

Request #: HUTRR55
Title: Sensor Batching Controls
Spec Release: 1.12
Received:
Requester: Nathan Sherman
Company: Microsoft Corporation
Phone: 425-882-8080
FAX: N/A
email: nathans@microsoft.com

Current Status: Approved
Priority: Normal
Submitted: 8 July 2015
Voting Starts: 14 August 2015
Voting Ends: 21 August 2015
Required Voter: Nathan Sherman, Microsoft (Chair)
Required Voter: Mark Lavelle, Logitech
Required Voter: Steve McGowan, Intel

Summary

This documentation defines new usages to support HID sensor devices that can buffer sensor data and deliver it to the host when the buffer becomes full.

Background

Primary mechanism that a host uses to get data from HID sensor devices is by specifying a report interval value. This is the elapsed time period for periodic input report generation. Instead of delivering input reports immediately to the host the device can instead buffer input reports and deliver it in batches. That is, we want to decouple the rate at which a device collects data and the rate at which it delivers it to the host. This can result in significant power savings in the processor, especially in the case of continuous sensing applications that must run when the processor goes to sleep. Instead of waking the processor for every single input report, data can be buffered and processed as a whole batch.

This change proposes usages for supporting such devices. We want to propose a usage for exposing a static value that the device will use to expose the maximum number of events that can be batched for a particular sensor. There is no guarantee that the device can always batch this many number of input reports, especially if the batch is being shared by multiple sensors in a HID device. Batching is done only on a best effort basis. We also want to propose a usage that the device can use to get a latency value from the host. This value indicates (along with the report interval) indicates the amount of input reports that the device should batch.

Proposal

We want add the following two new usages to the Sensors Usage Page.

Usage ID	Usage Name
0x031A	Property: Maximum FIFO Events
0x031B	Property: Report Latency

Remove the following from section 1.0, page 12:

- Omit 0x031A and 0x031B from Property: Reserved

Add the following to section 1.5, page 29

Maximum FIFO Events	SV – Indicates the maximum number of input reports that can be stored in a buffer by this sensor. The actual size can be smaller than this value since the FIFO can be shared by multiple sensors.
Report Latency	DV – Specifies the maximum latency that the host can tolerate before receiving a batch of input reports. The device should use this value (along with the report interval) to calculate the number of input reports it should batch before delivering it to the host. When the buffer becomes full, the device will deliver input reports to the host in the order in which the reports were buffered. That is, the oldest input report will be delivered first.

Add the following to section 4.1, page 74

```
#define HID_USAGE_SENSOR_PROPERTY_MAX_FIFO_EVENTS 0x0A,0x1A,0x03
#define HID_USAGE_SENSOR_PROPERTY_REPORT_INTERVAL 0x0A,0x1B,0x03
```

Add the following to section 4.2.2, page 81

```
#define HID_USAGE_SENSOR_PROPERTY_FRIENDLY_NAME 0x0A,0x01,0x03
#define HID_USAGE_SENSOR_PROPERTY_PERSISTENT_UNIQUE_ID 0x0A,0x02,0x03
#define HID_USAGE_SENSOR_PROPERTY_SENSOR_STATUS 0x0A,0x03,0x03
#define HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL 0x0A,0x04,0x03
#define HID_USAGE_SENSOR_PROPERTY_SENSOR_MANUFACTURER 0x0A,0x05,0x03
#define HID_USAGE_SENSOR_PROPERTY_SENSOR_MODEL 0x0A,0x06,0x03
#define HID_USAGE_SENSOR_PROPERTY_SENSOR_SERIAL_NUMBER 0x0A,0x07,0x03
#define HID_USAGE_SENSOR_PROPERTY_SENSOR_DESCRIPTION 0x0A,0x08,0x03
#define HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE 0x0A,0x09,0x03 //
Nary
#define HID_USAGE_SENSOR_PROPERTY_SENSOR_DEVICE_PATH 0x0A,0x0A,0x03
#define HID_USAGE_SENSOR_PROPERTY_HARDWARE_REVISION 0x0A,0x0B,0x03
#define HID_USAGE_SENSOR_PROPERTY_FIRMWARE_VERSION 0x0A,0x0C,0x03
#define HID_USAGE_SENSOR_PROPERTY_RELEASE_DATE 0x0A,0x0D,0x03
#define HID_USAGE_SENSOR_PROPERTY_REPORT_INTERVAL 0x0A,0x0E,0x03
#define HID_USAGE_SENSOR_PROPERTY_CHANGE_SENSITIVITY_ABS 0x0A,0x0F,0x03
#define HID_USAGE_SENSOR_PROPERTY_CHANGE_SENSITIVITY_RANGE_PCT 0x0A,0x10,0x03
#define HID_USAGE_SENSOR_PROPERTY_CHANGE_SENSITIVITY_REL_PCT 0x0A,0x11,0x03
#define HID_USAGE_SENSOR_PROPERTY_ACCURACY 0x0A,0x12,0x03
#define HID_USAGE_SENSOR_PROPERTY_RESOLUTION 0x0A,0x13,0x03
#define HID_USAGE_SENSOR_PROPERTY_RANGE_MAXIMUM 0x0A,0x14,0x03
#define HID_USAGE_SENSOR_PROPERTY_RANGE_MINIMUM 0x0A,0x15,0x03
#define HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE 0x0A,0x16,0x03 // Nary
#define HID_USAGE_SENSOR_PROPERTY_SAMPLING_RATE 0x0A,0x17,0x03
#define HID_USAGE_SENSOR_PROPERTY_RESPONSE_CURVE 0x0A,0x18,0x03
#define HID_USAGE_SENSOR_PROPERTY_POWER_STATE 0x0A,0x19,0x03 // Nary
#define HID_USAGE_SENSOR_PROPERTY_MAX_FIFO_SAMPLES 0x0A,0x1A,0x03
#define HID_USAGE_SENSOR_PROPERTY_REPORT_LATENCY 0x0A,0x1B,0x03
```

