(with Status=0x00) returned by the Host Controller after the Reject_Connection_Request command has been issued. The reason code issued in the Reason parameter of the Reject_Connection_Request command will also be sent over the air, so that it is returned in a Connection Complete event on the initiating side. Before this, the initiating side has issued a Create_Connection command or Add_SCO_Connection command, and has received a Command Status event (with Status=0x00).

6.16 HOST TIMEOUT (0X10)

Note: this error code is used to indicate a reason for rejecting an incoming connection. It is therefore called reason code in the following description.

Assume that a Connection Request event has been received by the Host and that the Host does not issue the Accept_Connection_Request or Reject_Connection_Request command before the connection accept timer expires (the connection accept timeout is set using Write_Connection_Accept_Timeout). In this case, the 'Host Timeout' reason code will be sent by the Host Controller in the Status parameter of a Connection Complete event. The reason code will also be sent over the air, so that it is returned in a Connection Complete event on the initiating side. The initiating side has before this issued a Create_Connection or Add_SCO_Connection command and has received a Command Status event (with Status=0x00).

6.17 UNSUPPORTED FEATURE OR PARAMETER VALUE (0X11)

The 'Unsupported Feature or Parameter Value' error code is returned by the Host Controller in the Status parameter in an event when the Host Controller has received a command where one or more parameters have values that are not supported by the hardware (the parameters are, however, within the allowed parameter range specified in this document). If the issued command is a command for which a Command Complete event should be returned, the event containing the error code is a Command Complete event. Otherwise, the event containing the error code is a Command Status event or the event associated with the issued command (following a Command Status event with Status=0x00) depending on the implementation.

6.18 INVALID HCI COMMAND PARAMETERS (0X12)

The 'Invalid HCI Command Parameters' error code is returned by the Host Controller in the Status parameter of an event when the total parameter length (or the value of one or more parameters in a received command) does not conform to what is specified in this document.

The error code can also be returned if a parameter value is currently not allowed although it is inside the allowed range for the parameter. One case is when a command requires a Connection Handle for an ACL connection but the

Host has given a Connection Handle for an SCO connection as a parameter instead. Another case is when a link key, a PIN code or a reply to an incoming connection has been requested by the Host Controller by using an event but the Host replies using a response command with a BD_ADDR for which no request has been made.

If the issued command is a command for which a Command Complete event should be returned, the event containing the error code is a Command Complete event. Otherwise, the event containing the error code is a Command Status event or the event associated with the issued command (following a Command Status event with Status=0x00), depending on the implementation.

6.19 OTHER END TERMINATED CONNECTION: ... (0X13-0X15)

Note: these error codes are used to indicate a reason for disconnecting a connection. They are therefore called reason codes in the following description.

When the Host issues the Disconnect command, one of these reason codes is used as value for the reason parameter. The 'Connection Terminated By Local Host' reason code will then be returned in the Reason parameter of the Disconnection Complete event that will follow the Command Status event (with Status=0x00) that is returned by the Host Controller after the Disconnect command has been issued. The reason code issued in the Reason parameter of the Disconnect command will also be sent over the air, so that it is returned in the Reason parameter of a Disconnection Complete event on the remote side.

6.20 CONNECTION TERMINATED BY LOCAL HOST (0X16)

See description in 6.19. This error code is called a reason code, since it is returned in the Reason parameter of a Disconnection Complete event.

6.21 REPEATED ATTEMPTS (0X17)

The 'Repeated Attempts' error code is returned by the Host Controller in the Status parameter in a Connection Complete event or Authentication Complete event when a device does not allow authentication or pairing because too little time has elapsed since an unsuccessful authentication or pairing attempt. See "Link Manager Protocol" on page 185 for a description of how repeated attempts work.

6.22 PAIRING NOT ALLOWED (0X18)

The 'Pairing Not Allowed' error code is returned by the Host Controller in the Status parameter in a Connection Complete event or Authentication Complete event when a device for some reason does not allow pairing. An example may be a PSTN adapter that only allows pairing during a certain time window after a button has been pressed on the adapter.

6.23 UNSUPPORTED REMOTE FEATURE (0X1A)

The 'Unsupported Remote Feature' error code is returned by the Host Controller in the Status parameter of the event associated with the issued command when a remote device that has been specified in the command parameters does not support the feature associated with the issued command. The 'Unsupported Remote Feature' error code can also be used as a value for the Reason parameter in the Disconnect command (as a reason code). The error code will then be sent over the air so that it is returned in the Reason parameter of a Disconnection Complete event on the remote side. In the Disconnection Complete event following a Command Status event (where Status=0x00) on the local side on which the Disconnect command has been issued, the Reason parameter will however contain the reason code 'Connection Terminated By Local Host'. (The 'Unsupported Remote Feature' error code is called 'Unsupported LMP Feature' in the LMP specification, see "Link Manager Protocol" on page 185.)

6.24 UNSPECIFIED ERROR (0X1F)

The 'Unspecified error' error code is used when no other error code specified in this document is appropriate to use.

6.25 UNSUPPORTED LMP PARAMETER VALUE (0X20)

The 'Unsupported LMP Parameter Value' error code is returned by the Host Controller in the Status parameter of the event associated with the issued command when a remote device that has been specified in the command parameters sent back an LMP message containing the LMP error code 0x20, 'Unsupported parameter values' (see "Link Manager Protocol" on page 185).

6.26 ROLE CHANGE NOT ALLOWED (0X21)

The 'Role Change Not Allowed' error code is returned by the Host Controller in the Status parameter in a Connection Complete event or Role Change event when role change is not allowed. If the local Host issues the Switch_Role command and the remote device rejects the role change, the error code will be returned in a Role Change event. If a connection fails because a device accepts an incoming ACL connection with a request for role change and the role change is rejected by the initiating device, the error code will be returned in a Connection Complete event on both sides.

6.27 LMP RESPONSE TIMEOUT (0X22)

The 'LMP Response Timeout' error code is returned by the Host Controller in the Status parameter in a Command Complete event or an event associated with the issued command following a Command Status event with Status=0x00, when the remote device does not respond to the LMP PDUs from the local device as a result of the issued command within LMP response time-out. (See "Link Manager Protocol" on page 185)

6.28 LMP ERROR TRANSACTION COLLISION (0X23)

The 'LMP Error Transaction Collision' error code is returned by the Host Controller in the Status parameter of the event associated with the issued command when a remote device that has been specified in the command parameters sends back an LMP message containing the LMP error code 0x23, "LMP Error Transaction Collision" (see "Link Manager Protocol" on page 185).

6.29 LMP PDU NOT ALLOWED (0X24)

The 'LMP PDU Not Allowed' error code is returned by the Host Controller in the Status parameter of the event associated with the issued command when a remote device that has been specified in the command parameters sends back an LMP message containing the LMP error code 0x24, "PDU Not Allowed" (see "Link Manager Protocol" on page 185).

7 LIST OF ACRONYMS AND ABBREVIATIONS

Acronym or abbreviation	Complete name
ACL	Asynchronous Connection Less
BD_ADDR	Bluetooth Device Address
DH	Data High rate
DIAC	Dedicated Inquiry Access Code
DM	Data Medium rate
DUT	Device Under Test
DV	Data Voice
GIAC	General Inquiry Access Code
HCI	Host Controller Interface
L2CAP	Logical Link Control and Adaptation Protocol
L_CH	Logical Channel
LAP	Lower Address Part
LC	Link Controller
LM	Link Manager
LMP	Link Manager Protocol
OCF	Opcode Command Field
OGF	OpCode Group Field
RF	Radio Frequency
RSSI	Received Signal Strength Indication
SCO	Synchronous Connection Oriented
TBD	To Be Defined
UA	User Asynchronous
UI	User Isochronous
US	User Synchronous
USB	Universal Serial Bus

Table 7.1: List of Acronyms and Abbreviations

8 LIST OF FIGURES

Bluetooth.

9 LIST OF TABLES

Part H:2

HCI USB TRANSPORT LAYER

An addendum to the HCI document

This document describes the USB transport layer (between a host and the host controller). HCI commands flow through this layer, but the layer does not decode the commands.

Bluetooth.

CONTENTS

1 OVERVIEW

This document discusses the requirements of the Universal Serial Bus (USB) interface for Bluetooth hardware. Readers should be familiar with USB, USB design issues, Advanced Configuration Power Interface (ACPI), the overall Bluetooth architecture, and the basics of the radio interface.

The reader should also be familiar with the Bluetooth Host Controller Interface.

Referring to Figure 1.1 below, notice that this document discusses the implementation details of the two-way arrow labelled 'USB Function'.

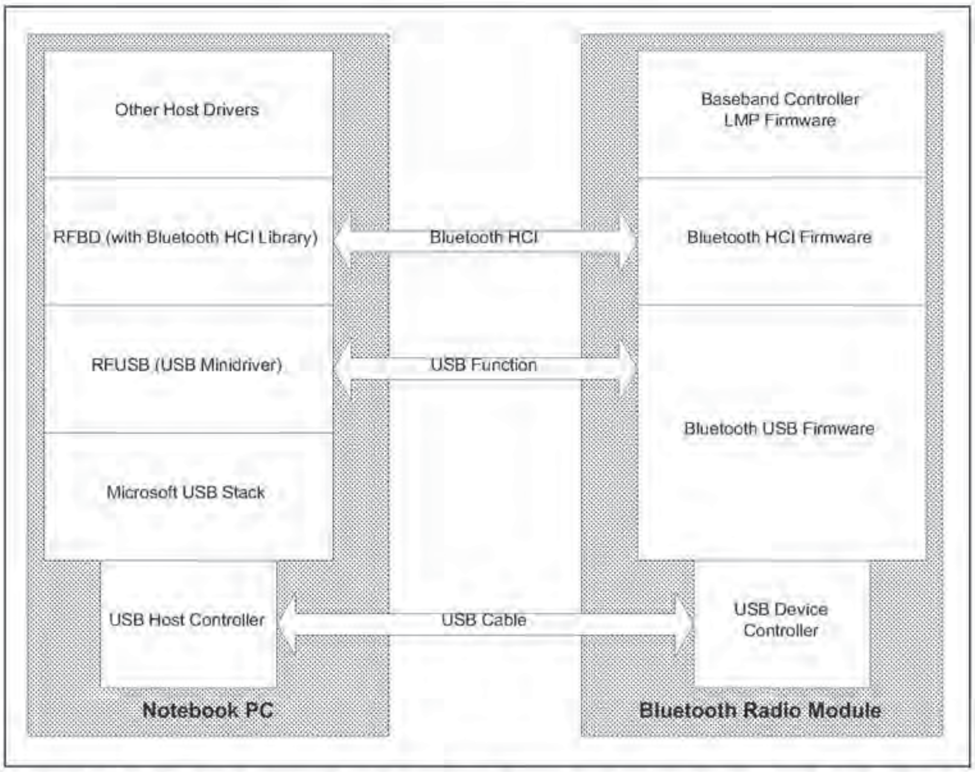


Figure 1.1: The Figure illustrates the relationship between the host and the Bluetooth Radio Module

The USB hardware can be embodied in one of two ways:

1. As a USB dongle, and
2. Integrated onto the motherboard of a notebook PC.

Finally, for an overview of the connection that is established between two Bluetooth devices, reference Figure 1.2, below.

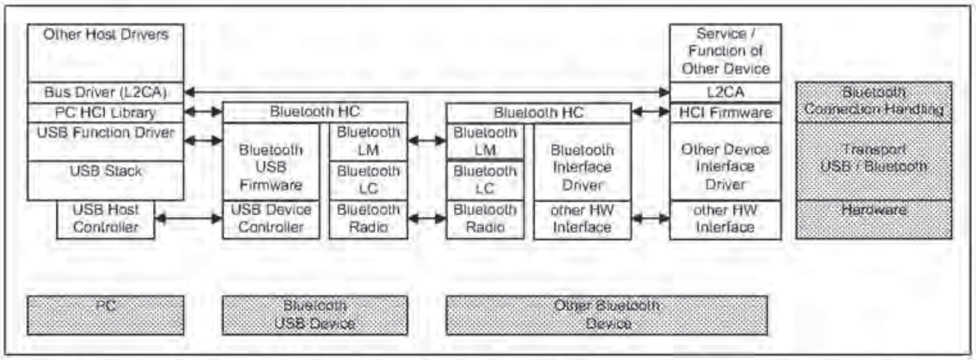


Figure 1.2: The figure illustrates the flow of data from one Bluetooth device to another

2 USB ENDPOINT EXPECTATIONS

This section outlines specific USB endpoints that are required in order to function properly with the host. This section assumes a basic familiarity with USB. The endpoint numbers (labelled 'Suggested Endpoint Address' below) may be dynamically recognized upon driver initialization – this depends on the implementation.

2.1 DESCRIPTOR OVERVIEW

The USB device is expected to be a high-speed device.

The firmware configuration consists of two interfaces. The first interface (interface zero) has no alternate settings and contains the bulk and interrupt endpoints. The second interface (interface one) provides scalable isochronous bandwidth consumption. The second interface has four alternate settings that provide different consumption based on the required isochronous bandwidth. The default interface is empty so that the device is capable of scaling down to no isochronous bandwidth.

An HCI frame - consisting of an HCI header and HCI data - should be contained in one USB transaction. A USB transaction is defined as one or more USB frames that contain the data from one IO request. For example, an ACL data packet containing 256 bytes (both HCI header and HCI data) would be sent over the bulk endpoint in one IO request. That IO request will require four 64-byte USB frames - and forms a transaction.

The endpoints are spread across two interfaces so, when adjusting isochronous bandwidth consumption (via select interface calls), any pending bulk and/or interrupt transactions do not have to be terminated and resubmitted.

The following table outlines the required configuration

Interface Number	Alternate Setting	Suggested Endpoint Address	Endpoint Type	Suggested Max Packet Size
HCI Commands				
0	0	0x00	Control	8/16/32/64
HCI Events				
0	0	0x81	Interrupt (IN)	16
ACL Data				
0	0	0x82	Bulk (IN)	32/64
0	0	0x02	Bulk (OUT)	32/64
No active voice channels (for USB compliance)				
1	0	0x83	Isoch (IN)	0
1	0	0x03	Isoch (OUT)	0
One voice channel with 8-bit encoding				
1	1	0x83	Isoch (IN)	9
1	1	0x03	Isoch (OUT)	9
Two voice channels with 8-bit encoding & One voice channel with 16-bit encoding				
1	2	0x83	Isoch (IN)	17
1	2	0x03	Isoch (OUT)	17
Three voice channels with 8-bit encoding				
1	3	0x83	Isoch (IN)	25
1	3	0x03	Isoch (OUT)	25
Two voice channels with 16-bit encoding				
1	4	0x83	Isoch (IN)	33
1	4	0x03	Isoch (OUT)	33
Three voice channels with 16-bit encoding				
1	5	0x83	Isoch (IN)	49
1	5	0x03	Isoch (OUT)	49

The following two examples are used to demonstrate the flow of data given the describe endpoints.

Number of voice channels	Duration of voice data	Encoding
One	3 ms per IO Request	8-bit

Time (ms	USB data (header refers to HCI header) (Receive & Send from the host)	Queued data (read / write)	Time (ms)	Air data	Amount Received / Sent (ms)
0	Receive 0 bytes Send 9 bytes (3 header, 6 data)	0 / 6	0	Send 0	0 / 0
		10 / 6	0.625	Receive 10	1.25 / 0
1	Receive 0 bytes Send 9 bytes (9 bytes HCI data)	10 / 15	1.25	Send 0	1.25 / 0
		20 / 15	1.875	Receive 10	2.50 / 0
2	Receive 0 bytes Send 9 bytes (9 bytes HCI data)	20 / 24	2.50	Send 0	2.50 / 0
		30 / 24	3.125	Receive 10	3.75 / 0
3	Receive 9 bytes (3 header, 6 data) Send 9 bytes (3 header, 6 data)	24 / 20	3.75	Send 10	3.75 / 1.25
4	Receive 9 bytes (9 bytes data) Send 9 bytes (9 bytes HCI data)	25 / 29	4.375	Receive 10	5.0 / 1.25
5	Receive 9 bytes (9 bytes data) Send 9 bytes (9 bytes HCI data)	16 / 28	5.0	Send 10	5.0 / 2.50
		26 / 28	5.625	Receive 10	6.25 / 2.50
6	Receive 9 bytes (3 header, 6 data) Send 9 bytes (3 header, 6 data)	20 / 24	6.25	Send 10	6.25 / 3.75
		30 / 24	6.875	Receive 10	7.5 / 3.75
7	Receive 9 bytes (9 bytes data) Send 9 bytes (9 bytes HCI data)	21 / 23	7.5	Send 10	7.5 / 5.0
8	Receive 9 bytes (9 bytes data) Send 9 bytes (9 bytes HCI data)	22 / 32	8.125	Receive 10	8.75 / 5.0
		22 / 22	8.75	Send 10	8.75 / 6.25
9	Receive 9 bytes (3 header, 6 data) Send 9 bytes (3 header, 6 data)	26 / 28	9.375	Receive 10	10.0 / 6.25

Table 2.1:

Time (ms	USB data (header refers to HCI header) (Receive & Send from the host)	Queued data (read / write)	Time (ms)	Air data	Amount Received / Sent (ms)
10	Receive 9 bytes (9 bytes data) Send 9 bytes (9 bytes HCI data)	17 / 27	10	Send 10	10.0 / 7.5
		27 / 27	10.625	Receive 10	11.25 / 7.5
11	Receive 9 bytes (9 bytes data) Send 9 bytes (9 bytes HCI data)	18 / 26	11.25	Send 10	11.25 / 8.75

Table 2.1:

Convergence is expected because the radio is sending out an average of 8 bytes of voice data every 1 ms and USB is sending 8 bytes of voice data every 1 ms.

Number of voice channels	Duration of voice data	Encoding
Two	3 ms per IO Request	8-bit

Time (ms	USB data (header refers to HCI header) (Receive & Send from the host)	Queued data (read / write)	Time (ms)	Air data	Amount Received / Sent (ms)
0	Receive 0 bytes for Channel #1 Send 17 bytes (3 header, 14 data) for Channel #1	C1- 0/14 C2- 0/0	0	Send 0 for C1	C1- 0/0 C2- 0/0
		C1- 20/14 C2- 0/0	0.625	Receive 20 for C1	C1- 2.5/0 C2- 0/0
1	Receive 0 bytes for Channel #1 Send 17 bytes (17 bytes HCI data) for Channel #1	C1- 20/31 C2- 0/0	1.25	Send 0 for C2	C1- 2.5/0 C2- 0/0
		C1- 20/31 C2- 20/0	1.875	Receive 20 for C2	C1- 2.5/0 C2- 2.5/0
2	Receive 0 bytes for Channel #1 Send 17 bytes (17 bytes HCI data) for Channel #1	C1- 20/28 C2- 20/0	2.50	Send 20 for C1	C1- 2.5/2.5 C2- 2.5/0
		C1- 40/28 C2- 0/0	3.125	Receive 20 for C1	C1- 5.0/2.5 C2- 2.5/0

Table 2.2:

Time (ms	USB data (header refers to HCI header) (Receive & Send from the host)	Queued data (read / write)	Time (ms)	Air data	Amount Received / Sent (ms)
3	Receive 0 bytes for Channel #2 Send 17 bytes (3 header, 14 data) for Channel #2	C1- 40/28 C2- 20/14	3.75	Send 0 for C2	C1- 5.0/2.5 C2- 2.5/0
4	Receive 0 bytes for Channel #2 Send 17 bytes (17 bytes HCI data) for Channel #2	C1- 40/28 C2- 40/31	4.375	Receive 20 for C2	C1- 5.0/2.5 C2- 5.0/0
5	Receive 0 bytes for Channel #2 Send 17 bytes (17 bytes HCI data) for Channel #2	C1- 40/8 C2- 40/48	5.0	Send 20 for C1	C1- 5.0/5.0 C2- 5.0/0
		C1- 60/8 C2- 40/48	5.625	Receive 20 for C1	C1- 7.5/5.0 C2- 5.0/0
6	Receive 17 bytes (3 header, 14 data) for Channel #1 Send 17 bytes (3 header, 14 data) for Channel #1	C1- 46/22 C2- 40/28	6.25	Send 20 for C2	C1- 7.5/5.0 C2- 5.0/2.5
		C1- 46/22 C2- 60/28	6.875	Receive 20 for C2	C1- 7.5/5.0 C2- 7.5/2.5
7	Receive 17 bytes (17 bytes data) for Channel #1 Send 17 bytes (17 bytes HCI data) for Channel #1	C1- 29/19 C2- 60/28	7.5	Send 20 for C1	C1- 7.5/7.5 C2- 7.5/2.5
8	Receive 17 bytes (17 bytes data) for Channel #1 Send 17 bytes (17 bytes HCI data) for Channel #1	C1- 32/36 C2- 60/28	8.125	Receive 20 for C1	C1- 10/7.5 C2- 7.5/2.5
		C1- 32/36 C2- 60/8	8.75	Send 20 for C2	C1- 10/7.5 C2- 7.5/5.0
9	Receive 17 bytes (3 header, 14 data) for Channel #2 Send 17 bytes (3 header, 14 data) for Channel #2	C1- 32/36 C2- 54/22	9.375	Receive 20 for C2	C1- 10/7.5 C2- 10/5.0
10	Receive 17 bytes (17 bytes data) for Channel #2 Send 17 bytes (17 bytes HCI data) for Channel #2	C1- 32/16 C2- 37/39	10	Send 20 for C1	C1- 10/10 C2- 10/5.0
		C1- 52/16 C2- 37/39	10.625	Receive 20 for C1	C1- 12.5/10 C2- 10/5.0

Table 2.2:

Time (ms	USB data (header refers to HCI header) (Receive & Send from the host)	Queued data (read / write)	Time (ms)	Air data	Amount Received / Sent (ms)
11	Receive 17 bytes (17 bytes data) for Channel #2 Send 17 bytes (17 bytes HCI data) for Channel #2	C1- 52/16 C2- 20/36	11.25	Send 20 for C2	C1- 12.5/10 C2- 10/7.5

Table 2.2:

2.2 CONTROL ENDPOINT EXPECTATIONS

Endpoint 0 is used to configure and control the USB device. Endpoint 0 will also be used to allow the host to send HCI-specific commands to the host controller. When the USB firmware receives a packet over this endpoint that has the Bluetooth class code, it should treat the packet as an HCI command packet.

2.3 BULK ENDPOINTS EXPECTATIONS

Data integrity is a critical aspect for ACL data. This, in combination with bandwidth requirements, is the reason for using a bulk endpoint. Multiple 64-byte packets can be shipped, per millisecond, across the bus.

Suggested bulk max packet size is 64 bytes. Bulk has the ability to transfer multiple 64-byte buffers per one millisecond frame, depending on available bus bandwidth.

Bulk has the ability to detect errors and correct them. Data flowing through this pipe might be destined for several different slaves. In order to avoid starvation, a flow control model similar to the shared endpoint model is recommended for the host controller.

2.4 INTERRUPT ENDPOINT EXPECTATIONS

An interrupt endpoint is necessary to ensure that events are delivered in a predictable and timely manner. Event packets can be sent across USB with a guaranteed latency.

The interrupt endpoint should have an interval of 1 ms.

The USB software and firmware requires no intimate knowledge of the events passed to the host controller.

2.5 ISOCHRONOUS ENDPOINTS EXPECTATIONS

These isochronous endpoints transfer SCO data to and from the host controller of the radio.

Time is the critical aspect for this type of data. The USB firmware should transfer the contents of the data to the host controllers' SCO FIFOs. If the FIFOs are full, the data should be overwritten with new data.

These endpoints have a one (1) ms interval, as required by Chapter 9 of the USB Specification, Versions 1.0 and 1.1.

The radio is capable of three (3) 64Kb/s voice channels (and can receive the data coded in different ways – 16-bit linear audio coding is the method that requires the most data). A suggested max packet size for this endpoint would be at least 64 bytes. (It is recommended that max packet sizes be on power of 2 boundaries for optimum throughput.) However, if it is not necessary to support three voice channels with 16-bit coding, 32 bytes could also be considered an acceptable max packet size.

3 CLASS CODE

A class code will be used that is specific to all USB Bluetooth devices. This will allow the proper driver stack to load, regardless of which vendor built the device. It also allows HCI commands to be differentiated from USB commands across the control endpoint.

The class code (bDeviceClass) is 0xE0 – Wireless Controller.

The SubClass code (bDeviceSubClass) is 0x01 – RF Controller.

The Protocol code (bDeviceProtocol) is 0x01 – Bluetooth programming.

4 DEVICE FIRMWARE UPGRADE

Firmware upgrade capability is not a required feature. But if implemented, the firmware upgrade shall be compliant with the "Universal Serial Bus Device Class Specification for Device Firmware Upgrade" (version 1.0 dated May 13, 1999) available on the USB Forum web site at http://www.usb.org.

5 LIMITATIONS

5.1 POWER SPECIFIC LIMITATIONS

Today, the host controller of USB-capable machines resides inside a chip known as PIIX4. Unfortunately, because of errata, the USB host controller will not receive power while the system is in S3 or S4. This means that a USB wake-up can only occur when the system is in S1 or S2.

Another issue with the USB host controller is that, while a device is attached, it continually snoops memory to see if there is any work that needs to be done. The frequency that it checks memory is 1ms. This prevents the processor from dropping into a low power state known as C3. Because the notebook processor is not able to enter the C3 state, significant power loss will occur. This is a real issue for business users – as a typical business user will spend almost 90% of their time in the C3 state.

5.2 OTHER LIMITATIONS

Data corruption may occur across isochronous endpoints. Endpoints one and two may suffer from data corruption.

USB provides 16-CRC on all data transfers. The USB has a bit error rate of 10^{-13}.

Note that when a dongle is removed from the system, the radio will lose power (assuming this is a bus-powered device). This means that devices will lose connection.

Part H:3

HCI RS232 TRANSPORT LAYER

An addendum to the HCI document

This document describes the RS232 transport layer (between the Host and the Host Controller). HCI command, event and data packets flow through this layer, but the layer does not decode them.

Bluetooth.

CONTENTS

1 GENERAL

The objective of the HCI RS232 Transport Layer is to make it possible to use the Bluetooth HCI over one physical RS232 interface between the Bluetooth Host and the Bluetooth Host Controller.

Figure 1.1:

2 OVERVIEW

There are four kinds of HCI packets that can be sent via the RS232 Transport Layer; i.e. HCI Command Packet, HCI Event Packet, HCI ACL Data Packet and HCI SCO Data Packet (see "Host Controller Interface Functional Specification" on page 517). HCI Command Packets can only be sent to the Bluetooth Host Controller, HCI Event Packets can only be sent from the Bluetooth Host Controller, and HCI ACL/SCO Data Packets can be sent both to and from the Bluetooth Host Controller.

However, HCI does not provide the ability to differentiate the four HCI packet types. Therefore, if the HCI packets are sent via a common physical interface, a HCI packet indicator has to be added according to the Table 2.1 below.

HCI packet type	HCI packet indicator
HCI Command Packet	0x01
HCI ACL Data Packet	0x02
HCI SCO Data Packet	0x03
HCI Event Packet	0x04
Error Message Packet*	0x05
Negotiation Packet*	0x06

Table 2.1: HCI RS232 Packet Header

In addition to those four HCI packet types, two additional packet types are introduced to support dynamic negotiation and error reporting. The Error Message Packet (0x05) is used by the receiver to report the nature of error to the transmitting side. The Negotiation Packet (0x06) is used to negotiate the communication settings and protocols.

The HCI packet indicator shall be followed by an 8-bit sequence number that is incremented by 1 every time any of the above packets are sent, except when the retransmission packets are sent as a part of the error recovery. The HCI packet shall immediately follow the sequence number field. All four kinds of HCI packets have a length field, which is used to determine how many bytes are expected for the HCI packet. The Error Message Packet and Negotiation Packet are fixed-length packets, although the negotiation packet can be extended up to 7 more bytes, based on the number in the extension field.

The frame of the basic RS232 Transport Packet is shown below.

LSB *MSB*

Packet Type (8-bit)	SEQ No (8-bit)	HCI Packet or Error Message/Negotiation Packet payload

The least significant byte is transmitted first.

3 NEGOTIATION PROTOCOL

Before sending any bytes over the RS232 link, the baud rate, parity type, number of stop bit and protocol mode should be negotiated between the Host Controller and the Host. Tdetect is the maximum time required for the transmitter to detect the CTS state change, plus the time it takes to flush the transmit buffer if RTS/CTS is used for error indication and re-synchronization. Otherwise, Tdetect represents the local-side interrupt latency. Host will first send a negotiation packet with the maximum suggested values, plus Host's Tdetect value with Ack code = 000b at the default UART settings specified below, using protocol mode = 0x13. At the same time, the Host Controller side also sets its UART settings to the same initiating parameters and waits for the negotiation packet from the Host.
If the Host Controller side can accept the suggested values from the Host, it sends back the negotiation packet with the same UART setting values, plus Host Controller's Tdetect value with Ack code = 001b. Then, the Host sends back the negotiation packet with the same UART setting values, plus Host's Tdetect with Ack code = 001b as the final acknowledgment, and then sets its Host's UART to the new value. After it has received the final acknowledgment packet from the Host, the Host Controller also changes its UART setting to the new values.

On the other hand, if the Host Controller side cannot accept the suggested value, it should send a set of new suggested values and its own Tdetect value with Ack code = 010b. Each side should continue these steps until both sides receive the accepted Ack code value. Error detection and error recovery during the initial negotiation are performed in the same manner as described in Section 5 on page 785 (Protocol Mode 0x13)

The negotiation phase can be initiated again by either side at any time in order to renegotiate the new values, or just to inform the new Tdetect time. When the negotiation is reinitiated during the data transfer, it should use the previously negotiated settings to exchange the new parameters rather than using the default values.

The initiating parameters:

baud rate: 9600 bps

parity type: no parity

number of data bit: 8 (Note: Only 8-bit data length is allowed.)

number of stop bit: 1

protocol mode: 0x13 (HDLC like framing with COBS/CCITT-CRC)

The negotiation packet format:

LSB *MSB*

Packet Type header 0x06 (8 bits)	SEQ No (8 bits)	UART Settings and ACK (8 bits)	Baud Rate (16 bits)	Tdetect Time (16 bits)	Protocol Mode (8 bit)

SEQ No:
This is an 8-bit number that is incremented by 1 each time a packet is transmitted, excluding the retransmission packet. The unsigned Little Endian format is used.

UART Settings and ACK Field

Bit 0-1	Bit 2	Bit 3	Bit 4	Bit 5-7
Reserved	Stop bit (1 bit)	Parity Enable (1 bit)	Parity Type (1 bit)	Ack Code (3 bits)

Stop Bit:

0: 1 stop bit
1: 2 stop bits

Parity Enable:

0: No parity
1: Parity

Parity Type:

0: Odd Parity
1: Even Parity

Ack Code:

000b: Request
001b: Accepted
010b: Not accepted with new suggested values
011b-111b: Reserved

Baud Rate:

N should be entered for baud rate where
Baud rate = 27,648,000 / N
N=0 is invalid

Maximum possible rate is therefore 27.648Mbps
Minimum possible rate is therefore 421.88bps

The unsigned Little Endian format is used, and the least significant byte should be transmitted first.

Tdetect Time:

This 16-bit field should be filled with the maximum required time for the transmitter to detect the CTS change, plus the time it takes to flush the transmit FIFO if RTS and CTS are used for resynchronization. Otherwise, it should be filled with the local interrupt latency.

The unit of time should be specified in 100 microseconds.

The unsigned Little Endian format is used, and the least significant byte should be transmitted first.

Protocol Mode

Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
CRC Used	Delimiter Used	RTS /CTS used	RTS/CTS Mode	Error Recovery Used	Ext0	Ext1	Ext2

CRC Used:

0: CRC-CCITT is not attached at the end of the packet.

1: CRC-CCITT is attached at the end of the packet. (Default)

16-bit CRC can be used with either RTS/CTS or delimiters, although this specification only describes a case when it is used with delimiters.
Generator Polynomial = $x^{16}+x^{12}+x^5+1$

Delimiter Used:

0: Delimiter, 0x7E, is not used.

1: Delimiter, 0x7E, is used with COBS. (Default)

RTS/CTS Used:

0: RTS/CTS is not used. (Default)

1: RTS/CTS is used.

RTS/CTS Mode:

0: RTS/CTS is used for Error indication and resynchronization. (Default)

1: RTS/CTS is used for hardware flow control. Please refer to "HCI UART Transport Layer" on page 795 for details.

Error Recovery Used:

0: Error Recovery is not supported.
Even if Error Recovery is not supported, Error Message has to be sent.

1: Error Recovery is supported. (Default)
Error Recovery retransmits the packet with error and all subsequent packets if RTS/CTS are used for synchronization. On the other hand, if 0x7E is used as a delimiter with COBS as a synchronization mechanism, then the error recovery retransmits only the packet with error. Please refer to following sections for details.

Ext2,Ext1,Ext0:

These three bits indicate the number of extra bytes attached to the negotiation packet for future expansion.

4 PACKET TRANSFER PROTOCOL

The packet can be transferred with parity enabled or disabled, and with or without CRC – depending on the environment – as a mechanism to detect the error.

As a synchronization mechanism, one can select either RTS/CTS, or delimiters. Usage of RTS/CTS reduces the computation time for COBS encoding, but it requires two extra copper wires which may not be suitable in some applications. If three-wire cable must be used, or programmable RTS and CTS are not available, delimiter, 0x7E, can be used with COBS.

However, error recovery for these two alternatives may differ slightly. If the RTS/CTS is used for resynchronization, it would be simpler to retransmit all the packets, starting with the packet that had an error. If delimiters are used, the transmitter should retransmit only the packet with an error. The error recovery can be disabled, but the error message packet should still be sent to the transmitter side when the receiver side detects an error.

The HCI RS232 transport layer always uses a data length of 8 bits, and this specification assumes the Little Endian format. Furthermore, the least significant byte should be transmitted first.

The Host Controller may choose to support only one protocol mode, but the Host should be able to support any combination.

Two common schemes (Protocol mode = 0x13 and 0x14) are defined in the following sections to illustrate the usage of each mode.

5 USING DELIMITERS WITH COBS FOR SYNCHRONIZATION

This section illustrates how delimiters with COBS are used for synchronization, and how error recovery procedure is performed if delimiters are used as a mechanism to synchronize. This is described using protocol mode 0x13.

5.1 USING DELIMITERS WITH COBS AND CRC, PROTOCOL MODE 0X13

In case RTS/CTS are not available, or if they are hard-wired to be used as a hardware flow control, the HDLC-like framing with the 16-bit CRC (CRC-CCITT) and delimiter 0x7E with COBS (Consistent Overhead Byte Stuffing) [2] are used as a means to detect an error and to resynchronize.

The CRC-CCITT uses the following generator polynomial for 16-bit checksum: $x^{16}+x^{12}+x^5+1$. The 16-bit CRC should be attached to the end of the packet, but right before the ending delimiter, 0x7E. The beginning delimiter, 0x7E, should be followed by the packet type indicator field.

The Consistent Overhead Byte Stuffing is a recent proposal to PPP that yields less than 0.5% overhead, regardless of the data pattern. It uses two steps to escape the delimiter, 0x7E. The first step is eliminating zeros and then replacing all 0x7E with 0x00 between the beginning and ending delimiters.

A simple error recovery scheme is adapted here to minimize the overhead of supporting the error recovery. When the receiving end detects any error, it should send the error message packet with an error type back to the transmitting side. This error message packet will contain a Sequence Number with Error field (SEQ No with Error) indicating in which packet the error was detected. The Sequence Number field (SEQ No) that is on every packet is an 8-bit field that is incremented by 1 each time any type of packet is transmitted, except for the retransmission packets. The retransmitted packets should contain the original sequence number in the SEQ No field.

The transmitting side should retransmit only the HCI packets that had an error, which is indicated by the SEQ No with Error field. It is the responsibility of the receiving end to reorder the packets in the right order. If the transmitting side doesn't have the packet with the correct sequence number in the retransmission holding buffer, it should send the error message packet with the Error Type equal to 0x81 and SEQ No with Error field with the missing sequence number for the retransmission packet, so that the receiving end can detect missing packets. In this case the full error recovery cannot be performed. However, the receiving side can at least detect the loss of packets.

The receiving side should wait at least 4 times the sum of remote Tdetect, local Tdetect and the transmission time of the error message packet, plus the retransmission packet, before it times out when it is waiting for the retransmission packet. When it times out, the receiver has an option of re-requesting it by sending another error message packet with error type = 0x09, or simply dropping it and reporting it to the higher layer.

5.2 FRAME FORMAT

BOF(0x7E), CRC-CCITT, and EOF(0x7E) are added as shown below to those basic packets described in this document. When the CRC is transmitted, the least significant byte should be transmitted first.

LSB *MSB*

0x7E BOF (8 bits)	Packet Type (8 bits)	SEQ No (8 bits)	....Payload....	CRC (16 bits)	0x7E EOF (8 bits)

5.3 ERROR MESSAGE PACKET

The error-message packet format is the following:

LSB *MSB*

Packet Type, 0x05 (8-bit field)	Sequence No (8-bit field)	Error Type (8-bit field)	SEQ No with Error (8-bit field)

Error Type	Description
0x00	Reserved
0x01	Overrun Error
0x02	Parity Error
0x03	Reserved
0x04	Framing Error
0x05-0x07	Reserved
0x08	CRC Error
0x09	Missing SEQ No
0x0A-0x80	Reserved
0x81	Missing Retransmission Packet
0x82- 0xFF	Reserved

Table 5.1: Error Type available

5.4 CONSISTENT OVERHEAD BYTE STUFFING

Code(n)	Followed by	Description
0x00		Unused.
0x01-0xCF	n-1 data bytes	The n-1 data bytes plus implicit trailing zero.
0xD0	n-1 data bytes	The n-1 data bytes without trailing zero.
0XD1		Unused.
0xD2		Reserved for future.
0xD3-0xDF	none	A run of n-0xD0 zeros.
0xE0-0xFE	n-E0 data bytes	The data bytes with two trailing zeros.
0xFF		Unused.

Table 5.2:

The COBS requires two step encodes.

The first step is the zero-elimination step. This step takes place after attaching the 16-bit CRC if CRC is enabled, but before adding the beginning and ending delimiters, 0x7E. Each COBS code block consists of the COBS code followed by zero or more data bytes. Code bytes 0x00, 0xD1, 0xD2 and 0xFF are never used. The COBS zero-elimination procedure searches the packet for the first occurrence of value zero. To simplify the encoding, a zero is added temporarily at the end of the packet, after the CRC, as a temporary place holder. The number of octets up to and including the first zero determines the code to be used. If this number is 207 or less, then the number itself is used as a COBS code byte, followed by the actual non-zero data bytes themselves, excluding the last byte, which is zero. On the other hand, if the number is more than 207, then the code byte 0xD0 is used, followed by the first 207 non-zero bytes. This process is repeated until all of the bytes of the packet, including the temporary place-holding zero at the end, have been encoded. If a pair of 0x00 is detected after 0 to 30 non-zero octets, the count of octets plus 0xE0 is used as the COBS code, followed by the non-zero octets, excluding the pair of zeros. If a run of three to fifteen 0x00 octets are detected, then the count of these 0x00 octets, plus 0xD0, is used as the code, followed by no other bytes.

The second step is replacing 0x7E with 0x00. The two steps can be done together in a loop, to reduce the encoding time.

For more details and a reference code, please refer to “PPP Consistent Overhead Byte Stuffing (COBS)” by J. Carlson et al [2].

6 USING RTS/CTS FOR SYNCHRONIZATION

This section illustrates how RTS and CTS are used to resynchronize, and how error-recovery procedure is performed if RTS and CTS are used as a mechanism to synchronize. This is described using protocol mode 0x14.

6.1 USING RTS/CTS FOR SYNC WITHOUT CRC, PROTOCOL MODE 0X14

The flow of HCI packet transfer is handled by two MODEM control/status signals, -RTS and -CTS. -RTS and -CTS are connected in a null MODEM fashion, meaning that the local-side -RTS should be connected to the remote-side -CTS, and the local-side -CTS should be connected to the remote-side -RTS. These MODEM control/status signals are used to notify the detection of an error to the other side, as well as to resynchronize the beginning of the packet after an error is detected. A very simple error-recovery scheme is adapted here to minimize the overhead of supporting this.

The HCI packet is transmitted only while CTS bit is 1. If the CTS bit changes to 0 during the HCI packet transfer or after the last byte is transmitted, this indicates that there was some error. The receiving end will deassert RTS as soon as it detects any error, and should send the error packet with an error type back to the transmission side. This error packet will contain a Sequence Number with Error field that indicates in which packet the error was detected. The sequence number field that is on every packet is an 8-bit field that is incremented by 1 each time any type of packet is transmitted, except for the retransmission packets. The retransmitted packets should contain the original sequence number in the SEQ No field.

When the transmitting end detects CTS bit changing from 1 to 0 at any time, the transmitting end should hold the transmission and wait until the error packet is received before resuming the transmission. When the receiving end is ready to receive the new data, it should assert RTS after the minimum of Tdetect time. Tdetect time is the maximum time required for the transmit side to detect the state change on CTS bit, plus the time it takes to flush the transmit buffer. The Tdetect value of each side should be informed to the other side during the negotiation phase. The local Tdetect value and the remote side Tdetect value together, along with the baud rate, can also be used to estimate the queue length required for the retransmission holding buffer. Before the receiving side asserts RTS line again, it should flush the RX buffer.

The transmission side should retransmit all of the HCI packets from the packet that had an error, which is indicated by SEQ No with Error field. Before it retransmits, it should flush the transmit buffer that may hold the leftover data from the aborted previous packet. As it retransmits the packets from the retransmission holding buffer, it should start transmitting the packet with the Sequence Number that matches the SEQ No with Error. If the transmitting side doesn't have the packet with the correct sequence number in the retransmission holding buffer, the transmitter should send an error message packet with error type 0x81, and it should skip to the packet with the sequence number that is available in the buffer. In this case, the full error recovery cannot be performed. However, the receiving side can, at least, detect the loss of packets.

6.2 ERROR MESSAGE PACKET

The error-message packet format is the following:

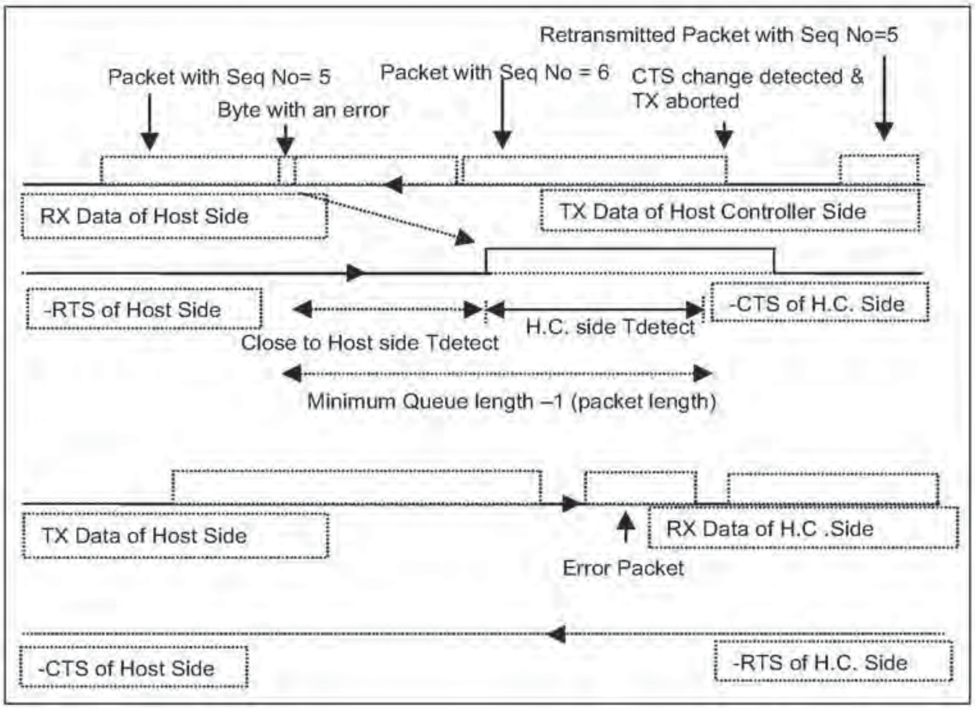
LSB *MSB*

Packet header 0x05 (8-bit field)	Sequence No (8-bit field)	Error Type (8-bit field)	SEQ No with Error (8-bit field)

Error Type	Description	Comment
0x00	Reserved	
0x01	Overrun Error	
0x02	Parity Error	
0x03	Reserved	
0x04	Framing Error	
0x05-0x07	Reserved	
0x08	CRC Error*	Not applicable In Mode 0x14
0x09	Missing SEQ No	
0x0A-0x80	Reserved	
0x81	Missing Retransmission Packet	
0x82- 0xFF	Reserved	

Table 6.1: Error Type available

6.3 EXAMPLE OF SIGNALLING



6.4 CONTROL FLOW EXAMPLES

6.4.1 Case 1, Normal Recovery Process

Controller Side	Host Side
0) RTS is asserted and the asserted CTS is detected.	0) RTS is asserted and the asserted CTS is detected.
	1) Ctrl/Data[n] is sent out and Ctrl/Data[n] is stored in the retransmission holding buffer.
2) Ctrl/Data[n] is received with an error.	
3) Deasserts RTS	
4a) Error message for [n] is sent and Error message for [n] is stored in the TX retransmit holding buffer. 4b) Empties the RX FIFO and waits for Tdetect (Host) amount of time.	4) Detects CTS deasserted.
	5a) Stops further transmission and waits until the TX FiFO is empty (or Flush the FIFO if it can.) 5b) Error message for [n] is received.
6) Asserts RTS	
.	7) The asserted CTS is detected.
	8) Retransmits Ctrl/Data[n].

6.4.2 Case 2, Both sides detect an error simultaneously

Controller Side	Host Side
0) RTS is asserted and the asserted CTS is detected.	0) RTS is asserted and the asserted CTS is detected.
1) Ctrl/Data[x] is sent and Ctrl/Data[x] is stored in the retransmission holding buffer.	1) Ctrl/Data[n] is sent and Ctrl/Data[n] is stored in the retransmission holding buffer.
2) Ctrl/Data[n] is received with an error.	2) Ctrl/Data[x] is received with an error.
3) Deasserts RTS.	3) Deasserts RTS.
4) Detects CTS deasserted.	4) Detects CTS deasserted.
5a) Stops further transmission and waits until the TX FiFO is empty (or Flush the FIFO if it can).	5a) Stops further transmission and waits until the TX FiFO is empty (or Flush the FIFO if it can).

Controller Side	Host Side
5b) Empties the RX FIFO and waits for Tdetect (Host) amount of time.	5b) Empties the RX FIFO and waits for Tdetect (Controller) amount of time.
6) Asserts RTS.	6) Asserts RTS.
7) The asserted CTS is detected.	7) The asserted CTS is detected.
8) Error message for [n] is sent and Error message for [n] is stored in the TX retransmit holding buffer.	8) Error message for [x] is sent and Error message for [x] is stored in the TX retransmit holding buffer.
9) Error message for [x] is received.	9) Error message for [n] is received.
10) Retransmits Ctrl/Data[x].	10) Retransmits Ctrl/Data[n].

6.4.3 Case 3, Error Message with an error

Controller Side	Host Side
0) RTS is asserted and the asserted CTS is detected.	0) RTS is asserted and the asserted CTS is detected.
	1) Ctrl/Data[n] is sent and Ctrl/Data[n] is stored in the retransmission holding buffer.
2) Ctrl/Data[n] is received with an error.	
3) Deasserts RTS.	
4a) Error message for [n] (Err[n]) is sent and Err[n] is stored in the TX retransmit holding buffer. 4b) Empties the RX FIFO and waits for Tdetect (Host) amount of time.	4) Detects CTS deasserted.
	5a) Stops further transmission and waits until the TX FiFO is empty (or Flush the FIFO if it can.) 5b) Error message for [n] is received with an error.
6) Asserts RTS.	6a) Deasserts RTS. 6b) Empties the RX FIFO and waits for Tdetect (Controller) amount of time.
7) Detects CTS deasserted.	
8) Stops further transmission and waits until the TX FiFO is empty (or Flush the FIFO if it can.)	8) The asserted CTS detected.

Controller Side	Host Side
	9a) Error message for Err[n] is sent and Error message for Err[n] is stored in the retransmission holding buffer. 9b) Asserts RTS.
10a) Error message for Err[n] is received. 10b) The asserted CTS detected.	
11) Retransmits Error message for [n].	
	12) Error message for [n] is received.
	13) Retransmit Ctrl/Data[n].

7 REFERENCES

[1] Bluetooth Host Controller Interface Function Specification

[2] J. Carlson, S. Cheshire, M. Baker, draft-ietf-pppext-cobs-00, "PPP Consistent Overhead Byte Stuffing (COBS)", November, 1997

[3] Bluetooth HCI UART Transport Layer Specification

Part H:4

HCI UART TRANSPORT LAYER

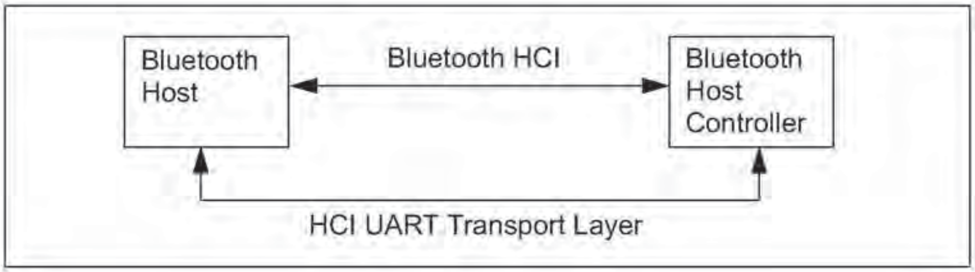
An addendum to the HCI document

This document describes the UART transport layer (between the Host and the Host Controller). HCI command, event and data packets flow through this layer, but the layer does not decode them.

CONTENTS

1 GENERAL

The objective of this HCI UART Transport Layer is to make it possible to use the Bluetooth HCI over a serial interface between two UARTs on the same PCB. The HCI UART Transport Layer assumes that the UART communication is free from line errors. See also "HCI RS232 Transport Layer" on page 775.



2 PROTOCOL

There are four kinds of HCI packets that can be sent via the UART Transport Layer; i.e. HCI Command Packet, HCI Event Packet, HCI ACL Data Packet and HCI SCO Data Packet (see "Host Controller Interface Functional Specification" on page 517). HCI Command Packets can only be sent to the Bluetooth Host Controller, HCI Event Packets can only be sent from the Bluetooth Host Controller, and HCI ACL/SCO Data Packets can be sent both to and from the Bluetooth Host Controller.

HCI does not provide the ability to differentiate the four HCI packet types. Therefore, if the HCI packets are sent via a common physical interface, a HCI packet indicator has to be added according to Table 2.11 below.

HCI packet type	**HCI packet indicator**
HCI Command Packet	0x01
HCI ACL Data Packet	0x02
HCI SCO Data Packet	0x03
HCI Event Packet	0x04

Table 2.1: HCI packet indicators

The HCI packet indicator shall be sent immediately before the HCI packet. All four kinds of HCI packets have a length field, which is used to determine how many bytes are expected for the HCI packet. When an entire HCI packet has been received, the next HCI packet indicator is expected for the next HCI packet. Over the UART Transport Layer, only HCI packet indicators followed by HCI packets are allowed.

3 RS232 SETTINGS

The HCI UART Transport Layer uses the following settings for RS232:

Baud rate:	**manufacturer-specific**
Number of data bits:	**8**
Parity bit:	**no parity**
Stop bit:	**1 stop bit**
Flow control:	**RTS/CTS**
Flow-off response time:	**3 ms**

Flow control with RTS/CTS is used to prevent temporary UART buffer overrun. It should not be used for flow control of HCI, since HCI has its own flow control mechanisms for HCI commands, HCI events and HCI data.

If CTS is 1, then the Host/Host Controller is allowed to send.
If CTS is 0, then the Host/Host Controller is not allowed to send.

The flow-off response time defines the maximum time from setting RTS to 0 until the byte flow actually stops.

The RS232 signals should be connected in a null-modem fashion; i.e. the local TXD should be connected to the remote RXD and the local RTS should be connected to the remote CTS and vice versa.

4 ERROR RECOVERY

If the Host or the Host Controller lose synchronization in the communication over RS232, then a reset is needed. A loss of synchronization means that an incorrect HCI packet indicator has been detected, or that the length field in an HCI packet is out of range.

If the UART synchronization is lost in the communication from Host to Host Controller, then the Host Controller shall send a Hardware Error Event to tell the Host about the synchronization error. The Host Controller will then expect to receive an HCI_Reset command from the Host in order to perform a reset. The Host Controller will also use the HCI_Reset command in the byte stream from Host to Host Controller to re-synchronize.

If the UART synchronization is lost in the communication from Host Controller to Host, then the Host shall send the HCI_Reset command in order to reset the Host Controller. The Host shall then re-synchronize by looking for the HCI Command Complete event for the HCI_Reset command in the byte stream from Host Controller to Host.

See "Host Controller Interface Functional Specification" on page 517 for HCI commands and HCI events.

Part I:1

BLUETOOTH TEST MODE

This document describes the test mode for hardware and low-level functionality tests of Bluetooth devices. The test mode includes transmitter tests (packets with constant bit patterns) and loop back tests.

CONTENTS

1 GENERAL DESCRIPTION

The test mode supports testing of the Bluetooth transmitter and receiver. It is intended mainly for certification/compliance testing of the radio and baseband layer, and may also be used for regulatory approval or in-production and after-sales testing.

A device in test mode must not support normal operation. For security reasons the test mode is designed such that it offers no benefit to the user. Therefore, no data output or acceptance on a HW or SW interface is allowed.

1.1 TEST SETUP

The setup consists of a device under test (DUT) and a tester. Optionally, additional measurement equipment may be used.

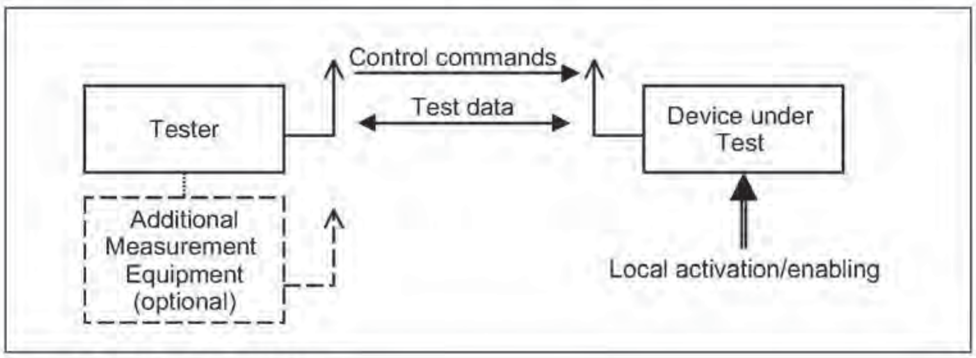


Figure 1.1: Setup for Test Mode

Tester and DUT form a piconet where the tester acts as master and has full control over the test procedure. The DUT acts as slave.

The control is done via the air interface using LMP commands (see Section 3 on page 817 and "Link Manager Protocol" on page 185). Hardware interfaces to the DUT may exist, but are not subject to standardization.

The test mode is a special state of the Bluetooth model. For security and type approval reasons, a device in test mode may not support normal operation. When the DUT leaves the test mode it enters the standby state. After power-off the Bluetooth device must return to standby state.

1.2 ACTIVATION

The activation may be carried out locally (via a HW or SW interface), or using the air interface.

- For activation over the air interface, entering the test mode must be locally enabled for security and type approval reasons. The implementation of this local enabling is not subject to standardization.

  The tester sends an LMP command that forces the DUT to enter test mode. The DUT terminates all normal operation before entering the test mode.

  The DUT shall return an LMP_Accepted on reception of an activation command. LMP_Not_Accepted shall be returned if the DUT is not locally enabled.

- If the activation is performed locally using a HW or SW interface, the DUT terminates all normal operation before entering the test mode.

  Until a connection to the tester exists, the device shall perform page scan and inquiry scan. Extended scan activity is recommended.

1.3 CONTROL

Control and configuration is performed using special LMP commands (see Section 3 on page 817). These commands must be rejected if the Bluetooth device is not in test mode. In this case, an LMP_not_accepted is returned. The DUT shall return an LMP_accepted on reception of a control command when in test mode.

A Bluetooth device in test mode must ignore all LMP commands not related to control of the test mode. LMP commands dealing with power control and the request for LMP features (LMP_features_req) are allowed in test mode; the normal procedures are also used to test the adaptive power control.

The DUT can be commanded to leave the test mode by an LMP_Detach command or by sending an LMP_test_control command with test scenario set to 'exit test mode'.

2 TEST SCENARIOS

2.1 TRANSMITTER TEST

The Bluetooth device transmits a constant bit pattern. This pattern is transmitted periodically with packets aligned to the slave TX timing of the piconet formed by tester and DUT. The same test packet is repeated for each transmission.

The transmitter test is started when the master sends the first POLL packet. In non-hopping mode agreed frequency is used for this POLL packet.

The tester transmits at his TX slots (control commands or POLL packets). The slave starts burst transmission in the following slave TX slot. The master's polling interval is fixed and defined beforehand. The device under test may transmit its burst according to the normal timing even if no packet from the tester was received.

The burst length may exceed the length of a one slot packet. In this case the tester may take the next free master TX slot for polling. The timing is illustrated in Figure 2.1.

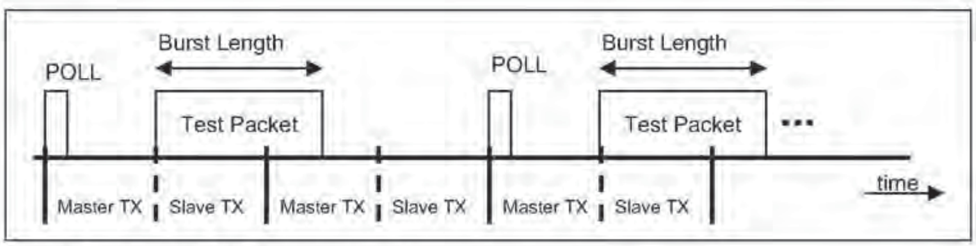


Figure 2.1: Timing for Transmitter Test

2.1.1 Packet Format

The test packet is a normal Bluetooth packet, see Figure 2.2. For the payload itself see below.

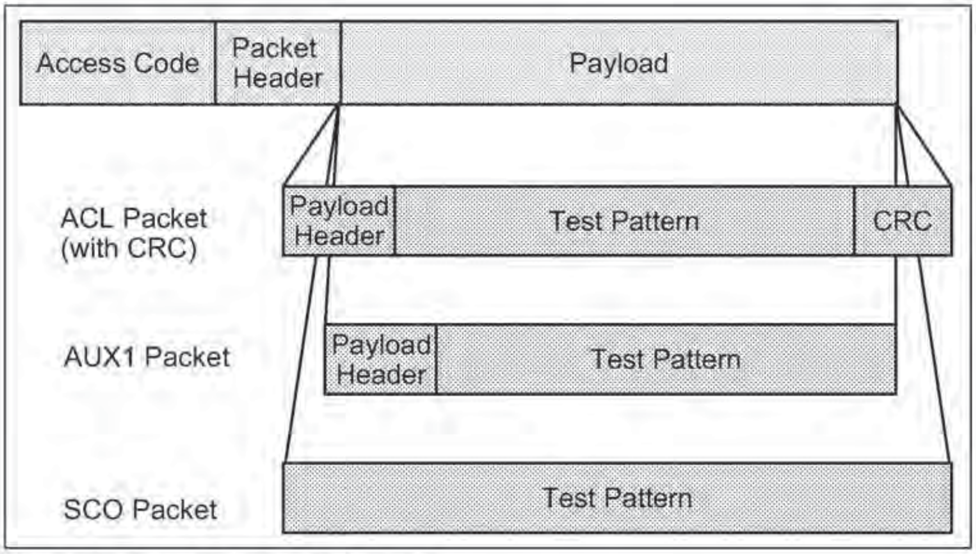


Figure 2.2: General Format of TX Packet

During configuration the tester defines:

- the packet type to be used
- payload length

For the payload length, the restrictions from the baseband specification apply (see "Baseband Specification" on page 33.). In case of ACL packets the payload structure defined in the baseband specification is preserved as well, see Figure 2.2.

For the transmitter test mode, only packets without FEC should be used; i.e. HV3, DH1, DH3, DH5and AUX1packets. Support of packet type is only mandatory up to the longest implemented packet type.

In transmitter test mode, the packets exchanged between tester and DUT are not scrambled with the whitening sequence. Whitening is turned off when the DUT has accepted to enter the transmitter test mode, and is turned on when the DUT has accepted to exit the transmitter test mode, see Figure 2.3.[1]

1. Note: Implementations must ensure that retransmissions of the LMP_Accepted messages use the same whitening status.

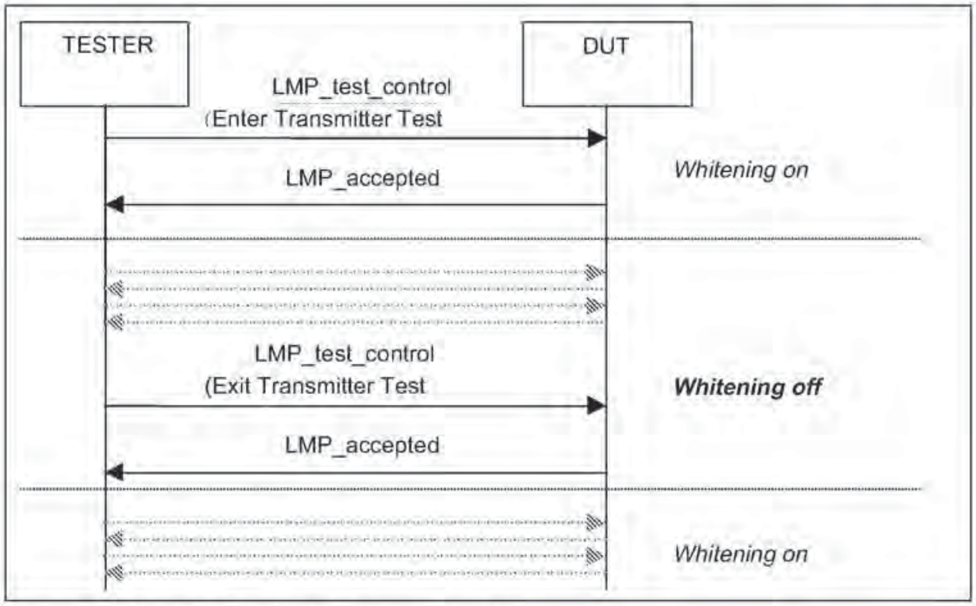


Figure 2.3: Use of whitening in Transmitter mode

2.1.2 Pseudorandom Sequence

In case of pseudorandom bit sequence, the same sequence of bits is used for each transmission (i.e. the packet is repeated, see above). A PRBS-9 Sequence[2] is used, see [2] and [3].

The properties of this sequence are as follows (see [3]). The sequence may be generated in a nine-stage shift register whose 5th and 9th stage outputs are added in a modulo-two addition stage (see Figure 2.4), and the result is fed back to the input of the first stage. The sequence begins with the first ONE of 9 consecutive ONEs; i.e. the shift register is initialized with nine ones.

- Number of shift register stages: 9
- Length of pseudo-random sequence: 2^9-1 = 511 bits
- Longest sequence of zeros: 8 (non-inverted signal)

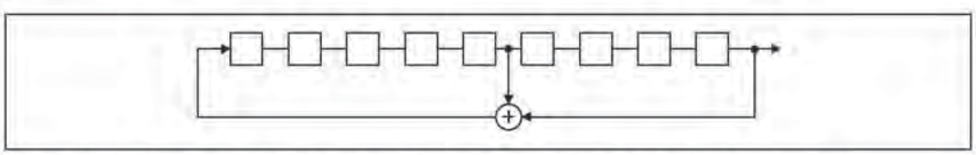


Figure 2.4: Linear Feedback Shift Register for Generation of the PRBS sequence

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

2.1.3 Reduced Hopping Sequence

To support quick testing of the radio over the complete frequency range, a reduced hopping mode is defined. Implementation of this mode is optional for Bluetooth devices and modules.

Reduced hopping uses only five frequencies, on which a sequential hopping is done (channels 0, 23, 46, 69 and 93 are used[3]), see Figure 2.5.

Channel	→ 0 →	23 →	46 →	69 →	93 →
$CLK_{27\text{-}1}$ mod 5 =	0	1	2	3	4

Figure 2.5: Reduced hopping scheme

The timing is based on the Bluetooth clock of the tester. The value of $CLK_{27\text{-}1}$ (i.e. not using CLK_0, representing half slots) modulo 5 is used to determine the transmit frequency.

2.1.4 Control of Transmit Parameters

The following parameters can be set to configure the transmitter test:

1. Bit pattern:
   - Constant zero
   - Constant one
   - Alternating 1010...[4]
   - Alternating 1111 0000 1111 0000...[4]
   - Pseudorandom bit pattern
   - Transmission off
2. Frequency selection:
   - Single frequency
   - Hopping Europe/USA
   - Hopping Japan
   - Hopping France
   - Hopping Spain
   - Reduced Hopping (implementation in Bluetooth devices and modules is optional)
3. TX frequency
   - $k \Rightarrow f := (2402 + k)$ MHz

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

4. It is recommended that the sequence starts with a one; but, as this is irrelevant for measurements, it is also allowed to start with a zero.

4. Default poll period in TDD frames (n * 1.25 ms)
5. Packet Type
6. Length of Test Sequence (user data of packet definition in Baseband Specification" on page 33.)

2.1.5 Power Control

If adaptive power control is tested, the normal LMP commands will be used. The DUT starts to transmit at the maximum power and reduces/increases its power by one step on every command received.

2.1.6 Switch between different Frequency Settings

A change in the frequency selection becomes effective when the LMP procedure is completed:

The tester switches to a new frequency or hopping pattern after the LMP_Accepted message has been received.

The DUT switches after the LMP_accepted message has been sent.

Note: Loss of the LMP_Accepted packet will eventually lead to a loss of frequency synchronization that cannot be recovered. Similar problems occur in normal operation, when the hopping pattern changes.

2.2 LOOPBACK TEST

The device under test receives normal baseband packets. The received packets are decoded in the DUT, and the payload is sent back using the same packet type. The return packet is sent back in either the TX slot directly following the transmission of the tester, or it is delayed and sent back in the slot after the next transmission of the tester (see Figure 2.7 to Figure 2.9 on page 815).

Alternatively, it is possible to implement a delayed loopback instead. Then the return packet is delayed to the following TX slot. There is no signalling to determine or control the mode. The device behavior must be fixed or adjusted by other means, but must not change randomly.

The tester can select, whether whitening is on or off. This setting holds for both up- and downlink. For switching the whitening status, the same rules as in Section 2.1 on page 808 (Figure 2.3) apply.

The following rules apply (for illustration see Figure 2.6 on page 814):

- Clearly, if the synch word was not detected, there will be no reply.
- If the header error check (HEC) fails, the DUT replies with a NULL packet with the ARQN bit set to NAK. It is not mandatory to return a NULL packet in this case; the DUT may send nothing.

- If the packet contains an LMP message relating to the control of the test mode this command is executed and the packet is not returned, though ACK or NAK is still returned as usual procedure. Other LMP commands are ignored and no packet is returned.
- The payload FEC is decoded and the payload is coded again for transmission. This allows testing of the FEC handling. If the pure bit error rate shall be determined the tester chooses a packet type without FEC.
- The CRC is evaluated. In case of a failure, the payload is returned with ARQN = NAK. The CRC for the return packet is calculated for the returned payload.
- If the CRC fails the number of bytes as indicated in the (possibly erroneous) payload header shall be looped back.

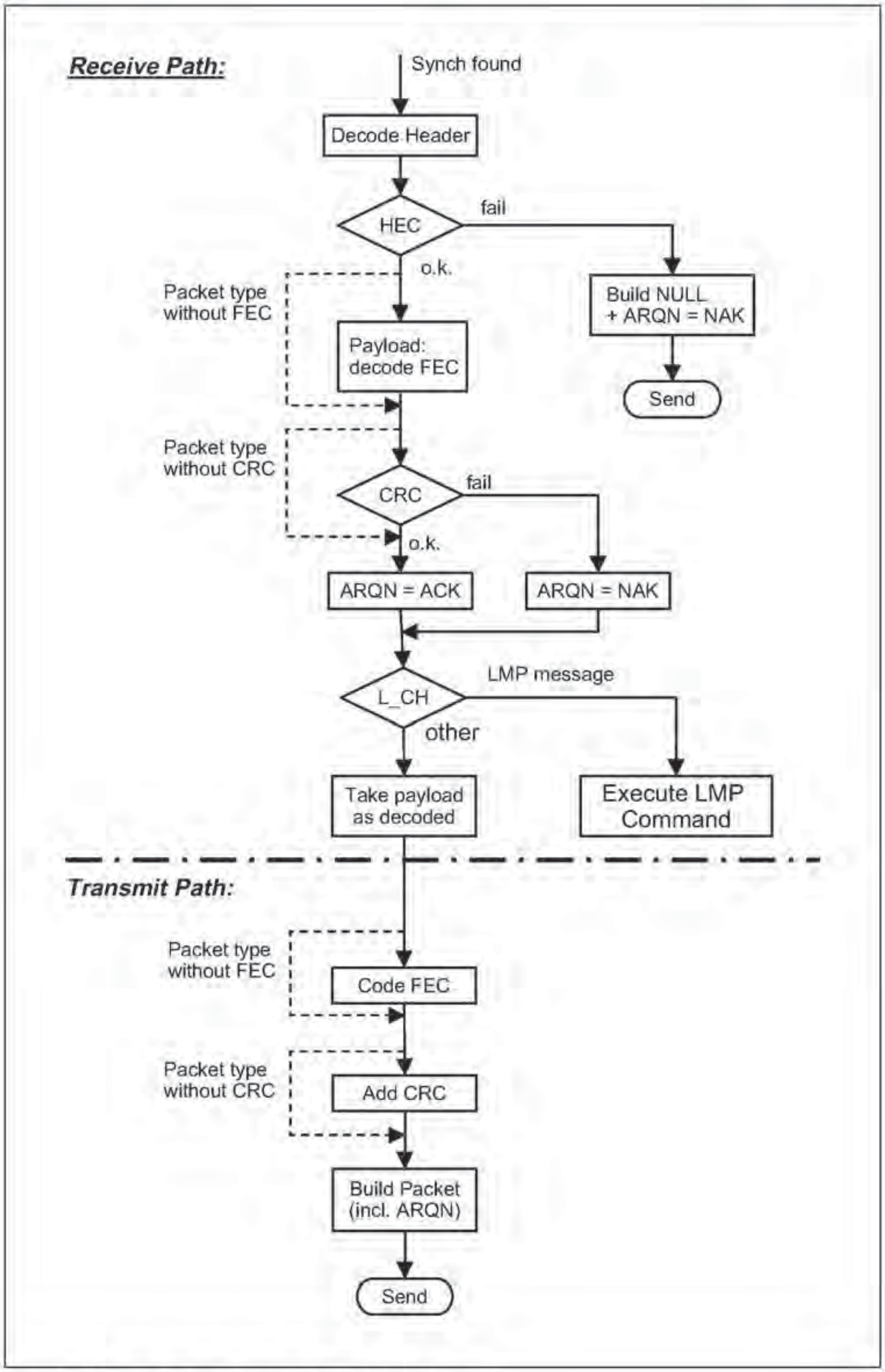


Figure 2.6: DUT Packet Handling in Loop Back Test

The timing for normal and delayed loopback is illustrated in Figure 2.7 to Figure 2.9:

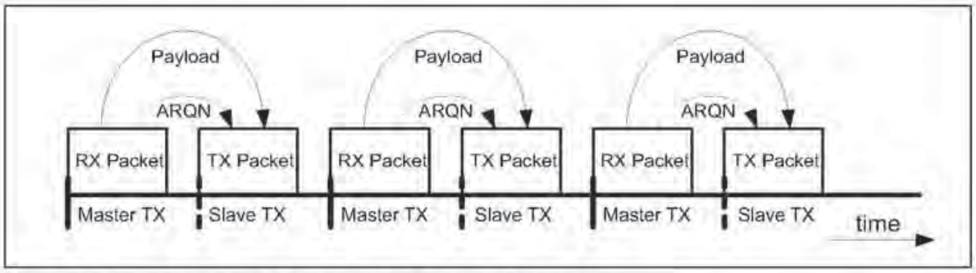


Figure 2.7: Payload & ARQN handling in normal loopback.

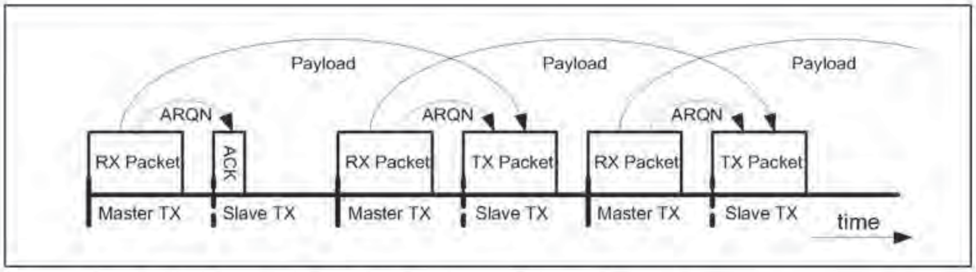


Figure 2.8: Payload & ARQN handling in delayed loopback - start.

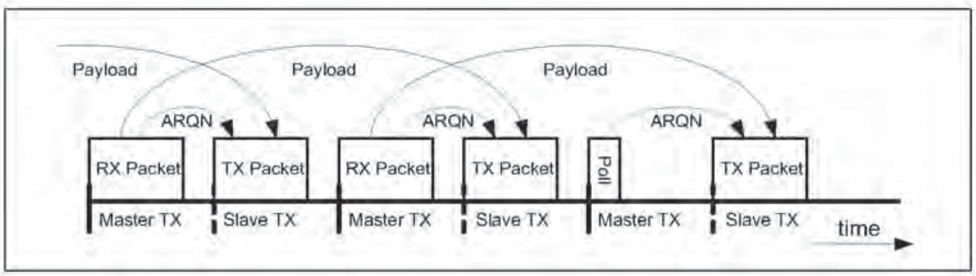


Figure 2.9: Payload & ARQN handling in delayed loopback - end.

The whitening is performed in the same way as it is used in normal active mode.

The following parameters can be set to configure the loop back test:

1. Packet Class[5]
    - ACL Packets
    - SCO Packets
    - ACL Packets without whitening
    - SCO Packets without whitening

5. This is included because, in the future, the packet type numbering may not remain unambiguous.

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

Hopping reduced uses only five frequencies on which a sequential hopping is done on (channel: 0, 23, 46, 69 and 93 is used).

3. Power level: (To be used according radio specification requirements)
   - power control or fixed TX power

The switch of the frequency setting is done exactly as for the transmitter test (see Section 2.1.6 on page 812).

3 OUTLINE OF PROPOSED LMP MESSAGES

Table 3.1 lists all LMP messages used for test mode (see Link Manager Protocol, Section 6 on page 237).

LMP PDU	PDU number	Possible Direction	Contents	Position in Payload
LMP_test_activate	56	m → s		
LMP_test_control	57	m → s	test scenario hopping mode TX frequency RX frequency power control mode poll period packet type length of test data	2 3 4 5 6 7 8 9-10
LMP_detach	7	m → s		
LMP_accepted	3	m ← s		
LMP_not_accepted	4	m ← s		

Table 3.1: LMP messages used for Test Mode

Name	Length (bytes)	Type	Unit	Detailed
Test scenario	1	u_int8		0 Pause (TX off) 1 Transmitter test – 0 pattern 2 Transmitter test – 1 pattern 3 Transmitter test – 1010 pattern 4 Pseudorandom bit sequence 5 Closed Loop Back – ACL packets 6 Closed Loop Back – SCO packets 7 ACL Packets without whitening 8 SCO Packets without whitening 9 Transmitter test – 1111 0000 pattern 10–254 reserved 255 Exit Test Mode
Hopping mode	1	u_int8		0 RX/TX on single frequency 1 Hopping Europe/USA 2 Hopping Japan 3 Hopping France 4 Hopping Spain 5 Reduced Hopping (optional) 6–255 reserved
TX frequency (for DUT)	1	u_int8		f = [2402 + k] MHz

Table 3.2: Parameters used in LMP_Test_Control PDU

Name	Length (bytes)	Type	Unit	Detailed
RX frequency (for DUT)	1	u_int8		f = [2402 + k] MHz
Power control mode	1	u_int8		0 fixed TX output power 1 adaptive power control
Poll period	1	u_int8	1.25 ms	
Packet type	1	u_int8		numbering as in packet header, see Baseband Specification)
length of test sequence (=length of user data in Baseband Specification)	2	u_int16	1 byte	unsigned binary number

Table 3.2: Parameters used in LMP_Test_Control PDU

The control PDU is used for both transmitter and loop back tests. The following restrictions apply for the parameter settings:

Parameter	Restrictions Transmitter Test	Restrictions Loopback Test
TX frequency	$0 \leq k \leq 93$	$0 \leq k \leq 93$
RX frequency	same as TX frequency	$0 \leq k \leq 93$
Poll period		not applicable (set to 0)
Length of test sequence	depends on packet type: DH1: $\leq$ 28 byte DH3: $\leq$ 181 byte DH5: $\leq$ 339 byte AUX1: $\leq$ 29 Byte HV3: = 30 byte	not applicable (set to 0)

Table 3.3: Restrictions for Parameters used in LMP_Test_Control PDU

4 REFERENCES

[1] Bluetooth Link Manager Protocol.

[2] CCITT Recommendation O.153 (1992), Basic parameters for the measurement of error performance at bit rates below the primary rate.

[3] ITU-T Recommendation O.150 (1996), General requirements for instrumentation for performance measurements on digital transmission equipment.

[4] Bluetooth Baseband Specification.

Part I:2

BLUETOOTH COMPLIANCE REQUIREMENTS

This document specifies the requirements for Bluetooth compliance.

Bluetooth.

CONTENTS

1 SCOPE

The Bluetooth Promoters and the Bluetooth Adopters have signed the Promoters' Agreement and the Adopters' Agreement respectively. These agreements grant Promoters and Adopters a Bluetooth license for "products which comply with the Specification".

This document specifies the requirements which must be met by a Promoter or Adopter to demonstrate that a particular product does "comply with the Specification", thereby qualifying that particular product to be subject to the rights extended by the Promoters' and Adopters' Agreements respectively.

The Bluetooth Qualification Program is the process by which a Promoter or an Adopter demonstrates that a particular product meets the requirements specified herein. This document provides an overview of the requirements and the Bluetooth Qualification Program. Further details are available through the Bluetooth Web site.

Regulatory requirements and governmental type approval requirements are outside the scope of this document.

2 TERMS USED

Bluetooth Trademark – As defined in the Promoters' Agreement and the Adopters' Agreement.

Bluetooth Brand – Covers all the brand elements specified in the "The Bluetooth Brand Book". Equal to the Bluetooth Trademark.

Bluetooth Logo or *Logo* – the brand element referred to as the 'figure mark' in the 'The Bluetooth Brand Book'.

Bluetooth License – all the rights, defined in the Promoters' and Adopters' Agreements respectively, that are granted by compliance with the specification, i.e. the Bluetooth Patent License and the Bluetooth Brand License.

Bluetooth Patent License – the applicable parts of the Bluetooth license consisting of patent rights or parts thereof as defined in the Promoters' and Adopters' Agreements respectively.

Bluetooth Brand License – the applicable parts of the Bluetooth license consisting of trademark rights as defined in the Promoters' and Adopters' Agreements respectively.

Protocol specification – defines the communication between two peer devices at a certain layer.

Profile specification – defines the usage of (parts of) the protocol stack for a certain Bluetooth usage model.

Bluetooth qualification process – the rules and procedures by which the manufacturer demonstrates compliance to the Bluetooth specification.

Bluetooth qualification program – the implementation of the Bluetooth qualification process.

Bluetooth Qualification Review Board (BQRB) – responsible for managing, reviewing and improving the Bluetooth qualification program. The original Bluetooth SIG companies will appoint BQRB initial members.

Bluetooth Qualification Test Facility (BQTF) – a test facility that is officially authorized by BQRB to test Bluetooth products.

Bluetooth Qualification Body (BQB) – a specific person authorized by the BQRB to be responsible for checking declarations and documents against requirements, reviewing product test reports, and listing products on the official database of Bluetooth qualified products.

Bluetooth Qualification Administrator (BQA) – a person responsible for administering the Bluetooth Qualification Program on behalf of BQRB.

Implementation Conformance Statement (ICS) – a document that the manufacturer attaches to the product when submitting it for qualification. It specifies all the implemented Bluetooth capabilities in detail.

Bluetooth Fellow Adopter – equal to Bluetooth Promoters + Bluetooth Adopters.

3 LEGAL ASPECTS

Rules and guidelines on how to use the Bluetooth Brand elements are stated in the document "The Bluetooth Brand Book" which is available on the Bluetooth Web site.

The Bluetooth Specification has been created, according to our best knowledge, to meet regulatory requirements worldwide. Regulatory certification as such is not a part of the Bluetooth qualification requirements, yet it is a requirement in all markets. It is the sole responsibility of each manufacturer to ensure that their products have all necessary regulatory approvals for the markets where their product(s) are intended to be sold or used.

A product must complete Bluetooth Qualification to meet the requirements for "complying with the Specification". The Bluetooth license granted by the Promoters' and Adopters' Agreements respectively is valid only for qualified products and is not transferable to other products.

In this document, the 'Bluetooth license' is sometimes divided into the 'Bluetooth patent license' and the 'Bluetooth brand license' for practical reasons. These terms correspond, respectively, to the terms 'necessary claims' and 'trademark' in the Promoters' and Adopters' Agreements respectively.

Sanctions will be invoked against any company responsible for producing or trading (a) products containing elements of the Bluetooth Interface, as defined in the Bluetooth Promoters' Agreement and Adopters' Agreement respectively, that do not comply with the Specification, or (b) products containing elements of the Bluetooth Interface that have not completed Bluetooth Qualification.

The Bluetooth SIG reserves the right to define a process for adding new Bluetooth profiles after the release of the Specification 1.0.

A Bluetooth brand license is granted by Ericsson to all Fellow Adopters for the use of the trademark in connection with products complying with the Specification.

Ericsson further provides Fellow Adopters a limited indemnity for costs and expenses incurred by the Fellow Adopter based upon the use of the trade mark within countries where Ericsson has registered the trademark. Ericsson does not take upon itself any liability regarding product, whether such liability is based on damages caused by the product for persons or property, or defects in the product itself.

4 THE VALUE OF THE BLUETOOTH BRAND

The purpose of this document is to define the requirements for Bluetooth compliance This has been done while bearing the basic Bluetooth philosophy in mind:

"Wireless Connections Made Easy"

Examples of important end-user experiences are:

- Reliable high-quality radio links,
- Interoperability between products of any brands,
- Easily understood product capabilities.

A reliable radio link experience depends upon all products demonstrating compliance with the Bluetooth radio link performance specifications. Interoperability is achieved by protocol and profile implementation conformance. Ease of use depends upon clear, consistent documentation of Bluetooth capabilities in product literature. All these elements are addressed in the requirements for Bluetooth compliance.

5 THE BLUETOOTH QUALIFICATION PROGRAM

This paragraph specifies the framework of the Bluetooth qualification program that a Bluetooth qualification applicant must perform. When completed, the full Bluetooth qualification program will be published at the Bluetooth web site.

The Bluetooth qualification program ('Program') establishes the rules and procedures by which the manufacturer demonstrates compliance to the Bluetooth specifications, and the process by which the Bluetooth license may be used by product manufacturers and distributors.

The Program defines the following entities:

- *Bluetooth Qualification Review Board (BQRB)* – responsible for managing, reviewing and improving the Bluetooth qualification program. The original Bluetooth SIG companies will appoint BQRB initial members.
- *Bluetooth Qualification Administrator (BQA)* – responsible for administering the Bluetooth Qualification Program on behalf of BQRB.
- *Bluetooth Qualification Test Facility (BQTF)* – a test facility that is officially authorized by BQRB to test Bluetooth products.
- *Bluetooth Qualification Body (BQB)* – a specific person authorized by the BQRB to be responsible for checking declarations and documents against requirements, reviewing product test reports, and listing products on the official database of Bluetooth qualified products.

Functions and relationships are illustrated in Figure 5.1.

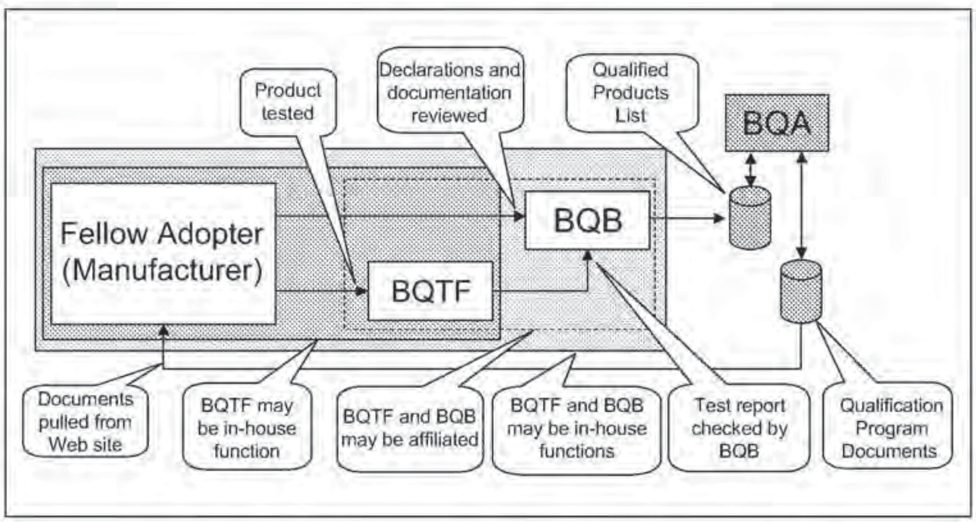


Figure 5.1: Bluetooth Qualification Process

Summary of the Qualification process:

1. The Fellow Adopter submits the product for Bluetooth qualification to a BQTF. The manufacturer must add temporary interfaces and/or functionality so that all implemented Bluetooth capabilities can be tested. The BQTF is not responsible for providing any secondary systems such as a LAN, PSTN or GSM network to facilitate the testing. Necessary documentation shall be provided; e.g. product description, user's manual and the Implementation Conformance Statement (a template for this document will be available at the Bluetooth web site). The BQTF tests each Bluetooth feature declared in the Implementation Conformance Statement according to the current Test Specification and BQRB policies and prepares a test report.
2. The test results and product documentation are then sent to the BQB. The Fellow Adopter sends an application to the BQB requesting that the product be listed as 'Bluetooth Qualified'. The application shall contain –
    a) Precise product description, and
    b) Declaration of Compliance with the Bluetooth Specification (including this entire document) and the Bluetooth Brand Book, signed by a duly authorized official of the Fellow Adopter.
3. When the application is complete, the BQB issues a Qualified Product notice and (with the applicant's permission) lists the product on the official Bluetooth Qualified products databases which can be viewed by all Fellow Adopters.

