

Ericsson ASIC Specific HCI Commands and Events for Baseband C

THE INFORMATION OF THIS DOCUMENT IS PROVIDED AS IS AND ERICSSON MAKES NO REPRESENTATIONS OR WARRANTIES WHATSOEVER, WHETHER EXPRESS OR IMPLIED, FOR THE ACCURACY OF THE INFORMATION PROVIDED HEREIN. IN NO EVENT SHALL ERICSSON BE LIABLE FOR ANY CONSEQUENCES ARISING FROM THE USE OF THIS INFORMATION NOR FOR ANY INFRINGEMENT OF PATENTS OR OTHER INTELLECTUAL PROPERTY RIGHTS OF THIRD PARTIES THAT MAY RESULT FROM THE USE. NO LICENSE IS GRANTED BY IMPLICATION OR OTHERWISE UNDER ANY INTELLECTUAL PROPERTY RIGHTS OF ERICSSON.

ALL RIGHTS RESERVED. NO PART OF THIS DOCUMENT MAY BE REPRODUCED IN ANY FORM WITHOUT THE WRITTEN PERMISSION OF THE COPYRIGHT HOLDER. THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE.

© Ericsson, 2001

CONTENTS

1	INTRODUCTION.....	6
1.1	ABOUT THIS DOCUMENT	6
1.2	PURPOSE AND USE	6
1.3	DELIVERY OBJECTS.....	6
1.4	SYSTEM LEVEL DESCRIPTION	6
1.5	ABBREVIATIONS	6
1.6	GLOSSARY.....	6
1.7	REFERENCES	7
2	ERICSSON SPECIFIC HCI COMMANDS	8
2.1	ERICSSON_READ_MEMORY	8
2.2	ERICSSON_WRITE_MEMORY	9
2.3	ERICSSON_READ_I2C.....	10
2.4	ERICSSON_WRITE_I2C.....	11
2.5	ERICSSON_SET_UART_BAUD_RATE	12
2.6	ERICSSON_READ_REVISION_INFORMATION	14
2.7	ERICSSON_SELF_TEST	15
2.8	ERICSSON_WRITE_PCM_SETTINGS	16
2.9	ERICSSON_BER	18
2.10	ERICSSON_TX_TEST.....	22
2.11	ERICSSON_SET_SCO_DATA_PATH	25
2.12	ERICSSON_STORE_IN_FLASH	26
2.12.1	<i>BD address</i>	27
2.12.2	<i>Country code</i>	28
2.12.3	<i>USB VID</i>	29
2.12.4	<i>USB PID</i>	30
2.12.5	<i>USB DID</i>	31
3	ERICSSON SPECIFIC EVENTS	32
3.1	ERICSSON_OSE_CRASH_EVENT.....	32
3.2	ERICSSON_OSE_POOL_EVENT.....	34
3.3	ERICSSON_STRING_EVENT.....	36
4	ERICSSON SPECIFIC HCI HARDWARE ERROR CODES	37

TABLE OF TABLES

TABLE 1-1. ABBREVIATIONS. 6

TABLE OF FIGURES

N/A.

1 INTRODUCTION

1.1 About this Document

This document describes HCI commands and events that are ASIC specific and that are therefore not described in reference [1]. Example code has been implemented for an Ericsson ASIC (Baseband C).

In the event that a referenced document is not included in the delivery, it is considered as unnecessary for immediate comprehension.

1.2 Purpose and Use

This document shall be read when an understanding of Ericsson ASIC specific HCI commands and events is needed.

1.3 Delivery Objects

N/A.

1.4 System Level Description

N/A.

1.5 Abbreviations

Abbreviations	Explanation
ASIC	Application Specific Integrated Circuit
BPS	Bits Per Second
DFU	Device Firmware Upgrade
FIFO	First In First Out
HCI	Host Controller Interface
LM	Link Manager
LMP	Link Manager Protocol
OCF	Opcode Command Field
OGF	Opcode Group Field
UART	Universal Asynchronous Receiver and Transmitter
USB	Universal Serial Bus
VOS	Virtual Operating System

Table 1-1. Abbreviations.

