



CSR Synergy Bluetooth 18.0.2

HIDH Human Interface Device Host Profile

Demo Description

September 2010



Cambridge Silicon Radio Limited

Churchill House
Cambridge Business Park
Cowley Road
Cambridge CB4 0WZ
United Kingdom

Registered in England and Wales 3665875

Tel: +44 (0)1223 692000

Fax: +44 (0)1223 692001

www.csr.com



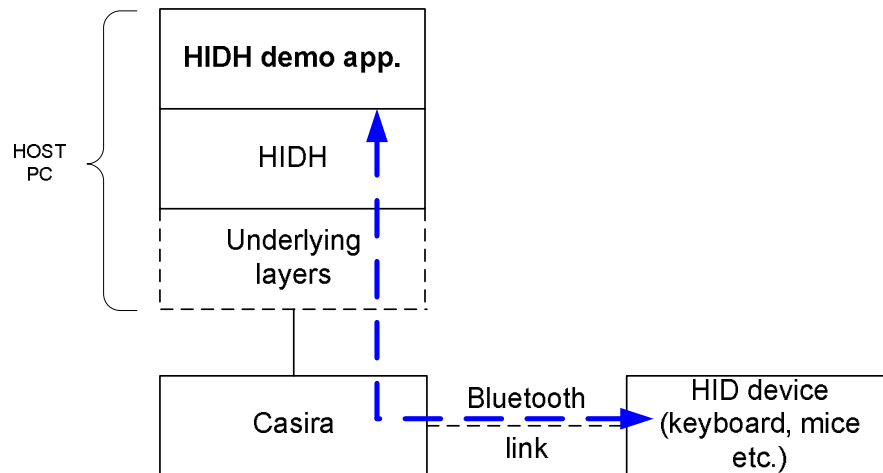
Contents

1	Human Interface Device Host Profile (HIDH)	3
1.1	Generally	3
1.2	Use of Program hci_hidh_demo_app.exe	3
2	Linux	7
	Terms and Definitions	8
	Document History	8
	TradeMarks, Patents and Licences	9
	Life Support Policy and Use in Safety-critical Compliance	9
	Performance and Conformance	9

1 Human Interface Device Host Profile (HIDH)

1.1 Generally

The HIDH demo application can be used for establishing a HID connection with a HID device, e.g. keyboards or mice. The purpose of the demo application is to show that the HID host profile (HIDH) developed by CSR is compliant with other HID devices and the specification proposed by the Bluetooth SIG. Furthermore, the demo application can be considered as an example of how to utilize the HIDH API.



The figure above shows the physical setup of the demo application scenario. The HID device is connected via a Casira Bluetooth development kit to a host PC running the host stack and the HIDH demo application.

The HIDH demo application may be connected to the Casira using either a serial connection using BCSP (`hci_hidh_demo_app.exe`), a serial connection using H4DS (`hci_hidh_demo_app_h4ds.exe`), or an USB connection (`hci_hidh_demo_app_usb.exe`)

The application has been made to run on Windows and Linux.

On Windows the HIDH demo application and the underlying layers are all running in the same process while on Linux it is possible to:

- 1) have the HIDH demo application running together with the underlying layers in one process.
- 2) have the HIDH demo application running in user space and the rest in kernel space.

The HIDH signalling consists of two “groups” of signals, namely:

- 1) Control signalling for doing discovery, connection, disconnection etc.
- 2) User signalling, which contains mouse movement, keyboard presses etc.

Both of these are in the Windows version all handled in the HIDH demo application while in the Linux version it is possible to have the user signals handled by an “external” module, so that when the HIDH profile is running in kernel space it is possible to connect to the Linux Input sub-system.

The next section describes how to use the HIDH demo application on windows but everything also holds for the Linux version of the HIDH demo application.

1.2 Use of Program `hci_hidh_demo_app.exe`

NOTE: This description is for CSR Synergy Bluetooth HCI using BCSP over UART as host transport. The functionality of the application is independent from the used host; the only difference is the naming of the

application. Note, that HIDH cannot be used with RFCOMM-builds due to limitations in handling the large SDP records used by typical HID devices.

