HCOMM Reference Manual

Document Number: 1000-2984
Document Revision: 0.3.2
Date: March 5, 2014

Table of Contents

List of Figures

1 Introduction

CEVA's Hillcrest Labs business unit's Freespace® MotionEngine™ is a complete solution for the creation of in-air pointing and motion control devices for a broad range of applications including pointing remote controls, body motion trackers, head mounted displays (HMDs), and mobile and tablet sensor fusion products. The solution includes the core motion sensing technology, multiple reference designs, development tools, manufacturing solutions, design expertise and intellectual property necessary to develop and produce superior motion control products.

An HCOMM hub is an implementation of Hillcrest's Freespace motion control solutions. An FSM-9 is an example of an HCOMM hub. An HCOMM hub contains both MotionEngine® and MotionHub. MotionEngine is Hillcrest's proprietary motion and sensor processing software. MotionHub is Hillcrest's sensor management software. HCOMM is the messaging protocol used to communicate with and control an HCOMM hub. An HCOMM hub is incorporated into a larger product and controlled by the processor of that product.

1.1 Reserved Values

All values marked "Reserved", "reserved", "RESERVED" or some abbreviation thereof must be set to 0 when they are produced. All such reserved values must be ignored when they are consumed.

All unspecified values of parameters are reserved and must not be used.

1.2 Revision History

Revision	Date	Description
0.3	x	Clarified sensor period request in section 5.2.1.7. Added clarifications to the MotionEngine power management record in section 6.2.4.2. Added defaults values for angular position smoothing record values in section 6.2.4.3. Added MotionEngine Output Format 3 to section 5.1.5. Changed reference to RF-on-motion to Notify-on-motion. In section 6.2.4.3 changed parameter description from signed to unsigned. Added default values for parameters to sections 6.2.4.1 and 6.2.4.2. Clarified meanings of activity classification and power management flags in section 5.1.5. Added MotionEngine Output Format 4 to section 5.1.5. Added fusion tick to MotionEngine Output Format 2 in section 5.1.5. Fixed scale factor for angular position in MEOut Format 1 in Figure 16. Added tilt detector.
0.2	03/13/2013	Added FRS records in section 6.2.4. Revised DCE Out V4 packet in section 5.1.8.
0.1	02/08/2013	First draft.

Figure 1: Document History

2 Sensors

MotionEngine works with a variety of sensors including accelerometers, gyroscopes, magnetometers, ambient light sensors, pressure sensors and proximity sensors.

2.1.1 Linear Accelerometers

MotionEngine uses a three axis linear accelerometer. The accelerometer is used for inclinometer functions, orientation compensation, wake-on-motion and other fusion operations. MotionEngine works with a variety of accelerometers. It uses calibration data provided at system design time or during production to obtain the best performance from the actual accelerometer being used.

2.1.2 Angular Velocity Sensors (Gyroscopes)

MotionEngine uses a three axis angular velocity sensor or gyroscope. The gyroscope measures rotation about the X, Y and Z axes of the hub. Angular velocity is an input used to produce cursor motion output, general motion output and other fused outputs. MotionEngine works with a variety of gyroscopes. It uses calibration data provided at system design time or during production to obtain the best performance from the actual gyroscope being used.

2.1.3 Magnetometers

MotionEngine uses a three axis magnetometer. The magnetometer measures the Earth's magnetic field and is used to determine absolute orientation. Absolute orientation can be thought of as determining which direction is north. MotionEngine works with a variety of magnetometers. It uses calibration data provided at system design time to obtain the best performance from the actual magnetometer being used.

2.1.4 Ambient Light Sensors

Future feature

2.1.5 Pressure Sensors

Future feature

2.1.6 Proximity Sensors

Future feature

3 HCOMM Hub Operation

This section describes the operating modes of an HCOMM hub.

3.1 Operating Modes

The FSP has several operating modes for power management. They are summarized in Figure 2.

Mode	Sensors	Outputs
Full Motion (FM)	On	Motion
Wake on Motion (WM)	Accelerometer in WM	Activity classification
Sleep(SL)	Off	None
Full Motion ON (FMON)	On	Motion
Notify on Motion (NM)	Accelerometer in WM	Activity classification

Figure 2: Summary of Operating Modes

Operating modes are changed by the FSP power management system or by a control packet from the host.

3.1.1 Full Motion

Full motion mode is used when motion output is desired. While in full motion, if the hub detects that it has becomes motionless (still) it will enter wake-on-motion.

3.1.2 Wake-on-Motion

In wake on motion mode, the accelerometer is used to detect large movements, such as a user picking up the device from a table, and causes the hub to enter full motion mode. Wake on motion is a low power mode with minimal motion sensing.

3.1.3 Sleep

Sleep mode is a very low power mode. No motion sensing is performed in sleep mode.

3.1.4 Notify-on-Motion

In NM on motion mode, the accelerometer is used to detect large movements, such as a user picking up the device from a table. The host processor is notified when motion is detected. Unlike WM, the hub remains in NM even after detecting motion. It is the responsibility of the host to change the operating mode of the hub when it is in NM. NM on motion is a low power mode with minimal motion sensing.

3.1.5 Full Motion On

Full motion on mode is used when motion output is desired, even if the hub detects that it has becomes motionless (still). The hub will not transition from FMON into WM. It is the responsibility of the host to change the operating mode of the hub when it is in FMON.

3.2 Host Interfaces

The host device connects to a hub via a physical interface. Supported interfaces include SPI and USB.

The hub exchanges packets with a host using the Hillcrest communication protocol (HCOMM). Over SPI it uses HCOMM SPI, and over USB it uses HCOMM USB. The hub sends packets formatted as USB HID reports for mouse and vendor-specific usages.

3.3 Device Firmware Upgrade

The hub supports in-field device firmware upgrade (DFU). DFU can be used to add new features and fix bugs. New firmware is provided to the hub in the form of an encrypted binary image.

4 HCOMM Overview

The Hillcrest communication protocol (HCOMM) provides a data passing scheme that can be used over all physical interfaces, including the system host.

As HCOMM packets are transported through a system they may transition into another system domain, such as USB HID. The HID domain has requirements that packets of a given class contain a class-specific HID report ID and have a consistent, pre-determined length. See Reference 1 and Reference 2 for more information regarding USB HID.

HCOMM packets are structured in such a manner as to be compatible with HID. The HCOMM class is synonymous with the HID report ID. HCOMM packets may need to be lengthened when transitioning into the HID domain and may need to be shortened when transitioning from the HID domain, as is the case if the stated length of the HCOMM packet is less than the maximum length of the class of the packet.

The format of HCOMM packets is shown in Figure 3.

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	Class							
1	Length							
2	Reserved	Reserved	Destination					
3	Reserved	Reserved	Source					
4-N	Data							

Figure 3: HCOMM Packet Format

Class:	indicates what sort of information this packet contains; this field is synonymous with the HID report ID
Length:	the length of the bytes to be sent using HCOMM, including the Class and Length bytes; all bytes specified by the length are significant and data-bearing. This length may be less than or equal to the maximum class length.
Destination:	contains the address of the packet recipient
Source	contains the address of the packet originator
Data	payload data

The addresses used by HCOMM packets are:

- 0 – host. This is typically a PC but could be the control processor connected to the hub
- 1 – dongle.
- 4 – hub.

5 HCOMM Packets

This section describes all HCOMM packet payloads.

All fields greater than 8 bits wide use network byte ordering (little-endian byte ordering) unless otherwise specified.

5.1 Inbound Packets

This section covers packets that are sent from a hub to a host.

5.1.1 Mouse Packet (Class 0x02)

This packet is a HID mouse packet with 8 buttons and a scroll wheel.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x02							
1	Length							
2	Destination							
3	Source							
4	Button 8	Button 7	Button 6	Button 5	Button 4	Button 3	Button 2	Button 1
5	Delta X							
6	Delta Y							
7	Delta Wheel							

Figure 4: Mouse Movement Packet

Button n: 0 – not pressed
1 – pressed

Delta X: change in pointer location along the x-axis in dimensionless units called mickeys. X positive moves the pointer right.

