**WTC401SPI 4 keys + 1 roller slider capacitive touch sensing chip**

**Specification (V1. 1)**

## 1. Product Overview

WTC401SPI has 4 keys and 1 roller/slider. Through the output key value of the SPI interface and the roller of finger touch, position data is 0 ~ 7FH, total 128 pieces. SSOP24A encapsulation

## 2. Ordering information

The WTC401SPI is divided into the sleeping mode and non-sleeping mode, which can be distinguished from the model. The user should provide the complete model when ordering

| Product model | Is there a mode of hibernation | Application notice |
|---|---|---|
| WTC401SPI | This model is not with sleeping mode | Used for the household electrical appliances |
| WTC401SPI-L | This model is with sleeping mode | It can satisfy the low power consumption application of battery power supply |

## 3. Typical Application

It needs to continually adjust the application of the material flow through finger sliding, is applied to various kinds of kitchen equipment, audio and video equipment, air conditioner, bathroom appliance, lamp switch, LED light color mixture and other applications.

## 4. Definitions of WTC401SPI Pins

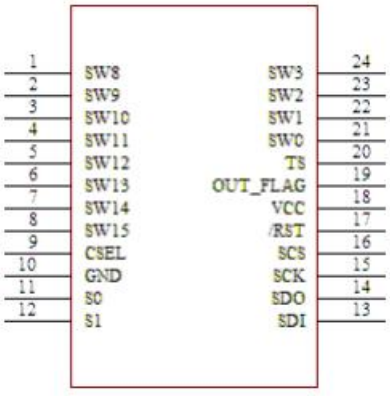


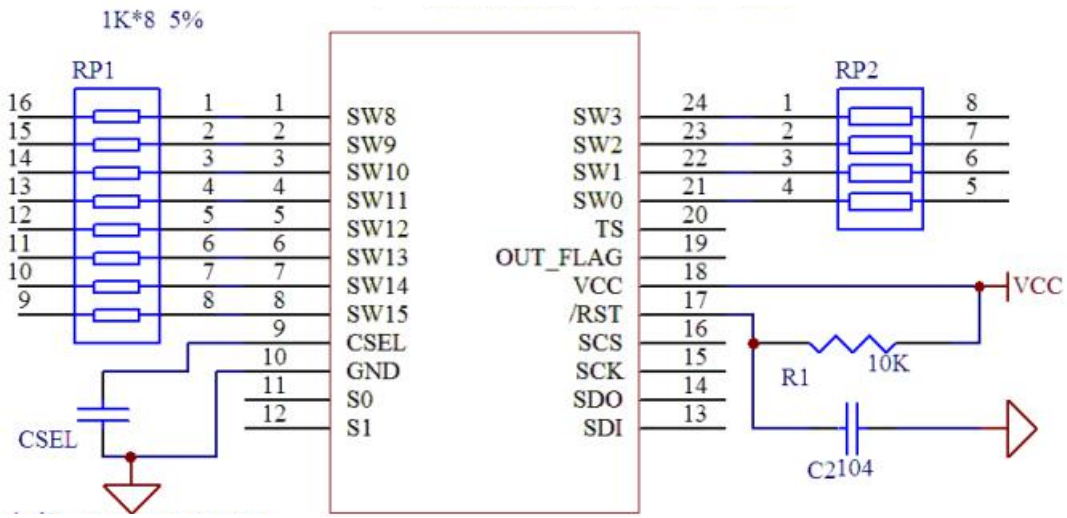

Figure 2：WTC401SPI Pin Diagram

Table 1: Definitions of WTC401SPI Pins

| Pin No. | Pin Name | Usage | Function Description |
|---|---|---|---|
| 1 | SW8 | I | Capacitive sensor (pulley sense element) interface 8 |
| 2 | SW9 | ! | Capacitive sensor (pulley sense element) interface 9 |
| 3 | SW10 | I | Capacitive sensor (pulley sense element) interface 10 |
| 4 | SW11 | I | Capacitive sensor (pulley sense element) interface 11 |
| 5 | SW12 | I | Capacitive sensor (pulley sense element) interface 12 |
| 6 | SW13 | I | Capacitive sensor (pulley sense element) interface 13 |
| 7 | SW14 | I | Capacitive sensor (pulley sense element) interface 14 |
| 8 | SW15 | I | Capacitive sensor (pulley sense element) interface 15 |
| 9 | CSEL | I | Sensitivity adjustable capacitor interface |
| 10 | GND | I | Power ground |
| 11 | S0 | O | The internal test leg Usually needs to suspend the S0 |
| 12 | S1 | O | The internal test leg usually needs to suspend the S1 |
| 13 | SDI | I | Data input line of the chip SPI interface is input the sensitivity adjustment data |
| 14 | SDO | O | The data output line of the chip SPI interface is output the key and slider data |
| 15 | SCK | I | The clock input line of the Chip SPI interface |
| 16 | SCS | I | The chip selection enabled line of Chip SPI interface |
| 17 | /RST | I | Chip reset pin |
| 18 | VCC | I | positive power supply output |
