

Programming Sequences and Tips for TSC2000/2200/230x Applications

Wendy X. Fang

DAP Group

ABSTRACT

To apply the touchscreen function of a TSC2000, TSC2200, or TSC230x device, certain software programming sequences are crucial during the TSC initialization and data reading and are also helpful during the TSC mode changing. This application report presents some practical programming sequences. Additionally, some program and debug tips are provided.

Introduction

The Texas Instruments (TI) high-performance touchscreen controllers TSC2000, TSC2200, TSC2300, TSC2301, and TSC2302 can be called the *first generation* of TI smart touchscreen devices. Each device contains a simple on-chip microprocessor and a series of memory registers. For detailed specifications of these devices, see the data sheets or the product files at the following TI Web-site location:

<http://focus.ti.com/analog/docs/analogprohome.tsp?templateId=4&familyId=82>

To apply any of these devices, some specific software programming sequences are required for touchscreen initialization or configuration, for touchscreen data reading, and for the ADC conversion mode changing from one mode to another. The initialization sequence is especially important and needs to be strictly followed due to a hardware bug with these devices, as stated in the TI documentation:

<http://focus.ti.com/lit/er/slaz003/slaz003.pdf>

The touchscreen programming sequence for all the devices is exactly the same. Therefore, the following discussion applies to all the devices. For simplification, this application report uses the term TSC to stand for each device.

Initialization Sequence

To use the TSC touchscreen function, the on-chip touchscreen control registers (located on memory page 1 of the TSC) must be initialized or set up properly in a specific order or sequence. Figure 1 shows the initialization sequence.

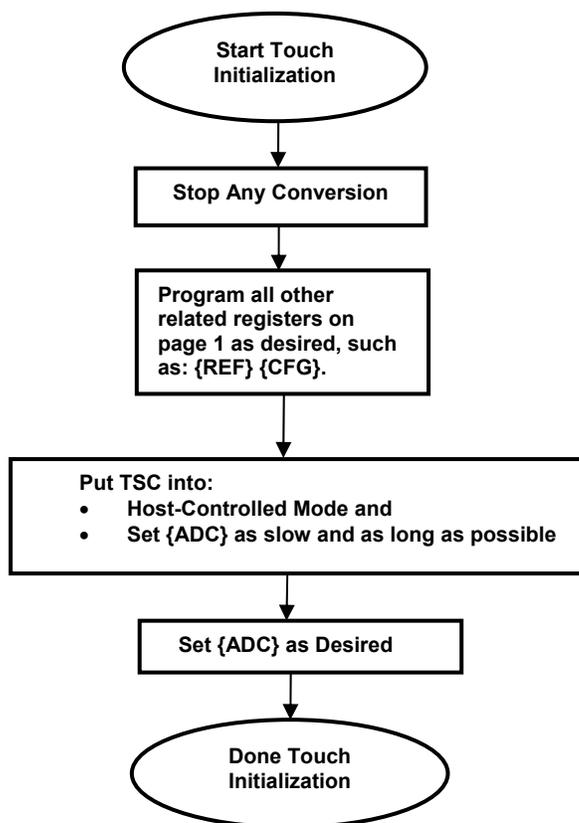


Figure 1. TSC Touch Function Initialization Sequence

As seen in Figure 1, the content within the brackets “{” and “}” denotes a TSC register. For example, {ADC} designates the ADC register that is located on page 1 and address 0x00.

Example 1: To initialize the TSC touchscreen to TSC-controlled XYZ mode, the initialization sequence could be:

```

/*****/
TSC Write Register { 0x0800, 0xC8A2 }; // to stop ADC conversion
TSC Write Register { 0x0860, 0x0017 }; // to set up {REF}
TSC Write Register { 0x08A0, 0x000A }; // to set up {CFG}
TSC Write Register { 0x08C0, 0xCBAD }; // to set up {CFG2}
TSC Write Register { 0x0800, 0x07FF }; // to put TSC to host and slowest mode
TSC Write Register { 0x0800, 0x88A2 }; // to set up (initialize) {ADC}
/*****/
  
```

In Example 1, the first 16-bit data in each line is the command word, and the second is the value written to the corresponding registers.