1.6 Glossary

N/A.

1.7 References

- [1] Title: “Host Controller Interface Functional Specification”,
Part H:1 of “Specification of the Bluetooth System – Core”
Author: Bluetooth Special Interest Group
Doc no.:
Version: 1.1

- [2] Title: “HCI UART Transport Layer”,
Part H:4 of “Specification of the Bluetooth System – Core”
Author: Bluetooth Special Interest Group
Doc no.:
Version: 1.1

2 ERICSSON SPECIFIC HCI COMMANDS

For all of the commands in this document, a Command Complete event is returned when the actions associated with the command have finished, even if the command implies that LMP messages are exchanged. This is not the case in [1] where a Command Status event followed by an event associated with the issued command is returned for most commands for which LMP messages are exchanged.

It is stated in [1] that all vendor specific commands have OGF 0x3F. This is valid also for Ericsson specific commands.

2.1 Ericsson_Read_Memory

Command	OCF	Command Parameters	Return Parameters
HCI_Ericsson_Read_Memory	0x0001	Memory_Address	Status, Memory_Content

Description:

This command is used to read the data stored at the address specified by the `Memory_Address` parameter. The data is returned in the return parameter `Memory_Content`.

Command Parameters:

Memory_Address: *Size: 4 Bytes*

Value	Parameter Description
	Memory address to read from. The memory address must be 16 bits aligned, i.e. the lowest bit must be equal to zero.

Return Parameters:

Status: *Size: 1 Byte*

Value	Parameter Description
0x00	Ericsson_Read_Memory command succeeded.
0x01-0xFF	Ericsson_Read_Memory command failed. See [1] for list of error codes.

Memory_Content: *Size: 2 Bytes*

Value	Parameter Description
	Data read from specified memory address.

Event(s) generated (unless masked away):

When the `Ericsson_Read_Memory` command has been completed, a Command Complete event will occur.

2.2 Ericsson_Write_Memory

Command	OCF	Command Parameters	Return Parameters
HCI_Ericsson_Write_Memory	0x0002	Memory_Address, Memory_Content	Status

Description:

This command is used to write the data in the `Memory_Content` parameter to the address specified by the `Memory_Address` parameter.

Command Parameters:

Memory_Address: *Size: 4 Bytes*

Value	Parameter Description
	Memory address to write to. The memory address must be 16 bits aligned, i.e. the lowest bit must be equal to zero.

Memory_Content: *Size: 2 Bytes*

Value	Parameter Description
	Data to write to the specified memory address.

Return Parameters:

Status: *Size: 1 Byte*

Value	Parameter Description
0x00	Ericsson_Write_Memory command succeeded.
0x01-0xFF	Ericsson_Write_Memory command failed. See [1] for list of error codes.

Event(s) generated (unless masked away):

When the `Ericsson_Write_Memory` command has been completed, a Command Complete event will occur.

2.3 Ericsson_Read_I2C

Command	OCF	Command Parameters	Return Parameters
HCI_Ericsson_Read_I2C	0x0005	Chip_Id, Address	Status, Data

Description:

This command is used to read from a register in a chip on the I2C bus. The `Chip_Id` parameter specifies the chip that should be read from. The parameter `Address` specifies the address of the register that should be read from in the specified chip. The data that is stored in the specified register is returned in the `Data` return parameter.

Command Parameters:

Chip_Id: *Size: 1 Byte*

Value	Parameter Description
	Identity of the chip to read from on the I2C bus.

Address: *Size: 1 Byte*

Value	Parameter Description
	Address of the registers to read from. Allowed range: See the specification of the specified <code>Chip_Id</code> .

