

x1C00 persistent remappable action

Version 0

PersistentRemappableAction

This feature describes user actions available on a device and allows them to be mapped to a behaviour that will be persistent in the device. The defined behaviour can be the sending of a single HID report or an internal function (e.g. show battery status).

[0] **getFeatureInfo()** → flags

[1] **getCount()** → count, numberOfHosts

[2] **getCidInfo(index, hostIndex)** → cid

[3] **getPersistentAction(cid, hostIndex)** → cid, hostIndex, actionId, hidUsage, modifierMask, cidStatus

[4] **setPersistentAction(cid, hostIndex, actionId, hidUsage, modifierMask)**

[5] **resetPersistentAction(cid, hostIndex)**

[6] **resetToFactorySettings(hostIdMask)**

Overview

Functions are provided to enumerate controls on the device which are interesting to software. This includes any control which can be reprogrammed or should be handled by the software in some special way. A control can be a physical input (e.g. key press, button) or any user input (e.g. long press, gesture, voice, etc.).

A particular control ID shall not appear more than once in the enumerated list of control IDs. If duplicate control IDs are desired a new control ID should be created to identify the secondary instance of the control.

Once enumerated, controls can be configured to perform actions different from their default. Controls are identified by control ID. The control IDs defined in this feature are identical to the control IDs defined in the x1b04 feature. The list of Control IDs can differ from one host to the other.

The control ID is used by software to determine the possible list of options the user may assign to the control.

The last 3 functions of the API reflect the persistence aspect of this feature with which the SW can remap a control to another behaviour that will be persistent in the device.

NOTE

if the device has both features x1b04 and x1c00, and the "divert" flag on feature 0x1b04 is set for a given CtrlId, then feature x1b04 takes precedence over feature x1c00, and the x1b04 hidpp "diverted" event is sent instead of the persistently remapped action on the x1c00.

NOTE

The make and break mechanism when a modifier key is pressed along with another key should be: on press: make(modifier) + make(key)
on release: break(key) + break(modifier)
For example: Win + Q should be:
on press: make(Win) + make(Q)
on release: break(Q) + break(Win)
Code and modifier are sent withing a single report.

Functions and Events

[0] getFeatureInfo() → flags

Parameters

none

Returns

flags

16-bit unsigned representing the capabilities of the device.

```

kbd report      = (0)/1 -> x1c00 is (not) capable of sending keyboard keys
mouse buttons   = (0)/1 -> x1c00 is (not) capable of sending mouse buttons
X disp          = (0)/1 -> x1c00 is (not) capable of sending mouse X displacement
Y disp          = (0)/1 -> x1c00 is (not) capable of sending mouse Y displacement
vert roller     = (0)/1 -> x1c00 is (not) capable of sending vertical roller
increments
horz roller     = (0)/1 -> x1c00 is (not) capable of sending horizontal roller
increments (AC PAN)
consumer ctrl   = (0)/1 -> x1c00 is (not) capable of sending consumer control codes
internal        = (0)/1 -> x1c00 is (not) capable of executing internal functions
power key       = (0)/1 -> x1c00 is (not) capable of sending power keys

```

Table 1. getFeatureInfo response packet format

| byte \ bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|-----------|---------------|-------------|-------------|--------|--------|---------------|------------|
| 0 | flags MSB | | | | | | | |
| | --- | --- | --- | --- | --- | --- | --- | power key |
| 1 | flags LSB | | | | | | | |
| | internal | consumer ctrl | horz roller | vert roller | Y disp | X disp | mouse buttons | Kbd report |
| 2-15 | reserved | | | | | | | |

[1] getCount() → count, numberOfHosts

Returns the number of items in the control ID table. These are sources (user stimuli) like controls (hot keys) and extra native functions (e.g. a given gesture) that a specific action can be mapped to.

Parameters

none

Returns

count

The number of items in the control ID table.

numberOfHosts

The number of hosts supported by the device.

NOTE The number of control IDs per Host in the scope of multi host devices.