Delta Y: change in pointer location along the y-axis in dimensionless units called mickeys. Y positive moves the pointer down.

Delta wheel: scroll wheel motion in detents (up is positive).

5.1.2 DataIn Packet (Class 0x05)

This packet is used as a container for command, control, status, testing and debugging information. Each packet contains a subclass. The subclass determines the type of information carried by the DataIn packet. All subclass values are reserved with the exception of those described in this section.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x05							
1	Length							
2	Destination							
3	Source							
4	Subclass							
5-11	Payload							

Figure 5: DataIn Packet

Subclass: identifies the specific type of information being carried within the DataIn packet

Payload: the remaining report payload, up to 7 bytes

5.1.2.1 FRS Write Response (Subclass 0x06)

This packet is sent to indicate status of the write operation of the flash record system (FRS). See section 6 for details of the flash record system.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x05							
1	Length							
2	Destination							
3	Source							
4	Subclass = 0x06							
5	Status/Error							
6	Word Offset LSB							
7	Word Offset MSB							

Figure 6: FRS Write Response Packet

Status/Error:
0 – word received
1 – unrecognized FRS type
2 – busy
3 – write completed
4 – write mode entered or ready
5 – write failed
6 – data received while not in write mode
7 – invalid length
8 – record valid (the complete record passed internal validation checks)
9 – record invalid (the complete record failed internal validation checks)
10 – device error (DFU flash memory device unavailable)

Word Offset: the number of words offset from the beginning of the record

5.1.2.2 Freespace Data Mode Control V2 Response (Subclass 0x14)

This packet reports the configuration of the generation of Freespace motion packets and the operating mode of the device.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x05							
1	Length							
2	Destination							
3	Source							
4	Subclass = 0x14							
5	Operating Mode Actual			Output Status	Operating Mode Requested			Operating Status
6	Packet Select							
7	Format Select							
8	FF7	FF6	FF5	FF4	FF3	FF2	FF1	FF0
9-11	Reserved							

Figure 7: Freespace Data Mode Control V2 Response Packet

Operating Status	0 – indicates a response to an operating mode update request 1 – indicates a response to an operating mode status request
Operating Mode Requested	Requests a change to the specified operating mode. The hub will enter the requested mode if allowed. The hub may leave the requested operating mode autonomously. 0 – Full Motion 1 – Sleep 2 – Deprecated 3 – Deprecated 4 – Full Motion On 5 – Notify-on-Motion
Output Status	0 – indicates a response to a packet and format selection update request 1 – indicates a response to a packet and format selection status request
Operating Mode Actual	The operating mode the device is currently in. The actual operating mode is reported independently of the requested operating mode. The actual operating mode reports the operating modes discussed in section 3.1. 0 – Full Motion 1 – Wake-on-Motion 2 – Reserved 3 – Sleep 4 – Reserved 5 – Notify-on-Motion
Packet select	Select the output packet sent by the FSP to report motion data. 0 – none 1 – Mouse

	2 – Reserved 3 – Reserved 4 – Reserved 5 – DCE out 6 – Reserved 7 – SuperSDA 8 – MotionEngine Output 9 – MotionEngine Output and Mouse 10-255 – reserved
Format Select	Select the format of the MotionEngine Output packet. See 5.1.5 for details.
FF7 – FF0	Format Flags – used to enable various portions of a MotionEngine Output packet. See 5.1.5 for details.

5.1.2.3 Sensor Period Response (Subclass 0x17)

This packet reports the sample period for a given sensor, including the sensor fusion operation. Sensor fusion is the operation of producing a motion output by using the inputs from several sensors. Sensor fusion may be performed at a rate different from the sample period of the sensors.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x05							
1	Length							
2	Destination							
3	Source							
4	Subclass = 0x17							
5	Reserved							
6	Sensor							
7	Reserved							
8	Period LSB							
9	…							
10	…							
11	Period MSB							

Figure 8: Sensor Period Response

Sensor	The sensor for which the period is being reported. 0 – accelerometer 1 – gyroscope 2 – magnetometer 3 – ambient light sensor 4 – pressure sensor 5 – proximity sensor 6 – sensor fusion 7 – step detector 8 – tap detector 9 – humidity sensor

	10 – ambient temperature sensor 11 – significant motion detector 12 – shake detector 13 – screen rotation accelerometer 14 – flip detector 15 – pickup detector 16 – stability detector 17 – personal activity classifier 18 – sleep detector 19 – gravity 20 – linear acceleration 21 – rotation vector 22 – game rotation vector 23 – geo magnetic rotation vector 24 – step counter 25 – gyroscope uncalibrated 26 – magnetometer uncalibrated 27 – tilt detector 28 – pocket detector 29 – circle detector 30-255 – reserved
Period	Sensor period in microseconds.

5.1.3 DataInLong Packet (Class 0x06)

This packet is used as a container for command, control, status, testing and debugging information. This packet is used when more data needs to be sent than the DataIn packet can accommodate. The DataIn packet is more power efficient, however, and can be sent more reliably over certain communication links, such as an RF link. Each packet contains a subclass. The subclass determines the type of information carried by the DataInLong packet. All subclass values are reserved with the exception of those described in this section.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x06							
1	Length							
2	Destination							
3	Source							
4	Subclass							
5-21	Payload							

Figure 9: DataInLong Packet

Subclass:	identifies the specific type of information being carried within the DataInLong packet
Payload:	the remaining report payload, up to 17 bytes

5.1.3.1 FRS Read Response (Subclass 0x08)

This packet is sent to convey an FRS record or portion of a record.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x06							
1	Length							
2	Destination							
3	Source							
4	Subclass = 0x08							
5	Data Length				Status			
6	Word Offset LSB							
7	Word Offset MSB							
8-19	Data							
20	FRS Type LSB							
21	FRS Type MSB							

Figure 10: FRS Read Response Packet

Data Length: the number of data words contained within the message, typically 3 words.

Status:
0 – no error
1 – unrecognized FRS type
2 – busy
3 – read record completed
4 – offset out of range
5 – record empty
6 – read block completed (if block size requested)
7 – read block completed and read record completed (if block size requested)
8 – device error (DFU flash memory device unavailable)

Word Offset: the number of words offset from the beginning of the record

Data: between 0 and 3 32-bit words of data from and FRS record.

FRS Type: indicates to which type of FRS record the data belongs (see Section 6.1)

5.1.3.2 Product ID Response (Subclass 0x09)

The product ID response carries information that identifies software running on the FSP and FSP unique ID information. This packet is sent automatically when an FSP boots and in response to a product ID request (Section 5.2.1.4).

Byte	Bit 7	6	5	4	3	2	1	0
0	Class = 0x06							
1	Length							
2	Destination							
3	Source							
4	Subclass = 0x09							
5	InvalidSN	Startup	Device Class					
6	SW Version Major							
7	SW Version Minor							
8	SW Part Number LSB							
9	SW Part Number …							
10	SW Part Number …							
11	SW Part Number MSB							
12	SW Build Number LSB							
13	SW Build Number …							
14	SW Build Number …							
15	SW Build Number MSB							
16	Serial Number LSB							
17	Serial Number …							
18	Serial Number …							
19	Serial Number MSB							
20	SW Version Patch LSB							
21	SW Version Patch MSB							

Figure 11: Product ID Response Packet

InvalidSN:	0 –serial number is valid 1 – serial number is invalid; retry until valid
Startup:	0 – device has not just started up 1 – device has just started up. This bit self clears after the first time this message is sent.
Device Class:	0 – device type not known 1 – non-data-generating device 2 – data-generating device 3 – reserved
SW Version:	software version major (8 bits).minor (8 bits).patch (16 bits)
SW Part Number:	32-bit value representing the software part number
SW Build Number:	32-bit software build number
Serial Number:	32-bit serial number. Also referred to as a tracking number.

