



WTC401SPI 4键+1滚轮滑条电容式触摸感应芯片

快速浏览

按键数量	1-4 个按键，一个机械按键
滑条/滚轮数	1 个编码器(滑条/滚轮)
按键反应模式	单键反应模式
技术机理	电容数字转换技术
按键感应盘大小	最小 3mm×3mm,最大 30mm×30mm，根据实际需要和面板厚度而定
按键感应盘间距	最小间距 2mm,根据实际需要而定
按键感应盘形状	任意形状多边形,圆形或椭圆形,中间可留孔或镂空.
按键感应盘材料	PCB 铜箔,金属片,平顶圆柱弹簧,导电橡胶,导电油墨,导电玻璃的 ITO 层等
对 PCB 的要求	双面 PCB
面板材质	绝缘材料,如有机玻璃,普通玻璃,钢化玻璃,塑胶,木材,纸张,陶瓷,石材等
面板厚度	0~5mm
按键灵敏度调节方式	改变基准电容 CSEL 的值可对按键的灵敏度进行调节
有效触摸反应时间	小于 100ms
防水性能	对面板撒水,喷水按键不发生误动,面板漫水,积水时触摸按键无异常反映,
抗射频干扰性能	能够有效消除 GSM 手机贴近面板拨打或接听电话,大功率对讲机贴近面板进行对讲操作产生的射频干扰。
工作电压范围	2.4V-5.5V
工作温度范围	-40℃ - +85℃
数据传输接口	按键输出为 SPI 接口,滑条/滚轮输出为 SPI 数据。
储存温度范围	-50℃ - +125℃
芯片封装形式	SSOP24(150MIL)
典型应用	各种触摸感应遥控器,例如 LED 调光混色遥控器等需要连续调节物理量的应用场合。.



目录

1	产品概述.....	3
2	订货信息.....	3
3	典型应用.....	4
4	WTC401SPI 引脚定义.....	4
5	芯片的休眠和唤醒.....	5
6	输出指示.....	6
7	WTC401SPI 的 SPI 接口.....	6
8	使用 SPI 接口的注意事项.....	7
9	灵敏度设定.....	8
10	WTC401SPI 使用的电容传感器（感应盘（sense element））.....	9
11	滑轮或滑条使用的电容传感器和 layout 建议.....	11
12	空置传感器通道的处理.....	12
13	WTC401SPI 封装及尺寸图.....	12



WTC401SPI 4键+1滚轮滑条电容式触摸感应芯片

规格书(V1.1)

1 产品概述

WTC401SPI 是有 4 个按键和 1 个滚轮/滑调, 通过 SPI 接口输出键值和手指触摸的滚轮位置数据 0~7FH, 共 128 个。SSOP24A 封装

1.1 概述

WTC401SPI 是为替代机械式轻触按键和机械旋钮, 实现防水防尘、密封隔离、坚固美观的操作界面而设计的一款触摸感应 IC。一个 WTC401SPI 可实现 4 个按键和 1 个滚轮(滑条)。

1.2 技术参数

工作电压: $2.4V < V_{CC} < 5.5V$

输出电压范围 $GND < V_{out} < V_{CC}$

感应厚度(绝缘介质): 0-8mm, 最大 10mm

有效触摸反映时间: 小于 150ms

工作温度: $-40^{\circ}C - +85^{\circ}C$

储存温度: $-50^{\circ}C - +125^{\circ}C$

静态电流: 18uA (3V 电源)

动态电流: 1.2mA (3V 电源)