The BQTF may either be a third-party test house or an internal function of the applying Fellow Adopter. Also the BQB can be either internal or external. Both the BQTF and the BQB must always be authorized by the BQRB.

It is the responsibility of the manufacturer to establish any necessary non-disclosure agreements with BQTF, BQB and (if required) BQA. In the event the BQB cannot determine compliance the BQB may, with the applicant's permission, submit information to the BQA for a ruling. In the event the BQRB must be consulted, the applicant will be requested to prepare a submission according to BQRB guidelines.

The Fellow Adopter will be invoiced directly from BQTF and BQB for their respective services and expenses. The BQRB will also charge a fee to finance the administration associated with the Qualification program. Initially, this fee will be $3000 per listed product. It will subsequently be adjusted once per year, to reflect the actual cost.

6 BLUETOOTH LICENSE REQUIREMENTS FOR PRODUCTS

This section summarizes the product requirements that must be met to complete a Bluetooth Qualification.

The product requirements are divided into:

- Bluetooth radio link requirements
- Bluetooth protocol requirements
- Bluetooth profile requirements
- Bluetooth information requirements

6.1 BLUETOOTH RADIO LINK REQUIREMENTS

6.1.1 Requirement description

The Bluetooth radio link shall meet certain minimum requirements, which are documented in the Test Specification. This is to establish and maintain the Bluetooth technology as the preferred choice for wireless short-range links. The Test Specification for the Bluetooth radio link requirements will be based on the Bluetooth specification Part A (Radio specification).

6.1.2 Qualification

The BQRB will issue a list of BQTFs that are allowed to qualify products against the Bluetooth performance requirements.

6.2 BLUETOOTH PROTOCOL REQUIREMENTS

The implementation of the lower layers of the Bluetooth protocol stack shall meet certain minimum requirements, which are documented in the Test Specification. In order to verify that these requirements are met, individual testing of these protocols will be performed. The verification will be done by accessing the upper interface of these protocols through the Bluetooth Test Control Interface, TCI. How this test control interface will be used during verification is described in the Test Specification.

The Test Specification for the Bluetooth protocol requirements will be based on the Bluetooth specification Part B, C, D and H (Base band, Link Manager, Logical Link Control and Adaptation and, if applicable, the Host Controller Interface).

6.2.1 Qualification

The BQRB will issue a list of BQTFs that are allowed to qualify products against the Bluetooth protocol requirements.

The manufacturer is allowed to modify both the HW and SW of the product, to make it possible to perform the protocol tests. If this is done, the manufacturer must guarantee that an identical implementation of Bluetooth specification Part B, C, D and H (Base band, Link Manager, Logical Link Control and Adaptation and, if applicable, the Host Controller Interface) is used in the real product.

6.3 BLUETOOTH PROFILE REQUIREMENTS

6.3.1 Requirement description

The Bluetooth products shall meet certain minimum Bluetooth profile requirements which, for each profile, is defined in the Test Specification. This is to ensure that the end user can benefit from interoperability between different products and brands. The Test Specification for the Bluetooth profile requirements will be based on the Bluetooth specification Part K (Profile specifications).

The following general Bluetooth profile requirements must always be met:

- The "Generic Access" profile must be complied with.
- All implemented Bluetooth services must be described in the "Implementation Conformance Statement".
- All Bluetooth profiles declared in the "Implementation Conformance Statement" must be implemented according to each profile specification.
- All mandatory features of a Bluetooth profile role shall be implemented. All implemented optional Bluetooth features of a profile role shall be implemented according to the profile specification.
- If a service, for which there exists a Bluetooth profile, shall be implemented, it must be done according to that profile. It is permitted to make improvements or add features to a profile, as long as interoperability is maintained with other products that have implemented the standard profile as described in the previous paragraph. Improvements or new features can only be activated after proper negotiation between two Bluetooth devices.

Notification: A Fellow Adopter that wants to implement a new service, for which there is no sufficient standardized Bluetooth profile specification available, is allowed to do so. However, this new service must never be referred to in a way that it could mistakenly be interpreted as being a standard Bluetooth profile and part of the Bluetooth specification. The manufacturer must inform the market as well as the end user in a clear and consistent way about these limitations in general interoperability.

6.3.2 Qualification

The BQRB will issue a list of BQTF that are allowed to qualify products against the Bluetooth profile requirements.

6.4 BLUETOOTH INFORMATION REQUIREMENTS

6.4.1 Requirement description

The manufacturer shall inform the market and end users in a clear and consistent way about the implemented Bluetooth capabilities.

6.4.2 Qualification

The product will be qualified against the Bluetooth information requirements.

6.5 REQUIREMENTS ON BLUETOOTH ACCESSORY PRODUCTS

6.5.1 Definition of 'Bluetooth accessory products'

A Bluetooth accessory product is defined as "A product marketed to the end user, containing at least the hardware for the Bluetooth radio and baseband, yet not being a stand-alone Bluetooth product. After being installed in a host system, the product acts like a complete Bluetooth product." Examples of Bluetooth accessory products: PC-Cards, serial port dongles, USB dongles.

Bluetooth accessory products must also pass through the complete Bluetooth qualification process. To facilitate testing, the Bluetooth accessory product and the provided Bluetooth SW will be installed in a host device that is provided by the manufacturer.

6.5.2 Qualification

Same as in Section 6.1 - Section 6.4 above.

6.6 REQUIREMENTS ON BLUETOOTH COMPONENTS

6.6.1 Definition of "Bluetooth components"

A Bluetooth component is defined as "A component product designed and marketed for the enabling of a complete Bluetooth product, which component product containing at least a subset of an existing Bluetooth Profile (see Section 6.3 on page 833), yet not being able to function as a complete Bluetooth product". For example, a Bluetooth component might be a complete module designed for integration on a PC board, or an integrated circuit implementing all Bluetooth baseband and protocol functions.

A Bluetooth component is typically purchased and integrated by an original equipment manufacturer (OEM) into a product designed for sale to an end user.

A Bluetooth component manufacturer will typically obtain a limited Bluetooth License enabling the manufacturer to identify the component's Bluetooth capabilities. A component manufacturer may also wish to minimize their OEM customer's qualification testing requirements through one-time qualification testing of a reference design based on the component. Qualifying a component is not necessary as long as the final product is qualified. The possibility has been created only to ease the marketing of Bluetooth components.

6.6.2 Requirement description

Bluetooth components must pass through the complete Bluetooth qualification process in a reference design configuration documented in an application note.

Bluetooth products incorporating a limitedly licensed Bluetooth component must also pass through the complete Bluetooth qualification process. However, certain tests may be waived if so indicated in the limited Bluetooth License valid for the component.

A component's limited Bluetooth License identifies specific qualification tests that may be considered pre-qualified by an OEM manufacturer using the component in an end-user product. Those specific qualification tests are identified by the BQTF, which performs qualification testing of the Bluetooth component in its reference design. BQTF identifies those tests based on the unique design characteristics of the component in consultation with the manufacturer.

6.6.3 Qualification

A product which includes an integrated Bluetooth component must be qualified as described in Section 6.1 - Section 6.4 above, possibly with some of the tests waived if so indicated in the limited Bluetooth License valid for the component.

7 BLUETOOTH LICENSE PROVISIONS FOR EARLY PRODUCTS

The process and conditions for qualifying early products (that may contain reasonable deviations from the Bluetooth specification) will be defined and published on the Bluetooth web site.

8 BLUETOOTH BRAND LICENSE PROVISIONS FOR SPECIAL PRODUCTS & MARKETING

This section defines the requirements for using the Bluetooth Brand elements for special products and marketing.

8.1 BLUETOOTH DEVELOPMENT TOOLS AND DEMOS

8.1.1 Definition of 'Bluetooth Development tools and demos'

Bluetooth Development tools and demos are products intended either for developing commercial Bluetooth products or for demonstrating the Bluetooth technology in a certain application. Neither one may be sold to ordinary consumers.

8.1.2 Requirement description

The manufacturer and/or seller of these products shall clearly inform the targeted audience/customer that the products are for development and/or demonstration purpose only, and that they have not been qualified to the Bluetooth specification.

8.1.3 Qualification

Qualification testing by a BQTF is not required. Qualification is based upon the applicant's declaration of compliance with the Specification and Brand Book.

8.2 MARKETING

The Bluetooth Brand elements may be used for general marketing and product announcements. The rules of the Bluetooth Brand Book must be followed.

If the Bluetooth Brand is used on a give-away item, where it is not obvious to everyone that the product doesn't contain a Bluetooth radio, then a clear disclaimer has to be displayed on the product (e.g. a give-away calculator with the Bluetooth brand must have a visible disclaimer since the idea of calculators with in-built Bluetooth actually makes sense).

9 RECOMMENDATIONS CONCERNING INFORMATION ABOUT A PRODUCT'S BLUETOOTH CAPABILITIES

In addition to the requirements set forth in the Brand Book, it is recommended that at least the following pieces of information are provided for the market and the end-user:

- The Bluetooth capabilities of the product should be stated, at least in brief, on the product box.
- The user's manual (or corresponding information) should contain a section where all the Bluetooth capabilities are described. A list of qualified standard profiles using the profile names listed in Appendix B should be contained. If applicable, revision numbers of the implemented profiles shall be included. For early products a list of interoperable products instead of profiles should be contained.

Important places for end-user information are user's manuals (user guides), leaflets, boxes and other advertisement material.

An example of information in a user's manual can be found in Appendix A on page 840.

It is important that new profiles, not sanctioned by the Bluetooth SIG, cannot be mistaken for profiles contained in the Bluetooth specification. In case of new profiles it is important that the manufacturer inform the market and the end user about what other *products* that interoperability can be expected with.

10 QUALITY MANAGEMENT, CONFIGURATION MANAGEMENT AND VERSION CONTROL

Each manufacturer is responsible for keeping a high quality level when mass-producing an approved product. Products that are put on the market shall meet the requirements for which the product has been qualified.

A Bluetooth Qualification covers specific product hardware and software versions. The product's manufacturer is responsible for ensuring that all production units perform identically to the qualified version, by maintaining appropriate quality management and configuration management programs.

Major hardware or software modifications related to the Bluetooth part of a qualified product, shall be documented and submitted to the BQB for review. Based on the manufacturer's representations, the BQB may certify that the product requires no further testing and allow the license to be updated to include the new version. In other cases, the BQB may identify a limited subset of tests that must be performed by a BQTF to qualify the new version.

Addition of Bluetooth capabilities requires a new qualification of the product.

11 APPENDIX A – EXAMPLE OF A "BLUETOOTH CAPABILITY STATEMENT"

This is an example of a Bluetooth Capability Statement in a User's Manual.

....

Bluetooth Capability Statement

This product is manufactured to meet the Bluetooth Specification 1.0. The following Bluetooth functions are supported:

- Service Discovery
- Cordless Telephony
- Local Telephony
- Headset

....

The profiles normally use roles. In most cases it is obvious which role a certain product has implemented. Where doubt or misunderstanding could arise, the implemented role shall be explicitly stated after the profile name.

12 APPENDIX B - MARKETING NAMES OF BLUETOOTH PROFILES

Bluetooth uses profiles to ensure interoperability between products and brands. The profile specifications are technical documents. In the marketing communication it is strongly recommended to use the names listed in Table 12.1.

Profile name	Marketing name	Comments
Generic Access	Generic Access	Mandatory
Service Discovery	Service Discovery	
Cordless Telephony	Cordless Telephony	
Intercom	Local Telephony	
Headset	Headset	
Speaker phone	Speakerphone	
Dial-up networking	Modem	
Fax	Fax	
LAN Access	Network access point	
Conferencing	Conferencing	
Serial port	Serial port	
Generic Object Exchange	Object Exchange	
Object Push	Object push	
File Transfer and Browsing	Data sharing	
Synchronization	Synchronization	

Table 12.1: Marketing names for Bluetooth profiles

Part I:3

TEST CONTROL INTERFACE

This document describes the Bluetooth Test Control Interface (TCI). The TCI provides a uniform method of accessing the upper interface of the System Under Test (SUT) and/or the Implementation Under Test (IUT).

Bluetooth.

CONTENTS

1 INTRODUCTION

1.1 TERMS USED

IUT = Implementation Under Test: An implementation of one or more OSI protocols in an adjacent user/provider relationship, being that part of a real open system which is to be studied by testing.
This term will be used when describing the test concept for Bluetooth accessory products and Bluetooth components. The definition of Bluetooth accessory products and Bluetooth components can be found in Part I:2 / Section 6.5.1 on page 834 and in Part I:2 / Section 6.6.1 on page 834.

SUT = System Under Test: The real open system in which the IUT resides. This term will be used when describing the test concept for Bluetooth products.

TCI = Test Control Interface: The interface and protocol used by the tester to send and receive commands and messages to and from the upper interface of the SUT/IUT.

1.2 THE NEEDS FOR A UNIFIED TEST INTERFACE

For all Bluetooth accessory products, Bluetooth components and Bluetooth products, protocol testing will be used to verify the implemented functionality in the lowest layers; i.e. conformance testing.

For this type of testing, an upper tester (UT) will be required to completely test the implementation.

In order to shield the tester from having to adopt to each and every implementation of IUTs or SUTs, the use of a standardized control interface is mandated. This concept puts some extra burden upon the manufacturer of the IUT/SUT. The manufacturer must:

- adopt the implementation-dependent interface to the TCI
- supply, with the IUT, the adapter needed (can be HW, SW or FW)

1.3 USAGE OF THE INTERFACE

The Bluetooth Test Control Interface, TCI, will be used when verifying the Bluetooth protocol requirements for a Bluetooth accessory product, Bluetooth component or a Bluetooth product. More specifically, the TCI will be used when verifying implemented functionality of the:

- Baseband layer, BB (the protocol-related part)
- Link Manager Protocol, LMP
- Logical Link Control and Adaptation Protocol, L2CAP

and, if support of the HCI is claimed by the manufacturer:

- Host Control Interface, HCI

2 GENERAL DESCRIPTION

The interface used between the tester and the SUT/IUT will be either of two types:

1. TCI-HCI
   This interface is semantically and syntactically identical to the HCI interface described in "Part H:1" on page 517.
2. TCI-L2CAP
   This interface is based on the HCI interface, and will be used during verification of the L2CAP layer of the SUT/IUT.

The proposed physical bearer is one of the transport layers specified for the HCI: USB, RS232 or UART, see "Part H:2" on page 759, "Part H:3" on page 775 or "Part H:4" on page 795. However, alternatives do exist. More details will be given in the following sections.

2.1 BASEBAND AND LINK MANAGEMENT VERIFICATION

For the verification of the link control part of the Baseband layer and for the Link Manager layer, the TCI-HCI interface will be used as the interface between the test system and the upper interface of the SUT/IUT. The test system accesses the upper interface of the SUT/IUT by sending HCI commands and receiving HCI events from the SUT/IUT as described in the "Host Controller Interface Functional Specification" on page 517. The supported functionality on the TCI-HCI interface depends on the implemented functionality of the BB and LM layers.

The transport bearer used between the tester and the SUT/IUT can be of either of two types:

1. A physical bearer of one of the types USB, RS232 or UART, as defined in Part H:2, Part H:3 or Part H:4. It is recommended to use one of these three physical bearers as transport bearer between the SUT/IUT and the test system.
2. A 'software' transfer bearer; i.e. there is no physical connection between the tester and the SUT/IUT. In this case, the manufacturer of the SUT/IUT must supply, when sending in the device for testing, a test software that can be operated by a test operator. The operator will receive instructions from the tester and will execute them on the SUT/IUT. The software must support the same functionality as if using the TCI-HCI with a physical bearer. Use of the 'software' interface must be agreed upon between the manufacturer of the SUT/IUT and the test facility that will perform the verification. The test facilities can themselves specify requirements placed on such an interface.

A schematic example is shown in Figure 2.1 of a possible test configuration for BB and LM verification of Bluetooth products which do not support HCI, and which use a physical transport bearer for the TCI-HCI interface. In this figure, the TC (Test Control) Software represents what the manufacturer has to supply with the SUT/IUT when sending it in for verification. The functionality of the TC software is to adapt the implementation-dependent interface to the TCI-HCI interface.

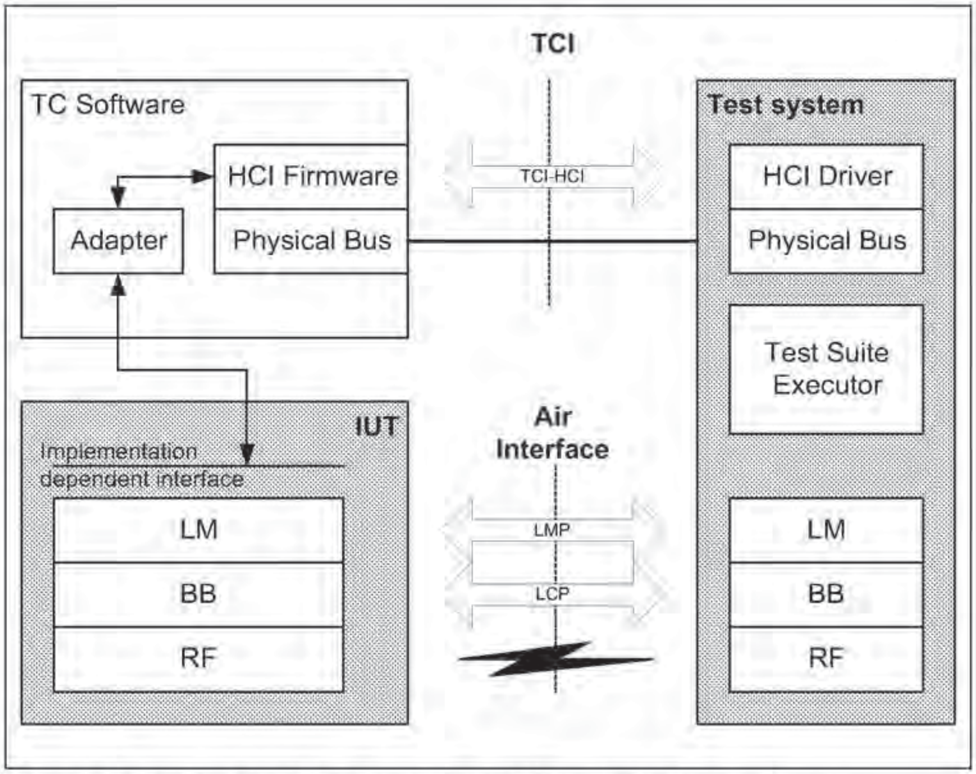


Figure 2.1: Baseband and LM verification without HCI – physical transport bearer

Figure 2.2 shows a schematic example of the test configuration for the same Bluetooth product using a 'software' transfer bearer for the TCI-HCI interface. Here, the role of the TC Software is to represent the application that can be controlled by the test operator.

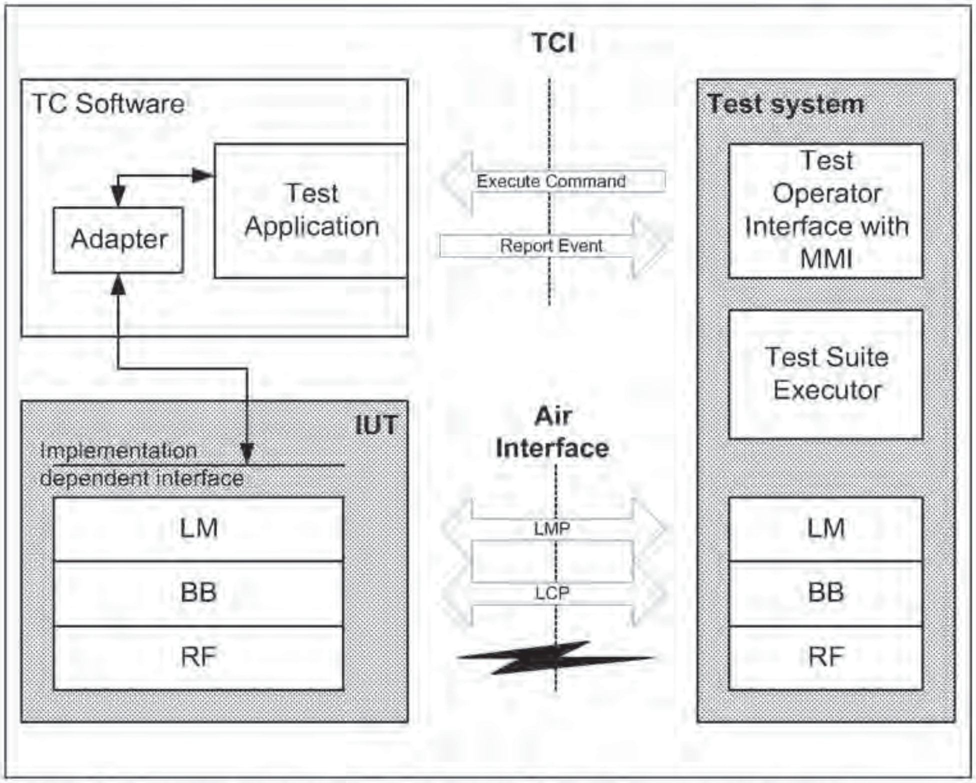


Figure 2.2: Baseband and LM verification without HCI – software transport bearer

2.2 HCI VERIFICATION

The TCI-HCI interface may also be used for HCI signalling verification. The HCI signalling will only be verified if support of the HCI functionality is claimed by the manufacturer.

The transport bearer between the tester and the SUT/IUT shall be one of the types USB, RS232, or UART, as defined in Part H:2, Part H:3 or Part H:4.

A schematic example is shown in Figure 2.3 of one possible test configuration for HCI verification of Bluetooth products, using a physical transport bearer for the TCI-HCI interface. As can be seen from the figure, no extra test control software is needed. Instead, the implemented HCI will be used as the interface to the tester.

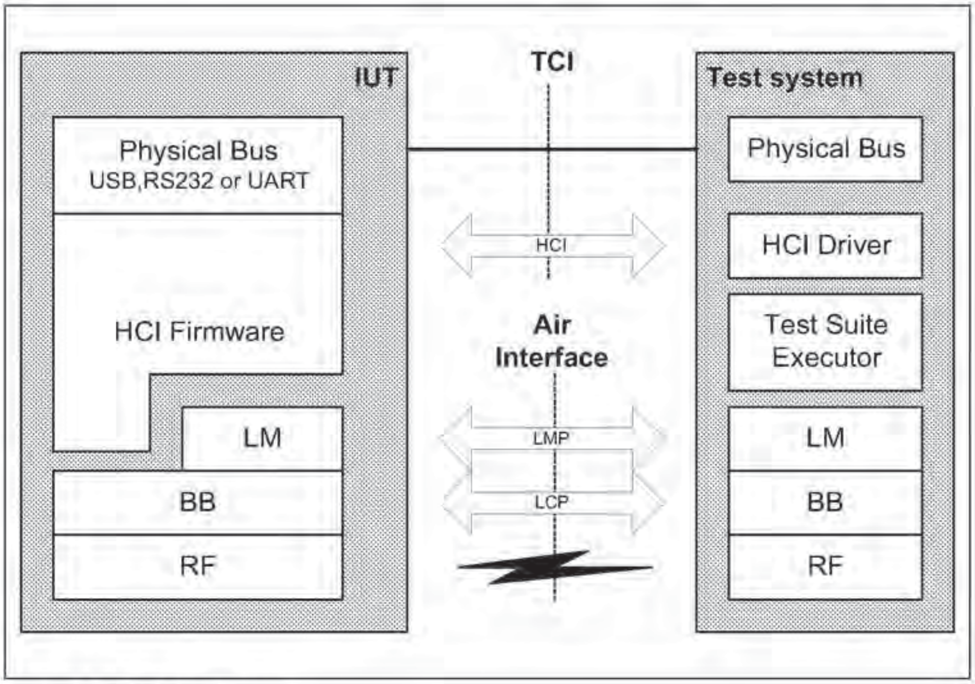


Figure 2.3: HCI verification

2.3 L2CAP VERIFICATION

The TCI-L2CAP interface is based on the HCI and will be used during verification of the L2CAP layer of the SUT/IUT. It uses the general event and command syntax as specified in Part H:1, and the mapping to transport layers is also identical to the ones defined in Part H:2, Part H:3 or Part H:4. Commands and events are defined according to the specified L2CAP service interface. See Part D / Section 7 on page 295.

The defined service primitives in the Logical Link and Control Layer specification, Part D / Section 7 on page 295, will be used as reference. However, the primitives for L2CAP events and commands must be converted into messages of the same format as used for the HCI events and commands. The mapping of the L2CAP events and commands to HCI format is described in Section 4 of this document.

A schematic example is shown in Figure 2.4. of how the test configuration can look for L2CAP verification of Bluetooth products, using a physical transport bearer for the TCI-L2CAP interface. In this figure, the TC (Test Control) Software represents what the manufacturer has to supply with the SUT/IUT when sending it in for verification. The functionality of the TC software is to adapt the implementation-dependent interface to the TCI-L2CAP interface.

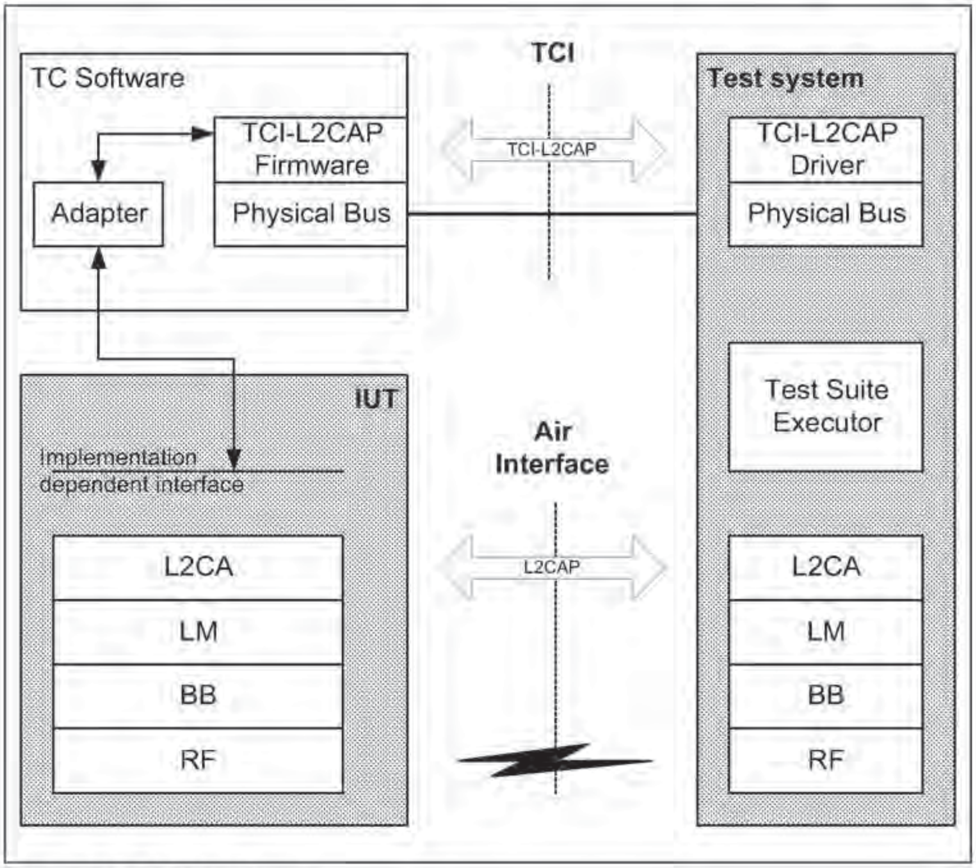


Figure 2.4: L2CAP verification

3 TEST CONFIGURATIONS

This section describes the test configurations that will be used when verifying the different Bluetooth requirements.

3.1 BLUETOOTH RF LINK REQUIREMENTS

For the verification of the Bluetooth RF Link requirements, the defined test mode will be used, see "Part I:1" on page 803.

The Test Specification for the Bluetooth radio link requirements will be based on the Bluetooth specification Parts A and B, and will contain the relevant test instructions that should be carried out on the SUT/IUT.

3.1.1 Required Interface(s)

For this type of verification, only the air interface is required. See Figure 3.1. As stated in Part I:1 / Section 1.2 on page 807, for security reasons, the test mode must be locally enabled. The implementation of this local enabling is not subject to standardization.

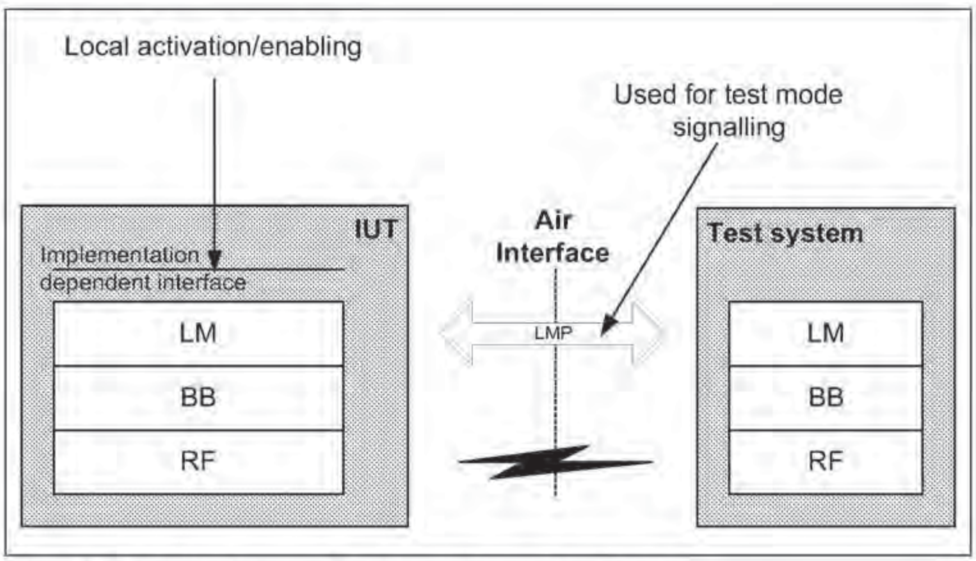


Figure 3.1: Test Configuration for RF link requirement verification

3.2 BLUETOOTH PROTOCOL REQUIREMENTS

Dependent on which Bluetooth layers BB, LM, HCI or L2CAP are implemented in the product sent in for verification, the amount of testing needed to verify the Bluetooth protocol requirements will differ. Also, the TCI used during the verification may be different.

The Test Specification for the Bluetooth protocol requirements will be based on the Bluetooth specification Part A to Part D and Part H, if applicable, and will contain the relevant test instructions that should be carried out on the SUT/IUT.

3.2.1 Required Interface(s)

For this type of verification, both the air interface of the SUT/IUT and the test control interface are required. The latter will be one of the types described in section 2.

3.3 BLUETOOTH PROFILE REQUIREMENTS

For each profile the Bluetooth product claims to conform to, profile testing will be performed to verify the Bluetooth profile requirements in order to ensure interoperability between products; i.e. interoperability testing.

The Test Specification for the Bluetooth profile requirements will be based on the Bluetooth specification Part K Volume 2, and will contain the relevant test instructions that should be carried out on the SUT.

3.3.1 Required Interface(s)

For this type of verification, both the air interface of the SUT and the supported MMI, as described in the profile, will be used during verification, see Figure 3.2.

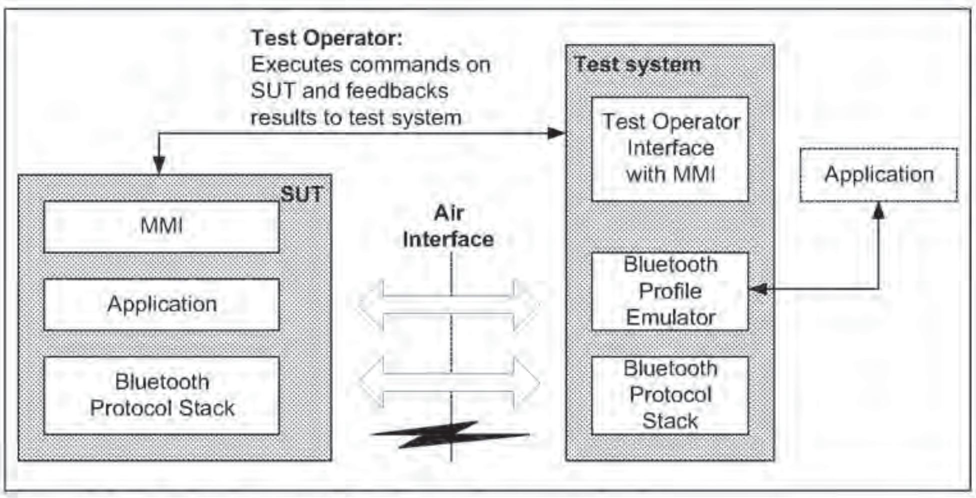


Figure 3.2: Test Configuration for Profile requirement verification

4 TCI-L2CAP SPECIFICATION

Note: This specification maps the L2CAP service interface to an appropriate TCI. This section is based on the 0.95b version of the L2CAP specification.

4.1 EVENTS

In the L2CAP service interface, indications are mapped to callback functions. The corresponding response parameters are submitted in the return parameter of these functions. For the TCI, the indications are mapped to events and the responses to commands.

A single event code is reserved for testing purposes: 0xFE. To distinguish the L2CAP events, a parameter 'Event_ID' is submitted as first parameter. This parameter is a single octet, resulting in 256 possible events. The assignment is given in Table 4.1.

Event_ID	L2CAP event
0x00	Reserved
0x01	L2CA_ConnectInd
0x02	L2CA_ConfigInd
0x03	L2CA_DisconnectInd
0x04	L2CA_QoSViolationInd
0x05 – 0xFF	Reserved

Table 4.1: Assignment of event IDs

The events in this test interface follow the HCI syntax as defined in Part H:1 / Section 4.4.2 on page 535.

4.1.1 Connect Indication

Event	Event Code	Event Parameters
L2CA_ConnectInd	0xFE	Event_ID, BD_ADDR, CID, PSM, Identifier

For more details and the event parameter, see Part D / Section 7.1 on page 295.

4.1.2 Configuration Indication

Event	Event Code	Event Parameters
L2CA_ConfigInd	0xFE	Event_ID, CID, OutMTU, InFlow, FlushTO

For more details and the event parameter, see Part D / Section 7.1 on page 295.

4.1.3 Disconnect Indication

Event	Event Code	Event Parameters
L2CA_DisconnectInd	0xFE	Event_ID, CID

For more details and the event parameter, see Part D / Section 7.1 on page 295.

4.1.4 Violation Indication

Event	Event Code	Event Parameters
L2CA_QoSViolationInd	0xFE	Event_ID, BD_ADDR

For more details and the event parameter, see Part D / Section 7.1 on page 295.

4.2 COMMANDS

The commands in this test interface follow the HCI syntax as defined in Part H:1 / Section 4.4.1 on page 532. The return parameters are sent back using a Command Complete event, see Part H:1 / Section 5.2.14 on page 723 and Part H:1 / Section 4.4.1 on page 532.

To distinguish the commands used for L2CAP testing from HCI commands, a single subgroup is reserved for the L2CAP test interface. Figure 4.1 shows how to code and decode the OpCode field in the HCI command packet used for testing.

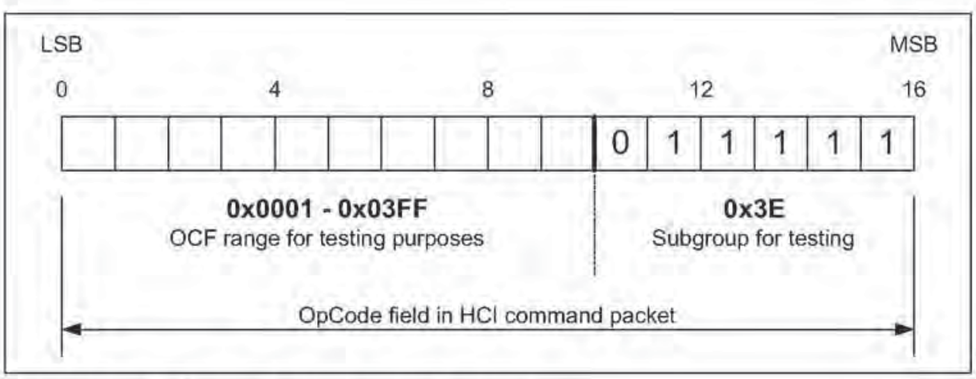


Figure 4.1: HCI Opcode field values used for testing

The assignment of the OpCode Command Field, OCF, for the L2CAP commands is summarized in Table 4.2. It is also detailed following the table, in a format similar to the HCI specification, see "Part H:1" on page 517.

OCF	L2CAP command
0x0000	Reserved
0x0001	L2CA_ConnectReq
0x0002	L2CA_DisconnectReq
0x0003	L2CA_ConfigReq
0x0004	L2CA_DisableCLT
0x0005	L2CA_EnableCLT
0x0006	L2CA_GroupCreate
0x0007	L2CA_GroupClose
0x0008	L2CA_GroupAddMember
0x0009	L2CA_GroupRemoveMember
0x000A	L2CA_GroupMemebership
0x000B	L2CA_WriteData
0x000C	L2CA_ReadData
0x000D	L2CA_Ping
0x000E	L2CA_GetInfo
0x000F	Reserved
0x0010	Reserved
0x0011	L2CA_ConnectRsp
0x0012	Reserved
0x0013	L2CA_ConfigRsp
0x0014 – 0x03FF	Reserved

Table 4.2: Assignment of Opcode Command Field values

4.2.1 Connection Establishment

Command	OCF	Command Parameters	Return Parameters
L2CA_ConnectReq	0x0001	PSM, BD_ADDR	LCID, Result, Status

Description:

Requests the creation of a channel representing a logical connection to a physical address (for more details and input/output parameter definition see Part D / Section 7.2 on page 296).

4.2.2 Connect Response

Command	OCF	Command Parameters	Return Parameters
L2CA_ConnectRsp	0x0011	BD_ADDR, Identifier, LCID, Response, Status	Result

Description:

Issues a response to a connection request event indication (for more details and input/output parameters definition see Part D / Section 7.3 on page 298)

4.2.3 Connection Release (Disconnect)

Command	OCF	Command Parameters	Return Parameters
L2CA_DisconnectReq	0x0002	CID	Result

Description:

Requests the disconnection of the channel. Input parameter is the *CID* representing the local channel endpoint (for more details and input/output parameter definition see Part D / Section 7.6 on page 302).

4.2.4 Configuration

Command	OCF	Command Parameters	Return Parameters
L2CA_ConfigReq	0x0003	CID, InMTU, OutFlow, FlushTO, LinkTO	Result, InMTU, OutFlow, FlushTO

Description:

Requests the initial or new configuration of a channel to a new set of channel parameters (for more details and input/output parameter definition see Part D / Section 7.4 on page 299).

4.2.5 Configure Response

Command	OCF	Command Parameters	Return Parameters
L2CA_ConfigRsp	0x0013	CID, OutMTU, InFlow	Result

Description:

Issues a response to a configuration request event indication (for more details and input/output parameter definition see Part D / Section 7.5 on page 301).

4.2.6 Disable Connectionless Traffic

Command	OCF	Command Parameters	Return Parameters
L2CA_DisableCLT	0x0004	N, List of PSMs	Result

Description:

For details and input/output parameter definition see Part D / Section 7.16 on page 311.

4.2.7 Enable Connectionless Traffic

Command	OCF	Command Parameters	Return Parameters
L2CA_EnableCLT	0x0005	N, List of PSMs	Result

Description:

For details and input/output parameter definition see Part D / Section 7.17 on page 312.

4.2.8 Group Create

Command	OCF	Command Parameters	Return Parameters
L2CA_GroupCreate	0x0006	PSM	CID

Description:

Request the creation of a channel identifier to represent a logical connection to multiple devices. On creation, the group is empty (for more details and input/output parameter definition see Part D / Section 7.9 on page 305).

4.2.9 Group Close

Command	OCF	Command Parameters	Return Parameters
L2CA_GroupClose	0x0007	CID	Result

Description:

This command closes down a Group (for more details and input/output parameter definition see Part D / Section 7.10 on page 305).

4.2.10 Group Add Member

Command	OCF	Command Parameters	Return Parameters
L2CA_GroupAddMember	0x0008	CID, BD_ADDR	Result

Description:

This command adds a member to the group (for more details and input/output parameter definition see Part D / Section 7.11 on page 306).

4.2.11 Group Remove Member

Command	OCF	Command Parameters	Return Parameters
L2CA_GroupRemoveMember	0x0009	CID, BD_ADDR	Result

Description:

Remove a member from the group (for more details and input/output parameter definition see Part D / Section 7.12 on page 307).

4.2.12 Group Membership

Command	OCF	Command Parameters	Return Parameters
L2CA_GroupMembership	0x000A	CID	Result, N, BD_ADDR_Lst

Description:

Get report of the members of the group (for more details and input/output parameter definition see Part D / Section 7.13 on page 308).

4.2.13 Ping

Command	OCF	Command Parameters	Return Parameters
L2CA_Ping	0x000D	BD_ADDR, ECHO_DATA	Result, ECHO_DATA

Description:

For more details and input/output parameter definition see Part D / Section 7.14 on page 309.

4.2.14 Get Info

Command	OCF	Command Parameters	Return Parameters
L2CA_GetInfo	0x000E	BD_ADDR, InfoType	Result, InfoData

Description:

For more details and input/output parameter definition see Part D / Section 7.15 on page 310.

4.3 DATA TRANSFER

Data transfer is modelled with read and write functions. Handling is like an L2CAP command.

To be able to send the amount of data that is needed to verify how the L2CAP implementation handles large chunks of data (i.e. segmentation and reassembly), and since it is not possible to use HCI Command packets as well as HCI Event packets to send the data, the use of HCI ACL Data packets will be used. The procedure/signalling used on the TCI-L2CAP interface to transfer data packets will be described with MSCs.

4.3.1 Write

Command	OCF	Command Parameters	Return Parameters
L2CA_WriteData	0x000B	CID, Length, OutBuffer	Size, Result

Description:

Parameters are the CID, the length of the data and the data itself. The data will be sent in a HCI ACL data packets as described in Figure 4.2. For more details and input/output parameter definition see Part D / Section 7.7 on page 303.

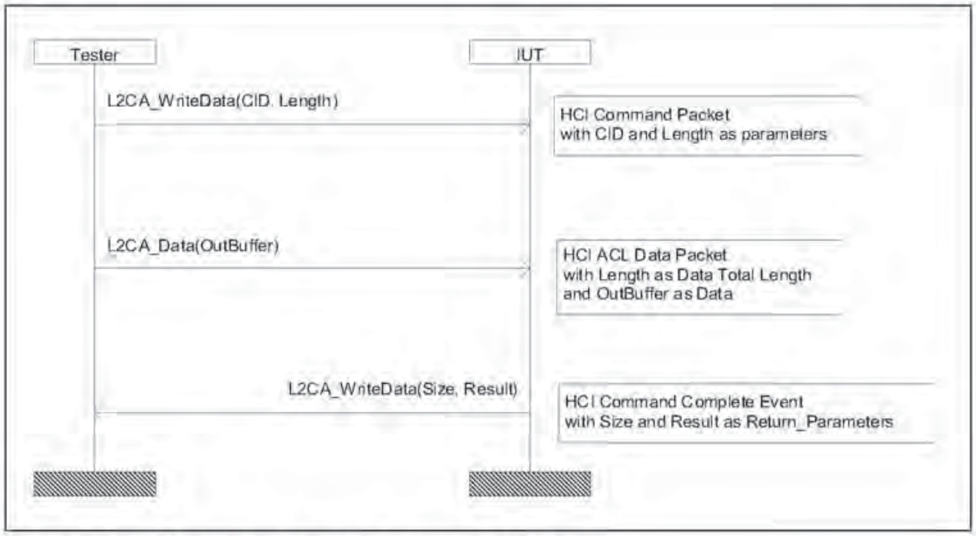


Figure 4.2: MSC showing how to write data to L2CAP

The L2CA_Data primitive is used as an abstract name for the data transmitted between the Tester and the IUT. The Tester will use Connection Handle 0x0001 for the data and will set the Flags-field to 0x02. The Data Total Length field will contain the length of the OutBuffer.

After the IUT has received the data, it shall send back an HCI Command Complete Event (named L2CAP_WriteData in the figure) with the N parameter set to 0x01, the OpCode parameter set to the corresponding OCF and subgroup (that is OCF = 0x000B and the subgroup = 0x3E). The Size and Result are sent in the Return_Parameters field of the HCI ACL Data packet.

4.3.2 Read

Command	OCF	Command Parameters	Return Parameters
L2CA_ReadData	0x000C	CID, Length, InBuffer	Result

Description:

Input parameter is the CID, length and the InBuffer. Output parameters are the result. The data will be sent in HCI ACL data packets as described in Figure 4.3. For more details and input/output parameter definition see Part D / Section 7.8 on page 304).

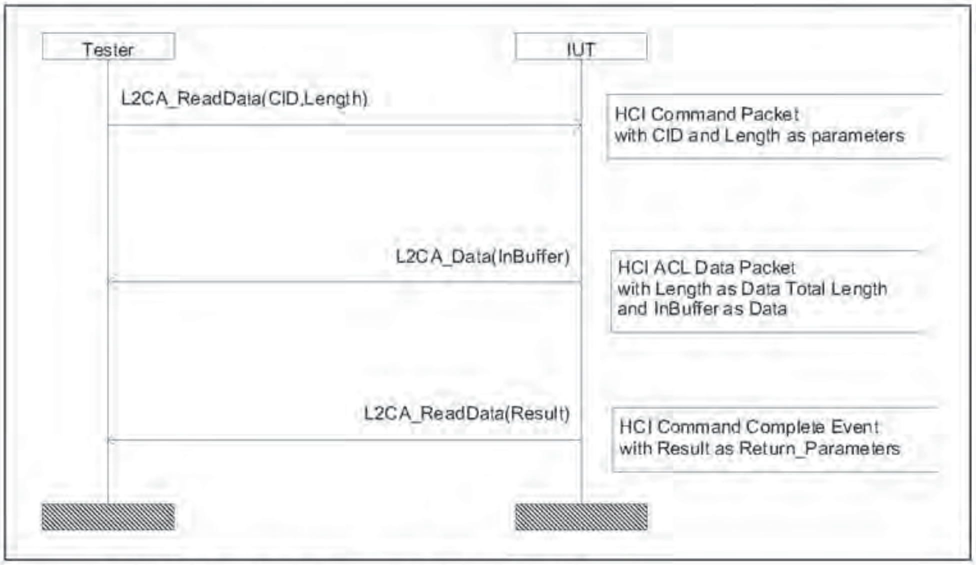


Figure 4.3: MSC showing how to read data from L2CAP

The L2CA_Data primitive is used as an abstract name for the data transmitted between the Tester and the IUT. The IUT shall use Connection Handle 0x0001 for the data and shall set the Flags-field to 0x02. The Data Total Length field shall contain the length of the InBuffer.

After the IUT has sent the data, it shall send back an HCI Command Complete Event (named L2CAP_ReadData in the figure) with the N parameter set to 0x01, the OpCode parameter set to the corresponding OCF and subgroup (that is OCF = 0x000C and the subgroup = 0x3E). The Size and Result are sent in the Return_Parameters field of the HCI ACL Data packet.

5 ABBREVIATIONS

BB	BaseBand (see LC)
FW	Firmware
HCI	Host Controller Interface
HW	Hardware
IUT	Implementation Under Test
L2CA	Logical Link Control And Management part of the Bluetooth protocol stack
L2CAP	Logical Link Control And Management Protocol
LC	Link Controller (or baseband) part of the Bluetooth protocol stack
LCP	Link Control Protocol
LM	Link Manager part of the Bluetooth Protocol Stack
LMP	Link Management Protocol
MMI	Man-Machine Interface
OCF	Opcode Command Field
RF	Radio part of the Bluetooth protocol stack
SUT	System Under Test
SW	Software
TC	Test Control layer for the test interface
TCI	Test Control Interface
UART	Universal Asynchronous receiver Transmitter
USB	Universal Serial Bus
UT	Upper Tester

Appendix I

REVISION HISTORY

Bluetooth.

Revision History

Part A / Radio Specification

Rev	Date	Comments
0.8	Jan 21st 1999	• System ambient temperature range • Power control step size • Transmit Spectrum mask • Frequency drift in a packet • New paragraph "Receiver susceptibility to frequency drift" • Adjacent interference levels • Measurement frequency for intermodulation test • Maximum useable level defined • New paragraph "Reference Interference-Signal Definition"
0.9	April 30th 1999	• Eye-diagram added • Out-of band blocking included • RSSI included
1.0 draft	July 5th 1999	• Appendix A and B added, (extreme conditions definition and test conditions) • Tolerances of the Eye-diagram added
1.0B	Dec 1st 1999	• Revised from a linguistic point of view. • Errata items previously published on the web has been included. These corrections and clarifications are marked with correction bars.

Part B / Baseband Specification

Rev	Date	Comments
0.7	Oct 19th 1998	• Minor changes in chapter 1-9. • "Link Monitoring" chapter removed (chapter 12 in v0.6). • Most parts of chapter 10 - 14 re-written.
0.8	Jan 21st 1999	• Some editorial changes in chapter 1-13. • Chapter 14 revised and partly re-written.
1.0 draft	July 5th 1999	• Please see revision bars in document.
1.0B	Dec 1st 1999	• Revised from a linguistic point of view. • Errata items previously published on the web has been included. These corrections and clarifications are marked with correction bars.

Part C / Link Manager Protocol

Rev	Date	Comments
0.7	Oct 19th 1998	• EEE address ⇒ BD_ADDR (or BD address) • M_ADDR ⇒ AM_ADDR • response nbr ⇒ PM_ADDR • Added section 3.1.4 • Added section 3.4 "Change the current link key" • Changed section 3.5 "Encryption". Added negotiation for encryption mode and encryption key size. • 3.14 "Sniff mode". Removed author's note in italic. It is now decided how to determine the first sniff slot. • 3.15 "Park mode". This section is totally changed. • 3.18 "Quality of Service". Added a parameter N_{BC} in one of the LMP messages. • 3.19 "SCO links". Removed author's note in italic. It is now decided how to determine the first SCO slot. • Added section 4 "Connection establishment" • 5.1 "Description of parameters" The length, type and unit of many parameters changed. Especially, all time-parameters are now measured in slots. • Added section 6 "Test modes"
0.8	Jan 21st 1999	• general: Changed "PDU nbr" to OpCode throughout the document. Two errors found in 0.80 were corrected. Table5.1: It says that the length of LMP_SCO_link_req is 7. This should be 8. Table 5.2: It says that the length of hold_time is 1. This should be 2. • 2. Added transactionID in bit0 of the byte in the payload where we have the OpCode. • 3.1 Removed LMP_accepted/not_accepted after receiving LMP_sres (3.1). • 3.1.2 Removed the last sentence "It the claimant is Ö initiate pairing, see 3.2.1" • 3.2 Major change since the initiator of the pairing procedure does not have to be the master. • 3.2.3 Minor clarification. • 3.3 Clearified that if the unit key is changed, the units must go through the initialisation procedure in order to change the link key. • 3.5.3 Modified the calculation of Kc according to changes in chapter 14 of the Baseband Specification. • 3.10 Switch can now be done anytime during the connection and it can be initiated by both master and slave. • 3.15 Clearified that the broadcast scan window is only valid for the current beacon. • 3.19.5 Added description that the PDU includes a reason parameter with information about why the SCO link is removed. • 3.21 This section was removed. Instead error handling is described in a separate chapter (CH. 7) • 4. Added LMP_host_connection_req and removed the LMP_accepted or LMP_not_accepted that the slave sends after the paging procedure. • 5.1 Added reason "invalid parameters" • 6. Added sequences. The slave can return LMP_not_accepted if not allowed to enter test or if not in test mode. Changed OpCodes for the two test mode PDU

0.9	April 30th 1999	• Name request procedure: Coding of the characters was changed from ASCII to UTF-8. The length of the name parameter was increased from 16 bytes to 248 bytes. • The detach reason and the reason parameter were merged into one parameter (reason). The coding of this parameter was changed and is now the same as in HCI. • Sniff procedure: Sniff interval and sniff offset parameters were changed from one byte to two bytes. Two parameters, sniff offset and sniff timeout, were also added. • The PDU LMP_slot_offset was added. • The PDU LMP_page_mode_req was added. • The PDU LMP_page_scan_mode_req was added. • Caption text was added to all figures, tables and sequences where such text was missing. • A maximum reply time was defined for response messages in LMP procedures. • The pairing procedure was modified to allow a claimant with fixed PIN to request to become verifier. • Changed some parameters in LMP_test_control to match the latest revision of "Bluetooth test modes". • The PDU LMP_supervision_timeout was added. • The parameter user data rate was removed from the PDU LMP_SCO_link request. • The length of the parameters CompId and SubVersNr included in the PDU LMP_version_req/res was increased to two bytes. • Some editorial changes and clarifications were made here and there.
1.0 draft	July 5th 1999	• Added figure 3 explaining sequence diagram conventions (Section 3). • Modified procedure to negotiate encryption key size to match changes in the Baseband specification (Section 3.6.2). • Added further description about the parameters in the version request (Section 3.10). • Clearified name request procedure and changed parameter name from "name" to "name fragment" (Section 3.13). • Added description that test mode can also be ended by sending LMP_test_control (Section 6.1). • Some editorial comments. • Decoupled RSSI and power control in the features parameter. Changed parameter "TBD" to "for future use". • Rewrote Section 3.20.
1.0a	July 26th 1999	• Correction of Table 5.3: Byte 1/Bit 1 “and power control” deleted Byte 2/Bit 2 “power control” added
1.0B	Dec 1st 1999	• Revised from a linguistic point of view. • Errata items previously published on the web has been included. These corrections and clarifications are marked with correction bars.

Part D
Logical Link Control and Adaptation Protocol Specification

Rev	Date	Comments
0.8	Jan 21st 1999	• Changes include the addition of an operational overview section, increasing CID lengths to 16 bits and removing the source CID from normal data flow. Moved all connection, termination, and configuration commands to a separate "CID". Changed the name of the "Connection ID" to the "Channel ID". • Added the state machine • Re-defined the timers to more clearly indicate their responsibilities. New Flags field in Configuration Request packet defined. • Several minor editorial corrections.
0.9	April 30th 1999	•
1.0 draft	July 5th 1999	• Replaced all references to "termination" with "disconnection". KTX timer removed - link loss results in channel loss. OSI naming terms applied to state machine. State machine table revised to remove superfluous states. Added some message sequence charts for clarification. Service interface defined in more detail and no longer specified as a guideline - this interface still needs work to complete. • Service interface revised corrections to state machine added, and security key management removed from service interface. • Editorial cleanup of various sections with the majority of the edits being in the state machine and service interface. • Added L2CA_Response service primitives. Clean up of state machined, closed outstanding issues dealing with MTU, • Implements major editorial changes to the configuration process (section 6.3), replacing SCID and DCID with RCID and LCID, removal of flush signalling PDU, and editorial comments from reviewers.
1.0B	Dec 1st 1999	• Revised from a linguistic point of view. • Errata items previously published on the web has been included. These corrections and clarifications are marked with correction bars.

Bluetooth.

Part E / Service Discovery Protocol (SDP)

Rev	Date	Comments
1.0 draft	July 5th 1999	• Removed the Icon10 attribute and the Icon data element type. Re-assigned the IconURL attribute ID to be 0x000C. Updated Example 3. Removed "notes to reviewers". Removed the list of TBD items, since it is empty. • A few corrections to example 3. Modified the BluetoothProfileList attribute to become the BluetoothProfileDescriptorList attribute, which contains a version number for each profile as well as the profile's UUID • Added a description of the protocol version number included in the BluetoothProfileDescriptorList attribute. No change to the 'phonebook' and 'calendar' data store indicators as the values in example 3 already match those of the synchronization profile. Replaced the browse hierarchy diagram in 2.8.1 because the previous version triggered a bug in MS-Word.
1.0B	Dec 1st 1999	• Revised from a linguistic point of view. • Errata items previously published on the web has been included. These corrections and clarifications are marked with correction bars.

Part F:1 / RFCOMM with TS 0710

Rev	Date	Comments
0.8	Jan 21st 1999	• Table 2.1 corrected. • Revised section 2.3. • Revised chapter 3. • Revised section 5.2, 5.3. • Clarifications added to Figure 5.1. • Changed title and contents of section 5.4 (old section 5.4 not needed any more). • Text removed from section 5.5 and 5.6. • Added text on Service Discovery in section 7.3.
0.9	April 30th 1999	• Lots of editorial changes and clarifications. • Added statement on baud rate settings vs. RFCOMM throughput in chapter 2. • Removed section 7.2 on flow control (information duplicated in TS 07.10). • Added DLC parameter negotiation command support (section 4.3, 5.7) • Added clarifications on session closure handling in section 5.2. • Major update regarding SDP; section 7.2. • Added section 7.3 on lower layer dependencies.
1.0 draft	July 5th 1999	• Editorial changes and clarifications in chapters 5 and 7. • Removed sections implying possibility to have more than one RFCOMM entity in a device.
1.0B	Dec 1st 1999	• Revised from a linguistic point of view. • Errata items previously published on the web has been included. These corrections and clarifications are marked with correction bars.

Part F:2 / IrDA Interoperability

Rev	Date	Comments
0.9	April 30th 1999	• Many linguistical changes made, Bluetooth OBEX related specifications chapter added, Separate Application profile chapters gathered into one chapter, OBEX operation requirements moved into the profile specifications, Refers to COMM-ports removed. • Service Records moved into the profile specifications, and added clarification for the use of connection-oriented OBEX
1.0 draft	July 5th 1999	• Changed reference to new version of IrOBEX specification, corrected wrong TCP port number, reference list updated • Updated Chapter 3.2
1.0B	Dec 1st 1999	• Revised from a linguistic point of view.

Part F:3 / Telephony Control Specification (TCS)

Rev	Date	Comments
1.0 draft	July 5th 1999	• Version for 1.0 Release, only editorial changes since 0.9
1.0B	Dec 1st 1999	• Revised from a linguistic point of view. • Errata items previously published on the web has been included. These corrections and clarifications are marked with correction bars. Figure A and Figure B in Appendix are replaced.

Part F:4
Interoperability Requirements for Bluetooth as a WAP Bearer

Rev	Date	Comments
1.0B	Dec 1st 1999	• Revised from a linguistic point of view.

Bluetooth.

Part H:1
Bluetooth Host Controller Interface Functional Specification

Rev	Date	Comments
0.8	Jan 21st 1999	• Many editorial corrections
0.9	April 30th 1999	•
1.0 draft	July 5th 1999	• Flow control for data changed. • Format of HCI Data Packet header changed. • Command Pending Event replaced by Command Status Event and functionality regarding which event should be returned when a command involving LMP actions can not start to execute due to an error changed. • Some commands and events have been renamed. • Changed parameters, descriptions and functionality for many commands and events. • Many new commands have been added. • HCI_Store_Clock_Offset command has been removed. This information is now provided at connection set up or name request as a parameter together with other new baseband related parameters. • Some new events have been added. • Descriptions of error codes added.
1.0B	Dec 1st 1999	• Revised from a linguistic point of view. • Errata items previously published on the web has been included. These corrections and clarifications are marked with correction bars.

Part H:2 / HCI USB Transport Layer

Rev	Date	Comments
0.8	Jan 21st 1999	• Added info about 64 byte isochronous endpoints. • Added a section detailing one of the mail messages - that discussed how SCO traffic would travel across the interface
0.9	April 30th 1999	• Updated revision # & HCI Header sizes •
1.0 draft	July 5th 1999	• Updated endpoint information, interface information relating to isoch, and Device Firmware Upgrade Requirements • Tidied up table describing interface/endpoint/alternate setting information
1.0B	Dec 1st 1999	• Revised from a linguistic point of view. • Errata items previously published on the web has been included. These corrections and clarifications are marked with correction bars.

Part H:3 / HCI RS232 Transport Layer

Rev	Date	Comments
0.9	April 30th 1999	• Assumption about error free link has been removed. • A simple error recovery, negotiation scheme and a resynchronisation/error indication scheme using RTS/CTS were added as proposed by IBM. • New document outline. • Added synchronisation using delimiters with COBS. • Added a support for CCITT-CRC. • Available Baud Rate Changed. • Only 8 bit data length is valid. • HCI Event packet type is added. • Available error type has been modified. • Editorial changes.
1.0 draft	July 5th 1999	• Assumption about error free link has been removed. • A simple error recovery, negotiation scheme and a resynchronisation/error indication scheme using RTS/CTS were added . • New document outline. • Added synchronisation using delimiters with COBS. • Added a support for CCITT-CRC • Available Baud Rate Changed. • Only 8 bit data length is valid. • HCI Event packet type is added. • Available error type has been modified. • Editorial changes.
1.0B	Dec 1st 1999	• Revised from a linguistic point of view.

Part H:4
HCI UART Transport Layer

Rev	Date	Comments
0.9	April 30th 1999	• was not a part of 0.8 • First revision. Based on HCI RS232 Transport Layer 0.80. • Added Default Settings, HW flow control and Error Recovery. • Improved description of RTS/CTS • Changed HCI packet indicator for HCI event packet
1.0 draft	July 5th 1999	• No changes since 0.9
1.0B	Dec 1st 1999	• Revised from a linguistic point of view.

Part I:1 / Test Mode

Rev	Date	Comments
0.9	April 30th 1999	• Transmitter Test: added pseudorandom bit sequence • Corresponding changes for LMP messages • Editorial Changes: Explanatory paragraph and figure added to the test packet format of the transmitter test • Proposal of a loopback alternative that has less demanding time constraints. • Reduced Hopping Sequence added • Modification/Clearification of TX packet format • Editorial changes • Statement in introduction that test mode may be used also for regulatory approval. • Description of delayed loopback added • proposed timing for reduced hopping scheme from Ericsson • timing for reduced hopping scheme refined • Loopback: transmission of NULL packet on failed HEC is not mandatory. • Added AUX1 packet explicitely to Figure 3 • Included AUX1 to Table 3 • Mentioned delayed loopback in first paragraph of Section 2.2 • Features request command is allowed, while in test mode. • Added codes for LMP packets according to LMP V0.9 review document Changed maximum packet length in Table 3 • FH timing: clearify that clock of tester is used • editorial changes • Clearifications about Whitening for both TX and Loopback mode • Clearification that RX and TX in the control command refer to the DUT. • Clearification in Table 3: For ACL packets a maximum length is given, for HV3 the exact length is given. • Editorial Changes
1.0 draft	July 5th 1999	• Description over use of whitening in transmitter test mode. • Added an exit command in the test mode control. • Editorial changes to whitening in transmitter test.
1.0B	Dec 1st 1999	• Revised from a linguistic point of view. • Errata items previously published on the web has been included. These corrections and clarifications are marked with correction bars.

Part I:2 Bluetooth Compliance Requirements

Rev	Date	Comments
1.0A	July 26th 1999	• The text in Section 6.6.1 "Definition of Bluetooth components" first paragraph has been revised.
1.0B	Dec 1st 1999	• Revised from a linguistic point of view.

Part I:3 Test Control Interface

Rev	Date	Comments
1.0B	Dec 1st 1999	• Revised from a linguistic point of view.

Appendix IV / Sample Data

Rev	Date	Comments
1.0B	Dec 1st 1999	• Revised from a linguistic point of view.

Appendix V / Bluetooth Audio

Rev	Date	Comments
1.0B	Dec 1st 1999	• Revised from a linguistic point of view.

Appendix VI / Baseband Timers

Rev	Date	Comments
1.0B	Dec 1st 1999	• Revised from a linguistic point of view. • Errata items previously published on the web has been included. These corrections and clarifications are marked with correction bars.

Appendix VII /Optional Paging Scheme

Rev	Date	Comments
1.0B	Dec 1st 1999	• Revised from a linguistic point of view.

Appendix VIII / Bluetooth Assigned Numbers

Rev	Date	Comments
1.0 draft	July 5th 1999	• Renamed some of the service class mnemonics. Added a bit to indicate if a node is in Limited Discoverable Mode (useful in a reply to GIAC). • Updated after review. Added the IAC LAP codes. • Minor editoral and clarifications, including section "Universally Unique Identifier (UUID) short forms".
1.0B	Dec 1st 1999	• Revised from a linguistic point of view. • Errata items previously published on the web has been included. These corrections and clarifications are marked with correction bars.

Appendix IX / Message Sequence Charts

Rev	Date	Comments
1.0 draft	July 5th 1999	• Command and Parameter updated; Chapter 7 added; Local- and Remote Hold-Mode removed; Chapter 8 clarified; No HCI-Num-Completed-Packets-Event in ACL- and SCO-Connection creation; LMP-feature-req/-res added in ACL-Connection-Setup; LMP-feature-req/-res removed in SCO-Connection-Setup • MSC "Onetime-Inquiry" and "Periodic-Inquiry" clarified with additional ID-Packet; • MSC "Local Loopback-Mode" exited with HCI_Write_Loopback_Mode instead with HCI_Reset; • MSC "Remote Loopback-Mode" exited with HCI_Write_Loopback_Mode instead with HCI_Reset; • MSC "Switch Role" updated with LMP_clkoffset_req/-res and LMP_slot_offset; • Editorial changes; • MSC "Switch Role": subscenario 2 added; Modified as Toru's proposal; • MSC "ACL-Connection Request": slot-offset and clock-offset exchange subscenario3; • Figure 4: no loopback to Page/Page-Res in case of Role-Switch; • LMP_features_req/-res: added in MSC "Pairing" and MSC "Authentication", removed in MSC "Encryption and Setup Complete"; • MSC "Local Loopback Mode": two additional SCO-Connections added;
1.0B	Dec 1st 1999	• Revised from a linguistic point of view. • Errata items previously published on the web has been included. These corrections and clarifications are marked with correction bars. The following figures are modified: 3.2, 3.3, 4.3 and 4.9

Appendix II

CONTRIBUTORS

Contributors

Part A / Radio Specification

Sven Mattisson	Ericsson Mobile Communications AB
Lars Nord (owner)	Ericsson Mobile Communications AB
Anders Svensson	Ericsson Mobile Communications AB
Paul Burgess	Nokia Mobile Phones
Olaf Joeressen	Nokia Mobile Phones
Thomas Muller	Nokia Mobile Phones
Troy Beukema	IBM Corporation
Brian Gaucher	IBM Corporation
Allen Huotari	Toshiba Corporation

Part B / Baseband Specification

Ayse Findikli	Ericsson Mobile Communications AB
Jaap Haartsen (owner)	Ericsson Mobile Communications AB
Joakim Persson	Ericsson Mobile Communications AB
Chatschik Bisdikian	IBM Corp.
Kris Fleming	Intel Corp.
Olaf Joeressen	Nokia Mobile Phones
Thomas Muller	Nokia Mobile Phones

Part C / Link Manager Protocol

Johannes Elg	Ericsson Mobile Communications AB
Jaap Haartsen	Ericsson Mobile Communications AB
Tobias Melin (owner)	Ericsson Mobile Communications AB
Chatschik Bisdikian	IBM Corporation
Kris Fleming	Intel Corporation
Thomas Busse	Nokia Mobile Phones
Olaf Joeressen	Nokia Mobile Phones
Thomas Müller	Nokia Mobile Phones
Dong Nguyen	Nokia Mobile Phones

Part D
Logical Link Control and Adaptation Protocol Specification

Burgess, Jon	3COM
Moran, Paul	3COM
Elg, Johannes	Ericsson Mobile Communications AB
Haartsen, Jaap	Ericsson Mobile Communications AB
Nilsson, Ingemar	Ericsson Mobile Communications AB
Runesson, Stefan	Ericsson Mobile Communications AB
Slot, Gerrit	Ericsson Mobile Communications AB
Sörensen, Johan	Ericsson Mobile Communications AB
Svennarp, Goran	Ericsson Mobile Communications AB
Aihara, Tohru	IBM Corporation
Bisdikian, Chatschik	IBM Corporation
Fleming, Kris	Intel Corporation
Gadamsetty, Uma	Intel Corporation
Hunter, Robert	Intel Corporation
Inouye, Jon (owner)	Intel Corporation
Lo, Steve C.	Intel Corporation
Zhu, Chunrong	Intel Corporation
Busse, Thomas	Nokia Mobile Phones
Makinen, Rauno	Nokia Mobile Phones
Müller, Thomas	Nokia Mobile Phones
Nykänen, Petri	Nokia Mobile Phones
Ollikainen, Peter	Nokia Mobile Phones
Kinoshita, Katsuhiro	Toshiba Corporation

Part E
Service Discovery Protocol (SDP)

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

Part F:1
RFCOMM with TS 07.10

Ingemar Nilsson	Ericsson Mobile Communications AB
Patrik Olsson	Ericsson Mobile Communications AB
Gerrit Slot	Ericsson Mobile Communications AB
Johan Sörensen (owner)	Ericsson Mobile Communications AB
Srikanth Kambhatla	Intel Corporation
Michael Camp	Nokia Mobile Phones
Riku Mettälä	Nokia Mobile Phones

Part F:2
IrDA Interoperability

David Kammer	3Com
Christian Andersson	Ericsson Mobile Communications AB
Johannes Elg	Ericsson Mobile Communications AB
Patrik Olsson	Ericsson Mobile Communications AB
Johan Sörensen	Ericsson Mobile Communications AB
Dave Suvak	Extended Systems
Chatschik Bisdikian	IBM
Brent Miller	IBM
Apratim Purakayastha	IBM
Aron Walker	IBM
Jon Inouye	Intel Corporation
Michael Camp	Nokia Mobile Phones
Riku Mettälä (owner)	Nokia Mobile Phones
Peter Ollikainen	Nokia Mobile Phones
James Scales	Nokia Mobile Phones
Steve Rybicki	Puma Technology
John Stossel	Puma Technology

Part F:3
Telephony Control Protocol Specification

Richard Shaw	3COM
Ken Morley	3COM
Olof Dellien	Ericsson Mobile Communications AB
Gert-jan van Lieshout	Ericsson Mobile Communications AB
Erik Slotboom (owner)	Ericsson Mobile Communications AB
Shridar Rajagopal	Intel Corporation
Ramu Ramakeshavan	Intel Corporation
Brian Redding	Motorola
Thomas Müller	Nokia Mobile Phones
Christian Zechlin	Nokia Mobile Phones
Jun'ichi Yoshizawa	Toshiba Corporation

Bluetooth.

Part F:4
Interoperability Requirements for Bluetooth as a WAP Bearer

Johannes Elg	Ericsson Mobile Communications AB
Robert Hed	Ericsson Mobile Communications AB
Jon Inouye	Intel Corporation
Michael T. Camp (owner)	Nokia Mobile Phones
Riku Mettala	Nokia Mobile Phones
Kevin Wagner	Nokia Mobile Phones

Part H:1
Bluetooth Host Controller Interface Functional Specification

Todor Cooklev	3Com Corporation
Johannes Elg	Ericsson Mobile Communications AB
Christian Johansson	Ericsson Mobile Communications AB
Patrik Lundin	Ericsson Mobile Communications AB
Tobias Melin	Ericsson Mobile Communications AB
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 (owner)	Intel Corporation
Robert Hunter	Intel Corporation
Jon Inouye	Intel Corporation
Srikanth Kambhatla	Intel Corporation
Steve Lo	Intel Corporation
Vijay Suthar	Intel Corporation
Greg Muchnik	Motorola
Thomas Busse	Nokia Mobile Phones
Thomas Muller	Nokia Mobile Phones
Dong Nguyen	Nokia Mobile Phones
Christian Zechlin	Nokia Mobile Phones
Masahiro Tada	Toshiba Corporation

Part H:2 / HCI USB Transport Layer

Patrik Lundin	Ericsson Mobile Communications AB
Nathan Lee	IBM Corporation
Les Cline	Intel Corporation
Brad Hosler	Intel Corporation
John Howard	Intel Corporation
Robert Hunter (owner)	Intel Corporation
Srikanth Kambhatla	Intel Corporation
Kosta Koeman	Intel Corporation
John McGrath	Intel Corporation
Uwe Gondrum	Nokia Mobile Phone

Part H:3 / HCI RS232 Transport Layer

Johannes Elg	Ericsson Mobile Communications AB
Sven Jerlhagen	Ericsson Mobile Communications AB
Patrik Lundin	Ericsson Mobile Communications AB
Chatschik Bisdikian	IBM Corporation
Edgar Kerstan	IBM Corporation
Nathan Lee (owner)	IBM Corporation
Robert Hunter	Intel Corporation
Patrick Kane	Motorola
Uwe Gondrum	Nokia Mobile Phone
Masahiro Tada	Toshiba Corporation

Part H:4 / HCI UART Transport Layer

Johannes Elg	Ericsson Mobile Communications AB
Sven Jerlhagen	Ericsson Mobile Communications AB
Patrik Lundin (owner)	Ericsson Mobile Communications AB
Lars Novak	Ericsson Mobile Communications AB
Edgar Kerstan	IBM Corporation
Nathan Lee	IBM Corporation
Robert Hunter	Intel Corporation
Patrick Kane	Motorola
Uwe Gondrum	Nokia Mobile Phone
Masahiro Tada	Toshiba Corporation

Part I:1 / Bluetooth Test Mode

Ayse Findikli	Ericsson Mobile Communications AB
Mårten Mattsson	Ericsson Mobile Communications AB
Tobias Melin	Ericsson Mobile Communications AB
Lars Nord	Ericsson Mobile Communications AB
Fredrik Töörn	Ericsson Mobile Communications AB
Jeffrey Schiffer	Intel Corporation
Daniel Bencak	Nokia Mobile Phones
Arno Kefenbaum	Nokia Mobile Phones
Thomas Müller (owner)	Nokia Mobile Phones
Roland Schmale	Nokia Mobile Phones

Part I:2 / Bluetooth Compliance Requirements

Lawrence Jones	ComBit,Inc.
Magnus Hansson (owner)	Ericsson Mobile Communications AB
Göran Svennarp	Ericsson Mobile Communications AB
Gary Robinson	IBM Corporation
John Webb	Intel Corporation
Petri Morko	Nokia Mobile Phones
Warren Allen	Toshiba Corporation

Part I:3 / Bluetooth Test Control Interface

Mårten Mattsson (owner)	Ericsson Mobile Communications AB
Dan Sönnerstam	Ericsson Mobile Communications AB
Thomas Müller	Nokia Mobile Phones

Appendix IV / Encryption Sample Data

Joakim Persson (owner)	Ericsson Mobile Communications AB
Thomas Müller	Nokia Mobile Phones
Thomas Sander	Nokia Mobile Phones

Appendix V / Bluetooth Audio

Joakim Persson (owner)	Ericsson Mobile Communication AB
Fisseha Mekuria	Ericsson Mobile Communication AB
Magnus Hansson	Ericsson Mobile Communication AB
Mats Omrin	Ericsson Mobile Communication AB

Appendix VI / Baseband Timers

Ayse Findikli	Ericsson Mobile Communications AB
Jaap Haartsen (owner)	Ericsson Mobile Communications AB
Joakim Persson	Ericsson Mobile Communications AB

Appendix VII / Optional Paging Scheme

Ayse Findikli	Ericsson Mobile Communications AB
Jaap Haartsen (owner)	Ericsson Mobile Communications AB
Joakim Persson	Ericsson Mobile Communications AB
Olaf Joeressen	Nokia Mobile Phones
Thomas Muller	Nokia Mobile Phones
Markus Schetelig	Nokia Mobile Phones

Appendix VIII / Bluetooth Assigned Numbers

Johannes Elg (owner)	Ericsson Mobile Communications AB
Numerous SIG members contributed directly or indirectly to this revision	

Appendix IX / Message Sequence Charts between

Todor Cooklev	3Com Corporation
Christian Johansson	Ericsson Mobile Communications AB
Tobias Melin	Ericsson Mobile Communications AB
Toru Aihara	IBM Corporation
Chatschik Bisdikian	IBM Corporation
Nathan Lee	IBM Corporation
Kris Fleming	Intel Cop.
Greg Muchnik	Motorola
Thomas Busse	Nokia Mobile Phones
Dong Nguyen (owner)	Nokia Mobile Phones

The Bluetooth Specification was compiled and edited by Dan Sonnerstam, Pyramid Communication AB

Appendix III

ACRONYMS AND ABBREVIATIONS

List of Acronyms and Abbreviations

Acronym or abbreviation	Writing out in full	Which means
ACK	Acknowledge	
ACL link	Asynchronous Connection-Less link	Provides a packet-switched connection.(Master to any slave)
ACO	Authenticated Ciphering Offset	
AM_ADDR	Active Member Address	
AR_ADDR	Access Request Address	
ARQ	Automatic Repeat reQuest	
B		
BB	BaseBand	
BCH	Bose, Chaudhuri & Hocquenghem	Type of code The persons who discovered these codes in 1959 (H) and 1960 (B&C)
BD_ADDR	Bluetooth Device Address	
BER	Bit Error Rate	
BT	Bandwidth Time	
BT	Bluetooth	
C		
CAC	Channel Access Code	
CC	Call Control	
CL	Connectionless	
CODEC	COder DECoder	
COF	Ciphering Offset	
CRC	Cyclic Redundancy Check	
CVSD	Continuous Variable Slope Delta Modulation	
D		
DAC	Device Access Code	
DCE	Data Communication Equipment	

Acronym or abbreviation	Writing out in full	Which means
DCE	Data Circuit-Terminating Equipment	In serial communications, DCE refers to a device between the communication endpoints whose sole task is to facilitate the communications process; typically a modem
DCI	Default Check Initialization	
DH	Data-High Rate	Data packet type for high rate data
DIAC	Dedicated Inquiry Access Code	
DM	Data - Medium Rate	Data packet type for medium rate data
DTE	Data Terminal Equipment	In serial communications, DTE refers to a device at the endpoint of the communications path; typically a computer or terminal.
DTMF	Dual Tone Multiple Frequency	
DUT	Device Under Test	
DV	Data Voice	Data packet type for data and voice
E		
ETSI	European Telecommunications Standards Institute	
F		
FCC	Federal Communications Commission	
FEC	Forward Error Correction code	
FH	Frequency Hopping	
FHS	Frequency Hop Synchronization	
FIFO	First In First Out	
FSK	Frequency Shift Keying	type of modulation
FW	Firmware	
G		
GEOP	Generic Object Exchange Profile	
GFSK	Gaussian Frequency Shift Keying	
GIAC	General Inquiry Access Code	
GM	Group Management	

Acronym or abbreviation	Writing out in full	Which means
H		
HA	Host Application	SW using Bluetooth
HCI	Host Controller Interface	
HEC	Header-Error-Check	
HID	Human Interface Device	
HV	High quality Voice	e.g. HV1 packet
HW	Hardware	
I		
IAC	Inquiry Access Code	
IEEE	Institute of Electronic and Electrical Engineering	
IETF	Internet Engineering Task Force	
IP	Internet Protocol	
IrDA	Infra-red Data Association	
IrMC	Ir Mobile Communications	
ISDN	Integrated Services Digital Net-works	
ISM	Industrial, Scientific, Medical	
IUT	Implementation Under Tes	
L		
L_CH	Logical Channel	
L2CA	Logical Link Control and Adaption	Logical Link Control And Manage-ment part of the Bluetooth protocol stack
L2CAP	Logical Link Control and Adaption Protocol	
LAP	Lower Address Part	
LC	Link Controller	Link Controller (or baseband) part of the Bluetooth protocol stack Low level Baseband protocol han-dler
LCP	Link Control Protocol	
LCSS	Link Controller Service Signalling	
LFSR	Linear Feedback Shift Register	
LM	Link Manager	

Acronym or abbreviation	Writing out in full	Which means
LMP	Link Manager Protocol	For LM peer to peer communication
LSB	Least Significant Bit	
M		
M	Master or Mandatory	
M_ADDR	Medium Access Control Address	
MAC	Medium Access Control	
MAPI	Messaging Application Procedure Interface	
MMI	Man Machine Interface	
MS	Mobile Station	
MS	Multiplexing sublayer	
MSB	Most Significant Bit	
MSC	Message Sequence Chart	
MTU	Maximum Transmission Unit	
MUX	Multiplexing Sublayer	a sublayer of the L2CAP layer
N		
NAK	Negative Acknowledge	
NAP	Non-significant Address Part	
O		
O	Optional	
OBEX	OBject EXchange protocol	
OCF	Opcode Command Field	
P		
PCM	Pulse Coded Modulation	
PCMCIA	Personal Computer Memory Card International Association	
PDU	Protocol Data Unit	a message
PIN	Personal Identification Number	
PM_ADDR	Parked Member Address	
PN	Pseudo-random Noise	
PnP	Plug and Play	
POTS	Plain Old Telephone system	

Acronym or abbreviation	Writing out in full	Which means
PPM	Part Per Million	
PPP	Point-to-Point Protocol	
PRBS	Pseudo Random Bit Sequence	
PRNG	Pseudo Random Noise Generation	
PSTN	Public Switched Telephone Network	
Q		
QoS	Quality of Service	
R		
RAND	Random number	
RF	Radio Frequency	
RFC	Request For Comments	
RFCOMM		Serial cable emulation protocol based on ETSI TS 07.10
RSSI	Received Signal Strength Indication	
RX	Receiver	
S		
S	Slave	
SAP	Service Access Points	
SAR	Segmentation and Reassembly	
SCO link	Synchronous Connection-Oriented link	Supports time-bounded information like voice. (Master to single slave)
SD	Service Discovery	
SDDB	Service Discovery Database	
SDP	Service Discovery Protocol	
SEQN	Sequential Numbering scheme	
SRES	Signed Response	
SS	Supplementary Services	
SSI	Signal Strength Indication	
SUT	System Under Test	
SW	Software	

Acronym or abbreviation	Writing out in full	Which means
T		
TAE	Terminal Adapter Equipment	
TBD	To Be Defined	
TC	Test Control	Test Control layer for the test interface
TCI	Test Control Interface	
TCP/IP	Transport Control Protocol/Internet Protocol	
TCS	Telephony Control protocol Specification	
TDD	Time-Division Duplex	
TTP	Tiny Transport Protocol between OBEX and UDP [TBD]	
TX	Transmit	
U		
UA	User Asynchronous	Asynchronous user data
UAP	Upper Address Part	
UART	Universial Asynchronous receiver Transmitte	
UC	User Control	
UDP/IP	User Datagram Protocol/Internet Protocol	
UI	User Isochronous	Isochronous user data
UIAC	Unlimited Inquiry Access Code	
US	User Synchronous	Synchronous user data
USB	Universal Serial Bus	
UT	Upper Tester	
W		
WAP	Wireless Application Prtocol	
WUG	Wireless User Group	

Definitions

Baseband. The Bluetooth baseband specifies the medium access and physical layers procedures to support the exchange of real-time voice and data information streams and ad-hoc networking between Bluetooth units.

Coverage area . The area where two Bluetooth units can exchange messages with acceptable quality and performance.

Host Terminal interface. Host terminal interface is the Interface between Bluetooth Host and Bluetooth Unit.

Inquiry. A Bluetooth unit transmits inquiry messages in order to discover the other Bluetooth units that are active within the coverage area. The Bluetooth units that capture inquiry messages may send a response to the inquiring Bluetooth unit. The response contains information about the Bluetooth unit itself and its Bluetooth Host.

Isochronous user channel . Channel used for time bounded information like i.e. compressed audio (ACL link).

Logical Channel. The different types of channels on a Physical Link.

Bluetooth Host. Bluetooth Host is a computing device, peripheral, cellular telephone, access point to PSTN network, etc. A Bluetooth Host attached to a Bluetooth unit may communicate with other Bluetooth Hosts attached to their Bluetooth units as well. The communication channel through the Bluetooth units provides almost wire-like transparency.

Bluetooth Unit. Bluetooth Unit is a voice/data circuit equipment for a short-range wireless communication link. It allows voice and data communications between Bluetooth Hosts.

Bluetooth. Bluetooth is a wireless communication link, operating in the unlicensed ISM band at 2,4 GHz using a frequency hopping tranceiver. It allows real-time voice and data communications between Bluetooth Hosts. The link protocol is based on time slots.

Packet. Format of aggregated bits that can be transmitted in 1, 3, or 5 time slots.

Paging. An Bluetooth unit transmits paging messages in order to set up a communication link to another Bluetooth unit who is active within the coverage area.

Physical Channel. Synchronized RF hopping sequence in a piconet

Physical Link. Connection between devices.

Piconet. In the Bluetooth system, the channel is shared among several Bluetooth units. The units sharing a common channel constitute a piconet.

RFCOMM Client. An RFCOMM client is an application that requests a connection to another application (RFCOMM server).

RFCOMM initiator. The device initiating the RFCOMM session, i.e.setting up RFCOMM channel on L2CAP and starting RFCOMM multiplexing with the SABM command on DLCI 0 (zero).

RFCOMM Server. An RFCOMM server is an application that awaits a connection from an RFCOMM client on another device. What happens after such a connection is established is out of scope of this definition.

RFCOMM Server Channel. This is a subfield of the TS 07.10 DLCI number. This abstraction is used to allow both server and client applications to reside on both sides of an RFCOMM session.

Service Discovery. The ability to discover the capability of connecting devices or hosts

Scatternet. Two or more piconets co-located in the same area (with or witout inter-piconet communication).

Time Slot. The Physical Channel is divided into 625 μs long time slots.

Appendix IV

SAMPLE DATA

This appendix contains sample data for various parts of the Bluetooth baseband specification. All sample data are provided for reference purpose only; they are intended as a complement to the definitions provided elsewhere in the specification. They can be used to check the behavior of an implementation and avoid misunderstandings. Fulfilling these sample data is a necessary but not sufficient condition for an implementation to be fully Bluetooth compliant.

Bluetooth.

CONTENTS

1 ENCRYPTION SAMPLE DATA

This part consist of four sets of sample data for the encryption process.

With respect to the functional description of the encryption engine in the Bluetooth baseband specification, the contents of registers and resulting concurrent values are listed as well. This by no means excludes different implementations (as far as they produce the same encryption stream) but is intended to describe the functional behavior.

In case of misunderstandings or inconsistencies, these sample data form the normative reference.

