PF1080 Aurora

PTT Framework
Design Specification

Author(s):

Henrik Eriksen

Revision History

Version	Date (newest first)	Author	Brief Description of Change
0.1	09-Oct-2018	HERI	Initial creation

Review

Version	Date	Review group	Comments

Approval

Version	Date	Approved by	Comments

Document Data

Attribute	Value
Total pages	70
File name	PTT Framework.docm
Last saved	18-Oct-2018 08:56:00 by Henrik Eriksen (HERI) (Consultant)
Created	09-Oct-2018 10:33:00
Template	ICC generic doc template.dot-2 {ICC doc template.dot-19}

Table of contents

1 Introduction

The purpose of this document is to provide a basic understanding of the PTT Framework for developers writing Tests for the Framework. It is assumed that the reader is familiar with NWTSIM and knows how to use NWTSIM to simulate Zephyr code execution on a standard Linux PC.

The PTT Framework is made for conformance testing of BLE functionality in the Zephyr Bluetooth implementation. The PTT Framework is using the simulator NWTSIM to execute the Zephyr Bluetooth stack in a simulated environment, where the Zephyr code is running on a standard Linux PC rather than a dedicated embedded system.

This setup provides the developer with a powerful environment, where tracing and debugging becomes a lot easier compared to the dedicated embedded system.

Test execution is decoupled from the execution environment running the simulated Zephyr Bluetooth stack, in the sense that tests are running in their own process communicating with the simulated Zephyr Bluetooth stack through Named Pipes.

Tests relies on a PTT application running in the simulated Zephyr Bluetooth execution environment. This PTT application is able to communicate with the Host Layer in the Bluetooth stack through standard HCI commands and HCI events. A simple command API provides tests access to HCI commands and HCI events thru Named Pipes connecting the two execution environments.

2 Framework

The Framework consists of a test execution part and a simulated Zephyr Bluetooth stack execution part as shown in figure 1. The two execution environments are connected thru Named Pipes. Since Bluetooth is all about communicating entities, a test needs access to more than a single DUT to exercise conformance tests. Rather than having the test execution environment communicate directly with each DUT, a multiplexer is introduced between the test execution environment and the Zephyr Bluetooth stack execution environment. The multiplexer, here referred to as the PTT BRIDGE, distributes byte streams between the test execution environment and the different DUTs. In principle any number of DUTs can be supported, but in reality only two DUTs are used for conformance testing. The setup is illustrated in figure 2.

In NWTSIM, the PTT BRIDGE is just another application being executed in parallel with the applications running in the individual DUTs. Each DUT is configured to run the PTT APP, implementing the HCI Command executor. The PTT BRIDGE application must know the number of DUTs to support and the identity of each DUT. This information is passed as run-time parameters to the PTT BRIDGE application. A typical execution of NWTSIM with the PTT BRIDGE and two DUTs could look as shown below.

```
./nwtsim_2G4_scheduler_v1 -s=Test -D=3 -sim_length=5e6 &
./nwtsim_device_PTT_bridge -s=Test -d=0 -D=2 -dev0=1 -dev1=2 -v=3
                           -RxWait=2.5e3 -AutoTerminate &
./nwtsim_nrf52_ptt_app -s=Test -d=1 -v=3 &
./nwtsim_nrf52_ptt_app -s=Test -d=2 -v=3
```

The simulator is configured with the name “Test” (-s=Test). It is told to host three applications (-D=3). The simulation length is set to 5 seconds (-sim_length=5e6).

Transport is conceptually a two-way FIFO for each DUT.

open(device_id), connect(device_id), close(device_id)
send(device_id, message), recv(device_id, count, timeout)
wait(time), get_time()

Commands: Atomic transactions that maps to calls on the DUT side. Cannot be interleaved (for simplicity).

Utils: Set of commands shared between tests.

start(), stop()
read(u8_t pbuffer, size_t count, int flags)
write(u8_t pbuffer, size_t count, int flags)

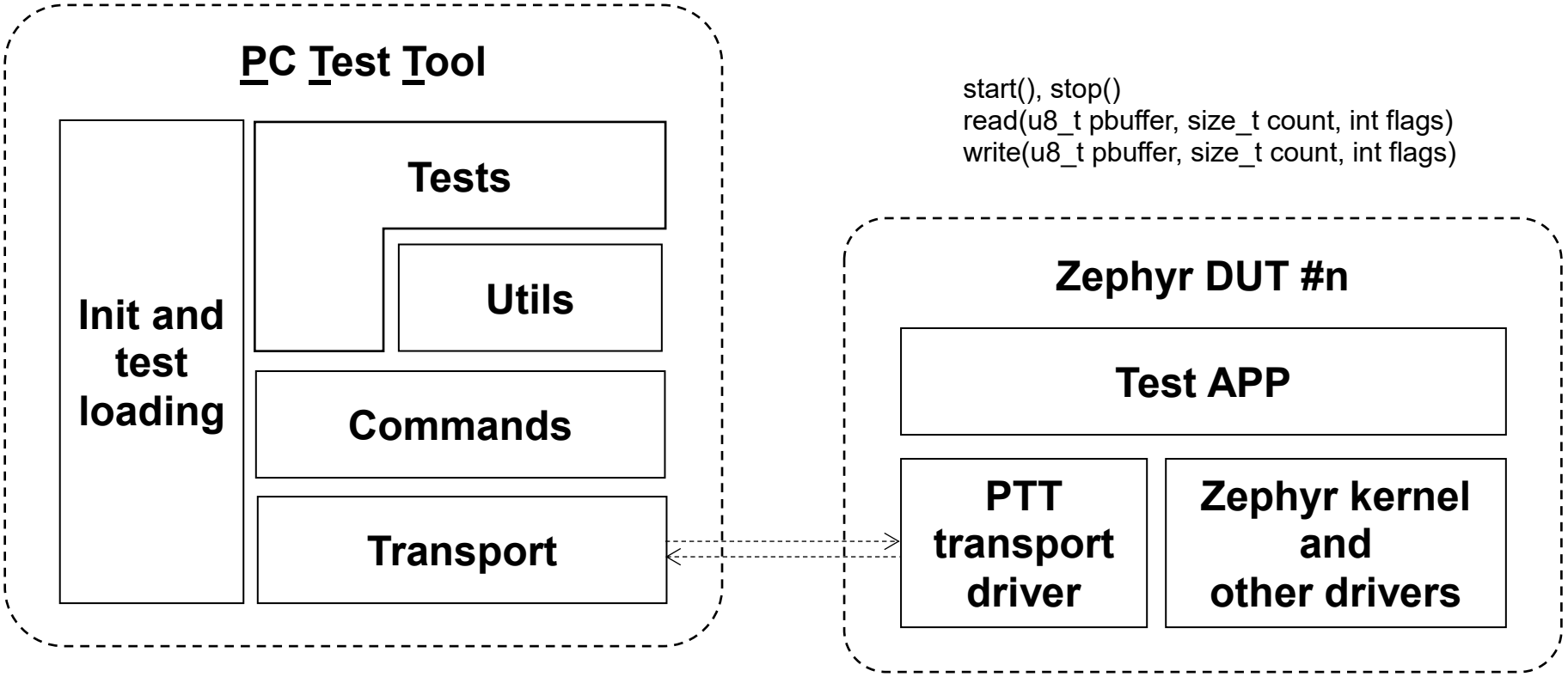

Figure 1 - PTT Framework

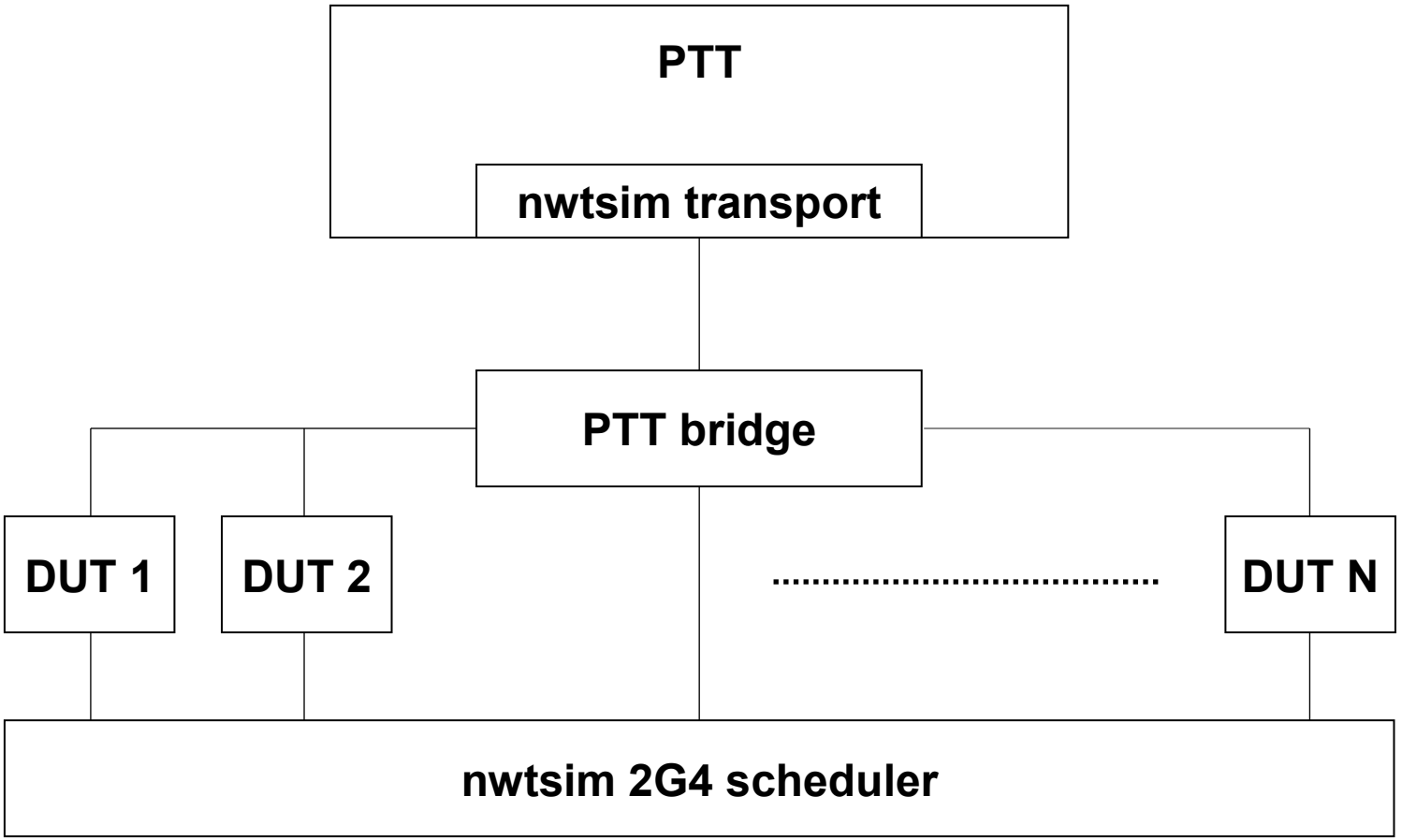

Figure 2 - PTT BRIDGE

The three applications are started. Beginning with PTT BRIDGE. The PTT BRIDGE is handed the name of the simulation “Test” (-s=Test), its own device identifier (-d=0), the number of DUTs to service (-D=2) and the device identifiers for the two DUTs (-dev0=1 and –dev1=2).

Last the PTT APP is started twice to simulate the two DUTs. The PTT APP is handed the name of the simulation “Test” (-s=Test) and its own device identifier (-d=1 or –d=2).

Now the simulation environment is started and ready to receive HCI Command requests from a test. See Running tests to see how the test execution is started.

3 Transport layer

The transport layer is responsible for transporting byte sequences between the PC and the DUT. On the PC side the transport layer is implemented in Python. On the DUT side the transport layer is implemented in C. The transport layer is implemented as Named Pipes functioning as simple FIFOs, two per DUT. One for writing and one for reading.

3.1 The Test side

On the Test side the transport layer is implemented in Python. The Test side communicates with the PTT Bridge that is responsible for writing and reading byte streams to and from the relevant DUTs. Communication with the PTT Bridge takes place via two Named Pipes. The Named Pipes are associated with files created in the temporary folder: /tmp/nwtsim_<user>/<device_id>/.

The two files are:

Device<bridge_id>**.ToPTT** (*opened for reading*) and

Device<bridge_id>**.ToBRIDGE** (*opened for writing*)

On the initial connection, the files are opened and the total number of simulated devices are read.

When sending byte streams to the PTT BRIDGE, the byte stream is prepended with a SEND command identifier, a DUT identifier and the length of the following sequence of bytes.

SEND command token: 2

SEND: <2><device-id><message-length><message>…

When receiving a non-empty byte stream, a RECEIVE command is sent to the PTT BRIDGE with the DUT identifier, a timeout and the number of bytes to receive.

RECEIVE command token: 3

RECEIVE: <3><device-id><timeout><message-length>

Following the RECEIVE command, the transport layer expects to be able to read a message from the PTT BRIDGE. The message starts with a status byte, followed by a time stamp and the actual byte stream.

REPLY: <status><time><message>…

When the transport layer is closed, the PTT Bridge FIFOs are closed and deleted.

3.2 The DUT side

On the DUT side the transport layer is implemented in C. The Named Pipes are associated with files created in the temporary folder: /tmp/nwtsim_<user>/.

The two files are:

Device<device_id>**.PTTin** (*opened for reading*) and

Device<device_id>**.PTTout** (*opened for writing*)

On the initial connection ptt_start(), the files are opened.

When sending and receiving byte streams to and from the PTT BRIDGE, the byte stream is passed through unaltered. Reads can be blocking (PTTT_BLOCK), waiting for the bytes to arrive, or non-blocking (PTTT_NONBLOCK) assuming that the expected number of bytes are waiting in the FIFO.

```
/**
 * Create FIFOs and synchrounize with the Bridge.
 */
bool ptt_start()

/**
 * Close and remove FIFOs.
 */
bool ptt_stop()

/**
 * Read from PTT Bridge.
 *  - pbuffer pointer to buffer to receive message
 *  - size    number of bytes to read
 *  - flags   use non-blocking (PTTT_NONBLOCK) or blocking (PTTT_BLOCK)
              I/O
 * Returns:
 *  - number of bytes read
 */
int ptt_read(u8_t *pbuffer, size_t size, int flags)

/**
 * Write to PTT Bridge.
 *  - pbuffer pointer to buffer with message to write
 *  - size    number of bytes to write
 *  - flags   N.U.
 * Returns:
 *  - number of bytes written
 */
int ptt_write(u8_t *pbuffer, size_t size, int flags)

/**
 * Enable PTT mode. PTT mode must be enabled to read and write FIFOs.
 */
void enable_ptt_mode()

/**
 * Enable PTT auto shutdown on read error.
 *  - mode true to enable auto shutdown; false otherwise
 */
void set ptt autoshutdown(bool mode)
```

Writes are always non-blocking, assuming that the FIFO is large enough to handle the request.

When the transport layer is closed, the PTT Bridge FIFOs are closed and deleted.

```
/**
 * Typical usage…
 */
enable_ptt_mode();
set_ptt_autoshutdown(true);
ptt_start();

while (1) {
      …
      ptt_read(…);
      ptt_write(…);
      …
}
```

4 HCI Command API

The PTT application running in the simulated Zephyr Bluetooth environment implements a HCI Command API that enables tests to execute HCI commands and receive the results of these executions as well as the HCI events generated. The HCI Command API is implemented as a request, reply protocol. In order to use this API, tests must pack commands into a byte stream that is sent over the Named Pipes to the PTT application. The PTT application will generate a reply that is packed into a byte stream and sent back to the test over the Named Pipes.

All requests and replies share a common syntax of the form:

<request> ::= <request_id> <request_size> [<request_parameter_list>]

<request_id> ::= 1 | 3 | 5 | … | 193

<request_size> ::= 0 | 1 | 2 | … | 255

<request_parameter_list> ::= <request_parameter> { <request_parameter> }

<request_parameter> ::= 0 | 1 | 2 | … | 255

<reply> ::= <reply_id> <reply_size> [<reply_parameter_list>]

<reply_id> ::= 2 | 4 | 6 | … | 194

<reply_size> ::= 0 | 1 | 2 | … | 255

<reply_parameter_list> ::= <reply_parameter> { <reply_parameter> }

<reply_parameter> ::= 0 | 1 | 2 | … | 255

<request_id>, <request_size>, <reply_id> and <reply_size> are little-endian 16 bit unsigned numbers. <request_parameter> and <reply_parameter> are 8 bit numbers.