唤醒方式: 触摸感应按键唤醒

2 订货信息

WTC401SPI 有带休眠模式和不带休眠模式之分, 从型号上加以区分, 用户订货时须提供完整的型号

产品型号	是否带休眠模式	使用注意事项
------	---------	--------



WTC401SPI WTC401SPI-L

WTC401SPI	此型号不带休眠模式	用于家电产品
WTC401SPI-L	此型号带休眠模式	可满足电池供电的低功耗应用

3 典型应用

需要通过手指滑动来连续调节物流量的应用，适用于各种厨房设备，音视频设备，空调器，卫浴电器，灯具开关，LED 调光混色等应用。

4 WTC401SPI 引脚定义

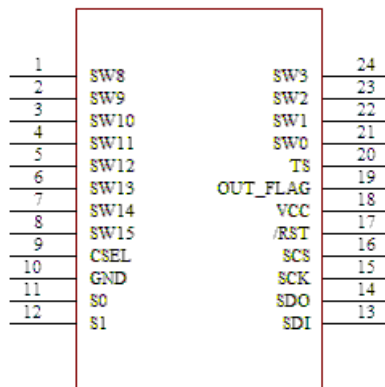


图 2:WTC401SPI 引脚图

表1: WTC401SPI引脚定义

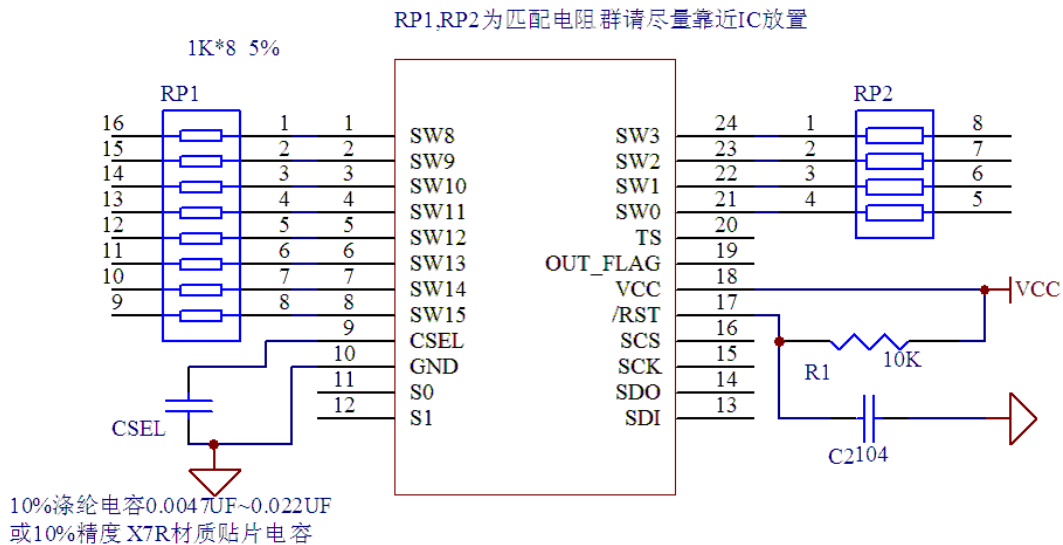
管脚序号	管脚名称	用法	功能描述
1	SW8	I	电容传感器（滑轮感应盘）接口8
2	SW9	I	电容传感器（滑轮感应盘）接口9
3	SW10	I	电容传感器（滑轮感应盘）接口10
4	SW11	I	电容传感器（滑轮感应盘）接口11
5	SW12	I	电容传感器（滑轮感应盘）接口12
6	SW13	I	电容传感器（滑轮感应盘）接口13
7	SW14	I	电容传感器（滑轮感应盘）接口14
8	SW15	I	电容传感器（滑轮感应盘）接口15
9	CSEL	I	灵敏度调整电容接口
10	GND	I	电源地
11	S0	O	内部测试脚平时需将S0悬空



WTC401SPI WTC401SPI-L

12	S1	O	内部测试脚平时需将S1悬空
13	SDI	I	芯片SPI接口的数据输入线，输入灵敏度调节数据
14	SDO	O	芯片SPI接口的数据输出线，输出按键和滑条数据
15	SCK	I	芯片SPI接口的时钟输入线
16	SCS	I	芯片SPI接口的片选使能线，
17	/RST	I	芯片复位脚
18	VCC	I	正电源输入
19	OUT FLAG	O	按键状态指示，有按键或滑条按下时输出低电平信号，手指离开按键或滑条输出高电平信号。上电初始为高电平输出
20	TS	O	内部测试脚平时需将TS悬空
21	SW0	I	电容传感器（按键感应盘）接口0
22	SW1	I	电容传感器（按键感应盘）接口1
23	SW2	I	电容传感器（按键感应盘）接口2
24	SW3	I	电容传感器（按键感应盘）接口3