1.1 GENERATING KC' FROM KC,

where Kc'(x) = g2(x)(Kc(x) mod g1(x)).

```
Note: All polynomials are in hexadecimal notation.
      'L' is the effective key length in bytes.
      The notation 'p: [m]' implies that deg(p(x)) = m.

                                MSB                                   LSB
L = 1
g1:                 [8]        00000000 00000000 00000000 0000011d
g2:                 [119]      00e275a0 abd218d4 cf928b9b bf6cb08f
Kc:                            a2b230a4 93f281bb 61a85b82 a9d4a30e
Kc mod g1:          [7]        00000000 00000000 00000000 0000009f
g2(Kc mod g1):      [126]      7aa16f39 59836ba3 22049a7b 87f1d8a5
-----------------------------------------------------------------
L = 2
g1:                 [16]       00000000 00000000 00000000 0001003f
g2:                 [112]      0001e3f6 3d7659b3 7f18c258 cff6efef
Kc:                            64e7df78 bb7ccaa4 61433123 5b3222ad
Kc mod g1:          [12]       00000000 00000000 00000000 00001ff0
g2(Kc mod g1):      [124]      142057bb 0bceac4c 58bd142e 1e710a50
-----------------------------------------------------------------
L = 3
g1:                 [24]       00000000 00000000 00000000 010000db
g2:                 [104]      000001be f66c6c3a b1030a5a 1919808b
Kc:                            575e5156 ba685dc6 112124ac edb2c179
Kc mod g1:          [23]       00000000 00000000 00000000 008ddbc8
g2(Kc mod g1):      [127]      d56d0adb 8216cb39 7fe3c591 1ff95618
-----------------------------------------------------------------
L = 4
g1:                 [32]       00000000 00000000 00000001 000000af
g2:                 [96]       00000001 6ab89969 de17467f d3736ad9
Kc:                            8917b4fc 403b6db2 1596b86d 1cb8adab
Kc mod g1:          [31]       00000000 00000000 00000000 aa1e78aa
g2(Kc mod g1):      [127]      91910128 b0e2f5ed a132a03e af3d8cda
-----------------------------------------------------------------
```

```
L = 5
g1:                [40]      00000000 00000000 00000100 00000039
g2:                [88]      00000000 01630632 91da50ec 55715247
Kc:                          785c915b dd25b9c6 0102ab00 b6cd2a68
Kc mod g1:         [38]      00000000 00000000 0000007f 13d44436
g2(Kc mod g1):     [126]     6fb5651c cb80c8d7 ea1ee56d f1ec5d02
=========================================================
L = 6
g1:                [48]      00000000 00000000 00010000 00000291
g2:                [77]      00000000 00002c93 52aa6cc0 54468311
Kc:                          5e77d19f 55ccd7d5 798f9a32 3b83e5d8
Kc mod g1:         [47]      00000000 00000000 000082eb 4af213ed
g2(Kc mod g1):     [124]     16096bcb afcf8def 1d226a1b 4d3f9a3d
=========================================================
```

```
L = 7
g1:                [56]    00000000 00000000 01000000 00000095
g2:                [71]    00000000 000000b3 f7fffce2 79f3a073
Kc:                        05454e03 8ddcfbe3 ed024b2d 92b7f54c
Kc mod g1:         [55]    00000000 00000000 0095b8a4 8eb816da
g2(Kc mod g1):     [126]   50f9c0d4 e3178da9 4a09fe0d 34f67b0e
------------------------------------------------------------
L = 8
g1:                [64]    00000000 00000001 00000000 0000001b
g2:                [63]    00000000 00000000 a1ab815b c7ec8025
Kc:                        7ce149fc f4b38ad7 2a5d8a41 eb15ba31
Kc mod g1:         [63]    00000000 00000000 8660806c 1865deec
g2(Kc mod g1):     [126]   532c36d4 5d0954e0 922989b6 826f78dc
------------------------------------------------------------
L = 9
g1:                [72]    00000000 00000100 00000000 00000609
g2:                [49]    00000000 00000000 0002c980 11d8b04d
Kc:                        5eeff7ca 84fc2782 9c051726 3df6f36e
Kc mod g1:         [71]    00000000 00000083 58ccb7d0 b95d3c71
g2(Kc mod g1):     [120]   016313f6 0d3771cf 7f8e4bb9 4aa6827d
------------------------------------------------------------
L = 10
g1:                [80]    00000000 00010000 00000000 00000215
g2:                [42]    00000000 00000000 0000058e 24f9a4bb
Kc:                        7b13846e 88beb4de 34e7160a fd44dc65
Kc mod g1:         [79]    00000000 0000b4de 34171767 f36981c3
g2(Kc mod g1):     [121]   023bc1ec 34a0029e f798dcfb 618ba58d
------------------------------------------------------------
L = 11
g1:                [88]    00000000 01000000 00000000 0000013b
g2:                [35]    00000000 00000000 0000000c a76024d7
Kc:                        bda6de6c 6e7d757e 8dfe2d49 9a181193
Kc mod g1:         [86]    00000000 007d757e 8dfe88aa 2fcee371
g2(Kc mod g1):     [121]   022e08a9 3aa51d8d 2f93fa78 85cc1f87
------------------------------------------------------------
L = 12
g1:                [96]    00000001 00000000 00000000 000000dd
g2:                [28]    00000000 00000000 00000000 1c9c26b9
Kc:                        e6483b1c 2cdb1040 9a658f97 c4efd90d
Kc mod g1:         [93]    00000000 2cdb1040 9a658fd7 5b562e41
g2(Kc mod g1):     [121]   030d752b 216fe29b b880275c d7e6f6f9
------------------------------------------------------------
L = 13
g1:                [104]   00000100 00000000 00000000 0000049d
g2:                [21]    00000000 00000000 00000000 0026d9e3
Kc:                        d79d281d a2266847 6b223c46 dc0ab9ee
Kc mod g1:         [100]   0000001d a2266847 6b223c45 e1fc5fa6
g2(Kc mod g1):     [121]   03f11138 9cebf919 00b93808 4ac158aa
------------------------------------------------------------
```

```
L = 14
g1:                  [112]        00010000 00000000 00000000 0000014f
g2:                  [14]         00000000 00000000 00000000 00004377
Kc:                               cad9a65b 9fca1c1d a2320fcf 7c4ae48e
Kc mod g1:           [111]        0000a65b 9fca1c1d a2320fcf 7cb6a909
g2(Kc mod g1):       [125]        284840fd f1305f3c 529f5703 76adf7cf
-----------------------------------------------------------
L = 15
g1:                  [120]        01000000 00000000 00000000 000000e7
g2:                  [7]          00000000 00000000 00000000 00000089
Kc:                               21f0cc31 049b7163 d375e9e1 06029809
Kc mod g1:           [119]        00f0cc31 049b7163 d375e9e1 0602840e
g2(Kc mod g1):       [126]        7f10b53b 6df84b94 f22e566a 3754a37e
-----------------------------------------------------------
L = 16
g1:                  [128]        00000001 00000000 00000000 00000000 00000000
g2:                  [0]          00000000 00000000 00000000 00000001
Kc:                               35ec8fc3 d50ccd32 5f2fd907 bde206de
Kc mod g1:           [125]        35ec8fc3 d50ccd32 5f2fd907 bde206de
g2(Kc mod g1):       [125]        35ec8fc3 d50ccd32 5f2fd907 bde206de
-----------------------------------------------------------
```

1.2 FIRST SET OF SAMPLE DATA

```
Initial values for the key, pan address and clock

K'c1[0]   = 00   K'c1[1]   = 00   K'c1[2]   = 00   K'c1[3]  = 00
K'c1[4]   = 00   K'c1[5]   = 00   K'c1[6]   = 00   K'c1[7]  = 00
K'c1[8]   = 00   K'c1[9]   = 00   K'c1[10]  = 00   K'c1[11] = 00
K'c1[12]  = 00   K'c1[13]  = 00   K'c1[14]  = 00   K'c1[15] = 00

Addr1[0] = 00   Addr1[1] = 00    Addr1[2] = 00
Addr1[3] = 00   Addr1[4] = 00    Addr1[5] = 00

Clk1[0]  = 00   Clk1[1]  = 00    Clk1[2]  = 00    Clk1[3]  = 00


=====================================================================================
        Fill LFSRs with initial data
=====================================================================================

  t  clk#    LFSR1    LFSR2     LFSR3      LFSR4           X1 X2 X3 X4   Z  C[t+1] C[t] C[t-1]

  0    0    0000000* 00000000* 000000000* 0000000000*     0  0  0  0    0  00      00   00
  1    1    0000000* 00000001* 000000000* 0000000001*     0  0  0  0    0  00      00   00
  2    2    0000000* 00000002* 000000000* 0000000003*     0  0  0  0    0  00      00   00
  3    3    0000000* 00000004* 000000000* 0000000007*     0  0  0  0    0  00      00   00
  4    4    0000000* 00000008* 000000000* 000000000E*     0  0  0  0    0  00      00   00
  5    5    0000000* 00000010* 000000000* 000000001C*     0  0  0  0    0  00      00   00
  6    6    0000000* 00000020* 000000000* 0000000038*     0  0  0  0    0  00      00   00
```

7	7	0000000*	00000040*	000000000*	0000000070*	0	0	0	0	0	00	00	00
8	8	0000000*	00000080*	000000000*	00000000E0*	0	0	0	0	0	00	00	00
9	9	0000000*	00000100*	000000000*	00000001C0*	0	0	0	0	0	00	00	00
10	10	0000000*	00000200*	000000000*	0000000380*	0	0	0	0	0	00	00	00
11	11	0000000*	00000400*	000000000*	0000000700*	0	0	0	0	0	00	00	00
12	12	0000000*	00000800*	000000000*	0000000E00*	0	0	0	0	0	00	00	00
13	13	0000000*	00001000*	000000000*	0000001C00*	0	0	0	0	0	00	00	00
14	14	0000000*	00002000*	000000000*	0000003800*	0	0	0	0	0	00	00	00
15	15	0000000*	00004000*	000000000*	0000007000*	0	0	0	0	0	00	00	00
16	16	0000000*	00008000*	000000000*	000000E000*	0	0	0	0	0	00	00	00
17	17	0000000*	00010000*	000000000*	000001C000*	0	0	0	0	0	00	00	00
18	18	0000000*	00020000*	000000000*	0000038000*	0	0	0	0	0	00	00	00
19	19	0000000*	00040000*	000000000*	0000070000*	0	0	0	0	0	00	00	00
20	20	0000000*	00080000*	000000000*	00000E0000*	0	0	0	0	0	00	00	00
21	21	0000000*	00100000*	000000000*	00001C0000*	0	0	0	0	0	00	00	00
22	22	0000000*	00200000*	000000000*	0000380000*	0	0	0	0	0	00	00	00
23	23	0000000*	00400000*	000000000*	0000700000*	0	0	0	0	0	00	00	00
24	24	0000000*	00800000*	000000000*	0000E00000*	0	1	0	0	1	00	00	00
25	25	0000000*	01000000*	000000000*	0001C00000*	0	0	0	0	0	00	00	00
26	26	0000000	02000000*	000000000*	0003800000*	0	0	0	0	0	00	00	00
27	27	0000000	04000000*	000000000*	0007000000*	0	0	0	0	0	00	00	00
28	28	0000000	08000000*	000000000*	000E000000*	0	0	0	0	0	00	00	00
29	29	0000000	10000000*	000000000*	001C000000*	0	0	0	0	0	00	00	00
30	30	0000000	20000000*	000000000*	0038000000*	0	0	0	0	0	00	00	00
31	31	0000000	40000000*	000000000*	0070000000*	0	0	0	0	0	00	00	00
32	32	0000000	00000001	000000000*	00E0000000*	0	0	0	1	1	00	00	00
33	33	0000000	00000002	000000000*	01C0000000*	0	0	0	1	1	00	00	00
34	34	0000000	00000004	000000000	0380000000*	0	0	0	1	1	00	00	00
35	35	0000000	00000008	000000000	0700000000*	0	0	0	0	0	00	00	00
36	36	0000000	00000010	000000000	0E00000000*	0	0	0	0	0	00	00	00
37	37	0000000	00000020	000000000	1C00000000*	0	0	0	0	0	00	00	00
38	38	0000000	00000040	000000000	3800000000*	0	0	0	0	0	00	00	00
39	39	0000000	00000080	000000000	7000000000*	0	0	0	0	0	00	00	00

==

Start clocking Summation Combiner

==

40	1	0000000	00000100	000000000	6000000001	0	0	0	0	0	00	00	00
41	2	0000000	00000200	000000000	4000000003	0	0	0	0	0	00	00	00
42	3	0000000	00000400	000000000	0000000007	0	0	0	0	0	00	00	00
43	4	0000000	00000800	000000000	000000000E	0	0	0	0	0	00	00	00
44	5	0000000	00001001	000000000	000000001D	0	0	0	0	0	00	00	00
45	6	0000000	00002002	000000000	000000003B	0	0	0	0	0	00	00	00
46	7	0000000	00004004	000000000	0000000077	0	0	0	0	0	00	00	00
47	8	0000000	00008008	000000000	00000000EE	0	0	0	0	0	00	00	00
48	9	0000000	00010011	000000000	00000001DD	0	0	0	0	0	00	00	00
49	10	0000000	00020022	000000000	00000003BB	0	0	0	0	0	00	00	00
50	11	0000000	00040044	000000000	0000000777	0	0	0	0	0	00	00	00
51	12	0000000	00080088	000000000	0000000EEE	0	0	0	0	0	00	00	00
52	13	0000000	00100110	000000000	0000001DDD	0	0	0	0	0	00	00	00
53	14	0000000	00200220	000000000	0000003BBB	0	0	0	0	0	00	00	00
54	15	0000000	00400440	000000000	0000007777	0	0	0	0	0	00	00	00
55	16	0000000	00800880	000000000	000000EEEE	0	1	0	0	1	00	00	00
56	17	0000000	01001100	000000000	000001DDDD	0	0	0	0	0	00	00	00
57	18	0000000	02002200	000000000	000003BBBB	0	0	0	0	0	00	00	00
58	19	0000000	04004400	000000000	0000077777	0	0	0	0	0	00	00	00
59	20	0000000	08008800	000000000	00000EEEEE	0	0	0	0	0	00	00	00
60	21	0000000	10011000	000000000	00001DDDDD	0	0	0	0	0	00	00	00

61	22	0000000	20022000	000000000	00003BBBBB	0	0	0	0	0	00	00	00
62	23	0000000	40044000	000000000	0000777777	0	0	0	0	0	00	00	00
63	24	0000000	00088001	000000000	0000EEEEEE	0	0	0	0	0	00	00	00
64	25	0000000	00110003	000000000	0001DDDDDD	0	0	0	0	0	00	00	00
65	26	0000000	00220006	000000000	0003BBBBBB	0	0	0	0	0	00	00	00
66	27	0000000	0044000C	000000000	0007777777	0	0	0	0	0	00	00	00
67	28	0000000	00880018	000000000	000EEEEEEE	0	1	0	0	1	00	00	00
68	29	0000000	01100031	000000000	001DDDDDDC	0	0	0	0	0	00	00	00
69	30	0000000	02200062	000000000	003BBBBBB8	0	0	0	0	0	00	00	00
70	31	0000000	044000C4	000000000	0077777770	0	0	0	0	0	00	00	00
71	32	0000000	08800188	000000000	00EEEEEEE0	0	1	0	1	0	01	00	00
72	33	0000000	11000311	000000000	01DDDDDDC1	0	0	0	1	0	00	01	00
73	34	0000000	22000622	000000000	03BBBBBB83	0	0	0	1	1	11	00	01
74	35	0000000	44000C44	000000000	0777777707	0	0	0	0	1	10	11	00
75	36	0000000	08001888	000000000	0EEEEEEE0E	0	0	0	1	1	01	10	11
76	37	0000000	10003111	000000000	1DDDDDDC1D	0	0	0	1	0	01	01	10
77	38	0000000	20006222	000000000	3BBBBBB83B	0	0	0	1	0	11	01	01
78	39	0000000	4000C444	000000000	7777777077	0	0	0	0	1	01	11	01
79	40	0000000	00018888	000000000	6EEEEEE0EF	0	0	0	1	0	10	01	11
80	41	0000000	00031110	000000000	5DDDDDC1DE	0	0	0	1	1	00	10	01
81	42	0000000	00062220	000000000	3BBBBB83BC	0	0	0	1	1	01	00	10
82	43	0000000	000C4440	000000000	7777770779	0	0	0	0	1	01	01	00
83	44	0000000	00188880	000000000	6EEEEE0EF2	0	0	0	1	0	11	01	01
84	45	0000000	00311100	000000000	5DDDDC1DE5	0	0	0	1	0	10	11	01
85	46	0000000	00622200	000000000	3BBBB83BCB	0	0	0	1	1	01	10	11
86	47	0000000	00C44400	000000000	7777707797	0	1	0	0	0	01	01	10
87	48	0000000	01888801	000000000	6EEEE0EF2F	0	1	0	1	1	11	01	01
88	49	0000000	03111003	000000000	5DDDC1DE5E	0	0	0	1	0	10	11	01
89	50	0000000	06222006	000000000	3BBB83BCBC	0	0	0	1	1	01	10	11
90	51	0000000	0C44400C	000000000	7777077979	0	0	0	0	1	00	01	10
91	52	0000000	18888018	000000000	6EEE0EF2F2	0	1	0	1	0	10	00	01
92	53	0000000	31110030	000000000	5DDC1DE5E5	0	0	0	1	1	11	10	00
93	54	0000000	62220060	000000000	3BB83BCBCB	0	0	0	1	0	00	11	10
94	55	0000000	444400C1	000000000	7770779797	0	0	0	0	0	10	00	11
95	56	0000000	08880183	000000000	6EE0EF2F2F	0	1	0	1	0	00	10	00
96	57	0000000	11100307	000000000	5DC1DE5E5F	0	0	0	1	1	01	00	10
97	58	0000000	2220060E	000000000	3B83BCBCBF	0	0	0	1	0	00	01	00
98	59	0000000	44400C1C	000000000	770779797E	0	0	0	0	0	11	00	01
99	60	0000000	08801838	000000000	6E0EF2F2FC	0	1	0	0	0	01	11	00
100	61	0000000	11003070	000000000	5C1DE5E5F8	0	0	0	0	1	11	01	11
101	62	0000000	220060E0	000000000	383BCBCBF0	0	0	0	0	1	01	11	01
102	63	0000000	4400C1C0	000000000	70779797E0	0	0	0	0	1	11	01	11
103	64	0000000	08018380	000000000	60EF2F2FC1	0	0	0	1	0	10	11	01
104	65	0000000	10030701	000000000	41DE5E5F82	0	0	0	1	1	01	10	11
105	66	0000000	20060E02	000000000	03BCBCBF04	0	0	0	1	0	01	01	10
106	67	0000000	400C1C05	000000000	0779797E09	0	0	0	0	1	10	01	01
107	68	0000000	0018380A	000000000	0EF2F2FC12	0	0	0	1	1	00	10	01
108	69	0000000	00307015	000000000	1DE5E5F825	0	0	0	1	1	01	00	10
109	70	0000000	0060E02A	000000000	3BCBCBF04B	0	0	0	1	0	00	01	00
110	71	0000000	00C1C055	000000000	779797E097	0	1	0	1	0	10	00	01
111	72	0000000	018380AA	000000000	6F2F2FC12F	0	1	0	0	1	11	10	00
112	73	0000000	03070154	000000000	5E5E5F825E	0	0	0	0	1	11	11	10
113	74	0000000	060E02A8	000000000	3CBCBF04BC	0	0	0	1	0	11	11	11
114	75	0000000	0C1C0550	000000000	79797E0979	0	0	0	0	1	00	11	11
115	76	0000000	18380AA0	000000000	72F2FC12F2	0	0	0	1	1	10	00	11
116	77	0000000	30701541	000000000	65E5F825E5	0	0	0	1	1	11	10	00
117	78	0000000	60E02A82	000000000	4BCBF04BCB	0	1	0	1	1	00	11	10

118	79	0000000	41C05505	000000000	1797E09796	0	1	0	1	0	11	00	11
119	80	0000000	0380AA0A	000000000	2F2FC12F2C	0	1	0	0	0	01	11	00
120	81	0000000	07015415	000000000	5E5F825E59	0	0	0	0	1	11	01	11
121	82	0000000	0E02A82A	000000000	3CBF04BCB2	0	0	0	1	0	10	11	01
122	83	0000000	1C055054	000000000	797E097964	0	0	0	0	0	01	10	11
123	84	0000000	380AA0A8	000000000	72FC12F2C9	0	0	0	1	0	01	01	10
124	85	0000000	70154151	000000000	65F825E593	0	0	0	1	0	11	01	01
125	86	0000000	602A82A3	000000000	4BF04BCB26	0	0	0	1	0	10	11	01
126	87	0000000	40550546	000000000	17E097964C	0	0	0	1	1	01	10	11
127	88	0000000	00AA0A8D	000000000	2FC12F2C99	0	1	0	1	1	01	01	10
128	89	0000000	0154151A	000000000	5F825E5932	0	0	0	1	0	11	01	01
129	90	0000000	02A82A34	000000000	3F04BCB264	0	1	0	0	0	10	11	01
130	91	0000000	05505468	000000000	7E097964C9	0	0	0	0	0	01	10	11
131	92	0000000	0AA0A8D0	000000000	7C12F2C992	0	1	0	0	0	01	01	10
132	93	0000000	154151A1	000000000	7825E59324	0	0	0	0	1	10	01	01
133	94	0000000	2A82A342	000000000	704BCB2648	0	1	0	0	1	00	10	01
134	95	0000000	55054684	000000000	6097964C91	0	0	0	1	1	01	00	10
135	96	0000000	2A0A8D09	000000000	412F2C9923	0	0	0	0	1	01	01	00
136	97	0000000	54151A12	000000000	025E593246	0	0	0	0	1	10	01	01
137	98	0000000	282A3424	000000000	04BCB2648D	0	0	0	1	1	00	10	01
138	99	0000000	50546848	000000000	097964C91A	0	0	0	0	0	01	00	10
139	100	0000000	20A8D090	000000000	12F2C99235	0	1	0	1	1	00	01	00
140	101	0000000	4151A120	000000000	25E593246A	0	0	0	1	1	11	00	01
141	102	0000000	02A34240	000000000	4BCB2648D5	0	1	0	1	1	01	11	00
142	103	0000000	05468481	000000000	17964C91AB	0	0	0	1	0	10	01	11
143	104	0000000	0A8D0903	000000000	2F2C992357	0	1	0	0	1	00	10	01
144	105	0000000	151A1206	000000000	5E593246AE	0	0	0	0	0	01	00	10
145	106	0000000	2A34240C	000000000	3CB2648D5C	0	0	0	1	0	00	01	00
146	107	0000000	54684818	000000000	7964C91AB8	0	0	0	0	0	11	00	01
147	108	0000000	28D09030	000000000	72C9923571	0	1	0	1	1	01	11	00
148	109	0000000	51A12060	000000000	6593246AE2	0	1	0	1	1	10	01	11
149	110	0000000	234240C0	000000000	4B2648D5C5	0	0	0	0	0	00	10	01
150	111	0000000	46848180	000000000	164C91AB8A	0	1	0	0	1	01	00	10
151	112	0000000	0D090301	000000000	2C99235714	0	0	0	1	0	00	01	00
152	113	0000000	1A120602	000000000	593246AE28	0	0	0	0	0	11	00	01
153	114	0000000	34240C04	000000000	32648D5C51	0	0	0	0	1	10	11	00
154	115	0000000	68481809	000000000	64C91AB8A2	0	0	0	1	1	01	10	11
155	116	0000000	50903012	000000000	4992357144	0	1	0	1	1	01	01	10
156	117	0000000	21206024	000000000	13246AE288	0	0	0	0	1	10	01	01
157	118	0000000	4240C048	000000000	2648D5C511	0	0	0	0	0	00	10	01
158	119	0000000	04818090	000000000	4C91AB8A23	0	1	0	1	0	00	00	10
159	120	0000000	09030120	000000000	1923571446	0	0	0	0	0	00	00	00
160	121	0000000	12060240	000000000	3246AE288D	0	0	0	0	0	00	00	00
161	122	0000000	240C0480	000000000	648D5C511B	0	0	0	1	1	00	00	00
162	123	0000000	48180900	000000000	491AB8A237	0	0	0	0	0	00	00	00
163	124	0000000	10301200	000000000	123571446F	0	0	0	0	0	00	00	00
164	125	0000000	20602400	000000000	246AE288DF	0	0	0	0	0	00	00	00
165	126	0000000	40C04800	000000000	48D5C511BE	0	1	0	1	0	01	00	00
166	127	0000000	01809001	000000000	11AB8A237D	0	1	0	1	1	00	01	00
167	128	0000000	03012002	000000000	23571446FA	0	0	0	0	0	11	00	01
168	129	0000000	06024004	000000000	46AE288DF5	0	0	0	1	0	01	11	00
169	130	0000000	0C048008	000000000	0D5C511BEA	0	0	0	0	1	11	01	11
170	131	0000000	18090011	000000000	1AB8A237D5	0	0	0	1	0	10	11	01
171	132	0000000	30120022	000000000	3571446FAA	0	0	0	0	0	01	10	11
172	133	0000000	60240044	000000000	6AE288DF55	0	0	0	1	0	01	01	10
173	134	0000000	40480089	000000000	55C511BEAA	0	0	0	1	0	11	01	01
174	135	0000000	00900113	000000000	2B8A237D54	0	1	0	1	1	10	11	01

175	136	0000000	01200227	000000000	571446FAA8	0	0	0	0	0	01	10	11
176	137	0000000	0240044E	000000000	2E288DF550	0	0	0	0	1	00	01	10
177	138	0000000	0480089C	000000000	5C511BEAA0	0	1	0	0	1	11	00	01
178	139	0000000	09001138	000000000	38A237D540	0	0	0	1	0	01	11	00
179	140	0000000	12002270	000000000	71446FAA81	0	0	0	0	1	11	01	11
180	141	0000000	240044E0	000000000	6288DF5503	0	0	0	1	0	10	11	01
181	142	0000000	480089C0	000000000	4511BEAA06	0	0	0	0	0	01	10	11
182	143	0000000	10011381	000000000	0A237D540D	0	0	0	0	1	00	01	10
183	144	0000000	20022702	000000000	1446FAA81A	0	0	0	0	0	11	00	01
184	145	0000000	40044E04	000000000	288DF55035	0	0	0	1	0	01	11	00
185	146	0000000	00089C08	000000000	511BEAA06A	0	0	0	0	1	11	01	11
186	147	0000000	00113810	000000000	2237D540D5	0	0	0	0	1	01	11	01
187	148	0000000	00227021	000000000	446FAA81AA	0	0	0	0	1	11	01	11
188	149	0000000	0044E042	000000000	08DF550355	0	0	0	1	0	10	11	01
189	150	0000000	0089C085	000000000	11BEAA06AA	0	1	0	1	0	10	10	11
190	151	0000000	0113810A	000000000	237D540D54	0	0	0	0	0	10	10	10
191	152	0000000	02270215	000000000	46FAA81AA9	0	0	0	1	1	10	10	10
192	153	0000000	044E042A	000000000	0DF5503553	0	0	0	1	1	10	10	10
193	154	0000000	089C0854	000000000	1BEAA06AA7	0	1	0	1	0	01	10	10
194	155	0000000	113810A8	000000000	37D540D54E	0	0	0	1	0	01	01	10
195	156	0000000	22702150	000000000	6FAA81AA9D	0	0	0	1	0	11	01	01
196	157	0000000	44E042A0	000000000	5F5503553A	0	1	0	0	0	10	11	01
197	158	0000000	09C08540	000000000	3EAA06AA75	0	1	0	1	0	10	10	11
198	159	0000000	13810A80	000000000	7D540D54EA	0	1	0	0	1	10	10	10
199	160	0000000	27021500	000000000	7AA81AA9D5	0	0	0	1	1	10	10	10
200	161	0000000	4E042A00	000000000	75503553AB	0	0	0	0	0	10	10	10
201	162	0000000	1C085400	000000000	6AA06AA756	0	0	0	1	1	10	10	10
202	163	0000000	3810A800	000000000	5540D54EAC	0	0	0	0	0	10	10	10
203	164	0000000	70215000	000000000	2A81AA9D58	0	0	0	1	1	10	10	10
204	165	0000000	6042A001	000000000	5503553AB0	0	0	0	0	0	10	10	10
205	166	0000000	40854002	000000000	2A06AA7561	0	1	0	0	1	10	10	10
206	167	0000000	010A8004	000000000	540D54EAC3	0	0	0	0	0	10	10	10
207	168	0000000	02150009	000000000	281AA9D586	0	0	0	0	0	10	10	10
208	169	0000000	042A0012	000000000	503553AB0C	0	0	0	0	0	10	10	10
209	170	0000000	08540024	000000000	206AA75618	0	0	0	0	0	10	10	10
210	171	0000000	10A80048	000000000	40D54EAC30	0	1	0	1	0	01	10	10
211	172	0000000	21500091	000000000	01AA9D5861	0	0	0	1	0	01	01	10
212	173	0000000	42A00122	000000000	03553AB0C3	0	1	0	0	0	11	01	01
213	174	0000000	05400244	000000000	06AA756186	0	0	0	1	0	10	11	01
214	175	0000000	0A800488	000000000	0D54EAC30D	0	1	0	0	1	01	10	11
215	176	0000000	15000911	000000000	1AA9D5861A	0	0	0	1	0	01	01	10
216	177	0000000	2A001223	000000000	3553AB0C35	0	0	0	0	1	10	01	01
217	178	0000000	54002446	000000000	6AA756186A	0	0	0	1	1	00	10	01
218	179	0000000	2800488D	000000000	554EAC30D5	0	0	0	0	0	01	00	10
219	180	0000000	5000911B	000000000	2A9D5861AA	0	0	0	1	0	00	01	00
220	181	0000000	20012236	000000000	553AB0C355	0	0	0	0	0	11	00	01
221	182	0000000	4002446C	000000000	2A756186AA	0	0	0	0	1	10	11	00
222	183	0000000	000488D9	000000000	54EAC30D54	0	0	0	1	1	01	10	11
223	184	0000000	000911B2	000000000	29D5861AA8	0	0	0	1	0	01	01	10
224	185	0000000	00122364	000000000	53AB0C3550	0	0	0	1	0	11	01	01
225	186	0000000	002446C8	000000000	2756186AA0	0	0	0	0	1	01	11	01
226	187	0000000	00488D90	000000000	4EAC30D540	0	0	0	1	0	10	01	11
227	188	0000000	00911B20	000000000	1D5861AA81	0	1	0	0	1	00	10	01
228	189	0000000	01223640	000000000	3AB0C35502	0	0	0	1	1	01	00	10
229	190	0000000	02446C80	000000000	756186AA05	0	0	0	0	1	01	01	00
230	191	0000000	0488D901	000000000	6AC30D540B	0	1	0	1	1	11	01	01
231	192	0000000	0911B203	000000000	55861AA817	0	0	0	1	0	10	11	01

232	193	0000000	12236407	000000000	2B0C35502F	0	0	0	0	0	01	10	11
233	194	0000000	2446C80E	000000000	56186AA05F	0	0	0	0	1	00	01	10
234	195	0000000	488D901C	000000000	2C30D540BF	0	1	0	0	1	11	00	01
235	196	0000000	111B2039	000000000	5861AA817E	0	0	0	0	1	10	11	00
236	197	0000000	22364072	000000000	30C35502FD	0	0	0	1	1	01	10	11
237	198	0000000	446C80E4	000000000	6186AA05FB	0	0	0	1	0	01	01	10
238	199	0000000	08D901C8	000000000	430D540BF6	0	1	0	0	0	11	01	01
239	200	0000000	11B20391	000000000	061AA817EC	0	1	0	0	0	10	11	01

```
Z[0]  = 3D
Z[1]  = C1
Z[2]  = F0
Z[3]  = BB
Z[4]  = 58
Z[5]  = 1E
Z[6]  = 42
Z[7]  = 42
Z[8]  = 4B
Z[9]  = 8E
Z[10] = C1
Z[11] = 2A
Z[12] = 40
Z[13] = 63
Z[14] = 7A
Z[15] = 1E

=====================================================================================
        Reload this pattern into the LFSRs
        Hold content of Summation Combiner regs and calculate new C[t+1] and Z values
=====================================================================================
LFSR1  <= 04B583D
LFSR2  <= 208E1EC1
LFSR3  <= 063C142F0
LFSR4  <= 0F7A2A42BB
C[t+1] <= 10

=====================================================================================
        Generating 125 key symbols (encryption/decryption sequence)
=====================================================================================
```

240	1	04B583D	208E1EC1	063C142F0	0F7A2A42BB	0	1	0	0	0	10	11	01
241	2	096B07A	411C3D82	0C78285E1	1EF4548577	1	0	1	1	1	10	10	11
242	3	12D60F4	02387B04	18F050BC3	3DE8A90AEF	0	0	1	1	0	01	10	10
243	4	05AC1E9	0470F609	11E0A1786	7BD15215DF	0	0	0	1	0	01	01	10
244	5	0B583D2	08E1EC13	03C142F0C	77A2A42BBF	1	1	0	1	0	00	01	01
245	6	16B07A5	11C3D827	078285E18	6F4548577E	0	1	0	0	1	11	00	01
246	7	0D60F4B	2387B04F	0F050BC30	5E8A90AEFD	1	1	1	1	1	00	11	00
247	8	1AC1E97	470F609E	1E0A17860	3D15215DFA	1	0	1	0	0	11	00	11
248	9	1583D2E	0E1EC13D	1C142F0C0	7A2A42BBF4	0	0	1	0	0	01	11	00
249	10	0B07A5D	1C3D827B	18285E181	74548577E9	1	0	1	0	1	10	01	11
250	11	160F4BB	387B04F7	1050BC302	68A90AEFD2	0	0	0	1	1	00	10	01
251	12	0C1E976	70F609EE	00A178605	515215DFA5	1	1	0	0	0	00	00	10
252	13	183D2ED	61EC13DD	0142F0C0B	22A42BBF4B	1	1	0	1	1	01	00	00
253	14	107A5DA	43D827BA	0285E1817	4548577E97	0	1	0	0	0	00	01	00
254	15	00F4BB4	07B04F74	050BC302F	0A90AEFD2E	0	1	0	1	0	10	00	01
255	16	01E9769	0F609EE8	0A178605E	15215DFA5C	0	0	1	0	1	11	10	00
256	17	03D2ED3	1EC13DD0	142F0C0BD	2A42BBF4B9	0	1	0	0	0	00	11	10
257	18	07A5DA7	3D827BA0	085E1817B	548577E972	0	1	1	1	1	11	00	11

258	19	0F4BB4F	7B04F740	10BC302F6	290AEFD2E5	1	0	0	0	0	01	11	00
259	20	1E9769F	7609EE80	0178605ED	5215DFA5CA	1	0	0	0	0	10	01	11
260	21	1D2ED3F	6C13DD01	02F0C0BDA	242BBF4B94	1	0	0	0	1	00	10	01
261	22	1A5DA7E	5827BA03	05E1817B4	48577E9729	1	0	0	0	1	01	00	10
262	23	14BB4FC	304F7407	0BC302F69	10AEFD2E53	0	0	1	1	1	00	01	00
263	24	09769F9	609EE80E	178605ED2	215DFA5CA7	1	1	0	0	0	10	00	01
264	25	12ED3F2	413DD01C	0F0C0BDA4	42BBF4B94F	0	0	1	1	0	00	10	00
265	26	05DA7E5	027BA038	1E1817B49	0577E9729F	0	0	1	0	1	01	00	10
266	27	0BB4FCA	04F74071	1C302F693	0AEFD2E53F	1	1	1	1	1	11	01	00
267	28	1769F95	09EE80E3	18605ED27	15DFA5CA7F	0	1	1	1	0	11	11	01
268	29	0ED3F2B	13DD01C6	10C0BDA4F	2BBF4B94FE	1	1	0	1	0	10	11	11
269	30	1DA7E56	27BA038D	01817B49F	577E9729FD	1	1	0	0	0	10	10	11
270	31	1B4FCAD	4F74071B	0302F693E	2EFD2E53FB	1	0	0	1	0	01	10	10
271	32	169F95B	1EE80E37	0605ED27D	5DFA5CA7F7	0	1	0	1	1	01	01	10
272	33	0D3F2B7	3DD01C6E	0C0BDA4FB	3BF4B94FEF	1	1	1	1	1	00	01	01
273	34	1A7E56F	7BA038DC	1817B49F6	77E9729FDE	1	1	1	1	0	01	00	01
274	35	14FCADF	774071B9	102F693ED	6FD2E53FBD	0	0	0	1	0	00	01	00
275	36	09F95BE	6E80E373	005ED27DB	5FA5CA7F7B	1	1	0	1	1	10	00	01
276	37	13F2B7C	5D01C6E7	00BDA4FB6	3F4B94FEF7	0	0	0	0	0	11	10	00
277	38	07E56F9	3A038DCE	017B49F6C	7E9729FDEE	0	0	0	1	0	00	11	10
278	39	0FCADF2	74071B9C	02F693ED8	7D2E53FBDD	1	0	0	0	1	10	00	11
279	40	1F95BE5	680E3738	05ED27DB0	7A5CA7F7BA	1	0	0	0	1	11	10	00
280	41	1F2B7CA	501C6E71	0BDA4FB60	74B94FEF74	1	0	1	1	0	01	11	10
281	42	1E56F94	2038DCE2	17B49F6C0	69729FDEE8	1	0	0	0	0	10	01	11
282	43	1CADF29	4071B9C4	0F693ED80	52E53FBDD1	1	0	1	1	1	11	10	01
283	44	195BE53	00E37389	1ED27DB01	25CA7F7BA3	1	1	1	1	1	01	11	10
284	45	12B7CA6	01C6E713	1DA4FB602	4B94FEF747	0	1	1	1	0	01	01	11
285	46	056F94C	038DCE26	1B49F6C04	1729FDEE8E	0	1	1	0	1	11	01	01
286	47	0ADF299	071B9C4D	1693ED808	2E53FBDD1C	1	0	0	0	0	10	11	01
287	48	15BE532	0E37389A	0D27DB011	5CA7F7BA38	0	0	1	1	0	10	10	11
288	49	0B7CA64	1C6E7135	1A4FB6022	394FEF7471	1	0	1	0	0	01	10	10
289	50	16F94C9	38DCE26A	149F6C044	729FDEE8E2	0	1	0	1	1	01	01	10
290	51	0DF2993	71B9C4D4	093ED8089	653FBDD1C4	1	1	1	0	0	00	01	01
291	52	1BE5327	637389A9	127DB0112	4A7F7BA388	1	0	0	0	1	11	00	01
292	53	17CA64E	46E71353	04FB60224	14FEF74710	0	1	0	1	1	01	11	00
293	54	0F94C9C	0DCE26A6	09F6C0448	29FDEE8E21	1	1	1	1	1	01	01	11
294	55	1F29939	1B9C4D4D	13ED80890	53FBDD1C42	1	1	0	1	0	00	01	01
295	56	1E53272	37389A9A	07DB01121	27F7BA3884	1	0	0	1	0	10	00	01
296	57	1CA64E5	6E713534	0FB602242	4FEF747108	1	0	1	1	1	00	10	00
297	58	194C9CB	5CE26A69	1F6C04485	1FDEE8E210	1	1	1	1	0	11	00	10
298	59	1299397	39C4D4D3	1ED80890A	3FBDD1C420	0	1	1	1	0	00	11	00
299	60	053272F	7389A9A6	1DB011214	7F7BA38840	0	1	1	0	0	11	00	11
300	61	0A64E5E	6713534C	1B6022428	7EF7471081	1	0	1	1	0	00	11	00
301	62	14C9CBD	4E26A699	16C044850	7DEE8E2102	0	0	0	1	1	10	00	11
302	63	099397A	1C4D4D32	0D80890A0	7BDD1C4205	1	0	1	1	1	00	10	00
303	64	13272F4	389A9A65	1B0112141	77BA38840B	0	1	1	1	1	00	00	10
304	65	064E5E8	713534CB	160224283	6F74710817	0	0	0	0	0	00	00	00
305	66	0C9CBD1	626A6997	0C0448507	5EE8E2102E	1	0	1	1	1	01	00	00
306	67	19397A3	44D4D32E	180890A0E	3DD1C4205C	1	1	1	1	1	11	01	00
307	68	1272F46	09A9A65D	10112141D	7BA38840B8	0	1	0	1	1	10	11	01
308	69	04E5E8C	13534CBA	00224283A	7747108171	0	0	0	0	0	01	10	11
309	70	09CBD19	26A69975	004485075	6E8E2102E3	1	1	0	1	0	10	01	10
310	71	1397A32	4D4D32EB	00890A0EA	5D1C4205C7	0	0	0	0	0	00	10	01
311	72	072F465	1A9A65D7	0112141D5	3A38840B8F	0	1	0	0	1	01	00	10
312	73	0E5E8CA	3534CBAF	0224283AA	747108171F	1	0	0	0	0	00	01	00
313	74	1CBD194	6A69975E	044850755	68E2102E3E	1	0	0	1	0	10	00	01
314	75	197A329	54D32EBC	0890A0EAB	51C4205C7D	1	1	1	1	0	01	10	00

315	76	12F4653	29A65D79	112141D56	238840B8FA	0	1	0	1	1	01	01	10
316	77	05E8CA6	534CBAF2	024283AAD	47108171F4	0	0	0	0	1	10	01	01
317	78	0BD194D	269975E5	04850755B	0E2102E3E9	1	1	0	0	0	11	10	01
318	79	17A329A	4D32EBCB	090A0EAB6	1C4205C7D2	0	0	1	0	0	00	11	10
319	80	0F46535	1A65D797	12141D56D	38840B8FA5	1	0	0	1	0	11	00	11
320	81	1E8CA6A	34CBAF2F	04283AADA	7108171F4B	1	1	0	0	1	01	11	00
321	82	1D194D5	69975E5F	0850755B4	62102E3E97	1	1	1	0	0	01	01	11
322	83	1A329AA	532EBCBF	10A0EAB68	44205C7D2F	1	0	0	0	0	11	01	01
323	84	1465355	265D797F	0141D56D1	0840B8FA5E	0	0	0	0	1	01	11	01
324	85	08CA6AB	4CBAF2FF	0283AADA2	108171F4BC	1	1	0	1	0	01	01	11
325	86	1194D56	1975E5FF	050755B45	2102E3E979	0	0	0	0	1	10	01	01
326	87	0329AAD	32EBCBFF	0A0EAB68A	4205C7D2F3	0	1	1	0	0	11	10	01
327	88	065355A	65D797FF	141D56D14	040B8FA5E7	0	1	0	0	0	00	11	10
328	89	0CA6AB4	4BAF2FFF	083AADA28	08171F4BCF	1	1	1	0	1	11	00	11
329	90	194D569	175E5FFF	10755B450	102E3E979E	1	0	0	0	0	01	11	00
330	91	129AAD3	2EBCBFFF	00EAB68A1	205C7D2F3C	0	1	0	0	0	10	01	11
331	92	05355A6	5D797FFF	01D56D142	40B8FA5E78	0	0	0	1	1	00	10	01
332	93	0A6AB4D	3AF2FFFE	03AADA285	0171F4BCF1	1	1	0	0	0	00	00	10
333	94	14D569B	75E5FFFD	0755B450A	02E3E979E2	0	1	0	1	0	01	00	00
334	95	09AAD37	6BCBFFFA	0EAB68A15	05C7D2F3C4	1	1	1	1	1	11	01	00
335	96	1355A6E	5797FFF4	1D56D142A	0B8FA5E788	0	1	1	1	0	11	11	01
336	97	06AB4DC	2F2FFFE8	1AADA2854	171F4BCF11	0	0	1	0	0	11	11	11
337	98	0D569B8	5E5FFFD0	155B450A9	2E3E979E23	1	0	0	0	0	11	11	11
338	99	1AAD370	3CBFFFA1	0AB68A153	5C7D2F3C46	1	1	1	0	0	10	11	11
339	100	155A6E0	797FFF43	156D142A7	38FA5E788D	0	0	0	1	1	01	10	11
340	101	0AB4DC0	72FFFE87	0ADA2854E	71F4BCF11B	1	1	1	1	1	10	01	10
341	102	1569B81	65FFFD0E	15B450A9D	63E979E236	0	1	0	1	0	11	10	01
342	103	0AD3703	4BFFFA1C	0B68A153B	47D2F3C46C	1	1	1	1	1	01	11	10
343	104	15A6E07	17FFF438	16D142A76	0FA5E788D8	0	1	0	1	1	10	01	11
344	105	0B4DC0F	2FFFE870	0DA2854EC	1F4BCF11B0	1	1	1	0	1	11	10	01
345	106	169B81F	5FFFD0E1	1B450A9D8	3E979E2360	0	1	1	1	0	01	11	10
346	107	0D3703F	3FFFA1C3	168A153B0	7D2F3C46C1	1	1	0	0	1	10	01	11
347	108	1A6E07E	7FFF4386	0D142A761	7A5E788D83	1	1	1	0	1	11	10	01
348	109	14DC0FD	7FFE870C	1A2854EC2	74BCF11B07	0	1	1	1	0	01	11	10
349	110	09B81FB	7FFD0E19	1450A9D84	6979E2360E	1	1	0	0	1	10	01	11
350	111	13703F6	7FFA1C33	08A153B09	52F3C46C1C	0	1	1	1	1	11	10	01
351	112	06E07EC	7FF43867	1142A7612	25E788D838	0	1	0	1	1	00	11	10
352	113	0DC0FD8	7FE870CF	02854EC25	4BCF11B071	1	1	0	1	1	11	00	11
353	114	1B81FB1	7FD0E19E	050A9D84B	179E2360E3	1	1	0	1	0	00	11	00
354	115	1703F62	7FA1C33D	0A153B096	2F3C46C1C7	0	1	1	0	0	11	00	11
355	116	0E07EC4	7F43867B	142A7612C	5E788D838E	1	0	0	0	0	01	11	00
356	117	1C0FD88	7E870CF6	0854EC259	3CF11B071C	1	1	1	1	1	01	01	11
357	118	181FB11	7D0E19ED	10A9D84B3	79E2360E38	1	0	0	1	1	11	01	01
358	119	103F622	7A1C33DA	0153B0967	73C46C1C71	0	0	0	1	0	10	11	01
359	120	007EC45	743867B5	02A7612CE	6788D838E3	0	0	0	1	1	01	10	11
360	121	00FD88B	6870CF6B	054EC259C	4F11B071C6	0	0	0	0	1	00	01	10
361	122	01FB117	50E19ED7	0A9D84B38	1E2360E38C	0	1	1	0	0	10	00	01
362	123	03F622F	21C33DAE	153B09671	3C46C1C718	0	1	0	0	1	11	10	00
363	124	07EC45F	43867B5C	0A7612CE2	788D838E30	0	1	1	1	0	01	11	10
364	125	0FD88BF	070CF6B9	14EC259C4	711B071C61	1	0	0	0	0	10	01	11

1.3 SECOND SET OF SAMPLE DATA

```
Initial values for the key, BD_ADDR and clock

K'c2[0]   = 00  K'c2[1]   = 00   K'c2[2]   = 00   K'c2[3]  = 00
K'c2[4]   = 00  K'c2[5]   = 00   K'c2[6]   = 00   K'c2[7]  = 00
K'c2[8]   = 00  K'c2[9]   = 00   K'c2[10]  = 00   K'c2[11] = 00
K'c2[12]  = 00  K'c2[13]  = 00   K'c2[14]  = 00   K'c2[15] = 00

Addr2[0] = 00   Addr2[1] = 00    Addr2[2] = 00
Addr2[3] = 00   Addr2[4] = 00    Addr2[5] = 00

Clk2[0]  = 00   Clk2[1]  = 00    Clk2[2]  = 00    Clk2[3]  = 03


=====================================================================================
        Fill LFSRs with initial data
=====================================================================================
```

t	clk#	LFSR1	LFSR2	LFSR3	LFSR4	X1	X2	X3	X4	Z	C[t+1]	C[t]	C[t-1]
0	0	0000000*	00000000*	000000000*	0000000000*	0	0	0	0	0	00	00	00
1	1	0000001*	00000001*	000000001*	0000000001*	0	0	0	0	0	00	00	00
2	2	0000002*	00000002*	000000002*	0000000003*	0	0	0	0	0	00	00	00
3	3	0000004*	00000004*	000000004*	0000000007*	0	0	0	0	0	00	00	00
4	4	0000008*	00000008*	000000008*	000000000E*	0	0	0	0	0	00	00	00
5	5	0000010*	00000010*	000000010*	000000001C*	0	0	0	0	0	00	00	00
6	6	0000020*	00000020*	000000020*	0000000038*	0	0	0	0	0	00	00	00
7	7	0000040*	00000040*	000000040*	0000000070*	0	0	0	0	0	00	00	00
8	8	0000080*	00000080*	000000080*	00000000E0*	0	0	0	0	0	00	00	00
9	9	0000100*	00000100*	000000100*	00000001C0*	0	0	0	0	0	00	00	00
10	10	0000200*	00000200*	000000200*	0000000380*	0	0	0	0	0	00	00	00
11	11	0000400*	00000400*	000000400*	0000000700*	0	0	0	0	0	00	00	00
12	12	0000800*	00000800*	000000800*	0000000E00*	0	0	0	0	0	00	00	00
13	13	0001000*	00001000*	000001000*	0000001C00*	0	0	0	0	0	00	00	00
14	14	0002000*	00002000*	000002000*	0000003800*	0	0	0	0	0	00	00	00
15	15	0004000*	00004000*	000004000*	0000007000*	0	0	0	0	0	00	00	00
16	16	0008000*	00008000*	000008000*	000000E000*	0	0	0	0	0	00	00	00
17	17	0010000*	00010000*	000010000*	000001C000*	0	0	0	0	0	00	00	00
18	18	0020000*	00020000*	000020000*	0000038000*	0	0	0	0	0	00	00	00
19	19	0040000*	00040000*	000040000*	0000070000*	0	0	0	0	0	00	00	00
20	20	0080000*	00080000*	000080000*	00000E0000*	0	0	0	0	0	00	00	00
21	21	0100000*	00100000*	000100000*	00001C0000*	0	0	0	0	0	00	00	00
22	22	0200000*	00200000*	000200000*	0000380000*	0	0	0	0	0	00	00	00
23	23	0400000*	00400000*	000400000*	0000700000*	0	0	0	0	0	00	00	00
24	24	0800000*	00800000*	000800000*	0000E00000*	1	1	0	0	0	01	00	00
25	25	1000000*	01000000*	001000000*	0001C00000*	0	0	0	0	0	00	00	00
26	26	0000001	02000000*	002000000*	0003800000*	0	0	0	0	0	00	00	00
27	27	0000002	04000000*	004000000*	0007000000*	0	0	0	0	0	00	00	00
28	28	0000004	08000000*	008000000*	000E000000*	0	0	0	0	0	00	00	00
29	29	0000008	10000000*	010000000*	001C000000*	0	0	0	0	0	00	00	00
30	30	0000010	20000000*	020000000*	0038000000*	0	0	0	0	0	00	00	00
31	31	0000020	40000000*	040000000*	0070000000*	0	0	0	0	0	00	00	00
32	32	0000040	00000001	080000000*	00E0000000*	0	0	1	1	0	01	00	00
33	33	0000080	00000002	100000000*	01C0000000*	0	0	0	1	1	00	00	00
34	34	0000101	00000004	000000001	0380000000*	0	0	0	1	1	00	00	00

```
35  35  0000202  00000008  000000002  0700000000*   0 0 0 0  0 00  00 00
36  36  0000404  00000010  000000004  0E00000000*   0 0 0 0  0 00  00 00
37  37  0000808  00000020  000000008  1C00000000*   0 0 0 0  0 00  00 00
38  38  0001011  00000040  000000011  3800000000*   0 0 0 0  0 00  00 00
39  39  0002022  00000080  000000022  7000000000*   0 0 0 0  0 00  00 00
=====================================================================================

        Start clocking Summation Combiner

=====================================================================================
40   1  0004044  00000100  000000044  6000000001    0 0 0 0  0 00  00 00
41   2  0008088  00000200  000000088  4000000003    0 0 0 0  0 00  00 00
42   3  0010111  00000400  000000111  0000000007    0 0 0 0  0 00  00 00
43   4  0020222  00000800  000000222  000000000E    0 0 0 0  0 00  00 00
44   5  0040444  00001001  000000444  000000001D    0 0 0 0  0 00  00 00
45   6  0080888  00002002  000000888  000000003B    0 0 0 0  0 00  00 00
46   7  0101111  00004004  000001111  0000000077    0 0 0 0  0 00  00 00
47   8  0202222  00008008  000002222  00000000EE    0 0 0 0  0 00  00 00
48   9  0404444  00010011  000004444  00000001DD    0 0 0 0  0 00  00 00
49  10  0808888  00020022  000008888  00000003BB    1 0 0 0  1 00  00 00
50  11  1011110  00040044  000011111  0000000777    0 0 0 0  0 00  00 00
51  12  0022221  00080088  000022222  0000000EEE    0 0 0 0  0 00  00 00
52  13  0044442  00100110  000044444  0000001DDD    0 0 0 0  0 00  00 00
53  14  0088884  00200220  000088888  0000003BBB    0 0 0 0  0 00  00 00
54  15  0111109  00400440  000111111  0000007777    0 0 0 0  0 00  00 00
55  16  0222212  00800880  000222222  000000EEEE    0 1 0 0  1 00  00 00
56  17  0444424  01001100  000444444  000001DDDD    0 0 0 0  0 00  00 00
57  18  0888848  02002200  000888888  000003BBBB    1 0 0 0  1 00  00 00
58  19  1111090  04004400  001111110  0000077777    0 0 0 0  0 00  00 00
59  20  0222120  08008800  002222220  00000EEEEE    0 0 0 0  0 00  00 00
60  21  0444240  10011000  004444440  00001DDDDD    0 0 0 0  0 00  00 00
61  22  0888480  20022000  008888880  00003BBBBB    1 0 0 0  1 00  00 00
62  23  1110900  40044000  011111100  0000777777    0 0 0 0  0 00  00 00
63  24  0221200  00088001  022222200  0000EEEEEE    0 0 0 0  0 00  00 00
64  25  0442400  00110003  044444400  0001DDDDDD    0 0 0 0  0 00  00 00
65  26  0884800  00220006  088888800  0003BBBBBB    1 0 1 0  0 01  00 00
66  27  1109000  0044000C  111111000  0007777777    0 0 0 0  1 01  01 00
67  28  0212001  00880018  022222001  000EEEEEEE    0 1 0 0  0 11  01 01
68  29  0424002  01100031  044444002  001DDDDDDC    0 0 0 0  1 01  11 01
69  30  0848004  02200062  088888004  003BBBBBB8    1 0 1 0  1 10  01 11
70  31  1090008  044000C4  111110008  0077777770    0 0 0 0  0 00  10 01
71  32  0120010  08800188  022220010  00EEEEEEE0    0 1 0 1  0 00  00 10
72  33  0240020  11000311  044440020  01DDDDDDC1    0 0 0 1  1 00  00 00
73  34  0480040  22000622  088880040  03BBBBBB83    0 0 1 1  0 01  00 00
74  35  0900081  44000C44  111100080  0777777707    1 0 0 0  0 00  01 00
75  36  1200103  08001888  022200101  0EEEEEEE0E    0 0 0 1  1 11  00 01
76  37  0400207  10003111  044400202  1DDDDDDC1D    0 0 0 1  0 01  11 00
77  38  080040E  20006222  088800404  3BBBBBB83B    1 0 1 1  0 01  01 11
78  39  100081C  4000C444  111000808  7777777077    0 0 0 0  1 10  01 01
79  40  0001038  00018888  022001010  6EEEEEE0EF    0 0 0 1  1 00  10 01
80  41  0002070  00031110  044002020  5DDDDDC1DE    0 0 0 1  1 01  00 10
81  42  00040E0  00062220  088004040  3BBBBB83BC    0 0 1 1  1 00  01 00
82  43  00081C1  000C4440  110008081  7777770779    0 0 0 0  0 11  00 01
83  44  0010383  00188880  020010103  6EEEEE0EF2    0 0 0 1  0 01  11 00
84  45  0020707  00311100  040020206  5DDDDC1DE5    0 0 0 1  0 10  01 11
85  46  0040E0E  00622200  08004040C  3BBBB83BCB    0 0 1 1  0 11  10 01
86  47  0081C1D  00C44400  100080819  7777707797    0 1 0 0  0 00  11 10
87  48  010383A  01888801  000101032  6EEEE0EF2F    0 1 0 1  0 11  00 11
88  49  0207075  03111003  000202064  5DDDC1DE5E    0 0 0 1  0 01  11 00
```

89	50	040E0EA	06222006	0004040C8	3BBB83BCBC	0	0	0	1	0	10	01	11
90	51	081C1D5	0C44400C	000808191	7777077979	1	0	0	0	1	00	10	01
91	52	10383AB	18888018	001010323	6EEE0EF2F2	0	1	0	1	0	00	00	10
92	53	0070756	31110030	002020646	5DDC1DE5E5	0	0	0	1	1	00	00	00
93	54	00E0EAC	62220060	004040C8C	3BB83BCBCB	0	0	0	1	1	00	00	00
94	55	01C1D59	444400C1	008081919	7770779797	0	0	0	0	0	00	00	00
95	56	0383AB2	08880183	010103232	6EE0EF2F2F	0	1	0	1	0	01	00	00
96	57	0707565	11100307	020206464	5DC1DE5E5F	0	0	0	1	0	00	01	00
97	58	0E0EACA	2220060E	04040C8C8	3B83BCBCBF	1	0	0	1	0	10	00	01
98	59	1C1D594	44400C1C	080819191	770779797E	1	0	1	0	0	00	10	00
99	60	183AB28	08801838	101032323	6E0EF2F2FC	1	1	0	0	0	00	00	10
100	61	1075650	11003070	002064647	5C1DE5E5F8	0	0	0	0	0	00	00	00
101	62	00EACA1	220060E0	0040C8C8E	383BCBCBF0	0	0	0	0	0	00	00	00
102	63	01D5943	4400C1C0	00819191D	70779797E0	0	0	0	0	0	00	00	00
103	64	03AB286	08018380	01032323A	60EF2F2FC1	0	0	0	1	1	00	00	00
104	65	075650C	10030701	020646475	41DE5E5F82	0	0	0	1	1	00	00	00
105	66	0EACA18	20060E02	040C8C8EA	03BCBCBF04	1	0	0	1	0	01	00	00
106	67	1D59430	400C1C05	0819191D4	0779797E09	1	0	1	0	1	00	01	00
107	68	1AB2861	0018380A	1032323A9	0EF2F2FC12	1	0	0	1	0	10	00	01
108	69	15650C3	00307015	006464752	1DE5E5F825	0	0	0	1	1	11	10	00
109	70	0ACA186	0060E02A	00C8C8EA4	3BCBCBF04B	1	0	0	1	1	00	11	10
110	71	159430C	00C1C055	019191D48	779797E097	0	1	0	1	0	11	00	11
111	72	0B28618	018380AA	032323A90	6F2F2FC12F	1	1	0	0	1	01	11	00
112	73	1650C30	03070154	064647520	5E5E5F825E	0	0	0	0	1	11	01	11
113	74	0CA1860	060E02A8	0C8C8EA40	3CBCBF04BC	1	0	1	1	0	11	11	01
114	75	19430C0	0C1C0550	19191D480	79797E0979	1	0	1	0	1	11	11	11
115	76	1286180	18380AA0	12323A900	72F2FC12F2	0	0	0	1	0	11	11	11
116	77	050C301	30701541	046475201	65E5F825E5	0	0	0	1	0	11	11	11
117	78	0A18602	60E02A82	08C8EA402	4BCBF04BCB	1	1	1	1	1	10	11	11
118	79	1430C04	41C05505	1191D4804	1797E09796	0	1	0	1	0	10	10	11
119	80	0861808	0380AA0A	0323A9008	2F2FC12F2C	1	1	0	0	0	01	10	10
120	81	10C3011	07015415	064752011	5E5F825E59	0	0	0	0	1	00	01	10
121	82	0186022	0E02A82A	0C8EA4022	3CBF04BCB2	0	0	1	1	0	10	00	01
122	83	030C045	1C055054	191D48044	797E097964	0	0	1	0	1	11	10	00
123	84	061808A	380AA0A8	123A90088	72FC12F2C9	0	0	0	1	0	00	11	10
124	85	0C30115	70154151	047520111	65F825E593	1	0	0	1	0	11	00	11
125	86	186022A	602A82A3	08EA40222	4BF04BCB26	1	0	1	1	0	00	11	00
126	87	10C0455	40550546	11D480444	17E097964C	0	0	0	1	1	10	00	11
127	88	01808AA	00AA0A8D	03A900888	2FC12F2C99	0	1	0	1	0	00	10	00
128	89	0301155	0154151A	075201111	5F825E5932	0	0	0	1	1	01	00	10
129	90	06022AA	02A82A34	0EA402222	3F04BCB264	0	1	1	0	1	00	01	00
130	91	0C04555	05505468	1D4804445	7E097964C9	1	0	1	0	0	10	00	01
131	92	1808AAA	0AA0A8D0	1A900888A	7C12F2C992	1	1	1	0	1	00	10	00
132	93	1011555	154151A1	152011115	7825E59324	0	0	0	0	0	01	00	10
133	94	0022AAB	2A82A342	0A402222B	704BCB2648	0	1	1	0	1	00	01	00
134	95	0045556	55054684	148044457	6097964C91	0	0	0	1	1	11	00	01
135	96	008AAAC	2A0A8D09	0900888AE	412F2C9923	0	0	1	0	0	01	11	00
136	97	0115559	54151A12	12011115D	025E593246	0	0	0	0	1	11	01	11
137	98	022AAB2	282A3424	0402222BA	04BCB2648D	0	0	0	1	0	10	11	01
138	99	0455564	50546848	080444575	097964C91A	0	0	1	0	1	01	10	11
139	100	08AAAC8	20A8D090	100888AEA	12F2C99235	1	1	0	1	0	10	01	10
140	101	1155591	4151A120	0011115D5	25E593246A	0	0	0	1	1	00	10	01
141	102	02AAB22	02A34240	002222BAA	4BCB2648D5	0	1	0	1	0	00	00	10
142	103	0555644	05468481	004445755	17964C91AB	0	0	0	1	1	00	00	00
143	104	0AAAC88	0A8D0903	00888AEAA	2F2C992357	1	1	0	0	0	01	00	00
144	105	1555911	151A1206	011115D55	5E593246AE	0	0	0	0	1	01	01	00
145	106	0AAB222	2A34240C	02222BAAA	3CB2648D5C	1	0	0	1	1	11	01	01

146	107	1556445	54684818	044457555	7964C91AB8	0	0	0	0	1	01	11	01
147	108	0AAC88B	28D09030	0888AEAAA	72C9923571	1	1	1	1	1	01	01	11
148	109	1559117	51A12060	11115D555	6593246AE2	0	1	0	1	1	11	01	01
149	110	0AB222F	234240C0	0222BAAAB	4B2648D5C5	1	0	0	0	0	10	11	01
150	111	156445F	46848180	044575557	164C91AB8A	0	1	0	0	1	01	10	11
151	112	0AC88BF	0D090301	088AEAAAE	2C99235714	1	0	1	1	0	10	01	10
152	113	159117F	1A120602	1115D555D	593246AE28	0	0	0	0	0	00	10	01
153	114	0B222FE	34240C04	022BAAABA	32648D5C51	1	0	0	0	1	01	00	10
154	115	16445FD	68481809	045755574	64C91AB8A2	0	0	0	1	0	00	01	00
155	116	0C88BFA	50903012	08AEAAAE8	4992357144	1	1	1	1	0	01	00	01
156	117	19117F5	21206024	115D555D1	13246AE288	1	0	0	0	0	00	01	00
157	118	1222FEA	4240C048	02BAAABA2	2648D5C511	0	0	0	0	0	11	00	01
158	119	0445FD5	04818090	057555744	4C91AB8A23	0	1	0	1	1	01	11	00
159	120	088BFAA	09030120	0AEAAAE88	1923571446	1	0	1	0	1	10	01	11
160	121	1117F55	12060240	15D555D11	3246AE288D	0	0	0	0	0	00	10	01
161	122	022FEAA	240C0480	0BAAABA22	648D5C511B	0	0	1	1	0	00	00	10
162	123	045FD54	48180900	175557444	491AB8A237	0	0	0	0	0	00	00	00
163	124	08BFAA9	10301200	0EAAAE889	123571446F	1	0	1	0	0	01	00	00
164	125	117F553	20602400	1D555D113	246AE288DF	0	0	1	0	0	00	01	00
165	126	02FEAA7	40C04800	1AAABA227	48D5C511BE	0	1	1	1	1	10	00	01
166	127	05FD54F	01809001	15557444F	11AB8A237D	0	1	0	1	0	00	10	00
167	128	0BFAA9F	03012002	0AAAE889E	23571446FA	1	0	1	0	0	00	00	10
168	129	17F553F	06024004	1555D113D	46AE288DF5	0	0	0	1	1	00	00	00
169	130	0FEAA7E	0C048008	0AABA227A	0D5C511BEA	1	0	1	0	0	01	00	00
170	131	1FD54FC	18090011	1557444F5	1AB8A237D5	1	0	0	1	1	00	01	00
171	132	1FAA9F9	30120022	0AAE889EB	3571446FAA	1	0	1	0	0	10	00	01
172	133	1F553F2	60240044	155D113D7	6AE288DF55	1	0	0	1	0	00	10	00
173	134	1EAA7E4	40480089	0ABA227AE	55C511BEAA	1	0	1	1	1	00	00	10
174	135	1D54FC9	00900113	157444F5D	2B8A237D54	1	1	0	1	1	01	00	00
175	136	1AA9F93	01200227	0AE889EBA	571446FAA8	1	0	1	0	1	00	01	00
176	137	1553F26	0240044E	15D113D75	2E288DF550	0	0	0	0	0	11	00	01
177	138	0AA7E4C	0480089C	0BA227AEA	5C511BEAA0	1	1	1	0	0	00	11	00
178	139	154FC98	09001138	17444F5D4	38A237D540	0	0	0	1	1	10	00	11
179	140	0A9F931	12002270	0E889EBA9	71446FAA81	1	0	1	0	0	00	10	00
180	141	153F262	240044E0	1D113D753	6288DF5503	0	0	1	1	0	00	00	10
181	142	0A7E4C5	480089C0	1A227AEA7	4511BEAA06	1	0	1	0	0	01	00	00
182	143	14FC98B	10011381	1444F5D4F	0A237D540D	0	0	0	0	1	01	01	00
183	144	09F9316	20022702	0889EBA9E	1446FAA81A	1	0	1	0	1	11	01	01
184	145	13F262D	40044E04	1113D753D	288DF55035	0	0	0	1	0	10	11	01
185	146	07E4C5A	00089C08	0227AEA7A	511BEAA06A	0	0	0	0	0	01	10	11
186	147	0FC98B4	00113810	044F5D4F5	2237D540D5	1	0	0	0	0	01	01	10
187	148	1F93169	00227021	089EBA9EB	446FAA81AA	1	0	1	0	1	11	01	01
188	149	1F262D2	0044E042	113D753D7	08DF550355	1	0	0	1	1	10	11	01
189	150	1E4C5A4	0089C085	027AEA7AE	11BEAA06AA	1	1	0	1	1	10	10	11
190	151	1C98B48	0113810A	04F5D4F5C	237D540D54	1	0	0	0	1	10	10	10
191	152	1931691	02270215	09EBA9EB8	46FAA81AA9	1	0	1	1	1	01	10	10
192	153	1262D22	044E042A	13D753D71	0DF5503553	0	0	0	1	0	01	01	10
193	154	04C5A44	089C0854	07AEA7AE2	1BEAA06AA7	0	1	0	1	1	11	01	01
194	155	098B488	113810A8	0F5D4F5C4	37D540D54E	1	0	1	1	0	11	11	01
195	156	1316910	22702150	1EBA9EB89	6FAA81AA9D	0	0	1	1	1	11	11	11
196	157	062D220	44E042A0	1D753D712	5F5503553A	0	1	1	0	1	11	11	11
197	158	0C5A440	09C08540	1AEA7AE25	3EAA06AA75	1	1	1	1	1	10	11	11
198	159	18B4880	13810A80	15D4F5C4B	7D540D54EA	1	1	0	0	0	10	10	11
199	160	1169100	27021500	0BA9EB897	7AA81AA9D5	0	0	1	1	0	01	10	10
200	161	02D2201	4E042A00	1753D712E	75503553AB	0	0	0	0	1	00	01	10
201	162	05A4403	1C085400	0EA7AE25C	6AA06AA756	0	0	1	1	0	10	00	01
202	163	0B48807	3810A800	1D4F5C4B8	5540D54EAC	1	0	1	0	0	00	10	00

203	164	169100F	70215000	1A9EB8971	2A81AA9D58	0	0	1	1	0	00	00	10
204	165	0D2201E	6042A001	153D712E3	5503553AB0	1	0	0	0	1	00	00	00
205	166	1A4403C	40854002	0A7AE25C6	2A06AA7561	1	1	1	0	1	01	00	00
206	167	1488079	010A8004	14F5C4B8D	540D54EAC3	0	0	0	0	1	01	01	00
207	168	09100F2	02150009	09EB8971B	281AA9D586	1	0	1	0	1	11	01	01
208	169	12201E5	042A0012	13D712E37	503553AB0C	0	0	0	0	1	01	11	01
209	170	04403CA	08540024	07AE25C6E	206AA75618	0	0	0	0	1	11	01	11
210	171	0880795	10A80048	0F5C4B8DD	40D54EAC30	1	1	1	1	1	11	11	01
211	172	1100F2A	21500091	1EB8971BA	01AA9D5861	0	0	1	1	1	11	11	11
212	173	0201E54	42A00122	1D712E374	03553AB0C3	0	1	1	0	1	11	11	11
213	174	0403CA9	05400244	1AE25C6E9	06AA756186	0	0	1	1	1	11	11	11
214	175	0807952	0A800488	15C4B8DD3	0D54EAC30D	1	1	0	0	1	11	11	11
215	176	100F2A5	15000911	0B8971BA6	1AA9D5861A	0	0	1	1	1	11	11	11
216	177	001E54A	2A001223	1712E374C	3553AB0C35	0	0	0	0	1	00	11	11
217	178	003CA94	54002446	0E25C6E98	6AA756186A	0	0	1	1	0	11	00	11
218	179	0079528	2800488D	1C4B8DD31	554EAC30D5	0	0	1	0	0	01	11	00
219	180	00F2A50	5000911B	18971BA62	2A9D5861AA	0	0	1	1	1	10	01	11
220	181	01E54A0	20012236	112E374C4	553AB0C355	0	0	0	0	0	00	10	01
221	182	03CA940	4002446C	025C6E988	2A756186AA	0	0	0	0	0	01	00	10
222	183	0795280	000488D9	04B8DD310	54EAC30D54	0	0	0	1	0	00	01	00
223	184	0F2A500	000911B2	0971BA620	29D5861AA8	1	0	1	1	1	10	00	01
224	185	1E54A00	00122364	12E374C40	53AB0C3550	1	0	0	1	0	00	10	00
225	186	1CA9400	002446C8	05C6E9880	2756186AA0	1	0	0	0	1	01	00	10
226	187	1952800	00488D90	0B8DD3101	4EAC30D540	1	0	1	1	0	11	01	00
227	188	12A5000	00911B20	171BA6202	1D5861AA81	0	1	0	0	0	10	11	01
228	189	054A000	01223640	0E374C404	3AB0C35502	0	0	1	1	0	10	10	11
229	190	0A94000	02446C80	1C6E98808	756186AA05	1	0	1	0	0	01	10	10
230	191	1528001	0488D901	18DD31011	6AC30D540B	0	1	1	1	0	10	01	10
231	192	0A50003	0911B203	11BA62023	55861AA817	1	0	0	1	0	11	10	01
232	193	14A0006	12236407	0374C4047	2B0C35502F	0	0	0	0	1	11	11	10
233	194	094000C	2446C80E	06E98808E	56186AA05F	1	0	0	0	0	11	11	11
234	195	1280018	488D901C	0DD31011D	2C30D540BF	0	1	1	0	1	11	11	11
235	196	0500030	111B2039	1BA62023A	5861AA817E	0	0	1	0	0	11	11	11
236	197	0A00060	22364072	174C40475	30C35502FD	1	0	0	1	1	11	11	11
237	198	14000C0	446C80E4	0E98808EA	6186AA05FB	0	0	1	1	1	11	11	11
238	199	0800180	08D901C8	1D31011D5	430D540BF6	1	1	1	0	0	10	11	11
239	200	1000301	11B20391	1A62023AB	061AA817EC	0	1	1	0	0	10	10	11

```
Z[0]  = 25
Z[1]  = 45
Z[2]  = 6B
Z[3]  = 55
Z[4]  = 5F
Z[5]  = C2
Z[6]  = 20
Z[7]  = E5
Z[8]  = C4
Z[9]  = F8
Z[10] = 3A
Z[11] = F1
Z[12] = FF
Z[13] = 89
Z[14] = 02
Z[15] = 35
```

```
================================================================================================
        Reload this pattern into the LFSRs
```

```
        Hold content of Summation Combiner regs and calculate new C[t+1] and Z values
=============================================================================================
LFSR1  <= 1C45F25
LFSR2  <= 7FF8C245
LFSR3  <= 1893A206B
LFSR4  <= 1A02F1E555
C[t+1] <= 10

=============================================================================================
        Generating 125 key symbols (encryption/decryption sequence)
=============================================================================================
240    1   1C45F25  7FF8C245  1893A206B  1A02F1E555    1  1  1  0    1  10     10  11
241    2   188BE4A  7FF1848B  1127440D7  3405E3CAAB    1  1  0  0    0  01     10  10
242    3   1117C95  7FE30917  024E881AF  680BC79557    0  1  0  0    0  01     01  10
243    4   022F92B  7FC6122F  049D1035E  50178F2AAF    0  1  0  0    0  11     01  01
244    5   045F257  7F8C245E  093A206BD  202F1E555E    0  1  1  0    1  10     11  01
245    6   08BE4AE  7F1848BC  127440D7A  405E3CAABC    1  0  0  0    1  01     10  11
246    7   117C95C  7E309178  04E881AF4  00BC795579    0  0  0  1    0  01     01  10
247    8   02F92B8  7C6122F0  09D1035E8  0178F2AAF2    0  0  1  0    0  11     01  01
248    9   05F2570  78C245E1  13A206BD0  02F1E555E5    0  1  0  1    1  10     11  01
249   10   0BE4AE1  71848BC2  07440D7A0  05E3CAABCA    1  1  0  1    1  10     10  11
250   11   17C95C3  63091784  0E881AF40  0BC7955795    0  0  1  1    0  01     10  10
251   12   0F92B87  46122F09  1D1035E80  178F2AAF2B    1  0  1  1    0  10     01  10
252   13   1F2570F  0C245E12  1A206BD01  2F1E555E56    1  0  1  0    0  11     10  01
253   14   1E4AE1F  1848BC25  1440D7A03  5E3CAABCAC    1  0  0  0    0  00     11  10
254   15   1C95C3E  3091784A  0881AF407  3C79557958    1  1  1  0    1  11     00  11
255   16   192B87D  6122F094  11035E80F  78F2AAF2B1    1  0  0  1    1  01     11  00
256   17   12570FA  4245E128  0206BD01E  71E555E562    0  0  0  1    0  10     01  11
257   18   04AE1F4  048BC250  040D7A03D  63CAABCAC5    0  1  0  1    0  11     10  01
258   19   095C3E8  091784A0  081AF407A  479557958A    1  0  1  1    0  01     11  10
259   20   12B87D1  122F0941  1035E80F4  0F2AAF2B14    0  0  0  0    1  11     01  11
260   21   0570FA3  245E1283  006BD01E9  1E555E5628    0  0  0  0    1  01     11  01
261   22   0AE1F46  48BC2506  00D7A03D2  3CAABCAC50    1  1  0  1    0  01     01  11
262   23   15C3E8C  11784A0C  01AF407A5  79557958A0    0  0  0  0    1  10     01  01
263   24   0B87D18  22F09419  035E80F4A  72AAF2B140    1  1  0  1    1  11     10  01
264   25   170FA30  45E12832  06BD01E94  6555E56280    0  1  0  0    0  00     11  10
265   26   0E1F460  0BC25065  0D7A03D28  4AABCAC501    1  1  1  1    0  00     00  11
266   27   1C3E8C0  1784A0CB  1AF407A50  1557958A03    1  1  1  0    1  01     00  00
267   28   187D181  2F094196  15E80F4A0  2AAF2B1406    1  0  0  1    1  00     01  00
268   29   10FA302  5E12832C  0BD01E941  555E56280C    0  0  1  0    1  11     00  01
269   30   01F4604  3C250658  17A03D283  2ABCAC5019    0  0  0  1    0  01     11  00
270   31   03E8C09  784A0CB0  0F407A506  557958A033    0  0  1  0    0  10     01  11
271   32   07D1812  70941960  1E80F4A0C  2AF2B14066    0  1  1  1    1  11     10  01
272   33   0FA3024  612832C1  1D01E9419  55E56280CD    1  0  1  1    0  01     11  10
273   34   1F46049  42506583  1A03D2832  2BCAC5019A    1  0  1  1    0  01     01  11
274   35   1E8C093  04A0CB07  1407A5065  57958A0335    1  1  0  1    0  00     01  01
275   36   1D18127  0941960F  080F4A0CB  2F2B14066B    1  0  1  0    0  10     00  01
276   37   1A3024F  12832C1F  101E94196  5E56280CD7    1  1  0  0    0  00     10  00
277   38   146049F  2506583E  003D2832C  3CAC5019AE    0  0  0  1    1  01     00  10
278   39   08C093E  4A0CB07D  007A50658  7958A0335D    1  0  0  0    0  00     01  00
279   40   118127C  141960FA  00F4A0CB0  72B14066BA    0  0  0  1    1  11     00  01
280   41   03024F8  2832C1F4  01E941961  656280CD74    0  0  0  0    1  10     11  00
281   42   06049F1  506583E9  03D2832C2  4AC5019AE9    0  0  0  1    1  01     10  11
282   43   0C093E2  20CB07D2  07A506585  158A0335D3    1  1  0  1    0  10     01  10
283   44   18127C5  41960FA5  0F4A0CB0B  2B14066BA7    1  1  1  0    1  11     10  01
284   45   1024F8A  032C1F4B  1E9419616  56280CD74F    0  0  1  0    0  00     11  10
285   46   0049F15  06583E97  1D2832C2C  2C5019AE9F    0  0  1  0    1  10     00  11
```

286	47	0093E2B	0CB07D2F	1A5065859	58A0335D3E	0	1	1	1	1	00	10	00
287	48	0127C56	1960FA5E	14A0CB0B2	314066BA7D	0	0	0	0	0	01	00	10
288	49	024F8AD	32C1F4BC	094196164	6280CD74FB	0	1	1	1	0	11	01	00
289	50	049F15A	6583E978	12832C2C8	45019AE9F6	0	1	0	0	0	10	11	01
290	51	093E2B5	4B07D2F0	050658591	0A0335D3ED	1	0	0	0	1	01	10	11
291	52	127C56B	160FA5E0	0A0CB0B22	14066BA7DA	0	0	1	0	0	01	01	10
292	53	04F8AD7	2C1F4BC1	141961645	280CD74FB5	0	0	0	0	1	10	01	01
293	54	09F15AF	583E9783	0832C2C8A	5019AE9F6A	1	0	1	0	0	11	10	01
294	55	13E2B5E	307D2F06	106585915	20335D3ED5	0	0	0	0	1	11	11	10
295	56	07C56BD	60FA5E0D	00CB0B22B	4066BA7DAA	0	1	0	0	0	11	11	11
296	57	0F8AD7A	41F4BC1B	019616457	00CD74FB54	1	1	0	1	0	10	11	11
297	58	1F15AF4	03E97836	032C2C8AF	019AE9F6A9	1	1	0	1	1	10	10	11
298	59	1E2B5E9	07D2F06C	06585915E	0335D3ED52	1	1	0	0	0	01	10	10
299	60	1C56BD2	0FA5E0D8	0CB0B22BC	066BA7DAA4	1	1	1	0	0	10	01	10
300	61	18AD7A5	1F4BC1B0	196164578	0CD74FB549	1	0	1	1	1	11	10	01
301	62	115AF4B	3E978361	12C2C8AF0	19AE9F6A92	0	1	0	1	1	00	11	10
302	63	02B5E96	7D2F06C2	0585915E0	335D3ED524	0	0	0	0	0	10	00	11
303	64	056BD2D	7A5E0D85	0B0B22BC1	66BA7DAA49	0	0	1	1	0	00	10	00
304	65	0AD7A5B	74BC1B0A	161645783	4D74FB5493	1	1	0	0	0	00	00	10
305	66	15AF4B6	69783615	0C2C8AF07	1AE9F6A926	0	0	1	1	0	01	00	00
306	67	0B5E96D	52F06C2B	185915E0F	35D3ED524C	1	1	1	1	1	11	01	00
307	68	16BD2DB	25E0D857	10B22BC1F	6BA7DAA499	0	1	0	1	1	10	11	01
308	69	0D7A5B7	4BC1B0AF	01645783F	574FB54933	1	1	0	0	0	10	10	11
309	70	1AF4B6F	1783615F	02C8AF07F	2E9F6A9266	1	1	0	1	1	01	10	10
310	71	15E96DF	2F06C2BF	05915E0FF	5D3ED524CC	0	0	0	0	1	00	01	10
311	72	0BD2DBF	5E0D857F	0B22BC1FE	3A7DAA4998	1	0	1	0	0	10	00	01
312	73	17A5B7F	3C1B0AFE	1645783FD	74FB549331	0	0	0	1	1	11	10	00
313	74	0F4B6FF	783615FD	0C8AF07FA	69F6A92662	1	0	1	1	0	01	11	10
314	75	1E96DFF	706C2BFB	1915E0FF5	53ED524CC4	1	0	1	1	0	01	01	11
315	76	1D2DBFE	60D857F6	122BC1FEB	27DAA49988	1	1	0	1	0	00	01	01
316	77	1A5B7FD	41B0AFEC	045783FD7	4FB5493310	1	1	0	1	1	10	00	01
317	78	14B6FFA	03615FD8	08AF07FAE	1F6A926620	0	0	1	0	1	11	10	00
318	79	096DFF4	06C2BFB1	115E0FF5D	3ED524CC40	1	1	0	1	0	01	11	10
319	80	12DBFE8	0D857F63	02BC1FEBB	7DAA499881	0	1	0	1	1	10	01	11
320	81	05B7FD0	1B0AFEC6	05783FD77	7B54933103	0	0	0	0	0	00	10	01
321	82	0B6FFA1	3615FD8C	0AF07FAEF	76A9266206	1	0	1	1	1	00	00	10
322	83	16DFF42	6C2BFB18	15E0FF5DE	6D524CC40C	0	0	0	0	0	00	00	00
323	84	0DBFE85	5857F631	0BC1FEBBD	5AA4998819	1	0	1	1	1	01	00	00
324	85	1B7FD0B	30AFEC62	1783FD77A	3549331033	1	1	0	0	1	00	01	00
325	86	16FFA16	615FD8C5	0F07FAEF5	6A92662067	0	0	1	1	0	10	00	01
326	87	0DFF42D	42BFB18B	1E0FF5DEA	5524CC40CE	1	1	1	0	1	00	10	00
327	88	1BFE85B	057F6317	1C1FEBBD5	2A4998819C	1	0	1	0	0	00	00	10
328	89	17FD0B7	0AFEC62E	183FD77AA	5493310339	0	1	1	1	1	01	00	00
329	90	0FFA16F	15FD8C5C	107FAEF55	2926620672	1	1	0	0	1	00	01	00
330	91	1FF42DF	2BFB18B9	00FF5DEAA	524CC40CE5	1	1	0	0	0	10	00	01
331	92	1FE85BF	57F63172	01FEBBD55	24998819CA	1	1	0	1	1	00	10	00
332	93	1FD0B7F	2FEC62E4	03FD77AAA	4933103394	1	1	0	0	0	00	00	10
333	94	1FA16FF	5FD8C5C9	07FAEF555	1266206728	1	1	0	0	0	01	00	00
334	95	1F42DFF	3FB18B93	0FF5DEAAA	24CC40CE51	1	1	1	1	1	11	01	00
335	96	1E85BFF	7F631727	1FEBBD554	4998819CA3	1	0	1	1	0	11	11	01
336	97	1D0B7FE	7EC62E4F	1FD77AAA9	1331033947	1	1	1	0	0	10	11	11
337	98	1A16FFC	7D8C5C9F	1FAEF5553	266206728E	1	1	1	0	1	10	10	11
338	99	142DFF9	7B18B93F	1F5DEAAA7	4CC40CE51D	0	0	1	1	0	01	10	10
339	100	085BFF3	7631727F	1EBBD554E	198819CA3B	1	0	1	1	0	10	01	10
340	101	10B7FE6	6C62E4FF	1D77AAA9C	3310339477	0	0	1	0	1	00	10	01
341	102	016FFCC	58C5C9FE	1AEF55538	66206728EE	0	1	1	0	0	00	00	10
342	103	02DFF98	318B93FC	15DEAAA70	4C40CE51DC	0	1	0	0	1	00	00	00

343	104	05BFF31	631727F8	0BBD554E1	18819CA3B9	0	0	1	1	0	01	00	00
344	105	0B7FE62	462E4FF1	177AAA9C2	3103394772	1	0	0	0	0	00	01	00
345	106	16FFCC5	0C5C9FE2	0EF555384	6206728EE4	0	0	1	0	1	11	00	01
346	107	0DFF98A	18B93FC4	1DEAAA709	440CE51DC9	1	1	1	0	0	00	11	00
347	108	1BFF315	31727F88	1BD554E12	0819CA3B93	1	0	1	0	0	11	00	11
348	109	17FE62A	62E4FF11	17AAA9C24	1033947726	0	1	0	0	0	01	11	00
349	110	0FFCC54	45C9FE22	0F5553849	206728EE4C	1	1	1	0	0	01	01	11
350	111	1FF98A8	0B93FC44	1EAAA7093	40CE51DC99	1	1	1	1	1	00	01	01
351	112	1FF3150	1727F889	1D554E127	019CA3B933	1	0	1	1	1	10	00	01
352	113	1FE62A0	2E4FF112	1AAA9C24F	0339477267	1	0	1	0	0	00	10	00
353	114	1FCC541	5C9FE225	15553849E	06728EE4CF	1	1	0	0	0	00	00	10
354	115	1F98A82	393FC44B	0AAA7093C	0CE51DC99F	1	0	1	1	1	01	00	00
355	116	1F31504	727F8897	1554E1279	19CA3B933E	1	0	0	1	1	00	01	00
356	117	1E62A09	64FF112F	0AA9C24F2	339477267D	1	1	1	1	0	01	00	01
357	118	1CC5412	49FE225E	1553849E4	6728EE4CFB	1	1	0	0	1	00	01	00
358	119	198A824	13FC44BC	0AA7093C9	4E51DC99F7	1	1	1	0	1	10	00	01
359	120	1315049	27F88979	154E12792	1CA3B933EE	0	1	0	1	0	00	10	00
360	121	062A093	4FF112F3	0A9C24F24	39477267DC	0	1	1	0	0	00	00	10
361	122	0C54127	1FE225E6	153849E48	728EE4CFB8	1	1	0	1	1	01	00	00
362	123	18A824E	3FC44BCD	0A7093C91	651DC99F71	1	1	1	0	0	11	01	00
363	124	115049C	7F88979A	14E127922	4A3B933EE2	0	1	0	0	0	10	11	01
364	125	02A0938	7F112F35	09C24F244	1477267DC5	0	0	1	0	1	01	10	11