For a list of implemented HCI requests and replies please refer to Appendix A.

5 Running tests

A special Python wrapper has been made to make test execution easy. The wrapper is pttool.py. The syntax is shown below. The <system-id> is a string that must match the <system-id> used for the execution of the PTT APP in the simulator. The <bridge-id> is the device identifier assigned to the PTT BRIDGE. The <test> argument is the name of the Python file (without .py) holding the code for the test to execute. The <trace-level> argument is number that can be used to limit the trace output from the test.

```
./pttool.py -s=<system-id> -d=<bridge-id> --transport nwtsim -T <test>
            -v=<trace-level>

Example:
./pttool.py -s=Test -d=0 --transport nwtsim -T ll_verification -v=5
```

6 Repositories

NWTSIM, PTT Tool and PTT applications all resides in different GIT repositories.

nwtsim resides in the repository: http://gitswarm.kitenet.com:1080/nwtsim/nwtsim.

nrf52_nwtsim resides in the repository: http://gitswarm.kitenet.com:1080/zephyr/nrf52_nwtsim.

ptt_app resides in the repository: http://gitswarm.kitenet.com:1080/zephyr/ptt_app.

PTTool resides in the repository: http://gitswarm.kitenet.com:1080/zephyr/PTTool.

Appendix A. HCI Commands

This Appendix contains a complete list of all the HCI Command requests and responses implemented by the PTT application.

A.1 Reset

Synopsis: The Reset command will reset the Controller and the Link Manager on the BR/ EDR Controller, the PAL on an AMP Controller, or the Link Layer on an LE Controller. If the Controller supports both BR/EDR and LE then the Reset command shall reset the Link Manager, Baseband and Link Layer. The Reset command shall not affect the used HCI transport layer since the HCI transport layers may have reset mechanisms of their own. After the reset is completed, the current operational state will be lost, the Controller will enter standby mode and the Controller will automatically revert to the default values for the parameters for which default values are defined in the specification.

Parameters:

request_id = CMD_RESET_REQ (3)
request_size = 0

Returns:

reply_id = CMD_RESET_RSP (4)
reply_size = 1

Status 0x00 – Reset command succeeded, was received and will be executed.
0x01-0xFF – Reset command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.2 Inquire

Synopsis: This command causes the BR/EDR Controller to enter Inquiry Mode. Inquiry Mode is used to discover other nearby BR/EDR Controllers. The LAP input parameter contains the LAP from which the inquiry access code shall be derived when the inquiry procedure is made. The Inquiry_Length parameter specifies the total duration of the Inquiry Mode and, when this time expires, Inquiry will be halted. When Extended_Inquiry_Length is greater than zero, the duration of the Inquiry Mode may be changed to (Inquiry_Length + Extended_Inquiry_Length). The Num_Responses parameter specifies the number of responses that can be received before the Inquiry is halted. Inquiry Result, Inquiry Result with RSSI, or Extended Inquiry Result events will be sent to report the details of nearby BR/EDR Controllers that have responded to this inquiry. The Inquiry Complete event is sent to report that Inquiry Mode has ended.

Parameters:

request_id = CMD_INQUIRE_REQ (5)
request_size = 5

LAP 0x9E8B00–0x9E8B3F – This is the LAP from which the inquiry access code should be derived when the inquiry procedure is made; see Bluetooth Assigned Numbers.

Inquiry_Length Maximum amount of time specified before the Inquiry is halted.
Size: 1 octet
Range: 0x01 – 0x30
Time = N * 1.28 s
Range: 1.28 – 61.44 s

Num_Responses 0x00 – Unlimited number of responses.
0xXX – Maximum number of responses from the Inquiry before the Inquiry is halted.
Range: 0x01 – 0xFF

Returns:

reply_id = CMD_INQUIRE_RSP (6)
reply_size = 4

Status 0x00 – Command currently in pending.
0x01-0xFF – Command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Num_HCI_Command_Packets The Number of HCI command packets which are allowed to be sent to the Controller from the Host.

Command_Opcode Opcode of the command which caused this event and is pending completion.

A.3 Read Buffer Size

Synopsis: The Read_Buffer_Size command is used to read the maximum size of the data portion of HCI ACL and synchronous Data Packets sent from the Host to the Controller. The Host will segment the data to be transmitted from the Host to the Controller according to these sizes, so that the HCI Data Packets will contain data with up to these sizes. The Read_Buffer_Size command also returns the total number of HCI ACL and synchronous Data Packets that can be stored in the data buffers of the Controller. The Read_Buffer_Size command must be issued by the Host before it sends any data to the Controller.

Parameters:

request_id = CMD_READ_BUFFER_SIZE_REQ (7)
request_size = 0

Returns:

reply_id = CMD_READ_BUFFER_SIZE_RSP (8)
reply_size = 8

Status 0x00 – Read_Buffer_Size command succeeded.
0x01-0xFF – Read_Buffer_Size command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

HC_ACL_Data_Packet_Length Maximum length (in octets) of the data portion of each HCI ACL Data Packet that the Controller is able to accept.

HC_Synchronous_Data_Packet_Length Maximum length (in octets) of the data portion of each HCI Synchronous Data Packet that the Controller is able to accept.

HC_Total_Num_ACL_Data_Packets Total number of HCI ACL Data Packets that can be stored in the data buffers of the Controller.

HC_Total_Num_Synchronous_Data_Packets Total number of HCI Synchronous Data Packets that can be stored in the data buffers of the Controller.

A.4 LE Read Buffer Size

Synopsis: The LE_Read_Buffer_Size command is used to read the maximum size of the data portion of HCI LE ACL Data Packets sent from the Host to the Controller. The Host will segment the data transmitted to the Controller according to these values, so that the HCI Data Packets will contain data with up to this size. The LE_Read_Buffer_Size command also returns the total number of HCI LE ACL Data Packets that can be stored in the data buffers of the Controller. The LE_Read_Buffer_Size command must be issued by the Host before it sends any data to an LE Controller (see Section 4.1.1).

Parameters:

request_id = CMD_LE_READ_BUFFER_SIZE_REQ (9)
request_size = 0

Returns:

reply_id = CMD_LE_READ_BUFFER_SIZE_RSP (10)
reply_size = 4

Status 0x00 – Read_Buffer_Size command succeeded.
0x01-0xFF – Read_Buffer_Size command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

HC_LE_Data_Packet_Length 0x0000 – No dedicated LE Buffer – use Read_Buffer_Size command.
0x0001 – 0xFFFF – Maximum length (in octets) of the data portion of each HCI ACL Data Packet that the Controller is able to accept.

HC_Total_Num_LE_Data_Packets 0x00 – No dedicated LE Buffer – use Read_Buffer_Size command.
0x01 – 0xFF – Total number of HCI ACL Data Packets that can be stored in the data buffers of the Controller.

A.5 Read Local Version Information

Synopsis: This command reads the values for the version information for the local Controller. The HCI Version information defines the version information of the HCI layer. The LMP/PAL Version information defines the version of the LMP or PAL. The Manufacturer_Name information indicates the manufacturer of the local device. The HCI Revision and LMP/PAL Subversion are implementation dependent.

Parameters:

request_id = CMD_READ_LOCAL_VERSION_INFORMATION_REQ (11)
request_size = 0

Returns:

reply_id = CMD_READ_LOCAL_VERSION_INFORMATION_RSP (12)
reply_size = 9

Status 0x00 – Read_Local_Version_Information command succeeded.
0x01-0xFF – Read_Local_Version_Information command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

HCI_Version See Bluetooth Assigned Numbers

HCI_Revision Revision of the Current HCI in the BR/EDR Controller.

LMP/PAL_Version Version of the Current LMP or PAL in the Controller. See Bluetooth Assigned Numbers

Manufacturer_Name Manufacturer Name of the BR/EDR Controller. See Bluetooth Assigned Numbers

LMP/PAL_Subversion Subversion of the Current LMP or PAL in the Controller. This value is implementation dependent.

A.6 Read Local Supported Commands

Synopsis: This command reads the list of HCI commands supported for the local Controller. This command shall return the Supported_Commands configuration parameter. It is implied that if a command is listed as supported, the feature underlying that command is also supported.

Parameters:

request_id = CMD_READ_LOCAL_SUPPORTED_COMMANDS_REQ (13)
request_size = 0

Returns:

reply_id = CMD_READ_LOCAL_SUPPORTED_COMMANDS_RSP (14)
reply_size = 65

Status 0x00 – Read_Local_Supported_Commands command succeeded.
0x01-0xFF – Read_Local_Supported_Commands command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Supported_Commands Bit mask for each HCI Command. If a bit is 1, the Controller supports the corresponding command and the features required for the command. Unsupported or undefined commands shall be set to 0. See section 6.27, “Supported Commands,” on page 498.

A.7 Read Local Supported Features

Synopsis: This command requests a list of the supported features for the local BR/EDR Controller. This command will return a list of the LMP features. For details see [Vol 2] Part C, Link Manager Protocol Specification.

Parameters:

request_id = CMD_READ_LOCAL_SUPPORTED_FEATURES_REQ (15)
request_size = 0

Returns:

reply_id = CMD_READ_LOCAL_SUPPORTED_FEATURES_RSP (16)
reply_size = 9

Status 0x00 – Read_Local_Supported_Features command succeeded.
0x01-0xFF – Read_Local_Supported_Features command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

LMP_Features Bit Mask List of LMP features. For details see Part C, Link Manager Protocol Specification on page 227.

A.8 Read BD_ADDR

Synopsis: On an LE Controller, this command shall read the Public Device Address as defined in [Vol 6] Part B, Section 1.3. If this Controller does not have a Public Device Address, the value 0x000000000000 shall be returned. On a BR/EDR/LE Controller, the public address shall be the same as the BD_ADDR.

Parameters:

request_id = CMD_READ_BD_ADDR_REQ (17)
request_size = 0

Returns:

reply_id = CMD_READ_BD_ADDR_RSP (18)
reply_size = 7

Status 0x00 – Read_BD_ADDR command succeeded.
0x01-0xFF – Read_BD_ADDR command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

BD_ADDR Bluetooth address of the Device

A.9 LE Read Local Supported Features

Synopsis: This command requests the list of the supported LE features for the Controller.

Parameters:

request_id = CMD_LE_READ_LOCAL_SUPPORTED_FEATURES_REQ (19)
request_size = 0

Returns:

reply_id = CMD_LE_READ_LOCAL_SUPPORTED_FEATURES_RSP (20)
reply_size = 9

Status 0x00 – LE_Read_Local_Supported_Features command succeeded.
0x01-0xFF – LE_Read_Local_Supported_Features command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

LE_Features Bit Mask List of LE features. See [Vol 6] Part B, Section 4.6

A.10 LE Read Supported States

Synopsis: The LE_Read_Supported_States command reads the states and state combinations that the link layer supports. See [Vol 6] Part B, Section 1.1.1. LE_States is an 8-octet bit field. If a bit is set to 1 then this state or state combination is supported by the Controller. Multiple bits in LE_States may be set to 1 to indicate support for multiple state and state combinations.

Parameters:

request_id = CMD_LE_READ_SUPPORTED_STATES_REQ (21)
request_size = 0

Returns:

reply_id = CMD_LE_READ_SUPPORTED_STATES_RSP (22)
reply_size = 9

Status 0x00 – LE_Read_Supported_States command succeeded.
0x01-0xFF – LE_Read_Supported_States command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

LE_States Bit Mask List of LE states or state combinations.

A.11 LE Set Advertising Enable

Synopsis: The LE_Set_Advertising_Enable command is used to request the Controller to start or stop advertising. The Controller manages the timing of advertisements as per the advertising parameters given in the LE_Set_Advertising_Parameters command.

Parameters:

request_id = CMD_LE_SET_ADVERTISING_ENABLE_REQ (23)
request_size = 1

Advertising_Enable 0x00 – Advertising is disabled (default).
0x01 – Advertising is enabled.
0x02 – 0xFF – Reserved for future use.

Returns:

reply_id = CMD_LE_SET_ADVERTISING_ENABLE_RSP (24)
reply_size = 1

Status 0x00 – LE_Set_Advertising_Enable command succeeded.
0x01-0xFF – LE_Set_Advertising_Enable command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.12 LE Set Advertising Data

Synopsis: The LE_Set_Advertising_Data command is used to set the data used in advertising packets that have a data field.

Parameters:

request_id = CMD_LE_SET_ADVERTISING_DATA_REQ (25)
request_size = 32

Advertising_Data_Length The number of significant octets in the Advertising_Data.

Advertising_Data 31 octets of advertising data formatted as defined in [Vol 3] Part C, Section 11. All octets zero (default).

Returns:

reply_id = CMD_LE_SET_ADVERTISING_DATA_RSP (26)
reply_size = 1

Status 0x00 – LE_Set_Advertising_Data command succeeded.
0x01-0xFF – LE_Set_Advertising_Data command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.13 LE Set Advertising Parameters

Synopsis: The LE_Set_Advertising_Parameters command is used by the Host to set the advertising parameters.

Parameters:

request_id = CMD_LE_SET_ADVERTISING_PARAMETERS_REQ (27)
request_size = 15

Advertising_Interval_Min Minimum advertising interval for undirected and low duty cycle directed advertising.
Range: 0x0020 to 0x4000.
Default: N = 0x0800 (1.28 second).
Time = N * 0.625 ms.
Time Range: 20 ms to 10.24 sec.

Advertising_Interval_Max Maximum advertising interval for undirected and low duty cycle directed advertising.
Range: 0x0020 to 0x4000.
Default: N = 0x0800 (1.28 second).
Time = N * 0.625 ms.
Time Range: 20 ms to 10.24 sec.

Advertising_Type 0x00 – Connectable undirected advertising (ADV_IND) (default).
0x01 – Connectable high duty cycle directed advertising (ADV_DIRECT_IND, high duty cycle).
0x02 – Scannable undirected advertising (ADV_SCAN_IND).
0x03 – Non connectable undirected advertising (ADV_NONCONN_IND).
0x04 – Connectable low duty cycle directed advertising (ADV_DIRECT_IND, low duty cycle).
0x05 – 0xFF – Reserved for future use.

Own_Address_Type 0x00 – Public Device Address (default).
0x01 – Random Device Address.
0x02 – Controller generates Resolvable Private Address based on the local IRK from the resolving list.
If the resolving list contains no matching entry, use the public address.
0x03 – Controller generates Resolvable Private Address based on the local IRK from the resolving list.
If the resolving list contains no matching entry, use the random address from LE_Set_Random_Address.
0x04 – 0xFF – Reserved for future use.

Peer_Address_Type 0x00 – Public Device Address (default).
0x01 – Random Device Address.
0x02 – 0xFF – Reserved for future use.

Peer_Address Public Device Address or Random Device Address of the device to be connected.

Advertising_Channel_Map 00000000b – Reserved for future use.
Xxxxxxx1b – Enable channel 37 use.
xxxxxx1xb – Enable channel 38 use.
xxxxx1xxb – Enable channel 39 use.
xxxxx111b – Default (all channels enabled).

Advertising_Filter_Policy 0x00 – Process scan and connection requests from all devices (i.e., the White List is not in use) (default).
0x01 – Process connection requests from all devices and only scan requests from devices that are in the White List.
0x02 – Process scan requests from all devices and only connection requests from devices that are in the White List.
0x03 – Process scan and connection requests only from devices in the White List.
0x04 – 0xFF – Reserved for future use.

Returns:

reply_id = CMD_LE_SET_ADVERTISING_PARAMETERS_RSP (28)
reply_size = 1

Status 0x00 – LE_Set_Advertising_Parameters command succeeded.
0x01-0xFF – LE_Set_Advertising_Parameters command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.14 LE Set Random Address

Synopsis: The LE_Set_Random_Address command is used by the Host to set the LE Random Device Address in the Controller (see [Vol 6] Part B, Section 1.3).

Parameters:

request_id = CMD_LE_SET_RANDOM_ADDRESS_REQ (29)
request_size = 6

Random_Addess Random Device Address as defined by [Vol 6] Part B, Section 1.3.

Returns:

reply_id = CMD_LE_SET_RANDOM_ADDRESS_RSP (30)
reply_size = 1

Status 0x00 – LE_Set_Random_Address command succeeded.
0x01-0xFF – LE_Set_Random_Address command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.15 LE Set Scan Response Data

Synopsis: This command is used to provide data used in Scanning Packets that have a data field.