5.1.4 DataInBLE Packet (Class 0x08)

This packet is used as a container for command, control, status, testing and debugging information. This packet is sized to fit in a Bluetooth Low Energy packet payload. Each packet contains a subclass. The subclass determines the type of information carried by the DataInBLE packet. All subclass values are reserved with the exception of those described in this section.

Byte	Bit 7	6	5	4	3	2	1	0
0	Class = 0x08							
1	Length							
2	Destination							
3	Source							
4	Subclass							
5-19	Payload							

Figure 12: DataInBLE Packet

Subclass: identifies the specific type of information being carried within the DataInBLE packet
Payload: the remaining report payload, up to 15 bytes

5.1.4.1 BLE FRS Read Response (Subclass 0x15)

This packet is sent to convey an FRS record or portion of a record.

Byte	Bit 7	6	5	4	3	2	1	0
0	Class = 0x08							
1	Length							
2	Destination							
3	Source							
4	Subclass = 0x15							
5	Data Length				Status			
6	Word Offset LSB							
7	Word Offset MSB							
8-15	Data							
16-17	Reserved							
18	FRS Type LSB							
19	FRS Type MSB							

Figure 13: BLE FRS Read Response Packet

Data Length: the number of data words contained within the message.
Status: 0 – no error
1 – unrecognized FRS type
2 – busy
3 – read record completed
4 – offset out of range
5 – record empty
6 – read block completed (if block size requested)
7 – read block completed and read record completed (if block size requested)
8 – device error (DFU flash memory device unavailable)
Word Offset: the number of words offset from the beginning of the record
Data: between 0 and 2 32-bit words of data from and FRS record.
FRS Type: indicates to which type of FRS record the data belongs (see Section 6.1)

5.1.4.2 BLE Product ID Response (Subclass 0x16)

The product ID response carries information that identifies software running on the FSP and FSP unique ID information. This packet is sent automatically when an FSP boots and in response to a product ID request (Section 5.2.1.4).

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x08							
1	Length							
2	Destination							
3	Source							
4	Subclass = 0x16							
5	InvalidSN	Startup	Device Class					
6	SW Version Major							
7	SW Version Minor							
8	SW Part Number LSB							
9	SW Part Number …							
10	SW Part Number …							
11	SW Part Number MSB							
12	Serial Number LSB							
13	Serial Number …							
14	Serial Number …							
15	Serial Number MSB							
16	SW Build Number LSB							
17	SW Build Number MSB							
18	SW Version Patch LSB							
19	SW Version Patch MSB							

Figure 14: BLE Product ID Response Packet

InvalidSN:	0 – serial number is valid
	1 – serial number is invalid; retry until valid
Startup:	0 – device has not just started up
	1 – device has just started up. This bit self clears after the first time this message is sent.
Device Class:	0 – device type not known
	1 – non-data-generating device
	2 – data-generating device
	3 – reserved
SW Version:	software version major (8 bits).minor (8 bits).patch (16 bits)
SW Part Number:	32-bit value representing the software part number
SW Build Number:	16-bit software build number
Serial Number:	32-bit serial number. Also referred to as a tracking number.

5.1.5 MotionEngine Output (Class 0x26)

This packet conveys the Motion Engine output. It has a flexible format. Format selection is controlled by the Data Mode Control V2 Request. See 5.2.1.6. The format flags bits are used to enable various portions of the packet. If a particular flag is set then that portion of the output is present. If it is not set then that portion is not present. If a portion of the output is not present then the remaining fields are moved ahead in the packet to fill in the gap left by the missing portion. In other words, the packet is packed. Trailing zeros are omitted. By using the format flags, an application can tailor the MotionEngine output to best suit its needs as well as minimize HCOMM packet sizes and communication times. Some formats and some feature flags may not be supported on a specific platform. Also, some platforms may have a size limitation and therefore are unable to support the full 54 bytes of the MotionEngine Output message. If more feature flags are enabled than will fit into a platform's message size limit then only the ones that fit will actually be set. The others will be ignored. The priority of setting flags starts from FF0 (highest) down to FF6 (lowest).

Motion outputs are formatted as 16-bit, 2's complement, fixed-point. The binary point of each type of data is different and is specified in the field descriptions below. The binary point is indicated by Qn where n is the number of fractional bits. Thus, Q10 indicates 1 sign bit, 5 integer bits and 10 fractional bits.

The Format 0 packet is shown below.

Byte	Bit 7	6	5	4	3	2	1	0
0	Class = 0x26							
1	Length							
2	Destination							
3	Source							
4	Format Select = 0							
5	FF7	FF6	FF5	FF4	FF3	FF2	FF1	FF0
6	Sequence Number LSB							
7	…							
8	…							
9	Sequence Number MSB							
	FF0 Section							
10	Button8	Button7	Button6	Button5	Button4	Button3	Button2	Button1
11	Pointer Delta X							
12	Reserved							
13	Pointer Delta Y							
14	Reserved							
15	Delta Wheel							
	FF1 Section							
16	Linear Acceleration X LSB							
17	Linear Acceleration X MSB							
18	Linear Acceleration Y LSB							
19	Linear Acceleration Y MSB							
20	Linear Acceleration Z LSB							
21	Linear Acceleration Z MSB							
	FF2 Section							
22	Linear Acceleration No Gravity X LSB							
23	Linear Acceleration No Gravity X MSB							
24	Linear Acceleration No Gravity Y LSB							
24	Linear Acceleration No Gravity Y MSB							
26	Linear Acceleration No Gravity Z LSB							
27	Linear Acceleration No Gravity Z MSB							
	FF3 Section							
28	Angular Velocity X LSB							
29	Angular Velocity X MSB							
30	Angular Velocity Y LSB							
31	Angular Velocity Y MSB							
32	Angular Velocity Z LSB							
33	Angular Velocity Z MSB							
	FF4 Section							
34	Magnetometer X LSB							
35	Magnetometer X MSB							
36	Magnetometer Y LSB							
37	Magnetometer Y MSB							
38	Magnetometer Z LSB							
39	Magnetometer Z MSB							

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	FF5 Section							
40	Temperature LSB							
41	Temperature MSB							
	FF6 Section							
42	Angular Position A LSB							
43	Angular Position A MSB							
44	Angular Position B LSB							
45	Angular Position B MSB							
56	Angular Position C LSB							
47	Angular Position C MSB							
48	Angular Position D LSB							
49	Angular Position D MSB							
	FF7 Section							
50	Reserved							
51	Reserved							
52	Reserved							
53	Reserved							

Figure 15: MotionEngine Output Packet, Format 0

Format Select	Indicates the format of the MotionEngine Output packet
Format Flags	Indicates which portions of the MotionEngine Output packet are present.
Sequence Number:	monotonically increasing integer
Button n:	0 – not pressed 1 – pressed
Delta X:	Q0. Changes in pointer location along the x-axis in dimensionless units called mickeys. X positive moves the pointer right.
Delta Y:	Q0. Changes in pointer location along the y-axis in dimensionless units called mickeys. Y positive moves the pointer down.
Delta wheel:	scroll wheel motion in detents. Up is positive.
Linear Acceleration	Q10. Reported in meters / second^2. X positive is forward, Y positive is to the right, and Z positive is down with respect to the handheld device frame of reference.
Linear Acceleration No Gravity	Q10. Reported meters / second^2. X positive is forward, Y positive is to the right, and Z positive is down with respect to the handheld device frame of reference.
Angular Velocity	Q10. Reported in radians / second. X positive is tilt right (roll), Y positive is tilt up (pitch), and Z positive is turn right (yaw), with respect to the handheld device frame of reference.
Magnetometer	Q12. Reported in gauss. X positive is forward, Y positive is to the right and Z positive is down with respect to the handheld device frame of reference.
Temperature	Q7. Reported in degrees Celsius.

Angular Position:	Q14. Reported in dimensionless units. The axes are given in quaternion form where A, B, C, and D represent the real, i, j, and k coefficients
Reserved	Reserved.