芯片的应用原理图



5 芯片的休眠和唤醒

当触摸滑轮或触摸按键都没有被触摸的情况下 15~20 秒后芯片会自动进入省电的休眠状态，以节省工作电流。此时芯片的耗电流在 3V 条件下只有 18~20uA。休眠状态下触摸滑轮不再起作用。

芯片在休眠状态下触摸任何一个按键感应盘都可以将芯片唤醒。唤醒后芯片的工作电



流在 3V 条件下为 1.2mA 左右。进入休眠后触摸滑轮不能唤醒芯片。

6 输出指示

当检测到感应按键或滑轮上有效触摸发生 150ms 内 WTC401SPI 的 OUT_FL 脚输出低电平信号，低电平会保持一直到手指离开按键或滑轮才会变为高电平。

7 WTC401SPI 的 SPI 接口

7.1 SPI 时序图

SPI 时序图如图 4:

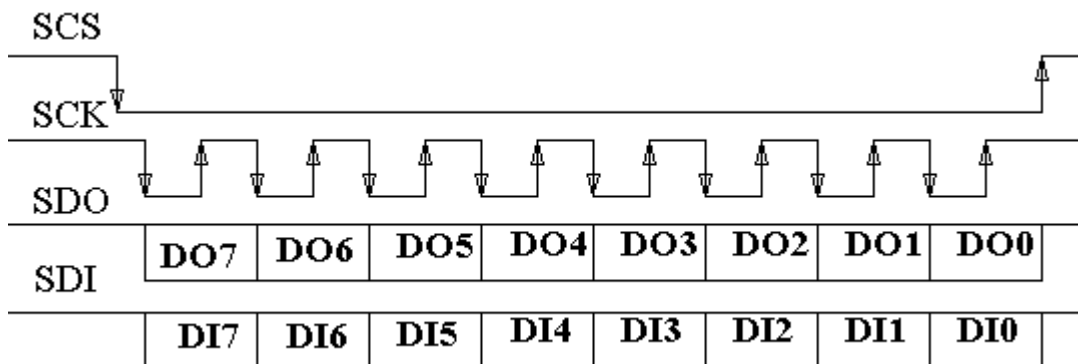


图 4: WTC401SPI 的 SPI 接口时序图

7.2 WTC401SPI 的 SPI 口的工作状态

WTC401SPI 的 SPI 口工作在从机状态

7.3 SPI 信号线功能

SCS 是 SPI 口的片选使能信号，由用户 MCU 设定。SCS = 1 时 SPI 口不工作。SCS = 0 时打开 SPI 口功能。

SCK 是 SPI 口的时钟信号，由用户 MCU 产生。初始电平应为高。

SDO 是 WTC401SPI 的 SPI 数据输出信号，用户 MCU 读取 SDO 的状态获取按键数据。

SDI 是 WTC401SPI 的 SPI 数据输入信号，用户 MCU 通过设置 SDI 可以用软件设置触



摸按键的灵敏度。不使用时请务必将 SDI 置为高电平。

7.4 SPI 的数据传输规则

.SPI 的数据都是从最高位开始传输。

7.5 SDI 信号的处理

如果不使用软件通过 SDI 设置触摸灵敏度，请将 SDI 接 VCC。

如果需要使用 SDI 那么在读 SDO 操作前务必将 SDI 置为高电平。否则可能会错误设置触摸灵敏度为 0 级。

8 使用 SPI 接口的注意事项

8.1 两次读取时间间隔不小于 4ms

WTC401SPI 的 SPI 接口如果连续读写会造成器件工作的不稳定，两次读写的时间请间隔 4ms 以上。

8.2 SCS 和 SDI,SDO 任何时候都不允许悬空

如果触控板上没有与 SPI 接口的主控 MCU，则必须在 WTC401SPI 的 SCS 和 SDI,SDO 上加上拉电阻，以保证芯片能够稳定工作。SCS 与 SDI,SDO 任何时候都不能悬空。