Parameters:

request_id = CMD_LE_SET_SCAN_RESPONSE_DATA_REQ (31)
request_size = 32

Scan_Response_Data_Length The number of significant octets in the Scan_Response_Data.

Scan_Response_Data 31 octets of Scan_Response_Data formatted as defined in [Vol 3] Part C, Section 11. All octets zero (default).

Returns:

reply_id = CMD_LE_SET_SCAN_RESPONSE_DATA_RSP (32)
reply_size = 1

Status 0x00 – LE_Set_Scan_Response_Data command succeeded.
0x01-0xFF – LE_Set_Scan_Response_Data command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.16 LE Set Scan Enable

Synopsis: The LE_Set_Scan_Enable command is used to start scanning. Scanning is used to discover advertising devices nearby.

Parameters:

request_id = CMD_LE_SET_SCAN_ENABLE_REQ (33)
request_size = 2

LE_Scan_Enable 0x00 – Scanning disabled.
0x01 – Scanning enabled.
0x02 – 0xFF – Reserved for future use.

Filter_Duplicates 0x00 – Duplicate filtering disabled.
0x01 – Duplicate filtering enabled.
0x02 – 0xFF – Reserved for future use.

Returns:

reply_id = CMD_LE_SET_SCAN_ENABLE_RSP (34)
reply_size = 1

Status 0x00 – LE_Set_Scan_Enable command succeeded.
0x01-0xFF – LE_Set_Scan_Enable command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.17 LE Set Scan Parameters

Synopsis: The LE_Set_Scan_Parameters command is used to set the scan parameters. The LE_Scan_Type parameter controls the type of scan to perform.

Parameters:

request_id = CMD_LE_SET_SCAN_PARAMETERS_REQ (35)
request_size = 7

LE_Scan_Type 0x00 – Passive Scanning. No SCAN_REQ packets shall be sent (default).
0x01 – Active scanning. SCAN_REQ packets may be sent.
0x02 – 0xFF – Reserved for future use.

LE_Scan_Interval The Scan Interval is defined as the time interval from when the Controller started its last LE scan until it begins the subsequent LE scan.
Range: 0x0004 to 0x4000.
Default: 0x0010 (10 ms).
Time = N * 0.625 ms.
Time Range: 2.5 ms to 10.24 seconds.

LE_Scan_Window The duration of the LE scan. LE_Scan_Window shall be less than or equal to LE_Scan_Interval.
Range: 0x0004 to 0x4000.
Default: 0x0010 (10 ms).
Time = N * 0.625 ms.
Time Range: 2.5 ms to 10.24 seconds.

Own_Address_Type 0x00 – Public Device Address (default).
0x01 – Random Device Address.
0x02 – Controller generates Resolvable Private Address based on the local IRK from the resolving list.

If the resolving list contains no matching entry, use the public address.
0x03 – Controller generates Resolvable Private Address based on the local IRK from the resolving list.
If the resolving list contains no matching entry, use the random address from LE_Set_Random_Address.
All other values reserved for future use.

Scanning_Filter_Policy 0x00 – Accept all advertising packets except directed advertising packets not addressed to this device (default).
0x01 – Accept only advertising packets from devices where the advertiser's address is in the White List.
Directed advertising packets which are not addressed to this device shall be ignored.
0x02 – Accept all advertising packets except directed advertising packets where the initiator's identity address does not address this device.
Note: Directed advertising packets where the initiator's address is a resolvable private address that cannot be resolved are also accepted.
0x03 – Accept all advertising packets except:
• advertising packets where the advertiser's identity address is not in the White List; and
• directed advertising packets where the initiator's identity address does not address this device.
Note: Directed advertising packets where the initiator's address is a resolvable private address that cannot be resolved are also accepted.
0x04 – 0xFF – Reserved for future use.

Returns:

reply_id = CMD_LE_SET_SCAN_PARAMETERS_RSP (36)
reply_size = 1

Status 0x00 – LE_Set_Scan_Parameters command succeeded.
0x01-0xFF – LE_Set_Scan_Parameters command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.18 Disconnect

Synopsis: The Disconnect command is used to terminate an existing connection.

Parameters:

request_id = CMD_DISCONNECT_REQ (37)
request_size = 3

Connection_Handle Connection handle for the connection being disconnected.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

Reason Authentication Failure error code (0x05),
Other End Terminated Connection error codes (0x13-0x15),
Unsupported Remote Feature error code (0x1A),
Pairing with Unit Key Not Supported error code (0x29) and
Unacceptable Connection Interval error code (0x3B), see Part D, Error Codes on page 379 for a list of error codes and descriptions.

Returns:

reply_id = CMD_DISCONNECT_RSP (38)
reply_size = 4

Status 0x00 – Command currently in pending.
0x01-0xFF – Command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Num_HCI_Command_Packets The Number of HCI command packets which are allowed to be sent to the Controller from the Host.

Command_Opcode Opcode of the command which caused this event and is pending completion.

A.19 LE Create Connection Cancel

Synopsis: The LE_Create_Connection_Cancel command is used to cancel the LE_Create_Connection or LE_Extended_Create_Connection commands. This command shall only be issued after the LE_Create_Connection or LE_Extended_Create_Connection commands have been issued, a Command Status event has been received for the LE Create Connection or LE_Extended_Create_Connection commands, and before the LE Connection Complete or LE Enhanced Connection Complete events.

Parameters:

request_id = CMD_LE_CREATE_CONNECTION_CANCEL_REQ (39)
request_size = 0

Returns:

reply_id = CMD_LE_CREATE_CONNECTION_CANCEL_RSP (40)
reply_size = 1

Status 0x00 – LE_Create_Connection_Cancel command succeeded.
0x01-0xFF – LE_Create_Connection_Cancel command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.20 LE Create Connection

Synopsis: The LE_Create_Connection command is used to create a Link Layer connection to a connectable advertiser.

Parameters:

request_id = CMD_LE_CREATE_CONNECTION_REQ (41)
request_size = 25

LE_Scan_Interval The Scan Interval is defined as the time interval from when the Controller started its last LE scan until it begins the subsequent LE scan.
Range: 0x0004 to 0x4000.
Default: 0x0010 (10 ms).
Time = N * 0.625 ms.
Time Range: 2.5 ms to 10.24 seconds.

LE_Scan_Window Amount of time for the duration of the LE scan. LE_Scan_Window shall be less than or equal to LE_Scan_Interval.
Range: 0x0004 to 0x4000.
Default: 0x0010 (10 ms).

Time = N * 0.625 ms.
Time Range: 2.5 ms to 10.24 seconds.

Initiator_Filter_Policy 0x00 – White list is not used to determine which advertiser to connect to. Peer_Address_Type and Peer_Address shall be used.
0x01 – White list is used to determine which advertiser to connect to. Peer_Address_Type and Peer_Address shall be ignored.
0x02 – 0xFF – Reserved for future use.

Peer_Address_Type 0x00 – Public Device Address.
0x01 – Random Device Address.
0x02 – Public Identity Address (Corresponds to peer's Resolvable Private Address).
This value shall only be used by the Host if either the Host or the Controller does not support the LE Set Privacy Mode command.
0x03 – Random (static) Identity Address (Corresponds to peer's Resolvable Private Address).
This value shall only be used by a Host if either the Host or the Controller does not support the LE Set Privacy Mode command.
0x04 – 0xFF – Reserved for future use.

Peer_Address Public Device Address, Random Device Address, Public Identity Address, or Random (static) Identity Address of the device to be connected.

Own_Address_Type 0x00 – Public Device Address (default).
0x01 – Random Device Address.
0x02 – Controller generates Resolvable Private Address based on the local IRK from the resolving list.
If the resolving list contains no matching entry, use the public address.
0x03 – Controller generates Resolvable Private Address based on the local IRK from the resolving list.
If the resolving list contains no matching entry, use the random address from the most recent successful LE_Set_Random_Address Command.
0x04 – 0xFF – Reserved for future use.

Conn_Interval_Min Minimum value for the connection event interval. This shall be less than or equal to Conn_Interval_Max.
Range: 0x0006 to 0x0C80.
Time = N * 1.25 ms.
Time Range: 7.5 ms to 4 seconds.

Conn_Interval_Max Maximum value for the connection event interval. This shall be greater than or equal to Conn_Interval_Min.
Range: 0x0006 to 0x0C80.
Time = N * 1.25 ms.
Time Range: 7.5 ms to 4 seconds.

Conn_Latency Slave latency for the connection in number of connection events.
Range: 0x0000 to 0x01F3.

Supervision_Timeout Supervision timeout for the LE Link. (See [Vol 6] Part B, Section 4.5.2).
Range: 0x000A to 0x0C80.
Time = N * 10 ms.
Time Range: 100 ms to 32 seconds.

Minimum_CE_Length Information parameter about the minimum length of connection event needed for this LE connection.
Range: 0x0000 – 0xFFFF.
Time = N * 0.625 ms.

Maximum_CE_Length Information parameter about the maximum length of connection event needed for this LE connection.
Range: 0x0000 – 0xFFFF.
Time = N * 0.625 ms.

Returns:

reply_id = CMD_LE_CREATE_CONNECTION_RSP (42)
reply_size = 4

Status 0x00 – Command currently in pending.
0x01-0xFF – Command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Num_HCI_Command_Packets The Number of HCI command packets which are allowed to be sent to the Controller from the Host.

Command_Opcode Opcode of the command which caused this event and is pending completion.

A.21 Read Remote Version Information

Synopsis: This command will obtain the values for the version information for the remote device identified by the Connection_Handle parameter.

Parameters:

request_id = CMD_READ_REMOTE_VERSION_INFORMATION_REQ (43)
request_size = 2

Connection_Handle Specifies which Connection Handle's version information to get.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

Returns:

reply_id = CMD_READ_REMOTE_VERSION_INFORMATION_RSP (44)
reply_size = 4

Status 0x00 – Command currently in pending.
0x01-0xFF – Command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Num_HCI_Command_Packets The Number of HCI command packets which are allowed to be sent to the Controller from the Host.

Command_Opcode Opcode of the command which caused this event and is pending completion.

A.22 LE Read Remote Features

Synopsis: This command requests, from the remote device identified by the connection handle, the features used on the connection and the features supported by the remote device. For details see [Vol 6] Part B, Section 4.6.

Parameters:

request_id = CMD_LE_READ_REMOTE_FEATURES_REQ (45)
request_size = 2

Connection_Handle Connection handle to be used to identify a connection.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

Returns:

reply_id = CMD_LE_READ_REMOTE_FEATURES_RSP (46)
reply_size = 4

Status 0x00 – Command currently in pending.
0x01-0xFF – Command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Num_HCI_Command_Packets The Number of HCI command packets which are allowed to be sent to the Controller from the Host.

Command_Opcode Opcode of the command which caused this event and is pending completion.

A.23 LE Connection Update

Synopsis: The LE_Connection_Update command is used to change the Link Layer connection parameters of a connection. This command may be issued on both the master and slave.

Parameters:

request_id = CMD_LE_CONNECTION_UPDATE_REQ (47)
request_size = 14

Connection_Handle Connection handle to be used to identify a connection.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

Conn_Interval_Min Minimum value for the connection event interval. This shall be less than or equal to Conn_Interval_Max.
Range: 0x0006 to 0x0C80.
Time = N * 1.25 ms.
Time Range: 7.5 ms to 4 seconds.

Conn_Interval_Max Maximum value for the connection event interval. This shall be greater than or equal to Conn_Interval_Min.
Range: 0x0006 to 0x0C80.
Time = N * 1.25 ms.
Time Range: 7.5 ms to 4 seconds.

Conn_Latency Slave latency for the connection in number of connection events.
Range: 0x0000 to 0x01F3.

Supervision_Timeout Supervision timeout for the LE Link. (See [Vol 6] Part B, Section 4.5.2).
Range: 0x000A to 0x0C80.
Time = N * 10 ms.
Time Range: 100 ms to 32 seconds.

Minimum_CE_Length Information parameter about the minimum length of connection event needed for this LE connection.
Range: 0x0000 – 0xFFFF.
Time = N * 0.625 ms.

Maximum_CE_Length Information parameter about the maximum length of connection event needed for this LE connection.
Range: 0x0000 – 0xFFFF.
Time = N * 0.625 ms.

Returns:

reply_id = CMD_LE_CONNECTION_UPDATE_RSP (48)
reply_size = 4

Status 0x00 – Command currently in pending.
0x01-0xFF – Command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Num_HCI_Command_Packets The Number of HCI command packets which are allowed to be sent to the Controller from the Host.

Command_Opcode Opcode of the command which caused this event and is pending completion.

A.24 LE Remote Connection Parameter Request Reply

Synopsis: Both the master Host and the slave Host use this command to reply to the HCI LE Remote Connection Parameter Request event. This indicates that the Host has accepted the remote device's request to change connection parameters.

Parameters:

request_id =
CMD_LE_REMOTE_CONNECTION_PARAMETER_REQUEST_REPLY_REQ (49)
request_size = 14

Connection_Handle Connection handle to be used to identify a connection.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use

Interval_Min Minimum value of the connection interval.
Range: 0x0006 to 0x0C80.
Time = N * 1.25 ms.
Time Range: 7.5 ms to 4 seconds.

Interval_Max Maximum value of the connection interval.
Range: 0x0006 to 0x0C80.
Time = N * 1.25 ms.
Time Range: 7.5 ms to 4 seconds.

Latency Maximum allowed slave latency for the connection specified as the number of connection events.
Range: 0x0000 to 0x01F3 (499).

Timeout Supervision timeout for the connection.
Range: 0x000A to 0x0C80.
Time = N * 10 ms.
Time Range: 100 ms to 32 seconds.

Minimum_CE_Length Information parameter about the minimum length of connection event needed for this LE connection.
Range: 0x0000 – 0xFFFF.
Time = N * 0.625 ms.

Maximum_CE_Length Information parameter about the maximum length of connection event needed for this LE connection.
Range: 0x0000 – 0xFFFF.
Time = N * 0.625 ms.

Returns:

reply_id = CMD_LE_REMOTE_CONNECTION_PARAMETER_REQUEST_REPLY_RSP (50)
reply_size = 3

Status 0x00 – LE_Remote_Connection_Parameter_Request_Reply command succeeded.
0x01-0xFF – LE_Remote_Connection_Parameter_Request_Reply command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Connection_Handle Connection handle to be used to identify a connection.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

A.25 LE Remote Connection Parameter Request Negative Reply

Synopsis: Both the master Host and the slave Host use this command to reply to the HCI LE Remote Connection Parameter Request event. This indicates that the Host has rejected the remote device's request to change connection parameters. The reason for the rejection is given in the Reason parameter.

Parameters:

request_id = CMD_LE_REMOTE_CONNECTION_PARAMETER_REQUEST_NEGATIVE_REPLY_REQ (51)
request_size = 3

Connection_Handle Connection handle to be used to identify a connection.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

Reason The reason that the connection parameter request was rejected. See [Vol 2] Part D, Error Codes for a list of error codes and descriptions.

Returns:

reply_id = CMD_LE_REMOTE_CONNECTION_PARAMETER_REQUEST_NEGATIVE_REPLY_RSP (52)
reply_size = 3

Status 0x00 – LE_Remote_Connection_Parameter_Request_Negative_Reply command succeeded.
0x01-0xFF – LE_Remote_Connection_Parameter_Request_Negative_Reply command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Connection_Handle Connection handle to be used to identify a connection.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

A.26 LE Set Host Channel Classification

Synopsis: The LE_Set_Host_Channel_Classification command allows the Host to specify a channel classification for data channels based on its “local information”. This classification persists until overwritten with a subsequent LE_Set_Host_Channel_Classification command or until the Controller is reset using the Reset command (see [Vol 6] Part B, Section 4.5.8.1).

Parameters:

request_id = CMD_LE_SET_HOST_CHANNEL_CLASSIFICATION_REQ (53)
request_size = 5

Channel_Map This parameter contains 37 1-bit fields. The n'th such field (in the range 0 to 36) contains the value for the link layer channel index n.
Channel n is bad = 0.
Channel n is unknown = 1.
The most significant bits are reserved and shall be set to 0. At least one channel shall be marked as unknown.

