



Embedded Computer Systems

Touch Panel Driver Software Specification

Specification: 110025-10062

Specification Version: 1.1

Last Revision: September 8, 2005

Introduction

Touch panels are clear plates placed in front of a graphical display that are used to detect user input. Most ADS single-board computer products include an interface to drive analog-resistive touch panels.

This document describes how the Windows CE driver controls the touch panel. It also describes the registry settings that affect touch panel driver operation.

Theory of Operation

Four-wire analog-resistive touch panels consist of two layers of conductive plastic adhered to a clear, rigid substrate (glass or Plexiglas). The two layers are separated by small, clear spacers. One layer of the touch panel returns information about the x-axis location of user touches, while the other returns y-axis information. When pressure is applied to the surface of the touch panel, the touch panel plates act electrically like a variable resistor.

The touch panel drive sequence starts when a user touches the panel. This action drops the resistance of the touch panel and generates an interrupt to the CPU via the touch panel control circuit. When the touch panel generates the initial interrupt, the driver waits a period of time, and then samples to confirm that the panel is still being touched. If not, it delays a period of time before re-enabling to filter out spurious data.

The CPU then uses the touch panel control circuit to energize one pair of wires (X or Y), then read the voltage back on the other. Only one axis can be read at a time, so the controller alternates energizing the X and Y axes.

Driver Logic Diagram

The following flow diagram describes the internal logic of the ADS touch panel driver. The key provides points in the logic that can be affected through registry settings (described in detail later). The keys are in the following format:

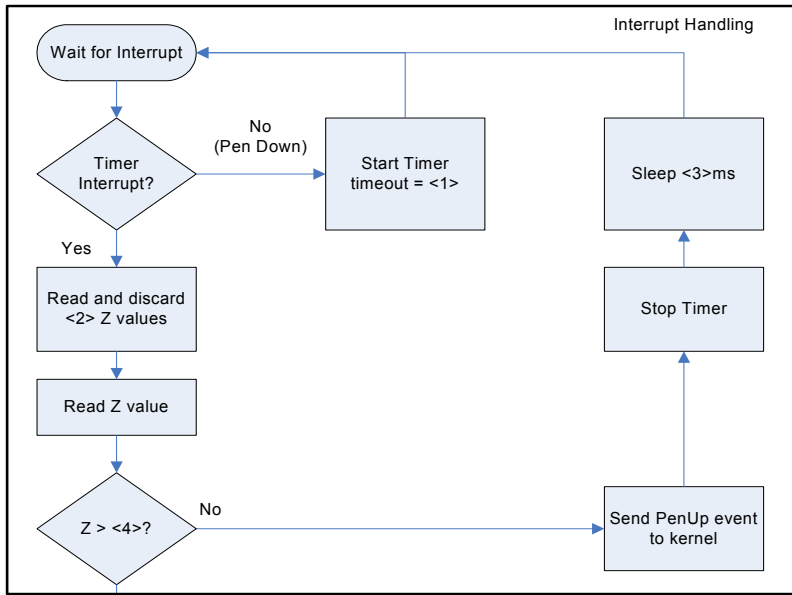
<n> RegValue : TypicalValue

Where:

<n> = substitution point in the flow chart

RegValue = the DWORD registry key value that can be used to override the default value. The value must be placed under the [HKEY_LOCAL_MACHINE\Drivers\Touch] key