The Format 1 packet is used to retrieve data in the format used by Windows 8. The Format 1 packet is shown below.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x26							
1	Length							
2	Destination							
3	Source							
4	Format Select = 1							
5	FF7	FF6	FF5	FF4	FF3	FF2	FF1	FF0
6	Sequence Number LSB							
7	…							
8	…							
9	Sequence Number MSB							
	FF0 Section							
10	Acceleration X LSB							
11	Acceleration X MSB							
12	Acceleration Y LSB							
13	Acceleration Y MSB							
14	Acceleration Z LSB							
15	Acceleration Z MSB							
	FF1 Section							
16	Linear Acceleration X LSB							
17	Linear Acceleration X MSB							
18	Linear Acceleration Y LSB							
19	Linear Acceleration Y MSB							
20	Linear Acceleration Z LSB							
21	Linear Acceleration Z MSB							
	FF2 Section							
22	Angular Velocity X LSB							
23	Angular Velocity X MSB							
24	Angular Velocity Y LSB							
24	Angular Velocity Y MSB							
26	Angular Velocity Z LSB							
27	Angular Velocity Z MSB							
	FF3 Section							
28	Magnetometer X LSB							
29	Magnetometer X MSB							
30	Magnetometer Y LSB							
31	Magnetometer Y MSB							
32	Magnetometer Z LSB							
33	Magnetometer Z MSB							
	FF4 Section							
34	Inclination X LSB							
35	Inclination X MSB							
36	Inclination Y LSB							
37	Inclination Y MSB							
38	Inclination Z LSB							
39	Inclination Z MSB							

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	FF5 Section							
40	Compass Heading LSB							
41	Compass Heading MSB							
	FF6 Section							
42	Angular Position X LSB							
43	Angular Position X MSB							
44	Angular Position Y LSB							
45	Angular Position Y MSB							
46	Angular Position Z LSB							
47	Angular Position Z MSB							
48	Angular Position W LSB							
49	Angular Position W MSB							
	FF7 Section							
50	Activity Classification							
51	Power Management Flags							

Figure 16: MotionEngine Output Packet, Format 1

Format Select	Indicates the format of the MotionEngine Output packet
Format Flags	Indicates which portions of the MotionEngine Output packet are present.
Sequence Number:	monotonically increasing integer generated by the HCOMM Hub.
Acceleration	The total acceleration of the device. Reported in units of 0.01g. X positive is to the right, Y positive is forward, and Z positive is up with respect to the handheld device frame of reference.
Linear Acceleration	The linear acceleration of the device minus gravity. Reported in units of 0.01g. X positive is to the right, Y positive is forward, and Z positive is up with respect to the handheld device frame of reference.
Angular Velocity	Reported in units of 0.1 degrees / second. X positive is tilt up (pitch), Y positive is tilt right (roll), and Z positive is turn left (yaw), with respect to the handheld device frame of reference.
Magnetometer	Reported in units of 0.001 gauss. X positive is forward, Y positive is to the right and Z positive is down with respect to the handheld device frame of reference.
Inclination	Reported in units of 0.1 degrees. X positive is tilt up (pitch), Y positive is tilt right (roll), and Z positive is turn left (yaw), with respect to the handheld device frame of reference.
Compass heading	Reported in units of 0.1 degrees. The compass output is in the range of 0-359.9°.
Angular Position:	Reported in dimensionless units of 0.0001. The axes are given in quaternion form where w, x, y, and z represent the real, i, j, and k coefficients

Activity Classification	Activity classification is the state if the device based on its movement. Activity classification is updated only when MotionEngine is running. See section 6.2.4.2 for information about motion thresholds and power management configurations. 0 – unknown 1 – on table. The hub is at rest on a stable surface with very little vibration. 2 – stationary. The hub’s motion is below the stable threshold but the stable duration requirement has not been met. 3 – stable. The hub’s motion has met the stable threshold and duration requirements. 4 – motion. The hub is moving.
Power Management Flags	Power management flags is a bit field that indicates occurrences of events that can be used to manage power. Flags may be set periodically during normal operation. Multiple flags may be set. See section 6.2.4.2 for information about motion thresholds and power management configurations. Bit 0 – on table. The hub is at rest on a stable surface with very little vibration. Bit 1 – is stable. The hub’s motion has met the stable threshold and duration requirements. Bit 2 – motion detected by the accelerometer Bit 3 – motion detected by MotionEngine

The Format 2 packet is used to retrieve a stream of raw sensor data. The Format 2 packet is shown below.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x26							
1	Length							
2	Destination							
3	Source							
4	Format Select = 2							
5	FF7	FF6	FF5	FF4	FF3	FF2	FF1	FF0
6	Sequence Number LSB							
7	…							
8	…							
9	Sequence Number MSB							
	Data Section (maximum of 44 bytes)							
10-53	Channels							
	Channel format for 3-axis sensor							
N	Channel ID (0-7)							
N + 1	Channel sequence number							
N + 2	X-axis LSB							
N + 3	X-axis MSB							
N + 4	Y-axis LSB							
N + 5	Y-axis MSB							
N + 6	Z-axis LSB							
N + 7	Z-axis MSB							
	Channel format for a 1-axis sensor							
N	Channel ID (0-7)							
N + 1	Channel sequence number							
N + 2	LSB							
N + 3	MSB							

Figure 17: MotionEngine Output Packet, Format 2

Format Select	Indicates the format of the MotionEngine Output packet
Format Flags	Indicates which streams of the MotionEngine Output packet are active. FF0 – accelerometer (3-axis) FF1 – gyroscope (3-axis) FF2 – magnetometer (3-axis) FF3 – ambient light sensor (1-axis) FF4 – pressure sensor (1-axis) FF5 – proximity sensor (1-axis) FF6 – temperature (1-axis) FF7 – fusion tick (1-axis)
Channels:	A channel consists of a channel ID and a channel sequence number followed by either 2 or 6 bytes of sensor data. The data section of a Format 2 packet contains as many channels as will

	fit. There is no fixed order to the channels. A Format 2 packet is constructed by populating it with sensor samples as sensors are sampled and at the rate sensors are sampled. Once the packet is filled, it is sent. For example, a Format 2 packet may contain an accelerometer channel (8 bytes), a gyroscope channel (8 bytes), a temperature channel (4 bytes), a magnetometer channel (8 bytes), a gyroscope channel (8 bytes), a temperature channel (4 bytes) and an empty channel (4 bytes in this case). This set of channels fills the available 44 bytes.
Channel ID	Identifies which channel is being reported. The channel ID's 0-7 correspond to FF0 – FF7. A channel ID of 255 indicates the empty channel. ID's outside of these values are reserved. The empty channel is defined to fill all remaining bytes in the packet. In other words, if the empty channel is encountered there is no more data in the packet.
Channel sequence number	A monotonically increment sequence number for the given channel. Each channel has its own independent sequence number.
X/Y/Z-axis LSB/MSB	The raw sample data for each axis of the sensor.

The Format 3 packet is used for systems with larger ranges for accelerometer and gyroscope output. The tradeoff in using this output is a loss of some resolution. Format 3 also uses µT rather than gauss for the magnetic field. The Format 3 packet is shown below.