Returns:

reply_id = CMD_LE_SET_HOST_CHANNEL_CLASSIFICATION_RSP (54)
reply_size = 1

Status 0x00 – LE_Set_Host_Channel_Classification command succeeded.
0x01-0xFF – LE_Set_Host_Channel_Classification command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.27 Host Buffer Size

Synopsis: The Host_Buffer_Size command is used by the Host to notify the Controller about the maximum size of the data portion of HCI ACL and synchronous Data Packets sent from the Controller to the Host.

Parameters:

request_id = CMD_HOST_BUFFER_SIZE_REQ (55)
request_size = 7

Host_ACL_Data_Packet_Length Maximum length (in octets) of the data portion of each HCI ACL Data Packet that the Host is able to accept.

Host_Synchronous_Data_Packet_Length Maximum length (in octets) of the data portion of each HCI synchronous Data Packet that the Host is able to accept.

Host_Total_Num_ACL_Data_Packets Total number of HCI ACL Data Packets that can be stored in the data buffers of the Host.

Host_Total_Num_Synchronous_Data_Packets Total number of HCI synchronous Data Packets that can be stored in the data buffers of the Host.

Returns:

reply_id = CMD_HOST_BUFFER_SIZE_RSP (56)
reply_size = 1

Status 0x00 – Host_Buffer_Size command succeeded.
0x01-0xFF – Host_Buffer_Size command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.28 Set Event Mask

Synopsis: The Set_Event_Mask command is used to control which events are generated by the HCI for the Host. If the bit in the Event_Mask is set to a one, then the event associated with that bit will be enabled. For an LE Controller, the “LE Meta Event” bit in the Event_Mask shall enable or disable all LE events in the LE Meta Event (see Section 7.7.65).

Parameters:

request_id = CMD_SET_EVENT_MASK_REQ (57)
request_size = 8

Event_Mask 0x0000000000000000 – No events specified.
0x0000000000000001 – Inquiry Complete Event.
0x0000000000000002 – Inquiry Result Event.
0x0000000000000004 – Connection Complete Event.
0x0000000000000008 – Connection Request Event.
0x0000000000000010 – Disconnection Complete Event.
0x0000000000000020 – Authentication Complete Event.
0x0000000000000040 – Remote Name Request Complete Event.
0x0000000000000080 – Encryption Change Event.
0x0000000000000100 – Change Connection Link Key Complete Event.
0x0000000000000200 – Master Link Key Complete Event.
0x0000000000000400 – Read Remote Supported Features Complete Event.
0x0000000000000800 – Read Remote Version Information Complete Event.
0x0000000000001000 – QoS Setup Complete Event.
0x0000000000002000 – Reserved.
0x0000000000004000 – Reserved.
0x0000000000008000 – Hardware Error Event.
0x0000000000010000 – Flush Occurred Event.
0x0000000000020000 – Role Change Event.
0x0000000000040000 – Reserved.
0x0000000000080000 – Mode Change Event.
0x0000000000100000 – Return Link Keys Event.
0x0000000000200000 – PIN Code Request Event.
0x0000000000400000 – Link Key Request Event.
0x0000000000800000 – Link Key Notification Event.
0x0000000001000000 – Loopback Command Event.
0x0000000002000000 – Data Buffer Overflow Event.
0x0000000004000000 – Max Slots Change Event.
0x0000000008000000 – Read Clock Offset Complete Event.
0x0000000010000000 – Connection Packet Type Changed Event.
0x0000000020000000 – QoS Violation Event.
0x0000000040000000 – Page Scan Mode Change Event [deprecated].
0x0000000080000000 – Page Scan Repetition Mode Change Event.
0x0000000100000000 – Flow Specification Complete Event.
0x0000000200000000 – Inquiry Result with RSSI Event.
0x0000000400000000 – Read Remote Extended Features Complete Event.
0x0000000800000000 – Reserved.
0x0000001000000000 – Reserved.
0x0000002000000000 – Reserved.
0x0000004000000000 – Reserved.
0x0000008000000000 – Reserved.
0x0000010000000000 – Reserved.
0x0000020000000000 – Reserved.
0x0000040000000000 – Reserved.

0x0000080000000000 – Synchronous Connection Complete Event.
0x0000100000000000 – Synchronous Connection Changed Event.
0x0000200000000000 – Sniff Subrating Event.
0x0000400000000000 – Extended Inquiry Result Event.
0x0000800000000000 – Encryption Key Refresh Complete Event.
0x0001000000000000 – IO Capability Request Event.
0x0002000000000000 – IO Capability Request Reply Event.
0x0004000000000000 – User Confirmation Request Event.
0x0008000000000000 – User Passkey Request Event.
0x0010000000000000 – Remote OOB Data Request Event.
0x0020000000000000 – Simple Pairing Complete Event.
0x0040000000000000 – Reserved.
0x0080000000000000 – Link Supervision Timeout Changed Event.
0x0100000000000000 – Enhanced Flush Complete Event.
0x0200000000000000 – Reserved.
0x0400000000000000 – User Passkey Notification Event.
0x0800000000000000 – Keypress Notification Event.
0x1000000000000000 – Remote Host Supported Features Notification Event.
0x2000000000000000 – LE Meta-Event.
0x00001FFFFFFFFFFF – Default.
0xC000000000000000 – Reserved for future use.

Returns:

reply_id = CMD_SET_EVENT_MASK_RSP (58)
reply_size = 1

Status 0x00 – Set_Event_Mask command succeeded.
0x01-0xFF – Set_Event_Mask command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.29 Set Controller To Host Flow Control

Synopsis: This command is used by the Host to turn flow control on or off for data and/or voice sent in the direction from the Controller to the Host.

Parameters:

request_id = CMD_SET_CONTROLLER_TO_HOST_FLOW_CONTROL_REQ (59)
request_size = 1

Flow_Control_Enable 0x00 – Flow control off in direction from Controller to Host (default).
0x01 – Flow control on for HCI ACL Data Packets and off for HCI synchronous Data Packets in direction from Controller to Host.
0x02 – Flow control off for HCI ACL Data Packets and on for HCI synchronous Data Packets in direction from Controller to Host.
0x03 – Flow control on both for HCI ACL Data Packets and HCI synchronous Data Packets in direction from Controller to Host.
0x04-0xFF – Reserved.

Returns:

reply_id = CMD_SET_CONTROLLER_TO_HOST_FLOW_CONTROL_RSP (60)
reply_size = 1

Status 0x00 – Set_Controller_To_Host_Flow_Control command succeeded.
0x01-0xFF – Set_Controller_To_Host_Flow_Control command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.30 Host Number Of Completed Packets

Synopsis: The Host_Number_Of_Completed_Packets command is used by the Host to indicate to the Controller the number of HCI Data Packets that have been completed for each Connection_Handle since the previous Host_Number_Of_Completed_Packets command was sent to the Controller.

Parameters:

request_id = CMD_HOST_NUMBER_OF_COMPLETED_PACKETS_REQ (61)
request_size = 1 + 4 * Number_Of_Handles

Number_Of_Handles (N) Number of Connection Handles and Host_Num_Of_Completed_Packets parameters pairs contained in this command. Range: 0-255.

Connection_Handle[i] Connection Handle.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

Host_Num_Of_Completed_Packets [i] Number of HCI Data Packets that have been completed for the associated Connection Handle since the previous time the event was returned.
Range for N: 0x0000-0xFFFF.

Returns:

reply_id = CMD_HOST_NUMBER_OF_COMPLETED_PACKETS_RSP (62)
reply_size = 4

Status 0x00 – Command currently in pending.
0x01-0xFF – Command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Num_HCI_Command_Packets The Number of HCI command packets which are allowed to be sent to the Controller from the Host.

Command_Opcode Opcode of the command which caused this event and is pending completion.

A.31 Set Event Mask Page 2

Synopsis: The Set_Event_Mask_Page_2 command is used to control which events are generated by the HCI for the Host. The Event_Mask_Page_2 is a logical extension to the Event_Mask parameter of the Set_Event_Mask command.

Parameters:

request_id = CMD_SET_EVENT_MASK_PAGE_2_REQ (63)
request_size = 8

Event_Mask_Page_2 0x0000000000000000 – No events specified (default).
0x0000000000000001 – Physical Link Complete Event.
0x0000000000000002 – Channel Selected Event.
0x0000000000000004 – Disconnection Physical Link Event.
0x0000000000000008 – Physical Link Loss Early Warning Event.
0x0000000000000010 – Physical Link Recovery Event.
0x0000000000000020 – Logical Link Complete Event.
0x0000000000000040 – Disconnection Logical Link Complete Event.
0x0000000000000080 – Flow Spec Modify Complete Event.

0x0000000000000100 – Number of Completed Data Blocks Event.
0x0000000000000200 – AMP Start Test Event.
0x0000000000000400 – AMP Test End Event.
0x0000000000000800 – AMP Receiver Report Event.
0x0000000000001000 – Short Range Mode Change Complete Event.
0x0000000000002000 – AMP Status Change Event.
0x0000000000004000 – Triggered Clock Capture Event.
0x0000000000008000 – Synchronization Train Complete Event.
0x0000000000010000 – Synchronization Train Received Event.
0x0000000000020000 – Connectionless Slave Broadcast Receive Event.
0x0000000000040000 – Connectionless Slave Broadcast Timeout Event.
0x0000000000080000 – Truncated Page Complete Event.
0x0000000000100000 – Slave Page Response Timeout Event.
0x0000000000200000 – Connectionless Slave Broadcast Channel Map Change Event.
0x0000000000400000 – Inquiry Response Notification Event.
0x0000000000800000 – Authenticated Payload Timeout Expired Event.
0xFFFFFFFFFF000000 – Reserved for future use.

Returns:

reply_id = CMD_SET_EVENT_MASK_PAGE_2_RSP (64)
reply_size = 1

Status 0x00 – Set_Event_Mask_Page_2 command succeeded.
0x01-0xFF – Set_Event_Mask_Page_2 command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.32 LE Add Device To White List

Synopsis: The LE_Add_Device_To_White_List command is used to add a single device to the White List stored in the Controller.

Parameters:

request_id = CMD_LE_ADD_DEVICE_TO_WHITE_LIST_REQ (65)
request_size = 7

Address_Type 0x00 – Public Device Address.
0x01 – Random Device Address.
0x02 – 0xFF – Reserved for future use.

Address Public Device Address or Random Device Address of the device to be added to the white list.

Returns:

reply_id = CMD_LE_ADD_DEVICE_TO_WHITE_LIST_RSP (66)
reply_size = 1

Status 0x00 – LE_Add_Device_To_White_List command succeeded.
0x01-0xFF – LE_Add_Device_To_White_List command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.33 LE Clear White List

Synopsis: The LE_Clear_White_List command is used to clear the White List stored in the Controller.

Parameters:

request_id = CMD_LE_CLEAR_WHITE_LIST_REQ (67)
request_size = 0

Returns:

reply_id = CMD_LE_CLEAR_WHITE_LIST_RSP (68)
reply_size = 1

Status 0x00 – LE_Clear_White_List command succeeded.
0x01-0xFF – LE_Clear_White_List command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.34 LE Read White List Size

Synopsis: The LE_Read_White_List_Size command is used to read the total number of White List entries that can be stored in the Controller.

Parameters:

request_id = CMD_LE_READ_WHITE_LIST_SIZE_REQ (69)
request_size = 0

Returns:

reply_id = CMD_LE_READ_WHITE_LIST_SIZE_RSP (70)
reply_size = 2

Status 0x00 – LE_Read_White_List_Size command succeeded.
0x01-0xFF – LE_Read_White_List_Size command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

White_List_Size 0x01 – 0xFF – Total number of white list entries that can be stored in the Controller.
0x00 – Reserved for future use.

A.35 LE Remove Device From White List

Synopsis: The LE_Remove_Device_From_White_List command is used to remove a single device from the White List stored in the Controller.

Parameters:

request_id = CMD_LE_REMOVE_DEVICE_FROM_WHITE_LIST_REQ (71)
request_size = 7

Address_Type 0x00 – Public Device Address.
0x01 – Random Device Address.
0x02 – 0xFF – Reserved for future use.

Address Public Device Address or Random Device Address of the device to be removed from the white list.

Returns:

reply_id = CMD_LE_REMOVE_DEVICE_FROM_WHITE_LIST_RSP (72)
reply_size = 1

Status 0x00 – LE_Remove_Device_From_White_List command succeeded.
0x01-0xFF – LE_Remove_Device_From_White_List command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.36 LE Set Event Mask

Synopsis: The LE_Set_Event_Mask command is used to control which LE events are generated by the HCI for the Host.

Parameters:

request_id = CMD_LE_SET_EVENT_MASK_REQ (73)
request_size = 8

LE_Event_Mask 0x0000000000000000 – No LE events specified.
0x0000000000000001 – LE Connection Complete Event.
0x0000000000000002 – LE Advertising Report Event.
0x0000000000000004 – LE Connection Update Complete Event.
0x0000000000000008 – LE Read Remote Used Features Complete Event.
0x0000000000000010 – LE Long Term Key Request Event.
0x0000000000000020 – LE Remote Connection Parameter Request Event.
0x000000000000001F – Default.
0xFFFFFFFFFFFFFFC0 – Reserved for future use.

Returns:

reply_id = CMD_LE_SET_EVENT_MASK_RSP (74)
reply_size = 1

Status 0x00 – LE_Set_Event_Mask command succeeded.
0x01-0xFF – LE_Set_Event_Mask command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.37 Read Transmit Power Level

Synopsis: This command reads the values for the Transmit_Power_Level parameter for the specified Connection_Handle. The Connection_Handle shall be a Connection_Handle for an ACL connection.

Parameters:

request_id = CMD_READ_TRANSMIT_POWER_LEVEL_REQ (75)
request_size = 3

Connection_Handle Specifies which Connection Handle's Transmit Power Level setting to read.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

Type 0x00 – Read Current Transmit Power Level.
0x01 – Read Maximum Transmit Power Level.
0x02-0xFF – Reserved.

Returns:

reply_id = CMD_READ_TRANSMIT_POWER_LEVEL_RSP (76)
reply_size = 4

Status 0x00 – Read_Transmit_Power_Level command succeeded.
0x01-0xFF – Read_Transmit_Power_Level command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Connection_Handle Specifies which Connection Handle's Transmit Power Level setting is returned. Range: 0x0000-0x0EFF (0x0F00 - 0x0FFF Reserved for future use).

Transmit_Power_Level Size: 1 Octet (signed integer).
Range: -30 ≤ N ≤ 20.
Units: dBm.

A.38 Read RSSI

Synopsis: This command reads the Received Signal Strength Indication (RSSI) value from a Controller.

Parameters:

request_id = CMD_READ_RSSI_REQ (77)
request_size = 2

Handle The Handle for the connection for which the RSSI is to be read. The Handle is a Connection_Handle for a BR/EDR Controller and a Physical_Link_Handle for an AMP Controller.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

Returns:

reply_id = CMD_READ_RSSI_RSP (78)
reply_size = 4

Status 0x00 – Read_RSSI command succeeded.
0x01-0xFF – Read_RSSI command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Handle The Handle for the connection for which the RSSI has been read. The Handle is a Connection_Handle for a BR/EDR Controller and a Physical_Link_Handle for an AMP Controller.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

RSSI LE: Range: -127 to 20, 127 (signed integer).
Units: dBm.

A.39 LE Read Advertising Channel TX Power

Synopsis: The LE_Read_Advertising_Channel_TX_Power command is used by the Host to read the transmit power level used for LE advertising channel packets.

Parameters:

request_id = CMD_LE_READ_ADVERTISING_CHANNEL_TX_POWER_REQ (79)
request_size = 0

Returns:

reply_id = CMD_LE_READ_ADVERTISING_CHANNEL_TX_POWER_RSP (80)
reply_size = 2

Status 0x00 – LE_Read_Advertising_Channel_TX_Power command succeeded.
0x01-0xFF – LE_Read_Advertising_Channel_TX_Power failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Transmit_Power_Level Size: 1 Octet (signed integer).
Range: -20 ≤ N ≤ 10.
Units: dBm.
Accuracy: +/- 4 dB.

A.40 LE Read Channel Map

Synopsis: The LE_Read_Channel_Map command returns the current Channel_Map for the specified Connection_Handle.

Parameters:

request_id = CMD_LE_READ_CHANNEL_MAP_REQ (81)
request_size = 2

Connection_Handle The Connection_Handle for the Connection for which the Channel_Map is to be read.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

Returns:

reply_id = CMD_LE_READ_CHANNEL_MAP_RSP (82)
reply_size = 8

Status 0x00 – LE_Read_Channel_Map command succeeded.
0x01-0xFF – LE_Read_Channel_Map command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Connection_Handle Connection handle to be used to identify a connection.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

Channel_Map This parameter contains 37 1-bit fields. The n'th such field (in the range 0 to 36) contains the value for the link layer channel index n.
Channel n is unused = 0.
Channel n is used = 1.
The most significant bits are reserved and shall be set to 0.

