

Sensors Page Update HID Proposal

Request #	HUTRR59
Title	New Sensor Page Usages for Wearables
Spec Release	HUT 1.12
Received	17 Mar 2016
Requester	Nathan Sherman
Company	Microsoft Corporation
Phone	425-882-8080
Fax	
Email	nathans@microsoft.com
Current Status	Approved
Priority	
Submitted	21 Mar 2016
Voting Starts	11 Apr 2016
Voting Ends	18 Apr 2016
Required Voter	Microsoft (Chair, Submitter)
Required Voter	Intel
Required Voter	Apple
Required Voter	Wacom

Summary

This documentation defines new usages to support various sensors which have recently become available on computing devices.

Background

Recent devices are increasingly surfacing new sensors geared for mobile and emerging wearable platforms which have advanced hardware capabilities and software algorithms to provide health/fitness monitoring and personal activity tracking. A majority of the proposed sensor types target sensors which are already on these embedded devices. In general, the current list of sensor types supported by HID Sensors page needs to be updated to reflect the arrival of this new class of sensors in addition to other newly available sensors applicable to already existing categories.

Below is a list of the new sensors which will be supported organized by their respective categories. New categories being proposed are shown in italics:

1. Biometric
 - a. Blood Pressure
 - b. Body Temperature
 - c. Heart Rate
 - d. Heart Rate Variability
 - e. Peripheral Oxygen Saturation
 - f. Respiratory Rate
2. Environmental

- a. Air Quality
- b. Heat Index
- c. Surface Temperature
- d. Volatile Organic Compounds (VOC)
- 3. Light
 - a. Infrared Light
 - b. Visible Light
 - c. Ultraviolet Light
- 4. Motion
 - a. Gravity Vector
 - b. Linear Accelerometer
- 5. Orientation
 - a. Relative Orientation
 - b. Simple Orientation
- 6. *Personal Activity*
 - a. Activity Detection
 - b. Device Position
 - c. Pedometer
 - d. Step Detection
- 7. *Orientation Extended*
 - a. Geomagnetic Orientation
 - b. Magnetometer

Proposal

Refer to the table of contents for the organization of change texts for the proposed usages.

Change text section and page numbers refer to the original HID Sensors Usage Page Proposal document HUTRR39b: <http://www.usb.org/developers/hidpage/HUTRR39b.pdf>.

Table of Contents

1. Sensor Page Modifications.....	5
-----------------------------------	---

2. Sensor Usages	20
2.1 Sensor Device Usages	21
2.2 Sensor Field Usages: Properties	24
2.3 Biometric Sensor Field Usages	25
2.4 Environmental Sensor Field Usages	26
2.5 Light Sensor Field Usages	27
2.6 Orientation Sensor Field Usages	28
2.7 Time Sensor Field Usages	29
2.8 Custom Sensor Field Usages.....	30
2.9 Personal Activity Sensor Field Usages	32
3. Illustrative Examples	34
Biometric: Blood Pressure.....	34
Biometric: Body Temperature	35
Biometric: Heart Rate	37
Biometric: Heart Rate Variability	38
Biometric: Peripheral Oxygen Saturation	40
Biometric: Respiration Rate	41
Environmental: Air Quality.....	43
Environmental: Surface Temperature	44
Environmental: Heat Index	46
Environmental: Volatile Organic Compounds.....	47
Light: Infrared Light.....	49
Light: Ultraviolet Light.....	50
Light: Visible Light	52
Motion: Gravity Vector	53
Motion: Linear Accelerometer.....	55
Orientation: Relative Orientation	57
Orientation: Simple Orientation	59
Orientation Extended: Geomagnetic Orientation	60
Orientation Extended: Magnetometer	61
Personal Activity: Activity Detection.....	63
Personal Activity: Device Position.....	67
Personal Activity: Pedometer	68

Personal Activity: Step Detection 71

1. Sensor Page Modifications

Add the highlighted text in yellow to Section 1, page 7:

The Usage IDs are numerically segregated into sections for convenience.

- The lowest-numbered IDs from 0x00 to 0xFF are Usages applied to Collections and represent sensor objects (may equate to sensor Categories or Types).
- The IDs from 0x0100 to 0x07FF are Usages applied to Properties and Data Fields. These are grouped by the sensor Category where the Usages are commonly employed, but this arrangement is arbitrary. Usages may be reported by any sensor (or more than one sensor) if it makes sense to do so. **Properties and Data Fields can also apply to Collections within a Collection described by a Categories or Types Usage.**
- The IDs from 0x0800 to 0x0FFF are Selector Usages used with Properties or Data Fields that are Named Array enumerations. **Selectors can also apply to Collections within a Collection described by a Categories or Types Usage.**
- The IDs from 0x1000 to 0xEFFF are Properties or Data Fields from the 0x0100 – 0x0FFF range with “Modifiers” OR-ed in to the top 4 bits.
- The IDs from 0xF000 upward are reserved for proprietary use by vendors.

Below is the current HID Sensors Usage Page table modified to include the all required usages to standardize the sensors listed in the background section. New table rows are highlighted in green, modified rows in yellow, and deleted rows in red.

Usage ID	Usage Name	Usage Type	Section
00	Undefined		
01	Sensor	CA,CP	1.1
02-0F	Sensor: Reserved		
	<i>(for Properties commonly used with all Sensors, please look at Usage range 0300 – 03ff)</i>		1.5
	<i>(Data Field Timestamp is also commonly used with all Sensors, and its Usage is 0529)</i>		1.15
10	Biometric	CA,CP	1.1,1.6
11	Biometric: Human Presence	CA,CP	1.1,1.6
12	Biometric: Human Proximity	CA,CP	1.1,1.6
13	Biometric: Human Touch	CA,CP	1.1,1.6
14	Biometric: Blood Pressure	CA,CP	1.1,1.6
15	Biometric: Body Temperature	CA,CP	1.1,1.6
16	Biometric: Heart Rate	CA,CP	1.1,1.6
17	Biometric: Heart Rate Variability	CA,CP	1.1,1.6
18	Biometric: Peripheral Oxygen Saturation	CA,CP	1.1,1.6
19	Biometric: Respiratory Rate	CA,CP	1.1,1.6
1A-1F	Biometric: Reserved		
	<i>(for Data Fields commonly used with Biometric sensors, please look at Usage range 04b0 – 04cf)</i>		1.6
20	Electrical	CA,CP	1.1,1.7
21	Electrical: Capacitance	CA,CP	1.1,1.7
22	Electrical: Current	CA,CP	1.1,1.7
23	Electrical: Power	CA,CP	1.1,1.7

24	Electrical: Inductance	CA,CP	1.1,1.7
25	Electrical: Resistance	CA,CP	1.1,1.7
26	Electrical: Voltage	CA,CP	1.1,1.7
27	Electrical: Potentiometer	CA,CP	1.1,1.7
28	Electrical: Frequency	CA,CP	1.1,1.7
29	Electrical: Period	CA,CP	1.1,1.7
2A-2F	Electrical: Reserved		
	<i>(for Data Fields commonly used with Electrical sensors, please look at Usage range 0500 – 051f)</i>		1.7
30	Environmental	CA,CP	1.1,1.8
31	Environmental: Atmospheric Pressure	CA,CP	1.1,1.8
32	Environmental: Humidity	CA,CP	1.1,1.8
33	Environmental: Temperature	CA,CP	1.1,1.8
34	Environmental: Wind Direction	CA,CP	1.1,1.8
35	Environmental: Wind Speed	CA,CP	1.1,1.8
36	Environmental: Air Quality	CA,CP	1.1,1.8
37	Environmental: Heat Index	CA,CP	1.1,1.8
38	Environmental: Surface Temperature	CA,CP	1.1,1.8
39	Environmental: Volatile Organic Compounds	CA,CP	1.1,1.8
3A-3F	Environmental: Reserved		
	<i>(for Data Fields commonly used with Environmental sensors, please look at Usage range 0430 – 043f)</i>		1.8
	<i>(for Properties commonly used with Environmental sensors, please look at Usage range 0440 – 044f)</i>		1.8
40	Light	CA,CP	1.1,1.9
41	Light: Ambient Light	CA,CP	1.1,1.9
42	Light: Consumer Infrared	CA,CP	1.1,1.9
43	Light: Infrared Light	CA,CP	1.1,1.9
44	Light: Visible Light	CA,CP	1.1,1.9
45	Light: Ultraviolet Light	CA,CP	1.1,1.9
46-4F	Light: Reserved		
	<i>(for Data Fields commonly used with Light sensors, please look at Usage range 04d0 – 04ef)</i>		1.9
	<i>(Property Response Curve is also commonly used with Light sensors, and its Usage is 0318)</i>		1.5
50	Location	CA,CP	1.1,1.10
51	Location: Broadcast	CA,CP	1.1,1.10
52	Location: Dead Reckoning	CA,CP	1.1,1.10
53	Location: GPS (Global Positioning System)	CA,CP	1.1,1.10
54	Location: Lookup	CA,CP	1.1,1.10
55	Location: Other	CA,CP	1.1,1.10
56	Location: Static	CA,CP	1.1,1.10
57	Location: Triangulation	CA,CP	1.1,1.10
58-5F	Location: Reserved		
	<i>(for Data Fields commonly used with Location sensors, please look at Usage range 0400 – 0429)</i>		1.10
	<i>(for Properties commonly used with Location sensors, please look at Usage range 042a – 042f)</i>		1.10
60	Mechanical	CA,CP	1.1,1.11
61	Mechanical: Boolean Switch	CA,CP	1.1,1.11

62	Mechanical: Boolean Switch Array	CA,CP	1.1,1.11
63	Mechanical: Multivalued Switch	CA,CP	1.1,1.11
64	Mechanical: Force	CA,CP	1.1,1.11
65	Mechanical: Pressure	CA,CP	1.1,1.11
66	Mechanical: Strain	CA,CP	1.1,1.11
67	Mechanical: Weight	CA,CP	1.1,1.11
68	Mechanical: Haptic Vibrator	CA,CP	1.1,1.11
69	Mechanical: Hall Effect Switch	CA,CP	1.1,1.11
6A-6F	<i>Mechanical: Reserved</i>		
	<i>(for Data Fields commonly used with Mechanical sensors, please look at Usage range 0490 – 049f)</i>		1.11
	<i>(for Properties commonly used with Mechanical sensors, please look at Usage range 04a0 – 04af)</i>		1.11
70	Motion	CA,CP	1.1,1.12
71	Motion: Accelerometer 1D	CA,CP	1.1,1.12
72	Motion: Accelerometer 2D	CA,CP	1.1,1.12
73	Motion: Accelerometer 3D	CA,CP	1.1,1.12
74	Motion: Gyrometer 1D	CA,CP	1.1,1.12
75	Motion: Gyrometer 2D	CA,CP	1.1,1.12
76	Motion: Gyrometer 3D	CA,CP	1.1,1.12
77	Motion: Motion Detector	CA,CP	1.1,1.12
78	Motion: Speedometer	CA,CP	1.1,1.12
79	Motion: Accelerometer (any number of axes)	CA,CP	1.1,1.12
7A	Motion: Gyrometer (any number of axes)	CA,CP	1.1,1.12
7B	Motion: Gravity Vector (any number of axes)	CA,CP	1.1,1.12
7C	Motion: Linear Accelerometer (any number of axes)	CA,CP	1.1,1.12
7D-7F	<i>Motion: Reserved</i>		
	<i>(for Data Fields commonly used with Motion sensors, please look at Usage range 0450 – 046f)</i>		1.12
80	Orientation	CA,CP	1.1,1.13
81	Orientation: Compass 1D	CA,CP	1.1,1.13
82	Orientation: Compass 2D	CA,CP	1.1,1.13
83	Orientation: Compass 3D	CA,CP	1.1,1.13
84	Orientation: Inclinator 1D	CA,CP	1.1,1.13
85	Orientation: Inclinator 2D	CA,CP	1.1,1.13
86	Orientation: Inclinator 3D	CA,CP	1.1,1.13
87	Orientation: Distance 1D	CA,CP	1.1,1.13
88	Orientation: Distance 2D	CA,CP	1.1,1.13
89	Orientation: Distance 3D	CA,CP	1.1,1.13
8A	Orientation: Device Orientation	CA,CP	1.1,1.13
8B	Orientation: Compass (any number of axes)	CA,CP	1.1,1.13
8C	Orientation: Inclinator (any number of axes)	CA,CP	1.1,1.13
8D	Orientation: Distance (any number of axes)	CA,CP	1.1,1.13
8E	Orientation: Relative Orientation	CA,CP	1.1,1.13
8F	Orientation: Simple Orientation	CA,CP	1.1,1.13
	<i>Orientation: Reserved</i>		
	<i>(for additional Orientation sensor types, please look at Orientation Extended category at Usage range C0-CF)</i>		
	<i>(for Data Fields commonly used with Orientation/Orientation Extended sensors, please look at Usage range 0470 – 048f)</i>		1.13

90	Scanner	CA,CP	1.1,1.14
91	Scanner: Barcode	CA,CP	1.1,1.14
92	Scanner: RFID	CA,CP	1.1,1.14
93	Scanner: NFC	CA,CP	1.1,1.14
94-9F	Scanner: Reserved		
	<i>(for Data Fields commonly used with Scanner sensors, please look at Usage range 04f0 – 04f7)</i>		1.14
	<i>(for Properties commonly used with Scanner sensors, please look at Usage range 04f8 – 04ff)</i>		1.14
A0	Time	CA,CP	1.1,1.15
A1	Time: Alarm Timer	CA,CP	1.1,1.15
A2	Time: Real Time Clock	CA,CP	1.1,1.15
A3-AF	Time: Reserved		
	<i>(for Data Fields commonly used with Time sensors, please look at Usage range 0520 – 052f)</i>		1.15
	<i>(for Properties commonly used with Time sensors, please look at Usage range 0530 – 053f)</i>		1.15
B0	Personal Activity	CA,CP	1.1,1.16
B1	Personal Activity: Activity Detection	CA,CP	1.1,1.16
B2	Personal Activity: Device Position	CA,CP	1.1,1.16
B3	Personal Activity: Pedometer	CA,CP	1.1,1.16
B4	Personal Activity: Step Detection	CA,CP	1.1,1.16
B5-BF	Personal Activity: Reserved		
	<i>(for Data Fields commonly used with Personal Activity sensors, please look at Usage range 0590 – 059f)</i>		1.16
	<i>(for Properties commonly used with Personal Activity sensors, please look at Usage range 05A0 – 05Af)</i>		1.16
C0	Orientation Extended	CA,CP	1.1,1.13
C1	Orientation Extended: Geomagnetic Orientation	CA,CP	1.1,1.13
C2	Orientation Extended: Magnetometer	CA,CP	1.1,1.13
C3-CF	Orientation Extended: Reserved		
	<i>(for Data Fields commonly used with Orientation/Orientation Extended sensors, please look at Usage range 0470 – 048f)</i>		1.13
D0-DF	Reserved		
E0	Other	CA,CP	1.1
E1	Other: Custom	CA,CP	1.1,1.16
E2	Other: Generic	CA,CP	1.1,1.17
E3	Other: Generic Enumerator	CA,CP	1.1,1.17
E5-EF	Other: Reserved		
	<i>(for Data Fields commonly used with Custom sensors, please look at Usage range 0540 – 055f)</i>		1.16
	<i>(for Properties commonly used with Generic sensors, please look at Usage range 0560 – 057f)</i>		1.17
F0-FF	Reserved for Vendors/OEMs	CA,CP	
	<i>(for Vendor-reserved Data Fields and Properties commonly used with Vendor-reserved sensors, please use</i>		

