

# RW BLE HID Over GATT Profile Interface Specification

---

Interface Specification

RW-BLE-HOGP-IS

Version 9.0

2017-03-09

---



## Revision History

Version	Date	Revision Description	Author
1.0	2012-10-24	Initial release	LT
2.0	2012-12-03	Client Multi-Profile API	LT
2.1	2013-04-05	Updated API	LT
7.00	2014-11-28	Updated for BLE 4.1	FBE
8.00	2015-07-29	Updated for BLE 4.2	CM
8.01	2015-08-14	HOGPD_ENABLE_REQ API change	CM
9.0	2017-03-09	Updated for BLE 5	LT



## Table of Contents

1	Overview.....	5
1.1	Document Overview .....	5
1.2	General HID Protocol Overview.....	5
1.2.1	Boot Reports .....	5
1.2.1.1	Boot Keyboard Input Report .....	6
1.2.1.2	Boot Keyboard Output Report .....	6
1.2.1.3	Boot Mouse Input Report .....	7
1.3	Bluetooth Low Energy HID Over GATT Profile Overview .....	7
2	HOGP Device Role API .....	8
2.1	Environment.....	8
2.2	Initialization / Database Creation.....	8
2.3	API Messages.....	9
2.3.1	Connection Messages .....	9
2.3.1.1	HOGPD_ENABLE_REQ .....	9
2.3.1.2	HOGPD_ENABLE_RSP .....	9
2.3.2	Communication Messages .....	9
2.3.2.1	HOGPD_NTF_CFG_IND.....	9
2.3.2.2	HOGPD_PROTO_MODE_REQ_IND .....	10
2.3.2.3	HOGPD_PROTO_MODE_CFM.....	10
2.3.2.4	HOGPD_CTNL_PT_IND .....	10
2.3.2.5	HOGPD_REPORT_UPD_REQ .....	11
2.3.2.6	HOGPD_REPORT_UPD_RSP .....	11
2.3.2.7	HOGPD_REPORT_REQ_IND.....	11
2.3.2.8	HOGPD_REPORT_CFM.....	12
3	HOGP Report Host Role API .....	13
3.1	Environment.....	13
3.2	Initialization .....	13
3.3	API Messages.....	14
3.3.1	Connection Messages .....	14
3.3.1.1	HOGPRH_ENABLE_REQ.....	14
3.3.1.2	HOGPRH_ENABLE_RSP.....	14
3.3.2	Communication Messages .....	15
3.3.2.1	HOGPRH_READ_INFO_REQ .....	15
3.3.2.2	HOGPRH_READ_INFO_RSP .....	15



---

3.3.2.3	HOGPRH_WRITE_REQ .....	15
3.3.2.4	HOGPRH_WRITE_RSP .....	16
3.3.2.5	HOGPRH_REPORT_IND .....	16
4	HOGP Boot Host Role API .....	18
4.1	Environment .....	18
4.2	Initialization .....	18
4.3	API Messages .....	19
4.3.1	Connection Messages .....	19
4.3.1.1	HOGPBH_ENABLE_REQ .....	19
4.3.1.2	HOGPBH_ENABLE_RSP .....	20
4.3.2	Communication Messages .....	21
4.3.2.1	HOGPBH_READ_INFO_REQ .....	21
4.3.2.2	HOGPBH_READ_INFO_RSP .....	21
4.3.2.3	HOGPBH_WRITE_REQ .....	21
4.3.2.4	HOGPBH_WRITE_RSP .....	22
4.3.2.5	HOGPBH_BOOT_REPORT_IND .....	22
5	Miscellaneous .....	23
6	Abbreviations .....	28
7	References .....	29

## 1 Overview

### 1.1 Document Overview

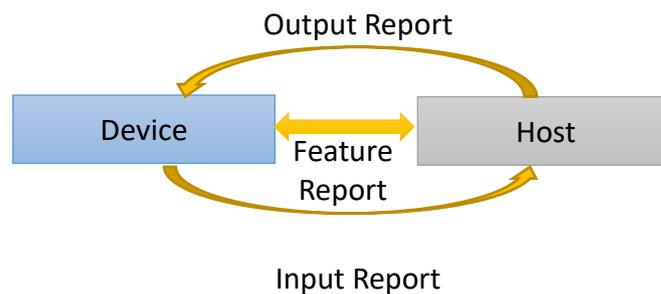
This document describes the non-standard interface of the RivieraWaves (RW) Bluetooth Low Energy (BLE) HID OVER GATT Profile (HOGP) implementation. Along this document, the interface messages will be referred to as API messages for the profile block(s).

Their description will include their utility and reason for implementation for a better understanding of the user and the developer that may one day need to interface them from a higher application.

### 1.2 General HID Protocol Overview

Two entities are defined in the Human Interface Device (HID) protocol: the "host" and the "device".

The device is the entity that directly interacts with a human, such as a keyboard or mouse. The host communicates with the device and receives input reports from the device on actions performed by the human. Output reports flow from the host to the device and then to the human. The most common example of a host is a PC or a tablet. A feature report specifies configuration information for a device.



The HID protocol makes implementation of devices very simple. Devices define their data packets and then present a "HID descriptor" (known as Report Map in the BLE HOGP specification) to the host. The Report Map is a hard coded array of bytes that describe the device's data packets.

This includes: how many packets the device supports, how large are the packets, and the purpose of each byte and bit in the packet.

The above mechanism describes what is known as HID "report protocol". Because it was understood that not all hosts would be capable of parsing HID descriptors, HID also defines "boot protocol". In boot protocol, only specific devices are supported with only specific features because fixed data packet formats are used. The Report Map is not used in this mode so innovation is limited. However, the benefit is that minimal functionality is still possible on hosts that otherwise would be unable to support HID. The only devices supported in boot protocol are:

- Keyboard: Any of the first 256 key codes ("Usages") defined in the HID Usage Tables will be available.
- Mouse: Only the X-axis, Y-axis, and the first 3 buttons will be available.

More information about the HID protocol and the way to use a HID Device can be found in [5] and [6].