A.41 LE Encrypt

Synopsis: The LE_Encrypt command is used to request the Controller to encrypt the Plaintext_Data in the command using the Key given in the command and returns the Encrypted_Data to the Host.

Parameters:

request_id = CMD_LE_ENCRYPT_REQ (83)
request_size = 32

Key 128 bit key for the encryption of the data given in the command. The most significant octet of the key corresponds to key[0] using the notation specified in FIPS 197.

Plaintext_Data 128 bit data block that is requested to be encrypted. The most significant octet of the Plaintext_Data corresponds to in[0] using the notation specified in FIPS 197.

Returns:

reply_id = CMD_LE_ENCRYPT_RSP (84)
reply_size = 17

Status 0x00 – LE_Encrypt command succeeded.
0x01-0xFF – LE_Encrypt command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Encrypted_Data 128 bit encrypted data block. The most significant octet of the Encrypted_Data corresponds to out[0] using the notation specified in FIPS 197.

A.42 LE Long Term Key Request Reply

Synopsis: The LE_Long_Term_Key_Request_Reply command is used to reply to an LE Long Term Key Request event from the Controller, and specifies the Long_Term_Key parameter that shall be used for this Connection_Handle. The Long_Term_Key is used as defined in [Vol 6] Part B, Section 5.1.3.

Parameters:

request_id = CMD_LE_LONG_TERM_KEY_REQUEST_REPLY_REQ (85)
request_size = 18

Connection_Handle Connection handle to be used to identify a connection.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

Long_Term_Key 128 bit long term key for the given connection.

Returns:

reply_id = CMD_LE_LONG_TERM_KEY_REQUEST_REPLY_RSP (86)
reply_size = 3

Status 0x00 – LE_Long_Term_Key_Request_Reply command succeeded.
0x01-0xFF – LE_Long_Term_Key_Request_Reply command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Connection_Handle Connection handle to be used to identify a connection.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

A.43 LE Long Term Key Request Negative Reply

Synopsis: The LE_Long_Term_Key_Request_Negative_Reply command is used to reply to an LE Long Term Key Request event from the Controller if the Host cannot provide a Long Term Key for this Connection_Handle.

Parameters:

request_id = CMD_LE_LONG_TERM_KEY_REQUEST_NEGATIVE_REPLY_REQ (87)
request_size = 2

Connection_Handle Connection handle to be used to identify a connection.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

Returns:

reply_id = CMD_LE_LONG_TERM_KEY_REQUEST_NEGATIVE_REPLY_RSP (88)
reply_size = 3

Status 0x00 – LE_Long_Term_Key_Request_Negative_Reply command succeeded.
0x01-0xFF – LE_Long_Term_Key_Request_Negative_Reply command failed.
See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Connection_Handle Connection handle to be used to identify a connection.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

A.44 LE Rand

Synopsis: The LE_Rand command is used to request the Controller to generate 8 octets of random data to be sent to the Host. The Random_Number shall be generated according to [Vol 2] Part H, Section 2 if the LE Feature (LE Encryption) is supported.

Parameters:

request_id = CMD_LE_RAND_REQ (89)
request_size = 0

Returns:

reply_id = CMD_LE_RAND_RSP (90)
reply_size = 9

Status 0x00 – LE_Rand command succeeded.
0x01-0xFF – LE_Rand command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Random_Number Random Number.

A.45 LE Start Encryption

Synopsis: The LE_Start_Encryption command is used to authenticate the given encryption key associated with the remote device specified by the Connection_Handle, and once authenticated will encrypt the connection.

Parameters:

request_id = CMD_LE_START_ENCRYPTION_REQ (91)
request_size = 28

Connection_Handle Connection handle to be used to identify a connection.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

Random_Number 64 bit random number.

Encrypted_Diversifier 16 bit encrypted diversifier.

Long_Term_Key 128 bit long term key.

Returns:

reply_id = CMD_LE_START_ENCRYPTION_RSP (92)
reply_size = 4

Status 0x00 – Command currently in pending.
0x01-0xFF – Command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Num_HCI_Command_Packets The Number of HCI command packets which are allowed to be sent to the Controller from the Host.

Command_Opcode Opcode of the command which caused this event and is pending completion.

A.46 Write Authenticated Payload Timeout

Synopsis: This command writes the Authenticated_Payload_Timeout (authenticatedPayloadTO, see [Vol 2] Part B, Section Appendix B and [Vol 6] Part B, Section 5.4 for the LE connection) parameter in the Primary Controller for the specified Connection_Handle.

Parameters:

request_id = CMD_WRITE_AUTHENTICATED_PAYLOAD_TIMEOUT_REQ (93)
request_size = 4

Connection_Handle Connection handle used to identify a connection.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

Authenticated_Payload_Timeout Maximum amount of time specified between packets authenticated by a valid MIC.
Range: 0x0001 to 0xFFFF.
Time = N * 10 ms.
Time Range: 10 ms to 655,350 ms.

Returns:

reply_id = CMD_WRITE_AUTHENTICATED_PAYLOAD_TIMEOUT_RSP (94)
reply_size = 3

Status 0x00 – Write_Authenticated_Payload_Timeout command succeeded.
0x01-0xFF – Write_Authenticated_Payload_Timeout command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Connection_Handle Connection handle used to identify a connection.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

A.47 Read Authenticated Payload Timeout

Synopsis: This command reads the Authenticated_Payload_Timeout (authenticatedPayloadTO, see [Vol 2] Part B, Section Appendix B for BR/EDR connections and [Vol 6] Part B, Section 5.4 for LE connections) parameter in the Primary Controller on the specified Connection_Handle.

Parameters:

request_id = CMD_READ_AUTHENTICATED_PAYLOAD_TIMEOUT_REQ (95)
request_size = 2

Connection_Handle Connection handle used to identify a connection.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

Returns:

reply_id = CMD_READ_AUTHENTICATED_PAYLOAD_TIMEOUT_RSP (96)
reply_size = 5

Status 0x00 – Read_Authenticated_Payload_Timeout command succeeded.
0x01-0xFF – Read_Authenticated_Payload_Timeout command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Connection_Handle Connection handle used to identify a connection.
Range: 0x0000-0x0EFF.
0x0F00 - 0x0FFF Reserved for future use.

Authenticated_Payload_Timeout Maximum amount of time specified between packets authenticated by a MIC.
Default = 0x0BB8 (30 seconds).
Range: 0x0001 to 0xFFFF.
Time = N * 10 ms.
Time Range: 10 ms to 655,350 ms.

A.48 LE Receiver Test

Synopsis: This command is used to start a test where the DUT receives test reference packets at a fixed interval. The tester generates the test reference packets.

Parameters:

request_id = CMD_LE_RECEIVER_TEST_REQ (97)
request_size = 1

RX_Channel N = (F – 2402) / 2.
Range: 0x00 – 0x27.
Frequency Range: 2402 MHz to 2480 MHz.

Returns:

reply_id = CMD_LE_RECEIVER_TEST_RSP (98)
reply_size = 1

Status 0x00 – LE_Receiver_Test command succeeded.
0x01-0xFF – LE_Receiver_Test command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.49 LE Transmitter Test

Synopsis: This command is used to start a test where the DUT generates test reference packets at a fixed interval. The Controller shall transmit at maximum power.

Parameters:

request_id = CMD_LE_TRANSMITTER_TEST_REQ (99)
request_size = 3

RX_Channel N = (F – 2402) / 2.
Range: 0x00 – 0x27.
Frequency Range: 2402 MHz to 2480 MHz.

Length_Of_Test_Data 0x00-0x25 – Length in bytes of payload data in each packet.
0x26-0xFF – Reserved for future use.

Packet_Payload 0x00 – Pseudo-Random bit sequence 9.
0x01 – Pattern of alternating bits '11110000'.
0x02 – Pattern of alternating bits '10101010'.
0x03 – Pseudo-Random bit sequence 15.
0x04 – Pattern of All '1' bits.
0x05 – Pattern of All '0' bits.
0x06 – Pattern of alternating bits '00001111'.
0x07 – Pattern of alternating bits '0101'.
0x08-0xFF – Reserved for future use.

Returns:

reply_id = CMD_LE_TRANSMITTER_TEST_RSP (100)
reply_size = 1

Status 0x00 – LE_Transmitter_Test command succeeded. 0x01-0xFF – LE_Transmitter_Test command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.50 LE Test End

Synopsis: This command is used to stop any test which is in progress. The Number_Of_Packets for a transmitter test shall be reported as 0x0000. The Number_Of_Packets is an unsigned number and contains the number of received packets.

Parameters:

request_id = CMD_LE_TEST_END_REQ (101)
request_size = 0

Returns:

reply_id = CMD_LE_TEST_END_RSP (102)
reply_size = 3

Status 0x00 – LE_Test_End command succeeded.
0x01-0xFF – LE_Test_End command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Number_Of_Packets Number of packets received.

A.51 Write LE Host Support

Synopsis: The Write_LE_Host_Support command is used to set the LE Supported (Host) and Simultaneous LE and BR/EDR to Same Device Capable (Host) Link Manager Protocol feature bits. These Link Manager Protocol feature bits are used by a remote Host. See [Vol 2] Part C, Section 3.2.

Parameters:

request_id = CMD_WRITE_LE_HOST_SUPPORT_REQ (103)
request_size = 2

LE_Supported_Host Supported host parameter. See Section 6.34.

Simultaneous_LE_Host Simultaneous host parameter. See Section 6.35.

Returns:

reply_id = CMD_WRITE_LE_HOST_SUPPORT_RSP (104)
reply_size = 1

Status 0x00 – Write_LE_Host_Support command succeeded.
0x01-0xFF – Write_LE_Host_Support command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.52 LE Set Data Length

Synopsis: The LE_Set_Data_Length command allows the Host to suggest maximum transmission packet size and maximum packet transmission time (connMaxTxOctets and connMaxTxTime - see [Vol 6] Part B, Section 4.5.10) to be used for a given connection. The Controller may use smaller or larger values based on local information.

Parameters:

request_id = CMD_LE_SET_DATA_LENGTH_REQ (105)
request_size = 6

Connection_Handle Connection handle.
Range 0x0000-0x0EFF.
All other values reserved for future use.

TxOctets Preferred maximum number of payload octets that the local Controller should include in a single Link Layer packet on this connection.
Range 0x001B-0x00FB.
All other values reserved for future use.

TxTime Preferred maximum number of microseconds that the local Controller should use to transmit a single Link Layer packet on this connection.
Range 0x0148-0x4290.
All other values reserved for future use.

Returns:

reply_id = CMD_LE_SET_DATA_LENGTH_RSP (106)
reply_size = 1

Status 0x00 – LE_Set_Data_Length command succeeded.
0x01-0xFF – LE_Set_Data_Length command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.53 LE Read Suggested Default Data Length

Synopsis: The LE_Read_Suggested_Default_Data_Length command allows the Host to read the Host's suggested values (SuggestedMaxTxOctets and SuggestedMaxTxTime) for the Controller's maximum transmitted number of payload octets and maximum packet transmission time to be used for new connections (see [Vol 6] Part B, Section 4.5.10).

Parameters:

request_id = CMD_LE_READ_SUGGESTED_DEFAULT_DATA_LENGTH_REQ (107)
request_size = 0

Returns:

reply_id = CMD_LE_READ_SUGGESTED_DEFAULT_DATA_LENGTH_RSP (108)
reply_size = 5

Status 0x00 – LE_Read_Suggested_Default_Data_Length command succeeded.
0x01-0xFF – LE_Read_Suggested_Default_Data_Length command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

SuggestedMaxTxOctets Host's suggested value for the Controller's maximum transmitted number of payload octets to be used for new connections.
Range 0x001B-0x00FB.
All other values reserved for future use.
Default: 0x001B.

SuggestedMaxTxTime Host's suggested value for the Controller's maximum packet transmission time to be used for new connections.
Range 0x0148-0x4290 (all other values reserved for future use).
Default: 0x0148.

A.54 LE Write Suggested Default Data Length

Synopsis: The LE_Write_Suggested_Default_Data_Length command allows the Host to specify its suggested values for the Controller's maximum transmission number of payload octets and maximum packet transmission time to be used for new connections. The Controller may use smaller or larger values for connInitialMaxTxOctets and connInitialMaxTxTime based on local information (see [Vol 6] Part B, Section 4.5.10).

Parameters:

request_id = CMD_LE_WRITE_SUGGESTED_DEFAULT_DATA_LENGTH_REQ (109)
request_size = 4

SuggestedMaxTxOctets Host's suggested value for the Controller's maximum transmitted number of payload octets to be used for new connections.
Range 0x001B-0x00FB.
All other values reserved for future use.
Default: 0x001B.

SuggestedMaxTxTime Host's suggested value for the Controller's maximum packet transmission time to be used for new connections.
Range 0x0148-0x4290 (all other values reserved for future use).

Returns:

reply_id = CMD_LE_WRITE_SUGGESTED_DEFAULT_DATA_LENGTH_RSP (110)
reply_size = 1

Status 0x00 – LE_Write_Suggested_Default_Data_Length command succeeded.
0x01-0xFF – LE_Write_Suggested_Default_Data_Length command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.55 LE Read Maximum Data Length

Synopsis: The LE_Read_Maximum_Data_Length command allows the Host to read the Controller's maximum supported payload octets and packet duration times for transmission and reception (supportedMaxTxOctets and supportedMaxTxTime, supportedMaxRxOctets, and supportedMaxRxTime, see [Vol 6] Part B, Section 4.5.10).

Parameters:

request_id = CMD_LE_READ_MAXIMUM_DATA_LENGTH_REQ (111)
request_size = 0

Returns:

reply_id = CMD_LE_READ_MAXIMUM_DATA_LENGTH_RSP (112)
reply_size = 9

Status 0x00 – LE_Read_Maximum_Data_Length command succeeded.
0x01-0xFF – LE_Read_Maximum_Data_Length command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

supportedMaxTxOctets Maximum number of payload octets that the local Controller supports for transmission of a single Link Layer packet on a data connection.
Range 0x001B-0x00FB.
All other values reserved for future use.

supportedMaxTxTime Maximum time, in microseconds, that the local Controller supports for transmission of a single Link Layer packet on a data connection.
Range 0x0148-0x4290.
All other values reserved for future use.

supportedMaxRxOctets Maximum number of payload octets that the local Controller supports for reception of a single Link Layer packet on a data connection.
Range 0x001B-0x00FB.
All other values reserved for future use.

supportedMaxRxTime Maximum time, in microseconds, that the local Controller supports for reception of a single Link Layer packet on a data connection.
Range 0x0148-0x4290.
All other values reserved for future use.

A.56 LE Set Resolvable Private Address Timeout

Synopsis: The LE_Set_Resolvable_Private_Address_Timeout command set the length of time the Controller uses a Resolvable Private Address before a new resolvable private address is generated and starts being used.

Parameters:

request_id = CMD_LE_SET_RESOLVABLE_PRIVATE_ADDRESS_TIMEOUT_REQ (113)
request_size = 2

RPA_Timeout RPA timeout measured in seconds.
Range for N: 0x0001 – 0xA1B8 (1 second – approximately 11.5 hours).
Default: N= 0x0384 (900 s or 15 minutes).

Returns:

reply_id = CMD_LE_SET_RESOLVABLE_PRIVATE_ADDRESS_TIMEOUT_RSP (114)
reply_size = 1

Status 0x00 – LE_Set_Resolvable_Private_Address_Timeout command succeeded.
0x01-0xFF – LE_Set_Resolvable_Private_Address_Timeout command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.57 LE Set Address Resolution Enable

Synopsis: The LE_Set_Address_Resolution_Enable command is used to enable resolution of Resolvable Private Addresses in the Controller. This causes the Controller to use the resolving list whenever the Controller receives a local or peer Resolvable Private Address.

Parameters:

request_id = CMD_LE_SET_ADDRESS_RESOLUTION_ENABLE_REQ (115)
request_size = 1

Address_Resolution_Enable 0x00 – Address Resolution in Controller disabled (default).
0x01 – Address Resolution in Controller enabled.
0x02 – 0xFF – Reserved for future use.