	<i>Usage range f000-ffff)</i>		
	<i>(Modifiers are Usage Switches used in conjunction with other Usages. The value of the Modifier is OR-ed in to the top 4 bits of the un-modified Usage ID)</i>		
0	Modifier: None	US	1.2
1	Modifier: Change Sensitivity Absolute	US	1.2
2	Modifier: Maximum	US	1.2
3	Modifier: Minimum	US	1.2
4	Modifier: Accuracy	US	1.2
5	Modifier: Resolution	US	1.2
6	Modifier: Threshold High	US	1.2
7	Modifier: Threshold Low	US	1.2
8	Modifier: Calibration Offset	US	1.2
9	Modifier: Calibration Multiplier	US	1.2
A	Modifier: Report Interval	US	1.2
B	Modifier: Frequency Max	US	1.2
C	Modifier: Period Max	US	1.2
D	Modifier: Change Sensitivity Percent of Range	US	1.2
E	Modifier: Change Sensitivity Percent Relative	US	1.2
F	<i>Modifier: Reserved for Vendors/OEMs</i>	US	1.2
	<i>(These Events are commonly used by all Sensors)</i>		
0200	Event		
0201	Event: Sensor State	NAry	1.3
0800	Sensor State: Undefined	Sel	1.3
0801	Sensor State: Ready	Sel	1.3
0802	Sensor State: Not Available	Sel	1.3
0803	Sensor State: No Data	Sel	1.3
0804	Sensor State: Initializing	Sel	1.3
0805	Sensor State: Access Denied	Sel	1.3
0806	Sensor State: Error	Sel	1.3
0202	Event: Sensor Event	NAry	1.4
0810	Sensor Event: Unknown	Sel	1.4
0811	Sensor Event: State Changed	Sel	1.4
0812	Sensor Event: Property Changed	Sel	1.4
0813	Sensor Event: Data Updated	Sel	1.4
0814	Sensor Event: Poll Response	Sel	1.4
0815	Sensor Event: Change Sensitivity	Sel	1.4
0816	Sensor Event: Range Maximum Reached	Sel	1.4
0817	Sensor Event: Range Minimum Reached	Sel	1.4
0818	Sensor Event: High Threshold Cross Upward	Sel	1.4
0819	Sensor Event: High Threshold Cross Downward	Sel	1.4
081A	Sensor Event: Low Threshold Cross Upward	Sel	1.4
081B	Sensor Event: Low Threshold Cross Downward	Sel	1.4
081C	Sensor Event: Zero Threshold Cross Upward	Sel	1.4
081D	Sensor Event: Zero Threshold Cross Downward	Sel	1.4
081E	Sensor Event: Period Exceeded	Sel	1.4
081F	Sensor Event: Frequency Exceeded	Sel	1.4
0820	Sensor Event: Complex Trigger	Sel	1.4
0203-02FF	<i>Event: Reserved</i>		

	<i>(These Properties are commonly used by all Sensors)</i>		
0300	Property		1.5
0301	Property: Friendly Name	SV	1.5
0302	Property: Persistent Unique ID	DV	1.5
0303	Property: Sensor Status	DV	1.5
0304	Property: Minimum Report Interval (<i>default Unit: milliseconds</i>)	SV	1.5
0305	Property: Sensor Manufacturer	SV	1.5
0306	Property: Sensor Model	SV	1.5
0307	Property: Sensor Serial Number	SV	1.5
0308	Property: Sensor Description	SV	1.5
0309	Property: Sensor Connection Type	NArY	1.5
0830	Connection Type: PC Integrated	Sel	1.5
0831	Connection Type: PC Attached	Sel	1.5
0832	Connection Type: PC External	Sel	1.5
030A	Property: Sensor Device Path	DV	1.5
030B	Property: Hardware Revision	SV	1.5
030C	Property: Firmware Version	SV	1.5
030D	Property: Release Date	SV	1.5
030E	Property: Report Interval (<i>default Unit: milliseconds</i>)	DV	1.5
030F	Property: Change Sensitivity Absolute	DV	1.5
0310	Property: Change Sensitivity Percent of Range	DV	1.5
0311	Property: Change Sensitivity Percent Relative	DV	1.5
0312	Property: Accuracy	DV	1.5
0313	Property: Resolution	DV	1.5
0314	Property: Maximum	DV	1.5
0315	Property: Minimum	DV	1.5
0316	Property: Reporting State	NArY	1.5
0840	Reporting State: Report No Events	Sel	1.5
0841	Reporting State: Report All Events	Sel	1.5
0842	Reporting State: Report Threshold Events	Sel	1.5
0843	Reporting State: Wake On No Events	Sel	1.5
0844	Reporting State: Wake On All Events	Sel	1.5
0845	Reporting State: Wake On Threshold Events	Sel	1.5
0317	Property: Sampling Rate (<i>default Unit: milliseconds</i>)	DV	1.5
0318	Property: Response Curve	DV	1.5
0319	Property: Power State	NArY	1.5
0850	Power State: Undefined	Sel	1.5
0851	Power State: D0 Full Power	Sel	1.5
0852	Power State: D1 Low Power	Sel	1.5
0853	Power State: D2 Standby Power with Wakeup	Sel	1.5
0854	Power State: D3 Sleep with Wakeup	Sel	1.5
0855	Power State: D4 Power Off	Sel	1.5
031A	Property: Maximum FIFO Events	SV	1.5
031B	Property: Report Latency	DV	1.5
031C	Property: Flush FIFO Events	DF	1.5
031D	Property: Maximum Power Consumption	DV	1.5
031E-03FF	Property: Reserved		
	<i>(These Data Fields are commonly used by Location sensors)</i>		
0400	Data Field: Location	SV	1.10

0401	<i>Data Field: Location Reserved</i>		
0402	Data Field: Altitude Antenna Sea Level (<i>default Unit: meters</i>)	SV	1.10
0403	Data Field: Differential Reference Station ID	SV	1.10
0404	Data Field: Altitude Ellipsoid Error (<i>default Unit: meters</i>)	SV	1.10
0405	Data Field: Altitude Ellipsoid (<i>default Unit: meters</i>)	SV	1.10
0406	Data Field: Altitude Sea Level Error (<i>default Unit: meters</i>)	SV	1.10
0407	Data Field: Altitude Sea Level (<i>default Unit: meters</i>)	SV	1.10
0408	Data Field: Differential GPS Data Age (<i>default Unit: seconds</i>)	SV	1.10
0409	Data Field: Error Radius (<i>default Unit: meters</i>)	SV	1.10
040A	Data Field: Fix Quality	NAry	1.10
0870	Fix Quality: No Fix	Sel	1.10
0871	Fix Quality: GPS	Sel	1.10
0872	Fix Quality: DGPS	Sel	1.10
040B	Data Field: Fix Type	NAry	1.10
0880	Fix Type: No Fix	Sel	1.10
0881	Fix Type: GPS SPS Mode, Fix Valid	Sel	1.10
0882	Fix Type: DGPS SPS Mode, Fix Valid	Sel	1.10
0883	Fix Type: GPS PPS Mode, Fix Valid	Sel	1.10
0884	Fix Type: Real Time Kinematic	Sel	1.10
0885	Fix Type: Float RTK	Sel	1.10
0886	Fix Type: Estimated (dead reckoned)	Sel	1.10
0887	Fix Type: Manual Input Mode	Sel	1.10
0888	Fix Type: Simulator Mode	Sel	1.10
040C	Data Field: Geoidal Separation (<i>default Unit: meters</i>)	SV	1.10
040D	Data Field: GPS Operation Mode	NAry	1.10
0890	GPS Operation Mode: Manual	Sel	1.10
0891	GPS Operation Mode: Automatic	Sel	1.10
040E	Data Field: GPS Selection Mode	SV	1.10
08A0	GPS Selection Mode: Autonomous	Sel	1.10
08A1	GPS Selection Mode: DGPS	Sel	1.10
08A2	GPS Selection Mode: Estimated (dead reckoned)	Sel	1.10
08A3	GPS Selection Mode: Manual Input	Sel	1.10
08A4	GPS Selection Mode: Simulator	Sel	1.10
08A5	GPS Selection Mode: Data Not Valid	Sel	1.10
040F	Data Field: GPS Status	NAry	1.10
08B0	GPS Status: Data Valid	Sel	1.10
08B1	GPS Status: Data Not Valid	Sel	1.10
0410	Data Field: Position Dilution of Precision	SV	1.10
0411	Data Field: Horizontal Dilution of Precision	SV	1.10
0412	Data Field: Vertical Dilution of Precision	SV	1.10
0413	Data Field: Latitude (<i>default Unit: degrees</i>)	SV	1.10
0414	Data Field: Longitude (<i>default Unit: degrees</i>)	SV	1.10
0415	Data Field: True Heading (<i>default Unit: degrees</i>)	SV	1.10
0416	Data Field: Magnetic Heading (<i>default Unit: degrees</i>)	SV	1.10
0417	Data Field: Magnetic Variation (<i>default Unit: degrees</i>)	SV	1.10
0418	Data Field: Speed (<i>default Unit: knots</i>)	SV	1.10
0419	Data Field: Satellites in View	SV	1.10

041A	Data Field: Satellites in View Azimuth	SV	1.10
041B	Data Field: Satellites in View Elevation	SV	1.10
041C	Data Field: Satellites in View IDs	SV	1.10
041D	Data Field: Satellites in View PRNs	SV	1.10
041E	Data Field: Satellites in View S/N Ratios	SV	1.10
041F	Data Field: Satellites Used Count	SV	1.10
0420	Data Field: Satellites Used PRNs	SV	1.10
0421	Data Field: NMEA Sentence	SV	1.10
0422	Data Field: Address Line 1	SV	1.10
0423	Data Field: Address Line 2	SV	1.10
0424	Data Field: City	SV	1.10
0425	Data Field: State or Province	SV	1.10
0426	Data Field: Country or Region (ISO 3166)	SV	1.10
0427	Data Field: Postal Code	SV	1.10
0428-0429	<i>Data Field: Location Reserved</i>		1.10
	<i>(These Properties are commonly used by Location sensors)</i>		
042A	Property: Location		1.10
042B	Property: Location Desired Accuracy	NAry	1.10
0860	Accuracy: Default	Sel	1.10
0861	Accuracy: High	Sel	1.10
0862	Accuracy: Medium	Sel	1.10
0863	Accuracy: Low	Sel	1.10
042C-042F	<i>Property: Location Reserved</i>		
	<i>(These Data Fields are commonly used by Environmental sensors)</i>		
0430	Data Field: Environmental	SV	1.8
0431	Data Field: Atmospheric Pressure (default Unit: bars)	SV	1.8
0432	<i>Data Field: Reserved</i>		1.8
0433	Data Field: Relative Humidity (percent)	SV	1.8
0434	Data Field: Temperature (default Unit: degrees Celsius)	SV	1.8
0435	Data Field: Wind Direction (default Unit: degrees)	SV	1.8
0436	Data Field: Wind Speed (default Unit: meters/second)	SV	1.8
0437	Data Field: Air Quality Index (default Unit: air quality index defined by the United States EPA)	SV	1.8
0438	Data Field: Equivalent CO2 (default Unit: percentage)	SV	1.8
0439	Data Field: Volatile Organic Compound Concentration (default Unit: percentage)	SV	1.8
043A-043F	<i>Data Field: Environmental Reserved</i>		
	<i>(These Properties are commonly used by Environmental sensors)</i>		
0440	Property: Environmental	SV	1.8
0441	Property: Reference Pressure (default Unit: bars)	SV	1.8
0442-044F	<i>Property: Environmental Reserved</i>		
	<i>(These Data Fields are commonly used by Motion sensors)</i>		
0450	Data Field: Motion	SV	1.12
0451	Data Field: Motion State	SF	1.12

0452	Data Field: Acceleration (<i>default Unit: G's</i>)	SV	1.12
0453	Data Field: Acceleration Axis X (<i>default Unit: G's</i>)	SV	1.12
0454	Data Field: Acceleration Axis Y (<i>default Unit: G's</i>)	SV	1.12
0455	Data Field: Acceleration Axis Z (<i>default Unit: G's</i>)	SV	1.12
0456	Data Field: Angular Velocity (<i>default Unit: degrees/second</i>)	SV	1.12
0457	Data Field: Angular Velocity X about Axis (<i>default Unit: degrees/second</i>)	SV	1.12
0458	Data Field: Angular Velocity Y about Axis (<i>default Unit: degrees/second</i>)	SV	1.12
0459	Data Field: Angular Velocity Z about Axis (<i>default Unit: degrees/second</i>)	SV	1.12
045A	Data Field: Angular Position (<i>default Unit: degrees</i>)	SV	1.12
045B	Data Field: Angular Position about X Axis (<i>default Unit: degrees</i>)	SV	1.12
045C	Data Field: Angular Position about Y Axis (<i>default Unit: degrees</i>)	SV	1.12
045D	Data Field: Angular Position about Z Axis (<i>default Unit: degrees</i>)	SV	1.12
045E	Data Field: Motion Speed (<i>default Unit: meters/second</i>)	SV	1.12
045F	Data Field: Motion Intensity (<i>percent</i>)	SV	1.12
0460-046F	<i>Data Field: Motion Reserved</i>		
	<i>(These Data Fields are commonly used by Orientation sensors)</i>		
0470	Data Field: Orientation	SV	1.13
0471	Data Field: Heading (<i>default Unit: degrees</i>)	SV	1.13
0472	Data Field: Heading X Axis (<i>default Unit: degrees</i>)	SV	1.13
0473	Data Field: Heading Y Axis (<i>default Unit: degrees</i>)	SV	1.13
0474	Data Field: Heading Z Axis (<i>default Unit: degrees</i>)	SV	1.13
0475	Data Field: Heading Compensated Magnetic North (<i>default Unit: degrees</i>)	SV	1.13
0476	Data Field: Heading Compensated True North (<i>default Unit: degrees</i>)	SV	1.13
0477	Data Field: Heading Magnetic North (<i>default Unit: degrees</i>)	SV	1.13
0478	Data Field: Heading True North (<i>default Unit: degrees</i>)	SV	1.13
0479	Data Field: Distance (<i>default Unit: meters</i>)	SV	1.13
047A	Data Field: Distance X Axis (<i>default Unit: meters</i>)	SV	1.13
047B	Data Field: Distance Y Axis (<i>default Unit: meters</i>)	SV	1.13
047C	Data Field: Distance Z Axis (<i>default Unit: meters</i>)	SV	1.13
047D	Data Field: Distance Out-of-Range	SF	1.13
047E	Data Field: Tilt (<i>default Unit: degrees</i>)	SV	1.13
047F	Data Field: Tilt X Axis (<i>default Unit: degrees</i>)	SV	1.13
0480	Data Field: Tilt Y Axis (<i>default Unit: degrees</i>)	SV	1.13
0481	Data Field: Tilt Z Axis (<i>default Unit: degrees</i>)	SV	1.13
0482	Data Field: Rotation Matrix	SV	1.13
0483	Data Field: Quaternion	SV	1.13
0484	Data Field: Magnetic Flux (<i>default Unit: milligauss</i>)	SV	1.13
0485	Data Field: Magnetic Flux X Axis (<i>default Unit: milligauss</i>)	SV	1.13
0486	Data Field: Magnetic Flux Y Axis (<i>default Unit:</i>	SV	1.13