1.4 THIRD SET OF SAMPLES

```
Initial values for the key, pan address and clock

K'c3[0]  = FF   K'c3[1]  = FF   K'c3[2]  = FF   K'c3[3]  = FF
K'c3[4]  = FF   K'c3[5]  = FF   K'c3[6]  = FF   K'c3[7]  = FF
K'c3[8]  = FF   K'c3[9]  = FF   K'c3[10] = FF   K'c3[11] = FF
K'c3[12] = FF   K'c3[13] = FF   K'c3[14] = FF   K'c3[15] = FF

Addr3[0] = FF   Addr3[1] = FF   Addr3[2] = FF
Addr3[3] = FF   Addr3[4] = FF   Addr3[5] = FF

Clk3[0]  = FF   Clk3[1]  = FF   Clk3[2]  = FF   Clk3[3]  = 03


=====================================================================================
        Fill LFSRs with initial data
=====================================================================================
```

t	clk#	LFSR1	LFSR2	LFSR3	LFSR4	X1	X2	X3	X4	Z	C[t+1]	C[t]	C[t-1]
0	0	0000000*	00000000*	000000000*	0000000000*	0	0	0	0	0	00	00	00
1	1	0000001*	00000001*	000000001*	0000000001*	0	0	0	0	0	00	00	00
2	2	0000003*	00000002*	000000003*	0000000003*	0	0	0	0	0	00	00	00
3	3	0000007*	00000004*	000000007*	0000000007*	0	0	0	0	0	00	00	00
4	4	000000F*	00000009*	00000000F*	000000000F*	0	0	0	0	0	00	00	00
5	5	000001F*	00000013*	00000001F*	000000001F*	0	0	0	0	0	00	00	00
6	6	000003F*	00000027*	00000003F*	000000003F*	0	0	0	0	0	00	00	00
7	7	000007F*	0000004F*	00000007F*	000000007F*	0	0	0	0	0	00	00	00
8	8	00000FF*	0000009F*	0000000FF*	00000000FF*	0	0	0	0	0	00	00	00
9	9	00001FF*	0000013F*	0000001FF*	00000001FF*	0	0	0	0	0	00	00	00
10	10	00003FF*	0000027F*	0000003FF*	00000003FF*	0	0	0	0	0	00	00	00
11	11	00007FF*	000004FF*	0000007FF*	00000007FF*	0	0	0	0	0	00	00	00
12	12	0000FFF*	000009FF*	000000FFF*	0000000FFF*	0	0	0	0	0	00	00	00
13	13	0001FFF*	000013FF*	000001FFF*	0000001FFF*	0	0	0	0	0	00	00	00
14	14	0003FFF*	000027FF*	000003FFF*	0000003FFF*	0	0	0	0	0	00	00	00
15	15	0007FFF*	00004FFF*	000007FFF*	0000007FFF*	0	0	0	0	0	00	00	00
16	16	000FFFF*	00009FFF*	00000FFFF*	000000FFFF*	0	0	0	0	0	00	00	00
17	17	001FFFF*	00013FFF*	00001FFFF*	000001FFFF*	0	0	0	0	0	00	00	00
18	18	003FFFF*	00027FFF*	00003FFFF*	000003FFFF*	0	0	0	0	0	00	00	00
19	19	007FFFF*	0004FFFF*	00007FFFF*	000007FFFF*	0	0	0	0	0	00	00	00
20	20	00FFFFF*	0009FFFF*	0000FFFFF*	00000FFFFF*	0	0	0	0	0	00	00	00
21	21	01FFFFF*	0013FFFF*	0001FFFFF*	00001FFFFF*	0	0	0	0	0	00	00	00
22	22	03FFFFF*	0027FFFF*	0003FFFFF*	00003FFFFF*	0	0	0	0	0	00	00	00
23	23	07FFFFF*	004FFFFF*	0007FFFFF*	00007FFFFF*	0	0	0	0	0	00	00	00
24	24	0FFFFFF*	009FFFFF*	000FFFFFF*	0000FFFFFF*	1	1	0	0	0	01	00	00
25	25	1FFFFFF*	013FFFFF*	001FFFFFF*	0001FFFFFF*	1	0	0	0	1	00	00	00
26	26	1FFFFFF	027FFFFF*	003FFFFFF*	0003FFFFFF*	1	0	0	0	1	00	00	00
27	27	1FFFFFF	04FFFFFF*	007FFFFFF*	0007FFFFFF*	1	1	0	0	0	01	00	00
28	28	1FFFFFF	09FFFFFF*	00FFFFFFF*	000FFFFFFF*	1	1	0	0	0	01	00	00
29	29	1FFFFFF	13FFFFFF*	01FFFFFFF*	001FFFFFFF*	1	1	0	0	0	01	00	00
30	30	1FFFFFF	27FFFFFF*	03FFFFFFF*	003FFFFFFF*	1	1	0	0	0	01	00	00
31	31	1FFFFFF	4FFFFFFF*	07FFFFFFF*	007FFFFFFF*	1	1	0	0	0	01	00	00
32	32	1FFFFFF	1FFFFFFF	0FFFFFFFF*	00FFFFFFFF*	1	1	1	1	0	10	00	00
33	33	1FFFFFF	3FFFFFFE	1FFFFFFFF*	01FFFFFFFF*	1	1	1	1	0	10	00	00
34	34	1FFFFFF	7FFFFFFC	1FFFFFFFF	03FFFFFFFF*	1	1	1	1	0	10	00	00

```
35   35   1FFFFFF  7FFFFFF9  1FFFFFFFF  07FFFFFFFF*   1 1 1 1   0 10   00 00
36   36   1FFFFFF  7FFFFFF3  1FFFFFFFF  0FFFFFFFFF*   1 1 1 1   0 10   00 00
37   37   1FFFFFF  7FFFFFE7  1FFFFFFFF  1FFFFFFFFF*   1 1 1 1   0 10   00 00
38   38   1FFFFFF  7FFFFFCF  1FFFFFFFF  3FFFFFFFFF*   1 1 1 1   0 10   00 00
39   39   1FFFFFF  7FFFFF9F  1FFFFFFFF  7FFFFFFFFF*   1 1 1 1   0 10   00 00
================================================================================

          Start clocking Summation Combiner

================================================================================
40    1   1FFFFFF  7FFFFF3F  1FFFFFFFF  7FFFFFFFFF    1 1 1 1   0 01   10 00
41    2   1FFFFFF  7FFFFE7F  1FFFFFFFF  7FFFFFFFFF    1 1 1 1   1 10   01 10
42    3   1FFFFFF  7FFFFCFF  1FFFFFFFF  7FFFFFFFFF    1 1 1 1   0 10   10 01
43    4   1FFFFFF  7FFFF9FF  1FFFFFFFF  7FFFFFFFFF    1 1 1 1   0 00   10 10
44    5   1FFFFFF  7FFFF3FF  1FFFFFFFF  7FFFFFFFFF    1 1 1 1   0 11   00 10
45    6   1FFFFFF  7FFFE7FE  1FFFFFFFF  7FFFFFFFFF    1 1 1 1   1 00   11 00
46    7   1FFFFFF  7FFFCFFC  1FFFFFFFF  7FFFFFFFFF    1 1 1 1   0 00   00 11
47    8   1FFFFFF  7FFF9FF9  1FFFFFFFF  7FFFFFFFFF    1 1 1 1   0 10   00 00
48    9   1FFFFFF  7FFF3FF3  1FFFFFFFF  7FFFFFFFFF    1 1 1 1   0 01   10 00
49   10   1FFFFFF  7FFE7FE6  1FFFFFFFF  7FFFFFFFFF    1 1 1 1   1 10   01 10
50   11   1FFFFFE  7FFCFFCC  1FFFFFFFE  7FFFFFFFFF    1 1 1 1   0 10   10 01
51   12   1FFFFFC  7FF9FF99  1FFFFFFFC  7FFFFFFFFF    1 1 1 1   0 00   10 10
52   13   1FFFFF8  7FF3FF33  1FFFFFFF8  7FFFFFFFFF    1 1 1 1   0 11   00 10
53   14   1FFFFF0  7FE7FE67  1FFFFFFF0  7FFFFFFFFF    1 1 1 1   1 00   11 00
54   15   1FFFFE0  7FCFFCCF  1FFFFFFE1  7FFFFFFFFF    1 1 1 1   0 00   00 11
55   16   1FFFFC0  7F9FF99F  1FFFFFFC3  7FFFFFFFFF    1 1 1 1   0 10   00 00
56   17   1FFFF80  7F3FF33E  1FFFFFF87  7FFFFFFFFE    1 0 1 1   1 00   10 00
57   18   1FFFF00  7E7FE67C  1FFFFFF0F  7FFFFFFFFC    1 0 1 1   1 00   00 10
58   19   1FFFE01  7CFFCCF8  1FFFFFE1E  7FFFFFFFF8    1 1 1 1   0 10   00 00
59   20   1FFFC03  79FF99F0  1FFFFFC3C  7FFFFFFFF0    1 1 1 1   0 01   10 00
60   21   1FFF807  73FF33E0  1FFFFF878  7FFFFFFFE1    1 1 1 1   1 10   01 10
61   22   1FFF00F  67FE67C0  1FFFFF0F0  7FFFFFFFC3    1 1 1 1   0 10   10 01
62   23   1FFE01E  4FFCCF80  1FFFFE1E1  7FFFFFFF87    1 1 1 1   0 00   10 10
63   24   1FFC03C  1FF99F00  1FFFFC3C3  7FFFFFFF0F    1 1 1 1   0 11   00 10
64   25   1FF8078  3FF33E01  1FFFF8787  7FFFFFFE1E    1 1 1 1   1 00   11 00
65   26   1FF00F0  7FE67C02  1FFFF0F0F  7FFFFFFC3C    1 1 1 1   0 00   00 11
66   27   1FE01E1  7FCCF805  1FFFE1E1E  7FFFFFF878    1 1 1 1   0 10   00 00
67   28   1FC03C3  7F99F00A  1FFFC3C3C  7FFFFFF0F0    1 1 1 1   0 01   10 00
68   29   1F80787  7F33E015  1FFF87878  7FFFFFE1E1    1 0 1 1   0 10   01 10
69   30   1F00F0F  7E67C02A  1FFF0F0F0  7FFFFFC3C3    1 0 1 1   1 11   10 01
70   31   1E01E1E  7CCF8054  1FFE1E1E1  7FFFFF8787    1 1 1 1   1 01   11 10
71   32   1C03C3C  799F00A9  1FFC3C3C3  7FFFFF0F0F    1 1 1 1   1 01   01 11
72   33   1807878  733E0152  1FF878787  7FFFFE1E1E    1 0 1 1   0 00   01 01
73   34   100F0F0  667C02A5  1FF0F0F0F  7FFFFC3C3C    0 0 1 1   0 10   00 01
74   35   001E1E0  4CF8054B  1FE1E1E1F  7FFFF87878    0 1 1 1   1 00   10 00
75   36   003C3C1  19F00A96  1FC3C3C3F  7FFFF0F0F0    0 1 1 1   1 00   00 10
76   37   0078783  33E0152C  1F878787F  7FFFE1E1E1    0 1 1 1   1 01   00 00
77   38   00F0F07  67C02A59  1F0F0F0FF  7FFFC3C3C3    0 1 1 1   0 11   01 00
78   39   01E1E0E  4F8054B3  1E1E1E1FF  7FFF878787    0 1 1 1   0 11   11 01
79   40   03C3C1C  1F00A966  1C3C3C3FF  7FFF0F0F0F    0 0 1 1   1 11   11 11
80   41   0787838  3E0152CC  1878787FF  7FFE1E1E1E    0 0 1 1   1 11   11 11
81   42   0F0F070  7C02A598  10F0F0FFF  7FFC3C3C3C    1 0 0 1   1 11   11 11
82   43   1E1E0E0  78054B30  01E1E1FFF  7FF8787878    1 0 0 1   1 11   11 11
83   44   1C3C1C0  700A9660  03C3C3FFE  7FF0F0F0F0    1 0 0 1   1 11   11 11
84   45   1878380  60152CC0  078787FFC  7FE1E1E1E0    1 0 0 1   1 11   11 11
85   46   10F0700  402A5980  0F0F0FFF8  7FC3C3C3C0    0 0 1 1   1 11   11 11
86   47   01E0E00  0054B300  1E1E1FFF0  7F87878780    0 0 1 1   1 11   11 11
87   48   03C1C00  00A96601  1C3C3FFE0  7F0F0F0F00    0 1 1 0   1 11   11 11
88   49   0783800  0152CC03  18787FFC0  7E1E1E1E01    0 0 1 0   0 11   11 11
```

89	50	0F07000	02A59806	10F0FFF80	7C3C3C3C03	1	1	0	0	1	11	11	11
90	51	1E0E000	054B300D	01E1FFF00	7878787807	1	0	0	0	0	11	11	11
91	52	1C1C001	0A96601A	03C3FFE01	70F0F0F00F	1	1	0	1	0	10	11	11
92	53	1838003	152CC035	0787FFC03	61E1E1E01E	1	0	0	1	0	10	10	11
93	54	1070007	2A59806B	0F0FFF807	43C3C3C03C	0	0	1	1	0	01	10	10
94	55	00E000F	54B300D7	1E1FFF00F	0787878078	0	1	1	1	0	10	01	10
95	56	01C001F	296601AE	1C3FFE01F	0F0F0F00F1	0	0	1	0	1	00	10	01
96	57	038003F	52CC035C	187FFC03F	1E1E1E01E2	0	1	1	0	0	00	00	10
97	58	070007F	259806B8	10FFF807F	3C3C3C03C4	0	1	0	0	1	00	00	00
98	59	0E000FE	4B300D71	01FFF00FE	7878780788	1	0	0	0	1	00	00	00
99	60	1C001FD	16601AE2	03FFE01FD	70F0F00F10	1	0	0	1	0	01	00	00
100	61	18003FA	2CC035C5	07FFC03FB	61E1E01E21	1	1	0	1	0	11	01	00
101	62	10007F4	59806B8B	0FFF807F7	43C3C03C43	0	1	1	1	0	11	11	01
102	63	0000FE8	3300D717	1FFF00FEE	0787807887	0	0	1	1	1	11	11	11
103	64	0001FD0	6601AE2F	1FFE01FDC	0F0F00F10E	0	0	1	0	0	11	11	11
104	65	0003FA0	4C035C5F	1FFC03FB8	1E1E01E21D	0	0	1	0	0	11	11	11
105	66	0007F40	1806B8BE	1FF807F70	3C3C03C43B	0	0	1	0	0	11	11	11
106	67	000FE81	300D717C	1FF00FEE1	7878078877	0	0	1	0	0	11	11	11
107	68	001FD02	601AE2F8	1FE01FDC2	70F00F10EF	0	0	1	1	1	11	11	11
108	69	003FA05	4035C5F0	1FC03FB84	61E01E21DE	0	0	1	1	1	11	11	11
109	70	007F40B	006B8BE0	1F807F708	43C03C43BC	0	0	1	1	1	11	11	11
110	71	00FE816	00D717C0	1F00FEE11	0780788778	0	1	1	1	0	10	11	11
111	72	01FD02C	01AE2F81	1E01FDC23	0F00F10EF1	0	1	1	0	0	10	10	11
112	73	03FA059	035C5F02	1C03FB847	1E01E21DE3	0	0	1	0	1	10	10	10
113	74	07F40B3	06B8BE05	1807F708F	3C03C43BC7	0	1	1	0	0	01	10	10
114	75	0FE8166	0D717C0B	100FEE11E	780788778F	1	0	0	0	0	01	01	10
115	76	1FD02CD	1AE2F817	001FDC23D	700F10EF1F	1	1	0	0	1	11	01	01
116	77	1FA059B	35C5F02F	003FB847A	601E21DE3F	1	1	0	0	1	10	11	01
117	78	1F40B37	6B8BE05E	007F708F4	403C43BC7F	1	1	0	0	0	10	10	11
118	79	1E8166E	5717C0BD	00FEE11E9	00788778FF	1	0	0	0	1	10	10	10
119	80	1D02CDC	2E2F817A	01FDC23D3	00F10EF1FE	1	0	0	1	0	01	10	10
120	81	1A059B9	5C5F02F5	03FB847A6	01E21DE3FD	1	0	0	1	1	01	01	10
121	82	140B373	38BE05EB	07F708F4C	03C43BC7FB	0	1	0	1	1	11	01	01
122	83	08166E7	717C0BD7	0FEE11E98	0788778FF7	1	0	1	1	0	11	11	01
123	84	102CDCF	62F817AE	1FDC23D31	0F10EF1FEF	0	1	1	0	1	11	11	11
124	85	0059B9F	45F02F5C	1FB847A63	1E21DE3FDE	0	1	1	0	1	11	11	11
125	86	00B373E	0BE05EB9	1F708F4C7	3C43BC7FBC	0	1	1	0	1	11	11	11
126	87	0166E7D	17C0BD72	1EE11E98F	788778FF78	0	1	1	1	0	10	11	11
127	88	02CDCFB	2F817AE5	1DC23D31F	710EF1FEF1	0	1	1	0	0	10	10	11
128	89	059B9F7	5F02F5CA	1B847A63F	621DE3FDE2	0	0	1	0	1	10	10	10
129	90	0B373EF	3E05EB94	1708F4C7F	443BC7FBC4	1	0	0	0	1	10	10	10
130	91	166E7DF	7C0BD728	0E11E98FF	08778FF788	0	0	1	0	1	10	10	10
131	92	0CDCFBE	7817AE50	1C23D31FF	10EF1FEF10	1	0	1	1	1	01	10	10
132	93	19B9F7D	702F5CA1	1847A63FE	21DE3FDE21	1	0	1	1	0	10	01	10
133	94	1373EFB	605EB942	108F4C7FC	43BC7FBC43	0	0	0	1	1	00	10	01
134	95	06E7DF7	40BD7285	011E98FF8	0778FF7886	0	1	0	0	1	01	00	10
135	96	0DCFBEF	017AE50A	023D31FF0	0EF1FEF10D	1	0	0	1	1	00	01	00
136	97	1B9F7DF	02F5CA15	047A63FE1	1DE3FDE21A	1	1	0	1	1	10	00	01
137	98	173EFBF	05EB942B	08F4C7FC3	3BC7FBC434	0	1	1	1	1	00	10	00
138	99	0E7DF7F	0BD72856	11E98FF87	778FF78869	1	1	0	1	1	00	00	10
139	100	1CFBEFF	17AE50AC	03D31FF0F	6F1FEF10D3	1	1	0	0	0	01	00	00
140	101	19F7DFE	2F5CA159	07A63FE1E	5E3FDE21A7	1	0	0	0	0	00	01	00
141	102	13EFBFC	5EB942B3	0F4C7FC3C	3C7FBC434F	0	1	1	0	0	10	00	01
142	103	07DF7F8	3D728566	1E98FF878	78FF78869F	0	0	1	1	0	00	10	00
143	104	0FBEFF0	7AE50ACD	1D31FF0F0	71FEF10D3E	1	1	1	1	0	11	00	10
144	105	1F7DFE1	75CA159B	1A63FE1E1	63FDE21A7D	1	1	1	1	1	00	11	00
145	106	1EFBFC3	6B942B36	14C7FC3C3	47FBC434FB	1	1	0	1	1	11	00	11

146	107	1DF7F86	5728566D	098FF8786	0FF78869F7	1	0	1	1	0	00	11	00
147	108	1BEFF0C	2E50ACDB	131FF0F0C	1FEF10D3EF	1	0	0	1	0	11	00	11
148	109	17DFE19	5CA159B6	063FE1E19	3FDE21A7DF	0	1	0	1	1	01	11	00
149	110	0FBFC33	3942B36D	0C7FC3C32	7FBC434FBF	1	0	1	1	0	01	01	11
150	111	1F7F866	728566DB	18FF87865	7F78869F7E	1	1	1	0	0	00	01	01
151	112	1EFF0CC	650ACDB6	11FF0F0CB	7EF10D3EFC	1	0	0	1	0	10	00	01
152	113	1DFE199	4A159B6D	03FE1E196	7DE21A7DF9	1	0	0	1	0	00	10	00
153	114	1BFC333	142B36DB	07FC3C32C	7BC434FBF3	1	0	0	1	0	00	00	10
154	115	17F8666	28566DB6	0FF878659	778869F7E6	0	0	1	1	0	01	00	00
155	116	0FF0CCC	50ACDB6D	1FF0F0CB3	6F10D3EFCC	1	1	1	0	0	11	01	00
156	117	1FE1999	2159B6DA	1FE1E1966	5E21A7DF99	1	0	1	0	1	10	11	01
157	118	1FC3332	42B36DB5	1FC3C32CC	3C434FBF33	1	1	1	0	1	10	10	11
158	119	1F86664	0566DB6B	1F8786599	78869F7E67	1	0	1	1	1	01	10	10
159	120	1F0CCC8	0ACDB6D6	1F0F0CB33	710D3EFCCE	1	1	1	0	0	10	01	10
160	121	1E19991	159B6DAC	1E1E19666	621A7DF99D	1	1	1	0	1	11	10	01
161	122	1C33323	2B36DB58	1C3C32CCC	4434FBF33B	1	0	1	0	1	00	11	10
162	123	1866647	566DB6B0	187865999	0869F7E676	1	0	1	0	0	11	00	11
163	124	10CCC8F	2CDB6D60	10F0CB333	10D3EFCCEC	0	1	0	1	1	01	11	00
164	125	019991E	59B6DAC0	01E196666	21A7DF99D9	0	1	0	1	1	10	01	11
165	126	033323C	336DB580	03C32CCCD	434FBF33B3	0	0	0	0	0	00	10	01
166	127	0666478	66DB6B01	07865999A	069F7E6766	0	1	0	1	0	00	00	10
167	128	0CCC8F0	4DB6D603	0F0CB3334	0D3EFCCECD	1	1	1	0	1	01	00	00
168	129	19991E1	1B6DAC07	1E1966669	1A7DF99D9B	1	0	1	0	1	00	01	00
169	130	13323C3	36DB580E	1C32CCCD3	34FBF33B37	0	1	1	1	1	10	00	01
170	131	0664786	6DB6B01C	1865999A7	69F7E6766F	0	1	1	1	1	00	10	00
171	132	0CC8F0D	5B6D6039	10CB3334F	53EFCCECDF	1	0	0	1	0	00	00	10
172	133	1991E1A	36DAC073	01966669E	27DF99D9BF	1	1	0	1	1	01	00	00
173	134	1323C35	6DB580E6	032CCCD3C	4FBF33B37E	0	1	0	1	1	00	01	00
174	135	064786A	5B6B01CD	065999A78	1F7E6766FC	0	0	0	0	0	11	00	01
175	136	0C8F0D5	36D6039B	0CB3334F0	3EFCCECDF9	1	1	1	1	1	00	11	00
176	137	191E1AA	6DAC0737	1966669E1	7DF99D9BF3	1	1	1	1	0	00	00	11
177	138	123C354	5B580E6E	12CCCD3C3	7BF33B37E7	0	0	0	1	1	00	00	00
178	139	04786A9	36B01CDC	05999A787	77E6766FCE	0	1	0	1	0	01	00	00
179	140	08F0D53	6D6039B8	0B3334F0E	6FCCECDF9C	1	0	1	1	0	11	01	00
180	141	11E1AA6	5AC07370	166669E1D	5F99D9BF38	0	1	0	1	1	10	11	01
181	142	03C354C	3580E6E0	0CCCD3C3A	3F33B37E70	0	1	1	0	0	10	10	11
182	143	0786A99	6B01CDC0	1999A7875	7E6766FCE1	0	0	1	0	1	10	10	10
183	144	0F0D533	56039B81	13334F0EB	7CCECDF9C2	1	0	0	1	0	01	10	10
184	145	1E1AA66	2C073703	06669E1D6	799D9BF385	1	0	0	1	1	01	01	10
185	146	1C354CC	580E6E06	0CCD3C3AC	733B37E70B	1	0	1	0	1	11	01	01
186	147	186A998	301CDC0C	199A78759	66766FCE17	1	0	1	0	1	10	11	01
187	148	10D5331	6039B818	1334F0EB2	4CECDF9C2F	0	0	0	1	1	01	10	11
188	149	01AA662	40737031	0669E1D65	19D9BF385E	0	0	0	1	0	01	01	10
189	150	0354CC5	00E6E063	0CD3C3ACB	33B37E70BD	0	1	1	1	0	00	01	01
190	151	06A998A	01CDC0C6	19A787596	6766FCE17B	0	1	1	0	0	10	00	01
191	152	0D53315	039B818C	134F0EB2C	4ECDF9C2F6	1	1	0	1	1	00	10	00
192	153	1AA662A	07370318	069E1D659	1D9BF385ED	1	0	0	1	0	00	00	10
193	154	154CC54	0E6E0630	0D3C3ACB3	3B37E70BDB	0	0	1	0	1	00	00	00
194	155	0A998A8	1CDC0C60	1A7875967	766FCE17B6	1	1	1	0	1	01	00	00
195	156	1533151	39B818C0	14F0EB2CE	6CDF9C2F6C	0	1	0	1	1	00	01	00
196	157	0A662A3	73703180	09E1D659D	59BF385ED8	1	0	1	1	1	10	00	01
197	158	14CC547	66E06301	13C3ACB3A	337E70BDB0	0	1	0	0	1	11	10	00
198	159	0998A8E	4DC0C602	078759675	66FCE17B61	1	1	0	1	0	01	11	10
199	160	133151D	1B818C05	0F0EB2CEB	4DF9C2F6C2	0	1	1	1	0	01	01	11
200	161	0662A3B	3703180B	1E1D659D6	1BF385ED85	0	0	1	1	1	11	01	01
201	162	0CC5477	6E063017	1C3ACB3AC	37E70BDB0B	1	0	1	1	0	11	11	01
202	163	198A8EF	5C0C602F	187596759	6FCE17B617	1	0	1	1	0	10	11	11

203	164	13151DE	3818C05F	10EB2CEB2	5F9C2F6C2F	0	0	0	1	1	01	10	11
204	165	062A3BC	703180BF	01D659D65	3F385ED85E	0	0	0	0	1	00	01	10
205	166	0C54779	6063017E	03ACB3ACB	7E70BDB0BD	1	0	0	0	1	11	00	01
206	167	18A8EF2	40C602FD	075967597	7CE17B617B	1	1	0	1	0	00	11	00
207	168	1151DE4	018C05FA	0EB2CEB2F	79C2F6C2F7	0	1	1	1	1	11	00	11
208	169	02A3BC9	03180BF5	1D659D65E	7385ED85EE	0	0	1	1	1	01	11	00
209	170	0547793	063017EB	1ACB3ACBC	670BDB0BDC	0	0	1	0	0	10	01	11
210	171	0A8EF27	0C602FD6	159675978	4E17B617B9	1	0	0	0	1	00	10	01
211	172	151DE4E	18C05FAD	0B2CEB2F1	1C2F6C2F73	0	1	1	0	0	00	00	10
212	173	0A3BC9C	3180BF5A	1659D65E3	385ED85EE6	1	1	0	0	0	01	00	00
213	174	1477938	63017EB5	0CB3ACBC6	70BDB0BDCC	0	0	1	1	1	00	01	00
214	175	08EF270	4602FD6A	19675978D	617B617B99	1	0	1	0	0	10	00	01
215	176	11DE4E1	0C05FAD5	12CEB2F1A	42F6C2F733	0	0	0	1	1	11	10	00
216	177	03BC9C3	180BF5AA	059D65E34	05ED85EE67	0	0	0	1	0	00	11	10
217	178	0779387	3017EB55	0B3ACBC68	0BDB0BDCCF	0	0	1	1	0	11	00	11
218	179	0EF270F	602FD6AA	1675978D0	17B617B99F	1	0	0	1	1	01	11	00
219	180	1DE4E1F	405FAD54	0CEB2F1A1	2F6C2F733F	1	0	1	0	1	10	01	11
220	181	1BC9C3F	00BF5AA9	19D65E342	5ED85EE67F	1	1	1	1	0	10	10	01
221	182	179387F	017EB552	13ACBC684	3DB0BDCCFE	0	0	0	1	1	10	10	10
222	183	0F270FF	02FD6AA5	075978D09	7B617B99FC	1	1	0	0	0	01	10	10
223	184	1E4E1FF	05FAD54A	0EB2F1A12	76C2F733F9	1	1	1	1	1	10	01	10
224	185	1C9C3FE	0BF5AA94	1D65E3425	6D85EE67F2	1	1	1	1	0	10	10	01
225	186	19387FD	17EB5529	1ACBC684B	5B0BDCCFE4	1	1	1	0	1	01	10	10
226	187	1270FFA	2FD6AA53	15978D096	3617B99FC9	0	1	0	0	0	01	01	10
227	188	04E1FF5	5FAD54A7	0B2F1A12C	6C2F733F93	0	1	1	0	1	11	01	01
228	189	09C3FEB	3F5AA94E	165E34258	585EE67F27	1	0	0	0	0	10	11	01
229	190	1387FD7	7EB5529C	0CBC684B1	30BDCCFE4F	0	1	1	1	1	10	10	11
230	191	070FFAE	7D6AA538	1978D0962	617B99FC9E	0	0	1	0	1	10	10	10
231	192	0E1FF5C	7AD54A70	12F1A12C4	42F733F93D	1	1	0	1	1	01	10	10
232	193	1C3FEB9	75AA94E1	05E342588	05EE67F27A	1	1	0	1	0	10	01	10
233	194	187FD73	6B5529C3	0BC684B10	0BDCCFE4F4	1	0	1	1	1	11	10	01
234	195	10FFAE6	56AA5386	178D09621	17B99FC9E8	0	1	0	1	1	00	11	10
235	196	01FF5CC	2D54A70C	0F1A12C43	2F733F93D0	0	0	1	0	1	10	00	11
236	197	03FEB98	5AA94E19	1E3425887	5EE67F27A1	0	1	1	1	1	00	10	00
237	198	07FD731	35529C33	1C684B10F	3DCCFE4F42	0	0	1	1	0	00	00	10
238	199	0FFAE63	6AA53866	18D09621F	7B99FC9E84	1	1	1	1	0	10	00	00
239	200	1FF5CC6	554A70CD	11A12C43F	7733F93D09	1	0	0	0	1	11	10	00

Z[0] = 59
Z[1] = 3B
Z[2] = EF
Z[3] = 07
Z[4] = 13
Z[5] = 70
Z[6] = 9B
Z[7] = B7
Z[8] = 52
Z[9] = 8F
Z[10] = 3E
Z[11] = B9
Z[12] = A5
Z[13] = AC
Z[14] = EA
Z[15] = 9E

```
==============================================================================================
        Reload this pattern into the LFSRs
        Hold content of Summation Combiner regs and calculate new C[t+1] and Z values
==============================================================================================
LFSR1  <= 1521359
LFSR2  <= 528F703B
LFSR3  <= 0AC3E9BEF
LFSR4  <= 4FEAB9B707
C[t+1] <= 00

==============================================================================================
        Generating 125 key symbols (encryption/decryption sequence)
==============================================================================================
```

240	1	1521359	528F703B	0AC3E9BEF	4FEAB9B707	0	1	1	1	1	00	10	00
241	2	0A426B3	251EE076	1587D37DE	1FD5736E0F	1	0	0	1	0	00	00	10
242	3	1484D67	4A3DC0ED	0B0FA6FBD	3FAAE6DC1E	0	0	1	1	0	01	00	00
243	4	0909ACF	147B81DA	161F4DF7A	7F55CDB83D	1	0	0	0	0	00	01	00
244	5	121359E	28F703B5	0C3E9BEF5	7EAB9B707B	0	1	1	1	1	10	00	01
245	6	0426B3C	51EE076B	187D37DEB	7D5736E0F6	0	1	1	0	0	00	10	00
246	7	084D679	23DC0ED6	10FA6FBD7	7AAE6DC1EC	1	1	0	1	1	00	00	10
247	8	109ACF2	47B81DAC	01F4DF7AF	755CDB83D8	0	1	0	0	1	00	00	00
248	9	01359E4	0F703B59	03E9BEF5E	6AB9B707B1	0	0	0	1	1	00	00	00
249	10	026B3C8	1EE076B3	07D37DEBD	55736E0F63	0	1	0	0	1	00	00	00
250	11	04D6791	3DC0ED67	0FA6FBD7A	2AE6DC1EC7	0	1	1	1	1	01	00	00
251	12	09ACF22	7B81DACF	1F4DF7AF4	55CDB83D8F	1	1	1	1	1	11	01	00
252	13	1359E44	7703B59E	1E9BEF5E8	2B9B707B1F	0	0	1	1	1	10	11	01
253	14	06B3C88	6E076B3C	1D37DEBD0	5736E0F63F	0	0	1	0	1	01	10	11
254	15	0D67911	5C0ED678	1A6FBD7A1	2E6DC1EC7E	1	0	1	0	1	01	01	10
255	16	1ACF223	381DACF0	14DF7AF42	5CDB83D8FD	1	0	0	1	1	11	01	01
256	17	159E446	703B59E0	09BEF5E85	39B707B1FA	0	0	1	1	1	10	11	01
257	18	0B3C88C	6076B3C0	137DEBD0A	736E0F63F4	1	0	0	0	1	01	10	11
258	19	1679118	40ED6780	06FBD7A15	66DC1EC7E8	0	1	0	1	1	01	01	10
259	20	0CF2231	01DACF00	0DF7AF42A	4DB83D8FD1	1	1	1	1	1	00	01	01
260	21	19E4463	03B59E01	1BEF5E854	1B707B1FA3	1	1	1	0	1	10	00	01
261	22	13C88C6	076B3C03	17DEBD0A9	36E0F63F47	0	0	0	1	1	11	10	00
262	23	079118C	0ED67807	0FBD7A152	6DC1EC7E8E	0	1	1	1	0	01	11	10
263	24	0F22318	1DACF00E	1F7AF42A4	5B83D8FD1D	1	1	1	1	1	01	01	11
264	25	1E44630	3B59E01C	1EF5E8548	3707B1FA3B	1	0	1	0	1	11	01	01
265	26	1C88C61	76B3C039	1DEBD0A91	6E0F63F477	1	1	1	0	0	11	11	01
266	27	19118C3	6D678073	1BD7A1523	5C1EC7E8EF	1	0	1	0	1	11	11	11
267	28	1223187	5ACF00E6	17AF42A46	383D8FD1DE	0	1	0	0	0	11	11	11
268	29	044630E	359E01CC	0F5E8548D	707B1FA3BD	0	1	1	0	1	11	11	11
269	30	088C61C	6B3C0399	1EBD0A91A	60F63F477B	1	0	1	1	0	10	11	11
270	31	1118C39	56780733	1D7A15234	41EC7E8EF6	0	0	1	1	0	10	10	11
271	32	0231872	2CF00E67	1AF42A468	03D8FD1DEC	0	1	1	1	1	01	10	10
272	33	04630E5	59E01CCE	15E8548D1	07B1FA3BD8	0	1	0	1	1	01	01	10
273	34	08C61CB	33C0399D	0BD0A91A3	0F63F477B1	1	1	1	0	0	00	01	01
274	35	118C396	6780733A	17A152347	1EC7E8EF63	0	1	0	1	0	10	00	01
275	36	031872D	4F00E674	0F42A468E	3D8FD1DEC7	0	0	1	1	0	00	10	00
276	37	0630E5A	1E01CCE8	1E8548D1D	7B1FA3BD8E	0	0	1	0	1	01	00	10
277	38	0C61CB5	3C0399D0	1D0A91A3B	763F477B1C	1	0	1	0	1	00	01	00
278	39	18C396A	780733A0	1A1523477	6C7E8EF639	1	0	1	0	0	10	00	01
279	40	11872D5	700E6741	142A468EF	58FD1DEC72	0	0	0	1	1	11	10	00
280	41	030E5AB	601CCE83	08548D1DF	31FA3BD8E5	0	0	1	1	1	00	11	10
281	42	061CB57	40399D07	10A91A3BF	63F477B1CB	0	0	0	1	1	10	00	11
282	43	0C396AF	00733A0F	01523477E	47E8EF6396	1	0	0	1	0	00	10	00
283	44	1872D5F	00E6741F	02A468EFD	0FD1DEC72C	1	1	0	1	1	00	00	10

284	45	10E5ABE	01CCE83F	0548D1DFA	1FA3BD8E58	0	1	0	1	0	01	00	00
285	46	01CB57C	0399D07F	0A91A3BF4	3F477B1CB0	0	1	1	0	1	00	01	00
286	47	0396AF9	0733A0FE	1523477E9	7E8EF63961	0	0	0	1	1	11	00	01
287	48	072D5F3	0E6741FD	0A468EFD2	7D1DEC72C3	0	0	1	0	0	01	11	00
288	49	0E5ABE7	1CCE83FA	148D1DFA4	7A3BD8E587	1	1	0	0	1	10	01	11
289	50	1CB57CE	399D07F4	091A3BF49	7477B1CB0F	1	1	1	0	1	11	10	01
290	51	196AF9D	733A0FE9	123477E92	68EF63961E	1	0	0	1	1	00	11	10
291	52	12D5F3B	66741FD2	0468EFD25	51DEC72C3C	0	0	0	1	1	10	00	11
292	53	05ABE77	4CE83FA4	08D1DFA4B	23BD8E5879	0	1	1	1	1	00	10	00
293	54	0B57CEE	19D07F49	11A3BF496	477B1CB0F2	1	1	0	0	0	00	00	10
294	55	16AF9DC	33A0FE92	03477E92C	0EF63961E4	0	1	0	1	0	01	00	00
295	56	0D5F3B8	6741FD25	068EFD259	1DEC72C3C9	1	0	0	1	1	00	01	00
296	57	1ABE771	4E83FA4B	0D1DFA4B3	3BD8E58793	1	1	1	1	0	01	00	01
297	58	157CEE2	1D07F496	1A3BF4967	77B1CB0F26	0	0	1	1	1	00	01	00
298	59	0AF9DC5	3A0FE92D	1477E92CE	6F63961E4D	1	0	0	0	1	11	00	01
299	60	15F3B8B	741FD25A	08EFD259C	5EC72C3C9B	0	0	1	1	1	01	11	00
300	61	0BE7716	683FA4B4	11DFA4B39	3D8E587937	1	0	0	1	1	10	01	11
301	62	17CEE2D	507F4968	03BF49672	7B1CB0F26E	0	0	0	0	0	00	10	01
302	63	0F9DC5B	20FE92D0	077E92CE4	763961E4DC	1	1	0	0	0	00	00	10
303	64	1F3B8B6	41FD25A0	0EFD259C9	6C72C3C9B9	1	1	1	0	1	01	00	00
304	65	1E7716D	03FA4B40	1DFA4B393	58E5879373	1	1	1	1	1	11	01	00
305	66	1CEE2DB	07F49680	1BF496727	31CB0F26E6	1	1	1	1	1	11	11	01
306	67	19DC5B7	0FE92D00	17E92CE4E	63961E4DCD	1	1	0	1	0	10	11	11
307	68	13B8B6F	1FD25A00	0FD259C9C	472C3C9B9A	0	1	1	0	0	10	10	11
308	69	07716DF	3FA4B400	1FA4B3938	0E58793735	0	1	1	0	0	01	10	10
309	70	0EE2DBF	7F496800	1F4967271	1CB0F26E6A	1	0	1	1	0	10	01	10
310	71	1DC5B7F	7E92D000	1E92CE4E2	3961E4DCD4	1	1	1	0	1	11	10	01
311	72	1B8B6FF	7D25A001	1D259C9C4	72C3C9B9A9	1	0	1	1	0	01	11	10
312	73	1716DFF	7A4B4002	1A4B39389	6587937352	0	0	1	1	1	10	01	11
313	74	0E2DBFF	74968005	149672713	4B0F26E6A5	1	1	0	0	0	11	10	01
314	75	1C5B7FE	692D000B	092CE4E26	161E4DCD4B	1	0	1	0	1	00	11	10
315	76	18B6FFC	525A0017	1259C9C4D	2C3C9B9A96	1	0	0	0	1	10	00	11
316	77	116DFF8	24B4002F	04B39389B	587937352C	0	1	0	0	1	11	10	00
317	78	02DBFF1	4968005F	096727136	30F26E6A58	0	0	1	1	1	00	11	10
318	79	05B7FE3	12D000BF	12CE4E26C	61E4DCD4B1	0	1	0	1	0	11	00	11
319	80	0B6FFC7	25A0017F	059C9C4D8	43C9B9A963	1	1	0	1	0	00	11	00
320	81	16DFF8E	4B4002FF	0B39389B1	07937352C6	0	0	1	1	0	11	00	11
321	82	0DBFF1C	168005FF	167271363	0F26E6A58C	1	1	0	0	1	01	11	00
322	83	1B7FE38	2D000BFF	0CE4E26C7	1E4DCD4B18	1	0	1	0	1	10	01	11
323	84	16FFC70	5A0017FF	19C9C4D8F	3C9B9A9631	0	0	1	1	0	11	10	01
324	85	0DFF8E1	34002FFF	139389B1E	7937352C62	1	0	0	0	0	00	11	10
325	86	1BFF1C3	68005FFF	07271363D	726E6A58C4	1	0	0	0	1	10	00	11
326	87	17FE387	5000BFFE	0E4E26C7B	64DCD4B188	0	0	1	1	0	00	10	00
327	88	0FFC70F	20017FFD	1C9C4D8F6	49B9A96311	1	0	1	1	1	00	00	10
328	89	1FF8E1F	4002FFFB	19389B1ED	137352C623	1	0	1	0	0	01	00	00
329	90	1FF1C3F	0005FFF7	1271363DB	26E6A58C46	1	0	0	1	1	00	01	00
330	91	1FE387F	000BFFEE	04E26C7B6	4DCD4B188C	1	0	0	1	0	10	00	01
331	92	1FC70FF	0017FFDC	09C4D8F6D	1B9A963118	1	0	1	1	1	00	10	00
332	93	1F8E1FF	002FFFB8	1389B1EDA	37352C6231	1	0	0	0	1	01	00	10
333	94	1F1C3FF	005FFF70	071363DB4	6E6A58C462	1	0	0	0	0	00	01	00
334	95	1E387FE	00BFFEE0	0E26C7B68	5CD4B188C5	1	1	1	1	0	01	00	01
335	96	1C70FFC	017FFDC1	1C4D8F6D1	39A963118A	1	0	1	1	0	11	01	00
336	97	18E1FF9	02FFFB82	189B1EDA2	7352C62315	1	1	1	0	0	11	11	01
337	98	11C3FF2	05FFF705	11363DB45	66A58C462B	0	1	0	1	1	11	11	11
338	99	0387FE4	0BFFEE0A	026C7B68B	4D4B188C56	0	1	0	0	0	11	11	11
339	100	070FFC9	17FFDC15	04D8F6D16	1A963118AD	0	1	0	1	1	11	11	11
340	101	0E1FF92	2FFFB82B	09B1EDA2C	352C62315A	1	1	1	0	0	10	11	11

341	102	1C3FF24	5FFF7057	1363DB458	6A58C462B4	1	1	0	0	0	10	10	11
342	103	187FE48	3FFEE0AE	06C7B68B0	54B188C569	1	1	0	1	1	01	10	10
343	104	10FFC90	7FFDC15C	0D8F6D161	2963118AD2	0	1	1	0	1	01	01	10
344	105	01FF920	7FFB82B9	1B1EDA2C2	52C62315A5	0	1	1	1	0	00	01	01
345	106	03FF240	7FF70573	163DB4584	258C462B4B	0	1	0	1	0	10	00	01
346	107	07FE481	7FEE0AE6	0C7B68B08	4B188C5696	0	1	1	0	0	00	10	00
347	108	0FFC902	7FDC15CD	18F6D1610	163118AD2D	1	1	1	0	1	00	00	10
348	109	1FF9204	7FB82B9A	11EDA2C20	2C62315A5B	1	1	0	0	0	01	00	00
349	110	1FF2408	7F705735	03DB45841	58C462B4B6	1	0	0	1	1	00	01	00
350	111	1FE4810	7EE0AE6B	07B68B082	3188C5696C	1	1	0	1	1	10	00	01
351	112	1FC9021	7DC15CD6	0F6D16105	63118AD2D8	1	1	1	0	1	00	10	00
352	113	1F92042	7B82B9AD	1EDA2C20B	462315A5B0	1	1	1	0	1	00	00	10
353	114	1F24084	7705735A	1DB458416	0C462B4B61	1	0	1	0	0	01	00	00
354	115	1E48108	6E0AE6B5	1B68B082C	188C5696C3	1	0	1	1	0	11	01	00
355	116	1C90211	5C15CD6A	16D161059	3118AD2D86	1	0	0	0	0	10	11	01
356	117	1920422	382B9AD5	0DA2C20B3	62315A5B0D	1	0	1	0	0	10	10	11
357	118	1240845	705735AA	1B4584167	4462B4B61A	0	0	1	0	1	10	10	10
358	119	048108A	60AE6B55	168B082CF	08C5696C34	0	1	0	1	0	01	10	10
359	120	0902114	415CD6AB	0D161059E	118AD2D869	1	0	1	1	0	10	01	10
360	121	1204228	02B9AD56	1A2C20B3D	2315A5B0D2	0	1	1	0	0	11	10	01
361	122	0408451	05735AAD	14584167B	462B4B61A4	0	0	0	0	1	11	11	10
362	123	08108A2	0AE6B55B	08B082CF7	0C5696C348	1	1	1	0	0	10	11	11
363	124	1021144	15CD6AB6	1161059EF	18AD2D8690	0	1	0	1	0	10	10	11
364	125	0042289	2B9AD56C	02C20B3DE	315A5B0D20	0	1	0	0	1	10	10	10

1.5 FOURTH SET OF SAMPLES

```
Initial values for the key, pan address and clock

K'c4[0]  = 21   K'c4[1]  = 87   K'c4[2]  = F0   K'c4[3]  = 4A
K'c4[4]  = BA   K'c4[5]  = 90   K'c4[6]  = 31   K'c4[7]  = D0
K'c4[8]  = 78   K'c4[9]  = 0D   K'c4[10] = 4C   K'c4[11] = 53
K'c4[12] = E0   K'c4[13] = 15   K'c4[14] = 3A   K'c4[15] = 63

Addr4[0] = 2C   Addr4[1] = 7F   Addr4[2] = 94
Addr4[3] = 56   Addr4[4] = 0F   Addr4[5] = 1B

Clk4[0]  = 5F   Clk4[1]  = 1A   Clk4[2]  = 00   Clk4[3]  = 02

=====================================================================================
        Fill LFSRs with initial data
=====================================================================================
```

t	clk#	LFSR1	LFSR2	LFSR3	LFSR4	X1	X2	X3	X4	Z	C[t+1]	C[t]	C[t-1]
0	0	0000000*	00000000*	000000000*	0000000000*	0	0	0	0	0	00	00	00
1	1	0000000*	00000001*	000000001*	0000000001*	0	0	0	0	0	00	00	00
2	2	0000001*	00000002*	000000002*	0000000003*	0	0	0	0	0	00	00	00
3	3	0000002*	00000004*	000000004*	0000000007*	0	0	0	0	0	00	00	00
4	4	0000004*	00000009*	000000008*	000000000F*	0	0	0	0	0	00	00	00
5	5	0000008*	00000013*	000000010*	000000001E*	0	0	0	0	0	00	00	00
6	6	0000010*	00000027*	000000021*	000000003D*	0	0	0	0	0	00	00	00
7	7	0000021*	0000004F*	000000043*	000000007A*	0	0	0	0	0	00	00	00
8	8	0000042*	0000009F*	000000087*	00000000F4*	0	0	0	0	0	00	00	00
9	9	0000084*	0000013F*	00000010F*	00000001E9*	0	0	0	0	0	00	00	00
10	10	0000108*	0000027F*	00000021F*	00000003D2*	0	0	0	0	0	00	00	00
11	11	0000211*	000004FE*	00000043E*	00000007A5*	0	0	0	0	0	00	00	00
12	12	0000422*	000009FC*	00000087C*	0000000F4A*	0	0	0	0	0	00	00	00
13	13	0000845*	000013F8*	0000010F8*	0000001E94*	0	0	0	0	0	00	00	00
14	14	000108B*	000027F0*	0000021F1*	0000003D29*	0	0	0	0	0	00	00	00
15	15	0002117*	00004FE1*	0000043E3*	0000007A52*	0	0	0	0	0	00	00	00
16	16	000422E*	00009FC2*	0000087C6*	000000F4A4*	0	0	0	0	0	00	00	00
17	17	000845D*	00013F84*	000010F8C*	000001E948*	0	0	0	0	0	00	00	00
18	18	00108BA*	00027F08*	000021F18*	000003D290*	0	0	0	0	0	00	00	00
19	19	0021174*	0004FE10*	000043E30*	000007A520*	0	0	0	0	0	00	00	00
20	20	00422E8*	0009FC21*	000087C61*	00000F4A41*	0	0	0	0	0	00	00	00
21	21	00845D1*	0013F842*	00010F8C3*	00001E9482*	0	0	0	0	0	00	00	00
22	22	0108BA3*	0027F084*	00021F186*	00003D2905*	0	0	0	0	0	00	00	00
23	23	0211747*	004FE109*	00043E30C*	00007A520B*	0	0	0	0	0	00	00	00
24	24	0422E8F*	009FC213*	00087C619*	0000F4A417*	0	1	0	0	1	00	00	00
25	25	0845D1E*	013F8426*	0010F8C32*	0001E9482F*	1	0	0	0	1	00	00	00
26	26	108BA3D	027F084D*	0021F1864*	0003D2905E*	0	0	0	0	0	00	00	00
27	27	011747B	04FE109B*	0043E30C9*	0007A520BC*	0	1	0	0	1	00	00	00
28	28	022E8F6	09FC2136*	0087C6192*	000F4A4179*	0	1	0	0	1	00	00	00
29	29	045D1EC	13F8426C*	010F8C325*	001E9482F2*	0	1	0	0	1	00	00	00
30	30	08BA3D9	27F084D8*	021F1864B*	003D2905E5*	1	1	0	0	0	01	00	00
31	31	11747B3	4FE109B0*	043E30C97*	007A520BCA*	0	1	0	0	1	00	00	00
32	32	02E8F67	1FC21360	087C6192E*	00F4A41795*	0	1	1	1	1	01	00	00
33	33	05D1ECF	3F8426C1	10F8C325C*	01E9482F2B*	0	1	0	1	0	01	00	00
34	34	0BA3D9F	7F084D82	01F1864B8	03D2905E56*	1	0	0	1	0	01	00	00

35	35	1747B3E	7E109B04	03E30C970	07A520BCAC*	0	0	0	1	1	00	00	00
36	36	0E8F67C	7C213608	07C6192E1	0F4A417958*	1	0	0	0	1	00	00	00
37	37	1D1ECF8	78426C11	0F8C325C3	1E9482F2B1*	1	0	1	1	1	01	00	00
38	38	1A3D9F0	7084D822	1F1864B86	3D2905E563*	1	1	1	0	1	01	00	00
39	39	147B3E1	6109B044	1E30C970C	7A520BCAC6*	0	0	1	0	1	00	00	00

===

Start clocking Summation Combiner

===

40	1	08F67C2	42136088	1C6192E18	74A417958D	1	0	1	1	1	01	00	00
41	2	11ECF84	0426C111	18C325C30	69482F2B1B	0	0	1	0	0	00	01	00
42	3	03D9F08	084D8222	11864B861	52905E5637	0	0	0	1	1	11	00	01
43	4	07B3E10	109B0444	030C970C3	2520BCAC6E	0	1	0	0	0	01	11	00
44	5	0F67C21	21360889	06192E186	4A417958DC	1	0	0	0	0	10	01	11
45	6	1ECF843	426C1112	0C325C30C	1482F2B1B8	1	0	1	1	1	11	10	01
46	7	1D9F086	04D82225	1864B8619	2905E56370	1	1	1	0	0	01	11	10
47	8	1B3E10D	09B0444B	10C970C32	520BCAC6E1	1	1	0	0	1	10	01	11
48	9	167C21B	13608897	0192E1865	2417958DC3	0	0	0	0	0	00	10	01
49	10	0CF8436	26C1112F	0325C30CB	482F2B1B87	1	1	0	0	0	00	00	10
50	11	19F086D	4D82225E	064B86197	105E56370F	1	1	0	0	0	01	00	00
51	12	13E10DB	1B0444BC	0C970C32F	20BCAC6E1F	0	0	1	1	1	00	01	00
52	13	07C21B7	36088979	192E1865E	417958DC3F	0	0	1	0	1	11	00	01
53	14	0F8436E	6C1112F2	125C30CBD	02F2B1B87F	1	0	0	1	1	01	11	00
54	15	1F086DD	582225E4	04B86197B	05E56370FF	1	0	0	1	1	10	01	11
55	16	1E10DBA	30444BC9	0970C32F7	0BCAC6E1FF	1	0	1	1	1	11	10	01
56	17	1C21B75	60889793	12E1865EE	17958DC3FF	1	1	0	1	0	01	11	10
57	18	18436EA	41112F27	05C30CBDD	2F2B1B87FF	1	0	0	0	0	10	01	11
58	19	1086DD4	02225E4E	0B86197BA	5E56370FFF	0	0	1	0	1	00	10	01
59	20	010DBA8	0444BC9D	170C32F74	3CAC6E1FFF	0	0	0	1	1	01	00	10
60	21	021B750	0889793A	0E1865EE8	7958DC3FFF	0	1	1	0	1	00	01	00
61	22	0436EA0	1112F274	1C30CBDD0	72B1B87FFE	0	0	1	1	0	10	00	01
62	23	086DD40	2225E4E9	186197BA1	656370FFFC	1	0	1	0	0	00	10	00
63	24	10DBA81	444BC9D3	10C32F743	4AC6E1FFF8	0	0	0	1	1	01	00	10
64	25	01B7502	089793A7	01865EE86	158DC3FFF1	0	1	0	1	1	00	01	00
65	26	036EA05	112F274E	030CBDD0D	2B1B87FFE3	0	0	0	0	0	11	00	01
66	27	06DD40B	225E4E9C	06197BA1A	56370FFFC6	0	0	0	0	1	10	11	00
67	28	0DBA817	44BC9D39	0C32F7434	2C6E1FFF8D	1	1	1	0	1	10	10	11
68	29	1B7502E	09793A72	1865EE868	58DC3FFF1B	1	0	1	1	1	01	10	10
69	30	16EA05D	12F274E5	10CBDD0D0	31B87FFE36	0	1	0	1	1	01	01	10
70	31	0DD40BA	25E4E9CB	0197BA1A1	6370FFFC6D	1	1	0	0	1	11	01	01
71	32	1BA8174	4BC9D397	032F74343	46E1FFF8DA	1	1	0	1	0	11	11	01
72	33	17502E8	1793A72F	065EE8687	0DC3FFF1B4	0	1	0	1	1	11	11	11
73	34	0EA05D0	2F274E5E	0CBDD0D0F	1B87FFE369	1	0	1	1	0	10	11	11
74	35	1D40BA0	5E4E9CBD	197BA1A1F	370FFFC6D2	1	0	1	0	0	10	10	11
75	36	1A81741	3C9D397B	12F74343F	6E1FFF8DA5	1	1	0	0	0	01	10	10
76	37	1502E82	793A72F6	05EE8687F	5C3FFF1B4B	0	0	0	0	1	00	01	10
77	38	0A05D05	7274E5ED	0BDD0D0FF	387FFE3696	1	0	1	0	0	10	00	01
78	39	140BA0B	64E9CBDA	17BA1A1FF	70FFFC6D2C	0	1	0	1	0	00	10	00
79	40	0817416	49D397B4	0F74343FE	61FFF8DA59	1	1	1	1	0	11	00	10
80	41	102E82C	13A72F69	1EE8687FD	43FFF1B4B3	0	1	1	1	0	00	11	00
81	42	005D058	274E5ED2	1DD0D0FFA	07FFE36966	0	0	1	1	0	11	00	11
82	43	00BA0B0	4E9CBDA5	1BA1A1FF5	0FFFC6D2CD	0	1	1	1	0	00	11	00
83	44	0174160	1D397B4A	174343FEA	1FFF8DA59B	0	0	0	1	1	10	00	11
84	45	02E82C0	3A72F695	0E8687FD4	3FFF1B4B37	0	0	1	1	0	00	10	00
85	46	05D0580	74E5ED2B	1D0D0FFA9	7FFE36966E	0	1	1	1	1	00	00	10
86	47	0BA0B00	69CBDA56	1A1A1FF53	7FFC6D2CDC	1	1	1	1	0	10	00	00
87	48	1741600	5397B4AC	14343FEA6	7FF8DA59B8	0	1	0	1	0	00	10	00
88	49	0E82C01	272F6959	08687FD4D	7FF1B4B370	1	0	1	1	1	00	00	10

89	50	1D05802	4E5ED2B3	10D0FFA9A	7FE36966E0	1	0	0	1	0	01	00	00
90	51	1A0B004	1CBDA566	01A1FF535	7FC6D2CDC0	1	1	0	1	0	11	01	00
91	52	1416009	397B4ACC	0343FEA6B	7F8DA59B80	0	0	0	1	0	10	11	01
92	53	082C013	72F69599	0687FD4D7	7F1B4B3701	1	1	0	0	0	10	10	11
93	54	1058026	65ED2B33	0D0FFA9AF	7E36966E03	0	1	1	0	0	01	10	10
94	55	00B004D	4BDA5667	1A1FF535E	7C6D2CDC06	0	1	1	0	1	01	01	10
95	56	016009B	17B4ACCE	143FEA6BD	78DA59B80D	0	1	0	1	1	11	01	01
96	57	02C0137	2F69599D	087FD4D7B	71B4B3701A	0	0	1	1	1	10	11	01
97	58	058026F	5ED2B33B	10FFA9AF6	636966E034	0	1	0	0	1	01	10	11
98	59	0B004DF	3DA56677	01FF535ED	46D2CDC068	1	1	0	1	0	10	01	10
99	60	16009BF	7B4ACCEF	03FEA6BDB	0DA59B80D0	0	0	0	1	1	00	10	01
100	61	0C0137F	769599DF	07FD4D7B7	1B4B3701A1	1	1	0	0	0	00	00	10
101	62	18026FE	6D2B33BE	0FFA9AF6E	36966E0342	1	0	1	1	1	01	00	00
102	63	1004DFC	5A56677D	1FF535EDD	6D2CDC0684	0	0	1	0	0	00	01	00
103	64	0009BF9	34ACCEFB	1FEA6BDBB	5A59B80D09	0	1	1	0	0	10	00	01
104	65	00137F2	69599DF7	1FD4D7B76	34B3701A12	0	0	1	1	0	00	10	00
105	66	0026FE5	52B33BEF	1FA9AF6EC	6966E03424	0	1	1	0	0	00	00	10
106	67	004DFCA	256677DF	1F535EDD8	52CDC06848	0	0	1	1	0	01	00	00
107	68	009BF94	4ACCEFBE	1EA6BDBB0	259B80D091	0	1	1	1	0	11	01	00
108	69	0137F29	1599DF7C	1D4D7B760	4B3701A123	0	1	1	0	1	10	11	01
109	70	026FE53	2B33BEF9	1A9AF6EC0	166E034246	0	0	1	0	1	01	10	11
110	71	04DFCA7	56677DF2	1535EDD81	2CDC06848D	0	0	0	1	0	01	01	10
111	72	09BF94F	2CCEFBE4	0A6BDBB03	59B80D091B	1	1	1	1	1	00	01	01
112	73	137F29E	599DF7C9	14D7B7607	33701A1236	0	1	0	0	1	11	00	01
113	74	06FE53C	333BEF93	09AF6EC0E	66E034246C	0	0	1	1	1	01	11	00
114	75	0DFCA79	6677DF26	135EDD81D	4DC06848D8	1	0	0	1	1	10	01	11
115	76	1BF94F2	4CEFBE4D	06BDBB03B	1B80D091B1	1	1	0	1	1	11	10	01
116	77	17F29E5	19DF7C9A	0D7B76077	3701A12363	0	1	1	0	1	00	11	10
117	78	0FE53CA	33BEF934	1AF6EC0EF	6E034246C6	1	1	1	0	1	11	00	11
118	79	1FCA794	677DF269	15EDD81DF	5C06848D8C	1	0	0	0	0	01	11	00
119	80	1F94F29	4EFBE4D2	0BDBB03BE	380D091B19	1	1	1	0	0	01	01	11
120	81	1F29E53	1DF7C9A5	17B76077D	701A123633	1	1	0	0	1	11	01	01
121	82	1E53CA6	3BEF934B	0F6EC0EFB	6034246C66	1	1	1	0	0	11	11	01
122	83	1CA794D	77DF2696	1EDD81DF6	406848D8CD	1	1	1	0	0	10	11	11
123	84	194F29B	6FBE4D2C	1DBB03BED	00D091B19B	1	1	1	1	0	11	10	11
124	85	129E536	5F7C9A59	1B76077DA	01A1236337	0	0	1	1	1	00	11	10
125	86	053CA6C	3EF934B3	16EC0EFB4	034246C66E	0	1	0	0	1	10	00	11
126	87	0A794D9	7DF26967	0DD81DF69	06848D8CDD	1	1	1	1	0	01	10	00
127	88	14F29B3	7BE4D2CF	1BB03BED3	0D091B19BB	0	1	1	0	1	01	01	10
128	89	09E5366	77C9A59F	176077DA6	1A12363377	1	1	0	0	1	11	01	01
129	90	13CA6CD	6F934B3F	0EC0EFB4D	34246C66EF	0	1	1	0	1	10	11	01
130	91	0794D9B	5F26967F	1D81DF69A	6848D8CDDF	0	0	1	0	1	01	10	11
131	92	0F29B37	3E4D2CFE	1B03BED35	5091B19BBE	1	0	1	1	0	10	01	10
132	93	1E5366F	7C9A59FD	16077DA6B	212363377C	1	1	0	0	0	11	10	01
133	94	1CA6CDF	7934B3FB	0C0EFB4D6	4246C66EF9	1	0	1	0	1	00	11	10
134	95	194D9BE	726967F6	181DF69AD	048D8CDDF2	1	0	1	1	1	11	00	11
135	96	129B37D	64D2CFED	103BED35B	091B19BBE5	0	1	0	0	0	01	11	00
136	97	05366FA	49A59FDA	0077DA6B7	12363377CA	0	1	0	0	0	10	01	11
137	98	0A6CDF5	134B3FB4	00EFB4D6E	246C66EF95	1	0	0	0	1	00	10	01
138	99	14D9BEA	26967F69	01DF69ADD	48D8CDDF2B	0	1	0	1	0	00	00	10
139	100	09B37D4	4D2CFED2	03BED35BB	11B19BBE56	1	0	0	1	0	01	00	00
140	101	1366FA8	1A59FDA5	077DA6B77	2363377CAC	0	0	0	0	1	01	01	00
141	102	06CDF51	34B3FB4A	0EFB4D6EF	46C66EF959	0	1	1	1	0	00	01	01
142	103	0D9BEA2	6967F695	1DF69ADDF	0D8CDDF2B2	1	0	1	1	1	10	00	01
143	104	1B37D45	52CFED2A	1BED35BBF	1B19BBE564	1	1	1	0	1	00	10	00
144	105	166FA8A	259FDA54	17DA6B77E	363377CAC8	0	1	0	0	1	01	00	10
145	106	0CDF515	4B3FB4A9	0FB4D6EFC	6C66EF9591	1	0	1	0	1	00	01	00

146	107	19BEA2B	167F6952	1F69ADDF8	58CDDF2B22	1	0	1	1	1	10	00	01
147	108	137D457	2CFED2A5	1ED35BBF1	319BBE5645	0	1	1	1	1	00	10	00
148	109	06FA8AF	59FDA54A	1DA6B77E2	63377CAC8B	0	1	1	0	0	00	00	10
149	110	0DF515F	33FB4A95	1B4D6EFC4	466EF95916	1	1	1	0	1	01	00	00
150	111	1BEA2BF	67F6952A	169ADDF88	0CDDF2B22C	1	1	0	1	0	11	01	00
151	112	17D457F	4FED2A55	0D35BBF10	19BBE56459	0	1	1	1	0	11	11	01
152	113	0FA8AFE	1FDA54AB	1A6B77E20	3377CAC8B3	1	1	1	0	0	10	11	11
153	114	1F515FD	3FB4A957	14D6EFC40	66EF959166	1	1	0	1	1	10	10	11
154	115	1EA2BFA	7F6952AF	09ADDF880	4DDF2B22CC	1	0	1	1	1	01	10	10
155	116	1D457F4	7ED2A55F	135BBF100	1BBE564598	1	1	0	1	0	10	01	10
156	117	1A8AFE8	7DA54ABF	06B77E200	377CAC8B31	1	1	0	0	0	11	10	01
157	118	1515FD0	7B4A957F	0D6EFC401	6EF9591663	0	0	1	1	1	00	11	10
158	119	0A2BFA1	76952AFE	1ADDF8803	5DF2B22CC7	1	1	1	1	0	00	00	11
159	120	1457F42	6D2A55FD	15BBF1007	3BE564598E	0	0	0	1	1	00	00	00
160	121	08AFE84	5A54ABFB	0B77E200F	77CAC8B31C	1	0	1	1	1	01	00	00
161	122	115FD09	34A957F7	16EFC401F	6F95916639	0	1	0	1	1	00	01	00
162	123	02BFA12	6952AFEF	0DDF8803E	5F2B22CC73	0	0	1	0	1	11	00	01
163	124	057F424	52A55FDF	1BBF1007D	3E564598E7	0	1	1	0	1	01	11	00
164	125	0AFE848	254ABFBF	177E200FA	7CAC8B31CF	1	0	0	1	1	10	01	11
165	126	15FD090	4A957F7E	0EFC401F5	795916639E	0	1	1	0	0	11	10	01
166	127	0BFA121	152AFEFD	1DF8803EA	72B22CC73C	1	0	1	1	0	01	11	10
167	128	17F4243	2A55FDFA	1BF1007D4	6564598E78	0	0	1	0	0	10	01	11
168	129	0FE8486	54ABFBF4	17E200FA8	4AC8B31CF0	1	1	0	1	1	11	10	01
169	130	1FD090C	2957F7E8	0FC401F51	15916639E1	1	0	1	1	0	01	11	10
170	131	1FA1219	52AFEFD1	1F8803EA3	2B22CC73C2	1	1	1	0	0	01	01	11
171	132	1F42432	255FDFA2	1F1007D47	564598E785	1	0	1	0	1	11	01	01
172	133	1E84865	4ABFBF44	1E200FA8F	2C8B31CF0B	1	1	1	1	1	11	11	01
173	134	1D090CB	157F7E88	1C401F51E	5916639E17	1	0	1	0	1	11	11	11
174	135	1A12196	2AFEFD11	18803EA3C	322CC73C2E	1	1	1	0	0	10	11	11
175	136	142432C	55FDFA23	11007D479	64598E785C	0	1	0	0	1	01	10	11
176	137	0848659	2BFBF446	0200FA8F2	48B31CF0B9	1	1	0	1	0	10	01	10
177	138	1090CB2	57F7E88C	0401F51E4	116639E173	0	1	0	0	1	00	10	01
178	139	0121964	2FEFD118	0803EA3C8	22CC73C2E6	0	1	1	1	1	00	00	10
179	140	02432C9	5FDFA230	1007D4791	4598E785CD	0	1	0	1	0	01	00	00
180	141	0486593	3FBF4461	000FA8F23	0B31CF0B9B	0	1	0	0	0	00	01	00
181	142	090CB26	7F7E88C3	001F51E47	16639E1736	1	0	0	0	1	11	00	01
182	143	121964D	7EFD1187	003EA3C8F	2CC73C2E6C	0	1	0	1	1	01	11	00
183	144	0432C9B	7DFA230E	007D4791E	598E785CD8	0	1	0	1	1	10	01	11
184	145	0865936	7BF4461C	00FA8F23C	331CF0B9B0	1	1	0	0	0	11	10	01
185	146	10CB26D	77E88C38	01F51E479	6639E17361	0	1	0	0	0	00	11	10
186	147	01964DA	6FD11870	03EA3C8F2	4C73C2E6C2	0	1	0	0	1	10	00	11
187	148	032C9B4	5FA230E1	07D4791E4	18E785CD84	0	1	0	1	0	00	10	00
188	149	0659368	3F4461C2	0FA8F23C9	31CF0B9B09	0	0	1	1	0	00	00	10
189	150	0CB26D0	7E88C384	1F51E4793	639E173612	1	1	1	1	0	10	00	00
190	151	1964DA0	7D118709	1EA3C8F27	473C2E6C24	1	0	1	0	0	00	10	00
191	152	12C9B41	7A230E12	1D4791E4E	0E785CD848	0	0	1	0	1	01	00	10
192	153	0593683	74461C24	1A8F23C9C	1CF0B9B091	0	0	1	1	1	00	01	00
193	154	0B26D06	688C3848	151E47938	39E1736123	1	1	0	1	1	10	00	01
194	155	164DA0D	51187091	0A3C8F271	73C2E6C247	0	0	1	1	0	00	10	00
195	156	0C9B41A	2230E123	14791E4E3	6785CD848F	1	0	0	1	0	00	00	10
196	157	1936835	4461C247	08F23C9C6	4F0B9B091E	1	0	1	0	0	01	00	00
197	158	126D06A	08C3848E	11E47938D	1E1736123C	0	1	0	0	0	00	01	00
198	159	04DA0D5	1187091C	03C8F271B	3C2E6C2478	0	1	0	0	1	11	00	01
199	160	09B41AA	230E1238	0791E4E37	785CD848F1	1	0	0	0	0	01	11	00
200	161	1368354	461C2470	0F23C9C6F	70B9B091E3	0	0	1	1	1	10	01	11
201	162	06D06A9	0C3848E1	1E47938DF	61736123C6	0	0	1	0	1	00	10	01
202	163	0DA0D52	187091C3	1C8F271BE	42E6C2478D	1	0	1	1	1	00	00	10

203	164	1B41AA4	30E12387	191E4E37C	05CD848F1A	1	1	1	1	0	10	00	00
204	165	1683549	61C2470F	123C9C6F9	0B9B091E34	0	1	0	1	0	00	10	00
205	166	0D06A92	43848E1E	047938DF3	1736123C68	1	1	0	0	0	00	00	10
206	167	1A0D524	07091C3C	08F271BE7	2E6C2478D1	1	0	1	0	0	01	00	00
207	168	141AA49	0E123879	11E4E37CF	5CD848F1A2	0	0	0	1	0	00	01	00
208	169	0835492	1C2470F3	03C9C6F9F	39B091E345	1	0	0	1	0	10	00	01
209	170	106A925	3848E1E6	07938DF3F	736123C68B	0	0	0	0	0	11	10	00
210	171	00D524A	7091C3CD	0F271BE7E	66C2478D16	0	1	1	1	0	01	11	10
211	172	01AA495	6123879B	1E4E37CFD	4D848F1A2D	0	0	1	1	1	10	01	11
212	173	035492A	42470F36	1C9C6F9FB	1B091E345B	0	0	1	0	1	00	10	01
213	174	06A9255	048E1E6C	1938DF3F6	36123C68B7	0	1	1	0	0	00	00	10
214	175	0D524AB	091C3CD8	1271BE7EC	6C2478D16E	1	0	0	0	1	00	00	00
215	176	1AA4957	123879B1	04E37CFD8	5848F1A2DD	1	0	0	0	1	00	00	00
216	177	15492AF	2470F363	09C6F9FB0	3091E345BA	0	0	1	1	0	01	00	00
217	178	0A9255E	48E1E6C7	138DF3F61	6123C68B75	1	1	0	0	1	00	01	00
218	179	1524ABD	11C3CD8F	071BE7EC3	42478D16EB	0	1	0	0	1	11	00	01
219	180	0A4957B	23879B1F	0E37CFD87	048F1A2DD6	1	1	1	1	1	00	11	00
220	181	1492AF6	470F363F	1C6F9FB0E	091E345BAD	0	0	1	0	1	10	00	11
221	182	09255EC	0E1E6C7F	18DF3F61D	123C68B75B	1	0	1	0	0	00	10	00
222	183	124ABD9	1C3CD8FF	11BE7EC3A	2478D16EB6	0	0	0	0	0	01	00	10
223	184	04957B3	3879B1FE	037CFD874	48F1A2DD6D	0	0	0	1	0	00	01	00
224	185	092AF66	70F363FD	06F9FB0E9	11E345BADB	1	1	0	1	1	10	00	01
225	186	1255ECD	61E6C7FA	0DF3F61D3	23C68B75B7	0	1	1	1	1	00	10	00
226	187	04ABD9B	43CD8FF5	1BE7EC3A7	478D16EB6E	0	1	1	1	1	00	00	10
227	188	0957B37	079B1FEA	17CFD874E	0F1A2DD6DD	1	1	0	0	0	01	00	00
228	189	12AF66F	0F363FD4	0F9FB0E9C	1E345BADBB	0	0	1	0	0	00	01	00
229	190	055ECDE	1E6C7FA9	1F3F61D39	3C68B75B76	0	0	1	0	1	11	00	01
230	191	0ABD9BC	3CD8FF53	1E7EC3A73	78D16EB6EC	1	1	1	1	1	00	11	00
231	192	157B379	79B1FEA7	1CFD874E6	71A2DD6DD9	0	1	1	1	1	11	00	11
232	193	0AF66F3	7363FD4E	19FB0E9CD	6345BADBB2	1	0	1	0	1	01	11	00
233	194	15ECDE6	66C7FA9D	13F61D39A	468B75B765	0	1	0	1	1	10	01	11
234	195	0BD9BCC	4D8FF53A	07EC3A735	0D16EB6ECA	1	1	0	0	0	11	10	01
235	196	17B3799	1B1FEA75	0FD874E6A	1A2DD6DD94	0	0	1	0	0	00	11	10
236	197	0F66F33	363FD4EA	1FB0E9CD5	345BADBB28	1	0	1	0	0	11	00	11
237	198	1ECDE67	6C7FA9D5	1F61D39AA	68B75B7650	1	0	1	1	0	00	11	00
238	199	1D9BCCF	58FF53AB	1EC3A7354	516EB6ECA0	1	1	1	0	1	11	00	11
239	200	1B3799E	31FEA756	1D874E6A8	22DD6DD940	1	1	1	1	1	00	11	00

Z[0] = 3F
Z[1] = B1
Z[2] = 67
Z[3] = D2
Z[4] = 2F
Z[5] = A6
Z[6] = 1F
Z[7] = B9
Z[8] = E6
Z[9] = 84
Z[10] = 43
Z[11] = 07
Z[12] = D8
Z[13] = 1E
Z[14] = E7
Z[15] = C3

==

Reload this pattern into the LFSRs

Hold content of Summation Combiner regs and calculate new C[t+1] and Z values

==

LFSR1 <= 0E62F3F
LFSR2 <= 6C84A6B1
LFSR3 <= 11E431F67
LFSR4 <= 61E707B9D2
C[t+1] <= 00

==

Generating 125 key symbols (encryption/decryption sequence)

==

240	1	0E62F3F	6C84A6B1	11E431F67	61E707B9D2	1	1	0	1	0	00	11	00
241	2	1CC5E7F	59094D63	03C863ECE	43CE0F73A5	1	0	0	1	0	11	00	11
242	3	198BCFF	32129AC6	0790C7D9D	079C1EE74A	1	0	0	1	1	01	11	00
243	4	13179FE	6425358C	0F218FB3A	0F383DCE94	0	0	1	0	0	10	01	11
244	5	062F3FD	484A6B19	1E431F675	1E707B9D28	0	0	1	0	1	00	10	01
245	6	0C5E7FB	1094D632	1C863ECEB	3CE0F73A50	1	1	1	1	0	11	00	10
246	7	18BCFF7	2129AC64	190C7D9D7	79C1EE74A1	1	0	1	1	0	00	11	00
247	8	1179FEE	425358C8	1218FB3AE	7383DCE942	0	0	0	1	1	10	00	11
248	9	02F3FDD	04A6B190	0431F675D	6707B9D285	0	1	0	0	1	11	10	00
249	10	05E7FBB	094D6320	0863ECEBB	4E0F73A50B	0	0	1	0	0	00	11	10
250	11	0BCFF77	129AC640	10C7D9D77	1C1EE74A16	1	1	0	0	0	11	00	11
251	12	179FEEE	25358C80	018FB3AEE	383DCE942C	0	0	0	0	1	10	11	00
252	13	0F3FDDC	4A6B1900	031F675DD	707B9D2859	1	0	0	0	1	01	10	11
253	14	1E7FBB8	14D63200	063ECEBBA	60F73A50B3	1	1	0	1	0	10	01	10
254	15	1CFF771	29AC6401	0C7D9D774	41EE74A167	1	1	1	1	0	10	10	01
255	16	19FEEE2	5358C803	18FB3AEE9	03DCE942CE	1	0	1	1	1	01	10	10
256	17	13FDDC4	26B19007	11F675DD2	07B9D2859C	0	1	0	1	1	01	01	10
257	18	07FBB88	4D63200E	03ECEBBA4	0F73A50B38	0	0	0	0	1	10	01	01
258	19	0FF7711	1AC6401D	07D9D7748	1EE74A1670	1	1	0	1	1	11	10	01
259	20	1FEEE23	358C803B	0FB3AEE91	3DCE942CE1	1	1	1	1	1	01	11	10
260	21	1FDDC47	6B190076	1F675DD23	7B9D2859C2	1	0	1	1	0	01	01	11
261	22	1FBB88F	563200ED	1ECEBBA47	773A50B385	1	0	1	0	1	11	01	01
262	23	1F7711E	2C6401DB	1D9D7748F	6E74A1670A	1	0	1	0	1	10	11	01
263	24	1EEE23D	58C803B6	1B3AEE91E	5CE942CE15	1	1	1	1	0	11	10	11
264	25	1DDC47A	3190076C	1675DD23D	39D2859C2B	1	1	0	1	0	01	11	10
265	26	1BB88F4	63200ED9	0CEBBA47A	73A50B3856	1	0	1	1	0	01	01	11
266	27	17711E8	46401DB2	19D7748F5	674A1670AD	0	0	1	0	0	11	01	01
267	28	0EE23D0	0C803B64	13AEE91EA	4E942CE15B	1	1	0	1	0	11	11	01
268	29	1DC47A0	190076C8	075DD23D4	1D2859C2B7	1	0	0	0	0	11	11	11
269	30	1B88F41	3200ED90	0EBBA47A9	3A50B3856E	1	0	1	0	1	11	11	11
270	31	1711E83	6401DB20	1D7748F53	74A1670ADC	0	0	1	1	1	11	11	11
271	32	0E23D07	4803B641	1AEE91EA7	6942CE15B8	1	0	1	0	1	11	11	11
272	33	1C47A0F	10076C82	15DD23D4F	52859C2B71	1	0	0	1	1	11	11	11
273	34	188F41E	200ED905	0BBA47A9E	250B3856E3	1	0	1	0	1	11	11	11
274	35	111E83C	401DB20A	17748F53D	4A1670ADC7	0	0	0	0	1	00	11	11
275	36	023D078	003B6414	0EE91EA7A	142CE15B8E	0	0	1	0	1	10	00	11
276	37	047A0F0	0076C828	1DD23D4F5	2859C2B71C	0	0	1	0	1	11	10	00
277	38	08F41E1	00ED9050	1BA47A9EA	50B3856E39	1	1	1	1	1	01	11	10
278	39	11E83C2	01DB20A0	1748F53D5	21670ADC72	0	1	0	0	0	10	01	11
279	40	03D0785	03B64141	0E91EA7AA	42CE15B8E4	0	1	1	1	1	11	10	01
280	41	07A0F0A	076C8283	1D23D4F54	059C2B71C8	0	0	1	1	1	00	11	10
281	42	0F41E14	0ED90507	1A47A9EA9	0B3856E390	1	1	1	0	1	11	00	11
282	43	1E83C29	1DB20A0F	148F53D52	1670ADC720	1	1	0	0	1	01	11	00
283	44	1D07853	3B64141E	091EA7AA5	2CE15B8E40	1	0	1	1	0	01	01	11

284	45	1A0F0A6	76C8283C	123D4F54B	59C2B71C81	1	1	0	1	0	00	01	01
285	46	141E14C	6D905079	047A9EA97	33856E3902	0	1	0	1	0	10	00	01
286	47	083C299	5B20A0F2	08F53D52F	670ADC7204	1	0	1	0	0	00	10	00
287	48	1078533	364141E4	11EA7AA5E	4E15B8E408	0	0	0	0	0	01	00	10
288	49	00F0A67	6C8283C8	03D4F54BC	1C2B71C811	0	1	0	0	0	00	01	00
289	50	01E14CE	59050791	07A9EA978	3856E39022	0	0	0	0	0	11	00	01
290	51	03C299C	320A0F23	0F53D52F1	70ADC72045	0	0	1	1	1	01	11	00
291	52	0785339	64141E47	1EA7AA5E2	615B8E408A	0	0	1	0	0	10	01	11
292	53	0F0A673	48283C8E	1D4F54BC4	42B71C8115	1	0	1	1	1	11	10	01
293	54	1E14CE6	1050791C	1A9EA9788	056E39022B	1	0	1	0	1	00	11	10
294	55	1C299CD	20A0F239	153D52F10	0ADC720456	1	1	0	1	1	11	00	11
295	56	185339B	4141E472	0A7AA5E20	15B8E408AC	1	0	1	1	0	00	11	00
296	57	10A6736	0283C8E4	14F54BC41	2B71C81158	0	1	0	0	1	10	00	11
297	58	014CE6C	050791C9	09EA97882	56E39022B0	0	0	1	1	0	00	10	00
298	59	0299CD9	0A0F2393	13D52F104	2DC7204561	0	0	0	1	1	01	00	10
299	60	05339B3	141E4726	07AA5E208	5B8E408AC3	0	0	0	1	0	00	01	00
300	61	0A67366	283C8E4C	0F54BC411	371C811587	1	0	1	0	0	10	00	01
301	62	14CE6CC	50791C98	1EA978822	6E39022B0F	0	0	1	0	1	11	10	00
302	63	099CD99	20F23930	1D52F1045	5C7204561E	1	1	1	0	0	01	11	10
303	64	1339B33	41E47260	1AA5E208B	38E408AC3D	0	1	1	1	0	01	01	11
304	65	0673666	03C8E4C0	154BC4117	71C811587A	0	1	0	1	1	11	01	01
305	66	0CE6CCC	0791C980	0A978822E	639022B0F5	1	1	1	1	1	11	11	01
306	67	19CD999	0F239301	152F1045C	47204561EB	1	0	0	0	0	11	11	11
307	68	139B332	1E472603	0A5E208B9	0E408AC3D6	0	0	1	0	0	11	11	11
308	69	0736664	3C8E4C06	14BC41172	1C811587AD	0	1	0	1	1	11	11	11
309	70	0E6CCC8	791C980C	0978822E5	39022B0F5A	1	0	1	0	1	11	11	11
310	71	1CD9990	72393019	12F1045CB	7204561EB4	1	0	0	0	0	11	11	11
311	72	19B3320	64726033	05E208B97	6408AC3D69	1	0	0	0	0	11	11	11
312	73	1366640	48E4C067	0BC41172F	4811587AD3	0	1	1	0	1	11	11	11
313	74	06CCC81	11C980CF	178822E5E	1022B0F5A6	0	1	0	0	0	11	11	11
314	75	0D99903	2393019E	0F1045CBC	204561EB4C	1	1	1	0	0	10	11	11
315	76	1B33206	4726033D	1E208B979	408AC3D699	1	0	1	1	1	10	10	11
316	77	166640D	0E4C067B	1C41172F2	011587AD33	0	0	1	0	1	10	10	10
317	78	0CCC81B	1C980CF6	18822E5E5	022B0F5A66	1	1	1	0	1	01	10	10
318	79	1999036	393019EC	11045CBCA	04561EB4CD	1	0	0	0	0	01	01	10
319	80	133206C	726033D9	0208B9794	08AC3D699B	0	0	0	1	0	11	01	01
320	81	06640D9	64C067B3	041172F29	11587AD337	0	1	0	0	0	10	11	01
321	82	0CC81B3	4980CF66	0822E5E53	22B0F5A66F	1	1	1	1	0	11	10	11
322	83	1990366	13019ECC	1045CBCA6	4561EB4CDF	1	0	0	0	0	00	11	10
323	84	13206CC	26033D98	008B9794D	0AC3D699BE	0	0	0	1	1	10	00	11
324	85	0640D98	4C067B31	01172F29B	1587AD337C	0	0	0	1	1	11	10	00
325	86	0C81B30	180CF662	022E5E537	2B0F5A66F9	1	0	0	0	0	00	11	10
326	87	1903660	3019ECC5	045CBCA6F	561EB4CDF3	1	0	0	0	1	10	00	11
327	88	1206CC1	6033D98A	08B9794DE	2C3D699BE6	0	0	1	0	1	11	10	00
328	89	040D983	4067B315	1172F29BD	587AD337CC	0	0	0	0	1	11	11	10
329	90	081B306	00CF662A	02E5E537A	30F5A66F98	1	1	0	1	0	10	11	11
330	91	103660C	019ECC55	05CBCA6F4	61EB4CDF31	0	1	0	1	0	10	10	11
331	92	006CC19	033D98AB	0B9794DE8	43D699BE62	0	0	1	1	0	01	10	10
332	93	00D9833	067B3156	172F29BD0	07AD337CC5	0	0	0	1	0	01	01	10
333	94	01B3066	0CF662AC	0E5E537A0	0F5A66F98B	0	1	1	0	1	11	01	01
334	95	03660CD	19ECC559	1CBCA6F41	1EB4CDF317	0	1	1	1	0	11	11	01
335	96	06CC19B	33D98AB2	19794DE83	3D699BE62F	0	1	1	0	1	11	11	11
336	97	0D98336	67B31565	12F29BD06	7AD337CC5F	1	1	0	1	0	10	11	11
337	98	1B3066D	4F662ACA	05E537A0C	75A66F98BF	1	0	0	1	0	10	10	11
338	99	1660CDB	1ECC5594	0BCA6F418	6B4CDF317E	0	1	1	0	0	01	10	10
339	100	0CC19B7	3D98AB29	1794DE831	5699BE62FC	1	1	0	1	0	10	01	10
340	101	198336F	7B315653	0F29BD062	2D337CC5F9	1	0	1	0	0	11	10	01

341	102	13066DE	7662ACA7	1E537A0C5	5A66F98BF2	0	0	1	0	0	00	11	10
342	103	060CDBC	6CC5594F	1CA6F418B	34CDF317E4	0	1	1	1	1	11	00	11
343	104	0C19B78	598AB29F	194DE8317	699BE62FC9	1	1	1	1	1	00	11	00
344	105	18336F1	3315653F	129BD062E	5337CC5F92	1	0	0	0	1	10	00	11
345	106	1066DE2	662ACA7E	0537A0C5C	266F98BF25	0	0	0	0	0	11	10	00
346	107	00CDBC5	4C5594FD	0A6F418B9	4CDF317E4B	0	0	1	1	1	00	11	10
347	108	019B78B	18AB29FA	14DE83172	19BE62FC96	0	1	0	1	0	11	00	11
348	109	0336F16	315653F4	09BD062E5	337CC5F92C	0	0	1	0	0	01	11	00
349	110	066DE2D	62ACA7E8	137A0C5CA	66F98BF258	0	1	0	1	1	10	01	11
350	111	0CDBC5B	45594FD1	06F418B95	4DF317E4B1	1	0	0	1	0	11	10	01
351	112	19B78B6	0AB29FA2	0DE83172B	1BE62FC962	1	1	1	1	1	01	11	10
352	113	136F16C	15653F45	1BD062E57	37CC5F92C5	0	0	1	1	1	10	01	11
353	114	06DE2D9	2ACA7E8B	17A0C5CAE	6F98BF258B	0	1	0	1	0	11	10	01
354	115	0DBC5B2	5594FD16	0F418B95D	5F317E4B16	1	1	1	0	0	01	11	10
355	116	1B78B64	2B29FA2C	1E83172BB	3E62FC962C	1	0	1	0	1	10	01	11
356	117	16F16C8	5653F458	1D062E577	7CC5F92C58	0	0	1	1	0	11	10	01
357	118	0DE2D91	2CA7E8B0	1A0C5CAEF	798BF258B1	1	1	1	1	1	01	11	10
358	119	1BC5B23	594FD161	1418B95DF	7317E4B163	1	0	0	0	0	10	01	11
359	120	178B647	329FA2C2	083172BBF	662FC962C7	0	1	1	0	0	11	10	01
360	121	0F16C8E	653F4584	1062E577F	4C5F92C58E	1	0	0	0	0	00	11	10
361	122	1E2D91C	4A7E8B09	00C5CAEFE	18BF258B1C	1	0	0	1	0	11	00	11
362	123	1C5B238	14FD1613	018B95DFC	317E4B1639	1	1	0	0	1	01	11	00
363	124	18B6471	29FA2C27	03172BBF9	62FC962C72	1	1	0	1	0	01	01	11
364	125	116C8E2	53F4584E	062E577F3	45F92C58E4	0	1	0	1	1	11	01	01

Bluetooth.