Returns:

reply_id = CMD_LE_SET_ADDRESS_RESOLUTION_ENABLE_RSP (116)
reply_size = 1

Status 0x00 – LE_Set_Address_Resolution_Enable command succeeded.
0x01-0xFF – LE_Set_Address_Resolution_Enable command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.58 LE Add Device To Resolving List

Synopsis: The LE_Add_Device_To_Resolving_List command is used to add one device to the list of address translations used to resolve Resolvable Private Addresses in the Controller.

Parameters:

request_id = CMD_LE_ADD_DEVICE_TO_RESOLVING_LIST_REQ (117)
request_size = 39

Peer_Identity_Address_Type 0x00 – Public Identity Address.
0x01 – Random (static) Identity Address.
0x02 – 0xFF – Reserved for future use.

Peer_Identity_Address Public or Random (static) Identity address of the peer device.

Peer_IRK IRK of the peer device.

Local_IRK IRK of the local device.

Returns:

reply_id = CMD_LE_ADD_DEVICE_TO_RESOLVING_LIST_RSP (118)
reply_size = 1

Status 0x00 – LE_Add_Device_To_Resolving_List command succeeded.
0x01-0xFF – LE_Add_Device_To_Resolving_List command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.59 LE Remove Device From Resolving List

Synopsis: The LE_Remove_Device_From_Resolving_List command is used to remove one device from the list of address translations used to resolve Resolvable Private Addresses in the Controller.

Parameters:

request_id = CMD_LE_REMOVE_DEVICE_FROM_RESOLVING_LIST_REQ (119)
request_size = 7

Peer_Identity_Address_Type 0x00 – Public Identity Address.
0x01 – Random (static) Identity Address.
0x02 – 0xFF – Reserved for future use.

Peer_Device_Address Public or Random (static) Identity Address of the peer device.

Returns:

reply_id = CMD_LE_REMOVE_DEVICE_FROM_RESOLVING_LIST_RSP (120)
reply_size = 1

Status 0x00 – LE_Remove_Device_From_Resolving_List command succeeded.
0x01-0xFF – LE_Remove_Device_From_Resolving_List command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.60 LE Clear Resolving List

Synopsis: The LE_Clear_Resolving_List command is used to remove all devices from the list of address translations used to resolve Resolvable Private Addresses in the Controller.

Parameters:

request_id = CMD_LE_CLEAR_RESOLVING_LIST_REQ (121)
request_size = 0

Returns:

reply_id = CMD_LE_CLEAR_RESOLVING_LIST_RSP (122)
reply_size = 1

Status 0x00 – LE_Clear_Resolving_List command succeeded.
0x01-0xFF – LE_Clear_Resolving_List command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.61 LE Read Resolving List Size

Synopsis: The LE_Read_Resolving_List_Size command is used to read the total number of address translation entries in the resolving list that can be stored in the Controller. Note: The number of entries that can be stored is not fixed and the Controller can change it at any time (e.g. because the memory used to store the list can also be used for other purposes).

Parameters:

request_id = CMD_LE_READ_RESOLVING_LIST_SIZE_REQ (123)
request_size = 0

Returns:

reply_id = CMD_LE_READ_RESOLVING_LIST_SIZE_RSP (124)
reply_size = 2

Status 0x00 – LE_Read_Resolving_List_Size command succeeded.
0x01-0xFF – LE_Read_Resolving_List_Size command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Resolving_List_Size Number of address translation entries in the resolving list.

A.62 LE Set Privacy Mode

Synopsis: The LE_Set_Privacy_Mode command is used to allow the Host to specify the privacy mode to be used for a given entry on the resolving list. The effect of this setting is specified in [Vol 6] Part B, Section 4.7.

Parameters:

request_id = CMD_LE_SET_PRIVACY_MODE_REQ (125)
request_size = 8

Peer_Identity_Address_Type 0x00 – Public Identity Address.
0x01 – Random (static) Identity Address.
All other values reserved for future use.

Peer Identity_Address Public Identity Address or Random (static) Identity Address of the advertiser.

Privacy_Mode 0x00 – Use Network Privacy Mode for this peer device (default).
0x01 – Use Device Privacy Mode for this peer device.
All other values reserved for future use.

Returns:

reply_id = CMD_LE_SET_PRIVACY_MODE_RSP (126)
reply_size = 1

Status 0x00 – LE_Set_Privacy_Mode command succeeded.
0x01-0xFF – LE_Set_Privacy_Mode command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.63 LE Read Peer Resolvable Address

Synopsis: The LE_Read_Peer_Resolvable_Address command is used to get the current peer Resolvable Private Address being used for the corresponding peer Public and Random (static) Identity Address. The peer's resolvable address being used may change after the command is called.

Parameters:

request_id = CMD_LE_READ_PEER_RESOLVABLE_ADDRESS_REQ (127)
request_size = 7

Peer_Identity_Address_Type 0x00 – Public Identity Address.
0x01 – Random (static) Identity Address.
0x02 – 0xFF – Reserved for future use.

Peer_Identity_Address Public or Random (static) Identity Address of the peer device.

Returns:

reply_id = CMD_LE_READ_PEER_RESOLVABLE_ADDRESS_RSP (128)
reply_size = 7

Status 0x00 – LE_Read_Peer_Resolvable_Address command succeeded.
0x01-0xFF – LE_Read_Peer_Resolvable_Address command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Peer_Resolvable_Address Resolvable Private Address being used by the peer device.

A.64 LE Read Local Resolvable Address

Synopsis: The LE_Read_Local_Resolvable_Address command is used to get the current local Resolvable Private Address being used for the corresponding peer Identity Address. The local's resolvable address being used may change after the command is called.

Parameters:

request_id = CMD_LE_READ_LOCAL_RESOLVABLE_ADDRESS_REQ (129)
request_size = 7

Peer_Identity_Address_Type 0x00 Public Identity Address.
0x01 Random (static) Identity Address.
0x02 – 0xFF Reserved for future use.

Peer_Identity_Address Public Identity Address or Random (static) Identity Address of the peer device, 48 bit value.

Returns:

reply_id = CMD_LE_READ_LOCAL_RESOLVABLE_ADDRESS_RSP (130)
reply_size = 7

Status 0x00 – LE_Read_Local_Resolvable_Address command succeeded.
0x01-0xFF – LE_Read_Local_Resolvable_Address command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Peer_Resolvable_Address Resolvable Private Address being used by the local device.

A.65 LE Read PHY

Synopsis: The LE_Read_PHY command is used to read the current transmitter PHY and receiver PHY on the connection identified by the Connection_Handle.

Parameters:

request_id = CMD_LE_READ_PHY_REQ (131)
request_size = 2

Connection_Handle Connection handle.
Range: 0x0000-0x0EFF.
All other values reserved for future use.

Returns:

reply_id = CMD_LE_READ_PHY_RSP (132)
reply_size = 5

Status 0x00 – LE_Read_PHY command succeeded.
0x01-0xFF – LE_Read_PHY command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Connection_Handle Connection handle.
Range: 0x0000-0x0EFF.
All other values reserved for future use.

TX_PHY 0x01 – The transmitter PHY for the connection is LE 1M.
0x02 – The transmitter PHY for the connection is LE 2M.
0x03 – The transmitter PHY for the connection is LE Coded.
All other values reserved for future use.

RX_PHY 0x01 – The receiver PHY for the connection is LE 1M
0x02 – The receiver PHY for the connection is LE 2M
0x03 – The receiver PHY for the connection is LE Coded
All other values reserved for future use

A.66 LE Set Default PHY

Synopsis: The LE_Set_Default_PHY command allows the Host to specify its preferred values for the transmitter PHY and receiver PHY to be used for all subsequent connections over the LE transport.

Parameters:

request_id = CMD_LE_SET_DEFAULT_PHY_REQ (133)
request_size = 3

ALL_PHYS Bit 0 – The Host has no preference among the transmitter PHYs supported by the Controller.
Bit 1 – The Host has no preference among the receiver PHYs supported by the Controller.
Bit 2-7 – Reserved for future use.

TX_PHYS Bit 0 – The Host prefers to use the LE 1M transmitter PHY (possibly among others).
Bit 1 – The Host prefers to use the LE 2M transmitter PHY (possibly among others).
Bit 2 – The Host prefers to use the LE Coded transmitter PHY (possibly among others).
Bit 3 – 7 – Reserved for future use.

RX_PHYS Bit 0 – The Host prefers to use the LE 1M receiver PHY (possibly among others).
Bit 1 – The Host prefers to use the LE 2M receiver PHY (possibly among others).
Bit 2 – The Host prefers to use the LE Coded receiver PHY (possibly among others).
Bit 3 – 7 – Reserved for future use.

Returns:

reply_id = CMD_LE_SET_DEFAULT_PHY_RSP (134)
reply_size = 1

Status 0x00 – LE_Set_Default_PHY command succeeded.
0x01-0xFF – LE_Set_Default_PHY command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.67 LE Set PHY

Synopsis: The LE_Set_PHY command is used to set the PHY preferences for the connection identified by the Connection_Handle. The Controller might not be able to make the change (e.g. because the peer does not support the requested PHY) or may decide that the current PHY is preferable.

Parameters:

request_id = CMD_LE_SET_PHY_REQ (135)
request_size = 7

Connection_Handle Connection handle.
Range: 0x0000-0x0EFF.
All other values reserved for future use.

ALL_PHYS Bit 0 – The Host has no preference among the transmitter PHYs supported by the Controller.
Bit 1 – The Host has no preference among the receiver PHYs supported by the Controller.
Bit 2 – 7 – Reserved for future use.

TX_PHYS Bit 0 – The Host prefers to use the LE 1M transmitter PHY (possibly among others).
Bit 1 – The Host prefers to use the LE 2M transmitter PHY (possibly among others).
Bit 2 – The Host prefers to use the LE Coded transmitter PHY (possibly among others).
Bit 3 – 7 – Reserved for future use.

RX_PHYS Bit 0 – The Host prefers to use the LE 1M receiver PHY (possibly among others).
Bit 1 – The Host prefers to use the LE 2M receiver PHY (possibly among others).
Bit 2 – The Host prefers to use the LE Coded receiver PHY (possibly among others).
Bit 3 – 7 – Reserved for future use.

PHY_options Bit 0 – 1: 0 = the Host has no preferred coding when transmitting on the LE Coded PHY.
1 = the Host prefers that S=2 coding be used when transmitting on the LE Coded PHY.
2 = the Host prefers that S=8 coding be used when transmitting on the LE Coded PHY.
3 = Reserved for future use.
Bit 2 – 15: Reserved for future use.

Returns:

reply_id = CMD_LE_SET_PHY_RSP (136)
reply_size = 4

Status 0x00 – Command currently in pending.
0x01-0xFF – Command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Num_HCI_Command_Packets The Number of HCI command packets which are allowed to be sent to the Controller from the Host.

Command_Opcode Opcode of the command which caused this event and is pending completion.

A.68 LE Enhanced Transmitter Test

Synopsis: This command is used to start a test where the DUT generates test reference packets at a fixed interval. The Controller shall transmit at maximum power.

Parameters:

request_id = CMD_LE_ENHANCED_TRANSMITTER_TEST_REQ (137)
request_size = 4

TX_Channel N = (F-2402) / 2.
Range: 0x00 – 0x27.
Frequency Range: 2402 MHz to 2480 MHz.

Length_Of_Test_Data Length in bytes of payload data in each packet.

Packet_Payload 0x00 – PRBS9 sequence '11111111100000111101…' (in transmission order) as described in [Vol 6] Part F, Section 4.1.5.
0x01 – Repeated '11110000' (in transmission order) sequence as described in [Vol 6] Part F, Section 4.1.5.
0x02 – Repeated '10101010' (in transmission order) sequence as described in [Vol 6] Part F, Section 4.1.5.
0x03 – PRBS15 sequence as described in [Vol 6] Part F, Section 4.1.5.
0x04 – Repeated '11111111' (in transmission order) sequence.
0x05 – Repeated '00000000' (in transmission order) sequence.
0x06 – Repeated '00001111' (in transmission order) sequence.
0x07 – Repeated '01010101' (in transmission order) sequence.
0x08 – 0xFF – Reserved for future use.

PHY 0x00 – Reserved for future use.
0x01 – Transmitter set to use the LE 1M PHY.
0x02 – Transmitter set to use the LE 2M PHY.
0x03 – Transmitter set to use the LE Coded PHY with S=8 data coding.
0x04 – Transmitter set to use the LE Coded PHY with S=2 data coding.
0x05 – 0xFF – Reserved for future use.

Returns:

reply_id = CMD_LE_ENHANCED_TRANSMITTER_TEST_RSP (138)
reply_size = 1

Status 0x00 – LE_Enhanced_Transmitter_Test command succeeded.
0x01-0xFF – LE_Enhanced_Transmitter_Test command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.69 LE Enhanced Receiver Test

Synopsis: This command is used to start a test where the DUT receives test reference packets at a fixed interval. The tester generates the test reference packets.

Parameters:

request_id = CMD_LE_ENHANCED_RECEIVER_TEST_REQ (139)
request_size = 3

RX_Channel N = (F-2402) / 2.
Range: 0x00 – 0x27.
Frequency Range: 2402 MHz to 2480 MHz.

PHY 0x00 – Reserved for future use.
0x01 – Receiver set to use the LE 1M PHY.
0x02 – Receiver set to use the LE 2M PHY.
0x03 – Receiver set to use the LE Coded PHY.
0x04 – 0xFF – Reserved for future use.

Modulation_Index 0x00 – Assume transmitter will have a standard modulation index.
0x01 – Assume transmitter will have a stable modulation index.
0x02 – 0xFF – Reserved for future use.

Returns:

reply_id = CMD_LE_ENHANCED_RECEIVER_TEST_RSP (140)
reply_size = 1

Status 0x00 –LE_Enhanced_Receiver_Test command succeeded.
0x01-0xFF – LE_Enhanced_Receiver_Test command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.70 LE Set Extended Advertising Parameters

Synopsis: The LE_Set_Extended_Advertising_Parameters command is used by the Host to set the advertising parameters.

Parameters:

request_id = CMD_LE_SET_EXTENDED_ADVERTISING_PARAMETERS_REQ (141)
request_size = 25

Advertising_Handle 0x00 – 0xEF – Used to identify an advertising set.
All other values reserved for future use.

Advertising_Event_Properties 0 – Connectable advertising.
1 – Scannable advertising.
2 – Directed advertising.
3 – High Duty Cycle Directed Connectable advertising (≤ 3.75 ms Advertising Interval).
4 – Use legacy advertising PDUs.
5 – Omit advertiser's address from all PDUs ("anonymous advertising").
6 – Include TxPower in the extended header of the advertising PDU.
All other bits reserved for future use.

Primary_Advertising_Interval_Min Minimum advertising interval for undirected and low duty cycle directed advertising.
Range: 0x000020 to 0xFFFFFF.
Time = N * 0.625 ms.
Time Range: 20 ms to 10,485.759375 seconds.

Primary_Advertising_Interval_Max Maximum advertising interval for undirected and low duty cycle directed advertising.
Range: 0x000020 to 0xFFFFFF.
Time = N * 0.625 ms.
Time Range: 20 ms to 10,485.759375 seconds.

Primary_Advertising_Channel_Map 0 – Channel 37 shall be used.
1 – Channel 38 shall be used.
2 – Channel 39 shall be used.
All other bits reserved for future use.

Own_Address_Type 0x00 – Public Device Address.
0x01 – Random Device Address.
0x02 – Controller generates the Resolvable Private Address based on the local IRK from the resolving list.
If the resolving list contains no matching entry, use the public address.
0x03 – Controller generates the Resolvable Private Address based on the local IRK from the resolving list.
If the resolving list contains no matching entry, use the random address from LE_Set_Advertising_Set_Random_Address.
All other values reserved for future use.

Peer_Address_Type 0x00 – Public Device Address or Public Identity Address.
0x01 – Random Device Address or Random (static) Identity Address.
All other values reserved for future use.

Peer_Address Public Device Address, Random Device Address, Public Identity Address, or Random (static) Identity Address of the device to be connected.

Advertising_Filter_Policy 0x00 – Process scan and connection requests from all devices (i.e., the White List is not in use).
0x01 – Process connection requests from all devices and only scan requests from devices that are in the White List.
0x02 – Process scan requests from all devices and only connection requests from devices that are in the White List.
0x03 – Process scan and connection requests only from devices in the White List.
All other values reserved for future use.