Start the program hci_hidh_demo_app.exe:

The `hci_hidh_demo_app.exe` program must be executed using certain execution parameters. This can be done using e.g. a 'command prompt'. The following parameters must be specified:

- B <baudrate> specifies the baud rate for the COM port connected to the Casira. If no parameter is given, the default value of 115200 baud will be used.
- C <COM port> specifies the COM port number connected to the Casira. If no parameter is specified, com1 will be used as default value. **(On Linux use /dev/ttyS0 etc)**
- A < BD addr.> specifies the Bluetooth address of the remote device used for the default connection. This parameter is optional.

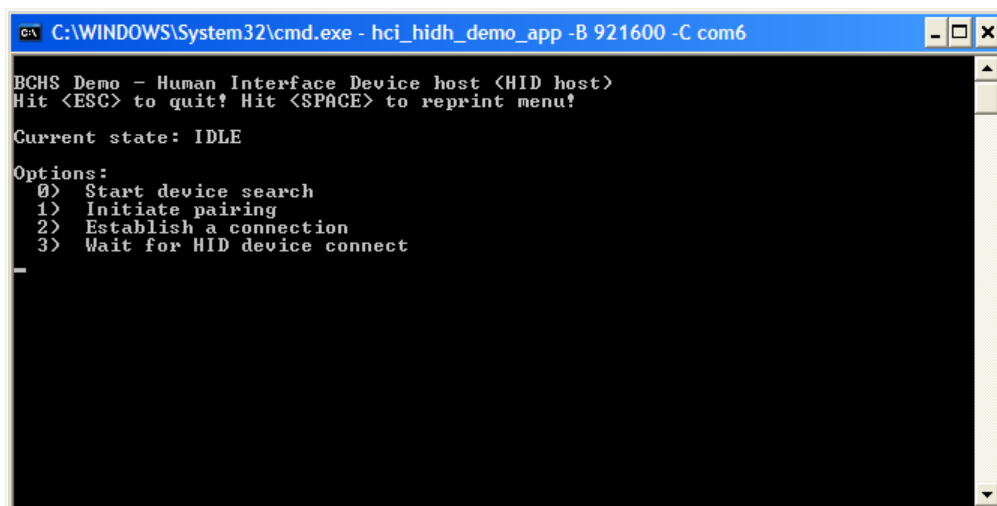
An example of the program execution using com6 with baud rate 921600 is given below:

```
hci_hidh_demo_app.exe -B 921600 -C com6
```

NOTE: If USB is used for host transport, the application (`hci_hidh_demo_app_usb.exe`) should be started without any parameters, except the optional '-A' parameter.

Using the HIDH demo application:

Once the application is started the following screen appears:



A valid use of the demo application follows the following procedure:

1. Make a search to find a HID device. This is done by choosing menu option '0'. The devices within range appear in a list – cancel the search by pressing 'C' and select the device by entering its entry number in the list. If search is not performed the default device address will be used. The default device address can be specified using the '-A' parameter during application start.
2. The host can initiate pairing with the device selected by pressing '1'. Pairing is optional and can be omitted.
3. A connection is established with the selected device by pressing '2'. Once the connection is established the menu will change and a new list of options is given:

```

C:\WINDOWS\System32\cmd.exe - hci_hidh_demo_app -B 921600 -C com6
BCHS Demo - Human Interface Device host <HID host>
Hit <ESC> to quit! Hit <SPACE> to reprint menu!

Current state: CONNECTED

Options:
0> Unplug
1> Disconnect
2> Suspend
3> Soft device reset
4> Get Report
5> Set Report
6> Get Protocol
7> Set Protocol
8> Get Idle
9> Set Idle
x> Send output report

Please insert passkey for 0007:61:1915D4: 0000
Report(1): ff 00
Report(1): ff 00
Report(1): ff 00
Report(1): ff 00
Bond indication received from device: 0007:61:1915D4
Connection encrypted

```

If a passkey is requested, enter a code and repeat the code on the device. A passkey is only required if the HID device has 'keyboard-functionality' and the devices have not previously bonded. Entering the passkey will enable encryption of the Bluetooth connection.

4. If the host previously has had a connection with a device (a known device), it is possible to let the host wait for the known HID device to connect by pressing '3'. A HID device is known once a connection to the device has successfully been established and will remain so until the host and device unplugs. If there are any known devices, the user is requested to select the device.