Add the following to section 4.2.2, page 82

```
HID_USAGE_SENSOR_PROPERTY_REPORT_INTERVAL,
HID_LOGICAL_MIN_8(0),
```

```

HID_LOGICAL_MAX_32(0xFF,0xFF,0xFF,0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(1),
// HID_USAGE_SENSOR_UNITS_MILLISECOND,
HID_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),

HID_USAGE_SENSOR_PROPERTY_MAX_FIFO_SAMPLES,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_32(0xFF,0xFF,0xFF,0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),

HID_USAGE_SENSOR_PROPERTY_REPORT_LATENCY,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_32(0xFF,0xFF,0xFF,0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(1),
// HID_USAGE_SENSOR_UNITS_MILLISECOND,
HID_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),

```

Sample Report Descriptor

A sample report descriptor of a 3D accelerometer sensor that supports batching.

```

// 3D Accelerometer that supports batching
const unsigned char accel3_report_descriptor[] = {
HID_USAGE_PAGE_SENSOR,
HID_USAGE_SENSOR_TYPE MOTION_ACCELEROMETER_3D,
HID_COLLECTION(Physical),
//feature reports (xmit/receive)
HID_USAGE_PAGE_SENSOR,
HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
HID_REPORT_SIZE(8),
HID_REPORT_COUNT(1),
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE_NO_EVENTS,
    HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE_ALL_EVENTS,
    HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE_THRESHOLD_EVENTS,
    HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE_NO_EVENTS_WAKE,
    HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE_ALL_EVENTS_WAKE,
    HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE_THRESHOLD_EVENTS_WAKE,
HID_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_PROPERTY_SENSOR_STATUS,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_32(0xFF,0xFF,0xFF,0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(1),
HID_FEATURE(Data_Var_Abs), // up to VT_UI4 worth of status info
HID_USAGE_SENSOR_PROPERTY_REPORT_INTERVAL,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_32(0xFF,0xFF,0xFF,0xFF),

```

```

HID_REPORT_SIZE(32),
HID_REPORT_COUNT(1),
// HID_USAGE_SENSOR_UNITS_MILLISECOND,
HID_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE, // NArY
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(2),
HID_REPORT_SIZE(8),
HID_REPORT_COUNT(1),
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_PROPERTY_CONNECTION_TYPE_PC_INTEGRATED,
    HID_USAGE_SENSOR_PROPERTY_CONNECTION_TYPE_PC_ATTACHED,
    HID_USAGE_SENSOR_PROPERTY_CONNECTION_TYPE_PC_EXTERNAL,
HID_FEATURE(Const_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_PROPERTY_CHANGE_SENSITIVITY_ABS,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF,0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
// HID_USAGE_SENSOR_UNITS_G,
HID_UNIT_EXPONENT(0x0E), // scale default unit Gs to "centi-Gs" to provide 2
digits past Gs decimal point
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_MOTION_ACCELERATION,HID_USAGE_SEN
SOR_DATA_MOD_MAX),
HID_LOGICAL_MIN_16(0x01,0x80), // LOGICAL_MINIMUM (-32767)
HID_LOGICAL_MAX_16(0xFF,0x7F), // LOGICAL_MAXIMUM (32767)
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
// HID_USAGE_SENSOR_UNITS_G,
HID_UNIT_EXPONENT(0x0E), // scale default unit Gs to "centi-Gs" to provide 2
digits past Gs decimal point
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_MOTION_ACCELERATION,HID_USAGE_SEN
SOR_DATA_MOD_MIN),
HID_LOGICAL_MIN_16(0x01,0x80), // LOGICAL_MINIMUM (-32767)
HID_LOGICAL_MAX_16(0xFF,0x7F), // LOGICAL_MAXIMUM (32767)
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
// HID_USAGE_SENSOR_UNITS_G,
HID_UNIT_EXPONENT(0x0E), // scale default unit Gs to "centi-Gs" to provide 2
digits past Gs decimal point
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MAX_FIFO_SAMPLES,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_32(0xFF,0xFF,0xFF,0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_REPORT_LATENCY,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_32(0xFF,0xFF,0xFF,0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(1),