Table 2. getCount() response packet format

| byte \ bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|---------------|---|---|---|---|---|---|---|
| 0 | count | | | | | | | |
| 1 | numberOfHosts | | | | | | | |
| 2-15 | reserved | | | | | | | |

[2] getCidInfo(index, hostIndex) → cid

Returns a row from the control ID table. This table information describes capabilities and desired software handling for physical controls in the device. All control IDs will be the exact same per host.

Parameters

index

The zero based row index to retrieve.

hostIndex

8-bit unsigned representing the host number where the control ID is remapped (0 based index). When setting hostIndex to 0xff, the device shall select the current host.

Table 3. getCidInfo request packet format

| byte \ bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|-----------|---|---|---|---|---|---|---|
| 0 | index | | | | | | | |
| 1 | hostIndex | | | | | | | |

| byte \ bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|----------|---|---|---|---|---|---|---|
| 2-15 | reserved | | | | | | | |

Returns

cid

16-bit unsigned representing the control ID of the control (physical button or user action). Each control ID is unique in this list.

Table 4. *getCidInfo()* response packet format

| byte \ bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|----------|---|---|---|---|---|---|---|
| 0 | cid MSB | | | | | | | |
| 1 | cid LSB | | | | | | | |
| 2-15 | reserved | | | | | | | |

[3] **getPersistentAction(cid, hostIndex) → cid, hostIndex, actionId, hidUsage, modifierMask, cidStatus**

For a given control ID and hostIndex supported by the device, returns the corresponding actionId, hidUsage and modifierMask from the non-volatile memory in the device.

Parameters

cid

16-bit unsigned representing the control ID of the control being requested (physical button or user action).

hostIndex

8-bit unsigned representing the host number where the control ID is remapped (0 based index). When setting hostIndex to 0xff, the device shall select the current host.

Table 5. *getPersistentAction* request packet format

| byte \ bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|-----------|---|---|---|---|---|---|---|
| 0 | cid MSB | | | | | | | |
| 1 | cid LSB | | | | | | | |
| 2 | hostIndex | | | | | | | |
| 3-15 | reserved | | | | | | | |

Returns

cid

16-bit unsigned representing the control ID of the control being addressed (physical button or user action).

hostIndex

8-bit unsigned representing the host number where the control ID has to be remapped (0 based index). When setting hostIndex to 0xff, the device shall select the current host.

actionId

8-bit unsigned representing the action that has to be performed by the device.

0x01 - send keyboard/keypad report (usage page 7)

0x02 - send mouse button report (usage page 9)

0x03 - send X displacement (usage page 1, code 0x30)

0x04 - send Y displacement (usage page 1, code 0x31)

0x05 - send vertical roller/wheel displacement (usage page 1, code 0x38)

0x06 - send horizontal roller/AC pan displacement (usage page 12, code 0x0238)

0x07 - send consumer control report (usage page 12)

0x08 - execute internal function (use value parameter as function index)

0x09 - send power key report (usage page 1)

All other values are reserved for internal FW use.

value

16-bit unsigned standard HID code sent when control ID is triggered by the user. It is an unsigned HID usage code for keyboard and consumer control keys. A HID usage code or a bit-map (TBD) for mouse buttons. A signed value for all displacements. An index to the internal function within the device(e.g. show battery status).

modifierMask

8-bit unsigned, this is the HID usage value for standard modifier keys like Win, Left Shift, Right Shift on Windows, corresponding to the hid usage page above.

cidStatus

8-bit-field that contains the remapped bit (equals 0 if default behaviour is mapped to this control ID)

NOTE Modifiers will only work for keyboard reports.