	<i>milligauss)</i>		
0487	Data Field: Magnetic Flux Z Axis (<i>default Unit: milligauss)</i>	SV	1.13
0488	Data Field: Magnetometer Accuracy	NAry	
08E0	Magnetometer Accuracy: Low	Sel	1.13
08E1	Magnetometer Accuracy: Medium	Sel	1.13
08E2	Magnetometer Accuracy: High	Sel	1.13
0489	Data Field: Simple Orientation Direction	NAry	1.13
08F0	Simple Orientation Direction: Not Rotated	Sel	1.13
08F1	Simple Orientation Direction: Rotated 90 Degrees CCW	Sel	1.13
08F2	Simple Orientation Direction: Rotated 180 Degrees CCW	Sel	1.13
08F3	Simple Orientation Direction: Rotated 270 Degrees CCW	Sel	1.13
08F4	Simple Orientation Direction: Face Up	Sel	1.13
08F5	Simple Orientation Direction: Face Down	Sel	1.13
048A-048F	<i>Data Field: Orientation Reserved</i>		
	<i>(These Data Fields are commonly used by Mechanical sensors)</i>		
0490	Data Field: Mechanical	SV	1.11
0491	Data Field: Boolean Switch State	SF	1.11
0492	Data Field: Boolean Switch Array States	SV	1.11
0493	Data Field: Multivalue Switch Value	SV	1.11
0494	Data Field: Force (<i>default Unit: Newtons)</i>	SV	1.11
0495	Data Field: Absolute Pressure (<i>default Unit: Pascals)</i>	SV	1.11
0496	Data Field: Gauge Pressure (<i>default Unit: Pascals)</i>	SV	1.11
0497	Data Field: Strain (<i>percent</i>)	SV	1.11
0498	Data Field: Weight (<i>default Unit: kilograms)</i>	SV	1.11
0498-049F	<i>Data Field: Mechanical Reserved</i>		1.11
	<i>(These Properties are commonly used by Mechanical sensors)</i>		1.11
04A0	Property: Mechanical	DV	1.11
04A1	Property: Vibration State	DF	1.11
04A2	Property: Forward Vibration Speed (<i>percent</i>)	DV	1.11
04A3	Property: Backward Vibration Speed (<i>percent</i>)	DV	1.11
04A4-04AF	<i>Property: Mechanical Reserved</i>		
	<i>(These Data Fields are commonly used by Biometric sensors)</i>		
04B0	Data Field: Biometric	SV	1.6
04B1	Data Field: Human Presence	SF	1.6
04B2	Data Field: Human Proximity Range (<i>default Unit: meters</i>)	SV	1.6
04B3	Data Field: Human Proximity Out of Range	SF	1.6
04B4	Data Field: Human Touch State	SF	1.6
04B5	Data Field: Blood Pressure (<i>default Units: mm Hg</i>)	SF	1.6
04B6	Data Field: Blood Pressure Diastolic (<i>default Units: mm Hg</i>)	SF	1.6
04B7	Data Field: Blood Pressure Systolic (<i>default Units:</i>	SF	1.6

	<i>mm Hg)</i>		
04B8	Data Field: Heart Rate (default Units: heartbeats per minute)	SV	1.6
04B9	Data Field: Resting Heart Rate (default Units: heartbeats per minute)	SV	1.6
04BA	Data Field: Heartbeat Interval (default Units: ms)	SV	1.6
04BB	Data Field: Respiratory Rate (default Units: breaths per minute)	SV	1.6
04BC	Data Field: SpO2 (default Units: percent)	SV	1.6
04BD-04CF	Data Field: Biometric Reserved		
	<i>(These Data Fields are commonly used by Light sensors)</i>		
04D0	Data Field: Light	SV	1.9
04D1	Data Field: Illuminance (default Unit: Lux)	SV	1.9
04D2	Data Field: Color Temperature (default Unit: degrees Kelvin)	SV	1.9
04D3	Data Field: Chromaticity	SV	1.9
04D4	Data Field: Chromaticity X (default Unit: CIE 1931 x)	SV	1.9
04D5	Data Field: Chromaticity Y (default Unit: CIE 1931 y)	SV	1.9
04D6	Data Field: Consumer IR Sentence Receive	SV	1.9
04D7	Data Field: Infrared Light (default Units: W/mm ²)	SV	1.9
04D8	Data Field: Red Light (default Units: W/mm ²)	SV	1.9
04D9	Data Field: Green Light (default Units: W/mm ²)	SV	1.9
04DA	Data Field: Blue Light (default Units: W/mm ²)	SV	1.9
04DB	Data Field: Ultraviolet A Light (default Units: mW/cm ²)	SV	1.9
04DC	Data Field: Ultraviolet B Light (default Units: mW/cm ²)	SV	1.9
04DD	Data Field: Ultraviolet Index (default Units: none)	SV	1.9
04DE-04DF	Data Field: Light Reserved		
	<i>(These Properties are commonly used by Light sensors)</i>		
04E0	Property: Light	DV	1.9
04E1	Property: Consumer IR Sentence Send	DV	1.9
04E2-04EF	Property: Light Reserved		
	<i>(Property Response Curve is also commonly used with Light sensors; it is Usage 0318)</i>		1.5
	<i>(These Data Fields are commonly used by Scanner sensors)</i>		
04F0	Data Field: Scanner	SV	1.14
04F1	Data Field: RFID Tag 40 Bit	SV	1.14
04F2	Data Field: NFC Sentence Receive	SV	1.14
04F3-04F7	Data Field: Scanner Reserved		
	<i>(These Properties are commonly used by Scanner sensors)</i>		
04F8	Property: Scanner	SV	1.14
04F9	Property: NFC Sentence Send	SV	1.14
04FA-04FF	Property: Scanner Reserved		
	<i>(These Data Fields are commonly used by Electrical sensors)</i>		
0500	Data Field: Electrical	SV	1.7
0501	Data Field: Capacitance (default Unit: Farads)	SV	1.7

0502	Data Field: Current (<i>default Unit: Amperes</i>)	SV	1.7
0503	Data Field: Electrical Power (<i>default Unit: Watts</i>)	SV	1.7
0504	Data Field: Inductance (<i>default Unit: Henrys</i>)	SV	1.7
0505	Data Field: Resistance (<i>default Unit: Ohms</i>)	SV	1.7
0506	Data Field: Voltage (<i>default Unit: Volts</i>)	SV	1.7
0507	Data Field: Frequency (<i>default Unit: Hertz</i>)	SV	1.7
0508	Data Field: Period (<i>default Unit: milliseconds</i>)	SV	1.7
0509	Data Field: Percent of Range	SV	1.7
050A-051F	<i>Data Field: Electrical Reserved</i>		
	<i>(These Data Fields are commonly used by Time sensors)</i>		
0520	Data Field: Time	SV	1.15
0521	Data Field: Year	SV	1.15
0522	Data Field: Month	SV	1.15
0523	Data Field: Day	SV	1.15
0524	Data Field: Day of Week	NAry	1.15
08C0	Day of Week: Sunday	Sel	1.15
08C1	Day of Week: Monday	Sel	1.15
08C2	Day of Week: Tuesday	Sel	1.15
08C3	Day of Week: Wednesday	Sel	1.15
08C4	Day of Week: Thursday	Sel	1.15
08C5	Day of Week: Friday	Sel	1.15
08C6	Day of Week: Saturday	Sel	1.15
0525	Data Field: Hour	SV	1.15
0526	Data Field: Minute	SV	1.15
0527	Data Field: Second	SV	1.15
0528	Data Field: Millisecond	SV	1.15
0529	Data Field: Timestamp	SV	1.15
052A	Data Field: Julian Day of Year	SV	1.15
052B	Data Field: Time Since System Boot	SV	1.15
052C-052F	<i>Data Field: Time Reserved</i>		
	<i>(These Properties are commonly used by Time sensors)</i>		
0530	Property: Time	DV	1.15
0531	Property: Time Zone Offset from UTC (<i>default Unit: minutes</i>)	DV	1.15
0532	Property: Time Zone Name	DV	1.15
0533	Property: Daylight Savings Time Observed	DF	1.15
0534	Property: Time Trim Adjustment	DV	1.15
0535	Property: Arm Alarm	DF	1.15
0535-053F	<i>Property: Time Reserved</i>		
	<i>(These Data Fields are commonly used by Custom sensors)</i>		
0540	Data Field: Custom	SV	1.16
0541	Data Field: Custom Usage	SV	1.16
0542	Data Field: Custom Boolean Array	SV	1.16
0543	Data Field: Custom Value	SV	1.16
0544	Data Field: Custom Value 1	SV	1.16
0545	Data Field: Custom Value 2	SV	1.16
0546	Data Field: Custom Value 3	SV	1.16
0547	Data Field: Custom Value 4	SV	1.16
0548	Data Field: Custom Value 5	SV	1.16

0549	Data Field: Custom Value 6	SV	1.16
054A	Data Field: Custom Value 7	SV	1.16
054B	Data Field: Custom Value 8	SV	1.16
054C	Data Field: Custom Value 9	SV	1.16
054D	Data Field: Custom Value 10	SV	1.16
054E	Data Field: Custom Value 11	SV	1.16
054F	Data Field: Custom Value 12	SV	1.16
0550	Data Field: Custom Value 13	SV	1.16
0551	Data Field: Custom Value 14	SV	1.16
0552	Data Field: Custom Value 15	SV	1.16
0553	Data Field: Custom Value 16	SV	1.16
0554	Data Field: Custom Value 17	SV	1.16
0555	Data Field: Custom Value 18	SV	1.16
0556	Data Field: Custom Value 19	SV	1.16
0557	Data Field: Custom Value 20	SV	1.16
0558	Data Field: Custom Value 21	SV	1.16
0559	Data Field: Custom Value 22	SV	1.16
055A	Data Field: Custom Value 23	SV	1.16
055B	Data Field: Custom Value 24	SV	1.16
055C	Data Field: Custom Value 25	SV	1.16
055D	Data Field: Custom Value 26	SV	1.16
055E	Data Field: Custom Value 27	SV	1.16
055F	Data Field: Custom Value 28	SV	1.16
05B0	Data Field: Custom Type ID	SV	1.16
05B1-05BF	<i>Data Field: Custom Reserved</i>		
	<i>(These Data Fields are commonly used by Personal Activity sensors)</i>		
0590	Data Field: Personal Activity	SV	1.18
0591	Data Field: Activity Type	NAry	1.18
0930	Activity Type: Unknown	Sel	1.18
0931	Activity Type: Stationary	Sel	1.18
0932	Activity Type: Fidgeting	Sel	1.18
0933	Activity Type: Walking	Sel	1.18
0934	Activity Type: Running	Sel	1.18
0935	Activity Type: In Vehicle	Sel	1.18
0936	Activity Type: Biking	Sel	1.18
0937	Activity Type: Idle	Sel	1.18
0592	Data Field: Activity State	NAry	1.18
0960	Activity State: No State Change	Sel	1.18
0961	Activity State: Start Activity	Sel	1.18
0962	Activity State: End Activity	Sel	1.18
0593	Data Field: Device Position	NAry	1.18
0980	Device Position: Unknown	Sel	1.18
0981	Device Position: Unchanged	Sel	1.18
0982	Device Position: On Desk	Sel	1.18
0983	Device Position: In Hand	Sel	1.18
0984	Device Position: Moving in Bag	Sel	1.18
0985	Device Position: Stationary in Bag	Sel	1.18
0594	Data Field: Step Count	SV	1.18
0595	Data Field: Step Count Reset	DF	1.18
0596	Data Field: Step Duration	SV	1.18

0597	Data Field: Step Type	NAry	1.18
0990	Step Type: Unknown	Sel	1.18
0991	Step Type: Running	Sel	1.18
0992	Step Type: Walking	Sel	1.18
0597-059F	<i>Data Field: Personal Activity Reserved</i>		
05A0	Property: Minimum Activity Detection Interval	DV	1.18
05A1	Property: Supported Activity Types	NAry	1.18
05A2	Property: Subscribed Activity Types	NAry	1.18
05A3	Property: Supported Step Types	NAry	1.18
05A4	Property: Subscribed Step Types	NAry	1.18
05A5	Property: Floor Height	DV	1.18
05A6-05AF	<i>Property: Personal Activity Reserved</i>		
	<i>(These Data Fields are commonly used by Generic sensors)</i>		
0560	Data Field: Generic	SV	1.17
0561	Data Field: Generic GUID or PROPERTYKEY	SV	1.17
0562	Data Field: Generic Category GUID	SV	1.17
0563	Data Field: Generic Type GUID	SV	1.17
0564	Data Field: Generic Event PROPERTYKEY	SV	1.17
0565	Data Field: Generic Property PROPERTYKEY	SV	1.17
0566	Data Field: Generic Data Field PROPERTYKEY	SV	1.17
0567	Data Field: Generic Event	SV	1.17
0568	Data Field: Generic Property	SV	1.17
0569	Data Field: Generic Data Field	SV	1.17
056A	Data Field: Enumerator Table Row Index	SV	1.17
056B	Data Field: Enumerator Table Row Count	SV	1.17
056C	Data Field: Generic GUID or PROPERTYKEY kind	NAry	1.17
08D0	Kind: Category	Sel	1.17
08D1	Kind: Type	Sel	1.17
08D2	Kind: Event	Sel	1.17
08D3	Kind: Property	Sel	1.17
08D4	Kind: Data Field	Sel	1.17
056D	Data Field: Generic GUID	SV	1.17
056E	Data Field: Generic PROPERTYKEY	SV	1.17
056F	Data Field: Generic Top Level Collection ID	SV	1.17
0570	Data Field: Generic Report ID	SV	1.17
0571	Data Field: Generic Report Item Position Index	SV	1.17
0572	Data Field: Generic Firmware VARTYPE	NAry	1.17
0900	VT_NULL: Empty	Sel	1.17
0901	VT_BOOL: Boolean	Sel	1.17
0902	VT_UI1: Byte	Sel	1.17
0903	VT_I1: Character	Sel	1.17
0904	VT_UI2: Unsigned Short	Sel	1.17
0905	VT_I2: Short	Sel	1.17
0906	VT_UI4: Unsigned Long	Sel	1.17
0907	VT_I4: Long	Sel	1.17
0908	VT_UI8: Unsigned Long Long	Sel	1.17
0909	VT_I8: Long Long	Sel	1.17
090A	VT_R4: Float	Sel	1.17
090B	VT_R8: Double	Sel	1.17
090C	VT_WSTR: Wide String	Sel	1.17
090D	VT_STR: Narrow String	Sel	1.17

090E	VT_CLSID: Guid	Sel	1.17
090F	VT_VECTOR VT_UI1: Opaque Structure	Sel	1.17
0910	VT_F16E0: HID 16-bit Float with Unit Exponent 0	Sel	1.17
0911	VT_F16E1: HID 16-bit Float with Unit Exponent 1	Sel	1.17
0912	VT_F16E2: HID 16-bit Float with Unit Exponent 2	Sel	1.17
0913	VT_F16E3: HID 16-bit Float with Unit Exponent 3	Sel	1.17
0914	VT_F16E4: HID 16-bit Float with Unit Exponent 4	Sel	1.17
0915	VT_F16E5: HID 16-bit Float with Unit Exponent 5	Sel	1.17
0916	VT_F16E6: HID 16-bit Float with Unit Exponent 6	Sel	1.17
0917	VT_F16E7: HID 16-bit Float with Unit Exponent 7	Sel	1.17
0918	VT_F16E8: HID 16-bit Float with Unit Exponent 8	Sel	1.17
0919	VT_F16E9: HID 16-bit Float with Unit Exponent 9	Sel	1.17
091A	VT_F16EA: HID 16-bit Float with Unit Exponent A	Sel	1.17
091B	VT_F16EB: HID 16-bit Float with Unit Exponent B	Sel	1.17
091C	VT_F16EC: HID 16-bit Float with Unit Exponent C	Sel	1.17
091D	VT_F16ED: HID 16-bit Float with Unit Exponent D	Sel	1.17
091E	VT_F16EE: HID 16-bit Float with Unit Exponent E	Sel	1.17
091F	VT_F16EF: HID 16-bit Float with Unit Exponent F	Sel	1.17
0920	VT_F32E0: HID 32-bit Float with Unit Exponent 0	Sel	1.17
0921	VT_F32E1: HID 32-bit Float with Unit Exponent 1	Sel	1.17
0922	VT_F32E2: HID 32-bit Float with Unit Exponent 2	Sel	1.17
0923	VT_F32E3: HID 32-bit Float with Unit Exponent 3	Sel	1.17
0924	VT_F32E4: HID 32-bit Float with Unit Exponent 4	Sel	1.17
0925	VT_F32E5: HID 32-bit Float with Unit Exponent 5	Sel	1.17
0926	VT_F32E6: HID 32-bit Float with Unit Exponent 6	Sel	1.17
0927	VT_F32E7: HID 32-bit Float with Unit Exponent 7	Sel	1.17
0928	VT_F32E8: HID 32-bit Float with Unit Exponent 8	Sel	1.17
0929	VT_F32E9: HID 32-bit Float with Unit Exponent 9	Sel	1.17
092A	VT_F32EA: HID 32-bit Float with Unit Exponent A	Sel	1.17
092B	VT_F32EB: HID 32-bit Float with Unit Exponent B	Sel	1.17
092C	VT_F32EC: HID 32-bit Float with Unit Exponent C	Sel	1.17
092D	VT_F32ED: HID 32-bit Float with Unit Exponent D	Sel	1.17
092E	VT_F32EE: HID 32-bit Float with Unit Exponent E	Sel	1.17
092F	VT_F32EF: HID 32-bit Float with Unit Exponent F	Sel	1.17
0573	Data Field: Generic Unit of Measure	NAry	1.17
0940	Unit: Not Specified	Sel	1.17
0941	Unit: Lux	Sel	1.17
0942	Unit: Degrees Kelvin	Sel	1.17
0943	Unit: Degrees Celsius	Sel	1.17
0944	Unit: Pascal	Sel	1.17
0945	Unit: Newton	Sel	1.17
0946	Unit: Meters/Second	Sel	1.17
0947	Unit: Kilogram	Sel	1.17
0948	Unit: Meter	Sel	1.17
0949	Unit: Meters/Second/Second	Sel	1.17
094A	Unit: Farad	Sel	1.17
094B	Unit: Ampere	Sel	1.17
094C	Unit: Watt	Sel	1.17
094D	Unit: Henry	Sel	1.17
094E	Unit: Ohm	Sel	1.17
094F	Unit: Volt	Sel	1.17
0950	Unit: Hertz	Sel	1.17
0951	Unit: Bar	Sel	1.17