8.3 SDI 可以固定接 VCC

如果不使用软件设置 WTC401SPI 的灵敏度则 SDI 必须加上拉电阻或直接接 VCC 固定置为高。

8.4 SCS 不允许固定接地

SCS 不允许固定接地

WTC401SPI 的 SPI 数据格式



从 SDO 输出（主机读出的按键或滑条数据）

Bit7	为 1 表示触摸按键的键值数据（键值数据有效值为 80H~87H） 为 0 表示触摸滚轮的位置数据（位置数据有效值为 0~7FH）
Bit6	键值数据或位置数据的第 6 位（键值或位置数据由 Bit7 决定）
Bit5	键值数据或位置数据的第 5 位（键值或位置数据由 Bit7 决定）
Bit4	键值数据或位置数据的第 4 位（键值或位置数据由 Bit7 决定）
Bit3	键值数据或位置数据的第 3 位（键值或位置数据由 Bit7 决定）
Bit2	键值数据或位置数据的第 2 位（键值或位置数据由 Bit7 决定）
Bit1	键值数据或位置数据的第 1 位（键值或位置数据由 Bit7 决定）
Bit0	键值数据或位置数据的第 0 位（键值或位置数据由 Bit7 决定）

从 SDI 输入（主机写入的灵敏度调节数据）

Bit7	必须为 0，为 1 的话本次传输无效
Bit6	必须为 0，为 1 的话本次传输无效
Bit5	为 1 表示本次数据设定的是触摸按键的灵敏度 为 0 表示本次数据设定的是触摸滑轮的灵敏度
Bit4	按键灵敏度或滑轮灵敏度的第 4 位（按键或滑轮数据由 Bit5 决定）
Bit3	按键灵敏度或滑轮灵敏度的第 3 位（按键或滑轮数据由 Bit5 决定）
Bit2	按键灵敏度或滑轮灵敏度的第 2 位（按键或滑轮数据由 Bit5 决定）
Bit1	按键灵敏度或滑轮灵敏度的第 1 位（按键或滑轮数据由 Bit5 决定）
Bit0	按键灵敏度或滑轮灵敏度的第 0 位（按键或滑轮数据由 Bit5 决定）

对于芯片 SPI 接口的使用，用户也可以直接参考规格书篇幅最后附录的 DEMO 程序，这样会比较直观和简单

9 灵敏度设定

9.1 灵敏度调节方式

WTC401SPI 的灵敏度设定可以让用户使用各种厚度的隔离介质实现可靠，灵活的触摸功能。

WTC401SPI 的灵敏度设定使用软、硬件双重设定。用户可以将 SDI 固定接高电平，从而不使用软件灵敏度调整功能，只调整 Csel 来实现灵敏度的设定，这样主控 MCU 可以节约



一条 I/O 口线，而且可以减少主控 MCU 的软件空间。

9.2 基准电容 CSEL 的材质、取值范围及精度要求

1: 选择合适的电容 Csel

首先用户根据自己的使用情况选择合适的电容 Csel，隔离介质越厚使用的 Csel 容量越大，一般建议在 0.0047UF 和 0.022UF 之间由小到大地选择合适的电容。**Csel 建议最好使用温度系数小的百分之五精度涤纶电容**。百分之十精度的**涤纶电容**也可使用。如需使用贴片电容则必须使用 10%或更高精度的 **NPO 材质** 电容或 **X7R 材质** 电容。

2: 软件设置触摸按键感应灵敏度的注意事项

主控 MCU 每次设定灵敏度后触摸芯片都会重新复位，复位需要 80ms 的时间。在触摸芯片重新复位的 80ms 时间内，不论读写都不会得到正确的结果。所以主控 MCU 设定灵敏度完成后必须等待 80ms 以上的时间才能对 WTC401SPI 的串行接口进行读写。