When the connection is established, the user has the following options which are described briefly:

Key	Description
'0'	Unplug. Initiates the procedure of releasing the connection and removing the stored knowledge of the device in the host.
'1'	Disconnect. Initiates the release of the connection with the device.
'2'	Suspend. Makes the host send a message to the device that the host will be in-active which allows the device to make more effective power saving.
'3'	Soft reset. Makes the host send a message to the device that it should make a software reset.
'4'	Get report. Retrieves the first input report of the device and print the report on the screen.
'5'	Set report. Sends the first input report filled with zeroes to the device.
'6'	Get protocol. This will retrieve the current HID protocol from the device.
'7'	Set protocol. This will set HID protocol to use by the device. It toggles among report and boot protocol.
'8'	Get idle. Retrieves the idle rate from the device.
'9'	Set idle. Sets the idle rate of the device. It toggles between infinite (0) and 1 second.
'x'	Send output report. Sends the first device output report (if any) filled with zeroes to the device.

```

C:\WINDOWS\System32\cmd.exe - hci_hidh_demo_app -B 921600 -C com6
BCHS Demo - Human Interface Device host <HID host>
Hit <ESC> to quit! Hit <SPACE> to reprint menu!

Current state: CONNECTED

Options:
0> Unplug
1> Disconnect
2> Suspend
3> Soft device reset
4> Get Report
5> Set Report
6> Get Protocol
7> Set Protocol
8> Get Idle
9> Set Idle
x> Send output report
Get report...

Report(1): 01 00 00 00 00 00 00 00 00
Get protocol...

Get protocol returned: Report
Get idle...

Get idle returned: 0
Report(1): 01 00 00 28 00 00 00 00 00
Report(1): 01 00 00 00 00 00 00 00 00
Report(1): 03 e9 00 00 00
Report(1): 03 00 00 00 00
Report(1): 04 01
Report(1): 04 00
Report(1): 03 00 00 00 00
Sending output report...

Set protocol...

Received HANDSHAKE, result code: 0x00
Report(1): 01 00 00 06 00 00 00 00 00
Report(1): 01 00 00 00 00 00 00 00 00
Report(1): 01 00 00 16 00 00 00 00 00
Report(1): 01 00 00 00 00 00 00 00 00
Report(1): 01 00 00 15 00 00 00 00 00
Report(1): 01 00 00 00 00 00 00 00 00

```

The reports arriving from the device are printed on the screen. The screen can be cleared and the menu reprinted by pressing '<space>'.

2 Linux

This section describes how to build and run the HIDH demo application on Linux.

The basic HIDH demo application (located in `./applications/hid_host`) may be compiled for Linux by means of:

```
> make clean all TARGET_ARCH=Linux-2.6-x86
```

This will output three files: `hci_hidh_demo_app`, `hci_hidh_demo_app_h4ds`, and `hci_hidh_demo_app_usb`, for serial BCSP, H4DS, and USB communication respectively.

Terms and Definitions

BlueCore®	Group term for CSR's range of Bluetooth wireless technology chips
Bluetooth®	Set of technologies providing audio and data transfer over short-range radio connections
CSR	Cambridge Silicon Radio
UniFi™	Group term for CSR's range of chips designed to meet IEEE 802.11 standards

Document History

Revision	Date	History
1	13 APR 10	Ready for release 18.0.0
2	31 AUG 10	Ready for release 18.0.2

TradeMarks, Patents and Licences

Unless otherwise stated, words and logos marked with TM or [®] are trademarks registered or owned by CSR plc or its affiliates. Bluetooth[®] and the Bluetooth logos are trademarks owned by Bluetooth SIG, Inc. and licensed to CSR. Other products, services and names used in this document may have been trademarked by their respective owners.

The publication of this information does not imply that any licence is granted under any patent or other rights owned by CSR plc.

CSR reserves the right to make technical changes to its products as part of its development programme.

While every care has been taken to ensure the accuracy of the contents of this document, CSR cannot accept responsibility for any errors.

No statements or representations in this document are to be construed as advertising, marketing, or offering for sale in the United States imported covered products subject to the Cease and Desist Order issued by the U.S. International Trade Commission in its Investigation No. 337-TA-602. Such products include SiRFstarIIITM chips that operate with SiRF software that supports SiRFInstantFixTM, and/or SiRFLoc[®] servers, or contains SyncFreeNav functionality.

Life Support Policy and Use in Safety-critical Compliance

CSR's products are not authorised for use in life-support or safety-critical applications. Use in such applications is done at the sole discretion of the customer. CSR will not warrant the use of its devices in such applications.

Performance and Conformance

Refer to www.csrsupport.com for compliance and conformance to standards information.