#### 1.2.1 Boot Reports

The Device Class Definition for Human Interface Devices (HID) document ([6], p59, Appendix B: Boot Interface Descriptors) describes the content of the Boot Reports packets.



“The HID Subclass 1 defines two descriptors for Boot Devices. Devices may append additional data to these boot reports, **but the first 8 bytes of keyboard reports and the first 3 bytes of mouse reports must conform to the format defined by the Boot Report descriptor in order for the data to be correctly interpreted by the BIOS. The report may not exceed 8 bytes in length.** The BIOS will ignore any extensions to reports. These descriptors describe reports that the BIOS expects to see.”

Some implementation point are based on these paragraph.

- The length of a Boot Keyboard Input Report is equal to 8.
- The length of a Boot Mouse Input Report must be in a range between 3 and 8.

### 1.2.1.1 Boot Keyboard Input Report

This section represents the content of a Boot Keyboard Input Report value as defined in [6].

Byte	Description
0	Modifier keys
1	Reserved
2	Keycode 1
3	Keycode 2
4	Keycode 3
5	Keycode 4
6	Keycode 5
7	Keycode 6

### 1.2.1.2 Boot Keyboard Output Report

This section represents the content of a Boot Keyboard Output Report value as defined in [6].

Bit	Description
0	NUM LOCK
1	CAPS LOCK
2	SCROLL LOCK
3	COMPOSE
4	KANA
5 to 7	CONSTANT



### 1.2.1.3 Boot Mouse Input Report

This section represents the content of a Boot Mouse Input Report value as defined in [6].

Byte	Description
0	Bit 0 : Button 1
	Bit 1 : Button 2
	Bit 2 : Button 3
	Bit 4 to 7 : Device specific
1	X displacement
2	Y displacement
3 to n (max(n) = 8)	Device specific (Optional)

## 1.3 Bluetooth Low Energy HID Over GATT Profile Overview

The BLE HOGP enables the user to exchange HID reports between a device and a host (boot or report).

Within the profile, three roles can be supported: Device role, Boot Host role and the Report Host role. Both of the Hosts must support the GAP Central Role and the Device, the GAP Peripheral role. The profile requires a connection to be established between the two devices for its functionality.

The various documents edited by the Bluetooth SIG present different use cases for this profile, their GATT, GAP and security, mandatory and optional requirements. The HOGP and the HID Service (HIDS) specifications have been adopted by the Bluetooth SIG on December 27th 2011 ([1] and [3]). Their related Test Specifications have been released at the same time and are referenced in [2] and [4].

The profile is implemented in the RW-BLE software stack as three tasks, one for each role. Each task has an API decided after the study of the profile specifications and test specifications, and it is considered to be minimalistic and designed for a future application which would combine the profile functionality with the device connectivity and security procedures.



## 2 HOGP Device Role API

### 2.1 Environment

A HID device may have more than one instance of the HID Service and in each of these instances it may have several instances of the Report Characteristic. As we currently have a static implementation, some firmware limitations have been defined for this role (in the hogpd.h file):

- The maximal number of HID Service instances is 2 (HOGPD\_NB\_HIDS\_INST\_MAX).
- The maximal number of Report Characteristics that can be present in a HID Service is 5 (HOGPD\_NB\_REPORT\_INST\_MAX).
- The maximal length of a Report Characteristic value has been set to 45 bytes.

As specified in [3], the maximal length for a Report Map Characteristic value is 512 bytes.

**Important Note** : In addition to the HID Service, a HID Device database shall contains one or more instances of the Battery Service (BAS), one and only one instance of the Device Information Service (DIS). It could also contains one and only one instance of the Scan Parameters Service (SCPS). The application shall include these services in the database using the BAS, DIS and SCPP APIs.

The Device Information Service shall include the PnP ID characteristic.

When several HID service instances are required by application, the HOGPD task allocates those services on a contiguous handle range into the attribute database.

HID over GATT service task is a mono-instantiated task; it means that connection index is **not** present into task index.

### 2.2 Initialization / Database Creation

During the initialization phase of the device, to use the HID over GATT Service task, the HOGPD task has to be allocated and corresponding attribute database initialized, using GAPM API. Application has to send GAPM\_PROFILE\_TASK\_ADD\_CMD [8] with specific device required security level and following parameters.

**Parameters:**

Type	Parameters	Description
uint8_t	hids_nb	Number of HID Service instances to add in the database.
struct hogpd_hids_cfg	cfg[HOGPD_NB_HIDS_INST_MAX]	Configuration for each HID Service you want to add (see Table 5-1).

Multiples service instances of the HID Service should allow implementers to define HID Devices whose combined functions require more than 512 octets of data to describe. Thus, the second instance of the HID Service shall exist only if the Report Map Characteristic value exceeds 512 bytes.

**Note about the External Report Reference feature:** The External Report Reference descriptor is an optional descriptor linked to the Report Map characteristic. It allows a HID Host to map information from the Report Map characteristic value to the characteristic UUID of external service characteristics used to transfer the associated data.

**Note about the Report type and the Report ID values:** For each Report Characteristic, the Report Type value has to be defined in the report\_char\_cfg parameter (see Table 5-3). The Report ID should match the report type; this value is provided by the application.



## 2.3 API Messages

### 2.3.1 Connection Messages

#### 2.3.1.1 HOGPD\_ENABLE\_REQ

Source: TASK\_APP

Destination: TASK\_HOGPD

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection Index.
uint16_t	ntf_cfg[HOGPD_NB_HIDS_INST_MAX]	Saved notification configurations (see Table 5-7).

Response: HOGPD\_ENABLE\_RSP

Description: This API message can be used after the connection with a peer device has been established in order to restore BOND data of a known device.

#### 2.3.1.2 HOGPD\_ENABLE\_RSP

Source: TASK\_HOGPD

Destination: TASK\_APP

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection Index.
uint8_t	status	Status code (see [7])

Description: Inform application if restoring bond data for peer device succeed or not.

### 2.3.2 Communication Messages

#### 2.3.2.1 HOGPD\_NTF\_CFG\_IND

Source: TASK\_HOGPD

Destination: TASK\_APP

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection Index.
uint16_t	ntf_cfg[HOGPD_NB_HIDS_INST_MAX]	Saved notification configurations (see Table 5-7).

