1.3 LIST OF ERROR CODES

The error code of 0x00 means Success. The possible range of failure error codes is 0x01-0xFF. Section 2 provides an error code usage description for each failure error code.

Values marked as "Reserved for Future Use", can be used in future versions of the specification. A host shall consider any error code that it does not explicitly understand equivalent to the "Unspecified Error (0x1F)."

Error Code	Name
0x00	Success
0x01	Unknown HCI Command
0x02	Unknown Connection Identifier
0x03	Hardware Failure
0x04	Page Timeout
0x05	Authentication Failure
0x06	PIN or Key Missing
0x07	Memory Capacity Exceeded
0x08	Connection Timeout
0x09	Connection Limit Exceeded
0x0A	Synchronous Connection Limit To A Device Exceeded
0x0B	ACL Connection Already Exists
0x0C	Command Disallowed
0x0D	Connection Rejected due to Limited Resources
0x0E	Connection Rejected Due To Security Reasons
0x0F	Connection Rejected due to Unacceptable BD_ADDR
0x10	Connection Accept Timeout Exceeded
0x11	Unsupported Feature or Parameter Value
0x12	Invalid HCI Command Parameters
0x13	Remote User Terminated Connection
0x14	Remote Device Terminated Connection due to Low Resources
0x15	Remote Device Terminated Connection due to Power Off
0x16	Connection Terminated By Local Host
0x17	Repeated Attempts
0x18	Pairing Not Allowed

Table 1.1: List of Possible Error Codes

Error Code	Name
0x19	Unknown LMP PDU
0x1A	Unsupported Remote Feature / Unsupported LMP Feature
0x1B	SCO Offset Rejected
0x1C	SCO Interval Rejected
0x1D	SCO Air Mode Rejected
0x1E	Invalid LMP Parameters
0x1F	Unspecified Error
0x20	Unsupported LMP Parameter Value
0x21	Role Change Not Allowed
0x22	LMP Response Timeout
0x23	LMP Error Transaction Collision
0x24	LMP PDU Not Allowed
0x25	Encryption Mode Not Acceptable
0x26	Link Key Can Not be Changed
0x27	Requested QoS Not Supported
0x28	Instant Passed
0x29	Pairing With Unit Key Not Supported
0x2A	Different Transaction Collision
0x2B	Reserved
0x2C	QoS Unacceptable Parameter
0x2D	QoS Rejected
0x2E	Channel Classification Not Supported
0x2F	Insufficient Security
0x30	Parameter Out Of Mandatory Range
0x31	Reserved
0x32	Role Switch Pending
0x33	Reserved
0x34	Reserved Slot Violation
0x35	Role Switch Failed

Table 1.1: List of Possible Error Codes

2 ERROR CODE DESCRIPTIONS

2.1 UNKNOWN HCI COMMAND (0X01)

The Unknown HCI Command error code indicates that the controller does not understand the HCI Command Packet OpCode that the host sent. The OpCode given might not correspond to any of the OpCodes specified in this document, or any vendor-specific OpCodes, or the command may not have been implemented.

2.2 UNKNOWN CONNECTION IDENTIFIER (0X02)

The Unknown Connection Identifier error code indicates that a command was sent from the host that should identify a connection, but that connection does not exist.

2.3 HARDWARE FAILURE (0X03)

The Hardware Failure error code indicates to the host that something in the controller has failed in a manner that cannot be described with any other error code. The meaning implied with this error code is implementation dependent.

2.4 PAGE TIMEOUT (0X04)

The Page Timeout error code indicates that a page timed out because of the Page Timeout configuration parameter. This error code may occur only with the HCI_Remote_Name_Request and HCI_Create_Connection commands.

2.5 AUTHENTICATION FAILURE (0X05)

The Authentication Failure error code indicates that pairing or authentication failed due to incorrect results in the pairing or authentication procedure. This could be due to an incorrect PIN or Link Key.

2.6 PIN OR KEY MISSING (0X06)

The PIN or Key Missing error code is used when pairing failed because of a missing PIN, or authentication failed because of a missing Key.

2.7 MEMORY CAPACITY EXCEEDED (0X07)

The Memory Capacity Exceeded error code indicates to the host that the controller has run out of memory to store new parameters.

2.8 CONNECTION TIMEOUT (0X08)

The Connection Timeout error code indicates that the link supervision timeout has expired for a given connection.

2.9 CONNECTION LIMIT EXCEEDED (0X09)

The Connection Limit Exceeded error code indicates that an attempt to create another connection failed because the controller is already at its limit of the number of connections it can support. The number of connections a device can support is implementation dependent.

2.10 SYNCHRONOUS CONNECTION LIMIT TO A DEVICE EXCEEDED (0X0A)

The Synchronous Connection Limit to a Device Exceeded error code indicates that the controller has reached the limit to the number of synchronous connections that can be achieved to a device. The number of synchronous connections a device can support is implementation dependent.

2.11 ACL CONNECTION ALREADY EXISTS (0X0B)

The ACL Connection Already Exists error code indicates that an attempt to create a new ACL Connection to a device when there is already a connection to this device.

2.12 COMMAND DISALLOWED (0X0C)

The Command Disallowed error code indicates that the command requested cannot be executed because the controller is in a state where it cannot process this command at this time. This error shall not be used for command OpCodes where the error code Unknown HCI Command is valid.

2.13 CONNECTION REJECTED DUE TO LIMITED RESOURCES (0X0D)

The Connection Rejected Due To Limited Resources error code indicates that an incoming connection was rejected due to limited resources.

2.14 CONNECTION REJECTED DUE TO SECURITY REASONS (0X0E)

The Connection Rejected Due To Security Reasons error code indicates that a connection was rejected due to security requirements not being fulfilled, like authentication or pairing.

2.15 CONNECTION REJECTED DUE TO UNACCEPTABLE BD_ADDR (0X0F)

The Connection Rejected due to Unacceptable BD_ADDR error code indicates that a connection was rejected because this device does not accept the BD_ADDR. This may be because the device will only accept connections from specific BD_ADDRs.

2.16 CONNECTION ACCEPT TIMEOUT EXCEEDED (0X10)

The Connection Accept Timeout Exceeded error code indicates that the Connection Accept Timeout has been exceeded for this connection attempt.

2.17 UNSUPPORTED FEATURE OR PARAMETER VALUE (0X11)

The Unsupported Feature Or Parameter Value error code indicates that a feature or parameter value in the HCI Command is not supported. This error code shall not be used in an LMP PDU.

2.18 INVALID HCI COMMAND PARAMETERS (0X12)

The Invalid HCI Command Parameters error code indicates that at least one of the HCI command parameters is invalid.

This shall be used when:

- the parameter total length is invalid.
- a command parameter is an invalid type.
- a connection identifier does not match the corresponding event.
- a parameter value must be even.
- a parameter is outside of the specified range.
- two or more parameter values have inconsistent values.

Note: An invalid type can be, for example, when an SCO connection handle is used where an ACL connection handle is required.

2.19 REMOTE USER TERMINATED CONNECTION (0X13)

The Remote User Terminated Connection error code indicates that the user on the remote device terminated the connection.

2.20 REMOTE DEVICE TERMINATED CONNECTION DUE TO LOW RESOURCES (0X14)

The Remote Device Terminated Connection due to Low Resources error code indicates that the remote device terminated the connection because of low resources.

2.21 REMOTE DEVICE TERMINATED CONNECTION DUE TO POWER OFF (0X15)

The Remote Device Terminated Connection due to Power Off error code indicates that the remote device terminated the connection because the device is about to power off.

2.22 CONNECTION TERMINATED BY LOCAL HOST (0X16)

The Connection Terminated By Local Host error code indicates that the local device terminated the connection.

2.23 REPEATED ATTEMPTS (0X17)

The Repeated Attempts error code indicates that the controller is disallowing an authentication or pairing procedure because too little time has elapsed since the last authentication or pairing attempt failed.

2.24 PAIRING NOT ALLOWED (0X18)

The Pairing Not Allowed error code indicates that the device does not allow pairing. For example, when a device only allows pairing during a certain time window after some user input allows pairing.

2.25 UNKNOWN LMP PDU (0X19)

The Unknown LMP PDU error code indicates that the controller has received an unknown LMP opcode.

2.26 Unsupported Remote Feature / Unsupported LMP Feature (0X1A)

The Unsupported Remote Feature error code indicates that the remote device does not support the feature associated with the issued command or LMP PDU.

2.27 SCO OFFSET REJECTED (0X1B)

The SCO Offset Rejected error code indicates that the offset requested in the LMP_SCO_link_req message has been rejected.

2.28 SCO INTERVAL REJECTED (0X1C)

The SCO Interval Rejected error code indicates that the interval requested in the LMP_SCO_link_req message has been rejected.

2.29 SCO AIR MODE REJECTED (0X1D)

The SCO Air Mode Rejected error code indicates that the air mode requested in the LMP_SCO_link_req message has been rejected.

2.30 INVALID LMP PARAMETERS (0X1E)

The Invalid LMP Parameters error code indicates that some LMP message parameters were invalid. This shall be used when:

- the PDU length is invalid.
- a parameter value must be even.
- a parameter is outside of the specified range.
- two or more parameters have inconsistent values.

2.31 UNSPECIFIED ERROR (0X1F)

The Unspecified Error error code indicates that no other error code specified is appropriate to use.

2.32 UNSUPPORTED LMP PARAMETER VALUE (0X20)

The Unsupported LMP Parameter Value error code indicates that an LMP message contains at least one parameter value that is not supported by the controller at this time. This is normally used after a long negotiation procedure, for example during an LMP_hold_req, LMP_sniff_req and LMP_encryption_key_size_req message exchanges.

2.33 ROLE CHANGE NOT ALLOWED (0X21)

The Role Change Not Allowed error code indicates that a controller will not allow a role change at this time.

2.34 LMP RESPONSE TIMEOUT (0X22)

The LMP Response Timeout error code indicates that an LMP transaction failed to respond within the LMP response timeout.

2.35 LMP ERROR TRANSACTION COLLISION (0X23)

The LMP Error Transaction Collision error code indicates that an LMP transaction has collided with the same transaction that is already in progress.

2.36 LMP PDU NOT ALLOWED (0X24)

The LMP PDU Not Allowed error code indicates that a controller sent an LMP message with an opcode that was not allowed.

2.37 ENCRYPTION MODE NOT ACCEPTABLE (0X25)

The Encryption Mode Not Acceptable error code indicates that the requested encryption mode is not acceptable at this time.

2.38 LINK KEY CAN NOT BE CHANGED (0X26)

The Link Key Can Not be Changed error code indicates that a link key can not be changed because a fixed unit key is being used.

2.39 REQUESTED QoS NOT SUPPORTED (0X27)

The Requested QoS Not Supported error code indicates that the requested Quality of Service is not supported.

2.40 INSTANT PASSED (0X28)

The Instant Passed error code indicates that an LMP PDU that includes an instant can not be performed because the instant when this would have occurred has passed.

2.41 PAIRING WITH UNIT KEY NOT SUPPORTED (0X29)

The Pairing With Unit Key Not Supported error code indicates that it was not possible to pair as a unit key was requested and it is not supported.

2.42 DIFFERENT TRANSACTION COLLISION (0X2A)

The Different Transaction Collision error code indicates that an LMP transaction was started that collides with an ongoing transaction.

2.43 QoS UNACCEPTABLE PARAMETER (0X2C)

The QoS Unacceptable Parameter error code indicates that the specified quality of service parameters could not be accepted at this time, but other parameters may be acceptable.

2.44 QoS REJECTED (0X2D)

The QoS Rejected error code indicates that the specified quality of service parameters can not be accepted and QoS negotiation should be terminated.

2.45 CHANNEL CLASSIFICATION NOT SUPPORTED (0X2E)

The Channel Classification Not Supported error code indicates that the controller can not perform channel classification because it is not supported.

2.46 INSUFFICIENT SECURITY (0X2F)

The Insufficient Security error code indicates that the HCI command or LMP message sent is only possible on an encrypted link.

2.47 PARAMETER OUT OF MANDATORY RANGE (0X30)

The Parameter Out Of Mandatory Range error code indicates that a parameter value requested is outside the mandatory range of parameters for the given HCI command or LMP message.

2.48 ROLE SWITCH PENDING (0X32)

The Role Switch Pending error code indicates that a Role Switch is pending. This can be used when an HCI command or LMP message can not be accepted because of a pending role switch. This can also be used to notify a peer device about a pending role switch.

2.49 RESERVED SLOT VIOLATION (0X34)

The Reserved Slot Violation error code indicates that the current Synchronous negotiation was terminated with the negotiation state set to Reserved Slot Violation.

2.50 ROLE SWITCH FAILED (0X35)

The Role Switch Failed error code indicates that a role switch was attempted but it failed and the original piconet structure is restored. The switch may have failed because the TDD switch or piconet switch failed.



HOST CONTROLLER INTERFACE FUNCTIONAL SPECIFICATION

This document describes the functional specification for the Host Controller Interface (HCI).
The HCI provides a command interface to the baseband controller and link manager, and access to configuration parameters. This interface provides a uniform method of accessing the Bluetooth baseband capabilities.

CONTENTS

1 Introduction 343
1.1 Lower Layers of the Bluetooth Software Stack 343
2 Overview of Host Controller Transport Layer 345
3 Overview of Commands and Events 347
3.1 Generic Events 348
3.2 Device Setup 348
3.3 Controller Flow Control 349
3.4 Controller Information 349
3.5 Controller Configuration 350
3.6 Device Discovery 351
3.7 Connection Setup 353
3.8 Remote Information 355
3.9 Synchronous Connections 356
3.10 Connection State 357
3.11 Piconet Structure 358
3.12 Quality of Service 359
3.13 Physical Links 360
3.14 Host Flow Control 361
3.15 Link Information 362
3.16 Authentication and Encryption 363
3.17 Testing 365
3.18 Alphabetical List of Commands and Events 366
4 HCI Flow Control 371
4.1 Host to Controller Data Flow Control 371
4.2 Controller to Host Data Flow Control 372
4.3 Disconnection Behavior 373
4.4 Command Flow Control 373
4.5 Command Error Handling 374
5 HCI Data Formats 375
5.1 Introduction 375
5.2 Data and Parameter Formats 375
5.3 Connection Handles 376
5.4 Exchange of HCI-Specific Information 377
5.4.1 HCI Command Packet 377
5.4.2 HCI ACL Data Packets 379
5.4.3 HCI Synchronous Data Packets 381
5.4.4 HCI Event Packet 382
6 HCI Configuration Parameters 383

6.1 Scan Enable 383
6.2 Inquiry Scan Interval 383
6.3 Inquiry Scan Window 384
6.4 Inquiry Scan Type 384
6.5 Inquiry Mode 384
6.6 Page Timeout 385
6.7 Connection Accept Timeout 385
6.8 Page Scan Interval 386
6.9 Page Scan Window 386
6.10 Page Scan Period Mode (Deprecated) 386
6.11 Page Scan Type 387
6.12 Voice Setting 387
6.13 PIN Type 388
6.14 Link Key 388
6.15 Authentication Enable 388
6.16 Encryption Mode 389
6.17 Failed Contact Counter 390
6.18 Hold Mode Activity 390
6.19 Link Policy Settings 391
6.20 Flush Timeout 392
6.21 Num Broadcast Retransmissions 392
6.22 Link Supervision Timeout 393
6.23 Synchronous Flow Control Enable 393
6.24 Local Name 394
6.25 Class Of Device 394
6.26 Supported Commands 395

7 HCI Commands and Events 399

7.1 Link Control Commands 399
7.1.1 Inquiry Command 399
7.1.2 Inquiry Cancel Command 401
7.1.3 Periodic Inquiry Mode Command 402
7.1.4 Exit Periodic Inquiry Mode Command 405
7.1.5 Create Connection Command 406
7.1.6 Disconnect Command 409
7.1.7 Create Connection Cancel Command 410
7.1.8 Accept Connection Request Command 412
7.1.9 Reject Connection Request Command 414
7.1.10 Link Key Request Reply Command 415
7.1.11 Link Key Request Negative Reply Command 417
7.1.12 PIN Code Request Reply Command 418
7.1.13 PIN Code Request Negative Reply Command 420

7.1.14 Change Connection Packet Type Command 421
7.1.15 Authentication Requested Command.....424
7.1.16 Set Connection Encryption Command 425
7.1.17 Change Connection Link Key Command 426
7.1.18 Master Link Key Command 427
7.1.19 Remote Name Request Command 428
7.1.20 Remote Name Request Cancel Command 430
7.1.21 Read Remote Supported Features Command.....431
7.1.22 Read Remote Extended Features Command 432
7.1.23 Read Remote Version Information Command.....433
7.1.24 Read Clock Offset Command.....434
7.1.25 Read LMP Handle Command 435
7.1.26 Setup Synchronous Connection Command 437
7.1.27 Accept Synchronous Connection Request Command 442
7.1.28 Reject Synchronous Connection Request Command .446
7.2 Link Policy Commands.....447
7.2.1 Hold Mode Command 447
7.2.2 Sniff Mode Command.....449
7.2.3 Exit Sniff Mode Command.....452
7.2.4 Park State Command 453
7.2.5 Exit Park State Command 455
7.2.6 QoS Setup Command 456
7.2.7 Role Discovery Command.....458
7.2.8 Switch Role Command.....459
7.2.9 Read Link Policy Settings Command 460
7.2.10 Write Link Policy Settings Command 461
7.2.11 Read Default Link Policy Settings Command 463
7.2.12 Write Default Link Policy Settings Command 464
7.2.13 Flow Specification Command 465
7.3 Controller & Baseband Commands.....467
7.3.1 Set Event Mask Command.....467
7.3.2 Reset Command 469
7.3.3 Set Event Filter Command 470
7.3.4 Flush Command 475
7.3.5 Read PIN Type Command 477
7.3.6 Write PIN Type Command.....478
7.3.7 Create New Unit Key Command 479
7.3.8 Read Stored Link Key Command 480

7.3.9 Write Stored Link Key Command 481
7.3.10 Delete Stored Link Key Command 483
7.3.11 Write Local Name Command 484
7.3.12 Read Local Name Command 485
7.3.13 Read Connection Accept Timeout Command 486
7.3.14 Write Connection Accept Timeout Command 487
7.3.15 Read Page Timeout Command 488
7.3.16 Write Page Timeout Command 489
7.3.17 Read Scan Enable Command 490
7.3.18 Write Scan Enable Command 491
7.3.19 Read Page Scan Activity Command 492
7.3.20 Write Page Scan Activity Command 494
7.3.21 Read Inquiry Scan Activity Command 495
7.3.22 Write Inquiry Scan Activity Command 496
7.3.23 Read Authentication Enable Command 497
7.3.24 Write Authentication Enable Command 498
7.3.25 Read Encryption Mode Command 499
7.3.26 Write Encryption Mode Command 500
7.3.27 Read Class of Device Command 501
7.3.28 Write Class of Device Command 502
7.3.29 Read Voice Setting Command 503
7.3.30 Write Voice Setting Command 504
7.3.31 Read Automatic Flush Timeout Command 505
7.3.32 Write Automatic Flush Timeout Command 506
7.3.33 Read Num Broadcast Retransmissions Command 507
7.3.34 Write Num Broadcast Retransmissions Command 508
7.3.35 Read Hold Mode Activity Command 509
7.3.36 Write Hold Mode Activity Command 510
7.3.37 Read Transmit Power Level Command 511
7.3.38 Read Synchronous Flow Control Enable Command ... 513
7.3.39 Write Synchronous Flow Control Enable Command ... 514
7.3.40 Set Controller To Host Flow Control Command 515
7.3.41 Host Buffer Size Command 516
7.3.42 Host Number Of Completed Packets Command 518
7.3.43 Read Link Supervision Timeout Command 520
7.3.44 Write Link Supervision Timeout Command 521
7.3.45 Read Number Of Supported IAC Command 523
7.3.46 Read Current IAC LAP Command 524