0952	Unit: Degrees Anti-clockwise	Sel	1.17
0953	Unit: Degrees Clockwise	Sel	1.17
0954	Unit: Degrees	Sel	1.17
0955	Unit: Degrees/Second	Sel	1.17
0956	Unit: Degrees/Second/Second	Sel	1.17
0957	Unit: Knot	Sel	1.17
0958	Unit: Percent	Sel	1.17
0959	Unit: Second	Sel	1.17
095A	Unit: Millisecond	Sel	1.17
095B	Unit: G	Sel	1.17
095C	Unit: Bytes	Sel	1.17
095D	Unit: Milligauss	Sel	1.17
095E	Unit: Bits	Sel	1.17
0574	Data Field: Generic Unit Exponent	NAry	1.17
0970	Exponent 0: 1	Sel	1.17
0971	Exponent 1: 10	Sel	1.17
0972	Exponent 2: 100	Sel	1.17
0973	Exponent 3: 1 000	Sel	1.17
0974	Exponent 4: 10 000	Sel	1.17
0975	Exponent 5: 100 000	Sel	1.17
0976	Exponent 6: 1 000 000	Sel	1.17
0977	Exponent 7: 10 000 000	Sel	1.17
0978	Exponent 8: 0.00 000 001	Sel	1.17
0979	Exponent 9: 0.0 000 001	Sel	1.17
097A	Exponent A: 0.000 001	Sel	1.17
097B	Exponent B: 0.00 001	Sel	1.17
097C	Exponent C: 0.0 001	Sel	1.17
097D	Exponent D: 0.001	Sel	1.17
097E	Exponent E: 0.01	Sel	1.17
097F	Exponent F: 0.1	Sel	1.17
0575	Data Field: Generic Report Size	SV	1.17
0576	Data Field: Generic Report Count	SV	1.17
0577-057F	Data Field: Generic Reserved		
	(These Properties are commonly used by Generic sensors)		
0580	Property: Generic	DV	1.17
0581	Property: Enumerator Table Row Index	DV	1.17
0582	Property: Enumerator Table Row Count	SV	1.17
0583-058F	Property: Generic Reserved		
05C0-07FF	Reserved for future use as Sensor Types, Data Fields and Properties		
09A0-09FF	Reserved for use as Selection Values		
1000-	Reserved for use as Data Fields with Modifiers		1.2
FFFF			
F000-	Reserved for Vendors/OEMs		
FFFF			

2. Sensor Usages

This section provides descriptions for the proposed usages to be added in Section 1, pages 7 to 45. Requested text modifications to existing descriptions are highlighted in yellow.

2.1 Sensor Device Usages

Add to the following to table in Section 1.1, page 20:

Biometric: Blood Pressure	CA,CP – An application-level or physical collection that identifies a device that measures the systolic/diastolic blood pressure of the device user.
Biometric: Body Temperature	CA,CP – An application-level or physical collection that identifies a device that measures temperature where the temperature source is a location on the device user’s body.
Biometric: Heart Rate	CA,CP – An application-level or physical collection that identifies a device that measures the heart rate of the device user.
Biometric: Heart Rate Variability	CA,CP – An application-level or physical collection that identifies a device that measures the variation in the time interval between heartbeats of the device user.
Biometric: Peripheral Oxygen Saturation	CA,CP – An application-level or physical collection that identifies a device that measures the peripheral oxygen saturation (SpO ₂) as a percentage of the hemoglobin in the device user’s blood which contains oxygen.
Biometric: Respiratory Rate	CA,CP – An application-level or physical collection that identifies a device that measures the respiratory rate (number of breaths taken over time) of the device user.
Environmental: Air Quality	CA,CP – An application-level or physical collection that identifies a device that measures the amount of pollutants in the air.
Environmental: Heat Index	CA,CP – An application-level or physical collection that identifies a device that measures the human-perceived temperature based on the current humidity and air temperature levels.
Environmental: Surface Temperature	CA,CP – An application-level or physical collection that identifies a device that measures the temperature of a surface which the device is currently in contact with.
Environmental: Volatile Organic Compounds	CA,CP – An application-level or physical collection that identifies a device that measures the amount of volatile organic compounds in the air.
Light: Infrared Light	CA,CP – An application-level or physical collection that identifies a device that measures levels of infrared light (wavelengths of approximately 700nm to 1mm on the electromagnetic spectrum).
Light: Visible Light	CA,CP – An application-level or physical collection that identifies a device that measures levels of visible light (wavelengths of approximately 390nm to 700nm on the electromagnetic spectrum).
Light: Ultraviolet Light	CA,CP – An application-level or physical collection that identifies a device that measures levels of Ultraviolet light (wavelengths of approximately 10nm to 390nm on the electromagnetic spectrum).
Motion: Gravity Vector	CA,CP – An application-level or physical collection that identifies a device that measures exclusively the force of Earth’s gravity along any number of axes.
Motion: Linear Accelerometer	CA,CP – An application-level or physical collection that identifies

	a device that measures the linear acceleration (acceleration excluding the force of Earth’s gravity) along any number of axes. Note that this differs from a standard accelerometer in that at rest, a standard accelerometer displays 1g due to earth’s gravitational pull while a linear accelerometer will show 0g.
Orientation: Relative Orientation	CA,CP – An application-level or physical collection that identifies a device that measures device orientation where yaw values are relative to the starting position of the device when powered on.
Orientation: Simple Orientation	CA,CP – An application-level or physical collection that identifies a device that measures the orientation by assigning an enumerated type to represent the current positional rotation of the face of the device.
Personal Activity	CA,CP – An application-level or physical collection that identifies a device that measures information regarding personal, day-to-day activities a user experiences.
Personal Activity: Activity Detection	CA,CP – An application-level or physical collection that identifies a device that measures confidence levels for detecting the device user’s current activity.
Personal Activity: Device Position	CA,CP – An application-level or physical collection that identifies a device that detects the type of position in which the device is currently placed.
Personal Activity: Pedometer	CA,CP – An application-level or physical collection that identifies a device that measures the cumulative number of steps taken, category of steps taken, and length of time spent stepping per category by the device user.
Personal Activity: Step Detection	CA,CP - An application-level or physical collection that identifies a device that detects when the user has taken a step.
Orientation Extended: Geomagnetic Orientation	CA,CP – An application-level or physical collection that identifies a device that measures device orientation in three axes where yaw is relative to Earth’s magnetic North (as opposed to true North). Note that this sensor normally is implemented through use of only an accelerometer and magnetometer, no gyroscope.
Orientation Extended: Magnetometer	CA,CP – An application-level or physical collection that identifies a device that measures magnetic field strengths.

To further clarify the differences between Temperature, Body Temperature and Surface Temperature sensor types, modify the Temperature sensor description of table entry in Section 1.1, page 21 to:

Environmental: Temperature	CA,CP – An application-level or physical collection that identifies a device that measures temperature, such as a thermometer or a thermocouple. This sensor’s temperature source primarily is the ambient air (see Biometric: Body Temperature and Environmental: Surface Temperature for other temperature sensor types).
-----------------------------------	---

To further clarify the differences between Geomagnetic Orientation and Device Orientation, modify the Device Orientation in Section 1.1, page 23 to:

Orientation: Device Orientation	CA,CP – An application-level or physical collection that identifies a device that measures device orientation in three axes (typically through the combined use of an accelerometer and gyroscope).
--	---

To not conflict with the description of Linear Accelerometer, remove the use of “linear” from the Accelerometer table entries in Section 1.1, page 22:

Motion: Accelerometer	CA,CP – An application-level or physical collection that identifies a device that measures linear acceleration along any number of axes
Motion: Accelerometer 1D	CA,CP – An application-level or physical collection that identifies a device that measures linear acceleration along 1 axis.
Motion: Accelerometer 2D	CA,CP – An application-level or physical collection that identifies a device that measures linear acceleration along 2 axes.
Motion: Accelerometer 3D	CA,CP – An application-level or physical collection that identifies a device that measures linear acceleration along 3 axes.

Add the usages highlighted in red to section 4.1, page 71/72:

```
//sensor category biometric
#define HID_USAGE_SENSOR_CATEGORY_BIOMETRIC 0x09,0x10
#define HID_USAGE_SENSOR_TYPE_BIOMETRIC_PRESENCE 0x09,0x11
#define HID_USAGE_SENSOR_TYPE_BIOMETRIC_PROXIMITY 0x09,0x12
#define HID_USAGE_SENSOR_TYPE_BIOMETRIC_TOUCH 0x09,0x13
#define HID_USAGE_SENSOR_TYPE_BLOOD_PRESSURE 0x09,0x14
#define HID_USAGE_SENSOR_TYPE_BODY_TEMPERATURE 0x09,0x15
#define HID_USAGE_SENSOR_TYPE_HEART_RATE 0x09,0x16
#define HID_USAGE_SENSOR_TYPE_HEART_RATE_VARIABILITY 0x09,0x17
#define HID_USAGE_SENSOR_TYPE_PERIPHERAL_OXYGEN_SATURATION 0x09,0x18
#define HID_USAGE_SENSOR_TYPE_RESPIRATORY_RATE 0x09,0x19
...
//sensor category environmental
#define HID_USAGE_SENSOR_CATEGORY_ENVIRONMENTAL 0x09,0x30
#define HID_USAGE_SENSOR_TYPE_ENVIRONMENTAL_ATMOSPHERIC_PRESSURE 0x09,0x31
#define HID_USAGE_SENSOR_TYPE_ENVIRONMENTAL_HUMIDITY 0x09,0x32
#define HID_USAGE_SENSOR_TYPE_ENVIRONMENTAL_TEMPERATURE 0x09,0x33
#define HID_USAGE_SENSOR_TYPE_ENVIRONMENTAL_WIND_DIRECTION 0x09,0x34
#define HID_USAGE_SENSOR_TYPE_ENVIRONMENTAL_WIND_SPEED 0x09,0x35
#define HID_USAGE_SENSOR_TYPE_ENVIRONMENTAL_AIR_QUALITY 0x09,0x36
#define HID_USAGE_SENSOR_TYPE_ENVIRONMENTAL_HEAT_INDEX 0x09,0x37
#define HID_USAGE_SENSOR_TYPE_ENVIRONMENTAL_SURFACE_TEMPERATURE 0x09,0x38
#define HID_USAGE_SENSOR_TYPE_ENVIRONMENTAL_VOLATILE_ORGANIC_COMOUNDS 0x09,0x39
...
//sensor category light
#define HID_USAGE_SENSOR_CATEGORY_LIGHT 0x09,0x40
#define HID_USAGE_SENSOR_TYPE_LIGHT_AMBIENTLIGHT 0x09,0x41
#define HID_USAGE_SENSOR_TYPE_LIGHT_CONSUMER_INFRARED 0x09,0x42
#define HID_USAGE_SENSOR_TYPE_LIGHT_INFRARED_LIGHT 0x09,0x43
#define HID_USAGE_SENSOR_TYPE_LIGHT_VISIBLE_LIGHT 0x09,0x44
#define HID_USAGE_SENSOR_TYPE_LIGHT_ULTRAVIOLET_LIGHT 0x09,0x45
```

```

...
//sensor category motion
#define HID_USAGE_SENSOR_CATEGORY_MOTION 0x09,0x70
#define HID_USAGE_SENSOR_TYPE_MOTION_ACCELEROMETER_1D 0x09,0x71
#define HID_USAGE_SENSOR_TYPE_MOTION_ACCELEROMETER_2D 0x09,0x72
#define HID_USAGE_SENSOR_TYPE_MOTION_ACCELEROMETER_3D 0x09,0x73
#define HID_USAGE_SENSOR_TYPE_MOTION_GYROMETER_1D 0x09,0x74
#define HID_USAGE_SENSOR_TYPE_MOTION_GYROMETER_2D 0x09,0x75
#define HID_USAGE_SENSOR_TYPE_MOTION_GYROMETER_3D 0x09,0x76
#define HID_USAGE_SENSOR_TYPE_MOTION_MOTION_DETECTOR 0x09,0x77
#define HID_USAGE_SENSOR_TYPE_MOTION_SPEEDOMETER 0x09,0x78
#define HID_USAGE_SENSOR_TYPE_MOTION_ACCELEROMETER 0x09,0x79
#define HID_USAGE_SENSOR_TYPE_MOTION_GYROMETER 0x09,0x7A
#define HID_USAGE_SENSOR_TYPE_MOTION_GRAVITY_VECTOR 0x09,0x7B
#define HID_USAGE_SENSOR_TYPE_MOTION_LINEAR_ACCELEROMETER 0x09,0x7C
...
//sensor category orientation
#define HID_USAGE_SENSOR_CATEGORY_ORIENTATION 0x09,0x80
#define HID_USAGE_SENSOR_TYPE_ORIENTATION_COMPASS_1D 0x09,0x81
#define HID_USAGE_SENSOR_TYPE_ORIENTATION_COMPASS_2D 0x09,0x82
#define HID_USAGE_SENSOR_TYPE_ORIENTATION_COMPASS_3D 0x09,0x83
#define HID_USAGE_SENSOR_TYPE_ORIENTATION_INCLINOMETER_1D 0x09,0x84
#define HID_USAGE_SENSOR_TYPE_ORIENTATION_INCLINOMETER_2D 0x09,0x85
#define HID_USAGE_SENSOR_TYPE_ORIENTATION_INCLINOMETER_3D 0x09,0x86
#define HID_USAGE_SENSOR_TYPE_ORIENTATION_DISTANCE_1D 0x09,0x87
#define HID_USAGE_SENSOR_TYPE_ORIENTATION_DISTANCE_2D 0x09,0x88
#define HID_USAGE_SENSOR_TYPE_ORIENTATION_DISTANCE_3D 0x09,0x89
#define HID_USAGE_SENSOR_TYPE_ORIENTATION_DEVICE_ORIENTATION 0x09,0x8A
#define HID_USAGE_SENSOR_TYPE_ORIENTATION_COMPASS 0x09,0x8B
#define HID_USAGE_SENSOR_TYPE_ORIENTATION_INCLINOMETER 0x09,0x8C
#define HID_USAGE_SENSOR_TYPE_ORIENTATION_DISTANCE 0x09,0x8D
#define HID_USAGE_SENSOR_TYPE_ORIENTATION_RELATIVE_ORIENTATION 0x09,0x8E
#define HID_USAGE_SENSOR_TYPE_ORIENTATION_SIMPLE_ORIENTATION 0x09,0x8F
...
//sensor category time
#define HID_USAGE_SENSOR_CATEGORY_TIME 0x09,0xA0
#define HID_USAGE_SENSOR_TYPE_TIME_ALARM 0x09,0xA1
#define HID_USAGE_SENSOR_TYPE_TIME_RTC 0x09,0xA2
//sensor category personal activity
#define HID_USAGE_SENSOR_CATEGORY_PERSONAL_ACTIVITY 0x09,0xB0
#define HID_USAGE_SENSOR_TYPE_PERSONAL_ACTIVITY_ACTIVITY_DETECTION 0x09,0xB1
#define HID_USAGE_SENSOR_TYPE_PERSONAL_ACTIVITY_DEVICE_POSITON 0x09,0xB2
#define HID_USAGE_SENSOR_TYPE_PERSONAL_ACTIVITY_FLOOR_TRACKER 0x09,0xB3
#define HID_USAGE_SENSOR_TYPE_PERSONAL_ACTIVITY_PEDOMETER 0x09,0xB4
#define HID_USAGE_SENSOR_TYPE_PERSONAL_ACTIVITY_STEP_DETECTION 0x09,0xB5
//sensor category orientation extended
#define HID_USAGE_SENSOR_CATEGORY_ORIENTATION_EXTENDED 0x09,0xC0
#define HID_USAGE_SENSOR_TYPE_ORIENTATION_EXTENDED_GEOMAGNETIC_ORIENTATION 0x09,0xC1
#define HID_USAGE_SENSOR_TYPE_ORIENTATION_MAGNETOMETER 0x09,0xC2