Return Parameters:

Status: *Size: 1 Byte*

Value	Parameter Description
0x00	Ericsson_Read_I2C command succeeded.
0x01-0xFF	Ericsson_Read_I2C command failed. See [1] for list of error codes.

Data: *Size: 1 Byte*

Value	Parameter Description
	Data read.

Event(s) generated (unless masked away):

When the `Ericsson_Read_I2C` command has been completed, a Command Complete event will occur.

2.4 Ericsson_Write_I2C

Command	OCF	Command Parameters	Return Parameters
HCI_Ericsson_Write_I2C	0x0006	Chip_Id, Address, Data	Status

Description:

This command is used to write to a register in a chip on the I2C bus. The `Chip_Id` parameter specifies the chip that should be written to. The parameter `Address` specifies the address of the register that should be written to in the specified chip. The parameter `Data` contains the data that should be written to the specified register.

Command Parameters:

Chip_Id: *Size: 1 Byte*

Value	Parameter Description
	Identity of the chip to write to on the I2C bus.

Address: *Size: 1 Byte*

Value	Parameter Description
	Address of the registers to write to. Allowed range: See the specification of the specified <code>Chip_Id</code> .

Data: *Size: 1 Byte*

Value	Parameter Description
	Data to write. Allowed range: See the specification for the specified <code>Chip_Id</code> and <code>Address</code> .

Return Parameters:

Status: *Size: 1 Byte*

Value	Parameter Description
0x00	Ericsson_Write_I2C command succeeded.
0x01-0xFF	Ericsson_Write_I2C command failed. See reference [1] for list of error codes.

Event(s) generated (unless masked away):

When the `Ericsson_Write_I2C` command has been completed, a Command Complete event will occur.

2.5 Ericsson_Set_UART_Baud_Rate

Command	OCF	Command Parameters	Return Parameters
HCI_Ericsson_Set_UART_Baud_Rate	0x0009	Baud_Rate	Status

Description:

This command is used to change the baud rate for the UART that is used by the HCI interface. The Host must change its baud rate within 0.5 seconds after the HCI_Ericsson_Set_UART_Baud_Rate command has been issued in order to be able to receive the Command Complete event associated with the command. There should not be any outstanding commands when the HCI_Ericsson_Set_UART_Baud_Rate command is issued and no additional commands should be issued before the Command Complete event associated with the command has been received. Furthermore, there should not be any connections when the command is issued.

Note 1: The baud rate is only changed to the default value after a hardware reset. The Baud_Rate is not changed when the HCI_Reset command is issued.

Note 2: Make sure that the platform used supports the desired baud rate. Some platforms support a lower maximum baud rate than what the used ASIC supports. Using a higher baud rate than what the hardware supports will lead to bit errors.

Command Parameters:

Baudrate

Size: 1byte

Value	Parameter Description
0x00	460.8 kbps
0x01	230.4 kbps
0x02	115.2 kbps
0x03	57.6 kbps
0x04	28.8 kbps
0x05	14.4 kbps
0x06	7200 bps
0x07	3600 bps
0x08	1800 bps
0x09	900 bps
0x10	153.6 kbps
0x11	76.8 kbps
0x12	38.4 kbps
0x13	19.2 kbps
0x14	9600 bps
0x15	4800 bps
0x16	2400 bps
0x17	1200 bps

0x18	600 bps
0x19	300 bps
Default value: 0x03 (57600 bps), system clock frequency 13 MHz.	

Return Parameters:

Status:

Size: 1 Byte

Value	Parameter Description
0x00	Ericsson_Set_UART_Baud_Rate command succeeded.
0x01-0xFF	Ericsson_Set_UART_Baud_Rate command failed. See [1] for list of error codes.

Event(s) generated (unless masked away):

When the `Ericsson_Set_UART_Baud_Rate` command has been completed, a Command Complete event will occur.

