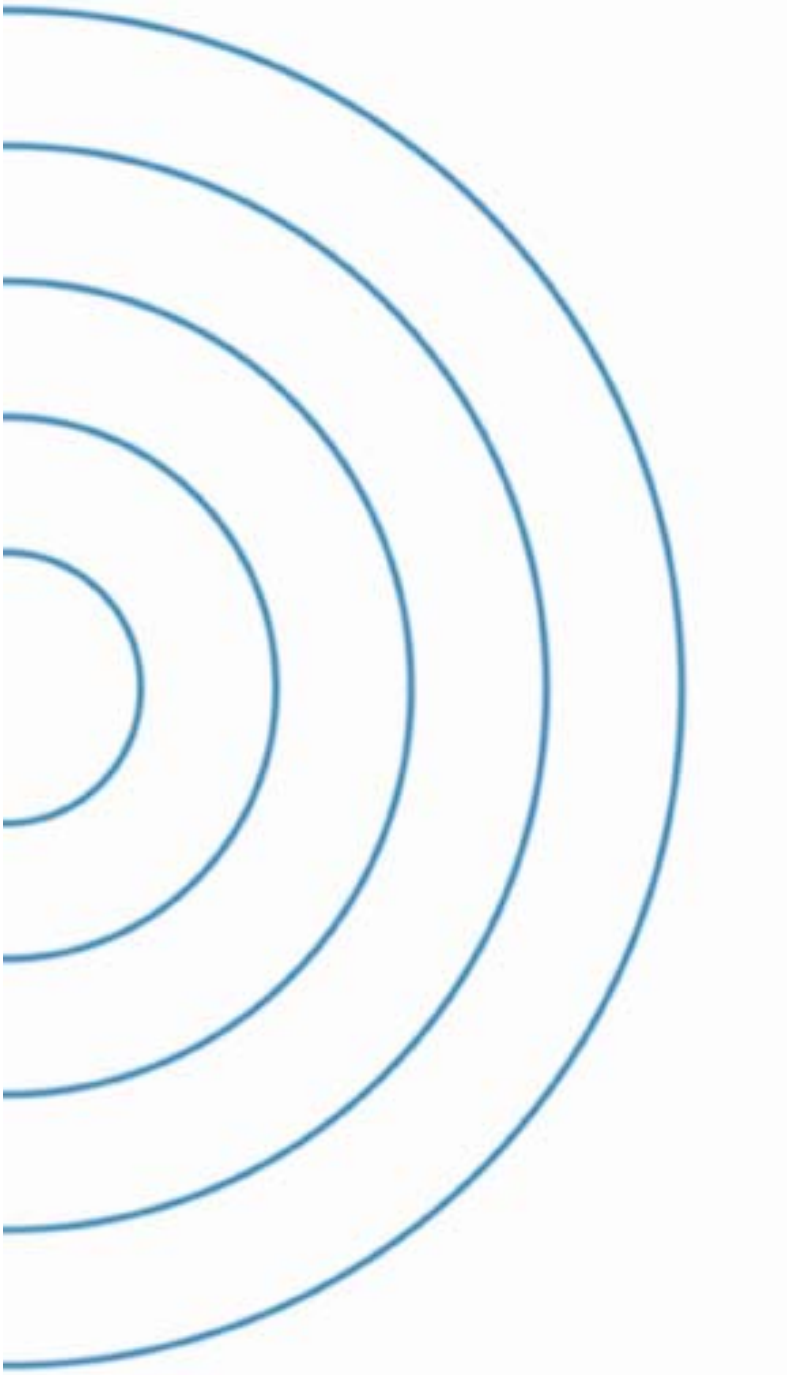
Errata Service Release to *Bluetooth*® Specification versions 1.1, 1.2, 2.0 + EDR, and Profiles

Adopted Errata

Specification of the Bluetooth System

Core & Profiles



ESR02 V10r00
June 21st 2005

Revision History

Revision History	Date	Comments
D10r00	21.03.05	Multi-specification ESR compiled and submitted for review
D10r01	27.03.05	Updated to include erratum 674
V10r00	21.06.05	Adopted by the Bluetooth Board of Directors

Bluetooth System Specifications are updated by errata. Errata published but not yet adopted are for information only. Once adopted and published, errata implementation is permitted and recommended.

This draft Errata Service Release (ESR) documents errata that has not been adopted. Upon adoption of these errata, Members are encouraged but not required to implement the errata. Errata contained within this draft ESR may be selectively implemented upon adoption, except where dependency among errata is expressly indicated.

For the purpose of Bluetooth qualification, a product change incorporating Errata Service Release errata is considered a change within a specific System Specification version, unless otherwise noted in the errata. Implemented errata shall be reviewed and documented as a product change within the Compliance Folder, and listing documentation shall be appropriately updated indicating errata implemented. ESR compliance may be indicated if all errata within the ESR are implemented.

DISCLAIMER AND COPYRIGHT NOTICE

TABLE OF CONTENTS

1 CORE SPECIFICATION VERSION 1.1

1.1 COMPLIANCE REQUIREMENTS

1.1.1 Erratum 736 Replacement Text for Bluetooth Compliance Requirements Part of the Bluetooth Specifications

[Vol 1>Part I:2>Bluetooth Compliance Requirements: Page 847, Replace entire Part with the text below]

Bluetooth Compliance Requirements

1. INTRODUCTION

The Bluetooth Qualification Program Reference Document (PRD) is the primary reference document for the Bluetooth Qualification Program and defines its requirements, functions, and policies. The PRD is available on the Bluetooth Web site.

Passing the Bluetooth Qualification Process demonstrates a certain measure of compliance and interoperability, but because products are not tested for every aspect of this Bluetooth Specification, qualification does not guarantee compliance. Passing the Bluetooth Qualification Process only satisfies one condition of the license grant. The Member has the ultimate responsibility to ensure that the qualified product complies with this Bluetooth Specification and interoperates with other products.

2. SCOPE

This part of the Bluetooth Specification defines some fundamental terms used in the Bluetooth Qualification Program.

3. DEFINITIONS

Bluetooth Qualification Process – The process defined in the Bluetooth Qualification Program Reference Document (PRD) to qualify a design used in implementations of Bluetooth wireless technology.

Bluetooth Qualification Program – The Bluetooth Qualification Process together with other related requirements and processes as defined in the PRD.

Bluetooth Product – Any product containing an implementation of Bluetooth wireless technology that implements, at a minimum, all the mandatory requirements of either one or more of any of the protocol and profile parts of the Specification in compliance with such portion of the Specification.

3.1 Types of Bluetooth products

All Bluetooth Products shall be one of the following:

- Bluetooth End Product
- Bluetooth Controller Subsystem Product
- Bluetooth Host Subsystem Product
- Bluetooth Profile Subsystem Product
- Bluetooth Component Product
- Bluetooth Test Equipment
- Bluetooth Development Tool

Bluetooth End Product - A Bluetooth Product that implements, at a minimum, all mandatory requirements in Radio, Baseband, Link Manager, Logical Link Control and Adaptation Protocol, Service Discovery Protocol and Generic Access Profile parts of the Specification.

Bluetooth Subsystem Product - A Bluetooth Product that implements only a portion of the Specification in compliance with such portion of the Specification and in accordance with the mandatory requirements as defined herein. Bluetooth Subsystem Products can be qualified solely for distribution and the use of Bluetooth wireless technology in Bluetooth Subsystem Products require such Bluetooth Subsystem Products to be combined with one or more Bluetooth Product(s) such that the resulting combination satisfies the definition of either Bluetooth End Product, Bluetooth Development Tool or Bluetooth Test Equipment. There are three types of Bluetooth Subsystem Products as defined below:

- *Bluetooth Host Subsystem Product* – A Bluetooth Subsystem Product containing, at a minimum, all the mandatory requirements defined in the Host Controller Interface, Logical Link Control and Adaptation Protocol, Service Discovery Protocol and Generic Access Profile parts of this Specification, but none of the

protocols below Host Controller Interface (HCI). In addition, a Bluetooth Host Subsystem Product may contain, at a minimum, all the mandatory requirements defined in one or more of the protocols and profiles above HCI.

- *Bluetooth Controller Subsystem Product* – A Bluetooth Subsystem Product containing, at a minimum, all the mandatory requirements defined in the Bluetooth Radio, Baseband, Link Manager and HCI parts of this Specification, but none of the Protocols and Profiles above HCI.

- *Bluetooth Profile Subsystem Product* – A Bluetooth Subsystem Product containing, at a minimum, all the mandatory requirements defined in one or more of the profile specifications.

Bluetooth Component Product - A Bluetooth Product that does not meet the definition of a Bluetooth End Product. Bluetooth Component Products can be qualified solely for distribution and the use of Bluetooth wireless technology in Bluetooth Component Products require such Bluetooth Component Products to be combined with one or more Bluetooth Product(s) such that the resulting combination satisfies the definition of either Bluetooth End Product, Bluetooth Development Tool or Bluetooth Test Equipment.

Bluetooth Development Tool - A Bluetooth Product, solely intended to facilitate the development of new Bluetooth designs. Bluetooth Development Tools can be qualified solely for distribution and the use of the Bluetooth wireless technology for the purposes of development of new Bluetooth Products.

Bluetooth Test Equipment - A Bluetooth Product, intended to facilitate the testing of new Bluetooth Products. Bluetooth Test Equipment can be qualified solely for distribution and the use of the Bluetooth wireless technology in testing of new Bluetooth Products. Where necessary, Bluetooth Test Equipment may deviate from the Specification in order to fulfill the test purposes in the Bluetooth Test Specifications.

1.2 RADIO SPECIFICATION

1.2.1 Erratum 347 Interference Performance

[Vol 1>Part A>Radio Specification: Page 25, Section 4.2 original text states]

Requirement	Ratio
Co-Channel interference, $C/I_{co\text{-}channel}$	11 dB[1)]
Adjacent (1 MHz) interference, C/I_{1MHz}	0 dB[1]
Adjacent (2 MHz) interference, C/I_{2MHz}	-30 dB
Adjacent (≥3 MHz) interference, $C/I_{\geq 3MHz}$	-40 dB
Image frequency Interference[2) 3)], C/I_{Image}	-9 dB[1]
Adjacent (1 MHz) interference to in-band image frequency, $C/I_{Image\pm 1MHz}$	-20 dB[1]

Table 4.1: Interference performance

Note 1. These specifications are tentative and will be fixed within 18 months after the release of the Bluetooth specification version 1.0. Implementations have to fulfil the final specification after a 3-years' convergence period starting at the release of the Bluetooth specification version 1.0. During the convergence period, devices need to achieve a co-channel interference resistance of +14 dB, an ACI (@1MHz) resistance of +4 dB, Image frequency interference resistance of –6 dB and an ACI to in-band image frequency resistance of –16 dB.

Note 2. In-band image frequency

Note 3. If the image frequency ≠ n*1 MHz, than the image reference frequency is defined as the closest n*1 MHz frequency.

Note 4. If two adjacent channel specifications from Table 4.1 are applicable to the same channel, the more relaxed specification applies.

[Replace with]

Requirement	Ratio
Co-Channel interference, $C/I_{co\text{-}channel}$	11 dB
Adjacent (1 MHz) interference, C/I_{1MHz}	0 dB
Adjacent (2 MHz) interference, C/I_{2MHz}	-30 dB
Adjacent (≥3 MHz) interference, $C/I_{\geq 3MHz}$	-40 dB
Image frequency interference[1,2], C/I_{Image}	-9 dB
Adjacent (1 MHz) interference to in-band image frequency, $C/I_{Image\pm 1MHz}$	-20 dB

Table 4.1: Interference performance

Note 1. In-band image frequency

Note 2. If the image frequency ≠ n*1 MHz, than the image reference frequency is defined as the closest n*1 MHz frequency.

Note 3. If two adjacent channel specifications from Table 4.1 are applicable to the same channel, the more relaxed specification applies.

1.3 BASEBAND SPECIFICATION

1.3.1 Erratum 278 Sniff Transition Mode

[Vol 2>Part B>Baseband Specification: Page 111, add a new section after 10.8.2 stating]

10.8.2.1 Sniff Transition Mode

Sniff transition mode is a special mode which is used during the transition between Sniff and active mode. It is required because during this transition it is unclear which mode (Sniff or Active) the slave is in and it is necessary to ensure that the slave is polled correctly regardless of which mode it is in.

In sniff transition mode the master shall maintain the active mode poll interval in case the slave is in active mode. In addition the master shall poll the slave at least once in the sniff attempt transmits slots starting at each sniff instant: note that this transmission counts for the active mode polling as well. The master must use its high power accurate clock when in Sniff Transition Mode.

The precise circumstances under which the master enters Sniff Transition Mode are defined in the LMP specification in section 3.16 on page 214."

The requirement to poll the slave during the sniff attempt slots is required: without it we might as well not have a sniff transition mode. Consider the case where the sniff interval is large and sniff attempts is small: normal poll operation could mean that no transmission occurs while the slave is listening because it would be too close to a previous active mode poll. The wording here ensures that there is at least a chance that the LMP message will get through. Only LMP data will ever be transferred in sniff transition mode. The ACL-C logical link shall carry control information exchanged between the link managers of the master and the slave(s). The ACL-C logical link shall use DM1 packets. The ACL-C logical link is indicated by the LLID code 11 in the payload header.

[End of changes for Erratum 278]

1.4 LINK MANAGER PROTOCOL

1.4.1 Erratum 717 Master or Slave Requests Sniff Mode

[Vol 2>Part B>Link Manager Protocol: Page 214, Section 3.16.1 original text states]

The master or the slave can request to enter sniff mode. Upon receipt of the request, the same request with modified parameters can be returned or the negotiation can be terminated. If an agreement is seen LMP_accepted terminates the negotiation and the ACL link is placed in sniff mode. If no agreement is seen, LMP_not_accepted with the

reason code *unsupported parameter value* terminates the negotiation and sniff mode is not entered.

[Replace with]

Either the master or the slave may request entry to sniff mode.

The process is initiated by sending an LMP_sniff_req PDU containing a set of parameters. The receiving LM shall then decide whether to reject the attempt by sending an LMP_not_accepted PDU, to suggest different parameters by replying with an LMP_sniff_req PDU or to accept the request.

Before the first time that the master sends LMP_sniff_req it shall enter sniff transition mode. If the master receives or sends an LMP_not_accepted PDU it shall exit from sniff transition mode. If the master receives an LMP_sniff_req PDU it shall enter sniff transition mode.

If the master decides to accept the request it shall send an LMP_accepted PDU. When the master receives the baseband acknowledgement for this PDU it shall exit sniff transition mode and enter sniff mode.

If the master receives an LMP_accepted PDU the master shall exit from sniff transition mode and enter sniff mode.

If the slave receives an LMP_sniff_req PDU it must decide whether to accept the request. If the slave does not wish to enter sniff mode then it replies with an LMP_not_accepted PDU. If it is happy to enter sniff mode but requires a different set of parameters it shall respond with an LMP_sniff_req PDU containing the new parameters. If the slave decides that the parameters are acceptable then it shall send an LMP_accepted PDU and enter sniff mode. If the slave receives an LMP_not_accepted PDU it shall terminate the attempt to enter sniff mode.

[Vol 2>Part B>Link Manager Protocol: Page 215, Section 3.16.2 original text states]

Sniff mode is ended by sending the PDU LMP_unsniff_req. The requested device must reply with LMP_accepted. If the slave requests it will enter active mode after receiving LMP_accepted. If the master requests, the slave will enter active mode after receiving LMP_unsniff_req.

[Replace with]

Sniff mode may be exited by either the master or the slave sending an LMP_unsniff_req PDU. The requested device must reply with an LMP_accepted PDU.

If the master requests an exit from Sniff mode it shall enter sniff transition mode

and then send an LMP_unsniff_req PDU. When the slave receives the LMP_unsniff_req it shall exit from sniff mode and reply with an LMP_accepted PDU. When the master receives the LMP_accepted PDU it shall exit from sniff transition mode and enter active mode.

If the slave requests an exit from sniff mode it shall send an LMP_unsniff_req PDU. When the master receives the LMP_unsniff_req PDU it shall enter sniff transition mode and then send an LMP_accepted PDU. When the slave receives the LMP_accepted PDU it shall exit from Sniff mode and enter active mode. When the master receives the baseband acknowledgement for the LMP_accepted PDU it shall leave sniff transition mode and enter active mode.

1.5 HCI FUNCTIONAL SPECIFICATION

1.5.1 Erratum 400 Clarification for Future

[Part H:1>HCI Functional Specification: Page 766, Section 6.1 add text to end of first paragraph]

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

[End of changes for Erratum 400]

1.5.2 Erratum 556 Clarification for Future

[Part H:1>HCI Functional Specification: Page 551, Section 4.3 add new bullet to end of bullet list]

- 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.

[End of changes for Erratum 556]

1.5.3 Erratum 567 Link Supervision Only Allowed for the Master

[Part H:1>HCI Functional Specification: Page 688, Section 4.7.44 at the end of the first paragraph add the text]

This command shall only be issued on the master for the given connection handle. If this command is issued on a slave, the command shall be rejected with the Command Disallowed.

[End of changes for Erratum 567]

1.6 TEST MODE

1.6.1 Erratum 131 Removal of 23 Hop Option

[Part I:1>Test Mode: Page 835, Section 2.1.2 delete Note 2]

2. Some uncertainties about Japanese regulatory requirements have been reported. If necessary for regulatory type approval in Japan, some features might be added; e.g. a longer PN sequence.

[Part I:1>Test Mode: Page 836, Section 2.1.4 item 2 original text states]

2. Frequency selection:
   - Single frequency
   - Hopping Europe/USA
   - Hopping France
   - Reduced Hopping (implementation in Bluetooth devices and modules is optional)

[Replace with]

2. Frequency selection:
   - Single frequency
   - Normal Hopping
   - Reduced Hopping (implementation in Bluetooth devices and modules is optional)

[Part I:1>Test Mode: Page 836, Section 2.1.4 Note 3 original text states]

3. The range is chosen to test the whole frequency range, which covers the normal 79 channels, as well as the French hopping scheme. The frequency assignment rule is the same as for the fixed TX frequency: f = (2402 + k) MHz.

[Replace with]

3. The range is chosen to test the whole frequency range. The frequency assignment rule is the same as for the fixed TX frequency: f = (2402 + k) MHz.

[Part I:1>Test Mode: Page 841, Section 2.2 item 2 original text states]

2. Frequency selection:
   - Single frequency (independent for RX and TX)
   - Hopping Europe/USA
   - Hopping France
   - Hopping reduced (optional)

[Replace with]

2. Frequency selection:
   - Single frequency (independent for RX and TX)
   - Normal Hopping
   - Hopping reduced (optional)

[Part I:1>Test Mode: Page 843, Table 3.2 original row named Hopping mode states]

Name	Length (bytes)	Type	Unit	Detailed
Hopping mode	1	u_int8		0 RX/TX on single frequency 1 Hopping Europe/USA 2 Reserved 3 Hopping France 4 Reserved 5 Reduced Hopping (optional) 6–255 reserved The value is XORed with 0x55.

[Replace with]

Name	Length (bytes)	Type	Unit	Detailed
Hopping mode	1	u_int8		0 RX/TX on single frequency 1 Normal Hopping 2 Reserved 3 Reserved 4 Reserved 5 Reduced Hopping (optional) 6-255 reserved The value is XORed with 0x55.

[End of changes for Erratum 131]

1.7 COMPLIANCE REQUIREMENTS – TEST SPECIFICATION IMPACTS

1.7.1 Erratum 736 Test Specification Impact

No test specification impact.

1.8 RADIO SPECIFICATION – TEST SPECIFICATION IMPACTS

1.8.1 Erratum 347 Test Specification Impact

Record Details:		**Specification Details:**	
Category:	Clarification	**Test Spec:**	RF Conformance
TSE#:	748		
Record Date:	2005-03-12	**Test Spec Version:**	
Subject:	Interference Performance		
Status:	Approved - pending, wait for Core Specification Errata release	**Core Spec Version:**	1.1
Requestor:	Magnus Sommansson		

Test Case Number Description:

OTHER
RCV/CA/03/C

Problem Statement & Suggested Change:

RF Spec (table 4.1) as well as Test Spec contains tentative, relaxed requirements regarding interference performance. SE 347 will be part of the ESR for 1.1. That errata removes the tentative requirements in the Spec and hence those should also be removed from the Test Spec

Proposed Change:

In 5.1.15.4, REMOVE Note 3 In 5.1.15.8 Notes, REMOVE The whole paragraph starting "These specifications are tentative.....

Supporting File:

No supporting files provided.

1.9 BASEBAND SPECIFICATION – TEST SPECIFICATION IMPACTS

1.9.1 Erratum 278 Test Specification Impact

See Test Specification Errata Request Number 659 below:

TS ERRATA REQUEST 659 Sniff transition mode (E278)

Requested by:	Company:	E-mail address:	Date of Request:
John Moring	Bluetooth SIG	john@moring.net	2004-9-15

Reference to Bluetooth Test Specifications

Test Specification: **Link Manager**	Version: **LM TS 1.1.0**	Test Case (if applicable): TP/LIH/BV-14-C TP/LIH/BV-15-C TP/LIH/BV-16-C TP/LIH/BV-18-C TP/LIH/BV-19-C	Test Case Category (if applicable):
Page:	Section:	Paragraph:	

Owner of TS:	E-mail address:	Status/date (Submitted, Review, Sign off, Closed)	
TS errata class: (1-not testable anymore, 2–still testable)	TS errata type: (Correction/Clarification/Editorial) ***Clarification***	ID No.: 659	Approved:

Problem Statement (and / or existing text)

A sniff transition mode is defined in erratum 278 that affects test cases.

New Test Case Proposal

A note should be added in order to specify that during the transition from Sniff to Active mode, or Active to Sniff, then Sniff Transition Mode is used. It should be verified that this mode finishes before checking that the IUT starts sending/answering all packets.

Add "After one sniff period it is verified on baseband level that the IUT..."

TTCN Code should be modified in order to wait for this time before starting to check packets.

5.7.8.2 TP/LIH/BV-14-C (Enter Sniff Mode)
Modify the note in *Figure 5.56: TP/LIH/BV-14-C, MSC* as follows:

After one sniff period it
~~it~~ is verified on baseband level that the
IUT answers to the DM1 packets necessary.

5.7.8.3 TP/LIH/BV-15-C (Initiate Sniff Mode, Slave)
Modify the note in *Figure 5.58: TP/LIH/BV-15-C, MSC* as follows:

After one sniff period it
~~it~~ is verified on baseband level that the IUT
answers to the DM1 packets necessary.

5.7.8.4 TP/LIH/BV-16-C (Exit Sniff Mode)

Modify the note in *Figure 5.60: TP/LIH/BV-16-C* as follows:

After one sniff period it
~~it~~ is verified on baseband level that the IUT answers to the POLL packets necessary.

5.7.9.2 TP/LIH/BV-18-C Initiate Sniff Mode, Master)

Modify the note in *Figure 5.62: TP/LIH/BV-18-C, MSC* as follows:

After one sniff period it
~~it~~ is verified on baseband level that the IUT transmits the name request in allowed SNIFF slots.

5.7.9.3 TP/LIH/BV-19-C (Request Sniff Mode Exit)

Modify the note in *Figure 5.64: TP/LIH/BV-19-C* as follows:

After one sniff period it
~~it~~ is verified on baseband level that the IUT starts to POLL the Tester again as in active mode.

1.10 LINK MANAGER PROTOCOL – TEST SPECIFICATION IMPACTS

1.10.1 Erratum 717 Test Specification Impact

No test specification impact.

1.11 HCI FUNCTIONAL SPECIFICATION – TEST SPECIFICATION IMPACTS

1.11.1 Erratum 400 Test Specification Impact

No test specification impact.

1.11.2 Erratum 556 Test Specification Impact

No test specification impact.

1.11.3 Erratum 567 Test Specification Impact

No test specification impact.

1.12 TEST MODE – TEST SPECIFICATION IMPACTS

1.12.1 Erratum 131 Test Specification Impact

No test specification impact.

2 CORE SPECIFICATION VERSION 1.2

2.1 COMPLIANCE REQUIREMENTS

2.1.1 Erratum 664 Replacement Text for Bluetooth Compliance Requirements Part of the Bluetooth Specifications

[Vol 0>Part B>Bluetooth Compliance Requirements: Page 41, Replace entire Part with the text below]

Bluetooth Compliance Requirements

1. INTRODUCTION

The Bluetooth Qualification Program Reference Document (PRD) is the primary reference document for the Bluetooth Qualification Program and defines its requirements, functions, and policies. The PRD is available on the Bluetooth Web site.

Passing the Bluetooth Qualification Process demonstrates a certain measure of compliance and interoperability, but because products are not tested for every aspect of this Bluetooth Specification, qualification does not guarantee compliance. Passing the Bluetooth Qualification Process only satisfies one condition of the license grant. The Member has the ultimate responsibility to ensure that the qualified product complies with this Bluetooth Specification and interoperates with other products.

2. SCOPE

This part of the Bluetooth Specification defines some fundamental terms used in the Bluetooth Qualification Program.

3. DEFINITIONS

Bluetooth Qualification Process – The process defined in the Bluetooth Qualification Program Reference Document (PRD) to qualify a design used in implementations of Bluetooth wireless technology.

Bluetooth Qualification Program – The Bluetooth Qualification Process together with other related requirements and processes as defined in the PRD.

Bluetooth Product – Any product containing an implementation of Bluetooth wireless technology that implements, at a minimum, all the mandatory requirements of either one or more of any of the protocol and profile parts of the Specification in compliance with such portion of the Specification.

3.1 Types of Bluetooth products

All Bluetooth Products shall be one of the following:

- Bluetooth End Product
- Bluetooth Controller Subsystem Product
- Bluetooth Host Subsystem Product
- Bluetooth Profile Subsystem Product
- Bluetooth Component Product
- Bluetooth Test Equipment
- Bluetooth Development Tool

Bluetooth End Product - A Bluetooth Product that implements, at a minimum, all mandatory requirements in Radio, Baseband, Link Manager, Logical Link Control and Adaptation Protocol, Service Discovery Protocol and Generic Access Profile parts of the Specification.

Bluetooth Subsystem Product - A Bluetooth Product that implements only a portion of the Specification in compliance with such portion of the Specification and in accordance with the mandatory requirements as defined herein. Bluetooth Subsystem Products can be qualified solely for distribution and the use of Bluetooth wireless technology in Bluetooth Subsystem Products require such Bluetooth Subsystem Products to be combined with one or more Bluetooth Product(s) such that the resulting combination satisfies the definition of either Bluetooth End Product, Bluetooth Development Tool or Bluetooth Test Equipment. There are three types of Bluetooth Subsystem Products as defined below:

- *Bluetooth Host Subsystem Product* – A Bluetooth Subsystem Product containing, at a minimum, all the mandatory requirements defined in the Host Controller Interface, Logical Link Control and Adaptation Protocol, Service Discovery Protocol and Generic Access Profile parts of this Specification, but none of the

protocols below Host Controller Interface (HCI). In addition, a Bluetooth Host Subsystem Product may contain, at a minimum, all the mandatory requirements defined in one or more of the protocols and profiles above HCI.

- *Bluetooth Controller Subsystem Product* – A Bluetooth Subsystem Product containing, at a minimum, all the mandatory requirements defined in the Bluetooth Radio, Baseband, Link Manager and HCI parts of this Specification, but none of the Protocols and Profiles above HCI.

- *Bluetooth Profile Subsystem Product* – A Bluetooth Subsystem Product containing, at a minimum, all the mandatory requirements defined in one or more of the profile specifications.

Bluetooth Component Product - A Bluetooth Product that does not meet the definition of a Bluetooth End Product. Bluetooth Component Products can be qualified solely for distribution and the use of Bluetooth wireless technology in Bluetooth Component Products require such Bluetooth Component Products to be combined with one or more Bluetooth Product(s) such that the resulting combination satisfies the definition of either Bluetooth End Product, Bluetooth Development Tool or Bluetooth Test Equipment.

Bluetooth Development Tool - A Bluetooth Product, solely intended to facilitate the development of new Bluetooth designs. Bluetooth Development Tools can be qualified solely for distribution and the use of the Bluetooth wireless technology for the purposes of development of new Bluetooth Products.

Bluetooth Test Equipment - A Bluetooth Product, intended to facilitate the testing of new Bluetooth Products. Bluetooth Test Equipment can be qualified solely for distribution and the use of the Bluetooth wireless technology in testing of new Bluetooth Products. Where necessary, Bluetooth Test Equipment may deviate from the Specification in order to fulfill the test purposes in the Bluetooth Test Specifications.

2.2 APPENDIX

2.2.1 Erratum 180 Incorrect Company Reference

[Vol 0>Part C>Appendix: Page 67, Section 2.3.5 Part E]

Len Ott is referenced twice and the reference to him with Motorola shall be deleted.

[End of changes for Erratum 180]

2.3 ARCHITECTURE

2.3.1 Erratum 577 Mixing of Different Core Package Controller and Host Versions

[Add a new core part in volume 1 "Architecture and Terminology Overview", after section* 4 Deprecated Specifications*]

Erratum Proposal

Submitted by:	Company:	E-mail address:	Date of submission:
Henrik Hedlund	Ericsson Technology Licensing	henrik.hedlund@ericsson.com	2004-02-23

References to Bluetooth Specification

Core/Profile/ Test:	Version/Draft:	Volume:	Part:	Section:	Page:	Paragraph/Tab/Eq/Fig:
Core	BT 1.2	Vol 1	New part	New section	See below	See below

Section-owner:	E-mail address:	Erratum ID title:	
Henrik Hedlund	henrik.hedlund@ericsson.com	Mixing of different Core Package Controller and Host versions	
Type of erratum (Clarification, Correction, etc)	Review forum (BARB, BTI, BQRB, WGs, Associates, etc)	Erratum ID No/	Related errata (ID):
Clarification	Radio WG, BQRB, BARB	<X00xx>	<X00xx>
Problem statement: This erratum gives a clarification how to handle mixed Core Package Controller and Host versions, i e the erratum adds a description and some normative statements how to mix Controller and Host implementations that are based either on BT 1.2, BT 1.1 or older versions.			

Editor Guidelines

Add a new core part in volume 1 "Architecture and Terminology Overview", after section *4 Deprecated Specifications*, with the following title and content:

Part D Mixing of Specification Versions

This part provides a description how different versions of the Core System Packages can be mixed in Bluetooth implementations. The Core System Packages consist of a Controller Package (see volume 2) and a Host Package (see volume 3).

In order to describe how these packages can be mixed, one needs to distinguish between four categories of features specified in the different specification versions. The four categories are:

Type 1	Feature that exists below HCI and cannot be configured via HCI
Type 2	Feature that exists below HCI and can be configured/enabled via HCI
Type 3	Feature that exists below and above HCI and requires HCI command/events to function
Type 4	Feature that exists only above HCI

The outcome of mixing different core system packages are derived from the feature definitions in the table above:

- If an implementation contains features of type 1 or type 4, these features can function with any combination of Host Package and- Controller Package versions.
- If an implementation contains features of type 2, these features can only be used under a default condition if a Host Package v1.1 (or older version) is mixed with a Controller Package v1.2.
- In order to fully use the feature under all conditions, the Host Package and Controller Package must be in version 1.2.
- If an implementation contains features of type 3, these features can only function with a Host Package and a Controller Package both in version 1.2

Editor's note: in a separate section create the following table that lists the V1.2 features and their types.

Feature	**Version**	**Type**
Basic AFH operation	V1.2	1
Enhanced inquiry	V1.2	1
Configuration of AFH (setting channels and enabling/disabling channel assessment)	V1.2	2
Enhanced synchronization capability	V1.2	2
Interlaced inquiry scan	V1.2	2
Interlaced page scan	V1.2	2
Broadcast encryption	V1.2	2
Enhanced flow specification and flush time-out	V1.2	3
Extended SCO links	V1.2	3
Inquiry Result with RSSI	V1.2	3
L2CAP flow and error control	V1.2	4

[End of changes for Erratum 577]

2.3.2 Erratum 563 Definition of Interoperability Requested

[Vol 1>Part A> Architecture: Page 16, Section 1.2, Table 1.1 Nomenclature, add the following definition]

Interoperability	The ability of two or more systems or components to exchange information and to use the information that has been exchanged.

[End of changes for Errata 563]

2.4 CHANGES FROM BLUETOOTH SPECIFICATION V1.1

2.4.1 Erratum 531 Missing Logic with Respect to "Deprecation"

[Vol 1>Part C>Changes from Bluetooth Specification v 1.1: Page 79, Section 4, Paragraph 3 original text states]

Specifications that have been deprecated will not be included in the updated Bluetooth Specifications, but the earlier versions are still available and may be used according to the rules set forth in the Qualification Policy (PRD).

[Replace with]

Specifications that have been deprecated will not be included in the updated Bluetooth Specifications. Deprecation also discontinues all further maintenance of specific associated Test Specifications, PICS and TCRL-records. After deprecation, qualification remains enabled for the earlier versions according to the rules set forth in Qualification Policy (PRD).

[End of changes for Erratum 531]

2.5 BASEBAND SPECIFICATION

2.5.1 Erratum 551 ACL-C Packet Type Mismatch

[Vol 2>Part B>Baseband Specification: Page 95, Section 5.2 original text states]

The ACL-C logical link shall carry control information exchanged between the link managers of the master and the slave(s). The ACL-C logical link shall use DM1 packets. The ACL-C logical link is indicated by the LLID code 11 in the payload header.

[Replace with]

The ACL-C logical link shall carry control information exchanged between the link managers of the master and the slave(s). The ACL-C link shall use DM1 or DV packets. DV packets shall only be used on the ACL-C link if the ACL-C message is less than or equal to 9 bytes and an HV1 synchronous logical transport is in use. The ACL-C logical link is indicated by the LLID code 11 in the payload header.

[End of changes for Erratum 551]

2.5.2 Erratum 547 Use of Initialization in Timing Flags for Slave-Requested eSCO

[Vol 2>Part B>Baseband Specification: Page 150, Section 8.6.3, 2nd Paragraph original text states]

To enter eSCO, the master or slave shall send an eSCO setup command via the LM protocol. This message shall contain the eSCO interval T_{ESCO} and an offset D_{ESCO}. In order to prevent clock wrap-around problems, an initialization

flag in the LMP setup message indicates whether initialization procedure 1 or 2 shall be used. The slave shall apply the initialization method as indicated by the initialization flag. The master shall use initialization 1 when the MSB of the current master clock (CLK_{27}) is 0; it shall use initialization 2 when the MSB of the current master clock (CLK_{27}) is 1. The master-to-slave eSCO slots reserved by the master and the slave shall be initialized on the slots for which the clock satisfies the applicable equation:

[Replace with]

To enter eSCO, the master or slave shall send an eSCO setup command via the LM protocol. This message shall contain the eSCO interval TESCO and an offset DESCO. In order to prevent clock wrap-around problems, an initialization flag in the LMP setup message indicates whether initialization procedure 1 or 2 shall be used. The initiating device shall use initialization 1 when the MSB of the current master clock (CLK27) is 0; it shall use initialization 2 when the MSB of the current master clock (CLK27) is 1. The responding device shall apply the initialization method as indicated by the initialization flag. The master-to-slave eSCO slots reserved by the master and the slave shall be initialized on the slots for which the clock satisfies the applicable equation:

[Vol 2>Part C>Link Manager Protocol: Page 261, Section 4.6.2.2, following Sequence 67 also add the text]

Note that the slave shall use the initialization flag appropriate to the master's BT clock. See Baseband section 8.6.3.

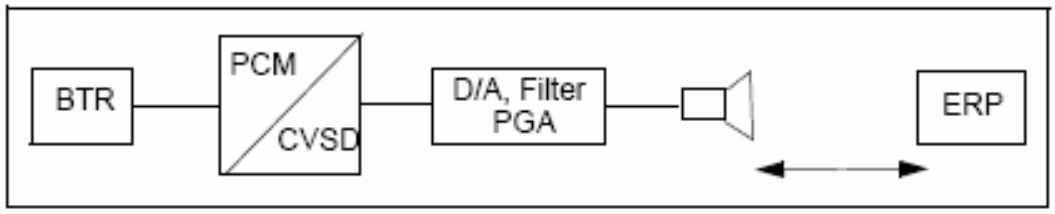
[End of changes for Erratum 547]

2.5.3 Erratum 680 Error in Loudspeaker Path Diagram

[Vol 2>Part B>Baseband Specification: APPENDIX A: GENERAL AUDIO RECOMMENDATIONS, Page 183 original model]

LOUDSPEAKER PATH

RLR measurement model

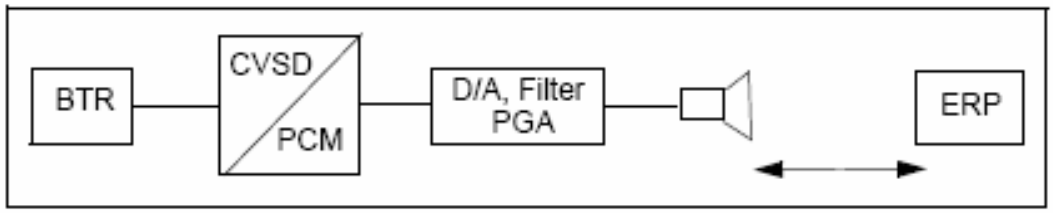


RLR measurement set-up.

[Replace with]

LOUDSPEAKER PATH

RLR measurement model



RLR measurement set-up.

[End of changes for Erratum 680]

2.6 LINK MANAGER PROTOCOL

2.6.1 Erratum 561 EV4/EV5 Mandatory Parameter Ranges

[Vol 2>Part C>Link Manager Protocol: Page 283, Section 5.2 original table 5.3]

	EV3	EV4	EV5
D_{eSCO}	0-4 (even)	0-14 (even)	0-14 (even)
T_{eSCO}	6	16 (even)	16 (even)
W_{eSCO}	0-4 (even)	0-6 (even)	0-6 (even)
eSCO packet type M->S	EV3	EV3, EV4	EV3, EV5
eSCO packet type S->M	EV3	EV3, EV4	EV3, EV5
packet length M->S	30	1-120	1-180
packet length S->M	30	1-120	1-180
air mode	At least one of A-law, mu-law, CVSD, transparent	transparent	transparent

Table 5.3: Mandatory parameter ranges for eSCO packet types

[Replace with]

	Single Slot Packets	3-slot Packets
D_{eSCO}	0 to T_{eSCO}-2 (even)	0 to T_{eSCO}-2 (even)
T_{eSCO}	EV3: 6	EV4: 16 EV5: 16
W_{eSCO}	0, 2 and 4	0 and 6
Packet length M->S	$10*T_{eSCO}/2$	$10*T_{eSCO}/2$
Packet length S->M	$10*T_{eSCO}/2$	$10*T_{eSCO}/2$

Air mode	At least one of A-law, mu-law, CVSD, transparent	Transparent

Table 5.3: Mandatory parameter ranges for eSCO packet types

[End of changes for Erratum 561]

2.7 ERROR CODES

2.7.1 Erratum 497 Clarification for Future

[Vol 2>Part D>Error Codes: Page 296, Section 1.3 after the first paragraph add the text]

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)."

[End of changes for Erratum 497]

2.7.2 Erratum 566 PIN Missing

[Vol 2>Part D>Error Codes: Page 299, Section 2.6 original text states]

2.6 PIN MISSING (0X06)
The PIN Missing error code is used when pairing failed because of a missing PIN.

[Replace with]

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.

[End of changes for Erratum 566]

2.7.3 Erratum 565 Error Code Muddling

[Vol 2>Part D>Error Codes: Page 301, Section 2.17 original text stated]

The Unsupported Feature Or Parameter Value error code indicates that a feature or parameter value in an LMP message or HCI Command is not supported.

[Replace with]

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

[End of changes for Erratum 565]

2.8 HCI FUNCTIONAL SPECIFICATION

2.8.1 Erratum 564 Clarification for Future

[Vol 2>Part E>HCI Functional Specification: Page 350, Section 5.2 add new text within a bullet at the end of this section]

- 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.

[End of changes for Erratum 564]

2.8.2 Erratum 254 Typo – Bad Wording

[Vol 2>Part E>HCI Functional Specification: Page 542, Section 7.7.4 original Class_of_Device table was]

Class_of_Device: *Size: 3 Octets*

Value	Parameter Description
0xXXXXXX	Class of Device for the device, which request the connection.
0x000000	Unknown Class of Device

[Replace with]

Class_of_Device: *Size: 3 Octets*

Value	Parameter Description
0xXXXXXX	Class of Device for the device, which requests the connection.
0x000000	Unknown Class of Device

[End of changes for Erratum 254]

2.8.3 Erratum 540 Wrong errorcode Stated in Create Connection Cancel and Remote_Name_Request_Cancel Commands

[Vol 2>Part E>HCI Functional Specification: Page 385, Section 7.1.7 remove the text]

"The error codes Unspecified Error (0x1F) and Command Disallowed (0x0C) may be used if the Controller does not support this command."

[Vol 2>Part E>HCI Functional Specification: Page 404, Section 7.1.20 remove the text]

"The error codes Unspecified Error (0x1F) and Command Disallowed (0x0C) may be used if the Controller does not support this command."

[End of changes for Erratum 540]

2.8.4 Erratum 570 Description “note”

[Vol 2>Part E>HCI Functional Specification: Page 573, Section 7.7.33 delete the text]

“Note: the only difference between the Inquiry Result with RSSI event and the Inquiry Result event is the additional RSSI parameter.”

[End of changes for Erratum 570]

2.8.5 Erratum 567 Link Supervision Only Allowed for the Master

[Vol 2>Part E>HCI Functional Specification: Page 494, Section 7.3.44 at the end of the first paragraph add the text]

This command shall only be issued on the master for the given connection handle. If this command is issued on a slave, the command shall be rejected with the Command Disallowed.

[End of changes for Erratum 567]

2.8.6 Erratum 410 Specifying BB in HCI

[Vol 2>Part E>HCI Functional Specification: Page 366, Section 6.21 original text stated]

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 parameter defines the number of times the device will retransmit a broadcast data packet. 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 130) This parameter should be adjusted as the link quality measurement changes.

[Replace with]

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 148) This parameter should be adjusted as the link quality measurement changes.

[by removing the 3rd sentence of the original paragraph text]

"This parameter defines the number of times the device will retransmit a broadcast data packet."

[End of changes for Erratum 410]

2.8.7 Erratum 455 Command Parameters are in the Wrong Order

[Vol 2>Part E>HCI Functional Specification: Page 529, Section7.5.6 original table]

Command	OCF	Command Parameters	Return Parameters
HCI_Read_Clock	0x0007	Which_Clock Connection_Handle	Status, Connection_Handle, Clock, Accuracy

[Replace with]

Command	OCF	Command Parameters	Return Parameters
HCI_Read_Clock	0x0007	Connection_Handle Which_Clock	Status, Connection_Handle, Clock, Accuracy

[End of changes for Erratum 455]

2.8.8 Erratum 439 Return Parameter Accuracy

[Vol 2>Part E>HCI Functional Specification: Page 529, Section 7.5.6 original table]

Accuracy: *Size: 2 Octets*

Value	Parameter Description
0xXX	+/- maximum Bluetooth Clock error. Value of 0xFFFF means Unknown. Accuracy = +/- N * 0.3125 msec (1 Bluetooth Clock) Range for N: 0x00-0xFFFE, 0xFFFF Time Range for N: 0 - 20479.375 sec

[Replace with]

Accuracy: *Size: 2 Octets*

Value	Parameter Description
0xXXXX	+/- maximum Bluetooth Clock error. Value of 0xFFFF means Unknown. Accuracy = +/- N * 0.3125 msec (1 Bluetooth Clock) Range for N: 0x0000 - 0xFFFE Time Range for N: 0 - 20479.375 msec

[End of changes for Erratum 439]

2.8.9 Erratum 694 Remove Page Scan Period Mode

[Vol 2>Part E>HCI Functional Specification: Page 328, Section 3.7, Table 3.7 remove the two rows]

Read Page Scan Period Mode Command	1.1	The Read Page Scan Period Mode command is used to read the mandatory Page Scan Period Mode of the local Bluetooth device.
Write Page Scan Period Mode Command	1.1	The Write Page Scan Period Mode command is used to write the mandatory Page Scan Period Mode of the local Bluetooth device.

Table 3.7: Connection setup

[Vol 2>Part E>HCI Functional Specification: Page 342 and 344, Section 3.18, Table 3.18 remove the rows]

Read Page Scan Period Mode Command	Connection Setup

Write Page Scan Period Mode Command	Connection Setup

[Vol 2>Part E>HCI Functional Specification: Page 360, Section 6.10 deprecate this configuration parameter]

6.10 PAGE SCAN PERIOD MODE

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 45 [Part B]. (Keyword: SP-Mode, FHS-Packet, T_mandatory_pscan, Inquiry-Response).

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

[Vol 2>Part E>HCI Functional Specification: Page 371, Section 6.26, change to reserved]

5	Read Page Scan Period Mode
6	Write Page Scan Period Mode

[Vol 2>Part E>HCI Functional Specification: Page 500 and 501, Sections 7.3.48 and 7.3.49, deprecate these commands]

7.3.48 Read Page Scan Period Mode Command

7.3.49 Write Page Scan Period Mode Command

[Vol 2>Part E>HCI Functional Specification: Page 501, Section 7.3.49, original Command Parameters Table]

Command Parameters:

Page_Scan_Period_Mode: *Size: 1 Octet*

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

[Replace with]

Command Parameters:

Page_Scan_Period_Mode: *Size: 1 Octet*

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

[Vol 2>Part E>HCI Functional Specification: Page 538, Section 7.72 original table]

Event	Event Code	Event Parameters
Inquiry Result	0x02	Num_Responses, BD_ADDR[i], Page_Scan_Repetition_Mode[i], Page_Scan_Period_Mode[i], Reserved[i], Class_of_Device[i] Clock_Offset[i]

[Replace with]

Event	Event Code	Event Parameters
Inquiry Result	0x02	Num_Responses, BD_ADDR[i], Page_Scan_Repetition_Mode[i], Reserved[i], Reserved[i], Class_of_Device[i] Clock_Offset[i]

[Vol 2>Part E>HCI Functional Specification: Page 539, Section 7.72, original tables]

Page_Scan_Period_Mode[i]: *Size: 1 Octet * Num_Responses*

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

Reserved[i]: *Size: 1 Octet * Num_Responses*

Value	Parameter Description
0xXX	Reserved, must be set to 0x00.

[Replace with]

Reserved[i][1] *Size: 1 Octet * Num_Responses*

Value	Parameter Description
0xXX	Reserved

1. This was the Page_Scan_Period_Mode parameter in the v1.1 specification. This parameter has no meaning in v1.2 and no default value.

Reserved[i][2] *Size: 1 Octet * Num_Responses*

Value	Parameter Description
0xXX	Reserved, must be set to 0x00.

2. This was the Page_Scan_Mode parameter in the v1.1 specification.

[Vol 2>Part E>HCI Functional Specification: Page 573, Section 7.7.33, original table]

Event	Event Code	Event Parameters
Inquiry Result with RSSI	0x22	Num_responses, BD_ADDR[i], Page_Scan_Repetition_Mode[i], Page_Scan_Period_Mode[i], Class_of_Device[i], Clock_Offset[i], RSSI[i]

[Replace with]

Event	Event Code	Event Parameters
Inquiry Result with RSSI	0x22	Num_responses, BD_ADDR[i], Page_Scan_Repetition_Mode[i], Reserved[i], Class_of_Device[i], Clock_Offset[i], RSSI[i]

[Vol 2>Part E>HCI Functional Specification: Page 574, Section 7.7.33, original table]

Page_Scan_Period_Mode[i]: *Size: 1 Octet* Num_Responses*

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

[Replace with]

Reserved[i][1] *Size: 1 Octet * Num_Responses*

Value	Parameter Description
0xXX	Reserved

1. This was the Page_Scan_Period_Mode parameter in the v1.1 specification.

This parameter has no meaning in v1.2 and no default value.

[Vol 1>Part C>Deprecated Features: Page 79, Section 4.1, original text states]

4.1 DEPRECATED FEATURES

Features deprecated in version 1.2 are:
• The use of Unit Keys for security
• Optional Paging schemes
• 23 channel hopping sequence

[Replace with]

4.1 DEPRECATED FEATURES

Features deprecated in version 1.2 are:
• The use of Unit Keys for security
• Optional Paging schemes
• 23 channel hopping sequence
• Page Scan Period Mode and associated commands

[End of changes for Erratum 694]

2.9 LOGICAL LINK CONTROL AND ADAPTATION PROTOCOL SPECIFICATION

2.9.1 Erratum 559 L2CAP Features Mask

[Vol 3>Part A>Logical Link Control and Adaptation Protocol Specification: Page 55, Section 4.11 original text]

Note: L2CAP entities of versions prior to version 1.2, receiving an Information Request with InfoType = 0x0002 for an L2CAP feature discovery, will return an Information Response with result code "Not supported".

[Replace with]

Note: L2CAP entities of versions prior to version 1.2, receiving an Information Request with InfoType = 0x0002 for an L2CAP feature discovery, will return an Information Response with result code "Not supported". L2CAP entities at version 1.2 or later that have an all zero extended features mask may return an Information Response with result code "Not supported".

[End of changes for Erratum 559]

2.9.2 Erratum 494 Clarification of Reconfiguration During SDU Transmission

[Vol 3>Part A>Logical Link Control and Adaptation Protocol Specification: Page 73, Section 6.1.5 add text after Table 6.7]

Note: The outgoing SDU shall be completed from the view of the remote entity. Therefore all PDUs forming the SDU shall have been reliably transmitted by the local entity and acknowledged by the remote entity, before entering the configuration state.

[End of changes for Erratum 494]

2.10 GENERIC ACCESS PROFILE

2.10.1 Erratum 536 Minor Terminology Updates in GAP

[Vol 3>Part C>Generic Access Profile: Page 212, Section 7.2.1 original text]

The purpose of the channel establishment procedure is to establish a Bluetooth channel (a logical link) between two Bluetooth devices using [3].

[Replace with]

The purpose of the channel establishment procedure is to establish a Bluetooth channel (L2CAP channel) between two Bluetooth devices using [3].

[Vol 3>Part C>Generic Access Profile: Page 215, Section 7.4 original text of 2nd bullet]

• A second channel on the same link, and/or

[Replace with]

• A second channel on the same logical link, and/or

[End of changes for Erratum 536]

2.10.2 Erratum 573 Longer PINS & Alphanumeric

[Vol 3>Part C>Generic Access Profile: Page 186, Section 3.2.3.3 original text]

The Bluetooth PIN has different representations on different levels. PINBB is used on baseband level, and PINUI is used on user interface level.

PINBB is the PIN used by [1] for calculating the initialization key during the pairing procedure. PINUI is the character representation of the PIN that is entered on UI level. The transformation from PINUI to PINBB shall be according to UTF-8 and decimal digits shall be within the Unicode range 0x00 - 0x7F.

According to [1], PINBB can be 128 bits (16 bytes). I.e. if a device supports entry of characters outside the Unicode range 0x00 - 0x7F, the maximum number of characters in the PINUI may be less than 16.

Examples:

User-entered code	Corresponding PIN_{BB}[0..length-1] (value as a sequence of octets in hexadecimal notation)
'0123'	length = 4, value = 0x30 0x31 0x32 0x33
'Ärlich'	length = 7, value = 0xC3 0x84 0x72 0x6C 0x69 0x63 0x68

All Bluetooth devices that support the bonding procedure and support PIN handling on UI level shall support UI level handling of PINs consisting of decimal digits. In addition, devices may support UI level handling of PINs consisting of general characters.

If a device has a fixed PIN (i.e. PIN is stored in the device and cannot be entered on UI level during pairing), the PIN shall be defined using decimal digits. A device that is expected to pair with a remote device that has restricted UI capabilities should ensure that the PIN can be entered on UI level as decimal digits.

[Replace with]

The Bluetooth PIN has different representations on different levels. PINBB is used on baseband level, and PINUI is used on user interface level. PINBB is the PIN used by [1] for calculating the initialization key during the pairing procedure. PINUI is the character representation of the PIN that is entered on UI level. The transformation from PINUI to PINBB shall be according to UTF-8. According to [1], PINBB can be 128 bits (16 bytes).

PIN codes may be up to 16 characters. In order to take advantage of the full level of security all PINs should be 16 characters long. Variable PINs should be composed of alphanumeric characters chosen from within the Unicode range 0x00-0x7F. If the PIN contains any decimal digits these shall be encoded using the Unicode Basic Latin characters (i.e. code points 0x30 to 0x39) (Note 1).

For compatibility with devices with numeric keypads fixed PINs shall be composed of only decimal digits, and variable PINS may be composed of only decimal digits.

If a device supports entry of characters outside the Unicode range 0x00-0x7F other Unicode code points may be used (Note 2), except the Halfwidth and Fullwidth Forms from within the Unicode range FF00 – FFEF shall not be used (Note 3).

Examples:

User-entered code	Corresponding PINBB[0..length-1] (value as a sequence of octets in hexadecimal notation)
’0196554200906493’	length = 16, value = 0x30 0x31 0x39 0x36 0x35 0x35 0x34 0x32 0x30 0x30 0x39 0x30 0x36 0x34 0x39 0x33
’Børnelitteratur’	length = 16, value = 0x42 0xC3 0xB8 0x72 0x6e 0x65 0x6c 0x69 0x74 0x74 0x65 0x72 0x61 0x74 0x75 0x72

Note 1: This is to prevent interoperability problems since there are decimal digits at other code points (e.g. the Fullwidth digits at code points 0xff10 to 0xff19).

Note 2: Unicode characters outside the Basic Latin range (0x00 - 0x7F) encode to multiple bytes, therefore when characters outside the Basic Latin range are used the maximum number of characters in the PINUI will be less than 16. The second example illustrates a case where a 15 character string encodes to 16 bytes because the character ø is outside the Basic Latin range and encodes to two bytes (0xC3 0xB8).

Note 3: This is to prevent interoperability problems since the Halfwidth and Fullwidth forms contain alternative variants of ASCII, Katakana, Hangul, punctuation and symbols. All of the characters in the Halfwidth and Fullwidth forms have other more commonly used Unicode code points.

[End of changes for Erratum 573]

2.11 COMPLIANCE REQUIREMENTS – TEST SPECIFICATION IMPACTS

2.11.1 Erratum 664 Test Specification Impact

No test specification impact.

2.12 APPENDIX – TEST SPECIFICATION IMPACTS

2.12.1 Erratum 180 Test Specification Impact

No test specification impact.

2.13 ARCHITECTURE – TEST SPECIFICATION IMPACTS

2.13.1 Erratum 577 Test Specification Impact

No test specification impact.

2.13.2 Erratum 563 Test Specification Impact

No test specification impact.

2.14 CHANGES FROM BLUETOOTH SPECIFICATION V1.1 – TEST SPECIFICATION IMPACTS

2.14.1 Erratum 531 Test Specification Impact

No test specification impact.

2.15 BASEBAND SPECIFICATION – TEST SPECIFICATION IMPACTS

2.15.1 Erratum 551 Test Specification Impact

No test specification impact.

2.15.2 Erratum 547 Test Specification Impact

See Test Specification Errata Request Number 655 below:

TS ERRATA REQUEST 655 Timing Control Flags (E547)

Requested by:	Company:	E-mail address:	Date of Request:
John Moring	Bluetooth SIG	john@moring.net	2004-9-15

Reference to Bluetooth Test Specifications

Test Specification:	Version:	Test Case (if applicable):	Test Case Category (if applicable):
Link Manager	**20.B.355/1.2.4**	TP/LIH/BV-111-C	
Page:	Section:	Paragraph: 5.7.21.2	

Owner of TS:	E-mail address:	Status/date (Submitted, Review, Sign off, Closed)	
TS errata class: (1-not testable anymore, 2–still testable)	TS errata type: (Correction/Clarification/Editorial) ***Correction***	ID No.: 655	Approved:

Problem Statement (and / or existing text)

TC_LIH_BV_111 should be modified in the following way:
"Timing Control Flags: Derived from IUT's master clock".

New Test Case Proposal

Change the message definition in the Test Procedure as follows:

The LMP_eSCO_link_req shall have the following content:

eSCO handle: Any valid number.
eSCO LT_ADDR: Any valid number.
Timing control flags: ~~Any valid value~~ Derived from IUT's master clock.
Desco: Any number in the range [0, Tesco - 2].
Tesco: 6 slots.
Wesco: 2 slots.
Packet type M→S: EV3.
Packet type S→M: EV3.
Packet length M→S: 30 bytes.
Packet length S→M: 30 bytes.
Air mode: Any supported air mode.
Negotiation Flag: Initiate Negotiation.

2.15.3 Erratum 680 Test Specification Impact

No test specification impact.

2.16 LINK MANAGER PROTOCOL – TEST SPECIFICATION IMPACTS

2.16.1 Erratum 561 Test Specification Impact

See Test Specification Errata Request Numbers 665 and 666 below:

TS ERRATA REQUEST 665 EV4/EV5 mandatory parameter ranges (E561)

Requested by:	Company:	E-mail address:	Date of Request:
John Moring	Bluetooth SIG	john@moring.net	2004-9-15

Reference to Bluetooth Test Specifications

Test Specification:	Version:	Test Case (if applicable):	Test Case Category (if applicable):
Link Manager	**20.B.355/1.2.4**	TP/LIH/BV-101-C, TP/LIH/BV-102-C	
Page:	Section:	Paragraph: 5.7.20.2, 5.7.20.3	

Owner of TS:	E-mail address:	Status/date (Submitted, Review, Sign off, Closed)	
TS errata class: (1-not testable anymore, 2–still testable)	TS errata type: (Correction/Clarification/Editorial) ***Correction***	ID No.: 665	Approved:

Problem Statement (and / or existing text)

Core erratum 561 requires a change to the packet length from 120 bytes to 80 bytes in the following

two LMP testcases: 5.7.20.2 TP/LIH/BV-101-C (Accept EV4 eSCO Request) & 5.7.20.3 TP/LIH/BV-102-C (Accept EV5 eSCO Request).

New Test Case Proposal

Modify each of the two packet lengths specified in the LMP_eSCO_link_req to 80 in both test cases.

In 5.7.20.2 TP/LIH/BV-101-C:
The LMP_eSCO_link_req shall have the following content:

eSCO handle: Any valid number.
eSCO LT_ADDR: Any valid number.
Timing control flags: Derived from master clock.
Desco: Any number in the range [0, Tesco - 2].
Tesco: 16 slots.
Wesco: 6 slots.
Packet type M→S: EV4.
Packet type S→M: EV4.
Packet length M→S: ~~120~~ 80 bytes.
Packet length S→M: ~~120~~ 80 bytes.
Air mode: Transparent.
Negotiation Flag: Initiate Negotiation.

In 5.7.20.3 TP/LIH/BV-102-C
The LMP_eSCO_link_req shall have the following content:

eSCO handle: Any valid number.
eSCO LT_ADDR: Any valid number.
Timing control flags: Derived from master clock.
Desco: Any number in the range [0, Tesco - 2].
Tesco: 16 slots.
Wesco: 6 slots.
Packet type M→S: EV5.
Packet type S→M: EV5.
Packet length M→S: ~~180~~ 80 bytes.
Packet length S→M: ~~180~~ 80 bytes.
Air mode: Transparent.
Negotiation Flag: Initiate Negotiation.

TS ERRATA REQUEST 666 EV4/EV5 mandatory parameter ranges (E561)

Requested by:	Company:	E-mail address:	Date of Request:
Alberto Salcedo	RFMD	asalcedo@rfmd.com	2004-9-22

Reference to Bluetooth Test Specifications

Test Specification: **Baseband**	Version: **20.B.354/1.2.1**	Test Case (if applicable): TP/PROT/COD/BV-18-C, TP/PROT/COD/BV-19-C	Test Case Category (if applicable):
Page:	Section:	Paragraph: 5.4.1.4, 5.4.1.5	

Owner of TS:	E-mail address:	Status/date (Submitted, Review, Sign off, Closed)	
TS errata class: (1-not testable anymore, 2–still testable)	TS errata type: (Correction/Clarification/Editorial) ***Correction***	ID No.: 666	Approved:

Problem Statement (and / or existing text)

Core erratum 561 requires a change to the packet length from 120 bytes to 80 bytes in the following two BB testcases: TP/COD/BV-18-C and TP/COD/BV-19-C.

New Test Case Proposal

Change payload used to 80 bytes in both test cases.

In 5.4.1.14 TP/PROT/COD/BV18-C:

Packet Header:

LT_ADDR: Logical Transport Address of the slave.
TYPE: ‘1100’B.
FLOW: ‘1’B.
ARQN: ‘1’B.
SEQN: Depends on the former transmission of the tester.
HEC: Generated by the polynomial ‘647’O in respect to the UAP of the master.

Payload header:

N/A.

Payload:

~~120~~ 80 bytes PRBS plus 16 bit CRC.

In 5.4.1.15 TP/PROT/COD/BV19-C:

Packet Header:

LT_ADDR: Logical Transport Address of the slave.
TYPE: ‘1101’B.
FLOW: ‘1’B.
ARQN: ‘1’B.
SEQN: depends on the former transmission of the tester.
HEC: Generated by the polynomial ‘647’O in respect to the UAP of the master.

Payload header:
N/A.
Payload:
~~180~~ 80 bytes PRBS plus 16 bit CRC.

2.17 ERROR CODES – TEST SPECIFICATION IMPACTS

2.17.1 Erratum 497 Test Specification Impact

No test specification impact.

2.17.2 Erratum 566 Test Specification Impact

See Test Specification Errata Request Numbers 656 and 657 below:

TS ERRATA REQUEST 656 Error code "key missing" (E566)

Requested by:	Company:	E-mail address:	Date of Request:
John Moring	Bluetooth SIG	john@moring.net	2004-9-15

Reference to Bluetooth Test Specifications

Test Specification:	Version:	Test Case (if applicable):	Test Case Category (if applicable):
Link Manager	**20.B.355/1.2.4**	TP/AUT/BV-01-C	
Page:	Section:	Paragraph: 5.4.1.2	

Owner of TS:	E-mail address:	Status/date (Submitted, Review, Sign off, Closed)	
TS errata class: (1-not testable anymore, 2–still testable)	TS errata type: (Correction/Clarification/Editorial) ***Correction***	ID No.: 656	Approved:

Problem Statement (and / or existing text)

Erratum 566 clarifies the use of error codes, which affects the test case.

The error code "Key Missing" should be changed to "PIN or Key missing" in MSC and Pass Verdict.

New Test Case Proposal

Change Figure 5.7: TP/AUT/BV-01-C as follows:

LMP_not accepted
(OpCode LMP_au_rand,
reason=PIN or key missing)

Change the text in the Expected Outcome as follows:

Expected Outcome

Pass verdict:

The IUT transmits the PDU LMP_not_accepted containing the opcode for PDU LMP_au_rand and "PIN or Key missing" upon reception PDU LMP_au_rand.

Fail verdict:

The IUT does not transmit PDU LMP_not_accepted upon reception PDU LMP_au_rand.

TS ERRATA REQUEST 657 Error code "key missing" (E566)

Requested by:	Company:	E-mail address:	Date of Request:
John Moring	Bluetooth SIG	john@moring.net	2004-9-15

Reference to Bluetooth Test Specifications

Test Specification:	Version:	Test Case (if applicable):	Test Case Category (if applicable):
GAP	**21.B.358/1.2.1**	TP/SEC/AUT/BV-01-C, other	
Page:	Section:	Paragraph:	
		5.2.4, 5.4.1.2	

Owner of TS:	E-mail address:	Status/date (Submitted, Review, Sign off, Closed)	
TS errata class: (1-not testable anymore, 2–still testable)	TS errata type: (Correction/Clarification/Editorial)	ID No.:	Approved:
	Correction	657	

Problem Statement (and / or existing text)

Core erratum 566 has changed the error code name "key missing" to "PIN or key missing." References in the test spec must be updated.

New Test Case Proposal

In 5.2.4 (Figure 5.5) change "key missing" to "PIN or key missing":

LMP_not_accepted
(opcode: LMP_au_rand,
reason: PIN or key missing)

In TP/SEC/AUT/BV-01-C (Figure 5.14) change "key missing" to "PIN or key missing":
LMP_not_accepted
(opcode: LMP_au_rand,
reason: PIN or key missing)

2.17.3 Erratum 565 Test Specification Impact

See Test Specification Errata Request Numbers 641 below:

TS ERRATA REQUEST 641 Specify reason code instead of reason code name to avoid confusion

Requested by:	Company:	E-mail address:	Date of Request:
Tom Cargill	RFMD	tcargill@rfmd.com	2004-7-29

Reference to Bluetooth Test Specifications

Test Specification:	Version:	Test Case (if applicable):	Test Case Category (if applicable):
Link Manager	**20.B.355/1.2.1**	TP/ENC/BV-03-C TP/ENC/BV-12-C TP/LIH/BV-03-C TP/LIH/BV-12-C TP/LIH/BV-20-C TP/LIH/BV-34-C TP/LIH/BV-60-C TP/LIH/BV-71-C TP/LIH/BV-72-C TP/LIH/BV-116-C	

Page:	Section:	Paragraph:
various		various

Owner of TS:	E-mail address:	Status/date (Submitted, Review, Sign off, Closed)	
TS errata class: (1-not testable anymore, 2–still testable)	TS errata type: (Correction/Clarification/Editorial) ***Clarification***	ID No.: 641	Approved:

Problem Statement (and / or existing text)

See Recommended Core Erratum #565 (#559 in old system). Some confusion in reason codes was introduced when the V1.2 Core spec was published. The HCI and LMP error code tables were combined and one error reason 0x1A previously had different descriptions for HCI and LMP. The description "Unsupported LMP Feature" was erroneously deleted for reason 0x1A, even though it is referred to in the error handling area of the core spec, and used throughout the test specification. Core erratum 565 assigns both descriptions to reason 0x1A and prohibits the use of reason code 0x11 in link manager.

New Test Case Proposal

In 5.5.1.4 TP/ENC/BV-03-C (Reject Encryption)

In Figure 5.18: TP/ENC/BV-03-C:

LMP_not_accepted
(OpCode LMP_encryption_mode_req, ~~unsupported feature~~ Reason=0x1A)

In Expected Outcome:
Pass verdict:
The IUT transmits the PDU LMP_not_accepted containing "~~Unsupported LMP Feature~~ Reason=0x1A" upon reception of PDU LMP_encryption_mode_req.
Fail verdict:
The IUT does not transmit PDU LMP_not_accepted containing "~~Unsupported LMP Feature~~ Reason=0x1A" upon reception of PDU LMP_encryption_mode_req.

In 5.5.1.8 TP/ENC/BV-12-C (Reject Broadcast Encryption)

In Figure 5.22: TP/ENC/BV-12-C:

LMP_not_accepted
(opcode LMP_encryption_key_size_mask_req,
reason: ~~unsupported feature~~ Reason=0x1A)

LMP_not_accepted
(OpCode LMP_encryption_mode_req,
reason: ~~unsupported feature~~ Reason=0x1A)

In Expected Outcome:

Pass verdict:
The IUT transmits the PDU LMP_not_accepted containing "~~Unsupported LMP Feature~~ Reason=0x1A" upon reception of the PDU LMP_encryption_key_size_mask_req.

Fail verdict:
The IUT does not transmit the PDU LMP_not_accepted containing "~~Unsupported LMP Feature~~ Reason=0x1A" upon reception of the PDU LMP_encryption_key_size_mask_req.

In 5.7.3.2 TP/LIH/BV-03-C (Unsupported Role Switch):

In Figure 5.47: TP/LIH/BV-03-C:
LMP_not_accepted
(OpCode LMP_switch_req, ~~unsupported LMP feature~~ Reason=0x1A)

In Expected Outcome:

Pass verdict:

The IUT transmits PDU LMP_not_accepted containing "~~Unsupported LMP Feature~~ Reason=0x1A" upon reception of PDU LMP_switch_req.
Fail verdict:
The IUT does not transmit PDU LMP_not_accepted containing "~~Unsupported LMP Feature~~ Reason=0x1A" upon reception of PDU LMP_switch_req.

In 5.7.7.2 TP/LIH/BV-12-C (Hold Mode Unsupported):

In Figure 5.55: TP/LIH/BV-12-C:
LMP_not_accepted
(OpCode LMP_hold_req, ~~unsupported LMP feature~~ Reason=0x1A)

In Expected Outcome:
Pass verdict:
The IUT transmits PDU LMP_not_accepted containing "~~Unsupported LMP Feature~~ Reason=0x1A" upon reception of PDU LMP_hold_req.
Fail verdict:
The IUT does not transmit PDU LMP_not_accepted containing "~~Unsupported LMP Feature~~ Reason=0x1A" upon reception of PDU LMP_hold_req.

In *5.7.10.2 TP/LIH/BV-20-C (Sniff Mode Reject):*

In Figure 5.65: TP/LIH/BV-20-C:
LMP_not_accepted
(OpCode LMP_sniff_req, ~~unsupported LMP feature~~ Reason=0x1A)

In Expected Outcome:
Pass verdict:
The IUT transmits PDU LMP_not_accepted containing "~~Unsupported LMP Feature~~ Reason=0x1A" upon reception of PDU LMP_sniff_req.
Fail verdict:
The IUT does not transmit PDU LMP_not_accepted containing "~~Unsupported LMP Feature~~ Reason=0x1A" upon reception of PDU LMP_sniff_req.

In *5.7.13.2 TP/LIH/BV-34-C (Park Reject):*

In Figure 5.83: TP/LIH/BV-34-C:
LMP_not_accepted)
(OpCode LMP_park_req,
~~unsupported LMP feature~~ Reason=0x1A)

In Expected Outcome:
Pass verdict:
The IUT transmits PDU LMP_not_accepted containing "~~unsupported~~

~~LMP feature~~ Reason=0x1A " upon reception of PDU LMP_park_req.
Fail verdict:
The IUT does not transmit PDU LMP_not_accepted containing "~~unsupported LMP feature~~ Reason=0x1A " upon reception of PDU LMP_park_req.

In *5.7.19.2 TP/LIH/BV-60-C (Reject SCO Request):*

In Figure 5.109: TP/LIH/BV-60-C:
LMP_not_accepted
(OpCode LMP_SCO_link_req,
Reason=~~Unsupported LMP feature~~ 0x1A)

In Expected Outcome:
Pass verdict:
The IUT transmits PDU LMP_not_accepted containing "~~Unsupported LMP Feature~~ Reason=0x1A" upon reception of PDU LMP_SCO_link_req.
Fail verdict:
The IUT does not transmit PDU LMP_not_accepted containing "~~Unsupported LMP Feature~~ Reason=0x1A" upon reception of PDU LMP_SCO_link_req.

In *5.7.21.7 TP/LIH/BV116-C (Reject eSCO Request):*

In Figure 5.126: TP/LIH/BV116-C:
LMP_not_accepted_ext
(OpCode LMP_eSCO_link_req,
Reason=~~Unsupported LMP feature~~ 0x1A)

In Expected Outcome:
Pass verdict:
The IUT transmits PDU LMP_not_accepted_ext containing "~~Unsupported LMP feature~~ Reason=0x1A" upon reception of PDU LMP_eSCO_link_req.
Fail verdict:
The IUT does not transmit PDU LMP_not_accepted_ext containing "~~Unsupported LMP feature~~ Reason=0x1A" upon reception of PDU LMP_eSCO_link_req.

In 5.7.24.2 TP/LIH/BV-71-C (Reject Page Mode Negotiation):

In Figure 5.130: TP/LIH/BV-71-C:
LMP_not_accepted
(OpCode LMP_page_mode_req,
reason=~~unsupported LMP feature~~ 0x1A)

In Expected Outcome:
Pass verdict:
The IUT transmits PDU LMP_not_accepted containing "~~unsupported LMP feature~~ Reason=0x1A " upon reception of LMP_page_mode_req.

Fail verdict:
The IUT does not transmit PDU LMP_not_accepted containing "Reason=0x1A" upon reception of PDU LMP_page_mode_req.

In 5.7.24.3 TP/LIH/BV-72-C (Reject Page Scan Negotiation):

In Figure 5.131: TP/LIH/BV-72-C:
LMP_not_accepted
(OpCode LMP_page_scan_mode_req,
reason=~~unsupported LMP feature~~ 0x1A)

In Expected Outcome:
Pass verdict:
The IUT transmits PDU LMP_not_accepted containing "~~Unsupported LMP Feature~~ Reason=0x1A" upon reception of PDU LMP_page_scan_mode_req.
Fail verdict:
The IUT does not transmit PDU LMP_not_accepted containing "~~Unsupported LMP Feature~~ Reason=0x1A" upon reception of PDU LMP_page_scan_mode_req.

2.18 HCI FUNCTIONAL SPECIFICATION – TEST SPECIFICATION IMPACTS

2.18.1 Erratum 564 Test Specification Impact

No test specification impact.

2.18.2 Erratum 542 Test Specification Impact

No test specification impact.

2.18.3 Erratum 540 Test Specification Impact

No test specification impact.

2.18.4 Erratum 570 Test Specification Impact

No test specification impact.

2.18.5 Erratum 410 Test Specification Impact

No test specification impact.

2.18.6 Erratum 455 Test Specification Impact

No test specification impact.

2.18.7 Erratum 439 Test Specification Impact

No test specification impact.

2.18.8 Erratum 694 Test Specification Impact

No test specification impact.

2.19 LOGICAL LINK CONTROL AND ADAPTATION PROTOCOL SPECIFICATION – TEST SPECIFICATION IMPACTS

2.19.1 Erratum 559 Test Specification Impact (addressed by approved TSE 576)

No test specification impact.

2.19.2 Erratum 494 Test Specification Impact

No test specification impact.

2.20 GENERIC ACCESS PROFILE – TEST SPECIFICATION IMPACTS

2.20.1 Erratum 536 Test Specification Impact

No test specification impact.

2.20.2 Erratum 573 Test Specification Impact

No test specification impact.

3 CORE SPECIFICATION VERSION 2.0 + EDR

3.1 COMPLIANCE REQUIREMENTS

3.1.1 Erratum 737 Replacement Text for Bluetooth Compliance Requirements Part of the Bluetooth Specifications

[Vol 0>Part B>Bluetooth Compliance Requirements: Page 39, Replace entire Part with the text below]

Bluetooth Compliance Requirements

1. INTRODUCTION

The Bluetooth Qualification Program Reference Document (PRD) is the primary reference document for the Bluetooth Qualification Program and defines its requirements, functions, and policies. The PRD is available on the Bluetooth Web site.

Passing the Bluetooth Qualification Process demonstrates a certain measure of compliance and interoperability, but because products are not tested for every aspect of this Bluetooth Specification, qualification does not guarantee compliance. Passing the Bluetooth Qualification Process only satisfies one condition of the license grant. The Member has the ultimate responsibility to ensure that the qualified product complies with this Bluetooth Specification and interoperates with other products.

2. SCOPE

This part of the Bluetooth Specification defines some fundamental terms used in the Bluetooth Qualification Program.

3. DEFINITIONS

Bluetooth Qualification Process – The process defined in the Bluetooth Qualification Program Reference Document (PRD) to qualify a design used in implementations of Bluetooth wireless technology.

Bluetooth Qualification Program – The Bluetooth Qualification Process together with other related requirements and processes as defined in the PRD.

Bluetooth Product – Any product containing an implementation of Bluetooth wireless technology that implements, at a minimum, all the mandatory requirements of either one or more of any of the protocol and profile parts of the Specification in compliance with such portion of the Specification.

3.1 Types of Bluetooth products

All Bluetooth Products shall be one of the following:

- Bluetooth End Product
- Bluetooth Controller Subsystem Product
- Bluetooth Host Subsystem Product
- Bluetooth Profile Subsystem Product
- Bluetooth Component Product
- Bluetooth Test Equipment
- Bluetooth Development Tool

Bluetooth End Product - A Bluetooth Product that implements, at a minimum, all mandatory requirements in Radio, Baseband, Link Manager, Logical Link Control and Adaptation Protocol, Service Discovery Protocol and Generic Access Profile parts of the Specification.

Bluetooth Subsystem Product - A Bluetooth Product that implements only a portion of the Specification in compliance with such portion of the Specification and in accordance with the mandatory requirements as defined herein. Bluetooth Subsystem Products can be qualified solely for distribution and the use of Bluetooth wireless technology in Bluetooth Subsystem Products require such Bluetooth Subsystem Products to be combined with one or more Bluetooth Product(s) such that the resulting combination satisfies the definition of either Bluetooth End Product, Bluetooth Development Tool or Bluetooth Test Equipment. There are three types of Bluetooth Subsystem Products as defined below:

- *Bluetooth Host Subsystem Product* – A Bluetooth Subsystem Product containing, at a minimum, all the mandatory requirements defined in the Host Controller Interface, Logical Link Control and Adaptation Protocol, Service Discovery Protocol and Generic Access Profile parts of this Specification, but none of the

protocols below Host Controller Interface (HCI). In addition, a Bluetooth Host Subsystem Product may contain, at a minimum, all the mandatory requirements defined in one or more of the protocols and profiles above HCI.

- *Bluetooth Controller Subsystem Product* – A Bluetooth Subsystem Product containing, at a minimum, all the mandatory requirements defined in the Bluetooth Radio, Baseband, Link Manager and HCI parts of this Specification, but none of the Protocols and Profiles above HCI.

- *Bluetooth Profile Subsystem Product* – A Bluetooth Subsystem Product containing, at a minimum, all the mandatory requirements defined in one or more of the profile specifications.

Bluetooth Component Product - A Bluetooth Product that does not meet the definition of a Bluetooth End Product. Bluetooth Component Products can be qualified solely for distribution and the use of Bluetooth wireless technology in Bluetooth Component Products require such Bluetooth Component Products to be combined with one or more Bluetooth Product(s) such that the resulting combination satisfies the definition of either Bluetooth End Product, Bluetooth Development Tool or Bluetooth Test Equipment.

Bluetooth Development Tool - A Bluetooth Product, solely intended to facilitate the development of new Bluetooth designs. Bluetooth Development Tools can be qualified solely for distribution and the use of the Bluetooth wireless technology for the purposes of development of new Bluetooth Products.

Bluetooth Test Equipment - A Bluetooth Product, intended to facilitate the testing of new Bluetooth Products. Bluetooth Test Equipment can be qualified solely for distribution and the use of the Bluetooth wireless technology in testing of new Bluetooth Products. Where necessary, Bluetooth Test Equipment may deviate from the Specification in order to fulfill the test purposes in the Bluetooth Test Specifications.

3.2 COMPLIANCE REQUIREMENTS – TEST SPECIFICATION IMPACTS

3.2.1 Erratum 737 Test Specification Impact

No test specification impact.

4 PROFILES

4.1 ADVANCED AUDIO DISTRIBUTION PROFILE SPECIFICATION (A2DP)

4.1.1 Erratum 447 ATRAC Service Capabilities

[Page 31, Section 4.6.2.1, Paragraph 1 original text states]

Table 4.18 shows the value of *Version* field for ATRAC family. The *Version* field contains one specific version of ATRAC family. Therefore, if the device supports both ATRAC and ATRAC3, for example, two sets of *Service Capabilities* shall be exchanged.

[Replace with]

Table 4.18 shows the value of *Version* field for ATRAC family. The *Version* field contains one specific version of ATRAC family. Therefore, if e.g. the device supports both ATRAC and ATRAC3, two different Stream End Points shall be used.

[End of changes for Erratum 447]

4.2 AUDIO/VIDEO DISTRIBUTION TRANSPORT PROTOCOL SPECIFICATION (AVDTP)

4.2.1 Erratum 310 Corrupted Messages

[Page 64, Section 8.4.6, paragraph 1 original text states]

The receiver of an AVDTP signalling message **shall** not interpret or pass corrupted messages to the upper layer. Those messages are discarded and not signalled to the local upper layer; no signalling message is returned to the sender. Possible corrupted messages are:

[Replace with]

'The receiver of an AVDTP signalling message shall not interpret corrupted messages. Those messages are discarded and no signalling message is returned to the sender if no error code is applicable. Possible corrupted messages are:

[End of changes for Erratum 310]

4.2.2 Erratum 502 Reporting Capability Error

[Page 82, Section 8.18.6.2, after ERROR CODE tables insert text]

In case the format of the Reporting Service Capability is not correct, the BAD_LENGTH or BAD_PAYLOAD_FORMAT error code shall be used.

[End of changes for Erratum 502]

4.2.3 Erratum 501 Discover Response Clarification

[Page 66, Section 8.6.2, add text to the end of paragraph 1]

There shall be at least one SEP in an AVDTP_DISCOVER_RSP.

[End of changes for Erratum 501]

4.2.4 Erratum 500 Symmetric Requests

[Page 82, Section 8.18.6.2, original text states]

ACP to INT, Procedure Error Codes			
Error ID	*Related Signalling command*	*Error Abbreviation*	*Error Description*
0x31	All messages	BAD_STATE	Indicates that the ACP state machine is in an invalid state in order to process the signal.

[Replace with]

ACP to INT, Procedure Error Codes			
Error ID	**Related Signalling command**	**Error Abbreviation**	**Error Description**
0x31	All messages	BAD_STATE	Indicates that the ACP state machine is in an invalid state in order to process the signal. This also includes the situation when an INT receives a request for the same command that it is currently expecting a response for, see 9.11.

[Page 99, Section 9.11, add the following text below paragraph 1]

In case an INT receives a request for the same command that it is expecting a response for, it may reject the command and may use a random time offset for a retransmission to avoid deadlock.

[End of changes for Erratum 500]

4.2.5 Erratum 322 Update of RTP Reference

[Page 100, Section 10, original reference text states]

[3] IETF RFC1889 – RTP, A Transport Protocol for Real-Time Applications

[Replace with]

[3] IETF RFC3550 / RFC1889 (obsolete) RTP, A Transport Protocol for Real-Time Applications

[End of changes for Erratum 322]

4.2.6 Erratum 304 Clarify Relation SEP to Codec

[Page 26, Section 5.3, after last paragraph in section add new paragraph stating]

Each codec supported by the application shall use a separate Stream End Point, but there can be multiple Stream End Points for the same codec.

[End of changes for Erratum 304]

4.2.7 Erratum 470 Multiple Stream Establishments

[Page 39, Section 6.10, after paragraph in section add new paragraph stating]

For the establishment of multiple transport sessions to one remote device, the Stream Establishment Procedure for one session, including all related L2CAP channel establishments, shall be completed before the Stream Establishment Procedure for the next session is initiated.

[End of changes for Erratum 470]

4.2.8 Erratum 332 Service Category in Reject

[Page 68, Section 8.8.3, Note 2 original text states]

Note 2. Contains the value of the first Service Category to fail.

[Replace with]

Note 2. Contains the value of the first Service Category to fail. In case no Service Category applies due to the nature of the error, the Service Category field shall be set to 0x0. The INT should ignore the Service Category if it is not applicable to the error code.

[End of changes for Erratum 332]

4.3 AUDIO/VIDEO REMOTE CONTROL PROFILE (AVRCP)

4.3.1 Erratum 90 AV/C Reference Update

[Page 43, Section 12, original reference text states]

[1] 1394 Trade Association , AV/C Digital Interface Command Set – General Specification, Version 4.0, Document No. 1999026 (http://www.1394ta.org)

[Replace with]

[1] 1394 Trade Association , AV/C Digital Interface Command Set – General Specification, Version 4.0, Document No. 1999026 and AV/C Digital Interface Command Set General Specification Version 4.1, Document No. 2001012 (http://www.1394ta.org)

[End of changes for Erratum 90]

4.3.2 Erratum 71 L2CAP Requirements

[Page 36, Section 6.2, original text states]

6.2 Signalling

Only the CT may issue an L2CAP Connection Request within the execution of this profile. Other than that, AVRCP does not impose any additional restrictions or requirements on L2CAP signalling.

[Replace with]

6.2 Signalling

AVRCP does not impose any restrictions or requirements on L2CAP signalling.

[End of changes for Erratum 71]

4.4 GENERIC OBJECT EXCHANGE PROFILE (GOEP)

4.4.1 Erratum 582 Non Pairable Mode Should Not be Mandatory

[Page 27, Table 7.1 item number 3 original text states]

3	Pairing modes		
	Non-pairable mode	N/A	M
	Pairable mode	N/A	M

Table 7.1: Modes

[Replace with]

3	Pairing modes		
	Non-parable mode	N/A	O
	Pairable mode	N/A	M

[End of changes for Erratum 582]

4.5 HANDS-FREE PROFILE (HFP)

4.5.1 Erratum 586 Interop Problems Require Clarification of AT+CHUP Command Action

[Page 56, Section 4.24.2 original bullet states]

- **AT+CHUP**

  Standard hang-up AT command. Refer to Section 6.5 in [2].

[Replace with]

- **AT+CHUP**

  Standard hang-up AT command. Execution command causes the AG to terminate the currently active call. This command shall have no impact on the state of any held call.

[End of changes for Erratum 586]

4.5.2 Erratum 635 ATD>nnn; Limits the Number of Dialing Digits

[Page 37 and 54, Sections 4.14 and 4.24.2 for the original text]

ATD>nnn

[Replace with]

ATD>nnn..

[End of changes for Erratum 635]

4.5.3 Erratum 13 Contradicting Requirements in References to GAP

[Page 69, Table 6.2, original text states]

Procedure	Support in AG
Initiation of general inquiry	M
Initiation of general bonding	M
Initiation of dedicated bonding	M

Table 6.2 Idle mode procedures

[Replace with]

Procedure	Support in AG
Initiation of general inquiry	M
Initiation of general bonding	M
Initiation of dedicated bonding	O

Table 6.2 Idle mode procedures

[End of changes for Erratum 13]

4.6 HUMAN INTERFACE DEVICE (HID)

4.6.1 Erratum 650 Inconsistency with the Declaration of HIDParserVersion in the Specification

[Page 83, Table 18 original text states]

Attribute	ID	Type & Size[1]	Required	Section
HIDBatteryPower	0x0209	Bool8	O	7.11.9
HIDBootDevice	0x020E	Bool8	M	7.11.11
HIDCountryCode	0x0203	uint8	M	7.11.3
HIDDescriptorList	0x0206	Sequence	M	7.11.6
HIDDeviceReleaseNumber	0x0200	uint16	O	7.11.1
HIDDeviceSubclass	0x0202	uint8	M	7.11.2
HIDLANGIDBaseList	0x0207	Sequence	M	7.11.7
HIDNormallyConnectable	0x020D	Bool 8	O	7.11.13
HIDParserVersion	0x020B	Uint16	M	7.11.14
HIDProfileVersion	0x0201	uint16	M	0
HIDReconnectInitiate	0x0205	Bool8	M	7.11.5
HIDRemoteWake	0x020A	Bool8	O	7.11.10
HIDSDPDisable	0x0208	Bool8	O	7.11.8
HIDSupervisionTimeout	0x020C	Uint16	O	7.11.12
HIDVirtualCable	0x0204	Bool8	M	7.11.4

[Replace with]

Attribute	ID	Type & Size[1]	Required	Section
HIDBatteryPower	0x0209	Bool8	O	7.11.9
HIDBootDevice	0x020E	Bool8	M	7.11.11
HIDCountryCode	0x0203	Uint8	M	7.11.3
HIDDescriptorList	0x0206	Sequence	M	7.11.6
HIDDeviceReleaseNumber	0x0200	Uint16	O	7.11.1
HIDDeviceSubclass	0x0202	Uint8	M	7.11.2
HIDLANGIDBaseList	0x0207	Sequence	M	7.11.7
HIDNormallyConnectable	0x020D	Bool8	O	7.11.13
HIDParserVersion	0x0201	Uint16	M	0
HIDProfileVersion	0x020B	Uint16	M	7.11.14
HIDReconnectInitiate	0x0205	Bool8	M	7.11.5
HIDRemoteWake	0x020A	Bool8	O	7.11.10
HIDSDPDisable	0x0208	Bool8	O	7.11.8
HIDSupervisionTimeout	0x020C	Uint16	O	7.11.12
HIDVirtualCable	0x0204	Bool8	M	7.11.4

[End of changes for Erratum 650]

4.6.2 Erratum 542 Correction to HIDSDPDisable Handling Description

[Page 89, Section 7.11.8, second paragraph original text states]

If the HIDSDPDisable attribute is true, the host shall issue an LMP_DisconnectReq primitive to the SDP channel of the device before attempting to open either the Control or the Interrupt channel. If after opening the Control and Interrupt channels the host decides that it wants to access SDP information on the device, then the host shall first close both the Control and Interrupt channels before opening the SDP channel.

[Replace with]

If the HIDSDPDisable attribute is true, the host shall issue an L2CA_DisconnectReq primitive to the SDP channel of the device, and await a successful L2CA_DisconnectRsp from the device, before attempting to open either the Control or the Interrupt channel. If after opening the Control and Interrupt channels the host decides that it wants to access SDP information on the device, then the host shall first close both the Control and Interrupt channels before opening the SDP channel.

[End of changes for Erratum 542]

4.6.3 Erratum 674 HIDDeviceReleaseNumber

[Page 84, Section 7.11.1, original text states]

7.11.1 HIDDeviceReleaseNumber

Attribute Name	Attribute ID	Attribute Value Type
HIDDeviceReleaseNumber	0x0200	16-bit unsigned integer

Description A numeric expression identifying the device release number in Binary-Coded Decimal. This is a vendor-assigned field, which defines the version of the product identified by the Bluetooth Device Identification [13] VendorID and ProductID attributes. This attribute is intended to differentiate between versions of products with identical VendorIDs and ProductIDs. The value of the field is 0xJJMN for version JJ.M.N (JJ – major version number, M – minor version number, N – sub-minor version number); e.g., version 2.1.3 is represented with value 0x0213 and version 2.0.0 is represented with a value of 0x0200. When upward-compatible changes are made to the device, the minor version number will be incremented. If incompatible changes are made to the device, the major version number will be incremented. NOTE: This attribute should not be used for new designs as it is redundant with the Device Identification Rev. 1.0 “Version Attribute”. HIDParserVersion

Attribute Name	Attribute ID	Attribute Value Type
HIDParserVersion	0x0201	16-bit unsigned integer

Description

Each version of a profile is assigned a 16-bit unsigned integer version number of the base HID Specification [4] that the device was designed to. The value of the field is 0xJJMN for version JJ.M.N (JJ – major version number, M – minor version number, N – sub-minor version number); e.g., version 2.1.3 is represented with value 0x0213 and version 2.0.0 is represented with a value of 0x0200.

[Replace with]

7.11.1 HIDDeviceReleaseNumber

Attribute Name	Attribute ID	Attribute Value Type
HIDDeviceReleaseNumber	0x0200	16-bit unsigned integer

Description A numeric expression identifying the device release number in Binary-Coded Decimal. This is a vendor-assigned field, which defines the version of the product identified by the Bluetooth Device Identification [13] VendorID and ProductID attributes. This attribute is intended to differentiate between versions of products with identical VendorIDs and ProductIDs. The value of the field is 0xJJMN for version JJ.M.N (JJ – major version number, M – minor version number, N – sub-minor version number); e.g., version 2.1.3 is represented with value 0x0213 and version 2.0.0 is represented with a value of 0x0200. When upward-compatible changes are made to the device, the minor version number will be incremented. If incompatible changes are made to the device, the major version number will be incremented. NOTE: This attribute should not be used for new designs as it is redundant with the Device Identification Rev. 1.0 “Version Attribute”.

7.11.2 HIDParserVersion

Attribute Name	Attribute ID	Attribute Value Type
HIDParserVersion	0x0201	16-bit unsigned integer

Description

Each version of a profile is assigned a 16-bit unsigned integer version number of the base HID Specification [4] that the device was designed to. The value of the field is 0xJJMN for version JJ.M.N (JJ – major version number, M – minor version number, N – sub-minor version number); e.g., version 2.1.3 is represented with value 0x0213 and version 2.0.0 is represented with a value of 0x0200.

[Pages 85-92, Sections 7.11.2 – 7.11.14, increment all third level headings by one. These Sections are referenced throughout the HID Specification. This erratum causes a total of 75 updates to occur.]

Example:
7.11.2 shall become 7.11.3

[End of changes for Erratum 674]

4.6.4 Erratum 241 Note About Drift in Low Power Modes Contains Wrong Figures

[Page 39, Section 6.5.1.4 second paragraph original text states]

Note: Due to the Bluetooth clock accuracy specifications (+/- 20 ppm), a worst-case clock accuracy situation could potentially require a maximum beacon interval of less than 250 m s in order for devices in Park, Sniff, or Hold modes to stay synchronized. The best link manager implementations will widen the receive window as the beacon interval is increased, but at the expense of additional power consumption. If loss of synchronization occurs, link loss will occur and the link must be re-established by the normal paging and connection establishment procedures.

[Replace with]

Note: Due to the Bluetooth clock accuracy specifications (+/- 250 ppm for Park, Sniff and Hold mode), a worst-case clock accuracy situation could potentially require a maximum beacon interval of less than 20 ms in order for devices in Park, Sniff or Hold modes to stay synchronized if those do not support widening the receive window in the low power modes. Normally link controller implementations will widen the receive window (like proposed in the Baseband specification) to compensate clock drift on local and remote side as the beacon interval is increased. The expense of additional power consumption for widening the synchronization window is marginal compared to the power saving achieved using the low power oscillator in Park, Sniff or Hold mode. In general if loss of synchronization occurs, link loss will occur and the link must be re-established by the normal paging and connection establishment procedures. This is common to all (Active, Park, Sniff and Hold) modes.

[End of changes for Erratum 241]

4.7 ADVANCED AUDIO DISTRIBUTION PROFILE SPECIFICATION (A2DP) – TEST SPECIFICATION IMPACTS

4.7.1 Erratum 447 Test Specification Impact

No test specification impact.

4.8 AUDIO/VIDEO DISTRIBUTION TRANSPORT PROTOCOL SPECIFICATION (AVDTP)– TEST SPECIFICATION IMPACTS

4.8.1 Erratum 310 Test Specification Impact

No test specification impact.

4.8.2 Erratum 502 Test Specification Impact

See Test Specification Errata Request Number 662 below:

TS ERRATA REQUEST 662; Reporting Capability Error (E502)

Record Details:

Category: Correction

TSE#: 662

Record Date: 2004-09-15

Subject: Reporting Capability Error (E502)

Status: Approved

Requestor: John Moring

Specification Details:

Test Spec: Audio/Video Dist. Transport Protocol Conformance

Test Spec Version:

Core Spec Version: Profiles

Test Case Number Description:

TP/LIH/BV-15-C
TP/TRA/REP/BI-01-C

Problem Statement & Suggested Change:

Erratum 502 adds the requirement 'In case the format of the Reporting Service Capability is not correct, the BAD_LENGTH or BAD_PAYLOAD_FORMAT error code shall be used. ' This affects test case TP/TRA/REP/BI-01-C.

Proposed Change:

In Table 3, adjacent to the first TP/TRA/REP/BI-01-C, change "BAD_REPORT_TYPE" to "BAD_LENGTH." In Table 3, adjacent to the second TP/TRA/REP/BI-01-C, change "BAD_REPORT_FORMAT" to "BAD_PAYLOAD_FORMAT." In the Success Verdict of TP/TRA/REP/BI-01-C, change "code = 0x65 (BAD_REPORT_FORMAT)" to "code = 0x11 (BAD_LENGTH) or code 0x18 (BAD_PAYLOAD_FORMAT)".

4.8.3 Erratum 501 Test Specification Impact

No test specification impact.

4.8.4 Erratum 500 Test Specification Impact

No test specification impact.

4.8.5 Erratum 322 Test Specification Impact

No test specification impact.

4.8.6 Erratum 304 Test Specification Impact

No test specification impact.

4.8.7 Erratum 470 Test Specification Impact

No test specification impact.

4.8.8 Erratum 332 Test Specification Impact

No test specification impact.

4.9 AUDIO/VIDEO REMOTE CONTOL PROFILE (AVRCP)– TEST SPECIFICATION IMPACTS

4.9.1 Erratum 90 Test Specification Impact

No test specification impact.

4.9.2 Erratum 71 Test Specification Impact

No test specification impact.

4.10 GENERIC OBJECT EXCHANGE PROFILE (GOEP) – TEST SPECIFICATION IMPACTS

4.10.1 Erratum 582 Test Specification Impact

No test specification impact.

4.11 HANDS-FREE PROFILE (HFP) – TEST SPECIFICATION IMPACTS

4.11.1 Erratum 586 Test Specification Impact

No test specification impact.

4.11.2 Erratum 635 Test Specification Impact

No test specification impact.

4.11.3 Erratum 13 Test Specification Impact

See Test Specification Errata Request Number 658 below:

TS ERRATA REQUEST 658 Security modes 2 & 3; general bonding (E13)

Requested by:	Company:	E-mail address:	Date of Request:
John Moring	Bluetooth SIG	john@moring.net	2004-9-15

Reference to Bluetooth Test Specifications

Test Specification:	Version:	Test Case (if applicable):	Test Case Category (if applicable):
Hands Free Profile PICS	**HFP_PICS_1.1**		
Page:	Section:	Paragraph:	

Owner of TS:	E-mail address:	Status/date (Submitted, Review, Sign off, Closed)	
TS errata class: (1-not testable anymore, 2–still testable)	TS errata type: (Correction/Clarification/Editorial)	ID No.:	Approved:
	Clarification	658	

Problem Statement (and / or existing text)

A clarification of the mandatory HFP features is suggested in erratum 13. This would be reflected in the PICS via changing Table 6, line 4, "Initiation of dedicated bonding" from M to O; or better yet, remove HFP PICS Table 6, line 4, "Initiation of dedicated bonding" altogether, as it offers no change to the inherited GAP requirement.

Correction Proposal

Table 6: Requirements towards the Generic Access Profile (AG)

Item	Capability	Reference to PICS	Status	Support
1	Pairable mode	K:1.1/7	M	☐
2	Initiation of general inquiry	K:1.3/1	M	☐
3	Initiation of general bonding	K:1.3/5	M	☐
4	Initiation of dedicated bonding	K:1.3/6	M	☐

4.12 HUMAN INTERFACE DEVICE (HID) – TEST SPECIFICATION IMPACTS

4.12.1 Erratum 650 Test Specification Impact

No test specification impact.

4.12.2 Erratum 542 Test Specification Impact

No test specification impact.

4.12.3 Erratum 674 Test Specification Impact

No test specification impact.

4.12.4 Erratum 241 Test Specification Impact

No test specification impact.