```

2.2 Sensor Field Usages: Properties

Add to the following to table in Section 1.5, page 27:

Flush FIFO Events	DF – TRUE indicates any batched input reports in the device’s buffer should be immediately flushed to the host where device is to reset this property to FALSE upon completion.
-------------------	---

Max Power Consumption	SV – Indicates the worst-case power consumption by the device. Default unit of measure is milli-watts, can be overridden using explicit Unit and/or Unit Exponent.
------------------------------	--

Modify Reporting State table entry in Section 1.5, page 29 to:

Reporting State	<p>NAry – Indicates the current reporting state of the sensor. The reporting state may be:</p> <ul style="list-style-type: none"> • Sel – Report No Events = no asynchronous Input reports are sent, • Sel – Report All Events = all Input reports are sent without any filtering. • Sel – Report Threshold Events = Input reports are sent only when it exceeds a pre-programmed threshold, • Sel – Wake On No Events = no asynchronous Input reports are sent and a Wake On event is never performed, • Sel – Wake On All Events = all Input reports are sent without any filtering and a Wake On event is performed, • Sel – Wake On Threshold Events = Input reports are sent only when it exceeds a pre-programmed threshold and a Wake On event is performed.
------------------------	---

Add the usages highlighted in red to section 4.1, page 73:

```
//begin reporting state selectors
#define HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE_NO_EVENTS          0x0A,0x40,0x08 // Sel
#define HID_USAGE_REPORTING_STATE_ON_NONE      HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE_NO_EVENTS
#define HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE_ALL_EVENTS        0x0A,0x41,0x08 // Sel
#define HID_USAGE_REPORTING_STATE_ON_ALL      HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE_ALL_EVENTS
#define HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE_THRESHOLD_EVENTS  0x0A,0x42,0x08 // Sel
#define
HID_USAGE_REPORTING_STATE_ON_THRESHOLD HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE_THRESHOLD_EVENTS
#define HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE_NO_EVENTS_WAKE    0x0A,0x43,0x08 // Sel
#define HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE_ALL_EVENTS_WAKE   0x0A,0x44,0x08 // Sel
#define HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE_THRESHOLD_EVENTS_WAKE 0x0A,0x45,0x08 // Sel
#define HID_USAGE_SENSOR_PROPERTY_REPORTING_STATE_ANYTIME           0x0A,0x46,0x08 // Sel
...
#define HID_USAGE_SENSOR_PROPERTY_MAX_FIFO_SAMPLES                  0x0A,0x1A,0x03
#define HID_USAGE_SENSOR_PROPERTY_REPORT_LATENCY                   0x0A,0x1B,0x03
#define HID_USAGE_SENSOR_PROPERTY_FLUSH_FIFO_EVENTS                0x0A,0x1C,0x03
#define HID_USAGE_SENSOR_PROPERTY_MAXIMUM_POWER_CONSUMPTION       0x0A,0x1D,0x03
```

2.3 Biometric Sensor Field Usages

Add to the following to table in Section 1.6, page 30:

Blood Pressure	SV – Indicates a blood pressure without respect to which blood pressure type (systolic and diastolic). This is usually used as a composite value for specifying min, max and accuracy for
-----------------------	---

	related blood pressures. Default units is mmHg; cannot be overridden.
Blood Pressure Diastolic	SV – Indicates the diastolic blood pressure of the device user. Default units is mmHg; cannot be overridden.
Blood Pressure Systolic	SV – Indicates the systolic blood pressure of the device user. Default units is mmHg; cannot be overridden.
Heart Rate	SV – Indicates the current heart rate of the device user. Default unit of measure is number of heart beats per minute; cannot be overridden.
Resting Heart Rate	SV – Indicates the current resting heart rate or the heart rate of the device user who has not had any recent physical exertion or stimulation. Default unit of measure is number of heart beats per minute; cannot be overridden.
Heartbeat Interval	SV – Indicates the timespan between two heart beats (also known as RR interval). Default unit of measure is ms, can be overridden using explicit Unit and/or Unit Exponent.
Respiratory Rate	SV – Indicates the current respiratory rate or rate of breath. Default unit of measure is number of breaths per minute; cannot be overridden.
SpO2	SV – Measures the percentage of hemoglobin containing oxygen in the blood of the device user. Default units is percent; cannot be overridden.

Add the usages highlighted in red to section 4.1, page 75:

```
//data type biometric
//data field usages (input report)
#define HID_USAGE_SENSOR_DATA_BIOMETRIC 0x0A, 0xB0, 0x04
#define HID_USAGE_SENSOR_DATA_BIOMETRIC_HUMAN_PRESENCE 0x0A, 0xB1, 0x04
#define HID_USAGE_SENSOR_DATA_BIOMETRIC_HUMAN_PROXIMITY_RANGE 0x0A, 0xB2, 0x04
#define HID_USAGE_SENSOR_DATA_BIOMETRIC_HUMAN_PROXIMITY_OUT_OF_RANGE 0x0A, 0xB3, 0x04
#define HID_USAGE_SENSOR_DATA_BIOMETRIC_HUMAN_TOUCH_STATE 0x0A, 0xB4, 0x04
#define HID_USAGE_SENSOR_DATA_BIOMETRIC_BLOOD_PRESSURE 0x0A, 0xB5, 0x04
#define HID_USAGE_SENSOR_DATA_BIOMETRIC_BLOOD_PRESSURE_DIASTOLIC 0x0A, 0xB6, 0x04
#define HID_USAGE_SENSOR_DATA_BIOMETRIC_BLOOD_PRESSURE_SYSTOLIC 0x0A, 0xB7, 0x04
#define HID_USAGE_SENSOR_DATA_BIOMETRIC_HEART_RATE 0x0A, 0xB8, 0x04
#define HID_USAGE_SENSOR_DATA_BIOMETRIC_RESTING_HEART_RATE 0x0A, 0xB9, 0x04
#define HID_USAGE_SENSOR_DATA_BIOMETRIC_HEARTBEAT_INTERVAL 0x0A, 0xBA, 0x04
#define HID_USAGE_SENSOR_DATA_BIOMETRIC_RESPIRATORY_RATE 0x0A, 0xBB, 0x04
#define HID_USAGE_SENSOR_DATA_BIOMETRIC_SP02 0x0A, 0xBC, 0x04
```

2.4 Environmental Sensor Field Usages

Add to the following to table in Section 1.8, page 31:

Air Quality Index	SV – Measures the air quality (or amount of pollutants in the surrounding air). Default unit of measure is AQI defined by the Environmental Protection Agency; cannot be overridden.
--------------------------	--

Equivalent CO2	SV – Measures the equivalent carbon-dioxide concentration in the surrounding air. Default unit of measure is percent; cannot be overridden.
Volatile Organic Compound Concentration	SV – Measures the total volatile organic compounds (TVOC) concentration. Default unit of measure is in percent; cannot be overridden.

Add the usages highlighted in red to section 4.1, page 74:

```
//data type environmental
//data field usages (input report)
#define HID_USAGE_SENSOR_DATA_ENVIRONMENTAL 0x0A,0x30,0x04
#define HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_ATMOSPHERIC_PRESSURE 0x0A,0x31,0x04
#define HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_REFERENCE_PRESSURE 0x0A,0x32,0x04
#define HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_RELATIVE_HUMIDITY 0x0A,0x33,0x04
#define HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_TEMPERATURE 0x0A,0x34,0x04
#define HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_WIND_DIRECTION 0x0A,0x35,0x04
#define HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_WIND_SPEED 0x0A,0x36,0x04
#define HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_AIR_QUALITY_INDEX 0x0A,0x37,0x04
#define HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_EQUIVALENT_CO2 0x0A,0x38,0x04
#define HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_VOLATILE_ORGANIC_COMPOUNDS 0x0A,0x39,0x04
```

2.5 Light Sensor Field Usages

Add to the following to table in Section 1.9, page 31:

Infrared Light	SV – Measures the amount of infrared light (wavelength of approximately 700nm to 1mm). Default unit of measure is W/mm ² , can be overridden using explicit Unit and/or Unit Exponent.
Red Light	SV – Measures the amount of red light (wavelength of approximately 620nm to 700nm). Default unit of measure is W/mm ² , can be overridden using explicit Unit and/or Unit Exponent.
Green Light	SV – Measures the amount of green light (wavelength of approximately 495nm to 570nm). Default unit of measure is W/mm ² , can be overridden using explicit Unit and/or Unit Exponent.
Blue Light	SV – Measures the amount of blue light (wavelength of approximately 450nm to 495nm). Default unit of measure is W/mm ² , can be overridden using explicit Unit and/or Unit Exponent.
Ultraviolet A Light	SV – Measures the amount of ultraviolet A light (wavelength of approximately 315nm to 400nm). Default unit of measure is mW/cm ² , can be overridden using explicit Unit and/or Unit Exponent.
Ultraviolet B Light	SV – Measures the amount of ultraviolet B light (wavelength of approximately 280nm to 315nm). Default unit of measure is

	mW/cm ² , can be overridden using explicit Unit and/or Unit Exponent.
Ultraviolet Index	SV – Measures the strength of ultraviolet radiation defined by the World Health Organization ultraviolet index standard. Default unit of measure is ultraviolet index unit and cannot be overridden.

Add the usages highlighted in red to section 4.1, page 75:

```
//data type light sensor
//data field usages (input report)
#define HID_USAGE_SENSOR_DATA_LIGHT 0x0A, 0xD0, 0x04
#define HID_USAGE_SENSOR_DATA_LIGHT_ILLUMINANCE 0x0A, 0xD1, 0x04
#define HID_USAGE_SENSOR_DATA_LIGHT_COLOR_TEMPERATURE 0x0A, 0xD2, 0x04
#define HID_USAGE_SENSOR_DATA_LIGHT_CHROMATICITY 0x0A, 0xD3, 0x04
#define HID_USAGE_SENSOR_DATA_LIGHT_CHROMATICITY_X 0x0A, 0xD4, 0x04
#define HID_USAGE_SENSOR_DATA_LIGHT_CHROMATICITY_Y 0x0A, 0xD5, 0x04
#define HID_USAGE_SENSOR_DATA_LIGHT_CONSUMER_IR_SENTENCE_RECEIVE 0x0A, 0xD6, 0x04
#define HID_USAGE_SENSOR_DATA_LIGHT_INFRARED_LIGHT 0x0A, 0xD7, 0x04
#define HID_USAGE_SENSOR_DATA_LIGHT_RED_LIGHT 0x0A, 0xD8, 0x04
#define HID_USAGE_SENSOR_DATA_LIGHT_GREEN_LIGHT 0x0A, 0xD9, 0x04
#define HID_USAGE_SENSOR_DATA_LIGHT_BLUE_LIGHT 0x0A, 0xDA, 0x04
#define HID_USAGE_SENSOR_DATA_LIGHT_ULTRAVIOLET_A_LIGHT 0x0A, 0xDB, 0x04
#define HID_USAGE_SENSOR_DATA_LIGHT_ULTRAVIOLET_B_LIGHT 0x0A, 0xDC, 0x04
#define HID_USAGE_SENSOR_DATA_LIGHT_ULTRAVIOLET_INDEX 0x0A, 0xDD, 0x04
```

2.6 Orientation Sensor Field Usages

Add to the following to table in Section 1.13, page 36:

Simple Orientation Direction	<p>NArY – Indicates the current orientation of the device with respect to the following types:</p> <ul style="list-style-type: none"> • Sel – Not Rotated, front face of device is in its default orientation • Sel – Rotated 90 Degrees CCW, front face of device is rotated 90 degrees counter clock-wise its default position • Sel – Rotated 180 Degrees CCW, front face of device is rotated 180 degrees counter clock-wise its default position • Sel – Rotated 270 Degrees CCW, front face of device is rotated 270 degrees counter clock-wise its default position • Sel – Face Up, front face of device is pointing away from the ground • Sel – Face Down, front face of device is pointing towards the ground
-------------------------------------	--

Add the usages highlighted in red to section 4.1, page 75:

```
//data type orientation
//data field usages (input report)
#define HID_USAGE_SENSOR_DATA_ORIENTATION 0x0A, 0x70, 0x04
```

```

#define HID_USAGE_SENSOR_DATA_ORIENTATION_MAGNETIC_HEADING 0x0A,0x71,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_MAGNETIC_HEADING_X 0x0A,0x72,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_MAGNETIC_HEADING_Y 0x0A,0x73,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_MAGNETIC_HEADING_Z 0x0A,0x74,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_COMPENSATED_MAGNETIC_NORTH 0x0A,0x75,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_COMPENSATED_TRUE_NORTH 0x0A,0x76,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_MAGNETIC_NORTH 0x0A,0x77,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_TRUE_NORTH 0x0A,0x78,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_DISTANCE 0x0A,0x79,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_DISTANCE_X 0x0A,0x7A,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_DISTANCE_Y 0x0A,0x7B,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_DISTANCE_Z 0x0A,0x7C,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_DISTANCE_OUT_OF_RANGE 0x0A,0x7D,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_TILT 0x0A,0x7E,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_TILT_X 0x0A,0x7F,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_TILT_Y 0x0A,0x80,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_TILT_Z 0x0A,0x81,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_ROTATION_MATRIX 0x0A,0x82,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_QUATERNION 0x0A,0x83,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_MAGNETIC_FLUX 0x0A,0x84,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_MAGNETIC_FLUX_X_AXIS 0x0A,0x85,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_MAGNETIC_FLUX_Y_AXIS 0x0A,0x86,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_MAGNETIC_FLUX_Z_AXIS 0x0A,0x87,0x04
#define HID_USAGE_SENSOR_DATA_ORIENTATION_MAGNETOMETER_ACCURACY 0x0A,0x88,0x04 //NArY
//begin magnetometer accuracy selectors
#define HID_USAGE_SENSOR_DATA_MAGNETOMETER_ACCURACY_LOW 0x0A,0xE0,0x08 // Sel
#define HID_USAGE_SENSOR_DATA_MAGNETOMETER_ACCURACY_MEDIUM 0x0A,0xE1,0x08 // Sel
#define HID_USAGE_SENSOR_DATA_MAGNETOMETER_ACCURACY_HIGH 0x0A,0xE2,0x08 // Sel
//end magnetometer accuracy selectors
#define HID_USAGE_SENSOR_DATA_ORIENTATION_SIMPLE_ORIENTATION_DIRECTION 0x0A,0x89,0x04 // NArY
//begin simple orientation direction selectors
#define HID_USAGE_SENSOR_DATA_SIMPLE_ORIENTATION_DIRECTION_NOT_ROTATED 0x0A,0xF0,0x08 // Sel
#define HID_USAGE_SENSOR_DATA_SIMPLE_ORIENTATION_DIRECTION_ROTATED_90_DEGREES_CCW 0x0A,0xF1,0x08 // Sel
#define HID_USAGE_SENSOR_DATA_SIMPLE_ORIENTATION_DIRECTION_ROTATED_180_DEGREES_CCW 0x0A,0xF2,0x08 // Sel
#define HID_USAGE_SENSOR_DATA_SIMPLE_ORIENTATION_DIRECTION_ROTATED_270_DEGREES_CCW 0x0A,0xF3,0x08 // Sel
#define HID_USAGE_SENSOR_DATA_SIMPLE_ORIENTATION_DIRECTION_FACE_UP 0x0A,0xF4,0x08 // Sel
#define HID_USAGE_SENSOR_DATA_SIMPLE_ORIENTATION_DIRECTION_FACE_DOWN 0x0A,0xF5,0x08 // Sel
//end simple orientation direction selectors