```

```

// HID_USAGE_SENSOR_UNITS_MILLISECOND,
HID_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs)

//input reports (transmit)
HID_USAGE_PAGE_SENSOR,
HID_USAGE_SENSOR_STATE,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(6),
HID_REPORT_SIZE(8),
HID_REPORT_COUNT(1),
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_STATE_UNKNOWN,
    HID_USAGE_SENSOR_STATE_READY,
    HID_USAGE_SENSOR_STATE_NOT_AVAILABLE,
    HID_USAGE_SENSOR_STATE_NO_DATA,
    HID_USAGE_SENSOR_STATE_INITIALIZING,
    HID_USAGE_SENSOR_STATE_ACCESS_DENIED,
    HID_USAGE_SENSOR_STATE_ERROR,
HID_INPUT(Const_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(16),
HID_REPORT_SIZE(8),
HID_REPORT_COUNT(1),
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_EVENT_UNKNOWN,
    HID_USAGE_SENSOR_EVENT_STATE_CHANGED,
    HID_USAGE_SENSOR_EVENT_PROPERTY_CHANGED,
    HID_USAGE_SENSOR_EVENT_DATA_UPDATED,
    HID_USAGE_SENSOR_EVENT_POLL_RESPONSE,
    HID_USAGE_SENSOR_EVENT_CHANGE_SENSITIVITY,
    HID_USAGE_SENSOR_EVENT_MAX_REACHED,
    HID_USAGE_SENSOR_EVENT_MIN_REACHED,
    HID_USAGE_SENSOR_EVENT_HIGH_THRESHOLD_CROSS_UPWARD,
    HID_USAGE_SENSOR_EVENT_HIGH_THRESHOLD_CROSS_DOWNWARD,
    HID_USAGE_SENSOR_EVENT_LOW_THRESHOLD_CROSS_UPWARD,
    HID_USAGE_SENSOR_EVENT_LOW_THRESHOLD_CROSS_DOWNWARD,
    HID_USAGE_SENSOR_EVENT_ZERO_THRESHOLD_CROSS_UPWARD,
    HID_USAGE_SENSOR_EVENT_ZERO_THRESHOLD_CROSS_DOWNWARD,
    HID_USAGE_SENSOR_EVENT_PERIOD_EXCEEDED,
    HID_USAGE_SENSOR_EVENT_FREQUENCY_EXCEEDED,
    HID_USAGE_SENSOR_EVENT_COMPLEX_TRIGGER,
HID_INPUT(Const_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_MOTION_ACCELERATION_X_AXIS,
HID_LOGICAL_MIN_16(0x01,0x80), // LOGICAL_MINIMUM (-32767)
HID_LOGICAL_MAX_16(0xFF,0x7F), // LOGICAL_MAXIMUM (32767)
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
// HID_USAGE_SENSOR_UNITS_G,
HID_UNIT_EXPONENT(0x0E), // scale default unit Gs to "centi-Gs" to provide 2
digits past Gs decimal point
HID_INPUT(Const_Var_Abs),
HID_USAGE_SENSOR_DATA_MOTION_ACCELERATION_Y_AXIS,
HID_LOGICAL_MIN_16(0x01,0x80), // LOGICAL_MINIMUM (-32767)

```

```
HID_LOGICAL_MAX_16(0xFF,0x7F), // LOGICAL_MAXIMUM (32767)
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
// HID_USAGE_SENSOR_UNITS_G,
HID_UNIT_EXPONENT(0x0E), // scale default unit Gs to "centi-Gs" to provide 2
digits past Gs decimal point
HID_INPUT(Const_Var_Abs),
HID_USAGE_SENSOR_DATA_MOTION_ACCELERATION_Z_AXIS,
HID_LOGICAL_MIN_16(0x01,0x80), // LOGICAL_MINIMUM (-32767)
HID_LOGICAL_MAX_16(0xFF,0x7F), // LOGICAL_MAXIMUM (32767)
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
// HID_USAGE_SENSOR_UNITS_G,
HID_UNIT_EXPONENT(0x0E), // scale default unit Gs to "centi-Gs" to provide 2
digits past Gs decimal point
HID_INPUT(Const_Var_Abs)
```