Table 6. *getPersistentAction()* response packet format

| byte \ bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|-----------|---|---|---|---|---|---|---|
| 0 | cid MSB | | | | | | | |
| 1 | cid LSB | | | | | | | |
| 2 | hostIndex | | | | | | | |
| 3 | actionId | | | | | | | |
| 4 | value MSB | | | | | | | |
| 5 | value LSB | | | | | | | |

| byte \ bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|--------------|-------|---------|--------|-------|-------|---------|----------|
| 6 | modifierMask | | | | | | | |
| | R-GUI | R-Alt | R-Shift | R-Ctrl | L-GUI | L-Alt | L-Shift | L-Ctrl |
| 7 | cidStatus | | | | | | | |
| | --- | --- | --- | --- | --- | --- | --- | remapped |
| 8-15 | reserved | | | | | | | |

[4] setPersistentAction(cid, hostIndex, actionId, hidUsage, modifierMask)

For a given control ID and hostIndex supported by the device, sets the corresponding actionId, value and modifierMask to the non-volatile memory in the device.

Parameters

cid

16-bit unsigned representing the control ID of the control being addressed (physical button or user action).

hostIndex

8-bit unsigned representing the host number where the control ID has to be remapped (0 based index). When setting hostIndex to 0xff, the device shall select the current host.

actionId

8-bit unsigned representing the action that has to be performed by the device.

0x01 - send keyboard/keypad report (usage page 7)

0x02 - send mouse button report (usage page 9)

0x03 - send X displacement (usage page 1, code 0x30)

0x04 - send Y displacement (usage page 1, code 0x31)

0x05 - send vertical roller/wheel displacement (usage page 1, code 0x38)

0x06 - send horizontal roller/AC pan displacement (usage page 12, code 0x238)

0x07 - send consumer control report (usage page 12)

0x08 - execute internal function (use value parameter as function index)

0x09 - send power key report (usage page 1)

All other values are reserved for internal FW use.

value

16-bit unsigned standard HID code to be sent when control ID is triggered by the user. It is an unsigned HID usage code for keyboard and consumer control keys. A HID usage code or a bit-map (TBD) for mouse buttons. A signed value for all displacements. An index to the internal function within the device(e.g. show battery status).

modifierMask

8-bit unsigned, this is the HID usage value for standard modifier keys like Win, Left Shift, Right

Shift on Windows, corresponding to the hid usage page above.

NOTE Modifiers will only work for keyboard reports.

Table 7. *setPersistentAction request packet format*

| byte \ bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|--------------|-------|---------|--------|-------|-------|---------|--------|
| 0 | cid MSB | | | | | | | |
| 1 | cid LSB | | | | | | | |
| 2 | hostIndex | | | | | | | |
| 3 | actionId | | | | | | | |
| 4 | value MSB | | | | | | | |
| 5 | value LSB | | | | | | | |
| 6 | modifierMask | | | | | | | |
| | R-GUI | R-Alt | R-Shift | R-Ctrl | L-GUI | L-Alt | L-Shift | L-Ctrl |
| 7-15 | reserved | | | | | | | |

[5] resetPersistentAction(cid, hostIndex)

For a given control ID and hostIndex supported by the device, resets the persistent action to the default factory settings on the device.

Parameters

cid

16-bit unsigned representing the control ID of the control being addressed (physical button or user action).

hostIndex

8-bit unsigned representing the host number where the control ID has to be reset (0 based index). When setting hostIndex to 0xff, the device shall select the current host.

Table 8. *resetPersistentAction request packet format*

| byte \ bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|-----------|---|---|---|---|---|---|---|
| 0 | cid MSB | | | | | | | |
| 1 | cid LSB | | | | | | | |
| 2 | hostIndex | | | | | | | |
| 3-15 | reserved | | | | | | | |

Returns

none

[6] resetToFactorySettings(hostIdMask)

For a given hostIdMask supported by the device, resets the persistent action to the default factory settings on the device for all control IDs.

Parameters

hostIdMask

8-bit-filed representing the hosts where the control IDs has to be reset to default factory settings.

Table 9. resetToFactorySettings request packet

| byte \ bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|------------|-----|-----|-----|-----|-------|-------|-------|
| 0 | hostIdMask | | | | | | | |
| | --- | --- | --- | --- | --- | host3 | host2 | host1 |
| 1-15 | reserved | | | | | | | |

Returns

none

ChangeLog

- Version 0: Initial version