```

2.7 Time Sensor Field Usages

Add to the following to table in Section 1.15, page 39:

Time Since System Boot	SV – Specifies the amount of time that has passed since the host system’s boot. Default unit of measure is seconds; can be overridden using explicit Unit and/or Unit Exponent.
-------------------------------	---

Add the usages highlighted in red to section 4.1, page 76:

```

//data type time
//data field usages (input report)
#define HID_USAGE_SENSOR_DATA_TIME 0x0A,0x20,0x05
#define HID_USAGE_SENSOR_DATA_TIME_YEAR 0x0A,0x21,0x05
#define HID_USAGE_SENSOR_DATA_TIME_MONTH 0x0A,0x22,0x05
#define HID_USAGE_SENSOR_DATA_TIME_DAY 0x0A,0x23,0x05
#define HID_USAGE_SENSOR_DATA_TIME_DAY_OF_WEEK 0x0A,0x24,0x05
#define HID_USAGE_SENSOR_DATA_TIME_HOUR 0x0A,0x25,0x05
#define HID_USAGE_SENSOR_DATA_TIME_MINUTE 0x0A,0x26,0x05
#define HID_USAGE_SENSOR_DATA_TIME_SECOND 0x0A,0x27,0x05
#define HID_USAGE_SENSOR_DATA_TIME_MILLISECOND 0x0A,0x28,0x05
#define HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP 0x0A,0x29,0x05
#define HID_USAGE_SENSOR_DATA_TIME_JULIAN_DAY_OF_YEAR 0x0A,0x2A,0x05
#define HID_USAGE_SENSOR_DATA_TIME_TIME_SINCE_SYSTEM_BOOT 0x0A,0x2B,0x05

```

2.8 Custom Sensor Field Usages

Insert the following in the table in Section 1.16, page 40 after the “Custom Value” row:

Custom Type ID	SV – A 16-byte GUID to uniquely identify a Custom Sensor instance.
-----------------------	--

Add the following to end of the table in Section 1.16, page 40:

Custom Value 7	SV – A seventh custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 8	SV – A eighth custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 9	SV – A ninth custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 10	SV – A tenth custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 11	SV – A eleventh custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 12	SV – A twelfth custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 13	SV – A thirteenth custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 14	SV – A fourteenth custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 15	SV – A fifteenth custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 16	SV – A sixteenth custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 17	SV – A seventeenth custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 18	SV – A eighteenth custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 19	SV – A nineteenth custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 20	SV – A twentieth custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 21	SV – A twenty-first custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 22	SV – A twenty-second custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 23	SV – A twenty-third custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 24	SV – A twenty-fourth custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 25	SV – A twenty-fifth custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.

Custom Value 26	SV – A twenty-sixth custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 27	SV – A twenty-seventh custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.
Custom Value 28	SV – A twenty-eighth custom value. Units are specified by the Units usage and scaling by the Unit Exponent usage.

To further clarify the difference between Custom Type ID and Custom Usage, modify the Custom Usage table entry in Section 1.16 page 40 to:

Custom Usage	SV – Indicates the HID Sensor Usage. See custom data field Custom Type ID if searching for a data field to assign a unique identifier for a custom sensor.
---------------------	--

Add the usages highlighted in red to section 4.1, page 76:

```
//data type custom
//data field usages (input report)
#define HID_USAGE_SENSOR_DATA_CUSTOM 0x0A,0x40,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_USAGE 0x0A,0x41,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_BOOLEAN_ARRAY 0x0A,0x42,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE 0x0A,0x43,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_1 0x0A,0x44,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_2 0x0A,0x45,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_3 0x0A,0x46,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_4 0x0A,0x47,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_5 0x0A,0x48,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_6 0x0A,0x49,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_7 0x0A,0x4A,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_8 0x0A,0x4B,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_9 0x0A,0x4C,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_10 0x0A,0x4D,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_11 0x0A,0x4E,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_12 0x0A,0x4F,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_13 0x0A,0x50,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_14 0x0A,0x51,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_15 0x0A,0x52,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_16 0x0A,0x53,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_17 0x0A,0x54,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_18 0x0A,0x55,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_19 0x0A,0x56,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_20 0x0A,0x57,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_21 0x0A,0x58,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_22 0x0A,0x59,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_23 0x0A,0x5A,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_24 0x0A,0x5B,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_25 0x0A,0x5C,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_26 0x0A,0x5D,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_27 0x0A,0x5E,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_VALUE_28 0x0A,0x5F,0x05
#define HID_USAGE_SENSOR_DATA_CUSTOM_TYPE_ID 0x0A,0xB0,0x05
```

2.9 Personal Activity Sensor Field Usages

Create new section 1.18, “Personal Activity Sensor Field Usages” which has the following table:

Activity Type	<p>NARY – Indicates a type of Activity. Activity types may be:</p> <ul style="list-style-type: none"> • Sel – Unknown, the device cannot determine the current activity • Sel – Biking, the device detects the user is riding a bike • Sel – Fidgeting, the device detects the user is fidgeting or moving restlessly • Sel – Idle, the device detects the user is idle, or not moving but still actively using the device • Sel – In Vehicle, the device detects the user is in a moving vehicle • Sel – Running, the device detects the user is running • Sel – Stationary, the device itself (not the user) is laying still and not moving • Sel – Walking, the device detects the user is walking
Supported Activity Types	<p>NARY – Indicates the activity types which can be detected by the device. This data field reuses the selectors mentioned in the Activity Type data field.</p>
Subscribed Activity Types	<p>NARY – Indicates the activity types which the host request to receive input reports for. This data field reuses the selectors mentioned in the Activity Type data field.</p>
Activity State	<p>NARY – Indicates Activity state change. State changes might be:</p> <ul style="list-style-type: none"> • Sel – No State change • Sel – Start Activity • Sel – End Activity
Activity Minimum Detection Interval	<p>SV – Indicates the minimum time required by the device to detect an activity. This data field is per supported activity. Default unit of measure is seconds; can be overridden using explicit Unit and/or Unit Exponent.</p>
Device Position	<p>NARY – Indicates a type of position or placement of the device. Values may be:</p> <ul style="list-style-type: none"> • Sel – Unknown, the device cannot detect its current placement • Sel – Unchanged, the device’s placement has not changed since the previous report. • Sel – On Desk, the user has placed the device on a desk or table • Sel - In Hand, the user is holding the device in their hands • Sel – Moving in Bag, the device is moving inside a bag • Sel – Stationary in Bag, the device is stationary inside a bag
Step Count	<p>SV – Indicates the number of footsteps take since last device reset.</p>
Step Count Reset	<p>DF – If present in feature report, the host setting this to TRUE indicates device is to reset it’s step counter to 0, where the</p>

	<p>device is to set the flag back to FALSE upon completion of the reset.</p> <p>If present in input report, TRUE indicates the step count has been reset since the previous input report, where a reset could be due to either the device itself being reset or the device counter rolling over.</p>
Step Duration	SV – Indicates the cumulative length of time of footsteps taken since last device reset.
Step Type	<p>NArY – Indicates a type of footstep. Step types may be:</p> <ul style="list-style-type: none"> • Sel – Unknown, the device cannot determine the current step type, note that this is also to be used if the device does not have the ability to distinguish between different step types • Sel – Running, the footsteps were taken while the device user was running • Sel – Walking, the footsteps were taken while the device user was walking
Supported Step Types	Nary – Indicates the step types which can be detected by the device. This data field reuses the selectors mentioned in the Step Type data field.
Subscribed Step Types	Nary – Indicates the step types which the host request to receive input reports for. This data field reuses the selectors mentioned in the Step Type data field.

Add the usages highlighted in red to section 4.1, page 76:

```

//data type personal activity
//data field usages (input report)
#define HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY           0x0A,0x90,0x05
#define HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_ACTIVITY_TYPE 0x0A,0x91,0x05 // NArY
//begin activity type selectors
#define HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_UNKNOWN       0x0A,0x30,0x09 // Sel
#define HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_STATIONARY    0x0A,0x31,0x09 // Sel
#define HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_FIDGETING     0x0A,0x32,0x09 // Sel
#define HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_WALKING       0x0A,0x33,0x09 // Sel
#define HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_RUNNING       0x0A,0x34,0x09 // Sel
#define HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_IN_VEHICLE    0x0A,0x35,0x09 // Sel
#define HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_BIKING        0x0A,0x36,0x09 // Sel
#define HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_IDLE          0x0A,0x37,0x09 // Sel
//end activity type selectors
#define HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_ACTIVITY_STATE 0x0A,0x92,0x05 // Nary
//begin activity state selectors
#define HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_NO_STATE_CHANGE 0x0A,0x60,0x09 // Sel
#define HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_START_ACTIVITY 0x0A,0x61,0x09 // Sel
#define HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_END_ACTIVITY   0x0A,0x62,0x09 // Sel
//end activity state selectors
#define HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_DEVICE_POSITION 0x0A,0x93,0x05 // Nary
//begin device position type selectors
#define HID_USAGE_SENSOR_DATA_DEVICE_POSITION_UNKNOWN     0x0A,0x80,0x09 // Sel
#define HID_USAGE_SENSOR_DATA_DEVICE_POSITION_UNCHANGED   0x0A,0x81,0x09 // Sel
#define HID_USAGE_SENSOR_DATA_DEVICE_POSITION_ON_DESK     0x0A,0x82,0x09 // Sel
#define HID_USAGE_SENSOR_DATA_DEVICE_POSITION_IN_HAND     0x0A,0x83,0x09 // Sel
#define HID_USAGE_SENSOR_DATA_DEVICE_POSITION_MOVING_IN_BAG 0x0A,0x84,0x09 // Sel
#define HID_USAGE_SENSOR_DATA_DEVICE_POSITION_STATIONARY_IN_BAG 0x0A,0x85,0x09 // Sel
//end device position type selectors
#define HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_STEP_COUNT 0x0A,0x94,0x05
#define HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_STEP_COUNT_RESET 0x0A,0x95,0x05
#define HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_STEP_DURATION 0x0A,0x96,0x05
#define HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_STEP_TYPE 0x0A,0x97,0x05 // Nary
//begin elevation mode selectors

```

```

#define HID_USAGE_SENSOR_DATA_STEP_TYPE_UNKNOWN 0x0A,0x90,0x09 // Sel
#define HID_USAGE_SENSOR_DATA_STEP_TYPE_WALKING 0x0A,0x91,0x09 // Sel
#define HID_USAGE_SENSOR_DATA_STEP_TYPE_RUNNING 0x0A,0x92,0x09 // Sel
//end elevation mode selectors
//property usages (get/set feature report)
#define HID_USAGE_SENSOR_PROPERTY_PERSONAL_ACTIVITY_MINIMUM_ACTIVITY_DETECTION_INTERVAL 0x0A,0xA0,0x05
#define HID_USAGE_SENSOR_PROPERTY_PERSONAL_ACTIVITY_SUPPORTED_ACTIVITY_TYPES 0x0A,0xA1,0x05
#define HID_USAGE_SENSOR_PROPERTY_PERSONAL_ACTIVITY_SUBSCRIBED_ACTIVITY_TYPES 0x0A,0xA2,0x05
#define HID_USAGE_SENSOR_PROPERTY_PERSONAL_ACTIVITY_SUPPORTED_STEP_TYPES 0x0A,0xA3,0x05
#define HID_USAGE_SENSOR_PROPERTY_PERSONAL_ACTIVITY_SUBSCRIBED_STEP_TYPES 0x0A,0xA4,0x05
#define HID_USAGE_SENSOR_PROPERTY_PERSONAL_ACTIVITY_FLOOR_HEIGHT 0x0A,0xA4,0x05

```

3. Illustrative Examples

Biometric: Blood Pressure

// For reference: [Complete HID report descriptor](#)

```

const unsigned char blood_pressure_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_BIOMETRIC_BLOOD_PRESSURE,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_POWER_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_POWER_STATE_UNDEFINED,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
        HID_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_UNIT_EXPONENT(0),
    HID_FEATURE(Data_Var_Abs),
    HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
    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_DATA(HID_USAGE_SENSOR_DATA_BIOMETRIC_BLOOD_PRESSURE,
    HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_BIOMETRIC_BLOOD_PRESSURE_SYSTOLIC,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(0xFF),
HID_REPORT_SIZE(8),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_INPUT(Data_Var_Abs),
HID_USAGE_SENSOR_DATA_BIOMETRIC_BLOOD_PRESSURE_DIASTOLIC,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(0xFF),
HID_REPORT_SIZE(8),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_INPUT(Data_Var_Abs),
HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(2),
HID_UNIT_EXPONENT(0x09),
HID_INPUT(Data_Var_Abs),

HID_END_COLLECTION
};

```

Biometric: Body Temperature

```

// For reference: Complete HID report descriptor
const unsigned char body_temperature_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_BIOMETRIC_BODY_TEMPERATURE,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_POWER_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_POWER_STATE_UNDEFINED,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
    HID_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
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),
// reuse environment's temperature data field
HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_TEMPERATURE, HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
    HID_END_COLLECTION,
    // reuse environment's temperature data field
    HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_TEMPERATURE,
    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_UNIT_EXPONENT(0),
    HID_INPUT(Data_Var_Abs),
    HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
    HID_LOGICAL_MIN_8(0),
    HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
    HID_REPORT_SIZE(32),
    HID_REPORT_COUNT(2),
    HID_UNIT_EXPONENT(0x09),
    HID_INPUT(Data_Var_Abs),

    HID_END_COLLECTION
};

```

Biometric: Heart Rate

```

// For reference: Complete HID report descriptor
const unsigned char heart_rate_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_BIOMETRIC_HEART_RATE,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_POWER_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_POWER_STATE_UNDEFINED,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
        HID_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
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_DATA(HID_USAGE_SENSOR_DATA_BIOMETRIC_HEART_RATE, HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_BIOMETRIC_HEART_RATE,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_INPUT(Data_Var_Abs),
HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(2),
HID_UNIT_EXPONENT(0x09),
HID_INPUT(Data_Var_Abs),

HID_END_COLLECTION
};

```

Biometric: Heart Rate Variability

```

// For reference: Complete HID report descriptor
const unsigned char heart_rate_variability_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_BIOMETRIC_HEART_RATE_VARIABILITY,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_POWER_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_POWER_STATE_UNDEFINED,
  HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
  HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
  HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
  HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
  HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
  HID_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
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_DATA(HID_USAGE_SENSOR_DATA_BIOMETRIC_HEARTBEAT_INTERVAL,
  HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,

```

```

HID_USAGE_SENSOR_DATA_BIOMETRIC_HEARTBEAT_INTERVAL,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_INPUT(Data_Var_Abs),
HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(2),
HID_UNIT_EXPONENT(0x09),
HID_INPUT(Data_Var_Abs),

HID_END_COLLECTION
};

```

Biometric: Peripheral Oxygen Saturation

```

// For reference: Complete HID report descriptor
const unsigned char peripheral_oxygen_saturation_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_BIOMETRIC_PERIPHERAL_OXYGEN_SATURATION,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_POWER_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_POWER_STATE_UNDEFINED,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
        HID_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_UNIT_EXPONENT(0),
    HID_FEATURE(Data_Var_Abs),
    HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
    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_DATA(HID_USAGE_SENSOR_DATA_BIOMETRIC_SPO2, HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(0x64), // max 100%
HID_REPORT_SIZE(8),
HID_REPORT_COUNT(1),
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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_BIOMETRIC_SPO2,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(0x64), // max 100%
HID_REPORT_SIZE(8),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_INPUT(Data_Var_Abs),
HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(2),
HID_INPUT(Data_Var_Abs),

HID_END_COLLECTION
};