| 19 | OUT FLAG | O | Key status indicator has the output low level signal when pressing the key or slide bar, and the output high level signal when the finger leaves the key or slide bar. The initial of the electrifying is the high level output |
| 20 | TS | O | The internal test leg needs to suspend the TS |
| 21 | SW0 | I | Capacitive sensor (key sense element) interface 0 |
| 22 | SW1 | I | Capacitive sensor (key sense element) interface 1 |
| 23 | SW2 | I | Capacitive sensor (key sense element) interface 2 |
| 24 | SW3 | I | Capacitive sensor (key sense element) interface 3 |

Chip Application Schematic Diagram

RP1 and RP2 are the match resistance group. Please try to place them close to the IC




10 % polyester capacitance 0.0047UF-0.022UF
Or 10% precision X7R texture chip capacitor

## 4. The sleep and awakening of the chip

When the touch pulley or touch key is not touched, after 15 ~ 20 seconds, the chip will automatically enter the power saving sleeping mode to save the working current. At this time, he power consumption of the chips is only 18 ~ 20uA under 3V condition. The touch pulley is no longer working in the sleeping mode. The chip can be awakened by touching any keystroke sensor under the sleeping mode. The working current of the chip after being awakened is about 1.2 mA under 3V condition. After entering sleeping, the touch pulley cannot wake the chip.

## 5. Output Display

When the effective touch on the sensor key or pulley is detected, the OUT_ FL WTC401SPI in 150ms will output a low level signal, and the low level will remain and will become high level until the finger has left the key or the pulley.

## 6. The SPI interface of WTC401SPI

### 6.1 SPI sequence diagram

The SPI time sequence diagram is shown in figure 4:

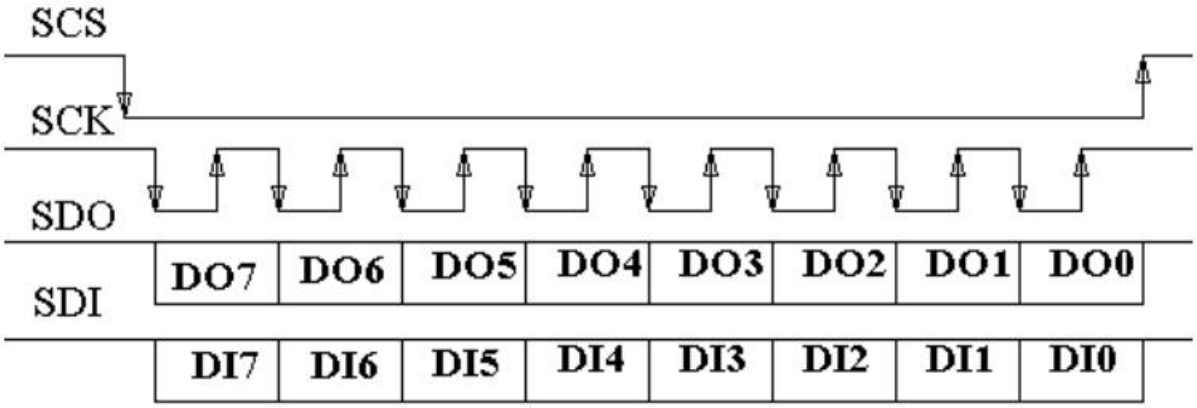



Figure 4: Sequence diagram of the SPI interface of WTC401SPI

### 6.2. SPI port work status of WTC401SPI

The working subordinate machine state of SPI port of WTC401SPI

### 6.3. PI signal line function

SCS is the enable signal of chip selection of SPI port, is set by the user MCU. The SPI port does not work when SCS = 1. SPI port function is opened when SCS = 0.
SCK is the clock signal of SPI port, generated by user MCU. The initial electrical level should be high.
SDO is the SPI data output signal of WTC401SPI, and the user MCU reads the status of SDO to obtain key data.
SDI is the SPI data input signal of WTC401SPI, and the user MCU can set the sensitivity of the touch key with software by setting SDI. Please be sure to set the SDI to be high level when not in use.