7.3.47 Write Current IAC LAP Command....525
7.3.48 Read Page Scan Period Mode Command (Deprecated)....527
7.3.49 Write Page Scan Period Mode Command (Deprecated)....528
7.3.50 Set AFH Host Channel Classification Command....529
7.3.51 Read Inquiry Scan Type Command....530
7.3.52 Write Inquiry Scan Type Command....531
7.3.53 Read Inquiry Mode Command....532
7.3.54 Write Inquiry Mode Command....533
7.3.55 Read Page Scan Type Command....534
7.3.56 Write Page Scan Type Command....535
7.3.57 Read AFH Channel Assessment Mode Command....536
7.3.58 Write AFH Channel Assessment Mode Command....537
7.4 Informational Parameters....539
7.4.1 Read Local Version Information Command....539
7.4.2 Read Local Supported Commands Command....541
7.4.3 Read Local Supported Features Command....542
7.4.4 Read Local Extended Features Command....543
7.4.5 Read Buffer Size Command....545
7.4.6 Read BD_ADDR Command....547
7.5 Status Parameters....548
7.5.1 Read Failed Contact Counter Command....548
7.5.2 Reset Failed Contact Counter Command....550
7.5.3 Read Link Quality Command....551
7.5.4 Read RSSI Command....552
7.5.5 Read AFH Channel Map Command....554
7.5.6 Read Clock Command....556
7.6 Testing Commands....558
7.6.1 Read Loopback Mode Command....558
7.6.2 Write Loopback Mode Command....559
7.6.3 Enable Device Under Test Mode Command....562
7.7 Events....563
7.7.1 Inquiry Complete Event....563
7.7.2 Inquiry Result Event....564
7.7.3 Connection Complete Event....566
7.7.4 Connection Request Event....567
7.7.5 Disconnection Complete Event....569
7.7.6 Authentication Complete Event....570

7.7.7 Remote Name Request Complete Event 571
7.7.8 Encryption Change Event 572
7.7.9 Change Connection Link Key Complete Event 573
7.7.10 Master Link Key Complete Event 574
7.7.11 Read Remote Supported Features Complete Event ... 575
7.7.12 Read Remote Version Information Complete Event ... 576
7.7.13 QoS Setup Complete Event 577
7.7.14 Command Complete Event 579
7.7.15 Command Status Event 580
7.7.16 Hardware Error Event 581
7.7.17 Flush Occurred Event 581
7.7.18 Role Change Event 582
7.7.19 Number Of Completed Packets Event 583
7.7.20 Mode Change Event 584
7.7.21 Return Link Keys Event 586
7.7.22 PIN Code Request Event 587
7.7.23 Link Key Request Event 588
7.7.24 Link Key Notification Event 589
7.7.25 Loopback Command Event 590
7.7.26 Data Buffer Overflow Event 590
7.7.27 Max Slots Change Event 591
7.7.28 Read Clock Offset Complete Event 592
7.7.29 Connection Packet Type Changed Event 593
7.7.30 QoS Violation Event 596
7.7.31 Page Scan Repetition Mode Change Event 597
7.7.32 Flow Specification Complete Event 598
7.7.33 Inquiry Result with RSSI Event 600
7.7.34 Read Remote Extended Features Complete Event 602
7.7.35 Synchronous Connection Complete Event 603
7.7.36 Synchronous Connection Changed event 605

8 List of Figures 607

9 List of Tables 609

10 Appendix 609

1 INTRODUCTION

This document describes the functional specifications for the Host Controller Interface (HCI). The HCI provides a uniform interface method of accessing the Bluetooth controller capabilities. The next two sections provide a brief overview of the lower layers of the Bluetooth software stack and of the Bluetooth hardware. Section 2, provides an overview of the Lower HCI Device Driver Interface on the host device. Section 4, describes the flow control used between the Host and the Controller. Section 7, describes each of the HCI Commands in details, identifies parameters for each of the commands, and lists events associated with each command.

1.1 LOWER LAYERS OF THE BLUETOOTH SOFTWARE STACK

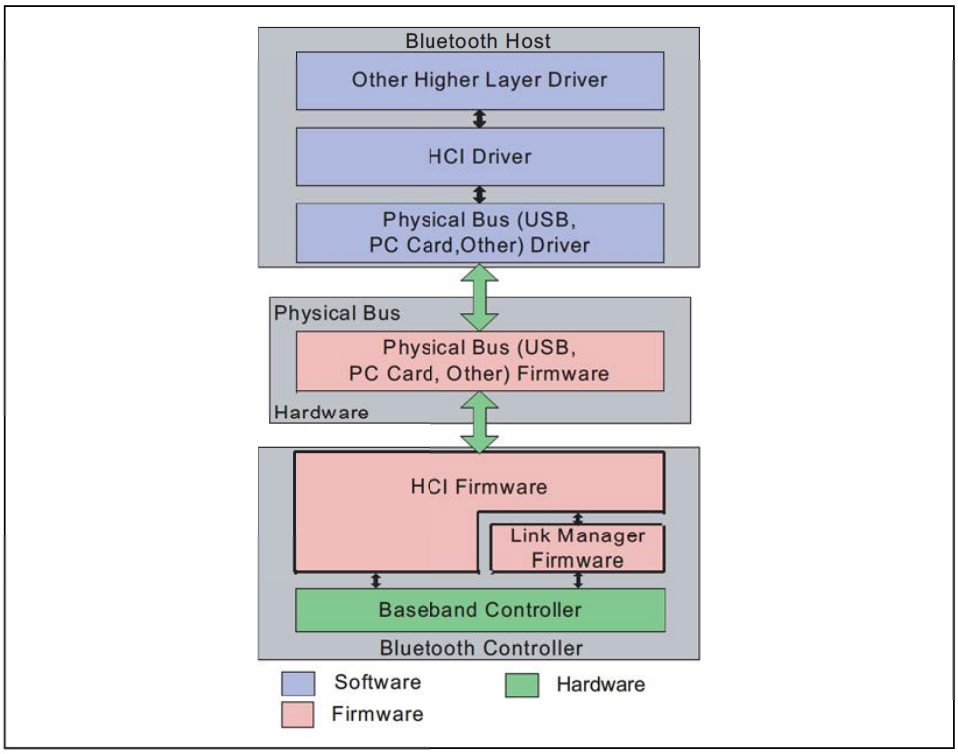


Figure 1.1: Overview of the Lower Software Layers

Figure 1.1, provides an overview of the lower software layers. The HCI firmware implements the HCI Commands for the Bluetooth hardware by accessing baseband commands link manager commands, hardware status registers, control registers, and event registers.

Several layers may exist between the HCI driver on the host system and the HCI firmware in the Bluetooth hardware. These intermediate layers, the Host

Controller Transport Layer, provide the ability to transfer data without intimate knowledge of the data.

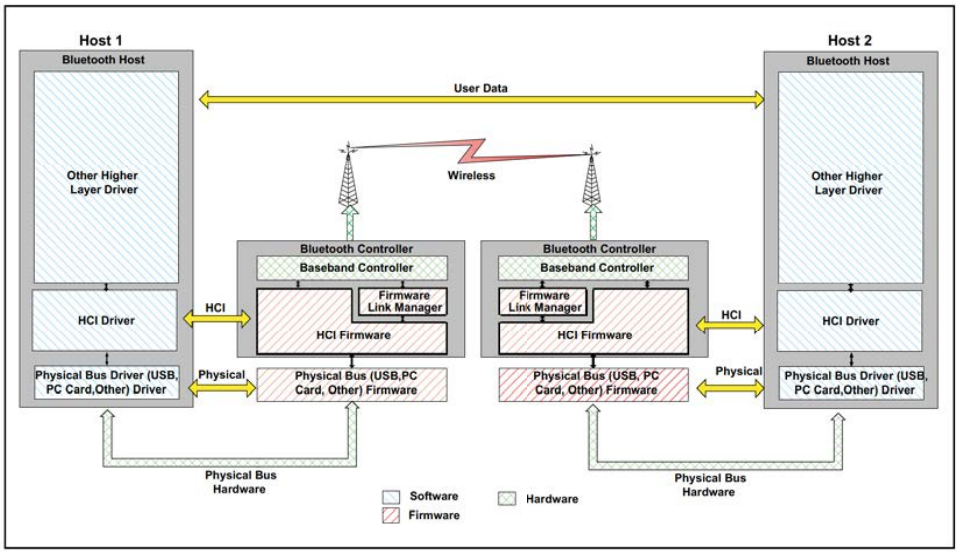


Figure 1.2: End to End Overview of Lower Software Layers to Transfer Data

Figure 1.2, illustrates the path of a data transfer from one device to another. The HCI driver on the Host exchanges data and commands with the HCI firmware on the Bluetooth hardware. The Host Control Transport Layer (i.e. physical bus) driver provides both HCI layers with the ability to exchange information with each other.

The Host will receive asynchronous notifications of HCI events independent of which Host Controller Transport Layer is used. HCI events are used for notifying the Host when something occurs. When the Host discovers that an event has occurred it will then parse the received event packet to determine which event occurred.

2 OVERVIEW OF HOST CONTROLLER TRANSPORT LAYER

The host driver stack has a transport layer between the Host Controller driver and the Host.

The main goal of this transport layer is transparency. The Host Controller driver (which interfaces to the Controller) should be independent of the underlying transport technology. Nor should the transport require any visibility into the data that the Host Controller driver passes to the Controller. This allows the interface (HCI) or the Controller to be upgraded without affecting the transport layer.

The specified Host Controller Transport Layers are described in a separate volume. (See specification volume 4)

3 OVERVIEW OF COMMANDS AND EVENTS

The commands and events are sent between the Host and the Controller. These are grouped into logical groups by function.

Generic Events	The generic events can occur due to multiple commands, or events that can occur at any time.
Device Setup	The device setup commands are used to place the Controller into a known state.
Controller Flow Control	The controller flow control commands and events are used to control data flow from the Host to the controller.
Controller Information	The controller information commands allow the Host to discover local information about the device.
Controller Configuration	The controller configuration commands and events allow the global configuration parameters to be configured.
Device Discovery	The device discovery commands and events allow a device to discover other devices in the surrounding area.
Connection Setup	The connection setup commands and events allow a device to make a connection to another device.
Remote Information	The remote information commands and events allow information about a remote device's configuration to be discovered.
Synchronous Connections	The synchronous connection commands and events allow synchronous connections to be created
Connection State	The connection state commands and events allow the configuration of a link, especially for low power operation.
Piconet Structure	The piconet structure commands and events allow the discovery and reconfiguration of piconet.
Quality of Service	The quality of service commands and events allow quality of service parameters to be specified.
Physical Links	The physical link commands and events allow the configuration of a physical link.
Host Flow Control	The Host flow control commands and events allow flow control to be used towards the Host.
Link Information	The link information commands and events allow information about a link to be read.
Authentication and Encryption	The authentication and encryption commands and events allow authentication of a remote device and then encryption of the link.
Testing	The testing commands and events allow a device to be placed into test mode.

Table 3.1: Overview of commands and events

The version information in this section denotes the version number of the specification that this command or event was first specified

3.1 GENERIC EVENTS

The generic events occur due to multiple commands, or events that can occur at any time.

Name	Vers.	Summary description
Command Complete Event	1.1	The Command Complete event is used by the Controller to pass the return status of a command and the other event parameters for each HCI Command.
Command Status Event	1.1	The Command Status event is used to indicate that the command described by the Command_Opcode parameter has been received and the Controller is currently performing the task for this command.
Hardware Error Event	1.1	The Hardware Error event is used to indicate some type of hardware failure for the Controller.

Table 3.2: Generic events

3.2 DEVICE SETUP

The device setup group of commands are used to place the Controller into a known state.

Name	Vers.	Summary description
Reset Command	1.1	The Reset command will reset the Controller, Link Manager, and the Bluetooth radio.

Device setup

3.3 CONTROLLER FLOW CONTROL

The controller flow control group of commands and events are used to control data flow from the Host to the Controller.

Name	Vers.	Summary description
Read Buffer Size Command	1.1	The Read_Buffer_Size command returns the size of the HCI buffers. These buffers are used by the Controller to buffer data that is to be transmitted.
Number Of Completed Packets Event	1.1	The Number Of Completed Packets event is used by the Controller to indicate to the Host how many HCI Data Packets have been completed for each Connection Handle since the previous Number Of Completed Packets event was sent.

Table 3.3: Controller flow control

3.4 CONTROLLER INFORMATION

The controller information group of commands allows the Host to discover local information about the device.

Name	Vers.	Summary description
Read Local Version Information Command	1.1	The Read Local Version Information command will read the version information for the local Bluetooth device.
Read Local Supported Commands Command	1.2	The Read Local Supported Commands command requests a list of the supported HCI commands for the local device.
Read Local Supported Features Command	1.1	The Read Local Supported Features command requests a list of the supported features for the local device.
Read Local Extended Features Command	1.2	The Read Local Extended Features command requests a list of the supported extended features for the local device
Read BD_ADDR Command	1.1	The Read BD_ADDR command will read the value for the BD_ADDR parameter.

Table 3.4: Controller information

3.5 CONTROLLER CONFIGURATION

The controller configuration group of commands and events allows the global configuration parameters to be configured.

Name	Vers.	Summary description
Read Local Name Command	1.1	The Read Local Name command provides the ability to read the stored user-friendly name for the Bluetooth device.
Write Local Name Command	1.1	The Write Local Name command provides the ability to modify the user-friendly name for the Bluetooth device.
Read Class of Device Command	1.1	The Read Class of Device command will read the value for the Class of Device configuration parameter, which is used to indicate its capabilities to other devices.
Write Class of Device Command	1.1	The Write Class of Device command will write the value for the Class_of_Device configuration parameter, which is used to indicate its capabilities to other devices.
Read Number Of Supported IAC Command	1.1	The Read Number of Supported IAC command will read the value for the number of Inquiry Access Codes (IAC) that the local Bluetooth device can simultaneously listen for during an Inquiry Scan.
Read Current IAC LAP Command	1.1	The Read Current IAC LAP command will read the LAP(s) used to create the Inquiry Access Codes (IAC) that the local Bluetooth device is simultaneously scanning for during Inquiry Scans.
Write Current IAC LAP Command	1.1	The Write Current IAC LAP command will write the LAP(s) used to create the Inquiry Access Codes (IAC) that the local Bluetooth device is simultaneously scanning for during Inquiry Scans.
Read Scan Enable Command	1.1	The Read Scan Enable command will read the value for the Scan Enable configuration parameter, which controls whether or not the Bluetooth device will periodically scan for page attempts and/or inquiry requests from other Bluetooth devices.
Write Scan Enable Command	1.1	The Write Scan Enable command will write the value for the Scan Enable configuration parameter, which controls whether or not the Bluetooth device will periodically scan for page attempts and/or inquiry requests from other Bluetooth devices.

Table 3.5: Controller configuration

3.6 DEVICE DISCOVERY

The device discovery group of commands and events allow a device to discovery other devices in the surrounding area.

Name	Vers.	Summary description
Inquiry Command	1.1	The Inquiry command will cause the Bluetooth device to enter Inquiry Mode. Inquiry Mode is used to discovery other nearby Bluetooth devices.
Inquiry Result Event	1.1	The Inquiry Result event indicates that a Bluetooth device or multiple Bluetooth devices have responded so far during the current Inquiry process.
Inquiry Result with RSSI Event	1.2	The Inquiry Result with RSSI event indicates that a Bluetooth device or multiple Bluetooth devices have responded so far during the current Inquiry process.
Inquiry Cancel Command	1.1	The Inquiry Cancel command will cause the Bluetooth device to stop the current Inquiry if the Bluetooth device is in Inquiry Mode.
Inquiry Complete Event	1.1	The Inquiry Complete event indicates that the Inquiry is finished.
Periodic Inquiry Mode Command	1.1	The Periodic Inquiry Mode command is used to configure the Bluetooth device to perform an automatic Inquiry based on a specified period range.
Exit Periodic Inquiry Mode Command	1.1	The Exit Periodic Inquiry Mode command is used to end the Periodic Inquiry mode when the local device is in Periodic Inquiry Mode.
Read Inquiry Scan Activity Command	1.1	The Read Inquiry Scan Activity command will read the value for Inquiry Scan Interval and Inquiry Scan Window configuration parameters. Inquiry Scan Interval defines the amount of time between consecutive inquiry scans. Inquiry Scan Window defines the amount of time for the duration of the inquiry scan.
Write Inquiry Scan Activity Command	1.1	The Write Inquiry Scan Activity command will write the value for Inquiry Scan Interval and Inquiry Scan Window configuration parameters. Inquiry Scan Interval defines the amount of time between consecutive inquiry scans. Inquiry Scan Window defines the amount of time for the duration of the inquiry scan.
Read Inquiry Scan Type Command	1.2	The Read Inquiry Scan Type command is used to read the Inquiry Scan Type configuration parameter of the local Bluetooth device. The Inquiry Scan Type configuration parameter can set the inquiry scan to either normal or interlaced scan.

Table 3.6: Device discovery

Name	Vers.	Summary description
Write Inquiry Scan Type Command	1.2	The Write Inquiry Scan Type command is used to write the Inquiry Scan Type configuration parameter of the local Bluetooth device. The Inquiry Scan Type configuration parameter can set the inquiry scan to either normal or interlaced scan.
Read Inquiry Mode Command	1.2	The Read Inquiry Mode command is used to read the Inquiry Mode configuration parameter of the local Bluetooth device.
Write Inquiry Mode Command	1.2	The Write Inquiry Mode command is used to write the Inquiry Mode configuration parameter of the local Bluetooth device.

Table 3.6: Device discovery

3.7 CONNECTION SETUP

The connection setup group of commands and events are used to allow a device to make a connection to another device.

Name	Vers.	Summary description
Create Connection Command	1.1	The Create Connection command will cause the link manager to create an ACL connection to the Bluetooth device with the BD_ADDR specified by the command parameters.
Connection Request Event	1.1	The Connection Request event is used to indicate that a new incoming connection is trying to be established.
Accept Connection Request Command	1.1	The Accept Connection Request command is used to accept a new incoming connection request.
Reject Connection Request Command	1.1	The Reject Connection Request command is used to decline a new incoming connection request.
Create Connection Cancel Command	1.2	The Create Connection Cancel Command is used to cancel an ongoing Create Connection.
Connection Complete Event	1.1	The Connection Complete event indicates to both of the Hosts forming the connection that a new connection has been established.
Disconnect Command	1.1	The Disconnect command is used to terminate an existing connection.
Disconnection Complete Event	1.1	The Disconnection Complete event occurs when a connection has been terminated.
Read Page Timeout Command	1.1	The Read Page Timeout command will read the value for the Page Reply Timeout configuration parameter, which determines the time the Bluetooth controller will wait for the remote device to respond to a connection request before the local device returns a connection failure.
Write Page Timeout Command	1.1	The Write Page Timeout command will write the value for the Page Reply Timeout configuration parameter, which allows the Bluetooth hardware to define the amount of time a connection request will wait for the remote device to respond before the local device returns a connection failure.
Read Page Scan Activity Command	1.1	The Read Page Scan Activity command will read the values for the Page Scan Interval and Page Scan Window configuration parameters. Page Scan Interval defines the amount of time between consecutive page scans. Page Scan Window defines the duration of the page scan.