```

Biometric: Respiration Rate

```

// For reference: Complete HID report descriptor
const unsigned char respiratory_rate_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_BIOMETRIC_RESPIRATORY_RATE,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_POWER_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_POWER_STATE_UNDEFINED,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
    HID_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
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_DATA(HID_USAGE_SENSOR_DATA_BIOMETRIC_RESPIRATORY_RATE, HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_BIOMETRIC_RESPIRATORY_RATE,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),

```

```

HID_INPUT(Data_Var_Abs),
HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(2),
HID_UNIT_EXPONENT(0x09),
HID_INPUT(Data_Var_Abs),

HID_END_COLLECTION
};

```

Environmental: Air Quality

```

// For reference: Complete HID report descriptor
const unsigned char air_quality_report_descriptor[] = {
HID_USAGE_PAGE_SENSOR,
HID_USAGE_SENSOR_TYPE_ENVIRONMENTAL_AIR_QUALITY,
HID_COLLECTION(Physical),

//feature reports (xmit/receive)
HID_USAGE_PAGE_SENSOR,
HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
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(Data_Arr_Abs),
HID_END_COLLECTION,
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_POWER_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_POWER_STATE_UNDEFINED,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
    HID_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
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_DATA(HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_AIR_QUALITY_INDEX,
    HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_EQUIVALENT_CO2,
    HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(0x64), // max 100%
HID_REPORT_SIZE(8),
HID_REPORT_COUNT(1),
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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_AIR_QUALITY_INDEX,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_INPUT(Data_Var_Abs),
HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_EQUIVALENT_CO2,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(0x64), // max 100%
HID_REPORT_SIZE(8),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_INPUT(Data_Var_Abs),
HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(2),
HID_UNIT_EXPONENT(0x09),
HID_INPUT(Data_Var_Abs),

HID_END_COLLECTION
};

```

Environmental: Surface Temperature

```

// For reference: Complete HID report descriptor
const unsigned char surface_temperature_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_ENVIRONMENTAL_SURFACE_TEMPERATURE,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
    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(Data_Arr_Abs),
HID_END_COLLECTION,
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_POWER_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_POWER_STATE_UNDEFINED,
HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
HID_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
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),
// reuse environment's temperature data field
HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_TEMPERATURE, HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
    HID_END_COLLECTION,
    // reuse environment's temperature data field
    HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_TEMPERATURE,
    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_UNIT_EXPONENT(0),
    HID_INPUT(Data_Var_Abs),
    HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
    HID_LOGICAL_MIN_8(0),
    HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
    HID_REPORT_SIZE(32),
    HID_REPORT_COUNT(2),
    HID_UNIT_EXPONENT(0x09),
    HID_INPUT(Data_Var_Abs),

    HID_END_COLLECTION
};

```

Environmental: Heat Index

```

// For reference: Complete HID report descriptor
const unsigned char heat_index_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_ENVIRONMENTAL_HEAT_INDEX,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_POWER_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_POWER_STATE_UNDEFINED,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
        HID_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
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),
// reuse environment's temperature data field
HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_TEMPERATURE, HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
// reuse environment's temperature data field
HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_TEMPERATURE,
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_UNIT_EXPONENT(0),
HID_INPUT(Data_Var_Abs),
HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(2),
HID_UNIT_EXPONENT(0x09),
HID_INPUT(Data_Var_Abs),

HID_END_COLLECTION
};

```

Environmental: Volatile Organic Compounds

```

// For reference: Complete HID report descriptor
const unsigned char voc_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_ENVIRONMENTAL_VOLATILE_ORGANIC_COMPOUNDS,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
    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(Data_Arr_Abs),
};

```

```

HID_END_COLLECTION,
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_POWER_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_POWER_STATE_UNDEFINED,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
    HID_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
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_DATA(HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_VOLATILE_ORGANIC_COMPOUND_CONCENTRATION,
    HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(0x64), // max 100%
HID_REPORT_SIZE(8),
HID_REPORT_COUNT(1),
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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
    HID_END_COLLECTION,
    HID_USAGE_SENSOR_DATA_ENVIRONMENTAL_VOLATILE_ORGANIC_COMPOUND_CONCENTRATION,
    HID_LOGICAL_MIN_8(0),
    HID_LOGICAL_MAX_8(0x64), // max 100%
    HID_REPORT_SIZE(8),
    HID_REPORT_COUNT(1),
    HID_UNIT_EXPONENT(0),
    HID_INPUT(Data_Var_Abs),
    HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
    HID_LOGICAL_MIN_8(0),
    HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
    HID_REPORT_SIZE(32),
    HID_REPORT_COUNT(2),
    HID_UNIT_EXPONENT(0x09),
    HID_INPUT(Data_Var_Abs),

    HID_END_COLLECTION
};

```

Light: Infrared Light

```

// For reference: Complete HID report descriptor
const unsigned char infrared_light_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_LIGHT_ULTRAVIOLET_LIGHT,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_POWER_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_POWER_STATE_UNDEFINED,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
        HID_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
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_DATA(HID_USAGE_SENSOR_DATA_LIGHT, HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
    HID_END_COLLECTION,
    HID_USAGE_SENSOR_DATA_LIGHT_INFRARED_LIGHT,
    HID_LOGICAL_MIN_8(0),
    HID_LOGICAL_MAX_16(0xFF, 0xFF),
    HID_REPORT_SIZE(16),
    HID_REPORT_COUNT(1),
    HID_UNIT_EXPONENT(0),
    HID_INPUT(Data_Var_Abs),
    HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
    HID_LOGICAL_MIN_8(0),
    HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
    HID_REPORT_SIZE(32),
    HID_REPORT_COUNT(2),
    HID_UNIT_EXPONENT(0x09),
    HID_INPUT(Data_Var_Abs),

    HID_END_COLLECTION
};

```

Light: Ultraviolet Light

```

// For reference: Complete HID report descriptor
const unsigned char uv_light_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_LIGHT_ULTRAVIOLET_LIGHT,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_POWER_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_POWER_STATE_UNDEFINED,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
    HID_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
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_DATA(HID_USAGE_SENSOR_DATA_LIGHT,HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_LIGHT_ULTRAVIOLET_A_LIGHT,
HID_LOGICAL_MIN_8(0),

```

```

HID_LOGICAL_MAX_16(0xFF,0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_INPUT(Data_Var_Abs),
HID_USAGE_SENSOR_DATA_LIGHT_ULTRAVIOLET_B_LIGHT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF,0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_INPUT(Data_Var_Abs),
HID_USAGE_SENSOR_DATA_LIGHT_ULTRAVIOLET_INDEX,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_INPUT(Data_Var_Abs),
HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(2),
HID_UNIT_EXPONENT(0x09),
HID_INPUT(Data_Var_Abs),

HID_END_COLLECTION
};

```

Light: Visible Light

```

// For reference: Complete HID report descriptor
const unsigned char visible_light_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_LIGHT_VISISBLE,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_POWER_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_POWER_STATE_UNDEFINED,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
        HID_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
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_DATA(HID_USAGE_SENSOR_DATA_LIGHT,HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_LIGHT_RED_LIGHT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF,0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_INPUT(Data_Var_Abs),
HID_USAGE_SENSOR_DATA_LIGHT_GREEN_LIGHT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF,0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_INPUT(Data_Var_Abs),
HID_USAGE_SENSOR_DATA_LIGHT_BLUE_LIGHT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF,0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_INPUT(Data_Var_Abs),
HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(2),
HID_UNIT_EXPONENT(0x09),
HID_INPUT(Data_Var_Abs),

HID_END_COLLECTION
};

```

Motion: Gravity Vector

```

// For reference: Complete HID report descriptor
const unsigned char gravity_vector_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,

```

```

HID_USAGE_SENSOR_TYPE_MOTION_GRAVITY_VECTOR,
HID_COLLECTION(Physical),

//feature reports (xmit/receive)
HID_USAGE_PAGE_SENSOR,
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(Data_Arr_Abs),
HID_END_COLLECTION,
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_POWER_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_POWER_STATE_UNDEFINED,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
    HID_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_MOTION_ACCELERATION, HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
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_SENSOR_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_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_SENSOR_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_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_REPORT_LATENCY,
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_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),

//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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_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_UNIT_EXPONENT(0x0E), // scale default unit Gs to centi-Gs to provide 2 digits past Gs decimal point
HID_INPUT(Data_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_UNIT_EXPONENT(0x0E), // scale default unit Gs to centi-Gs to provide 2 digits past Gs decimal point
HID_INPUT(Data_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_UNIT_EXPONENT(0x0E), // scale default unit Gs to centi-Gs to provide 2 digits past Gs decimal point
HID_INPUT(Data_Var_Abs),
HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
HID_LOGICAL_MIN_8(0x00),
HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(2),
HID_UNIT_EXPONENT(0x09),
HID_INPUT(Data_Var_Abs),

HID_END_COLLECTION
};

```

Motion: Linear Accelerometer

```

// For reference: Complete HID report descriptor
const unsigned char linear_acc_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_MOTION_LINEAR_ACCELEROMETER,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_POWER_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_POWER_STATE_UNDEFINED,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
    HID_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_MOTION_ACCELERATION, HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
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_SENSOR_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_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_SENSOR_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_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_REPORT_LATENCY,
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_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),

//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(Data_Arr_Abs),
    HID_END_COLLECTION,
    HID_USAGE_SENSOR_EVENT,
    HID_LOGICAL_MIN_8(0),
    HID_LOGICAL_MAX_8(5),
    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_INPUT(Data_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_UNIT_EXPONENT(0x0E), // scale default unit Gs to centi-Gs to provide 2 digits past Gs decimal point
    HID_INPUT(Data_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_UNIT_EXPONENT(0x0E), // scale default unit Gs to centi-Gs to provide 2 digits past Gs decimal point
    HID_INPUT(Data_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_UNIT_EXPONENT(0x0E), // scale default unit Gs to centi-Gs to provide 2 digits past Gs decimal point
    HID_INPUT(Data_Var_Abs),
    HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
    HID_LOGICAL_MIN_8(0x00), // LOGICAL_MINIMUM (-32767)
    HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF), // LOGICAL_MAXIMUM (32767)
    HID_REPORT_SIZE(32),
    HID_REPORT_COUNT(2),
    HID_UNIT_EXPONENT(0x09), // scale default unit Gs to centi-Gs to provide 2 digits past Gs decimal point
    HID_INPUT(Data_Var_Abs),
    HID_END_COLLECTION
};

```

Orientation: Relative Orientation

```

// For reference: Complete HID report descriptor
const unsigned char relative_orientation_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_ORIENTATION_RELATIVE_ORIENTATION,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_POWER_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_POWER_STATE_UNDEFINED,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
        HID_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_UNIT_EXPONENT(0),
    HID_FEATURE(Data_Var_Abs),
    HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
    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_DATA(HID_USAGE_SENSOR_DATA_ORIENTATION, HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
    HID_LOGICAL_MIN_8(0),
    HID_LOGICAL_MAX_16(0xFF, 0xFF),
    HID_REPORT_SIZE(16),
    HID_REPORT_COUNT(1),
    HID_UNIT_EXPONENT(0x0E),
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    HID_USAGE_SENSOR_EVENT,
    HID_LOGICAL_MIN_8(0),
    HID_LOGICAL_MAX_8(5),
    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_INPUT(Data_Arr_Abs),
    HID_END_COLLECTION,
    HID_USAGE_SENSOR_DATA_ORIENTATION_ROTATION_MATRIX,
    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(9),
    HID_UNIT_EXPONENT(0x0E),
    HID_INPUT(Data_Var_Abs),
    HID_USAGE_SENSOR_DATA_ORIENTATION_QUATERNION,
    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(4),
    HID_UNIT_EXPONENT(0x0E),
    HID_INPUT(Data_Var_Abs),

    HID_END_COLLECTION,
};

```

Orientation: Simple Orientation

```
// For reference: Complete HID report descriptor
const unsigned char simple_orientation_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_ORIENTATION_SIMPLE_ORIENTATION,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_POWER_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_POWER_STATE_UNDEFINED,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
        HID_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_UNIT_EXPONENT(0),
    HID_FEATURE(Data_Var_Abs),
    HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
    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),

    //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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_ORIENTATION_SIMPLE_ORIENTATION_DIRECTION,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
HID_REPORT_SIZE(8),
HID_REPORT_COUNT(1),
HID_COLLECTION(Logical)
    HID_USAGE_SENSOR_DATA_ORIENTATION_SIMPLE_ORIENTATION_DIRECTION_NOT_ROTATED
    HID_USAGE_SENSOR_DATA_ORIENTATION_SIMPLE_ORIENTATION_DIRECTION_FACE_UP
    HID_USAGE_SENSOR_DATA_ORIENTATION_SIMPLE_ORIENTATION_DIRECTION_FACE_DOWN
    HID_USAGE_SENSOR_DATA_ORIENTATION_SIMPLE_ORIENTATION_DIRECTION_ROTATED_90_DEGREES_CCW
    HID_USAGE_SENSOR_DATA_ORIENTATION_SIMPLE_ORIENTATION_DIRECTION_ROTATED_180_DEGREES_CCW
    HID_USAGE_SENSOR_DATA_ORIENTATION_SIMPLE_ORIENTATION_DIRECTION_ROTATED_270_DEGREES_CCW
    HID_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(2),
HID_UNIT_EXPONENT(0x09),
HID_INPUT(Data_Var_Abs),

HID_END_COLLECTION
};

```

Orientation Extended: Geomagnetic Orientation

```

// For reference: Complete HID report descriptor
const unsigned char geomagnetic_orientation_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_ORIENTATION_EXTENDED_GEOMAGNETIC_ORIENTATION,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_POWER_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_POWER_STATE_UNDEFINED,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
        HID_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
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_DATA(HID_USAGE_SENSOR_DATA_ORIENTATION, HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0x0E),
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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_ORIENTATION_ROTATION_MATRIX,
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(9),
HID_UNIT_EXPONENT(0x0E),
HID_INPUT(Data_Var_Abs),
HID_USAGE_SENSOR_DATA_ORIENTATION_QUATERNION,
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(4),
HID_UNIT_EXPONENT(0x0E),
HID_INPUT(Data_Var_Abs),

HID_END_COLLECTION,
};

```

Orientation Extended: Magnetometer

```

// For reference: Complete HID report descriptor
const unsigned char magnetometer_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_ORIENTATION_MAGNETOMETER,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)

```

```

HID_USAGE_PAGE_SENSOR,
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(Data_Arr_Abs),
HID_END_COLLECTION,
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_POWER_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_POWER_STATE_UNDEFINED,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
    HID_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_ORIENTATION_MAGNETIC_FLUX, HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0x0E), // scale to provide 2 digits past decimal point
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_ORIENTATION_MAGNETIC_FLUX, HID_USAGE_SENSOR_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_UNIT_EXPONENT(0x0E), // scale to provide 2 digits past decimal point
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_ORIENTATION_MAGNETIC_FLUX, HID_USAGE_SENSOR_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_UNIT_EXPONENT(0x0E), // scale to provide 2 digits past decimal point
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(Data_Arr_Abs),
    HID_END_COLLECTION,
    HID_USAGE_SENSOR_EVENT,
    HID_LOGICAL_MIN_8(0),
    HID_LOGICAL_MAX_8(5),
    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_INPUT(Data_Arr_Abs),
    HID_END_COLLECTION,
    HID_USAGE_SENSOR_DATA_ORIENTATION_MAGNETIC_FLUX_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_UNIT_EXPONENT(0x0E), // scale to provide 2 digits past decimal point
    HID_INPUT(Data_Var_Abs),
    HID_USAGE_SENSOR_DATA_ORIENTATION_MAGNETIC_FLUX_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_UNIT_EXPONENT(0x0E), // scale to provide 2 digits past decimal point
    HID_INPUT(Data_Var_Abs),
    HID_USAGE_SENSOR_DATA_ORIENTATION_MAGNETIC_FLUX_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_UNIT_EXPONENT(0x0E), // scale to provide 2 digits past decimal point
    HID_INPUT(Data_Var_Abs),
    HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
    HID_LOGICAL_MIN_8(0x00), // LOGICAL_MINIMUM (-32767)
    HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF), // LOGICAL_MAXIMUM (32767)
    HID_REPORT_SIZE(32),
    HID_REPORT_COUNT(2),
    HID_UNIT_EXPONENT(0x0E), // scale to provide 2 digits past decimal point
    HID_INPUT(Data_Var_Abs),
    HID_END_COLLECTION
};

