x6100 - Touchpad Raw XY

Version 0

Touchpad Raw XY

[0] **GetTouchpadInfo**() → XSize, YSize, ZDataRange, AreaDataRange, TimeStampUnits, MaxFingerCount, …
[1] **GetRawReportState**() → reportBitmap
[2] **SetRawReportState**(reportBitmap)
[event0] **DualXYData**() → TSTAMP, CPT, X, Y, CTS, Z, FORCE , AREA, FID, BTN, EOF, NUMFING

Overview

The Touchpad raw data feature describes the parameters and possible raw report formats of a touchpad. If the raw reporting is turned on, then any keyboard/mouse reporting is automatically turned off.

Functions and Events

[0] GetTouchpadInfo() → XSize, YSize, ZDataRange, AreaDataRange, TimeStampUnits, MaxFingerCount, …

This function returns a structure describing the characteristics of the touchpad.

Parameters

none

Returns

XSize, YSize

The extent of the touch pad coordinates, in native resolution.

ZDataRange

0x00 means no range, 0x0f means 16-bit.

AreaDataRange

0x0f means 16-bit.

TimeStampUnits

Number of 0.1 milliseconds per timestamp increment.

MaxFingerCount

Maximum number of fingers that can be tracked.

Origin

Position of the origin, 0x00 is reserved, 0x01 = LOWER-LEFT, 0x02 = LOWER-RIGHT, 0x03 = UPPER-LEFT, 0x04 = UPPER-RIGHT. The corners are defined by looking at the device from above, with the lower edge towards the user and the upper facing the PC screen

PenSupport

0x00 = no support, 0x01 = support.

RawReportMappingVersion

For example 0x02 (see below).

Table 1. GetTouchpadInfo() response packet format

byte \ bit	7	6	5	4	3	2	1	0
0	XSize [15:8]							
1	XSize [7:0]							
2	YSize [15:8]							
3	YSize [7:0]							
4	ZDataRange							
5	AreaDataRange							
6	TimeStampUnits							
7	MaxFingerCount							
8	Origin							
9	PenSupport							
10..11	reserved							
12	RawReportMappingVersion							
13	DPI (MSB)							
14	DPI (LSB)							
15	reserved							

[1] GetRawReportState() → reportBitmap

Returns a bitmap, indicating the currently selected state for reporting.

Parameters

none

Returns

reportBitmap

bitmap of report setting

```
bit 0 : RAW
        0 = raw report disabled, 1 = raw reporting enabled
bit 1 : FA
        0 = do not add force data, 1 = add force data to 16-bit reporting (deprecated)
bit 2 : ENH
        0 = enhanced reporting disabled, 1 = enhanced reporting enabled
bit 3 : WH
        0 = do not report Width/Height, 1 = report Width/Height instead of Area
bit 4 : NG
        0 = report of native gesture disabled, 1 = reporting of native gestures
enabled
bit 5 : MM
        0 = reporting of Major/Minor/Orientation disabled, 1 = reporting of
Major/Minor/Orientation enabled
bit 6 : WH8
        0 = do not report Width and Height bytes, 1 = report Width and Height bytes
instead of Area
bit 7 :  reserved
```

Table 2. GetRawReportState() response packet format

byte \ bit	7	6	5	4	3	2	1	0
0	reportBitmap							
	reserved	WH8	MM	NG	WH	ENH	FA	RAW
1..15	reserved							

NOTE

Some of these state bits are mutually exclusive. In particular, only the following combinations should be used:
0x00 = no raw reporting.
0x05 = raw reporting, with area being reported as a 16-bit value.
0x09 = raw reporting enabled, force is zero and area is width/height in 4 bits each.
0x21 = raw reporting, with major/minor and orientation.
0x41 = raw reporting, with 8-bit width and 8-bit height.

[2] SetRawReportState(reportBitmap)

Returns a bitmap, indicating the currently selected state for reporting.