建议主控 MCU 在程序中不要频繁的对触摸芯片的灵敏度进行设定。只要在程序初始化时设定一次就可以了。

触摸按键感应灵敏度分为 32 级，相应的级数数据为 0~31。级数越高触摸按键的灵敏度就越高。但设定的级数数据如果超过了 31 本次设置将无效，无效设置发送结束位后触摸芯片不会重新复位，也不会进行内部参数的调整，仍然会使用以前的灵敏度参数。

触摸芯片初次上电后灵敏度内部自动设定为第 31 级，用户也可以不使用软件调整灵敏度。直接使用芯片的默认参数。

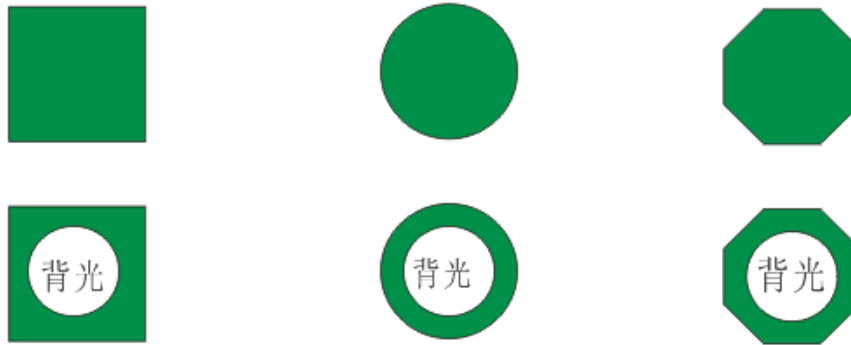
9.3 感应盘面积大小对灵敏度的影响

加大感应盘（sense element）的面积有利于提高触摸感应的穿透能力。

10 WTC401SPI 使用的电容传感器（感应盘（sense element））

10.1 触摸按键电容传感器的材料和形状

电容传感器可以是任何形状的导体，但要保证一定的平面面积。建议使用直径大于 12mm 的圆形金属片或其他导体。常用的感应盘（sense element）有 PCB 板上的铜箔、弹簧、薄膜线路以及 ITO 玻璃等。如图 9 所示。



按键感应盘可以是实心或中空矩形、圆形、多边形

图 9:PCB 铜箔感应盘 (sense element)

10.2 感应盘的面积

每个感应盘 (sense element) 的面积应尽量保持相同,以确保灵敏度相同。

10.3 感应盘与面板之间的贴合

电容传感器应紧贴玻璃等绝缘面板,或使用弹性贴合。

常用的贴合方法有:

- A 使用带弹簧的感应盘 (sense element)
- B 用圆柱状导电橡胶进行弹性连接
- C 将感应盘 (sense element) 用进口强力双面胶粘在面板上, 双面胶层不能太厚。

如图 10 所示



图 10:弹簧感应盘 (sense element)



10.4 消除贴合间隙

感应盘 (sense element) 和绝缘面板的贴合面必须要平整, 确保它们之间是无间隙的密贴合。

10.5 按键和滚轮 (滑条) 灵敏度的均衡

如果感应按键和感应滚轮两者的灵敏度不一致可以通过调整感应通道上串接的匹配电阻阻值来调整。匹配电阻阻值越大相应感应通道的灵敏度就越低。但匹配电阻的值不要小于 1K。滚轮感应通道上串接的匹配电阻的值必须一致。

11 滚轮或滑条使用的电容传感器和 layout 建议

- 采用双面 PCB 板进行设计。
- 电容传感器 (感应盘): 用 PCB 上的铜箔做电容传感器 (按键感应盘) 和滚轮 (滑条) 的电容传感器 (电容传感器)。每个电容传感器的面积应尽量保持相同, 以确保灵敏度相同。电容传感器应放置在 TOP 层。
- 触摸芯片放置于 BOTTOM 层适中的位置。
- 电容传感器与触摸芯片的引脚间的连线全部走 BOTTOM 层
- 电容传感器与触摸芯片的引脚间的连线尽量短和细 (5~8mil)。
- 连线的背面和周围 0.2mm 不要铺铜和放置其它回路, 以保证传感器有好的灵敏度并避免误触发。
- 触摸滚轮 (滑条)