2.6 Ericsson_Read_Revision_Information

Command	OCF	Command Parameters	Return Parameters
HCI_Ericsson_Read_Revision_Information	0x000F		Status, Revision_Info

Description:

This command is used to return the date when the file was created and revision information. This information is returned in the Revision_Info return parameter, which contains a null (0x00) terminated ASCII string.

Command Parameters:

N/A.

Return Parameters:

Status:

Size: 1 Byte

Value	Parameter Description
0x00	Ericsson_Read_Revision_Information command succeeded.
0x01-0xFF	Ericsson_Read_Revision_Information command failed. See reference [1] for list of error codes.

Revision_Info:

Size: 101 Bytes

Value	Parameter Description
	A null (0x00) terminated ASCII string. The string contains the date when the file was created and revision information.

Event(s) generated (unless masked away):

When the `Ericsson_Read_Revision_Information` command has been completed, a Command Complete event will occur.

2.7 Ericsson_Self_Test

Command	OCF	Command Parameters	Return Parameters
HCI_Ericsson_Self_Test	0x0010	Self_Test_Bit_Mask	Status

Description:

This command is used to perform a self-test of the local device. The `Self_Test_Bit_Mask` parameter contains a bit-mask used to specify what should be tested. The specified values can be ORed. After this command has been issued, zero or more Hardware Error events (see [1]) will be sent to the Host, one for every hardware error that is detected. See Chapter 0 for a description of the Ericsson specific HCI hardware error codes. This is followed by a Command Complete event, which will be sent to the Host when the self-test has completed.

Command Parameters:

Self_Test_Bit_Mask:

Size: 1 Byte

Value	Parameter Description
00000001 _b	Check flash code check sum.
Bits 1-7	Reserved.

Return Parameters:

Status:

Size: 1 Byte

Value	Parameter Description
0x00	Ericsson_Self_Test command succeeded.
0x01-0xFF	Ericsson_Self_Test command failed. See [1] for list of error codes.

Event(s) generated (unless masked away):

For every hardware error that is detected, a Hardware Error event will be sent to the Host. This is followed by a Command Complete event, which will be sent to the Host when the self-test has completed.

2.8 Ericsson_Write_PCM_Settings

Command	OCF	Command Parameters	Return Parameters
HCI_Ericsson_Write_PCM_Settings	0x0007	PCM_Settings	Status

Description:

This command is utilized to set if the PCM interface should be master or slave, the PCM direction and the source for PCM clock synchronization.

Command Parameters:

PCM_Settings:

Size: 1 Byte

Value	Parameter Description
xxxxxxx0 _b	PCM direction: port A receive, port B transmit.
xxxxxxx1 _b	PCM direction: port A transmit, port B receive.
xxxxxx0x _b	PCM interface is slave, external PCM clock and synchronisation.
xxxxxx1x _b	PCM interface is master, internal PCM clock and synchronisation.
xxxxx0xx _b	PcmSync pace is generated from system clock.
xxxxx1xx _b	PcmSync pace is generated from frame pace.
Xxxxx111 _b	Default value

Return Parameters:

Status:

Size: 1 Byte

Value	Parameter Description
0x00	Ericsson_Write_PCM_Settings command succeeded.
0x01-0xFF	Ericsson_Write_PCM_Settings command failed. See [1] for list of error codes.

Event(s) generated (unless masked away):

When the `Ericsson_Write_PCM_Settings` command has completed, a Command Complete event will occur.

2.9 Ericsson_BER

Command	OCF	Command Parameters	Return Parameters
HCI_Ericsson_BER	0x0015	Connection_Handle, RX_On_Start, Synt_On_Start, TX_On_Start, Phd_Off_Start, BER_Hopping_Mode, TX_Channel_Master, TX_Channel_Slave, Whitening_Enable, Nbr_Of_Packets, BER_Packet_Type, Test_Data_Type, Test_Data, PX_On_Start, BER_Interval	Status

Description:

This command is used to measure BER when fully loaded DH1, DH3, DH5, DM1, DM3 or DM5 packets are sent from master to slave on the link that is identified by the `Connection_Handle` parameter. The `Connection_Handle` must be a `Connection_Handle` for an ACL connection. The slave will XOR each received octet with the expected value, so each bit that equals 1 in the data sent to the Host represents a transmission error. Note that if the DM1, DM3 or DM5 packet type is used, error correction will be performed before this is completed, consequently some transmission errors will go unnoticed to the Host. Packets where the length field in the payload header has been subjected to transmission errors will be discarded. Packets are sent every `BER_Interval`th frame, to prevent overflow on the receiving side. Furthermore, the command `HCI_Ericsson_BER` should only be sent on the master side (with one exception as described below). In addition to this, no SCO connections are allowed to exist when the command is issued. The command `HCI_Enable_Device_Under_Test_Mode` must also have been issued on the slave side before the command `HCI_Ericsson_BER` is issued on the master side.

Note that since flow control is not utilized over the air during the BER test, the physical interface between the Host and the Host Controller on the slave side must support a sufficiently high data rate. What counts as sufficiently high depends upon the values of the `BER_Interval` and `BER_Packet_Type` parameters. The lower the value of the `BER_Interval` parameter is, the higher the supported data rate must be. A data rate that is too low may result in a crash of the operating system. It is recommended that the supported data rate is at least twice as large as the data rate on the air interface during the BER test, which is equal to $(\text{size of BER_Packet_Type}) / (\text{BER_Interval})$.

Command Parameters:

Connection_Handle:

Size: 2 Bytes

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

RX_On_Start:

Size: 1 Byte

Value	Parameter Description
1xxxxxxx _b	No change.
0xxxxxxx _b	Use timing as indicated in bits 0:6. Unit is 2 μs.

Synt_On_Start:

Size: 1 Byte

Value	Parameter Description
1xxxxxxx _b	No change.
0xxxxxxx _b	Use timing as indicated in bits 0:6. Unit is 2 μs.

TX_On_Start:

Size: 1 Byte

Value	Parameter Description
1xxxxxxx _b	No change.
0xxxxxxx _b	Use timing as indicated in bits 0:6. Unit is 2 μs.

Phd_Off_Start:

Size: 1 Byte

Value	Parameter Description
1xxxxxxx _b	No change.
0xxxxxxx _b	Use timing as indicated in bits 0:6. Unit is 2 μs.

BER_Hopping_Mode:

Size: 1 Byte

Value	Parameter Description
0x00	North America & Europe (except France)
0x01	Use fixed channels as determined by the parameters TX_Channel_Master and TX_Channel_Slave.
0x02-0xFF	Reserved.

TX_Channel_Master:

Size: 1 Byte

Value	Parameter Description
0x00-0x5D	(Master transmission frequency-2402) MHz.

TX_Channel_Slave:

Size: 1 Byte

Value	Parameter Description
0x00-0x5D	(Slave transmission frequency-2402) MHz.

Whitening_Enable:

Size: 1 Byte

Value	Parameter Description
0x00	Whitening is disabled.
0x01	Whitening is enabled.
0x02-0xFF	Reserved.

Nbr_Of_Packets:

Size: 2 Bytes

Value	Parameter Description
0x0000	An unlimited number of baseband packets will be transmitted.
0x0001-0xFFFF	Number of baseband packets that will be transmitted.

BER_Packet_Type:

Size: 1 Byte

Value	Parameter Description
0x00	DH1
0x01	DH3
0x02	DH5
0x03	DM1
0x04	DM3
0x05	DM5
0x06-0xFF	Reserved.

Test_Data_Type:

Size: 1 Byte

Value	Parameter Description
0x00	Send PRBS (same as in Bluetooth test mode).
0x01	Every octet that is sent equals Test_Data.
0x02-0xFF	Reserved.

Test_Data:

Size: 1 Byte

Value	Parameter Description
	Data to send (all octets will be the same).