Parameters

reportBitmap

bitmap of report setting

```
bit 0 : RAW
        0 = raw report disabled, 1 = raw reporting enabled
bit 1 : FA
        0 = do not add force data, 1 = add force data to 16-bit reporting (deprecated)
bit 2 : ENH
        0 = enhanced reporting disabled, 1 = enhanced reporting enabled
bit 3 : WH
        0 = do not report Width/Height, 1 = report Width/Height instead of Area
bit 4 : NG
        0 = report of native gesture disabled, 1 = reporting of native gestures
enabled
bit 5 : MM
        0 = reporting of Major/Minor/Orientation disabled, 1 = reporting of
Major/Minor/Orientation enabled
bit 6 : WH8
        0 = do not report Width and Height bytes, 1 = report Width and Height bytes
instead of Area
bit 7 :  reserved
```

Table 3. SetRawReportState() request packet format

byte \ bit	7	6	5	4	3	2	1	0
0	reportBitmap							
	reserved	WH8	MM	NG	WH	ENH	FA	RAW
1..15	reserved							

Returns

none

[event0] DualXYData() → TSTAMP, CPT, X, Y, CTS, Z, FORCE , AREA, FID, BTN, EOF, NUMFING

This event is sent whenever a new frame of touch data is available from the sensor. Since the frame can only describe 2 fingers at a time, multiple events are indicated if more than 2 touch points are present.
In this case, the timestamp will be the same for each event. Also, the last frame will indicate so by setting the EOF bit.

Table 4. DualXYData() response packet format

byte \ bit	7	6	5	4	3	2	1	0
0	TSTAMP [15:8]							
1	TSTAMP [7:0]							
2	CPT1		X1 [13:8]					
3	X1 [7:0]							
4	CTS1		Y1 [13:8]					

byte \ bit	7	6	5	4	3	2	1	0
5	Y1 [7:0]							
6	Z1 / FORCE1							
7	AREA1							
8	FID1				-	BTN	SP1	EOF
9	CPT2		X2 [13:8]					
10	X2 [7:0]							
11	CTS2		Y2 [13:8]					
12	Y2 [7:0]							
13	Z2 / FORCE2							
14	AREA2							
15	FID2				NUMFING			

TSTAMP

A running timestamp for the touch frame. Note that the value is encoded in Big-Endian

CTP1, CTP2

Contact Type for touch 1/2. 0b00 = finger, 0b01/0b10/0b11 = reserved.

X1, Y1, X2, Y2

The coordinates in device units of the center of the touch point.

CTS1, CTS2

Contact Status for touch 1/2. 0b00 = hover, 0b01 = touch, 0b10/0b11 = reserved.

Z1, Z2

The Z coordinate of the touch point. This is roughly the distance of the finger from the surface.

FORCE1, FORCE2

TBD

AREA1, AREA2

The area of the touch point in arbitrary units.

FID1, FID2

A unique finger ID per touch point.

BTN

Indicates whether the pysical switch underneath the touch surface is being pressed or not. 0b0 = not pressed, 0b1 = pressed.

EOF

End of frame flag. 0b0 = there are more reports for this frame that follow, 0b1 = this is the last

event for this frame.

NUMFING

The total number of fingers in the frame.

Examples

Depending on the state bits that are set in addition to turn on the raw mode, the following changes can be expected in the DualXYReport.

If bit FORCE16BITS (0x02) is set, then the force value is reported in bytes 6, 7 and 13, 14 like so:

byte \ bit	7	6	5	4	3	2	1	0
0	TSTAMP [15:8]							
1	TSTAMP [7:0]							
2	CPT1		X1 [13:8]					
3	X1 [7:0]							
4	CTS1		Y1 [13:8]					
5	Y1 [7:0]							
6	FORCE1 [15:8]							
7	FORCE1 [7:0]							
8	FID1				-	BTN	SP1	EOF
9	CPT2		X2 [13:8]					
10	X2 [7:0]							
11	CTS2		Y2 [13:8]					
12	Y2 [7:0]							
13	FORCE2 [15:8]							
14	FORCE2 [7:0]							
15	FID2				NUMFING			