Description: This API message is sent to the application each time the host enables or disables sending of notifications for a characteristic.



### 2.3.2.2 HOGPD\_PROTO\_MODE\_REQ\_IND

Source: TASK\_HOGPD

Destination: TASK\_APP

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection Index.
uint8_t	operation	Operation requested (see Table 5-8)
uint8_t	Hids_idx	HID Service instance. Between 0 and (HOGPD_NB_HIDS_INST_MAX-1)
uint8_t	proto_mode	New Protocol Mode Characteristic value: HOGP_BOOT_PROTOCOL_MODE or HOGP_REPORT_PROTOCOL_MODE.

Response: HOGPD\_PROTO\_MODE\_CFM

Description: This API message is used by the HID Device role to inform the application that a Protocol Mode characteristic value has been written by a peer device.

If we consider the HID Device is operating as a Keyboard and that the Host has set the Protocol Mode Characteristic value to Boot Protocol Mode, Reports shall be sent using the Boot Keyboard Input Report Characteristic instead of the classic Report Characteristic.

The application shall confirm or not modification of the protocol mode with HOGPD\_PROTO\_MODE\_CFM message.

### 2.3.2.3 HOGPD\_PROTO\_MODE\_CFM

Source: TASK\_APP

Destination: TASK\_HOGPD

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection Index.
uint8_t	status	Status code of the request (see [7])
uint8_t	Hids_idx	HID Service instance. Between 0 and (HOGPD_NB_HIDS_INST_MAX-1)
uint8_t	proto_mode	New Protocol Mode Characteristic value: HOGP_BOOT_PROTOCOL_MODE or HOGP_REPORT_PROTOCOL_MODE.

Description: This API message is used to confirm or not modification of the protocol mode.

### 2.3.2.4 HOGPD\_CTLN\_PT\_IND

Source: TASK\_HOGPD

Destination: TASK\_APP

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection Index.
uint8_t	hids_idx	HID Service instance. Between 0 and (HOGPD_NB_HIDS_INST_MAX-1)
uint8_t	hid_ctln_pt	New HID Control Point Characteristic value:



		HOGP_CTLN_PT_SUSPEND or HOGP_CTLN_PT_EXIT_SUSPEND.
--	--	---

Description: This API message is used by the HID Device role to inform the application that a HID Control Point Characteristic value has been written by a peer device.

### 2.3.2.5 HOGPD\_REPORT\_UPD\_REQ

Source: TASK\_APP

Destination: TASK\_HOGPD

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection Index.
struct hogpd_report_info	report	Report Info (see Table 5-9)

Response: HOGPD\_NTF\_SENT\_CFM

Description: This API message is used to update the value of the Report Characteristic stored in the database and to notify the Host about this new value if sending of notifications has been enabled for it.

### 2.3.2.6 HOGPD\_REPORT\_UPD\_RSP

Source: TASK\_HOGPD

Destination: TASK\_APP

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection Index.
uint8_t	status	Status code (see [7])

Description: This message is sent to the application after reception of the HOGPD\_REPORT\_UPD\_REQ to inform it if a notification has been sent to the Host or if an error has been raised.

### 2.3.2.7 HOGPD\_REPORT\_REQ\_IND

Source: TASK\_HOGPD

Destination: TASK\_APP

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection Index.
uint8_t	operation	Operation requested (see Table 5-8)
struct hogpd_report_info	report	Report Info (see Table 5-9)

Response: HOGPD\_REPORT\_CFM

Description: This message is sent to the application after the peer Host has written the value of one of the Report Characteristics or if peer device request information of a report value.



### 2.3.2.8 HOGPD\_REPORT\_CFM

Source: TASK\_HOGPD

Destination: TASK\_APP

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection Index.
uint8_t	operation	Operation requested (see Table 5-8)
uint8_t	status	Status code (see [7])
struct hogpd_report_info	report	Report Info (see Table 5-9)

Description: This message is sent by the application to confirm read or modification of a report.



### 3 HOGP Report Host Role API

#### 3.1 Environment

The BLE HOGP Report Host role has been designed to allow a collector to communicate with a HID device. An application which would use this role shall support a HID Parser and be able to handle arbitrary format for data transfers.

The table below shown Report Host characteristic requirements:

Characteristic	Requirements
Report Map	Mandatory
Report	Mandatory
Boot Keyboard Input Report	Excluded
Boot Keyboard Output Report	Excluded
Boot Mouse Input Report	Excluded
HID Information	Mandatory
HID Control Point	C.1
Protocol Mode	Optional

C.1: Mandatory if the Host supports Suspend Mode, otherwise optional.

Thus, the HOGP Report Host role task will only look for the non-excluded characteristics during the discovery process.

Some restrictions have been defined in the BLE HOGP specification and shall be respected by the application designer:

- A Report Host shall not concurrently be a Boot Host.
- The Report Host shall use the GAP Central role.

As we currently have a static implementation, some firmware limitations have been defined for this role (in the hogprh.h file):

- The maximal number of HID Service instances that can be handled has been limited to 2 (HOGPBH\_NB\_HIDS\_INST\_MAX).
- The maximal number of Report Characteristics instances that can be handled has been limited to 5 (HOGPRH\_NB\_REPORT\_INST\_MAX).
- The maximal length of a Report Characteristic value has been limited to 45 bytes.

Within the HOGPRH task, three state are defined: FREE, IDLE, BUSY.

**Important Note:** The TASK\_HOPRH task is multi-instantiated, one instance is created for each connection for which the profile will be enabled and each of these instances will have a different task ID. To communicate with the peer device, the corresponding connection index has to be used to calculate the HOGPRH task instance.

The term TASK\_HOGPRH\_IDX will be used in the rest of the document to refer to any instance of the HID Over GATT Profile Report Host Role Task. The term TASK\_HOGPRH will refer to the first instance of this task.