2 FREQUENCY HOPPING SAMPLE DATA—MANDATORY SCHEME

These sets of sample data show the mandatory frequency hopping scheme for three different combinations of addresses and initial clock values. The first part is for the 79-hop system, the second is for the 23-hop system.

2.1 THE 79-HOP SYSTEM SAMPLE DATA

Bluetooth.

2.1.1 First set

```
==================================================================================
                                  79 HOP SYSTEM
==================================================================================

Hop sequence {k} for PAGE SCAN/INQUIRY SCAN:
CLKN start:     0x0000000
ULAP:           0x00000000
#ticks:         0000 | 1000 | 2000 | 3000 | 4000 | 5000 | 6000 | 7000 |
                ----------------------------------------------------------
0x0000000:         0 |    2 |    4 |    6 |    8 |   10 |   12 |   14 |
0x0008000:        16 |   18 |   20 |   22 |   24 |   26 |   28 |   30 |
0x0010000:        32 |   34 |   36 |   38 |   40 |   42 |   44 |   46 |
0x0018000:        48 |   50 |   52 |   54 |   56 |   58 |   60 |   62 |
0x0020000:         0 |    2 |    4 |    6 |    8 |   10 |   12 |   14 |
0x0028000:        16 |   18 |   20 |   22 |   24 |   26 |   28 |   30 |
0x0030000:        32 |   34 |   36 |   38 |   40 |   42 |   44 |   46 |
0x0038000:        48 |   50 |   52 |   54 |   56 |   58 |   60 |   62 |

Hop sequence {k} for PAGE STATE/INQUIRY STATE:
CLKE start:     0x0000000
ULAP:           0x00000000
#ticks:         00 01 02 03 | 04 05 06 07 | 08 09 0a 0b | 0c 0d 0e 0f |
                ----------------------------------------------------------
0x0000000:      48 50 09 13 | 52 54 41 45 | 56 58 11 15 | 60 62 43 47 |
0x0000010:      00 02 64 68 | 04 06 17 21 | 08 10 66 70 | 12 14 19 23 |
0x0000020:      48 50 09 13 | 52 54 41 45 | 56 58 11 15 | 60 62 43 47 |
0x0000030:      00 02 64 68 | 04 06 17 21 | 08 10 66 70 | 12 14 19 23 |
...
0x0001000:      48 18 09 05 | 20 22 33 37 | 24 26 03 07 | 28 30 35 39 |
0x0001010:      32 34 72 76 | 36 38 25 29 | 40 42 74 78 | 44 46 27 31 |
0x0001020:      48 18 09 05 | 20 22 33 37 | 24 26 03 07 | 28 30 35 39 |
0x0001030:      32 34 72 76 | 36 38 25 29 | 40 42 74 78 | 44 46 27 31 |
...
0x0002000:      16 18 01 05 | 52 54 41 45 | 56 58 11 15 | 60 62 43 47 |
0x0002010:      00 02 64 68 | 04 06 17 21 | 08 10 66 70 | 12 14 19 23 |
0x0002020:      16 18 01 05 | 52 54 41 45 | 56 58 11 15 | 60 62 43 47 |
0x0002030:      00 02 64 68 | 04 06 17 21 | 08 10 66 70 | 12 14 19 23 |
...
0x0003000:      48 50 09 13 | 52 22 41 37 | 24 26 03 07 | 28 30 35 39 |
0x0003010:      32 34 72 76 | 36 38 25 29 | 40 42 74 78 | 44 46 27 31 |
0x0003020:      48 50 09 13 | 52 22 41 37 | 24 26 03 07 | 28 30 35 39 |
0x0003030:      32 34 72 76 | 36 38 25 29 | 40 42 74 78 | 44 46 27 31 |

Hop sequence {k} for SLAVE PAGE RESPONSE STATE:
CLKN* =         0x0000010
ULAP:           0x00000000
#ticks:         00 | 02 04 | 06 08 | 0a 0c | 0e 10 | 12 14 | 16 18 | 1a 1c | 1e
                ------------------------------------------------------------------
0x0000012:      64 | 02 68 | 04 17 | 06 21 | 08 66 | 10 70 | 12 19 | 14 23 | 16
0x0000032:      01 | 18 05 | 20 33 | 22 37 | 24 03 | 26 07 | 28 35 | 30 39 | 32
0x0000052:      72 | 34 76 | 36 25 | 38 29 | 40 74 | 42 78 | 44 27 | 46 31 | 48
0x0000072:      09 | 50 13 | 52 41 | 54 45 | 56 11 | 58 15 | 60 43 | 62 47 | 00

Hop sequence {k} for MASTER PAGE RESPONSE STATE:
Offset value:   24
CLKE* =         0x0000012
ULAP:           0x00000000
#ticks:         00 02 | 04 06 | 08 0a | 0c 0e | 10 12 | 14 16 | 18 1a | 1c 1e |
                ------------------------------------------------------------------
0x0000014:      02 68 | 04 17 | 06 21 | 08 66 | 10 70 | 12 19 | 14 23 | 16 01 |
0x0000034:      18 05 | 20 33 | 22 37 | 24 03 | 26 07 | 28 35 | 30 39 | 32 72 |
0x0000054:      34 76 | 36 25 | 38 29 | 40 74 | 42 78 | 44 27 | 46 31 | 48 09 |
0x0000074:      50 13 | 52 41 | 54 45 | 56 11 | 58 15 | 60 43 | 62 47 | 00 64 |
```

```
Hop sequence {k} for CONNECTION STATE:
CLK start:       0x0000010
ULAP:            0x00000000
#ticks:          00 02 | 04 06 | 08 0a | 0c 0e | 10 12 | 14 16 | 18 1a | 1c 1e |
                 ----------------------------------------------------------------
0x0000010:       08 66 | 10 70 | 12 19 | 14 23 | 16 01 | 18 05 | 20 33 | 22 37 |
0x0000030:       24 03 | 26 07 | 28 35 | 30 39 | 32 72 | 34 76 | 36 25 | 38 29 |
0x0000050:       40 74 | 42 78 | 44 27 | 46 31 | 48 09 | 50 13 | 52 41 | 54 45 |
0x0000070:       56 11 | 58 15 | 60 43 | 62 47 | 32 17 | 36 19 | 34 49 | 38 51 |
0x0000090:       40 21 | 44 23 | 42 53 | 46 55 | 48 33 | 52 35 | 50 65 | 54 67 |
0x00000b0:       56 37 | 60 39 | 58 69 | 62 71 | 64 25 | 68 27 | 66 57 | 70 59 |
0x00000d0:       72 29 | 76 31 | 74 61 | 78 63 | 01 41 | 05 43 | 03 73 | 07 75 |
0x00000f0:       09 45 | 13 47 | 11 77 | 15 00 | 64 49 | 66 53 | 68 02 | 70 06 |
0x0000110:       01 51 | 03 55 | 05 04 | 07 08 | 72 57 | 74 61 | 76 10 | 78 14 |
0x0000130:       09 59 | 11 63 | 13 12 | 15 16 | 17 65 | 19 69 | 21 18 | 23 22 |
0x0000150:       33 67 | 35 71 | 37 20 | 39 24 | 25 73 | 27 77 | 29 26 | 31 30 |
0x0000170:       41 75 | 43 00 | 45 28 | 47 32 | 17 02 | 21 04 | 19 34 | 23 36 |
0x0000190:       33 06 | 37 08 | 35 38 | 39 40 | 25 10 | 29 12 | 27 42 | 31 44 |
0x00001b0:       41 14 | 45 16 | 43 46 | 47 48 | 49 18 | 53 20 | 51 50 | 55 52 |
0x00001d0:       65 22 | 69 24 | 67 54 | 71 56 | 57 26 | 61 28 | 59 58 | 63 60 |
0x00001f0:       73 30 | 77 32 | 75 62 | 00 64 | 49 34 | 51 42 | 57 66 | 59 74 |
0x0000210:       53 36 | 55 44 | 61 68 | 63 76 | 65 50 | 67 58 | 73 03 | 75 11 |
0x0000230:       69 52 | 71 60 | 77 05 | 00 13 | 02 38 | 04 46 | 10 70 | 12 78 |
0x0000250:       06 40 | 08 48 | 14 72 | 16 01 | 18 54 | 20 62 | 26 07 | 28 15 |
0x0000270:       22 56 | 24 64 | 30 09 | 32 17 | 02 66 | 06 74 | 10 19 | 14 27 |
0x0000290:       04 70 | 08 78 | 12 23 | 16 31 | 18 03 | 22 11 | 26 35 | 30 43 |
0x00002b0:       20 07 | 24 15 | 28 39 | 32 47 | 34 68 | 38 76 | 42 21 | 46 29 |
0x00002d0:       36 72 | 40 01 | 44 25 | 48 33 | 50 05 | 54 13 | 58 37 | 62 45 |
0x00002f0:       52 09 | 56 17 | 60 41 | 64 49 | 34 19 | 36 35 | 50 51 | 52 67 |
0x0000310:       38 21 | 40 37 | 54 53 | 56 69 | 42 27 | 44 43 | 58 59 | 60 75 |
0x0000330:       46 29 | 48 45 | 62 61 | 64 77 | 66 23 | 68 39 | 03 55 | 05 71 |
0x0000350:       70 25 | 72 41 | 07 57 | 09 73 | 74 31 | 76 47 | 11 63 | 13 00 |
0x0000370:       78 33 | 01 49 | 15 65 | 17 02 | 66 51 | 70 67 | 03 04 | 07 20 |
0x0000390:       68 55 | 72 71 | 05 08 | 09 24 | 74 59 | 78 75 | 11 12 | 15 28 |
0x00003b0:       76 63 | 01 00 | 13 16 | 17 32 | 19 53 | 23 69 | 35 06 | 39 22 |
0x00003d0:       21 57 | 25 73 | 37 10 | 41 26 | 27 61 | 31 77 | 43 14 | 47 30 |
0x00003f0:       29 65 | 33 02 | 45 18 | 49 34 | 19 04 | 21 08 | 23 20 | 25 24 |
```

2.1.2 Second set

```
Set mode:
Set clock:
Set ULAP:

================================================================================
                                   79 HOP SYSTEM
================================================================================

Hop sequence {k} for PAGE SCAN/INQUIRY SCAN:
CLKN start:      0x0000000
ULAP:            0x2a96ef25
#ticks:          0000 | 1000 | 2000 | 3000 | 4000 | 5000 | 6000 | 7000 |
                 --------------------------------------------------------
0x0000000:         49 |   13 |   17 |   51 |   55 |   19 |   23 |   53 |
0x0008000:         57 |   21 |   25 |   27 |   31 |   74 |   78 |   29 |
0x0010000:         33 |   76 |    1 |   35 |   39 |    3 |    7 |   37 |
0x0018000:         41 |    5 |    9 |   43 |   47 |   11 |   15 |   45 |
0x0020000:         49 |   13 |   17 |   51 |   55 |   19 |   23 |   53 |
0x0028000:         57 |   21 |   25 |   27 |   31 |   74 |   78 |   29 |
0x0030000:         33 |   76 |    1 |   35 |   39 |    3 |    7 |   37 |
0x0038000:         41 |    5 |    9 |   43 |   47 |   11 |   15 |   45 |

Hop sequence {k} for PAGE STATE/INQUIRY STATE:
CLKE start:      0x0000000
ULAP:            0x2a96ef25
#ticks:          00 01 02 03 | 04 05 06 07 | 08 09 0a 0b | 0c 0d 0e 0f |
                 --------------------------------------------------------
0x0000000:       41 05 10 04 | 09 43 06 16 | 47 11 18 12 | 15 45 14 32 |
0x0000010:       49 13 34 28 | 17 51 30 24 | 55 19 26 20 | 23 53 22 40 |
0x0000020:       41 05 10 04 | 09 43 06 16 | 47 11 18 12 | 15 45 14 32 |
0x0000030:       49 13 34 28 | 17 51 30 24 | 55 19 26 20 | 23 53 22 40 |
...
0x0001000:       41 21 10 36 | 25 27 38 63 | 31 74 65 59 | 78 29 61 00 |
0x0001010:       33 76 02 75 | 01 35 77 71 | 39 03 73 67 | 07 37 69 08 |
0x0001020:       41 21 10 36 | 25 27 38 63 | 31 74 65 59 | 78 29 61 00 |
0x0001030:       33 76 02 75 | 01 35 77 71 | 39 03 73 67 | 07 37 69 08 |
...
0x0002000:       57 21 42 36 | 09 43 06 16 | 47 11 18 12 | 15 45 14 32 |
0x0002010:       49 13 34 28 | 17 51 30 24 | 55 19 26 20 | 23 53 22 40 |
0x0002020:       57 21 42 36 | 09 43 06 16 | 47 11 18 12 | 15 45 14 32 |
0x0002030:       49 13 34 28 | 17 51 30 24 | 55 19 26 20 | 23 53 22 40 |
...
0x0003000:       41 05 10 04 | 09 27 06 63 | 31 74 65 59 | 78 29 61 00 |
0x0003010:       33 76 02 75 | 01 35 77 71 | 39 03 73 67 | 07 37 69 08 |
0x0003020:       41 05 10 04 | 09 27 06 63 | 31 74 65 59 | 78 29 61 00 |
0x0003030:       33 76 02 75 | 01 35 77 71 | 39 03 73 67 | 07 37 69 08 |

Hop sequence {k} for SLAVE PAGE RESPONSE STATE:
CLKN* =          0x0000010
ULAP:            0x2a96ef25
#ticks:          00 | 02 04 | 06 08 | 0a 0c | 0e 10 | 12 14 | 16 18 | 1a 1c | 1e
                 ----------------------------------------------------------------
0x0000012:       34 | 13 28 | 17 30 | 51 24 | 55 26 | 19 20 | 23 22 | 53 40 | 57
0x0000032:       42 | 21 36 | 25 38 | 27 63 | 31 65 | 74 59 | 78 61 | 29 00 | 33
0x0000052:       02 | 76 75 | 01 77 | 35 71 | 39 73 | 03 67 | 07 69 | 37 08 | 41
0x0000072:       10 | 05 04 | 09 06 | 43 16 | 47 18 | 11 12 | 15 14 | 45 32 | 49

Hop sequence {k} for MASTER PAGE RESPONSE STATE:
Offset value:    24
CLKE* =          0x0000012
ULAP:            0x2a96ef25
#ticks:          00 02 | 04 06 | 08 0a | 0c 0e | 10 12 | 14 16 | 18 1a | 1c 1e |
                 ----------------------------------------------------------------
0x0000014:       13 28 | 17 30 | 51 24 | 55 26 | 19 20 | 23 22 | 53 40 | 57 42 |
0x0000034:       21 36 | 25 38 | 27 63 | 31 65 | 74 59 | 78 61 | 29 00 | 33 02 |
0x0000054:       76 75 | 01 77 | 35 71 | 39 73 | 03 67 | 07 69 | 37 08 | 41 10 |
0x0000074:       05 04 | 09 06 | 43 16 | 47 18 | 11 12 | 15 14 | 45 32 | 49 34 |
```

Hop sequence {k} for CONNECTION STATE:
CLK start: 0x0000010
ULAP: 0x2a96ef25

#ticks:	00 02	04 06	08 0a	0c 0e	10 12	14 16	18 1a	1c 1e
0x0000010:	55 26	19 20	23 22	53 40	57 42	21 36	25 38	27 63
0x0000030:	31 65	74 59	78 61	29 00	33 02	76 75	01 77	35 71
0x0000050:	39 73	03 67	07 69	37 08	41 10	05 04	09 06	43 16
0x0000070:	47 18	11 12	15 14	45 32	02 66	47 60	49 64	04 54
0x0000090:	06 58	51 52	53 56	08 70	10 74	55 68	57 72	59 14
0x00000b0:	61 18	27 12	29 16	63 30	65 34	31 28	33 32	67 22
0x00000d0:	69 26	35 20	37 24	71 38	73 42	39 36	41 40	75 46
0x00000f0:	77 50	43 44	45 48	00 62	26 11	69 05	73 07	36 17
0x0000110:	40 19	04 13	08 15	38 25	42 27	06 21	10 23	12 48
0x0000130:	16 50	59 44	63 46	14 56	18 58	61 52	65 54	28 64
0x0000150:	32 66	75 60	00 62	30 72	34 74	77 68	02 70	20 01
0x0000170:	24 03	67 76	71 78	22 09	58 43	24 37	26 41	68 47
0x0000190:	70 51	36 45	38 49	72 55	74 59	40 53	42 57	44 78
0x00001b0:	46 03	12 76	14 01	48 07	50 11	16 05	18 09	60 15
0x00001d0:	62 19	28 13	30 17	64 23	66 27	32 21	34 25	52 31
0x00001f0:	54 35	20 29	22 33	56 39	19 04	62 63	66 00	07 73
0x0000210:	11 10	54 69	58 06	23 75	27 12	70 71	74 08	76 33
0x0000230:	01 49	44 29	48 45	13 35	17 51	60 31	64 47	05 41
0x0000250:	09 57	52 37	56 53	21 43	25 59	68 39	72 55	78 65
0x0000270:	03 02	46 61	50 77	15 67	51 36	17 18	19 34	41 24
0x0000290:	43 40	09 22	11 38	57 28	59 44	25 26	27 42	29 63
0x00002b0:	31 00	76 61	78 77	45 67	47 04	13 65	15 02	37 71
0x00002d0:	39 08	05 69	07 06	53 75	55 12	21 73	23 10	33 16
0x00002f0:	35 32	01 14	03 30	49 20	75 60	39 48	43 56	00 66
0x0000310:	04 74	47 62	51 70	08 68	12 76	55 64	59 72	61 18
0x0000330:	65 26	29 14	33 22	69 20	73 28	37 16	41 24	77 34
0x0000350:	02 42	45 30	49 38	06 36	10 44	53 32	57 40	63 50
0x0000370:	67 58	31 46	35 54	71 52	28 13	73 03	75 11	34 17
0x0000390:	36 25	02 15	04 23	42 21	44 29	10 19	12 27	14 48
0x00003b0:	16 56	61 46	63 54	22 52	24 60	69 50	71 58	30 64
0x00003d0:	32 72	77 62	00 70	38 68	40 76	06 66	08 74	18 01
0x00003f0:	20 09	65 78	67 07	26 05	44 29	32 23	36 25	70 43

2.1.3 Third set

```
Set mode:
Set clock:
Set ULAP:

=================================================================================
                                   79 HOP SYSTEM
=================================================================================

Hop sequence {k} for PAGE SCAN/INQUIRY SCAN:
CLKN start:      0x0000000
ULAP:            0x6587cba9
#ticks:          0000 | 1000 | 2000 | 3000 | 4000 | 5000 | 6000 | 7000 |
                 --------------------------------------------------------
0x0000000:         16 |   65 |   67 |   18 |   20 |   53 |   55 |    6 |
0x0008000:          8 |   57 |   59 |   10 |   12 |   69 |   71 |   22 |
0x0010000:         24 |   73 |   75 |   26 |   28 |   45 |   47 |   77 |
0x0018000:          0 |   49 |   51 |    2 |    4 |   61 |   63 |   14 |
0x0020000:         16 |   65 |   67 |   18 |   20 |   53 |   55 |    6 |
0x0028000:          8 |   57 |   59 |   10 |   12 |   69 |   71 |   22 |
0x0030000:         24 |   73 |   75 |   26 |   28 |   45 |   47 |   77 |
0x0038000:          0 |   49 |   51 |    2 |    4 |   61 |   63 |   14 |

Hop sequence {k} for PAGE STATE/INQUIRY STATE:
CLKE start:      0x0000000
ULAP:            0x6587cba9
#ticks:          00 01 02 03 | 04 05 06 07 | 08 09 0a 0b | 0c 0d 0e 0f |
                 --------------------------------------------------------
0x0000000:       00 49 36 38 | 51 02 42 40 | 04 61 44 46 | 63 14 50 48 |
0x0000010:       16 65 52 54 | 67 18 58 56 | 20 53 60 62 | 55 06 66 64 |
0x0000020:       00 49 36 38 | 51 02 42 40 | 04 61 44 46 | 63 14 50 48 |
0x0000030:       16 65 52 54 | 67 18 58 56 | 20 53 60 62 | 55 06 66 64 |
...
0x0001000:       00 57 36 70 | 59 10 74 72 | 12 69 76 78 | 71 22 03 01 |
0x0001010:       24 73 05 07 | 75 26 11 09 | 28 45 13 30 | 47 77 34 32 |
0x0001020:       00 57 36 70 | 59 10 74 72 | 12 69 76 78 | 71 22 03 01 |
0x0001030:       24 73 05 07 | 75 26 11 09 | 28 45 13 30 | 47 77 34 32 |
...
0x0002000:       08 57 68 70 | 51 02 42 40 | 04 61 44 46 | 63 14 50 48 |
0x0002010:       16 65 52 54 | 67 18 58 56 | 20 53 60 62 | 55 06 66 64 |
0x0002020:       08 57 68 70 | 51 02 42 40 | 04 61 44 46 | 63 14 50 48 |
0x0002030:       16 65 52 54 | 67 18 58 56 | 20 53 60 62 | 55 06 66 64 |
...
0x0003000:       00 49 36 38 | 51 10 42 72 | 12 69 76 78 | 71 22 03 01 |
0x0003010:       24 73 05 07 | 75 26 11 09 | 28 45 13 30 | 47 77 34 32 |
0x0003020:       00 49 36 38 | 51 10 42 72 | 12 69 76 78 | 71 22 03 01 |
0x0003030:       24 73 05 07 | 75 26 11 09 | 28 45 13 30 | 47 77 34 32 |

Hop sequence {k} for SLAVE PAGE RESPONSE STATE:
CLKN* =          0x0000010
ULAP:            0x6587cba9
#ticks:          00 | 02 04 | 06 08 | 0a 0c | 0e 10 | 12 14 | 16 18 | 1a 1c | 1e
                 ----------------------------------------------------------------
0x0000012:       52 | 65 54 | 67 58 | 18 56 | 20 60 | 53 62 | 55 66 | 06 64 | 08
0x0000032:       68 | 57 70 | 59 74 | 10 72 | 12 76 | 69 78 | 71 03 | 22 01 | 24
0x0000052:       05 | 73 07 | 75 11 | 26 09 | 28 13 | 45 30 | 47 34 | 77 32 | 00
0x0000072:       36 | 49 38 | 51 42 | 02 40 | 04 44 | 61 46 | 63 50 | 14 48 | 16

Hop sequence {k} for MASTER PAGE RESPONSE STATE:
Offset value:    24
CLKE* =          0x0000012
ULAP:            0x6587cba9
#ticks:          00 02 | 04 06 | 08 0a | 0c 0e | 10 12 | 14 16 | 18 1a | 1c 1e |
                 ----------------------------------------------------------------
0x0000014:       65 54 | 67 58 | 18 56 | 20 60 | 53 62 | 55 66 | 06 64 | 08 68 |
0x0000034:       57 70 | 59 74 | 10 72 | 12 76 | 69 78 | 71 03 | 22 01 | 24 05 |
0x0000054:       73 07 | 75 11 | 26 09 | 28 13 | 45 30 | 47 34 | 77 32 | 00 36 |
0x0000074:       49 38 | 51 42 | 02 40 | 04 44 | 61 46 | 63 50 | 14 48 | 16 52 |
```

```
Hop sequence {k} for CONNECTION STATE:
CLK start:      0x0000010
ULAP:           0x6587cba9
#ticks:         00 02 | 04 06 | 08 0a | 0c 0e | 10 12 | 14 16 | 18 1a | 1c 1e |
                ----------------------------------------------------------------
0x0000010:      20 60 | 53 62 | 55 66 | 06 64 | 08 68 | 57 70 | 59 74 | 10 72 |
0x0000030:      12 76 | 69 78 | 71 03 | 22 01 | 24 05 | 73 07 | 75 11 | 26 09 |
0x0000050:      28 13 | 45 30 | 47 34 | 77 32 | 00 36 | 49 38 | 51 42 | 02 40 |
0x0000070:      04 44 | 61 46 | 63 50 | 14 48 | 50 05 | 16 07 | 20 09 | 48 11 |
0x0000090:      52 13 | 06 15 | 10 17 | 38 19 | 42 21 | 08 23 | 12 25 | 40 27 |
0x00000b0:      44 29 | 22 31 | 26 33 | 54 35 | 58 37 | 24 39 | 28 41 | 56 43 |
0x00000d0:      60 45 | 77 62 | 02 64 | 30 66 | 34 68 | 00 70 | 04 72 | 32 74 |
0x00000f0:      36 76 | 14 78 | 18 01 | 46 03 | 72 29 | 42 39 | 44 43 | 74 41 |
0x0000110:      76 45 | 46 47 | 48 51 | 78 49 | 01 53 | 50 63 | 52 67 | 03 65 |
0x0000130:      05 69 | 54 55 | 56 59 | 07 57 | 09 61 | 58 71 | 60 75 | 11 73 |
0x0000150:      13 77 | 30 15 | 32 19 | 62 17 | 64 21 | 34 31 | 36 35 | 66 33 |
0x0000170:      68 37 | 38 23 | 40 27 | 70 25 | 27 61 | 72 71 | 76 73 | 25 75 |
0x0000190:      29 77 | 78 00 | 03 02 | 31 04 | 35 06 | 01 16 | 05 18 | 33 20 |
0x00001b0:      37 22 | 07 08 | 11 10 | 39 12 | 43 14 | 09 24 | 13 26 | 41 28 |
0x00001d0:      45 30 | 62 47 | 66 49 | 15 51 | 19 53 | 64 63 | 68 65 | 17 67 |
0x00001f0:      21 69 | 70 55 | 74 57 | 23 59 | 53 22 | 35 12 | 37 28 | 67 14 |
0x0000210:      69 30 | 23 32 | 25 48 | 55 34 | 57 50 | 39 40 | 41 56 | 71 42 |
0x0000230:      73 58 | 27 36 | 29 52 | 59 38 | 61 54 | 43 44 | 45 60 | 75 46 |
0x0000250:      77 62 | 15 00 | 17 16 | 47 02 | 49 18 | 31 08 | 33 24 | 63 10 |
0x0000270:      65 26 | 19 04 | 21 20 | 51 06 | 06 54 | 65 42 | 69 58 | 18 46 |
0x0000290:      22 62 | 55 64 | 59 01 | 08 68 | 12 05 | 71 72 | 75 09 | 24 76 |
0x00002b0:      28 13 | 57 66 | 61 03 | 10 70 | 14 07 | 73 74 | 77 11 | 26 78 |
0x00002d0:      30 15 | 47 32 | 51 48 | 00 36 | 04 52 | 63 40 | 67 56 | 16 44 |
0x00002f0:      20 60 | 49 34 | 53 50 | 02 38 | 38 78 | 12 05 | 14 13 | 44 07 |
0x0000310:      46 15 | 16 17 | 18 25 | 48 19 | 50 27 | 24 33 | 26 41 | 56 35 |
0x0000330:      58 43 | 20 21 | 22 29 | 52 23 | 54 31 | 28 37 | 30 45 | 60 39 |
0x0000350:      62 47 | 00 64 | 02 72 | 32 66 | 34 74 | 08 01 | 10 09 | 40 03 |
0x0000370:      42 11 | 04 68 | 06 76 | 36 70 | 70 31 | 42 35 | 46 43 | 74 39 |
0x0000390:      78 47 | 48 49 | 52 57 | 01 53 | 05 61 | 56 65 | 60 73 | 09 69 |
0x00003b0:      13 77 | 50 51 | 54 59 | 03 55 | 07 63 | 58 67 | 62 75 | 11 71 |
0x00003d0:      15 00 | 32 17 | 36 25 | 64 21 | 68 29 | 40 33 | 44 41 | 72 37 |
0x00003f0:      76 45 | 34 19 | 38 27 | 66 23 | 11 71 | 05 18 | 07 22 | 13 20 |
```

2.2 THE 23-HOP SYSTEM SAMPLE DATA

2.2.1 First set

```
Set mode:
Set clock:
Set ULAP:

================================================================================
=
                                  23 HOP SYSTEM
================================================================================
=


Hop sequence {k} for PAGE SCAN/INQUIRY SCAN:
CLKN start:      0x0000000
ULAP:            0x00000000
#ticks:          0000 | 1000 | 2000 | 3000 | 4000 | 5000 | 6000 | 7000 |
                 ------------------------------------------------------
0x0000000:          0 |    2 |    4 |    6 |    8 |   10 |   12 |   14 |
0x0008000:         16 |   18 |   20 |   22 |    1 |    3 |    5 |    7 |
0x0010000:          0 |    2 |    4 |    6 |    8 |   10 |   12 |   14 |
0x0018000:         16 |   18 |   20 |   22 |    1 |    3 |    5 |    7 |
0x0020000:          0 |    2 |    4 |    6 |    8 |   10 |   12 |   14 |
0x0028000:         16 |   18 |   20 |   22 |    1 |    3 |    5 |    7 |
0x0030000:          0 |    2 |    4 |    6 |    8 |   10 |   12 |   14 |
0x0038000:         16 |   18 |   20 |   22 |    1 |    3 |    5 |    7 |

Hop sequence {k} for PAGE STATE/INQUIRY STATE:
CLKE start:      0x0000000
ULAP:            0x00000000
#ticks:          00 01 02 03 | 04 05 06 07 | 08 09 0a 0b | 0c 0d 0e 0f |
                 ------------------------------------------------------
0x0000000:       16 18 02 10 | 20 22 06 14 | 01 03 04 12 | 05 07 08 16 |
0x0000010:       00 02 09 17 | 04 06 13 21 | 08 10 11 19 | 12 14 15 00 |
0x0000020:       16 18 02 10 | 20 22 06 14 | 01 03 04 12 | 05 07 08 16 |
0x0000030:       00 02 09 17 | 04 06 13 21 | 08 10 11 19 | 12 14 15 00 |
...
0x0001000:       18 20 10 06 | 22 01 14 04 | 03 05 12 08 | 07 00 16 09 |
0x0001010:       02 04 17 13 | 06 08 21 11 | 10 12 19 15 | 14 16 00 02 |
0x0001020:       18 20 10 06 | 22 01 14 04 | 03 05 12 08 | 07 00 16 09 |
0x0001030:       02 04 17 13 | 06 08 21 11 | 10 12 19 15 | 14 16 00 02 |
...
0x0002000:       20 22 06 14 | 01 03 04 12 | 05 07 08 16 | 00 02 09 17 |
0x0002010:       04 06 13 21 | 08 10 11 19 | 12 14 15 00 | 16 18 02 10 |
0x0002020:       20 22 06 14 | 01 03 04 12 | 05 07 08 16 | 00 02 09 17 |
0x0002030:       04 06 13 21 | 08 10 11 19 | 12 14 15 00 | 16 18 02 10 |
...
0x0003000:       22 01 14 04 | 03 05 12 08 | 07 00 16 09 | 02 04 17 13 |
0x0003010:       06 08 21 11 | 10 12 19 15 | 14 16 00 02 | 18 20 10 06 |
0x0003020:       22 01 14 04 | 03 05 12 08 | 07 00 16 09 | 02 04 17 13 |
0x0003030:       06 08 21 11 | 10 12 19 15 | 14 16 00 02 | 18 20 10 06 |

Hop sequence {k} for SLAVE PAGE RESPONSE STATE:
CLKN* =          0x0000010
ULAP:            0x00000000
#ticks:          00 | 02 04 | 06 08 | 0a 0c | 0e 10 | 12 14 | 16 18 | 1a 1c | 1e
                 ----------------------------------------------------------------
0x0000012:       09 | 02 17 | 04 13 | 06 21 | 08 11 | 10 19 | 12 15 | 14 00 | 16
0x0000032:       02 | 18 10 | 20 06 | 22 14 | 01 04 | 03 12 | 05 08 | 07 16 | 00
0x0000052:       09 | 02 17 | 04 13 | 06 21 | 08 11 | 10 19 | 12 15 | 14 00 | 16
0x0000072:       02 | 18 10 | 20 06 | 22 14 | 01 04 | 03 12 | 05 08 | 07 16 | 00

Hop sequence {k} for MASTER PAGE RESPONSE STATE:
Offset value:    24
CLKE* =          0x0000012
ULAP:            0x00000000
#ticks:          00 02 | 04 06 | 08 0a | 0c 0e | 10 12 | 14 16 | 18 1a | 1c 1e |
                 ----------------------------------------------------------------
0x0000014:       02 17 | 04 13 | 06 21 | 08 11 | 10 19 | 12 15 | 14 00 | 16 02 |
0x0000034:       18 10 | 20 06 | 22 14 | 01 04 | 03 12 | 05 08 | 07 16 | 00 09 |
```

```
0x0000054:     02 17 | 04 13 | 06 21 | 08 11 | 10 19 | 12 15 | 14 00 | 16 02 |
0x0000074:     18 10 | 20 06 | 22 14 | 01 04 | 03 12 | 05 08 | 07 16 | 00 09 |

Hop sequence {k} for CONNECTION STATE:
CLK start:     0x0000010
ULAP:          0x00000000
#ticks:        00 02 | 04 06 | 08 0a | 0c 0e | 10 12 | 14 16 | 18 1a | 1c 1e |
               ----------------------------------------------------------------
0x0000010:     08 11 | 10 19 | 12 15 | 14 00 | 16 02 | 18 10 | 20 06 | 22 14 |
0x0000030:     01 04 | 03 12 | 05 08 | 07 16 | 16 02 | 18 10 | 20 06 | 22 14 |
0x0000050:     01 04 | 03 12 | 05 08 | 07 16 | 09 18 | 11 03 | 13 22 | 15 07 |
0x0000070:     17 20 | 19 05 | 21 01 | 00 09 | 09 18 | 13 03 | 11 20 | 15 05 |
0x0000090:     17 22 | 21 07 | 19 01 | 00 09 | 02 11 | 06 19 | 04 13 | 08 21 |
0x00000b0:     10 15 | 14 00 | 12 17 | 16 02 | 02 11 | 06 19 | 04 13 | 08 21 |
0x00000d0:     10 15 | 14 00 | 12 17 | 16 02 | 18 04 | 22 12 | 20 06 | 01 14 |
0x00000f0:     03 08 | 07 16 | 05 10 | 09 18 | 18 04 | 20 20 | 22 08 | 01 01 |
0x0000110:     11 06 | 13 22 | 15 10 | 17 03 | 03 12 | 05 05 | 07 16 | 09 09 |
0x0000130:     19 14 | 21 07 | 00 18 | 02 11 | 11 20 | 13 13 | 15 01 | 17 17 |
0x0000150:     04 22 | 06 15 | 08 03 | 10 19 | 19 05 | 21 21 | 00 09 | 02 02 |
0x0000170:     12 07 | 14 00 | 16 11 | 18 04 | 04 13 | 08 06 | 06 15 | 10 08 |
0x0000190:     20 17 | 01 10 | 22 19 | 03 12 | 12 21 | 16 14 | 14 00 | 18 16 |
0x00001b0:     05 02 | 09 18 | 07 04 | 11 20 | 20 06 | 01 22 | 22 08 | 03 01 |
0x00001d0:     13 10 | 17 03 | 15 12 | 19 05 | 05 14 | 09 07 | 07 16 | 11 09 |
0x00001f0:     21 18 | 02 11 | 00 20 | 04 13 | 13 22 | 06 07 | 17 03 | 10 11 |
0x0000210:     21 15 | 14 00 | 02 19 | 18 04 | 15 01 | 08 09 | 19 05 | 12 13 |
0x0000230:     00 17 | 16 02 | 04 21 | 20 06 | 06 15 | 22 00 | 10 19 | 03 04 |
0x0000250:     14 08 | 07 16 | 18 12 | 11 20 | 08 17 | 01 02 | 12 21 | 05 06 |
0x0000270:     16 10 | 09 18 | 20 14 | 13 22 | 22 08 | 15 16 | 01 10 | 17 18 |
0x0000290:     07 01 | 00 09 | 09 03 | 02 11 | 03 12 | 19 20 | 05 14 | 21 22 |
0x00002b0:     11 05 | 04 13 | 13 07 | 06 15 | 15 01 | 08 09 | 17 03 | 10 11 |
0x00002d0:     00 17 | 16 02 | 02 19 | 18 04 | 19 05 | 12 13 | 21 07 | 14 15 |
0x00002f0:     04 21 | 20 06 | 06 00 | 22 08 | 08 17 | 16 10 | 12 21 | 20 14 |
0x0000310:     01 02 | 09 18 | 05 06 | 13 22 | 10 19 | 18 12 | 14 00 | 22 16 |
0x0000330:     03 04 | 11 20 | 07 08 | 15 01 | 01 10 | 09 03 | 05 14 | 13 07 |
0x0000350:     17 18 | 02 11 | 21 22 | 06 15 | 03 12 | 11 05 | 07 16 | 15 09 |
0x0000370:     19 20 | 04 13 | 00 01 | 08 17 | 17 03 | 02 19 | 19 05 | 04 21 |
0x0000390:     10 11 | 18 04 | 12 13 | 20 06 | 21 07 | 06 00 | 00 09 | 08 02 |
0x00003b0:     14 15 | 22 08 | 16 17 | 01 10 | 10 19 | 18 12 | 12 21 | 20 14 |
0x00003d0:     03 04 | 11 20 | 05 06 | 13 22 | 14 00 | 22 16 | 16 02 | 01 18 |
0x00003f0:     07 08 | 15 01 | 09 10 | 17 03 | 03 12 | 05 16 | 11 20 | 13 01 |
```

2.2.2 Second set

```
Set mode:
Set clock:
Set ULAP:

===============================================================================

                                   23 HOP SYSTEM
===============================================================================


Hop sequence {k} for PAGE SCAN/INQUIRY SCAN:
CLKN start:     0x0000000
ULAP:           0x2a96ef25
#ticks:         0000 | 1000 | 2000 | 3000 | 4000 | 5000 | 6000 | 7000 |
                ---------------------------------------------------------
0x0000000:         7 |   18 |    3 |    8 |   16 |    4 |   12 |    1 |
0x0008000:         9 |   20 |    5 |    6 |   14 |    2 |   10 |   22 |
0x0010000:         7 |   18 |    3 |    8 |   16 |    4 |   12 |    1 |
0x0018000:         9 |   20 |    5 |    6 |   14 |    2 |   10 |   22 |
0x0020000:         7 |   18 |    3 |    8 |   16 |    4 |   12 |    1 |
0x0028000:         9 |   20 |    5 |    6 |   14 |    2 |   10 |   22 |
0x0030000:         7 |   18 |    3 |    8 |   16 |    4 |   12 |    1 |
0x0038000:         9 |   20 |    5 |    6 |   14 |    2 |   10 |   22 |

Hop sequence {k} for PAGE STATE/INQUIRY STATE:
CLKE start:     0x0000000
ULAP:           0x2a96ef25
#ticks:         00 01 02 03 | 04 05 06 07 | 08 09 0a 0b | 0c 0d 0e 0f |
                ---------------------------------------------------------
0x0000000:      09 20 18 00 | 05 06 02 04 | 14 02 06 11 | 10 22 13 12 |
0x0000010:      07 18 14 19 | 03 08 21 08 | 16 04 10 15 | 12 01 17 16 |
0x0000020:      09 20 18 00 | 05 06 02 04 | 14 02 06 11 | 10 22 13 12 |
0x0000030:      07 18 14 19 | 03 08 21 08 | 16 04 10 15 | 12 01 17 16 |
...
0x0001000:      20 05 00 02 | 06 14 04 06 | 02 10 11 13 | 22 07 12 14 |
0x0001010:      18 03 19 21 | 08 16 08 10 | 04 12 15 17 | 01 09 16 18 |
0x0001020:      20 05 00 02 | 06 14 04 06 | 02 10 11 13 | 22 07 12 14 |
0x0001030:      18 03 19 21 | 08 16 08 10 | 04 12 15 17 | 01 09 16 18 |
...
0x0002000:      05 06 02 04 | 14 02 06 11 | 10 22 13 12 | 07 18 14 19 |
0x0002010:      03 08 21 08 | 16 04 10 15 | 12 01 17 16 | 09 20 18 00 |
0x0002020:      05 06 02 04 | 14 02 06 11 | 10 22 13 12 | 07 18 14 19 |
0x0002030:      03 08 21 08 | 16 04 10 15 | 12 01 17 16 | 09 20 18 00 |
...
0x0003000:      06 14 04 06 | 02 10 11 13 | 22 07 12 14 | 18 03 19 21 |
0x0003010:      08 16 08 10 | 04 12 15 17 | 01 09 16 18 | 20 05 00 02 |
0x0003020:      06 14 04 06 | 02 10 11 13 | 22 07 12 14 | 18 03 19 21 |
0x0003030:      08 16 08 10 | 04 12 15 17 | 01 09 16 18 | 20 05 00 02 |

Hop sequence {k} for SLAVE PAGE RESPONSE STATE:
CLKN* =         0x0000010
ULAP:           0x2a96ef25
#ticks:         00 | 02 04 | 06 08 | 0a 0c | 0e 10 | 12 14 | 16 18 | 1a 1c | 1e
                ----------------------------------------------------------------
0x0000012:      14 | 18 19 | 03 21 | 08 08 | 16 10 | 04 15 | 12 17 | 01 16 | 09
0x0000032:      18 | 20 00 | 05 02 | 06 04 | 14 06 | 02 11 | 10 13 | 22 12 | 07
0x0000052:      14 | 18 19 | 03 21 | 08 08 | 16 10 | 04 15 | 12 17 | 01 16 | 09
0x0000072:      18 | 20 00 | 05 02 | 06 04 | 14 06 | 02 11 | 10 13 | 22 12 | 07

Hop sequence {k} for MASTER PAGE RESPONSE STATE:
Offset value:   24
CLKE* =         0x0000012
ULAP:           0x2a96ef25
#ticks:         00 02 | 04 06 | 08 0a | 0c 0e | 10 12 | 14 16 | 18 1a | 1c 1e |
                ----------------------------------------------------------------
0x0000014:      18 19 | 03 21 | 08 08 | 16 10 | 04 15 | 12 17 | 01 16 | 09 18 |
0x0000034:      20 00 | 05 02 | 06 04 | 14 06 | 02 11 | 10 13 | 22 12 | 07 14 |
```

```
0x0000054:    18 19 | 03 21 | 08 08 | 16 10 | 04 15 | 12 17 | 01 16 | 09 18 |
0x0000074:    20 00 | 05 02 | 06 04 | 14 06 | 02 11 | 10 13 | 22 12 | 07 14 |

Hop sequence {k} for CONNECTION STATE:
CLK start:    0x0000010
ULAP:         0x2a96ef25
#ticks:       00 02 | 04 06 | 08 0a | 0c 0e | 10 12 | 14 16 | 18 1a | 1c 1e |
              ----------------------------------------------------------------
0x0000010:    16 10 | 04 15 | 12 17 | 01 16 | 09 18 | 20 00 | 05 02 | 06 04 |
0x0000030:    14 06 | 02 11 | 10 13 | 22 12 | 00 07 | 11 12 | 19 14 | 01 01 |
0x0000050:    09 03 | 20 08 | 05 10 | 17 09 | 02 11 | 13 16 | 21 18 | 22 20 |
0x0000070:    07 22 | 18 04 | 03 06 | 15 05 | 14 02 | 04 05 | 12 09 | 17 15 |
0x0000090:    02 19 | 15 22 | 00 03 | 10 00 | 18 04 | 08 07 | 16 11 | 13 13 |
0x00000b0:    21 17 | 11 20 | 19 01 | 06 21 | 07 18 | 20 21 | 05 02 | 10 08 |
0x00000d0:    18 12 | 08 15 | 16 19 | 03 16 | 11 20 | 01 00 | 09 04 | 06 06 |
0x00000f0:    14 10 | 04 13 | 12 17 | 22 14 | 02 09 | 05 22 | 21 01 | 03 18 |
0x0000110:    19 20 | 22 10 | 15 12 | 11 11 | 04 13 | 07 03 | 00 05 | 01 14 |
0x0000130:    17 16 | 20 06 | 13 08 | 09 07 | 18 02 | 21 15 | 14 17 | 19 11 |
0x0000150:    12 13 | 15 03 | 08 05 | 04 04 | 20 06 | 00 19 | 16 21 | 17 07 |
0x0000170:    10 09 | 13 22 | 06 01 | 02 00 | 09 20 | 14 08 | 07 12 | 12 02 |
0x0000190:    05 06 | 10 17 | 03 21 | 20 18 | 13 22 | 18 10 | 11 14 | 08 00 |
0x00001b0:    01 04 | 06 15 | 22 19 | 16 16 | 02 13 | 07 01 | 00 05 | 05 18 |
0x00001d0:    21 22 | 03 10 | 19 14 | 13 11 | 06 15 | 11 03 | 04 07 | 01 16 |
0x00001f0:    17 20 | 22 08 | 15 12 | 09 09 | 20 00 | 08 05 | 12 07 | 02 02 |
0x0000210:    06 04 | 17 09 | 21 11 | 18 06 | 22 08 | 10 13 | 14 15 | 00 17 |
0x0000230:    04 19 | 15 01 | 19 03 | 16 21 | 13 16 | 01 21 | 05 00 | 18 18 |
0x0000250:    22 20 | 10 02 | 14 04 | 11 22 | 15 01 | 03 06 | 07 08 | 16 10 |
0x0000270:    20 12 | 08 17 | 12 19 | 09 14 | 04 09 | 17 12 | 19 16 | 13 11 |
0x0000290:    15 15 | 05 18 | 07 22 | 06 13 | 08 17 | 21 20 | 00 01 | 09 03 |
0x00002b0:    11 07 | 01 10 | 03 14 | 02 05 | 20 02 | 10 05 | 12 09 | 06 04 |
0x00002d0:    08 08 | 21 11 | 00 15 | 22 06 | 01 10 | 14 13 | 16 17 | 02 19 |
0x00002f0:    04 00 | 17 03 | 19 07 | 18 21 | 15 10 | 18 00 | 22 02 | 05 20 |
0x0000310:    09 22 | 12 12 | 16 14 | 13 01 | 17 03 | 20 16 | 01 18 | 03 04 |
0x0000330:    07 06 | 10 19 | 14 21 | 11 08 | 08 03 | 11 16 | 15 18 | 21 13 |
0x0000350:    02 15 | 05 05 | 09 07 | 06 17 | 10 19 | 13 09 | 17 11 | 19 20 |
0x0000370:    00 22 | 03 12 | 07 14 | 04 01 | 22 19 | 04 07 | 06 11 | 16 06 |
0x0000390:    18 10 | 00 21 | 02 02 | 01 08 | 03 12 | 08 00 | 10 04 | 12 13 |
0x00003b0:    14 17 | 19 05 | 21 09 | 20 15 | 15 12 | 20 00 | 22 04 | 09 22 |
0x00003d0:    11 03 | 16 14 | 18 18 | 17 01 | 19 05 | 01 16 | 03 20 | 05 06 |
0x00003f0:    07 10 | 12 21 | 14 02 | 13 08 | 19 17 | 07 22 | 15 15 | 02 20 |
```

2.2.3 Third set

```
Set mode:
Set clock:
Set ULAP:

================================================================================
                                23 HOP SYSTEM
================================================================================

Hop sequence {k} for PAGE SCAN/INQUIRY SCAN:
CLKN start:     0x0000000
ULAP:           0x6587cba9
#ticks:         0000 | 1000 | 2000 | 3000 | 4000 | 5000 | 6000 | 7000 |
                --------------------------------------------------------
0x0000000:         0 |   10 |   14 |   12 |   16 |    9 |   13 |   11 |
0x0008000:        15 |    2 |    6 |    4 |    8 |   17 |   21 |   19 |
0x0010000:         0 |   10 |   14 |   12 |   16 |    9 |   13 |   11 |
0x0018000:        15 |    2 |    6 |    4 |    8 |   17 |   21 |   19 |
0x0020000:         0 |   10 |   14 |   12 |   16 |    9 |   13 |   11 |
0x0028000:        15 |    2 |    6 |    4 |    8 |   17 |   21 |   19 |
0x0030000:         0 |   10 |   14 |   12 |   16 |    9 |   13 |   11 |
0x0038000:        15 |    2 |    6 |    4 |    8 |   17 |   21 |   19 |

Hop sequence {k} for PAGE STATE/INQUIRY STATE:
CLKE start:     0x0000000
ULAP:           0x6587cba9
#ticks:         00 01 02 03 | 04 05 06 07 | 08 09 0a 0b | 0c 0d 0e 0f |
                --------------------------------------------------------
0x0000000:      15 02 05 11 | 06 04 19 13 | 08 17 21 22 | 21 19 07 01 |
0x0000010:      00 10 09 15 | 14 12 00 17 | 16 09 02 18 | 13 11 03 20 |
0x0000020:      15 02 05 11 | 06 04 19 13 | 08 17 21 22 | 21 19 07 01 |
0x0000030:      00 10 09 15 | 14 12 00 17 | 16 09 02 18 | 13 11 03 20 |
...
0x0001000:      02 06 11 19 | 04 08 13 21 | 17 21 22 07 | 19 00 01 09 |
0x0001010:      10 14 15 00 | 12 16 17 02 | 09 13 18 03 | 11 15 20 05 |
0x0001020:      02 06 11 19 | 04 08 13 21 | 17 21 22 07 | 19 00 01 09 |
0x0001030:      10 14 15 00 | 12 16 17 02 | 09 13 18 03 | 11 15 20 05 |
...
0x0002000:      06 04 19 13 | 08 17 21 22 | 21 19 07 01 | 00 10 09 15 |
0x0002010:      14 12 00 17 | 16 09 02 18 | 13 11 03 20 | 15 02 05 11 |
0x0002020:      06 04 19 13 | 08 17 21 22 | 21 19 07 01 | 00 10 09 15 |
0x0002030:      14 12 00 17 | 16 09 02 18 | 13 11 03 20 | 15 02 05 11 |
...
0x0003000:      04 08 13 21 | 17 21 22 07 | 19 00 01 09 | 10 14 15 00 |
0x0003010:      12 16 17 02 | 09 13 18 03 | 11 15 20 05 | 02 06 11 19 |
0x0003020:      04 08 13 21 | 17 21 22 07 | 19 00 01 09 | 10 14 15 00 |
0x0003030:      12 16 17 02 | 09 13 18 03 | 11 15 20 05 | 02 06 11 19 |

Hop sequence {k} for SLAVE PAGE RESPONSE STATE:
CLKN* =         0x0000010
ULAP:           0x6587cba9
#ticks:         00 | 02 04 | 06 08 | 0a 0c | 0e 10 | 12 14 | 16 18 | 1a 1c | 1e
                ----------------------------------------------------------------
0x0000012:      09 | 10 15 | 14 00 | 12 17 | 16 02 | 09 18 | 13 03 | 11 20 | 15
0x0000032:      05 | 02 11 | 06 19 | 04 13 | 08 21 | 17 22 | 21 07 | 19 01 | 00
0x0000052:      09 | 10 15 | 14 00 | 12 17 | 16 02 | 09 18 | 13 03 | 11 20 | 15
0x0000072:      05 | 02 11 | 06 19 | 04 13 | 08 21 | 17 22 | 21 07 | 19 01 | 00

Hop sequence {k} for MASTER PAGE RESPONSE STATE:
Offset value:   24
CLKE* =         0x0000012
ULAP:           0x6587cba9
#ticks:         00 02 | 04 06 | 08 0a | 0c 0e | 10 12 | 14 16 | 18 1a | 1c 1e |
                ----------------------------------------------------------------
0x0000014:      10 15 | 14 00 | 12 17 | 16 02 | 09 18 | 13 03 | 11 20 | 15 05 |
0x0000034:      02 11 | 06 19 | 04 13 | 08 21 | 17 22 | 21 07 | 19 01 | 00 09 |
```

```
0x0000054:    10 15 | 14 00 | 12 17 | 16 02 | 09 18 | 13 03 | 11 20 | 15 05 |
0x0000074:    02 11 | 06 19 | 04 13 | 08 21 | 17 22 | 21 07 | 19 01 | 00 09 |

Hop sequence {k} for CONNECTION STATE:
CLK start:    0x0000010
ULAP:         0x6587cba9
#ticks:       00 02 | 04 06 | 08 0a | 0c 0e | 10 12 | 14 16 | 18 1a | 1c 1e |
              ----------------------------------------------------------------
0x0000010:    16 02 | 09 18 | 13 03 | 11 20 | 15 05 | 02 11 | 06 19 | 04 13 |
0x0000030:    08 21 | 17 22 | 21 07 | 19 01 | 16 02 | 03 08 | 07 16 | 05 10 |
0x0000050:    09 18 | 02 11 | 06 19 | 04 13 | 08 21 | 18 04 | 22 12 | 20 06 |
0x0000070:    01 14 | 10 15 | 14 00 | 12 17 | 09 18 | 19 22 | 21 07 | 00 03 |
0x0000090:    02 11 | 18 04 | 20 12 | 22 08 | 01 16 | 11 20 | 13 05 | 15 01 |
0x00000b0:    17 09 | 03 06 | 05 14 | 07 10 | 02 11 | 12 15 | 14 00 | 16 19 |
0x00000d0:    18 04 | 11 20 | 13 05 | 15 01 | 17 09 | 04 13 | 06 21 | 08 17 |
0x00000f0:    10 02 | 19 22 | 21 07 | 00 03 | 03 12 | 05 02 | 09 18 | 07 04 |
0x0000110:    11 20 | 04 13 | 08 06 | 06 15 | 10 08 | 12 21 | 16 14 | 14 00 |
0x0000130:    18 16 | 20 17 | 01 10 | 22 19 | 19 05 | 21 18 | 02 11 | 00 20 |
0x0000150:    04 13 | 20 06 | 01 22 | 22 08 | 03 01 | 05 14 | 09 07 | 07 16 |
0x0000170:    11 09 | 13 10 | 17 03 | 15 12 | 12 21 | 14 09 | 16 02 | 18 13 |
0x0000190:    20 06 | 13 22 | 15 15 | 17 03 | 19 19 | 21 07 | 00 00 | 02 11 |
0x00001b0:    04 04 | 06 01 | 08 17 | 10 05 | 05 14 | 07 02 | 09 18 | 11 06 |
0x00001d0:    13 22 | 06 15 | 08 08 | 10 19 | 12 12 | 14 00 | 16 16 | 18 04 |
0x00001f0:    20 20 | 22 17 | 01 10 | 03 21 | 13 22 | 17 05 | 21 07 | 02 13 |
0x0000210:    06 15 | 22 08 | 03 10 | 07 16 | 11 18 | 15 01 | 19 03 | 00 09 |
0x0000230:    04 11 | 01 12 | 05 14 | 09 20 | 06 15 | 10 21 | 14 00 | 18 06 |
0x0000250:    22 08 | 15 01 | 19 03 | 00 09 | 04 11 | 08 17 | 12 19 | 16 02 |
0x0000270:    20 04 | 17 05 | 21 07 | 02 13 | 22 08 | 05 12 | 07 16 | 13 20 |
0x0000290:    15 01 | 08 17 | 10 21 | 16 02 | 18 06 | 01 10 | 03 14 | 09 18 |
0x00002b0:    11 22 | 12 19 | 14 00 | 20 04 | 15 01 | 21 05 | 00 09 | 06 13 |
0x00002d0:    08 17 | 01 10 | 03 14 | 09 18 | 11 22 | 17 03 | 19 07 | 02 11 |
0x00002f0:    04 15 | 05 12 | 07 16 | 13 20 | 16 02 | 04 15 | 08 17 | 20 08 |
0x0000310:    01 10 | 17 03 | 21 05 | 10 19 | 14 21 | 02 11 | 06 13 | 18 04 |
0x0000330:    22 06 | 19 07 | 00 09 | 12 00 | 09 18 | 20 08 | 01 10 | 13 01 |
0x0000350:    17 03 | 10 19 | 14 21 | 03 12 | 07 14 | 18 04 | 22 06 | 11 20 |
0x0000370:    15 22 | 12 00 | 16 02 | 05 16 | 02 11 | 15 22 | 17 03 | 08 15 |
0x0000390:    10 19 | 03 12 | 05 16 | 19 05 | 21 09 | 11 20 | 13 01 | 04 13 |
0x00003b0:    06 17 | 07 14 | 09 18 | 00 07 | 18 04 | 08 15 | 10 19 | 01 08 |
0x00003d0:    03 12 | 19 05 | 21 09 | 12 21 | 14 02 | 04 13 | 06 17 | 20 06 |
0x00003f0:    22 10 | 00 07 | 02 11 | 16 00 | 15 01 | 01 18 | 17 03 | 03 20 |
```

3 ACCESS CODE SAMPLE DATA

Different access codes (GIAC, DIACs, others...)

```
LAP with LSB as rightmost bit.

                 Bit transmit order on air
              --------------------------------->
LAP:        | Preamble:| Sync word:        | Trailer: |
------------------------------------------------------
000000      |    5     | 7e7041e3 4000000d |    5     |
ffffff      |    a     | e758b522 7ffffff2 |    a     |
9e8b33      |    5     | 475c58cc 73345e72 |    a     |
9e8b34      |    5     | 28ed3c34 cb345e72 |    a     |
9e8b36      |    5     | 62337b64 1b345e72 |    a     |
9e8b39      |    a     | c05747b9 e7345e72 |    a     |
9e8b3d      |    5     | 7084eab0 2f345e72 |    a     |
9e8b42      |    5     | 64c86d2b 90b45e72 |    a     |
9e8b48      |    a     | e3c3725e 04b45e72 |    a     |
9e8b4f      |    a     | 8c7216a6 bcb45e72 |    a     |
9e8b57      |    a     | b2f16c30 fab45e72 |    a     |
9e8b60      |    5     | 57bd3b22 c1b45e72 |    a     |
9e8b6a      |    a     | d0b62457 55b45e72 |    a     |
9e8b75      |    a     | 81843a39 abb45e72 |    a     |
9e8b81      |    5     | 0ca96681 e0745e72 |    a     |
9e8b8e      |    a     | aecd5a5c 1c745e72 |    a     |
9e8b9c      |    5     | 17453fbf ce745e72 |    a     |
9e8bab      |    a     | f20968ad f5745e72 |    a     |
9e8bbb      |    5     | 015f4a1e f7745e72 |    a     |
9e8bcc      |    a     | d8c695a0 0cf45e72 |    a     |
9e8bde      |    5     | 614ef043 def45e72 |    a     |
9e8bf1      |    a     | ba81ddc7 a3f45e72 |    a     |
9e8c05      |    5     | 64a7dc4f 680c5e72 |    a     |
9e8c1a      |    5     | 3595c221 960c5e72 |    a     |
9e8c30      |    a     | cb35cc0d 830c5e72 |    a     |
9e8c47      |    5     | 12ac13b3 788c5e72 |    a     |
9e8c5f      |    5     | 2c2f6925 3e8c5e72 |    a     |
9e8c78      |    5     | 3a351c84 078c5e72 |    a     |
9e8c92      |    5     | 7396d0f3 124c5e72 |    a     |
9e8cad      |    5     | 5b0fdfc4 6d4c5e72 |    a     |
9e8cc9      |    a     | aea2eb38 e4cc5e72 |    a     |
9e8ce6      |    5     | 756dc6bc 99cc5e72 |    a     |
9e8d04      |    5     | 214cf934 882c5e72 |    a     |
9e8d23      |    5     | 37568c95 b12c5e72 |    a     |
9e8d43      |    5     | 72281560 f0ac5e72 |    a     |
9e8d64      |    5     | 643260c1 c9ac5e72 |    a     |
9e8d86      |    a     | e044f493 986c5e72 |    a     |
9e8da9      |    5     | 3b8bd917 e56c5e72 |    a     |
9e8dcd      |    a     | ce26edeb 6cec5e72 |    a     |
9e8df2      |    a     | e6bfe2dc 13ec5e72 |    a     |
9e8e18      |    a     | 82dcde3d c61c5e72 |    a     |
9e8e3f      |    a     | 94c6ab9c ff1c5e72 |    a     |
```

9e8e67	a	969059a6	799c5e72	a
9e8e90	a	c4dfccef	425c5e72	a
9e8eba	5	3a7fc2c3	575c5e72	a
9e8ee5	5	57985401	69dc5e72	a
9e8f11	5	0ae2a363	623c5e72	a
9e8f3e	a	d12d8ee7	1f3c5e72	a
9e8f6c	5	547063a8	0dbc5e72	a
9e8f9b	5	063ff6e1	367c5e72	a
9e8fcb	a	c9bc5cfe	f4fc5e72	a
9e8ffc	5	2cf00bec	cffc5e72	a
9e902e	a	8ec5052f	5d025e72	a
9e9061	5	1074b15e	61825e72	a
9e9095	a	9d59ede6	2a425e72	a
9e90ca	a	f0be7b24	14c25e72	a
9e9100	5	10e10dd0	c0225e72	a
9e9137	a	f5ad5ac2	fb225e72	a
9e916f	a	f7fba8f8	7da25e72	a
9e91a8	5	2f490e5b	c5625e72	a
9e91e2	a	94979982	91e25e72	a
9e921d	5	26cda478	2e125e72	a
9e9259	a	aacb81dd	26925e72	a
9e9296	a	bfac7f5b	da525e72	a
9e92d4	a	c9a7b0a7	cad25e72	a
9e9313	a	c142bdde	32325e72	a
616cec	5	586a491f	0dcda18d	5
616ceb	5	37db2de7	b5cda18d	5
616ce9	5	7d056ab7	65cda18d	5
616ce6	a	df61566a	99cda18d	5
616ce2	5	6fb2fb63	51cda18d	5
616cdd	5	472bf454	2ecda18d	5
616cd7	a	c020eb21	bacda18d	5
616cd0	a	af918fd9	02cda18d	5
616cc8	a	9112f54f	44cda18d	5
616cbf	5	488b2af1	bf4da18d	5
616cb5	a	cf803584	2b4da18d	5
616caa	a	9eb22bea	d54da18d	5
616c9e	a	a49cb509	9e4da18d	5
616c91	5	06f889d4	624da18d	5
616c83	a	bf70ec37	b04da18d	5
616c74	a	ed3f797e	8b8da18d	5
616c64	5	1e695bcd	898da18d	5
616c53	a	fb250cdf	b28da18d	5
616c41	5	42ad693c	608da18d	5
616c2e	a	a5b7cc14	dd0da18d	5
616c1a	a	9f9952f7	960da18d	5
616c05	a	ceab4c99	680da18d	5
616bef	a	d403ddde	fdf5a18d	5
616bd8	5	314f8acc	c6f5a18d	5
616bc0	5	0fccf05a	80f5a18d	5
616ba7	5	25030d57	7975a18d	5
616b8d	a	dba3037b	6c75a18d	5
616b72	5	4439ce17	13b5a18d	5

```
616b56       |   a   | 8d417247 5ab5a18d |    5    |
616b39       |   5   | 6a5bd76f e735a18d |    5    |
616b1b       |   5   | 592e8166 b635a18d |    5    |
616afc       |   5   | 28609d46 cfd5a18d |    5    |
616adc       |   5   | 51cb8c1f 4ed5a18d |    5    |
616abb       |   5   | 7b047112 b755a18d |    5    |
616a99       |   5   | 4871271b e655a18d |    5    |
616a76       |   5   | 24bdc8c4 9b95a18d |    5    |
616a52       |   a   | edc57494 d295a18d |    5    |
616a2d       |   a   | f989f30f 6d15a18d |    5    |
616a07       |   5   | 0729fd23 7815a18d |    5    |
6169e0       |   a   | 8bf0ba4f 81e5a18d |    5    |
6169b8       |   a   | 89a64875 0765a18d |    5    |
61698f       |   5   | 6cea1f67 3c65a18d |    5    |
616965       |   5   | 2549d310 29a5a18d |    5    |
61693a       |   5   | 48ae45d2 1725a18d |    5    |
61690e       |   5   | 7280db31 5c25a18d |    5    |
6168e1       |   a   | ce1b9f34 61c5a18d |    5    |
6168b3       |   5   | 4b46727b 7345a18d |    5    |
616884       |   a   | ae0a2569 4845a18d |    5    |
616854       |   a   | ea5fc581 4a85a18d |    5    |
616823       |   5   | 33c61a3f b105a18d |    5    |
6167f1       |   a   | c49fb8c5 63f9a18d |    5    |
6167be       |   5   | 5a2e0cb4 5f79a18d |    5    |
61678a       |   5   | 60009257 1479a18d |    5    |
616755       |   a   | 86314e62 eab9a18d |    5    |
61671f       |   5   | 3defd9bb be39a18d |    5    |
6166e8       |   a   | bff7e728 c5d9a18d |    5    |
6166b0       |   a   | bda11512 4359a18d |    5    |
616677       |   5   | 6513b3b1 fb99a18d |    5    |
61663d       |   a   | decd2468 af19a18d |    5    |
616602       |   a   | f6542b5f d019a18d |    5    |
6165c6       |   a   | dc44b49b d8e9a18d |    5    |
616589       |   5   | 42f500ea e469a18d |    5    |
61654b       |   a   | bf2885e1 34a9a18d |    5    |
61650c       |   a   | ec4c69b5 4c29a18d |    5    |
```

4 HEC AND PACKET HEADER SAMPLE DATA

Test vectors for HEC and header. Note that UAP, Data, and HEC are in hexadecimal notation, while the header is in octal notation. The header is transmitted from left to right over air.

UAP	Data	HEC	Header (octal not.)
00	123	e1	770007 007070 000777
47	123	06	770007 007007 700000
00	124	32	007007 007007 007700
47	124	d5	007007 007070 707077
00	125	5a	707007 007007 077070
47	125	bd	707007 007070 777707
00	126	e2	077007 007007 000777
47	126	05	077007 007070 700000
00	127	8a	777007 007007 070007
47	127	6d	777007 007070 770770
00	11b	9e	770770 007007 777007
47	11b	79	770770 007070 077770
00	11c	4d	007770 007070 770070
47	11c	aa	007770 007007 070707
00	11d	25	707770 007070 700700
47	11d	c2	707770 007007 000077
00	11e	9d	077770 007070 777007
47	11e	7a	077770 007007 077770
00	11f	f5	777770 007070 707777
47	11f	12	777770 007007 007000

5 CRC SAMPLE DATA

```
Sample CRC generation


Data:
-----

data[0] = 0x4e
data[1] = 0x01
data[2] = 0x02
data[3] = 0x03
data[4] = 0x04
data[5] = 0x05
data[6] = 0x06
data[7] = 0x07
data[8] = 0x08
data[9] = 0x09

UAP = 0x47


==> CRC = 6d d2

Codeword (hexadecimal notation):
---------------------------------

4e 01 02 03 04 05 06 07 08 09 6d d2


NB: Over air each byte in the codeword
    is sent with the LSB first.
```

6 COMPLETE SAMPLE PACKETS

6.1 EXAMPLE OF DH1 PACKET

```
Packet header: (MSB...LSB)
--------------------------
AM_ADDR = 011
TYPE = 0100 (DH1)
FLOW = 0
ARQN = 1
SEQN = 0

Payload: (MSB...LSB)
--------------------
payload length: 5 bytes
logical channel = 10 (UA/I, Start L2CAP message)
flow = 1
data byte 1 = 00000001
data byte 2 = 00000010
data byte 3 = 00000011
data byte 4 = 00000100
data byte 5 = 00000101

HEC and CRC initialization: (MSB...LSB)
---------------------------------------
uap = 01000111

NO WHITENING USED




AIR DATA (LSB...MSB)

Packet header (incl HEC):
-------------------------
111111000
000000111000
000111000
000111111000000000000000

Payload (incl payload header and CRC):
-------------------------------------------------------------
01110100
10000000
01000000
11000000
00100000
10100000
1110110000110110
```

6.2 EXAMPLE OF DM1 PACKET

```
Packet header: (MSB...LSB)
--------------------------
AM_ADDR = 011
TYPE = 0011 (DM1)
FLOW = 0
ARQN = 1
SEQN = 0

Payload: (MSB...LSB)
--------------------
payload length: 5 bytes
logical channel = 10 (UA/I, Start L2CAP message)
flow = 1
data byte 1 = 00000001
data byte 2 = 00000010
data byte 3 = 00000011
data byte 4 = 00000100
data byte 5 = 00000101

HEC and CRC initialization: (MSB...LSB)
---------------------------------------
uap = 01000111

NO WHITENING USED


AIR DATA (LSB...MSB)

Packet header (incl HEC):
-------------------------
111111000
111111000000
000111000
111000000111111111111000

Payload (incl payload header, FEC23, CRC and 6 padded zeros):
-------------------------------------------------------------
0111010010 11001
0000000100 01011
0000110000 11110
0000100000 00111
1010000011 01100
1011000011 00010
0110000000 10001
```

7 WHITENING SEQUENCE SAMPLE DATA

```
Whitening sequence generator.


   Whitening  Whitening
   Sequence     LFSR
     (=D7)     D7.....D0
-----------------------
        1      1111111
        1      1101111
        1      1001111
        0      0001111
        0      0011110
        0      0111100
        1      1111000
        1      1100001
        1      1010011
        0      0110111
        1      1101110
        1      1001101
        0      0001011
        0      0010110
        0      0101100
        1      1011000
        0      0100001
        1      1000010
        0      0010101
        0      0101010
        1      1010100
        0      0111001
        1      1110010
        1      1110101
        1      1111011
        1      1100111
        1      1011111
        0      0101111
        1      1011110
        0      0101101
        1      1011010
        0      0100101
        1      1001010
        0      0000101
        0      0001010
        0      0010100
        0      0101000
        1      1010000
        0      0110001
        1      1100010
        1      1010101
        0      0111011
        1      1110110
```

1	1111101
1	1101011
1	1000111
0	0011111
0	0111110
1	1111100
1	1101001
1	1000011
0	0010111
0	0101110
1	1011100
0	0101001
1	1010010
0	0110101
1	1101010
1	1000101
0	0011011
0	0110110
1	1101100
1	1001001
0	0000011
0	0000110
0	0001100
0	0011000
0	0110000
1	1100000
1	1010001
0	0110011
1	1100110
1	1011101
0	0101011
1	1010110
0	0111101
1	1111010
1	1100101
1	1011011
0	0100111
1	1001110
0	0001101
0	0011010
0	0110100
1	1101000
1	1000001
0	0010011
0	0100110
1	1001100
0	0001001
0	0010010
0	0100100
1	1001000
0	0000001
0	0000010

0	0000100
0	0001000
0	0010000
0	0100000
1	1000000
0	0010001
0	0100010
1	1000100
0	0011001
0	0110010
1	1100100
1	1011001
0	0100011
1	1000110
0	0011101
0	0111010
1	1110100
1	1111001
1	1100011
1	1010111
0	0111111
1	1111110
1	1101101
1	1001011
0	0000111
0	0001110
0	0011100
0	0111000
1	1110000
1	1110001
1	1110011
1	1110111
1	1111111

8 FEC SAMPLE DATA

```
=================================================
 Rate 2/3 FEC -- (15,10) Shortened Hamming Code
=================================================

Data is in hexadecimal notation, the codewords are in binary notation. The codeword bits
are sent from left to right over the air interface. The space in the codeword indicates
the start of parity bits.

Data:           Codeword:

0x001       1000000000 11010
0x002       0100000000 01101
0x004       0010000000 11100
0x008       0001000000 01110
0x010       0000100000 00111
0x020       0000010000 11001
0x040       0000001000 10110
0x080       0000000100 01011
0x100       0000000010 11111
0x200       0000000001 10101
```

9 ENCRYPTION KEY SAMPLE DATA

Explanation:

Key [i]: denotes the ith sub-key in Ar or A'r;
round r: denotes the input to the rth round;
added ->: denotes the input to round 3 in A'r after adding original input (of round 1).