PX_On_Start:

Size: 1 Byte

Value	Parameter Description
1xxxxxxx _b	No change.
0xxxxxxx _b	Use timing as indicated in bits 0:6. Unit is 0.25 μs.

BER_Interval:

Size: 1 Byte

Value	Parameter Description
0x01-0xFF	This parameter specifies how often a packet is to be sent. A packet will be sent every BER_Interval th frame.

Return Parameters:

Status:

Size: 1 Byte

Value	Parameter Description
0x00	Ericsson_BER command succeeded.
0x01-0xFF	Ericsson_BER command failed. See [1] for list of error codes.

Event(s) generated (unless masked away):

When the `Ericsson_BER` command has been completed, a Command Complete event will occur.

2.10 Ericsson_TX_Test

Command	OCF	Command Parameters	Return Parameters
HCI_Ericsson_TX_Test	0x0019	RX_On_Start, Synt_On_Start, TX_On_Start, Phd_Off_Start, Test_Scenario, Hopping_Mode, TX_Frequency, RX_Frequency, TX_Test_Interval, Test_Packet_Type, Length_Of_Test_Data	Status

Description:

This command is utilized to measure the TX spectrum. The Host Controller will go into the connection as master (without a slave) after the command has been received. The local device, which is the DUT, will then transmit packets without whitening according to the specified parameters. The local device is then said to be in TX test mode. When in TX test mode, the Host can again send the command to change the parameters or to end the test mode. After ending the test mode, an HCI_Reset command is needed to reset the firmware.

Command Parameters:

RX_On_Start:

Size: 1 Byte

Value	Parameter Description
1xxxxxxx _b	No change.
0xxxxxxx _b	Use timing as indicated in bits 0:6. Unit is 2 μs.

Synt_On_Start:

Size: 1 Byte

Value	Parameter Description
1xxxxxxx _b	No change.
0xxxxxxx _b	Use timing as indicated in bits 0:6. Unit is 2 μs.

TX_On_Start:

Size: 1 Byte

Value	Parameter Description
1xxxxxxx _b	No change.
0xxxxxxx _b	Use timing as indicated in bits 0:6. Unit is 2 μs.

Phd_Off_Start:

Size: 1 Byte

Value	Parameter Description
1xxxxxxx _b	No change.
0xxxxxxx _b	Use timing as indicated in bits 0:6. Unit is 2 μs.

Test_Scenario:

Size: 1 Byte

Value	Parameter Description
	Note: Only transmitter tests are allowed.

Hopping_Mode:

Size: 1 Byte

Value	Parameter Description

TX_Frequency:

Size: 1 Byte

Value	Parameter Description
0x00-0x5D	(DUT transmission frequency-2402) MHz.

RX_Frequency:

Size: 1 Byte

Value	Parameter Description
0x00-0x5D	(DUT reception frequency-2402) MHz.

TX_Test_Interval:

Size: 1 Byte

Value	Parameter Description
0x00-0xFF	Number of empty frames between subsequent transmissions.

Test_Packet_Type:

Size: 1 Byte

Value	Parameter Description
	Note: Only DH1, DH3 or DH5 is allowed.

Length_Of_Test_Data:

Size: 2 Bytes

Value	Parameter Description

Return Parameters:

Status:

Size: 1 Byte

Value	Parameter Description
0x00	Ericsson_TX_Test command succeeded.
0x01-0xFF	Ericsson_TX_Test command failed. See [1] for list of error codes.

Event(s) generated (unless masked away):

When the `Ericsson_TX_Test` command has been completed, a Command Complete event will occur.