#### 3.2 Initialization

During the initialization phase of the device, to use the HID over GATT Report Host task, the HOGPRH task has to be allocated using GAPM API. Application has to send GAPM\_PROFILE\_TASK\_ADD\_CMD [8].



### 3.3 API Messages

#### 3.3.1 Connection Messages

##### 3.3.1.1 HOGPRH\_ENABLE\_REQ

Source: TASK\_APP

Destination: TASK\_HOGPRH\_IDX

Parameters:

Type	Parameters	Description
uint8_t	con_type	Connection type
uint8_t	hids_nb	Number of instances of the HID Service that have been found during the last discovery.
struct hogprh_hids_content	hids[HOGPRH_NB_HIDS_INST_MAX]	Information about the HID Service instances that have been found during the last discovery. (see Table 5-18)

Response: HOGPRH\_ENABLE\_RSP

Description: This API message is used for enabling the Report Host role of the HOGP. This Application message contains BLE connection handle, the connection type and the previously saved discovered HIDS details on peer.

The connection type may be PRF\_CON\_DISCOVERY (0x00) for discovery/initial configuration or PRF\_CON\_NORMAL (0x01) for a normal connection with a bonded device. Application shall save those information to reuse them for other connections. During normal connection, previously discovered device information can be reused.

If it is a discovery /configuration type of connection, it is useless to fill the HIDS parameters (hids\_nb and hids) are useless. Otherwise they will contain pertinent data which will be kept in the Boot Host environment while enabled.

For a normal connection, the response to this request is sent right away after saving the HIDS content in the environment and registering HOGPRH in GATT to receive the notifications for the known attribute handles in HIDS that would be notified (Report Characteristic). For a discovery connection, discovery of the peer HIDS is started and the response will be sent at the end of the discovery with the discovered attribute details.

The Task for this profile role will go from IDLE state to CONNECTED state for a normal connection, and to DISCOVERING state for a discovery/configuration type of connection.

##### 3.3.1.2 HOGPRH\_ENABLE\_RSP

Source: TASK\_HOGPRH\_IDX

Destination: TASK\_APP

Parameters:

Type	Parameters	Description
uint8_t	status	Discovery status code (see [7])
uint8_t	hids_nb	Number of instances of the HID Service that have been found during the discovery.
struct hogprh_hids_content	hids[HOGPRH_NB_HIDS_INST_MAX]	Information about the HID Service instances that have been found during the discovery. (see Table 5-18)

Description: This API message is used by the Report Host to either send the discovery results of HIDS on the HID device and confirm enabling of the Report Host role, or to simply confirm enabling of Report Host role if it is a normal connection and the attribute details are already known.



### 3.3.2 Communication Messages

#### 3.3.2.1 HOGPRH\_READ\_INFO\_REQ

Source: TASK\_APP

Destination: TASK\_HOGPRH\_IDX

Parameters:

Type	Parameters	Description
uint8_t	info	Characteristic info (see Table 5-11)
uint8_t	hid_idx	HID Service Instance - From 0 to HOGPRH_NB_HIDS_INST_MAX-1
uint8_t	report_idx	HID Report Index: only relevant for: - info = HOGPRH_REPORT - info = HOGPRH_REPORT_REF - info = HOGPRH_REPORT_NTF_CFG

Response: HOGPRH\_READ\_INFO\_RSP

Description: This API message shall be used to read the value of a characteristic or a descriptor in the HID Device database.

#### 3.3.2.2 HOGPRH\_READ\_INFO\_RSP

Source: TASK\_HOGPRH\_IDX

Destination: TASK\_APP

Parameters:

Type	Parameters	Description
uint8_t	status	Status code (see [7])
uint8_t	info	Characteristic info (see Table 5-11)
uint8_t	hid_idx	HID Service Instance - From 0 to HOGPRH_NB_HIDS_INST_MAX-1
uint8_t	report_idx	HID Report Index: only relevant for: - info = HOGPRH_REPORT - info = HOGPRH_REPORT_REF - info = HOGPRH_REPORT_NTF_CFG
union hogprh_data	data	Information data (see Table 5-12)

Description: This response contains value of requested data or an error code if nothing has been found.

#### 3.3.2.3 HOGPRH\_WRITE\_REQ

Source: TASK\_APP

Destination: TASK\_HOGPRH\_IDX

Parameters:

Type	Parameters	Description
uint8_t	info	Characteristic info (see Table 5-11)



uint8_t	hid_idx	HID Service Instance - From 0 to HOGPRH_NB_HIDS_INST_MAX-1
uint8_t	report_idx	HID Report Index: only relevant for: - info = HOGPRH_REPORT - info = HOGPRH_REPORT_NTF_CFG
bool	wr_cmd	Write type ( Write without Response True or Write Request) - only valid for HOGPRH_REPORT
union hogprh_data	data	Information data (see Table 5-12)

Response: HOGPRH\_WRITE\_RSP

Description: This API message shall be used to:

- Enable or disable the notifications for a Report Characteristic instance.
- Write the HID Control Point Characteristic value in the peer device database.
- Set the protocol mode of a HID Service instance to the Report Protocol Mode.
- Write the value of a Report Characteristic in the peer device database

### 3.3.2.4 HOGPRH\_WRITE\_RSP

Source: TASK\_HOGPRH\_IDX

Destination: TASK\_APP

Parameters:

Type	Parameters	Description
uint8_t	status	Status code (see [7])
uint8_t	info	Characteristic info (see Table 5-11)
uint8_t	hid_idx	HID Service Instance - From 0 to HOGPRH_NB_HIDS_INST_MAX-1
uint8_t	report_idx	HID Report Index: only relevant for: - info = HOGPRH_REPORT - info = HOGPRH_REPORT_NTF_CFG

Description: The API message is used to inform the application about the write request status.

### 3.3.2.5 HOGPRH\_REPORT\_IND

Source: TASK\_HOGPRH\_IDX

Destination: TASK\_APP

Parameters:

Type	Parameters	Description
uint8_t	hid_idx	HID Service Instance - From 0 to HOGPRH_NB_HIDS_INST_MAX-1
uint8_t	report_idx	HID Report Index
struct hogprh_report	report	Report data (see Table 5-13)

Description: The API message is used to inform the application about a modification of report on peer database.