### 6.4. Data transmission rules of SPI

The data of SPI is transmitted beginning from the highest level.

### 6.5. SDI signal processing

If you do not use software to set touch sensitivity through setting SDI, please connect SDI to VCC.
If SDI is needed to be used, SDI must be set to be high level before reading the SDO operation.
Otherwise, the touch sensitivity may be wrong set to be 0.

## 7. Notices for using the SPI interface

### 7.1. The two read intervals are not less than 4ms

If the SPI interface of WTC401SPI is continuous read-write, it will cause the instability of the device working. The time of reading and writing twice should be more than 4ms.

### 7.2. SCS and SDI, SDO are not allowed to be suspended at any time

If there is no master control MCU of the SPI interface on the touchpad, then the pulling resistance must be added on the SCS and SDI and SD0 of WTC401SPI, to ensure that the chip can steadily operate.
SCS and SDI, SD0 cannot be suspended at any time.

### 7.3. SDI can be fixed to connect VCC

If the sensitivity of WTC401SPI is not set by using the software, the pulling resistance must be added to SDI Or SDI must directly connect to fix the VCC to set the high.

7.4. SCS is not allowed to be fixed grounding

SCS is not allowed to be fixed grounding

The SPI data format of WTC401SPI
From the SDO output (the key or slide bar data read by the host )

| | |
|---|---|
| Bit7 | When it is 1, it indicates the key value data of the touch key (the k valid value of the key value data is 80H ~ 87H).<br>When it is 0, it indicates the position data of the touch roller (the valid value of the location data is 0~7FH). |
| Bit6 | The 6th place of the Key value data or location data (key value or location data is determined by Bit7) |
| Bit5 | The 5th place of the Key value data or location data (key value or location data is determined by Bit7) |
| Bit4 | The 4th place of the Key value data or location data (key value or location data is determined by Bit7) |
| Bit3 | The 3rd place of the Key value data or location data (key value or location data is determined by Bit7) |
| Bit2 | The 2nd place of the Key value data or location data (key value or location data is determined by Bit7) |
| Bit1 | The 1st place of the Key value data or location data (key value or location data is determined by Bit7) |
| Bit0 | The 0 place of the Key value data or location data (key value or location data is determined by Bit7) |

From the SDI input (the Sensitivity adjustment data write by the host)

| | |
|---|---|
| Bit7 | It must be 0. If it is 1, the transmission this time is not valid |
| Bit6 | It must be 0. If it is 1, the transmission this time is not valid |
| Bit5 | When it is 1, it indicates the data set is the sensitivity of the touch key.<br>When it is 0., it indicates that data set this time is the sensitivity of the touch pulley |
| Bit4 | The 4th position of the key sensitivity or pulley sensitivity (the data of key or pulley is determined by Bit5). |
| Bit3 | The 3rd position of the key sensitivity or pulley sensitivity (the data of key or pulley is determined by Bit5). |
| Bit2 | The 2rd position of the key sensitivity or pulley sensitivity (the data of key or pulley is determined by Bit5). |
| Bit1 | The 1rd position of the key sensitivity or pulley sensitivity (the data of key or pulley is determined by Bit5). |
| Bit0 | The 0 position of the key sensitivity or pulley sensitivity (the data of key or pulley is determined by Bit5). |
| | |

For the use of the chip SPI interface, the user can also directly refer to the DEM0 program of the final appendix of the specification, which will be more intuitive and simple.

## 8. Sensitivity Setting

### 8.1. Sensitivity adjustment mode

The sensitivity setting of WTC401SPI enables the user to use isolated media of various thicknesses to implement reliable and flexible touch function.

The sensitivity of WTC401SPI is set by using the double setting of soft and hardware. The user can be fix the SDI to connect the high level, so that it does not use the sensitivity adjustment function of software, just adjusts the Csel to realize the sensitivity setting, so that the master control MCU can save a I/O port line, and also reduce the software space of the master MCU.

### 8.2. The material of Benchmark capacitance CSEL, , value range precision requirements

1. Selection of Suitable Capacitor CSEL

The user can select the appropriate capacitance Csel according to the own use case. The larger the separation medium is, the larger the Csel capacity is. It is generally recommended to choose the appropriate capacitance between 0.0047 UF and 0.022 UF from large to small. It is recommends that it is best to use the A 5 % precision polyester capacitor with a small temperature coefficient for Csel. 10 percent accuracy of polyester capacitance can also be used. If needing use of patch capacitors, 10% or higher precision NPO material or X7R capacitance needs to be used.