9.1 FOUR TESTS OF E1

```
rand     :00000000000000000000000000000000
address :000000000000
key      :00000000000000000000000000000000
round  1:00000000000000000000000000000000
Key [ 1]:00000000000000000000000000000000
Key [ 2]:4697b1baa3b7100ac537b3c95a28ac64
round  2:78d19f9307d2476a523ec7a8a026042a
Key [ 3]:ecabaac66795580df89af66e66dc053d
Key [ 4]:8ac3d8896ae9364943bfebd4969b68a0
round  3:600265247668dda0e81c07bbb30ed503
Key [ 5]:5d57921fd5715cbb22c1be7bbc996394
Key [ 6]:2a61b8343219fdfb1740e6511d41448f
round  4:d7552ef7cc9dbde568d80c2215bc4277
Key [ 7]:dd0480dee731d67f01a2f739da6f23ca
Key [ 8]:3ad01cd1303e12a1cd0fe0a8af82592c
round  5:fb06bef32b52ab8f2a4f2b6ef7f6d0cd
Key [ 9]:7dadb2efc287ce75061302904f2e7233
Key [10]:c08dcfa981e2c4272f6c7a9f52e11538
round  6:b46b711ebb3cf69e847a75f0ab884bdd
Key [11]:fc2042c708e409555e8c147660ffdfd7
Key [12]:fa0b21001af9a6b9e89e624cd99150d2
round  7:c585f308ff19404294f06b292e978994
Key [13]:18b40784ea5ba4c80ecb48694b4e9c35
Key [14]:454d54e5253c0c4a8b3fcca7db6baef4
round  8:2665fadb13acf952bf74b4ab12264b9f
Key [15]:2df37c6d9db52674f29353b0f011ed83
Key [16]:b60316733b1e8e70bd861b477e2456f1
Key [17]:884697b1baa3b7100ac537b3c95a28ac
round  1:158ffe43352085e8a5ec7a88e1ff2ba8
Key [ 1]:e9e5dfc1b3a79583e9e5dfc1b3a79583
Key [ 2]:7595bf57e0632c59f435c16697d4c864
round  2:0b5cc75febcdf7827ca29ec0901b6b5b
Key [ 3]:e31b96afcc75d286ef0ae257cbbc05b7
Key [ 4]:0d2a27b471bc0108c6263aff9d9b3b6b
round  3:e4278526c8429211f7f2f0016220aef4
added ->:f1b68365fd6217f952de6a89831fd95c
Key [ 5]:98d1eb5773cf59d75d3b17b3bc37c191
Key [ 6]:fd2b79282408ddd4ea0aa7511133336f
round  4:d0304ad18337f86040145d27aa5c8153
Key [ 7]:331227756638a41d57b0f7e071ee2a98
```

```
Key [ 8]:aa0dd8cc68b406533d0f1d64aabacf20
round  5:84db909d213bb0172b8b6aaf71bf1472
Key [ 9]:669291b0752e63f806fce76f10e119c8
Key [10]:ef8bdd46be8ee0277e9b78adef1ec154
round  6:f835f52921e903dfa762f1df5abd7f95
Key [11]:f3902eb06dc409cfd78384624964bf51
Key [12]:7d72702b21f97984a721c99b0498239d
round  7:ae6c0b4bb09f25c6a5d9788a31b605d1
Key [13]:532e60bceaf902c52a06c2c283ecfa32
Key [14]:181715e5192efb2a64129668cf5d9dd4
round  8:744a6235b86cc0b853cc9f74f6b65311
Key [15]:83017c1434342d4290e961578790f451
Key [16]:2603532f365604646ff65803795ccce5
Key [17]:882f7c907b565ea58dae1c928a0dcf41
sres    :056c0fe6
aco     :48afcdd4bd40fef76693b113
-----------------------------------------
rand    :bc3f30689647c8d7c5a03ca80a91eceb
address :7ca89b233c2d
key     :159dd9f43fc3d328efba0cd8a861fa57
round  1:bc3f30689647c8d7c5a03ca80a91eceb
Key [ 1]:159dd9f43fc3d328efba0cd8a861fa57
Key [ 2]:326558b3c15551899a97790e65ff669e
round  2:3e950edf197615638cc19c09f8fedc9b
Key [ 3]:62e879b65b9f53bbfbd020c624b1d682
Key [ 4]:73415f30bac8ab61f410adc9442992db
round  3:6a7640791cb536678936c5ecd4ae5a73
Key [ 5]:5093cfa1d31c1c48acd76df030ea3c31
Key [ 6]:0b4acc2b8f1f694fc7bd91f4a70f3009
round  4:fca2c022a577e2ffb2aa007589693ec7
Key [ 7]:2ca43fc817947804ecff148d50d6f6c6
Key [ 8]:3fcd73524b533e00b7f7825bea2040a4
round  5:e97f8ea4ed1a6f4a36ffc179dc6bb563
Key [ 9]:6c67bec76ae8c8cc4d289f69436d3506
Key [10]:95ed95ee8cb97e61d75848464bffb379
round  6:38b07261d7340d028749de1773a415c7
Key [11]:ff566c1fc6b9da9ac502514550f3e9d2
Key [12]:ab5ce3f5c887d0f49b87e0d380e12f47
round  7:58241f1aed7c1c3e047d724331a0b774
Key [13]:a2cab6f95eac7d655dbe84a6cd4c47f5
Key [14]:f5caff88af0af8c42a20b5bbd2c8b460
round  8:3d1aaeff53c0910de63b9788b13c490f
Key [15]:185099c1131cf97001e2f36fda415025
Key [16]:a0ebb82676bc75e8378b189eff3f6b1d
Key [17]:cf5b348aaee27ae332b4f1bfa10289a6
round  1:2e4b417b9a2a9cfd7d8417d9a6a556eb
Key [ 1]:fe78b835f26468ab069fd3991b086fda
Key [ 2]:095c5a51c6fa6d3ac1d57fa19aa382bd
round  2:b8bca81d6bb45af9d92beadd9300f5ed
Key [ 3]:1af866df817fd9f4ec00bc704192cffc
Key [ 4]:f4a8a059c1f575f076f5fbb24bf16590
round  3:351aa16dec2c3a4787080249ed323eae
```

```
added ->:1b65e2167656d6bafa8c19904bd79445
Key [ 5]:8c9d18d9356a9954d341b4286e88ea1f
Key [ 6]:5c958d370102c9881bf753e69c7da029
round  4:2ce8fef47dda6a5bee74372e33e478a2
Key [ 7]:7eb2985c3697429fbe0da334bb51f795
Key [ 8]:af900f4b63a1138e2874bfb7c628b7b8
round  5:572787f563e1643c1c862b7555637fb4
Key [ 9]:834c8588dd8f3d4f31117a488420d69b
Key [10]:bc2b9b81c15d9a80262f3f48e9045895
round  6:16b4968c5d02853c3a43aa4cdb5f26ac
Key [11]:f08608c9e39ad3147cba61327919c958
Key [12]:2d4131decf4fa3a959084714a9e85c11
round  7:10e4120c7cccef9dd4ba4e6da8571b01
Key [13]:c934fd319c4a2b5361fa8eef05ae9572
Key [14]:4904c17aa47868e40471007cde3a97c0
round  8:f9081772498fed41b6ffd72b71fcf6c6
Key [15]:ea5e28687e97fa3f833401c86e6053ef
Key [16]:1168f58252c4ecfccafbdb3af857b9f2
Key [17]:b3440f69ef951b78b5cbd6866275301b
sres    :8d5205c5
aco     :3ed75df4abd9af638d144e94
----------------------------------------
rand    :0891caee063f5da1809577ff94ccdcfb
address :c62f19f6ce98
key     :45298d06e46bac21421ddfbed94c032b
round  1:0891caee063f5da1809577ff94ccdcfb
Key [ 1]:45298d06e46bac21421ddfbed94c032b
Key [ 2]:8f03e1e1fe1c191cad35a897bc400597
round  2:1c6ca013480a685c1b28e0317f7167e1
Key [ 3]:4f2ce3a092dde854ef496c8126a69e8e
Key [ 4]:968caee2ac6d7008c07283daec67f2f2
round  3:06b4915f5fcc1fc551a52048f0af8a26
Key [ 5]:ab0d5c31f94259a6bf85ee2d22edf56c
Key [ 6]:dfb74855c0085ce73dc17b84bfd50a92
round  4:077a92b040acc86e6e0a877db197a167
Key [ 7]:8f888952662b3db00d4e904e7ea53b5d
Key [ 8]:5e18bfcc07799b0132db88cd6042f599
round  5:7204881fb300914825fdc863e8ceadf3
Key [ 9]:bfca91ad9bd3d1a06c582b1d5512dddf
Key [10]:a88bc477e3fa1d5a59b5e6cf793c7a41
round  6:27031131d86cea2d747deb4f756143aa
Key [11]:f3cfb8dac8aea2a6a8ef95af3a2a2767
Key [12]:77beb90670c5300b03aa2b2232d3d40c
round  7:fc8c13d49149b1ce8d86f96e44a00065
Key [13]:b578373650af36a06e19fe335d726d32
Key [14]:6bcee918c7d0d24dfdf42237fcf99d53
round  8:04ef5f5a7ddf846cda0a07782fc23866
Key [15]:399f158241eb3e079f45d7b96490e7ea
Key [16]:1bcfbe98ecde2add52aa63ea79fb917a
Key [17]:ee8bc03ec08722bc2b075492873374af
round  1:d989d7a40cde7032d17b52f8117b69d5
Key [ 1]:2ecc6cc797cc41a2ab02007f6af396ae
```

```
Key [ 2]:acfaef7609c12567d537ae1cf9dc2198
round  2:8e76eb9a29b2ad5eea790db97aee37c1
Key [ 3]:079c8ff9b73d428df879906a0b87a6c8
Key [ 4]:19f2710baf403a494193d201f3a8c439
round  3:346bb7c35b2539676375aafe3af69089
added ->:edf48e675703a955b2f0fc062b71f95c
Key [ 5]:d623a6498f915cb2c8002765247b2f5a
Key [ 6]:900109093319bc30108b3d9434a77a72
round  4:fafb6c1f3ebbd2477be2da49dd923f69
Key [ 7]:e28e2ee6e72e7f4e5b5c11f10d204228
Key [ 8]:8e455cd11f8b9073a2dfa5413c7a4bc5
round  5:7c72230df588060a3cf920f9b0a08f06
Key [ 9]:28afb26e2c7a64238c41cefc16c53e74
Key [10]:d08dcafc2096395ba0d2dddd0e471f4d
round  6:55991df991db26ff00073a12baa3031d
Key [11]:fcffdcc3ad8faae091a7055b934f87c1
Key [12]:f8df082d77060252c02d91e55bd6a7d6
round  7:70ec682ff864375f63701fa4f6be5377
Key [13]:bef3706e523d708e8a44147d7508bc35
Key [14]:3e98ab283ca2422d56a56cf8b06caeb3
round  8:172f12ec933da85504b4ea5c90f8f0ea
Key [15]:87ad9625d06645d22598dd5ef811ea2c
Key [16]:8bd3db0cc8168009e5da90877e13a36f
Key [17]:0e74631d813a8351ac7039b348c41b42
sres    :00507e5f
aco     :2a5f19fbf60907e69f39ca9f
-----------------------------------------
rand    :0ecd61782b4128480c05dc45542b1b8c
address :f428f0e624b3
key     :35949a914225fabad91995d226de1d92
round  1:0ecd61782b4128480c05dc45542b1b8c
Key [ 1]:35949a914225fabad91995d226de1d92
Key [ 2]:ea6b3dcccc8ee5d88de349fa5010404f
round  2:8935e2e263fbc4b9302cabdfc06bce3e
Key [ 3]:920f3a0f2543ce535d4e7f25ad80648a
Key [ 4]:ad47227edf9c6874e80ba80ebb95d2c9
round  3:b4c8b878675f184a0c72f3dab51f8f05
Key [ 5]:81a941ca7202b5e884ae8fa493ecac3d
Key [ 6]:bcde1520bee3660e86ce2f0fb78b9157
round  4:77ced9f2fc42bdd5c6312b87fc2377c5
Key [ 7]:c8eee7423d7c6efa75ecec0d2cd969d3
Key [ 8]:910b3f838a02ed441fbe863a02b4a1d0
round  5:fe28e8056f3004d60bb207e628b39cf2
Key [ 9]:56c647c1e865eb078348962ae070972d
Key [10]:883965da77ca5812d8104e2b640aec0d
round  6:1f2ba92259d9e88101518f145a33840f
Key [11]:61d4cb7e4f8868a283327806a9bd8d4d
Key [12]:9f57de3a3ff310e21dc1e696ce060304
round  7:cc9b5d0218d29037e88475152ebebb2f
Key [13]:7aa1d8adc1aeed7127ef9a18f6eb2d8e
Key [14]:b4db9da3bf865912acd14904c7f7785d
round  8:b04d352bedc02682e4a7f59d7cda1dba
```

```
Key [15]:a13d7141ef1f6c7d867e3d175467381b
Key [16]:08b2bc058e50d6141cdd566a307e1acc
Key [17]:057b2b4b4be5dc0ac49e50489b8006c9
round  1:5cfacc773bae995cd7f1b81e7c9ec7df
Key [ 1]:1e717950f5828f3930fe4a9395858815
Key [ 2]:d1623369b733d98bbc894f75866c544c
round  2:d571ffa21d9daa797b1a0a3c962fc64c
Key [ 3]:4abf27664ae364cc8a7e5bcf88214cc4
Key [ 4]:2aaedda8dc4933dd6aeaf6e5c0d5a482
round  3:e17c8e498a00f125bf654c938c23f36d
added ->:bd765a3eb1ae8a796856048df0c1bab2
Key [ 5]:bc7f8ab2d86000f47b1946cc8d7a7a2b
Key [ 6]:6b28544cb13ec6c5d98470df2cf900b7
round  4:a9727c26f2f06bd9920e83c8605dcd76
Key [ 7]:1be840d9107f2c9523f66bb19f5464a1
Key [ 8]:61d6fb1aa2f0c2b26fb2a3d6de8c177c
round  5:aeff751f146eab7e4626b2e2c9e2fb39
Key [ 9]:adabfc82570c568a233173099f23f4c2
Key [10]:b7df6b55ad266c0f1ff7452101f59101
round  6:cf412b95f454d5185e67ca671892e5bd
Key [11]:8e04a7282a2950dcbaea28f300e22de3
Key [12]:21362c114433e29bda3e4d51f803b0cf
round  7:16165722fe4e07ef88f8056b17d89567
Key [13]:710c8fd5bb3cbb5f132a7061de518bd9
Key [14]:0791de7334f4c87285809343f3ead3bd
round  8:28854cd6ad4a3c572b15490d4b81bc3f
Key [15]:4f47f0e5629a674bfcd13770eb3a3bd9
Key [16]:58a6d9a16a284cc0aead2126c79608a1
Key [17]:a564082a0a98399f43f535fd5cefad34
sres    :80e5629c
aco     :a6fe4dcde3924611d3cc6ba1


========================================
```

9.2 FOUR TESTS OF E21

```
rand    :00000000000000000000000000000000
address :000000000000
round  1:00000000000000000000000000000000
Key [ 1]:00000000000000000000000000000006
Key [ 2]:4697b1baa3b7100ac537b3c95a28dc94
round  2:98611307ab76bbde9a86af1ce8cad412
Key [ 3]:ecabaac66795580df89af66e665d863d
Key [ 4]:8ac3d8896ae9364943bfebd4a2a768a0
round  3:820999ad2e6618f4b578974beeedf9e7
added ->:820999ad2e6618f4b578974beeedf9e7
Key [ 5]:5d57921fd5715cbb22c1bedb1c996394
Key [ 6]:2a61b8343219fdfb1740e9541d41448f
round  4:acd6edec87581ac22dbdc64ea4ced3a2
Key [ 7]:dd0480dee731d67f01ba0f39da6f23ca
Key [ 8]:3ad01cd1303e12a18dcfe0a8af82592c
```

```
round  5:1c7798732f09fbfe25795a4a2fbc93c2
Key [ 9]:7dadb2efc287ce7b0c1302904f2e7233
Key [10]:c08dcfa981e2f4572f6c7a9f52e11538
round  6:c05b88b56aa70e9c40c79bb81cd911bd
Key [11]:fc2042c708658a555e8c147660ffdfd7
Key [12]:fa0b21002605a6b9e89e624cd99150d2
round  7:abacc71b481c84c798d1bdf3d62f7e20
Key [13]:18b407e44a5ba4c80ecb48694b4e9c35
Key [14]:454d57e8253c0c4a8b3fcca7db6baef4
round  8:e8204e1183ae85cf19edb2c86215b700
Key [15]:2d0b946d9db52674f29353b0f011ed83
Key [16]:76c316733b1e8e70bd861b477e2456f1
Key [17]:8e4697b1baa3b7100ac537b3c95a28ac
Ka      :d14ca028545ec262cee700e39b5c39ee
-----------------------------------------
rand    :2dd9a550343191304013b2d7e1189d09
address :cac4364303b6
round  1:cac4364303b6cac4364303b6cac43643
Key [ 1]:2dd9a550343191304013b2d7e1189d0f
Key [ 2]:14c4335b2c43910c5dcc71d81a14242b
round  2:e169f788aad45a9011f11db5270b1277
Key [ 3]:55bfb712cba168d1a48f6e74cd9f4388
Key [ 4]:2a2b3aacca695caef2821b0fb48cc253
round  3:540f9c76652e92c44987c617035037bf
added ->:9ed3d23566e45c007fcac9a1c9146dfc
Key [ 5]:a06aab22d9a287384042976b4b6b00ee
Key [ 6]:c229d054bb72e8eb230e6dcdb32d16b7
round  4:83659a41675f7171ea57909dc5a79ab4
Key [ 7]:23c4812ab1905ddf77dedaed4105649a
Key [ 8]:40d87e272a7a1554ae2e85e3638cdf52
round  5:0b9382d0ed4f2fccdbb69d0db7b130a4
Key [ 9]:bdc064c6a39f6b84fe40db359f62a3c4
Key [10]:58228db841ce3cee983aa721f36aa1b9
round  6:c6ebda0f8f489792f09c189568226c1f
Key [11]:a815bacd6fa747a0d4f52883ac63ebe7
Key [12]:a9ce513b38ea006c333ecaaefcf1d0f8
round  7:75a8aba07e69c9065bcd831c40115116
Key [13]:3635e074792d4122130e5b824e52cd60
Key [14]:511bdb61bb28de72a5d794bffbf407df
round  8:57a6e279dcb764cf7dd6a749dd60c735
Key [15]:a32f5f21044b6744b6d913b13cdb4c0a
Key [16]:9722bbaeef281496ef8c23a9d41e92f4
Key [17]:807370560ad7e8a13a054a65a03b4049
Ka      :e62f8bac609139b3999aedbc9d228042
-----------------------------------------
rand    :dab3cffe9d5739d1b7bf4a667ae5ee24
address :02f8fd4cd661
round  1:02f8fd4cd66102f8fd4cd66102f8fd4c
Key [ 1]:dab3cffe9d5739d1b7bf4a667ae5ee22
Key [ 2]:e315a8a65d809ec7c289e69c899fbdcc
round  2:ef85ff081b8709405e19f3e275cec7dc
Key [ 3]:df6a119bb50945fc8a3394e7216448f3
```

```
Key [ 4]:87fe86fb0d58b5dd0fb3b6b1dab51d07
round  3:aa25c21bf577d92dd97381e3e9edcc54
added ->:a81dbf5723d8dbd524bf5782ebe5c918
Key [ 5]:36cc253c506c0021c91fac9d8c469e90
Key [ 6]:d5fda00f113e303809b7f7d78a1a2b0e
round  4:9e69ce9b53caec3990894d2baed41e0d
Key [ 7]:c14b5edc10cabf16bc2a2ba4a8ae1e40
Key [ 8]:74c6131afc8dce7e11b03b1ea8610c16
round  5:a5460fa8cedca48a14fd02209e01f02e
Key [ 9]:346cfc553c6cbc9713edb55f4dcbc96c
Key [10]:bddf027cb059d58f0509f8963e9bdec6
round  6:92b33f11eadcacc5a43dd05f13d334dd
Key [11]:8eb9e040c36c4c0b4a7fd3dd354d53c4
Key [12]:c6ffecdd5e135b20879b9dfa4b34bf51
round  7:fb0541aa5e5df1a61c51aef606eb5a41
Key [13]:bf12f5a6ba08dfc4fda4bdfc68c997d9
Key [14]:37c4656b9215f3c959ea688fb64ad327
round  8:f0bbd2b94ae374346730581fc77a9c98
Key [15]:e87bb0d86bf421ea4f779a8eee3a866c
Key [16]:faa471e934fd415ae4c0113ec7f0a5ad
Key [17]:95204a80b8400e49db7cf6fd2fd40d9a
Ka       :b0376d0a9b338c2e133c32b69cb816b3
----------------------------------------
rand     :13ecad08ad63c37f8a54dc56e82f4dc1
address :9846c5ead4d9
round  1:9846c5ead4d99846c5ead4d99846c5ea
Key [ 1]:13ecad08ad63c37f8a54dc56e82f4dc7
Key [ 2]:ad04f127bed50b5e671d6510d392eaed
round  2:97374e18cdd0a6f7a5aa49d1ac875c84
Key [ 3]:57ad159e5774fa222f2f3039b9cd5101
Key [ 4]:9a1e9e1068fede02ef90496e25fd8e79
round  3:9dd3260373edd9d5f4e774826b88fd2d
added ->:0519ebe9a7c6719331d1485bf3cec2c7
Key [ 5]:378dce167db62920b0b392f7cfca316e
Key [ 6]:db4277795c87286faee6c9e9a6b71a93
round  4:40ec6563450299ac4e120d88672504d6
Key [ 7]:ec01aa2f5a8a793b36c1bb858d254380
Key [ 8]:2921a66cfa5bf74ac535424564830e98
round  5:57287bbb041bd6a56c2bd931ed410cd4
Key [ 9]:07018e45aab61b3c3726ee3d57dbd5f6
Key [10]:627381f0fa4c02b0c7d3e7dfbffc3333
round  6:66affa66a8dcd36e36bf6c3f1c6a276e
Key [11]:33b57c925bd5551999f716e138efbe79
Key [12]:a6dc7f9aa95bcc9243aebd12608f657a
round  7:450e65184fd8c72c578d5cdecb286743
Key [13]:a6a6db00fd8c72a28ea57ea542f6e102
Key [14]:dcf3377daeb2e24e61f0ad6620951c1f
round  8:e5eb180b519a4e673f21b7c4f4573f3d
Key [15]:621240b9506b462a7fa250da41844626
Key [16]:ae297810f01f43dc35756cd119ee73d6
Key [17]:b959835ec2501ad3894f8b8f1f4257f9
Ka       :5b61e83ad04d23e9d1c698851fa30447
```

```
========================================
```

9.3 THREE TESTS OF E22

(for K_master and overlay generation)

```
rand      :001de169248850245a5f7cc7f0d6d633
PIN       :d5a51083a04a1971f18649ea8b79311a
round   1:001de169248850245a5f7cc7f0d6d623
Key [ 1]:d5a51083a04a1971f18649ea8b79311a
Key [ 2]:7317cdbff57f9b99f9810a2525b17cc7
round   2:5f05c143347b59acae3cb002db23830f
Key [ 3]:f08bd258adf1d4ae4a54d8ccb26220b2
Key [ 4]:91046cbb4ccc43db18d6dd36ca7313eb
round   3:c8f3e3300541a25b6ac5a80c3105f3c4
added ->:c810c45921c9f27f302424cbc1dbc9e7
Key [ 5]:67fb2336f4d9f069da58d11c82f6bd95
Key [ 6]:4fed702c75bd72c0d3d8f38707134c50
round   4:bd5e0c3a97fa55b91a3bbbf306ebb978
Key [ 7]:41c947f80cdc0464c50aa89070af314c
Key [ 8]:680eecfa8daf41c7109c9a5cb1f26d75
round   5:21c1a762c3cc33e75ce8976a73983087
Key [ 9]:6e33fbd94d00ff8f72e8a7a0d2cebc4c
Key [10]:f4d726054c6b948add99fabb5733ddc3
round   6:56d0df484345582f6b574a449ba155eb
Key [11]:4eda2425546a24cac790f49ef2453b53
Key [12]:cf2213624ed1510408a5a3e00b7333df
round   7:120cf9963fe9ff22993f7fdf9600d9b8
Key [13]:d04b1a25b0b8fec946d5ecfa626d04c9
Key [14]:01e5611b0f0e140bdb64585fd3ae5269
round   8:a6337400ad8cb47fefb91332f5cb2713
Key [15]:f15b2dc433f534f61bf718770a3698b1
Key [16]:f990d0273d8ea2b9e0b45917a781c720
Key [17]:f41b3cc13d4301297bb6bdfcb3e5a1dd
Ka        :539e4f2732e5ae2de1e0401f0813bd0d
----------------------------------------
rand      :67ed56bfcf99825f0c6b349369da30ab
PIN       :7885b515e84b1f082cc499976f1725ce
round   1:67ed56bfcf99825f0c6b349369da30bb
Key [ 1]:7885b515e84b1f082cc499976f1725ce
Key [ 2]:72445901fdaf506beb036f4412512248
round   2:6b160b66a1f6c26c1f3432f463ef5aa1
Key [ 3]:59f0e4982e97633e5e7fd133af8f2c5b
Key [ 4]:b4946ec77a41bf7c729d191e33d458ab
round   3:3f22046c964c3e5ca2a26ec9a76a9f67
added ->:580f5ad359e5c003ae0da25ace44cfdc
Key [ 5]:eb0b839f97bdf534183210678520bbef
Key [ 6]:cff0bc4a94e5c8b2a2d24d9f59031e19
round   4:87aa61fc0ff88e744c195249b9a33632
Key [ 7]:592430f14d8f93db95dd691af045776d
Key [ 8]:3b55b404222bf445a6a2ef5865247695
round   5:83dcf592a854226c4dcd94e1ecf1bc75
```

```
Key [ 9]:a9714b86319ef343a28b87456416bd52
Key [10]:e6598b24390b3a0bf2982747993b0d78
round  6:dee0d13a52e96bcf7c72045a21609fc6
Key [11]:62051d8c51973073bff959b032c6e1e2
Key [12]:29e94f4ab73296c453c833e217a1a85b
round  7:08488005761e6c7c4dbb203ae453fe3a
Key [13]:0e255970b3e2fc235f59fc5acb10e8ce
Key [14]:d0dfbb3361fee6d4ffe45babf1cd7abf
round  8:0d81e89bddde7a7065316c47574feb8f
Key [15]:c12eee4eb38b7a171f0f736003774b40
Key [16]:8f962523f1c0abd9a087a0dfb11643d3
Key [17]:24be1c66cf8b022f12f1fb4c60c93fd1
Ka       :04435771e03a9daceb8bb1a493ee9bd8
-----------------------------------------
rand     :40a94509238664f244ff8e3d13b119d3
PIN      :1ce44839badde30396d03c4c36f23006
round  1:40a94509238664f244ff8e3d13b119c3
Key [ 1]:1ce44839badde30396d03c4c36f23006
Key [ 2]:6dd97a8f91d628be4b18157af1a9dcba
round  2:0eac5288057d9947a24eabc1744c4582
Key [ 3]:fef9583d5f55fd4107ad832a725db744
Key [ 4]:fc3893507016d7c1db2bd034a230a069
round  3:60b424f1082b0cc3bd61be7b4c0155f0
added ->:205d69f82bb17031f9604c465fb26e33
Key [ 5]:0834d04f3e7e1f7f85f0c1db685ab118
Key [ 6]:1852397f9a3723169058e9b62bb3682b
round  4:2c6b65a49d66af6566675afdd6fa7d7d
Key [ 7]:6c10da21d762ae4ac1ba22a96d9007b4
Key [ 8]:9aa23658b90470a78d686344b8a9b0e7
round  5:a2c537899665113a42f1ac24773bdc31
Key [ 9]:137dee3bf879fe7bd02fe6d888e84f16
Key [10]:466e315a1863f47d0f93bc6827cf3450
round  6:e26982980d79b21ed3e20f8c3e71ba96
Key [11]:0b33cf831465bb5c979e6224d7f79f7c
Key [12]:92770660268ede827810d707a0977d73
round  7:e7b063c4e2e3110b89b7e1631c762dd5
Key [13]:7be30ae4961cf24ca17625a77bb7a9f8
Key [14]:be65574a33ae30e6e82dbd2826d3cc1a
round  8:7a963e37b2c2e76b489cfe40a2cf00e5
Key [15]:ed0ba7dd30d60a5e69225f0a33011e5b
Key [16]:765c990f4445e52b39e6ed6105ad1c4f
Key [17]:52627bf9f35d94f30d5b07ef15901adc
Ka       :9cde4b60f9b5861ed9df80858bac6f7f


=========================================
```

9.4 TESTS OF E22 WITH PIN AUGMENTING

for PIN lengths 1,...,16 bytes

```
rand     :24b101fd56117d42c0545a4247357048
PIN length =16 octets
PIN      :fd397c7f5c1f937cdf82d8816cc377e2
round  1:24b101fd56117d42c0545a4247357058
Key [ 1]:fd397c7f5c1f937cdf82d8816cc377e2
Key [ 2]:0f7aac9c9b53f308d9fdbf2c78e3c30e
round  2:838edfe1226266953ccba8379d873107
Key [ 3]:0b8ac18d4bb44fad2efa115e43945abc
Key [ 4]:887b16b062a83bfa469772c25b456312
round  3:8cd0c9283120aba89a7f9d635dd4fe3f
added ->:a881cad5673128ea5ad3f7211a096e67
Key [ 5]:2248cbe6d299e9d3e8fd35a91178f65b
Key [ 6]:b92af6237385bd31f8fb57fb1bdd824e
round  4:2648d9c618a622b10ef80c4dbaf68b99
Key [ 7]:2bf5ffe84a37878ede2d4c30be60203b
Key [ 8]:c9cb6cec60cb8a8f29b99fcf3e71e40f
round  5:b5a7d9e96f68b14ccebf361de3914d0f
Key [ 9]:5c2f8a702e4a45575b103b0cce8a91c6
Key [10]:d453db0c9f9ddbd11e355d9a34d9b11b
round  6:632a091e7eefe1336857ddafd1ff3265
Key [11]:32805db7e59c5ed4acabf38d27e3fece
Key [12]:fde3a8eedfa3a12be09c1a8a00890fd7
round  7:048531e9fd3efa95910540150f8b137b
Key [13]:def07eb23f3a378f059039a2124bc4c2
Key [14]:2608c58f23d84a09b9ce95e5caac1ab4
round  8:461814ec7439d412d0732f0a6f799a6a
Key [15]:0a7ed16481a623e56ee1442ffa74f334
Key [16]:12add59aca0d19532f1516979954e369
Key [17]:dd43d02d39ffd6a386a4b98b4ac6eb23
Ka       :a5f2adf328e4e6a2b42f19c8b74ba884
-----------------------------------------
rand     :321964061ac49a436f9fb9824ac63f8b
PIN length =15 octets
PIN      :ad955d58b6b8857820ac1262d617a6
address :0314c0642543
round  1:321964061ac49a436f9fb9824ac63f9b
Key [ 1]:ad955d58b6b8857820ac1262d617a603
Key [ 2]:f281736f68e3d30b2ac7c67f125dc416
round  2:7c4a4ece1398681f4bafd309328b7770
Key [ 3]:43c157f4c8b360387c32ab330f9c9aa8
Key [ 4]:3a3049945a298f6d076c19219c47c3cb
round  3:9672b00738bdfaf9bd92a855bc6f3afb
added ->:a48b1401228194bad23161d7f6357960
Key [ 5]:c8e2eaa6d73b7de18f3228ab2173bc69
Key [ 6]:8623f44488222e66a293677cf30bf2bb
round  4:9b30247aad3bf133712d034b46d21c68
Key [ 7]:f3e500902fba31db9bae50ef30e484a4
Key [ 8]:49d4b1137c18f4752dd9955a5a8d2f43
```

```
round  5:4492c25fda08083a768b4b5588966b23
Key [ 9]:9d59c451989e74785cc097eda7e42ab8
Key [10]:251de25f3917dcd99c18646107a641fb
round  6:21ae346635714d2623041f269978c0ee
Key [11]:80b8f78cb1a49ec0c3e32a238e60fddf
Key [12]:beb84f4d20a501e4a24ecfbde481902b
round  7:9b56a3d0f8932f20c6a77a229514fb00
Key [13]:852571b44f35fd9d9336d3c1d2506656
Key [14]:d0a0d510fb06ba76e69b8ee3ebc1b725
round  8:6cd8492b2fd31a86978bcdf644eb08a8
Key [15]:c7ffd523f32a874ed4a93430a25976de
Key [16]:16cdcb25e62964876d951fdcc07030d3
Key [17]:def32c0e12596f9582e5e3c52b303f52
Ka      :c0ec1a5694e2b48d54297911e6c98b8f
-----------------------------------------
rand    :d4ae20c80094547d7051931b5cc2a8d6
PIN length =14 octets
PIN     :e1232e2c5f3b833b3309088a87b6
address :fabecc58e609
round  1:d4ae20c80094547d7051931b5cc2a8c6
Key [ 1]:e1232e2c5f3b833b3309088a87b6fabe
Key [ 2]:5f0812b47cd3e9a30d7707050fffa1f2
round  2:1f45f16be89794bef33e4547c9c0916a
Key [ 3]:77b681944763244ffa3cd71b248b79b5
Key [ 4]:e2814e90e04f485958ce58c9133e2be6
round  3:b10d2f4ac941035263cee3552d774d2f
added ->:65bb4f82c9d5572f131f764e7139f5e9
Key [ 5]:520acad20801dc639a2c6d66d9b79576
Key [ 6]:c72255cdb61d42be72bd45390dd25ba5
round  4:ead4dc34207b6ea721c62166e155aaad
Key [ 7]:ebf04c02075bf459ec9c3ec06627d347
Key [ 8]:a1363dd2812ee800a4491c0c74074493
round  5:f507944f3018e20586d81d7f326aae9d
Key [ 9]:b0b6ba79493dc833d7f425be7b8dadb6
Key [10]:08cd23e536b9b9b53e85eb004cba3111
round  6:fff450f4302a2b3571e8405e148346da
Key [11]:fec22374c6937dcd26171f4d2edfada3
Key [12]:0f1a8ef5979c69ff44f620c2e007b6e4
round  7:de558779589897f3402a90ee78c3f921
Key [13]:901fb66f0779d6aad0c0fba1fe812cb5
Key [14]:a0cab3cd15cd23603adc8d4474efb239
round  8:b2df0aa0c9f07fbbaa02f510a29cf540
Key [15]:18edc3f4296dd6f1dea13f7c143117a1
Key [16]:8d3d52d700a379d72ded81687f7546c7
Key [17]:5927badfe602f29345f840bb53e1dea6
Ka      :d7b39be13e3692c65b4a9e17a9c55e17
-----------------------------------------
rand    :272b73a2e40db52a6a61c6520549794a
PIN length =13 octets
PIN     :549f2694f353f5145772d8ae1e
address :20487681eb9f
round  1:272b73a2e40db52a6a61c6520549795a
```

```
Key [ 1]:549f2694f353f5145772d8ae1e204876
Key [ 2]:42c855593d66b0c458fd28b95b6a5fbf
round  2:d7276dc8073f7677c31f855bde9501e2
Key [ 3]:75d0a69ae49a2da92e457d767879df52
Key [ 4]:b3aa7e7492971afaa0fb2b64827110df
round  3:71aae503831133d19bc452da4d0e409b
added ->:56d558a1671ee8fbf12518884857b9c1
Key [ 5]:9c8cf1604a98e9a503c342e272de5cf6
Key [ 6]:d35bc2df6b85540a27642106471057d9
round  4:f41a709c89ea80481aa3d2b9b2a9f8ca
Key [ 7]:b454dda74aeb4eff227ba48a58077599
Key [ 8]:bcba6aec050116aa9b7c6a9b7314d796
round  5:20fdda20f4a26b1bd38eb7f355a7be87
Key [ 9]:d41f8a9de0a716eb7167a1b6e321c528
Key [10]:5353449982247782d168ab43f17bc4d8
round  6:a70e316997eeed49a5a9ef9ba5e913b5
Key [11]:32cbc9cf1a81e36a45153972347ce4ac
Key [12]:5747619006cf4ef834c749f2c4b9feb6
round  7:e66f2317a825f589f76b47b6aa6e73fb
Key [13]:f9b68beba0a09d2a570a7dc88cc3c3c2
Key [14]:55718f9a4f0b1f9484e8c6b186a41a4b
round  8:5f68f940440a9798e074776019804ada
Key [15]:4ecc29be1b4d78433f6aa30db974a7fb
Key [16]:8470a066ffb00cda7b08059599f919f5
Key [17]:f39a36d74e960a051e1ca98b777848f4
Ka      :9ac64309a37c25c3b4a584fc002a1618
-----------------------------------------
rand    :7edb65f01a2f45a2bc9b24fb3390667e
PIN length =12 octets
PIN     :2e5a42797958557b23447ca8
address :04f0d2737f02
round  1:7edb65f01a2f45a2bc9b24fb3390666e
Key [ 1]:2e5a42797958557b23447ca804f0d273
Key [ 2]:18a97c856561eb23e71af8e9e1be4799
round  2:3436e12db8ffdc1265cb5a86da2fed0b
Key [ 3]:7c0908dcbc73201e17c4f7aa1ab8aec8
Key [ 4]:7cb58833602fbe4194c7cc797ce8c454
round  3:caed6af4226f67e4ad1914620803ef2a
added ->:b4c8cf04389eac4611b438993b935544
Key [ 5]:f4dce7d607b5234562d0ebb2267b08b8
Key [ 6]:560b75c5545751fd8fa99fa4346e654b
round  4:ee67c87d6f74bb75db98f68bff0192c1
Key [ 7]:32f10cefd8d3e6424c6f91f1437808af
Key [ 8]:a934a46045be30fb3be3a5f3f7b18837
round  5:792398dcbeb8d10bdb07ae3c819e943c
Key [ 9]:a0f12e97c677a0e8ac415cd2c8a7ca88
Key [10]:e27014c908785f5ca03e8c6a1da3bf13
round  6:e778b6e0c3e8e7edf90861c7916d97a8
Key [11]:1b4a4303bcc0b2e0f41c72d47654bd9f
Key [12]:4b1302a50046026d6c9054fc8387965a
round  7:1fafddc7efa5f04c1dec1869d3f2d9bb
Key [13]:58c334bb543d49eca562cdbe0280e0fc
```

```
Key [14]:bdb60d383c692d06476b76646c8dec48
round  8:3d7c326d074bd6aa222ea050f04a3c7f
Key [15]:78c0162506be0b5953e8403c01028f93
Key [16]:24d7dbbe834dbd7b67f57fcf0d39d60f
Key [17]:2e74f1f3331c0f6585e87b2f715e187e
Ka      :d3af4c81e3f482f062999dee7882a73b
----------------------------------------
rand    :26a92358294dce97b1d79ec32a67e81a
PIN length =11 octets
PIN     :05fbad03f52fa9324f7732
address :b9ac071f9d70
round  1:26a92358294dce97b1d79ec32a67e80a
Key [ 1]:05fbad03f52fa9324f7732b9ac071f9d
Key [ 2]:2504c9691c04a18480c8802e922098c0
round  2:0be20e3d76888e57b6bf77f97a8714fb
Key [ 3]:576b2791d1212bea8408212f2d43e77e
Key [ 4]:90ae36dcce8724adb618f912d1b27297
round  3:19696670607644532 57d906b7e58bd5b
added ->:3f12892849c312c494542ea854bfa551
Key [ 5]:bc492c42c9e87f56ec31af5474e9226e
Key [ 6]:c135d1dbed32d9519acfb4169f3e1a10
round  4:ac404205118fe771e54aa6f392da1153
Key [ 7]:83ccbdbbaf17889b7d18254dc9252fa1
Key [ 8]:80b90a1767d3f2848080802764e21711
round  5:41795e89ae9a0cf776ffece76f47fd7a
Key [ 9]:cc24e4a86e8eed129118fd3d5223a1dc
Key [10]:7b1e9c0eb9dab083574be7b7015a62c9
round  6:29ca9e2f87ca00370ef1633505bfba4b
Key [11]:888e6d88cf4beb965cf7d4f32b696baa
Key [12]:6d642f3e5510b0b043a44daa2cf5eec0
round  7:81fc891c3c6fd99acc00028a387e2366
Key [13]:e224f85da2ab63a23e2a3a036e421358
Key [14]:c8dc22aaa739e2cb85d6a0c08226c7d0
round  8:e30b537e7a000e3d2424a9c0f04c4042
Key [15]:a969aa818c6b324bae391bedcdd9d335
Key [16]:6974b6f2f07e4c55f2cc0435c45bebd1
Key [17]:134b925ebd98e6b93c14aee582062fcb
Ka      :be87b44d079d45a08a71d15208c5cb50
----------------------------------------
rand    :0edef05327eab5262430f21fc91ce682
PIN length =10 octets
PIN     :8210e47390f3f48c32b3
address :7a3cdfe377d1
round  1:0edef05327eab5262430f21fc91ce692
Key [ 1]:8210e47390f3f48c32b37a3cdfe377d1
Key [ 2]:c6be4c3e425e749b620a94c779e33a7e
round  2:07ca3c7a7a6bcbc31d79a856d9cffc0e
Key [ 3]:2587cec2a4b8e4f996a9ed664350d5dd
Key [ 4]:70e4bf72834d9d3dbb7eb2c239216dc0
round  3:792ad2ac4e4559d1463714d2f161b6f4
added ->:7708c2ff692f0ef7626706cd387d9c66
Key [ 5]:6696e1e7f8ac037e1fff3598f0c164e2
```

```
Key [ 6]:23dbfe4d0b561bea08fbcef25e49b648
round  4:7d8c71a9d7fbdcbd851bdf074550b100
Key [ 7]:b03648acd021550edee904431a02f00c
Key [ 8]:cb169220b7398e8f077730aa4bf06baa
round  5:b6fcaa45064ffd557e4b7b30cfbb83e0
Key [ 9]:af602c2ba16a454649951274c2be6527
Key [10]:5d60b0a7a09d524143eca13ad680bc9c
round  6:b3416d391a0c26c558843debd0601e9e
Key [11]:9a2f39bfe558d9f562c5f09a5c3c0263
Key [12]:72cae8eebd7fabd9b1848333c2aab439
round  7:abe4b498d9c36ea97b8fd27d7f813913
Key [13]:15f27ea11e83a51645d487b81371d7dc
Key [14]:36083c8666447e03d33846edf444eb12
round  8:8032104338a945ba044d102eabda3b22
Key [15]:0a3a8977dd48f3b6c1668578befadd02
Key [16]:f06b6675d78ca0ee5b1761bdcdab516d
Key [17]:cbc8a7952d33aa0496f7ea2d05390b23
Ka      :bf0706d76ec3b11cce724b311bf71ff5
-----------------------------------------
rand    :86290e2892f278ff6c3fb917b020576a
PIN length = 9 octets
PIN     :3dcdffcfd086802107
address :791a6a2c5cc3
round  1:86290e2892f278ff6c3fb917b0205765
Key [ 1]:3dcdffcfd086802107791a6a2c5cc33d
Key [ 2]:b4962f40d7bb19429007062a3c469521
round  2:1ec59ffd3065f19991872a7863b0ef02
Key [ 3]:eb9ede6787dd196b7e340185562bf28c
Key [ 4]:2964e58aacf7287d1717a35b100ae23b
round  3:f817406f1423fc2fe33e25152679eaaf
added ->:7e404e47861574d08f7dde02969941ca
Key [ 5]:6abf9a314508fd61e486fa4e376c3f93
Key [ 6]:6da148b7ee2632114521842cbb274376
round  4:e9c2a8fac22b8c7cf0c619e2b3f890ed
Key [ 7]:df889cc34fda86f01096d52d116e620d
Key [ 8]:5eb04b147dc39d1974058761ae7b73fc
round  5:444a8aac0efee1c02f8d38f8274b7b28
Key [ 9]:8426cc59eee391b2bd50cf8f1efef8b3
Key [10]:8b5d220a6300ade418da791dd8151941
round  6:9185f983db150b1bccab1e5c12eb63a1
Key [11]:82ba4ddef833f6a4d18b07aa011f2798
Key [12]:ce63d98794682054e73d0359dad35ec4
round  7:5eded2668f5916dfd036c09e87902886
Key [13]:da794357652e80c70ad8b0715dbe33d6
Key [14]:732ef2c0c3220b31f3820c375e27bb29
round  8:88a5291b4acbba009a85b7dd6a834b3b
Key [15]:3ce75a61d4b465b70c95d7ccd5799633
Key [16]:5df9bd2c3a17a840cdaafb76c171db7c
Key [17]:3f8364b089733d902bccb0cd3386846f
Ka      :cdb0cc68f6f6fbd70b46652de3ef3ffb
-----------------------------------------
rand    :3ab52a65bb3b24a08eb6cd284b4b9d4b
```

```
PIN length = 8 octets
PIN      :d0fb9b6838d464d8
address :25a868db91ab
round  1:3ab52a65bb3b24a08eb6cd284b4b9d45
Key [ 1]:d0fb9b6838d464d825a868db91abd0fb
Key [ 2]:2573f47b49dad6330a7a9155b7ae8ba1
round  2:ad2ffdff408fcfab44941016a9199251
Key [ 3]:d2c5b8fb80cba13712905a589adaee71
Key [ 4]:5a3381511b338719fae242758dea0997
round  3:2ddc17e570d7931a2b1d13f6ace928a5
added ->:17914180cb12b7baa5d3e0dee734c5e0
Key [ 5]:e0a4d8ac27fbe2783b7bcb3a36a6224d
Key [ 6]:949324c6864deac3eca8e324853e11c3
round  4:62c1db5cf31590d331ec40ad692e8df5
Key [ 7]:6e67148088a01c2d4491957cc9ddc4aa
Key [ 8]:557431deab7087bb4c03fa27228f60c6
round  5:9c8933bc361f4bde4d1bda2b5f8bb235
Key [ 9]:a2551aca53329e70ade3fd2bb7664697
Key [10]:05d0ad35de68a364b54b56e2138738fe
round  6:9156db34136aa06655bf28a05be0596a
Key [11]:1616a6b13ce2f2895c722e8495181520
Key [12]:b12e78a1114847b01f6ed2f5a1429a23
round  7:84dcc292ed836c1c2d523f2a899a2ad5
Key [13]:316e144364686381944e95afd8a026bb
Key [14]:1ab551b88d39d97ea7a9fe136dbfe2e1
round  8:87bdcac878d777877f4eccf042cfee5e
Key [15]:70e21ab08c23c7544524b64492b25cc9
Key [16]:35f730f2ae2b950a49a1bf5c8b9f8866
Key [17]:2f16924c22db8b74e2eadf1ba4ebd37c
Ka       :983218718ca9aa97892e312d86dd9516
-----------------------------------------
rand     :a6dc447ff08d4b366ff96e6cf207e179
PIN length = 7 octets
PIN      :9c57e10b4766cc
address :54ebd9328cb6
round  1:a6dc447ff08d4b366ff96e6cf207e174
Key [ 1]:9c57e10b4766cc54ebd9328cb69c57e1
Key [ 2]:00a609f4d61db26993c8177e3ee2bba8
round  2:1ed26b96a306d7014f4e5c9ee523b73d
Key [ 3]:646d7b5f9aaa528384bda3953b542764
Key [ 4]:a051a42212c0e9ad5c2c248259aca14e
round  3:a53f526db18e3d7d53edbfc971104led
added ->:031b961241lb884b3ce62da583172299
Key [ 5]:d1bd5e64930e7f838d8a33994462d8b2
Key [ 6]:5dc7e2291e32435665ebd6956bec3414
round  4:9438be308ec83f35c560e2796f4e0559
Key [ 7]:10552f45af63b0f15e2919ab37f64fe7
Key [ 8]:c44d5717c114a58b09207392ebe341f8
round  5:b79a7b14386066d339f799c40479cb3d
Key [ 9]:6886e47b782325568eaf59715a75d8ff
Key [10]:8e1e335e659cd36b132689f78c147bda
round  6:ef232462228aa166438d10c34e17424b
```

```
Key [11]:8843efeedd5c2b7c3304d647f932f4d1
Key [12]:13785aaedd0adf67abb4f01872392785
round  7:02d133fe40d15f1073673b36bba35abd
Key [13]:837d7ca2722419e6be3fae35900c3958
Key [14]:93f8442973e7fccf2e7232d1d057c73a
round  8:275506a3d08c84e94cc58ed60054505e
Key [15]:8a7a9edffa3c52918bc6a45f57d91f5d
Key [16]:f214a95d777f763c56109882c4b52c84
Key [17]:10e2ee92c5ea1ddc5eb010e55510c403
Ka      :9cd6650ead86323e87cafb1ff516d1e0
-----------------------------------------
rand    :3348470a7ea6cc6eb81b40472133262c
PIN length = 6 octets
PIN     :fcad169d7295
address :430d572f8842
round  1:3348470a7ea6cc6eb81b404721332620
Key [ 1]:fcad169d7295430d572f8842fcad169d
Key [ 2]:b3479d4d4fd178c43e7bc5b0c7d8983c
round  2:af976da9225066d563e10ab955e6fc32
Key [ 3]:7112462b37d82dd81a2a35d9eb43cb7c
Key [ 4]:c5a7030f8497945ac7b84600d1d161fb
round  3:d08f826ebd55a0bd7591c19a89ed9bde
added ->:e3d7c964c3fb6cd3cdac01dda820c1fe
Key [ 5]:84b0c6ef4a63e4dff19b1f546d683df5
Key [ 6]:f4023edfc95d1e79ed4bb4de9b174f5d
round  4:6cd952785630dfc7cf81eea625e42c5c
Key [ 7]:ea38dd9a093ac9355918632c90c79993
Key [ 8]:dbba01e278ddc76380727f5d7135a7de
round  5:93573b2971515495978264b88f330f7f
Key [ 9]:d4dc3a31be34e412210fafa6eca00776
Key [10]:39d1e190ee92b0ff16d92a8be58d2fa0
round  6:b3f01d5e7fe1ce6da7b46d8c389baf47
Key [11]:1eb081328d4bcf94c9117b12c5cf22ac
Key [12]:7e047c2c552f9f1414d946775fabfe30
round  7:0b833bff6106d5bae033b4ce5af5a924
Key [13]:e78e685d9b2a7e29e7f2a19d1bc38ebd
Key [14]:1b582272a3121718c4096d2d8602f215
round  8:23de0bbdc70850a7803f4f10c63b2c0f
Key [15]:8569e860530d9c3d48a0870dac33f676
Key [16]:6966b528fdd1dc222527052c8f6cf5a6
Key [17]:a34244c757154c53171c663b0b56d5c2
Ka      :98f1543ab4d87bd5ef5296fb5e3d3a21
-----------------------------------------
rand    :0f5bb150b4371ae4e5785293d22b7b0c
PIN length = 5 octets
PIN     :b10d068bca
address :b44775199f29
round  1:0f5bb150b4371ae4e5785293d22b7b07
Key [ 1]:b10d068bcab44775199f29b10d068bca
Key [ 2]:aec70d1048f1bbd2c18040318a8402ad
round  2:342d2b79d7fb7cd110379742b9842c79
Key [ 3]:6d8d5cf338f29ef4420639ef488e4fa9
```

```
Key [ 4]:a1584117541b759ba6d9f7eb2bedcbba
round  3:9407e8e3e810603921bf81cfda62770a
added ->:9b6299b35c477addc437d35c088df20d
Key [ 5]:09a20676666aeed6f22176274eb433f4
Key [ 6]:840472c001add5811a054be5f5c74754
round  4:9a3ba953225a7862c0a842ed3d0b2679
Key [ 7]:fad9e45c8bf70a972fcd9bff0e8751f5
Key [ 8]:e8f30ff666dfd212263416496ff3b2c2
round  5:2c573b6480852e875df34b28a5c44509
Key [ 9]:964cdba0cf8d593f2fc40f96daf8267a
Key [10]:bcd65c11b13e1a70bcd4aafba8864fe3
round  6:21b0cc49e880c5811d24dee0194e6e9e
Key [11]:468c8548ea9653c1a10df6288dd03c1d
Key [12]:5d252d17af4b09d3f4b5f7b5677b8211
round  7:e6d6bdcd63e1d37d9883543ba86392fd
Key [13]:e814bf307c767428c67793dda2df95c7
Key [14]:4812b979fdc20f0ff0996f61673a42cc
round  8:e3dde7ce6bd7d8a34599aa04d6a760ab
Key [15]:5b1e2033d1cd549fc4b028146eb5b3b7
Key [16]:0f284c14fb8fe706a5343e3aa35af7b1
Key [17]:b1f7a4b7456d6b577fded6dc7a672e37
Ka      :c55070b72bc982adb972ed05d1a74ddb
----------------------------------------
rand    :148662a4baa73cfadb55489159e476e1
PIN length = 4 octets
PIN     :fb20f177
address :a683bd0b1896
round  1:148662a4baa73cfadb55489159e476eb
Key [ 1]:fb20f177a683bd0b1896fb20f177a683
Key [ 2]:47266cefbfa468ca7916b458155dc825
round  2:3a942eb6271c3f4e433838a5d3fcbd27
Key [ 3]:688853a6d6575eb2f6a2724b0fbc133b
Key [ 4]:7810df048019634083a2d9219d0b5fe0
round  3:9c835b98a063701c08879435967807 69
added ->:8809bd3c1a0aace6d3dcdca4cf5c7d82
Key [ 5]:c78f6dcf56da1bbd413828b33f5865b3
Key [ 6]:eb3f3d407d160df3d293a76d1a513c4a
round  4:7e68c4bafa020a4a59b5a1968105bab5
Key [ 7]:d330e038d6b19d5c9bb0d7285a360064
Key [ 8]:9bd3ee50347c00753d165faced702d9c
round  5:227bad0cf0838bdb15b3b3872c24f592
Key [ 9]:9543ad0fb3fe74f83e0e2281c6d4f5f0
Key [10]:746cd0383c17e0e80e6d095a87fd0290
round  6:e026e98c71121a0cb739ef6f59e14d26
Key [11]:fa28bea4b1c417536608f11f406ea1dd
Key [12]:3aee0f4d21699df9cb8caf5354a780ff
round  7:cd6a6d8137d55140046f8991da1fa40a
Key [13]:372b71bc6d1aa6e785358044fbcf05f4
Key [14]:00a01501224c0405de00aa2ce7b6ab04
round  8:52cd7257fe8d0c782c259bcb6c9f5942
Key [15]:c7015c5c1d7c030e00897f104a006d4a
Key [16]:260a9577790c62e074e71e19fd2894df
```

```
Key [17]:c041b7a231493acd15ddcdaee94b9f52
Ka      :7ec864df2f1637c7e81f2319ae8f4671
-----------------------------------------
rand    :193a1b84376c88882c8d3b4ee93ba8d5
PIN length = 3 octets
PIN     :a123b9
address :4459a44610f6
round  1:193a1b84376c88882c8d3b4ee93ba8dc
Key [ 1]:a123b94459a44610f6a123b94459a446
Key [ 2]:5f64d384c8e990c1d25080eb244dde9b
round  2:3badbd58f100831d781ddd3ccedefd3f
Key [ 3]:5abc00eff8991575c00807c48f6dbea5
Key [ 4]:127521158ad6798fb6479d1d2268abe6
round  3:0b53075a49c6bf2df2421c655fdedf68
added ->:128d22de7e3247a5decf572bb61987b4
Key [ 5]:f2a1f620448b8e56665608df2ab3952f
Key [ 6]:7c84c0af02aad91dc39209c4edd220b1
round  4:793f4484fb592e7a78756fd4662f990d
Key [ 7]:f6445b647317e7e493bb92bf6655342f
Key [ 8]:3cae503567c63d3595eb140ce60a84c0
round  5:9e46a8df925916a342f299a8306220a0
Key [ 9]:734ed5a806e072bbecb4254993871679
Key [10]:cda69ccb4b07f65e3c8547c11c0647b8
round  6:6bf9cd82c9e1be13fc58eae0b936c75a
Key [11]:c48e531d3175c2bd26fa25cc8990e394
Key [12]:6d93d349a6c6e9ff5b26149565b13d15
round  7:e96a9871471240f198811d4b8311e9a6
Key [13]:5c4951e85875d663526092cd4cbdb667
Key [14]:f19f7758f5cde86c3791efaf563b3fd0
round  8:e94ca67d3721d5fb08ec069191801a46
Key [15]:bf0c17f3299b37d984ac938b769dd394
Key [16]:7edf4ad772a6b9048588f97be25bde1c
Key [17]:6ee7ba6afefc5b561abbd8d6829e8150
Ka      :ac0daabf17732f632e34ef193658bf5d
-----------------------------------------
rand    :1453db4d057654e8eb62d7d62ec3608c
PIN length = 2 octets
PIN     :3eaf
address :411fbbb51d1e
round  1:1453db4d057654e8eb62d7d62ec36084
Key [ 1]:3eaf411fbbb51d1e3eaf411fbbb51d1e
Key [ 2]:c3a1a997509f00fb4241aba607109c64
round  2:0b78276c1ebc65707d38c9c5fa1372bd
Key [ 3]:3c729833ae1ce7f84861e4dbad6305cc
Key [ 4]:c83a43c3a66595cb8136560ed29be4ff
round  3:23f3f0f6441563d4c202cee0e5cb2335
added ->:3746cbbb418bb73c2964a536cb8e83b1
Key [ 5]:18b26300b86b70acdd1c8f5cbc7c5da8
Key [ 6]:04efc75309b98cd8f1cef5513c18e41e
round  4:c61afa90d3c14bdf588320e857afdc00
Key [ 7]:517c789cecadc455751af73198749fb8
Key [ 8]:fd9711f913b5c844900fa79dd765d0e2
```

```
round  5:a8a0e02ceb556af8bfa321789801183a
Key [ 9]:bb5cf30e7d3ceb930651b1d16ee92750
Key [10]:3d97c7862ecab42720e984972f8efd28
round  6:0b58e922438d224db34b68fca9a5ea12
Key [11]:4ce730344f6b09e449dcdb64cd466666
Key [12]:38828c3a56f922186adcd9b713cdcc31
round  7:b90664c4ac29a8b4bb26debec9ffc5f2
Key [13]:d30fd865ea3e9edcff86a33a2c319649
Key [14]:1fdb63e54413acd968195ab6fa424e83
round  8:6934de3067817cefd811abc5736c163b
Key [15]:a16b7c655bbaa262c807cba8ae166971
Key [16]:7903dd68630105266049e23ca607cda7
Key [17]:888446f2d95e6c2d2803e6f4e815ddc9
Ka       :1674f9dc2063cc2b83d3ef8ba692ebef
-----------------------------------------
rand     :1313f7115a9db842fcedc4b10088b48d
PIN length = 1 octets
PIN      :6d
address :008aa9be62d5
round  1:1313f7115a9db842fcedc4b10088b48a
Key [ 1]:6d008aa9be62d56d008aa9be62d56d00
Key [ 2]:46ebfeafb6657b0a1984a8dc0893accf
round  2:839b23b83b5701ab095bafd162ec0ac7
Key [ 3]:8e15595edcf058af62498ee3c1dc6098
Key [ 4]:dd409c3444e94b9cc08396ae967542a0
round  3:c0a2010cc44f2139427f093f4f97ae68
added ->:d3b5f81d9eecd97bbe6ccd8e4f1f62e2
Key [ 5]:487deff5d519f6a6481e947b926f633c
Key [ 6]:5b4b6e3477ed5c2c01f6e607d3418963
round  4:1a5517a0efad3575931d8ea3bee8bd07
Key [ 7]:34b980088d2b5fd6b6a2aceeda99c9c4
Key [ 8]:e7d06d06078acc4ecdbc8da800b73078
round  5:d3ce1fdfe716d72c1075ff37a8a2093f
Key [ 9]:7d375bad245c3b757380021af8ecd408
Key [10]:14dac4bc2f4dc4929a6cceec47f4c3a3
round  6:47e90cb55be6e8dd0f583623c2f2257b
Key [11]:66cfda3c63e464b05e2e7e25f8743ad7
Key [12]:77cfccda1ad380b9fdf1df10846b50e7
round  7:f866ae6624f7abd4a4f5bd24b04b6d43
Key [13]:3e11dd84c031a470a8b66ec6214e44cf
Key [14]:2f03549bdb3c511eea70b65ddbb08253
round  8:02e8e17cf8be4837c9c40706b613dfa8
Key [15]:e2f331229ddfcc6e7bea08b01ab7e70c
Key [16]:b6b0c3738c5365bc77331b98b3fba2ab
Key [17]:f5b3973b636119e577c5c15c87bcfd19
Ka       :38ec0258134ec3f08461ae5c328968a1

=========================================
```

9.5 FOUR TESTS OF E3

```
rand      :00000000000000000000000000000000
aco       :48afcdd4bd40fef76693b113
key       :00000000000000000000000000000000
round   1:00000000000000000000000000000000
Key [ 1]:00000000000000000000000000000000
Key [ 2]:4697b1baa3b7100ac537b3c95a28ac64
round   2:78d19f9307d2476a523ec7a8a026042a
Key [ 3]:ecabaac66795580df89af66e66dc053d
Key [ 4]:8ac3d8896ae9364943bfebd4969b68a0
round   3:600265247668dda0e81c07bbb30ed503
Key [ 5]:5d57921fd5715cbb22c1be7bbc996394
Key [ 6]:2a61b8343219fdfb1740e6511d41448f
round   4:d7552ef7cc9dbde568d80c2215bc4277
Key [ 7]:dd0480dee731d67f01a2f739da6f23ca
Key [ 8]:3ad01cd1303e12a1cd0fe0a8af82592c
round   5:fb06bef32b52ab8f2a4f2b6ef7f6d0cd
Key [ 9]:7dadb2efc287ce75061302904f2e7233
Key [10]:c08dcfa981e2c4272f6c7a9f52e11538
round   6:b46b711ebb3cf69e847a75f0ab884bdd
Key [11]:fc2042c708e409555e8c147660ffdfd7
Key [12]:fa0b21001af9a6b9e89e624cd99150d2
round   7:c585f308ff19404294f06b292e978994
Key [13]:18b40784ea5ba4c80ecb48694b4e9c35
Key [14]:454d54e5253c0c4a8b3fcca7db6baef4
round   8:2665fadb13acf952bf74b4ab12264b9f
Key [15]:2df37c6d9db52674f29353b0f011ed83
Key [16]:b60316733b1e8e70bd861b477e2456f1
Key [17]:884697b1baa3b7100ac537b3c95a28ac
round   1:5d3ecb17f26083df0b7f2b9b29aef87c
Key [ 1]:e9e5dfc1b3a79583e9e5dfc1b3a79583
Key [ 2]:7595bf57e0632c59f435c16697d4c864
round   2:de6fe85c5827233fe22514a16f321bd8
Key [ 3]:e31b96afcc75d286ef0ae257cbbc05b7
Key [ 4]:0d2a27b471bc0108c6263aff9d9b3b6b
round   3:7cd335b50d09d139ea6702623af85edb
added ->:211100a2ff6954e6e1e62df913a656a7
Key [ 5]:98d1eb5773cf59d75d3b17b3bc37c191
Key [ 6]:fd2b79282408ddd4ea0aa7511133336f
round   4:991dccb3201b5b1c4ceff65a3711e1e9
Key [ 7]:331227756638a41d57b0f7e071ee2a98
Key [ 8]:aa0dd8cc68b406533d0f1d64aabacf20
round   5:18768c7964818805fe4c6ecae8a38599
Key [ 9]:669291b0752e63f806fce76f10e119c8
Key [10]:ef8bdd46be8ee0277e9b78adef1ec154
round   6:82f9aa127a72632af43d1a17e7bd3a09
Key [11]:f3902eb06dc409cfd78384624964bf51
Key [12]:7d72702b21f97984a721c99b0498239d
round   7:1543d7870bf2d6d6efab3cbf62dca97d
Key [13]:532e60bceaf902c52a06c2c283ecfa32
Key [14]:181715e5192efb2a64129668cf5d9dd4
```

```
round  8:eee3e8744a5f8896de95831ed837ffd5
Key [15]:83017c1434342d4290e961578790f451
Key [16]:2603532f365604646ff65803795ccce5
Key [17]:882f7c907b565ea58dae1c928a0dcf41
kc      :cc802aecc7312285912e90af6a1e1154
---------------------------------------------
rand    :950e604e655ea3800fe3eb4a28918087
aco     :68f4f472b5586ac5850f5f74
key     :34e86915d20c485090a6977931f96df5
round  1:950e604e655ea3800fe3eb4a28918087
Key [ 1]:34e86915d20c485090a6977931f96df5
Key [ 2]:8de2595003f9928efaf37e5229935bdb
round  2:d46f5a04c967f55840f83d1cdb5f9afc
Key [ 3]:46f05ec979a97cb6ddf842ecc159c04a
Key [ 4]:b468f0190a0a83783521deae8178d071
round  3:e16edede9cb6297f32e1203e442ac73a
Key [ 5]:8a171624dedbd552356094daaadcf12a
Key [ 6]:3085e07c85e4b99313f6e0c837b5f819
round  4:805144e55e1ece96683d23366fc7d24b
Key [ 7]:fe45c27845169a66b679b2097d147715
Key [ 8]:44e2f0c35f64514e8bec66c5dc24b3ad
round  5:edbaf77af070bd22e9304398471042f1
Key [ 9]:0d534968f3803b6af447eaf964007e7b
Key [10]:f5499a32504d739ed0b3c547e84157ba
round  6:0dab1a4c846aef0b65b1498812a73b50
Key [11]:e17e8e456361c46298e6592a6311f3fb
Key [12]:ec6d14da05d60e8abac807646931711f
round  7:1e7793cac7f55a8ab48bd33bc9c649e0
Key [13]:2b53dde3d89e325e5ff808ed505706ae
Key [14]:41034e5c3fb0c0d4f445f0cf23be79b0
round  8:3723768baa78b6a23ade095d995404da
Key [15]:e2ca373d405a7abf22b494f28a6fd247
Key [16]:74e09c9068c0e8f1c6902d1b70537c30
Key [17]:767a7f1acf75c3585a55dd4a428b2119
round  1:39809afb773efd1b7510cd4cb7c49f34
Key [ 1]:1d0d48d485abddd3798b483a82a0f878
Key [ 2]:aed957e600a5aed5217984dd5fef6fd8
round  2:6436ddbabe92655c87a7d0c12ae5e5f6
Key [ 3]:fee00bb0de89b6ef0a289696a4faa884
Key [ 4]:33ce2f4411db4dd9b7c42cc586b8a2ba
round  3:cec690f7e0aa5f063062301e049a5cc5
added ->:f7462a0c97e85c1d4572fd52b35efbf1
Key [ 5]:b5116f5c6c29e05e4acb4d02a46a3318
Key [ 6]:ff4fa1f0f73d1a3c67bc2298abc768f9
round  4:dcdfe942e9f0163fc24a4718844b417d
Key [ 7]:5453650c0819e001e48331ad0e9076e0
Key [ 8]:b4ff8dda778e26c0dce08349b81c09a1
round  5:265a16b2f766afae396e7a98c189fda9
Key [ 9]:f638fa294427c6ed94300fd823b31d10
Key [10]:1ccfa0bd86a9879b17d4bc457e3e03d6
round  6:628576b5291d53d1eb8611c8624e863e
Key [11]:0eaee2ef4602ac9ca19e49d74a76d335
```

```
Key [12]:6e1062f10a16e0d378476da3943842e9
round  7:d7b9c2e9b2d5ea5c27019324cae882b3
Key [13]:40be960bd22c744c5b23024688e554b9
Key [14]:95c9902cb3c230b44d14ba909730d211
round  8:97fb6065498385e47eb3df6e2ca439dd
Key [15]:10d4b6e1d1d6798aa00aa2951e32d58d
Key [16]:c5d4b91444b83ee578004ab8876ba605
Key [17]:1663a4f98e2862eddd3ec2fb03dcc8a4
kc      :c1beafea6e747e304cf0bd7734b0a9e2
----------------------------------------
rand    :6a8ebcf5e6e471505be68d5eb8a3200c
aco     :658d791a9554b77c0b2f7b9f
key     :35cf77b333c294671d426fa79993a133
round  1:6a8ebcf5e6e471505be68d5eb8a3200c
Key [ 1]:35cf77b333c294671d426fa79993a133
Key [ 2]:c4524e53b95b4bf2d7b2f095f63545fd
round  2:ade94ec585db0d27e17474b58192c87a
Key [ 3]:c99776768c6e9f9dd3835c52cea8d18a
Key [ 4]:f1295db23823ba2792f21217fc01d23f
round  3:da8dc1a10241ef9e6e069267cd2c6825
Key [ 5]:9083db95a6955235bbfad8aeefec5f0b
Key [ 6]:8bab6bc253d0d0c7e0107feab728ff68
round  4:e6665ca0772ceecbc21222ff7be074f8
Key [ 7]:2fa1f4e7a4cf3ccd876ec30d194cf196
Key [ 8]:267364be247184d5337586a19df8bf84
round  5:a857a9326c9ae908f53fee511c5f4242
Key [ 9]:9aef21965b1a6fa83948d107026134c7
Key [10]:d2080c751def5dc0d8ea353cebf7b973
round  6:6678748a1b5f21ac05cf1b117a7c342f
Key [11]:d709a8ab70b0d5a2516900421024b81e
Key [12]:493e4843805f1058d605c8d1025f8a56
round  7:766c66fe9c460bb2ae39ec01e435f725
Key [13]:b1ed21b71daea03f49fe74b2c11fc02b
Key [14]:0e1ded7ebf23c72324a0165a698c65c7
round  8:396e0ff7b2b9b7a3b35c9810882c7596
Key [15]:b3bf4841dc92f440fde5f024f9ce8be9
Key [16]:1c69bc6c2994f4c84f72be8f6b188963
Key [17]:bb7b66286dd679a471e2792270f3bb4d
round  1:45654f2f26549675287200f07cb10ec9
Key [ 1]:1e2a5672e66529e4f427b0682a3a34b6
Key [ 2]:974944f1ce0037b1febcf61a2bc961a2
round  2:990cd869c534e76ed4f4af7b3bfbc6c8
Key [ 3]:8147631fb1ce95d624b480fc7389f6c4
Key [ 4]:6e90a2db33d284aa13135f3c032aa4f4
round  3:ceb662f875aa6b94e8192b5989abf975
added ->:8b1bb1d753fe01e1c08b2ba9f55c07bc
Key [ 5]:cbad246d24e36741c46401e6387a05f9
Key [ 6]:dcf52aaec5713110345a41342c566fc8
round  4:d4e000be5de78c0f56ff218f3c1df61b
Key [ 7]:8197537aa9d27e67d17c16b182c8ec65
Key [ 8]:d66e00e73d835927a307a3ed79d035d8
round  5:9a4603bdef954cfaade2052604bed4e4
```

```
Key [ 9]:71d46257ecc1022bcd312ce6c114d75c
Key [10]:f91212fa528379651fbd2c32890c5e5f
round  6:09a0fd197ab81eb933eece2fe0132dbb
Key [11]:283acc551591fadce821b02fb9491814
Key [12]:ca5f95688788e20d94822f162b5a3920
round  7:494f455a2e7a5db861ece816d4e363e4
Key [13]:ba574aef663c462d35399efb999d0e40
Key [14]:6267afc834513783fef1601955fe0628
round  8:37a819f91c8380fb7880e640e99ca947
Key [15]:fdcd9be5450eef0f8737e6838cd38e2b
Key [16]:8cfbd9b8056c6a1ce222b92b94319b38
Key [17]:4f64c1072c891c39eeb95e63318462e0
kc      :a3032b4df1cceba8adc1a04427224299
-----------------------------------------
rand    :5ecd6d75db322c75b6afbd799cb18668
aco     :63f701c7013238bbf88714ee
key     :b9f90c53206792b1826838b435b87d4d
round  1:5ecd6d75db322c75b6afbd799cb18668
Key [ 1]:b9f90c53206792b1826838b435b87d4d
Key [ 2]:15f74bbbde4b9d1e08f858721f131669
round  2:72abb85fc80c15ec2b00d72873ef9ad4
Key [ 3]:ef7fb29f0b01f82706c7439cc52f2dab
Key [ 4]:3003a6aecdee06b9ac295cce30dcdb93
round  3:2f10bab93a0f73742183c68f712dfa24
Key [ 5]:5fcdbb3afdf7df06754c954fc6340254
Key [ 6]:ddaa90756635579573fe8ca1f93d4a38
round  4:183b145312fd99d5ad08e7ca4a52f04e
Key [ 7]:27ca8a7fc703aa61f6d7791fc19f704a
Key [ 8]:702029d8c6e42950762317e730ec5d18
round  5:cbad52d3a026b2e38b9ae6fefffecc32
Key [ 9]:ff15eaa3f73f4bc2a6ccfb9ca24ed9c5
Key [10]:034e745246cd2e2cfc3bda39531ca9c5
round  6:ce5f159d0a1acaacd9fb4643272033a7
Key [11]:0a4d8ff5673731c3dc8fe87e39a34b77
Key [12]:637592fab43a19ac0044a21afef455a2
round  7:8a49424a10c0bea5aba52dbbffcbcce8
Key [13]:6b3fde58f4f6438843cdbe92667622b8
Key [14]:a10bfa35013812f39bf2157f1c9fca4e
round  8:f5e12da0e93e26a5850251697ec0b917
Key [15]:2228fe5384e573f48fdd19ba91f1bf57
Key [16]:5f174db2bc88925c0fbc6b5485bafc08
Key [17]:28ff90bd0dc31ea2bb479feb7d8fe029
round  1:0c75eed2b54c1cfb9ff522daef94ed4d
Key [ 1]:a21ceb92d3c027326b4de775865fe8d0
Key [ 2]:26f64558a9f0a1652f765efd546f3208
round  2:48d537ac209a6aa07b70000016c602e8
Key [ 3]:e64f9ef630213260f1f79745a0102ae5
Key [ 4]:af6a59d7cebfd0182dcca9a537c4add8
round  3:8b6d517ac893743a401b3fb7911b64e1
added ->:87e23fa87ddf90c1df10616d7eaf51ac
Key [ 5]:9a6304428b45da128ab64c8805c32452
Key [ 6]:8af4d1e9d80cb73ec6b44e9b6e4f39d8
```

```
round  4:9f0512260a2f7a5067efc35bf1706831
Key [ 7]:79cc2d138606f0fca4e549c34a1e6d19
Key [ 8]:803dc5cdde0efdbee7a1342b2cd4d344
round  5:0cfd7856edfafac51f29e86365de6f57
Key [ 9]:e8fa996448e6b6459ab51e7be101325a
Key [10]:2acc7add7b294acb444cd933f0e74ec9
round  6:2f1fa34bf352dc77c0983a01e8b7d622
Key [11]:f57de39e42182efd6586b86a90c86bb1
Key [12]:e418dfd1bb22ebf1bfc309cd27f5266c
round  7:ee4f7a53849bf73a747065d35f3752b1
Key [13]:80a9959133856586370854db6e0470b3
Key [14]:f4c1bc2f764a0193749f5fc09011a1ae
round  8:8fec6f7249760ebf69e370e9a4b80a92
Key [15]:d036cef70d6470c3f52f1b5d25b0c29d
Key [16]:d0956af6b8700888a1cc88f07ad226dc
Key [17]:1ce8b39c4c7677373c30849a3ee08794
kc      :ea520cfc546b00eb7c3a6cea3ecb39ed

========================================
```

Appendix V

BLUETOOTH AUDIO

This recommendation outlines some general guidelines for voice transmission over the Bluetooth air interface. The recommendation is formed such that a smooth audio interface of a Bluetooth terminal to other audio, electronic devices and cellular terminal equipment can be carried out.

CONTENTS

1 GENERAL AUDIO RECOMMENDATIONS

1.1 MAXIMUM SOUND PRESSURE

It is the sole responsibility of each manufacturer to design their audio products in a safe way with regards to injury to the human ear. Bluetooth doesn't specify maximum sound pressure from an audio device.

1.2 OTHER TELEPHONY NETWORK REQUIREMENTS

It is the sole responsibility of each manufacturer to design the Bluetooth audio product so that it meets the regulatory requirements of all telephony networks that it may be connected to.

1.3 AUDIO LEVELS FOR BLUETOOTH

Audio levels shall be calculated as Send Loudness Rating, SLR, and Receive Loudness Rating, RLR. The calculation methods are specified in ITU-T Recommendation P.79.

The physical test set-up for Handsets and Headsets is described in ITU-T Recommendation P.51 and P.57

The physical test set-up for speakerphones and "Vehicle handsfree systems" is specified in ITU-T Recommendation P.34.

A general equation for computation of loudness rating (LR) for telephone sets is given by ITU-T recommendations P.79 and is given by

$$LR = -\frac{10}{m}\log_{10}\left(\sum_{i=N_1}^{N_2} 10^{m(s_i - w_i)/10}\right), \qquad \text{(EQ 1)}$$

where

m is a constant (~ 0.2).

w_i = weighting coefficient (different for the various LRs).

S_i = the sensitivity at frequency F_i, of the electro-acoustic path

N_1,N_2, consecutive filter bank numbers (Art. Index: 200-4000 Hz)

(EQ 1) is used for calculating the (SLR) according to Figure 1.1:, and (RLR) according to Figure 1.2:. When calculating LRs one must only include those parts of the frequency band where an actual signal transmission can occur in order to ensure that the additive property of LRs is retained. Therefore ITU-T P.79 uses only the frequency band 200-4000 Hz in LR computations.

1.4 MICROPHONE PATH

1.4.1 SLR measurement model

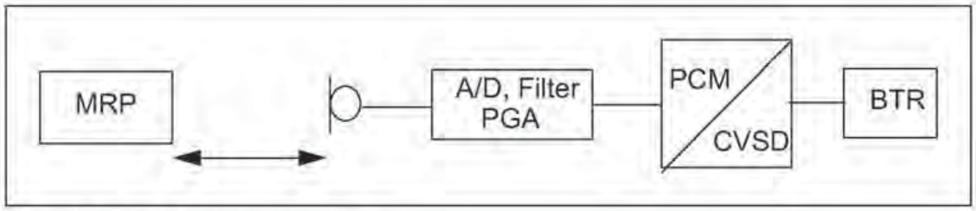


Figure 1.1: SLR measurement set-up.

1.5 LOUDSPEAKER PATH

1.5.1 RLR measurement model

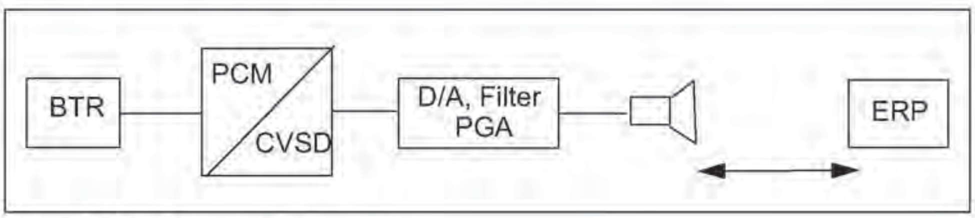


Figure 1.2: RLR measurement set-up.

1.6 BLUETOOTH VOICE INTERFACE

The specification for the Bluetooth voice interface should follow in the first place the *ITU-T Recommendations P.79*, which specifies the loudness ratings for telephone sets. These recommendations give general guidelines and specific algorithms used for calculating the loudness ratings of the audio signal with respect to Ear Reference Point (ERP).

For Bluetooth voice interface to the different cellular system terminals, loudness and frequency recommendations based on the cellular standards should be used. For example, GSM 03.50 gives recommendation for both the loudness ratings and frequency mask for a GSM terminal interconnection with Bluetooth.

1- The output of the CVSD decoder are 16-bit linear PCM digital samples, at a sampling frequency of 8 ksample/second. Bluetooth also supports 8-bit log PCM samples of A-law and μ-law type. The sound pressure at the ear reference point for a given 16-bit CVSD sample, should follow the sound pressure level given in the cellular standard specification.

2- A maximum sound pressure which can be represented by a 16-bit linear PCM sample at the output of the CVSD decoder should be specified according to the loudness rating, in ITU P.79 and at PGA value of 0 dB. Programmable Gain Amplifiers (PGAs) are used to control the audio level at the terminals by the user. For conversion between various PCM representations: A-law, μ-law and linear PCM, ITU-T G.711, G.712, G.714 give guidelines and PCM value relationships. Zero-code suppression based on ITU-T G.711 is also recommended to avoid network mismatches.

1.7 FREQUENCY MASK

For interfacing a Bluetooth terminal to a digital cellular mobile terminal, a compliance of the CVSD decoder signal to the frequency mask given in the cellular standard, is recommended to guarantee correct function of the speech coders. A recommendation for a frequency mask is given in Table 1.1. Figure 1.3: shows a plot of the frequency mask for Bluetooth (solid line). The GSM frequency mask (dotted line) is shown in Figure 1.3: for comparison.

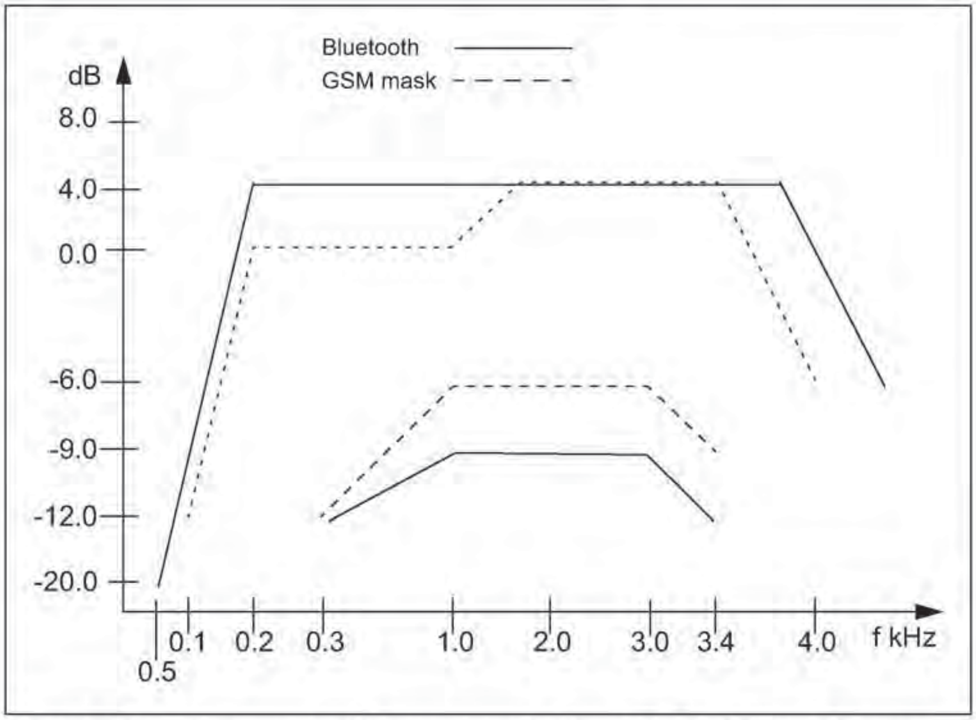


Figure 1.3: Plot of recommended frequency mask for Bluetooth. The GSM send frequency mask is given for comparison (dotted line)

Frequency (Hz)	Upper Limit (dB)	Lower Limit (dB)
50	-20	-
300	4	-12
1000	4	-9
2000	4	-9
3000	4	--9
3400	4	-12
4000	0	-

Table 1.1: Recommended Frequency Mask for Bluetooth

Appendix VI

BASEBAND TIMERS

This appendix contains a list of all timers defined the Baseband Specification.

CONTENTS

1 BASEBAND TIMERS

This appendix contains a list of all timers defined in this specification. Definitions and default values of the timers are listed below.

All timer values are given in slots.

1.1 LIST OF TIMERS

1.1.1 inquiryTO

The *inquiryTO* defines the number of slots the **inquiry** substate will last. Its value is determined by an HCI command.

1.1.2 pageTO

The *pageTO* defines the number of slots the **page** substate can last before a response is received. Its value is determined by an HCI command.

1.1.3 pagerespTO

In the slave, it defines the number of slots the slave awaits the master's response, FHS packet, after sending the page acknowledgment ID packet. In the master, *pagerespTO* defines the number of slots the master should wait for the FHS packet acknowledgment before returning to **page** substate. Both master and slave units should use the same value for this timeout, to ensure common page/scan intervals after reaching *pagerespTO*.

The *pagerespTO* default value is 8 slots.

1.1.4 inqrespTO

In the inquiry scan substate, when a device triggers on an inquiry, it waits a RAND random number of slots and returns to inquiry scan. The *inqRespTO* defines the number of slots the device will stay in the inquiry scan substate without triggering on an inquiry after the RAND wait period. The timeout value should preferably be in multiples of an inquiry train period. Upon reaching the *inqrespTO*, the device returns to **CONNECTION** or **STANDBY** state.

The *inqrespTO* default value is 128 slots.

1.1.5 newconnectionTO

Every time a new connection is started through paging, scanning, master-slave switch or unparking, the master sends a POLL packet as the first packet in the new connection. Transmission and acknowledgment of this POLL packet is used to confirm the new connection. If the POLL packet is not received by the slave or the response packet is not received by the master for *newconnectionTO* number of slots, both the master and the slave will return to the previous substate.

newconnectionTO default value is 32 slots.

1.1.6 supervisionTO

The *supervisionTO* is used by both the master and slave to monitor link loss. If a device does not receive any packets that pass the HEC check and have the proper AM_ADDR for a period of *supervisionTO*, it will reset the link *supervisionTO* will work through hold and sniff periods.

The *supervisionTO* value is determined by an HCI command. At the baseband level a default value that is equivalent to 20 seconds will be used.

Appendix VII

OPTIONAL PAGING SCHEMES

Bluetooth.

CONTENTS

1 GENERAL

For the access procedure, several paging schemes may be used. There is one mandatory paging scheme which has to be supported by all Bluetooth devices. This scheme has been described in Baseband Specification Section 10.6 on page 99 In addition to the mandatory scheme, a Bluetooth unit may support one or more optional paging schemes. The method used for page scan is indicated in the FHS payload, see Baseband Specification Section 4.4.1.4 on page 56. Three additional optional paging schemes are possible; only optional paging scheme ***I*** has been defined yet.

2 OPTIONAL PAGING SCHEME I

In this section the first optional paging scheme is described which may be used according to the rules specified in Baseband Specification Section 10 on page 95 and LMP Specification Section 3.23 on page 223. The paging code for optional scheme ***I*** is 1 (0 is used for the mandatory scheme), see also Baseband Specification Section 4.4.1.4 on page 56

The main difference between the first optional paging scheme and the mandatory scheme is the construction of the page train sent by the pager. In addition to transmission in the even master slots, the master is transmitting in the odd master slots as well. This allows the slave unit to reduce the scan window.

2.1 PAGE

The same 32 frequencies that are used for transmitting ID-packets in the mandatory paging scheme are used in the optional paging scheme ***I*** (for the construction of page trains, see Baseband Specification Section 11.3.2 on page 135. The 32 frequencies are also split into an **A-train** and **B train**. In contrast to the mandatory scheme, the same 32 frequencies that are used for transmitting are also used for reception trials, to catch the response from the addressed device.

The construction of the page train in optional page scheme ***I*** differs from the page train in the mandatory scheme in two ways:

- the page train consists of 10 slots, or 6.25 ms
- the first 8 slots of the train are used to transmit the ID packets, the 9th slot is used to send a marker packet, and the 10th slot is used for the return of a slave response

The marker packets precede the return slot, indicating the position where the slave can respond, and with which frequency. For the marker codes M_ID, bit-inverted page access codes are used. If a marker code is received at T_m with frequency f_k, a return is expected at nominally T_m+625µs at frequency f_k.

Note: The bit-inverted code M_ID to be used as marker code is beneficial for the implementation of the correlators, because the sign of the correlation peak can be used to identify the mark code during page scanning. Still, the transmitting party is uniquely identified, since inverted ID packets are not identical to the ID packets for the device with bit-wise inverted LAP.

The frequency ordering in the train and the frequencies used for the marker and receive slots change after every train. After 8 trains, all of which have a different appearance, the entire procedure is repeated. It is, therefore, more appropriate to talk about subtrains, each with length 6.25ms. Eight subtrains form a supertrain, which is repeated. An example of a supertrain with the eight subtrains is illustrated in Figure 2.1. The supertrain length is 50ms. In this example, the **A-train** is assumed with an estimated frequency of f_8; as a consequence, the frequencies selected for the train range from f_0 to f_{15}. The marker codes M_ID are indicated as **M**; the receive (half) slots are indicated as **R**.

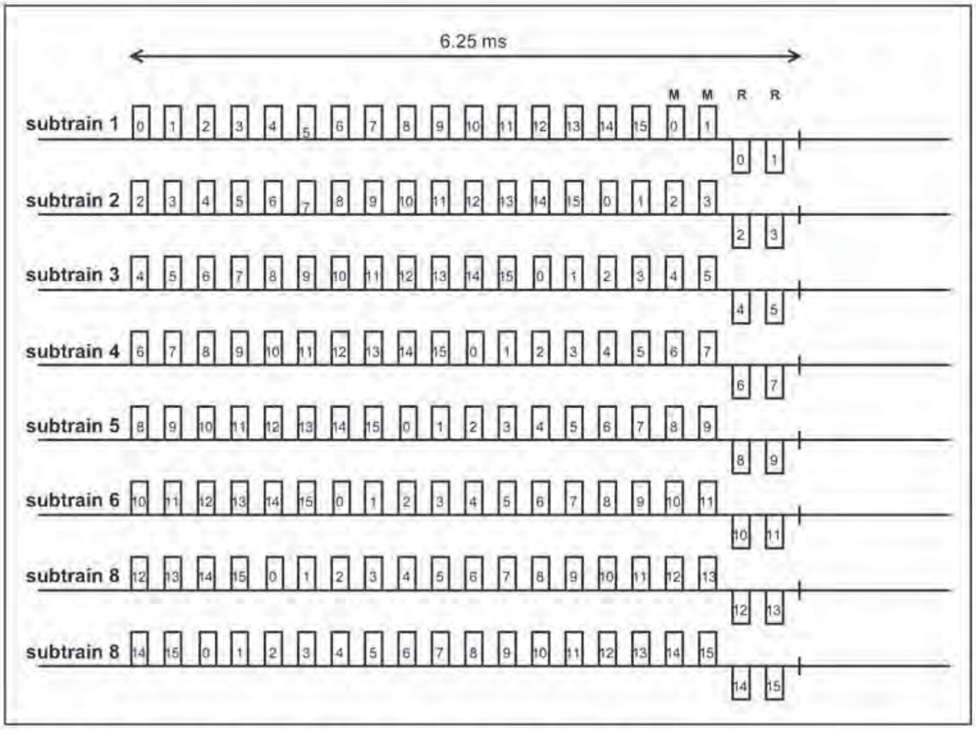


Figure 2.1: Example of train configuration for optional page scheme I.

Corresponding to the paging modes R0, R1 and R2 of the mandatory scheme, the optional scheme supports the same three modes as described for the mandatory scheme in Baseband Specification Section 10.6.2 on page 99

Since the subtrain length is now 10 slots, the 1.28s interval does not cover a multiple of (sub)trains any longer. Therefore, in contrast to the mandatory scheme, the exchange from **A-train** to **B-train** and vice versa is not based on the 1.28s interval, but instead on a multiple number of supertrains. For the R1 and R2 modes, the repetition of a supertrain N_{sup} is indicated in Table 2.1 below.

mode	No SCO link	One SCO link (HV3)	Two SCO links (HV3)
R1	N_{sup}=26	N_{sup}=52	N_{sup}=77
R2	N_{sup}=52	N_{sup}=103	N_{sup}=154

*Table 2.1: Relation between repetition duration of **A**- and **B**-trains and paging modes R1 and R2 when SCO links are present*

In accordance with the phase input to the hop selection scheme *Xp* in (EQ 4) on page 135 in the Baseband Specification (Section 11.3.2), the phase input *Xp_opt* in the optional mode is determined by:

$$Xp\_opt=[k_{offset\_opt} + \mathrm{ST}(cnt)] \bmod 32 \qquad \text{(EQ A1)}$$

where k_{offset_opt} is determined by the A/B selection and the clock estimation of the recipient:

$$k_{offset\_opt} = \begin{array}{ll} \mathrm{CLKE}_{16\text{-}12} + 24 & \text{A-train} \\ \mathrm{CLKE}_{16\text{-}12} + 8 & \text{B-train} \end{array} \qquad \text{(EQ A2)}$$

and ST is a function determining the structure of the sub- and supertrain:

$$\mathrm{ST}(cnt) = (\, cnt \bmod 160 - 2{*}\mathrm{INT}[\,((cnt \bmod 160) + 18)/20\,]\,) \bmod 16 \qquad \text{(EQ A3)}$$

k_{offset_opt} is determined once at the beginning of the repetition period. The CLKE value as is found at the beginning of the repetition interval is taken (the repetition interval being the interval in which the same supertrain is repeated all the time). As long as no train change takes place, k_{offset_opt} is not updated. *cnt* is a counter which is reset to zero at the beginning of the repetition interval and is incremented at the half-slot rate (3200 cycles/s)

The first two ID-packets of a train are transmitted in an even numbered slot.

2.2 PAGE SCAN

The basic page scanning is identical to the mandatory scheme except that a scan duration of 9.5 · 0.625 = 5.9375 ms is sufficient at the slave side.

If a device wants to scan concurrently for the mandatory and optional mode (e.g. after an inquiry response was sent), the device shall try to identify whether the paging party uses the optional scheme after an ID packet was caught. This can be done by train tracing; i.e. the device can determine whether transmission takes place in consecutive slots (optional paging scheme ***I***) or in every over slot (mandatory paging scheme), and/or whether mark codes are sent.

2.3 PAGE RESPONSE PROCEDURES

The page response procedures at the master and slave sides are almost identical to the procedures described in the mandatory mode (see Baseband Specification Section 10.6.4 on page 104). There are two differences:

- The page response routine starts after the transmission and reception of the marker code M_ID
- The ID packet sent by recipient is identical to the frequency in which the marker code was received

For the page response timing, see Figure 2.2 and Figure 2.3.

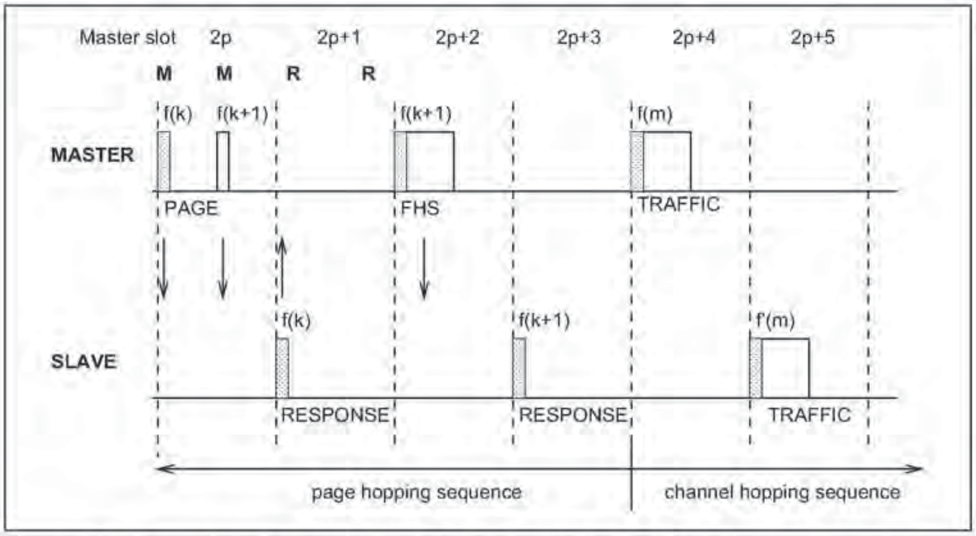


Figure 2.2: Messaging when marker code is received in first half slot of even master slot

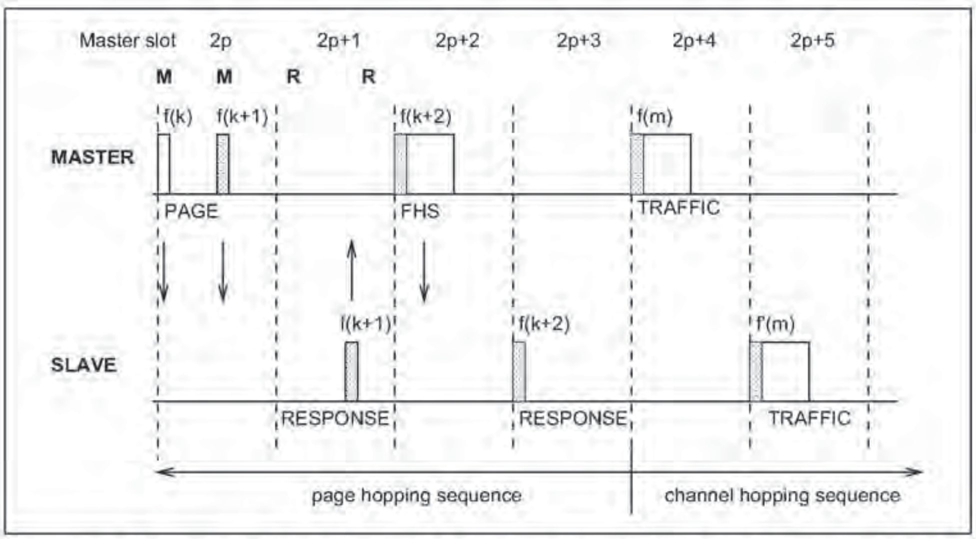


Figure 2.3: Messaging when marker code is received in second half slot of even master slot

2.4 TRAIN TRACING

This section outlines how a slave may search for the mark code although the current partitioning into A- and B-trains at the master side is not known. Train tracing means that the slave tries to receive as many page access codes from the train as possible, to catch a mark code as soon as possible. When searching for the mark codes, or trying to distinguish between the mandatory paging mode and the optional paging mode, a unit shall set up a hopping pattern for train tracing after the reception of the first access code. The hopping pattern shall ensure that the transmission and reception is performed with a 50% probability on the same frequency regardless of the actual frequency set (16 frequencies) used for paging.

Appendix VIII

BLUETOOTH ASSIGNED NUMBERS

This is a living document that lists assigned numbers, codes and identifiers in the Bluetooth standard.