Note: Here is an extract of the BLE HIDS specification,

*“Notification of characteristic values can contain at most [ATT\_MTU-3] bytes of data by definition. Data beyond [ATT\_MTU- 3] bytes long is not included in a notification, and must instead be read using the GATT Read Long Characteristic Value sub-procedure. The possibility that data to be notified in a Report characteristic value could change before the HID Host completed an outstanding Read Long Characteristic Value sub-procedure, and therefore be lost, exists. For this reason it is strongly recommended that HID Devices support an ATT\_MTU large enough to transfer their largest possible Report characteristic value in a single transaction.”*

Thus when a notification is received from peer device and suspect that MTU length exceed, the profile internally read the full attribute value before triggering the report message indication.



## 4 HOGP Boot Host Role API

### 4.1 Environment

The BLE HOGP Boot Host role has been designed to allow a collector to easily communicate with a HID Boot device (Boot Keyboard or Boot Mouse). All data exchanged within this role have a fixed length and each bit in a packet has a known meaning. There is no need of a Report descriptor, so no HID Parser is required in the application.

The table below shown Boot Host characteristic requirements:

Characteristic	Requirements
Report Map	Excluded
Report	Excluded
Boot Keyboard Input Report	C.2/C.3
Boot Keyboard Output Report	C.2/C.3
Boot Mouse Input Report	C.2
HID Information	Excluded
HID Control Point	Excluded
Protocol Mode	Mandatory
C.2: Mandatory to support at least one of these features.	
C.3: If one of these feature is supported, both features shall be supported.	

Thus, the HOGP Boot Host role task will only look for the non-excluded characteristics during the discovery process (more details in HOGPBH\_ENABLE\_REQ and HOGPBH\_ENABLE\_).

Some restrictions have been defined in BLE HOGP specification and shall be respected by the application designer:

- A Boot Host shall not concurrently be a Report Host.
- The Boot Host shall use the GAP Central role.

As we currently have a static implementation, some firmware limitations have been defined for this role (in the hogpbh.h file):

- The maximal number of HID Service instances that can be handled has been limited to 2 (HOGPBH\_NB\_HIDS\_INST\_MAX).

Within the HOGPBH task, three state are defined: IDLE, DISCOVERING, CONNECTED.

**Important Note:** The TASK\_HOGPBH task is multi-instantiated, one instance is created for each connection for which the profile will be enabled and each of these instances will have a different task ID. To communicate with the peer device, the corresponding connection index has to be used to calculate the HOGRH task instance.

The term TASK\_HOGPBH\_IDX will be used in the rest of the document to refer to any instance of the HID Over GATT Profile Boot Host Role Task. The term TASK\_HOGPBH will refer to the first instance of this task.

### 4.2 Initialization

During the initialization phase of the device, to use the HID over GATT Boot Host task, the HOGPBH task has to be allocated using GAPM API. Application has to send GAPM\_PROFILE\_TASK\_ADD\_CMD [8].



## 4.3 API Messages

This part presents all messages that can be sent between an higher application and the HOGP Boot Host role task.

### 4.3.1 Connection Messages

#### 4.3.1.1 HOGPBH\_ENABLE\_REQ

Source: TASK\_APP

Destination: TASK\_HOGPBH\_IDX

Required state: ALL

Parameters:

Type	Parameters	Description
uint8_t	con_type	Connection Type.
uint8_t	hids_nb	Number of instances of the HID Service that have been found during the last discovery.
struct hids_content	hids[HOGPBH_NB_HIDS_INST_MAX]	Information about HID Services that have been found during the last discovery. (see Table 5-17)

Response: HOGPBH\_ENABLE\_RSP

Description: This API message is used for enabling the Boot Host role of the HOGP. This Application message contains BLE connection handle, the connection type and the previously saved discovered HIDS details on peer.

The connection type may be PRF\_CON\_DISCOVERY (0x00) for discovery/initial configuration or PRF\_CON\_NORMAL (0x01) for a normal connection with a bonded device. Application shall save those information to reuse them for other connections. During normal connection, previously discovered device information can be reused.

If it is a discovery /configuration type of connection, it is useless to fill the HIDS parameters (hids\_nb and hids) are useless. Otherwise they will contain pertinent data which will be kept in the Boot Host environment while enabled.

For a normal connection, the response to this request is sent right away after saving the HIDS content in the environment and registering HOGPBH in GATT to receive the notifications for the known attribute handles in HIDS that would be notified (Boot Keyboard Input Report and Boot Mouse Input Report). For a discovery connection, discovery of the peer HIDS is started and the response will be sent at the end of the discovery with the discovered attribute details.

The Task for this profile role will go from IDLE state to CONNECTED state for a normal connection, and to DISCOVERING state for a discovery/configuration type of connection.



#### 4.3.1.2 HOGPBH\_ENABLE\_RSP

Source: **TASK\_HOGPBH\_IDX**

Destination: TASK\_APP

Parameters:

Type	Parameters	Description
uint8_t	status	Discovery status code (see [7])
uint8_t	hids_nb	Number of instances of the HID Service that have been found during the discovery.
struct hids_content	hids[HOGPBH_NB_HIDS_INST_MAX]	Information about HID Services that have been found during the discovery. (see Table 5-17)

Description: This API message is used by the Boot Host to either send the discovery results of HIDS on the HID device and confirm enabling of the Boot Host role, or to simply confirm enabling of Boot Host role if it is a normal connection and the attribute details are already known.



### 4.3.2 Communication Messages

#### 4.3.2.1 HOGPBH\_READ\_INFO\_REQ

Source: TASK\_APP

Destination: TASK\_HOGPBH\_IDX

Required state: CONNECTED

Parameters:

Type	Parameters	Description
uint8_t	info	Characteristic info (see Table 5-26)
uint8_t	hid_idx	HID Service Instance - From 0 to HOGPBH_NB_HIDS_INST_MAX-1

Response: HOGPBH\_READ\_INFO\_RSP

Description: This API message shall be used to read the value of a characteristic or a descriptor in the HID Device database.