Byte	Bit 7	6	5	4	3	2	1	0
0	Class = 0x26							
1	Length							
2	Destination							
3	Source							
4	Format Select = 3							
5	FF7	FF6	FF5	FF4	FF3	FF2	FF1	FF0
6	Sequence Number LSB							
7	…							
8	…							
9	Sequence Number MSB							
	FF0 Section							
10	Button8	Button7	Button6	Button5	Button4	Button3	Button2	Button1
11	Pointer Delta X							
12	Reserved							
13	Pointer Delta Y							
14	Reserved							
15	Delta Wheel							
	FF1 Section							
16	Linear Acceleration X LSB							
17	Linear Acceleration X MSB							
18	Linear Acceleration Y LSB							
19	Linear Acceleration Y MSB							
20	Linear Acceleration Z LSB							
21	Linear Acceleration Z MSB							
	FF2 Section							
22	Linear Acceleration No Gravity X LSB							
23	Linear Acceleration No Gravity X MSB							
24	Linear Acceleration No Gravity Y LSB							
24	Linear Acceleration No Gravity Y MSB							
26	Linear Acceleration No Gravity Z LSB							
27	Linear Acceleration No Gravity Z MSB							
	FF3 Section							
28	Angular Velocity X LSB							
29	Angular Velocity X MSB							
30	Angular Velocity Y LSB							
31	Angular Velocity Y MSB							
32	Angular Velocity Z LSB							
33	Angular Velocity Z MSB							
	FF4 Section							
34	Magnetometer X LSB							
35	Magnetometer X MSB							
36	Magnetometer Y LSB							
37	Magnetometer Y MSB							
38	Magnetometer Z LSB							
39	Magnetometer Z MSB							

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	FF5 Section							
40	Temperature LSB							
41	Temperature MSB							
	FF6 Section							
42	Angular Position A LSB							
43	Angular Position A MSB							
44	Angular Position B LSB							
45	Angular Position B MSB							
56	Angular Position C LSB							
47	Angular Position C MSB							
48	Angular Position D LSB							
49	Angular Position D MSB							
	FF7 Section							
50	Reserved							
51	Reserved							
52	Reserved							
53	Reserved							

Figure 18: MotionEngine Output Packet, Format 3

Format Select	Indicates the format of the MotionEngine Output packet
Format Flags	Indicates which portions of the MotionEngine Output packet are present.
Sequence Number:	monotonically increasing integer
Button n:	0 – not pressed 1 – pressed
Delta X:	Q0. Changes in pointer location along the x-axis in dimensionless units called mickeys. X positive moves the pointer right.
Delta Y:	Q0. Changes in pointer location along the y-axis in dimensionless units called mickeys. Y positive moves the pointer down.
Delta wheel:	scroll wheel motion in detents. Up is positive.
Linear Acceleration	Q8. Reported in meters / second^2. X positive is forward, Y positive is to the right, and Z positive is down with respect to the handheld device frame of reference.
Linear Acceleration No Gravity	Q8. Reported meters / second^2. X positive is forward, Y positive is to the right, and Z positive is down with respect to the handheld device frame of reference.
Angular Velocity	Q9. Reported in radians / second. X positive is tilt right (roll), Y positive is tilt up (pitch), and Z positive is turn right (yaw), with respect to the handheld device frame of reference.
Magnetometer	Q5. Reported in microteslas. X positive is forward, Y positive is to the right and Z positive is down with respect to the handheld device frame of reference.
Temperature	Q7. Reported in degrees Celsius.

Angular Position:	Q14. Reported in dimensionless units. The axes are given in quaternion form where A, B, C, and D represent the real, i, j, and k coefficients
Reserved	Reserved.

The Format 4 packet is used to retrieve a stream of any and all data from the HCOMM Hub. The Format 4 packet is shown below. In addition to the 16-bit number format, the format 4 packet uses a 32-bit, 2's complement, fixed-point format. This format is used for the 1-axis extended sensor. The Q point still indicates the number of fractional bits.

Byte	Bit 7	6	5	4	3	2	1	0
0	Class = 0x26							
1	Length							
2	Destination							
3	Source							
4	Format Select = 4							
5	FF7	FF6	FF5	FF4	FF3	FF2	FF1	FF0
6	Sequence Number LSB							
7	…							
8	…							
9	Sequence Number MSB							
	Data Section (maximum of 44 bytes)							
10-53	Channels							
	Channel format for 3-axis sensor							
N	Channel ID							
N + 1	Channel sequence number							
N + 2	X-axis LSB							
N + 3	X-axis MSB							
N + 4	Y-axis LSB							
N + 5	Y-axis MSB							
N + 6	Z-axis LSB							
N + 7	Z-axis MSB							
	If FF3 Is Enabled							
N + 8	Timestamp LSB							
N + 9	…							
N + 10	…							
N + 11	Timestamp MSB							
	Channel format for a 1-axis sensor							
N	Channel ID							
N + 1	Channel sequence number							
N + 2	LSB							
N + 3	MSB							
	If FF3 Is Enabled							
N + 4	Timestamp LSB							
N + 5	…							
N + 6	…							
N + 7	Timestamp MSB							
	Channel format for 4-axis sensor							
N	Channel ID							
N + 1	Channel sequence number							
N + 2	real coefficient LSB							
N + 3	real coefficient MSB							
N + 4	i coefficient LSB							
N + 5	i coefficient MSB							
N + 6	j coefficient LSB							
N + 7	k coefficient MSB							

N + 8	k coefficient LSB
N + 9	k coefficient MSB
	If FF3 Is Enabled
N + 10	Timestamp LSB
N + 11	…
N + 12	…
N + 13	Timestamp MSB
	Channel format for 1-axis extended sensor
N	Channel ID
N + 1	Channel sequence number
N + 2	LSB
N + 3	…
N + 4	**…**
N + 5	MSB
	If FF3 Is Enabled
N + 6	Timestamp LSB
N + 7	…
N + 8	…
N + 9	Timestamp MSB
	Channel format for 3+3-axis sensor
N	Channel ID
N + 1	Channel sequence number
N + 2	X-axis LSB
N + 3	X-axis MSB
N + 4	Y-axis LSB
N + 5	Y-axis MSB
N + 6	Z-axis LSB
N + 7	Z-axis MSB
N + 8	X-axis meta data LSB
N + 9	X-axis meta data MSB
N + 10	Y-axis meta data LSB
N + 11	Y-axis meta data MSB
N + 12	Z-axis meta data LSB
N + 13	Z-axis meta data MSB
	If FF3 Is Enabled
N + 14	Timestamp LSB
N + 15	…
N + 16	…
N + 17	Timestamp MSB
	Channel format for 3+1-axis sensor
N	Channel ID
N + 1	Channel sequence number
N + 2	X-axis LSB
N + 3	X-axis MSB
N + 4	Y-axis LSB
N + 5	Y-axis MSB
N + 6	Z-axis LSB

N + 7	Z-axis MSB
N + 8	Sensor meta data LSB
N + 9	Sensor meta data MSB
	If FF3 Is Enabled
N + 10	Timestamp LSB
N + 11	…
N + 12	…
N + 13	Timestamp MSB
	Channel format for 4+1-axis sensor
N	Channel ID
N + 1	Channel sequence number
N + 2	real coefficient LSB
N + 3	real coefficient MSB
N + 4	i coefficient LSB
N + 5	i coefficient MSB
N + 6	j coefficient LSB
N + 7	j coefficient MSB
N + 8	k coefficient LSB
N + 9	k coefficient MSB
N +10	Sensor meta data LSB
N +11	Sensor meta data MSB
	If FF3 Is Enabled
N + 12	Timestamp LSB
N + 13	…
N + 14	…
N + 15	Timestamp MSB

Figure 19: MotionEngine Output Packet, Format 4

Format Select — Indicates the format of the MotionEngine Output packet

Format Flags — Indicates which types of outputs are placed in the stream. All outputs that are produced for a given type are included in the stream. To disable an output, turn it off using sensor period request messages.
FF0 – raw data. Unprocessed data as it is received form the sensors. The units are ADC units.
FF1 – continuous data. Data produced from sensors that operate continuously. Examples include accelerometer, angular velocity and angular position.
FF2 – event data. Data produced from sensors that detect events. Examples include step detector and tap detector
FF3 – timestamp data. The timestamp for each reported channel is appended as a 4-byte unsigned integer to the end of the channel.
FF4 – reserved
FF5 – reserved
FF6 – reserved
FF7 – reserved

Channels:	A channel consists of a channel ID and a channel sequence number followed by 2, 4, 6, 8 or 12 bytes of sensor data. The data section of a Format 4 packet contains as many channels as will fit. There is no fixed order to the channels. A Format 4 packet is constructed by populating it with sensor samples as sensors are sampled and at the rate sensors are sampled. Once the packet is filled or an event sensor event is added, it is sent. For example, a Format 4 packet may contain an accelerometer channel (8 bytes), a gyroscope channel (8 bytes), a temperature channel (4 bytes), a magnetometer channel (8 bytes), a gyroscope channel (8 bytes), a temperature channel (4 bytes) and an empty channel (4 bytes in this case). This set of channels fills the available 44 bytes. A channel becomes 4 bytes larger if the FF3 (timestamp) flag is set, to make room for the channel timestamp. The timestamp is appended to the end of the channel.
Channel ID	Identifies which channel is being reported. The empty channel is defined to fill all remaining bytes in the packet. In other words, if the empty channel is encountered there is no more data in the packet. The channel IDs are defined in Figure 20.
Channel sequence number	A monotonically increment sequence number for the given channel. Each channel has its own independent sequence number.
X/Y/Z-axis LSB/MSB	The raw sample data for each axis of the sensor.