2.11 Ericsson_Set_SCO_Data_Path

Command	OCF	Command Parameters	Return Parameters
HCI_Ericsson_Set_SCO_Data_Path	0x001D	SCO_Data_Path	Status

Description:

This command is used to set whether the SCO data should be sent directly to the PCM codec or if it should be sent via the HCI interface (DMA), i.e. in HCI SCO Data Packets over the physical interface (e.g. USB). The setting is valid for all outgoing and incoming SCO connections made after the command `HCI_Ericsson_Set_SCO_Data_Path` has been successfully executed. The command does not change the SCO data path for already existing SCO connections. The restrictions described in [2] regarding how many SCO connections can be completed for PCM, DMA and in total have to be considered. LM will automatically reject an incoming SCO connection if no more SCO connections can be established for the current SCO data path. The Host will not be notified about the incoming SCO connection in this case.

Note: If the Host issues the command while one or more SCO connections are in the set-up state, the command will return a Command Complete event where Status=0x0C (Command Disallowed).

Command Parameters:

SCO_Data_Path:

Size: 1 Byte

Value	Parameter Description
0x00	DMA used for following SCO connections.
0x01	PCM used for following SCO connections. Default
0x02-0xFF	Reserved.

Return Parameters:

Status:

Size: 1 Byte

Value	Parameter Description
0x00	Ericsson_Set_SCO_Data_Path command succeeded.
0x01-0xFF	Ericsson_Set_SCO_Data_Path command failed. See [1] for list of error codes.

Event(s) generated (unless masked away):

When the `Ericsson_Set_SCO_Data_Path` command has been completed, a Command Complete event will occur.

2.12 Ericsson_Store_In_Flash

Command	OCF	Command Parameters	Return Parameters
HCI_Ericsson_Store_In_Flash	0x0022	User_Id, Flash_Length, Flash_Data	Status

Description:

This command is used to store data in flash. After the command has been successfully completed, the command `HCI_Reset` (see [1]) must be issued or a power reset must be made for the settings to take effect.

Command Parameters:

User_Id: *Size: 1 Byte*

Value	Parameter Description
	User_Id for data to be stored in flash.

Flash_Length: *Size: 1 Byte*

Value	Parameter Description
0x01-0xFC	The Flash_Length first bytes in Flash_Data will be stored in flash.

Flash_Data: *Size: 253 Bytes*

Value	Parameter Description
0x01-0xFC	Data to be stored in flash. Only the Flash_Length first bytes in Flash_Data will be stored.

Return Parameters:

Status: *Size: 1 Byte*

Value	Parameter Description
0x00	Ericsson_Store_In_Flash command succeeded.
0x01-0xFF	Ericsson_Store_In_Flash command failed. See reference [1] for list of error codes.

Event(s) generated (unless masked away):

When the `Ericsson_Store_In_Flash` command has been completed, a Command Complete event will occur.

2.12.1 BD address

Bluetooth device address used for identification of the device, see [3].

Name	Size	UserId	Comment
BD address	6 octets	254	Stored in little endian format, e.g. for the BD address 0x1a2b3c4d5e6f, 0x6f is stored in the LSB and 0x1a is stored in the MSB (0x4d5e6f is the LAP).

2.12.2 Country code

Country code, which determines the RF channels used by the device.

Name	Size	UserId	Comment
Country code	1 octet (bit 0 used)	249	The coding is as follows: 0b North America & Europe 1b France

2.12.3 USB VID

Vendor ID.

Name	Size	UserId	Comment
VID	2 bytes	245	Vendor ID for USB device. Low byte first.

2.12.4 USB PID

Product ID.

Name	Size	UserId	Comment
PID	2 bytes	246	Product ID for USB device. Low byte first.

2.12.5 USB DID

Device ID.

Name	Size	UserId	Comment
DID	2 bytes	247	Device ID for USB device. Low byte first.

3 ERICSSON SPECIFIC EVENTS

Vendor specific events have event code 0xFF (see [1]).

The `Event_Id` parameter is the first parameter of all Ericsson specific events described in this document. This parameter indicates which Ericsson specific event that is returned.

In the following subsections, Ericsson specific events are described. The Ericsson specific events described in this document are intended for testing and debugging.