#### 4.3.2.2 HOGPBH\_READ\_INFO\_RSP

Source: TASK\_HOGPBH\_IDX

Destination: TASK\_APP

Parameters:

Type	Parameters	Description
uint8_t	status	Status code (see [7])
uint8_t	info	Characteristic info (see Table 5-26)
uint8_t	hid_idx	HID Service Instance - From 0 to HOGPBH_NB_HIDS_INST_MAX-1
union hogpbh_data	data	Information data (see Table 5-28)

Description: The API message is used to inform the application about the read Client Characteristic Configuration Descriptor value.

#### 4.3.2.3 HOGPBH\_WRITE\_REQ

Source: TASK\_APP

Destination: TASK\_HOGPBH\_IDX

Required state: CONNECTED

Parameters:

Type	Parameters	Description
uint8_t	info	Characteristic info (see Table 5-26)
uint8_t	hid_idx	HID Service Instance - From 0 to HOGPBH_NB_HIDS_INST_MAX-1
bool	wr_cmd	Write type ( Write without Response True or Write Request) - only valid for HOGPBH_BOOT_KB_OUT_REPORT
union hogpbh_data	data	Information data (see Table 5-28)



Response: HOGPBH\_WRITE\_RSP

Description: This API message shall be used to:

- Enable or disable the notifications for the Boot Keyboard Input Characteristic
- Enable or disable the notifications for the Boot Mouse Input Characteristic.
- Write the value of one of the Boot Report Characteristics in the peer device database.

#### 4.3.2.4 HOGPBH\_WRITE\_RSP

Source: TASK\_HOGPBH\_IDX

Destination: TASK\_APP

Parameters:

Type	Parameters	Description
uint8_t	status	Status code (see [7])
uint8_t	info	Characteristic info (see Table 5-26)
uint8_t	hid_idx	HID Service Instance - From 0 to HOGPBH_NB_HIDS_INST_MAX-1

Description: The API message is used to inform the application about status of the write request

#### 4.3.2.5 HOGPBH\_BOOT\_REPORT\_IND

Source: TASK\_HOGPBH\_IDX

Destination: TASK\_APP

Parameters:

Type	Parameters	Description
uint8_t	info	Characteristic info (see <b>TODO</b> ): - HOGPBH_BOOT_KB_IN_REPORT - HOGPBH_BOOT_MOUSE_IN_REPORT
uint8_t	hid_idx	HID Service Instance - From 0 to HOGPBH_NB_HIDS_INST_MAX-1
struct hogpbh_boot_report	report	Boot Report Data (see Table 5-27)

Description: The API message is used to inform the application about a new value has been received within a read response message or a notification for one of the Boot Report Characteristics present in the peer device database.



## 5 Miscellaneous

Type	Parameters	Description
uint8_t	svc_features	Features supported in the HID Service (see Table 5-2).
uint8_t	report_nb	Value of the HID Information Characteristic.
uint8_t	report_char_cfg[HOGPD_NB_REPORT_INST_MAX]	Features supported by each of the Report Characteristics in the HID Service (see Table 5-3).
uint8_t	report_id[HOGPD_NB_REPORT_INST_MAX]	Report id number for a given report type
struct hids_hid_info	hid_info	Value of the HID Information Characteristic (see Table 5-4).
struct hogpd_ext_ref	ext_ref	External Report Reference (see Table 5-6)
uint16_t	ext_rep_ref_uuid	External Report Reference value.

Table 5-1 - HIDS Configuration structure (struct hogpd\_hids\_cfg)

Bit	Flag	Description
0	HOGPD_CFG_KEYBOARD	The HID device is operating as a keyboard. Add the Boot Keyboard Input Report Characteristic. Add the Boot Keyboard Output Report Characteristic.
1	HOGPD_CFG_MOUSE	The HID device is operating as a mouse. Add the Boot Mouse Input Report Characteristic.
2	HOGPD_CFG_PROTO_MODE	The HID Device supports the Boot Protocol Mode. Add the Protocol Mode Characteristic.
3	HOGPD_CFG_MAP_EXT_REF	The Report Map Characteristic value maps information to an external service characteristic. Add an External Report Reference descriptor in the Report Map Characteristic. Add an Included Service in the HID Service.
4	HOGPD_CFG_BOOT_KB_WR	The Boot Keyboard Input Report Characteristic value is writable.
5	HOGPD_CFG_BOOT_MOUSE_WR	The Boot Mouse Input Report Characteristic value is writable.
6		
7		

Table 5-2 – svc\_features Parameter Bit Flags

Bit	Flag	Description
0	HOGPD_CFG_REPORT_IN	The Report is an Input Report.
1	HOGPD_CFG_REPORT_OUT	The Report is an Output Report.
2	HOGPD_CFG_REPORT_FEAT	The Report is a Feature Report.
3	HOGPD_CFG_REPORT_WR	The Report Characteristic value is writable. Taken in account only if the Report is an Input Report.
4 to 7	-	Reserved.

Table 5-3– report\_char\_cfg Parameter Bit Flags

Type	Parameters	Description
uint16_t	bcdHID	HID Class Specification release number in binary-coded decimal (for example, 1.50 is 0x150).
uint8_t	bCountryCode	Hardware target country.
uint8_t	flags	Flags (see Table 5-5).



**Table 5-4 – HID Information structure (struct hids\_hid\_info)**

Bit	Flag	Description
0	HIDS_REMOTE_WAKE_CAPABLE	Inform if the HID Device is capable of providing wake-up signal to a HID host.
1	HIDS_NORM_CONNECTABLE	Inform if the HID Device is normally connectable.
2 to 7	-	Reserved for future use.