ID	Type	Axes	Units	Description
0	Reserved		Reserved	Reserved
1	Raw	3	ADC	Raw accelerometer
2	Raw	3+1	ADC	Raw gyroscope
			ADC	Raw gyroscope temperature
3	Raw	3	ADC	Raw magnetometer
4	Raw	1	ADC	Raw temperature
5	Raw	1E	ADC	Raw pressure
6	Raw	1E	ADC	Raw ambient light
7	Raw	1	ADC	Raw humidity
8	Raw	1	ADC	Raw proximity
9	Raw	1	none	Raw fusion tick
10	Continuous	3	m/s^2, Q8	Accelerometer
11	Continuous	3	rad/sec, Q9	Gyroscope
12	Continuous	3+3	rad/sec, Q9	Gyroscope uncalibrated
			rad/sec, Q9	Gyroscope drift estimate
13	Continuous	3	μTesla, Q5	Magnetometer
14	Continuous	3+3	μTesla, Q5	Magnetic field uncalibrated
			μTesla, Q5	Hard iron bias
15	Continuous	4+1	Unit quaternion, Q14	Rotation vector
			radians, Q12	Rotation vector accuracy
16	Continuous	4	Unit quaternion, Q14	Game rotation vector
17	Continuous	4+1	Unit quaternion, Q14	Geomagnetic rotation vector
			radians, Q12	Geomagnetic rotation vector accuracy
18	Continuous	3	m/s^2, Q8	Linear acceleration
19	Continuous	3	m/s^2, Q8	Gravity
20	Continuous	1E	hectoPascal, Q8	Pressure
21	Continuous	1E	lux, Q8	Ambient light
22	Continuous	1	percent, Q8	Relative humidity
23	Continuous	1	cm, Q4	Proximity
24	Continuous	1	°C, Q7	Temperature
25	Event	1	Tap events	Tap detector
26	Event	1	Step event	Step detector
27	Event	1	Enumeration	Activity classification
28	Event	1	Bit field	Power management
29	Event	1E	Step count	Step counter
30	Event	1	Sig. motion event	Significant motion detector
31	Event	1	Shake events	Shake detector
32	Event	1	Flip events	Flip detector
33	Event	1	Pickup events	Pickup detector
34	Continuous	3	m/s^2, Q8	Screen rotation accelerometer
35	Event	1	Stability classification	Stability classifier
36	Continuous	4+1	PAC confidences	Personal activity classifier
37	Event	1	Sleep stage	Sleep detector
38	Event	1	Tilt events	Tilt detector
39	Event	1	In Pocket	Pocket detector
40	Event	1	circle	Circle detector

41-254	Reserved		Reserved	Reserved
255	N/A	N/A	N/A	empty channel

Figure 20: Channel ID Definitions for Format 4

5.1.6 DCE Out V2 (Class 0x27)

This packet conveys 9-axis raw motion sensor data.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x27							
1	Length							
2	Destination							
3	Source							
4	Sequence Number LSB							
5	…							
6	…							
7	Sequence Number MSB							
8	Accelerometer X LSB							
9	Accelerometer X MSB							
10	Accelerometer Y LSB							
11	Accelerometer Y MSB							
12	Accelerometer Z LSB							
13	Accelerometer Z MSB							
14	Rotational Sensor X LSB							
15	Rotational Sensor X MSB							
16	Rotational Sensor Y LSB							
17	Rotational Sensor Y MSB							
18	Rotational Sensor Z LSB							
19	Rotational Sensor Z MSB							
20	Magnetometer X LSB							
21	Magnetometer X MSB							
22	Magnetometer Y LSB							
23	Magnetometer Y MSB							
24	Magnetometer Z LSB							
25	Magnetometer Z MSB							
26	Temperature LSB							
27	Temperature MSB							
28	Reserved	Reserved	Reserved	Reserved	Flag4	Flag3	Flag2	Flag1
29	Button8	Button7	Button6	Button5	Button4	Button3	Button2	Button1
30	Delta Wheel							

Figure 21: DCE Out V2 Packet

Sequence number 32-bit unsigned monotonically increasing number

Suppression Flags	Suppression flags indicate that a user interface event occurred that may cause unintended motion by the user. Motion would be suppressed for a predefined duration according to the flag selected. Flag1 – Set to indicate button press event occurs Flag2 – Set to indicate button release Flag3 – Set to indicate encoder up motion is detected Flag4 – Set to indicate encoder down motion is detected
Button n	Bit mapped field for state information for up to 8 buttons. 0 – not pressed. 1 – pressed.
Other fields	Self-explanatory

5.1.7 DCE Out V3 (Class 0x28)

This packet conveys the 6-axis raw motion sensor data in a smaller packet size of 20 bytes. It is generated at the sampling rate.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x28							
1	Length							
2	Destination							
3	Source							
4	Sequence Number							
5	Flag4	Flag3	Flag2	Flag1	Button4	Button3	Button2	Button1
6	Accelerometer X LSB							
7	Accelerometer X MSB							
8	Accelerometer Y LSB							
9	Accelerometer Y MSB							
10	Accelerometer Z LSB							
11	Accelerometer Z MSB							
12	Rotational Sensor X LSB							
13	Rotational Sensor X MSB							
14	Rotational Sensor Y LSB							
15	Rotational Sensor Y MSB							
16	Rotational Sensor Z LSB							
17	Rotational Sensor Z MSB							
18	Temperature LSB							
19	Temperature MSB							

Figure 22: DCE Out V3 Packet

Sequence number	8-bit unsigned monotonically increasing number
Suppression Flags	Suppression flags indicate that a user interface event occurred that may cause unintended motion by the user. Motion would be suppressed for a predefined duration according to the flag selected.

	Flag1 – Set to indicate button press event occurs Flag2 – Set to indicate button release Flag3 – Set to indicate encoder up motion is detected Flag4 – Set to indicate encoder down motion is detected
Button n	Bit mapped field for state information for up to 4 buttons. 0 – not pressed. 1 – pressed.
Other fields	Self-explanatory

5.1.8 DCE Out V4 (Class 0x29)

The DCE Out V4 packet is a flexible packet that can carry many different payloads. It uses a subclass filed similar to the way the generic in and out messages use a subclass field. The format for a general DCE Out V4 packet is shown in Figure 23.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x29							
1	Length							
2	Destination							
3	Source							
4	Subclass							
5	Sequence Number							
6-19	Payload							

Figure 23 - DCE Out V4 Packet

5.1.8.1 DCE Out V4 - AG (Subclass 0x00)

This packet carries raw accelerometer, gyroscope and temperature sensor sample data.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x29							
1	Length							
2	Destination							
3	Source							
4	Subclass = 0x00							
5	Sequence Number							
6	Accelerometer X LSB							
7	Accelerometer X MSB							
8	Accelerometer Y LSB							
9	Accelerometer Y MSB							
10	Accelerometer Z LSB							
11	Accelerometer Z MSB							
12	Rotational Sensor X LSB							
13	Rotational Sensor X MSB							
14	Rotational Sensor Y LSB							
15	Rotational Sensor Y MSB							
16	Rotational Sensor Z LSB							
17	Rotational Sensor Z MSB							
18	Temperature LSB							
19	Temperature MSB							

Figure 24 - DCE Out v4 Class

Sequence number:	8-bit unsigned monotonically increasing number
Other fields:	Self-explanatory