CONTENTS

1 BLUETOOTH BASEBAND

1.1 THE GENERAL- AND DEVICE-SPECIFIC INQUIRY ACCESS CODES (DIACS)

The Inquiry Access Code is the first level of filtering when finding Bluetooth devices and services. The main purpose of defining multiple IACs is to limit the number of responses that are received when scanning devices within range.

#	LAP value	Usage
0	0x9E8B33	General/Unlimited Inquiry Access Code (GIAC)
1	0x9E8B00	Limited Dedicated Inquiry Access Code (LIAC)
2-63	0x9E8B01-0x9E8B32, 0x9E8B34-0x9E8B3F	RESERVED FOR FUTURE USE

Table 1.1: The Inquiry Access Codes

The Limited Inquiry Access Code (LIAC) is only intended to be used for limited time periods in scenarios where both sides have been explicitly caused to enter this state, usually by user action. For further explanation of the use of the LIAC, please refer to the Generic Access Profile [7].

In contrast it is allowed to be continuously scanning for the General Inquiry Access Code (GIAC) and respond whenever inquired.

1.2 THE CLASS OF DEVICE/SERVICE FIELD

The Class of Device/Service (CoD) field has a variable format. The format is indicated using the 'Format Type field' within the CoD. The length of the Format Type field is variable and ends with two bits different from '11'. The version field starts at the least significant bit of the CoD and may extend upwards.

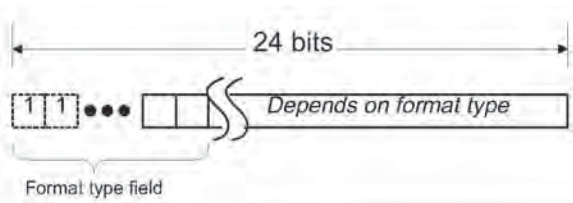


Figure 1.1: General format of Class of Device/Service

In the 'format #1' of the CoD (Format Type field = 00), 11 bits are assigned as a bit-mask (multiple bits can be set) each bit corresponding to a high level generic category of service class. Currently 7 categories are defined. These are primarily of a 'public service' nature. The remaining 11 bits are used to indicate device type category and other device-specific characteristics.

Any reserved but otherwise unassigned bits, such as in the Major Service Class field, should be set to 0.

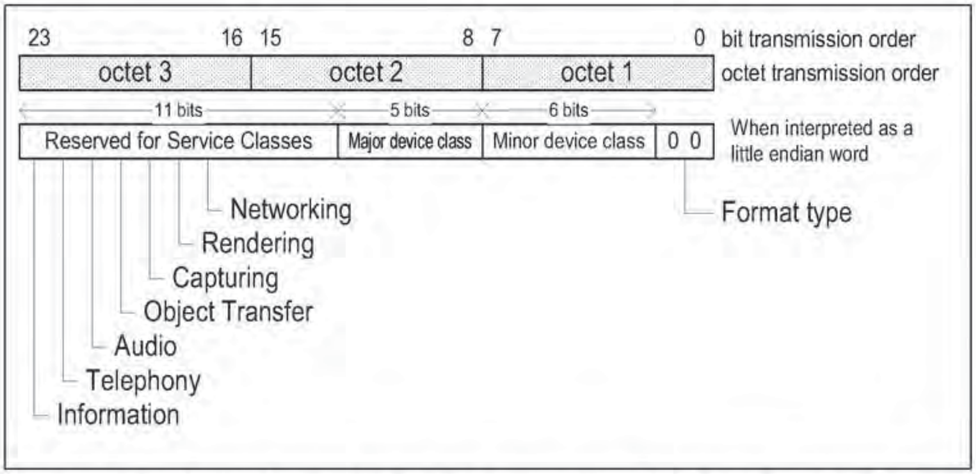


Figure 1.2: The Class of Device/Service field (format type 1). Note the order in which the octets are sent on the air and stored in memory.

1.2.1 Major Service Classes

Bit no	Major Service Class
13	Limited Discoverable Mode[1]
14	(reserved)
15	(reserved)
16	(reserved)
17	Networking (LAN, Adhoc, ...)
18	Rendering (Printing, Speaker, ...)
19	Capturing (Scanner, Microphone, ...)
20	Object Transfer (v-Inbox, v-Folder, ...)
21	Audio (Speaker, Microphone, Headset service, ...)
22	Telephony (Cordless telephony, Modem, Headset service, ...)
23	Information (WEB-server, WAP-server, ...)

Table 1.2: Major Service Classes

1. As defined in [7]

1.2.2 Major Device Classes

The Major Class segment is the highest level of granularity for defining a Bluetooth Device. The main function of a device is used to determine the major class grouping. There are 32 different possible major classes. The assignment of this Major Class field is defined in Table 1.3.

Code (bits)					Major Device Class
12	11	10	9	8	bit no of CoD
0	0	0	0	0	Miscellaneous[1]
0	0	0	0	1	Computer (desktop, notebook, PDA, organizers, ...)
0	0	0	1	0	Phone (cellular, cordless, payphone, modem, ...)
0	0	0	1	1	LAN Access Point
0	0	1	0	0	Audio (headset, speaker, stereo, ...)
0	0	1	0	1	Peripheral (mouse, joystick, keyboards, ...)
x	x	x	x	x	Range 0x06 to 0x1E reserved
1	1	1	1	1	Unclassified, specific device code not assigned

Table 1.3: Major Device Classes

1. Used where a more specific Major Device Class code is not suited (but only as specified in this document. Devices that do not have a major class code assigned can use the all-1 code until 'classified')

1.2.3 The Minor Device Class field

The 'Minor Device Class field' (bits 7 to 1 in the CoD), are to be interpreted only in the context of the Major Device Class (but independent of the Service Class field). Thus the meaning of the bits may change, depending on the value of the 'Major Device Class field'. When the Minor Device Class field indicates a device class, then the primary device class should be reported, e.g. a cellular phone that can also work as a cordless handset should use 'Cellular' in the minor device class field.

1.2.4 Minor Device Class field - Computer Major Class

Code (bits)						Minor Device Class
7	6	5	4	3	2	bit no of CoD
0	0	0	0	0	0	Unclassified, code for device not assigned
0	0	0	0	0	1	Desktop workstation
0	0	0	0	1	0	Server-class computer
0	0	0	0	1	1	Laptop
0	0	0	1	0	0	Handheld PC/PDA (clam shell)
0	0	0	1	0	1	Palm sized PC/PDA
x	x	x	x	x	x	Range 0x06-0x7F reserved

Table 1.4: Sub Device Class field for the 'Computer' Major Class

1.2.5 Minor Device Class field - Phone Major Class

Code (bits)						Minor Device Class
7	6	5	4	3	2	bit no of CoD
0	0	0	0	0	0	Unclassified, code not assigned
0	0	0	0	0	1	Cellular
0	0	0	0	1	0	Cordless
0	0	0	0	1	1	Smart phone
0	0	0	1	0	0	Wired modem or voice gateway
x	x	x	x	x	x	Range 0x05-0x7F reserved

Table 1.5: Sub Device Classes for the 'Phone' Major Class

1.2.6 Minor Device Class field - LAN Access Point Major Class

Code (bits)			Minor Device Class
7	6	5	bit no of CoD
0	0	0	Fully available
0	0	1	1-17% utilized
0	1	0	17 - 33% utilized
0	1	1	33 - 50% utilized
1	0	0	50 - 67% utilized
1	0	1	67 - 83% utilized
1	1	0	83 - 99% utilized
1	1	1	No Service Available[1]

Table 1.6: The LAN Access Point Load Factor field

1. "Device is fully utilized and cannot accept additional connections at this time, please retry later"

The exact loading formula is not standardized. It is up to each LAN Access Point implementation to determine what internal conditions to report as a utilization percentage. The only requirement is that the number reflects an ever-increasing utilization of communication resources within the box. As a recommendation, a client that locates multiple LAN Access Points should attempt to connect to the one reporting the lowest load.

Code (bits)			Minor Device Class
4	3	2	bit no of CoD
0	0	0	Unclassified (use this value if no other apply)
x	x	x	range 0x01-0x0F reserved

Table 1.7: Reserved sub-field for the LAN Access Point

1.2.7 Minor Device Class field - Audio Major Class

Code (bits)						**Minor Device Class**
7	6	5	4	3	2	bit no of CoD
0	0	0	0	0	0	Unclassified, code not assigned
0	0	0	0	0	1	Device conforms to the Headset profile [9]
x	x	x	x	x	x	Range 0x02-0x7F reserved

Table 1.8: Sub Device Classes for the 'Audio' Major Class

2 LINK MANAGER PROTOCOL (LMP)

2.1 THE LINK MANGER VERSION PARAMETER

Parameter name	Assigned values	
VersNr	0	Bluetooth LMP 1.0, [2]
	1-255	(reserved)

Table 2.1: The LMP Version Parameter Values

2.2 THE LMP_COMPID PARAMETER CODES

This is the parameter used in the LMP Version procedure.

Code	Company
0	Ericsson Mobile Communications
1	Nokia Mobile Phones
2	Intel Corp.
3	IBM Corp.
4	Toshiba Corp.
5 - 65534	(reserved)
65535	Unassigned. For use in internal and interoperability tests before a Company ID has been assigned. May not be used in products.

Table 2.2: The LMP_CompId parameter codes

3 LOGICAL LINK CONTROL AND ADAPTATION PROTOCOL (L2CAP)

Please see Section 4.3 for assigned PSM values.

3.1 CHANNEL IDENTIFIERS

Destination CID	Protocol/usage	Reference
0x0000	Illegal, should not be used	[3]
0x0001	L2CAP signalling channel	[3]
0x0002	L2CA connection less data	[3]
0x0003 - 0x003F	(reserved)	

Table 3.1: Pre-defined L2CAP Channel Identifiers

3.2 PROTOCOL AND SERVICE MULTIPLEXOR (PSM)

Protocol	PSM	Reference
SDP	0x0001	[4]
RFCOMM	0x0003	[5]
TCS-BIN	0x0005	[6]
TCS-BIN-CORDLESS	0x0007	[6]

Table 3.2: Assigned Protocol and Service Multiplexor values (PSM)

4 SERVICE DISCOVERY PROTOCOL (SDP)

4.1 UNIVERSALLY UNIQUE IDENTIFIER (UUID) SHORT FORMS

The Bluetooth Service Discovery Protocol (SDP) specification defines a way to represent a range of UUIDs (which are nominally 128-bits) in a shorter form. A *reserved* range of 2^{32} values can be represented using 32-bits (denoted uuid32). Of these, a sub-range of 2^{16} values can be represented using only 16-bits (denoted uuid16). Any value in the 2^{32} range that is not assigned in this document is reserved pending future revisions of this document. In other words, no value in this range may be used except as specified in this or future revisions of this document. UUID values outside of this range can be allocated as described in [19] for any purpose the allocater desires.

4.2 BASE UNIVERSALLY UNIQUE IDENTIFIER (UUID)

The Base UUID is used for calculating 128-bit UUIDs from 'short UUIDs' (uuid16 and uuid32) as described in the SDP Specification [4].

Mnemonic	UUID
BASE_UUID	00000000-0000-1000-8000-00805F9B34FB

4.3 PROTOCOLS

Mnemonic	UUID	Name	Ref.
SDP	uuid16: *0x0001*[1]	sdp.bt	[4]
RFCOMM	uuid16: *0x0003*	com.bt	[5]
TCS-BIN	uuid16: *0x0005*	tcs.bt	[6]
L2CAP	uuid16: *0x0100*		[3]
IP	uuid16: *0x0009*		
UDP	uuid16: *0x0002*		
TCP	uuid16: *0x0004*		
TCS-AT	uuid16: *0x0006*	modem	
OBEX	uuid16: *0x0008*	obex	
FTP	uuid16: *0x000A*	ftp	
HTTP	uuid16: *0x000C*	http	
WSP	uuid16: *0x000E*	wsp	

Table 4.1: Protocol Universally Unique Identifiers and Names

1. 'Short UUID'

4.4 SERVICE CLASSES

Mnemonic	UUID	Profile[1]	AbstractName
ServiceDiscoveryServerServiceClassID	uuid16: *0x1000*		
BrowseGroupDescriptorServiceClassID	uuid16: *0x1001*		
PublicBrowseGroup	uuid16: *0x1002*		
SerialPort	uuid16: *0x1101*	[7]	serial.bt
LANAccessUsingPPP	uuid16: *0x1102*		
DialupNetworking	uuid16: *0x1103*	[13]	
IrMCSync	uuid16: *0x1104*	[17]	
OBEXObjectPush	uuid16: *0x1105*	[16]	
OBEXFileTransfer	uuid16: *0x1106*	[15]	
IrMCSyncCommand	uuid16: *0x1107*	[17]	
Headset	uuid16: *0x1108*	[7]	headset
CordlessTelephony	uuid16: *0x1109*	[10]	
Intercom	uuid16: *0x1110*	[11]	
Fax	uuid16: *0x1111*	[12]	
HeadsetAudioGateway	uuid16: 0x1112	[7]	
PnPInformation	uuid16: *0x1200*		
GenericNetworking	uuid16: *0x1201*	n/a	
GenericFileTransfer	uuid16: *0x1202*	n/a	
GenericAudio	uuid16: *0x1203*	n/a	
GenericTelephony	uuid16: *0x1204*	n/a	

Table 4.2: Service Class Identifiers and Names

1. If the specified Service Class directly and exactly implies a certain Profile, the Profile is indicated here (i.e. for concrete Service Classes). Leave empty for abstract Service Classes.

The Profile column in Table 4.2 indicates which Service Class identifiers that also directly corresponds to a Bluetooth Profile. It is not allowed to use the Service Class UUID unless the service complies with the specified Profile. These UUIDs might also appear as Profile Identifiers in the BluetoothProfileDescriptorList attribute.

4.5 ATTRIBUTE IDENTIFIER CODES

Mnemonic	Attribute ID	Reference
ServiceRecordHandle	0x0000	[4] *Bluetooth Service Discovery Protocol (SDP)*, Bluetooth SIG
ServiceClassIDList	0x0001	
ServiceRecordState	0x0002	
ServiceID	0x0003	
ProtocolDescriptorList	0x0004	
BrowseGroupList	0x0005	
LanguageBaseAttributeIDList	0x0006	
ServiceInfoTimeToLive	0x0007	
ServiceAvailability	0x0008	
BluetoothProfileDescriptorList	0x0009	
DocumentationURL	0x000A	
ClientExecutableURL	0x000B	
Icon10	0x000C	
IconURL	0x000D	
Reserved	0x000E-0x01FF	
ServiceName	0x0000 + b[1]	
ServiceDescription	0x0001 + b	
ProviderName	0x0002 + b	
VersionNumberList	0x0200	
ServiceDatabaseState	0x0201	
GroupID	0x0200	
Remote audio volume control	0x0302[2]	[7]
External network	0x0301	[10]
Service Version	0x0300	
Supported Data Stores List	0x0301	[17]
Supported Formats List	0x0303	[16]

Table 4.3: Attribute Identifiers

Mnemonic	Attribute ID	Reference
Fax Class 1 Support	0x0302	[12]
Fax Class 2.0 Support	0x0303	
Fax Class 2 Support	0x0304	
Audio Feedback Support	0x0305	

Table 4.3: Attribute Identifiers

1. 'b' in this table represents a base offset as given by the LanguageBaseAttributeIDList attribute. For the primary language, 'b' must be equal to 0x0100 as described in the SDP specification.
2. Items in *italic* are tentative values in this version of the document.

4.6 PROTOCOL PARAMETERS

Protocol	Parameter mnemonic	Index
L2CAP	PSM	1
TCP or UDP	Port	1
RFCOMM	Channel	1

Table 4.4: Protocol Parameters

4.7 HOST OPERATING ENVIRONMENT IDENTIFIERS

4.7.1 ClientExecutableURL substitution strings

The operating environment identifier strings have the following format[1]:

<cpu_type>-<manufacturer>-[<kernel>-]<os>[<version>][-<object_format>]

The general rule is that is that a new identifier should only be defined as required to differentiate incompatible operating environments concerning an executable file image. That is, for example different <version>-tags should not be used for compatible versions of the same operating system.

1. It is based on a format used by the GNU AutoConfig tools

Currently defined tags:

CPU-Type ID	Description
alpha	Digital Alpha* compatible
arm	ARM* core or compatible
i86	Any Intel* 80x86-family compatible CPU
i960	Intel* i960 compatible
jvm	Java Virtual Machine*
mips	MIPS MIPS* compatible
ppc	IBM/Motorola PowerPC* compatible
sh3	Hitachi SH-3* compatible
sh4	Hitachi SH-4* compatible
sparc	Sun Sparc* compatible
Kernel ID	**Description**
chorus, linux, javaos, os9, qnx, vxworks	
<os>	An 'OS identifier' as listed below, might appear in the <kernel> field when the requested OS platform is Java based.
OS+Version-Identifiers	
amigaos, beos4.5, ejava, epocc, epoce, epocq, epocs, gnu, jre1.1, jre1.2, macos, macosx, os2, os9, palmos, pjava, pjava1.1, photon, plan9, qnx, rtjava, win95, win98, win2000, wince, winnt4	
Object Format Identifiers[1]	
aout, bout, coff, elf, jar	
Manufacturer Identifiers	
amiga*, apple*, be*, ericsson*, ibm*, intel*, lucent*, microsoft*, microware*, motorola*, nokia*, palm*, psion*, qnx*, sun*, symbian*, toshiba*, unknown[2]	

1. Only applicable when the object format is not otherwise uniquely implied by the identifier string.
2. Use when no other applies.

For Linux, the 'manufacturer' field may be used to indicate Linux distribution if so required (in which case <version> indicates the version of the distribution). Otherwise use 'unknown'.

Linux Distribution Identifiers
caldera, debian, dlx, doslinux, linuxpro, linuxware, mandrake, mklinux, redhat, slackware, stampede, suse, turbolinux, yggdrasil

Example Operating Environment Identifier Strings		
i86-microsoft-win95	ppc-apple-macos	i86-redhat-linux-gnu6
i86-microsoft-win98	m68k-apple-macos	ppc-mklinux-linux-gnu
i86-microsoft-winnt4	ppc-apple-macosx	
alpha-microsoft-winnt4	i86-apple-macosx	
i86-microsoft-win2000	m68k-amiga-amigaos	
alpha-microsoft-win2000	ppc-amiga-amigaos	
i86-be-beos4.5	jvm-sun-jre1.2	
ppc-be-beos4.5	jvm-sun-pjava1.1	
arm-symbian-epoc3	jvm-sun-ejava	
i86-unknown-linux-gnu	m68k-palm-palmos-coff	
sh3-microsoft-wince	ppc-ibm-vxworks-pjava1.2	
arm-microsoft-wince	sparc-sun-javaos-jre1.2	

4.7.2 IconURL substitution strings

The IconURL operating environment identifier strings have the following general format:

<horizontal_pixels>x<vertical_pixels>x<color_depth>[m].<file_format>

The optional tag 'm' indicates monochrome or grayscale. The host is free to try to match/request any graphics file format as indicated by a <file_format> tag, however at a minimum files conforming to the Portable Network Graphic standard [18] should be made available at the resulting URL (indicated by <file_format>=png)[2].

File format tag	Description
png	Portable Network Graphics [18]
gif	Graphics Interchange File format
bmp	Windows bitmap

Currently defined IconURL Icon format identifier strings:

Example Icon format Identifier Strings	
32x32x8.png	256 color 32 by 32 icon (or 255 colors + transparent)
16x16x8.png	
16x16x1m.png	Black and white (or monochrome + transparent)
10x10x2m.png	4 gray-scales

2. The use of PNG, and whether a subset of PNG should be required, is currently pending further investigation.

5 REFERENCES

[1] *Bluetooth Baseband Specification*, Bluetooth SIG

[2] *Bluetooth Link Manager Specification,* Bluetooth SIG

[3] *Logical Link Control and Adaptation Protocol Specification*, Bluetooth SIG

[4] *Bluetooth Service Discovery Protocol (SDP)*, Bluetooth SIG

[5] *RFCOMM with TS 07.10,* Bluetooth SIG

[6] *Bluetooth Telephony Control Specification / TCS Binary*, Bluetooth SIG

[7] *Generic Access Profile*, Bluetooth SIG

[8] *Serial Port Profile*, Bluetooth SIG

[9] *Headset Profile*, Bluetooth SIG

[10] *Cordless Telephony Profile*, Bluetooth SIG

[11] *Intercom Profile*, Bluetooth SIG

[12] *Fax Profile*, Bluetooth SIG

[13] *Dial-up Networking Profile*, Bluetooth SIG

[14] *IrDA Interoperability*, Bluetooth SIG

[15] *File Transfer Profile*, Bluetooth SIG

[16] *Object Push Profile*, Bluetooth SIG

[17] *Synchronization Profile*, Bluetooth SIG

[18] *Portable Network Graphics (PNG)*, http://www.w3.org/Graphics/PNG

[19] *UUIDs and GUIDs*, P. J. Leach et al, http://www.ietf.org/internet-drafts/draft-leach-uuids-guids-01.txt

6 TERMS AND ABBREVIATIONS

LMP	Link Management Protocol
L2CA	Logical Link Control and Adaptation, protocol multiplexer layer for Bluetooth
MTU	Maximum Transmission Unit
SAP	Service Access Points
Baseband	Baseband Protocol
Service Discovery	The ability to discover the capability of connecting devices or hosts.
PnP	Plug and Play
SAR	Segmentation and Reassembly
IP	Internet Protocol
IrDA	InfraRed Data Association
PPP	Point-to-Point Protocol
IETF	Internet Engineering Task Force
RFC	Request For Comments

7 LIST OF FIGURES

8 LIST OF TABLES

Appendix IX

MESSAGE SEQUENCE CHARTS

Between Host and Host Controller/Link Manager

This document shows examples of inter-working between HCI Commands and LM Protocol Data Units in form of message sequence charts. It helps to understand and to correctly use the HCI Commands.

Bluetooth.

CONTENTS

1 INTRODUCTION

The goal of this document is to show the interworkings of HCI-Commands and LM-PDUs. It focuses on the message sequence charts for the procedures specified in [3] "Bluetooth Host Controller Interface Functional Specification" with regard to LM Procedures from [2] "Link Manager Protocol".

We illustrate here the most useful scenarios, but we do not cover all possible alternatives. Furthermore, the message sequence charts do not consider the transfer error over Air Interface or Host Interface. In all message sequence charts it is assumed that all events are not masked, so the Host Controller will not filter out any events.

Notation used in the message sequence charts:

Box:

- Replaces a group of transactions
- Indicates the start of a procedure or a sub-scenario

Note: in a message sequence chart where several sub-scenarios exist, the sub-scenarios can be executed optionally, consequently, exclusively or independently from each other.

Hexagon:

- Indicates a condition that is needed to start the transaction below this hexagon

Arrow:

- Represents a message, signal or transaction

Comment:

- "/* ... */" indicates editor comments

2 SERVICES WITHOUT CONNECTION REQUEST

2.1 REMOTE NAME REQUEST

The service Remote Name Request is used to find out the name of the remote BT Device without an explicit ACL Connection request.

Sending an HCI_Remote_Name_Request (BD_ADDR, Page_Scan_ Repetition_Mode, Page_Scan_Mode, Clock_Offset), the Host expects that its local BT Device will automatically try to connect to the remote BT Device (with the specified BD_ADDR). Then the local BT Device should try to get the name, to disconnect, and finally to return the name of the remote BT Device back to the Host (see Figure 2.1 Remote Name Request: sub-scenario 1).

Note: if an ACL Connection already exists (see Figure 2.1 Remote Name Request: sub-scenario 2), the Remote Name Request procedure will be executed like an optional service. No Paging and no ACL Detachment need to be done.

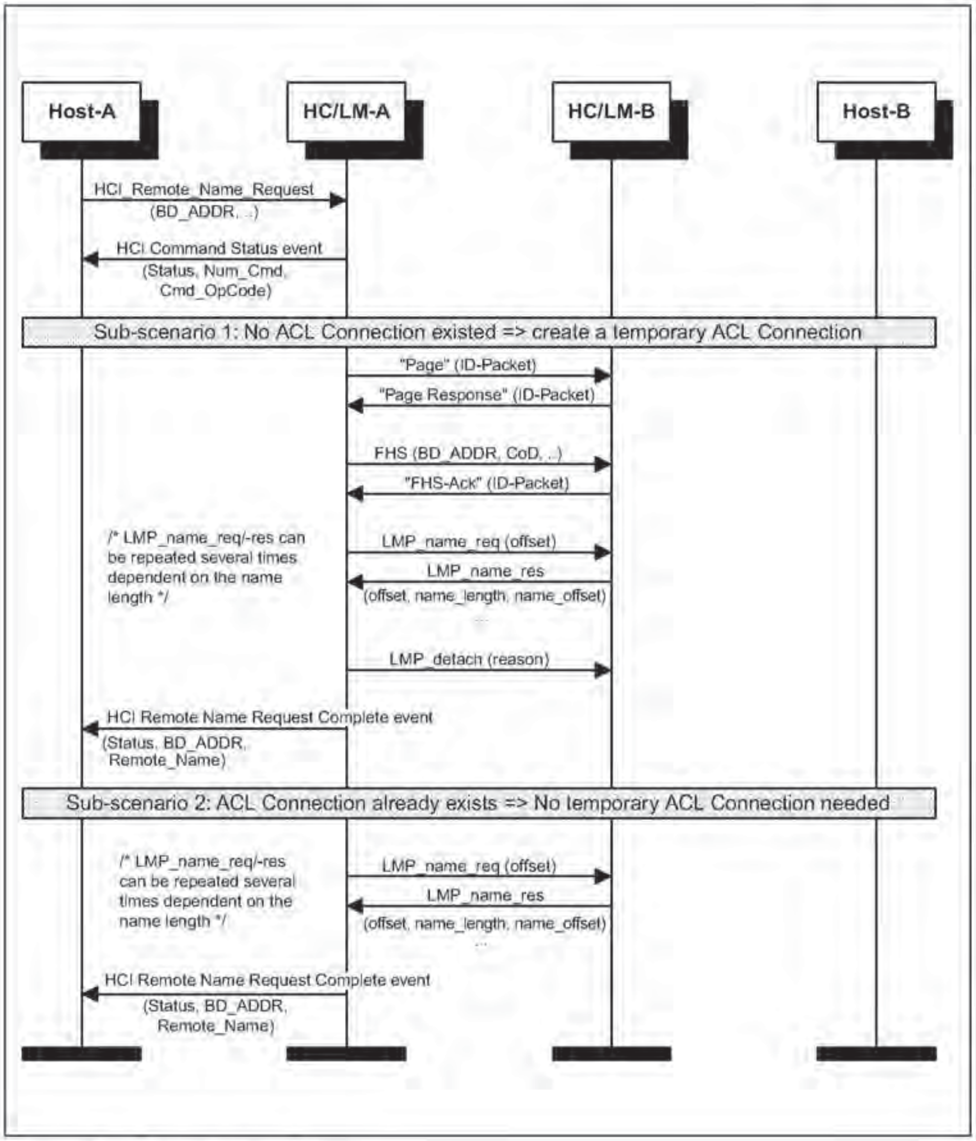


Figure 2.1: Remote Name Request

2.2 ONE-TIME INQUIRY

Inquiry is used to detect and collect nearby BT Devices. When receiving the command HCI_Inquiry (LAP, Inquiry_Length, Num_Responses), HC will start the baseband inquiry procedure with an Inquiry Access Code (derived from the specified LAP) and Inquiry Length. When Inquiry Responses are received, HC will filter out and then return the information related to the found BT Devices using one or several Inquiry Result events (Num_Responses, BD_ADDR[I], Page_Scan_Repetition_Mode[i], Page_Scan_Period_Mode[i], Page_Scan_Mode[i], Class_Of_Device[i], Clock_Offset[i]) to the Host.

The filtering of found BT Devices is specified in HCI_Set_Event_Filter (Filter_Type, Filter_Condition_Type, Condition) with the Filter_Type = Inquiry Result. When the Inquiry procedure is completed, Inquiry Complete event (Status, Num_Responses) must be returned to the Host. Otherwise, the command HCI_Inquiry_Cancel( ) will be used to directly stop the inquiry procedure.

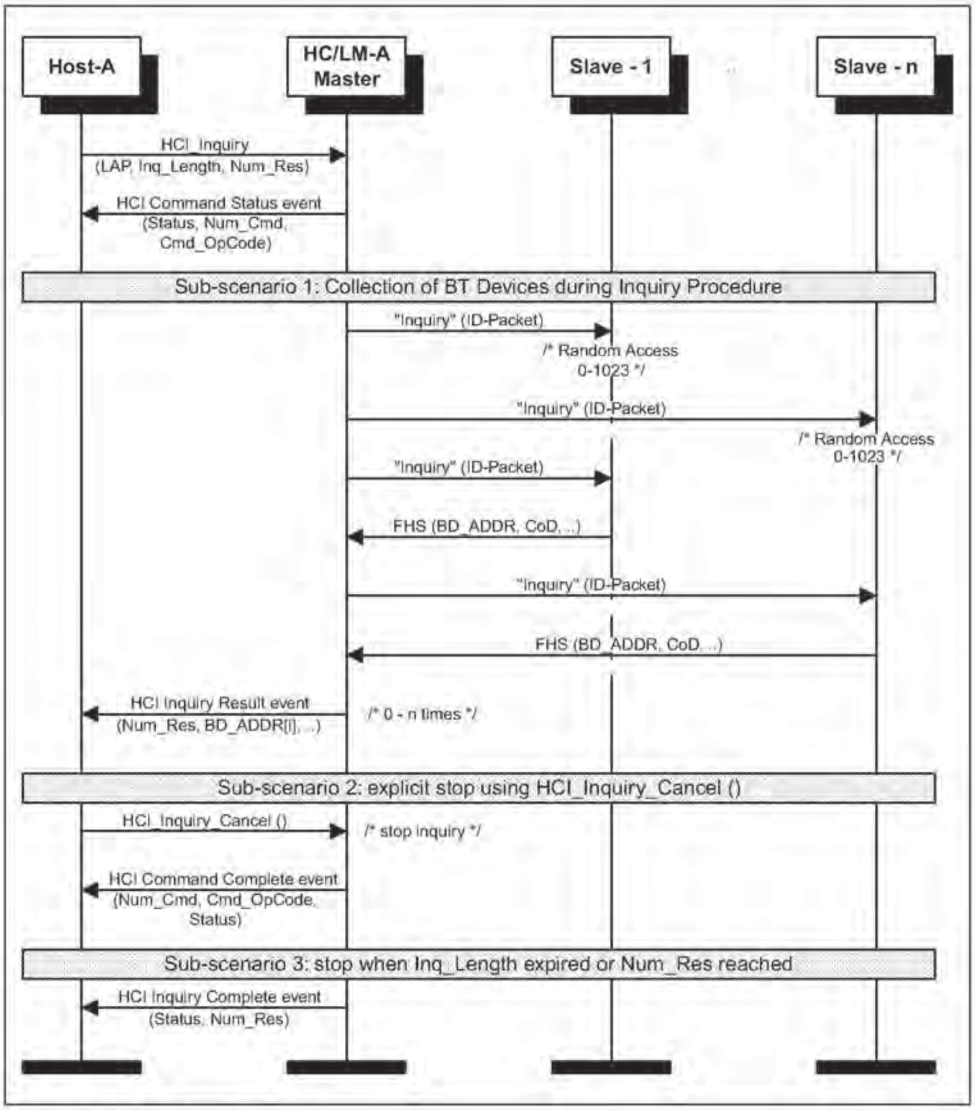


Figure 2.2: One-Time Inquiry

2.3 PERIODIC INQUIRY

Periodic inquiry is needed when the inquiry procedure is to be repeated periodically. Receipt of the command HCI_Periodic_Inquiry_Mode (Max_Period_Length, Min_Period_Length, LAP, Inquiry_Length, Num_Responses) HC will start the periodic Inquiry Mode with the specified parameters Max_Period_Length, Min_Period_Length, Inquiry_Access_code (derived from LAP) and Inquiry_Length. As in the one-time Inquiry procedure, only BT Devices that are specified in the HCI_Set_Event_Filter (Filter_Type, Filter_Condition_Type, Condition) with the Filter_Type = Inquiry Result will not be filtered out. Therefore, in the inquiry cycle, one or several Inquiry Result events (Num_Responses, BD_ADDR[i], Page_Scan_Repetition_Mode[i], Page_Scan_Period_Mode[i], Page_Scan_Mode[i], Class_Of_Device[i], Clock_Offset[i]) and Inquiry Complete event (Status, Num_Responses) will be returned to the Host with one, or a list of, found BT Devices. The periodic Inquiry can be stopped using HCI_Exit_Periodic_Inquiry_Mode( ).

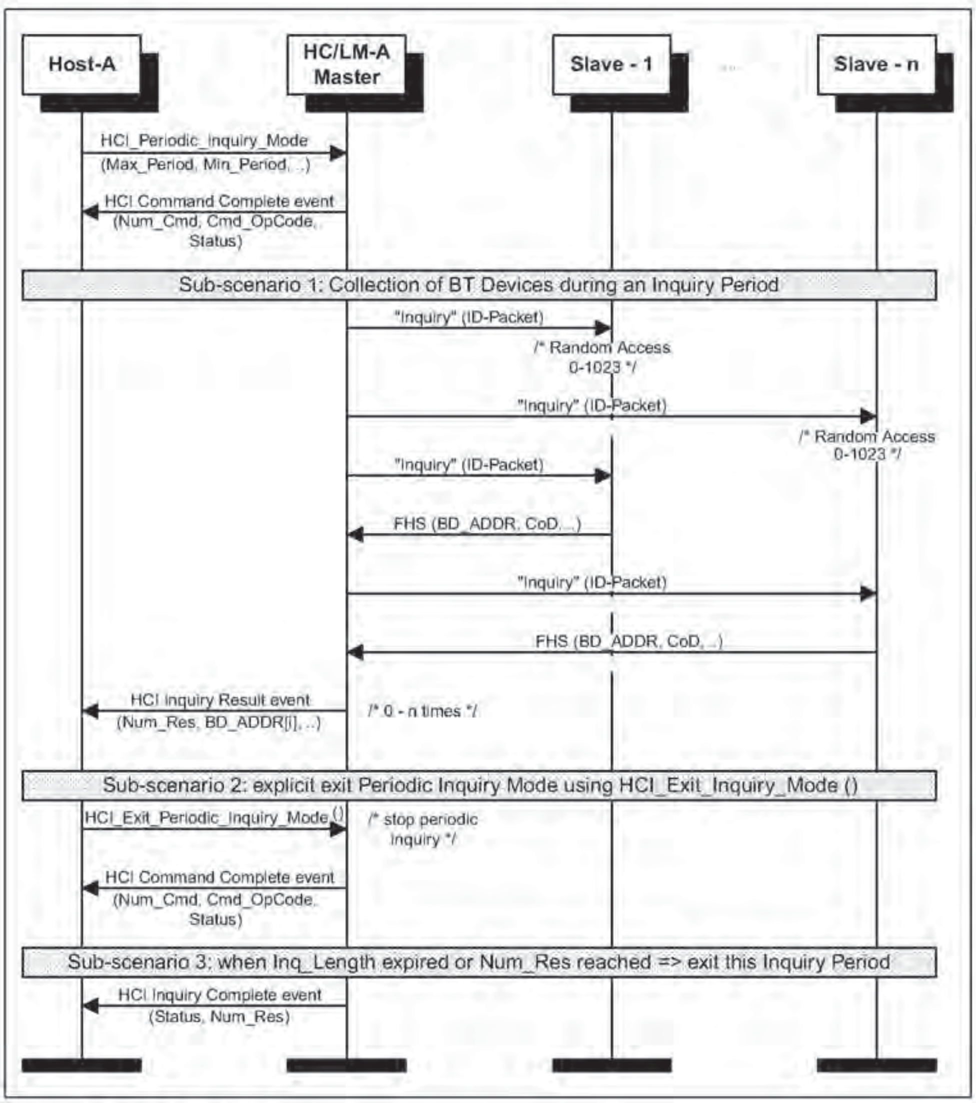


Figure 2.3: Periodic Inquiry

3 ACL CONNECTION ESTABLISHMENT AND DETACHMENT

The overview of the ACL Connection establishment and detachment is shown in Figure 3.1 Overview of ACL Connection establishment and detachment.

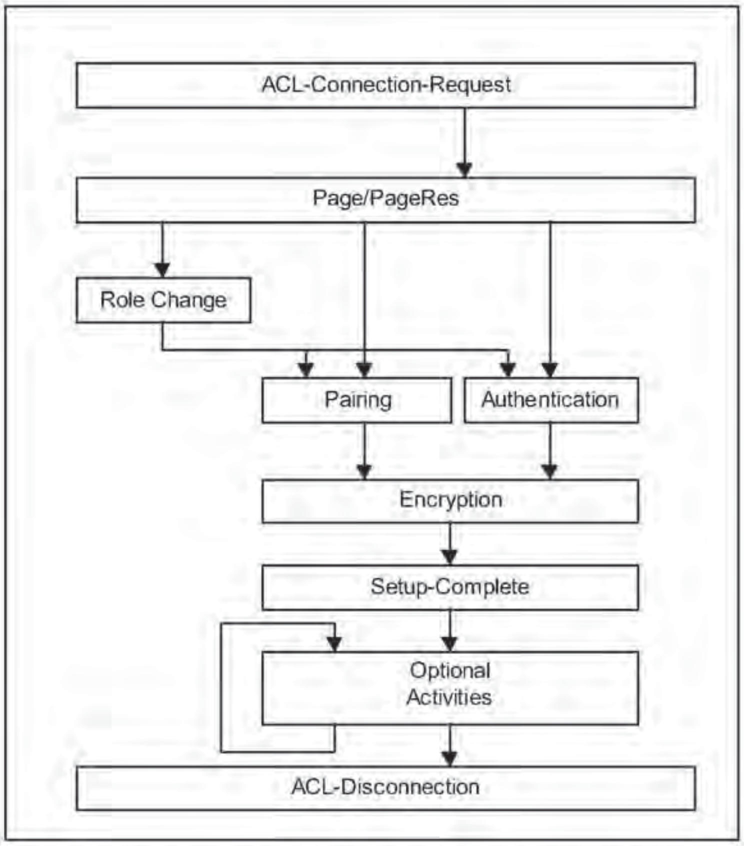


Figure 3.1: Overview of ACL Connection establishment and detachment

3.1 ACL CONNECTION REQUEST PHASE

The ACL Connection Request phase is identified between the HCI_Create_Connection (BD_ADDR, Packet_Type, Page_Scan_Repetition_Mode, Page_Scan_Mode, Clock_Offset, Allow_Role_Switch) from the master side and the response from the slave side with rejection or acceptation on the LM level. Three alternative sub-scenarios are shown in Figure 3.2, "ACL Connection Request phase," on page 1044.

Sub-scenario 1: Slave rejects ACL Connection Request

If the ACL Connection request is rejected by slave, a Connection Complete event (Status, Connection_Handle, BD_ADDR, Link_Type, Encryption_Mode) will be then returned to Host, whereby the Status will be copied from the Reason parameter of the command HCI_Reject_Connection_Request (Reason, BD_ADDR). The parameters Connection_Handle and Encryption_Mode will be meaningless.

Sub-scenario 2: Slave accepts ACL Connection Request

When the slave responds with LMP_accepted ( ) correspondent to LMP_host_connection_req ( ), the ACL Connection Request is accepted. The master will continue with the ACL Connection Setup, where pairing, authentication or encryption will be executed.

Sub-scenario 3: Slave accepts ACL Connection Request with Role Change

This case is identified when the slave sends an LMP_switch_req ( ) to initiate Role Change. If the master accepts, the baseband Master-Slave Switch will be executed. Thereafter, the ACL Connection Setup will continue.

Note: on the slave side, an incoming connection request can be automatically accepted by using HCI_Set_Event_Filter (Filter_Type, Filter_Condition_Type, Condition) with the Filter_Type = 0x02 /*Connection_Setup*/.

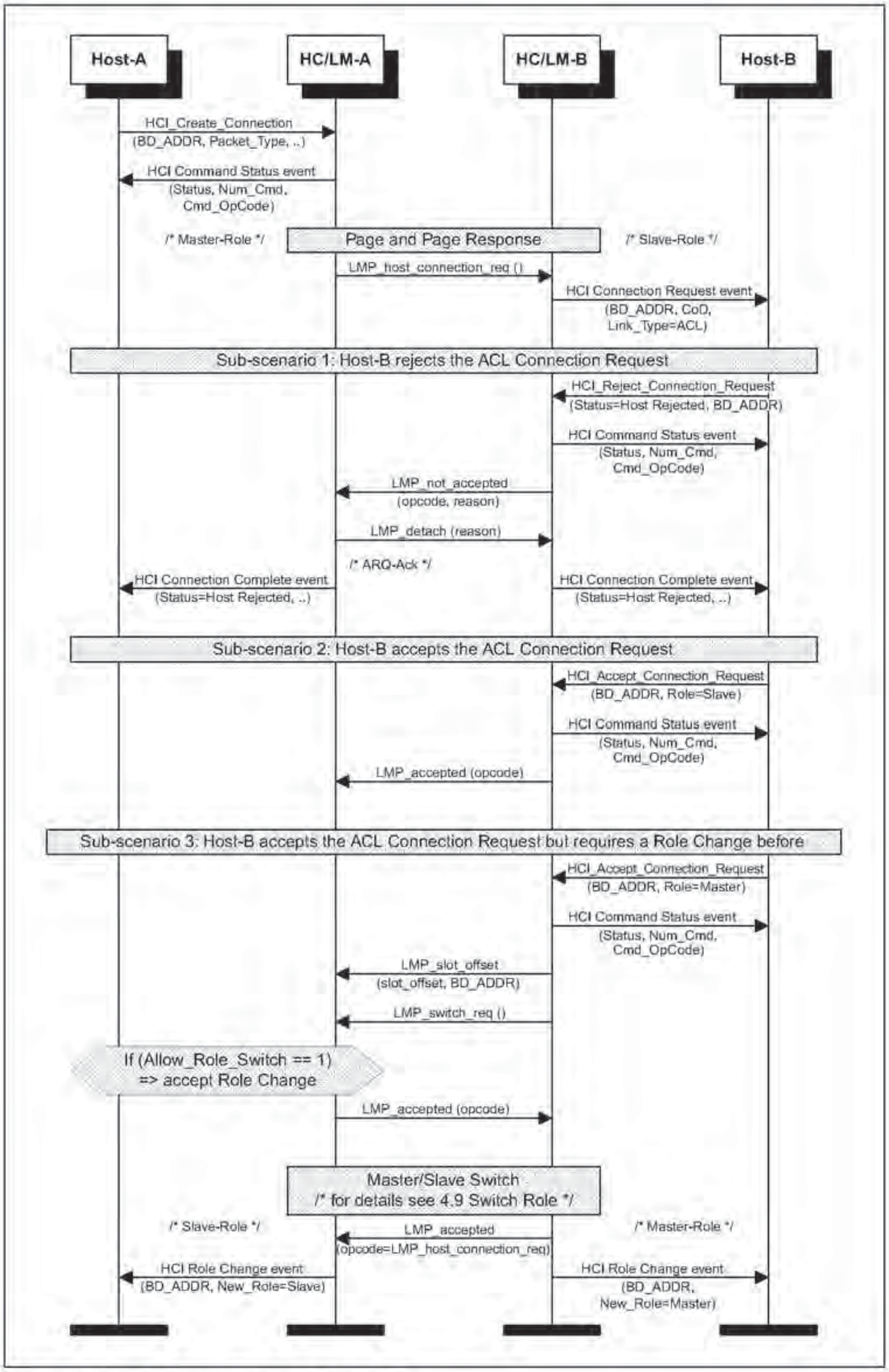


Figure 3.2: ACL Connection Request phase

3.2 ACL CONNECTION SETUP PHASE

If the ACL Connection Request phase was successful, the ACL Connection Setup phase will start, with the goal of executing security procedures like pairing, authentication and encryption. The ACL Connection Setup phase is successfully finished when LMP_setup_complete ( ) is exchanged and the Connection Complete event (Status=0x00, Connection_Handle, BD_ADDR, Link_Type, Encryption_Mode) is sent to the Host.

3.2.1 Pairing

If authentication is required and the BT Devices to be connected don't have a common link key, the pairing procedure on LM Level will be executed using the PIN Input from Host. During the pairing, the authentication- and link key creation procedures will be done. Note: the created Link Key can be stored either in the BT Device or in the Host.

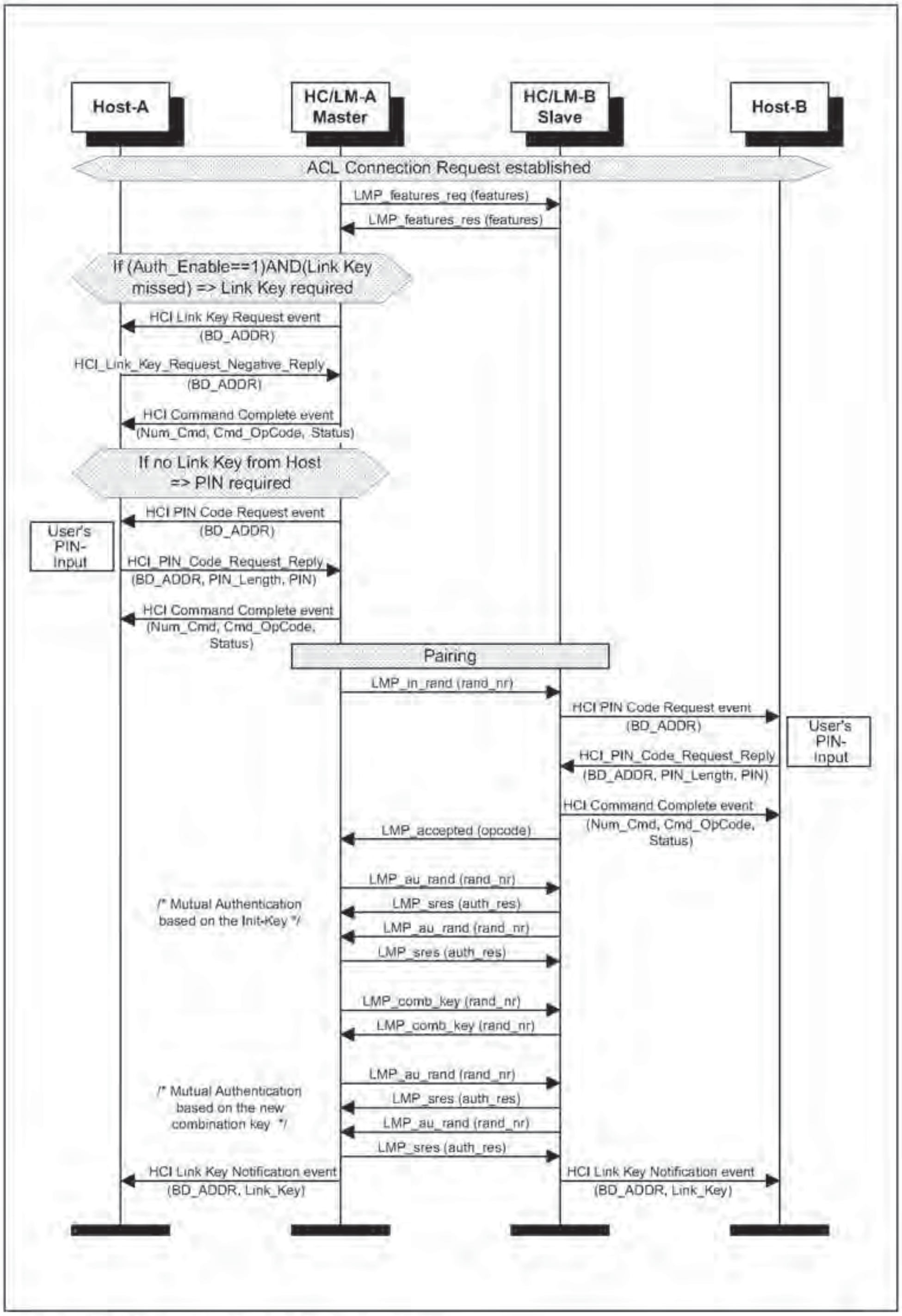


Figure 3.3: ACL Connection setup with pairing

3.2.2 Authentication

If a common link key already exists between the BT Devices, pairing is not needed. Note: a Link Key created during pairing can be stored either in the BT Device or in the Host. If the parameter Authentication_Enable is set, the authentication procedure has to be executed. Here, the MSC only shows the case when Authentication_Enable is set on both sides.

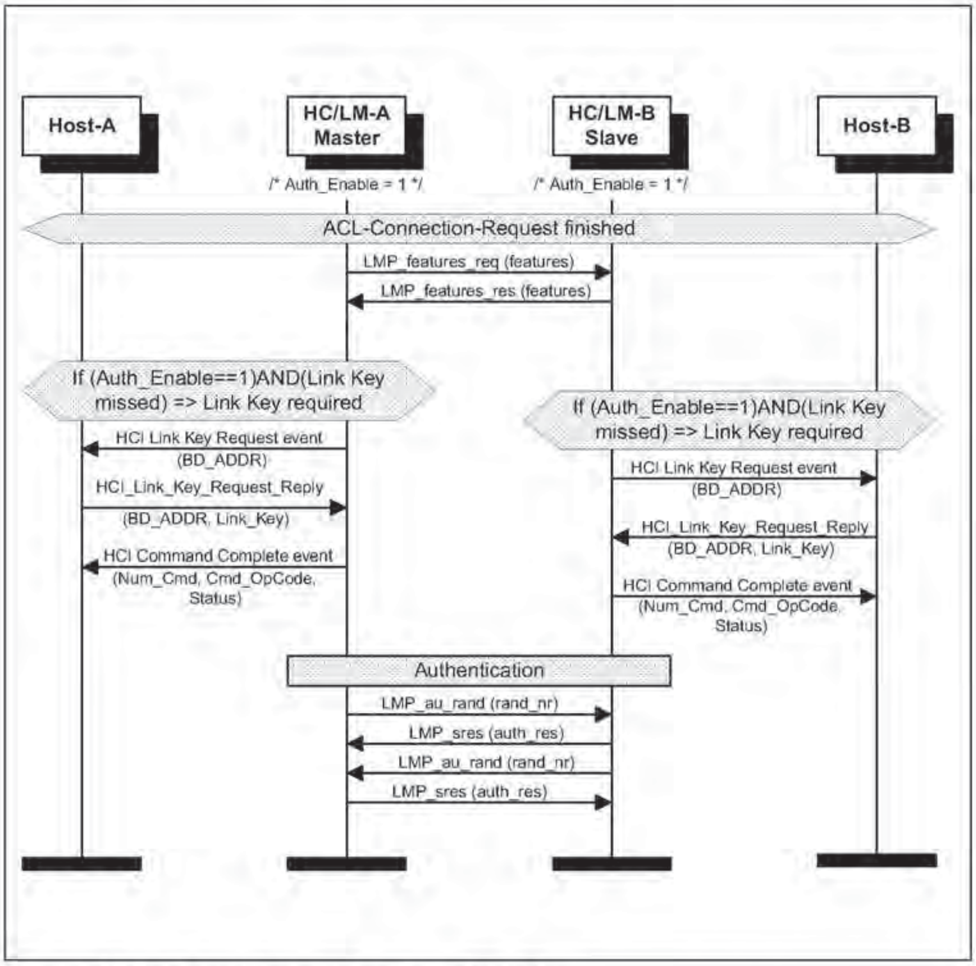


Figure 3.4: ACL Connection setup with authentication

3.3 ENCRYPTION AND CONNECTION SETUP COMPLETE

Once the pairing/authentication procedure is successful, the encryption procedure will be started. Here, the MSC only shows how to set up an encrypted point-to-point connection (Encryption_Mode = 1 /*point-to-point/). Note: an encrypted connection requires an established common link key.

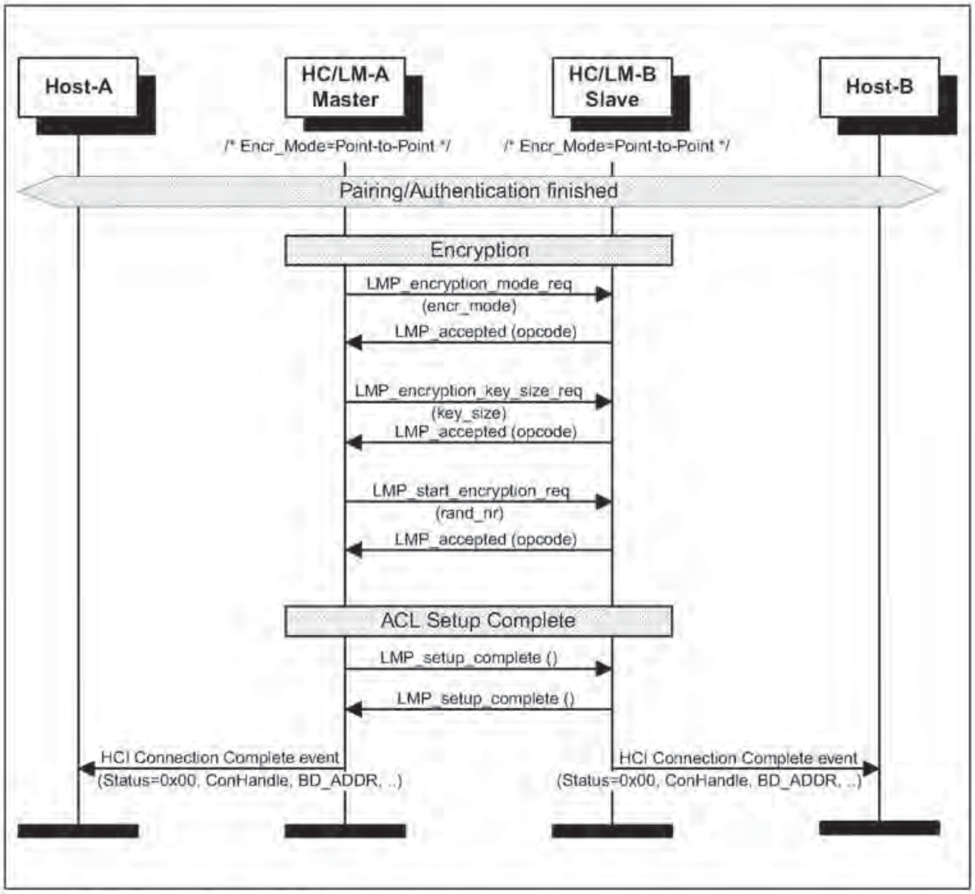


Figure 3.5: Encryption and Setup complete

3.4 ACL DISCONNECTION

At any time, an established ACL Connection can be detached by an HCI_Disconnect (Connection_Handle, Reason). If one or several SCO Connections exist, they must first be detached before the ACL Connection can be released.

Note: the disconnection procedure is one-sided and doesn't need an explicit acknowledgment from the remote LM. So the ARQ Acknowledgment from the LC is needed, to ensure that the remote LM has received the LMP_detach (reason).

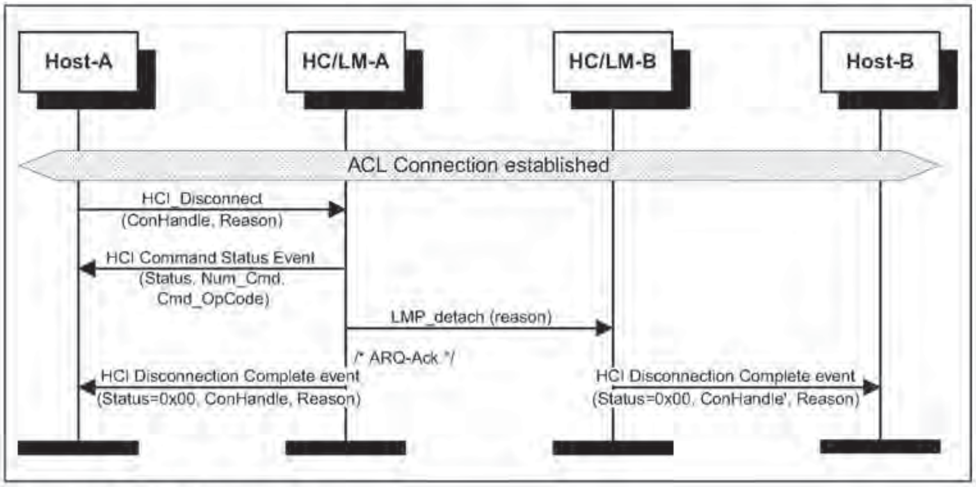


Figure 3.6: ACL Disconnection

4 OPTIONAL ACTIVITIES AFTER ACL CONNECTION ESTABLISHMENT

4.1 AUTHENTICATION REQUESTED

Authentication can be explicitly executed at any time after an ACL Connection has been established. If the Link Key was missed in HC/LM, the Link Key will be required from the Host, as in the authentication procedure (see 3.2.2).

Note: if the HC/LM and the Host don't have the Link Key a PIN Code Request event will be sent to the Host to request a PIN Code for pairing. A procedure identical to ACL Connection Setup with Pairing (see 3.2.1) will be used.

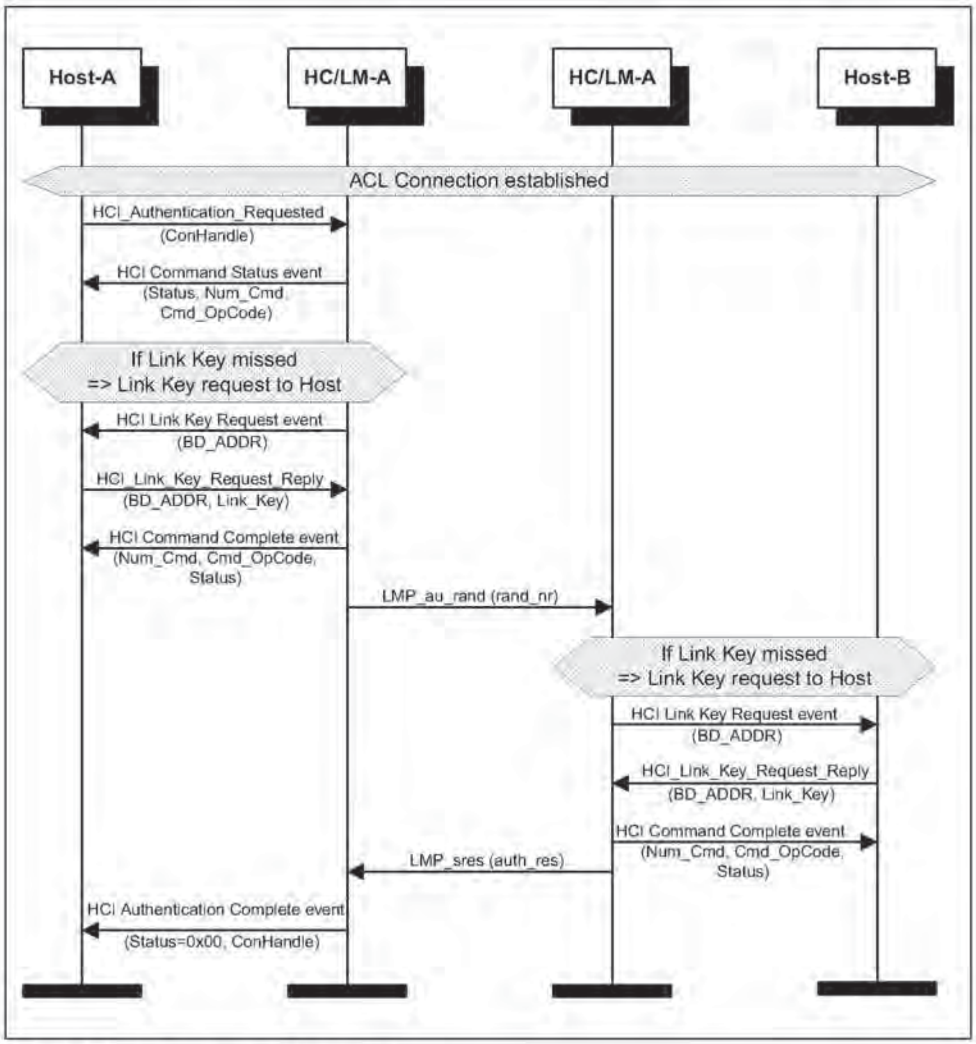


Figure 4.1: Authentication Requested

Bluetooth.

4.2 SET CONNECTION ENCRYPTION

Using the command HCI_Set_Connection_Encryption (Connection_Handle, Encryption_Enable), the Host is able to switch the encryption of a connection with the specified Connection_Handle to ON/OFF. This command can be applied on both the master- and slave sides (only the master side is shown in Figure 4.2 Set Connection Encryption). If this command occurs on the slave side, the only difference is that LMP_encryption_mode_req (encryption_mode) will be sent from the HC/LM Slave. LMP_encryption_key_size_req (key_size) and LMP_start_encryption_req (rand_nr) will still be requested from the HC/LM master.

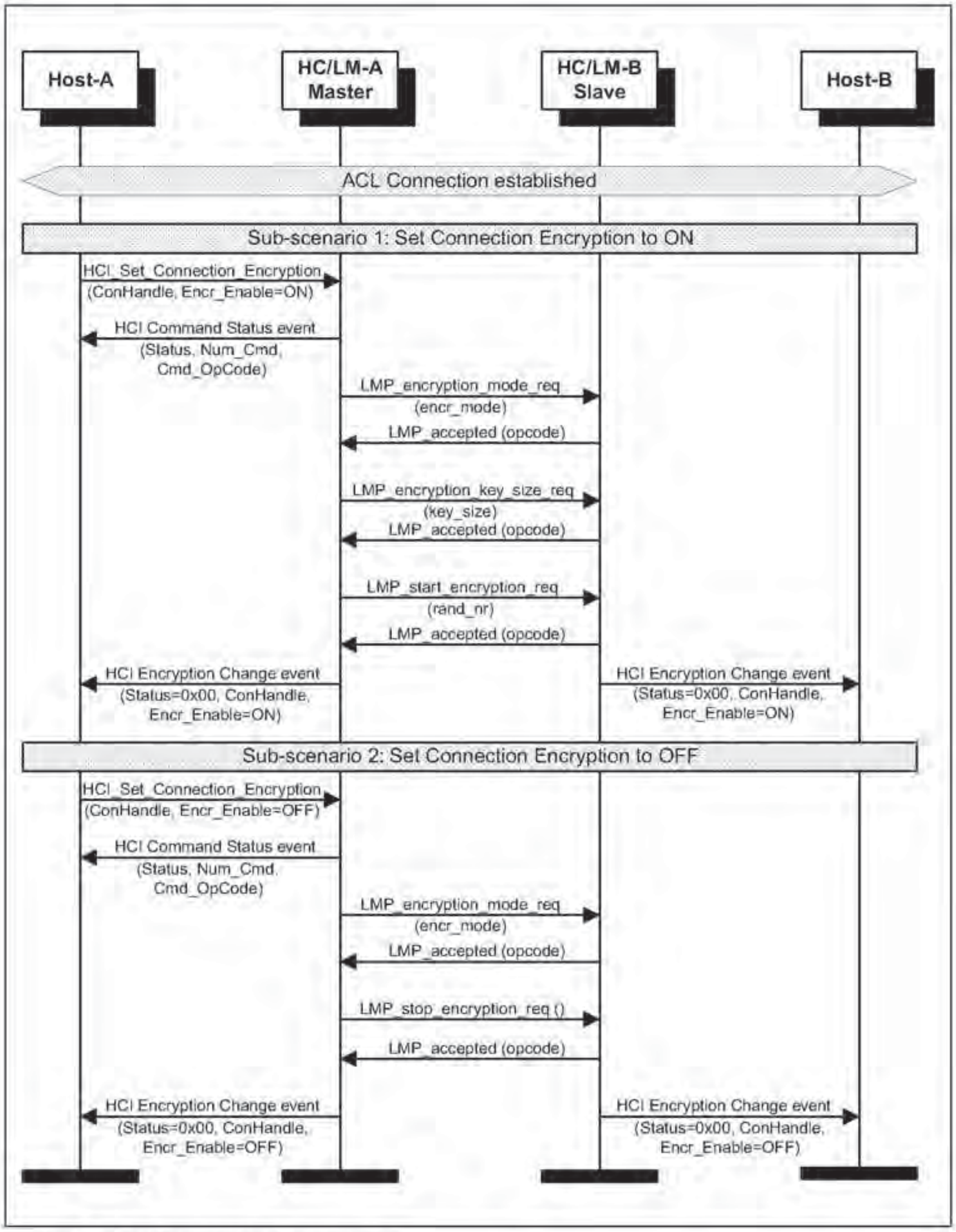


Figure 4.2: Set Connection Encryption

4.3 CHANGE CONNECTION LINK KEY

Using the command HCI_Change_Connection_Link_Key (Connection_Handle), the Host can explicitly change the common link key shared between the BT Devices.

Note: if the connection encryption was enabled and the temporary link key was used, it is the task of the BT Master to automatically restart the encryption (first stop and then restart) after the link key is successfully changed.

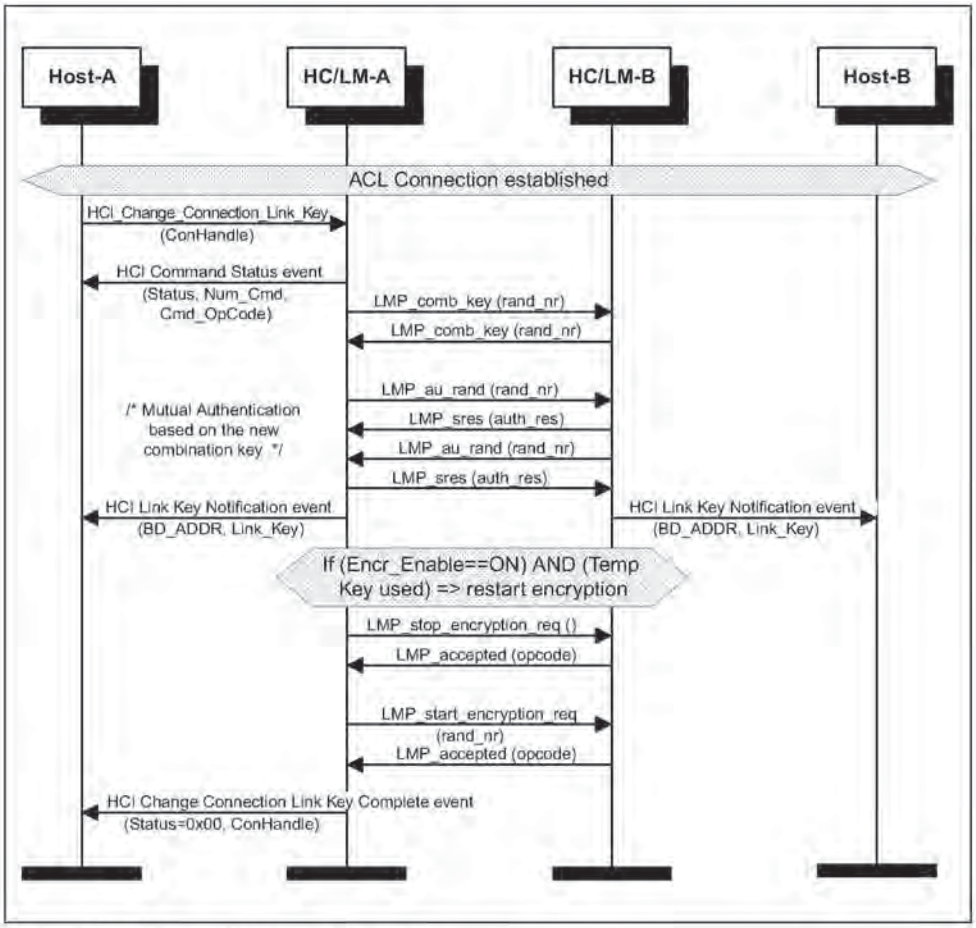


Figure 4.3: Change Connection Link Key

4.4 MASTER LINK KEY

The Figure 4.4 Master Link Key shows how the Host can explicitly switch between the temporary Link Key and the semi-permanent Link Key. Since this command can only be used for the BT Master, the Link Key switch will affect all connections.

Note: if encryption was enabled, it is the task of the BT Master to restart the encryption separately for each slave.

AFFLT0294281

Samsung Ex. 1419 p. 1053

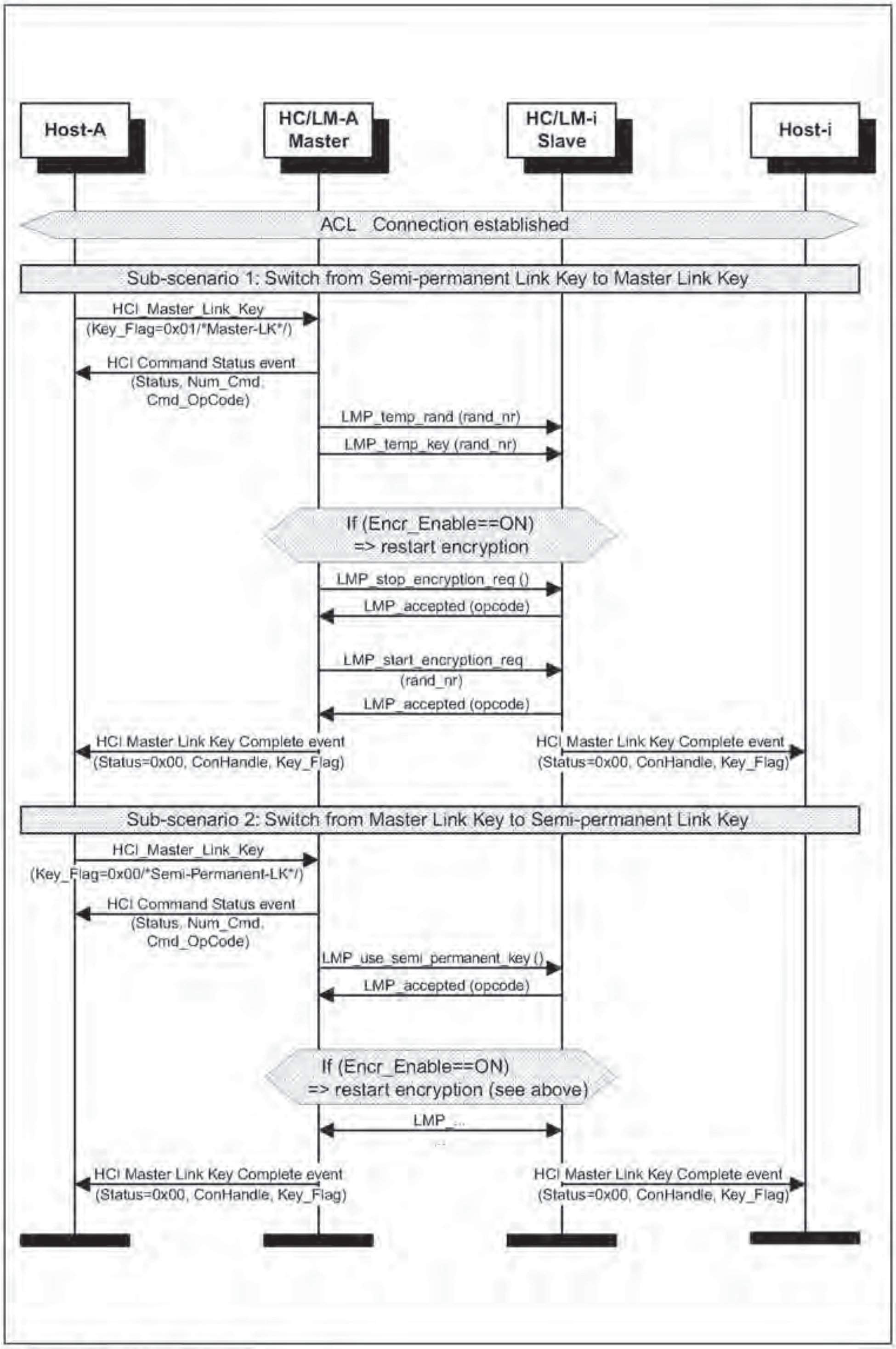


Figure 4.4: Master Link Key

4.5 READ REMOTE SUPPORTED FEATURES

Using the command HCI_Read_Remote_Supported_Features (Connection_Handle) the supported LMP Features of a remote BT Device can be read. These features contain supported packet types, supported modes, supported audio coding modes, etc.

Note: if the LMP Features was exchanged during ACL Connection Setup, the HC/LM A may return the Read Remote Supported Features Complete event (Status, Connection_Handle, LMP_Features) without exchange of LMP PDUs.

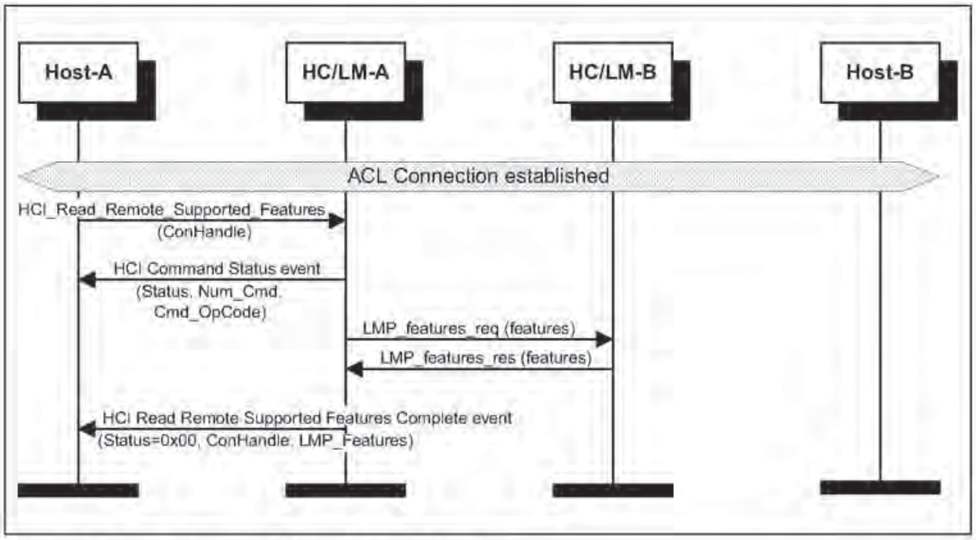


Figure 4.5: Read Remote Supported Features

4.6 READ CLOCK OFFSET

Using the command HCI_Read_Clock_Offset (Connection_Handle) the BT Master can read the Clock Offset of the BT Slaves. The Clock Offset can be used to speed up the paging procedure in a later connection attempt. If the command is requested from the slave device, the HC/LM Slave will directly return a Command Status event and an Read Clock Offset Complete event without exchange of LMP PDUs.

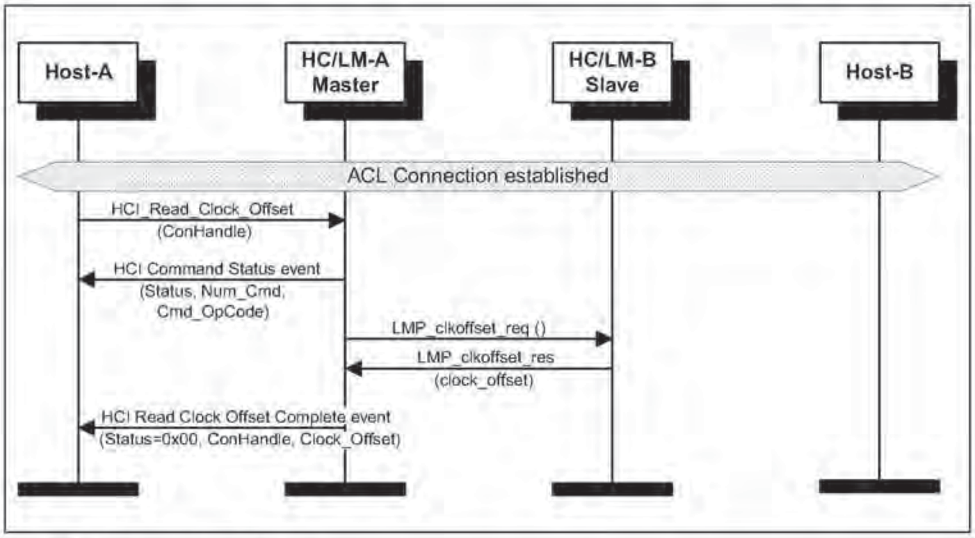


Figure 4.6: Read Clock Offset

4.7 READ REMOTE VERSION INFORMATION

Using HCI_Read_Remote_Version_Information (Connection_Handle) the version information consisting of LMP_Version, Manufacturer_Name and LMP_Subversion from the remote BT Device can be read.

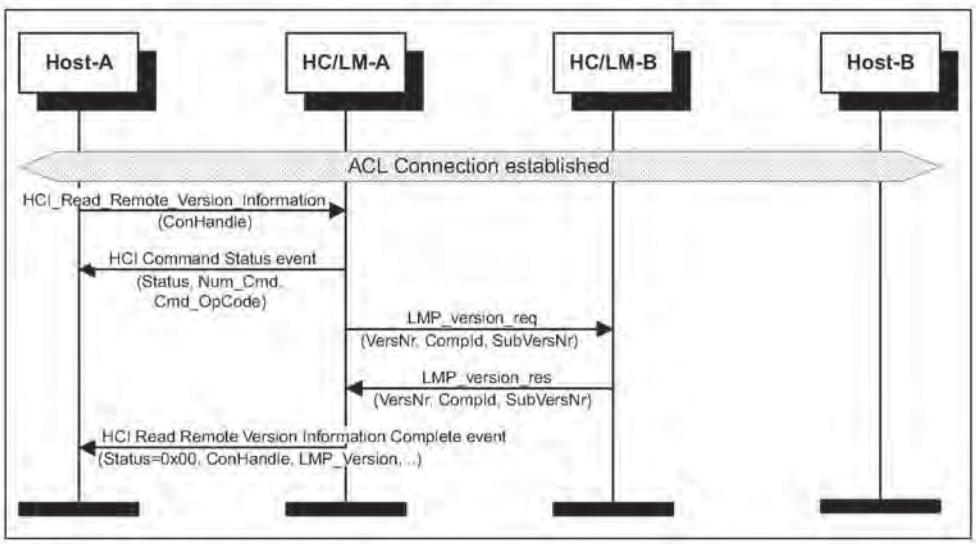


Figure 4.7: Read Remote Version Information

4.8 QOS SETUP

To set up the Quality of Service, the command HCI_QoS_Setup (Connection_Handle, Flags, Service_Type, Token_Rate, Peak_Bandwidth, Latency, Delay_Variation) is used.

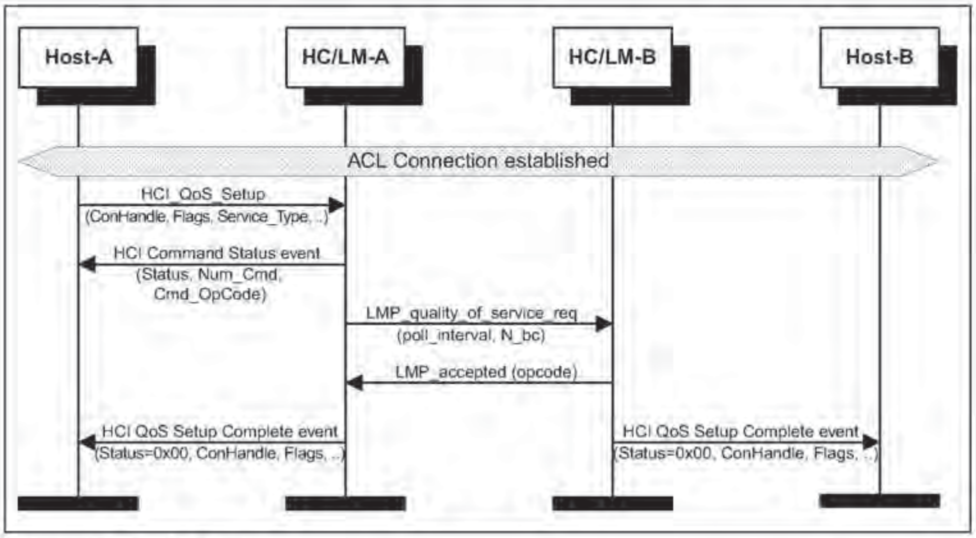


Figure 4.8: QoS Setup

4.9 SWITCH ROLE

The command HCI_Switch_Role (BD_ADDR, Role) can be used to explicitly switch the current role of the local BT Device for a particular connection with the specified BT Device (BD_ADDR). The local HC/LM has to check whether the switch is performed or not.

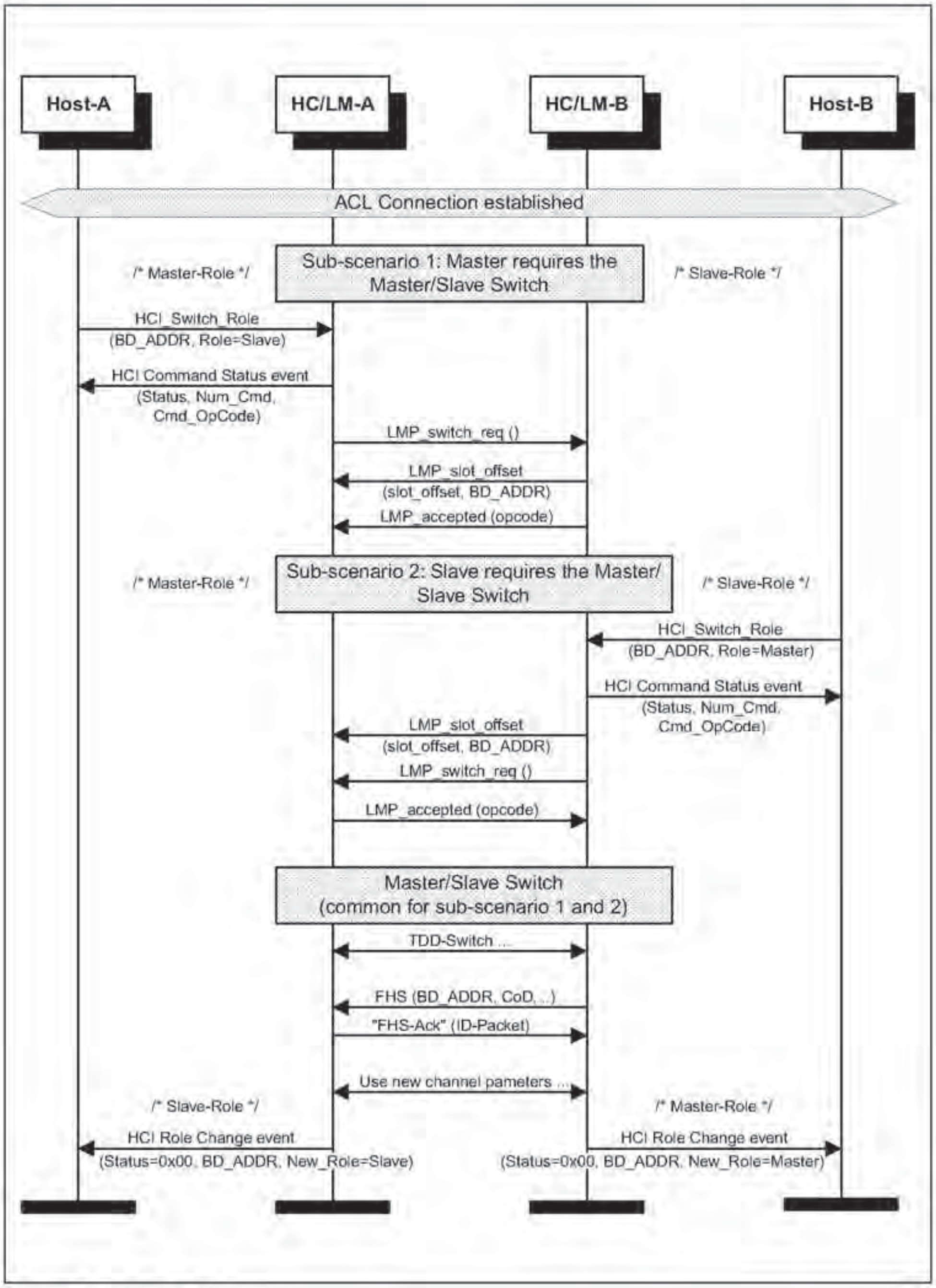


Figure 4.9: Switch Role

5 SCO CONNECTION ESTABLISHMENT AND DETACHMENT

5.1 SCO CONNECTION SETUP

SCO Connection setup requires an established ACL Connection. It is the task of the Host to create an ACL Connection first and then the SCO Link.

Note: On the slave side, an incoming connection request can be automatically accepted by using HCI_Set_Event_Filter (Filter_Type, Filter_Condition_Type, Condition) with the Filter_Type = 0x02 /*Connection_Setup*/. Furthermore, for each SCO Link to a BT Device, a separate SCO Connection Handle is needed.

5.1.1 Master activates the SCO Connection setup

To set up an SCO Connection, the HCI_Add_SCO_Connection (Connection_Handle, Packet_Type) command is used. The specified Connection_Handle is related to the ACL Connection that must have been created before the HCI_Add_SCO_Connection is issued.

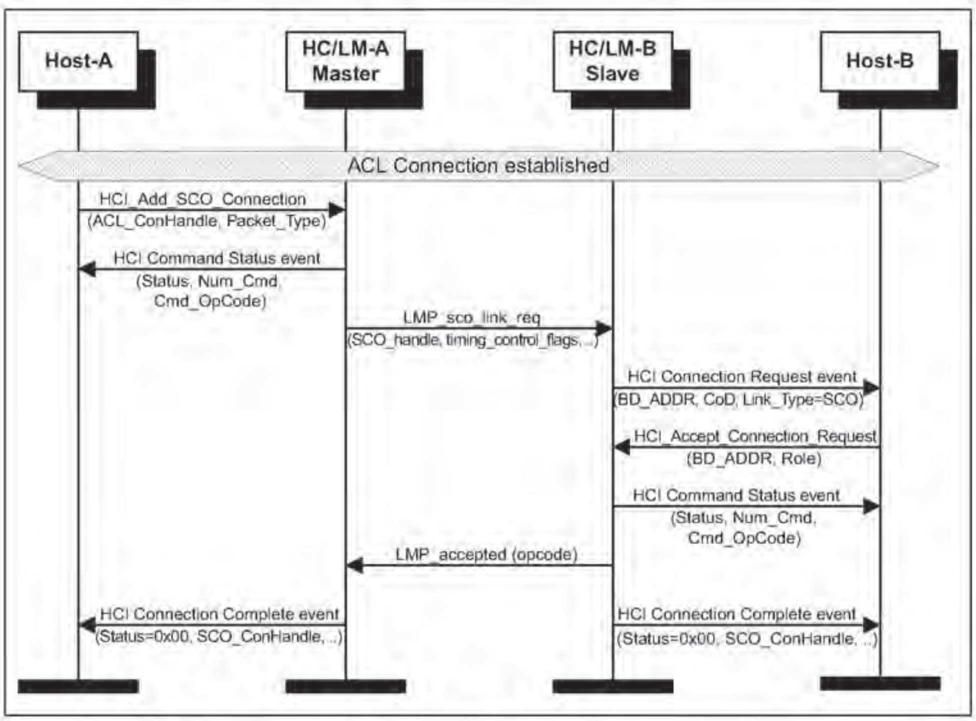


Figure 5.1: SCO Connection setup (activated from master)

5.1.2 Slave activates the SCO Connection setup

The same command HCI_Add_SCO_Connection (Connection_Handle, Packet_Type) can be used to create an SCO Link when the local BT Device is a BT Slave. Here the specified Connection_Handle belongs to the established ACL Connection between the BT Devices. Compared to 5.1.1, the only difference is that the HC/LM Slave starts the SCO Setup with LMP_sco_link_req first.