Table 3.7: Connection setup

Name	Vers.	Summary description
Write Page Scan Activity Command	1.1	The Write Page Scan Activity command will write the value for Page Scan Interval and Page Scan Window configuration parameters. Page Scan Interval defines the amount of time between consecutive page scans. Page Scan Window defines the duration of the page scan.
Page Scan Repetition Mode Change Event	1.1	The Page Scan Repetition Mode Change event indicates that the connected remote Bluetooth device with the specified Connection_Handle has successfully changed the Page_Scan_Repetition_Mode (SR)."
Read Page Scan Type Command	1.2	The Read Page Scan Type command is used to read the page scan type of the local Bluetooth device. The Page Scan Type configuration parameter can set the page scan to either normal or interlaced scan.
Write Page Scan Type Command	1.2	The Write Page Scan Type command is used to write the page scan type of the local Bluetooth device. The Page Scan Type configuration parameter can set the page scan to either normal or interlaced scan.
Read Connection Accept Timeout Command	1.1	The Read Connection Accept Timeout command will read the value for the Connection Accept Timeout configuration parameter, which allows the Bluetooth hardware to automatically deny a connection request after a specified period has occurred, and to refuse a new connection.
Write Connection Accept Timeout Command	1.1	The Write Connection Accept Timeout command will write the value for the Connection Accept Timeout configuration parameter, which allows the Bluetooth hardware to automatically deny a connection request after a specified period has occurred, and to refuse a new connection.

Table 3.7: Connection setup

3.8 REMOTE INFORMATION

The remote information group of commands and events allows information about a remote devices configuration to be discovered.

Name	Vers.	Summary description
Remote Name Request Command	1.1	The Remote Name Request command is used to obtain the user-friendly name of another Bluetooth device.
Remote Name Request Cancel Command	1.2	The Remote Name Request Cancel Command is used to cancel an ongoing Remote Name Request.
Remote Name Request Complete Event	1.1	The Remote Name Request Complete event is used to indicate a remote name request has been completed.
Read Remote Supported Features Command	1.1	The Read Remote Supported Features command requests a list of the supported features of a remote device.
Read Remote Supported Features Complete Event	1.1	The Read Remote Supported Features Complete event is used to indicate the completion of the process of the Link Manager obtaining the supported features of the remote Bluetooth device specified by the Connection Handle event parameter.
Read Remote Extended Features Command	1.2	The Read Remote Extended Features command requests a list of the supported extended features of a remote device
Read Remote Extended Features Complete Event	1.2	The Read Remote Extended Features Complete Event is used to indicate the completion of the process of the Link Manager obtaining the supported Extended features of the remote Bluetooth device specified by the connection handle event parameter.
Read Remote Version Information Command	1.1	The Read Remote Version Information command will read the values for the version information for the remote Bluetooth device.
Read Remote Version Information Complete Event	1.1	The Read Remote Version Information Complete event is used to indicate the completion of the process of the Link Manager obtaining the version information of the remote Bluetooth device specified by the Connection Handle event parameter.

Table 3.8: Remote information

3.9 SYNCHRONOUS CONNECTIONS

The synchronous connections group of commands and events allows synchronous connections to be created.

Name	Vers.	Summary description
Setup Synchronous Connection Command	1.2	The Setup Synchronous Connection command adds a new or modifies an existing synchronous logical transport (SCO or eSCO) on a physical link depending on the Connection Handle parameter specified.
Synchronous Connection Complete Event	1.2	The Synchronous Connection Complete event indicates to both the Hosts that a new Synchronous connection has been established.
Synchronous Connection Changed event	1.2	The Synchronous Connection Changed event indicates to the Host that an existing Synchronous connection has been reconfigured.
Accept Synchronous Connection Request Command	1.2	The Accept_Synchronous_Connection_Request command is used to accept an incoming request for a synchronous connection and to inform the local Link Manager about the acceptable parameter values for the synchronous connection.
Reject Synchronous Connection Request Command	1.2	The Reject_Synchronous_Connection_Request is used to decline an incoming request for a synchronous link.
Read Voice Setting Command	1.1	The Read Voice Setting command will read the values for the Voice Setting configuration parameter, which controls all the various settings for the voice connections.
Write Voice Setting Command	1.1	The Write Voice Setting command will write the values for the Voice Setting configuration parameter, which controls all the various settings for the voice connections.

Table 3.9: Synchronous connections

3.10 CONNECTION STATE

The connection state group of commands and events allows the configuration of a link, especially for low power operation.

Name	Vers.	Summary description
Mode Change Event	1.1	The Mode Change event is used to indicate that the current mode has changed.
Max Slots Change Event	1.1	The Max Slots Change event it used to indicate a change in the max slots by the LM.
Hold Mode Command	1.1	The Hold Mode command is used to initiate Hold Mode.
Sniff Mode Command	1.1	The Sniff Mode command is used to alter the behavior of the LM and have the LM place the local or remote device into the sniff mode.
Exit Sniff Mode Command	1.1	The Exit Sniff Mode command is used to end the sniff mode for a connection handle which is currently in sniff mode.
Park State Command	1.1	The Park State command is used to alter the behavior of the LM and have the LM place the local or remote device into the park state.
Exit Park State Command	1.1	The Exit Park State command is used to switch the Bluetooth device from park state back to active mode.
Read Link Policy Settings Command	1.1	The Read Link Policy Settings command will read the Link Policy configuration parameter for the specified Connection Handle. The Link Policy settings allow the Host to specify which Link Modes the LM can use for the specified Connection Handle.
Write Link Policy Settings Command	1.1	The Write Link Policy Settings command will write the Link Policy configuration parameter for the specified Connection Handle. The Link Policy settings allow the Host to specify which Link Modes the LM can use for the specified Connection Handle.
Read Default Link Policy Settings Command	1.2	The Read Default Link Policy Settings command will read the Default Link Policy configuration parameter for all new connections.
Write Default Link Policy Settings Command	1.2	The Write Default Link Policy Settings command will write the Default Link Policy configuration parameter for all new connections.

Table 3.10: Connection state

3.11 PICONET STRUCTURE

The piconet structure group of commands and events allows the discovery and reconfiguration a piconet.

Name	Vers.	Summary description
Role Discovery Command	1.1	The Role Discovery command is used for a Bluetooth device to determine which role the device is performing for a particular Connection Handle.
Switch Role Command	1.1	The Switch Role command is used to switch master and slave roles of the devices on either side of a connection.
Role Change Event	1.1	The Role Change event is used to indicate that the current Bluetooth role related to the particular connection has been changed.

Table 3.11: Piconet structure

3.12 QUALITY OF SERVICE

The quality of service group of commands and events allows the configuration of links to allow for quality of service parameters to be specified.

Name	Vers.	Summary description
Flow Specification Command	1.2	The Flow Specification command is used to specify the flow parameters for the traffic carried over the ACL connection identified by the Connection Handle.
Flow Specification Complete Event	1.2	The Flow Specification Complete event is used to inform the Host about the Quality of Service for the ACL connection the Controller is able to support.
QoS Setup Command	1.1	The QoS Setup command is used to specify Quality of Service parameters for a connection handle.
QoS Setup Complete Event	1.1	The QoS Setup Complete event is used to indicate that QoS is setup.
QoS Violation Event	1.1	The QoS Violation event is used to indicate the Link Manager is unable to provide the current QoS requirement for the Connection Handle.
Flush Command	1.1	The Flush command is used to discard all data that is currently pending for transmission in the Controller for the specified connection handle.
Flush Occurred Event	1.1	The Flush Occurred event is used to indicate that, for the specified Connection Handle, the data to be transmitted has been discarded.
Read Automatic Flush Timeout Command	1.1	The Read Automatic Flush Timeout will read the value for the Flush Timeout configuration parameter for the specified connection handle. The Flush Timeout parameter is only used for ACL connections.
Write Automatic Flush Timeout Command	1.1	The Write Automatic Flush Timeout will write the value for the Flush Timeout configuration parameter for the specified connection handle. The Flush Timeout parameter is only used for ACL connections.
Read Failed Contact Counter Command	1.1	The Read Failed Contact Counter will read the value for the Failed Contact Counter configuration parameter for a particular connection to another device.
Reset Failed Contact Counter Command	1.1	The Reset Failed Contact Counter will reset the value for the Failed Contact Counter configuration parameter for a particular connection to another device.
Read Num Broadcast Retransmissions Command	1.1	The Read Num Broadcast Retransmissions command will read the parameter value for the Number of Broadcast Retransmissions for the device.
Write Num Broadcast Retransmissions Command	1.1	The Write Num Broadcast Retransmissions command will write the parameter value for the Number of Broadcast Retransmissions for the device.

Table 3.12: Quality of service

3.13 PHYSICAL LINKS

The physical links commands and events allows configuration of the physical link.

Name	Vers.	Summary description
Read Link Supervision Timeout Command	1.1	The Read Link Supervision Timeout command will read the value for the Link Supervision Timeout configuration parameter for the device. This parameter is used by the device to determine link loss.
Write Link Supervision Timeout Command	1.1	The Write Link Supervision Timeout command will write the value for the Link Supervision Timeout configuration parameter for the device. This parameter is used by the device to determine link loss.
Read AFH Channel Assessment Mode Command	1.2	The Read AFH Channel Assessment Mode command will read the value for the AFH Channel Classification Mode parameter. This value is used to enable or disable the Controller's channel assessment scheme.
Write AFH Channel Assessment Mode Command	1.2	The Write AFH Channel Assessment Mode command will write the value for the Channel Classification Mode configuration parameter. This value is used to enable or disable the Controller's channel assessment scheme.
Set AFH Host Channel Classification Command	1.2	The Set AFH Host Channel Classification command allows the Host to specify a channel classification based on its "local information".
Change Connection Packet Type Command	1.1	The Change Connection Packet Type command is used to change which packet types can be used for a connection that is currently established.
Connection Packet Type Changed Event	1.1	The Connection Packet Type Changed event is used to indicate the completion of the process of the Link Manager changing the packet type mask used for the specified Connection Handle.

Table 3.13: Physical links

3.14 HOST FLOW CONTROL

The Host flow control group of commands and events allows flow control to be used towards the Host.

Name	Vers.	Summary description
Host Buffer Size Command	1.1	The Host Buffer Size command is used by the Host to notify the Controller about its buffer sizes for ACL and synchronous data. The Controller will segment the data to be transmitted from the Controller to the Host, so that data contained in HCI Data Packets will not exceed these sizes.
Set Event Mask Command	1.1	The Set Event Mask command is used to control which events are generated by the HCI for the Host.
Set Event Filter Command	1.1	The Set Event Filter command is used by the Host to specify different event filters. The Host may issue this command multiple times to request various conditions for the same type of event filter and for different types of event filters.
Set Controller To Host Flow Control Command	1.1	The Set Controller To Host Flow Control command is used by the Host to turn flow control on or off in the direction from the Controller to the Host.
Host Number Of Completed Packets Command	1.1	The Host Number Of Completed Packets command is used by the Host to indicate to the Controller when the Host is ready to receive more HCI packets for any connection handle.
Data Buffer Overflow Event	1.1	The Data Buffer Overflow event is used to indicate that the Controller's data buffers have overflowed, because the Host has sent more packets than allowed.
Read Synchronous Flow Control Enable Command	1.1	The Read Synchronous Flow Control Enable command provides the ability to read the Synchronous Flow Control Enable setting. By using this setting, the Host can decide if the Controller will send Number Of Completed Packets events for Synchronous Connection Handles.
Write Synchronous Flow Control Enable Command	1.1	The Write Synchronous Flow Control Enable command provides the ability to write the Synchronous Flow Control Enable setting. By using this setting, the Host can decide if the Controller will send Number Of Completed Packets events for Synchronous Connection Handles.

Table 3.14: Controller flow control.

3.15 LINK INFORMATION

The link information group of commands and events allows information about a link to be read.

Name	Vers.	Summary description
Read LMP Handle Command	1.2	The Read LMP Handle command will read the current LMP Handle associated with the Connection Handle.
Read Transmit Power Level Command	1.1	The Read Transmit Power Level command will read the values for the Transmit Power Level parameter for the specified Connection Handle.
Read Link Quality Command	1.1	The Read Link Quality command will read the value for the Link Quality for the specified Connection Handle.
Read RSSI Command	1.1	The Read RSSI command will read the value for the Received Signal Strength Indication (RSSI) for a connection handle to another Bluetooth device.
Read Clock Offset Command	1.1	The Read Clock Offset command allows the Host to read the clock offset of remote devices.
Read Clock Offset Complete Event	1.1	The Read Clock Offset Complete event is used to indicate the completion of the process of the LM obtaining the Clock offset information.
Read Clock Command	1.2	The Read Clock command will read an estimate of a piconet or the local Bluetooth Clock.
Read AFH Channel Map Command	1.2	The Read AFH Channel Map command will read the current state of the channel map for a connection.

Table 3.15: Link information

3.16 AUTHENTICATION AND ENCRYPTION

The authentication and encryption group of commands and events allows authentication of a remote device and then encryption of the link to one or more remote devices.

Name	Vers.	Summary description
Read Authentication Enable Command	1.1	The Read Authentication Enable command will read the value for the Authentication Enable parameter, which controls whether the Bluetooth device will require authentication for each connection with other Bluetooth devices.
Write Authentication Enable Command	1,1	The Write Authentication Enable command will write the value for the Authentication Enable parameter, which controls whether the Bluetooth device will require authentication for each connection with other Bluetooth devices.
Read Encryption Mode Command	1.1	The Read Encryption Mode command will read the value for the Encryption Mode parameter, which controls whether the Bluetooth device will require encryption for each connection with other Bluetooth devices.
Write Encryption Mode Command	1.1	The Write Encryption Mode command will write the value for the Encryption Mode parameter, which controls whether the Bluetooth device will require encryption for each connection with other Bluetooth devices.
Link Key Request Event	1.1	The Link Key Request event is used to indicate that a Link Key is required for the connection with the device specified in BD_ADDR.
Link Key Request Reply Command	1.1	The Link Key Request Reply command is used to reply to a Link Key Request event from the Controller, and specifies the Link Key stored on the Host to be used as the link key for the connection with the other Bluetooth device specified by BD_ADDR.
Link Key Request Negative Reply Command	1.1	The Link Key Request Negative Reply command is used to reply to a Link Key Request event from the Controller if the Host does not have a stored Link Key for the connection with the other Bluetooth Device specified by BD_ADDR.
PIN Code Request Event	1.1	The PIN Code Request event is used to indicate that a PIN code is required to create a new link key for a connection.
PIN Code Request Reply Command	1.1	The PIN Code Request Reply command is used to reply to a PIN Code Request event from the Controller and specifies the PIN code to use for a connection.

Table 3.16: Authentication and encryption

Name	Vers.	Summary description
PIN Code Request Negative Reply Command	1.1	The PIN Code Request Negative Reply command is used to reply to a PIN Code Request event from the Controller when the Host cannot specify a PIN code to use for a connection.
Link Key Notification Event	1.1	The Link Key Notification event is used to indicate to the Host that a new Link Key has been created for the connection with the device specified in BD_ADDR.
Authentication Requested Command	1.1	The Authentication Requested command is used to establish authentication between the two devices associated with the specified Connection Handle.
Authentication Complete Event	1.1	The Authentication Complete event occurs when authentication has been completed for the specified connection.
Set Connection Encryption Command	1.1	The Set Connection Encryption command is used to enable and disable the link level encryption.
Encryption Change Event	1.1	The Encryption Change event is used to indicate that the change in the encryption has been completed for the Connection Handle specified by the Connection Handle event parameter.
Change Connection Link Key Command	1.1	The Change Connection Link Key command is used to force both devices of a connection associated to the connection handle, to generate a new link key.
Change Connection Link Key Complete Event	1.1	The Change Connection Link Key Complete event is used to indicate that the change in the Link Key for the Connection Handle specified by the Connection Handle event parameter had been completed.
Master Link Key Command	1.1	The Master Link Key command is used to force both devices of a connection associated to the connection handle to use the temporary link key of the Master device or the regular link keys.
Master Link Key Complete Event	1.1	The Master Link Key Complete event is used to indicate that the change in the temporary Link Key or in the semi-permanent link keys on the Bluetooth master side has been completed.
Read PIN Type Command	1.1	The Read PIN Type command is used for the Host to read the value that is specified to indicate whether the Host supports variable PIN or only fixed PINs.
Write PIN Type Command	1.1	The Write PIN Type command is used for the Host to specify whether the Host supports variable PIN or only fixed PINs.
Read Stored Link Key Command	1.1	The Read Stored Link Key command provides the ability to read one or more link keys stored in the Controller.

Table 3.16: Authentication and encryption

Name	Vers.	Summary description
Return Link Keys Event	1.1	The Return Link Keys event is used to return stored link keys after a Read Stored Link Key command is used.
Write Stored Link Key Command	1.1	The Write Stored Link Key command provides the ability to write one or more link keys to be stored in the Controller.
Delete Stored Link Key Command	1.1	The Delete Stored Link Key command provides the ability to remove one or more of the link keys stored in the Controller.
Create New Unit Key Command	1.1	The Create New Unit Key command is used to create a new unit key.

Table 3.16: Authentication and encryption

3.17 TESTING

The testing group of commands and events allows a device to be placed into a special testing mode to allow for testing to be performed.

Name	Vers.	Summary description
Read Loopback Mode Command	1.1	The Read Loopback Mode will read the value for the setting of the Controllers Loopback Mode. The setting of the Loopback Mode will determine the path of information.
Write Loopback Mode Command	1.1	The Write Loopback Mode will write the value for the setting of the Controllers Loopback Mode. The setting of the Loopback Mode will determine the path of information.
Loopback Command Event	1.1	The Loopback Command event is used to loop back all commands that the Host sends to the Controller with some exceptions.
Enable Device Under Test Mode Command	1.1	The Enable Device Under Test Mode command will allow the local Bluetooth module to enter test mode via LMP test commands. The Host issues this command when it wants the local device to be the DUT for the Testing scenarios as described in the Bluetooth Test Mode document.

Table 3.17: Testing

3.18 ALPHABETICAL LIST OF COMMANDS AND EVENTS

Commands/Events	Group
Accept Connection Request Command	Connection Setup
Authentication Complete Event	Authentication and Encryption
Authentication Requested Command	Authentication and Encryption
Change Connection Link Key Command	Authentication and Encryption
Change Connection Link Key Complete Event	Authentication and Encryption
Change Connection Packet Type Command	Physical Links
Command Complete Event	Generic Events
Command Status Event	Generic Events
Connection Complete Event	Connection Setup
Connection Packet Type Changed Event	Physical Links
Connection Request Event	Connection Setup
Create Connection Cancel Command	Connection Setup
Create Connection Command	Connection Setup
Create New Unit Key Command	Authentication and Encryption
Data Buffer Overflow Event	Host Flow Control
Delete Stored Link Key Command	Authentication and Encryption
Disconnect Command	Connection Setup
Disconnection Complete Event	Connection Setup
Enable Device Under Test Mode Command	Testing
Encryption Change Event	Authentication and Encryption
Exit Park State Command	Connection State
Exit Periodic Inquiry Mode Command	Device Discovery
Exit Sniff Mode Command	Connection State
Flow Specification Command	Quality of Service
Flow Specification Complete Event	Quality of Service
Flush Command	Quality of Service
Flush Occurred Event	Quality of Service
Hardware Error Eventt	Generic Events
Hold Mode Command	Connection State
Host Buffer Size Command	Host Flow Control

Table 3.18: Alphabetical list of commands and events.