触摸感应滑条形状和尺寸如图 4, 可根据面板设计要求, 按比例适当缩放滑条电容传感器的尺寸, 并且可以将滑条设计成曲线。

触摸滑条

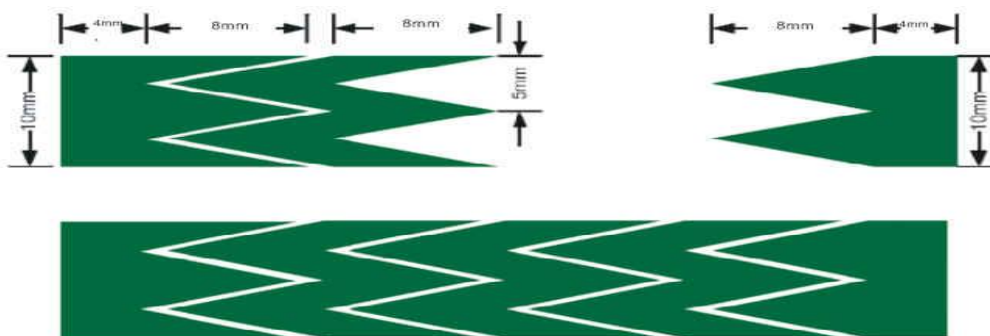


图 11: 触摸滑条的几何图形和尺寸



图 12:触摸滚轮的几何图形和尺寸

触摸感应滚轮的电容传感器的形状如图 5，电容传感器之间的间隙为 0.1-0.2mm,滚轮内外直径分别为 18mm、36mm，可根据面板设计要求,按比例适当缩放滚轮尺寸。

触摸感应滚轮电容传感器从左开始顺时针方向排列顺序为：

SW13,SW10.SW15,SW12,SW9,SW14,SW11,SW8,SW15.SW14,SW13,SW12,SW11,SW10,SW9,SW8

(具体可以参考我们提供的原理图和应用 DEMO 板的 PCB 图)。

12 空置传感器通道的处理

WTC401SPI 要求必须至少使用 2 个触摸按键，WTC401SPI 应用在少于 4 个键的场合时 SW3~SW0 会有空置不用的传感器输入通道。空置输入通道只要简单的悬空就可以，空置输入通道不能加任何上拉或下拉电阻。

如果需要使用 2 个以下的触摸按键，请和我们联系以获取相应的技术支持。

13 WTC401SPI 封装及尺寸图

Symbol	Dimensions in mil		
	Min.	Nom.	Max.
A	228	—	244
B	150	—	157
C	8	—	12
C'	386	—	394
D	54	—	60
E	—	25	—
F	4	—	10
G	22	—	28
H	7	—	10
α	0°	—	8°

