

Request #: HUTRR103
Title: Transducer Serial Number More Significant bits.
Spec Release: 1.22
Requester: Kenneth Albanowski
Company: Google Inc.

Pages Affected: Digitizers (0x0D)
Values checked: Matthew Williams (Chair)

Current Status: **Approved**
Priority:

Required Voter: Wacom
Required Voter: Logitech
Required Voter: Intel

Voting Begins: 4th August 2021
Voting Ends: 11th August 2021
Voting Result: 3-0

Summary:

Adds support for larger Transducer Serial Numbers by adding a new Usage to the Digitizers Page (0x0D) to represent (up to 32 more) significant bits of a serial number.

Background:

Styli used with current digitizers may require a serial number larger than can be described by a single HID Report Item (max unsigned 32bits).

We propose adding a new Usage that specifically represents a second (more-significant) part of the serial number, (e.g. bits 32 to 63). When used in conjunction with the original Usage ('Transducer Serial Number[0x5B]'), permits retrieval of a 33- to 64-bit Serial Number without processing any fields larger than 32 bits.

Note: There is evidence in the field of descriptors containing Transducer Serial Number fields larger than 32 bits. This is not permitted by the HID 1.11 spec (which has a 32-bit max size for a field.)

Proposal:

Add/modify to Table 16.1: Digitizers Page

Usage ID: 0x6E

Usage Name: Transducer Serial Number Part 2

Usage Types: SV

Usage ID: 0x6F

Usage Name: Reserved

Add following text in section 16.3.1: Digitizer-Specific Fields

Usage Name: Transducer Serial Number Part 2

Usage Type: SV

Description: More/most significant bits of a unique persistent identifier provided by the transducer currently in use. Expected to be used in conjunction with "Transducer Serial Number [5B]", which describes the lesser significant bits. This permits up to a 64bit (unsigned) serial number, spread across two items. Note: It is valid to have any number of bits, up to 32 (e.g. 8) for either "Transducer Serial Number Part 2" and/or "Transducer Serial Number", where the former describes the upper-bits and the latter describes the lower bits. (If combined these bits are <=32, then it is recommended to only use "Transducer Serial Number").

Sample Descriptor:

InputReport1 describes a 64bit transducer serial-number, spread across two items.

InputReport2 describes a 11bit transducer serial-number, spread across two items. (Note: While valid, there is no advantage of splitting the serial number across multiple items, a single 11bit with Usage Transducer Serial Number[91], would work equally well.)

```
0x05, 0x0D, // UsagePage(Digitizers[13])
0x09, 0x20, // UsageId(Stylus[32])
0xA1, 0x01, // Collection(Application)
0x85, 0x01, // ReportId(1)
0x09, 0x5B, // UsageId(Transducer Serial Number[91])
0x17, 0x00, 0x00, 0x00, 0x80, // LogicalMinimum(-2,147,483,648)
0x27, 0xFF, 0xFF, 0xFF, 0x7F, // LogicalMaximum(2,147,483,647)
0x95, 0x01, // ReportCount(1)
0x75, 0x20, // ReportSize(32)
0x81, 0x02, // Input(Data, Variable, Absolute)
0x09, 0x6E, // UsageId(Transducer Serial Number Part 2[110])
0x81, 0x02, // Input(Data, Variable, Absolute)
0x85, 0x02, // ReportId(2)
0x09, 0x5B, // UsageId(Transducer Serial Number[91])
0x15, 0x00, // LogicalMinimum(0)
0x25, 0x0F, // LogicalMaximum(15)
0x75, 0x04, // ReportSize(4)
0x81, 0x02, // Input(Data, Variable, Absolute)
0x09, 0x6E, // UsageId(Transducer Serial Number Part 2[110])
0x25, 0x7F, // LogicalMaximum(127)
0x75, 0x07, // ReportSize(7)
0x81, 0x02, // Input(Data, Variable, Absolute)
0x75, 0x05, // ReportSize(5)
0x81, 0x03, // Input(Constant, Variable, Absolute)
0xC0, // EndCollection()
```