Commands/Events	Group
Host Number Of Completed Packets Command	Host Flow Control
Inquiry Cancel Command	Device Discovery
Inquiry Command	Device Discovery
Inquiry Complete Event	Device Discovery
Inquiry Result Event	Device Discovery
Inquiry Result with RSSI Event	Device Discovery
Link Key Notification Event	Authentication and Encryption
Link Key Request Event	Authentication and Encryption
Link Key Request Negative Reply Command	Authentication and Encryption
Link Key Request Reply Command	Authentication and Encryption
Loopback Command Event	Testing
Master Link Key Command	Authentication and Encryption
Master Link Key Complete Event	Authentication and Encryption
Max Slots Change Event	Connection State
Mode Change Event	Connection State
Number Of Completed Packets Event	Controller Flow Control
Page Scan Repetition Mode Change Event	Connection Setup
Park State Command	Connection State
Periodic Inquiry Mode Command	Device Discovery
PIN Code Request Event	Authentication and Encryption
PIN Code Request Negative Reply Command	Authentication and Encryption
PIN Code Request Reply Command	Authentication and Encryption
QoS Setup Command	Quality of Service
QoS Setup Complete Event	Quality of Service
QoS Violation Event	Quality of Service
Read AFH Channel Assessment Mode Command	Physical Links
Read AFH Channel Map Command	Link Information
Read Authentication Enable Command	Authentication and Encryption
Read Automatic Flush Timeout Command	Quality of Service
Read BD_ADDR Command	Controller Information
Read Buffer Size Command	Controller Flow Control
Read Class of Device Command	Controller Information

Table 3.18: Alphabetical list of commands and events.

Commands/Events	Group
Read Clock Command	Link Information
Read Clock Offset Command	Link Information
Read Clock Offset Complete Event	Link Information
Read Connection Accept Timeout Command	Connection Setup
Read Current IAC LAP Command	Controller Information
Read Default Link Policy Settings Command	Connection State
Read Encryption Mode Command	Authentication and Encryption
Read Failed Contact Counter Command	Quality of Service
Read Hold Mode Activity Command	Connection State
Read Inquiry Mode Command	Device Discovery
Read Inquiry Scan Activity Command	Device Discovery
Read Inquiry Scan Type Command	Device Discovery
Read Link Policy Settings Command	Connection State
Read Link Quality Command	Link Information
Read Link Supervision Timeout Command	Physical Links
Read LMP Handle Command	Link Information
Read Local Extended Features Command	Controller Information
Read Local Name Command	Controller Configuration
Read Local Supported Commands Command	Controller Information
Read Local Supported Features Command	Controller Information
Read Local Version Information Command	Controller Information
Read Loopback Mode Command	Testing
Read Num Broadcast Retransmissions Command	Quality of Service
Read Number Of Supported IAC Command	Controller Information
Read Page Scan Activity Command	Connection Setup
Read Page Scan Type Command	Connection Setup
Read Page Timeout Command	Connection Setup
Read PIN Type Command	Authentication and Encryption
Read Remote Extended Features Command	Remote Information
Read Remote Extended Features Complete Event	Remote Information
Read Remote Supported Features Command	Remote Information
Read Remote Supported Features Complete Event	Remote Information

Table 3.18: Alphabetical list of commands and events.

Commands/Events	Group
Read Remote Version Information Command	Remote Information
Read Remote Version Information Complete Event	Remote Information
Read RSSI Command	Link Information
Read Scan Enable Command	Controller Information
Read Stored Link Key Command	Authentication and Encryption
Read Synchronous Flow Control Enable Command	Host Flow Control
Read Transmit Power Level Command	Link Information
Read Voice Setting Command	Synchronous Connections
Reject Connection Request Command	Connection Setup
Remote Name Request Cancel Command	Remote Information
Remote Name Request Command	Remote Information
Remote Name Request Complete Event	Remote Information
Reset Command	Device Setup
Reset Failed Contact Counter Command	Quality of Service
Return Link Keys Event	Authentication and Encryption
Role Change Event	Piconet Structure
Role Discovery Command	Piconet Structure
Set AFH Host Channel Classification Command	Physical Links
Set Connection Encryption Command	Authentication and Encryption
Set Controller To Host Flow Control Command	Host Flow Control
Set Event Filter Command	Host Flow Control
Set Event Mask Command	Host Flow Control
Setup Synchronous Connection Command	Synchronous Connections
Sniff Mode Command	Connection State
Switch Role Command	Piconet Structure
Synchronous Connection Changed event	Synchronous Connections
Synchronous Connection Complete Event	Synchronous Connections
Write AFH Channel Assessment Mode Command	Physical Links
Write Authentication Enable Command	Authentication and Encryption
Write Automatic Flush Timeout Command	Quality of Service
Write Class of Device Command	Controller Information
Write Connection Accept Timeout Command	Connection Setup

Table 3.18: Alphabetical list of commands and events.

Commands/Events	Group
Write Current IAC LAP Command	Controller Information
Write Default Link Policy Settings Command	Connection State
Write Encryption Mode Command	Authentication and Encryption
Write Hold Mode Activity Command	Connection State
Write Inquiry Mode Command	Device Discovery
Write Inquiry Scan Activity Command	Device Discovery
Write Inquiry Scan Type Command	Device Discovery
Write Link Policy Settings Command	Connection State
Write Link Supervision Timeout Command	Physical Links
Write Local Name Command	Controller Information
Write Loopback Mode Command	Testing
Write Num Broadcast Retransmissions Command	Quality of Service
Write Page Scan Activity Command	Connection Setup
Write Page Scan Type Command	Connection Setup
Write Page Timeout Command	Connection Setup
Write PIN Type Command	Authentication and Encryption
Write Scan Enable Command	Controller Information
Write Stored Link Key Command	Authentication and Encryption
Write Synchronous Flow Control Enable Command	Host Flow Control
Write Voice Setting Command	Synchronous Connections

Table 3.18: Alphabetical list of commands and events.

4 HCI FLOW CONTROL

Flow control shall be used in the direction from the Host to the Controller to avoid overflowing the Controller data buffers with ACL data destined for a remote device (using a connection handle) that is not responding. The Host manages the data buffers of the Controller.

4.1 HOST TO CONTROLLER DATA FLOW CONTROL

On initialization, the Host shall issue the Read Buffer Size command. Two of the return parameters of this command determine the maximum size of HCI ACL and synchronous Data Packets (excluding header) sent from the Host to the Controller. There are also two additional return parameters that specify the total number of HCI ACL and synchronous Data Packets that the Controller may have waiting for transmission in its buffers.

When there is at least one connection to another device, or when in local loop-back mode, the Controller shall use the Number Of Completed Packets event to control the flow of data from the Host. This event contains a list of connection handles and a corresponding number of HCI Data Packets that have been completed (transmitted, flushed, or looped back to the Host) since the previous time the event was returned (or since the connection was established, if the event has not been returned before for a particular connection handle).

Based on the information returned in this event, and the return parameters of the Read Buffer Size command that specify the total number of HCI ACL and synchronous Data Packets that can be stored in the Controller, the Host decides for which Connection Handles the following HCI Data Packets should be sent.

Every time it has sent an HCI Data Packet, the Host shall assume that the free buffer space for the corresponding link type (ACL, SCO or eSCO) in the Controller has decreased by one HCI Data Packet.

Each Number Of Completed Packets event received by the Host provides information about how many HCI Data Packets have been completed (transmitted or flushed) for each Connection Handle since the previous Number Of Completed Packets event was sent to the Host. It can then calculate the actual current buffer usage.

When the Controller has completed one or more HCI Data Packet(s) it shall send a Number Of Completed Packets event to the Host, until it finally reports that all the pending HCI Data Packets have been completed. The frequency at which this event is sent is manufacturer specific.

Note: The Number Of Completed Packets events will not report on synchronous connection handles if Synchronous Flow Control is disabled. (See Read/

Write Synchronous Flow Control Enable, Section 7.3.38 on page 513 and Section 7.3.39 on page 514)

For each individual Connection Handle, the data must be sent to the Controller in HCI Data Packets in the order in which it was created in the Host. The Controller shall also transmit data on the air that is received from the Host for a given Connection Handle in the same order as it is received from the Host.

Data that is received on the air from another device shall, for the corresponding Connection Handle, be sent in HCI Data Packets to the Host in the same order as it is received. This means the scheduling shall be decided separately for each Connection Handle basis. For each individual Connection Handle, the order of the data shall not be changed from the order in which the data has been created.

4.2 CONTROLLER TO HOST DATA FLOW CONTROL

In some implementations, flow control may also be necessary in the direction from the Controller to the Host. The Set Host Controller To Host Flow Control command can be used to turn flow control on or off in that direction.

On initialization, the Host uses the Host Buffer Size command to notify the Controller about the maximum size of HCI ACL and synchronous Data Packets sent from the Controller to the Host. The command also contains two additional command parameters to notify the Controller about the total number of ACL and synchronous Data Packets that can be stored in the data buffers of the Host.

The Host uses the Host Number Of Completed Packets command in exactly the same way as the Controller uses the Number Of Completed Packets event as was previously described in this section.

The Host Number Of Completed Packets command is a special command for which no command flow control is used, and which can be sent anytime there is a connection or when in local loopback mode. The command also has no event after the command has completed. This makes it possible for the flow control to work in exactly the same way in both directions, and the flow of normal commands will not be disturbed.

4.3 DISCONNECTION BEHAVIOR

When the Host receives a Disconnection Complete event, the Host shall assume that all unacknowledged HCI Data Packets that have been sent to the Controller for the returned Connection Handle have been flushed, and that the corresponding data buffers have been freed. The Controller does not have to notify the Host about this in a Number Of Completed Packets event.

If flow control is also enabled in the direction from the Controller to the Host, the Controller may, after it has sent a Disconnection Complete event, assume that the Host will flush its data buffers for the sent Connection Handle when it receives the Disconnection Complete event. The Host does not have to notify the Controller about this in a Host Number Of Completed Packets command.

4.4 COMMAND FLOW CONTROL

On initial power-on, and after a reset, the Host shall send a maximum of one outstanding HCI Command Packet until a Command Complete or Command Status event has been received.

The Command Complete and Command Status events contain a parameter called Num HCI Command Packets, which indicates the number of HCI Command Packets the Host is currently allowed to send to the Controller. The Controller may buffer one or more HCI command packets, but the Controller must start performing the commands in the order in which they are received. The Controller can start performing a command before it completes previous commands. Therefore, the commands do not always complete in the order they are started.

To indicate to the Host that the Controller is ready to receive HCI command packets, the Controller may generate a Command Complete event with the Command Opcode 0x0000, and the Num HCI Command Packets event parameter is set to 1 or more. Command Opcode 0x0000 is a NOP (No OPeration), and can be used to change the number of outstanding HCI command packets that the Host can send. The Controller may generate a Command Complete event with the Num HCI Command Packets event parameter set to zero to inform Host it must stop sending commands.

For most commands, a Command Complete event shall be sent to the Host when the Controller completes the command. Some commands are executed in the background and do not return a Command Complete event when they have been completed. Instead, the Controller shall send a Command Status event back to the Host when it has begun to execute the command. When the actions associated with the command have finished, an event that is associated with the command shall be sent by the Controller to the Host.

If the command does not begin to execute (for example, if there was a parameter error or the command is currently not allowed), the Command Status event shall be returned with the appropriate error code in the Status parameter, and the event associated with the sent command shall not be returned.

4.5 COMMAND ERROR HANDLING

If an error occurs for a command for which a Command Complete event is returned, the Return Parameters field may not contain all the return parameters specified for the command. The Status parameter, which explains the error reason and which is the first return parameter, shall always be returned. If there is a Connection Handle parameter or a BD_ADDR parameter right after the Status parameter, this parameter shall also be returned so that the Host can identify to which instance of a command the Command Complete event belongs. In this case, the Connection Handle or BD_ADDR parameter shall have exactly the same value as that in the corresponding command parameter. It is implementation specific whether more parameters will be returned in case of an error.

The above also applies to commands that have associated command specific completion events with a status parameter in their completion event, with two exceptions. The exceptions are the Connection Complete and the Synchronous Connection Complete events. On failure, for these two events only, the second parameter, Connection_Handle, is not valid and the third parameter, BD_ADDR, is valid for identification purposes. The validity of other parameters is likewise implementation specific for failed commands in this group.

Note: The BD_ADDR return parameter of the command Read BD_ADDR is not used to identify to which instance of the Read BD ADDR command the Command Complete event belongs. It is optional for the Controller to return this parameter in case of an error.

5 HCI DATA FORMATS

5.1 INTRODUCTION

The HCI provides a uniform command method of accessing the Bluetooth capabilities. The HCI Link commands provide the Host with the ability to control the link layer connections to other Bluetooth devices. These commands typically involve the Link Manager (LM) to exchange LMP commands with remote Bluetooth devices. For details see "Link Manager Protocol" on page 211 [Part C].

The HCI Policy commands are used to affect the behavior of the local and remote LM. These Policy commands provide the Host with methods of influencing how the LM manages the piconet. The Controller & Baseband, Informational, and Status commands provide the Host access to various registers in the Controller.

HCI commands may take different amounts of time to be completed. Therefore, the results of commands will be reported back to the Host in the form of an event. For example, for most HCI commands the Controller will generate the Command Complete event when a command is completed. This event contains the return parameters for the completed HCI command. For enabling the Host to detect errors on the HCI-Transport Layer, there needs to be a timeout between the transmission of the Host's command and the reception of the Controller's response (e.g. a Command Complete or Command Status event). Since the maximum response timeout is strongly dependent on the HCI-Transport Layer used, it is recommended to use a default value of one second for this timer. This amount of time is also dependent on the number of commands unprocessed in the command queue.

5.2 DATA AND PARAMETER FORMATS

- All values are in Binary and Hexadecimal Little Endian formats unless otherwise noted
- In addition, all parameters which can have negative values must use 2's complement when specifying values
- Arrayed parameters are specified using the following notation: ParameterA[i]. If more than one set of arrayed parameters are specified (e.g. ParameterA[i], ParameterB[i]), then the order of the parameters are as follows: ParameterA[0], ParameterB[0], ParameterA[1], ParameterB[1], ParameterA[2], ParameterB[2], … ParameterA[n], ParameterB[n]
- Unless noted otherwise, all parameter values are sent and received in Little Endian format (i.e. for multi-octet parameters the rightmost (Least Signification Octet) is transmitted first)
- All command and event parameters that are not-arrayed and all elements in an arrayed parameter have fixed sizes (an integer number of octets). The parameters and the size of each not arrayed parameter (or of each element in an arrayed parameter) contained in a command or an event is specified

for each command or event. The number of elements in an arrayed parameter is not fixed.

- Where bit strings are specified, the low order bit is the right hand bit, e.g. 0 is the low order bit in '10'.
- Values or parameters marked as Reserved for Future Use, shall be set to 0 unless explicitly stated otherwise on transmission, and shall be ignored on reception. Parameter values or opcodes that an implementation does not know how to interpret shall be ignored, and the operation that is being attempted shall be completed with the correct signaling. The host or controller shall not stop functioning because of receiving a reserved value.

5.3 CONNECTION HANDLES

Connection Handles are used to identify logical channels between the Host and Controller. Connection Handles are assigned by the Controller when a new logical link is created, using the Connection Complete or Synchronous Connection Complete events. Broadcast Connection Handles are handled differently, and are described below.

The first time the Host sends an HCI Data Packet with Broadcast_Flag set to 01b (active slave broadcast) or 10b (parked slave broadcast) after a power-on or a reset, the value of the Connection Handle parameter must be a value which is not currently assigned by the Host Controller. The Host must use different connection handles for active broadcast and piconet broadcast.

The Controller must then continue to use the same connection handles for each type of broadcast until a reset is made. Note: The Host Controller must not send a Connection Complete event containing a new Connection_Handle that it knows is used for broadcast.

Note: In some situations, it may happen that the Host Controller sends a Connection Complete event before having interpreted a Broadcast packet received from the Host, and that the Connection_Handles of both Connection Complete event and HCI Data packet are the same. This conflict has to be avoided as follows:

If a Connection Complete event is received containing one of the connection handles used for broadcast, the Host has to wait before sending any packets for the new connection until it receives a Number Of Completed Packets event indicating that there are no pending broadcast packets belonging to the connection handle. In addition, the Host must change the Connection_Handle used for the corresponding type of broadcast to a Connection_Handle which is currently not assigned by the Host Controller.

This Connection_Handle must then be used for all the following broadcasts of that type until a reset is performed or the same conflict situation happens again. However, this will occur very rarely.

The Host Controller must, in the above conflict case, be able to distinguish between the Broadcast message sent by the Host and the new connection made (this could be even a new synchronous link) even though the connection handles are the same.

For an HCI Data Packet sent from the Host Controller to the Host where the Broadcast_Flag is 01 or 10, the Connection_Handle parameter should contain the connection handle for the ACL connection to the master that sent the broadcast.

Note: Connection handles used for Broadcast do not identify an ACL point-to-point connection, so they must not be used in any command having a Connection_Handle parameter and they will not be returned in any event having a Connection_Handle parameter except the Number Of Completed Packets event.

5.4 EXCHANGE OF HCI-SPECIFIC INFORMATION

The Host Controller Transport Layer provides transparent exchange of HCI specific information. These transporting mechanisms provide the ability for the Host to send HCI commands, ACL data and synchronous data to the Controller. These transport mechanisms also provide the ability for the Host to receive HCI events, ACL data and synchronous data from the Controller. Since the Host Controller Transport Layer provides transparent exchange of HCI-specific information, the HCI specification specifies the format of the commands, events, and data exchange between the Host and the Controller. The next sections specify the HCI packet formats.

5.4.1 HCI Command Packet

The HCI Command Packet is used to send commands to the Controller from the Host. The format of the HCI Command Packet is shown in Figure 5.1, and the definition of each field is explained below.

The Controller must be able to accept HCI Command Packets with up to 255 bytes of data excluding the HCI Command Packet header.

Each command is assigned a 2 byte Opcode used to uniquely identify different types of commands. The Opcode parameter is divided into two fields, called the OpCode Group Field (OGF) and OpCode Command Field (OCF). The OGF occupies the upper 6 bits of the Opcode, while the OCF occupies the remaining 10 bits. The OGF of 0x3F is reserved for vendor-specific debug commands. The OGF of 0x3E is reserved for Bluetooth Logo Testing. The organization of the Opcodes allows additional information to be inferred without fully decoding the entire Opcode.

Note: the OGF composed of all ‘ones’ has been reserved for vendor-specific debug commands. These commands are vendor-specific and are used during manufacturing, for a possible method for updating firmware, and for debugging.

Note: the OGF composed of all ‘zeros’ and an OCF or all ‘zeros’ is the NOP command. This command Opcode may be used in Command Flow Control. (See Section 4.4 on page 373)

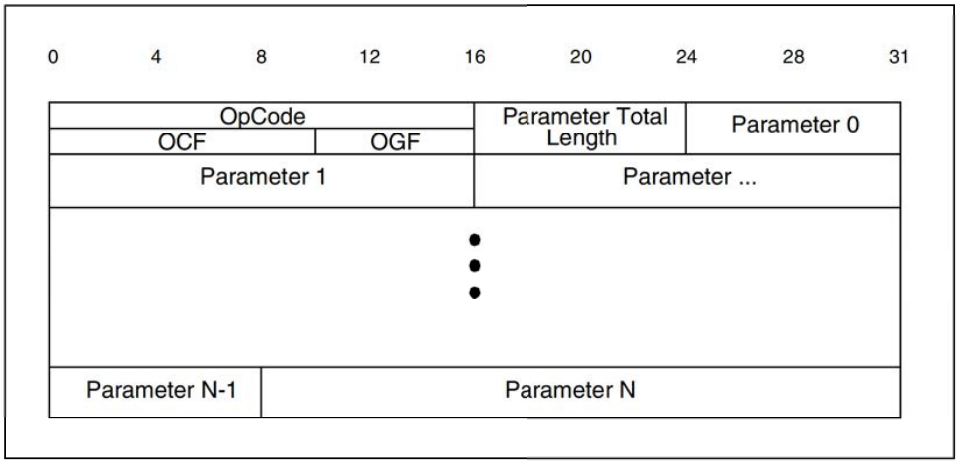


Figure 5.1: HCI Command Packet

Op_Code: *Size: 2 Octets*

Value	Parameter Description
0xXXXX	OGFRange (6 bits): 0x00-0x3F (0x3E reserved for Bluetooth logo testing and 0x3F reserved for vendor-specific debug commands) OCF Range (10 bits): 0x0000-0x03FF

Parameter_Total_Length: *Size: 1 Octet*

Value	Parameter Description
0xXX	Lengths of all of the parameters contained in this packet measured in octets. (N.B.: total length of parameters, not number of parameters)

Parameter 0 - N: *Size: Parameter Total Length*

Value	Parameter Description
0xXX	Each command has a specific number of parameters associated with it. These parameters and the size of each of the parameters are defined for each command. Each parameter is an integer number of octets in size.