2：Notices for the sensitivity of the touch key sensor set by the software

The master control MCU will reset every time after the sensitivity is set, and the reset will take 80ms. In the 80ms time when the touch chip is reset, regardless of whether it is reading or writing, it will not get the correct result. Therefore, the master control MCU must wait for more than 80ms after sitting the sensitivity of master control MCU is completed, then the read-write can be made to the serial interface of WTC401SPI.

It is recommended that the master control MCU should not frequently set the sensitivity of touch chip in the program. Just setting it once when the program is initialized.

The sensitivity of touch key is divided into 32 grades, and the corresponding series data is 0 ~31. The higher the series is, the higher the sensitivity of the touch key is. But if the set series data is more than 31, this set will be invalid. After the invalid set is transmitted to end bit, the touch chip won't to reset and won’t adjust internal parameters, can still use the previous sensitivity parameters.

After the touch chip is electrified for the first time, the inside setting of the sensitivity is automatically set to be the level 31. The user can adjust the sensitivity by not using the not use the software, to use the chip's default parameters directly.

8.3. The effect of the area and size of the sense element on sensitivity

Increasing the area of sense element can improve the penetration ability of touch sensing.
Appendix :

DEMO program read and written by SPI

```
//-------------------------------------------------------------------------------------------------------------------
//-------------------------------------------------------------------------------------------------------------------
void main(void)
{
   SCS = 1;        // Close the SPI port to enable
   SCK = 1; //SCK The initial electrical level is high
   SDI = 1; //SDI SDI is set to be high level
   set_Subtle_SPI(10 | 0x20);     //Delay_loop(5ms);
   Set the key sensitivity to 10 level delay_loop(5ms);
   delay_loop(5ms);       // 5MS Delayed
   set_Subtle_SPI(10);    // set the slider sensitivity to be the level 10
   delay_loop(5ms);        //5MS Delayed

   while(1)
   {
           SDI = 1;
           if(～OutFlag)
           {
                   delay_loop(5ms);                        //Delayed, in the actual program, it can read the SPI every 5MS
                   GetKey =    get_key_data();             //Read the key value
               if (GetKey & 0x80 == 1)
               {             //bit7 The bit7 which the read is 1 is the key data of the key
           GetKey &= 0x7F；                  //Set the 7 place of the data to be 0 to get the key value of the key
        }
        else
        {                //bit7 is the slider data read by 0
     }
   }
}

//-------------------------------------------------------------------------------------------------------------------
uchar     get_key_data(void)
{
            uchar     KeyData;
            uchar     i;
            KeyData = 0;
            i = 0;              //The initial of counter is 0
            SDI = 1; //SDI The setting of SD is high to avoid error setting the sensitivity of touch chip
            SCS = 0; // Open the SPI to enable
            do
            {
               KeyData《=1;          //MSB is the number one of data
               SCK = 0;//SCK Signal falling edge
            if(SDO == 1) //Read the data of SDO
```

```
            {               //SDI is the high level
                    KeyData |= 0x01;
            }
            else
            {           //SDI is the low level
                  KeyData &= 0xFE;
            }
            SCK =1;//SCK Signal rising edge
            i++;       // Counter adds 1
            }while (i < 8);  // Cycle reading 8 times
            SCS = 1;                    // Close the SPI port to enable
            return(KeyData); // Return the read key information
}

//-----------------------------------------------------------------------------------------------------------------------------
   void       set_Subtle_SPI(uchar temp)  //Sensitivity setting
{
      uchar i;
      i=0;                  //The initial of counter is 0
      SCS = 0;              //Open the SPI port to enable
                do
                {
                   if((temp & 0x80) != 0)      //The first place of the transmitted data is MSB

                {                               //The current value of the sensitivity setting value is 1
                            SDI = 1;
                }
                else
                {                              //The current value of the sensitivity setting value is 0
                      SDI = 0;
                }
                SCK = 0;    //The falling edge of SCK signal
                SCK = 1;      //The rising edge of SCK signal
                temp <<= 1;           //The first place of the transmitted data is MSB
                i++;//Counter adds 1
                }while (i < 8);          //Circular transmission 8 times
                SCS = 1;      //Close the SPI port to enable
                SDI = 1;                //SDI is set to be high level
```