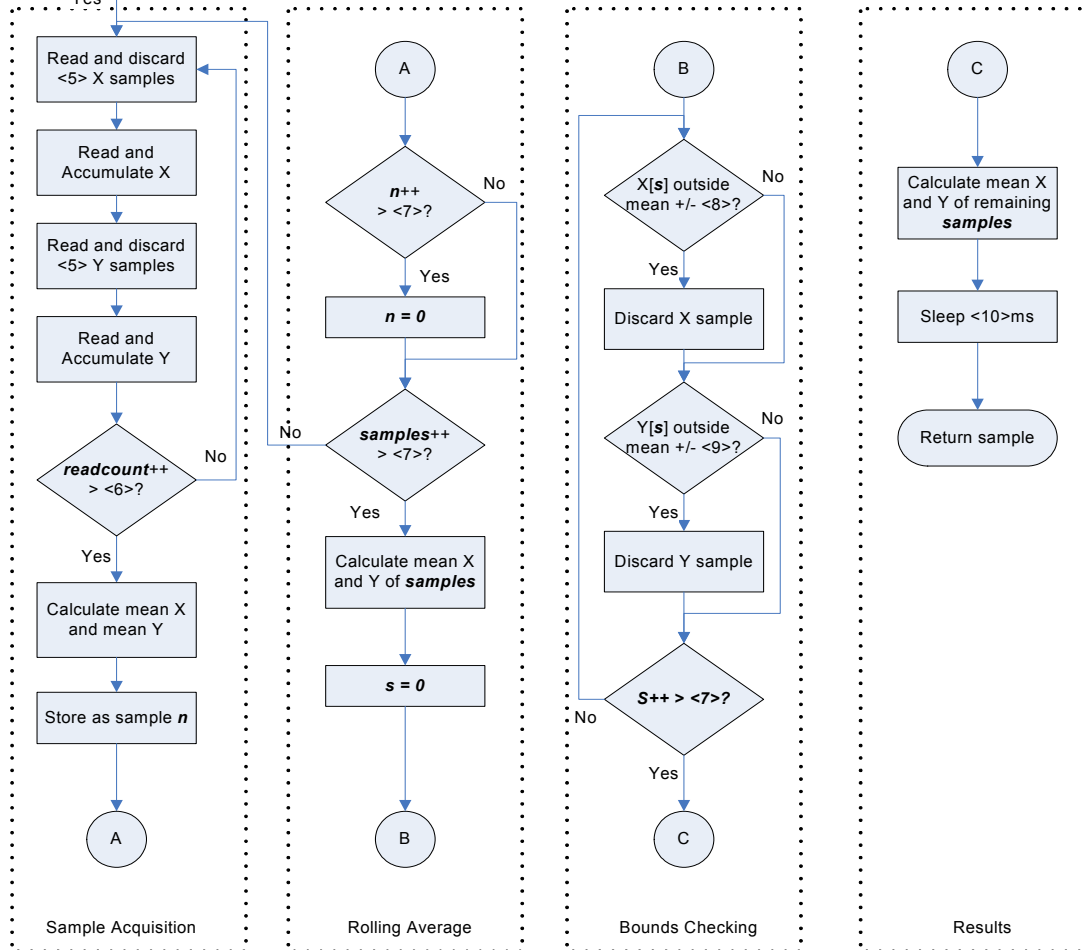
TypicalValue = a typical value (BitsyX default value)



ADS Touchscreen Driver Logic Specification 1.0

Registry Adjustable Values

- <0> SerialClockRate : 1
- <1> SamplePeriod : 300000
- <2> ReadsBeforePenDown : 1
- <3> InterruptReenableDelay : 10
- <4> ZThreshold : 100
- <5> LeadingReadsToIgnore : 10
- <6> ReadsPerSample : 4
- <7> SamplesPerPoint : 8
- <8> DeltaXCoordTolerance : 16
- <9> DeltaYCoordTolerance : 16
- <10> InterSampleDelay : 0



Using the Driver

For Windows CE 4.x devices that support touch panels drivers that meet spec version 1.0 or later, the driver must be explicitly enabled. That is to say that by default All ADS Windows CE 4.x images will use the pre-1.0 touch panel driver, even if the new driver binary is included in the image.

To enable the new driver, the following registry value must be set:

```
[HKEY_LOCAL_MACHINE\HARDWARE\DEVICEMAP\TOUCH]
    "Drivername"="touch.dll"
```

For Windows CE 5.0 devices that support touch panels drivers that meet spec version 1.0 or later, the driver is enabled by default. That is to say that by default All ADS Windows CE 5.0 images will use a 1.0 or later touch panel driver.

To enable the legacy driver, the following registry value must be set:

```
[HKEY_LOCAL_MACHINE\HARDWARE\DEVICEMAP\TOUCH]
    "Drivername"="pre10touch.dll"
```

Registry Settings

The following registry values can be used to modify the touch panel driver behavior. All are values under the [HKEY_LOCAL_MACHINE\Drivers\TOUCH] registry key. Adjusting these keys will usually impact multiple behaviors. Optimal values will vary depending on the characteristics of the specific touch panel used and trade-offs are often necessary to achieve desired performance characteristics.

NoTouchScreen (dword)

This value can be used to indicate to the system that no touch screen is installed on the device. When this value is non-zero the touch panel controller interrupt is disabled and the calibration screen will be suppressed at device startup.

SerialClockRate (dword)

This value sets the rate at which the system communicates with the touch panel controller (through an SPI interface). For most ADS products this value represents a clock divisor that sets the rate for the SPI bus. The actual rate is calculated by the following formula:

$$1,843,200 / (\text{SerialClockRate} + 1)$$

Value Effects

Low Value: Faster response
High Value: Less resource usage

SamplePeriod (dword)

This value sets the period interval in which the driver does sampling.

Value Effects

Low Value: Faster response
High Value: Less resource usage

ReadsBeforePenDown (dword)

This value gives the number of reads that will be thrown away before looking for a pen-down condition. This value provides the driver the ability to let the A/D values settle before actually reading

Value Effects

Low Value: Faster response, increased noise in pen-down detection
High Value: Slower response, more consistent pen-down reaction

InterruptReenableDelay (dword)

This value sets the delay after a pen up is detected and when the touch interrupt is re-enabled.