Tip 1: After the proper initialization, the /DAV pin on the TSC chip is in a logic-high state. The /DAV pin goes low (i.e., active) when the screen or panel is touched (and the ADC data are ready), if the device is under the TSC-controlled mode. If the /DAV is not high after the software initialization or never goes low when the screen has been touched, it is possible that the TSC register has not been properly written to or initialized.

Tip 2: The initialization sequence shown in Figure 1 is needed **ONLY** once after each power up or reset.

Note 1: The SPI™ on the TSC2000/2200/230x does not support multiple writing operations. In other words, a TSC writing procedure must contain and be limited to 32 SCLKs (or two 16-bits). Thus, after the /SS line goes active (low), the 32 synchronous clock pulses appear at the SCLK line, and then the /SS line goes back high. In the meantime, one 16-bit command word and one 16-bit data word go through the MOSI line. For details of the writing operation timing, see the relevant data sheet.

Note 2: The touchscreen operation can function under either the TSC-controlled mode or the host-controlled mode, selectable by bit 15 (D15) of the {ADC} register. Under the TSC-controlled mode, the analog-to-digital conversion automatically starts whenever the panel is touched. Under the host-controlled mode, the conversion starts only by command of the host processor. Therefore, under the TSC mode, the /DAV pin goes low when the panel has been touched. However, under the host mode, the /DAV pin is **NOT** active by just touching the panel.

Data Reading Sequence

After all of the request data are converted and ready for reading, the /DAC goes active (logic low) to notify the host processor. At this point, the host processor proceeds to read all data from the data registers located in page 0 of the TSC memory space.

Tip 3: All data converted have to be read from the TSC. Otherwise, no further analog-to-digital conversion can occur, and the /DAV cannot be reset (logic high). Therefore, monitor the /DAV line to see if all data have been read. If the /DAV stays low, it indicates that the data have not been properly read.

Figure 2 shows the data reading sequence. The touch or nontouch mode is defined and selected by bit 10 to bit 13 of the {ADC} register. To convert to X, Y, and Z is called the touch mode, and to convert to BAT1, BAT2, AUX1, AUX2, and temperature is called the nontouch mode.