5.4.2 HCI ACL Data Packets

HCI ACL Data Packets are used to exchange data between the Host and Controller. The format of the HCI ACL Data Packet is shown in Figure 5.2. The definition for each of the fields in the data packets is explained below.

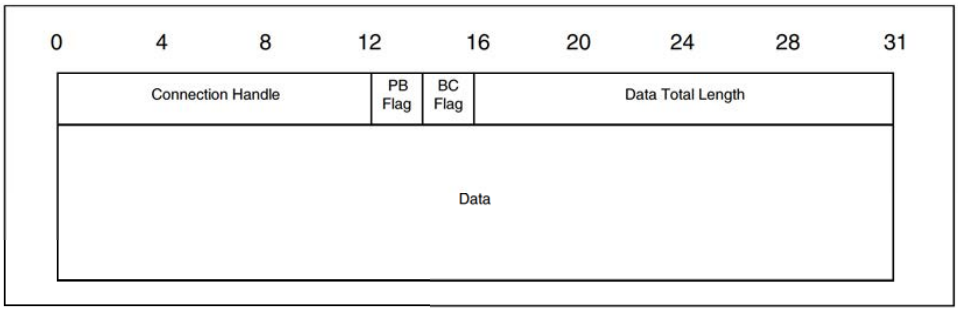


Figure 5.2: HCI ACL Data Packet

Connection_Handle: *Size: 12 Bits*

Value	Parameter Description
0xXXX	Connection Handle to be used for transmitting a data packet or segment. Range: 0x0000-0x0EFF (0x0F00 - 0x0FFF Reserved for future use)

Flags: *Size: 2 Bits*

The Flag Bits consist of the Packet_Boundary_Flag and Broadcast_Flag. The Packet_Boundary_Flag is located in bit 4 and bit 5, and the Broadcast_Flag is located in bit 6 and 7 in the second octet of the HCI ACL Data packet.

Packet_Boundary_Flag: *Size: 2 Bits*

Value	Parameter Description
00	Reserved for future use
01	Continuing fragment packet of Higher Layer Message
10	First packet of Higher Layer Message (i.e. start of an L2CAP packet)
11	Reserved for future use

Broadcast_Flag (in packet from Host to Controller): *Size: 2 Bits*

Value	Parameter Description
00	No broadcast. Only point-to-point.
01	Active Slave Broadcast: packet is sent to all active slaves (i.e. packet is usually not sent during park beacon slots), and it may be received by slaves in sniff mode or park state.
10	Parked Slave Broadcast: packet is sent to all slaves and all slaves in park state (i.e. packet is sent during park beacon slots if there are parked slaves), and it may be received by slaves in sniff mode.
11	Reserved for future use.

Broadcast_Flag (in packet from Controller to Host): *Size: 2 Bits*

Value	Parameter Description
00	Point-to-point
01	Packet received as a slave not in park state (either Active Slave Broadcast or Parked Slave Broadcast)
10	Packet received as a slave in park state (Parked Slave Broadcast)
11	Reserved for future use.

Note: active slave broadcast packets may be sent in park beacon slots.

Note: slaves in sniff mode may or may not receive a broadcast packet depending on whether they happen to be listening at sniff slots, when the packet is sent.

Data_Total_Length: *Size: 2 Octets*

Value	Parameter Description
0xXXXX	Length of data measured in octets.

5.4.3 HCI Synchronous Data Packets

HCI synchronous (SCO and eSCO) Data Packets are used to exchange synchronous data between the Host and Controller. The format of the synchronous Data Packet is shown in Figure 5.3. The definition for each of the fields in the data packets is explained below.



Figure 5.3: HCI Synchronous Data Packet

Connection_Handle: *Size: 12 Bits*

Value	Parameter Description
0xXXX	Connection handle to be used to for transmitting a synchronous data packet or segment. Range: 0x0000-0x0EFF (0x0F00- 0x0FFF Reserved for future use)

The Reserved Bits consist of four bits which are located from bit 4 to bit 7 in the second octet of the HCI Synchronous Data packet.

Reserved: *Size: 4 Bits*

Value	Parameter Description
XXXX	Reserved for future use.

Data_Total_Length: *Size: 1 Octet*

Value	Parameter Description
0xXX	Length of synchronous data measured in octets

5.4.4 HCI Event Packet

The HCI Event Packet is used by the Controller to notify the Host when events occur. The Host must be able to accept HCI Event Packets with up to 255 octets of data excluding the HCI Event Packet header. The format of the HCI Event Packet is shown in Figure 5.4, and the definition of each field is explained below.

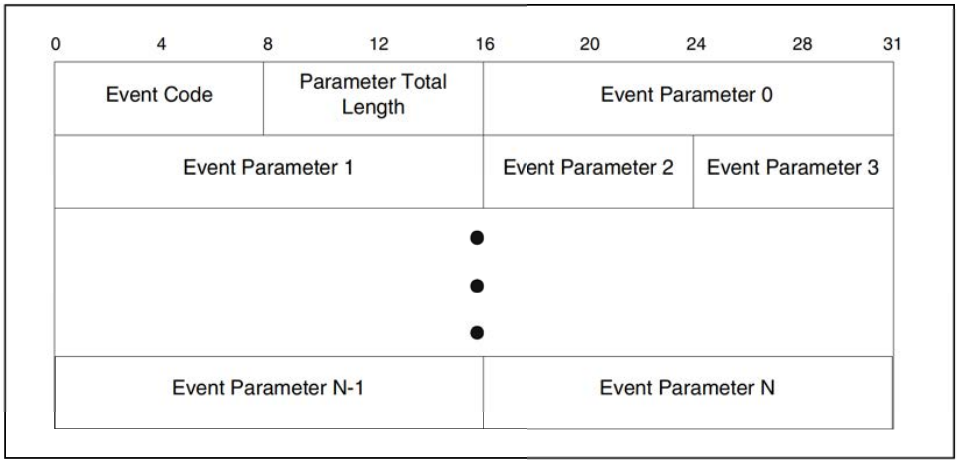


Figure 5.4: HCI Event Packet

Event_Code: *Size: 1 Octet*

Value	Parameter Description
0xXX	Each event is assigned a 1-Octet event code used to uniquely identify different types of events. Range: 0x00-0xFF (The event code 0xFF is reserved for the event code used for vendor-specific debug events. In addition, the event code 0xFE is also reserved for Bluetooth Logo Testing)

Parameter_Total_Length: *Size: 1 Octet*

Value	Parameter Description
0xXX	Length of all of the parameters contained in this packet, measured in octets

Event_Parameter 0 - N: *Size: Parameter Total Length*

Value	Parameter Description
0xXX	Each event has a specific number of parameters associated with it. These parameters and the size of each of the parameters are defined for each event. Each parameter is an integer number of octets in size.

6 HCI CONFIGURATION PARAMETERS

6.1 SCAN ENABLE

The Scan_Enable parameter controls whether or not the Bluetooth device will periodically scan for page attempts and/or inquiry requests from other Bluetooth devices. If Page_Scan is enabled, then the device will enter page scan mode based on the value of the Page_Scan_Interval and Page_Scan_Window parameters. If Inquiry_Scan is enabled, then the device will enter Inquiry Scan mode based on the value of the Inquiry_Scan_Interval and Inquiry_Scan_Window parameters.

Value	Parameter Description
0x00	No Scans enabled.
0x01	Inquiry Scan enabled. Page Scan always disabled.
0x02	Inquiry Scan disabled. Page Scan enabled.
0x03	Inquiry Scan enabled. Page Scan enabled.
0x04-0xFF	Reserved

6.2 INQUIRY SCAN INTERVAL

The Inquiry_Scan_Interval configuration parameter defines the amount of time between consecutive inquiry scans. This is defined as the time interval from when the Controller started its last inquiry scan until it begins the next inquiry scan.

Value	Parameter Description
N = 0xXXXX	Size: 2 Octets Range: 0x0012 to 0x1000; only even values are valid Default: 0x1000 Mandatory Range: 0x0012 to 0x1000 Time = N * 0.625 msec Time Range: 11.25 to 2560 msec Time Default: 2.56 sec

6.3 INQUIRY SCAN WINDOW

The Inquiry_Scan_Window configuration parameter defines the amount of time for the duration of the inquiry scan. The Inquiry_Scan_Window can only be less than or equal to the Inquiry_Scan_Interval.

Value	Parameter Description
N = 0xXXXX	Size: 2 Octets Range: 0x0011 to 0x1000 Default: 0x0012 Mandatory Range: 0x0011 to Inquiry Scan Interval Time = N * 0.625 msec Time Range: 10.625 msec to 2560 msec Time Default: 11.25 msec

6.4 INQUIRY SCAN TYPE

The Inquiry_Scan_Type configuration parameter indicates whether inquiry scanning will be done using non-interlaced scan or interlaced scan. Currently, one mandatory inquiry scan type and one optional inquiry scan type are defined. For details, see the Baseband Specification, "Inquiry scan substate" on page 164 [Part B].

Value	Parameter Description
0x00	Mandatory: Standard Scan (default)
0x01	Optional: Interlaced Scan
0x02-0xFF	Reserved

6.5 INQUIRY MODE

The Inquiry_Mode configuration parameter indicates whether inquiry returns Inquiry Result events in the standard format or with RSSI.

Value	Parameter Description
0x00	Standard Inquiry Result event format
0x01	Inquiry Result format with RSSI
0x02-0xFF	Reserved

6.6 PAGE TIMEOUT

The Page_Timeout configuration parameter defines the maximum time the local Link Manager will wait for a baseband page response from the remote device at a locally initiated connection attempt. If this time expires and the remote device has not responded to the page at baseband level, the connection attempt will be considered to have failed.

Value	Parameter Description
N = 0xXXXX	Size: 2 Octets Range: 0x0001 to 0xFFFF Default: 0x2000 Mandatory Range: 0x0016 to 0xFFFF Time = N * 0.625 msec Time Range: 0.625 msec to 40.9 sec Time Default: 5.12 sec

6.7 CONNECTION ACCEPT TIMEOUT

The Connection_Accept_Timeout configuration parameter allows the Bluetooth hardware to automatically deny a connection request after a specified time period has occurred and the new connection is not accepted. The parameter defines the time duration from when the Controller sends a Connection Request event until the Controller will automatically reject an incoming connection.

Value	Parameter Description
N = 0xXXXX	Size: 2 Octets Range: 0x0001 to 0xB540 Default: 0x1F40 Mandatory Range: 0x00A0 to 0xB540 Time = N * 0.625 msec Time Range: 0.625 msec to 29 sec Time Default: 5 sec

6.8 PAGE SCAN INTERVAL

The Page_Scan_Interval configuration parameter defines the amount of time between consecutive page scans. This time interval is defined from when the Controller started its last page scan until it begins the next page scan.

Value	Parameter Description
N = 0xXXXX	Size: 2 Octets Range: 0x0012 to 0x1000; only even values are valid Default: 0x0800 Mandatory Range: 0x0012 to 0x1000 Time = N * 0.625 msec Time Range: 11.25 msec to 2560 msec Time Default: 1.28 sec

6.9 PAGE SCAN WINDOW

The Page_Scan_Window configuration parameter defines the amount of time for the duration of the page scan. The Page_Scan_Window can only be less than or equal to the Page_Scan_Interval.

Value	Parameter Description
N = 0xXXXX	Size: 2 Octets Range: 0x0011 to 0x1000 Default: 0x0012 Mandatory Range: 0x0011 to Page Scan Interval Time = N * 0.625 msec Time Range: 10.625 msec to Page Scan Interval Time Default: 11.25 msec

6.10 PAGE SCAN PERIOD MODE (DEPRECATED)

Every time an inquiry response message is sent, the Bluetooth device will start a timer (T_mandatory_pscan), the value of which is dependent on the Page_Scan_Period_Mode. As long as this timer has not expired, the Bluetooth device will use the mandatory page scan mode for all following page scans. Note: the timer T_mandatory_pscan will be reset at each new inquiry response. For details see the "Baseband Specification" on page 55 [Part B]. (Keyword: SP-Mode, FHS-Packet, T_mandatory_pscan, Inquiry-Response).

Value	Parameter Description
0x00	P0
0x01	P1
0x02	P2
0x03-0xFF	Reserved.

6.11 PAGE SCAN TYPE

The Page_Scan_Type parameter indicates whether inquiry scanning will be done using non-interlaced scan or interlaced scan. For details, see the Baseband Specification, "Page scan substate" on page 154 [Part B].

Value	Parameter Description
0x00	Mandatory: Standard Scan (default)
0x01	Optional: Interlaced Scan
0x02-0xFF	Reserved

6.12 VOICE SETTING

The Voice_Setting parameter controls all the various settings for voice connections. The input settings apply to all voice connections, and **cannot** be set for individual voice connections. The Voice_Setting parameter controls the configuration for voice connections: Input Coding, Air coding format, input data format, Input sample size, and linear PCM parameter.The air coding format bits in the Voice_Setting command parameter specify which air coding format the local device requests. The air coding format bits do not specify which air coding format(s) the local device accepts when a remote device requests an air coding format. This is determined by the hardware capabilities of the local device.

Value	Parameter Description
00XXXXXXXX	Input Coding: Linear
01XXXXXXXX	Input Coding: μ-law Input Coding
10XXXXXXXX	Input Coding: A-law Input Coding
11XXXXXXXX	Reserved for Future Use
XX00XXXXXX	Input Data Format: 1's complement
XX01XXXXXX	Input Data Format: 2's complement
XX10XXXXXX	Input Data Format: Sign-Magnitude
XX11XXXXXX	Input Data Format: Unsigned
XXXX0XXXXX	Input Sample Size: 8-bit (only for linear PCM)
XXXX1XXXXX	Input Sample Size: 16-bit (only for linear PCM)
XXXXXnnnXX	Linear_PCM_Bit_Pos: # bit positions that MSB of sample is away from starting at MSB (only for Linear PCM).
XXXXXXXX00	Air Coding Format: CVSD
XXXXXXXX01	Air Coding Format: μ-law
XXXXXXXX10	Air Coding Format: A-law
XXXXXXXX11	Air Coding Format: Transparent Data

6.13 PIN TYPE

The PIN Type configuration parameter determines whether the Link Manager assumes that the Host supports variable PIN codes or a fixed PIN code. The host controller uses the PIN-type information during pairing.

Value	Parameter Description
0x00	Variable PIN.
0x01	Fixed PIN.

6.14 LINK KEY

The Controller can store a limited number of link keys for other Bluetooth devices. Link keys are shared between two Bluetooth devices, and are used for all security transactions between the two devices. A Host device may have additional storage capabilities, which can be used to save additional link keys to be reloaded to the Bluetooth Controller when needed. A Link Key is associated with a BD_ADDR.

Value	Parameter Description
0xXXXXXXXXXX XXXXXXXXXX XXXXXXXXXX	Link Key for an associated BD_ADDR.

6.15 AUTHENTICATION ENABLE

The Authentication_Enable parameter controls if the local device requires to authenticate the remote device at connection setup (between the Create_Connection command or acceptance of an incoming ACL connection and the corresponding Connection Complete event). At connection setup, only the device(s) with the Authentication_Enable parameter enabled will try to authenticate the other device.

Note: Changing this parameter does not affect existing connections.

Value	Parameter Description
0x00	Authentication not required.
0x01	Authentication required for all connections.
0x02-0xFF	Reserved

6.16 ENCRYPTION MODE

The Encryption_Mode parameter controls if the local device requires encryption to the remote device at connection setup (between the Create_Connection command or acceptance of an incoming ACL connection and the corresponding Connection Complete event). At connection setup, only devices with the Authentication_Enabled configuration parameter set to required and the Encryption_Mode configuration parameter set to required will try to encrypt the physical link to the other device.

Note: Changing this parameter does not affect existing connections.

A temporary link key is used when both broadcast and point-to-point traffic are encrypted.

The Host must not specify the Encryption_Mode parameter with more encryption capability than its local device currently supports, although the parameter is used to request the encryption capability to the remote device. Note that the Host must not request the command with the Encryption_Mode parameter set to 0x01, when the local device does not support encryption.

Note: for encryption to be used, both devices must support and enable encryption.

Value	Parameter Description
0x00	Encryption not required.
0x01	Encryption required for all connections.
0x02-0xFF	Reserved.

Note: in the Connection Complete event the Encryption_Mode parameter will show whether encryption was successfully turned on. The remote device may not support encryption or may have set Encryption_Mode to 0x01 when the local device has not, so the encryption mode returned in the Connection Complete event may not equal the encryption mode set in the HCI_Write_Encryption_Mode Command.

6.17 FAILED CONTACT COUNTER

The Failed_Contact_Counter records the number of consecutive incidents in which either the slave or master didn't respond after the flush timeout had expired, and the L2CAP packet that was currently being transmitted was automatically 'flushed'. When this occurs, the Failed_Contact_Counter is incremented by 1. The Failed_Contact_
Counter for a connection is reset to zero on the following conditions:

1. When a new connection is established
2. When the Failed_Contact_Counter is > zero and an L2CAP packet is acknowledged for that connection
3. When the Reset_Failed_Contact_Counter command has been issued

Value	Parameter Description
0xXXXX	Number of consecutive failed contacts for a connection corresponding to the connection handle.

6.18 HOLD MODE ACTIVITY

The Hold_Mode_Activity value is used to determine what activities should be suspended when the device is in hold mode. After the hold period has expired, the device will return to the previous mode of operation. Multiple hold mode activities may be specified for the Hold_Mode_Activity parameter by performing a bitwise OR operation of the different activity types. If no activities are suspended, then all of the current Periodic Inquiry, Inquiry Scan, and Page Scan settings remain valid during the Hold Mode. If the Hold_Mode_Activity parameter is set to Suspend Page Scan, Suspend Inquiry Scan, and Suspend Periodic Inquiries, then the device can enter a low-power state during the Hold Mode period, and all activities are suspended. Suspending multiple activities can be specified for the Hold_Mode_Activity parameter by performing a bitwise OR operation of the different activity types.The Hold Mode Activity is only valid if all connections are in Hold Mode.

Value	Parameter Description
0x00	Maintain current Power State.
0x01	Suspend Page Scan.
0x02	Suspend Inquiry Scan.
0x04	Suspend Periodic Inquiries.
0x08-0xFF	Reserved for Future Use.

6.19 LINK POLICY SETTINGS

The Link_Policy_Settings parameter determines the behavior of the local Link Manager when it receives a request from a remote device or it determines itself to change the master-slave role or to enter park state, hold, or sniff mode. The local Link Manager will automatically accept or reject such a request from the remote device, and may even autonomously request itself, depending on the value of the Link_Policy_Settings parameter for the corresponding Connection_Handle. When the value of the Link_Policy_Settings parameter is changed for a certain Connection_Handle, the new value will only be used for requests from a remote device or from the local Link Manager itself made after this command has been completed. By enabling each mode individually, the Host can choose any combination needed to support various modes of operation. Multiple LM policies may be specified for the Link_Policy_Settings parameter by performing a bitwise OR operation of the different activity types.

