPROM 值  
    IIC_WRITE(0x13,~IICReadData); //取反写入地址 0x13  
    while(1)  
    {  
        NOP();  
    }  
}
```

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 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, 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.