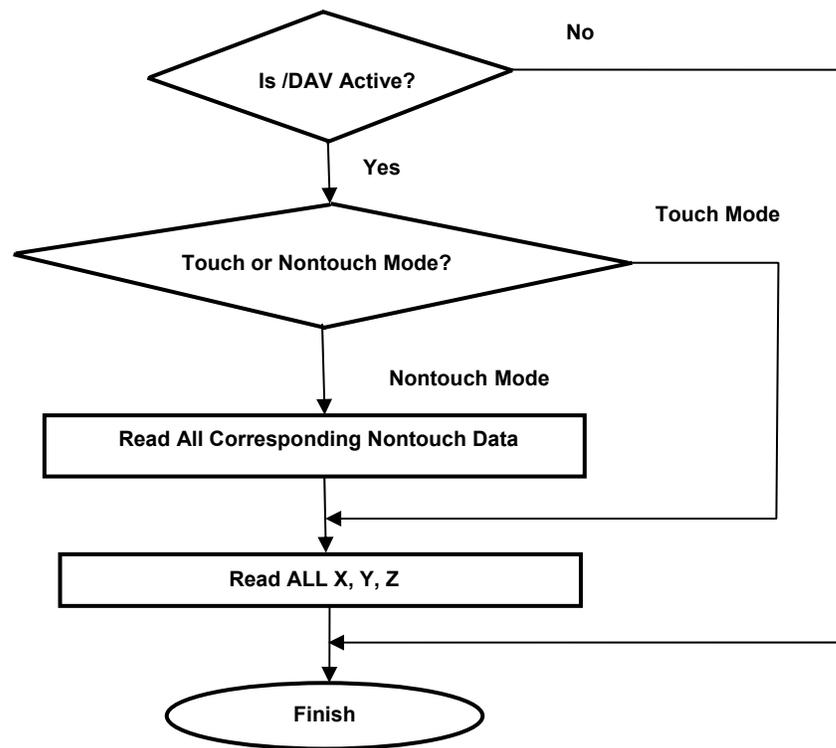


Figure 2. TSC Data Reading Sequence

Example 2: Read the X, Y, and Z (Z1 and Z2) data back for the TSC initialized in Example 1.

When /DAV goes active, the following occurs:

1. /SS goes low.
2. Write to TSC (MOSI pin) to ask to read page 0 and start at address 0x00 { 0x8000, 0x0000, 0x0000, 0x0000 }
3. In the meantime, read from TSC (MISO pin) the X, Y, Z1, and Z2 positions {Not Care, X, Y, Z1, and Z2 }
4. /SS goes back high. (/DAV also returns to logic high.)

Note 3: Unlike the writing operation, the SPI interface on the TSC2000/2200/230x does support multiple readings. That is, after /SS line goes low, the first 16-bit command word on the MOSI pin tells the starting address to begin reading from; the second 16 bits on the MISO are the data from the TSC register. Then, the address automatically increases to point to the next TSC register address until /SS returns to logic high.

Note 4: After the /SS goes low, ignore the first 16-bit word from the TSC (on the MISO line). The second to the rest of the 16-bit words being sent to the TSC (on the MOSI) could be any 16-bit word because all are ignored by the TSC.

Tip 4: Under the touch measurement, for example “X” or “X, Y”, the /DAV would NOT reset if ONLY the “X” or “X, Y” were read back. All “X, Y, Z” must be read back so as to reset the /DAV and make the ADC ready for the next conversion.

Tip 5: After reading back the required data, if the /DAV does not reset (goes back “high”), implement an X, Y, Z1, Z2 reading as described in Example 2 to reset the /DAV.

Tip 6: It is more efficient to use the /DAV pin, rather than the /PENIRQ, as the hardware interrupt to trigger and call the data reading routine or ISR.

Mode Changing Sequence

When shifting or changing from the TSC-controlled to host-controlled mode or vice versa (see the foregoing Note 2 or the relevant data sheet), the proper coding sequence can help to shorten the delay and to retrieve the correct data.

Example 3: The application requires that the TSC be in the TSC and XYZ mode most of the time, and every ½ second the BAT1 should be sampled so as to continuously monitor the battery status.

1. Thus, the power-up initialization should put the TSC into the TSC XYZ mode, as discussed in Example 1.

2. Then, every 500 ms, to change from X, Y, Z1, and Z2 to BAT1, the following sequence can be used:

```
/*******/
```

(Under the TSC and “X, Y, Z” mode):

```
TSC Write Register { 0x0800, 0xC8E6 }; // to stop ADC conversion
```

```
Check /DAV and make sure /DAV back high (in case not, read “X, Y, Z1, Z2”)
```

```
TSC Write Register { 0x0800, 0x18E6 }; // put the ADC to host and BAT1 mode
```

```
*****/
```

3. Read back the BAT1 data when /DAV low. See Figure 2

4. After reading BAT1, shift back to X, Y, Z1, and Z2 mode by using the following sequence:

```
/*******/
```

(Under the HOST and BAT1 mode):

```
Check /DAV and make sure that /DAV is back to high (if not, read X, Y, Z1, and Z2)
```

```
TSC Write Register { 0x0800, 0x88E6 }; // put the ADC to TSC and XYZ mode
```

```
/*******/
```

Tip 7: When shifting from a host to a TSC mode, the ADC does not need to be stopped. But when shifting from TSC to host mode, it is better to stop the ADC first.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		Applications	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments
Post Office Box 655303 Dallas, Texas 75265

Copyright © 2004, Texas Instruments Incorporated