Note: The local device may be forced into hold mode (regardless of whether the local device is master or slave) by the remote device regardless of the value of the Link_Policy_Settings parameter. The forcing of hold mode can however only be done once the connection has already been placed into hold mode through an LMP request (the Link_Policy_Settings determine if requests from a remote device should be accepted or rejected). The forcing of hold mode can after that be done as long as the connection lasts regardless of the setting for hold mode in the Link_Policy_Settings parameter.

Note that the previous description implies that if the implementation in the remote device is a "polite" implementation that does not force another device into hold mode via LMP PDUs, then the Link_Policy_Settings will never be overruled.

Value	Parameter Description
0x0000	Disable All LM Modes Default
0x0001	Enable Role switch.
0x0002	Enable Hold Mode.
0x0004	Enable Sniff Mode.
0x0008	Enable Park State.
0x0010 – 0x8000	Reserved for Future Use.

6.20 FLUSH TIMEOUT

The Flush_Timeout configuration parameter is used for ACL connections only. The Flush Timeout is defined in the Baseband specification Section 7.6.3, "Flushing payloads," on page 150. This parameter allows ACL packets to be automatically flushed without the Host device issuing a Flush command. This provides support for isochronous data, such as audio. When the L2CAP packet that is currently being transmitted is automatically 'flushed', the Failed Contact Counter is incremented by one.

Value	Parameter Description
0	Timeout = ∞; No Automatic Flush
N = 0xXXXX	Size: 2 Octets Range: 0x0001 to 0x07FF Mandatory Range: 0x0002 to 0x07FF Time = N * 0.625 msec Time Range: 0.625 msec to 1279.375 msec

6.21 NUM BROADCAST RETRANSMISSIONS

Broadcast packets are not acknowledged and are unreliable. The Number of Broadcast Retransmissions parameter, N, is used to increase the reliability of a broadcast message by retransmitting the broadcast message multiple times. This sets the value N_{BC} in the baseband to one greater than the Num Broadcast Retransmissions value. (See Baseband Specification, Section 7.6.5, on page 150) This parameter should be adjusted as the link quality measurement changes.

Value	Parameter Description
N = 0xXX	N_{BC} = N + 1 Range 0x00-0xFE

6.22 LINK SUPERVISION TIMEOUT

The Link_Supervision_Timeout parameter is used by the master or slave Bluetooth device to monitor link loss. If, for any reason, no Baseband packets are received from that Connection Handle for a duration longer than the Link_Supervision_Timeout, the connection is disconnected. The same timeout value is used for both synchronous and ACL connections for the device specified by the Connection Handle.

Note: Setting the Link_Supervision_Timeout to No Link_Supervision_Timeout (0x0000) will disable the Link_Supervision_Timeout check for the specified Connection Handle. This makes it unnecessary for the master of the piconet to unpark and then park each Bluetooth Device every ~40 seconds. By using the No Link_Supervision_Timeout setting, the scalability of the Park state is not limited.

Value	Parameter Description
0x0000	No Link_Supervision_Timeout.
N = 0xXXXX	Size: 2 Octets Range: 0x0001 to 0xFFFF Default: 0x7D00 Mandatory Range: 0x0190 to 0xFFFF Time = N * 0.625 msec Time Range: 0.625 msec to 40.9 sec Time Default: 20 sec

6.23 SYNCHRONOUS FLOW CONTROL ENABLE

The Synchronous Flow Control Enable configuration parameter allows the Host to decide if the Controller will send Number Of Completed Packets events for synchronous Connection Handles. This setting allows the Host to enable and disable synchronous flow control.

Value	Parameter Description
0x00	Synchronous Flow Control is disabled. No Number of Completed Packets events will be sent from the Controller for synchronous Connection Handles.
0x01	Synchronous Flow Control is enabled. Number of Completed Packets events will be sent from the Controller for synchronous Connection Handles.

6.24 LOCAL NAME

The user-friendly Local Name provides the user the ability to distinguish one Bluetooth device from another. The Local Name configuration parameter is a UTF-8 encoded string with up to 248 octets in length. The Local Name configuration parameter will be null terminated (0x00) if the UTF-8 encoded string is less than 248 octets.

Note: the Local Name configuration parameter is a string parameter. Endianess does therefore not apply to the Local Name configuration parameter. The first octet of the name is received first.

Value	Parameter Description
	A UTF-8 encoded User Friendly Descriptive Name for the device. If the name contained in the parameter is shorter than 248 octets, the end of the name is indicated by a NULL octet (0x00), and the following octets (to fill up 248 octets, which is the length of the parameter) do not have valid values.

6.25 CLASS OF DEVICE

The Class_of_Device parameter is used to indicate the capabilities of the local device to other devices.

Value	Parameter Description
0xXXXXXX	Class of Device for the device.

6.26 SUPPORTED COMMANDS

The Supported Commands configuration parameter lists which HCI commands the local controller supports. It is implied that if a command is listed as supported, the feature underlying that command is also supported.

The Supported Commands is a 64 octet bit field. If a bit is set to 1, then this command is supported.

Octet	Bit	Command Supported
0	0	Inquiry
	1	Inquiry Cancel
	2	Periodic Inquiry Mode
	3	Exit Periodic Inquiry Mode
	4	Create Connection
	5	Disconnect
	6	Add SCO Connection
	7	Cancel Create Connection
1	0	Accept Connection Request
	1	Reject Connection Request
	2	Link Key Request Reply
	3	Link Key Request Negative Reply
	4	PIN Code Request Reply
	5	PIN Code Request Negative Reply
	6	Change Connection Packet Type
	7	Authentication Request
2	0	Set Connection Encryption
	1	Change Connection Link Key
	2	Master Link Key
	3	Remote Name Request
	4	Cancel Remote Name Request
	5	Read Remote Supported Features
	6	Read Remote Extended Features
	7	Read Remote Version Information
3	0	Read Clock Offset
	1	Read LMP Handle
	2	Reserved
	3	Reserved
	4	Reserved
	5	Reserved
	6	Reserved
	7	Reserved

Octet	Bit	Command Supported
4	0	Reserved
	1	Hold Mode
	2	Sniff Mode
	3	Exit Sniff Mode
	4	Park State
	5	Exit Park State
	6	QoS Setup
	7	Role Discovery
5	0	Switch Role
	1	Read Link Policy Settings
	2	Write Link Policy Settings
	3	Read Default Link Policy Settings
	4	Write Default Link Policy Settings
	5	Flow Specification
	6	Set Event Mark
	7	Reset
6	0	Set Event Filter
	1	Flush
	2	Read PIN Type
	3	Write PIN Type
	4	Create New Unit Key
	5	Read Stored Link Key
	6	Write Stored Link Key
	7	Delete Stored Link Key
7	0	Write Local Name
	1	Read Local Name
	2	Read Connection Accept Timeout
	3	Write Connection Accept Timeout
	4	Read Page Timeout
	5	Write Page Timeout
	6	Read Scan Enable
	7	Write Scan Enable
8	0	Read Page Scan Activity
	1	Write Page Scan Activity
	2	Read Inquiry Scan Activity
	3	Write Inquiry Scan Activity
	4	Read Authentication Enable
	5	Write Authentication Enable
	6	Read Encryption Mode
	7	Write Encryption Mode

Octet	Bit	Command Supported
9	0	Read Class Of Device
	1	Write Class Of Device
	2	Read Voice Setting
	3	Write Voice Setting
	4	Read Automatic Flush Timeout
	5	Write Automatic Flush Timeout
	6	Read Num Broadcast Retransmissions
	7	Write Num Broadcast Retransmissions
10	0	Read Hold Mode Activity
	1	Write Hold Mode Activity
	2	Read Transmit Power Level
	3	Read Synchronous Flow Control Enable
	4	Write Synchronous Flow Control Enable
	5	Set Host Controller To Host Flow Control
	6	Host Buffer Size
	7	Host Number Of Completed Packets
11	0	Read Link Supervision Timeout
	1	Write Link Supervision Timeout
	2	Read Number of Supported IAC
	3	Read Current IAC LAP
	4	Write Current IAC LAP
	5	Reserved
	6	Reserved
	7	Read Page Scan Mode
12	0	Write Page Scan Mode
	1	Set AFH Channel Classification
	2	reserved
	3	reserved
	4	Read Inquiry Scan Type
	5	Write Inquiry Scan Type
	6	Read Inquiry Mode
	7	Write Inquiry Mode
13	0	Read Page Scan Type
	1	Write Page Scan Type
	2	Read AFH Channel Assessment Mode
	3	Write AFH Channel Assessment Mode
	4	Reserved
	5	Reserved
	6	Reserved
	7	Reserved

Octet	Bit	Command Supported
14	0	Reserved
	1	Reserved
	2	Reserved
	3	Read Local Version Information
	4	Reserved
	5	Read Local Supported Features
	6	Read Local Extended Features
	7	Read Buffer Size
15	0	Read Country Code
	1	Read BD ADDR
	2	Read Failed Contact Count
	3	Reset Failed Contact Count
	4	Get Link Quality
	5	Read RSSI
	6	Read AFH Channel Map
	7	Read BD Clock
16	0	Read Loopback Mode
	1	Write Loopback Mode
	2	Enable Device Under Test Mode
	3	Setup Synchronous Connection
	4	Accept Synchronous Connection
	5	Reject Synchronous Connection[
	6	Reserved
	7	Reserved

7 HCI COMMANDS AND EVENTS

7.1 LINK CONTROL COMMANDS

The Link Control commands allow the Controller to control connections to other Bluetooth devices. When the Link Control commands are used, the Link Manager (LM) controls how the Bluetooth piconets and scatternets are established and maintained. These commands instruct the LM to create and modify link layer connections with Bluetooth remote devices, perform Inquiries of other Bluetooth devices in range, and other LMP commands. For the Link Control commands, the OGF is defined as 0x01.

7.1.1 Inquiry Command

Command	OCF	Command Parameters	Return Parameters
HCI_Inquiry	0x0001	LAP, Inquiry_Length, Num_Responses	

Description:

This command will cause the Bluetooth device to enter Inquiry Mode. Inquiry Mode is used to discover other nearby Bluetooth devices. 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. The Num_Responses parameter specifies the number of responses that can be received before the Inquiry is halted. A Command Status event is sent from the Controller to the Host when the Inquiry command has been started by the Bluetooth device. When the Inquiry process is completed, the Controller will send an Inquiry Complete event to the Host indicating that the Inquiry has finished. The event parameters of Inquiry Complete event will have a summary of the result from the Inquiry process, which reports the number of nearby Bluetooth devices that responded. When a Bluetooth device responds to the Inquiry message, an Inquiry Result event will occur to notify the Host of the discovery.

A device which responds during an inquiry or inquiry period should always be reported to the Host in an Inquiry Result event if the device has not been reported earlier during the current inquiry or inquiry period and the device has not been filtered out using the command Set_Event_Filter. If the device has been reported earlier during the current inquiry or inquiry period, it may or may not be reported depending on the implementation (depending on if earlier results have been saved in the Controller and in that case how many responses that have been saved). It is recommended that the Controller tries to report a particular device only once during an inquiry or inquiry period.

Command Parameters:

LAP: *Size: 3 Octets*

Value	Parameter Description
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" (https://www.bluetooth.org/foundry/assignnumb/document/assigned_numbers).

Inquiry_Length: *Size: 1 Octet*

Value	Parameter Description
N = 0xXX	Maximum amount of time specified before the Inquiry is halted. Size: 1 octet Range: 0x01 – 0x30 Time = N * 1.28 sec Range: 1.28 – 61.44 Sec

Num_Responses: *Size: 1 Octet*

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

Return Parameters:

None.

Event(s) generated (unless masked away):

A Command Status event is sent from the Controller to the Host when the Controller has started the Inquiry process. An Inquiry Result event will be created for each Bluetooth device which responds to the Inquiry message. In addition, multiple Bluetooth devices which respond to the Inquire message may be combined into the same event. An Inquiry Complete event is generated when the Inquiry process has completed.
Note: no Command Complete event will be sent by the Controller to indicate that this command has been completed. Instead, the Inquiry Complete event will indicate that this command has been completed. No Inquiry Complete event will be generated for the canceled Inquiry process.

7.1.2 Inquiry Cancel Command

Command	OCF	Command Parameters	Return Parameters
HCI_Inquiry_Cancel	0x0002		Status

Description:

This command will cause the Bluetooth device to stop the current Inquiry if the Bluetooth device is in Inquiry Mode. This command allows the Host to interrupt the Bluetooth device and request the Bluetooth device to perform a different task. The command should only be issued after the Inquiry command has been issued, a Command Status event has been received for the Inquiry command, and before the Inquiry Complete event occurs.

Return Parameters:

Status: *Size: 1 Octet*

Value	Parameter Description
0x00	Inquiry_Cancel command succeeded.
0x01-0xFF	Inquiry_Cancel command failed. See "Error Codes" on page 319 [Part D] for list of Error Codes.

Event(s) generated (unless masked away):

When the Inquiry Cancel command has completed, a Command Complete event will be generated. No Inquiry Complete event will be generated for the canceled Inquiry process.

7.1.3 Periodic Inquiry Mode Command

Command	OCF	Command Parameters	Return Parameters
HCI_Periodic_ Inquiry_Mode	0x0003	Max_Period_Length, Min_Period_Length, LAP, Inquiry_Length, Num_Responses	Status

Description:

The Periodic_Inquiry_Mode command is used to configure the Bluetooth device to enter the Periodic Inquiry Mode that performs an automatic Inquiry. Max_Period_Length and Min_Period_Length define the time range between two consecutive inquiries, from the beginning of an inquiry until the start of the next inquiry. The Controller will use this range to determine a new random time between two consecutive inquiries for each Inquiry. 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 InquiryMode and, when time expires, Inquiry will be halted. The Num_Responses parameter specifies the number of responses that can be received before the Inquiry is halted. This command is completed when the Inquiry process has been started by the Bluetooth device, and a Command Complete event is sent from the Controller to the Host. When each of the periodic Inquiry processes are completed, the Controller will send an Inquiry Complete event to the Host indicating that the latest periodic Inquiry process has finished. When a Bluetooth device responds to the Inquiry message an Inquiry Result event will occur to notify the Host of the discovery.
Note: Max_Period_Length > Min_Period_Length > Inquiry_Length

A device which responds during an inquiry or inquiry period should always be reported to the Host in an Inquiry Result event if the device has not been reported earlier during the current inquiry or inquiry period and the device has not been filtered out using the command Set_Event_Filter. If the device has been reported earlier during the current inquiry or inquiry period, it may or may not be reported depending on the implementation (depending on if earlier results have been saved in the Controller and in that case how many responses that have been saved). It is recommended that the Controller tries to report a particular device only once during an inquiry or inquiry period.

Command Parameters:

Max_Period_Length: *Size: 2 Octets*

Value	Parameter Description
N = 0xXXXX	Maximum amount of time specified between consecutive inquiries. Size: 2 octets Range: 0x03 – 0xFFFF Time = N * 1.28 sec Range: 3.84 – 83884.8 Sec 0.0 – 23.3 hours

Min_Period_Length: *Size: 2 Octets*

Value	Parameter Description
N = 0xXXXX	Minimum amount of time specified between consecutive inquiries. Size: 2 octets Range: 0x02 – 0xFFFE Time = N * 1.28 sec Range: 2.56 – 83883.52 Sec 0.0 – 23.3 hours

LAP: *Size: 3 Octets*

Value	Parameter Description
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" (https://www.bluetooth.org/foundry/assignnumb/document/assigned_numbers).

Inquiry_Length: *Size: 1 Octet*

Value	Parameter Description
N = 0xXX	Maximum amount of time specified before the Inquiry is halted. Size: 1 octet Range: 0x01 – 0x30 Time = N * 1.28 sec Range: 1.28 – 61.44 Sec

Num_Responses: *Size: 1 Octet*

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

Return Parameters:

Status: *Size: 1 Octet*

Value	Parameter Description
0x00	Periodic Inquiry Mode command succeeded.
0x01-0xFF	Periodic Inquiry Mode command failed. See "Error Codes" on page 319 [Part D] for list of Error Codes.

Event(s) generated (unless masked away):

The Periodic Inquiry Mode begins when the Controller sends the Command Complete event for this command to the Host. An Inquiry Result event will be created for each Bluetooth device which responds to the Inquiry message. In addition, multiple Bluetooth devices which response to the Inquiry message may be combined into the same event. An Inquiry Complete event is generated when each of the periodic Inquiry processes has completed. No Inquiry Complete event will be generated for the canceled Inquiry process.

7.1.4 Exit Periodic Inquiry Mode Command

Command	OCF	Command Parameters	Return Parameters
HCI_Exit_Periodic_Inquiry_Mode	0x0004		Status

Description:

The Exit Periodic Inquiry Mode command is used to end the Periodic Inquiry mode when the local device is in Periodic Inquiry Mode. If the local device is currently in an Inquiry process, the Inquiry process will be stopped directly and the Controller will no longer perform periodic inquiries until the Periodic Inquiry Mode command is reissued.

Command Parameters:

None.

Return Parameters:

Status: *Size: 1 Octet*

Value	Parameter Description
0x00	Exit Periodic Inquiry Mode command succeeded.
0x01-0xFF	Exit Periodic Inquiry Mode command failed. See "Error Codes" on page 319 [Part D] for list of Error Codes.

Event(s) generated (unless masked away):

A Command Complete event for this command will occur when the local device is no longer in Periodic Inquiry Mode. No Inquiry Complete event will be generated for the canceled Inquiry process.

7.1.5 Create Connection Command

Command	OCF	Command Parameters	Return Parameters
HCI_Create_Connection	0x0005	BD_ADDR, Packet_Type, Page_Scan_Repetition_Mode, Reserved, Clock_Offset, Allow_Role_Switch	

Description:

This command will cause the Link Manager to create a connection to the Bluetooth device with the BD_ADDR specified by the command parameters. This command causes the local Bluetooth device to begin the Page process to create a link level connection. The Link Manager will determine how the new ACL connection is established. This ACL connection is determined by the current state of the device, its piconet, and the state of the device to be connected. The Packet_Type command parameter specifies which packet types the Link Manager shall use for the ACL connection. When sending HCI ACL Data Packets the Link Manager shall only use the packet type(s) specified by the Packet_Type command parameter or the always-allowed DM1 packet type. Multiple packet types may be specified for the Packet Type parameter by performing a bit-wise OR operation of the different packet types. The Link Manager may choose which packet type to be used from the list of acceptable packet types. The Page_Scan_Repetition_Mode parameter specifies the page scan repetition mode supported by the remote device with the BD_ADDR. This is the information that was acquired during the inquiry process. The Clock_Offset parameter is the difference between its own clock and the clock of the remote device with BD_ADDR. Only bits 2 through 16 of the difference are used, and they are mapped to this parameter as bits 0 through 14 respectively. A Clock_Offset_Valid_Flag, located in bit 15 of the Clock_Offset parameter, is used to indicate if the Clock Offset is valid or not. A Connection handle for this connection is returned in the Connection Complete event (see below). The Allow_Role_Switch parameter specifies if the local device accepts or rejects the request of a master-slave role switch when the remote device requests it at the connection setup (in the Role parameter of the Accept_Connection_Request command) (before the local Controller returns a Connection Complete event). For a definition of the different packet types see the “Baseband Specification” on page 55 [Part B].

Note: The Host should enable as many packet types as possible for the Link Manager to perform efficiently. However, the Host must not enable packet types that the local device does not support.

Command Parameters:

BD_ADDR: *Size: 6 Octets*

Value	Parameter Description
0xXXXXXXXXXXXX	BD_ADDR of the Device to be connected.