**Table 5-5 – HID Information value flags Parameter Bit Flags**

Type	Parameters	Description
uint16_t	inc_svc_hdl	External Report Reference - Included Service
uint16_t	rep_ref_uuid	External Report Reference - Characteristic UUID

**Table 5-6 – External Report Reference (struct hogpd\_ext\_ref)**

Flag	Value	Description
HOGPD_CFG_KEYBOARD	0x01	Keyboard Device NTF
HOGPD_CFG_MOUSE	0x02	Boot Mouse Input Report Characteristic Notification Configuration.
HOGPD_CFG_REPORT_NTF_EN	0x40 << report_idx	Report Notification Enabled (to be shift for each report index)

**Table 5-7 – HIDS Notification Configuration bit field**

Flag	Value	Description
HOGPD_OP_NO	0x01	No operation
HOGPD_OP_REPORT_READ	0x02	Read report value
HOGPD_OP_REPORT_WRITE	0x03	Modify/Set report value
HOGPD_OP_PROT_UPDATE	0x04	Modify Protocol mode

**Table 5-8 – Type of operation requested by peer device**

Type	Parameters	Description
uint8_t	hid_idx	HIDS Instance
uint8_t	type	type of report (see Table 5-10)
uint16_t	length	Report Length
uint8_t	idx	Report Instance - 0 for boot reports and report map
uint8_t	value[length]	Report data

**Table 5-9 - HID Report Info (struct hogpd\_report\_info)**

Flag	Value	Description
HOGPD_REPORT	0	The Report characteristic is used to exchange data between a HID Device and a HID Host.
HOGPD_REPORT_MAP	1	The Report Map characteristic
HOGPD_BOOT_KEYBOARD_INPUT_REPORT	2	Boot Keyboard Input Report
HOGPD_BOOT_KEYBOARD_OUTPUT_REPORT	3	Boot Keyboard Output Report
HOGPD_BOOT_MOUSE_INPUT_REPORT	4	Boot Mouse Input Report

**Table 5-10 - Type of reports (enum hogpd\_report\_type)**

Flag	Description
------	-------------



HOGPRH_PROTO_MODE	Protocol Mode
HOGPRH_REPORT_MAP	Report Map
HOGPRH_REPORT_MAP_EXT_REP_REF	Report Map Char. External Report Reference Descriptor
HOGPRH_HID_INFO	HID Information
HOGPRH_HID_CTLN_PT	HID Control Point
HOGPRH_REPORT	Report
HOGPRH_REPORT_REF	Report Char. Report Reference
HOGPRH_REPORT_NTF_CFG	Report Notification config

Table 5-11 Peer HID service info that can be read/write (enum hogprh\_info)

Type	Parameters	Description
uint8_t	proto_mode	Protocol Mode - info = HOGPRH_PROTO_MODE
struct hids_hid_info	hid_info	HID Information value (see Table 5-4) - info = HOGPRH_HID_INFO
uint8_t	hid_ctln_pt	HID Control Point value to write - info = HOGPRH_HID_CTLN_PT
struct hogprh_report	report	Report information (see Table 5-13) - info = HOGPRH_REPORT
uint16_t	report_cfg	Notification Configuration Value - info = HOGPRH_REPORT_NTF_CFG
struct hogprh_report_ref	report_ref	HID report Reference (see Table 5-16) - info = HOGPRH_REPORT_REF
struct hogprh_report_map	report_map	HID report MAP info (see Table 5-14) - info = HOGPRH_REPORT_MAP
struct hogprh_report_map_ref	report_map_ref	HID report MAP reference (see Table 5-15) - info = HOGPRH_REPORT_MAP_EXT_REP_REF

Table 5-12 - Information data Union used to read or write information (union hogprh\_data)

Type	Parameters	Description
uint16_t	length	Report Length
uint8_t	value[length]	Report value

Table 5-13 - HID report info (struct hogprh\_report)

Type	Parameters	Description
uint16_t	length	Report MAP Length
uint8_t	value[length]	Report MAP value

Table 5-14 - HID report MAP info (struct hogprh\_report\_map)

Type	Parameters	Description
uint8_t	uuid_length	Reference UUID length
uint8_t	uuid [uuid_length]	Reference UUID Value

Table 5-15 - HID report MAP reference (struct hogprh\_report\_map\_ref)

Type	Parameters	Description
uint16_t	id	Report ID.
uint16_t	type	Report Type.

Table 5-16 – Report Reference Descriptor value structure (struct hogprh\_report\_ref)

Type	Parameters	Description
struct prf_svc	svc	Start handle and End handle of the HID Service instance (see Table 5-19).



struct prf_char_inf	chars[HOGPBH_CHAR_MAX]	Characteristic information (see Table 5-20). See also Table 5-24 for indices.
struct prf_char_desc_inf	descs[HOGPBH_DESC_MAX]	Descriptors information (see Table 5-21). See also Table 5-25 for indices.
uint8_t	report_nb	Number of instances of the Report Characteristic that have been found.

Table 5-17 – HIDS content structure for a Boot Host (struct hogpbh\_hids\_content)

Type	Parameters	Description
struct prf_svc	svc	Start handle and End handle of the HID Service instance (see Table 5-19).
struct prf_char_inf	chars[HOGPRH_CHAR_MAX]	Characteristic information (see Table 5-20). See also Table 5-22 for indices.
struct prf_char_desc_inf	descs[HOGPRH_DESC_MAX]	Descriptors information (see Table 5-21). See also Table 5-23 for indices.

Table 5-18 – HIDS content structure for a Report Host (struct hogprh\_hids\_content)

Type	Parameters	Description
uint16_t	shdl	Start handle of the HID Service.
uint16_t	ehdl	End handle of the HID Service.

Table 5-19 – Service description structure (struct prf\_svc)

Type	Parameters	Description
uint16_t	char_hdl	Characteristic declaration attribute handle.
uint16_t	val_hdl	Characteristic value attribute handle.
uint8_t	prop	Properties
uint8_t	char_ehdl_off	Number of attribute within the Characteristic.