3.1 Ericsson OSE Crash Event

Event	Event Code	Event_Id	Event Parameters
Ericsson OSE Crash	0xFF	0x01	Event_Id, OSE_Error_Code, OSE_Process_Id, OSE_PCB_Address, OSE_Stack_Pointer, OSE_User_Or_OS_Error

Description:

The Ericsson OSE Crash event indicates that the operating system (OSE) has crashed. The event parameters contain information about the crash. The hardware will reset within a few seconds after this event has been sent.

Note: This event will be sent immediately when the operating system crashes. This is the case even if the Host has not yet finished sending another HCI Packet that is in the process of being sent to the Host. This fact makes it impossible to write Host software that can always detect an Ericsson OSE Crash event (since the HCI Event Packet containing the Ericsson OSE Crash event can begin in the middle of another packet that is being received, there is nothing to synchronize with). Therefore, it is best to manually inspect log files of data that is received from the Host Controller in order to find when an Ericsson OSE Crash event begins.

Event Parameters:

Event_Id: *Size: 1 Byte*

Value	Parameter Description
0x01 for HCI Ericsson OSE Crash	Indicates which Ericsson specific HCI event that is returned.

OSE_Error_Code: *Size: 1 Byte*

Value	Parameter Description

OSE_Process_Id: *Size: 1 Byte*

Value	Parameter Description

OSE_PCB_Address:

Size: 1 Bytes

Value	Parameter Description

OSE_Stack_Pointer:

Size: 1 Byte

Value	Parameter Description

OSE_User_Or_OSE_Error:

Size: 1 Byte

Value	Parameter Description
0x00	OSE user.
0x01	OS error.

3.2 Ericsson OSE Pool Event

Event	Event Code	Event_Id	Event Parameters
Ericsson OSE Pool	0xFF	0x04	Event_Id, Owner, Size, Header, Sender, Signal_Number

Description:

The Ericsson OSE Pool event indicates that the operating system (OSE) has crashed because the memory pool was empty. One event is in this case sent for each signal in the pool after the Ericsson OSE Crash event has been sent, see Chapter 3.1.

Event Parameters:

Event_Id: *Size: 1 Byte*

Value	Parameter Description
0x04 for HCI Ericsson OSE Pool	Indicates which Ericsson specific HCI event that is returned.

Owner: *Size: 1 Byte*

Value	Parameter Description

Size: *Size: 2 Bytes*

Value	Parameter Description

Header: *Size: 1 Byte*

Value	Parameter Description

Sender: *Size: 1 Byte*

Value	Parameter Description

Signal_Number:

Size: 2 Bytes

Value	Parameter Description

3.3 Ericsson String Event

Event	Event Code	Event_Id	Event Parameters
Ericsson String	0xFF	0x06	Event_Id, String_Length, String

Description:

The Ericsson String event is used to output a string of ASCII characters.

Event Parameters:

Event_Id: *Size: 1 Byte*

Value	Parameter Description
0x06 for HCI Ericsson String	Indicates which Ericsson specific HCI event that is returned.

String_Length: *Size: 1 Byte*

Value	Parameter Description
0x01-0xFD	Length of the String parameter (in number of bytes).

String: *Size: String_Length Bytes*

Value	Parameter Description
	String of ASCII characters. The string is not '\0' terminated. Note: The String parameter has variable length (length defined by the String_Length parameter). It is therefore a deviation from the fixed-length parameters that are specified in [1].

4 ERICSSON SPECIFIC HCI HARDWARE ERROR CODES

In this section, Ericsson specific HCI hardware error codes, which are returned in the `Hardware_Code` parameter of the Hardware Error event (see [1]), are listed.

Hardware_Code	Description
0x20	HCI UART Transport Layer synchronisation error, see [2].
0x21	Flash code check sum is incorrect. (Can be returned at <code>HCI_Ericsson_Self_Test</code> command.)
0x23	UART FIFO Rx interrupt where buffer is full.
0x24	UART FIFO Rx interrupt where buffer is empty.