Packet_Type: *Size: 2 Octets*

Value	Parameter Description
0x0001	Reserved for future use.
0x0002	2-DH1 may not be used.
0x0004	3-DH1 may not be used.
0x0008[1]	DM1 may be used.
0x0010	DH1 may be used.
0x0020	Reserved for future use.
0x0040	Reserved for future use.
0x0080	Reserved for future use.
0x0100	2-DH3 may not be used.
0x0200	3-DH3 may not be used.
0x0400	DM3 may be used.
0x0800	DH3 may be used.
0x1000	2-DH5 may not be used.
0x2000	3-DH5 may not be used.
0x4000	DM5 may be used.
0x8000	DH5 may be used.

1. This bit will be interpreted as set to 1 by Bluetooth V1.2 or later controllers.

Page_Scan_Repetition_Mode: *Size: 1 Octet*

Value	Parameter Description
0x00	R0
0x01	R1
0x02	R2
0x03 – 0xFF	Reserved.

Reserved: *Size: 1 Octet*

Value	Parameter Description
0x00	Reserved, must be set to 0x00. See "Page Scan Mode" on page 612.

Clock_Offset: *Size: 2 Octets*

Bit format	Parameter Description
Bit 14-0	Bit 16-2 of CLKslave-CLKmaster.
Bit 15	Clock_Offset_Valid_Flag Invalid Clock Offset = 0 Valid Clock Offset = 1

Allow_Role_Switch: *Size: 1 Octet*

Value	Parameter Description
0x00	The local device will be a master, and will not accept a role switch requested by the remote device at the connection setup.
0x01	The local device may be a master, or may become a slave after accepting a role switch requested by the remote device at the connection setup.
0x02-0xFF	Reserved for future use.

Return Parameters:

None.

Event(s) generated (unless masked away):

When the Controller receives the Create Connection command, the Controller sends the Command Status event to the Host. In addition, when the LM determines the connection is established, the Controller, on both Bluetooth devices that form the connection, will send a Connection Complete event to each Host. The Connection Complete event contains the Connection Handle if this command is successful.
Note: no Command Complete event will be sent by the Controller to indicate that this command has been completed. Instead, the Connection Complete event will indicate that this command has been completed.

7.1.6 Disconnect Command

Command	OCF	Command Parameters	Return Parameters
HCI_Disconnect	0x0006	Connection_Handle, Reason	

Description:

The Disconnection command is used to terminate an existing connection. The Connection_Handle command parameter indicates which connection is to be disconnected. The Reason command parameter indicates the reason for ending the connection. The remote Bluetooth device will receive the Reason command parameter in the Disconnection Complete event. All synchronous connections on a physical link should be disconnected before the ACL connection on the same physical connection is disconnected.

Command Parameters:

Connection_Handle: *Size: 2 Octets (12 Bits meaningful)*

Value	Parameter Description
0xXXXX	Connection Handle for the connection being disconnected. Range: 0x0000-0x0EFF (0x0F00 - 0x0FFF Reserved for future use)

Reason: *Size: 1 Octet*

Value	Parameter Description
0x05, 0x13-0x15, 0x1A, 0x29	Authentication Failure error code (0x05), Other End Terminated Connection error codes (0x13-0x15), Unsupported Remote Feature error code (0x1A) and Pairing with Unit Key Not Supported error code (0x29), see "Error Codes" on page 319 [Part D] for list of Error Codes.

Return Parameters:

None.

Event(s) generated (unless masked away):

When the Controller receives the Disconnect command, it sends the Command Status event to the Host. The Disconnection Complete event will occur at each Host when the termination of the connection has completed, and indicates that this command has been completed.
Note: No Command Complete event will be sent by the Controller to indicate that this command has been completed. Instead, the Disconnection Complete event will indicate that this command has been completed.

7.1.7 Create Connection Cancel Command

Command	OCF	Command Parameters	Return Parameters
HCI_Create_Connection _Cancel	0x0008	BD_ADDR	Status, BD_ADDR

Description:

This command is used to request cancellation of the ongoing connection creation process, which was started by a Create_Connection command of the local device.

Command Parameters:

BD_ADDR: *Size: 6 Octets*

Value	Parameter Description
0xXXXXXXXXXXXX	BD_ADDR of the Create Connection command request that was issued before and is subject of this cancellation request

Return Parameters:

Status: *Size: 1 Octet*

Value	Parameter Description
0x00	Create Connection Cancel command succeeded
0x01-0xff	Create Connection Cancel command failed. See "Error Codes" on page 319 [Part D] for list of error codes

BD_ADDR: *Size: 6 Octet*

Value	Parameter Description
0xXXXXXXXXXXXX	BD_ADDR of the Create Connection command that was issued before and is the subject of this cancellation request.

Event(s) generated (unless masked away):

When the Create Connection Cancel command has completed, a Command Complete event shall be generated.

If the connection is already established by the baseband, but the Controller has not yet sent the Connection Complete event, then the local device shall detach the link and return a Command Complete event with the status “Success”.

If the connection is already established, and the Connection Complete event has been sent, then the Controller shall return a Command Complete event with the error code *ACL Connection already exists* (0x0B).

If the Create Connection Cancel command is sent to the Controller without a preceding Create Connection command to the same device, the Controller shall return a Command Complete event with the error code *Unknown Connection Identifier* (0x02).

The Connection Complete event for the corresponding Create Connection Command shall always be sent. The Connection Complete event shall be sent after the Command Complete event for the Create Connection Cancel command. If the cancellation was successful, the Connection Complete event will be generated with the error code *Unknown Connection Identifier* (0x02).

7.1.8 Accept Connection Request Command

Command	OCF	Command Parameters	Return Parameters
HCI_Accept_Connection Request	0x0009	BD_ADDR, Role	

Description:

The Accept_Connection_Request command is used to accept a new incoming connection request. The Accept_Connection_Request command shall only be issued after a Connection Request event has occurred. The Connection Request event will return the BD_ADDR of the device which is requesting the connection. This command will cause the Link Manager to create a connection to the Bluetooth device, with the BD_ADDR specified by the command parameters. The Link Manager will determine how the new connection will be established. This will be determined by the current state of the device, its piconet, and the state of the device to be connected. The Role command parameter allows the Host to specify if the Link Manager shall request a role switch and become the Master for this connection. This is a preference and not a requirement. If the Role Switch fails then the connection will still be accepted, and the Role Discovery Command will reflect the current role.
Note: The Link Manager may terminate the connection if it would be low on resources if the role switch fails. The decision to accept a connection must be completed before the connection accept timeout expires on the local Bluetooth Module.
Note: when accepting synchronous connection request, the Role parameter is not used and will be ignored by the Controller.

Command Parameters:

BD_ADDR: *Size: 6 Octets*

Value	Parameter Description
0xXXXXXXXXXXXX	BD_ADDR of the Device to be connected

Role: *Size: 1 Octet*

Value	Parameter Description
0x00	Become the Master for this connection. The LM will perform the role switch.
0x01	Remain the Slave for this connection. The LM will **NOT** perform the role switch.

Return Parameters:

None.

Event(s) generated (unless masked away):

The Accept_Connection_Request command will cause the Command Status event to be sent from the Controller when the Controller begins setting up the connection. In addition, when the Link Manager determines the connection is established, the local Controller will send a Connection Complete event to its Host, and the remote Controller will send a Connection Complete event or a Synchronous Connection Complete event to the Host. The Connection Complete event contains the Connection Handle if this command is successful. Note: no Command Complete event will be sent by the Controller to indicate that this command has been completed. Instead, the Connection Complete event will indicate that this command has been completed.

7.1.9 Reject Connection Request Command

Command	OCF	Command Parameters	Return Parameters
HCI_Reject_Connection_Request	0x000A	BD_ADDR, Reason	

Description:

The Reject_Connection_Request command is used to decline a new incoming connection request. The Reject_Connection_Request command shall only be called after a Connection Request event has occurred. The Connection Request event will return the BD_ADDR of the device that is requesting the connection. The Reason command parameter will be returned to the connecting device in the Status parameter of the Connection Complete event returned to the Host of the connection device, to indicate why the connection was declined.

Command Parameters:

BD_ADDR: *Size: 6 Octets*

Value	Parameter Description
0xXXXXXXXXXXXX	BD_ADDR of the Device to reject the connection from.

Reason: *Size: 1 Octet*

Value	Parameter Description
0x0D-0x0F	Host Reject Error Code. See "Error Codes" on page 319 [Part D] for list of Error Codes and descriptions.

Return Parameters:

None.

Event(s) generated (unless masked away):

When the Controller receives the Reject_Connection_Request command, the Controller sends the Command Status event to the Host. Then, the local Controller will send a Connection Complete event to its host, and the remote Controller will send a Connection Complete event or a Synchronous Connection Complete event to the host. The Status parameter of the Connection Complete event, which is sent to the Host of the device attempting to make the connection, will contain the Reason command parameter from this command.
Note: no Command Complete event will be sent by the Controller to indicate that this command has been completed. Instead, the Connection Complete event will indicate that this command has been completed.

7.1.10 Link Key Request Reply Command

Command	OCF	Command Parameters	Return Parameters
HCI_Link_Key_Request_Reply	0x000B	BD_ADDR, Link_Key	Status, BD_ADDR

Description:

The Link_Key_Request_Reply command is used to reply to a Link Key Request event from the Controller, and specifies the Link Key stored on the Host to be used as the link key for the connection with the other Bluetooth Device specified by BD_ADDR. The Link Key Request event will be generated when the Controller needs a Link Key for a connection.

When the Controller generates a Link Key Request event in order for the local Link Manager to respond to the request from the remote Link Manager (as a result of a Create_Connection or Authentication_Requested command from the remote Host), the local Host must respond with either a Link_Key_Request_Reply or Link_Key_Request_Negative_Reply command before the remote Link Manager detects LMP response timeout. (See "Link Manager Protocol" on page 211 [Part C].)

Command Parameters:

BD_ADDR: *Size: 6 Octets*

Value	Parameter Description
0xXXXXXXXXXXXX	BD_ADDR of the Device of which the Link Key is for.

Link_Key: *Size: 16 Octets*

Value	Parameter Description
0xXXXXXXXXXX XXXXXXXXXX XXXXXXXXXX	Link Key for the associated BD_ADDR.

Return Parameters:

Status: *Size: 1 Octet*

Value	Parameter Description
0x00	Link_Key_Request_Reply command succeeded.
0x01-0xFF	Link_Key_Request_Reply command failed. See "Error Codes" on page 319 [Part D] for error codes and description.

BD_ADDR: *Size: 6 Octets*

Value	Parameter Description
0xXXXXXXXXXXXX	BD_ADDR of the Device of which the Link Key request reply has completed.

Event(s) generated (unless masked away):

The Link_Key_Request_Reply command will cause a Command Complete event to be generated.

7.1.11 Link Key Request Negative Reply Command

Command	OCF	Command Parameters	Return Parameters
HCI_Link_Key_Request_Negative_Reply	0x000C	BD_ADDR	Status, BD_ADDR

Description:

The Link_Key_Request_Negative_Reply command is used to reply to a Link Key Request event from the Controller if the Host does not have a stored Link Key for the connection with the other Bluetooth Device specified by BD_ADDR. The Link Key Request event will be generated when the Controller needs a Link Key for a connection.

When the Controller generates a Link Key Request event in order for the local Link Manager to respond to the request from the remote Link Manager (as a result of a Create_Connection or Authentication_Requested command from the remote Host), the local Host must respond with either a Link_Key_Request_Reply or Link_Key_Request_Negative_Reply command before the remote Link Manager detects LMP response timeout. (See "Link Manager Protocol" on page 211 [Part C].)

Command Parameters:

BD_ADDR: *Size: 6 Octets*

Value	Parameter Description
0xXXXXXXXXXXXX	BD_ADDR of the Device which the Link Key is for.

Return Parameters:

Status: *Size: 1 Octet*

Value	Parameter Description
0x00	Link_Key_Request_Negative_Reply command succeeded.
0x01-0xFF	Link_Key_Request_Negative_Reply command failed. See "Error Codes" on page 319 [Part D] for list of Error Codes.

BD_ADDR: *Size: 6 Octets*

Value	Parameter Description
0xXXXXXXXXXXXX	BD_ADDR of the Device which the Link Key request negative reply has completed.

Event(s) generated (unless masked away):

The Link_Key_Request_Negative_Reply command will cause a Command Complete event to be generated.

7.1.12 PIN Code Request Reply Command

Command	OCF	Command Parameters	Return Parameters
HCI_PIN_Code_Request_Reply	0x000D	BD_ADDR, PIN_Code_Length, PIN_Code	Status, BD_ADDR

Description:

The PIN_Code_Request_Reply command is used to reply to a PIN Code request event from the Controller, and specifies the PIN code to use for a connection. The PIN Code Request event will be generated when a connection with remote initiating device has requested pairing.

When the Controller generates a PIN Code Request event in order for the local Link Manager to respond to the request from the remote Link Manager (as a result of a Create_Connection or Authentication_Requested command from the remote Host), the local Host must respond with either a PIN_Code_Request_Reply or PIN_Code_Request_Negative_Reply command before the remote Link Manager detects LMP response timeout. (See "Link Manager Protocol" on page 211 [Part C].)

Command Parameters:

BD_ADDR: *Size: 6 Octets*

Value	Parameter Description
0xXXXXXXXXXXXX	BD_ADDR of the Device which the PIN code is for.

PIN_Code_Length: *Size: 1 Octet*

Value	Parameter Description
0xXX	The PIN code length specifics the length, in octets, of the PIN code to be used. Range: 0x01-0x10

PIN_Code: *Size: 16 Octets*

Value	Parameter Description
0xXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	PIN code for the device that is to be connected. The Host should insure that strong PIN Codes are used. PIN Codes can be up to a maximum of 128 bits. Note: the PIN_Code Parameter is a string parameter. Endianess does therefore not apply to the PIN_Code Parameter. The first octet of the PIN code should be transmitted first.

Return Parameters:

Status: *Size: 1 Octet*

Value	Parameter Description
0x00	PIN_Code_Request_Reply command succeeded.
0x01-0xFF	PIN_Code_Request_Reply command failed. See "Error Codes" on page 319 [Part D] for list of Error Codes.

BD_ADDR: *Size: 6 Octets*

Value	Parameter Description
0xXXXXXXXXXXXX	BD_ADDR of the Device which the PIN Code request reply has completed.

Event(s) generated (unless masked away):

The PIN_Code_Request_Reply command will cause a Command Complete event to be generated.

7.1.13 PIN Code Request Negative Reply Command

Command	OCF	Command Parameters	Return Parameters
HCI_PIN_Code_ Request_Negative Reply	0x000E	BD_ADDR	Status, BD_ADDR

Description:

The PIN_Code_Request_Negative_Reply command is used to reply to a PIN Code request event from the Controller when the Host cannot specify a PIN code to use for a connection. This command will cause the pair request with remote device to fail.

When the Controller generates a PIN Code Request event in order for the local Link Manager to respond to the request from the remote Link Manager (as a result of a Create_Connection or Authentication_Requested command from the remote Host), the local Host must respond with either a PIN_Code_Request_Reply or PIN_Code_Request_Negative_Reply command before the remote Link Manager detects LMP response timeout. (See "Link Manager Protocol" on page 211 [Part C].)

Command Parameters:

BD_ADDR: *Size: 6 Octets*

Value	Parameter Description
0xXXXXXXXXXXXX	BD_ADDR of the Device which this command is responding to.

Return Parameters:

Status: *Size: 1 Octet*

Value	Parameter Description
0x00	PIN_Code_Request_Negative_Reply command succeeded.
0x01-0xFF	PIN_Code_Request_Negative_Reply command failed. See "Error Codes" on page 319 [Part D] for list of Error Codes.

BD_ADDR: *Size: 6 Octets*

Value	Parameter Description
0xXXXXXXXXXXXX	BD_ADDR of the Device which the PIN Code request negative reply has completed.

Event(s) generated (unless masked away):

The PIN_Code_Request_Negative_Reply command will cause a Command Complete event to be generated.

7.1.14 Change Connection Packet Type Command

Command	OCF	Command Parameters	Return Parameters
HCI_Change_Connection_ Packet_Type	0x000F	Connection_Handle, Packet_Type	

Description:

The Change_Connection_Packet_Type command is used to change which packet types can be used for a connection that is currently established. This allows current connections to be dynamically modified to support different types of user data. The Packet_Type command parameter specifies which packet types the Link Manager can use for the connection. When sending HCI ACL Data Packets the Link Manager shall only use the packet type(s) specified by the Packet_Type command parameter or the always-allowed DM1 packet type. The interpretation of the value for the Packet_Type command parameter will depend on the Link_Type command parameter returned in the Connection Complete event at the connection setup. Multiple packet types may be specified for the Packet_Type command parameter by bitwise OR operation of the different packet types. For a definition of the different packet types see the "Baseband Specification" on page 55 [Part B].

Note: The Host should enable as many packet types as possible for the Link Manager to perform efficiently. However, the Host must not enable packet types that the local device does not support.

Note: to change an eSCO connection, use the Setup Synchronous Connection command.

Command Parameters:

Connection_Handle: *Size: 2 Octets (12 Bits meaningful)*

Value	Parameter Description
0xXXXX	Connection Handle to be used to for transmitting and receiving voice or data. Returned from creating a connection. Range: 0x0000-0x0EFF (0x0F00 - 0x0FFF Reserved for future use)

Packet_Type: *Size: 2 Octets*

For ACL Link_Type

Value	Parameter Description
0x0001	Reserved for future use.
0x0002	2-DH1 may not be used.
0x0004	3-DH1 may not be used.
0x0008[1]	DM1 may be used.
0x0010	DH1 may be used.
0x0020	Reserved for future use.
0x0040	Reserved for future use.
0x0080	Reserved for future use.
0x0100	2-DH3 may not be used.
0x0200	3-DH3 may not be used.
0x0400	DM3 may be used.
0x0800	DH3 may be used.
0x1000	2-DH5 may not be used.
0x2000	3-DH5 may not be used.
0x4000	DM5 may be used.
0x8000	DH5 may be used.

1. This bit will be interpreted as set to 1 by Bluetooth V1.2 or later controllers.

For SCO Link Type

Value	Parameter Description
0x0001	Reserved for future use.
0x0002	Reserved for future use.
0x0004	Reserved for future use.
0x0008	Reserved for future use.
0x0010	Reserved for future use.
0x0020	HV1
0x0040	HV2
0x0080	HV3
0x0100	Reserved for future use.
0x0200	Reserved for future use.
0x0400	Reserved for future use.
0x0800	Reserved for future use.
0x1000	Reserved for future use.
0x2000	Reserved for future use.
0x4000	Reserved for future use.
0x8000	Reserved for future use.

Return Parameters:

None.

Event(s) generated (unless masked away):

When the Controller receives the Change Connection Packet Type command, the Controller sends the Command Status event to the Host. In addition, when the Link Manager determines the packet type has been changed for the connection, the Controller on the local device will send a Connection Packet Type Changed event to the Host. This will be done at the local side only.
Note: no Command Complete event will be sent by the Controller to indicate that this command has been completed. Instead, the Connection Packet Type Changed event will indicate that this command has been completed.

7.1.15 Authentication Requested Command

Command	OCF	Command Parameters	Return Parameters
HCI_Authentication_ Requested	0x0011	Connection_Handle	

Description:

The Authentication_Requested command is used to try to authenticate the remote device associated with the specified Connection Handle. On an authentication failure, the Controller or Link Manager shall not automatically detach the link. The Host is responsible for issuing a Disconnect command to terminate the link if the action is appropriate.
Note: the Connection_Handle command parameter is used to identify the other Bluetooth device, which forms the connection. The Connection Handle should be a Connection Handle for an ACL connection.