if bit ENHANCED_SETTINGS (0x04) is set, then the force is not reported and the are is calculated in some arbitrary units:

byte \ bit	7	6	5	4	3	2	1	0
0	TSTAMP [15:8]							
1	TSTAMP [7:0]							
2	CPT1		X1 [13:8]					
3	X1 [7:0]							
4	CTS1		Y1 [13:8]					

byte \ bit	7	6	5	4	3	2	1	0
5	Y1 [7:0]							
6	0x00							
7	AREA1 [7:0]							
8	FID1				-	BTN	SP1	EOF
9	CPT2		X2 [13:8]					
10	X2 [7:0]							
11	CTS2		Y2 [13:8]					
12	Y2 [7:0]							
13	0x00							
14	ARE2 [7:0]							
15	FID2				NUMFING			

If bit WIDTH_HEIGHT (0x08) is set, then the force is reported as 8-bit and the area of the touch is split into two 4-bit values for width and height:

byte \ bit	7	6	5	4	3	2	1	0
0	TSTAMP [15:8]							
1	TSTAMP [7:0]							
2	CPT1		X1 [13:8]					
3	X1 [7:0]							
4	CTS1		Y1 [13:8]					
5	Y1 [7:0]							
6	FORCE1 [7:0]							
7	HEIGHT1[3:0]				WIDTH1[3:0]			
8	FID1				-	BTN	SP1	EOF
9	CPT2		X2 [13:8]					
10	X2 [7:0]							
11	CTS2		Y2 [13:8]					
12	Y2 [7:0]							
13	FORCE2 [7:0]							
14	HEIGHT2[3:0]				WIDTH2[3:0]			
15	FID2				NUMFING			

If bit MAJOR_MINOR_ORIENTATION (0x20) is set, then the entire report is packed more tightly and the ellipse data for a touch is reported with the major, minor and orientation values.

byte \ bit	7	6	5	4	3	2	1	0
0	TSTAMP [15:8]							
1	TSTAMP [7:0]							
2	X1 [11:4]							
3	X1 [3:0]				Y1 [11:8]			
2	Y1 [7:0]							
5	Z1 [7:0]							
6	MAJOR1 [5:0]						MINOR1 [5:4]	
3	MINOR1 [3:0]				ORIENT1 [3:0]			
8	FID1				-	BTN	-	EOF
9	X2 [11:4]							
10	X2 [3:0]				Y2 [11:8]			
11	Y2 [7:0]							
12	Z2 [7:0]							
13	MAJOR2 [5:0]						MINOR2 [5:4]	
14	MINOR2 [3:0]				ORIENT2 [3:0]			
15	FID2				NUMFING			

MAJOR1, MAJOR2, MINOR1, MINOR2

The major and minor axes of the ellipse of the touch point. Arbitrary units are used for these values. If the major value is bigger than the minor value, the angle/orientation is interpreted clockwise. If the major value is smaller than the minor value, then the orientation is interpreted counter-clockwise

ORIENT1, ORIENT2

The angle of the major axis to the vertical axis. The angle is mapped from 0..15 to 0..90 degrees.

If bit WIDTH8_HEIGHT8 (0x40) is set, then the the area of the touch is split into two 8-bit values for width and height

byte \ bit	7	6	5	4	3	2	1	0
0	TSTAMP [15:8]							
1	TSTAMP [7:0]							
2	CPT1		X1 [13:8]					
3	X1 [7:0]							
4	CTS1		Y1 [13:8]					
5	Y1 [7:0]							
6	WIDTH1							

byte \ bit	7	6	5	4	3	2	1	0
7	HEIGHT1							
8	FID1				-	BTN	SP1	EOF
9	CPT2		X2 [13:8]					
10	X2 [7:0]							
11	CTS2		Y2 [13:8]					
12	Y2 [7:0]							
13	WIDTH2							
14	HEIGHT2							
15	FID2				NUMFING			

ChangeLog

- Version 0: Initial version