Advertising_Tx_Power N = 0xXX Size: 1 Octet (signed integer).
Range: -127 ≤ N ≤ +126.
Units: dBm.
127 Host has no preference.

Primary_Advertising_PHY 0x01 – Primary advertisement PHY is LE 1M.
0x03 – Primary advertisement PHY is LE Coded.
All other values reserved for future use.

Secondary_Advertising_Max_Skip 0x00 – AUX_ADV_IND shall be sent prior to the next advertising event.
0x01-0xFF – Maximum advertising events the Controller can skip before sending the AUX_ADV_IND packets on the secondary advertising channel.

Secondary_Advertising_PHY 0x01 – Secondary advertisement PHY is LE 1M.
0x02 – Secondary advertisement PHY is LE 2M.
0x03 – Secondary advertisement PHY is LE Coded.
All other values reserved for future use.

Advertising_SID 0x00 – 0x0F – Value of the Advertising SID subfield in the ADI field of the PDU.
All other values reserved for future use.

Scan_Request_Notification_Enable 0x00 – Scan request notifications disabled.
0x01 – Scan request notifications enabled.
All other values reserved for future use.

Returns:

reply_id = CMD_LE_SET_EXTENDED_ADVERTISING_PARAMETERS_RSP (142)
reply_size = 1

Status 0x00 – LE_Set_Extended_Advertising_Parameters command succeeded.
0x01-0xFF – LE_Set_Extended_Advertising_Parameters command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Selected_Tx_Power Size: 1 Octet (signed integer).
Range: -127 ≤ N ≤ +126.
Units: dBm.

A.71 LE Set Extended Advertising Data

Synopsis: The LE_Set_Extended_Advertising_Data command is used to set the data used in advertising PDUs that have a data field. This command may be issued at any time after an advertising set identified by the Advertising_Handle parameter has been created using the LE Set Extended Advertising Parameters Command (see Section 7.8.53), regardless of whether advertising in that set is enabled or disabled.

Parameters:

request_id = CMD_LE_SET_EXTENDED_ADVERTISING_DATA_REQ (143)
request_size = 255

Advertising_Handle 0x00 – 0xEF – Used to identify an advertising set.
All other values reserved for future use.

Operation 0x00 – Intermediate fragment of fragmented extended advertising data.
0x01 – First fragment of fragmented extended advertising data.
0x02 – Last fragment of fragmented extended advertising data.
0x03 – Complete extended advertising data.
0x04 – Unchanged data (just update the Advertising DID).
All other values reserved for future use.

Fragment_Preference 0x00 – The Controller may fragment all Host advertising data.
0x01 – The Controller should not fragment or should minimize fragmentation of Host advertising data.
All other values reserved for future use.

Advertising_Data_Length 0 – 251 Number of octets in the Advertising_Data parameter.
All other values reserved for future use.

Advertising_Data Advertising data formatted as defined in [Vol 3] Part C, Section 11.
Note: This parameter has a variable length.

Returns:

reply_id = CMD_LE_SET_EXTENDED_ADVERTISING_DATA_RSP (144)
reply_size = 1

Status 0x00 – LE_Set_Extended_Advertising_Data command succeeded. 0x01-0xFF – LE_Set_Extended_Advertising_Data command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.72 LE Set Extended Scan Response Data

Synopsis:

Parameters:

request_id = CMD_LE_SET_EXTENDED_SCAN_RESPONSE_DATA_REQ (145)
request_size = 255

Advertising_Handle 0x00 – 0xEF Used to identify an advertising set.
All other values reserved for future use.

Operation 0x00 – Intermediate fragment of fragmented scan response data.
0x01 – First fragment of fragmented scan response data.
0x02 – Last fragment of fragmented scan response data.
0x03 – Complete scan response data.
All other values reserved for future use.

Fragment_Preference 0x00 – The Controller may fragment all scan response data.
0x01 – The Controller should not fragment or should minimize fragmentation of scan response data.
All other values reserved for future use.

Scan_Response_Data_Length 0 – 251 Number of octets in the Scan_Response_Data parameter.
All other values reserved for future use.

Scan_Response_Data Scan response data formatted as defined in [Vol 3] Part C, Section 11.
Note: This parameter has a variable length.

Returns:

reply_id = CMD_LE_SET_EXTENDED_SCAN_RESPONSE_DATA_RSP (146)
reply_size = 1

Status 0x00 – LE_Set_Extended_Scan_Response_Data command succeeded.
0x01-0xFF – LE_Set_Extended_Scan_Response_Data command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.73 LE Set Extended Advertising Enable

Synopsis: The LE_Set_Extended_Advertising_Enable command is used to request the Controller to enable or disable one or more advertising sets using the advertising sets identified by the Advertising_Handle[i] parameter. The Controller manages the timing of advertisements in accordance with the advertising parameters given in the LE_Set_Extended_Advertising_Parameters command. The Number_of_Sets parameter is the number of advertising sets contained in the parameter arrays. If Enable and Number_of_Sets are both set to 0x00, then all advertising sets are disabled.

Parameters:

request_id = CMD_LE_SET_EXTENDED_ADVERTISING_ENABLE_REQ (147)
request_size = 2 + 4 * Number_of_Sets

Enable 0x00 – Advertising is disabled.
0x01 – Advertising is enabled.
All other values reserved for future use.

Number_of_Sets 0x00 – Disable all advertising sets.
0x01 – 0x3F – Number of advertising sets to enable or disable.
All other values reserved for future use.

Advertising_Handle[i] 0x00 – 0xEF – Used to identify an advertising set.
All other values reserved for future use.

Duration[i] 0x0000 No advertising duration. Advertising to continue until the Host disables it.
N = 0xXXXX Advertising duration.
Range: 0x0001 – 0xFFFF.

Time = N * 10 ms.
Time Range: 10 ms to 655,350 ms.

Max_Extended_Advertising_Events[i] N = 0xXX Maximum number of extended advertising events the Controller shall attempt to send prior to terminating the extended advertising.
0x00 No maximum number of advertising events.

Returns:

reply_id = CMD_LE_SET_EXTENDED_ADVERTISING_ENABLE_RSP (148)
reply_size = 1

Status 0x00 – LE_Set_Extended_Advertising_Enable command succeeded.
0x01-0xFF – LE_Set_Extended_Advertising_Enable command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.74 LE Read Maximum Advertising Data Length

Synopsis: The LE_Read_Maximum_Advertising_Data_Length command is used to read the maximum length of data supported by the Controller for use as advertisement data or scan response data in an advertising event or as periodic advertisement data.

Parameters:

request_id = CMD_LE_READ_MAXIMUM_ADVERTISING_DATA_LENGTH_REQ (149)
request_size = 0

Returns:

reply_id = CMD_LE_READ_MAXIMUM_ADVERTISING_DATA_LENGTH_RSP (150)
reply_size = 3

Status 0x00 – LE_Read_Maximum_Advertising_Data_Length command succeeded.
0x01-0xFF – LE_Read_Maximum_Advertising_Data_Length command failed.
See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Maximum_Advertising_Data_Length 0x001F – 0x0672 Maximum supported advertising data length.
All other values reserved for future use.

A.75 LE Read Number of Supported Advertising Sets

Synopsis: The LE_Read_Number_of_Supported_Advertising_Sets command is used to read the maximum number of advertising sets supported by the advertising Controller at the same time. Note: The number of advertising sets that can be supported is not fixed and the Controller can change it at any time because the memory used to store advertising sets can also be used for other purposes.

Parameters:

request_id =
CMD_LE_READ_NUMBER_OF_SUPPORTED_ADVERTISING_SETS_REQ (151)
request_size = 0

Returns:

reply_id = CMD_LE_READ_NUMBER_OF_SUPPORTED_ADVERTISING_SETS_RSP (152)
reply_size = 2

Status 0x00 – LE_Read_Number_of_Supported_Advertising_Sets command succeeded.
0x01-0xFF – LE_Read_Number_of_Supported_Advertising_Sets command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Num_Supported_Advertising_Sets 0x01 – 0xF0 – Number of advertising sets supported at the same time.
All other values reserved for future use.

A.76 LE Remove Advertising Set

Synopsis: The LE_Remove_Advertising_Set command is used to remove an advertising set from the Controller.

Parameters:

request_id = CMD_LE_REMOVE_ADVERTISING_SET_REQ (153)
request_size = 1

Advertising_Handle 0x00 – 0xEF – Used to identify an advertising set.
All other values reserved for future use.

Returns:

reply_id = CMD_LE_REMOVE_ADVERTISING_SET_RSP (154)
reply_size = 1

Status 0x00 – LE_Remove_Advertising_Set command succeeded.
0x01-0xFF – LE_Remove_Advertising_Set command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.77 LE Clear Advertising Sets

Synopsis: The LE_Clear_Advertising_Sets command is used to remove all existing advertising sets from the Controller.

Parameters:

request_id = CMD_LE_CLEAR_ADVERTISING_SETS_REQ (155)
request_size = 0

Returns:

reply_id = CMD_LE_CLEAR_ADVERTISING_SETS_RSP (156)
reply_size = 1

Status 0x00 – LE_Clear_Advertising_Sets command succeeded.
0x01-0xFF – LE_Clear_Advertising_Sets command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.78 LE Set Periodic Advertising Parameters

Synopsis: The LE_Set_Periodic_Advertising_Parameters command is used by the Host to set the parameters for periodic advertising.

Parameters:

request_id = CMD_LE_SET_PERIODIC_ADVERTISING_PARAMETERS_REQ (157)
request_size = 7

Advertising_Handle 0x00 – 0xEF – Used to identify a periodic advertisement.
All other values reserved for future use.

Periodic_Advertising_Interval_Min Minimum advertising interval for periodic advertising.
Range: 0x0006 to 0xFFFF.
Time = N * 1.25 ms.
Time Range: 7.5ms to 81.91875 seconds.

Periodic_Advertising_Interval_Max Maximum advertising interval for periodic advertising.
Range: 0x0006 to 0xFFFF.
Time = N * 1.25 ms.
Time Range: 7.5ms to 81.91875 seconds.

Periodic_Advertising_Properties 6 Include TxPower in the advertising PDU.
All other bits reserved for future use.

Returns:

reply_id = CMD_LE_SET_PERIODIC_ADVERTISING_PARAMETERS_RSP (158)
reply_size = 1

Status 0x00 – LE_Set_Periodic_Advertising_Parameters command succeeded.
0x01-0xFF – LE_Set_Periodic_Advertising_Parameters command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.79 LE Set Periodic Advertising Data

Synopsis: The LE_Set_Periodic_Advertising_Data command is used to set the data used in periodic advertising PDUs. This command may be issued at any time after the advertising set identified by the Advertising_Handle parameter has been configured for periodic advertising using the LE_Set_Periodic_Advertising_Parameters Command (see Section 7.8.61), regardless of whether advertising in that set is enabled or disabled. If the advertising set has not been configured for periodic advertising, then the Controller shall return the error code Command Disallowed (0x0C).

Parameters:

request_id = CMD_LE_SET_PERIODIC_ADVERTISING_DATA_REQ (159)
request_size = 254

Advertising_Handle 0x00 – 0xEF – Used to identify an advertising set.
All other values reserved for future use.

Operation 0x00 – Intermediate fragment of fragmented periodic advertising data.
0x01 – First fragment of fragmented periodic advertising data.
0x02 – Last fragment of fragmented periodic advertising data.

0x03 – Complete periodic advertising data.
All other values reserved for future use.

Advertising_Data_Length 0 – 252 – The number of octets in the Advertising Data parameter.
All other values reserved for future use.

Advertising_Data Periodic advertising data formatted as defined in [Vol 3] Part C, Section 11.
Note: This parameter has a variable length.

Returns:

reply_id = CMD_LE_SET_PERIODIC_ADVERTISING_DATA_RSP (160)
reply_size = 1

Status 0x00 – LE_Set_Periodic_Advertising_Data command succeeded.
0x01-0xFF – LE_Set_Periodic_Advertising_Data command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.80 LE Set Periodic Advertising Enable

Synopsis: The LE_Set_Periodic_Advertising_Enable command is used to request the Controller to enable or disable the periodic advertising for the advertising set specified by the Advertising_Handle parameter (ordinary advertising is not affected).

Parameters:

request_id = CMD_LE_SET_PERIODIC_ADVERTISING_ENABLE_REQ (161)
request_size = 2

Enable 0x00 – Periodic advertising is disabled (default).
0x01 – Periodic advertising is enabled.
All other values reserved for future use.

Advertising_Handle 0x00 – 0xEF – Used to identify an advertising set.
All other values reserved for future use.

Returns:

reply_id = CMD_LE_SET_PERIODIC_ADVERTISING_ENABLE_RSP (162)
reply_size = 1

Status 0x00 – LE_Set_Periodic_Advertising_Enable command succeeded.
0x01-0xFF – LE_Set_Periodic_Advertising_Enable command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.81 LE Set Extended Scan Parameters

Synopsis: The LE_Set_Extended_Scan_Parameters command is used to set the extended scan parameters to be used on the advertising channels.

Parameters:

request_id = CMD_LE_SET_EXTENDED_SCAN_PARAMETERS_REQ (163)
request_size = 3 + 5 * number of bits set in Scanning_PHYs

Own_Address_Type 0x00 – Public Device Address.
0x01 – Random Device Address.
0x02 – Controller generates the Resolvable Private Address based on the local

IRK from the resolving list.
If the resolving list contains no matching entry, then use the public address.
0x03 – Controller generates the Resolvable Private Address based on the local IRK from the resolving list.
If the resolving list contains no matching entry, then use the random address from LE_Set_Random_Address.
All other values reserved for future use.

Scanning_Filter_Policy 0x00 – Accept all advertising packets except directed advertising packets not addressed to this device.
0x01 – Accept only advertising packets from devices where the advertiser's address is in the White List.
Directed advertising packets which are not addressed to this device shall be ignored.
0x02 – Accept all advertising packets except directed advertising packets where the initiator's identity address does not address this device.
Note: directed advertising packets where the initiator's address is a resolvable private address that cannot be resolved are also accepted.
0x03 – Accept all advertising packets except:
• advertising packets where the advertiser's identity address is not in the White List; and
• directed advertising packets where the initiator's identity address does not address this device.
Note: directed advertising packets where the initiator's address is a resolvable private address that cannot be resolved are also accepted.
All other values reserved for future use.

Scanning_PHYs Bit 0 – Scan advertisements on the LE 1M PHY.
Bit 2 – Scan advertisements on the LE Coded PHY.
All other bits reserved for future use.

Scan_Type[i] 0x00 – Passive Scanning. No scan request PDUs shall be sent.
0x01 – Active Scanning. Scan request PDUs may be sent.
All other values reserved for future use.

Scan_Interval[i] Time interval from when the Controller started its last scan until it begins the subsequent scan on the primary advertising channel.
Range: 0x0004 to 0xFFFF.
Time = N * 0.625 ms.
Time Range: 2.5 ms to 40.959375 seconds.

Scan_Window[i] Duration of the scan on the primary advertising channel.
Range: 0x0004 to 0xFFFF.
Time = N * 0.625 ms.
Time Range: 2.5 ms to 40.959375 seconds.

Returns:

reply_id = CMD_LE_SET_EXTENDED_SCAN_PARAMETERS_RSP (164)
reply_size = 1

Status 0x00 – LE_Set_Extended_Scan_Parameters command succeeded.
0x01-0xFF – LE_Set_Extended_Scan_Parameters command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.82 LE Set Extended Scan Enable

Synopsis: The LE_Set_Extended_Scan_Enable command is used to enable or disable scanning.

Parameters:

request_id = CMD_LE_SET_EXTENDED_SCAN_ENABLE_REQ (165)
request_size = 6

Enable 0x00 – Scanning disabled.
0x01 – Scanning enabled.
All other values reserved for future use.

Filter_Duplicates 0x00 – Duplicate filtering disabled.
0x01 – Duplicate filtering enabled.
0x02 – Duplicate filtering enabled, reset for each scan period.
All other values reserved for future use.

Duration 0x0000 – Scan continuously until explicitly disable.
N = 0xXXXX – Scan duration.
Range: 0x0001 – 0xFFFF.
Time = N * 10 ms.
Time Range: 10 ms to 655.35 seconds.

Period 0x0000 – Periodic scanning disabled.
N = 0xXXXX – Time interval from when the Controller started its last Scan Duration until it begins the subsequent Scan Duration.
Range: 0x0001 – 0xFFFF.
Time = N * 1.28 seconds.
Time Range: 1.28 seconds to 83,884.8 seconds.