Command Parameters:

Connection_Handle: *Size 2 Octets (12 Bits meaningful)*

Value	Parameter Description
0xXXXX	Connection Handle to be used to set up authentication for all Connection Handles with the same Bluetooth device end-point as the specified Connection Handle. Range: 0x0000-0x0EFF (0x0F00 - 0x0FFF Reserved for future use)

Return Parameters:

None.

Event(s) generated (unless masked away):

When the Controller receives the Authentication_Requested command, it sends the Command Status event to the Host. The Authentication Complete event will occur when the authentication has been completed for the connection and is indication that this command has been completed.
Note: no Command Complete event will be sent by the Controller to indicate that this command has been completed. Instead, the Authentication Complete event will indicate that this command has been completed.

Note: When the local or remote Controller does not have a link key for the specified Connection_Handle, it will request the link key from its Host, before the local Host finally receives the Authentication Complete event.

7.1.16 Set Connection Encryption Command

Command	OCF	Command Parameters	Return Parameters
HCI_Set_Connection_ Encryption	0x0013	Connection_Handle, Encryption_Enable	

Description:

The Set_Connection_Encryption command is used to enable and disable the link level encryption. Note: the Connection_Handle command parameter is used to identify the other Bluetooth device which forms the connection. The Connection Handle should be a Connection Handle for an ACL connection. While the encryption is being changed, all ACL traffic must be turned off for all Connection Handles associated with the remote device.

Command Parameters:

Connection_Handle: *Size 2 Octets (12 Bits meaningful)*

Value	Parameter Description
0xXXXX	Connection Handle to be used to enable/disable the link layer encryption for all Connection Handles with the same Bluetooth device end-point as the specified Connection Handle. Range: 0x0000-0x0EFF (0x0F00 - 0x0FFF Reserved for future use)

Encryption_Enable: *Size: 1 Octet*

Value	Parameter Description
0x00	Turn Link Level Encryption OFF.
0x01	Turn Link Level Encryption ON.

Return Parameters:

None.

Event(s) generated (unless masked away):

When the Controller receives the Set_Connection_Encryption command, the Controller sends the Command Status event to the Host. When the Link Manager has completed enabling/disabling encryption for the connection, the Controller on the local Bluetooth device will send an Encryption Change event to the Host, and the Controller on the remote device will also generate an Encryption Change event.
Note: no Command Complete event will be sent by the Controller to indicate that this command has been completed. Instead, the Encryption Change event will indicate that this command has been completed.

7.1.17 Change Connection Link Key Command

Command	OCF	Command Parameters	Return Parameters
HCI_Change_Connection_ Link_Key	0x0015	Connection_Handle	

Description:

The Change_Connection_Link_Key command is used to force both devices of a connection associated with the connection handle to generate a new link key. The link key is used for authentication and encryption of connections.
Note: the Connection_Handle command parameter is used to identify the other Bluetooth device forming the connection. The Connection Handle should be a Connection Handle for an ACL connection.

Command Parameters:

Connection_Handle: *Size 2 Octets (12 Bits meaningful)*

Value	Parameter Description
0xXXXX	Connection Handle used to identify a connection. Range: 0x0000-0x0EFF (0x0F00 - 0x0FFF Reserved for future use)

Return Parameters:

None.

Event(s) generated (unless masked away):

When the Controller receives the Change_Connection_Link_Key command, the Controller sends the Command Status event to the Host. When the Link Manager has changed the Link Key for the connection, the Controller on the local Bluetooth device will send a Link Key Notification event and a Change Connection Link Key Complete event to the Host, and the Controller on the remote device will also generate a Link Key Notification event. The Link Key Notification event indicates that a new connection link key is valid for the connection.
Note: no Command Complete event will be sent by the Controller to indicate that this command has been completed. Instead, the Change Connection Link Key Complete event will indicate that this command has been completed.

7.1.18 Master Link Key Command

Command	OCF	Command Parameters	Return Parameters
HCI_Master_Link_Key	0x0017	Key_Flag	

Description:

The Master Link Key command is used to force the device that is master of the piconet to use the temporary link key of the master device, or the semi-permanent link keys. The temporary link key is used for encryption of broadcast messages within a piconet, and the semi-permanent link keys are used for private encrypted point-to-point communication. The Key_Flag command parameter is used to indicate which Link Key (temporary link key of the Master, or the semi-permanent link keys) shall be used.

Command Parameters:

Key_Flag: *Size: 1 Octet*

Value	Parameter Description
0x00	Use semi-permanent Link Keys.
0x01	Use Temporary Link Key.

Return Parameters:

None.

Event(s) generated (unless masked away):

When the Controller receives the Master_Link_Key command, the Controller sends the Command Status event to the Host. When the Link Manager has changed link key, the Controller on both the local and the remote device will send a Master Link Key Complete event to the Host. The Connection Handle on the master side will be a Connection Handle for one of the existing connections to a slave. On the slave side, the Connection Handle will be a Connection Handle to the initiating master.

The Master Link Key Complete event contains the status of this command. Note: no Command Complete event will be sent by the Controller to indicate that this command has been completed. Instead, the Master Link Key Complete event will indicate that this command has been completed.

7.1.19 Remote Name Request Command

Command	OCF	Command Parameters	Return Parameters
HCI_Remote_Name_Request	0x0019	BD_ADDR, Page_Scan_Repetition_Mode, Reserved, Clock_Offset	

Description:

The Remote_Name_Request command is used to obtain the user-friendly name of another Bluetooth device. The user-friendly name is used to enable the user to distinguish one Bluetooth device from another. The BD_ADDR command parameter is used to identify the device for which the user-friendly name is to be obtained. The Page_Scan_Repetition_Mode parameter specifies the page scan repetition mode supported by the remote device with the BD_ADDR. This is the information that was acquired during the inquiry process. The Clock_Offset parameter is the difference between its own clock and the clock of the remote device with BD_ADDR. Only bits 2 through 16 of the difference are used and they are mapped to this parameter as bits 0 through 14 respectively. A Clock_Offset_Valid_Flag, located in bit 15 of the Clock_Offset command parameter, is used to indicate if the Clock Offset is valid or not.
Note: if no connection exists between the local device and the device corresponding to the BD_ADDR, a temporary link layer connection will be established to obtain the name of the remote device.

Command Parameters:

BD_ADDR: *Size: 6 Octets*

Value	Parameter Description
0xXXXXXXXXXXXX	BD_ADDR for the device whose name is requested.

Page_Scan_Repetition_Mode: *Size: 1 Octet*

Value	Parameter Description
0x00	R0
0x01	R1
0x02	R2
0x03 – 0xFF	Reserved.

Reserved: *Size: 1 Octet*

Value	Parameter Description
0x00	Reserved, must be set to 0x00. See "Page Scan Mode" on page 612.

Clock_Offset: *Size: 2 Octets*

Bit format	Parameter Description
Bit 14.0	Bit 16.2 of CLKslave-CLKmaster.
Bit 15	Clock_Offset_Valid_Flag Invalid Clock Offset = 0 Valid Clock Offset = 1

Return Parameters:

None.

Event(s) generated (unless masked away):

When the Controller receives the Remote_Name_Request command, the Controller sends the Command Status event to the Host. When the Link Manager has completed the LMP messages to obtain the remote name, the Controller on the local Bluetooth device will send a Remote Name Request Complete event to the Host. Note: no Command Complete event will be sent by the Controller to indicate that this command has been completed. Instead, only the Remote Name Request Complete event will indicate that this command has been completed.

7.1.20 Remote Name Request Cancel Command

Command	OCF	Command Parameters	Return Parameters
HCI_Remote_Name_Request_Cancel	0x001A	BD_ADDR	Status, BD_ADDR

Description:

This command is used to request cancellation of the ongoing remote name request process, which was started by the Remote Name Request command.

Command Parameters:

BD_ADDR: *Size: 6 Octets*

Value	Parameter Description
0xXXXXXXXXXXXX	BD_ADDR of the Remote Name Request command that was issued before and that is subject of this cancellation request

Return Parameters:

Status: *Size: 1 Octet*

Value	Parameter Description
0x00	Remote Name Request Cancel command succeeded
0x01-0xff	Remote Name Request Cancel command failed. See "Error Codes" on page 319 [Part D] for list of error codes

BD_ADDR: *Size: 6 Octets*

Value	Parameter Description
0xXXXXXXXXXXXX	BD_ADDR of the Remote Name Request Cancel command that was issued before and that was subject of this cancellation request

Event(s) generated (unless masked away):

When the Remote Name Request Cancel command has completed, a Command Complete event shall be generated.

If the Remote Name Request Cancel command is sent to the Controller without a preceding Remote Name Request command to the same device, the Controller will return a Command Complete event with the error code *Invalid HCI Command Parameters* (0x12).

The Remote Name Request Complete event for the corresponding Remote Name Request Command shall always be sent. The Remote Name Request

Complete event shall be sent after the Command Complete event for the Remote Name Request Cancel command. If the cancellation was successful, the Remote Name Request Complete event will be generated with the error code *Unknown Connection Identifier (0x02)*.

7.1.21 Read Remote Supported Features Command

Command	OCF	Command Parameters	Return Parameters
HCI_Read_Remote_Supported_Features	0x001B	Connection_Handle	

Description:

This command requests a list of the supported features for the remote device identified by the Connection_Handle parameter. The Connection_Handle must be a Connection_Handle for an ACL connection. The Read Remote Supported Features Complete event will return a list of the LMP features. For details see "Link Manager Protocol" on page 211 [Part C].

Command Parameters:

Connection_Handle: *Size: 2 Octets (12 Bits meaningful)*

Value	Parameter Description
0xXXXX	Specifies which Connection Handle's LMP-supported features list to get. Range: 0x0000-0x0EFF (0x0F00 - 0x0FFF Reserved for future use)

Return Parameters:

None.

Event(s) generated (unless masked away):

When the Controller receives the Read_Remote_Supported_Features command, the Controller sends the Command Status event to the Host. When the Link Manager has completed the LMP messages to determine the remote features, the Controller on the local Bluetooth device will send a Read Remote Supported Features Complete event to the Host. The Read Remote Supported Features Complete event contains the status of this command, and parameters describing the supported features of the remote device. Note: no Command Complete event will be sent by the Controller to indicate that this command has been completed. Instead, the Read Remote Supported Features Complete event will indicate that this command has been completed.

7.1.22 Read Remote Extended Features Command

Command	OCF	Command Parameters	Return Parameters
HCI_Read_Remote_Extended_Features	0x001C	Connection_Handle, Page Number	

Description:

The HCI_Read_Remote_Extended_Features command returns the requested page of the extended LMP features for the remote device identified by the specified connection handle. The connection handle must be the connection handle for an ACL connection. This command is only available if the extended features feature is implemented by the remote device. The Read Remote Extended Features Complete event will return the requested information.
For details see "Link Manager Protocol" on page 211 [Part C].

Command Parameters:

Connection_Handle: *Size: 2 Octets (12 Bits meaningful)*

Value	Parameter Description
0xXXXX	The connection handle identifying the remote device for which extended feature information is required. Range: 0x0000-0x0EFF (0x0F00-0x0FFF Reserved for future use)

Page Number: *Size: 1 Octet*

Value	Parameter Description
0x00	Requests the normal LMP features as returned by HCI_Read_Remote_Supported_Features
0x01-0xFF	Return the corresponding page of features

Return Parameters:

None.

Event(s) generated (unless masked away):

When the Controller receives the HCI_Read_Remote_Extended_Features command the Controller sends the Command Status command to the Host. When the Link Manager has completed the LMP sequence to determine the remote extended features the controller on the local device will generate a Read Remote Extended Features Complete event to the host. The Read Remote Extended Features Complete event contains the page number and the remote features returned by the remote device.

Note: no Command Complete event will ever be sent by the Controller to indicate that this command has been completed. Instead the Read Remote Extended Features Complete event will indicate that this command has been completed.

7.1.23 Read Remote Version Information Command

Command	OCF	Command Parameters	Return Parameters
HCI_Read_Remote_Version_Information	0x001D	Connection_Handle	

Description:

This command will obtain the values for the version information for the remote Bluetooth device identified by the Connection_Handle parameter. The Connection_Handle must be a Connection_Handle for an ACL connection.

Command Parameters:

Connection_Handle: *Size: 2 Octets (12 Bits meaningful)*

Value	Parameter Description
0xXXXX	Specifies which Connection Handle's version information to get. Range: 0x0000-0x0EFF (0x0F00 - 0x0FFF Reserved for future use)

Return Parameters:

None.

Event(s) generated (unless masked away):

When the Controller receives the Read_Remote_Version_Information command, the Controller sends the Command Status event to the Host. When the Link Manager has completed the LMP messages to determine the remote version information, the Controller on the local Bluetooth device will send a Read Remote Version Information Complete event to the Host. The Read Remote Version Information Complete event contains the status of this command, and parameters describing the version and subversion of the LMP used by the remote device.
Note: no Command Complete event will be sent by the Controller to indicate that this command has been completed. Instead, the Read Remote Version Information Complete event will indicate that this command has been completed.

7.1.24 Read Clock Offset Command

Command	OCF	Command Parameters	Return Parameters
HCI_Read_Clock_Offset	0x001F	Connection_Handle	

Description:

Both the System Clock and the clock offset to a remote device are used to determine what hopping frequency is used by a remote device for page scan. This command allows the Host to read clock offset to remote devices. The clock offset can be used to speed up the paging procedure when the local device tries to establish a connection to a remote device, for example, when the local Host has issued Create_Connection or Remote_Name_Request. The Connection_Handle must be a Connection_Handle for an ACL connection.

Command Parameters:

Connection_Handle: *Size: 2 Octets (12 bits meaningful)*

Value	Parameter Description
0xXXXX	Specifies which Connection Handle's Clock Offset parameter is returned. Range: 0x0000-0x0EFF (0x0F00 - 0x0FFF Reserved for future use)

Return Parameters:

None.

Event(s) generated (unless masked away):

When the Controller receives the Read_Clock_Offset command, the Controller sends the Command Status event to the Host. If this command was requested at the master and the Link Manager has completed the LMP messages to obtain the Clock Offset information, the Controller on the local Bluetooth device will send a Read Clock Offset Complete event to the Host.
Note: no Command Complete event will be sent by the Controller to indicate that this command has been completed. Instead, only the Read Clock Offset Complete event will indicate that this command has been completed. If the command is requested at the slave, the LM will immediately send a Command Status event and a Read Clock Offset Complete event to the Host, without an exchange of LMP PDU.

7.1.25 Read LMP Handle Command

Command	OCF	Command Parameters	Return Parameters
HCI_Read_LMP_Handle	0x0020	Connection_Handle	Status, Connection_Handle, LMP_Handle, Reserved

Description:

This command will read the current LMP Handle associated with the Connection_Handle. The Connection_Handle must be a SCO or eSCO Handle. If the Connection_Handle is a SCO connection handle, then this command shall read the LMP SCO Handle for this connection. If the Connection_Handle is an eSCO connection handle, then this command shall read the LMP eSCO Handle for this connection.

Command Parameters:

Connection_Handle: *Size 2 Octets (12 bits meaningful)*

Value	Parameter Description
0xXXXX	Connection Handle to be used to identify which connection to be used for reading the LMP Handle. This must be a synchronous handle. Range: 0x0000-0x0EFF (0x0F00 – 0x0FFF Reserved for future use)

Return Parameters:

Status: *Size: 1 Octet*

Value	Parameter Description
0x00	Read_LMP_Handle command succeeded.
0x01 – 0xFF	Read_LMP_Handle command failed. See "Error Codes" on page 319 [Part D] for list of Error Codes.

Connection_Handle: *Size: 2 Octets (12 bits meaningful)*

Value	Parameter Description
0xXXXX	The Connection Handle for the Connection for which the LMP_Handle has been read. Range: 0x0000-0x0EFF (0x0F00 – 0x0FFF Reserved for future use)

LMP_Handle: *Size: 1 Octet*

Value	Parameter Description
0xXX	The LMP Handle is the LMP Handle that is associated with this connection handle. For a synchronous handle, this would be the LMP Synchronous Handle used when negotiating the synchronous connection in the link manager.

Reserved: *Size: 4 Octets*

Value	Parameter Description
0xXXXXXXXX	This parameter is reserved, must to set to zero.

Events(s) generated (unless masked away):

When the Read_LMP_Handle command has completed, a Command Complete event will be generated.

7.1.26 Setup Synchronous Connection Command

Command	OCF	Command Parameters	Return Parameters
HCI_Setup_Synchronous_ Connection	0x0028	Connection_Handle Transmit_Bandwidth Receive_Bandwidth Max_Latency Voice_Setting Retransmission_Effort Packet Type	

Description:

The HCI Setup Synchronous Connection command adds a new or modifies an existing synchronous logical transport (SCO or eSCO) on a physical link depending on the Connection_Handle parameter specified. If the Connection_Handle refers to an ACL link a new synchronous logical transport will be added. If the Connection_Handle refers to an already existing synchronous logical transport (eSCO only) this link will be modified.The parameters are specified per connection. This synchronous connection can be used to transfer synchronous voice at 64kbps or transparent synchronous data.

When used to setup a new synchronous logical transport, the Connection_Handle parameter shall specify an ACL connection with which the new synchronous connection will be associated. The other parameters relate to the negotiation of the link, and may be reconfigured during the lifetime of the link. The transmit and receive bandwidth specify how much bandwidth shall be available for transmitting and for receiving data. While in many cases the receive and transmit bandwidth parameters may be equal, they may be different. The latency specifies an upper limit to the time in milliseconds between the eSCO (or SCO) instants, plus the size of the retransmission window, plus the length of the reserved synchronous slots for this logical transport. The content format specifies the settings for voice or transparent data on this connection. The retransmission effort specifies the extra resources that are allocated to this connection if a packet may need to be retransmitted. The Retransmission_Effort parameter shall be set to indicate the required behavior, or to don't care.

When used to modify an existing synchronous logical transport, the Transmit_Bandwidth, Receive_Bandwidth and Voice_Settings shall be set to the same values as were used during the initial setup. The Packet_Type, Retransmission_Effort and Max_Latency parameters may be modified.

The Packet_Type field is a bitmap specifying which packet types the LM shall accept in the negotiation of the link parameters. Multiple packet types are specified by bitwise OR of the packet type codes in the table. At least one packet type must be specified for each negotiation. It is recommended to enable as many packet types as possible. Note that it is allowed to enable packet types that are not supported by the local device.

A connection handle for the new synchronous connection will be returned in the synchronous connection complete event.

Note: The link manager may choose any combination of packet types, timing, and retransmission window sizes that satisfy the parameters given. This may be achieved by using more frequent transmissions of smaller packets. The link manager may choose to set up either a SCO or an eSCO connection, if the parameters allow, using the corresponding LMP sequences.

Note: To modify a SCO connection, use the Change Connection Packet Type command.

Note: If the lower layers cannot achieve the exact transmit and receive bandwidth requested subject to the other parameters, then the link shall be rejected.

A synchronous connection may only be created when an ACL connection already exists and when it is not in park state.

Command Parameters:

Connection_Handle: *2 octets (12 bits meaningful)*

Value	Parameter Description
0xXXXX	Connection Handle for the ACL connection being used to create a synchronous Connection or for the existing Connection that shall be modified. Range: 0x0000-0x0EFF (0x0F00 - 0x0FFF Reserved for future use)

Transmit_Bandwidth: *4 octets*

Value	Parameter Description
0xXXXXXXXX	Transmit bandwidth in octets per second.

Receive_Bandwidth: *4 octets*

Value	Parameter Description
0xXXXXXXXX	Receive bandwidth in octets per second.