Table 5-20 – Characteristic description structure (struct prf\_char\_inf)

Type	Parameters	Description
uint16_t	desc_hdl	Descriptor attribute handle

Table 5-21 – Descriptor description structure (struct prf\_char\_desc\_inf)

Flag	Description
HOGPRH_CHAR_REPORT_MAP	Report Map
HOGPRH_CHAR_HID_INFO	HID Information
HOGPRH_CHAR_HID_CTLN_PT	HID Control Point
HOGPRH_CHAR_PROTOCOL_MODE	Protocol Mode
HOGPRH_CHAR_REPORT+ [0:report_nb-1]	Report

Table 5-22 - HID Report Host Characteristics list (enum hogprh\_chars)

Flag	Description
HOGPRH_DESC_REPORT_MAP_EXT_REP_REF	Report Map Char. External Report Reference Descriptor
HOGPRH_DESC_REPORT_REF + [0:report_nb-1]	Report Char. Report Reference
HOGPRH_DESC_REPORT_CFG + [0:report_nb-1]	Report Client Config

Table 5-23 - HID Report Host Characteristic descriptors list (enum hogprh\_descs)

Flag	Description
------	-------------



HOGPBH_CHAR_PROTO_MODE	Protocol Mode Characteristic.
HOGPBH_CHAR_BOOT_KB_IN_REPORT	Boot Keyboard Input Report Characteristic.
HOGPBH_CHAR_BOOT_KB_OUT_REPORT	Boot Keyboard Output Report Characteristic.
HOGPBH_CHAR_BOOT_MOUSE_IN_REPORT	Boot Mouse Input Report Characteristic.

**Table 5-24 - Boot Host Role Characteristic Code Flags**

Flag	Description
HOGPBH_DESC_BOOT_KB_IN_REPORT_CFG	Boot Keyboard Input Report Characteristic Client Characteristic Configuration Descriptor.
HOGPBH_DESC_BOOT_MOUSE_IN_REPORT_CFG	Boot Mouse Input Report Characteristic Client Characteristic Configuration Descriptor.

**Table 5-25 - Boot Host Role Descriptors Code Flags**

Flag	Description
HOGPBH_PROTO_MODE	Protocol Mode
HOGPBH_BOOT_KB_IN_REPORT	Boot Keyboard Input Report
HOGPBH_BOOT_KB_OUT_REPORT	Boot Keyboard Output Report
HOGPBH_BOOT_MOUSE_IN_REPORT	Boot Mouse Input Report
HOGPBH_BOOT_KB_IN_NTF_CFG	Boot Keyboard Input Report Client Config
HOGPBH_BOOT_MOUSE_IN_NTF_CFG	Boot Mouse Input Report Client Config

**Table 5-26 - Peer HID service info that can be read/write (enum hogpbh\_info)**

Type	Parameters	Description
uint16_t	length	Boot Report Length
uint8_t	value[length]	Boot Report value

**Table 5-27 - HID Boot report info (struct hogpbh\_boot\_report)**

Type	Parameters	Description
uint8_t	proto_mode	Protocol Mode - info = HOGPBH_PROTO_MODE
uint16_t	ntf_cfg	Notification Configuration Value - info = HOGPBH_BOOT_KB_IN_NTF_CFG - info = HOGPBH_BOOT_MOUSE_IN_NTF_CFG
struct hogpbh_boot_report	report	Boot report information - info = HOGPBH_BOOT_KB_IN_REPORT - info = HOGPBH_BOOT_KB_OUT_REPORT - info = HOGPBH_BOOT_MOUSE_IN_REPORT

**Table 5-28 - Information data (union hogpbh\_data)**



## 6 Abbreviations

Abbreviation	Original Terminology
API	Application Programming Interface
BAS	Battery Service
BLE	Bluetooth Low Energy
DIS	Device Information Service
GAP	Generic Access Profile
GATT	Generic Attribute Profile
HID	Human Interface Device
HIDS	HID Service
HOGP	HID Over GATT Profile
HOGPBH	HID Over GATT Profile Boot Host
HOGPD	HID Over GATT Profile Device
HOGPRH	HID Over GATT Profile Report Host
RW	RivieraWaves
SCPS	Scan Parameters Service



## 7 References

<b>[1]</b>	<b>Title</b>	HID OVER GATT PROFILE SPECIFICATION		
	<b>Reference</b>	HOGP_SPEC_V10		
	<b>Version</b>	V10r00	<b>Date</b>	2011-12-27
	<b>Source</b>	Bluetooth SIG		

<b>[2]</b>	<b>Title</b>	HID OVER GATT PROFILE TEST SPECIFICATION		
	<b>Reference</b>	HOGP.TS.1.0.0		
	<b>Version</b>	1.0.0	<b>Date</b>	2011-12-27
	<b>Source</b>	Bluetooth SIG		

<b>[3]</b>	<b>Title</b>	HID SERVICE SPECIFICATION		
	<b>Reference</b>	HIDS_SPEC_V10		
	<b>Version</b>	V10r00	<b>Date</b>	2011-12-27
	<b>Source</b>	Bluetooth SIG		

<b>[4]</b>	<b>Title</b>	HID SERVICE TEST SPECIFICATION		
	<b>Reference</b>	HIDS.TS.1.0.0		
	<b>Version</b>	1.0.0	<b>Date</b>	2011-12-27
	<b>Source</b>	Bluetooth SIG		

<b>[5]</b>	<b>Title</b>	HID USAGE TABLE		
	<b>Reference</b>	HUT1_12v2		
	<b>Version</b>	1.12	<b>Date</b>	2004-10-28
	<b>Source</b>	USB Implementers' Forum		

<b>[6]</b>	<b>Title</b>	DEVICE CLASS DEFINITION FOR HUMAN INTERFACE DEVICES (HID)		
	<b>Reference</b>	HID1_11		
	<b>Version</b>	1.11	<b>Date</b>	2001-06-27
	<b>Source</b>	USB Implementers' Forum		

<b>[7]</b>	<b>Title</b>	RW BLE Host Error Code Interface Specification		
	<b>Reference</b>	RW-BLE-HOST-ERR-CODE-IS		
	<b>Version</b>	9.0	<b>Date</b>	2017-03-09
	<b>Source</b>	RivieraWaves SAS		

<b>[8]</b>	<b>Title</b>	GAP Interface Specification		
	<b>Reference</b>	RW-BLE-GAP-IS		
	<b>Version</b>	9.0	<b>Date</b>	2017-03-09
	<b>Source</b>	RivieraWaves SAS		