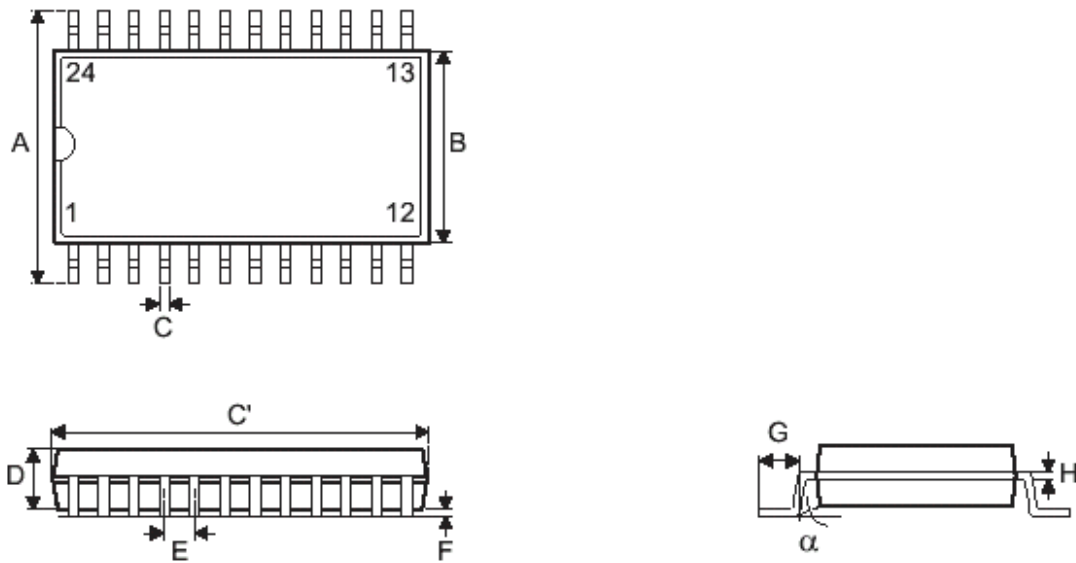


图 13: WTC401SPI 封装和尺寸图

附:

SPI 读写的 DEMO 程序

```
//-----  
//  
//-----  
void main(void)  
{  
    SCS = 1; //关闭 SPI 口使能  
    SCK = 1; //SCK 初始电平为高  
    SDI = 1; //SDI 设置为高电平  
    set_Subtle_SPI(10 | 0x20); //将按键灵敏度设定为 10 级  
    delay_loop(5ms); //延时 5MS  
    set_Subtle_SPI(10); //将滑条灵敏度设定为 10 级  
    delay_loop(5ms); //延时 5MS  
  
    while(1)  
    {  
        SDI = 1;  
    }  
}
```



WTC401SPI WTC401SPI-L

```
if(~OutFlag) //
{
    delay_loop(5ms); //延时，实际程序中每5MS读一次SPI
    GetKey = get_key_data(); //读取键值
    if (GetKey & 0x80 == 1)
    { //bit7为1 读取的是按键数据
        GetKey &= 0x7F; //将数据的第7位置为0，得到按键键值
    }
    else
    { //bit7为0 读取的是滑条数据
    }
}
}

//-----
uchar get_key_data(void)
{
    uchar KeyData;
    uchar i;
    KeyData = 0;
    i = 0; //计数器指初始为0
    SDI = 1; //SDI置为高，以避免错误设置触摸芯片的灵敏度
    SCS = 0; //打开SPI口使能
    do
    {
        KeyData <<= 1; //MSB为数据第一位
        SCK = 0; //SCK信号下降沿
        if(SDO == 1) //读取SDO的数据
        { //SDI为高电平
            KeyData |= 0x01;
        }
        else
        { //SDI为低电平
            KeyData &= 0xFE;
        }
        SCK = 1; //SCK信号上升沿
        i++; //计数器加1
    }while (i < 8); //循环读数8次
```



WTC401SPI WTC401SPI-L

```
SCS = 1;          //关闭 SPI 口使能
return(KeyData); //返回读取的按键信息
}

//-----
void set_Subtle_SPI(uchar temp) //灵敏度设定
{
    uchar i;
    i= 0;          //计数器指初始为 0
    SCS = 0;      //打开 SPI 口使能
    do
    {
        if((temp & 0x80) != 0) //发送数据的第一位为 MSB
        {
            //敏感度设定值的当前位值为 1
            SDI = 1;
        }
        else
        {
            //敏感度设定值的当前位值为 0
            SDI = 0;
        }
        SCK = 0; //SCK 信号下降沿
        SCK = 1; //SCK 信号上升沿
        temp <<= 1; //发送数据的第一位为 MSB
        i++; //计数器加 1
    }while (i < 8); //循环发送 8 次
    SCS = 1; //关闭 SPI 口使能
    SDI = 1; //SDI 设置为高电平
}
}
```