Value Effects

Low Value: Spurious interrupts can occur during pen up
High Value: Decreases double-tap responsiveness

ZThreshold (dword)

This is the A/D value that must be exceeded when the Z value is read to be considered a “pen down” condition. This is not equivalent to pressure, as the Z value for the same pressure will vary widely from corner to corner on the device.

Value Effects

Low Value: Spurious or continual interrupts without touching the panel
High Value: Loss of touch reaction in high resistance

LeadingReadsToIgnore (dword)

This value gives the number of reads that will be thrown away before actual sampling of the X and Y values from the panel. This value provides the driver the ability to let the A/D values settle before actually reading.

Value Effects

Low Value: Faster response, decreased ability to get reads at the high-resistance edges of the panel
High Value: More consistent edge reads, higher load on overall system, decreases double-tap performance

ReadsPerSample (dword)

This value sets the number of times the A/D is read to calculate a mean X or Y value in any *one* given interrupt period. The calculated means are then saved as a single point sample. This value is used for “fine” adjustments to touch position noise.

Value Effects

Low Value: Increases touch position noise, lower load on overall system
High Value: Decreases touch position noise, higher load on overall system

SamplesPerPoint (dword)

This value sets the number of samples that must be collected before they will be reported back to the driver as a touch position. This value is effectively the number of sample periods (see SamplePeriod value description) that must pass before the first touch is recognized. These values are reported back to the system using a rolling average, so once the buffer is filled after a pen down, collection of only a single sample is required before reporting back to the kernel. Increasing this value has less system impact than ReadsPerSample but has a more noticeable touch panel performance impact. This value is used for “coarse” adjustments to touch position noise.

Value Effects

Low Value: Increases touch position noise, lower load on overall system
High Value: Decreases double-tap responsiveness, hysteresis in drag motions, and higher load on overall system. Decreases double-tap performance

DeltaXCoordTolerance (dword)

This value gives the number variation tolerance of X position reads. When averaging samples to get a touch position, values that are found to be more than this value away from the mean of all samples in the buffer are ignored.

Value Effects

- Low Value: Filters out data point outliers of noise, increases local position noise, poor performance with large “stylus” (i.e. finger)
- High Value: Lower local position noise, allows data point outliers to affect position

DeltaYCoordTolerance (dword)

This value gives the number variation tolerance of Y position reads. When averaging samples to get a touch position, values that are found to be more than this value away from the mean of all samples in the buffer are ignored.

Value Effects

- Low Value: Filters out data point outliers of noise, increases local position noise, poor performance with large “stylus” (i.e. finger)
- High Value: Lower local position noise, allows data point outliers to affect position

InterSampleDelay (dword)

This value gives the number of milliseconds to sleep after acquiring a data sample

Value Effects

- Low Value: More responsive touch performance, higher load on overall system
- High Value: Less responsive touch performance, lower load on overall system

FlushRegOnCalibration (dword)

When non zero, the driver will call RegFlush after calibration to persist the device registry. This will prevent the calibration screen from being display on subsequent boots unless the persistent registry is erased. The default value for this key is TRUE (1).

Behavior

This section describes expected behaviors exhibited by the driver.

Calibration

At device boot, the driver will attempt to read calibration data from the following registry key value:

```
[HKEY_LOCAL_MACHINE\HARDWARE\DEVICEMAP\TOUCH]
    "CalibrationData"="{data depends on last calibration}"
```

If this value is missing or invalid, the calibration screen will be displayed. It is important to note that if the registry is not persisted, the calibration screen will always appear after a warm or cold boot of the device. See the *FlushRegOnCalibration* registry key detail for more information.

Implementation Matrix

The following table illustrates the ADS run-time image in which each version of the touch panel driver was included. For example, version 1.0 of the driver was first included in VGX image 4.20.23.05

ADS Product	Touch Panel Specification	
	v1.0	V1.1
BitsyX	4.20.16.03	4.20.18
AGX	4.20.15	4.20.25
VGX	4.20.23.05	4.20.32
GCX	4.20.11	4.20.12

Document History

The following list summarizes the changes made between releases of this document.

REV	DESCRIPTION	BY
1	Preliminary version for pre-specification drivers	2/9/05 TZK/ak
2	Internal release of Spec 1.0	3/2/05 ct
	Updates and clarifications based on reviewer comments	3/10/05 ct
3	Added FlushRegOnCalibration reg key and Behavior section	3/16/05 ct
4	Added NoTouchScreen reg key and behavior	9/8/05 ct

Specification History

The following list summarizes the changes made between versions of the specification.

REV	DESCRIPTION	BY
1.0	Preliminary release.	3/10/05 ct
	Public Release	3/17/05 ct
1.1	Public Release	9/8/05 ct