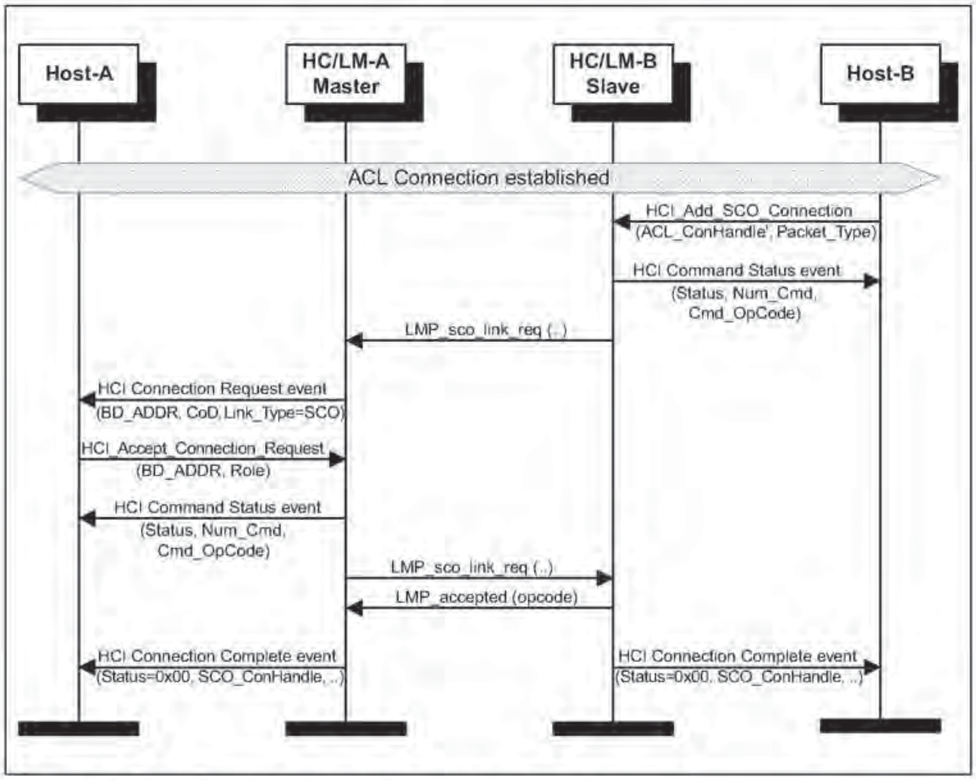


Figure 5.2: SCO Connection setup (activated from slave)

5.2 SCO DISCONNECTION

An established SCO Connection can be detached at any time. Since several SCO Connections can exist between a BT Master and a BT Slave, an SCO Disconnection only removes the SCO Link with the specified SCO Connection Handle. The other SCO Connections will still exist.

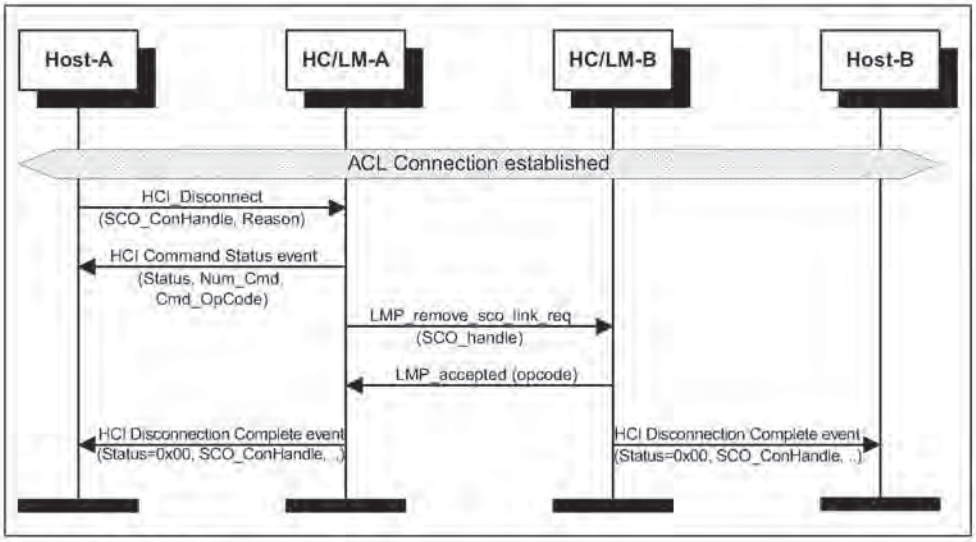


Figure 5.3: SCO Disconnection

6 SPECIAL MODES: SNIFF, HOLD, PARK

Entry into sniff, hold or park mode requires an established ACL Connection. The following table summarizes the modes and the BT Role that can request, force, activate or exit the modes.

	Sniff	Hold	Park
Request	Master/Slave	Master/Slave	Master/Slave
Force	Master	Master/Slave	Master
Activation	Master	Master/Slave	Master
Release	Master/Slave	Automatic	Master/Slave

Table 6.1: Summary of modes (Sniff, Hold, Park)

6.1 SNIFF MODE

Sniff Mode is used when a slave shall participate in the piconet only in a sniff interval. For the Sniff Mode negotiation, the Host specifies the Sniff_Max_Interval and the Sniff_Min_Interval so that HC/LM will be able to choose the one sniff interval in this range. The used command is HCI_Sniff_Mode (Connection_Handle, Sniff_Max_Interval, Sniff_Min_Interval, Sniff_Attempt, Sniff_Timeout).
Since Sniff Mode is a periodic mode, the command HCI_Exit_Sniff_Mode (Connection_Handle) is needed to return to Active Mode.

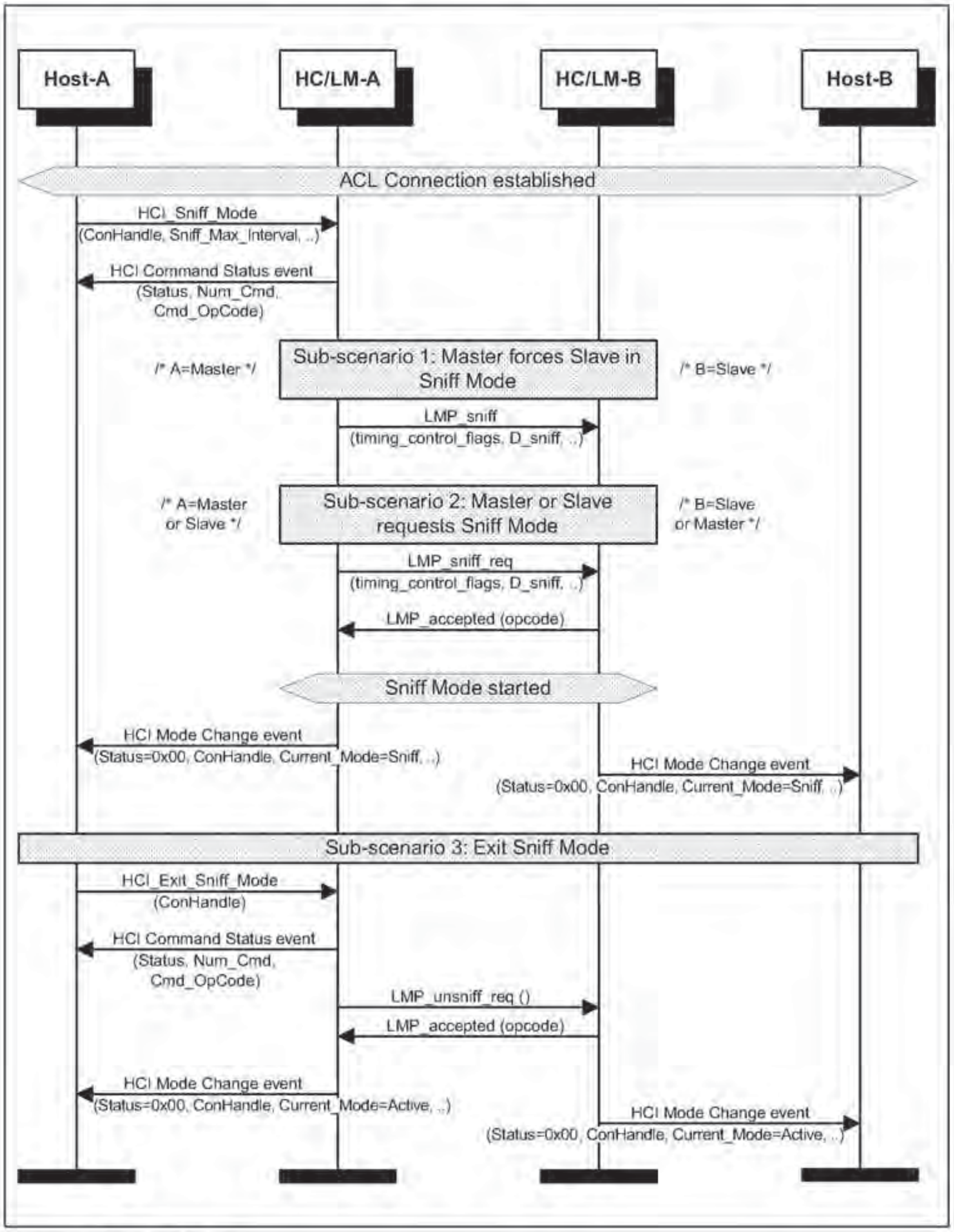


Figure 6.1: Sniff Mode

6.2 HOLD MODE

Hold Mode is useful when a BT Device doesn't want to participate in the connection for a Hold Mode Length. Using the command HCI_Hold_Mode (Connection_Handle, Hold_Max_Length, Hold_Min_Length), the Host specifies the Hold_Max_Length and Hold_Min_Length. The HC/LM will then be able to negotiate a Hold Mode Length in this range. When the hold mode is started or complete, Mode Change event (Status, Connection_ Handle, Current_Mode, Interval) will be used to inform the Host about the actual mode.

Note: the Hold Mode is exited when the Hold Mode Length has expired, so it is no guarantee that the remote BT Device is immediately active.

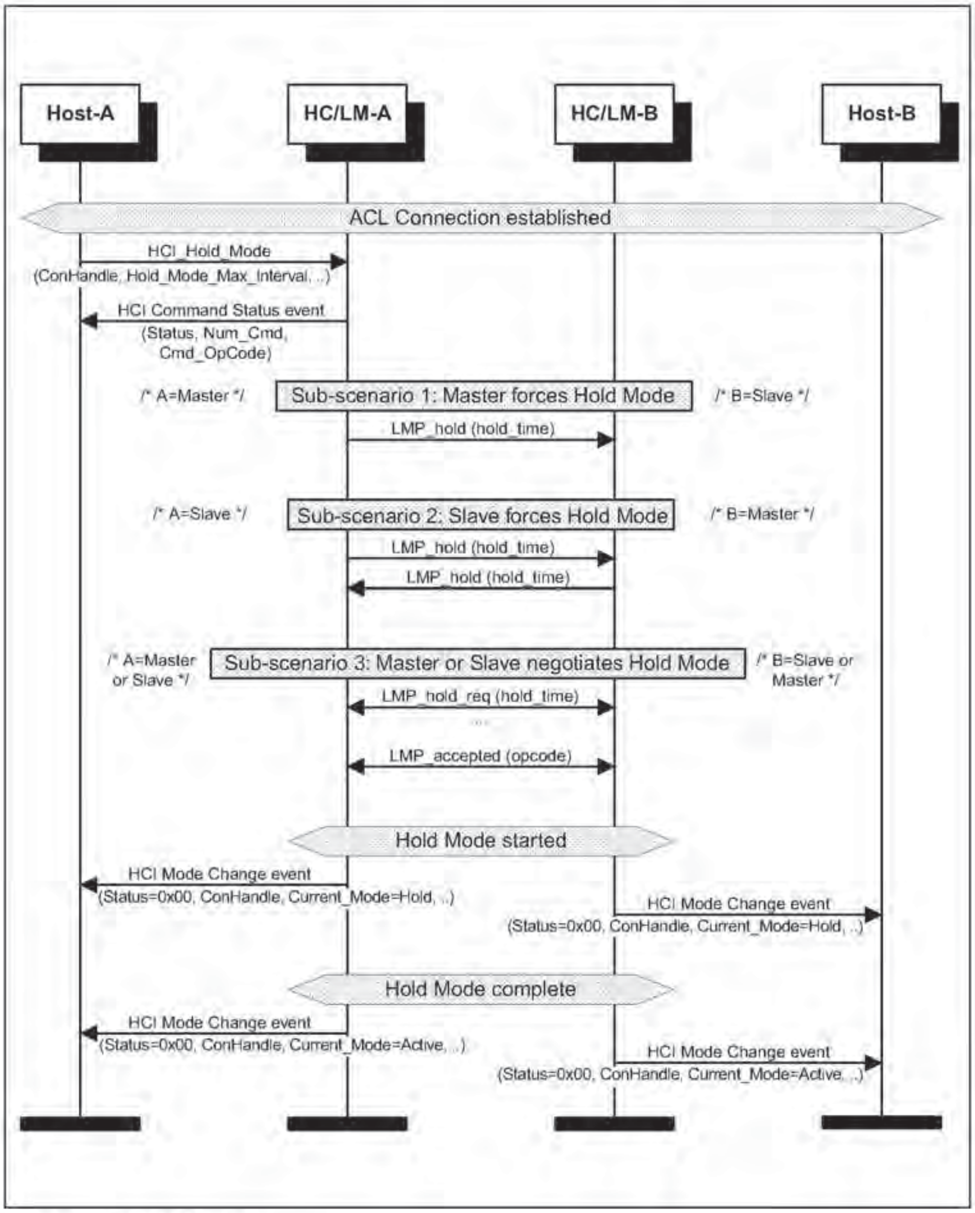


Figure 6.2: Hold Mode

6.3 PARK MODE

Park Mode is used to render the slaves inactive but still synchronized to the master using the beacon interval. In park mode, broadcast is performed.

6.3.1 Enter park mode

Using the command HCI_Park_Mode (Connection_Handle, Beacon_Max_Interval, Beacon_Min_Interval) the Host specifies the Beacon_Max_Interval and Beacon_Min_Interval so that HC/LM can set up a Beacon-Interval in this range for the BT Slaves. In Park Mode, the BT Slave gives up its AM_ADDR.

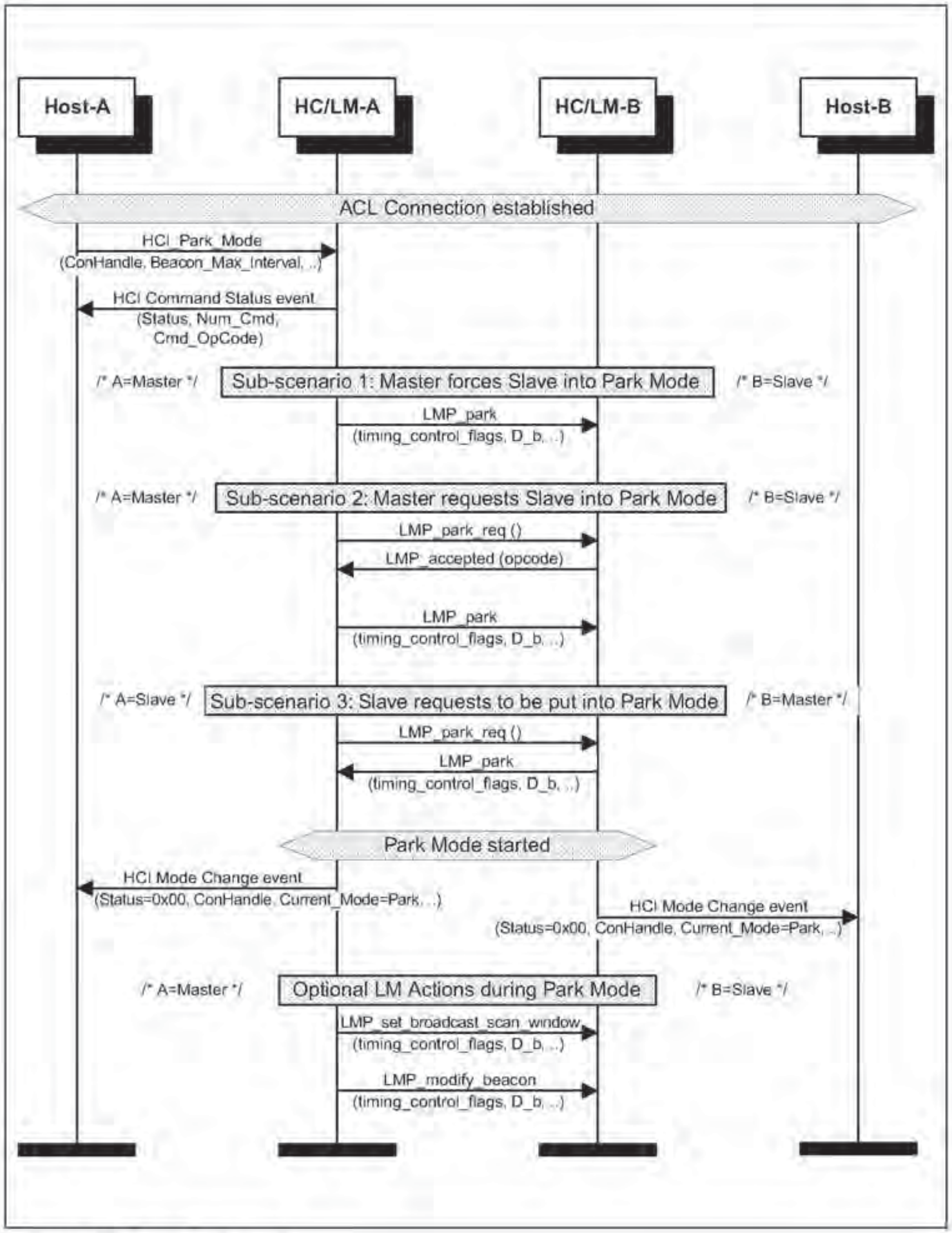


Figure 6.3: Enter Park Mode

6.3.2 Exit Park Mode

Since Park Mode is a periodic mode, the command HCI_Exit_Park_Mode (Connection_Handle) will be used to return to Active Mode. A parked BT Slave can send an Access_Request_Message to request to leave the Park Mode. It is the task of master HC/LM to use LMP_unpark_PM_ADDR_req (..) or LMP_unpark_BD_ADDR_req (..) to unpark a BT Slave.

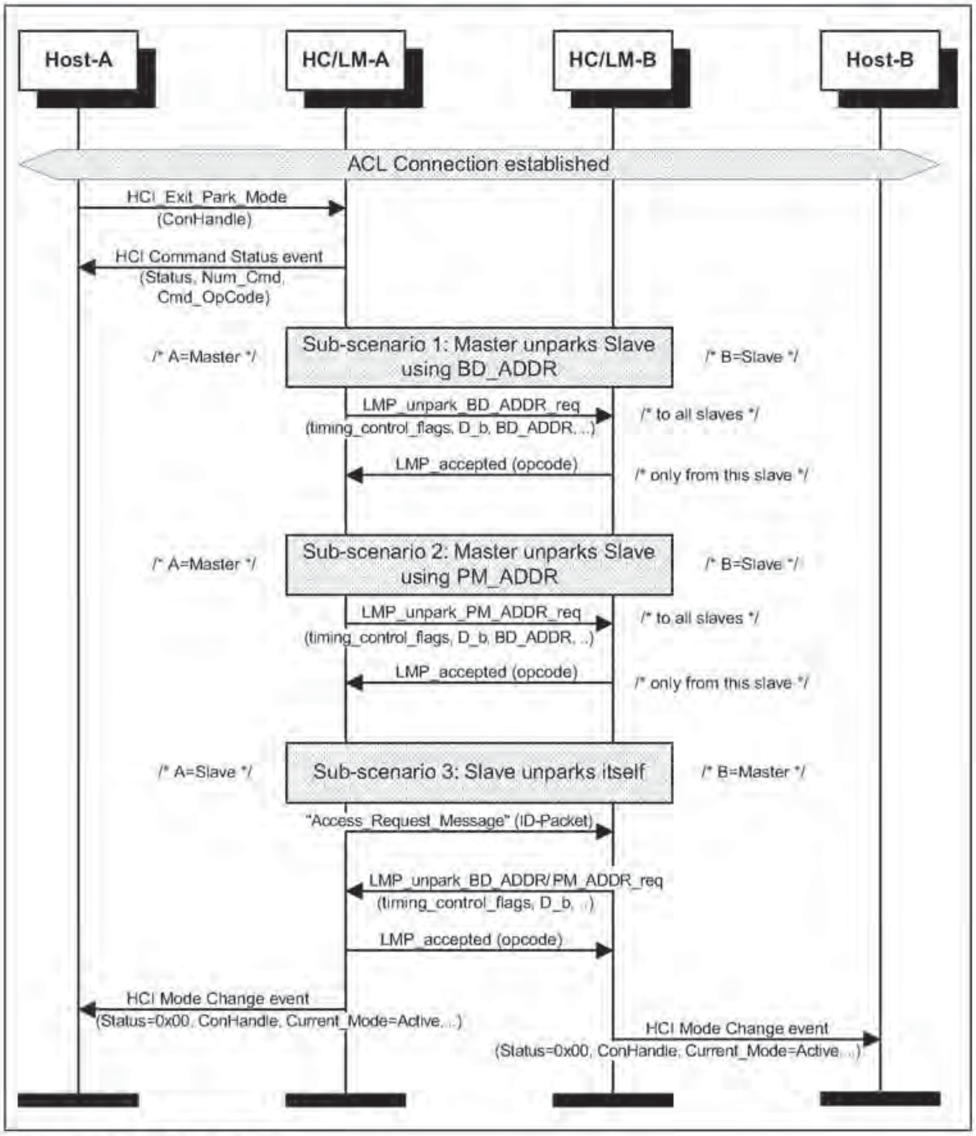


Figure 6.4: Exit Park Mode

7 BUFFER MANAGEMENT, FLOW CONTROL

The HC Data buffers are configured by the HC and managed by the Host. On initialization, the Host will issue HCI_Read_Buffer_Size. This specifies the maximum allowed length of HCI data packets sent from the Host to the HC, and the maximum number of ACL and SCO data packets that the HC can store in its buffer. After a connection is created, HC will frequently inform the Host about the number of sent packets using Number Of Completed Packets event (Number_of_Handles, Connection_Handle[I], HC_Num_Of_Completed_Packets[i]) (see Figure 7.1 Host-to-HC flow control).

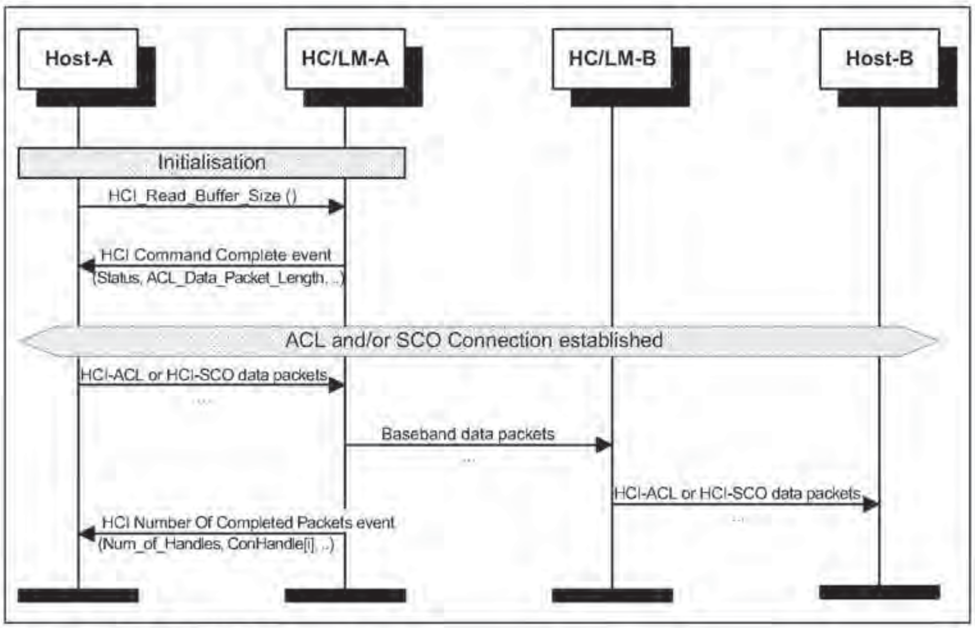


Figure 7.1: Host to HC flow control

Accordingly the HC to Host flow control can be applied in the same way so that during initialization the Host configures the Buffer Size and later the Host Controller will manage the Host Buffers.
Using HCI_Set_Host_Controller_To_Host_Flow_Control (Flow_Control_Enable) the Host can decide to apply the HC to Host flow control or not. For flow control itself HCI_Host_Buffer_Size (Host_ACL_Data_Packet_Length, Host_SCO_Data_Packet_Length, Host_Total_Num_ACL_Data_Packets, Host_Total_Num_SCO_Data_Packets) and HCI_Host_Number_Of_Completed_Packets (Number_of_Handles, Connection_Handle[I], Host_Num_Of_Completed_Packets[I]) will be used (for details see Figure 7.2 HC to Host Flow Control).

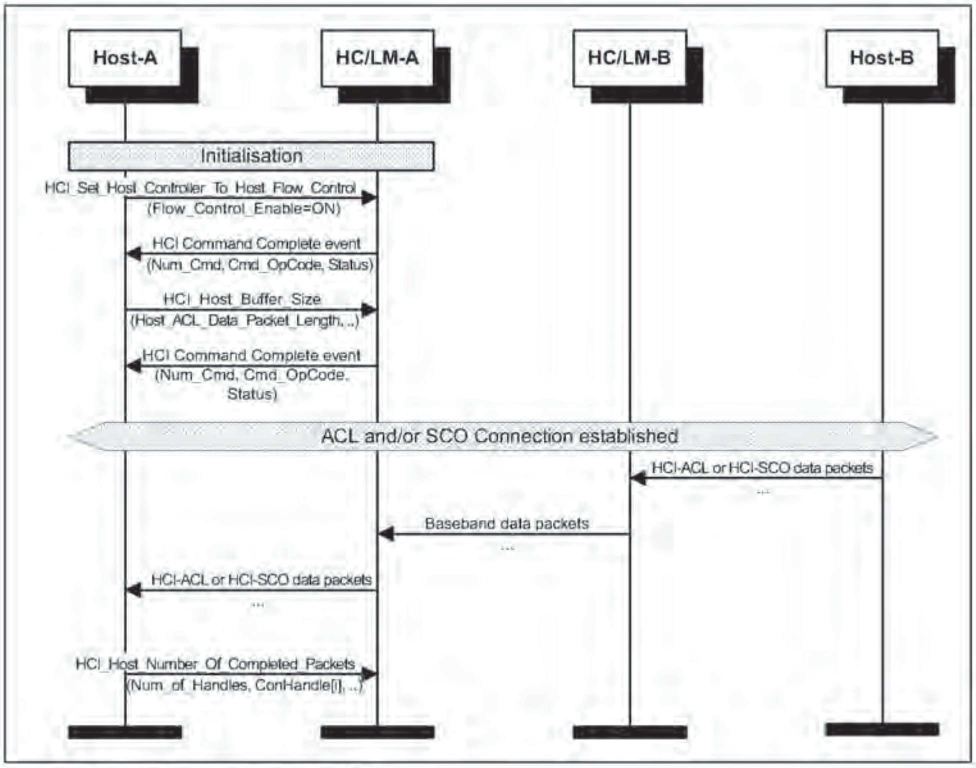


Figure 7.2: HC to Host Flow Control

8 LOOPBACK MODE

8.1 LOCAL LOOPBACK MODE

The local Loopback Mode is used to loopback received HCI Commands, and HCI ACL and HCI SCO packets sent from the Host.

The HC will send four Connection Complete events (one for ACL, three for SCO Connections) so that the Host can use the Connection_Handles to re-send HCI ACL and HCI SCO Packet to HC. To exit the local Loopback Mode, HCI_Write_Loopback_Mode (Loopback_Mode=0x00) or HCI_Reset ( ) will be used.

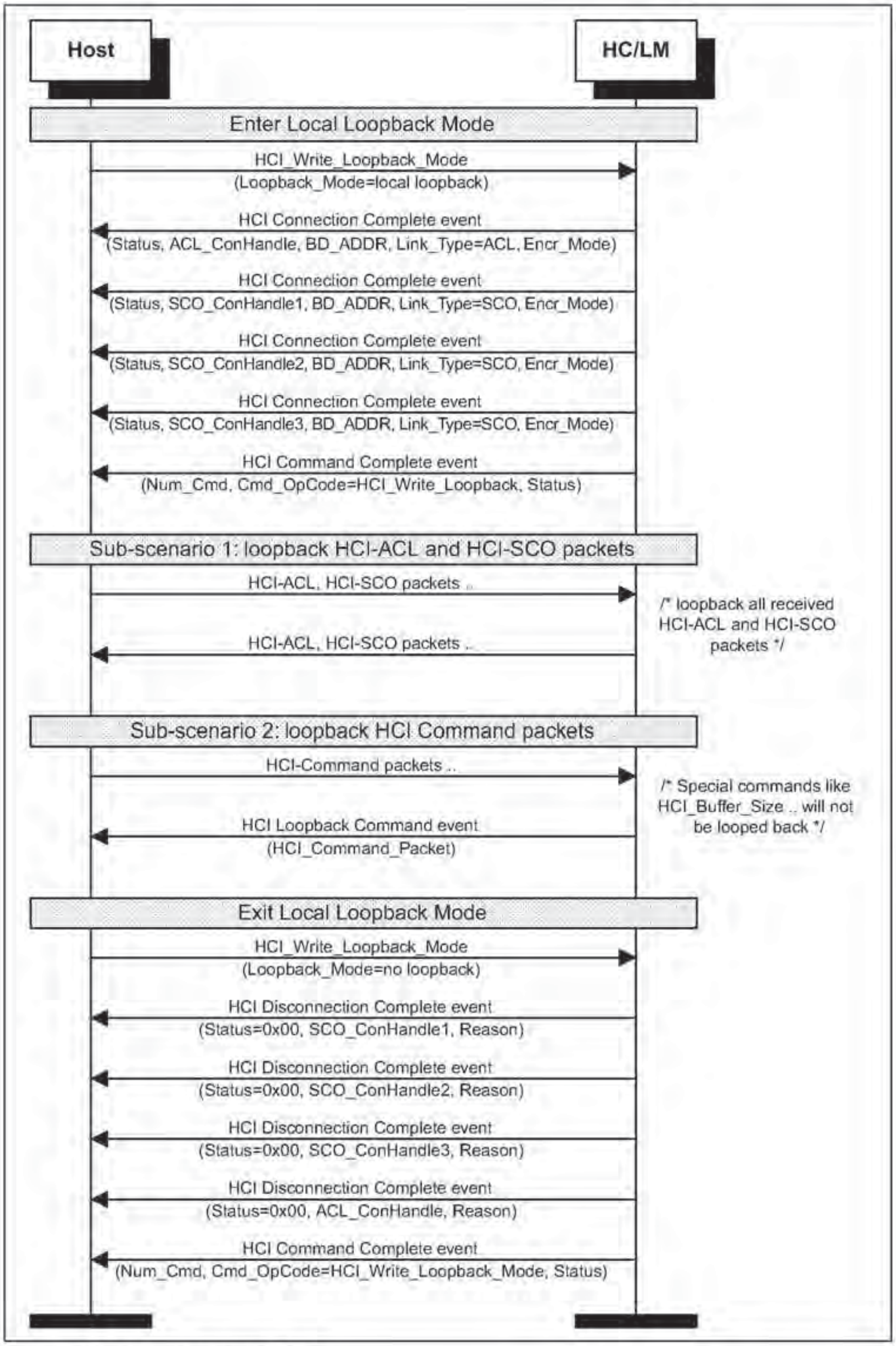


Figure 8.1: Local Loopback Mode

8.2 REMOTE LOOPBACK MODE

The remote Loopback Mode is used to loopback all received Baseband ACL and SCO Data received from a remote BT Device. During remote Loopback Mode, ACL and SCO Connection can be created. The remote Loopback Mode can be released with the command HCI_Write_Loopback_Mode (Loopback_Mode=0x00).

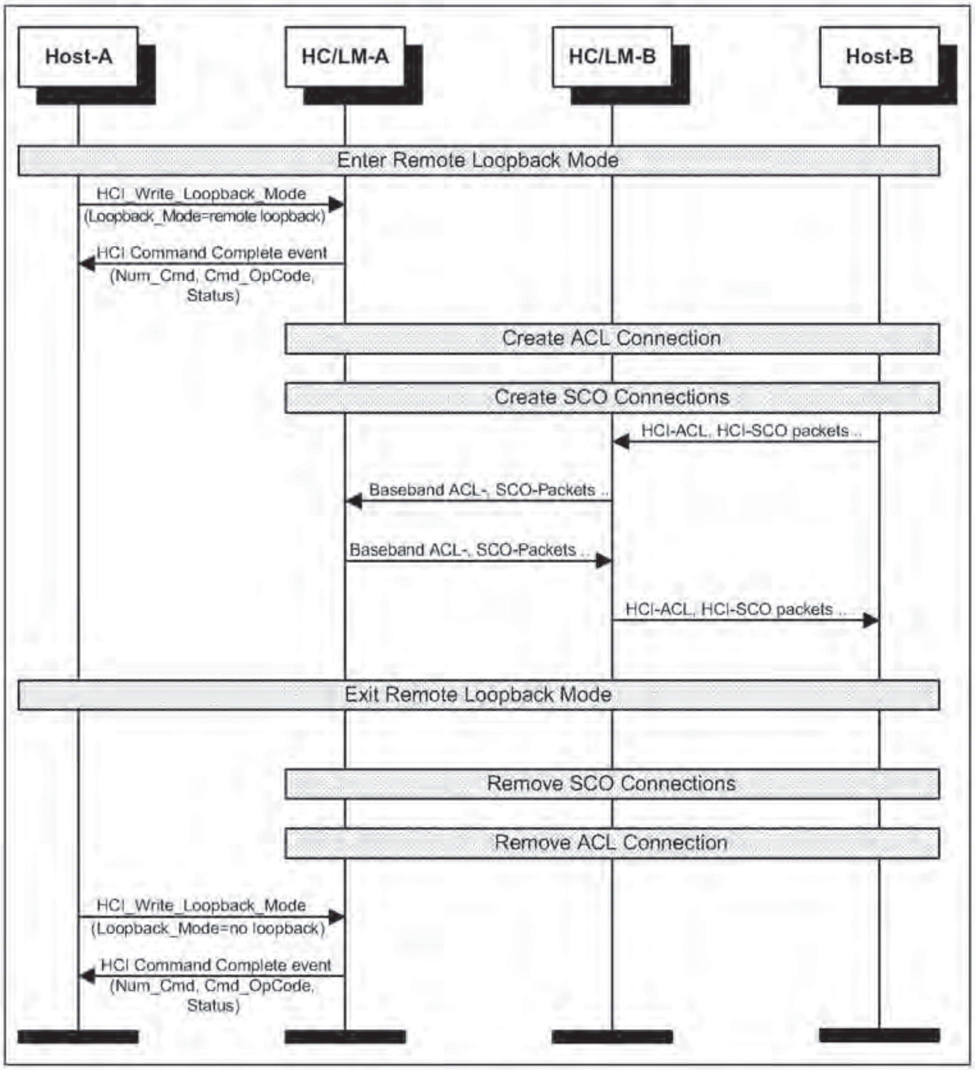


Figure 8.2: Remote Loopback Mode

9 LIST OF ACRONYMS AND ABBREVIATIONS

BT	Bluetooth
HC	Host Controller
HCI	Host Controller Interface
LAP	Lower Address Part
LC	Link Controller
LM	Link Manager
LMP	Link Manager Protocol
MSC	Message Sequence Chart
PDU	Protocol Data Unit

10 LIST OF FIGURES

11 LIST OF TABLES

Bluetooth.

12 REFERENCES

[1] "Baseband Specification" on page 33

[2] "Link Manager Protocol" on page 185

[3] "Host Controller Interface Functional Specification" on page 517

[4] "Logical Link Control and Adaptation Protocol Specification" on page 245

Bluetooth.

Alphabetical Index

Bluetooth.

X

Z

Specification Volume 2

Specification of the Bluetooth System

Wireless connections made easy

Profiles

Bluetooth™

v1.0 B
December 1st 1999

BLUETOOTH DOC	Date / Day-Month-Year 01 Dec 99	N.B.	Document No. 1.C.47/1.0 B
Responsible	e-mail address		Status

Bluetooth.

Profiles of the Bluetooth System

Version 1.0B

Revision History

The Revision History is shown in Appendix I on page 413

Contributors

The persons who contributed to this specification are listed in Appendix II on page 421.

Web Site

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

Disclaimer and copyright notice

Bluetooth.

MASTER TABLE OF CONTENTS

For the Core Specification, *see Volume 1*

Part K:1

GENERIC ACCESS PROFILE

Part K:2

SERVICE DISCOVERY APPLICATION PROFILE

Part K:3

CORDLESS TELEPHONY PROFILE

Contents 97

Part K:4

INTERCOM PROFILE

Contents 141

Part K:5

SERIAL PORT PROFILE

Part K:6

HEADSET PROFILE

Part K:7

DIAL-UP NETWORKING PROFILE

Part K:8

FAX PROFILE

Bluetooth.

Part K:9

LAN ACCESS PROFILE

Part K:10

GENERIC OBJECT EXCHANGE PROFILE

Bluetooth.

Part K:11

OBJECT PUSH PROFILE

Part K:12

FILE TRANSFER PROFILE

Part K:13

SYNCHRONIZATION PROFILE

Bluetooth.

Appendix I

Appendix II

Appendix III

Part K:1

GENERIC ACCESS PROFILE

This profile defines the generic procedures related to discovery of Bluetooth devices (idle mode procedures) and link management aspects of connecting to Bluetooth devices (connecting mode procedures). It also defines procedures related to use of different security levels. In addition, this profile includes common format requirements for parameters accessible on the user interface level.

CONTENTS

FOREWORD

Interoperability between devices from different manufacturers is provided for a specific service and use case, if the devices conform to a Bluetooth SIG-defined profile specification. A profile defines a selection of messages and procedures (generally termed *capabilities*) from the Bluetooth SIG specifications and gives an unambiguous description of the air interface for specified service(s) and use case(s).

All defined features are process-mandatory. This means that, if a feature is used, it is used in a specified manner. Whether the provision of a feature is mandatory or optional is stated separately for both sides of the Bluetooth air interface.

1 INTRODUCTION

1.1 SCOPE

The purpose of the Generic Access Profile is:

To introduce definitions, recommendations and common requirements related to modes and access procedures that are to be used by transport and application profiles.

To describe how devices are to behave in standby and connecting states in order to guarantee that links and channels always can be established between Bluetooth devices, and that multi-profile operation is possible. Special focus is put on discovery, link establishment and security procedures.

To state requirements on user interface aspects, mainly coding schemes and names of procedures and parameters, that are needed to guarantee a satisfactory user experience.

1.2 SYMBOLS AND CONVENTIONS

1.2.1 Requirement status symbols

In this document (especially in the profile requirements tables), the following symbols are used:

'M' for mandatory to support (used for capabilities that shall be used in the profile);

'O' for optional to support (used for capabilities that can be used in the profile);

'C' for conditional support (used for capabilities that shall be used in case a certain other capability is supported);

'X' for excluded (used for capabilities that may be supported by the unit but shall never be used in the profile);

'N/A' for not applicable (in the given context it is impossible to use this capability).

Some excluded capabilities are capabilities that, according to the relevant Bluetooth specification, are mandatory. These are features that may degrade operation of devices following this profile. Therefore, these features shall never be activated while a unit is operating as a unit within this profile.

In this specification, the word *shall* is used for mandatory requirements, the word *should* is used to express recommendations and the word *may* is used for options.

1.2.2 Signalling diagram conventions

The following arrows are used in diagrams describing procedures
:

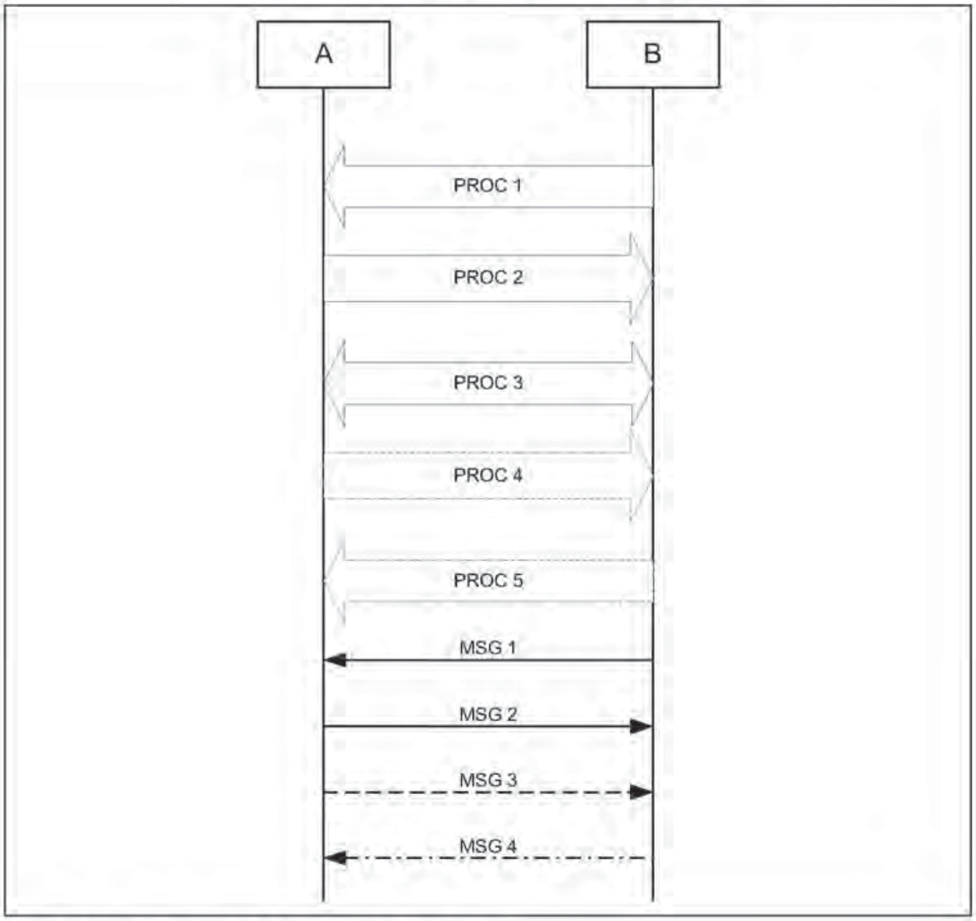


Figure 1.1: Arrows used in signalling diagrams

In the table above, the following cases are shown: PROC1 is a sub-procedure initiated by B. PROC2 is a sub-procedure initiated by A. PROC3 is a sub-procedure where the initiating side is undefined (may be both A or B). Dashed arrows denote optional steps. PROC4 indicates an optional sub-procedure initiated by A, and PROC5 indicates an optional sub-procedure initiated by B.

MSG1 is a message sent from B to A. MSG2 is a message sent from A to B. MSG3 indicates an optional message from A to B, and MSG4 indicates a conditional message from B to A.

1.2.3 Notation for timers and counters

Timers are introduced specific to this profile. To distinguish them from timers used in the Bluetooth protocol specifications and other profiles, these timers are named in the following format: 'T_{GAP}*(nnn)*'.

2 PROFILE OVERVIEW

2.1 PROFILE STACK

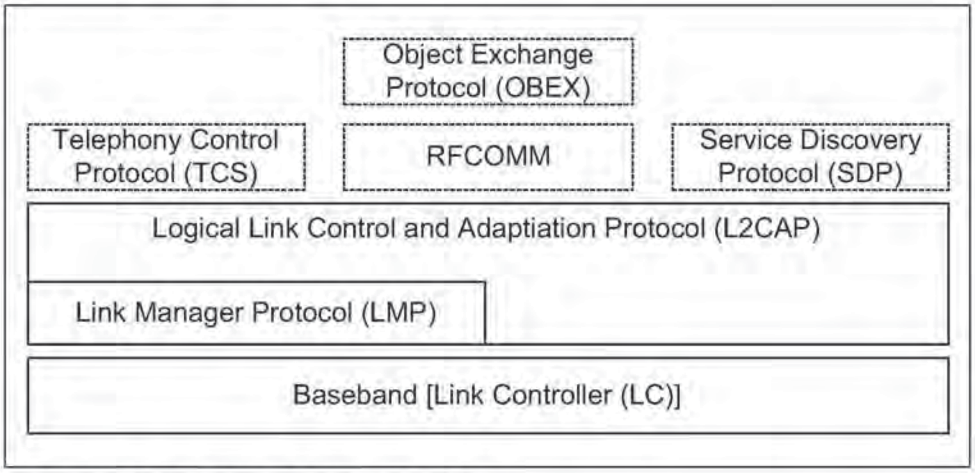


Figure 2.1: Profile stack covered by this profile.

The main purpose of this profile is to describe the use of the lower layers of the Bluetooth protocol stack (LC and LMP). To describe security related alternatives, also higher layers (L2CAP, RFCOMM and OBEX) are included.

2.2 CONFIGURATIONS AND ROLES

For the descriptions in this profile of the roles that the two devices involved in a Bluetooth communication can take, the generic notation of the A-party (the *paging device* in case of link establishment, or *initiator* in case of another procedure on an established link) and the B-party (*paged device* or *acceptor*) is used. The A-party is the one that, for a given procedure, initiates the establishment of the physical link or initiates a transaction on an existing link.

This profile handles the procedures between two devices related to discovery and connecting (link and connection establishment) for the case where none of the two devices has any link established as well as the case where (at least) one device has a link established (possibly to a third device) before starting the described procedure.

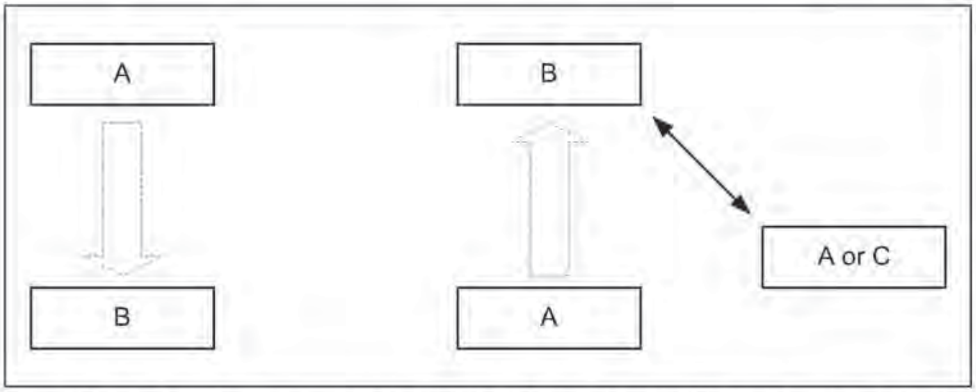


Figure 2.2: This profile covers procedures initiated by one device (A) towards another device (B) that may or may not have an existing Bluetooth link active.

The initiator and the acceptor generally operate the generic procedures according to this profile or another profile referring to this profile. If the acceptor operates according to several profiles simultaneously, this profile describes generic mechanisms for how to handle this.

2.3 USER REQUIREMENTS AND SCENARIOS

The Bluetooth user should in principle be able to connect a Bluetooth device to any other Bluetooth device. Even if the two connected devices don't share any common application, it should be possible for the user to find this out using basic Bluetooth capabilities. When the two devices do share the same application but are from different manufacturers, the ability to connect them should not be blocked just because manufacturers choose to call basic Bluetooth capabilities by different names on the user interface level or implement basic procedures to be executed in different orders.

2.4 PROFILE FUNDAMENTALS

This profile states the requirements on names, values and coding schemes used for names of parameters and procedures experienced on the user interface level.

This profile defines modes of operation that are not service- or profile-specific, but that are generic and can be used by profiles referring to this profile, and by devices implementing multiple profiles.

This profile defines the general procedures that can be used for discovering identities, names and basic capabilities of other Bluetooth devices that are in a mode where they can be discoverable. Only procedures where no channel or connection establishment is used are specified.

This profile defines the general procedure for how to create bonds (i.e. dedicated exchange of link keys) between Bluetooth devices.

This profile describes the general procedures that can be used for establishing connections to other Bluetooth devices that are in mode that allows them to accept connections and service requests.

2.5 CONFORMANCE

Bluetooth devices that do not conform to any other Bluetooth profile shall conform to this profile to ensure basic interoperability and co-existence.

Bluetooth devices that conform to another Bluetooth profile may use adaptations of the generic procedures as specified by that other profile. They shall, however, be compatible with devices compliant to this profile at least on the level of the supported generic procedures.

If conformance to this profile is claimed, all capabilities indicated mandatory for this profile shall be supported in the specified manner (process-mandatory). This also applies for all optional and conditional capabilities for which support is indicated. All mandatory capabilities, and optional and conditional capabilities for which support is indicated, are subject to verification as part of the Bluetooth certification program.

3 USER INTERFACE ASPECTS

3.1 THE USER INTERFACE LEVEL

In the context of this specification, the user interface level refers to places (such as displays, dialog boxes, manuals, packaging, advertising, etc.) where users of Bluetooth devices encounters names, values and numerical representation of Bluetooth terminology and parameters.

This profile specifies the generic terms that should be used on the user interface level. These terms should be translated into languages supported by the Bluetooth device according to tables provided by the Bluetooth SIG.

3.2 REPRESENTATION OF BLUETOOTH PARAMETERS

3.2.1 Bluetooth device address (BD_ADDR)

3.2.1.1 Definition

BD_ADDR is the unique address of a Bluetooth device as defined in [1]. It is received from a remote device during the device discovery procedure.

3.2.1.2 Term on user interface level

When the Bluetooth address is referred to on UI level, the term 'Bluetooth Device Address' should be used.

3.2.1.3 Representation

On BB level the BD_ADDR is represented as 48 bits [1].

On the UI level the Bluetooth address shall be represented as 12 hexadecimal characters, possibly divided into sub-parts separated by':'.
(E.g., '000C3E3A4B69' or '00:0C:3E:3A:4B:69'.) At UI level, any number shall have the MSB -> LSB (from left to right) 'natural' ordering (e.g., the number '16' shall be shown as '0x10').

3.2.2 Bluetooth device name (the user-friendly name)

3.2.2.1 Definition

The Bluetooth device name is the user-friendly name that a Bluetooth device presents itself with. It is a character string returned in LMP_name_res as response to a LMP_name_req.

3.2.2.2 Term on user interface level

When the Bluetooth device name is referred to on UI level, the term 'Bluetooth Device Name' should be used.

3.2.2.3 Representation

The Bluetooth device name can be up to 248 bytes maximum according to [2]. It shall be coded according to Unicode UTF-8 (i.e. name entered on UI level may be down to 82 characters if UCS-2 is used).

A device can not expect that a general remote device is able to handle more than the first 40 characters of the Bluetooth device name. If a remote device has limited display capabilities, it may use only the first 20 characters.

3.2.3 Bluetooth passkey (Bluetooth PIN)

3.2.3.1 Definition

The Bluetooth PIN is used to authenticate two Bluetooth devices (that have not previously exchanged link keys) to each other and create a trusted relationship between them. The PIN is used in the pairing procedure (see Section 10.2) to generate the initial link key that is used for further authentication.

The PIN may be entered on UI level but may also be stored in the device; e.g. in the case of a device without sufficient MMI for entering and displaying digits.

3.2.3.2 Terms at user interface level

When the Bluetooth PIN is referred to on UI level, the term 'Bluetooth Passkey' should be used.

3.2.3.3 Representation

The Bluetooth PIN has different representations on different level. PIN_{BB} is used on baseband level, and PIN_{UI} is used on user interface level.

PIN_{BB} is the PIN used by [1] for calculating the initialization key during the pairing procedure. PIN_{UI} is the character representation of the PIN that is entered on UI level. The transformation between PIN_{BB} and PIN_{UI} shall be according to Unicode UTF-8.

According to [1], PIN_{BB} can be 128 bits (16 bytes). When PIN is entered on UI level (PIN_{UI}), it is to be coded into PIN_{BB} according to Unicode UTF-8 (i.e. if a device supports entry of characters outside the Unicode range 0x00 - 0x7F, the maximum number of characters in the PIN_{UI} may be less than 16).

Examples:

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

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

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

3.2.4 Class of Device

3.2.4.1 Definition

Class of device is a parameter received during the device discovery procedure, indicating the type of device and which types of service that are supported.

3.2.4.2 Term on user interface level

The information within the Class of Device parameter should be referred to as 'Bluetooth Device Class' (i.e. the major and minor device class fields) and 'Bluetooth Service Type' (i.e. the service class field). The terms for the defined Bluetooth Device Types and Bluetooth Service Types are defined in [11].

When using a mix of information found in the Bluetooth Device Class and the Bluetooth Service Type, the term 'Bluetooth Device Type' should be used.

3.2.4.3 Representation

The Class of device is a bit field and is defined in [11]. The UI-level representation of the information in the Class of device is implementation specific.

3.3 PAIRING

Two procedures are defined that make use of the pairing procedure defined on LMP level (LMP-pairing, see Section 10.2). Either the user initiates the bonding procedure and enters the passkey with the explicit purpose of creating a bond (and maybe also a secure relationship) between two Bluetooth devices, or the user is requested to enter the passkey during the establishment procedure since the devices did not share a common link key beforehand. In the first case, the user is said to perform 'bonding (with entering of passkey)' and in the second case the user is said to 'authenticate using the passkey'.

4 MODES

	Procedure	Ref.	Support
1	Discoverability modes	4.1	
	Non-discoverable mode		C1
	Limited discoverable mode		C2
	General discoverable mode		C2
2	Connectability modes	4.1.3.3	
	Non-connectable mode		O
	Connectable mode		M
3	Pairing modes	4.2.2.2	
	Non-pairable mode		O
	Pairable mode		C3
C1: If limited discoverable mode is supported, non-discoverable mode is mandatory, otherwise optional.			
C2: A Bluetooth device shall support at least one discoverable mode (limited or/and general).			
C3: If the bonding procedure is supported, support for pairable mode is mandatory, otherwise optional.			

Table 4.1: Conformance requirements related to modes defined in this section

4.1 DISCOVERABILITY MODES

With respect to inquiry, a Bluetooth device shall be either in non-discoverable mode or in a discoverable mode. (The device shall be in one, and only one, discoverability mode at a time.) The two discoverable modes defined here are called limited discoverable mode and general discoverable mode. Inquiry is defined in [1].

When a Bluetooth device is in non-discoverable mode it does not respond to inquiry.

A Bluetooth device is said to be made discoverable, or set into a discoverable mode, when it is in limited discoverable mode or in general discoverable mode. Even when a Bluetooth device is made discoverable it may be unable to respond to inquiry due to other baseband activity [1]. A Bluetooth device that does not respond to inquiry for any of these two reasons is called a silent device.

After being made discoverable, the Bluetooth device shall be discoverable for at least $T_{GAP}(103)$.

4.1.1 Non-discoverable mode

4.1.1.1 Definition

When a Bluetooth device is in non-discoverable mode, it shall never enter the INQUIRY_RESPONSE state.

4.1.1.2 Term on UI-level

Bluetooth device is 'non-discoverable' or in 'non-discoverable mode'.

4.1.2 Limited discoverable mode

4.1.2.1 Definition

The limited discoverable mode should be used by devices that need to be discoverable only for a limited period of time, during temporary conditions or for a specific event. The purpose is to respond to a device that makes a limited inquiry (inquiry using the LIAC).

A Bluetooth device should not be in limited discoverable mode for more than $T_{GAP}(104)$. The scanning for the limited inquiry access code can be done either in parallel or in sequence with the scanning of the general inquiry access code. When in limited discoverable mode, one of the following options shall be used.

4.1.2.1.1 Parallel scanning

When a Bluetooth device is in limited discoverable mode, it shall enter the INQUIRY_SCAN state at least once in $T_{GAP}(102)$ and scan for the GIAC and the LIAC for at least $T_{GAP}(101)$.

4.1.2.1.2 Sequential scanning

When a Bluetooth device is in limited discoverable mode, it shall enter the INQUIRY_SCAN state at least once in $T_{GAP}(102)$ and scan for the GIAC for at least $T_{GAP}(101)$ and enter the INQUIRY_SCAN state more often than once in $T_{GAP}(102)$ and scan for the LIAC for at least $T_{GAP}(101)$.

If an inquiry message is received when in limited discoverable mode, the entry into the INQUIRY_RESPONSE state takes precedence over the next entries into INQUIRY_SCAN state until the inquiry response is completed.

4.1.2.2 Conditions

When a device is in limited discoverable mode it shall set bit no 13 in the Major Service Class part of the Class of Device/Service field [11].

4.1.2.3 Term on UI-level

Bluetooth device is 'discoverable' or in 'discoverable mode'.

4.1.3 General discoverable mode

4.1.3.1 Definition

The general discoverable mode shall be used by devices that need to be discoverable continuously or for no specific condition. The purpose is to respond to a device that makes a general inquiry (inquiry using the GIAC).

4.1.3.2 Conditions

When a Bluetooth device is in general discoverable mode, it shall enter the INQUIRY_SCAN state more often than once in $T_{GAP}(102)$ and scan for the GIAC for at least $T_{GAP}(101)$.

A device in general discoverable mode shall not respond to a LIAC inquiry.

4.1.3.3 Term on UI-level

Bluetooth device is 'discoverable' or in 'discoverable mode'.

4.2 CONNECTABILITY MODES

With respect to paging, a Bluetooth device shall be either in non-connectable mode or in connectable mode. Paging is defined in [1].

When a Bluetooth device is in non-connectable mode it does not respond to paging. When a Bluetooth device is in connectable mode it responds to paging.

4.2.1 Non-connectable mode

4.2.1.1 Definition

When a Bluetooth device is in non-connectable mode it shall never enter the PAGE_SCAN state.

4.2.1.2 Term on UI-level

Bluetooth device is 'non-connectable' or in 'non-connectable mode'.

4.2.2 Connectable mode

4.2.2.1 Definition

When a Bluetooth device is in connectable mode it shall periodically enter the PAGE_SCAN state.

4.2.2.2 Term on UI-level

Bluetooth device is 'connectable' or in 'connectable mode'.

4.3 PAIRING MODES

With respect to pairing, a Bluetooth device shall be either in non-pairable mode or in pairable mode. In pairable mode the Bluetooth device accepts paring – i.e. creation of bonds – initiated by the remote device, and in non-pairable mode it does not. Pairing is defined in [1] and [2].

4.3.1 Non-pairable mode

4.3.1.1 Definition

When a Bluetooth device is in non-pairable mode it shall respond to a received LMP_in_rand with LMP_not_accepted with the reason *pairing not allowed*.

4.3.1.2 Term on UI-level

Bluetooth device is 'non-bondable' or in 'non-bondable mode' or "does not accept bonding".

4.3.2 Pairable mode

4.3.2.1 Definition

When a Bluetooth device is in pairable mode it shall respond to a received LMP_in_rand with LMP_accepted (or with LMP_in_rand if it has a fixed PIN).

4.3.2.2 Term on UI-level

Bluetooth device is 'bondable' or in 'bondable mode' or "accepts bonding".

5 SECURITY ASPECTS

	Procedure	Ref.	Support
1	Authentication	5.1	C1
2	Security modes	5.2	
	Security mode 1		O
	Security mode 2		C2
	Security mode 3		C2
C1: If security mode 1 is the only security mode that is supported, support for authentication is optional, otherwise mandatory. (Note: support for LMP-authentication and LMP-pairing is mandatory according [2] independent of which security mode that is used.)			
C2: If security mode 1 is not the only security mode that is supported, then support for at least one of security mode 2 or security mode 3 is mandatory.			

Table 5.1: Conformance requirements related to the generic authentication procedure and the security modes defined in this section

5.1 AUTHENTICATION

5.1.1 Purpose

The generic authentication procedure describes how the LMP-authentication and LMP-pairing procedures are used when authentication is initiated by one Bluetooth device towards another, depending on if a link key exists or not and if pairing is allowed or not.

5.1.2 Term on UI level

'Bluetooth authentication'.

5.1.3 Procedure

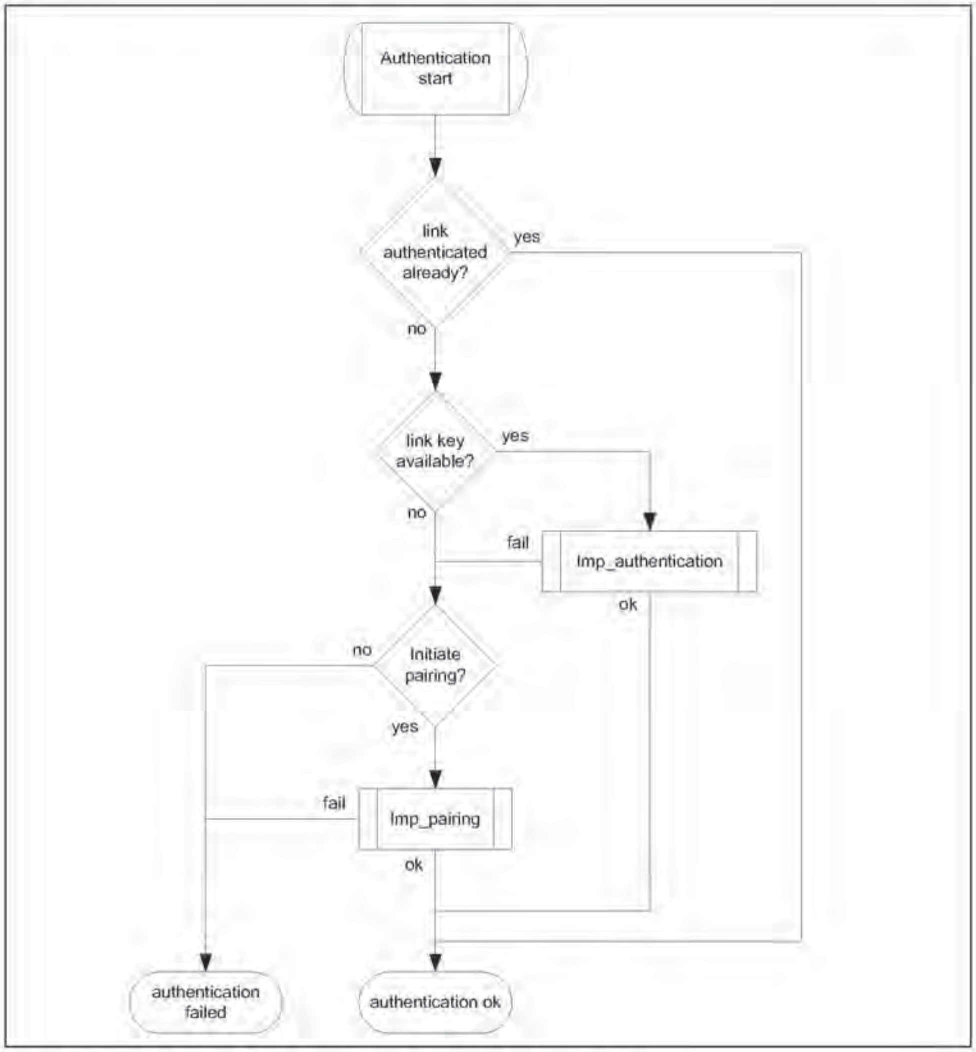


Figure 5.1: Definition of the generic authentication procedure.

5.1.4 Conditions

The device that initiates authentication has to be in security mode 2 or in security mode 3.

5.2 SECURITY MODES

The following flow chart describes where in the channel establishment procedures initiation of authentication takes place, depending on which security mode the Bluetooth device is in.

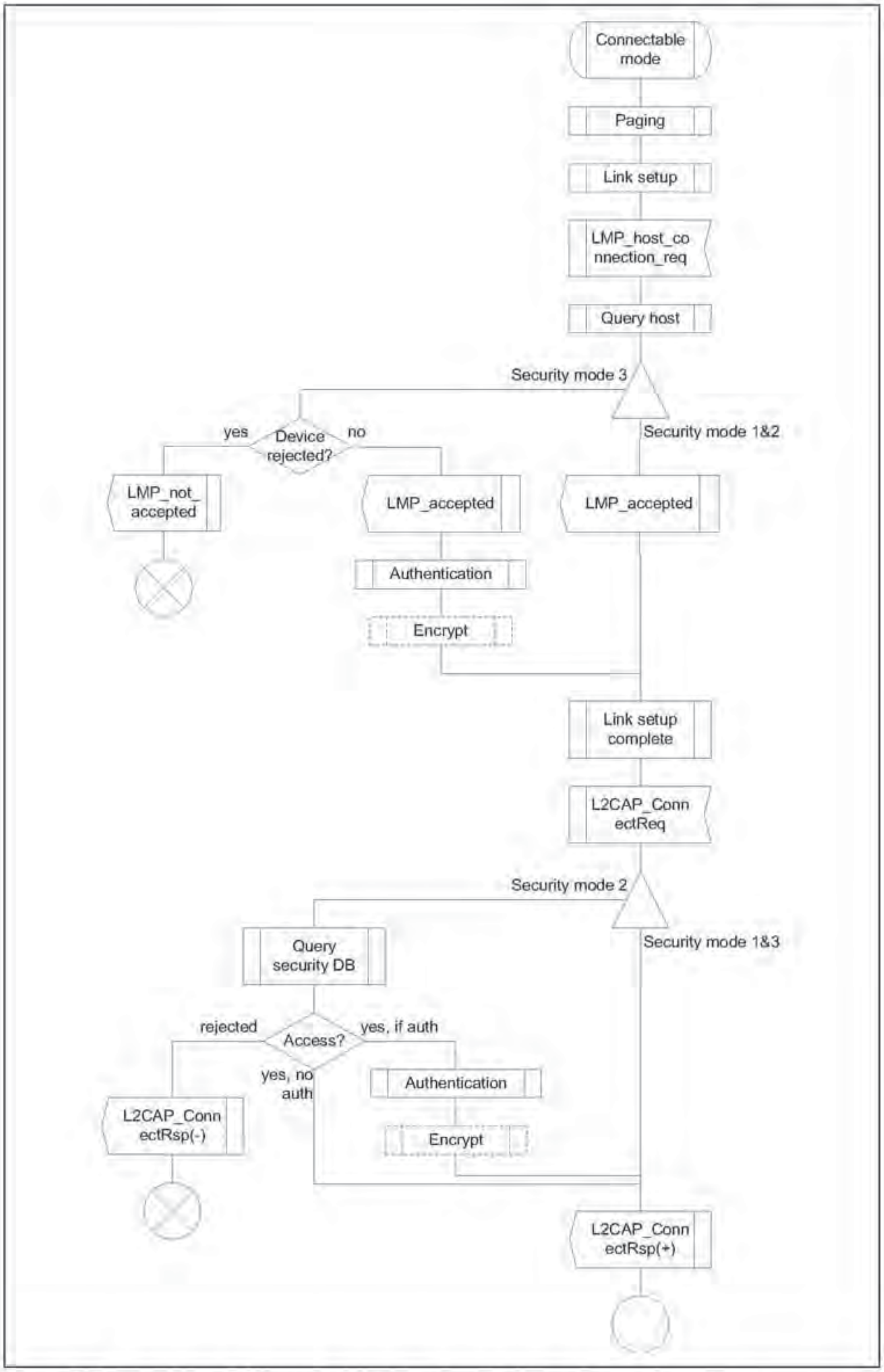


Figure 5.2: Illustration of channel establishment using different security modes.

When authentication is initiated towards a Bluetooth device, it shall act according to [2] and the current pairing mode, independent of which security mode it is in.

5.2.1 Security mode 1 (non-secure)

When a Bluetooth device is in security mode 1 it shall never initiate any security procedure (i.e., it shall never send LMP_au_rand, LMP_in_rand or LMP_encryption_mode_req).

5.2.2 Security mode 2 (service level enforced security)

When a Bluetooth device is in security mode 2 it shall not initiate any security procedure before a channel establishment request (L2CAP_ConnectReq) has been received or a channel establishment procedure has been initiated by itself. (The behavior of a device in security mode 2 is further described in [10].) Whether a security procedure is initiated or not depends on the security requirements of the requested channel or service.

A Bluetooth device in security mode 2 should classify the security requirements of its services using at least the following attributes:

- Authorization required;
- Authentication required;
- Encryption required.

Note: Security mode 1 can be considered (at least from a remote device point of view) as a special case of security mode 2 where no service has registered any security requirements.

5.2.3 Security modes 3 (link level enforced security)

When a Bluetooth device is in security mode 3 it shall initiate security procedures before it sends LMP_link_setup_complete. (The behavior of a device in security mode 3 is as described in [2].)

A Bluetooth device in security mode 3 may reject the host connection request (respond with LMP_not_accepted to the LMP_host_connection_req) based on settings in the host (e.g. only communication with pre-paired devices allowed).

6 IDLE MODE PROCEDURES

The inquiry and discovery procedures described here are applicable only to the device that initiates them (A). The requirements on the behavior of B is according to the modes specified in Section 4 and to [2].

	Procedure	Ref.	Support
1	General inquiry	6.1	C1
2	Limited inquiry	6.2	C1
3	Name discovery	6.3	O
4	Device discovery	6.4	O
5	Bonding	6.5	O
C1: If initiation of bonding is supported, support for at least one inquiry procedure is mandatory, otherwise optional. (Note: support for LMP-pairing is mandatory [2].)			

6.1 GENERAL INQUIRY

6.1.1 Purpose

The purpose of the general inquiry procedure is to provide the initiator with the Bluetooth device address, clock, Class of Device and used page scan mode of general discoverable devices (i.e. devices that are in range with regard to the initiator and are set to scan for inquiry messages with the General Inquiry Access Code). Also devices in limited discoverable mode will be discovered using general inquiry.

The general inquiry should be used by devices that need to discover devices that are made discoverable continuously or for no specific condition.

6.1.2 Term on UI level

'Bluetooth Device Inquiry'.

6.1.3 Description

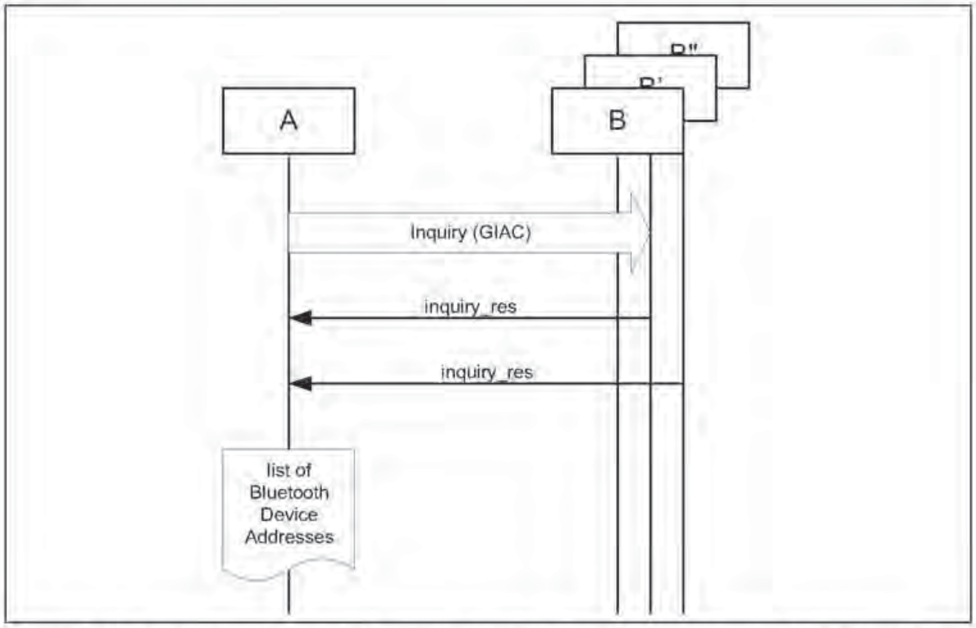


Figure 6.1: General inquiry ,where B is a device in non-discoverable mode, B´ is a device in limited discoverable mode and B" is a device in general discoverable mode. (Note that all discoverable devices are discovered using general inquiry, independent of which discoverable mode they are in.)

6.1.4 Conditions

When general inquiry is initiated by a Bluetooth device, it shall be in the INQUIRY state for at least $T_{GAP}(100)$ and perform inquiry using the GIAC.

In order to receive inquiry response, the remote devices in range have to be made discoverable (limited or general).

6.2 LIMITED INQUIRY

6.2.1 Purpose

The purpose of the limited inquiry procedure is to provide the initiator with the Bluetooth device address, clock, Class of Device and used page scan mode of limited discoverable devices. The latter devices are devices that are in range with regard to the initiator, and may be set to scan for inquiry messages with the Limited Inquiry Access Code, in addition to scanning for inquiry messages with the General Inquiry Access Code.

The limited inquiry should be used by devices that need to discover devices that are made discoverable only for a limited period of time, during temporary conditions or for a specific event. Since it is not guaranteed that the discoverable device scans for the LIAC, the initiating device may choose any inquiry procedure (general or limited). Even if the remote device that is to be discovered is expected to be made limited discoverable (e.g. when a dedicated bonding is to be performed), the limited inquiry should be done in sequence with a general inquiry in such a way that both inquiries are completed within the time the remote device is limited discoverable, i.e. at least $T_{GAP}(103)$.

6.2.2 Term on UI level

'Bluetooth Device Inquiry'.

6.2.3 Description

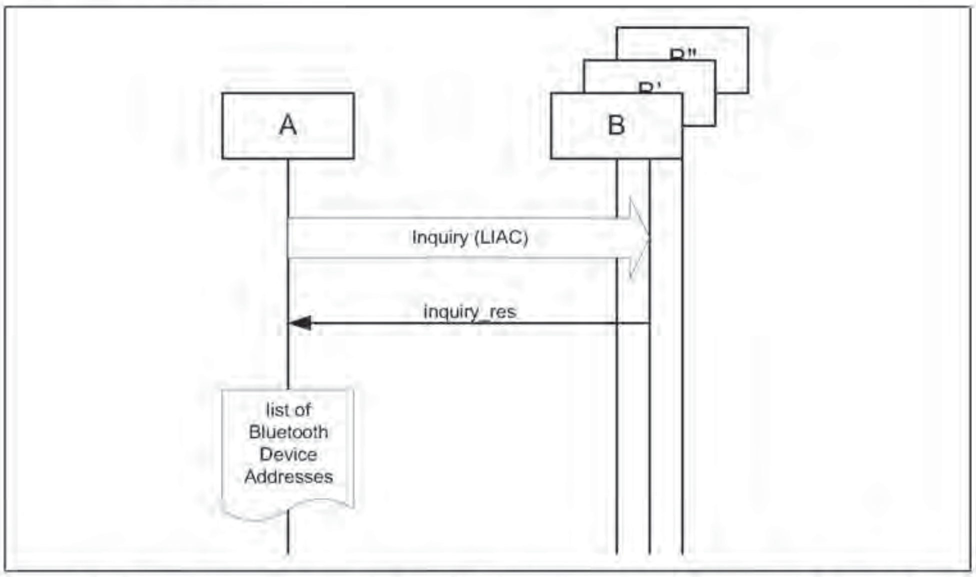


Figure 6.2: Limited inquiry where B is a device in non-discoverable mode, B' is a device in limited discoverable mode and B" is a device in general discoverable mode. (Note that only limited discoverable devices can be discovered using limited inquiry.)

6.2.4 Conditions

When limited inquiry is initiated by a Bluetooth device, it shall be in the INQUIRY state for at least $T_{GAP}(100)$ and perform inquiry using the LIAC.

In order to receive inquiry response, the remote devices in range has to be made limited discoverable.

6.3 NAME DISCOVERY

6.3.1 Purpose

The purpose of name discovery is to provide the initiator with the Bluetooth Device Name of connectable devices (i.e. devices in range that will respond to paging).

6.3.2 Term on UI level

'Bluetooth Device Name Discovery'.

6.3.3 Description

6.3.3.1 Name request

Name request is the procedure for retrieving the Bluetooth Device Name from a connectable Bluetooth device. It is not necessary to perform the full link establishment procedure (see Section 7.1) in order to just to get the name of another device.

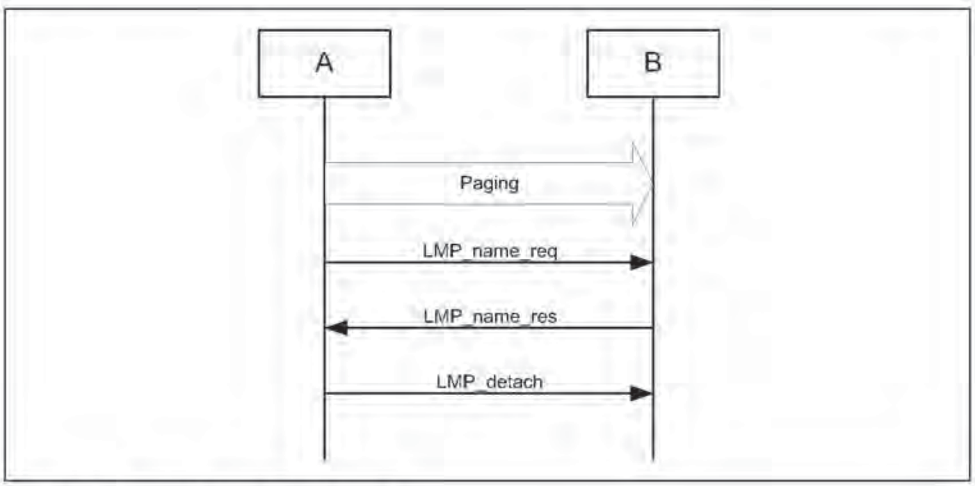


Figure 6.3: Name request procedure.

6.3.3.2 Name discovery

Name discovery is the procedure for retrieving the Bluetooth Device Name from connectable Bluetooth devices by performing name request towards known devices (i.e. Bluetooth devices for which the Bluetooth Device Addresses are available).

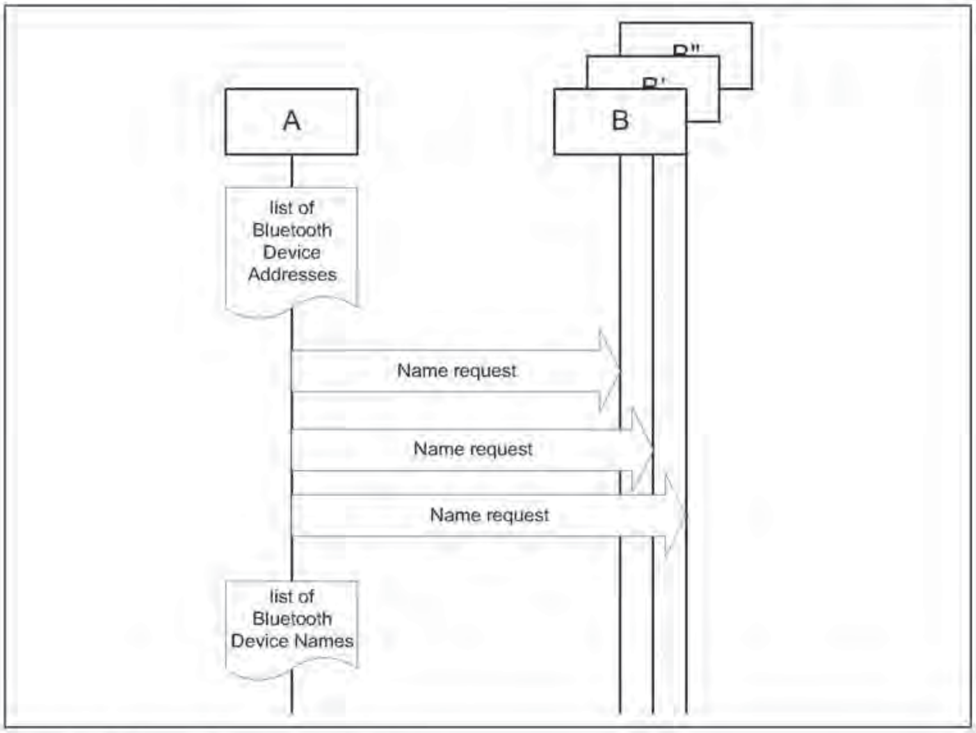


Figure 6.4: Name discovery procedure.

6.3.4 Conditions

In the name request procedure, the initiator will use the Device Access Code of the remote device as retrieved immediately beforehand – normally through an inquiry procedure.

6.4 DEVICE DISCOVERY

6.4.1 Purpose

The purpose of device discovery is to provide the initiator with the Bluetooth Address, clock, Class of Device, used page scan mode and Bluetooth device name of discoverable devices.

6.4.2 Term on UI level

'Bluetooth Device Discovery'.

6.4.3 Description

During the device discovery procedure, first an inquiry (either general or limited) is performed, and then name discovery is done towards some or all of the devices that responded to the inquiry.

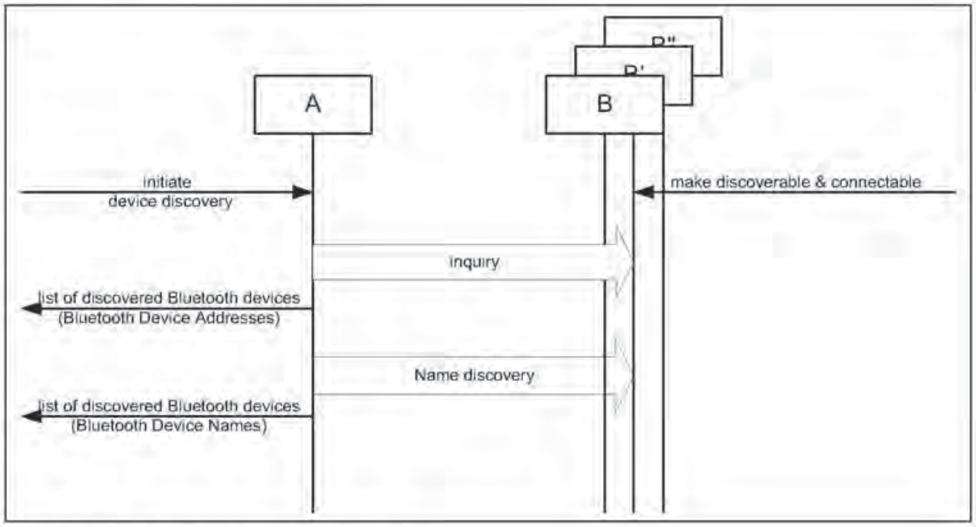


Figure 6.5: Device discovery procedure.

6.4.4 Conditions

Conditions for both inquiry (general or limited) and name discovery must be fulfilled (i.e. devices discovered during device discovery must be both discoverable and connectable).

6.5 BONDING

6.5.1 Purpose

The purpose of bonding is to create a relation between two Bluetooth devices based on a common link key (a bond). The link key is created and exchanged (pairing) during the bonding procedure and is expected to be stored by both Bluetooth devices, to be used for future authentication.

In addition to pairing, the bonding procedure can involve higher layer initialization procedures.

6.5.2 Term on UI level

'Bluetooth Bonding'

6.5.3 Description

Two aspects of the bonding procedure are described here. Dedicated bonding is what is done when the two devices are explicitly set to perform only a creation and exchange of a common link key.

General bonding is included to indicate that the framework for the dedicated bonding procedure is the same as found in the normal channel and connection establishment procedures. This means that pairing may be performed successfully if A has initiated bonding while B is in its normal connectable and security modes.

The main difference with bonding, as compared to a pairing done during link or channel establishment, is that for bonding it is the paging device (A) that must initiate the authentication.

6.5.3.1 General bonding

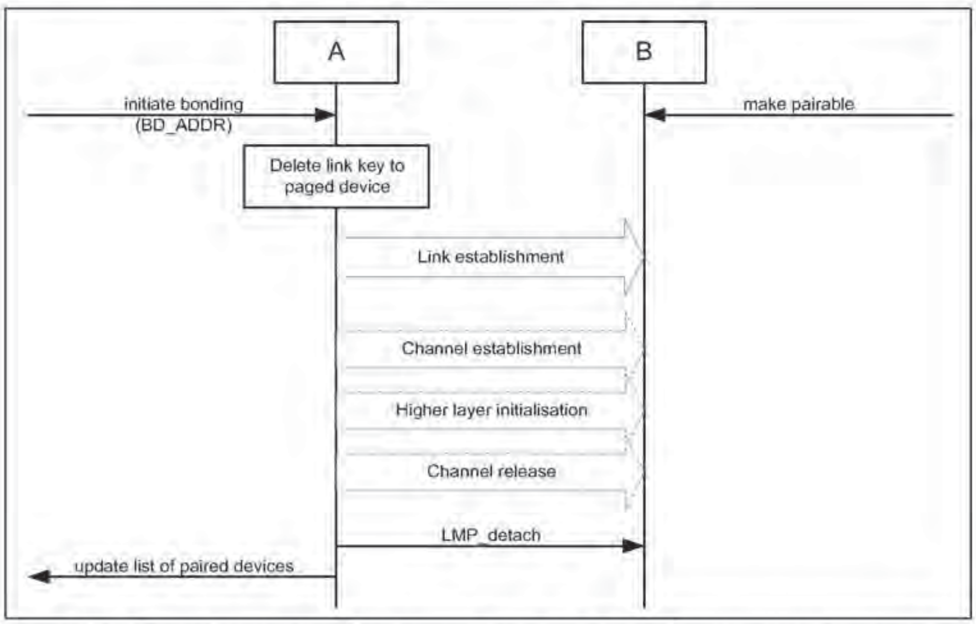


Figure 6.6: General description of bonding as being the link establishment procedure executed under specific conditions on both devices, followed by an optional higher layer initalization process.