x2150 - Thumbwheel

Version 0

Thumbwheel

Thumbwheel reporting

[0] **getThumbwheelInfo**() → info, capabilities
[1] **getThumbwheelStatus**() → reporting mode, inv_dir, touch, proxy
[2] **setThumbwheelReporting**(reporting mode, inv_dir)
[event0] **thumbwheelEvent**() → rotation, time stamp, rotation status, touch, proxy, single tap

Overview

A thumbwheel generates natively HID reports that are usually horizontal scroll events.
This feature gives the possibility to divert these reports (that is to send them in HID++ format instead) or to invert the rotation direction.
Additional info is available in the diverted reports, depending of the capabilities of the device, as proxy (proximity) / touch detection, single tap gesture, etc

Functions and Events

[0] getThumbwheelInfo() → info, capabilities

The function returns various characteristics of the thumbwheel as well as capabilities

Parameters

none

Returns

native_res

The number of ratchets generated by revolution when in native (HID) mode

diverted_res

The number of rotation increments generated by revolution when in diverted (HID++) mode

default_dir

Original (not inverted) rotation direction
0: positive when moving to the left or back of the device
1: positive when moving to the right or front of the device

c_time_stamp

Time stamp capability (1 = supported)
Elapsed time since last report, the unit being time_unit

c_touch

Touch capability (1 = supported)
Thumbwheels are often coupled with touch or proxy sensors, helpful for some gestures

c_proxy

Proxy capability (1 = supported)

c_single_tap

Single tap gesture capability (1 = supported)

time_unit

If time stamp is supported, it gives the unit in us (micro second); otherwise, set to 0

Table 1. getThumbwheelInfo() response packet format

byte \ bit	7	6	5	4	3	2	1	0
0	native_res (MSB)							
1	native_res (LSB)							
2	diverted_res (MSB)							
3	diverted_res (LSB)							
4	—	—	—	—	—	—	—	default_dir
5	—	—	—	—	c_single_tap	c_proxy	c_touch	c_time_stamp
6	time_unit (MSB)							
7	time_unit (LSB)							
8..15	reserved							

[1] getThumbwheelStatus() → reporting mode, inv_dir, touch, proxy

The function returns the thumbwheel status

Parameters

none

Returns

reporting_mode

The following values are possible:

```
Native (HID only)                = 0
Diverted (HID++ only)            = 1
```

- In **Diverted** mode, [event0] is sent in HID++

inv_dir

1 = the rotation direction is inverted (relatively to default_dir)

touch

1 = user is touching the thumbwheel

proxy

1 = user is close to the thumbwheel

Table 2. getThumbwheelStatus() response packet format

byte \ bit	7	6	5	4	3	2	1	0
0	Reporting_mode							
1	—	—	—	—	—	proxy	touch	inv_dir

[2] setThumbwheelReporting(reporting mode, inv_dir)

Set the reporting mode

Parameters

reporting_mode

The following values are possible:

```
Native (HID)                     = 0
Diverted (HID++)                 = 1
```

- In **Diverted** mode, [event0] is sent in HID++

inv_dir

1 = invert the rotation direction (relatively to default_dir). This setting applies in both native and diverted modes

Table 3. setThumbwheelReporting() request packet format

byte \ bit	7	6	5	4	3	2	1	0
0	Reporting_mode							
1	—	—	—	—	—	—	—	inv_dir

Returns

none

[event0] thumbwheelEvent() → rotation, time stamp, rotation status, touch, proxy, single tap

This event is sent whenever any information listed hereafter is updated and if in diverted mode

rotation

Relative wheel rotation (signed integer).

time_stamp

Time elapsed between the current and the previous rotation report, expressed in time_unit
time_stamp is zero if not supported or for the very first rotation report

rotation_status

It applies to wheel rotation. The following values are possible:

```
Inactive (no rotation)                = 0
Start    (first rotation report)      = 1
Active   (next rotation reports)      = 2
Stop     (release, no touch)          = 3
```

touch

1 = user is touching the thumbwheel; 0 = no touch or touch not supported

proxy

1 = user is close to the thumbwheel; 0 = user is not in proxymity or proxy not supported

single_tap

1 = single tap gesture detected; 0 = no tap gesture or not supported

Table 4. thumbwheelEvent() event packet format

byte \ bit	7	6	5	4	3	2	1	0
0	rotation (MSB)							
1	rotation (LSB)							
2	time_stamp (MSB)							
3	time_stamp (LSB)							
4	rotation_status							
5	—	—	—	—	single_tap	proxy	touch	—

ChangeLog

- Version 0: Initial version