```

Personal Activity: Activity Detection

```

// For reference: Complete HID report descriptor
// Activity Detection
const unsigned char activity_detection_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_PERSONAL_ACTIVITY_ACTIVITY_DETECTION,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_POWER_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_POWER_STATE_UNDEFINED,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
    HID_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
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_DATA(HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE,
    HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MAX_POWER_CONSUMPTION,
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_PERSONAL_ACTIVITY_SUPPORTED_ACTIVITY_TYPES,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(7),
HID_REPORT_SIZE(8),
HID_REPORT_COUNT(8),
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_UNKNOWN,
    HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_IDLE,
    HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_STATIONARY,
    HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_FIDGETING,
    HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_WALKING,
    HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_RUNNING,
    HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_IN_VEHICLE,
    HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_BIKING,
    HID_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_PROPERTY_PERSONAL_ACTIVITY_SUBSCRIBED_ACTIVITY_TYPES,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(7),
HID_REPORT_SIZE(8),
HID_REPORT_COUNT(8),
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_UNKNOWN,
    HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_IDLE,
    HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_STATIONARY,
    HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_FIDGETING,
    HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_WALKING,
    HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_RUNNING,
    HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_IN_VEHICLE,
    HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_BIKING,
    HID_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,

//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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_UNKNOWN,
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_UNKNOWN, HID_USAGE_SENSOR_DATA_MOD_ACCURACY),
    LOGICAL_MIN(0),
    LOGICAL_MAX(0x64),
    REPORT_COUNT(1),
    REPORT_SIZE(8),
    HID_INPUT(Data_Var_Abs),

    HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_ACTIVITY_STATE,
    LOGICAL_MIN(0),
    LOGICAL_MAX(2),
    REPORT_COUNT(1),
    REPORT_SIZE(8),
    HID_COLLECTION(Logical),
        HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_NO_STATE_CHANGE,
        HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_START_ACTIVITY,
        HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_END_ACTIVITY,
        HID_INPUT(Data_Arr_Abs),
    HID_END_COLLECTION,
HID_END_COLLECTION,

HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_IDLE,
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_IDLE, HID_USAGE_SENSOR_DATA_MOD_ACCURACY),
    LOGICAL_MIN(0),
    LOGICAL_MAX(0x64),
    REPORT_COUNT(1),
    REPORT_SIZE(8),
    HID_INPUT(Data_Var_Abs),

    HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_ACTIVITY_STATE,
    LOGICAL_MIN(0),
    LOGICAL_MAX(2),
    REPORT_COUNT(1),
    REPORT_SIZE(8),
    HID_COLLECTION(Logical),
        HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_NO_STATE_CHANGE,
        HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_START_ACTIVITY,
        HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_END_ACTIVITY,
        HID_INPUT(Data_Arr_Abs),
    HID_END_COLLECTION,
HID_END_COLLECTION,

HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_STATIONARY,
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_STATIONARY, HID_USAGE_SENSOR_DATA_MOD_ACCURACY),
    LOGICAL_MIN(0),
    LOGICAL_MAX(0x64),
    REPORT_COUNT(1),
    REPORT_SIZE(8),
    HID_INPUT(Data_Var_Abs),

    HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_ACTIVITY_STATE,
    LOGICAL_MIN(0),
    LOGICAL_MAX(2),
    REPORT_COUNT(1),
    REPORT_SIZE(8),
    HID_COLLECTION(Logical),
        HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_NO_STATE_CHANGE,
        HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_START_ACTIVITY,
        HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_END_ACTIVITY,
        HID_INPUT(Data_Arr_Abs),
    HID_END_COLLECTION,
HID_END_COLLECTION,

HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_FIDGETING,
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_FIDGETING, HID_USAGE_SENSOR_DATA_MOD_ACCURACY),
    LOGICAL_MIN(0),

```

```

LOGICAL_MAX(0x64),
REPORT_COUNT(1),
REPORT_SIZE(8),
HID_INPUT(Data_Var_Abs),

HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_ACTIVITY_STATE,
LOGICAL_MIN(0),
LOGICAL_MAX(2),
REPORT_COUNT(1),
REPORT_SIZE(8),
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_NO_STATE_CHANGE,
    HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_START_ACTIVITY,
    HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_END_ACTIVITY,
    HID_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_END_COLLECTION,

HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_WALKING,
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_WALKING, HID_USAGE_SENSOR_DATA_MOD_ACCURACY),
    LOGICAL_MIN(0),
    LOGICAL_MAX(0x64),
    REPORT_COUNT(1),
    REPORT_SIZE(8),
    HID_INPUT(Data_Var_Abs),

HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_ACTIVITY_STATE,
LOGICAL_MIN(0),
LOGICAL_MAX(2),
REPORT_COUNT(1),
REPORT_SIZE(8),
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_NO_STATE_CHANGE,
    HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_START_ACTIVITY,
    HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_END_ACTIVITY,
    HID_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_END_COLLECTION,

HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_RUNNING,
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_WALKING, HID_USAGE_SENSOR_DATA_MOD_ACCURACY),
    LOGICAL_MIN(0),
    LOGICAL_MAX(0x64),
    REPORT_COUNT(1),
    REPORT_SIZE(8),
    HID_INPUT(Data_Var_Abs),

HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_ACTIVITY_STATE,
LOGICAL_MIN(0),
LOGICAL_MAX(2),
REPORT_COUNT(1),
REPORT_SIZE(8),
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_NO_STATE_CHANGE,
    HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_START_ACTIVITY,
    HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_END_ACTIVITY,
    HID_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_END_COLLECTION,

HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_IN_VEHICLE,
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_IN_VEHICLE, HID_USAGE_SENSOR_DATA_MOD_ACCURACY),
    LOGICAL_MIN(0),
    LOGICAL_MAX(0x64),
    REPORT_COUNT(1),
    REPORT_SIZE(8),
    HID_INPUT(Data_Var_Abs),

HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_ACTIVITY_STATE,
LOGICAL_MIN(0),
LOGICAL_MAX(2),
REPORT_COUNT(1),
REPORT_SIZE(8),
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_NO_STATE_CHANGE,
    HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_START_ACTIVITY,
    HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_END_ACTIVITY,
    HID_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_END_COLLECTION,

HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_BIKING,
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA(HID_USAGE_SENSOR_DATA_ACTIVITY_TYPE_BIKING, HID_USAGE_SENSOR_DATA_MOD_ACCURACY),
    LOGICAL_MIN(0),
    LOGICAL_MAX(0x64),
    REPORT_COUNT(1),
    REPORT_SIZE(8),
    HID_INPUT(Data_Var_Abs),

HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_ACTIVITY_STATE,
LOGICAL_MIN(0),

```

```

        LOGICAL_MAX(2),
        REPORT_COUNT(1),
        REPORT_SIZE(8),
        HID_COLLECTION(Logical),
            HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_NO_STATE_CHANGE,
            HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_START_ACTIVITY,
            HID_USAGE_SENSOR_DATA_ACTIVITY_STATE_TYPE_END_ACTIVITY,
            HID_INPUT(Data_Arr_Abs),
        HID_END_COLLECTION,
    HID_END_COLLECTION,

    HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
    HID_LOGICAL_MIN_8(0x00),
    HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
    HID_REPORT_SIZE(32),
    HID_REPORT_COUNT(2),
    HID_UNIT_EXPONENT(0x09),
    HID_INPUT(Data_Var_Abs),

    HID_END_COLLECTION,
};

```

Personal Activity: Device Position

```

// For reference: Complete HID report descriptor
const unsigned char pedometer_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_PERSONAL_ACTIVITY_DEVICE_POSITION,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_POWER_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_POWER_STATE_UNDEFINED,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
        HID_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
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_MAX_POWER_CONSUMPTION,
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_DATA(HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_DEVICE_POSITION,
HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_DEVICE_POSITION,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
HID_REPORT_SIZE(8),
HID_REPORT_COUNT(1),
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA_DEVICE_POSITION_UNKNOWN,
    HID_USAGE_SENSOR_DATA_DEVICE_POSITION_UNCHANGED,
    HID_USAGE_SENSOR_DATA_DEVICE_POSITION_ON_DESK,
    HID_USAGE_SENSOR_DATA_DEVICE_POSITION_IN_HAND,
    HID_USAGE_SENSOR_DATA_DEVICE_POSITION_MOVING_IN_BAG,
    HID_USAGE_SENSOR_DATA_DEVICE_POSITION_STATIONARY_IN_BAG,
    HID_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
HID_LOGICAL_MIN_8(0x00),
HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(2),
HID_UNIT_EXPONENT(0x09),
HID_INPUT(Data_Var_Abs),

HID_END_COLLECTION,
};

```

Personal Activity: Pedometer

```

// For reference: Complete HID report descriptor
const unsigned char pedometer_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_PERSONAL_ACTIVITY_PEDOMETER,
    HID_COLLECTION(Physical),

```

```

//feature reports (xmit/receive)
HID_USAGE_PAGE_SENSOR,
HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
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(Data_Arr_Abs),
HID_END_COLLECTION,
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_POWER_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_POWER_STATE_UNDEFINED,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
    HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
    HID_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
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_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
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_MAX_POWER_CONSUMPTION,
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_DATA(HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_STEP_COUNT,
    HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_PERSONAL_ACTIVITY_SUPPORTED_STEP_TYPES,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(2),
HID_REPORT_SIZE(1),
HID_REPORT_COUNT(3),
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA_STEP_TYPE_UNKNOWN,
    HID_USAGE_SENSOR_DATA_STEP_TYPE_WALKING,
    HID_USAGE_SENSOR_DATA_STEP_TYPE_RUNNING,

```

```

        HID_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_PROPERTY_PERSONAL_ACTIVITY_SUBSCRIBED_STEP_TYPES,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(2),
HID_REPORT_SIZE(1),
HID_REPORT_COUNT(3),
HID_COLLECTION(Logical),
        HID_USAGE_SENSOR_DATA_STEP_TYPE_UNKNOWN,
        HID_USAGE_SENSOR_DATA_STEP_TYPE_WALKING,
        HID_USAGE_SENSOR_DATA_STEP_TYPE_RUNNING,
        HID_FEATURE(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_STEP_COUNT_RESET
HID_LOGICAL_MIN_8(0), // false, no reset has occurred
HID_LOGICAL_MAX_8(1), // true, a step count reset has occurred
HID_REPORT_SIZE(8),
HID_REPORT_COUNT(1),
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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_STEP_TYPE_UNKNOWN,
HID_COLLECTION(Logical),
        HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_STEP_COUNT,
        HID_LOGICAL_MIN_8(0),
        HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
        HID_REPORT_SIZE(32),
        HID_REPORT_COUNT(2),
        HID_INPUT(Data_Var_Abs),
        HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_STEP_DURATION,
        HID_LOGICAL_MIN_8(0),
        HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
        HID_REPORT_SIZE(32),
        HID_REPORT_COUNT(2),
        HID_INPUT(Data_Var_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_STEP_TYPE_WALKING,
HID_COLLECTION(Logical),
        HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_STEP_COUNT,
        HID_LOGICAL_MIN_8(0),
        HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
        HID_REPORT_SIZE(32),
        HID_REPORT_COUNT(2),
        HID_INPUT(Data_Var_Abs),
        HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_STEP_DURATION,
        HID_LOGICAL_MIN_8(0),
        HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
        HID_REPORT_SIZE(32),
        HID_REPORT_COUNT(2),
        HID_INPUT(Data_Var_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_STEP_TYPE_RUNNING,
HID_COLLECTION(Logical),
        HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_STEP_COUNT,
        HID_LOGICAL_MIN_8(0),
        HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
        HID_REPORT_SIZE(32),
        HID_REPORT_COUNT(2),
        HID_INPUT(Data_Var_Abs),
        HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_STEP_DURATION,
        HID_LOGICAL_MIN_8(0),
        HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
        HID_REPORT_SIZE(32),

```

```

        HID_REPORT_COUNT(2),
        HID_INPUT(Data_Var_Abs),
    HID_END_COLLECTION,

    HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_STEP_COUNT_RESET
    HID_LOGICAL_MIN_8(0), // false, no reset has occurred
    HID_LOGICAL_MAX_8(1), // true, a step count reset has occurred
    HID_REPORT_SIZE(8),
    HID_REPORT_COUNT(1),
    HID_INPUT(Data_Var_Abs),

    HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
    HID_LOGICAL_MIN_8(0x00),
    HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
    HID_REPORT_SIZE(32),
    HID_REPORT_COUNT(2),
    HID_UNIT_EXPONENT(0x09),
    HID_INPUT(Data_Var_Abs),

    HID_END_COLLECTION,
};

```

Personal Activity: Step Detection

```

// For reference: Complete HID report descriptor
const unsigned char pedometer_report_descriptor[] = {
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_TYPE_PERSONAL_ACTIVITY_STEP_DETECTION,
    HID_COLLECTION(Physical),

    //feature reports (xmit/receive)
    HID_USAGE_PAGE_SENSOR,
    HID_USAGE_SENSOR_PROPERTY_SENSOR_CONNECTION_TYPE,
    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(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_POWER_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_POWER_STATE_UNDEFINED,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D0_FULL_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D1_LOW_POWER,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D2_STANDBY_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D3_SLEEP_WITH_WAKE,
        HID_USAGE_SENSOR_PROPERTY_POWER_STATE_D4_POWER_OFF,
        HID_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_FEATURE(Data_Arr_Abs),
    HID_END_COLLECTION,
    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_UNIT_EXPONENT(0),
};

```

```

HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_MINIMUM_REPORT_INTERVAL,
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_MAX_POWER_CONSUMPTION,
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_DATA(HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_STEP_TYPE,
HID_USAGE_SENSOR_DATA_MOD_CHANGE_SENSITIVITY_ABS),
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_16(0xFF, 0xFF),
HID_REPORT_SIZE(16),
HID_REPORT_COUNT(1),
HID_UNIT_EXPONENT(0),
HID_FEATURE(Data_Var_Abs),
HID_USAGE_SENSOR_PROPERTY_ACTIVITY_SUPPORTED_STEP_TYPES, // uses bit-field where each step type selector
HID_LOGICAL_MIN_8(0), // not supported // maps to a bit, thus report size of 1 and report
HID_LOGICAL_MAX_8(1), // supported // count of 3 (for 3 step types). The first bit
HID_REPORT_SIZE(1), // maps to first selector, in this case step type
HID_REPORT_COUNT(3), // unknown and the last bit maps to step type running.
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_STEP_TYPE_UNKNOWN, // unknown's enum = 0 -> first bit
    HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_WALKING_TYPE_WALKING,
    HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_RUNNING_TYPE_RUNNING,
    HID_FEATURE(Data_Var_Abs),
HID_END_COLLECTION,

//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(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_EVENT,
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(5),
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_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_PERSONAL_ACTIVITY_STEP_TYPE
HID_LOGICAL_MIN_8(0),
HID_LOGICAL_MAX_8(3),
HID_REPORT_SIZE(8),
HID_REPORT_COUNT(1),
HID_COLLECTION(Logical),
    HID_USAGE_SENSOR_DATA_STEP_TYPE_UNKNOWN,
    HID_USAGE_SENSOR_DATA_STEP_TYPE_WALKING,
    HID_USAGE_SENSOR_DATA_STEP_TYPE_RUNNING,
    HID_INPUT(Data_Arr_Abs),
HID_END_COLLECTION,
HID_USAGE_SENSOR_DATA_TIME_TIMESTAMP,
HID_LOGICAL_MIN_8(0x00),
HID_LOGICAL_MAX_32(0xFF, 0xFF, 0xFF, 0xFF),
HID_REPORT_SIZE(32),
HID_REPORT_COUNT(2),
HID_UNIT_EXPONENT(0x09),
HID_INPUT(Data_Var_Abs),

HID_END_COLLECTION,
};

```