5.1.8.2 DCE Out V4 – M (Subclass 0x01)

This packet carries raw magnetometer sensor sample data, button states and control flags.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x29							
1	Length							
2	Destination							
3	Source							
4	Subclass = 0x01							
5	Sequence Number							
6	Magnetometer X LSB							
7	Magnetometer X MSB							
8	Magnetometer Y LSB							
9	Magnetometer Y MSB							
10	Magnetometer Z LSB							
11	Magnetometer Z MSB							
12	Reserved	Reserved	Reserved	Reserved	Flag4	Flag3	Flag2	Flag1
13	Button8	Button7	Button6	Button5	Button4	Button3	Button2	Button1
14	Delta Wheel							
15	Reserved							
16	Reserved							
17	Reserved							
18	Reserved							
19	Reserved							

Figure 25: DCE Out V4 Packet

Sequence number	8-bit unsigned monotonically increasing number
Suppression Flags	Suppression flags indicate that a user interface event occurred that may cause unintended motion by the user. Motion would be suppressed for a predefined duration according to the flag selected. Flag1 – Set to indicate button press event occurs Flag2 – Set to indicate button release Flag3 – Set to indicate encoder up motion is detected Flag4 – Set to indicate encoder down motion is detected
Button n	Bit mapped field for state information for up to 8 buttons. 0 – not pressed. 1 – pressed.
Other fields	Self-explanatory

5.2 Outbound Packets

This section covers packets that are sent to a hub from a host.

5.2.1 DataOut Packet (Class 0x07)

This packet is used as a container for command, control, status, testing, and debugging information. Each packet contains a subclass. The subclass determines the type of information carried with the DataOut packet. All subclass values are reserved with the exception of those described in this section

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x07							
1	Length							
2	Destination							
3	Source							
4	Subclass							
5-11	Payload							

Figure 26: DataOut Packet

Subclass: identifies the specific type of information being carried within the DataOut packet
Payload: the remaining report payload, up to 7 bytes

5.2.1.1 FRS Write Request (Subclass 0x06)

This request is sent to the device to initiate a flash record write. A length of 0 will cause the record to be cleared.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x07							
1	Length							
2	Destination							
3	Source							
4	Subclass = 0x06							
5	Reserved							
6	Length LSB							
7	Length MSB							
8	FRS Type LSB							
9	FRS Type MSB							

Figure 27: FRS Write Request Packet

Length: length in 32-bit words of record to be written
FRS Type: FRS record type (see Section 6.1)

5.2.1.2 FRS Write Data (Subclass 0x07)

This packet is sent to the device to write data to the record indicated by a previous write request. Only one FRS operation may be in progress at any one time.

Byte	Bit 7	6	5	4	3	2	1	0
0	Class = 0x07							
1	Length							
2	Destination							
3	Source							
4	Subclass = 0x07							
5	Reserved							
6	Offset LSB							
7	Offset MSB							
8	Data LSB							
9	Data …							
10	Data …							
11	Data MSB							

Figure 28: FRS Write Data Packet

Offset: offset, in 32-bit words, from the beginning of the record indicating where in the record the data is to be written

Data: 32-bit word of data to be written to the FRS record

5.2.1.3 FRS Read Request (Subclass 0x08)

This packet is sent to the device to request a flash record to be sent. The data sent starts from the word offset and continues through to the end of the record or up to the block size requested. The entire record is sent if the requested block size is 0.

Byte	Bit 7	6	5	4	3	2	1	0
0	Class = 0x07							
1	Length							
2	Destination							
3	Source							
4	Subclass = 0x08							
5	Reserved							
6	Read Offset LSB							
7	Read Offset MSB							
8	FRS Type LSB							
9	FRS Type MSB							
10	Block Size LSB							
11	Block Size MSB							

Figure 29: FRS Read Request Packet

Read Offset: offset, in 32-bit words, from the beginning of the FRS records at which to begin the read operation. The first word in an FRS record is word 0.

FRS Type: FRS record type to read (see Section 6.1)

Block Size: number of 32-bit words to read

5.2.1.4 Product ID Request (Subclass 0x09)

This request is sent from the host to a device to request the product ID information.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x07							
1	Length							
2	Destination							
3	Source							
4	Subclass = 0x09							
5	Reserved							

Figure 30: Product ID Request Packet

5.2.1.5 Activity Classification Notification (Subclass 0x12)

The activity classification notification is used by systems that perform algorithm processing on the host. This notification is used to communicate motion activity classification information back to the FSP for power management purposes.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x07							
1	Length							
2	Destination							
3	Source							
4	Subclass = 0x12							
5	Classification							

Figure 31: Activity Classification Notification

Classification:
- 0 – stable
- 1 – on table
- 2 – active
- 3 – watchdog pet. Used by host systems to notify sensor only systems that MotionEngine is still running

5.2.1.6 Freespace Data Mode Control V2 Request (Subclass 0x14)

This packet controls generation of Freespace motion packets.

Byte	Bit 7	6	5	4	3	2	1	0
0	Class = 0x07							
1	Length							
2	Destination							
3	Source							
4	Subclass = 0x14							
5	Reserved			Output Status	Operating Mode Requested			Operating Status
6	Packet Select							
7	Format Select							
8	FF7	FF6	FF5	FF4	FF3	FF2	FF1	FF0
9-11	Reserved							

Figure 32: Freespace Data Mode Control V2 Request Packet

Operating Status:	0 – update the current operating mode as indicated 1 – report the current operating mode without changing it
Operating Mode Requested	Requests a change to the specified operating mode. The hub will enter the requested mode if allowed. 0 – Full Motion 1 – Sleep 2 – Reserved 3 – Deprecated 4 – Full Motion On 5 – Notify-on-Motion
Output Status	0 – update the current packet and format selections as indicated 1 – report the current packet and format selections without changing them
Packet select	Select the output packet sent by the FSP to report motion data. 0 – none. 1 – Mouse 2 – Reserved 3 – Reserved 4 – Reserved 5 – Reserved 6 – Reserved 7 – SuperSDA (reserved) 8 – MotionEngine Output 9 – Mouse and MotionEngine Output. Both mouse and MotionEngine output packets are sent 10-255 – reserved
Format Select	Select the format of the MotionEngine Output packet. See 5.1.5 for details.
FF7 – FF0	Format Flags – used to enable various portions of a MotionEngine Output packet. See 5.1.5 for details.

5.2.1.7 Sensor Period Request (Subclass 0x16)

This packet is used to set or get the sampling period for each sensor or the fusion rate for sensor fusion processing. The period for each sensor must first be set. Once the periods have been set they must be committed. The last packet sent may both set and commit the periods. The sensor periods from previous set operations are retained until the hub is next reset. Therefore, once the periods have all been set, only the ones that need changing need to be set again. In order for a change to take effect it must be committed. Only the sensors that the user wants to operate should have their periods set to a non-zero value. The other periods should be set to 0. For sensor fusion only the fusion period needs to be non-zero. The hub will determine which sensors to operate and at what periods in order to satisfy the superset of requirements imposed by the periods requested for all the sensors.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x07							
1	Length							
2	Destination							
3	Source							
4	Subclass = 0x16							
5	Reserved						Set	Commit
6	Sensor							
7	Reserved							
8	Period LSB							
9	…							
10	…							
11	Period MSB							

Figure 33: Sensor Period Request

Commit	Commit Period change 0 – Don’t commit 1 – Commit change to system
Set	Set/get the sample period. 0 – set the sample period 1 – get the sample period
Sensor	Select which sensor period to set or get. 0 – accelerometer 1 – gyroscope 2 – magnetometer 3 – ambient light sensor 4 – pressure sensor 5 – proximity sensor 6 – sensor fusion 7 – step detector 8 – tap detector 9 – humidity sensor 10 – ambient temperature sensor 11 – significant motion detector

	12 – shake detector 13 – screen rotation accelerometer 14 – flip detector 15 – pickup detector 16 – stability detector 17 – personal activity classifier 18 – sleep detector 19 – gravity 20 – linear acceleration 21 – rotation vector 22 – game rotation vector 23 – geo magnetic rotation vector 24 – step counter 25 – gyroscope uncalibrated 26 – magnetometer uncalibrated 27 – tilt detector 28 – pocket detector 29 – circle detector 30-255 – reserved
Period	Sensor period in microseconds. The actual sample period resolution supported varies from sensor to sensor. A sensor will never be operated at a period it does not support. The sample period returned from a get operation will be the actual sample rate used, regardless of the rate requested.