Returns:

reply_id = CMD_LE_SET_EXTENDED_SCAN_ENABLE_RSP (166)
reply_size = 1

Status 0x00 – LE_Set_Extended_Scan_Enable command succeeded.
0x01-0xFF – LE_Set_Extended_Scan_Enable command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.83 LE Extended Create Connection

Synopsis: The LE_Extended_Create_Connection command is used to create a Link Layer connection to a connectable advertiser. LE_Extended_Create_Connection command can be used in place of LE_Create_Connection command.

Parameters:

request_id = CMD_LE_EXTENDED_CREATE_CONNECTION_REQ (167)
request_size = 10 + 16 * number of bits set in Initiating_PHYs

Initiating_Filter_Policy 0x00 – White List is not used to determine which advertiser to connect to. Peer_Address_Type and Peer_Address shall be used.
0x01 – White List is used to determine which advertiser to connect to. Peer_Address_Type and Peer_Address shall be ignored.
All other values reserved for future use.

Own_Address_Type 0x00 – Public Device Address.
0x01 – Random Device Address.

0x02 – Controller generates the Resolvable Private Address based on the local IRK from the resolving list.
If the resolving list contains no matching entry, then use the public address.
0x03 – Controller generates the Resolvable Private Address based on the local IRK from the resolving list.
If the resolving list contains no matching entry, then use the random address from the most recent successful LE_Set_Random_Address Command.
All other values reserved for future use.

Peer_Address_Type 0x00 – Public Device Address or Public Identity Address.
0x01 – Random Device Address or Random (static) Identity Address.
All other values reserved for future use.

Peer_Address Public Device Address, Random Device Address, Public Identity Address, or Random (static) Identity Address of the device to be connected.

Initiating_PHYs (N) Bit 0 – Scan connectable advertisements on the LE 1M PHY. Connection parameters for the LE 1M PHY are provided.
Bit 1 – Connection parameters for the LE 2M PHY are provided.
Bit 2 – Scan connectable advertisements on the LE Coded PHY. Connection parameters for the LE Coded PHY are provided.
All other bits reserved for future use.

Scan_Interval[i] Time interval from when the Controller started its last scan until it begins the subsequent scan on the primary advertising channel.
Range: 0x0004 to 0xFFFF.
Time = N * 0.625 ms.
Time Range: 2.5 ms to 40.959375 seconds.

Scan_Window[i] Duration of the scan on the primary advertising channel.
Range: 0x0004 to 0xFFFF.
Time = N * 0.625 ms.
Time Range: 2.5 ms to 40.959375 seconds.

Conn_Interval_Min[i] Minimum value for the connection interval. This shall be less than or equal to Conn_Interval_Max[i].
Range: 0x0006 to 0x0C80.
Time = N * 1.25 ms.
Time Range: 7.5 ms to 4 seconds.

Conn_Interval_Max[i] Maximum value for the connection interval. This shall be greater than or equal to Conn_Interval_Max[i].
Range: 0x0006 to 0x0C80.
Time = N * 1.25 ms.
Time Range: 7.5 ms to 4 seconds.

Conn_Latency[i] Slave latency for the connection in number of connection events.
Range: 0x0000 to 0x01F3.

Supervision_Timeout[i] Supervision timeout for the LE Link. (See [Vol 6] Part B, Section 4.5.2)
Range: 0x000A to 0x0C80.
Time = N * 10 ms.
Time Range: 100 ms to 32 seconds.

Minimum_CE_Length[i] Informative parameter recommending the minimum length of connection event needed for this LE connection.
Range: 0x0000 – 0xFFFF.
Time = N * 0.625 ms.

Maximum_CE_Length[i] Informative parameter recommending the maximum length of connection event needed for this LE connection.
Range: 0x0000 – 0xFFFF.
Time = N * 0.625 ms.

Returns:

reply_id = CMD_LE_EXTENDED_CREATE_CONNECTION_RSP (168)
reply_size = 4

Status 0x00 – Command currently in pending.
0x01-0xFF – Command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Num_HCI_Command_Packets The Number of HCI command packets which are allowed to be sent to the Controller from the Host.

Command_Opcode Opcode of the command which caused this event and is pending completion.

A.84 LE Periodic Advertising Create Sync

Synopsis: The LE_Periodic_Advertising_Create_Sync command is used to synchronize with periodic advertising from an advertiser and begin receiving periodic advertising packets.

Parameters:

request_id = CMD_LE_PERIODIC_ADVERTISING_CREATE_SYNC_REQ (169)
request_size = 14

Filter_Policy 0x00 – Use the Advertising_SID, Advertising_Address_Type, and Advertising_Address parameters to determine which advertiser to listen to.
0x01 – Use the Periodic Advertiser List to determine which advertiser to listen to.
All other values reserved for future use.

Advertising_SID 0x00 – 0x0F – Advertising SID subfield in the ADI field used to identify the Periodic Advertising.
All other values reserved for future use.

Advertising_Address_Type 0x00 – Public Device Address.
0x01 – Random Device Address.
All other values reserved for future use.

Advertising_Address Public Device Address, Random Device Address, Public Identity Address, or Random (static) Identity Address of the advertiser.

Skip The number of periodic advertising packets that can be skipped after a successful receive.
Range: 0x0000 to 0x01F3.

Sync_Timeout Synchronization timeout for the periodic advertising.
Range: 0x000A to 0x4000.
Time = N*10 ms.
Time Range: 100 ms to 163.84 seconds.

Unused 0x00 This value must be used by the Host.
All other values reserved for future use.

Returns:

reply_id = CMD_LE_PERIODIC_ADVERTISING_CREATE_SYNC_RSP (170)
reply_size = 4

Status 0x00 – Command currently in pending.
0x01-0xFF – Command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Num_HCI_Command_Packets The Number of HCI command packets which are allowed to be sent to the Controller from the Host.

Command_Opcode Opcode of the command which caused this event and is pending completion.

A.85 LE Periodic Advertising Create Sync Cancel

Synopsis: The LE_Periodic_Advertising_Create_Sync_Cancel command is used to cancel the LE_Periodic_Advertising_Create_Sync command while it is pending.

Parameters:

request_id = CMD_LE_PERIODIC_ADVERTISING_CREATE_SYNC_CANCEL_REQ (171)
request_size = 0

Returns:

reply_id = CMD_LE_PERIODIC_ADVERTISING_CREATE_SYNC_CANCEL_RSP (172)
reply_size = 1

Status 0x00 – LE_Periodic_Advertising_Create_Sync_Cancel command succeeded.
0x01-0xFF – LE_Periodic_Advertising_Create_Sync_Cancel command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.86 LE Periodic Advertising Terminate Sync

Synopsis: The LE_Periodic_Advertising_Terminate_Sync command is used to stop reception of the periodic advertising identified by the Sync_Handle parameter.

Parameters:

request_id = CMD_LE_PERIODIC_ADVERTISING_TERMINATE_SYNC_REQ (173)
request_size = 2

Sync_Handle Sync handle to be used to identify the periodic advertiser.
Range: 0x0000-0x0EFF.

Returns:

reply_id = CMD_LE_PERIODIC_ADVERTISING_TERMINATE_SYNC_RSP (174)
reply_size = 1

Status 0x00 – LE_Periodic_Advertising_Terminate_Sync command succeeded.
0x01-0xFF – LE_Periodic_Advertising_Terminate_Sync command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.87 LE Add Device To Periodic Advertiser List

Synopsis: The LE_Add_Device_To_Periodic_Advertiser_List command is used to add a single device to the Periodic Advertiser list stored in the Controller. Any additions to the Periodic Advertiser list take effect immediately. If the device is already on the list, the Controller shall return the error code Invalid HCI Command Parameters (0x12).

Parameters:

request_id = CMD_LE_ADD_DEVICE_TO_PERIODIC_ADVERTISER_LIST_REQ (175)
request_size = 8

Advertiser_Address_Type 0x00 – Public Device Address or Public Identity Address.
0x01 – Random Device Address or Random (static) Identity Address.
All other values reserved for future use.

Advertiser_Address Public Device Address, Random Device Address, Public Identity Address, or Random (static) Identity Address of the advertiser.

Advertising_SID 0x00-0x0F – Advertising_SID subfield in the ADI field used to identify the Periodic Advertising.
All other values reserved for future use.

Returns:

reply_id = CMD_LE_ADD_DEVICE_TO_PERIODIC_ADVERTISER_LIST_RSP (176)
reply_size = 1

Status 0x00 – LE_Add_Device_To_Periodic_Advertiser_List command succeeded.
0x01-0xFF – LE_Add_Device_To_Periodic_Advertiser_List command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.88 LE Remove Device From Periodic Advertiser List

Synopsis: The LE_Remove_Device_From_Periodic_Advertiser_List command is used to remove one device from the list of Periodic Advertisers stored in the Controller. Removals from the Periodic Advertisers List take effect immediately.

Parameters:

request_id =
CMD_LE_REMOVE_DEVICE_FROM_PERIODIC_ADVERTISER_LIST_REQ (177)
request_size = 8

Advertiser_Address_Type 0x00 – Public Device Address or Public Identity Address.
0x01 – Random Device Address or Random (static) Identity Address.
All other values reserved for future use.

Advertiser_Address Public Device Address, Random Device Address, Public Identity Address, or Random (static) Identity Address of the advertiser.

Advertising_SID 0x00-0x0F – Advertising SID subfield in the ADI field used to identify the Periodic Advertising.
All other values reserved for future use.

Returns:

reply_id = CMD_LE_REMOVE_DEVICE_FROM_PERIODIC_ADVERTISER_LIST_RSP (178)
reply_size = 1

Status 0x00 – LE_Remove_Device_From_Periodic_Advertiser_List command succeeded.
0x01-0xFF – LE_Remove_Device_From_Periodic_Advertiser_List command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.89 LE Clear Periodic Advertiser List

Synopsis: The LE_Clear_Periodic_Advertiser_List command is used to remove all devices from the list of Periodic Advertisers in the Controller.

Parameters:

request_id = CMD_LE_CLEAR_PERIODIC_ADVERTISER_LIST_REQ (179)
request_size = 0

Returns:

reply_id = CMD_LE_CLEAR_PERIODIC_ADVERTISER_LIST_RSP (180)
reply_size = 1

Status 0x00 – LE_Clear_Periodic_Advertiser_List command succeeded.
0x01-0xFF – LE_Clear_Periodic_Advertiser_List command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.90 LE Read Periodic Advertiser List Size

Synopsis: The LE_Read_Periodic_Advertiser_List_Size command is used to read the total number of Periodic Advertiser list entries that can be stored in the Controller. Note: The number of entries that can be stored is not fixed and the Controller can change it at any time (e.g., because the memory used to store the list can also be used for other purposes).

Parameters:

request_id = CMD_LE_READ_PERIODIC_ADVERTISER_LIST_SIZE_REQ (181)
request_size = 0

Returns:

reply_id = CMD_LE_READ_PERIODIC_ADVERTISER_LIST_SIZE_RSP (182)
reply_size = 2

Status 0x00 – LE_Read_Periodic_Advertiser_List_Size command succeeded.
0x01-0xFF – LE_Read_Periodic_Advertiser_List_Size command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

Periodic_Advertiser_List_Size 0x01 – 0xFF – Total number of Periodic Advertiser list entries that can be stored in the Controller.
0x00 – Reserved for future use.

A.91 LE Read RF Path Compensation

Synopsis: The LE_Read_RF_Path_Compensation command is used to read the RF Path Compensation Values parameter used in the Tx Power Level and RSSI calculation.

Parameters:

request_id = CMD_LE_READ_RF_PATH_COMPENSATION_REQ (183)
request_size = 0

Returns:

reply_id = CMD_LE_READ_RF_PATH_COMPENSATION_RSP (184)
reply_size = 5

Status 0x00 – LE_Read_RF_Path_Compensation command succeeded.
0x01-0xFF – LE_Read_RF_Path_Compensation command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

RF_Tx_Path_Compensation_Value Size: 2 Octets (signed integer).
Range: -128.0 dB (0xFB00) ≤ N ≤ 128.0 dB (0x0500).
Units: 0.1 dB.

RF_Rx_Path_Compensation_Value Size: 2 Octets (signed integer).
Range: -128.0 dB (0xFB00) ≤ N ≤ 128.0 dB (0x0500).
Units: 0.1 dB.

A.92 LE Write RF Path Compensation

Synopsis: The LE_Write_RF_Path_Compensation command is used to indicate the RF path gain or loss between the RF transceiver and the antenna contributed by intermediate components. A positive value means a net RF path gain and a negative value means a net RF path loss. The RF Tx Path Compensation Value parameter shall be used by the Controller to calculate radiative Tx Power Level used in the TxPower field in the Extended Header using the following equation: Radiative Tx Power Level = Tx Power Level at RF transceiver output + RF Tx Path Compensation Value For example, if the Tx Power Level is +4 (dBm) at RF transceiver output and the RF Path Compensation Value is -1.5 (dB), the radiative Tx Power Level is +4+(-1.5) = 2.5 (dBm).

Parameters:

request_id = CMD_LE_WRITE_RF_PATH_COMPENSATION_REQ (185)
request_size = 4

RF_Tx_Path_Compensation_Value Size: 2 Octets (signed integer).
Range: -128.0 dB (0xFB00) ≤ N ≤ 128.0 dB (0x0500).
Units: 0.1 dB.

RF_Rx_Path_Compensation_Value Size: 2 Octets (signed integer).
Range: -128.0 dB (0xFB00) ≤ N ≤ 128.0 dB (0x0500).
Units: 0.1 dB.

Returns:

reply_id = CMD_LE_WRITE_RF_PATH_COMPENSATION_RSP (186)
reply_size = 1

Status 0x00 – LE_Write_RF_Path_Compensation command succeeded.
0x01-0xFF – LE_Write_RF_Path_Compensation command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.

A.93 Flush Events

Synopsis: The Flush_Events command is used to flush the internal queue of HCI events.

Parameters:

request_id = CMD_FLUSH_EVENTS_REQ (187)
request_size = 0

Returns:

reply_id = CMD_FLUSH_EVENTS_RSP (188)
reply_size = 0

A.94 Get Event

Synopsis: The Get_Event command is used to retrieve an event from the internal queue of HCI events. Note that this function is blocking. It will wait for an event to arrive.

Parameters:

request_id = CMD_GET_EVENT_REQ (189)
request_size = 0

Returns:

reply_id = CMD_GET_EVENT_RSP (190)
reply_size = 6 + Event_Size

Time Timestamp for the event. Uptime in milliseconds. Four bytes long.

Event The event retrieved. One byte event code. Can have any one of the following values:
0x00 - BT_HCI_EVT_NONE
0x05 - BT_HCI_EVT_DISCONN_COMPLETE
0x08 - BT_HCI_EVT_ENCRYPT_CHANGE
0x0C - BT_HCI_EVT_REMOTE_VERSION_INFO
0x0E - BT_HCI_EVT_CMD_COMPLETE
0x0F - BT_HCI_EVT_CMD_STATUS
0x13 - BT_HCI_EVT_NUM_COMPLETED_PACKETS
0x1A - BT_HCI_EVT_DATA_BUF_OVERFLOW
0x30 - BT_HCI_EVT_ENCRYPT_KEY_REFRESH_COMPLETE
0x3E - BT_HCI_EVT_LE_META_EVENT
0x57 - BT_HCI_EVT_AUTH_PAYLOAD_TIMEOUT_EXP

Event_Size The number of bytes in Event_Data, holding data for the event.

Event_Data Data for the event. Actual layout and size depends on the event.

A.95 Has Event

Synopsis: The Has_Event command is used to check whether there are any events in the internal queue of HCI events.

Parameters:

request_id = CMD_HAS_EVENT_REQ (191)
request_size = 0

Returns:

reply_id = CMD_HAS_EVENT_RSP (192)
reply_size = 1

Has_Event True (1) if event(s) are available; false (0) otherwise.

A.96 Write BD_ADDR

Synopsis: The Write_BD_ADDR command is used to set the Public address of the Device.

Parameters:

request_id = CMD_WRITE_BD_ADDR_REQ (193)
request_size = 6

BD_ADDR Bluetooth address of the Device.

Returns:

reply_id = CMD_WRITE_BD_ADDR_RSP (194)
reply_size = 1

Status 0x00 – Write_BD_ADDR command succeeded.
0x01-0xFF – Write_BD_ADDR command failed. See Part D, Error Codes on page 379 for a list of error codes and descriptions.