

Specification of the Bluetooth System

Wireless connections made easy

Master Table of Contents & Compliance Requirements

Covered Core Package version:
2.0 + EDR
Current Master TOC issued:
4 November 2004

Revision History

The Revision History is shown in the Appendix.

Contributors

The persons who contributed to this specification are listed in the Appendix.

Web Site

This specification can also be found on the Bluetooth web site:
http://www.bluetooth.com

Disclaimer and Copyright Notice

The copyright in these specifications is owned by the Promoter Members of Bluetooth SIG, Inc. (“Bluetooth SIG”). Use of these specifications and any related intellectual property (collectively, the “Specification”), is governed by the Promoters Membership Agreement among the Promoter Members and Bluetooth SIG (the “Promoters Agreement”), certain membership agreements between Bluetooth SIG and its Adopter and Associate Members (the “Membership Agreements”) and the Bluetooth Specification Early Adopters Agreements (“1.2 Early Adopters Agreements”) among Early Adopter members of the unincorporated Bluetooth special interest group and the Promoter Members (the “Early Adopters Agreement”). Certain rights and obligations of the Promoter Members under the Early Adopters Agreements have been assigned to Bluetooth SIG by the Promoter Members.

Use of the Specification by anyone who is not a member of Bluetooth SIG or a party to an Early Adopters Agreement (each such person or party, a “Member”), is prohibited. The legal rights and obligations of each Member are governed by their applicable Membership Agreement, Early Adopters Agreement or Promoters Agreement. No license, express or implied, by estoppel or otherwise, to any intellectual property rights are granted herein.

Any use of the Specification not in compliance with the terms of the applicable Membership Agreement, Early Adopters Agreement or Promoters Agreement is prohibited and any such prohibited use may result in termination of the applicable Membership Agreement or Early Adopters Agreement and other liability permitted by the applicable agreement or by applicable law to Bluetooth SIG or any of its members for patent, copyright and/or trademark infringement.

THE SPECIFICATION IS PROVIDED “AS IS” WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, SATISFACTORY QUALITY, OR REASONABLE SKILL OR CARE, OR ANY WAR-

RANTY ARISING OUT OF ANY COURSE OF DEALING, USAGE, TRADE PRACTICE, PROPOSAL, SPECIFICATION OR SAMPLE.

Each Member hereby acknowledges that products equipped with the Bluetooth® technology ("Bluetooth® Products") may be subject to various regulatory controls under the laws and regulations of various governments worldwide. Such laws and regulatory controls may govern, among other things, the combination, operation, use, implementation and distribution of Bluetooth® Products. Examples of such laws and regulatory controls include, but are not limited to, airline regulatory controls, telecommunications regulations, technology transfer controls and health and safety regulations. Each Member is solely responsible for the compliance by their Bluetooth® Products with any such laws and regulations and for obtaining any and all required authorizations, permits, or licenses for their Bluetooth® Products related to such regulations within the applicable jurisdictions. Each Member acknowledges that nothing in the Specification provides any information or assistance in connection with securing such compliance, authorizations or licenses. NOTHING IN THE SPECIFICATION CREATES ANY WARRANTIES, EITHER EXPRESS OR IMPLIED, REGARDING SUCH LAWS OR REGULATIONS.

ALL LIABILITY, INCLUDING LIABILITY FOR INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHTS OR FOR NONCOMPLIANCE WITH LAWS, RELATING TO USE OF THE SPECIFICATION IS EXPRESSLY DISCLAIMED. BY USE OF THE SPECIFICATION, EACH MEMBER EXPRESSLY WAIVES ANY CLAIM AGAINST BLUETOOTH SIG AND ITS PROMOTER MEMBERS RELATED TO USE OF THE SPECIFICATION.

Bluetooth SIG reserves the right to adopt any changes or alterations to the Specification as it deems necessary or appropriate.

Copyright © 1999, 2000, 2001, 2002, 2003, 2004

Agere Systems, Inc.,
Ericsson Technology Licensing, AB,
IBM Corporation,
Intel Corporation,
Microsoft Corporation,
Motorola, Inc.,
Nokia Corporation,
Toshiba Corporation

*Third-party brands and names are the property of their respective owners.



MASTER TABLE OF CONTENTS

This table of contents (TOC) covers the entire Bluetooth Specification.
In addition each volume has a TOC and each part of a volume is preceded by a detailed TOC.

MASTER TOC FOR THE BLUETOOTH SPECIFICATION

In the following table:

- The TOC for each Volume starts at the top of a page.
- The Volume No. is followed by the name of the Volume written in red.

Note: Each Volume is a self contained book which is published and updated separately and is equipped with a TOC of its own. However, this Master TOC is also revised as soon as any of the other Volumes are updated.

- A Volume cover one or more Parts (A, B, etc.), each Part can be viewed independently and has its own TOC.

Red or blue text on the following pages indicates hypertext links that will take you directly to the indicated section, on condition that you have access to a complete specification.

Specification Volume 0
Master Table of Contents & Compliance Requirements

Part A
MASTER TABLE OF CONTENTS

Master TOC for the Bluetooth Specification 7

Part B
BLUETOOTH COMPLIANCE REQUIREMENTS

Contents 41
1 Introduction 43
2 Scope 45
3 Definitions 47
3.1 Types of Bluetooth Products 47
4 Core Configurations 49
4.1 Specification Naming Conventions 49
4.2 EDR Configurations 49

Part C
APPENDIX

Contents 53
1 Revision History 55
1.1 [vol 0] Master TOC & Compliance Requirements 55
1.1.1 Bluetooth Compliance Requirements 55
1.2 [Vol 1] Architecture & Terminology Overview 55
1.3 [Vol 2 & 3] Core System Package 56
2 Contributors 59
2.1 [vol 0] Master TOC & Compliance Requirements 59
2.1.1 Part B: Bluetooth Compliance Requirements 59
2.2 [vol 1] Architecture &Terminology Overview 59
2.2.1 Part A: Architectural Overview 59
2.2.2 Part B: Acronyms & Abbreviations 59
2.2.3 Part C: Changes from Bluetooth Specification v1.1 60
2.3 [Vol 2] Core System Package, Controller 61
2.3.1 Part A: Radio Specification 61
2.3.2 Part B: Baseband Specification 62

2.3.3 Part C: Link Manager Protocol 64
2.3.4 Part D: Error Codes 66
2.3.5 Part E: Bluetooth Host Controller Interface Functional Specification66
2.3.6 Part F: Message Sequence Charts 67
2.3.7 Part G: Sample Data 68
2.3.8 Part H: Security Specification 68
2.4 [Vol 3] Core System Package, Host 69
2.4.1 Part A: Logical Link Control and Adaptation Protocol Specification69
2.4.2 Part B: Service Discovery Protocol (SDP) 70
2.4.3 Part C Generic Access Profile 71
2.4.4 Part D: Test Support 71

Specification Volume 1
Architecture & Terminology Overview

Table of Contents **5**

Part A
ARCHITECTURE

Contents **11**

1 General Description **13**
1.1 Overview of Operation 13
1.2 Nomenclature 15

2 Core System Architecture **21**
2.1 Core Architectural Blocks 24
2.1.1 Channel manager 24
2.1.2 L2CAP resource manager 24
2.1.3 Device manager 25
2.1.4 Link manager 25
2.1.5 Baseband resource manager 25
2.1.6 Link controller 26
2.1.7 RF 26

3 Data Transport Architecture **27**
3.1 Core Traffic Bearers 28
3.1.1 Framed data traffic 29
3.1.2 Unframed data traffic 30
3.1.3 Reliability of traffic bearers 30
3.2 Transport Architecture Entities 32
3.2.1 Bluetooth generic packet structure 32
3.3 Physical Channels 34
3.3.1 Basic piconet channel 35
3.3.2 Adapted piconet channel 36
3.3.3 Inquiry scan channel 37
3.3.4 Page scan channel 38
3.4 Physical Links 39
3.4.1 Links supported by the basic and adapted piconet physical channel 40
3.4.2 Links supported by the scanning physical channels 41
3.5 Logical Links and Logical Transports 41
3.5.1 Casting 43
3.5.2 Scheduling and acknowledgement scheme 43

3.5.3 Class of data 44
3.5.4 Asynchronous connection-oriented (ACL) 44
3.5.5 Synchronous connection-oriented (SCO) 45
3.5.6 Extended synchronous connection-oriented (eSCO) 46
3.5.7 Active slave broadcast (ASB) 46
3.5.8 Parked slave broadcast (PSB) 47
3.5.9 Logical links 48
3.5.10 ACL Control Logical Link (ACL-C) 49
3.5.11 User Asynchronous/Isochronous Logical Link (ACL-U) 49
3.5.12 User Synchronous/Extended Synchronous Logical Links (SCO-S/eSCO-S) 49
3.6 L2CAP Channels 50

4 Communication Topology 51
4.1 Piconet Topology 51
4.2 Operational Procedures and Modes 53
4.2.1 Inquiry (Discovering) Procedure 53
4.2.2 Paging (Connecting) Procedure 54
4.2.3 Connected mode 54
4.2.4 Hold mode 55
4.2.5 Sniff mode 55
4.2.6 Parked state 56
4.2.7 Role switch procedure 56
4.2.8 Enhanced Data Rate 57

Part B
ACRONYMS & ABBREVIATIONS

1 List of Acronyms and Abbreviations 61
2 Abbreviations of the Specification Names 69

Part C
CORE SPECIFICATION CHANGE HISTORY

Contents 73
1 Changes from V1.1 to V1.2 75
1.1 New Features 75
1.2 Structure Changes 75
1.3 Deprecated Specifications 75
1.4 Deprecated Features 76
1.5 Changes in Wording 76
1.6 Nomenclature Changes 76

2 Changes from V1.2 to V2.0 + EDR 77
2.1 New Features 77
2.2 Deprecated Features 77

Part D
MIXING OF SPECIFICATION VERSIONS

1 Mixing of Specification Versions 81
1.1 features and their types 82

Part E
IEEE LANGUAGE

Contents 85

1 Use of IEEE Language 87
1.1 Shall 87
1.2 Must 88
1.3 Will 88
1.4 Should 88
1.5 May 88
1.6 Can 89

Specification Volume 2
Core System Package
[Controller volume]

Table of Contents **5**

Part A
RADIO SPECIFICATION

Contents **25**
1 Scope **27**
2 Frequency Bands and Channel Arrangement **29**
3 Transmitter Characteristics **31**
3.1 Basic Rate 32
3.1.1 Modulation Characteristics 32
3.1.2 Spurious Emissions 33
3.1.3 Radio Frequency Tolerance 34
3.2 Enhanced Data Rate 34
3.2.1 Modulation Characteristics 34
3.2.2 Spurious Emissions 37
3.2.3 Radio Frequency Tolerance 38
3.2.4 Relative Transmit Power 39
4 Receiver Characteristics **41**
4.1 Basic Rate 41
4.1.1 Actual Sensitivity Level 41
4.1.2 Interference Performance 41
4.1.3 Out-of-Band Blocking 42
4.1.4 Intermodulation Characteristics 42
4.1.5 Maximum Usable Level 43
4.1.6 Receiver Signal Strength Indicator 43
4.1.7 Reference Signal Definition 43
4.2 Enhanced Data Rate 43
4.2.1 Actual Sensitivity Level 43
4.2.2 BER Floor Performance 43
4.2.3 Interference Performance 43
4.2.4 Maximum Usable Level 44
4.2.5 Out-of-Band and Intermodulation Characteristics 45
4.2.6 Reference Signal Definition 45
5 Appendix A **47**

5.1 Nominal Test Conditions 47
5.1.1 Nominal temperature 47
5.1.2 Nominal power source 47
5.2 Extreme Test Conditions 48
5.2.1 Extreme temperatures 48
5.2.2 Extreme power source voltages 48
6 Appendix B 49
7 Appendix C 51
7.1 Enhanced Data Rate Modulation Accuracy 51

Part B
BASEBAND SPECIFICATION

Contents 57
1 General Description 61
1.1 Bluetooth Clock 62
1.2 Bluetooth Device Addressing 64
1.2.1 Reserved addresses 64
1.3 Access Codes 65
2 Physical Channels 67
2.1 Physical Channel Definition 68
2.2 Basic Piconet Physical Channel 68
2.2.1 Master-slave definition 68
2.2.2 Hopping characteristics 69
2.2.3 Time slots 69
2.2.4 Piconet clocks 70
2.2.5 Transmit/receive timing 70
2.3 Adapted Piconet Physical Channel 73
2.3.1 Hopping characteristics 73
2.4 Page Scan Physical Channel 74
2.4.1 Clock estimate for paging 74
2.4.2 Hopping characteristics 74
2.4.3 Paging procedure timing 75
2.4.4 Page response timing 76
2.5 Inquiry Scan Physical Channel 78
2.5.1 Clock for inquiry 78
2.5.2 Hopping characteristics 78
2.5.3 Inquiry procedure timing 78
2.5.4 Inquiry response timing 78
2.6 Hop Selection 80

2.6.1 General selection scheme 80
2.6.2 Selection kernel 84
2.6.3 Adapted hop selection kernel 87
2.6.4 Control word 88

3 Physical Links 93
3.1 Link Supervision 93

4 Logical Transports 95
4.1 General 95
4.2 Logical Transport Address (LT_ADDR) 95
4.3 Synchronous Logical Transports 96
4.4 Asynchronous Logical Transport 96
4.5 Transmit/Receive Routines 97
4.5.1 TX Routine 97
4.5.2 RX routine 100
4.5.3 Flow control 101
4.6 Active Slave Broadcast Transport 102
4.7 Parked Slave Broadcast Transport 103
4.7.1 Parked member address (PM_ADDR) 103
4.7.2 Access request address (AR_ADDR) 103

5 Logical Links 105
5.1 Link Control Logical Link (LC) 105
5.2 ACL Control Logical Link (ACL-C) 105
5.3 User Asynchronous/Isochronous Logical Link (ACL-U) 105
5.3.1 Pausing the ACL-U logical link 106
5.4 User Synchronous Data Logical Link (SCO-S) 106
5.5 User Extended Synchronous Data Logical Link (eSCO-S) 106
5.6 Logical Link Priorities 106

6 Packets 107
6.1 General Format 107
6.1.1 Basic Rate 107
6.1.2 Enhanced Data Rate 107
6.2 Bit Ordering 108
6.3 Access Code 109
6.3.1 Access code types 109
6.3.2 Preamble 110
6.3.3 Sync word 110
6.3.4 Trailer 113
6.4 Packet Header 114
6.4.1 LT_ADDR 114

6.4.2 TYPE 114
6.4.3 FLOW 115
6.4.4 ARQN 115
6.4.5 SEQN 115
6.4.6 HEC 115
6.5 Packet Types 116
6.5.1 Common packet types 117
6.5.2 SCO packets 121
6.5.3 eSCO packets 122
6.5.4 ACL packets 124
6.6 Payload Format 126
6.6.1 Synchronous data field 126
6.6.2 Asynchronous data field 128
6.7 Packet Summary 132
7 Bitstream Processing 135
7.1 Error Checking 136
7.1.1 HEC generation 136
7.1.2 CRC generation 137
7.2 Data Whitening 139
7.3 Error Correction 140
7.4 FEC Code: Rate 1/3 140
7.5 FEC Code: Rate 2/3 141
7.6 ARQ Scheme 142
7.6.1 Unnumbered ARQ 142
7.6.2 Retransmit filtering 145
7.6.3 Flushing payloads 148
7.6.4 Multi-slave considerations 148
7.6.5 Broadcast packets 148
8 Link Controller Operation 151
8.1 Overview of States 151
8.2 Standby State 152
8.3 Connection Establishment Substates 152
8.3.1 Page scan substate 152
8.3.2 Page substate 154
8.3.3 Page response substates 157
8.4 Device Discovery Substates 161
8.4.1 Inquiry scan substate 162
8.4.2 Inquiry substate 163
8.4.3 Inquiry response substate 164

8.5 Connection State 165
8.6 Active Mode 166
8.6.1 Polling in the active mode 167
8.6.2 SCO 167
8.6.3 eSCO 169
8.6.4 Broadcast scheme 171
8.6.5 Role switch 173
8.6.6 Scatternet 175
8.6.7 Hop sequence switching 176
8.6.8 Channel classification and channel map selection 179
8.6.9 Power Management 180
8.7 sniff Mode 181
8.7.1 Sniff Transition Mode 182
8.8 Hold Mode 183
8.9 Park State 183
8.9.1 Beacon train 184
8.9.2 Beacon access window 187
8.9.3 Parked slave synchronization 188
8.9.4 Parking 189
8.9.5 Master-initiated unparking 190
8.9.6 Slave-initiated unparking 190
8.9.7 Broadcast scan window 191
8.9.8 Polling in the park state 191

9 Audio 193
9.1 LOG PCM CODEC 193
9.2 CVSD CODEC 193
9.3 Error Handling 196
9.4 General Audio Requirements 196
9.4.1 Signal levels 196
9.4.2 CVSD audio quality 196

10 List of Figures 197

11 List of Tables 201

12 Appendix 201
Appendix A: General Audio Recommendations 202
Appendix B: Timers 205
Appendix C:Recommendations for AFH Operation in Park, Hold and Sniff 207

Part C
LINK MANAGER PROTOCOL

Contents **211**

1 Introduction **213**

2 General Rules **215**
2.1 Message Transport 215
2.2 Synchronization 215
2.3 Packet Format 216
2.4 Transactions 217
2.4.1 LMP Response Timeout 218
2.5 Error Handling 218
2.5.1 Transaction collision resolution 219
2.6 Procedure Rules 219
2.7 General Response Messages 220
2.8 LMP Message Constraints 220

3 Device Features **221**
3.1 General Description 221
3.2 Feature Definitions 221
3.3 Feature Mask Definition 226
3.4 Link Manager Interoperability policy 228

4 Procedure Rules **229**
4.1 Connection Control 229
4.1.1 Connection establishment 229
4.1.2 Detach 230
4.1.3 Power control 231
4.1.4 Adaptive frequency hopping 233
4.1.5 Channel classification 236
4.1.6 Link supervision 238
4.1.7 Channel quality driven data rate change (CQDDR) 239
4.1.8 Quality of service (QoS) 240
4.1.9 Paging scheme parameters 242
4.1.10 Control of multi-slot packets 243
4.1.11 Enhanced Data Rate 243
4.2 Security 245
4.2.1 Authentication 245
4.2.2 Pairing 247
4.2.3 Change link key 250
4.2.4 Change current link key type 251
4.2.5 Encryption 253

4.2.6 Request supported encryption key size 257
4.3 Informational Requests 258
4.3.1 Timing accuracy 258
4.3.2 Clock offset 259
4.3.3 LMP version 259
4.3.4 Supported features 260
4.3.5 Name request 262
4.4 Role Switch 263
4.4.1 Slot offset 263
4.4.2 Role switch 264
4.5 Modes of Operation 266
4.5.1 Hold mode 266
4.5.2 Park state 268
4.5.3 Sniff mode 274
4.6 Logical Transports 277
4.6.1 SCO logical transport 277
4.6.2 eSCO logical transport 280
4.7 Test Mode 285
4.7.1 Activation and deactivation of test mode 285
4.7.2 Control of test mode 286
4.7.3 Summary of test mode PDUs 287

5 Summary 291
5.1 PDU Summary 291
5.2 Parameter Definitions 299
5.3 Default Values 307

6 List of Figures 309

7 List of Tables 313

Part D
ERROR CODES

Contents 317

1 Overview of Error Codes 319
1.1 Usage Descriptions 319
1.2 HCI Command Errors 319
1.3 List of Error Codes 320

2 Error Code Descriptions 323
2.1 Unknown HCI Command (0X01) 323
2.2 Unknown Connection Identifier (0X02) 323
2.3 Hardware Failure (0X03) 323

2.4 Page Timeout (0X04) 323
2.5 Authentication Failure (0X05) 323
2.6 PIN or key Missing (0X06) 323
2.7 Memory Capacity Exceeded (0X07) 323
2.8 Connection Timeout (0X08) 324
2.9 Connection Limit Exceeded (0X09) 324
2.10 Synchronous Connection Limit to a Device Exceeded (0X0A) 324
2.11 ACL Connection Already Exists (0X0B) 324
2.12 Command Disallowed (0X0C) 324
2.13 Connection Rejected due to Limited Resources (0X0D) 324
2.14 Connection Rejected due to Security Reasons (0X0E) 324
2.15 Connection Rejected due to Unacceptable BD_ADDR (0X0F) 325
2.16 Connection Accept Timeout Exceeded (0X10) 325
2.17 Unsupported Feature or Parameter Value (0X11) 325
2.18 Invalid HCI Command Parameters (0X12) 325
2.19 Remote User Terminated Connection (0X13) 325
2.20 Remote Device Terminated Connection due to Low Resources (0X14) 326
2.21 Remote Device Terminated Connection due to Power Off (0X15). 326
2.22 Connection Terminated by Local Host (0X16) 326
2.23 Repeated Attempts (0X17) 326
2.24 Pairing not Allowed (0X18) 326
2.25 Unknown LMP PDU (0X19) 326
2.26 Unsupported Remote Feature / Unsupported LMP Feature (0X1A) 326
2.27 SCO Offset Rejected (0X1B) 326
2.28 SCO Interval Rejected (0X1C) 327
2.29 SCO Air Mode Rejected (0X1D) 327
2.30 Invalid LMP Parameters (0X1E) 327
2.31 Unspecified Error (0X1F) 327
2.32 Unsupported LMP Parameter Value (0X20) 327
2.33 Role Change Not Allowed (0X21) 327
2.34 LMP Response Timeout (0X22) 327
2.35 LMP Error Transaction Collision (0X23) 328
2.36 LMP PDU Not Allowed (0X24) 328
2.37 Encryption Mode Not Acceptable (0X25) 328
2.38 Link Key Can Not be Changed (0X26) 328
2.39 Requested Qos Not Supported (0X27) 328
2.40 Instant Passed (0X28) 328
2.41 Pairing with Unit Key Not Supported (0X29) 328
2.42 Different Transaction Collision (0x2a) 328
2.43 QoS Unacceptable Parameter (0X2C) 328

2.44 QoS Rejected (0X2D) 329
2.45 Channel Classification Not Supported (0X2E) 329
2.46 Insufficient Security (0X2F) 329
2.47 Parameter out of Mandatory Range (0X30) 329
2.48 Role Switch Pending (0X32) 329
2.49 Reserved Slot Violation (0X34) 329
2.50 Role Switch Failed (0X35) 329

Part E
HOST CONTROLLER INTERFACE FUNCTIONAL SPECIFICATION

Contents 333

1 Introduction 339
1.1 Lower Layers of the Bluetooth Software Stack 339

2 Overview of Host Controller Transport Layer 341

3 Overview of Commands and Events 343
3.1 Generic Events 344
3.2 Device Setup 344
3.3 Controller Flow Control 345
3.4 Controller Information 345
3.5 Controller Configuration 346
3.6 Device Discovery 347
3.7 Connection Setup 349
3.8 Remote Information 351
3.9 Synchronous Connections 352
3.10 Connection State 353
3.11 Piconet Structure 354
3.12 Quality of Service 355
3.13 Physical Links 356
3.14 Host Flow Control 357
3.15 Link Information 358
3.16 Authentication and Encryption 359
3.17 Testing 361
3.18 Alphabetical List of Commands and Events 362

4 HCI Flow Control 367
4.1 Host to Controller Data Flow Control 367
4.2 Controller to Host Data Flow Control 368
4.3 Disconnection Behavior 369
4.4 Command Flow Control 369
4.5 Command Error Handling 370

5 HCI Data Formats 371
5.1 Introduction 371

5.2 Data and Parameter Formats....371
5.3 Connection Handles....372
5.4 Exchange of HCI-Specific Information....373
5.4.1 HCI Command Packet....373
5.4.2 HCI ACL Data Packets....375
5.4.3 HCI Synchronous Data Packets....377
5.4.4 HCI Event Packet....378
6 HCI Configuration Parameters....379
6.1 Scan Enable....379
6.2 Inquiry Scan Interval....379
6.3 Inquiry Scan Window....380
6.4 Inquiry Scan Type....380
6.5 Inquiry Mode....380
6.6 Page Timeout....381
6.7 Connection Accept Timeout....381
6.8 Page Scan Interval....382
6.9 Page Scan Window....382
6.10 Page Scan Period Mode (Deprecated)....382
6.11 Page Scan Type....383
6.12 Voice Setting....383
6.13 PIN Type....384
6.14 Link Key....384
6.15 Authentication Enable....384
6.16 Encryption Mode....385
6.17 Failed Contact Counter....386
6.18 Hold Mode Activity....386
6.19 Link Policy Settings....387
6.20 Flush Timeout....388
6.21 Num Broadcast Retransmissions....388
6.22 Link Supervision Timeout....389
6.23 Synchronous Flow Control Enable....389
6.24 Local Name....390
6.25 Class Of Device....390
6.26 Supported Commands....391
7 HCI Commands and Events....395
7.1 Link Control Commands....395
7.1.1 Inquiry Command....395
7.1.2 Inquiry Cancel Command....397
7.1.3 Periodic Inquiry Mode Command....398
7.1.4 Exit Periodic Inquiry Mode Command....401
7.1.5 Create Connection Command....402

7.1.6 Disconnect Command 405
7.1.7 Create Connection Cancel Command 406
7.1.8 Accept Connection Request Command 408
7.1.9 Reject Connection Request Command 410
7.1.10 Link Key Request Reply Command 411
7.1.11 Link Key Request Negative Reply Command 413
7.1.12 PIN Code Request Reply Command 414
7.1.13 PIN Code Request Negative Reply Command 416
7.1.14 Change Connection Packet Type Command 417
7.1.15 Authentication Requested Command 420
7.1.16 Set Connection Encryption Command 421
7.1.17 Change Connection Link Key Command 422
7.1.18 Master Link Key Command 423
7.1.19 Remote Name Request Command 424
7.1.20 Remote Name Request Cancel Command 426
7.1.21 Read Remote Supported Features Command 427
7.1.22 Read Remote Extended Features Command 428
7.1.23 Read Remote Version Information Command 429
7.1.24 Read Clock Offset Command 430
7.1.25 Read LMP Handle Command 431
7.1.26 Setup Synchronous Connection Command 433
7.1.27 Accept Synchronous Connection Request Command 438
7.1.28 Reject Synchronous Connection Request Command . 442
7.2 Link Policy Commands 443
7.2.1 Hold Mode Command 443
7.2.2 Sniff Mode Command 445
7.2.3 Exit Sniff Mode Command 448
7.2.4 Park State Command 449
7.2.5 Exit Park State Command 451
7.2.6 QoS Setup Command 452
7.2.7 Role Discovery Command 454
7.2.8 Switch Role Command 455
7.2.9 Read Link Policy Settings Command 456
7.2.10 Write Link Policy Settings Command 457
7.2.11 Read Default Link Policy Settings Command 459
7.2.12 Write Default Link Policy Settings Command 460
7.2.13 Flow Specification Command 461
7.3 Controller & Baseband Commands 463

7.3.1 Set Event Mask Command....463
7.3.2 Reset Command....465
7.3.3 Set Event Filter Command....466
7.3.4 Flush Command....471
7.3.5 Read PIN Type Command....473
7.3.6 Write PIN Type Command....474
7.3.7 Create New Unit Key Command....475
7.3.8 Read Stored Link Key Command....476
7.3.9 Write Stored Link Key Command....477
7.3.10 Delete Stored Link Key Command....479
7.3.11 Write Local Name Command....480
7.3.12 Read Local Name Command....481
7.3.13 Read Connection Accept Timeout Command....482
7.3.14 Write Connection Accept Timeout Command....483
7.3.15 Read Page Timeout Command....484
7.3.16 Write Page Timeout Command....485
7.3.17 Read Scan Enable Command....486
7.3.18 Write Scan Enable Command....487
7.3.19 Read Page Scan Activity Command....488
7.3.20 Write Page Scan Activity Command....490
7.3.21 Read Inquiry Scan Activity Command....491
7.3.22 Write Inquiry Scan Activity Command....492
7.3.23 Read Authentication Enable Command....493
7.3.24 Write Authentication Enable Command....494
7.3.25 Read Encryption Mode Command....495
7.3.26 Write Encryption Mode Command....496
7.3.27 Read Class of Device Command....497
7.3.28 Write Class of Device Command....498
7.3.29 Read Voice Setting Command....499
7.3.30 Write Voice Setting Command....500
7.3.31 Read Automatic Flush Timeout Command....501
7.3.32 Write Automatic Flush Timeout Command....502
7.3.33 Read Num Broadcast Retransmissions Command....503
7.3.34 Write Num Broadcast Retransmissions Command....504
7.3.35 Read Hold Mode Activity Command....505
7.3.36 Write Hold Mode Activity Command....506
7.3.37 Read Transmit Power Level Command....507
7.3.38 Read Synchronous Flow Control Enable Command....509

7.3.39 Write Synchronous Flow Control Enable Command...510
7.3.40 Set Controller To Host Flow Control Command 511
7.3.41 Host Buffer Size Command.....512
7.3.42 Host Number Of Completed Packets Command 514
7.3.43 Read Link Supervision Timeout Command.....516
7.3.44 Write Link Supervision Timeout Command 517
7.3.45 Read Number Of Supported IAC Command.....519
7.3.46 Read Current IAC LAP Command 520
7.3.47 Write Current IAC LAP Command 521
7.3.48 Read Page Scan Period Mode Command (Deprecated) 523
7.3.49 Write Page Scan Period Mode Command (Deprecated) 524
7.3.50 Set AFH Host Channel Classification Command 525
7.3.51 Read Inquiry Scan Type Command 526
7.3.52 Write Inquiry Scan Type Command 527
7.3.53 Read Inquiry Mode Command 528
7.3.54 Write Inquiry Mode Command 529
7.3.55 Read Page Scan Type Command 530
7.3.56 Write Page Scan Type Command 531
7.3.57 Read AFH Channel Assessment Mode Command 532
7.3.58 Write AFH Channel Assessment Mode Command 533
7.4 Informational Parameters 535
7.4.1 Read Local Version Information Command 535
7.4.2 Read Local Supported Commands Command 537
7.4.3 Read Local Supported Features Command.....538
7.4.4 Read Local Extended Features Command 539
7.4.5 Read Buffer Size Command 541
7.4.6 Read BD_ADDR Command.....543
7.5 Status Parameters 544
7.5.1 Read Failed Contact Counter Command.....544
7.5.2 Reset Failed Contact Counter Command 546
7.5.3 Read Link Quality Command 547
7.5.4 Read RSSI Command.....548
7.5.5 Read AFH Channel Map Command 550
7.5.6 Read Clock Command 552
7.6 Testing Commands 554
7.6.1 Read Loopback Mode Command 554
7.6.2 Write Loopback Mode Command.....555

7.6.3 Enable Device Under Test Mode Command 558
7.7 Events 559
7.7.1 Inquiry Complete Event 559
7.7.2 Inquiry Result Event 560
7.7.3 Connection Complete Event 562
7.7.4 Connection Request Event 563
7.7.5 Disconnection Complete Event 565
7.7.6 Authentication Complete Event 566
7.7.7 Remote Name Request Complete Event 567
7.7.8 Encryption Change Event 568
7.7.9 Change Connection Link Key Complete Event 569
7.7.10 Master Link Key Complete Event 570
7.7.11 Read Remote Supported Features Complete Event 571
7.7.12 Read Remote Version Information Complete Event 572
7.7.13 QoS Setup Complete Event 573
7.7.14 Command Complete Event 575
7.7.15 Command Status Event 576
7.7.16 Hardware Error Event 577
7.7.17 Flush Occurred Event 577
7.7.18 Role Change Event 578
7.7.19 Number Of Completed Packets Event 579
7.7.20 Mode Change Event 580
7.7.21 Return Link Keys Event 582
7.7.22 PIN Code Request Event 583
7.7.23 Link Key Request Event 584
7.7.24 Link Key Notification Event 585
7.7.25 Loopback Command Event 586
7.7.26 Data Buffer Overflow Event 586
7.7.27 Max Slots Change Event 587
7.7.28 Read Clock Offset Complete Event 588
7.7.29 Connection Packet Type Changed Event 589
7.7.30 QoS Violation Event 592
7.7.31 Page Scan Repetition Mode Change Event 593
7.7.32 Flow Specification Complete Event 594
7.7.33 Inquiry Result with RSSI Event 596
7.7.34 Read Remote Extended Features Complete Event 598
7.7.35 Synchronous Connection Complete Event 599
7.7.36 Synchronous Connection Changed event 601

8 List of Figures 603

9 List of Tables 605

10 Appendix 605

Appendix A: Deprecated Commands, Events and Configuration Parameters 607

Part F
MESSAGE SEQUENCE CHARTS

Contents 617

1 Introduction 619

1.1 Notation 619
1.2 Flow of Control 620
1.3 Example MSC 620

2 Services Without Connection Request 621

2.1 Remote Name Request 621
2.2 One-time Inquiry 622
2.3 Periodic Inquiry 624

3 ACL Connection Establishment and Detachment 627

3.1 Connection Setup 628

4 Optional Activities After ACL Connection Establishment 635

4.1 Authentication Requested 635
4.2 Set Connection Encryption 636
4.3 Change Connection Link Key 637
4.4 Master Link Key 638
4.5 Read Remote Supported Features 640
4.6 Read Remote Extended Features 640
4.7 Read Clock Offset 641
4.8 Read Remote Version Information 641
4.9 QOS Setup 642
4.10 Switch Role 642

5 Synchronous Connection Establishment and Detachment 645

5.1 Synchronous Connection Setup 645

6 Sniff, Hold and Park 651

6.1 sniff Mode 651
6.2 Hold Mode 652
6.3 Park State 654

7 Buffer Management, Flow Control 657

8 Loopback Mode 659

8.1 Local Loopback Mode 659
8.2 Remote Loopback Mode 661

9 List of Figures....663

Part G
SAMPLE DATA

Contents 667
1 Encryption Sample Data 669
1.1 Generating Kc' from Kc, ...669
1.2 First Set of Sample Data 672
1.3 Second Set of Sample Data 680
1.4 Third Set of Samples....688
1.5 Fourth Set of Samples 696
2 Frequency Hopping Sample Data 705
2.1 First set 706
2.2 Second set 712
2.3 Third set 718
3 Access Code Sample Data 725
4 HEC and Packet Header Sample Data 729
5 CRC Sample Data 731
6 Complete Sample Packets....733
6.1 Example of DH1 Packet....733
6.2 Example of DM1 Packet....734
7 Whitening Sequence Sample Data 735
8 FEC Sample Data....739
9 Encryption Key Sample Data 741
9.1 Four Tests of E1 741
9.2 Four Tests of E21 746
9.3 Three Tests of E22 748
9.4 Tests of E22 With Pin Augmenting....750
9.5 Four Tests of E3 760

Part H
SECURITY SPECIFICATION

Contents 767
1 Security Overview 769
2 Random Number Generation 771
3 Key Management....773
3.1 Key Types 773
3.2 Key Generation and Initialization 775
3.2.1 Generation of initialization key, ...776

3.2.2 Authentication 776
3.2.3 Generation of a unit key 776
3.2.4 Generation of a combination key 777
3.2.5 Generating the encryption key 778
3.2.6 Point-to-multipoint configuration 779
3.2.7 Modifying the link keys 780
3.2.8 Generating a master key 780

4 Encryption 783
4.1 Encryption Key Size Negotiation 784
4.2 Encryption of Broadcast Messages 784
4.3 Encryption Concept 785
4.4 Encryption Algorithm 786
4.4.1 The operation of the cipher 788
4.5 LFSR Initialization 789
4.6 Key Stream Sequence 792

5 Authentication 793
5.1 Repeated Attempts 795

6 The Authentication And Key-Generating Functions 797
6.1 The Authentication Function E1 797
6.2 The Functions Ar and A'r 799
6.2.1 The round computations 799
6.2.2 The substitution boxes "e" and "l" 799
6.2.3 Key scheduling 800
6.3 E2-Key Generation Function for Authentication 801
6.4 E3-Key Generation Function for Encryption 803

7 List of Figures 805

8 List of Tables 807

Specification Volume 3
Core System Package
[Host volume]

Table of Contents 5

Part A
LOGICAL LINK CONTROL AND ADAPTATION PROTOCOL SPECIFICATION

Contents 15

1 Introduction 19
1.1 L2CAP Features 19
1.2 Assumptions 23
1.3 Scope 23
1.4 Terminology 24

2 General Operation 27
2.1 Channel Identifiers 27
2.2 Operation Between Devices 27
2.3 Operation Between Layers 29
2.4 Modes of Operation 29

3 Data Packet Format 31
3.1 Connection-oriented Channel in Basic L2CAP Mode 31
3.2 Connectionless Data Channel in Basic L2CAP Mode 32
3.3 Connection-oriented Channel in Retransmission/Flow Control Modes 33
3.3.1 L2CAP header fields 33
3.3.2 Control field (2 octets) 34
3.3.3 L2CAP SDU length field (2 octets) 36
3.3.4 Information payload field (0 to 65531 octets) 36
3.3.5 Frame check sequence (2 octets) 37
3.3.6 Invalid frame detection 38

4 Signalling Packet Formats 39
4.1 Command Reject (code 0x01) 41
4.2 Connection Request (code 0x02) 42
4.3 Connection Response (code 0x03) 44
4.4 Configuration Request (code 0x04) 45
4.5 Configuration Response (code 0x05) 48
4.6 Disconnection Request (code 0x06) 50
4.7 Disconnection Response (code 0x07) 51
4.8 Echo Request (code 0x08) 51

4.9 Echo Response (code 0x09) 52
4.10 Information Request (code 0x0A) 52
4.11 Information Response (code 0x0B) 53
4.12 Extended Feature Mask 54

5 Configuration Parameter Options 55
5.1 Maximum Transmission Unit (MTU) 55
5.2 Flush Timeout Option 57
5.3 Quality of Service (QoS) Option 58
5.4 Retransmission and Flow Control Option 62

6 State Machine 65
6.1 General rules for the state machine: 65
6.1.1 CLOSED state 66
6.1.2 WAIT_CONNECT_RSP state 67
6.1.3 WAIT_CONNECT state 67
6.1.4 CONFIG state 68
6.1.5 OPEN state 71
6.1.6 WAIT_DISCONNECT state 71
6.2 Timers events 73
6.2.1 RTX 73
6.2.2 ERTX 74

7 General Procedures 77
7.1 Configuration Process 77
7.1.1 Request path 78
7.1.2 Response path 78
7.2 Fragmentation and Recombination 79
7.2.1 Fragmentation of L2CAP PDUs 79
7.2.2 Recombination of L2CAP PDUs 80
7.3 Encapsulation of SDUs 81
7.3.1 Segmentation of L2CAP SDUs 81
7.3.2 Reassembly of L2CAP SDUs 82
7.3.3 Segmentation and fragmentation 82
7.4 Delivery of Erroneous L2CAP SDUs 83
7.5 Operation with Flushing 83
7.6 Connectionless Data Channel 84

8 Procedures for Flow Control and Retransmission 85
8.1 Information Retrieval 85
8.2 Function of PDU Types for Flow Control and Retransmission 85
8.2.1 Information frame (I-frame) 85
8.2.2 Supervisory Frame (S-frame) 85

8.3 Variables and Sequence Numbers....87
8.3.1 Sending peer....87
8.3.2 Receiving peer....89
8.4 Retransmission Mode....91
8.4.1 Transmitting frames....91
8.4.2 Receiving I-frames....93
8.4.3 I-frames pulled by the SDU reassembly function....93
8.4.4 Sending and receiving acknowledgements....94
8.4.5 Receiving REJ frames....95
8.4.6 Waiting acknowledgements....95
8.4.7 Exception conditions....95
8.5 Flow Control Mode....97
8.5.1 Transmitting I-frames....97
8.5.2 Receiving I-frames....98
8.5.3 I-frames pulled by the SDU reassembly function....98
8.5.4 Sending and receiving acknowledgements....98
8.5.5 Waiting acknowledgements....99
8.5.6 Exception conditions....99
9 List of Figures....101
10 List of Tables....103
11 Appendix....103
Appendix A: Configuration MSCs....105

Part B
SERVICE DISCOVERY PROTOCOL (SDP)

Contents....111
1 Introduction....113
1.1 General Description....113
1.2 Motivation....113
1.3 Requirements....113
1.4 Non-requirements and Deferred Requirements....114
1.5 Conventions....114
1.5.1 Bit And Byte Ordering Conventions....114
2 Overview....115
2.1 SDP Client-Server Interaction....115
2.2 Service Record....116
2.3 Service Attribute....118
2.4 Attribute ID....118

2.5 Attribute Value 119
2.6 Service Class 119
2.6.1 A Printer Service Class Example 120
2.7 Searching for Services 121
2.7.1 UUID 121
2.7.2 Service Search Patterns 122
2.8 Browsing for Services 122
2.8.1 Example Service Browsing Hierarchy 123

3 Data Representation 125
3.1 Data Element 125
3.2 Data Element Type Descriptor 125
3.3 Data Element Size Descriptor 126
3.4 Data Element Examples 127

4 Protocol Description 129
4.1 Transfer Byte Order 129
4.2 Protocol Data Unit Format 129
4.3 Partial Responses and Continuation State 131
4.4 Error Handling 131
4.4.1 SDP_ErrorResponse PDU 132
4.5 ServiceSearch Transaction 133
4.5.1 SDP_ServiceSearchRequest PDU 133
4.5.2 SDP_ServiceSearchResponse PDU 134
4.6 ServiceAttribute Transaction 136
4.6.1 SDP_ServiceAttributeRequest PDU 136
4.6.2 SDP_ServiceAttributeResponse PDU 138
4.7 ServiceSearchAttribute Transaction 139
4.7.1 SDP_ServiceSearchAttributeRequest PDU 139
4.7.2 SDP_ServiceSearchAttributeResponse PDU 141

5 Service Attribute Definitions 143
5.1 Universal Attribute Definitions 143
5.1.1 ServiceRecordHandle Attribute 143
5.1.2 ServiceClassIDList Attribute 144
5.1.3 ServiceRecordState Attribute 144
5.1.4 ServiceID Attribute 144
5.1.5 ProtocolDescriptorList Attribute 145
5.1.6 BrowseGroupList Attribute 146
5.1.7 LanguageBaseAttributeIDList Attribute 146
5.1.8 ServiceInfoTimeToLive Attribute 147
5.1.9 ServiceAvailability Attribute 148

5.1.10 BluetoothProfileDescriptorList Attribute....148
5.1.11 DocumentationURL Attribute....149
5.1.12 ClientExecutableURL Attribute....149
5.1.13 IconURL Attribute....150
5.1.14 ServiceName Attribute....150
5.1.15 ServiceDescription Attribute....151
5.1.16 ProviderName Attribute....151
5.1.17 Reserved Universal Attribute IDs....151
5.2 ServiceDiscoveryServer Service Class Attribute Definitions....152
5.2.1 ServiceRecordHandle Attribute....152
5.2.2 ServiceClassIDList Attribute....152
5.2.3 VersionNumberList Attribute....152
5.2.4 ServiceDatabaseState Attribute....153
5.2.5 Reserved Attribute IDs....153
5.3 BrowseGroupDescriptor Service Class Attribute Definitions....153
5.3.1 ServiceClassIDList Attribute....153
5.3.2 GroupID Attribute....154
5.3.3 Reserved Attribute IDs....154
6 Appendix....154
Appendix A – Background Information....155
Appendix B – Example SDP Transactions....156

Part C
GENERIC ACCESS PROFILE

Contents....171
Foreword....174
1 Introduction....175
1.1 Scope....175
1.2 Symbols and conventions....175
1.2.1 Requirement status symbols....175
1.2.2 Signaling diagram conventions....176
1.2.3 Notation for timers and counters....176
2 Profile overview....177
2.1 Profile stack....177
2.2 Configurations and roles....177
2.3 User requirements and scenarios....178
2.4 Profile fundamentals....179
2.5 Conformance....179
3 User interface aspects....181

3.1 The user interface level 181
3.2 Representation of Bluetooth parameters 181
3.2.1 Bluetooth device address (BD_ADDR) 181
3.2.2 Bluetooth device name (the user-friendly name) 181
3.2.3 Bluetooth passkey (Bluetooth PIN) 182
3.2.4 Class of Device 183
3.3 Pairing 184
4 Modes 185
4.1 Discoverability modes 185
4.1.1 Non-discoverable mode 186
4.1.2 Limited discoverable mode 186
4.1.3 General discoverable mode 187
4.2 Connectability modes 189
4.2.1 Non-connectable mode 189
4.2.2 Connectable mode 189
4.3 Pairing modes 191
4.3.1 Non-pairable mode 191
4.3.2 Pairable mode 191
5 Security aspects 193
5.1 Authentication 193
5.1.1 Purpose 193
5.1.2 Term on UI level 193
5.1.3 Procedure 194
5.1.4 Conditions 194
5.2 Security modes 194
5.2.1 Security mode 1 (non-secure) 196
5.2.2 Security mode 2 (service level enforced security) 196
5.2.3 Security modes 3 (link level enforced security) 196
6 Idle mode procedures 197
6.1 General inquiry 197
6.1.1 Purpose 197
6.1.2 Term on UI level 197
6.1.3 Description 198
6.1.4 Conditions 198
6.2 Limited inquiry 198
6.2.1 Purpose 198
6.2.2 Term on UI level 199
6.2.3 Description 199

6.2.4 Conditions 199
6.3 Name discovery 200
6.3.1 Purpose 200
6.3.2 Term on UI level 200
6.3.3 Description 200
6.3.4 Conditions 201
6.4 Device discovery 201
6.4.1 Purpose 201
6.4.2 Term on UI level 201
6.4.3 Description 202
6.4.4 Conditions 202
6.5 Bonding 202
6.5.1 Purpose 202
6.5.2 Term on UI level 202
6.5.3 Description 203
6.5.4 Conditions 204
7 Establishment procedures 205
7.1 Link establishment 205
7.1.1 Purpose 205
7.1.2 Term on UI level 205
7.1.3 Description 206
7.1.4 Conditions 207
7.2 Channel establishment 208
7.2.1 Purpose 208
7.2.2 Term on UI level 208
7.2.3 Description 208
7.2.4 Conditions 209
7.3 Connection establishment 210
7.3.1 Purpose 210
7.3.2 Term on UI level 210
7.3.3 Description 210
7.3.4 Conditions 211
7.4 Establishment of additional connection 211
8 Definitions 213
8.1 General definitions 213
8.2 Connection-related definitions 213
8.3 Device-related definitions 214
8.4 Procedure-related definitions 215
8.5 Security-related definitions 215

9 Appendix A (Normative): Timers and constants 217

10 Appendix B (Informative): Information flows of related procedures 219

10.1 Imp-authentication 219
10.2 Imp-pairing 220
10.3 Service discovery 221

11 References 223

Part D
TEST SUPPORT

Contents 226

1 Test Methodology 227

1.1 Test Scenarios 227
1.1.1 Test setup 227
1.1.2 Transmitter Test 228
1.1.3 LoopBack test 233
1.1.4 Pause test 236
1.2 References 236

2 Test Control Interface (TCI) 237

2.1 Introduction 237
2.1.1 Terms used 237
2.1.2 Usage of the interface 237
2.2 TCI Configurations 238
2.2.1 Bluetooth RF requirements 238
2.2.2 Bluetooth protocol requirements 239
2.2.3 Bluetooth profile requirements 240
2.3 TCI Configuration and Usage 241
2.3.1 Transport layers 241
2.3.2 Baseband and link manager qualification 242
2.3.3 HCI qualification 244



BLUETOOTH COMPLIANCE REQUIREMENTS

This document specifies the requirements for Bluetooth® compliance.



CONTENTS

1 Introduction 43

2 Scope 45

3 Definitions 47

3.1 Types of Bluetooth Products 47

4 Core Configurations 49

4.1 Specification Naming Conventions 49

4.2 EDR Configurations 49



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 specification defines some fundamental concepts 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.

3.1 TYPES OF BLUETOOTH PRODUCTS

Bluetooth Product – Any product containing an implementation of Bluetooth wireless technology. All Bluetooth Products shall be one of the following:

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

Bluetooth End Product - An implementation of Bluetooth wireless technology, which 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 - An implementation of Bluetooth wireless technology, which 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 a complementary Bluetooth End Product or one or more complementary Bluetooth Subsystem Products such that the resulting combination satisfies the requirements of a Bluetooth End Product. 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 - An implementation of Bluetooth wireless technology, which does not meet the requirements of a Bluetooth End Product, but 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. Bluetooth Component Products can be qualified solely for distribution and the use of the Bluetooth wireless technology in Bluetooth Component Products require such Bluetooth Component Products to be incorporated in Bluetooth End Products or Bluetooth Subsystem Products.

Bluetooth Development Tool - An implementation of Bluetooth wireless technology, 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 in development of new Bluetooth Products.

Bluetooth Test Equipment - An implementation of Bluetooth wireless technology, 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.

4 CORE CONFIGURATIONS

This section defines the set of features that are required for a product to be qualified to a specification name. Each core configuration is defined by a set of LMP feature bits or L2CAP feature bits that shall be supported to allow the configuration name to be used.

The configuration requirements imposed on a device depends on the profiles that it supports.

4.1 SPECIFICATION NAMING CONVENTIONS

Each specification is named by its core specification version number, followed by a list of the core configuration names that are implemented and qualified.

A complete specification name shall be stated as the core specification version number followed by "+", and then either a single core configuration name or a sequence of core configuration names separated by "+".

Examples of complete specification names including the core configuration names:

- Bluetooth v2.0
- Bluetooth v2.0 + EDR

In this example, a product claiming "Bluetooth v2.0" may implement some of the EDR features, following the requirements in other parts of the specifications, and be qualified for those features. If the full set required in Section 4.2 are not supported the "+EDR" configuration name shall not be used in product literature.

4.2 EDR CONFIGURATIONS

This section specifies additional compliance requirements that shall be followed if the configuration name "EDR" is used within the complete specification name. The configu-ration name "EDR" may only be used with the current core specification version number 2.0 or later versions of the specification.

Table 4.1 defines three categories of Transport Requirements that shall be satisfied subject to the following rules:

- A Bluetooth product shall support category 1 whenever it supports asynchronous transports for the profiles it incorporates.
- A Bluetooth product shall support category 2 whenever it supports asynchronous transports with multislot ACL packets for the profiles it incorporates.
- A Bluetooth product shall support category 3 whenever it supports eSCO syn-chronous transports for the profiles it incorporates.

.

No.	Transport Requirements	LMP Feature Bits Required	L2CAP Feature Bits Required
1	EDR for asynchronous transports (single slot)	Enhanced Data Rate ACL 2 Mbps mode (25) Enhanced Data Rate ACL 3 Mbps mode (26)	None
2	EDR for asynchronous transports (multi-slot)	3-slot Enhanced Data Rate ACL packets (39) 5-slot Enhanced Data Rate ACL packets (40)	None
3	EDR for synchronous transports	Enhanced Data Rate eSCO 2 Mbps mode (45) Enhanced Data Rate eSCO 3 Mbps mode (46)	None

Table 4.1: EDR configuration requirements

Note: No additional requirements are stated on the support of 2-EV5 and 3-EV5 packets.



APPENDIX



CONTENTS

1 Revision History 55
1.1 [vol 0] Master TOC & Compliance Requirements 55
1.1.1 Bluetooth Compliance Requirements 55
1.2 [Vol 1] Architecture & Terminology Overview 55
1.3 [Vol 2 & 3] Core System Package 56
2 Contributors 59
2.1 [vol 0] Master TOC & Compliance Requirements 59
2.1.1 Part B: Bluetooth Compliance Requirements 59
2.2 [vol 1] Architecture &Terminology Overview 59
2.2.1 Part A: Architectural Overview 59
2.2.2 Part B: Acronyms & Abbreviations 59
2.2.3 Part C: Changes from Bluetooth Specification v1.1 60
2.3 [Vol 2] Core System Package, Controller 61
2.3.1 Part A: Radio Specification 61
2.3.2 Part B: Baseband Specification 62
2.3.3 Part C: Link Manager Protocol 64
2.3.4 Part D: Error Codes 66
2.3.5 Part E: Bluetooth Host Controller Interface Functional Specification66
2.3.6 Part F: Message Sequence Charts 67
2.3.7 Part G: Sample Data 68
2.3.8 Part H: Security Specification 68
2.4 [Vol 3] Core System Package, Host 69
2.4.1 Part A: Logical Link Control and Adaptation Protocol Specification69
2.4.2 Part B: Service Discovery Protocol (SDP) 70
2.4.3 Part C Generic Access Profile 71
2.4.4 Part D: Test Support 71



1 REVISION HISTORY

Public versions are marked with bold in the tables below.

Beginning with the v1.2 of the Core System Package the core Bluetooth specification documents, protocols and profiles, are transferred to a new partitioning comprising 3 volumes and individual profile specifications are each contained in an individual document instead of the two volumes (Core and Profiles) previously used.

For more detailed information about changes between version published before v1.2, please see the Appendix I "Revision History" in v1.1.

1.1 [VOL 0] MASTER TOC & COMPLIANCE REQUIREMENTS

1.1.1 Bluetooth Compliance Requirements

Rev	Date	Comments
v2.0 + EDR	Oct 15 2004	This version of the specification is intended to be a separate Bluetooth Specification that has all the functional characteristics of the v1.2 Bluetooth Specification that adds the Enhanced Data Rate (EDR) feature which required changes to Volume 0, Part A, Master Table of Contents.
v1.2	Nov 05 2003	This Part was moved from the Core volume. No content changes been made to this document since v1.1

1.2 [VOL 1] ARCHITECTURE & TERMINOLOGY OVERVIEW

Rev	Date	Comments
v2.0 + EDR	Oct 15 2004	This version of the specification is intended to be a separate Bluetooth Specification that has all the functional characteristics of the v1.2 Bluetooth Specification that adds the Enhanced Data Rate (EDR) feature which incorporates changes to Volume 1, Part B, Acronyms and Abbreviations.
v1.2	Nov 05 2003	New volume with informational content. This volume will always be updated in parallel with the Core System volumes

1.3 [VOL 2 & 3] CORE SYSTEM PACKAGE

Rev	Date	Comments
v2.0 + EDR	Aug 01 2004	This version of the specification is intended to be a separate Bluetooth Specification. This specification was created by adding EDR and the errata for v1.2 ESR 1.
v1.2	Nov 05 2003	New features added in v1.2: - Architectural overview - Faster connection - Adaptive frequency hopping - Extended SCO links - Enhanced error detection and flow control - Enhanced synchronization capability - Enhanced flow specification The Core System Package now comprises two volumes and the text has gone through a radical change both in terms of structure and nomenclature. The language is also more precise and is adapted to meet the IEEE standard. The following parts are moved from the Core System Package to other volumes or has been deprecated: RFCOMM [vol 7], Object Exchange (IrDA Interoperability) [vol 8], TCS [vol 9], Interoperability Requirements for Bluetooth as a WAP Bearer [vol 6], HCI USB Transport Layer [vol4], HCI RS232 Transport Layer [vol 4], HCI UART Transport Layer [vol 4], Bluetooth Compliance Requirements [vol 0], Optional Paging Schemes [deprecated]
1.1	Feb 22nd 2001	The specification was updated with Errata items previously published on the web site. The Bluetooth Assigned Numbers appendix was lifted out from the specification to allow continuos maintenance on the web site.
1.0B	Dec. 1st 1999	The specification was updated with Errata items previously published on the web site and was revised from a linguistic point of view. The following parts was added: Interoperability Requirements for Bluetooth as a WAP Bearer, Test Control Interface, Sample Data (appendix), Bluetooth Audio (appendix), Baseband Timers (appendix) and Optional Paging Scheme (appendix)
1.0a	July 26th 1999	The first version of the Bluetooth Specification published on the public web site. Added part: Bluetooth Compliance Requirements.
1.0 draft	July 5th 1999	The following parts was added: Service Discovery Protocol (SDP), Telephony Control Specification (TCS), Bluetooth Assigned Numbers (appendix) and Message Sequence Charts (appendix)
0.9	April 30th 1999	The following parts was added: IrDA Interoperability, HCI RS232 Transport Layer, HCI UART Transport Layer and Test Mode
0.8	Jan 21st 1999	The following parts was added: Radio Specification, L2CAP, RFCOMM, HCI & HCI USB Transport Layer

Rev	Date	Comments
0.7	Oct 19th 1998	This first version only included Baseband and Link Manager Protocol



2 CONTRIBUTORS

2.1 [VOL 0] MASTER TOC & COMPLIANCE REQUIREMENTS

2.1.1 Part B: Bluetooth Compliance Requirements

BQRB (Editor)	
Wayne Park	3Com Corporation
Lawrence Jones	ComBit, Inc.
Gary Robinson	IBM Corporation
Georges Seuron	IBM Corporation
Rick Jessop	Intel Corporation
John Webb	Intel Corporation
Bruce Littlefield	Lucent Technologies, Inc.
Brian A. Redding	Motorola, Inc.
Waldemar Hontscha	Nokia Corporation
Petri Morko	Nokia Corporation
Magnus Hansson	Telefonaktiebolaget LM Ericsson
Magnus Sommansson	Telefonaktiebolaget LM Ericsson
Göran Svennarp	Telefonaktiebolaget LM Ericsson
Warren Allen	Toshiba Corporation
John Shi	Toshiba Corporation

2.2 [VOL 1] ARCHITECTURE &TERMINOLOGY OVERVIEW

2.2.1 Part A: Architectural Overview

Jennifer Bray	CSR
Robin Heydon	CSR
Henrik Hedlund	Ericsson
Martin van der Zee	Ericsson
Andy Glass	Microsoft
Brian Redding	Motorola
Jürgen Schnitzler	Nokia
Thomas Müller	Nokia
Joel Linsky	Silicon Wave
Terry Bourk	Silicon Wave
Michael Hasling	Tality
Yoshimitsu Shimojo	Toshiba
Toshiki Kizu	Toshiba
John Mersh	TTPCom

2.2.2 Part B: Acronyms & Abbreviations

Dan Sönnerstam	Pyramid Communication AB
Steve Koester	Bluetooth SIG, Inc.

2.2.3 Part C: Changes from Bluetooth Specification v1.1

Tom Siep	Bluetooth SIG Inc.
Robin Heydon	CSR
Henrik Hedlund	Ericsson
Rob Davies	Philips
Joel Linsky	Silicon Wave
John Mersh	TTPCom

2.3 [VOL 2] CORE SYSTEM PACKAGE, CONTROLLER

2.3.1 Part A: Radio Specification

Version 2.0 + EDR

Steven Hall	RF Micro Devices
Robert Young	CSR
Robert Kokke	Ericsson
Harald Kafemann	Nokia
Jukka Reunamäki	Nokia
Morton Gade	Digianswer
Mike Fitton	Toshiba
Oren Eliezer	Texas Instruments
Stephane Laurent-Michel	Tality

Version 1.2

Tom Siep	Bluetooth SIG Inc.
Jennifer Bray	CSR
Robin Heydon	CSR
Morten Gade	Digianswer/Motorola
Henrik Hedlund	Ericsson
Stefan Agnani	Ericsson
Robert Kokke	Ericsson
Roland Hellfajer	Infineon
Thomas Müller	Nokia
Antonio Salloum	Philips
Joel Linsky	Silicon Wave
Steven Hall	Silicon Wave
Oren Eliezer	Texas Instruments
Mike Fitton	Toshiba

Previous versions

Steve Williams	3Com Corporation
Todor V. Cooklov	Aware
Poul Hove Kristensen	Digianswer A/S
Kurt B. Fischer	Hyper Corporation
Kevin D. Marquess	Hyper Corporation
Troy Beukema	IBM Corporation
Brian Gaucher	IBM Corporation
Jeff Schiffer	Intel Corporation
James P. Gilb	Mobilian
Rich L. Ditch	Motorola, Inc.
Paul Burgess	Nokia Corporation
Olaf Joeressen	Nokia Corporation
Thomas Müller	Nokia Corporation

Arto T. Palin	Nokia Corporation
Steven J. Shellhammer	Symbol
Sven Mattisson	Telefonaktiebolaget LM Ericsson
Lars Nord (Section Owner)	Telefonaktiebolaget LM Ericsson
Anders Svensson	Telefonaktiebolaget LM Ericsson
Mary A. DuVal	Texas Instruments
Allen Hotari	Toshiba Corporation

2.3.2 Part B: Baseband Specification

Version 2.0 + EDR

Steven Hall	RF Micro Devices
Robert Young	CSR
Robert Kokke	Ericsson
Harald Kafemann	Nokia
Joel Linsky	RF Micro Devices
Terry Bourk	RF Micro Devices
Arto Palin	Nokia

Version 1.2

P G Madhavan	Agere
Hongbing Gan	Bandspeed, Inc.
Tod Sizer	Bell Labs
Alexander Thoukydides	CSR
Jennifer Bray	CSR
Robin Heydon	CSR
Kim Schneider	Digianswer/Motorola
Knud Dyring-Olsen	Digianswer/Motorola
Niels Nielsen	Digianswer/Motorola
Henrik Andersen	Digianswer/Motorola
Christian Gehrmann	Ericsson
Henrik Hedlund	Ericsson
Jan Åberg	Ericsson
Martin van der Zee	Ericsson
Rakesh Taori	Ericsson
Jaap Haartsen	Ericsson
Stefan Zürbes	Ericsson
Roland Hellfajer	Infineon
YC Maa	Integrated Programmable Communications, Inc.
HungKun Chen	Integrated Programmable Communications, Inc.
Steve McGowan	Intel
Adrian Stephens	Mobilian Corporation
Jim Lansford	Mobilian Corporation
Eric Meihofer	Motorola
Arto Palin	Nokia

Carmen Kühl	Nokia
Hannu Laine	Nokia
Jürgen Schnitzler	Nokia
Päivi Ruuska	Nokia
Thomas Müller	Nokia
Antonio Salloum	Philips
Harmke de Groot	Philips
Marianne van de Casteelee	Philips
Rob Davies	Philips
Roland Matthijssen	Philips
Joel Linsky (section owner)	Silicon Wave
Terry Bourk	Silicon Wave
Gary Schneider	Symbol Technologies, Inc.
Stephen J. Shellhammer	Symbol Technologies, Inc.
Michael Hasling	Tality
Amihai Kidron	Texas Instruments
Dan Michael	Texas Instruments
Eli Dekel	Texas Instruments
Jie Liang	Texas Instruments
Oren Eliezer	Texas Instruments
Tally Shina	Texas Instruments
Yariv Raveh	Texas Instruments
Anuj Batra	Texas Instruments
Katsuhiro Kinoshita	Toshiba
Toshiki Kizu	Toshiba
Yoshimitsu Shimojo	Toshiba
Charles Sturman	TTPCom
John Mersh	TTPCom
Sam Turner	TTPCom
Christoph Scholtz	University of Bonn
Simon Baatz	University of Bonn

Previous versions

Kevin D. Marquess	Hyper Corporation
Chatschik Bisdikian	IBM Corporation
Kris Fleming	Intel Corporation
James P. Gilb	Mobilian
David E. Cypher	NIST
Nada Golmie	NIST
Olaf Joeressen	Nokia Corporation
Thomas Müller	Nokia Corporation
Charlie Mellone	Motorola, Inc.
Harmke de Groot	Philips
Terry Bourk	Silicon Wave
Steven J. Shellhammer	Symbol

Jaap Haartsen	Telefonaktiebolaget LM Ericsson
Henrik Hedlund (Section Owner)	Telefonaktiebolaget LM Ericsson
Tobias Melin	Telefonaktiebolaget LM Ericsson
Joakim Persson	Telefonaktiebolaget LM Ericsson
Mary A. DuVal	Texas Instruments
Onn Haran	Texas Instruments
Thomas M. Siep	Texas Instruments
Ayse Findikli	Zeevo, Inc.

Previous versions [Encryption Sample Data, appendix]

Thomas Müller	Nokia Corporation
Thomas Sander	Nokia Corporation
Joakim Persson (Section Owner)	Telefonaktiebolaget LM Ericsson

Previous versions [Bluetooth Audio, appendix]

Magnus Hansson	Telefonaktiebolaget LM Ericsson
Fisseha Mekuria	Telefonaktiebolaget LM Ericsson
Mats Omrin	Telefonaktiebolaget LM Ericsson
Joakim Persson (Section Owner)	Telefonaktiebolaget LM Ericsson

Previous versions [Baseband Timers, appendix]

David E. Cyper	NIST
Jaap Haartsen (Section Owner)	Telefonaktiebolaget LM Ericsson
Joakim Persson	Telefonaktiebolaget LM Ericsson
Ayse Findikli	Zeevo, Inc.

2.3.3 Part C: Link Manager Protocol

Version 2.0 + EDR

John Mersh	TTPCom Ltd.
Joel Linsky	RF Micro Devices
Harald Kafemann	Nokia
Simon Morris	CSR

Version 1.2

Jennifer Bray	CSR
Robin Heydon	CSR
Simon Morris	CSR
Alexander Thoukydides	CSR
Kim Schneider	Digianswer/Motorola
Knud Dyring-Olsen	Digianswer/Motorola
Henrik Andersen	Digianswer/Motorola
Jan Åberg	Ericsson
Martin van der Zee	Ericsson

Roland Hellfajer	Infineon
YC Maa	Integrated Programmable Communications, Inc.
Steve McGowan	Intel
Tod Sizer	Lucent Technologies
Adrian Stephens	Mobilian
Jürgen Schnitzler	Nokia
Thomas Müller	Nokia
Carmen Kuhl	Nokia
Arto Palin	Nokia
Thomas Müller	Nokia
Roland Matthijssen	Philips
Rob Davies	Philips
Harmke de Groot	Philips
Antonio Salloum	Philips
Joel Linsky	Silicon Wave
Terry Bourk	Silicon Wave
Yariv Raveh	Texas Instruments
Tally Shina	Texas Instruments
Amihai Kidron	Texas Instruments
Yoshimitsu Shimojo	Toshiba
Toshiki Kizu	Toshiba
John Mersh (section owner)	TTPCom
Sam Turner	TTPCom

Previous versions

Kim Schneider	Digianswer A/S
Toru Aihara	IBM Corporation
Chatschik Bisdikian	IBM Corporation
Kris Fleming	Intel Corporation
David E. Cypher	NIST
Thomas Busse	Nokia Corporation
Julien Corthial	Nokia Corporation
Olaf Joeressen	Nokia Corporation
Thomas Müller	Nokia Corporation
Dong Nguyen	Nokia Corporation
Harmke de Groot	Philips
Terry Bourk	Silicon Wave
Johannes Elg	Telefonaktiebolaget LM Ericsson
Jaap Haartsen	Telefonaktiebolaget LM Ericsson
Tobias Melin (Section Owner)	Telefonaktiebolaget LM Ericsson
Mary A. DuVal	Texas Instruments
Onn Haran	Texas Instruments
John Mersh	TTPCom

2.3.4 Part D: Error Codes

Version 1.2

Robin Heydon (section owner)	CSR
Roland Hellfajer	Infineon
Joel Linsky	Silicon Wave
John Mersh	TTPCom

This part was earlier included in the LMP and HCI functional Specifications.

2.3.5 Part E: Bluetooth Host Controller Interface Functional Specification

Version 2.0 + EDR

Neil Stewart	Tality UK, Ltd.
Joel Linsky	RF Micro Devices
Robin Heydon	CSR

Version 1.2

Robin Heydon (section owner	CSR
Jennifer Bray	CSR
Alexander Thoukydides	CSR
Knud Dyring-Olsen	Digianswer/Motorola
Henrik Andersen	Digianswer/Motorola
Jan Åberg	Ericsson
Martin van der Zee	Ericsson
Don Liechty	Extended Systems
Kevin Marquess	Hyper Corp
Roland Hellfajer	Infineon
YC Maa	Integrated Programmable Communications, Inc.
Steve McGowan	Intel
Tod Sizer	Lucent Technologies
Tsuyoshi Okada	Matsushita Electric Industrial Co. Ltd
Andy Glass	Microsoft
Adrian Stephens	Mobilian
Jürgen Schnitzler	Nokia
Thomas Müller	Nokia
Rene Tischer	Nokia
Rob Davies	Philips
Antonio Salloum	Philips
Joel Linsky	Silicon Wave
Terry Bourk	Silicon Wave
Len Ott	Socket Communications
Randy Erman	Taiyo Yuden
Yoshimitsu Shimojo	Toshiba
Toshiki Kizu	Toshiba

Katsuhiro Kinoshita	Toshiba
Sam Turner	TTPCom
John Mersh	TTPCom

Previous versions

Todor Cooklev	3Com Corporation
Toru Aihara	IBM Corporation
Chatschik Bisdikian	IBM Corporation
Nathan Lee	IBM Corporation
Akihiko Mizutani	IBM Corporation
Les Cline	Intel Corporation
Bailey Cross	Intel Corporation
Kris Fleming	Intel Corporation
Robert Hunter	Intel Corporation
Jon Inouye	Intel Corporation
Srikanth Kambhatla	Intel Corporation
Steve Lo	Intel Corporation
Vijay Suthar	Intel Corporation
Bruce P. Kraemer	Intersil
Greg Muchnik	Motorola, Inc.
David E. Cypher	NIST
Thomas Busse	Nokia Corporation
Julien Courthial	Nokia Corporation
Thomas Müller	Nokia Corporation
Dong Nguyen	Nokia Corporation
Jürgen Schnitzler	Nokia Corporation
Fujio Watanabe	Nokia Corporation
Christian Zechlin	Nokia Corporation
Johannes Elg	Telefonaktiebolaget LM Ericsson
Christian Johansson (Section Owner)	Telefonaktiebolaget LM Ericsson
Patrik Lundin	Telefonaktiebolaget LM Ericsson
Tobias Melin	Telefonaktiebolaget LM Ericsson
Mary A. DuVal	Texas Instruments
Thomas M. Siep	Texas Instruments
Masahiro Tada	Toshiba Corporation
John Mersh	TTPCom

2.3.6 Part F: Message Sequence Charts

Version 1.2

Tom Siep	Bluetooth SIG Inc.
Robin Heydon (section owner)	CSR
Simon Morris	CSR
Jan Åberg	Ericsson
Christian Gehrmann	Ericsson
Joel Linsky	Silicon Wave
John Mersh	TTPCom

Previous versions

Todor Cooklev	3Com Corporation
Toru Aihara	IBM Corporation
Chatschik Bisdikian	IBM Corporation
Nathan Lee	IBM Corporation
Kris Fleming	Intel Corporation
Greg Muchnik	Motorola, Inc.
David E. Cypher	NIST
Thomas Busse	Nokia Corporation
Dong Nguyen (Section Owner)	Nokia Corporation
Fujio Watanabe	Nokia Corporation
Christian Johansson	Telefonaktiebolaget LM Ericsson
Tobias Melin	Telefonaktiebolaget LM Ericsson
Mary A. DuVal	Texas Instruments

2.3.7 Part G: Sample Data

Version 1.2

Joel Linsky	Silicon Wave

Previous versions

Thomas Müller	Nokia Corporation
Thomas Sander	Nokia Corporation
Joakim Persson (Section Owner)	Telefonaktiebolaget LM Ericsson

2.3.8 Part H: Security Specification

Please see Part B (Section 2.3.2 on page 62). The Security specification was until recently a part of the Baseband specification.

2.4 [VOL 3] CORE SYSTEM PACKAGE, HOST

2.4.1 Part A: Logical Link Control and Adaptation Protocol Specification

Version 1.2

Tom Siep	Bluetooth SIG Inc.
Carsten Andersen (section owner	CSR
Jennifer Bray	CSR
Jan Åberg	Ericsson
Martin van der Zee	Ericsson
Sam Ravnborg	Ericsson
Stefan Agnani	Ericsson
Steve McGowan	Intel Corporation
Joby Lafky	Microsoft
Doron Holan	Microsoft
Andy Glass	Microsoft
Brian Redding	Motorola
Jürgen Schnitzler	Nokia
Thomas Müller	Nokia
Rob Davies	Philips
Terry Bourk	Silicon Wave
Michael Hasling	Tality

Previous versions

Jon Burgess	3Com Corporation
Paul Moran	3Com Corporation
Doug Kogan	Extended Systems
Kevin D. Marquess	Hyper Corporation
Toru Aihara	IBM Corporation
Chatschik Bisdikian	IBM Corporation
Kris Fleming	Intel Corporation
Uma Gadamsetty	Intel Corporation
Robert Hunter	Intel Corporation
Jon Inouye	Intel Corporation
Steve C. Lo	Intel Corporation
Chunrong Zhu	Intel Corporation
Sergey Solyanik	Microsoft Corporation
David E. Cypher	NIST
Nada Golmie	NIST
Thomas Busse	Nokia Corporation
Rauno Makinen	Nokia Corporation
Thomas Müller	Nokia Corporation
Petri Nykänen	Nokia Corporation
Peter Ollikainen	Nokia Corporation
Petri O. Nurminen	Nokia Corporation

Johannes Elg	Telefonaktiebolaget LM Ericsson
Jaap Haartsen	Telefonaktiebolaget LM Ericsson
Elco Nijboer	Telefonaktiebolaget LM Ericsson
Ingemar Nilsson	Telefonaktiebolaget LM Ericsson
Stefan Runesson	Telefonaktiebolaget LM Ericsson
Gerrit Slot	Telefonaktiebolaget LM Ericsson
Johan Sörensen	Telefonaktiebolaget LM Ericsson
Goran Svennarp	Telefonaktiebolaget LM Ericsson
Mary A. DuVal	Texas Instruments
Thomas M. Siep	Texas Instruments
Kinoshita Katsuhiro	Toshiba Corporation

2.4.2 Part B: Service Discovery Protocol (SDP)

Version 1.2

Previous versions

Ned Plasson	3Com Corporation
John Avery	Convergence
Jason Kronz	Convergence
Chatschik Bisdikian	IBM Corporation
Parviz Kermani	IBM Corporation
Brent Miller	IBM Corporation
Dick Osterman	IBM Corporation
Bob Pascoe	IBM Corporation
Jon Inouye	Intel Corporation
Srikanth Kambhatla	Intel Corporation
Jay Eaglstun	Motorola, Inc.
Dale Farnsworth (Section Owner)	Motorola, Inc.
Jean-Michel Rosso	Motorola, Inc.
Jan Grönholm	Nokia Corporation
Kati Rantala	Nokia Corporation
Thomas Müller	Nokia Corporation
Johannes Elg	Telefonaktiebolaget LM Ericsson
Kazuaki Iwamura	Toshiba Corporation

2.4.3 Part C Generic Access Profile

Version 1.2

Jennifer Bray	CSR
Alexander Thoukydides	CSR
Christian Gehrmann	Ericsson
Henrik Hedlund	Ericsson
Jan Åberg	Ericsson
Stefan Agnani	Ericsson
Thomas Müller	Nokia
Joel Linsky	Silicon Wave
Terry Bourk	Silicon Wave
Katsuhiro Kinoshita	Toshiba

Previous versions

Ken Morley	3Com Corporation
Chatschik Bisdikian	IBM Corporation
Jon Inouye	Intel Corporation
Brian Redding	Motorola, Inc.
David E. Cypher	NIST
Stephane Bouet	Nokia Corporation
Thomas Müller	Nokia Corporation
Martin Roter	Nokia Corporation
Johannes Elg	Telefonaktiebolaget LM Ericsson
Patric Lind (Section Owner)	Telefonaktiebolaget LM Ericsson
Erik Slotboom	Telefonaktiebolaget LM Ericsson
Johan Sörensen	Telefonaktiebolaget LM Ericsson

2.4.4 Part D: Test Support

Version 1.2

Emilio Mira Escartis	Cetecom
Robin Heydon	CSR
Jennifer Bray	CSR
Stefan Agnani (section owner)	Ericsson
Terry Bourk	Silicon Wave
Joel Linsky	Silicon Wave
Michael Hasling	Tality

Previous versions [Test Mode]

Jeffrey Schiffer	Intel Corporation
David E. Cypher	NIST
Daniel Bencak	Nokia Corporation
Arno Kefenbaum	Nokia Corporation
Thomas Müller (Section Owner)	Nokia Corporation

Roland Schmale	Nokia Corporation
Fujio Watanabe	Nokia Corporation
Stefan Agnani	Telefonaktiebolaget LM Ericsson
Mårten Mattsson	Telefonaktiebolaget LM Ericsson
Tobias Melin	Telefonaktiebolaget LM Ericsson
Lars Nord	Telefonaktiebolaget LM Ericsson
Fredrik Töörn	Telefonaktiebolaget LM Ericsson
John Mersh	TTPCom
Ayse Findikli	Zeevo, Inc.

Previous versions [Test Control Interface]

Mike Feldman	Motorola, Inc.
Thomas Müller	Nokia Corporation
Stefan Agnani (Section Owner)	Telefonaktiebolaget LM Ericsson
Mårten Mattsson	Telefonaktiebolaget LM Ericsson
Dan Sönnerstam	Telefonaktiebolaget LM Ericsson





Specification of the Bluetooth System

Wireless connections made easy

Architecture & Terminology Overview

Covered Core Package version:
2.0 + EDR
Current Master TOC issued:
4 November 2004

Bluetooth®

Revision History

The Revision History is shown in the “Appendix” on page 51[vol. 0].

Contributors

The persons who contributed to this specification are listed in the Appendix.

Web Site

This specification can also be found on the official Bluetooth web site:
http://www.bluetooth.com

Disclaimer and Copyright Notice

The copyright in these specifications is owned by the Promoter Members of Bluetooth SIG, Inc. (“Bluetooth SIG”). Use of these specifications and any related intellectual property (collectively, the “Specification”), is governed by the Promoters Membership Agreement among the Promoter Members and Bluetooth SIG (the “Promoters Agreement”), certain membership agreements between Bluetooth SIG and its Adopter and Associate Members (the “Membership Agreements”) and the Bluetooth Specification Early Adopters Agreements (“1.2 Early Adopters Agreements”) among Early Adopter members of the unincorporated Bluetooth special interest group and the Promoter Members (the “Early Adopters Agreement”). Certain rights and obligations of the Promoter Members under the Early Adopters Agreements have been assigned to Bluetooth SIG by the Promoter Members.

Use of the Specification by anyone who is not a member of Bluetooth SIG or a party to an Early Adopters Agreement (each such person or party, a “Member”), is prohibited. The legal rights and obligations of each Member are governed by their applicable Membership Agreement, Early Adopters Agreement or Promoters Agreement. No license, express or implied, by estoppel or otherwise, to any intellectual property rights are granted herein.

Any use of the Specification not in compliance with the terms of the applicable Membership Agreement, Early Adopters Agreement or Promoters Agreement is prohibited and any such prohibited use may result in termination of the applicable Membership Agreement or Early Adopters Agreement and other liability permitted by the applicable agreement or by applicable law to Bluetooth SIG or any of its members for patent, copyright and/or trademark infringement.

THE SPECIFICATION IS PROVIDED “AS IS” WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, SATISFACTORY QUALITY, OR REASONABLE SKILL OR CARE, OR ANY WARRANTY ARISING OUT OF ANY COURSE OF DEALING, USAGE, TRADE PRACTICE, PROPOSAL, SPECIFICATION OR SAMPLE.

Each Member hereby acknowledges that products equipped with the Bluetooth® technology (“Bluetooth® Products”) may be subject to various regulatory controls under the laws and regulations of various governments worldwide. Such laws and regulatory controls may govern, among other things, the combination, operation, use, implementation and distribution of Bluetooth® Products. Examples of such laws and regulatory controls include, but are not limited to, airline regulatory controls, telecommunications regulations, technology transfer controls and health and safety regulations. Each Member is solely responsible for the compliance by their Bluetooth® Products with any such laws and regulations and for obtaining any and all required authorizations, permits, or licenses for their Bluetooth® Products related to such regulations within the applicable jurisdictions. Each Member acknowledges that nothing in the Specification provides any information or assistance in connection with securing such compliance, authorizations or licenses. NOTHING IN THE SPECIFICATION CREATES ANY WARRANTIES, EITHER EXPRESS OR IMPLIED, REGARDING SUCH LAWS OR REGULATIONS.

ALL LIABILITY, INCLUDING LIABILITY FOR INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHTS OR FOR NONCOMPLIANCE WITH LAWS, RELATING TO USE OF THE SPECIFICATION IS EXPRESSLY DISCLAIMED. BY USE OF THE SPECIFICATION, EACH MEMBER EXPRESSLY WAIVES ANY CLAIM AGAINST BLUETOOTH SIG AND ITS PROMOTER MEMBERS RELATED TO USE OF THE SPECIFICATION.

Bluetooth SIG reserves the right to adopt any changes or alterations to the Specification as it deems necessary or appropriate.

Copyright © 1999, 2000, 2001, 2002, 2003, 2004

Agere Systems, Inc.,
Ericsson Technology Licensing, AB,
IBM Corporation,
Intel Corporation,
Microsoft Corporation,
Motorola, Inc.,
Nokia Corporation,
Toshiba Corporation

*Third-party brands and names are the property of their respective owners.

Part A
ARCHITECTURE

Contents **11**

1 General Description **13**
1.1 Overview of Operation 13
1.2 Nomenclature 15

2 Core System Architecture **21**
2.1 Core Architectural Blocks 24
2.1.1 Channel manager 24
2.1.2 L2CAP resource manager 24
2.1.3 Device manager 25
2.1.4 Link manager 25
2.1.5 Baseband resource manager 25
2.1.6 Link controller 26
2.1.7 RF 26

3 Data Transport Architecture **27**
3.1 Core Traffic Bearers 28
3.1.1 Framed data traffic 29
3.1.2 Unframed data traffic 30
3.1.3 Reliability of traffic bearers 30
3.2 Transport Architecture Entities 32
3.2.1 Bluetooth generic packet structure 32
3.3 Physical Channels 34
3.3.1 Basic piconet channel 35
3.3.1.1 Overview 35
3.3.1.2 Characteristics 35
3.3.1.3 Topology 36
3.3.1.4 Supported layers 36
3.3.2 Adapted piconet channel 36
3.3.2.1 Overview 36
3.3.3 Inquiry scan channel 37
3.3.3.1 Overview 37
3.3.3.2 Characteristics 37
3.3.3.3 Topology 38
3.3.3.4 Supported layers 38
3.3.4 Page scan channel 38
3.3.4.1 Overview 38
3.3.4.2 Characteristics 38
3.3.4.3 Topology 39
3.3.4.4 Supported layers 39
3.4 Physical Links 39

3.4.1 Links supported by the basic and adapted piconet physical channel40
3.4.1.1 Active physical link 40
3.4.1.2 Parked physical link 40
3.4.2 Links supported by the scanning physical channels 41
3.5 Logical Links and Logical Transports 41
3.5.1 Casting 43
3.5.2 Scheduling and acknowledgement scheme 43
3.5.3 Class of data 44
3.5.4 Asynchronous connection-oriented (ACL) 44
3.5.5 Synchronous connection-oriented (SCO) 45
3.5.6 Extended synchronous connection-oriented (eSCO) 46
3.5.7 Active slave broadcast (ASB) 46
3.5.8 Parked slave broadcast (PSB) 47
3.5.9 Logical links 48
3.5.10 ACL Control Logical Link (ACL-C) 49
3.5.11 User Asynchronous/Isochronous Logical Link (ACL-U) 49
3.5.12 User Synchronous/Extended Synchronous Logical Links (SCO-S/eSCO-S)49
3.6 L2CAP Channels 50

4 Communication Topology 51
4.1 Piconet Topology 51
4.2 Operational Procedures and Modes 53
4.2.1 Inquiry (Discovering) Procedure 53
4.2.2 Paging (Connecting) Procedure 54
4.2.3 Connected mode 54
4.2.4 Hold mode 55
4.2.5 Sniff mode 55
4.2.6 Parked state 56
4.2.7 Role switch procedure 56
4.2.8 Enhanced Data Rate 57

Part B
ACRONYMS & ABBREVIATIONS

1 List of Acronyms and Abbreviations 61
2 Abbreviations of the Specification Names 69

Part C
CORE SPECIFICATION CHANGE HISTORY

Contents **73**

1 Changes from V1.1 to V1.2 ..... **75**
1.1 New Features 75
1.2 Structure Changes 75
1.3 Deprecated Specifications 75
1.4 Deprecated Features 76
1.5 Changes in Wording 76
1.6 Nomenclature Changes 76

2 Changes from V1.2 to V2.0 + EDR **77**
2.1 New Features 77
2.2 Deprecated Features 77

Part D
MIXING OF SPECIFICATION VERSIONS

Contents **80**

1 Mixing of Specification Versions **81**
1.1 features and their types 82

Part E
IEEE LANGUAGE

Contents **85**

1 Use of IEEE Language **87**
1.1 Shall 87
1.2 Must 88
1.3 Will 88
1.4 Should 88
1.5 May 88
1.6 Can 89



Architecture

CONTENTS

1 General Description 13
1.1 Overview of Operation 13
1.2 Nomenclature 15
2 Core System Architecture 21
2.1 Core Architectural Blocks 24
2.1.1 Channel manager 24
2.1.2 L2CAP resource manager 24
2.1.3 Device manager 25
2.1.4 Link manager 25
2.1.5 Baseband resource manager 25
2.1.6 Link controller 26
2.1.7 RF 26
3 Data Transport Architecture 27
3.1 Core Traffic Bearers 28
3.1.1 Framed data traffic 29
3.1.2 Unframed data traffic 30
3.1.3 Reliability of traffic bearers 30
3.2 Transport Architecture Entities 32
3.2.1 Bluetooth generic packet structure 32
3.3 Physical Channels 34
3.3.1 Basic piconet channel 35
3.3.1.1 Overview 35
3.3.1.2 Characteristics 35
3.3.1.3 Topology 36
3.3.1.4 Supported layers 36
3.3.2 Adapted piconet channel 36
3.3.2.1 Overview 36
3.3.3 Inquiry scan channel 37
3.3.3.1 Overview 37
3.3.3.2 Characteristics 37
3.3.3.3 Topology 38
3.3.3.4 Supported layers 38
3.3.4 Page scan channel 38
3.3.4.1 Overview 38
3.3.4.2 Characteristics 38
3.3.4.3 Topology 39
3.3.4.4 Supported layers 39
3.4 Physical Links 39

3.4.1 Links supported by the basic and adapted piconet physical channel 40
3.4.1.1 Active physical link 40
3.4.1.2 Parked physical link 40
3.4.2 Links supported by the scanning physical channels 41
3.5 Logical Links and Logical Transports 41
3.5.1 Casting 43
3.5.2 Scheduling and acknowledgement scheme 43
3.5.3 Class of data 44
3.5.4 Asynchronous connection-oriented (ACL) 44
3.5.5 Synchronous connection-oriented (SCO) 45
3.5.6 Extended synchronous connection-oriented (eSCO) 46
3.5.7 Active slave broadcast (ASB) 46
3.5.8 Parked slave broadcast (PSB) 47
3.5.9 Logical links 48
3.5.10 ACL Control Logical Link (ACL-C) 49
3.5.11 User Asynchronous/Isochronous Logical Link (ACL-U) 49
3.5.12 User Synchronous/Extended Synchronous Logical Links (SCO-S/eSCO-S) 49
3.6 L2CAP Channels 50

4 Communication Topology 51

4.1 Piconet Topology 51
4.2 Operational Procedures and Modes 53
4.2.1 Inquiry (Discovering) Procedure 53
4.2.2 Paging (Connecting) Procedure 54
4.2.3 Connected mode 54
4.2.4 Hold mode 55
4.2.5 Sniff mode 55
4.2.6 Parked state 56
4.2.7 Role switch procedure 56
4.2.8 Enhanced Data Rate 57

1 GENERAL DESCRIPTION

Bluetooth wireless technology is a short-range communications system intended to replace the cable(s) connecting portable and/or fixed electronic devices. The key features of Bluetooth wireless technology are robustness, low power, and low cost. Many features of the core specification are optional, allowing product differentiation.

The Bluetooth core system consists of an RF transceiver, baseband, and protocol stack. The system offers services that enable the connection of devices and the exchange of a variety of classes of data between these devices.

This chapter of the specification provides an overview of the Bluetooth system architecture, communication topologies and data transport features. The text in this chapter of the specification should be treated as informational and used as a background and for context-setting.

1.1 OVERVIEW OF OPERATION

The Bluetooth RF (physical layer) operates in the unlicensed ISM band at 2.4 GHz. The system employs a frequency hop transceiver to combat interference and fading and provides many FHSS carriers. RF operation uses a shaped, binary frequency modulation to minimize transceiver complexity. The symbol rate is 1 Megasymbol per second (Ms/s) supporting the bit rate of 1 Megabit per second (Mb/s) or, with Enhanced Data Rate, a gross air bit rate of 2 or 3Mb/s. These modes are known as Basic Rate and Enhanced Data Rate respectively.

During typical operation a physical radio channel is shared by a group of devices that are synchronized to a common clock and frequency hopping pattern. One device provides the synchronization reference and is known as the master. All other devices are known as slaves. A group of devices synchronized in this fashion form a piconet. This is the fundamental form of communication in the Bluetooth wireless technology.

Devices in a piconet use a specific frequency hopping pattern, which is algorithmically determined by certain fields in the Bluetooth address and clock of the master. The basic hopping pattern is a pseudo-random ordering of the 79 frequencies in the ISM band. The hopping pattern may be adapted to exclude a portion of the frequencies that are used by interfering devices. The adaptive hopping technique improves Bluetooth co-existence with static (non-hopping) ISM systems when these are co-located.

The physical channel is sub-divided into time units known as slots. Data is transmitted between Bluetooth devices in packets, that are positioned in these slots. When circumstances permit, a number of consecutive slots may be allocated to a single packet. Frequency hopping takes place between the transmis-

sion or reception of packets. Bluetooth technology provides the effect of full duplex transmission through the use of a Time-Division Duplex (TDD) scheme.

Above the physical channel there is a layering of links and channels and associated control protocols. The hierarchy of channels and links from the physical channel upwards is physical channel, physical link, logical transport, logical link and L2CAP channel. These are discussed in more detail in Section 3.3 on page 34 - Section 3.6 on page 50 but are introduced here to aid the understanding of the remainder of this section.

Within a physical channel, a physical link is formed between any two devices that transmit packets in either direction between them. In a piconet physical channel there are restrictions on which devices may form a physical link. There is a physical link between each slave and the master. Physical links are not formed directly between the slaves in a piconet.

The physical link is used as a transport for one or more logical links that support unicast synchronous, asynchronous and isochronous traffic, and broadcast traffic. Traffic on logical links is multiplexed onto the physical link by occupying slots assigned by a scheduling function in the resource manager.

A control protocol for the baseband and physical layers is carried over logical links in addition to user data. This is the link manager protocol (LMP). Devices that are active in a piconet have a default asynchronous connection-oriented logical transport that is used to transport the LMP protocol signalling. For historical reasons this is known as the ACL logical transport. The default ACL logical transport is the one that is created whenever a device joins a piconet. Additional logical transports may be created to transport synchronous data streams when this is required.

The Link Manager function uses LMP to control the operation of devices in the piconet and provide services to manage the lower architectural layers (radio layer and baseband layer). The LMP protocol is only carried on the default ACL logical transport and the default broadcast logical transport.

Above the baseband layer the L2CAP layer provides a channel-based abstraction to applications and services. It carries out segmentation and reassembly of application data and multiplexing and de-multiplexing of multiple channels over a shared logical link. L2CAP has a protocol control channel that is carried over the default ACL logical transport. Application data submitted to the L2CAP protocol may be carried on any logical link that supports the L2CAP protocol.

1.2 NOMENCLATURE

Where the following terms appear in the specification they have the meaning given in Table 1.1 on page 15.

Ad Hoc Network	A network typically created in a spontaneous manner. An ad hoc network requires no formal infrastructure and is limited in temporal and spatial extent.
Active Slave Broadcast (ASB)	The Active Slave Broadcast logical transport that is used to transport L2CAP user traffic to all active devices in the piconet. See Section 3.5.7 on page 46
Beacon Train	A pattern of reserved slots within a basic or adapted piconet physical channel. Transmissions starting in these slots are used to resynchronize parked devices.
Bluetooth	Bluetooth is a wireless communication link, operating in the unlicensed ISM band at 2.4 GHz using a frequency hopping transceiver. It allows real-time AV and data communications between Bluetooth Hosts. The link protocol is based on time slots.
Bluetooth Baseband	The part of the Bluetooth system that specifies or implements the medium access and physical layer procedures to support the exchange of real-time voice, data information streams, and ad hoc networking between Bluetooth Devices.
Bluetooth Clock	A 28 bit clock internal to a Bluetooth controller sub-system that ticks every 312.5µs. The value of this clock defines the slot numbering and timing in the various physical channels.
Bluetooth Controller	A sub-system containing the Bluetooth RF, baseband, resource controller, link manager, device manager and a Bluetooth HCI.
Bluetooth Device	A Bluetooth Device is a device that is capable of short-range wireless communications using the Bluetooth system.
Bluetooth Device Address	A 48 bit address used to identify each Bluetooth device.
BD_ADDR	The Bluetooth Device Address, BD_ADDR, is used to identify a Bluetooth device.
Bluetooth HCI	The Bluetooth Host Controller Interface (HCI) provides a command interface to the baseband controller and link manager and access to hardware status and control registers. This interface provides a uniform method of accessing the Bluetooth baseband capabilities.

Table 1.1: Nomenclature.

Bluetooth Host	A Bluetooth Host is a computing device, peripheral, cellular telephone, access point to PSTN network or LAN, etc. A Bluetooth Host attached to a Bluetooth Controller may communicate with other Bluetooth Hosts attached to their Bluetooth Controllers as well.
Channel	Either a physical channel or an L2CAP channel, depending on the context.
Connect (to service)	The establishment of a connection to a service. If not already done, this also includes establishment of a physical link, logical transport, logical link and L2CAP channel.
Connectable device	A Bluetooth device in range that periodically listens on its page scan physical channel and will respond to a page on that channel.
Connected devices	Two Bluetooth devices in the same piconet and with a physical link between them.
Connecting	A phase in the communication between devices when a connection between them is being established. (Connecting phase follows after the link establishment phase is completed.)
Connection	A connection between two peer applications or higher layer protocols mapped onto an L2CAP channel.
Connection establishment	A procedure for creating a connection mapped onto a channel.
coverage area	The area where two Bluetooth devices can exchange messages with acceptable quality and performance.
Creation of a secure connection	A procedure of establishing a connection, including authentication and encryption.
Creation of a trusted relationship	A procedure where the remote device is marked as a trusted device. This includes storing a common link key for future authentication and pairing (if the link key is not available).
Device discovery	A procedure for retrieving the Bluetooth device address, clock, class-of-device field and used page scan mode from discoverable devices.
Discoverable device	A Bluetooth device in range that periodically listens on an inquiry scan physical channel and will respond to an inquiry on that channel. Discoverable device are normally also connectable.
Inquiring device	A Bluetooth device that is carrying out the inquiry procedure.
Inquiry	A procedure where a Bluetooth device transmits inquiry messages and listens for responses in order to discover the other Bluetooth devices that are within the coverage area.

Table 1.1: Nomenclature.

Inquiry scan	A procedure where a Bluetooth device listens for inquiry messages received on its inquiry scan physical channel.
Interoperability	The ability of two or more systems or components to exchange information and to use the information that has been exchanged.
Isochronous data	Information in a stream where each information entity in the stream is bound by a time relationship to previous and successive entities.
Known device	A Bluetooth device for which at least the BD_ADDR is stored.
L2CAP Channel	A logical connection on L2CAP level between two devices serving a single application or higher layer protocol.
L2CAP Channel establishment	A procedure for establishing a logical connection on L2CAP level.
Link establishment	A procedure for establishing the default ACL link and hierarchy of links and channels between devices.
Link	Shorthand for a logical link.
Link key	A secret key that is known by two devices and is used in order to authenticate each device to the other
LMP authentication	An LMP level procedure for verifying the identity of a remote device.
LMP pairing	A procedure that authenticates two devices and creates a common link key that can be used as a basis for a trusted relationship or a (single) secure connection.
Logical Channel	Identical to an L2CAP channel, but deprecated due to an alternative meaning in Bluetooth 1.1
Logical link	The lowest architectural level used to offer independent data transport services to clients of the Bluetooth system.
Logical transport	Used in Bluetooth to represent commonality between different logical links due to shared acknowledgement protocol and link identifiers.
Name discovery	A procedure for retrieving the user-friendly name (the Bluetooth device name) of a connectable device.
Packet	Format of aggregated bits that are transmitted on a physical channel.
Page	The initial phase of the connection procedure where a device transmits a train of page messages until a response is received from the target device or a timeout occurs.
Page scan	A procedure where a device listens for page messages received on its page scan physical channel.

Table 1.1: Nomenclature.

Paging device	A Bluetooth device that is carrying out the page procedure.
Paired device	A Bluetooth device with which a link key has been exchanged (either before connection establishment was requested or during connecting phase).
Parked device	A device operating in a basic mode piconet that is synchronized to the master but has given up its default ACL logical transport.
Physical Channel	Characterized by synchronized occupancy of a sequence of RF carriers by one or more devices. A number of physical channel types exist with characteristics defined for their different purposes.
Physical Link	A Baseband-level connection between two devices established using paging.
Piconet	A collection of devices occupying a shared physical channel where one of the devices is the Piconet Master and the remaining devices are connected to it.
Piconet Physical Channel	A Channel that is divided into time slots in which each slot is related to an RF hop frequency. Consecutive hops normally correspond to different RF hop frequencies and occur at a standard hop rate of 1600 hops/s. These consecutive hops follow a pseudo-random hopping sequence, hopping through a 79 RF channel set.
Piconet Master	The device in a piconet whose Bluetooth Clock and Bluetooth Device Address are used to define the piconet physical channel characteristics.
Piconet Slave	Any device in a piconet that is not the Piconet Master, but is connected to the Piconet Master.
PIN	A user-friendly number that can be used to authenticate connections to a device before paring has taken place.
PMP	A Participant in Multiple Piconets. A device that is concurrently a member of more than one piconet, which it achieves using time division multiplexing (TDM) to interleave its activity on each piconet physical channel.
The Parked Slave Broadcast (PSB)	The Parked Slave Broadcast logical transport that is used for communications between the master and parked devices. Section 3.5.8 on page 47.
Scatternet	Two or more piconets that include one or more devices acting as PMPs.
Service Layer Protocol	A protocol that uses an L2CAP channel for transporting PDUs.
Service discovery	Procedures for querying and browsing for services offered by or through another Bluetooth device.

Table 1.1: Nomenclature.

Silent device	A Bluetooth device appears as silent to a remote device if it does not respond to inquiries made by the remote device.
Unknown device	A Bluetooth device for which no information (Bluetooth Device Address, link key or other) is stored.

Table 1.1: Nomenclature.

2 CORE SYSTEM ARCHITECTURE

The Bluetooth core system covers the four lowest layers and associated protocols defined by the Bluetooth specification as well as one common service layer protocol, the Service Discovery Protocol (SDP) and the overall profile requirements are specified in the Generic Access Profile (GAP). A complete Bluetooth application requires a number of additional service and higher layer protocols that are defined in the Bluetooth specification, but are not described here. The core system architecture is shown in Figure 2.1 on page 21 except for SDP that is not shown for clarity.

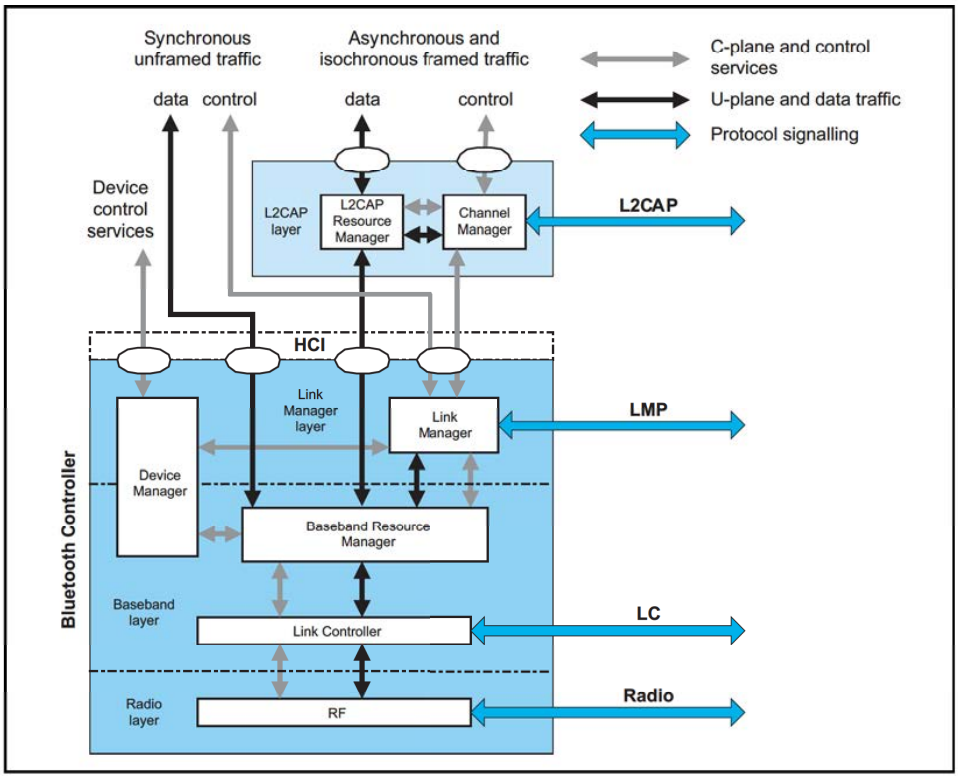


Figure 2.1: Bluetooth core system architecture

Figure 2.1 on page 21 shows the four lowest layers, each with its associated communication protocol. The lowest three layers are sometimes grouped into a subsystem known as the Bluetooth controller. This is a common implementation involving a standard physical communications interface between the Bluetooth controller and remainder of the Bluetooth system including the L2CAP, service and higher layers (known as the Bluetooth host.) Although this interface is optional the architecture is designed to allow for its existence and characteristics. The Bluetooth specification enables inter-operability between independent Bluetooth systems by defining the protocol messages exchanged between equivalent layers, and also inter-operability between independent

Bluetooth sub-systems by defining a common interface between Bluetooth controllers and Bluetooth hosts.

A number of functional blocks are shown and the path of services and data between these. The functional blocks shown in the diagram are informative; in general the Bluetooth specification does not define the details of implementations except where this is required for inter-operability. Thus the functional blocks in Figure 2.1 on page 21 are shown in order to aid description of the system behavior. An implementation may be different from the system shown in Figure 2.1 on page 21.

Standard interactions are defined for all inter-device operation, where Bluetooth devices exchange protocol signalling according to the Bluetooth specification. The Bluetooth core system protocols are the Radio (RF) protocol, Link Control (LC) protocol, Link Manager (LM) protocol and Logical Link Control and Adaptation protocol (L2CAP), all of that are fully defined in subsequent parts of the Bluetooth specification. In addition the Service Discovery Protocol (SDP) is a service layer protocol required by all Bluetooth applications.

The Bluetooth core system offers services through a number of service access points that are shown in the diagram as ellipses. These services consist of the basic primitives that control the Bluetooth core system. The services can be split into three types. There are device control services that modify the behavior and modes of a Bluetooth device, transport control services that create, modify and release traffic bearers (channels and links), and data services that are used to submit data for transmission over traffic bearers. It is common to consider the first two as belonging to the C-plane and the last as belonging to the U-plane.

A service interface to the Bluetooth controller sub-system is defined such that the Bluetooth controller may be considered a standard part. In this configuration the Bluetooth controller operates the lowest three layers and the L2CAP layer is contained with the rest of the Bluetooth application in a host system. The standard interface is called the Host to Controller Interface (HCI) and its service access points are represented by the ellipses on the upper edge of the Bluetooth controller sub-system in Figure 2.1 on page 21. Implementation of this standard service interface is optional.

As the Bluetooth architecture is defined with the possibility of separate host and controller communicating through an HCI, a number of general assumptions are made. The Bluetooth controller is assumed to have limited data buffering capabilities in comparison with the host. Therefore the L2CAP layer is expected to carry out some simple resource management when submitting L2CAP PDUs to the controller for transport to a peer device. This includes segmentation of L2CAP SDUs into more manageable PDUs and then the fragmentation of PDUs into start and continuation packets of a size suitable for the controller buffers, and management of the use of controller buffers to ensure availability for channels with Quality of Service (QoS) commitments.

The Baseband layer provides the basic ARQ protocol in Bluetooth. The L2CAP layer can optionally provide a further error detection and retransmission to the L2CAP PDUs. This feature is recommended for applications with requirements for a low probability of undetected errors in the user data. A further optional feature of L2CAP is a window-based flow control that can be used to manage buffer allocation in the receiving device. Both of these optional features augment the QoS performance in certain scenarios.

Although these assumptions may not be required for embedded Bluetooth implementations that combine all layers in a single system, the general architectural and QoS models are defined with these assumptions in mind, in effect a lowest common denominator.

Automated conformance testing of implementations of the Bluetooth core system is required. This is achieved by allowing the tester to control the implementation through the RF interface, which is common to all Bluetooth systems, and through the Test Control Interface (TCI), which is only required for conformance testing.

The tester uses exchanges with the Implementation Under Test (IUT) through the RF interface to ensure the correct responses to requests from remote devices. The tester controls the IUT through the TCI to cause the IUT to originate exchanges through the RF interface so that these can also be verified as conformant.

The TCI uses a different command-set (service interface) for the testing of each architectural layer and protocol. A subset of the HCI command-set is used as the TCI service interface for each of the layers and protocols within the Bluetooth Controller subsystem. A separate service interface is used for testing the L2CAP layer and protocol. As an L2CAP service interface is not defined in the Bluetooth core specification it is defined separately in the Test Control Interface specification. Implementation of the L2CAP service interface is only required for conformance testing.

2.1 CORE ARCHITECTURAL BLOCKS

This section describes the function and responsibility of each of the blocks shown in Figure 2.1 on page 21. An implementation is not required to follow the architecture described above, though every implementation shall conform to the protocol specifications described in subsequent parts of the Bluetooth specification, and shall implement the behavioral aspects of the system outlined below and specified in subsequent parts of the Bluetooth specification.

2.1.1 Channel manager

The channel manager is responsible for creating, managing and destroying L2CAP channels for the transport of service protocols and application data streams. The channel manager uses the L2CAP protocol to interact with a channel manager on a remote (peer) device to create these L2CAP channels and connect their endpoints to the appropriate entities. The channel manager interacts with its local link manager to create new logical links (if necessary) and to configure these links to provide the required quality of service for the type of data being transported.

2.1.2 L2CAP resource manager

The L2CAP resource manager block is responsible for managing the ordering of submission of PDU fragments to the baseband and some relative scheduling between channels to ensure that L2CAP channels with QoS commitments are not denied access to the physical channel due to Bluetooth controller resource exhaustion. This is required because the architectural model does not assume that the Bluetooth controller has limitless buffering, or that the HCI is a pipe of infinite bandwidth.

L2CAP Resource Managers may also carry out traffic conformance policing to ensure that applications are submitting L2CAP SDUs within the bounds of their negotiated QoS settings. The general Bluetooth data transport model assumes well-behaved applications, and does not define how an implementation is expected to deal with this problem.

2.1.3 Device manager

The device manager is the functional block in the baseband that controls the general behavior of the Bluetooth device. It is responsible for all operation of the Bluetooth system that is not directly related to data transport, such as inquiring for the presence of other nearby Bluetooth devices, connecting to other Bluetooth devices, or making the local Bluetooth device discoverable or connectable by other devices.

The device manager requests access to the transport medium from the baseband resource controller in order to carry out its functions.

The device manager also controls local device behavior implied by a number of the HCI commands, such as managing the device local name, any stored link keys, and other functionality.

2.1.4 Link manager

The link manager is responsible for the creation, modification and release of logical links (and, if required, their associated logical transports), as well as the update of parameters related to physical links between devices. The link manager achieves this by communicating with the link manager in remote Bluetooth devices using the Link Management Protocol (LMP.)

The LM protocol allows the creation of new logical links and logical transports between devices when required, as well as the general control of link and transport attributes such as the enabling of encryption on the logical transport, the adapting of transmit power on the physical link, or the adjustment of QoS settings for a logical link.

2.1.5 Baseband resource manager

The baseband resource manager is responsible for all access to the radio medium. It has two main functions. At its heart is a scheduler that grants time on the physical channels to all of the entities that have negotiated an access contract. The other main function is to negotiate access contracts with these entities. An access contract is effectively a commitment to deliver a certain QoS that is required in order to provide a user application with an expected performance.

The access contract and scheduling function must take account of any behavior that requires use of the Bluetooth radio. This includes (for example) the normal exchange of data between connected devices over logical links, and logical transports, as well as the use of the radio medium to carry out inquiries, make connections, be discoverable or connectable, or to take readings from unused carriers during the use of adaptive frequency hopping mode.

In some cases the scheduling of a logical link results in changing to a different physical channel from the one that was previously used. This may be (for example) due to involvement in scatternet, a periodic inquiry function, or page

scanning. When the physical channels are not time slot aligned, then the resource manager also accounts for the realignment time between slots on the original physical channel and slots on the new physical channel. In some cases the slots will be naturally aligned due to the same device clock being used as a reference for both physical channels.

2.1.6 Link controller

The link controller is responsible for the encoding and decoding of Bluetooth packets from the data payload and parameters related to the physical channel, logical transport and logical link.

The link controller carries out the link control protocol signalling (in close conjunction with the scheduling function of the resource manager), which is used to communicate flow control and acknowledgement and retransmission request signals. The interpretation of these signals is a characteristic of the logical transport associated with the baseband packet. Interpretation and control of the link control signalling is normally associated with the resource manager's scheduler.

2.1.7 RF

The RF block is responsible for transmitting and receiving packets of information on the physical channel. A control path between the baseband and the RF block allows the baseband block to control the timing and frequency carrier of the RF block. The RF block transforms a stream of data to and from the physical channel and the baseband into required formats.

3 DATA TRANSPORT ARCHITECTURE

The Bluetooth data transport system follows a layered architecture. This description of the Bluetooth system describes the Bluetooth core transport layers up to and including L2CAP channels. All Bluetooth operational modes follow the same generic transport architecture, which is shown in Figure 3.1 on page 27.

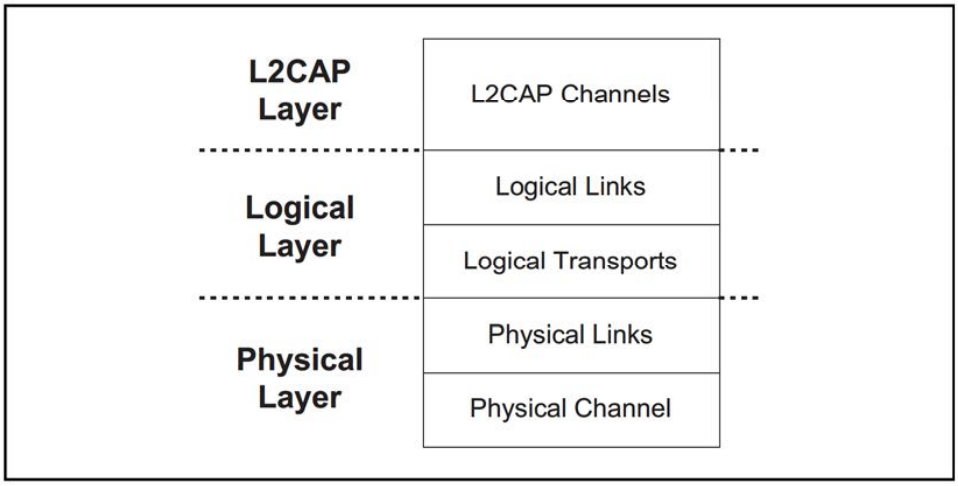


Figure 3.1: Bluetooth generic data transport architecture

For efficiency and legacy reasons, the Bluetooth transport architecture includes a sub-division of the logical layer, distinguishing between logical links and logical transports. This sub-division provides a general (and commonly understood) concept of a logical link that provides an independent transport between two or more devices. The logical transport sub-layer is required to describe the inter-dependence between some of the logical link types (mainly for reasons of legacy behavior.)

The Bluetooth 1.1 specification described the ACL and SCO links as physical links. With the addition of Extended SCO (eSCO) and for future expansion it is better to consider these as logical transport types, which more accurately encapsulates their purpose. However, they are not as independent as might be desired, due to their shared use of resources such as the LT_ADDR and acknowledgement/repeat request (ARQ) scheme. Hence the architecture is incapable of representing these logical transports with a single transport layer. The additional logical transport layer goes some way towards describing this behavior.

3.1 CORE TRAFFIC BEARERS

The Bluetooth core system provides a number of standard traffic bearers for the transport of service protocol and application data. These are shown in Figure 3.2 on page 28 below (for ease of representation this is shown with higher layers to the left and lower layers to the right).

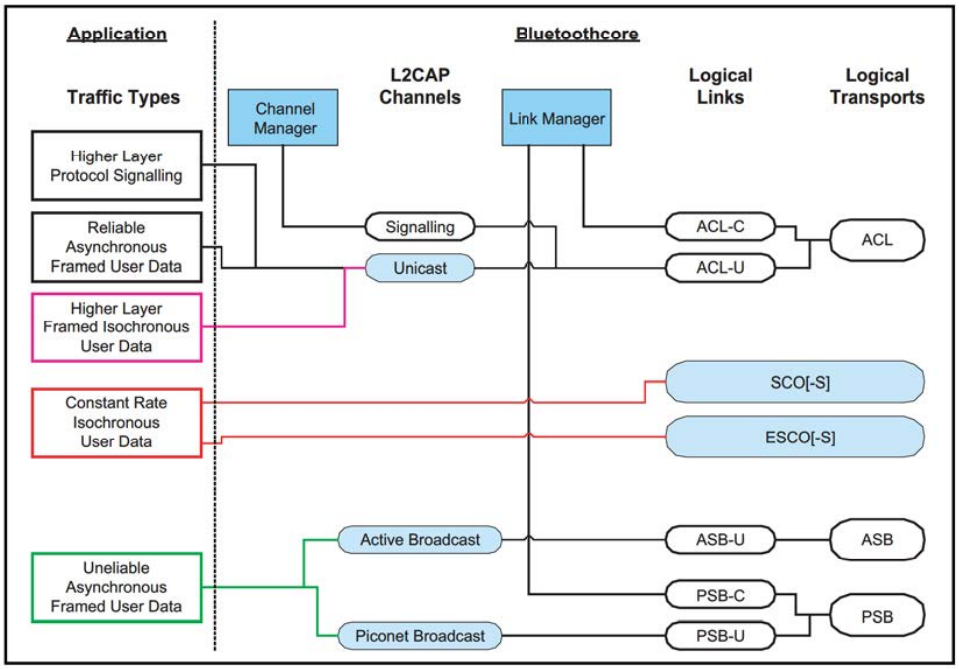


Figure 3.2: Bluetooth traffic bearers

The core traffic bearers that are available to applications are shown in Figure 3.2 on page 28 as the shaded rounded rectangles. The architectural layers that are defined to provide these services are described in Section 2 on page 21. A number of data traffic types are shown on the left of the diagram linked to the traffic bearers that are typically suitable for transporting that type of data traffic.

The logical links are named using the names of the associated logical transport and a suffix that indicates the type of data that is transported. (C for control links carrying LMP messages, U for L2CAP links carrying user data (L2CAP PDUs) and S for stream links carrying unformatted synchronous or isochronous data.) It is common for the suffix to be removed from the logical link without introducing ambiguity, thus a reference to the default ACL logical transport can be resolved to mean the ACL-C logical link in cases where the LMP protocol is being discussed, or the ACL-U logical link when the L2CAP layer is being discussed.

The mapping of application traffic types to Bluetooth core traffic bearers in Figure 3.2 on page 28 is based on matching the traffic characteristics with the

bearer characteristics. It is recommended to use these mappings as they provide the most natural and efficient method of transporting the data with its given characteristics.

However, an application (or an implementation of the Bluetooth core system) may choose to use a different traffic bearer, or a different mapping to achieve a similar result. For example, in a piconet with only one slave, the master may choose to transport L2CAP broadcasts over the ACL-U logical link rather than over the ASB-U or PSB-U logical links. This will probably be more efficient in terms of bandwidth (if the physical channel quality is not too degraded.) Use of alternative transport paths to those in Figure 3.2 on page 28 is only acceptable if the characteristics of the application traffic type are preserved.

Figure 3.2 on page 28 shows a number of application traffic types. These are used to classify the types of data that may be submitted to the Bluetooth core system. The original data traffic type may not be the same as the type that is submitted to the Bluetooth core system if an intervening process modifies it. For example, video data is generated at a constant rate but an intermediate coding process may alter this to variable rate, e.g. by MPEG4 encoding. For the purposes of the Bluetooth core system, only the characteristic of the submitted data is of interest.

3.1.1 Framed data traffic

The L2CAP layer services provide a frame-oriented transport for asynchronous and isochronous user data. The application submits data to this service in variable-sized frames (up to a negotiated maximum for the channel) and these frames are delivered in the same form to the corresponding application on the remote device. There is no requirement for the application to insert additional framing information into the data, although it may do so if this is required (such framing is invisible to the Bluetooth core system.)

Connection-oriented L2CAP channels may be created for transport of unicast (point-to-point) data between two Bluetooth devices. A connectionless L2CAP channel exists for broadcasting data. In the case of piconet topologies the master device is always the source of broadcast data and the slave device(s) are the recipients. Traffic on the broadcast L2CAP channel is uni-directional. Unicast L2CAP channels may be uni-directional or bi-directional.

L2CAP channels have an associated QoS setting that defines constraints on the delivery of the frames of data. These QoS settings may be used to indicate (for example) that the data is isochronous, and therefore has a limited lifetime after which it becomes invalid, or that the data should be delivered within a given time period, or that the data is reliable and should be delivered without error, however long this takes.

The L2CAP channel manager is responsible for arranging to transport the L2CAP channel data frames on an appropriate baseband logical link, possibly multiplexing this onto the baseband logical link with other L2CAP channels with similar characteristics.

3.1.2 Unframed data traffic

If the application does not require delivery of data in frames, possibly because it includes in-stream framing, or because the data is a pure stream, then it may avoid the use of L2CAP channels and make direct use of a baseband logical link.

The Bluetooth core system supports the direct transport of application data that is isochronous and of a constant rate (either bit-rate, or frame-rate for pre-framed data), using a SCO-S or eSCO-S logical link. These logical links reserve physical channel bandwidth and provide a constant rate transport locked to the piconet clock. Data is transported in fixed size packets at fixed intervals with both of these parameters negotiated during channel establishment. eSCO links provide a greater choice of bit-rates and also provide greater reliability by using limited retransmission in case of error. Enhanced Data Rate operation is supported for eSCO, but not for SCO logical transports. SCO and eSCO logical transports do not support multiplexed logical links or any further layering within the Bluetooth core. An application may choose to layer a number of streams within the submitted SCO/eSCO stream, provided that the submitted stream is, or has the appearance of being, a constant rate stream.

The application chooses the most appropriate type of logical link from those available at the baseband, and creates and configures it to transport the data stream, and releases it when completed. (The application will normally also use a framed L2CAP unicast channel to transport its C-plane information to the peer application on the remote device.)

If the application data is isochronous and of a variable rate, then this may only be carried by the L2CAP unicast channel, and hence will be treated as framed data.

3.1.3 Reliability of traffic bearers

Bluetooth is a wireless communications system. In poor RF environments, this system should be considered inherently unreliable. To counteract this the system provides levels of protection at each layer. The baseband packet header uses forward error correcting (FEC) coding to allow error correction by the receiver and a header error check (HEC) to detect errors remaining after correction. Certain Baseband packet types include FEC for the payload. Furthermore, some Baseband packet types include a cyclic redundancy error check (CRC).

On ACL logical transports the results of the error detection algorithm are used to drive a simple acknowledgement/repeat request (ARQ) protocol. This provides an enhanced reliability by re-transmitting packets that do not pass the receiver's error checking algorithm. It is possible to modify this scheme to support latency-sensitive packets by discarding an unsuccessfully transmitted packet at the transmitter if the packet's useful life has expired. eSCO links use

a modified version of this scheme to improve reliability by allowing a limited number of retransmissions.

The resulting reliability gained by this ARQ scheme is only as dependable as the ability of the HEC and CRC codes to detect errors. In most cases this is sufficient, however it has been shown that for the longer packet types the probability of an undetected error is too high to support typical applications, especially those with a large amount of data being transferred.

The L2CAP layer provides an additional level of error control that is designed to detect the occasional undetected errors in the baseband layer and request retransmission of the affected data. This provides the level of reliability required by typical Bluetooth applications.

Broadcast links have no feedback route, and are unable to use the ARQ scheme (although the receiver is still able to detect errors in received packets.) Instead each packet is transmitted several times in the hope that the receiver is able to receive at least one of the copies successfully. Despite this approach there are still no guarantees of successful receipt, and so these links are considered unreliable.

In summary, if a link or channel is characterized as reliable this means that the receiver is capable of detecting errors in received packets and requesting retransmission until the errors are removed. Due to the error detection system used some residual (undetected) errors may still remain in the received data. For L2CAP channels the level of these is comparable to other communication systems, although for logical links the residual error level is somewhat higher.

The transmitter may remove packets from the transmit queue such that the receiver does not receive all the packets in the sequence. If this happens detection of the missing packets is delegated to the L2CAP layer.

On an unreliable link the receiver is capable of detecting errors in received packets but cannot request retransmission. The packets passed on by the receiver may be without error, but there is no guarantee that all packets in the sequence are received. Hence the link is considered fundamentally unreliable. There are limited uses for such links, and these uses are normally dependent on the continuous repetition of data from the higher layers while it is valid.

Stream links have a reliability characteristic somewhere between a reliable and an unreliable link, depending on the current operating conditions.

3.2 TRANSPORT ARCHITECTURE ENTITIES

The Bluetooth transport architecture entities are shown in Figure 3.3 on page 32 and are described from the lowest layer upwards in the subsequent sections.

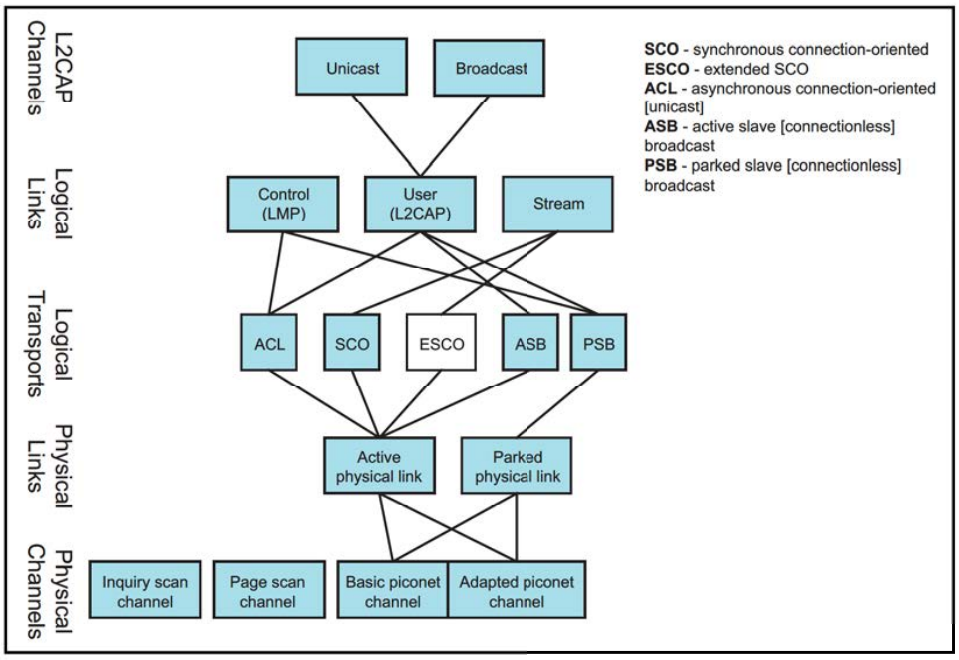


Figure 3.3: Overview of transport architecture entities and hierarchy

3.2.1 Bluetooth generic packet structure

The general packet structure nearly reflects the architectural layers found in the Bluetooth system. The packet structure is designed for optimal use in normal operation. It is shown in Figure 3.4 on page 32.

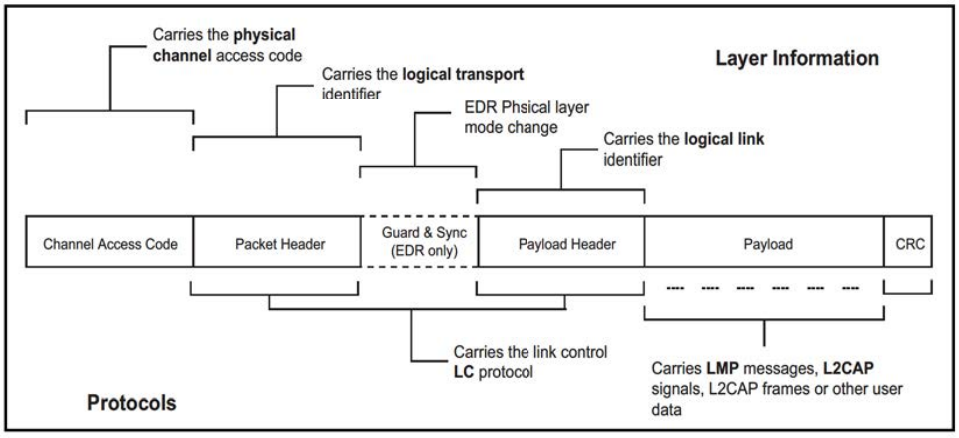


Figure 3.4: Bluetooth packet structure

Packets normally only include the fields that are necessary to represent the layers required by the transaction. Thus a simple inquiry request over an inquiry scan physical channel does not create or require a logical link or higher layer and therefore consists only of the channel access code (associated with the physical channel.) General communication within a piconet uses packets that include all of the fields, as all of the architectural layers are used.

All packets include the channel access code. This is used to identify communications on a particular physical channel, and to exclude or ignore packets on a different physical channel that happens to be using the same RF carrier in physical proximity.

There is no direct field within the Bluetooth packet structure that represents or contains information relating to physical links. This information is implied in the logical transport address (LT_ADDR) carried in the packet header.

Most packets include a packet header. The packet header is always present in packets transmitted on physical channels that support physical links, logical transports and logical links. The packet header carries the LT_ADDR, which is used by each receiving device to determine if the packet is addressed to the device and is used to route the packet internally.

The packet header also carries part of the link control (LC) protocol that is operated per logical transport (except for ACL and SCO transports that operate a shared LC protocol carried on either logical transport.)

The Enhanced Data Rate (EDR) packets have a guard time and synchronization sequence before the payload. This is a field used for physical layer change of modulation scheme.

The payload header is present in all packets on logical transports that support multiple logical links. The payload header includes a logical link identifier field used for routing the payload, and a field indicating the length of the payload. Some packet types also include a CRC after the packet payload that is used to detect most errors in received packets.

EDR packets have a trailer after the CRC.

The packet payload is used to transport the user data. The interpretation of this data is dependent on the logical transport and logical link identifiers. For ACL logical transports Link Manager Protocol (LMP) messages and L2CAP signals are transported in the packet payload, along with general user data from applications. For SCO and eSCO logical transports the payload contains the user data for the logical link.

3.3 PHYSICAL CHANNELS

The lowest architectural layer in the Bluetooth system is the physical channel. A number of types of physical channel are defined. All Bluetooth physical channels are characterized by an RF frequency combined with temporal parameters and restricted by spatial considerations. For the basic and adapted piconet physical channels frequency hopping is used to change frequency periodically to reduce the effects of interference and for regulatory reasons.

Two Bluetooth devices use a shared physical channel for communication. To achieve this their transceivers need to be tuned to the same RF frequency at the same time, and they need to be within a nominal range of each other.

Given that the number of RF carriers is limited and that many Bluetooth devices may be operating independently within the same spatial and temporal area there is a strong likelihood of two independent Bluetooth devices having their transceivers tuned to the same RF carrier, resulting in a physical channel collision. To mitigate the unwanted effects of this collision each transmission on a physical channel starts with an access code that is used as a correlation code by devices tuned to the physical channel. This channel access code is a property of the physical channel. The access code is always present at the start of every transmitted packet.

Four Bluetooth physical channels are defined. Each is optimized and used for a different purpose. Two of these physical channels (the basic piconet channel and adapted piconet channel) are used for communication between connected devices and are associated with a specific piconet. The remaining physical channels are used for discovering Bluetooth devices (the inquiry scan channel) and for connecting Bluetooth devices (the page scan channel.)

A Bluetooth device can only use one of these physical channels at any given time. In order to support multiple concurrent operations the device uses time-division multiplexing between the channels. In this way a Bluetooth device can appear to operate simultaneously in several piconets, as well as being discoverable and connectable.

Whenever a Bluetooth device is synchronized to the timing, frequency and access code of a physical channel it is said to be ‘connected’ to this channel (whether or not it is actively involved in communications over the channel.) The Bluetooth specification assumes that a device is only capable of connecting to one physical channel at any time. Advanced devices may capable of connecting simultaneously to more than one physical channel, but the specification does not assume that this is possible.

3.3.1 Basic piconet channel

3.3.1.1 Overview

The basic piconet channel is used for communication between connected devices during normal operation.

3.3.1.2 Characteristics

The basic piconet channel is characterized by a pseudo-random sequence hopping through the RF channels. The hopping sequence is unique for the piconet and is determined by the Bluetooth device address of the master. The phase in the hopping sequence is determined by the Bluetooth clock of the master. All Bluetooth devices participating in the piconet are time- and hop-synchronized to the channel.

The channel is divided into time slots where each slot corresponds to an RF hop frequency. Consecutive hops correspond to different RF hop frequencies. The time slots are numbered according to the Bluetooth clock of the piconet master. Packets are transmitted by Bluetooth devices participating in the piconet aligned to start at a slot boundary. Each packet starts with the channel's access code, which is derived from the Bluetooth device address of the piconet.

On the basic piconet channel the master controls access to the channel. The master starts its transmission in even-numbered time slots only. Packets transmitted by the master are aligned with the slot start and define the piconet timing. Packets transmitted by the master may occupy up to five time slots depending on the packet type.

Each master transmission is a packet carrying information on one of the logical transports. Slave devices may transmit on the physical channel in response. The characteristics of the response are defined by the logical transport that is addressed.

For example on the asynchronous connection-oriented logical transport the addressed slave device responds by transmitting a packet containing information for the same logical transport that is nominally aligned with the next (odd-numbered) slot start. Such a packet may occupy up to five time slots, depending on the packet type. On a broadcast logical transport no slaves are allowed to respond.

A special characteristic of the basic piconet physical channel is the use of some reserved slots to transmit a beacon train. The beacon train is only used if the piconet physical channel has parked slaves connected to it. In this situation the master transmits a packet in the reserved beacon train slots (these packets are used by the slave to resynchronize to the piconet physical channel.) The master may transmit packets from any logical transport onto these slots, providing there is a transmission starting in each of the slots. In the case where

there is information from the parked slave broadcast (PSB) logical transport to be transmitted then this is transmitted in the beacon train slots and takes priority over any other logical transport.

3.3.1.3 Topology

A basic piconet channel may be shared by any number of Bluetooth devices, limited only by the resources available on the piconet master device. Only one device is the piconet master, all others being piconet slaves. All communication is between the master and slave devices. There is no direct communication between slave devices on the piconet channel.

There is, however, a limitation on the number of logical transports that can be supported within a piconet. This means that although there is no theoretical limit to the number of Bluetooth devices that share a channel there is a limit to the number of these devices that can be actively involved in exchanging data with the master.

3.3.1.4 Supported layers

The basic piconet channel supports a number of physical links, logical transports, logical links and L2CAP channels used for general purpose communications.

3.3.2 Adapted piconet channel

3.3.2.1 Overview

The adapted piconet channel differs from the basic piconet channel in two ways. First the frequencies on which the slaves transmit are the same as the preceding master transmit frequency. In other words the frequency is not recomputed between master and subsequent slave packets. The second way in which the adapted piconet channel differs from the basic piconet channel is that the adapted type can be based on fewer than the full 79 frequencies. A number of frequencies may be excluded from the hopping pattern by being marked as “unused”. The remainder of the 79 frequencies are included. The two sequences are the same except that whenever the basic pseudo-random hopping sequence would have selected an unused frequency it is replaced with an alternative chosen from the used set.

Because the adapted piconet channel uses the same timing and access code as the basic piconet channel, the two channels are often coincident. This provides a deliberate benefit as it allows slaves in either the basic piconet channel or the adapted piconet channel to adjust their synchronization to the master.

The topology and supported layers of the adapted piconet physical channel are identical to the basic piconet physical channel.

3.3.3 Inquiry scan channel

3.3.3.1 Overview

In order for a device to be discovered an inquiry scan channel is used. A discoverable device listens for inquiry requests on its inquiry scan channel and then sends responses to these requests. In order for a device to discover other devices, it iterates (hops) through all possible inquiry scan channel frequencies in a pseudo-random fashion, sending an inquiry request on each frequency and listening for any response.

3.3.3.2 Characteristics

Inquiry scan channels follow a slower hopping pattern and use an access code to distinguish between occasional occupancy of the same radio frequency by two co-located devices using different physical channels.

The access code used on the inquiry scan channel is taken from a reserved set of inquiry access codes that are shared by all Bluetooth devices. One access code is used for general inquiries, and a number of additional access codes are reserved for limited inquiries. Each device has access to a number of different inquiry scan channels. As all of these channels share an identical hopping pattern, a device may concurrently occupy more than one inquiry scan channel if it is capable of concurrently correlating more than one access code.

A device using one of its inquiry scan channel remains passive until it receives an inquiry message on this channel from another Bluetooth device. This is identified by the appropriate inquiry access code. The inquiry scanning device will then follow the inquiry response procedure to return a response to the inquiring device.

In order for a device to discover other Bluetooth devices it uses the inquiry scan channel of these devices in order to send inquiry requests. As it has no prior knowledge of the devices to discover, it cannot know the exact characteristics of the inquiry scan channel.

The device takes advantage of the fact that inquiry scan channels have a reduced number of hop frequencies and a slower rate of hopping. The inquiring device transmits inquiry requests on each of the inquiry scan hop frequencies and listens for an inquiry response. This is done at a faster rate, allowing the inquiring device to cover all inquiry scan frequencies in a reasonably short time period.

3.3.3.3 Topology

Inquiring and discoverable devices use a simple exchange of packets to fulfil the inquiring function. The topology formed during this transaction is a simple and transient point-to-point connection.

3.3.3.4 Supported layers

During the exchange of packets between an inquiring and discoverable device it may be considered that a temporary physical link exists between these devices. However, the concept is quite irrelevant as it has no physical representation but is only implied by the brief transaction between the devices. No further architectural layers are considered to be supported.

3.3.4 Page scan channel

3.3.4.1 Overview

A connectable device (one that is prepared to accept connections) does so using an page scan channel. A connectable device listens for page requests on its page scan channel and enters into a sequence of exchanges with this device. In order for a device to connect to another device, it iterates (hops) through all page scan channel frequencies in a pseudo-random fashion, sending an page request on each frequency and listening for any response.

3.3.4.2 Characteristics

The page scan channel uses an access code derived from the scanning device's Bluetooth device address to identify communications on the channel. The page scan channel uses a slower hopping rate than the hop rate of the basic and adapted piconet channels. The hop selection algorithm uses the Bluetooth device clock of the scanning device as an input.

A device using its page scan channel remains passive until it receives a page request from another Bluetooth device. This is identified by the page scan channel access code. The two devices will then follow the page procedure to form a connection. Following a successful conclusion of the page procedure both devices switch to the basic piconet channel that is characterized by having the paging device as master.

In order for a device to connect to another Bluetooth device it uses the page scan channel of the target device in order to send page requests. If the paging device does not know the phase of the target device's page scan channel it therefore does not know the current hop frequency of the target device. The paging device transmits page requests on each of the page scan hop frequencies and listens for a page response. This is done at a faster hop rate, allowing

the paging device to cover all page scan frequencies in a reasonably short time period.

The paging device may have some knowledge of the target device's Bluetooth clock (indicated during a previous inquiry transaction between the two devices, or as a result of a previous involvement in a piconet with the device), in which case it is able to predict the phase of the target device's page scan channel. It may use this information to optimize the synchronization of the paging and page scanning process and speed up the formation of the connection.

3.3.4.3 Topology

Paging and connectable devices use a simple exchange of packets to fulfil the paging function. The topology formed during this transaction is a simple and transient point-to-point connection.

3.3.4.4 Supported layers

During the exchange of packets between an paging and connectable device it may be considered that a temporary physical link exists between these devices. However, the concept is quite irrelevant as it has no physical representation but is only implied by the brief transaction between the devices. No further architectural layers are considered to be supported.

3.4 PHYSICAL LINKS

A physical link represents a baseband connection between Bluetooth devices. A physical link is always associated with exactly one physical channel (although a physical channel may support more than one physical link.)

Within the Bluetooth system a physical link is a virtual concept that has no direct representation within the structure of a transmitted packet. The access code packet field, together with the clock and address of the master Bluetooth device are used to identify a physical channel. However there is no subsequent part of the packet that directly identifies the physical link. Instead the physical link may be identified by association with the logical transport, as each logical transport is only received on one physical link.

Some physical link types have properties that may be modified. An example of this is the transmit power for the link. Other physical link types have no such properties. In the case of physical links with modifiable properties the LM protocol is used to adapt these properties. As the LM protocol is supported at a higher layer (by a logical link) the appropriate physical link is identified by implication from the logical link that transports the LM signalling.

In the situation where a transmission is broadcasted over a number of different physical links, then the transmission parameters are selected to be suitable for all of the physical links.

3.4.1 Links supported by the basic and adapted piconet physical channel

The basic and adapted piconet physical channels support a physical link which may be active or parked. The physical link is a point-to-point link between the master and a slave. It is always present when the slave is synchronized in the piconet.

3.4.1.1 Active physical link

The physical link between a master and a slave device is active if a default ACL logical transport exists between the devices. Active physical links have no direct identification of their own, but are identified by association with the default ACL logical transport ID with which there is a one-to-one correspondence.

An active physical link has the associated properties of radio transmit power in each direction. Transmissions from slave devices are always directed over the active physical link to the master, and use the transmit power that is a property of this link in the slave to master direction. Transmissions from the master may be directed over a single active physical link (to a specific slave) or over a number of physical links (to a group of slaves in the piconet.) In the case of point-to-point transmissions the master uses the appropriate transmit power for the physical link in question. (In the case of point-to-multipoint transmissions the master uses a transmit power appropriate for the set of devices addressed.)

Active physical links may be placed into Hold or Sniff mode. The effect of these modes is to modify the periods when the physical link is active and may carry traffic. Logical transports that have defined scheduling characteristics are not affected by these modes and continue according to their pre-defined scheduling behavior. The default ACL logical transport and other links with undefined scheduling characteristics are subject to the mode of the active physical link.

3.4.1.2 Parked physical link

The physical link between a master and a slave device is parked when the slave remains synchronized in the piconet, but has no default ACL logical transport. Such a slave is also said to be parked. A beacon train is used to provide regular synchronization to all parked slaves connected to the piconet physical channel. A parked slave broadcast (PSB) logical transport is used to allow communication of a subset of LMP signalling and broadcast L2CAP to parked slaves. The PSB logical transport is closely associated with the beacon train.

A slave is parked (its active link is changed to a parked link) using the park procedure. The master is not allowed to park a slave that has any user created logical transport supported by the physical link. These logical transports are first removed, and any L2CAP channels that are built on these logical transports are also removed. The broadcast logical transport and default ACL logical transports are not considered as user created and are not explicitly removed. When the active link is replaced with a parked link the default ACL

logical transport is implicitly removed. The supported logical links and L2CAP channels remain in existence, but become suspended. It is not possible to use these links and L2CAP channels to transport signalling or data while the active link is absent.

A parked slave may become active using the unpark procedure. This procedure is requested by the slave at an access window and initiated by the master. Following the unpark procedure the parked physical link is changed to an active physical link and the default ACL logical transport is re-created. L2CAP channels that were suspended during the most recent park procedure are associated with the new default ACL logical transport and become active again.

Parked links do not support radio power control, as there is no feedback path from parked slaves to the piconet master that can be used to signal received signal strength at the slave or for the master to measure received signal strength from the slave. Transmissions are carried out at nominal power on parked links.

Parked links use the same physical channel as their associated active link. If a master manages a piconet that contains parked slaves using the basic piconet physical channel and also parked slaves using the adapted piconet physical channel then it must create a parked slave broadcast logical transport (and associated transport) for each of these physical channels.

A parked slave may use the inactive periods of the parked slave broadcast logical transport to save power, or it may carry out activities on other physical channels unrelated to the piconet within which it is parked.

3.4.2 Links supported by the scanning physical channels

In the case of the inquiry scan and page scan channels the physical link exists for a relatively short time and cannot be controlled or modified in any way. These types of physical link are not further elaborated.

3.5 LOGICAL LINKS AND LOGICAL TRANSPORTS

A variety of logical links are available to support different application data transport requirements. Each logical link is associated with a logical transport, which has a number of characteristics. These characteristics include flow control, acknowledgement/repeat mechanisms, sequence numbering and scheduling behavior. Logical transports are able to carry different types of logical links (depending on the type of the logical transport). In the case of some of the Bluetooth 1.1 logical links these are multiplexed onto the same logical transport. Logical transports may be carried by active physical links on either the basic or the adapted piconet physical channel.

Logical transport identification and real-time (link control) signalling are carried in the packet header, and for some logical links identification is carried in the

payload header. Control signalling that does not require single slot response times is carried out using the LMP protocol.

Table 3.1 on page 42 lists all of the logical transport types, the supported logical link types, which type of physical links and physical channels can support them, and a brief description of the purpose of the logical transport.

Logical transport	Links supported	Supported by	Overview
Asynchronous Connection-Oriented (ACL[1])	Control (LMP) ACL-C User (L2CAP) ACL-U	Active physical link, basic or adapted physical channel	Reliable or time-bounded, bi-directional, point-to-point.
Synchronous Connection-Oriented (SCO)	Stream (unframed) SCO-S	Active physical link, basic or adapted physical channel	Bi-directional, symmetric, point-to-point, AV channels. Used for 64Kb/s constant rate data.
Extended Synchronous Connection-Oriented (eSCO)	Stream (unframed) eSCO-S	Active physical link, basic or adapted physical channel	Bi-directional, symmetric or asymmetric, point-to-point, general regular data, limited retransmission. Used for constant rate data synchronized to the master Bluetooth clock.
Active slave broadcast (ASB)	User (L2CAP) ASB-U	Active physical link, basic or adapted physical channel.	Unreliable, uni-directional broadcast to any devices synchronized with the physical channel. Used for broadcast L2CAP groups.
Parked slave broadcast (PSB)	Control (LMP) PSB-C, User (L2CAP) PSB-U	Parked physical link, basic or adapted physical channel.	Unreliable, uni-directional broadcast to all piconet devices. Used for LMP and L2CAP traffic to parked devices, and for access requests from parked devices.

Table 3.1: Logical transport types.

1. It is clear that the most obvious abbreviation for Asynchronous Connection-Oriented logical transport is ACO. However, this acronym has an alternative meaning from the Bluetooth 1.1 specification. To avoid confusion between two possible meanings for ACO the decision was made to retain the ACL abbreviation for the Asynchronous Connection-oriented Logical transport.

The names given to the logical links and logical transports reflect some of the names used in Bluetooth 1.1, in order to provide some degree of familiarity and continuation. However these names do not reflect a consistent scheme, which is outlined below.

The classification of each link type follows from a selection procedure within three categories.

3.5.1 Casting

The first category is that of casting. This may be either unicast or broadcast. There are no multicast links defined in Bluetooth 1.2,

- *Unicast links.* Unicast links exist between exactly two endpoints. Traffic may be sent in either direction on unicast links. All unicast links are connection-oriented, meaning that a connection procedure takes place before the link may be used. In the case of the default ACL links, the connection procedure is an implicit step within the general paging procedure used to form ad-hoc piconets.
- *Broadcast links.* Broadcast links exist between one source device and zero or more receiver devices. Traffic is unidirectional, i e only sent from the source devices to the receiver devices. Broadcast links are connectionless, meaning that there is no procedure to create these links, and data may be sent over them at any time. Broadcast links are unreliable, and there is no guarantee that the data will be received.

3.5.2 Scheduling and acknowledgement scheme

The second category relates to the scheduling and acknowledgement scheme of the link, and implies the type of traffic that is supported by the link. These are synchronous, isochronous or asynchronous. There are no specific isochronous links defined in Bluetooth 1.2, though the default ACL link can be configured to operate in this fashion.

- *Synchronous links.* Synchronous links provide a method of associating the Bluetooth piconet clock with the transported data. This is achieved by reserving regular slots on the physical channel, and transmitting fixed size packets at these regular intervals. Such links are suitable for constant rate isochronous data.
- *Asynchronous links.* Asynchronous links provide a method for transporting data that has no time-based characteristics. The data is normally expected to be retransmitted until successfully received, and each data entity can be processed at any time after receipt, without reference to the time of receipt of any previous or successive entity in the stream (providing the ordering of data entities is preserved.)
- *Isochronous links.* Isochronous links provide a method for transporting data that has time-based characteristics. The data may be retransmitted until received or expired. The data rate on the link need not be constant (this being the main difference from synchronous links.)

3.5.3 Class of data

The final category is related to the class of data that is carried by the link. This is either control (LMP) data or user data. The user data category is sub-divided into L2CAP (or framed) data and stream (or unframed) data.

- *Control links.* Control links are only used for transporting LMP messages between two link managers. These links are invisible above the baseband layer, and cannot be directly instantiated, configured or released by applications, other than by the use of the connection and disconnection services that have this effect implicitly. Control links are always multiplexed with an equivalent L2CAP link onto an ACL logical transport. Subject to the rules defining the ARQ scheme, the control link traffic always takes priority over the L2CAP link traffic.
- *L2CAP links.* L2CAP links are used to transport L2CAP PDUs, which may carry the L2CAP signalling channel (on the default ACL-U logical link only) or framed user data submitted to user-instantiated L2CAP channels. L2CAP frames submitted to the baseband may be larger than the available baseband packets. A link control protocol embedded within the LLID field preserves the frame-start and frame-continuation semantics when the frame is transmitted in a number of fragments to the receiver.
- *Stream links.* Stream links are used to transport user data that has no inherent framing that should be preserved when delivering the data. Lost data may be replaced by padding at the receiver.

3.5.4 Asynchronous connection-oriented (ACL)

The asynchronous connection-oriented (ACL) logical transport is used to carry LMP and L2CAP control signalling and best effort asynchronous user data. The ACL logical transport uses a simple 1-bit ARQN/SEQN scheme to provide simple channel reliability. Every active slave device within a piconet has one ACL logical transport to the piconet master, known as the default ACL.

The default ACL is created between the master and the slave when a device joins a piconet (connects to the basic piconet physical channel). This default ACL is assigned a logical transport address (LT_ADDR) by the piconet master. This LT_ADDR is also used to identify the active physical link when required (or as a piconet active member identifier, effectively for the same purpose.)

The LT_ADDR for the default ACL is reused for synchronous connection-oriented logical transports between the same master and slave. (This is for reasons of compatibility with earlier Bluetooth specifications.) Thus the LT_ADDR is not sufficient on its own to identify the default ACL. However the packet types used on the ACL are different from those used on the synchronous connection-oriented logical transport. Therefore, the ACL logical transport can be identified by the LT_ADDR field in the packet header in combination with the packet type field.

The default ACL may be used for isochronous data transport by configuring it to automatically flush packets after the packets have expired.

If the default ACL is removed from the active physical link then all other logical transports that exist between the master and the slave are also removed. In the case of unexpected loss of synchronization to the piconet physical channel the physical link and all logical transports and logical links cease to exist at the time that this synchronization loss is detected.

A device may remove its default ACL (and by implication its active physical link) but remain synchronized to the piconet. This procedure is known as parking, and a device that is synchronized to the piconet, but has no active physical link is parked within that piconet.

When the device transitions to the parked state the default ACL logical links that are transported on the default ACL logical transport remain in existence, but become suspended. No data may be transferred across a suspended logical link. When the device transitions from the parked state back into active state, a new default ACL logical transport is created (it may have a different LT_ADDR from the previous one) and the suspended logical links are attached to this default ACL and become active once again.

3.5.5 Synchronous connection-oriented (SCO)

The synchronous connection-oriented (SCO) logical transport is a symmetric, point-to-point channel between the master and a specific slave. The SCO logical transport reserves slots on the physical channel and can therefore be considered as a circuit-switched connection between the master and the slave. SCO logical transports carry 64 kb/s of information synchronized with the piconet clock. Typically this information is an encoded voice stream. Three different SCO configurations exist, offering a balance between robustness, delay and bandwidth consumption.

Each SCO-S logical link is supported by a single SCO logical transport, which is assigned the same LT_ADDR as the default ACL logical transport between the devices. Therefore the LT_ADDR field is not sufficient to identify the destination of a received packet. Because the SCO links use reserved slots, a device uses a combination of the LT_ADDR, the slot numbers (a property of the physical channel) and the packet type to identify transmissions on the SCO link.

The reuse of the default ACL's LT_ADDR for SCO logical transports is due to legacy behavior from the Bluetooth 1.1 specification. In this earlier version of Bluetooth the LT_ADDR (then known as the active member address) was used to identify the piconet member associated with each transmission. This was not easily extensible for enabling more logical links, and so the purpose of this field was redefined for the new features. Some Bluetooth 1.1 features, however, do not cleanly fit into the more formally described architecture.

Although slots are reserved for the SCO, it is permissible to use a reserved slot for traffic from another channel that has a higher priority. This may be required as a result of QoS commitments, or to send LMP signalling on the default ACL when the physical channel bandwidth is fully occupied by SCOs. As SCOs carry different packet types to ACLs, the packet type is used to identify SCO traffic (in addition to the slot number and LT_ADDR.)

There are no further architectural layers defined by the Bluetooth core specification that are transported over an SCO link. A number of standard formats are defined for the 64 kb/s stream that is transported, or an unformatted stream is allowed where the application is responsible for interpreting the encoding of the stream.

3.5.6 Extended synchronous connection-oriented (eSCO)

The extended synchronous connection-oriented (eSCO) logical transport is a symmetric or asymmetric, point-to-point link between the master and a specific slave. The eSCO reserves slots on the physical channel and can therefore be considered as a circuit-switched connection between the master and the slave. eSCO links offer a number of extensions over the standard SCO links, in that they support a more flexible combination of packet types and selectable data contents in the packets and selectable slot periods, allowing a range of synchronous bit rates to be supported.

eSCO links also can offer limited retransmission of packets (unlike SCO links where there is no retransmission.) If these retransmissions are required they take place in the slots that follow the reserved slots, otherwise the slots may be used for other traffic.

Each eSCO-S logical link is supported by a single eSCO logical transport, identified by a LT_ADDR that is unique within the piconet for the duration of the eSCO. eSCO-S links are created using LM signalling and follow scheduling rules similar to SCO-S links.

There are no further architectural layers defined by the Bluetooth core specification that are transported over an eSCO-S link. Instead applications may use the data stream for whatever purpose they require, subject to the transport characteristics of the stream being suitable for the data being transported.

3.5.7 Active slave broadcast (ASB)

The active slave broadcast logical transport is used to transport L2CAP user traffic to all devices in the piconet that are currently connected to the physical channel that is used by the ASB. There is no acknowledgement protocol and the traffic is uni-directional from the piconet master to the slaves. The ASB channel may be used for L2CAP group traffic (a legacy of the 1.1 specification), and is never used for L2CAP connection-oriented channels, L2CAP control signalling or LMP control signalling.

The ASB logical transport is inherently unreliable because of the lack of acknowledgement. To improve the reliability, each packet is transmitted a number of times. An identical sequence number is used to assist with filtering retransmissions at the slave device.

The ASB logical transport is identified by a reserved LT_ADDR. (The reserved LT_ADDR address is also used by the PSB logical transport.) An active slave will receive traffic on both logical transports, and cannot readily distinguish between them. As the ASB logical transport does not carry LMP traffic an active slave can ignore packets received over the LMP logical link on the ASB logical transport. However L2CAP traffic transmitted over the PSB logical transport is also received by active slaves on the ASB logical transport and cannot be distinguished from L2CAP traffic sent on the ASB transport.

An ASB is implicitly created whenever a piconet exists, and there is always one ASB associated with each of the basic and adapted piconet physical channels that exist within the piconet. Because the basic and adapted piconet physical channels are mostly coincident a slave device cannot distinguish which of the ASB channels is being used to transmit the packets. This adds to the general unreliability of the ASB channel. (Although it is, perhaps, no more unreliable than general missed packets.)

A master device may decide to use only one of its two possible ASBs (when it has both a basic and adapted piconet physical channel), as with sufficient retransmissions it is possible to address both groups of slaves on the same ASB channel.

The ASB channel is never used to carry LMP or L2CAP control signals.

3.5.8 Parked slave broadcast (PSB)

The parked slave broadcast logical transport is used for communications between the master and slaves that are parked (have given up their default ACL logical transport.) The parked slave broadcast link is the only logical transport that exists between the piconet master and parked slaves.

The PSB logical transport is more complex than the other logical transports as it consists of a number of phases, each having a different purpose. These phases are the control information phase (used to carry the LMP logical link), the user information phase (used to carry the L2CAP logical link), and the access phase (carrying baseband signalling.) The control information and broadcast information phases are usually mutually exclusive as only one of them can be supported in a single beacon interval. (Even if there is no control or user information phase, the master is still required to transmit a packet in the beacon slots so that the parked slaves can resynchronize.) The access phase is normally present unless cancelled in a control information message.

The control information phase is used for the master to send information to the parked slaves containing modifications to the PSB transport attributes, modifi-

cations to the beacon train attributes, or a request for a parked slave to become active in the piconet (known as unparking). This control information is carried in LMP messages on the LMP logical link. (The control information phase is also present in the case of a user information phase where the user information requires more than one baseband packet.)

Packets in the control information phase are always transmitted in the physical channel beacon train slots, and cannot be transmitted on any other slots. The control information occupies a single DM1 packet and is repeated in every beacon train slot within a single beacon interval. (If there is no control information then there may be a user information phase that uses the beacon slots. If neither phase is used then the beacon slots are used for other logical transport traffic or for NULL packets.)

The user information phase is used for the master to send L2CAP packets that are destined for all piconet slaves. User information may occupy one or more baseband packets. If the user information occupies a single packet then the user information packet is repeated in each of the piconet channel beacon train slots.

If the user information occupies more than one baseband packet then it is transmitted in slots after the beacon train (the broadcast scan window) and the beacon slots are used to transmit a control information phase message that contains the timing attributes of this broadcast scan window. This is required so that the parked slaves remain connected to the piconet physical channel to receive the user information.

The access phase is normally present unless temporarily cancelled by a control message carried in the control information broadcast phase. The access window consists of a sequence of slots that follow the beacon train. In order for a parked slave to become active in the piconet, it may send such an access request to the piconet master during the access window. Each parked slave is allocated an access request address (not necessarily unique) that controls when during the access window the slave requests access.

The PSB logical transport is identified by the reserved LT_ADDR of 0. This reserved LT_ADDR address is also used by the ASB logical transport. Parked slaves are not normally confused by the duplicated use of the LT_ADDR as they are only connected to the piconet physical channel during the time that the PSB transport is being used.

3.5.9 Logical links

Some logical transports are capable of supporting different logical links, either concurrently multiplexed, or one of the choice. Within such logical transports, the logical link is identified by the logical link identifier (LLID) bits in the payload header of baseband packets that carry a data payload. The logical links distinguish between a limited set of core protocols that are able to transmit and receive data on the logical transports. Not all of the logical transports are able to carry all of the logical links (the supported mapping is shown in Figure 3.2 on

page 28.) In particular the SCO and eSCO logical transports are only able to carry constant data rate streams, and these are uniquely identified by the LT_ADDR. Such logical transports only use packets that do not contain a payload header, as their length is known in advance, and no LLID is necessary.

3.5.10 ACL Control Logical Link (ACL-C)

The ACL Control Logical Link (ACL-C) is used to carry LMP signalling between devices in the piconet. The control link is only carried on the default ACL logical transport and on the PSB logical transport (in the control information phase). The ACL-C link is always given priority over the ACL-U (see below) link when carried on the same logical transport.

3.5.11 User Asynchronous/Isochronous Logical Link (ACL-U)

The user asynchronous/isochronous logical link (ACL-U) is used to carry all asynchronous and isochronous framed user data. The ACL-U link is carried on all but the synchronous logical transports. Packets on the ACL-U link are identified by one of two reserved LLID values. One value is used to indicate whether the baseband packet contains the start of an L2CAP frame and the other indicates a continuation of a previous frame. This ensures correct synchronization of the L2CAP reassembler following flushed packets. The use of this technique removes the need for a more complex L2CAP header in every baseband packet (the header is only required in the L2CAP start packets), but adds the requirement that a complete L2CAP frame shall be transmitted before a new one is transmitted. (An exception to this rule being the ability to flush a partially transmitted L2CAP frame in favour of another L2CAP frame.)

3.5.12 User Synchronous/Extended Synchronous Logical Links (SCO-S/eSCO-S)

Synchronous (SCO-S) and extended synchronous (eSCO-S) logical links are used to support isochronous data delivered in a stream without framing. These links are associated with a single logical transport, where data is delivered in constant sized units at a constant rate. There is no LLID within the packets on these transports, as only a single logical link can be supported, and the packet length and scheduling period are pre-defined and remain fixed during the lifetime of the link.

Variable rate isochronous data cannot be carried by the SCO-S or eSCO-S logical links. In this case the data must be carried on ACL-U logical links, which use packets with a payload header. Bluetooth has some limitations when supporting variable-rate isochronous data concurrently with reliable user data.

3.6 L2CAP CHANNELS

L2CAP provides a multiplexing role allowing many different applications to share the resources of an ACL-U logical link between two devices. Applications and service protocols interface with L2CAP using a channel-oriented interface to create connections to equivalent entities on other devices.

L2CAP channel endpoints are identified to their clients by a Channel Identifier (CID). This is assigned by L2CAP, and each L2CAP channel endpoint on any device has a different CID.

L2CAP channels may be configured to provide an appropriate Quality of Service (QoS) to the application. L2CAP maps the channel onto the ACL-U logical link.

L2CAP supports channels that are connection-oriented and others that are group-oriented. Group-oriented channels may be mapped onto the ASB-U logical link, or implemented as iterated transmission to each member in turn over an ACL-U logical link.

Apart from the creation, configuration and dismantling of channels, the main role of L2CAP is to multiplex service data units (SDUs) from the channel clients onto the ACL-U logical links, and to carry out a simple level of scheduling, selecting SDUs according to relative priority.

L2CAP can provide a per channel flow control with the peer L2CAP layer. This option is selected by the application when the channel is established. L2CAP can also provide enhanced error detection and retransmission to (a) reduce the probability of undetected errors being passed to the application and (b) recover from loss of portions of the user data when the Baseband layer performs a flush on the ACL-U logical link.

In the case where an HCI is present, the L2CAP is also required to segment L2CAP SDUs into fragments that will fit into the baseband buffers, and also to operate a token based flow control procedure over the HCI, submitting fragments to the baseband only when allowed to do so. This may affect the scheduling algorithm.

4 COMMUNICATION TOPOLOGY

4.1 PICONET TOPOLOGY

Any time a Bluetooth link is formed it is within the context of a piconet. A piconet consists of two or more devices that occupy the same physical channel (which means that they are synchronized to a common clock and hopping sequence.) The common (piconet) clock is identical to the Bluetooth clock of one of the devices in the piconet, known as the master of the piconet, and the hopping sequence is derived from the master's clock and the master's Bluetooth device address. All other synchronized devices are referred to as slaves in the piconet. The terms master and slave are only used when describing these roles in a piconet.

Within a common location a number of independent piconets may exist. Each piconet has a different physical channel (that is a different master device and an independent piconet clock and hopping sequence.)

A Bluetooth device may participate concurrently in two or more piconets. It does this on a time-division multiplexing basis. A Bluetooth device can never be a master of more than one piconet. (Since the piconet is defined by synchronization to the master's Bluetooth clock it is impossible to be the master of two or more piconets.) A Bluetooth device may be a slave in many independent piconets.

A Bluetooth device that is a member of two or more piconets is said to be involved in a scatternet. Involvement in a scatternet does not necessarily imply any network routing capability or function in the Bluetooth device. The Bluetooth core protocols do not, and are not intended to offer such functionality, which is the responsibility of higher level protocols and is outside the scope of the Bluetooth core specification.

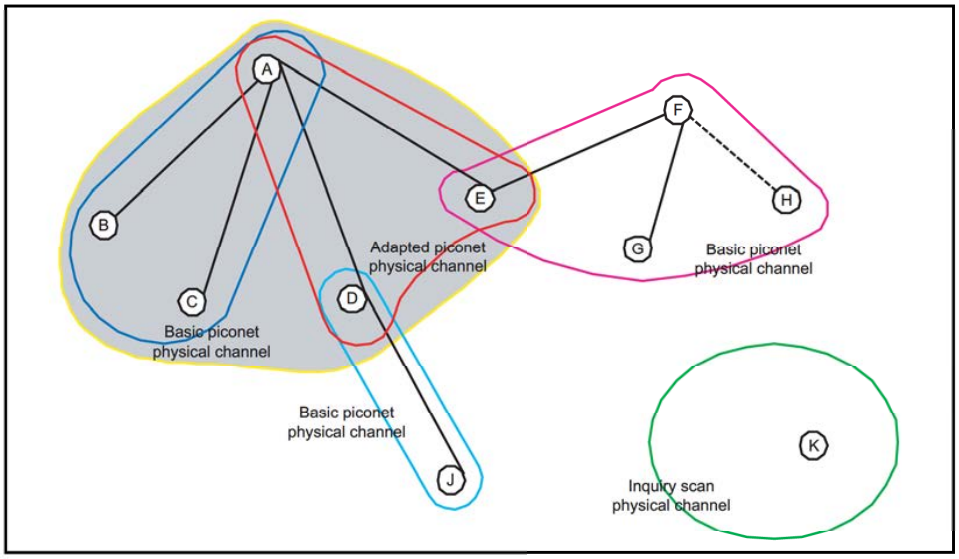


Figure 4.1: Example Bluetooth topology

In Figure 4.1 on page 52 an example topology is shown that demonstrates a number of the architectural features described below. Device A is a master in a piconet (represented by the shaded area, and known as piconet A) with devices B, C, D and E as slaves. Two other piconets are shown: a) one piconet with device F as master (known as piconet F) and devices E, G and H as slaves and b) one piconet with device D as master (known as piconet D) and device J as slave.

In piconet A there are two physical channels. Devices B and C are using the basic piconet physical channel (represented by the blue enclosure) as they do not support adaptive frequency hopping. Devices D and E are capable of supporting adaptive frequency hopping, and are using the adapted piconet physical channel (represented by the red enclosure.) Device A is capable of adaptive frequency hopping, and operates in a TDM basis on both physical channels according to which slave is being addressed.

Piconet D and piconet F are both using only a basic piconet physical channel (represented by the cyan and magenta enclosures respectively.) In the case of piconet D this is because device J does not support the adaptive hopping mode. Although device D supports adaptive hopping it cannot use it in this piconet. In piconet F device F does not support adaptive hopping, and therefore it cannot be used in this piconet.

Device K is shown in the same locality as the other devices. It is not currently a member of a piconet, but has services that it offers to other Bluetooth devices. It is currently listening on its inquiry scan physical channel (represented by the green enclosure), awaiting an inquiry request from another device.