5.2.1.8 Button Motion Suppression Request (Subclass 0x17)

Button motion suppression (BMS) requests inform the hub that it needs to suppress unintended cursor motion due to a button or scroll wheel user input action. The BMS request byte specifies the type of action to suppress.

The closer in time to the action the packet is received by the hub, the more effectively the hub can suppress motion due to the action. Response times under 8ms are optimal. Response times over 50ms may affect motion not correlated to the user action. This request is also used to enable /disable park. Park is a cursor output suppression mechanism. When park is enabled cursor output is set to 0. Cursor output remains at 0 until park is disabled.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	Class = 0x07							
1	Length							
2	Destination							
3	Source							
4	Subclass = 0x17							
5	BMS Request							
6-11	Reserved							

Figure 34: Button Motion Suppression Request

BMS Request	0 – button press 1 – button release 2 – scroll up 3 – scroll down 4 – park enable 5 – park disable 6 – 255 reserved

6 FSP Flash Record System

The FSP flash record system maintains a variety of records in flash memory. These records are used to store fixed system configuration, factory production configuration, customer product-specific configuration and dynamic information.

Each record may vary in length. The length of each record is a multiple of four bytes. There is at most one valid record of each type stored in flash.

Reserved bits must be set to 0. Undefined or inconsistent configurations result in unexpected behavior.

6.1 Record Types

The defined record types are listed in Figure 35.

Description	FRS Record Type	Scope	Producer
Tracking number record	0x4B4B	Device	Factory
Static calibration data record	0x7979	Device	Factory/Developer
Enable record	0x5627	Device	Factory
MotionEngine orientation record	0x2D3E	Product	Developer
MotionEngine power management record	0xD3E2	Product	Developer
Motion Engine angular position smoothing record	0x3E2D	Product	Developer

Figure 35: FSP Record Types

Each record can be described in terms of its scope and its producer. The scope indicates whether the record contains information that is specific to a device, hardware or product. Device scope is information that varies per individual device. Hardware scope is information that varies with each hardware implementation of a product. Product scope is information that varies with each overall implementation of a product. A given hardware implementation can support multiple overall implementations.

The producer indicates who produces the information to populate a record. “System” indicates that the firmware itself produces the data. “Factory” indicates that the factory production process produces the data. “Developer” indicates that an engineering product developer produces the data. Even though a developer produces the data, the record may be programmed onto a device during the factory production process.

6.2 Device Records

The tracking number and network ID records vary for each device.

6.2.1 Tracking Number

The tracking number record stores a 32-bit number used to identify an individual device. The format of a tracking number record is shown in Figure 36.

Word	Description
0	32-bit tracking number

Figure 36: Tracking Number Record

6.2.2 Static Calibration Data

Static calibration data is produced by the manufacturing process for per device calibration or by the design process for nominal calibration. The data format is proprietary to Hillcrest Labs.

6.2.3 Enable

The enable record stores product enable information. The data format is proprietary to Hillcrest Labs.

6.2.4 Product Records

6.2.4.1 MotionEngine Orientation

The MotionEngine Orientation record controls the rotation of MotionEngine output coordinate system relative to the sensor set's coordinate system. The rotation vector is a signed, 2's-complement fixed point number with a Q point of 30. The format of a MotionEngine Orientation record is shown in Figure 37 If the rotation vector is set to all 0's then no rotation will be applied. The default values are 0 for all for values.

Word	Description
0	Rotation quaternion X
1	Rotation quaternion Y
2	Rotation quaternion Z
3	Rotation quaternion W

Figure 37: MotionEngine Orientation Record

6.2.4.2 MotionEngine Power Management

The power management record controls the thresholds MotionEngine uses to determine when the device is at rest and when it is in motion. The format of a MotionEngine Power Management record is shown in Figure 38.

	Description			
Word	Byte 3	Byte 2	Byte 1	Byte 0
0	Delta orientation			
1	Stable threshold			
2	Gyro power save	Delta acceleration	Stable duration	

Figure 38: MotionEngine Power Management Record

Delta orientation

Once a device has been determined to be not moving based on the generation of ON_TABLE or IS_STABLE, delta orientation is the amount of change in device orientation required to recognize that the device is in motion. The units are radians. The delta orientation value is a signed, 2's-complement fixed point number with a Q point of 29. The default value is 0.09.

Stable threshold

The gyro output must be below this limit for the stable duration in order for an IS_STABLE notification to be

	generated. The units are radians per second. The stable threshold value is a signed, 2's-complement fixed point number with a Q point of 25. The default value is 0.1.
Stable duration	The amount of time in seconds that motion must be below the stable threshold before an IS_STABLE notification is generated. The default value is 10.
Delta acceleration	When using wake-on-motion, this is the amount of acceleration required for the accelerometer to determine that motion has occurred. The units are mg. The delta acceleration value is unsigned. The default value is 2.
Gyro power save	Determines the gyroscope power management scheme used by MotionEngine. Gyroscope power usage can be reduced by turning the gryoscope off when the hub is not moving and thus the gyroscope is not needed. Allowing the gyroscope to be turned off to save power results in a delay of several tens of milliseconds after motion starts until updated gyroscope outputs are available. The actual delay depends on the startup time of the particular gyroscope being used. The default setting is none. 0 – none 1 – when on table 2 – when stable 3 – when on table or stable 4 – when on table and stable 5-255 – reserved

6.2.4.3 MotionEngine Angular Position Smoothing

During large, fast motions the angular position output may sometimes become misaligned with the actual angular position. When the device slows or stops, angular position can be determined accurately and the angular position output updated accordingly. However, step updates to the output are undesirable in some applications. Angular position smoothing addresses this issue by correcting angular position errors only when the device is moving.

The angular position smoothing record controls the thresholds MotionEngine uses to angular position smoothing. The format of an Angular Position Smoothing record is shown in Figure 39. The angular position smoothing parameters are all unsigned, fixed point numbers. If this record is not configured, the hub uses the default values listed below.

Word	Description
0	Scaling
1	Max rotation
2	Max error
3	Stability magnitude

Figure 39: Angular Position Smoothing Record

Scaling	Scaling controls what fraction of the angular velocity can be used to correct angular position errors. The range for this parameter is 0 to 1.0. This parameter is dimensionless. The Q point is 30. The default scale is 0.2.
Max rotation	Max rotation is the maximum amount of angular correction that can be used to correct angular position errors. The settings for scaling and max rotation determine how aggressively angular errors are corrected. The range is 0 to 2*PI. The units are radians. The Q point is 29. The default value is 7.5 degrees or 0.131 radians.
Max error	Max error is the maximum angular error allowed to accumulate before the angular position output is updated in a single step. The units are radians. The range is 0 to 2*PI. The Q point is 29. The default value is 15 degrees or 0.262 radians.
Stability magnitude	Stability magnitude controls the amount of change in angular position that must occur before the angular position output is updated with a new value. The units are radians. The Q point is 29. The default value is 0.1 degrees or 0.00175 radians.

7 References

1. USB Device Class Definition for Human Interface Devices (HID) Version 1.11: http://www.usb.org/developers/devclass_docs/HID1_11.pdf.
2. USB HID Usage Tables Version 1.12: http://www.usb.org/developers/devclass_docs/Hut1_12.pdf.
3. 1000-1024 - Calibration Test Station Manual, Hillcrest Labs. (under NDA only)
4. http://libfreespace.hillcrestlabs.com.
5. 1000-3041 – Developing a Freespace Remote Control, Hillcrest Labs.

8 Legal Statements

www.ceva-dsp.com

FOR MORE
INFORMATION: