LMX9820/LMX9820A Bluetooth Serial Port Module - Software Users Guide

National Semiconductor
Software Users Guide
October 2006
Revision 1.6.4

Introduction

The National Semiconductor® LMX9820A Bluetooth Serial Port Module is a highly integrated radio, baseband, and memory implemented on a FR4 substrate. All hardware and firmware is included to provide a complete solution from antenna through the complete lower and upper layers of the bluetooth stack up to the application including the Generic Access Profile (GAP), the Service Discovery Application Profile (SDAP) and the Serial Port Profile (SPP). The module includes a configurable service database to fulfill service requests for additional profiles on the host.

The module offers an automatic slave mode without any configuration necessary from an external host. Additionally it offers a command set for hardware configuration and SPP full bluetooth operation.

This document is a reference for implementing the LMX9820A module into a system. A getting started session gives a very detailed entry point for starting development. The advance usage section describes all features and configuration parameters in detail and gives example for using the LMX9820A as active bluetooth node. Finally all commands and events are listed and explained in the command section.

This document is based on:

Table 0-1.

Item	Version
Hardware	LMX9820V5.2 LMX9820SB LMX9820ASM
Firmware	V5.05 and later
Actual Firmware Release	V6.23
SimplyBlueCommander	1.6.0.1

IMPORTANT:

LMX9820SB products delivered with software versions earlier than v5.07 require initialization to factory default settings (see Section 3.1.1.4 “Restore Factory Settings” on page 35).

Afterwards the parameters in flash are programmed for proper functionality.

This procedure only has to be done once before first usage of the device.

Table of Contents

1.0 LMX9820A Features

1.1 SYSTEM OVERVIEW

The LMX9820A is intended to be an add-on module to an existing microcontroller. In this function it either appears as cable like interface for the UART or can also be controlled with a simple application on the external microcontroller to establish links itself.

The LMX9820A includes the complete bluetooth stack including the following protocol layers.

- Link Controller
- Link Manager
- L2CAP (Logic Link Control and Adaptation)
- RFCOMM
- SDP (Service Discovery Protocol)

An on-chip application offers together with those protocol layers the following profiles:

- GAP (Generic Application Profile)
- SDAP (Service Discovery Application Profile)
- SPP (Serial Port Profile)

The application is managing all profile related interactions to the stack but also offers a simplified command interface over the UART. The interface is used for configuration of the device, setting up the link and receiving events from the module. The interface can handle either packaged data transmission for multipoint support or is able to handle RAW data by setting it into a transparent UART mode.

The firmware can also accept or establish synchronous links (SCO) to transmit audio data. Once the link has been established the firmware will route the synchronous data to and from the PCM interface, using predefined driver settings. No further host action required.

.

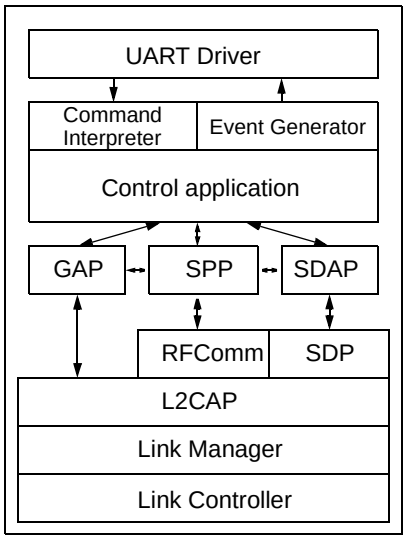


Figure 1-1. LMX9820A Firmware Implementation

1.2 SYSTEM PARAMETERS

The LMX9820A includes a non-volatile memory area to store system and bluetooth parameters. These parameters define the chip behaviour during bluetooth operation but also after a hardware or software reset. The values are configured by special commands through the UART interface. Please see Table 1-1 for a complete list of parameters.

Table 1-1. LMX9820A System Parameters

No.	Address	Parameter	Default Value	Description	Checked
1	0000-0005	BD_Addr	<prepro-grammed>	Bluetooth Device Address. On default prepro-grammed by National.	Runtime
2	0006	CountryCode	0x00	Used by BT core	Boot-up
3	0007	UnitKeyPre-sent	0xFF	Used by BT core, generated during pairing pro-cedure.	Runtime
4	0008-0017	UnitKey	0xFF..0xFF	Used by BT core, generated during pairing pro-cedure.	Runtime
5	001C	DeviceName-Length	0xFF	Length of Parameter 6 “Devicename”	Runtime
6	001D-0044	Device Name	‘Serial Port Device’	Friendly Name of the Bluetooth Device	Runtime
7	0045-0284	Link Keys	0xFF..0xFF	Link Keys for paired devices	Runtime
8	0285-0293	AssertInfo	0xFF..0xFF	NSC configured parameter, do not modify	Runtime
9	0294-02A5	RunErrorInfo	0xFF..0xFF	NSC configured parameter, do not modify	Runtime
10	02A6	Pin Length	0x04	The length of parameter 9, “PinCode”. In case set to 0, the LMX9820A will request pin from host.	Runtime
11	02A7-02B6	Pin Code	“0000”	Fixed PinCode used for pairing with other devices	Runtime
12	02B7-02B9	ClassOfDevice	0x000000	The ’Class of Device’ describes general func-tionality of the Bluetooth Device and is transmit-ted during the Inquiry process.	Runtime
13	02BE-034D	SDP Records	1 SPP Entry, Auth and Encr. Enabled	The Service Discovery Database of the Device. The database can be configured via Configura-tion Commands. On default one SPP port con-figured.	Runtime
14	0350-0353	SPP Ports to Open	0000 0001	Defines the RFCOMM channels to open. For each channel one RFCOMM instance will be created.	Runtime
15	0354-036E	Default Con-nections	<empty>	Default connections database, to be connected during boot-up or by sending a command.	Runtime
16	036F	PreferredMas-terRole	0x00, False	Preferred Master forces the device to switch to Master Role after being connected. The device will reject the link if command could not be exe-cuted.	Boot-up
17	0370	UART Parity	0x00, No Parity	Parity Setting for the Hardware UART Interface	Boot-up
18	0371	UARTNoofS-topbits	0x00, 1 Stopbit	Number of Stop Bits used on Hardware UART Interface	Boot-up
19	0372	UART speed	0x03, 9600kbit/s	Speed of the Hardware UART Interface	Boot-up
20	0373	Operation Mode	0x01, Automatic	Configures the general behaviour of the device. Please see Section 1.6 "Operation Modes" for details.	Boot-up
21	0374	Page Scan Mode	0x01, Enabled	Sets the connectability of the device. (enabled/ disabled)	Runtime

Table 1-1. LMX9820A System Parameters

No.	Address	Parameter	Default Value	Description	Checked
22	0375	Inquiry Scan Mode	0x01, General Inquriy	Sets the discoverability mode of the device.	Runtime
23	0376	Security Mode	0x02	Configures Service Level Security Mode.	Runtime
24	0377-0378	Default Link Policy	0x000F	Configures the default link policy for incoming and outgoing links.	Runtime
25	0379	EventFilter	0x01	Configures the level of events reported to the host. 0x00: No filter, all events reported 0x01: ACL events filtered, only API events reported. 0x02: All events filtered, only UART breaks indicated 0x03: All events filtered, including UART break.	Runtime
26	037A	PMM usage	0xFF	Bitmask to configure enhanced power management (PMM) functions 0x01: enhanced PMM: 1 - disabled (default); 0 - enabled (requires 32khz connected and external clock source like TCXO, don't use with crystal) All other bits are reserved and should be set to 1	Boot-up
27	037B	NVS Initialized	0x00	If not 0x00, device restores all values to these default values. Usually only happens during very first boot process.	Boot-up
28	037C-037D	Default Link Timeout	0x7D00, 20seconds	Configures the default link supervision timeout used for any incoming and outgoing link	Runtime
29	037E	CodecType	0x00	The audio codec settings used on the PCM interface: 0x00: None connected 0x01: Motorola MC145483 0x02: OKI MSM7717 0x03: PCM slave, see PCMSlaveConfig 0x04-0xFF: reserved	Boot-up
30	037F	AirFormat	0x00	The audio format used on the SCO link 0x00: CVSD 0x01: u-Law 0x02: A-Law 0x03-0xFF: reserved	Boot-up
31	0380-0381	Default Link Latency	0x0000	Configures the default poll period of master to slave. 0x0000: No requirement (default 40slots) 0x0002-0x0190: Valid link latency	Runtime

Table 1-1. LMX9820A System Parameters

No.	Address	Parameter	Default Value	Description	Checked
32	0382-0383	PcmSlaveConfig	0xFFFF	This 16-bit value (LSB first) is used to store the PCM format configuration for the PCM slave configuration. This setting is only use in case the PCM slave setting is activated (see No. 21). BIT0-1, Slot selection 00: use slot 0 01: use slot 1 10: use slot 2 11: use slot 3 BIT2-3: Number of slots per frame 00: 1 slot 01: 2 slots 10: 3 slots 11: 4 slots BIT4-6: PCM data format 000: Reserved 001: 8 bit A-law 010: 8 bit u-law 011: 13 bit linear 100: 14 bit linear 101: 15 bit linear 110: 16 bit linear 111: Reserved BIT7:Frame sync length 0: short frame sync 1: long frame sync BIT8: Data word length 0: 8-bit data word length 1: 16-bit data word length BIT9: Frame sync polarity 0: use inverted frame sync 1: use normal frame sync BIT10-15: Unused, set to 0	Boot-up
33	0384	PcmFcprs	0xFF	Unsigned integer indicating the frame clock prescaler for generic PCM slave	Boot-up
34	0385-0388	RfSetupReg4	0xFFFFFFFF	NSC configured parameter, do not modify	Boot-up
35	0389-038C	RfSetupReg15	0xFFFFFFFF	NSC configured parameter, do not modify	Boot-up
36	038D-0396	Reserved		NSC configured parameters, do not modify	

1.3 UART TRANSPARENT MODE

On default the LMX9820A is listening on the UART to a special package format described in chapter 4. Any incoming data will be analyzed and data or parameters will be extracted.
In case the LMX9820A has established a link to only one remote device and no configuration commands have to be sent to the LMX9820A, the UART interface can be switched to transparent mode.
This means data are directly routed to the bluetooth link and not interpreted. Also incoming data are not indicated as events, they are sent as RAW data to the UART.

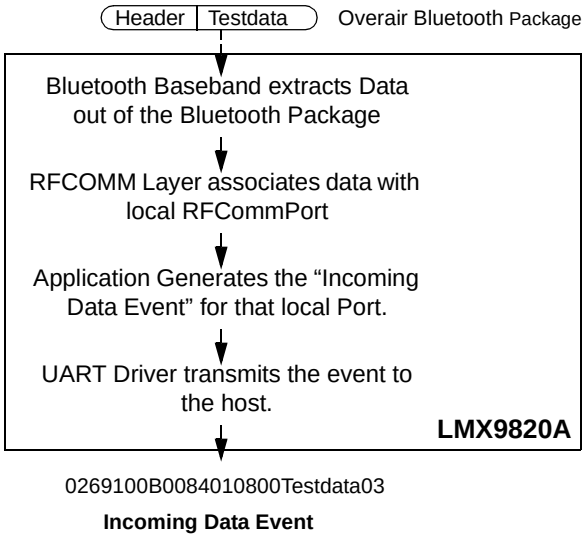


Figure 1-2. Receiving data in command mode

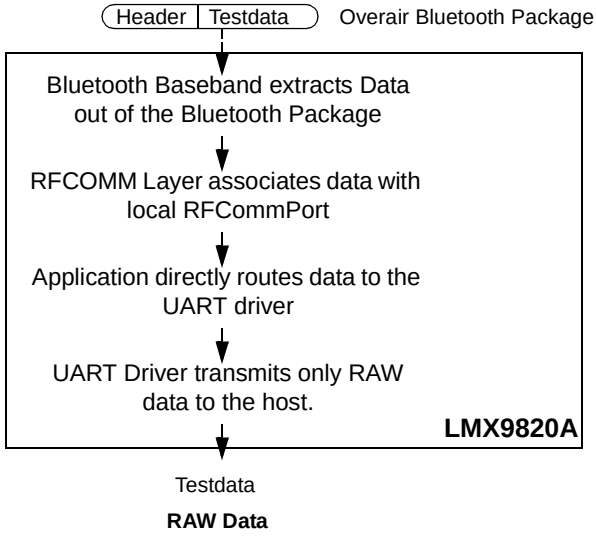


Figure 1-3. Receiving Data in Transparent Mode

Leaving transparent mode:

As the LMX9820A does not listen to commands, UART Break has to be used to tell the device to leave the transparent mode. See also Section 5.2.4.1 "Transparent Mode" on page 116 for details.

1.4 DEFAULT CONNECTIONS

The LMX9820A is able to store up to 3 connections into its own database. The parameters stored within the device are:

- Index
- Local RFCommPort
- BD_Addr
- Remote RFCommPort
- Transparent Mode (yes/no)
- Sniff Mode (yes/no). For Sniff Mode details, see “The LMX9820A power management is a combination of the firmware and the hardware supported low power modes. Depending on the system activity it decides to switch off as many hardware blocks as possible to reduce the current consumption.” (page 48).

Each connection can be addressed by its index. The connection to those devices can be established by the “Connect Default Connection” command.

If the device is in Automatic mode (parameter 13 in Table 1-1), the device will try to connect to those devices after boot-up or reset. The different modes are described Section 1.6 “Operation Modes” on page 14.

1.5 THE COMMAND INTERFACE

The LMX9820A offers a wide range of commands to configure the hardware and the bluetooth operation. As the command set is on top of the profiles, Bluetooth operational commands are reduced to high level commands controlling general bluetooth operation.

The commands have to be sent in a specific package format. The interface is based on an event mechanism. Any command sent will be confirmed by the appropriate confirmation event. Unexpected events will be reported by indication events or requests.

Please refer to Section 5.0 “LMX9820A Command Interface” on page 100 for a complete list of commands and their usage.

1.6 OPERATION MODES

The operation of the LMX9820A can be divided into different modes. Each mode represents special situations and describes the behaviour of the module.

The following modes can be defined for the LMX9820A:

- Automatic Idle
- Automatic Sniff
- Command Idle
- Command Mode Master
- Command Mode Slave
- Transparent Mode

The definition for the operation mode “automatic or command” mode is determined out of the parameter “operation mode” in the NVS (see Table 1-1) and has influence on the behaviour of the LMX9820A in different situations.

Figure 1-4 on page 16 describes the behaviour of the LMX9820A after boot-up or reset.
Figure 1-5 on page 17 and Figure 1-6 show the operation flow, if the LMX9820A is contacted by a remote device either in automatic or in command operation.
Figure 1-7 finally shows the operation as master using the command interface. For this operation there’s no difference if the device is configured to automatic or command mode.

1.6.1 Automatic/Command Idle

On default, e.g. after boot-up with no connection established, the LMX9820A stays within one of the Idle Modes. This means the device on default is visible for other devices, connectable and is answering to service requests.

Settings for connectablility and discoverability as well as the Service database are stored within the NVS and can be changed by commands.

1.6.2 Command Master Mode

The command interface can be used to inquire, doing service requests on the remote device and establishing connections. If the link has been established by the LMX9820A commands, it will be in the “Command Master” Mode. In this mode the device acts as master within the link, beeing able to set up a piconet with other devices and still listening to commands on the UART. Depending on the parameters for scanmode stored in NVS, the device still listens to requests on the radio.

1.6.3 Automatic/Command Slave Mode

If connected by another device, the LMX9820A will ask for authentication and if necessary pairing will be processed automatically using the stored pincode.
After successful connection setup the LMX9820A indicates the link establishment by sending the event “Incoming Link Established” (see Section 5.2.3.7) and by setting pin LSTAT1 to 1.

Depending on the Operation Mode (Parameter #13 in Table 1-1) the module will switch to one of the following modes:

- Operation Mode Automatic:
  LMX9820A automatically switches to Transparent Mode on the UART
- Operation Mode Automatic Sniff
  LMX9820A automatically switches to Transparent Mode on the UART, and put the incoming link into Sniff mode. The Sniff parameters used Should be set by the command “Set Default Sniff Mode Parameters”. For more informations about the Sniff mode, refer to “The LMX9820A power management is a combination of the firmware and the hardware supported low power modes. Depending on the system activity it decides to switch off as many hardware blocks as possible to reduce the current consumption.” (page 48).
- Operation Mode Command:
  LMX9820A switches to Command Slave Mode, still trying to interpret the data on the UART as commands. Incoming data will be indicated by the “Incoming Data” event.

If the link is lost the module will indicate this by setting LSTAT1 to 0 and sending a “Link Released” indicator to the host. In case the UART has been in transparent mode, an UART BREAK signal will be sent first.

1.6.4 Transparent Mode

If the LMX9820A established a link to only one remote device, it can be switched to UART transparent mode. (see also "UART Transparent Mode" on page 13).

In this mode the LMX9820A will not listen to commands anymore. The device will also not respond to inquiry or connection requests from remote devices.

Table 1-2. Operation Mode Features

Parameter	Automatic Idle	Automatic Sniff	Command Idle	Command Master	Command Slave	Transparent
P. #13 in NVS	Automatic	Automatic	Command	both	both	both
Discoverable[1]	yes	yes	yes	yes	no	no
Connectable[1]	yes	yes	yes	yes	no	no
Master/Slave	-	-	-	master	slave	both
Search for devices	yes	yes	yes	yes	yes[2]	no
Connect to remote devices (actively)	yes	yes	yes	yes	yes[2]	no
Send Raw Data	-	-	-	no	no	yes
Multipoint connections	-	-	-	yes	yes[2,3]	no
Request Sniff mode on incoming link	no	yes	no	no	no	no

1. Depending on parameters #14 and #15 in the non-volatile storage (default setting assumed).
2. Only if the active link is switched to one of the Low Power Modes (Sniffmode, Holdmode or Parkmode)
3. Master and Slave Scatternet scenario can only be supported temporarily, eg. in access point scenarios.

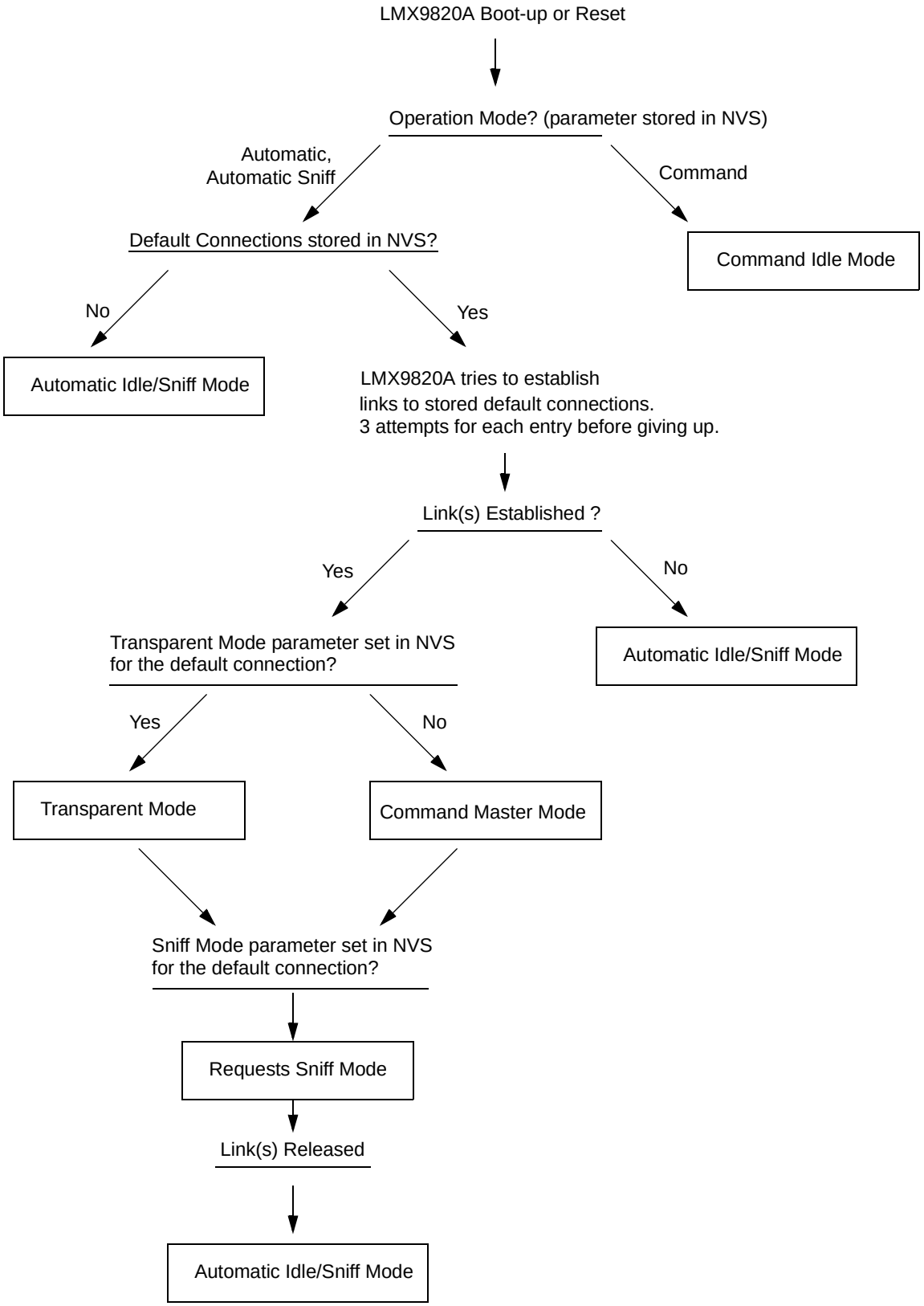


Figure 1-4. Operation Flow after boot-up or Reset

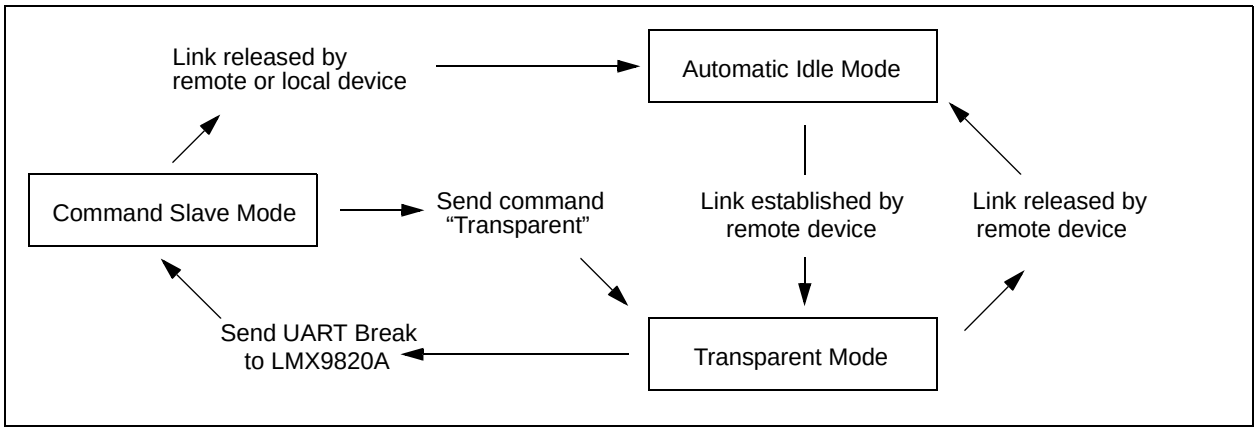


Figure 1-5. Automatic Slave Operation Flow

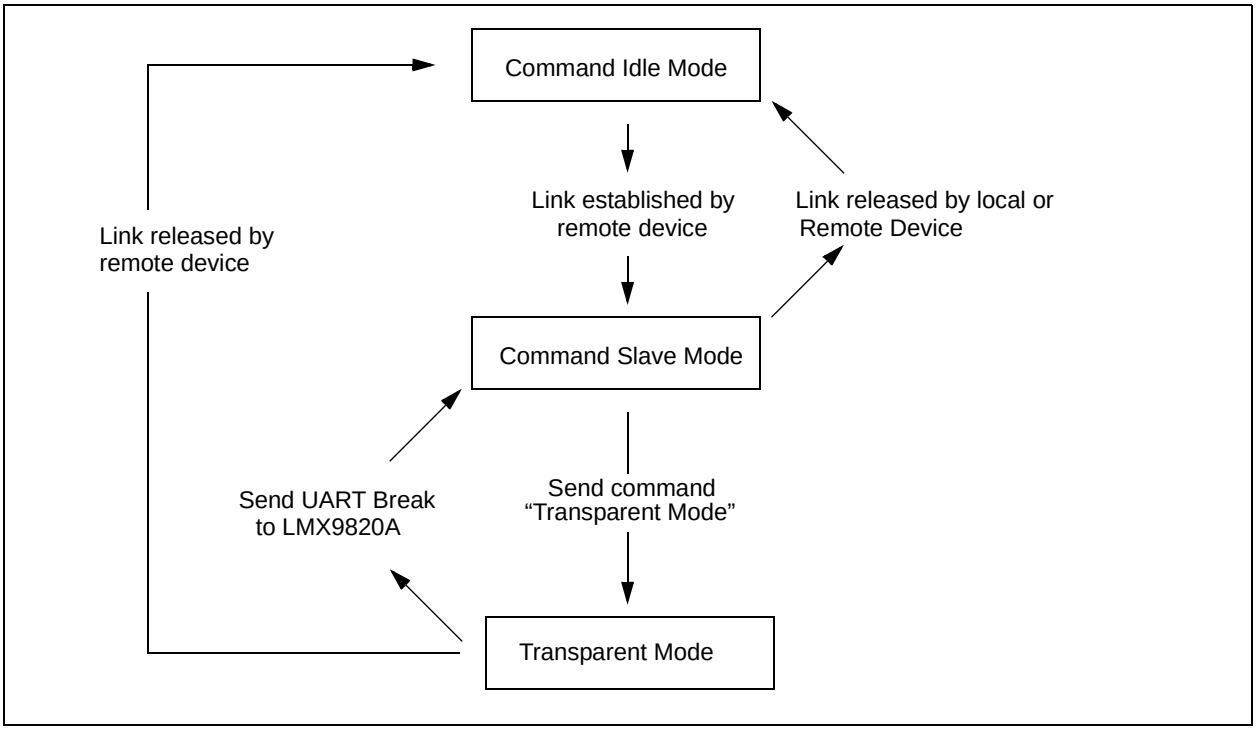


Figure 1-6. Command Slave Operation Flow

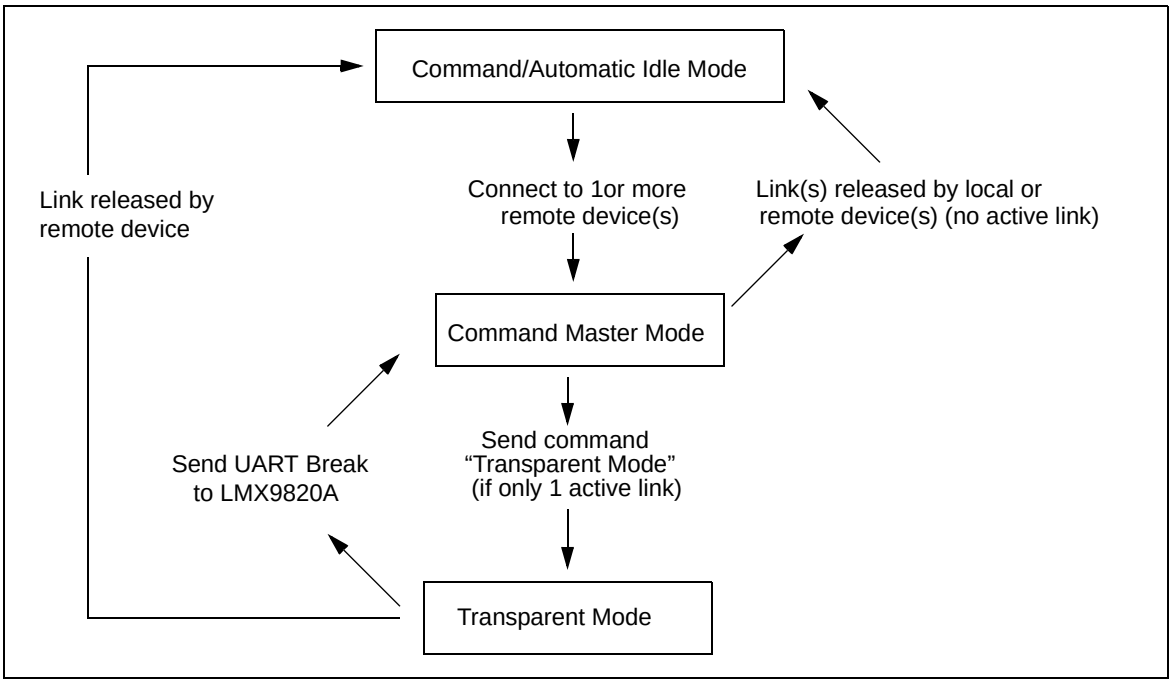


Figure 1-7. Command Master Operation Flow

1.7 AUDIO SUPPORT

Firmware versions later than 6.00 include additional commands allowing to establish or accept synchronous audio links. The audio data will be routed within the LMX9820A between the bluetooth baseband connection and the Advanced Audio interface. In order to encode or decode the PCM stream generated by the internal CVSD codec, an external codec or DSP has to be connected to the advanced audio interface. Alternatively the data can be used to be for further digital processing.

The AAI is configured by predefined codec settings, selected in NVS. The NVS setting configures the bitrate as well as the format like PCM log or linear. In addition on Firmware revision 6.23 or later, the interface can be configure to PCM Slave in which the external codec or DSP needs to provide the symchronous clock and frame sync signal. The settings for the PCM Slave mode are done by specific commands. See also “PCM codec configuration” (page 59).

Please refer to the LMX9820A Datasheet for detailed codec configurations.

The Bluetooth standard defines CVSD, u-Law and A-Law to be used as format on the bluetooth link. This settings is also configured within the NVS.

Figure 1-8 shows a typical application block diagram, how to connect the audio codec.

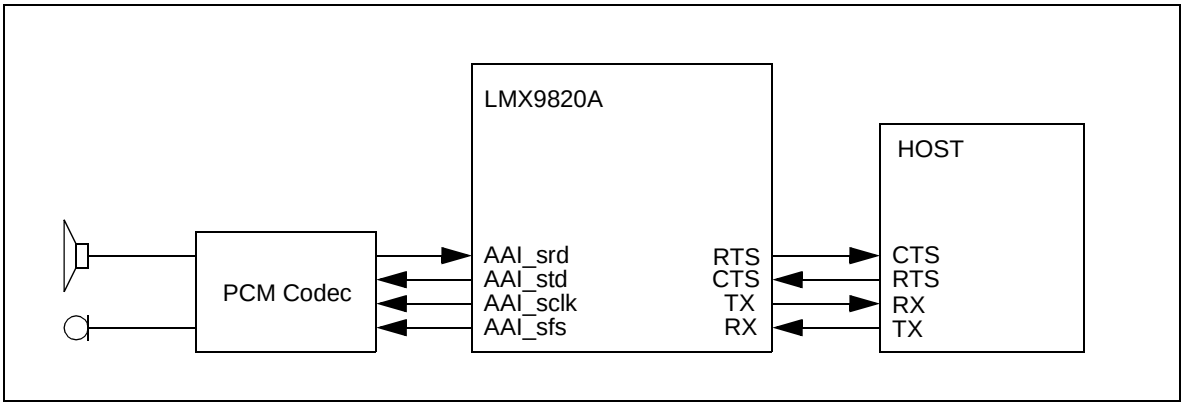


Figure 1-8. PCM Codec connection block diagram

2.0 Getting Started

This description is using Simply Blue Commander Log Entries to show the command structure and their results.

The log file is adding new commands on top of the screen, so all log file pictures show the first entry at the bottom of the picture, the last entry at the top.

2.1 THE UART COMMAND INTERFACE

The LMX9820A uses the UART interface to interact with a host. The UART interface between host and LMX9820A needs to be connected in Null Modem configuration, meaning RTS/CTS and TX/RX are crossed.

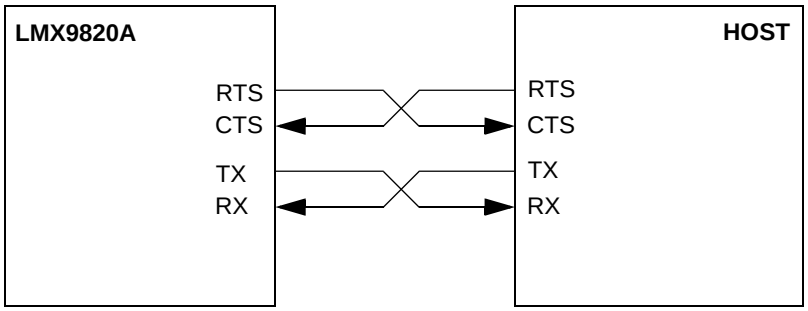


Figure 2-1. UART Null Modem connection

The command interface is based on a command/event based structure. Each command (also called “Request”) will be acknowledged with the appropriate status event (also called “Confirm”). Unexpected events, like incoming link establishment or data, are also sent as events, but signed with a different simply blue package type called “Indicator”. Please see Section 5.0 "LMX9820A Command Interface" on page 100 for the detailed description of the command interface.

The following sections shall give a short introduction into the basic link establishment.

2.2 USING THE AUTOMATIC SLAVE OPERATION

As described in Section 1.6 the LMX9820A on default is in a waiting mode (Automatic idle mode) after boot-up or reset. This means it waits for requests and automatically answers to connection requests.

If connected from a remote device, the LMX9820A establishes automatically a SPP link and indicates the established link to host by the “Link Established Event” and by setting LSTAT1 pin to 1.

Rx: Event: Incoming Link Established, BdAddr: BF8C03029000, Local Port: 01

Figure 2-2. Log of Incoming Link Established

Rx(RAW): 02,69,0C,07,00,7C,BF,8C,03,02,90,00,01,03

Figure 2-3. Hex Log of Incoming Link Established

The event indicates the local RFComm Port and the BD_Addr of the remote device.

Table 2-1. Example Incoming Link Established

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	69 0C 07 00 7C
07- 12	BD_Addr of remote device	BF 8C 03 02 90 00
13	Local RFComm Port	01
14	End Delimiter	03

After this event, the module automatically switches to transparent mode and routes all incoming and outgoing data from the RF side "unmodified" to the UART or vice versa. This switching process is not indicated to the host.

If the link is dropped, the LMX9820A will empty its buffers and

- send a UART break to the host,
- send "Transparent Mode" Indicator (Section 5.2.4.1 on page 116)
  — Indicates on protocol level to the host that transparent mode has been left.
- send "SPP Release Link" Indicator (Section 5.2.3.5 on page 114)
  — Indicates that link has been released.
- LSTAT1 pin will be set back to 0

```
Rx: Event: Link Released, Reason: 01, Local Port: 01
Rx: Event: Transparent Mode, Local Port: 01, Mode: 00
```

Figure 2-4. Log of Link Released Event

```
Rx(RAW): 02,69,0E,02,00,79,01,01,03
Rx(RAW): 02,69,11,02,00,7C,01,00,03
```

Figure 2-5. Hex Log of Link Released Event

The indicators "transparent mode" and "link released" report within their package the local RFComm port and the current mode respectively the reason of releasing the link.

Please see also Section 5.1 for a complete description of the package and header format.

Table 2-2. Example Transparent Mode Lost Indicator

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	69 11 02 00 7C
07	Local RFComm Port	01
08	Mode	00 (Command Mode)
09	End Delimiter	03

Table 2-3. Example Link Released Indicator

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	69 0E 02 00 79
07	Reason byte	01 (Remote device disconn.)
08	Local RFComm Port	01
09	End Delimiter	03

2.3 SETTING UP A LINK USING THE COMMAND INTERFACE

Setting up a bluetooth link between devices requires that the devices know specific parameters of each other. To get those parameters several steps have to be processed before a SPP link can be established to the device.

All commands necessary for this section can be found in “CreateSPPLink.dir” of the Simply Blue Commander.

The first commands will be explained very detailed for better understanding of the syntax and logic of the command interface.

Figure 2-6. CreateSPPLink.dir

2.3.1 Inquiry

The Inquiry process searches for devices in range and gets its BD_Addr (Bluetooth Device Address). This address is a unique address for each Bluetooth device on the market.
Also transmitted with it is the “Class of Device” of this device.

With LMX9820A this process can be started with the command “Inquiry” (Section 5.2.1.1).

The command results in two different events:

- Device Found Indicator
- Inquiry Complete Confirmation

Each found device will be indicated by the Device Found Indicator, including its BD_Addr and Class of Device.

The Inquiry Complete indicates the end of the Inquiry process. Figure 2-7 shows the log as interpreted by the Simply Blue Commander and Figure 2-8 the actual hex traffic on the UART.

```
Rx: Event: Inquiry, Status: 00
Rx: Event: Device Found, BdAddr: 469528D90A00, DeviceClass: 040252
Tx: Cmd: Inquiry, Length: 0A, NumResponces: 00, Mode: 00
```

Figure 2-7. Interpreted Log of an Inquiry

```
Rx(RAW): 02,43,00,01,00,44,00,03
Rx(RAW): 02,69,01,09,00,73,46,95,28,D9,0A,00,04,02,52,03
Tx(RAW): 02,52,00,03,00,55,0A,00,00,03
```

Figure 2-8. Hex Log of an Inquiry

Figure 2-8 shows the package format used on the Command interface. The TX indicates the Inquiry command sent to the device, the two Rx lines the events from the LMX9820A.

The following paragraphs explain the package format and usage in more detail. The complete package format is also described in Section 5.1.

a.) The Inquiry Command
Let’s first have a look on the Inquiry command:

02 52 00 03 00 55 0A 00 00 03

Any package, request or event, has a 6 byte header

- Startdelimiter (1 byte)

- Packet Type Identification (1 byte)
- Opcode (the actual command, 1 byte)
- Payload length (2 bytes)
- Checksum (1 byte)

In this case:

02 52 00 03 00 55

The **startdelimiter** is always 0x02.

The **packet type id** for a request is 0x52. (see Section 5.1.3 for the complete list of packet types)

The **opcode** for Inquiry is 00 (see Section 5.1.4 for the complete list of opcodes)

The **payload length** indicates literally the length of the payload after the checksum.

The payload for this package is

0A 00 00

so the length is 0x0003 (bytes).

The **checksum** is calculated as sum of packet type id, opcode and packet length,

0x52 + 0x00 + 0x03 + 0x00 = 0x55

The **payload** for this command consists of three parameters:

- Inquiry length - 0x0A (10 seconds)
- Number of responses - 0x00 (no limitation)
- Inquiry Mode - 0x00 (General Inquiry)

Table 2-4. Example Inquiry Command Package

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	52 00 03 00 55
07	Inquiry Length	0A (10 seconds)
08	Number of Responses	00
09	Inquiry Mode	00 (General Inquiry)
10	End Delimiter	03

All packages have to end with the **enddelimiter** 0x03.

b.) The Device Found Indicator

The first response to the inquiry command from the LMX9820A is the Device_Found_Indicator. In hex:

02 69 01 09 00 73 46 95 28 D9 0A 00 04 02 52 03

Package header:

- Startdelimiter - 0x02
- Packet Type - Indicator: 0x69
- Opcode - 0x01 (Indicator opcode, different from command opcode)
- Payload Length - 0x0009 (byte swapped in the package)
- Checksum - 0x69 + 0x01 + 0x09 + 0x00 = 0x73

The **Payload**:

46 95 28 D9 0A 00 04 02 52

- BD_Addr - 46 95 28 D9 0A 00
- Class of Device - 04 02 52

Because of the Little Endian format, both parameters have to be byte swapped. So the “real” values are:

BD_Addr: 00 0A D9 28 95 46

Class of Device: 52 02 04 (Mobile Phone)

Table 2-5. Example Device Found Indicator Package

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	69 01 09 00 73
07 - 12	BD_Addr	46 95 28 D9 0A 00
13 - 15	Class of Device	04 02 52
16	End Delimiter	03

c.) The Inquiry Confirm

Every command on the LMX9820A command interface is confirmed by an appropriate event. The confirmation always has the opcode as the command sent to the device. The event also indicates the success status of the command or any parameters requested. If no error occurred, error 0x00 will be returned. All other values have a specific reason. Please see Table 5-208 "Generic Error Codes" on page 182 for a complete list of error codes.

The confirmation in hex:

02 43 00 01 00 44 00 03

Package header:

- Startdelimiter - 0x02
- Packet type - confirm: 0x43
- Opcode - 0x00 (confirmation, same as command)
- Payload length - 0x0001 (byte swapped in the package)
- Checksum - 0x43 + 0x01 + 0x00 + 0x00 = 0x44

The payload of a confirmation consists at least of the status byte. In this case 0x00.

Table 2-6. Example Inquiry Confirm Package

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	43 00 01 00 44
07	Status/Error Code	00
08	End Delimiter	03

2.3.2 Create SDAP Connection

To create a SPP connection to another device, the local RFComm channel has to know which remote RFComm Channel to address. Each service is registered to a specific RFComm channel number. To get this number the local device has to do a Service Request on the remote device and get the service entry.

The first command necessary for this is the “Create SDAP Connection”. This command establishes a SDP based connection to the other device.

```
Rx: Event: SDAP Connect, Status: 00
Tx: Cmd: SDAP Connect, BdAddr: 469528D90A00
```

Figure 2-9. Log of the Create SDAP Command

```
Rx(RAW): 02,43,32,01,00,76,00,03
Tx(RAW): 02,52,32,06,00,8A,46,95,28,D9,0A,00,03
```

Figure 2-10. Hexadecimal Log of the Create SDAP Command.

Table 2-7. Example Create SDAP Connection

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Packet Header	52 32 06 00 8A
07 - 12	Remote BD_Addr	46 95 28 0D 0A 00
13	End delimiter	03

The only parameter of the command is the BD_Addr to connect to:

46 95 28 D9 0A 00 (byte swapped)

The command is confirmed by the LMX9820A with the appropriate confirmation event. If status is 0x00 the link has been established.

2.3.3 SDAP Service Browse for SPP

After the SDAP connection is established, the service request can be sent. To search for a remote SPP entry, UUID 1101 can be used.

As any multi-byte parameter the UUID has to be sent byte swapped to the LMX9820A within the command.

.

```
Rx: Event: Service Browse, Status: 00, Browse Group ID: 0210, Service ID: 0111, PortNo: 04,
Tx: Cmd: Service Browse, Browse Group ID: 0111
```

Figure 2-11. Log SDAP Browse for SPP

```
Rx(RAW): 02,43,35,2A,00,A2,00,02,02,10,01,11,04,
Tx(RAW): 02,52,35,02,00,89,01,11,03
```

Figure 2-12. Hex Log of SDAP Browse for SPP

- Opcode - 0x35
- Parameters:
  — UUID for the requested Service: 0x1101

Table 2-8. Example SDAP Browse

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	52 35 02 00 89
08 - 09	UUID	01 11
10	End Delimiter	03

The confirmation of the command includes all information about the registered services on the remote device for the requested UUID.

The full event includes the following parameters:

- Status byte (Error code) - 0x00
- Number of services - 0x02 (Number of services found)
- BrowseGroupID - 0x1002 (Public Browse Group)
- ServiceUUID - 0x1101 (The service found)
- RFComm Port Number - 0x04
- Number of bytes in the service name
- Name of the service

The following table shows the full confirm package for one SPP entry.

Table 2-9. Example SDAP Browse Confirm

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	43 35 0D 00 85
07	Status Byte	00
08	Number of Services	02
09 - 10	Browse Group ID	02 10
11 - 12	Service UUID	01 11
13	Remote RFComm Port Number	04
14	Number of bytes in name	05
15 - 18	Service Name	43 4F 4D 31 00 (COM1)
19	End Delimiter	03

The most important parameter out this event is parameter byte number 13, the RFComm Port Number. This will be needed to create a SPP Link to the other device.

2.3.4 SDAP Disconnect

After a successful Service Browse the connection has to be released again. As there can only be made one SDAP link at the time, the SDAP Disconnect command has no parameters.

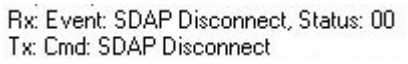


Figure 2-13. Log of SDAP Disconnect

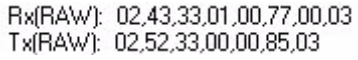


Figure 2-14. Hex Log of SDAP Disconnect

The confirmation of the command just returns the error/status code and is 0x00 is successful disconnection.

2.3.5 Create SPP Connection

Based on the information out of the Inquiry and the service request, a SPP connection can be established to the remote device. (assuming a SPP entry was found).

The following parameters are needed to establish a SPP link to a remote device.

- Command Opcode: 0x0A

- Local RFComm Port: Depending on local configuration, on default RFComm Port 1 is enabled
- Remote BD_Addr: out of Inquiry process
- Remote RFComm Port: out of SDAP Request

see also Figure 2-16 for the SPP Command (TX).

Table 2-10. Example Create SPP Link

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	52 0A 08 00 64
07	Local RFComm Port	01 (default at LMX9820A)
08 - 13	Remote BD_Addr	46 95 28 D9 0A 00
14	Remote RFComm Port	04 (out of SDAP Request)
15	End Delimiter	03

```
Rx: Event: Link Established, Status: 00, BdAddr: 469528D90A00, Local Port: 01, Remote Port Number: 04
Rx: Event: Port Status Changed, Local Port: 01, PortStatus: 0C, Break Length: 0000
Rx: Event: Establish Link, Status: 00, Local Port: 01
Tx: Cmd: Establish Link, Local Port: 01, BdAddr: 469528D90A00, Remote Port Number: 04
```

Figure 2-15. Log of Create SPP Connection

```
Rx(RAW): 02,69,0B,09,00,7D,00,46,95,28,D9,0A,00,01,04,03
Rx(RAW): 02,69,3E,04,00,AB,01,0C,00,00,03
Rx(RAW): 02,43,0A,02,00,4F,00,01,03
Tx(RAW): 02,52,0A,08,00,64,01,46,95,28,D9,0A,00,04,03
```

Figure 2-16. Hex Log of Create SPP Connection

The Log Window shows 3 events returned by the LMX9820A.

a) Establish Link Confirm
As any confirm the “Establish Link Confirm” (Section 5.2.3.1) has the same Opcode as the command sent. It includes the following parameters:

- Status/Error Code - 0x00
- Local RFComm Port - 0x01

The event means: “Got the request, trying to set up the link on port 1”.

Table 2-11. Example Establish Link Confirm

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	43 02 0A 00 4F
07	Status/Error Code	00 (for success)
08	Local RFComm Port	01
09	End delimiter	03

b) Port Status Changed Indicator
This event indicates that during the RFComm channel setup process the settings of the SPP link have changed.

The RFComm channel behaves like a virtual serial port with emulated handshaking and flow control.

Please see Section 5.2.10.3 for the detailed description of the event.

c) Link Established Indicator
To indicate an established link on top of the SPP, the LMX9820A uses the “Link Established Indicator” (Section 5.2.3.7). The event returns

- Status/Error code
- BD_Addr of the remote device
- Local RFComm port
- Remote RFComm port

Table 2-12. Example Link Established Indicator

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	69 0B 09 00 7D
07	Status/Error Code	00 (for success)
08 - 13	BD_Addr	46 95 28 0D 0A 00
14	Local RFComm Port	01
15	Remote RFComm Port	04
16	End delimiter	03

The package indicates which local RFComm port is now bound to a specific link.

2.3.6 Sending Data in Command Mode

After actively setting up a connection with the LMX9820A, the device is still listening to commands and returning status changes by events.

So to send data over the command interface the “Send Data” (Section 5.2.3.3) Command has to be used.

Besides the data which have to be sent, the local RFComm Port parameter has also to be sent to the LMX9820A. This enables the application to support multiple connections.

The Log windows in Figure 2-17 and Figure 2-18 show the transmission of the word “Test” over an established SPP link. The data is displayed in hex as ASCII values.

```
Rx: Event: Send Data, Status: 00, Local Port: 01
Tx: Cmd: Send Data, Local Port: 01, Payload Data: 54657374
```

Figure 2-17. Log of sending the Data “Test”

```
Rx(RAW): 02,43,0F,02,00,54,00,01,03
Tx(RAW): 02,52,0F,07,00,68,01,04,00,54,65,73,74,03
```

Figure 2-18. Hex log of sending the data “Test”

Table 2-13. Example Sending Data Package

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	52 0F 07 00 68
07	Local RFComm Port	01
08 - 09	Length of Data to send	04 00 (byte swapped)
10 - 13	Data to send	54 65 73 74 (“Test”)
14	End Delimiter	03

NOTE: The length of the data in the payload has influence on the package length within the package header and the length parameter within the payload itself.

Table 2-14 shows a second example with a longer data package, marking changed parameters in bold.

Table 2-14. Example Sending Data "Testdata"

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	52 0F **0B** 00 **6C**
07	Local Port	01
08 - 09	Length of Data	**08 00** (byte swapped)
10 - 13	Data to send	54 65 73 74 64 61 74 61 ("Testdata")
14	End Delimiter	03

2.3.7 Receiving Data in Command Mode

In command mode, incoming data from a remote device are indicated by the "Incoming Data" (Section 5.2.3.3) Event.

Besides the received data, the event also includes the local RFComm Port, on which the device has received the data.

```
Rx: Event: Incoming Data, Local Port: 01, Received Data: 74
Rx: Event: Incoming Data, Local Port: 01, Received Data: 73
Rx: Event: Incoming Data, Local Port: 01, Received Data: 65
Rx: Event: Incoming Data, Local Port: 01, Received Data: 54
```

Figure 2-19. Log of Incoming Data Event

```
Rx(RAW): 02,69,10,04,00,7D,01,01,00,74,03
Rx(RAW): 02,69,10,04,00,7D,01,01,00,73,03
Rx(RAW): 02,69,10,04,00,7D,01,01,00,65,03
Rx(RAW): 02,69,10,04,00,7D,01,01,00,54,03
```

Figure 2-20. Hex Log of Incoming Data Event

Figure 2-20 and Figure 2-19 show the log of 4 bytes received on local RFComm Port 01. The bytes together form the word "Test" again.

Table 2-15. Example Incoming Data Event

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	69 10 04 00 7D
07	Local RFComm Port	01
08 - 09	Length of Data received	01 00 (byte swapped)
10	Received Data	54 ("T")
11	End Delimiter	03

2.3.8 Releasing a SPP connection

To release an existing SPP connection the "Release Link" Command (Section 5.2.3.5) is used. The command is referring to the local RFComm port the connection has been established on.

```
Rx: Event: Link Released, Reason: 00, Local Port: 01
Rx: Event: Release Link, Status: 00, LocalPort: 01
Tx: Cmd: Release Link, Local Port: 01
```

Figure 2-21. Log for Release Link Command

```
Rx(RAW): 02,69,0E,02,00,79,00,01,03
Rx(RAW): 02,43,0D,02,00,52,00,01,03
Tx(RAW): 02,52,0D,01,00,60,01,03
```

Figure 2-22. Hex Log for Release Link Command

Table 2-16. Example Release Link Package

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	52 0D 01 00 60
07	Local RFComm Port	01
08	End Delimiter	03

The “Release Link” is confirmed by two events. Both include a status/error byte and the port number.

a) Release Link Confirm
The event confirms to the host that the command has been received and release is initiated.

b) Link Released Indicator
The event indicates that the LMX9820A released the Link on the RFComm Port returned.

Table 2-17. Example Link Released Indicator Package

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	69 0E 02 00 79
07	Reason Code	00
08	Local RFComm Port	01
09	End Delimiter	03

2.4 USING TRANSPARENT MODE

2.4.1 Activating Transparent Mode

In Transparent Mode, described in Section 1.3, the LMX9820A acts as cable replacement device. In this mode the LMX9820A does not interpret the packages sent to device. Instead, it is directly forwarding the data straight to the link previously set up.

If a bluetooth link to the LMX9820A has been established, this mode is automatically activated if the Operation Mode “Automatic” is set in NVS (default).

If the link was set up manually via the LMX9820A command interface (see Section 2.3), the LMX9820A still listens to commands and data have to be sent via the “Send Data” command.

As the LMX9820A routes the data directly to the bluetooth link, transparent mode can only be activated if only one active link exists. The following command has to be used to switch into transparent mode:

- “Transparent Mode” (page 116)

The “Transparent mode” command is referring to the local RFComm port the link was created on.

The command is confirmed by the appropriate event. Afterwards the LMX9820A is routing all incoming data directly to the remote device.

```
Rx: Event: Transparent Mode, Status: 00, Local Port: 01
Tx: Cmd: Transparent Mode, Local Port: 01
```

Figure 2-23. Log of Set Transparent Mode

```
Rx(RAW): 02,43,11,02,00,56,00,01,03
Tx(RAW): 02,52,11,01,00,64,01,03
```

Figure 2-24. Hex Log of Set Transparent Mode

Table 2-18. Example Set Transparent Mode

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	52 11 01 00 64
07	Local RFComm Port	01
08	End Delimiter	03

Table 2-19. Example Set Transparent Mode Confirm

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	43 11 02 00 56
07	Status/Error Code	00
08	Local RFComm Port	01
09	End Delimiter	03

Any data received will be routed directly to the UART.

```
Tx(RAW): 54,65,73,74
Rx(RAW): 54,65,73,74
```

Figure 2-25. Sending/Receiving “Test” in transparent mode

The LMX9820A is leaving the transparent mode, when a break signal is sent on the UART (see Section 2.4.2).

The break signal can also be used if the LMX9820A has been connected from a remote device and switched automatically to transparent (Automatic slave mode).

The recognized BREAK is confirmed by the “Transparent Mode” Event.

```
Rx: Event: Transparent Mode, Local Port: 01, Mode: 00
```

Figure 2-26. Log of Transparent Mode Event

```
Rx(RAW): 02,69,11,02,00,7C,01,00,03
```

Figure 2-27. Hex Log of Transparent Mode Indicator

Table 2-20. Example Transparent Mode Indicator

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	69 11 02 00 7C
07	Local RFComm Port	01
08	Operation Mode	00 (Command)
09	End Delimiter	03

2.4.2 Leaving transparent mode with UART BREAK

The UART Break is defined as the contiguous transmission of “0” (space) for a certain length of time. The CCITT “blue book” specification states that the time duration for this is larger than 2M+3 bit time (where M is the character length). After the break sequence, another 2M+3 bit time consisting of the contiguous transmission of “1” (mark) is required to start the next character.

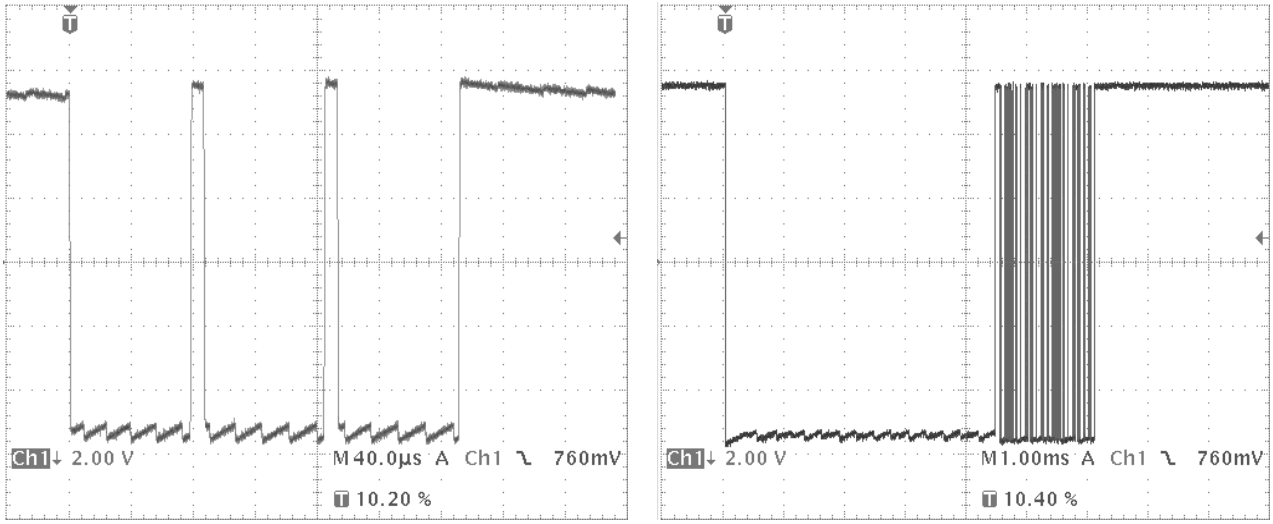


Figure 2-28. Difference between a Standard 0 transmission and BREAK signal

Figure 2-28 shows the difference between the signal of a normal 0 and the BREAK signal. The left picture shows the signalling of 3 Zeros at 115.2kbit/s. Each character is started and ended with Start and Stopbit.

The picture on the right shows a BREAK signalled by the LMX9820A after a released link. The signal is held low for over 4 ms. Theoretical minimum value for a BREAK at this speed would be about 165µS.

2.5 EXAMPLES

The following log files show the typical hex values sent to respectively returned from the LMX9820A. The level of events returned by the 9820 depends on the event filter level set within the NVS. The tables with the log entries also show the filter level, in which the messages are reported. Default filter setting is 01, so only those events will be reported.

2.5.1 Automatic Slave

Table 2-21 shows the event returned from the LMX9820A if it was connected from outside. The LMX9820A just returns one event indicating the BD_Addr of the remote device and the local RFComm port it connected to.

Table 2-21. Log File of Incoming Link as automatic slave

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
00	RX	Indicator	02,69,50,07,00,C0,12,34,56,78,9A, BC,00,03	Rx: Event: ACL Established, BdAddr: 123456789ABC, Status: 00
00 / 01	RX	Indicator	02,69,0C,07,00,7C,12,34,56,78,9A ,BC,01,03	Rx: Event: Incoming Link Established, BdAddr: 123456789ABC, Local Port: 01

Table 2-22 shows the events of typical procedure if a link was released from the other device.

Table 2-22. Log File of a Released Link as Automatic Slave

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
00 / 01	RX	UART Break		Rx(RAW): 00
00 / 01	RX	Indicator	02,69,11,02,00,7C,01,00,03	Rx: Event: Transparent Mode, Local Port: 01, Mode: 00
00 / 01	RX	Indicator	02,69,0E,02,00,79,01,01,03	Rx: Event: Link Released, Reason: 01, Local Port: 01
01	RX	Indicator	02,69,51,07,00,C1,12,34,56,78, 9A,BC,13,03	Rx: Event: ACL Terminated, BdAddr: 123456789ABC, Reason: 13

2.5.2 Setting up a link

As documented in Section 2.3, setting up one or more links to another device in general requires the knowledge of the BD_Addr and the RFComm Port to connect to. Table 2-23 shows all commands necessary from scratch to establish a link to another device.

Table 2-23. Log File of a complete link setup

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,00,03,00,55,0A,00,00,03	Tx: Cmd: Inquiry, Length: 0A, NumResponces: 00, Mode: 00
00 / 01	RX	Indicator	02,69,01,09,00,73,12,34,56,78, 9A,BC,00,00,00,03	Rx: Event: Device Found, BdAddr: 123456789ABC, DeviceClass: 000000
00 / 01	RX	Confirm	02,43,00,01,00,44,00,03	Rx: Event: Inquiry, Status: 00
	TX	Request	02,52,32,06,00,8A,12,34,56,78, 9A,BC,03	Tx: Cmd: SDAP Connect, BdAddr: 123456789ABC
01	RX	Indicator	02,69,50,07,00,C0,12,34,56,78, 9A,BC,00,03	Rx: Event: ACL Established, BdAddr: 123456789ABC, Status: 00
00 / 01	RX	Confirm	02,43,32,01,00,76,00,03	Rx: Event: SDAP Connect, Status: 00
	TX	Request	02,52,35,02,00,89,01,11,03	Tx: Cmd: Service Browse, Browse Group ID: 0111

Table 2-23. Log File of a complete link setup

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
00 / 01	RX	Confirm	02,43,35,0D,00,85,00,01,02,10, 01,11,01,05,43,4F,4D,31,00,03	Rx: Event: Service Browse, Status: 00, Browse Group ID: 0210, Service ID: 0111, PortNo: 01, Service Name: COM1.
	TX	Request	02,52,33,00,00,85,03	Tx: Cmd: SDAP Disconnect
01	RX	Indicator	02,69,51,07,00,C1,12,34,56,78, 9A,BC,16,03	Rx: Event: ACL Terminated, BdAddr: 123456789ABC, Reason: 16
00 / 01	RX	Confirm	02,43,33,01,00,77,00,03	Rx: Event: SDAP Disconnect, Status: 00
	TX	Request	02,52,0A,08,00,64,01,12,34,56, 78,9A,BC,01,03	Tx: Cmd: Establish Link, Local Port: 01, BdAddr: 123456789ABC, Remote Port Number: 01
00 / 01	RX	Confirm	02,43,0A,02,00,4F,00,01,03	Rx: Event: Establish Link, Status: 00, Local Port: 01
01	RX	Indicator	02,69,50,07,00,C0,12,34,56,78, 9A,BC,00,03	Rx: Event: ACL Established, BdAddr: 123456789ABC, Status: 00
00 / 01	RX	Indicator	02,69,3E,04,00,AB,01,0C,00,00 ,03	Rx: Event: Port Status Changed, Local Port: 01, PortStatus: 0C, Break Length: 0000
00 / 01	RX	Indicator	02,69,0B,09,00,7D,00,12,34,56, 78,9A,BC,01,01,03	Rx: Event: Link Established, Status: 00, BdAddr: 123456789ABC, Local Port: 01, Remote Port Number: 01
	TX	Request	02,52,11,01,00,64,01,03	Tx: Cmd: Transparent Mode, Local Port: 01
00 / 01	RX	Confirm	02,43,11,02,00,56,00,01,03	Rx: Event: Transparent Mode, Status: 00, Local Port: 01

3.0 Advanced Usage

The LMX9820A offers a wide variety of functions for different usage models. This section points out the most important features and scenarios covered by the LMX9820A.

3.1 LOCAL CONFIGURATION

3.1.1 Hardware Configuration

The LMX9820A has several commands to configure the local hardware. Those include settings for the UART speed and configuration, a soft reset and also include settings to set the device into special test modes. Please check also Section 5.2.18 “Local Hardware Configuration” on page 169.

3.1.1.1 Change UART settings

The UART speed in general is determined by the choice of the ISEL pins of the LMX9820A. The pins and UART settings in NVS are only checked during the software boot-up process so also after a Reset command.

Table 3-1. UART settings

ISEL1	ISEL2	Interface Speed (baud)	UART Settings
1	1	921.6k	Check NVS
0	1	115.2k	Check NVS
1	0	9.6k	No Parity, One Stop bit
0	0	Check NVS	Check NVS

In case another speed than the default needs to be used, the speed configuration has to be done in NVS. By default a speed of 9.6kbit/s is stored.

Example: Configuring LMX9820A for UART speed of 57.6kbit/s:

- Set ISEL1 and ISEL2 to 0
- Power up / Reset the device, the UART is configured on 9.6kbit/s on default. LMX9820A answers with “SimplyBlue LMX9820A Ready”. If set to an unknown speed, please set ISEL 1 and 2 to a dedicated speed and try again.
- Send the following command:
  — “Change NVS UART Speed” (page 175) with parameter 0x06.

Table 3-2. Change NVS UART Speed to 57.6kbit/s

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	52 23 01 00 76
07	UART speed	06
08	End Delimiter	03

- Await the Confirmation Event from the LMX9820A.
- Send the following command do a hardware reset
  — “Reset” (page 174)

Afterwards the LMX9820A will communicate with a UART speed of 57.6kbit/s

3.1.1.2 Bluetooth Testmode

The LMX9820A supports the standard Bluetooth “Device Under Test” Mode and a “Local Loopback” Mode. The “Device under Test” Mode is the standard testmode is used by any bluetooth tester. If activated the tester will be able to take control over the LMX9820A and put it into the specific testmodes needed for bluetooth qualification.

The “local loopback” mode is a simple UART loopback mode to test the UART communication interface.

The testmodes are enabled by the command “Test Mode”.

“Bluetooth Device Under Test Mode” can be left by software Reset, “Local Loopback test” requires a hardware reset.

Please see Section 5.2.18.11 “Bluetooth Test Mode” on page 176 for details for the command.

3.1.1.3 RF Testmodes

Bluetooth qualification, FCC and CEPT qualifications also require continuous transmit modes. For this the transmitter has to be set to a specific transmitting only status, with which test houses can make for instance spurious emission testings.

The detailed command is described in Section 5.2.18.12 “Initiate RF Test Mode” on page 177.

3.1.1.4 Restore Factory Settings

The LMX9820A is delivered with standard settings in NVS which can be seen in Table 1-1 "LMX9820A System Parameters" on page 10. Those parameters are changing during usage or testing the part.

The restore to factory settings gives the ability to restore all default values as listed in the table and deletes all additional entries. The following changes will be made.

- Reset of the Service Database entry to the default
- Deleting Link Keys (all paired devices)
- UART Speed Settings
- Event Filter
- Ports to open
- Master Role off
- Operation Mode
- Pin
- Local Name
- Default Link Policy
- Default Link Timeout

Command details can be found in Section 5.2.18.8 “Restore Factory Settings” on page 174.

3.1.1.5 Read RSSI

After link establishment the radio is measuring the “Receive Signal Strength Indicator”, a parameter indicating the signal strength of the incoming packages. Typically the Link Manager forces the remote devices to decrease or increase its output power to improve the receiving performance.

This command offers the ability to read out the RSSI and indicate the current status of the signal strength. (See Section 5.2.18.13 on page 178).

The value indicates:

- 0xFF: Signal too low
- 0x00: Signal OK
- 0x01: Signal too strong

3.1.1.6 Start Firmware Upgrade

The LMX9820A includes an “In-System-Programming” (ISP) code which allows to update the on-chip firmware over the UART interface. The code is activated by either the “Environment Pins” ENV or by sending this command to the chip. The command will initiate a software Jump to the ISP code, which can only be left by a hardware reset. In ISP no bluetooth functionality will be provided.

Commands available:

- “Firmware Upgrade” (page 179)

NOTE: The Command “Firmware Upgrade” sets the empty flag on flash memory. This means the LMX9820A needs to be programmed after that command, ended by the ISP “GO” command. Otherwise the flag will stay marked as empty and the module will not come up to Bluetooth software after reset.

3.1.1.7 Event Filter

The LMX9820A indicates status changes and confirms commands with specific events. To address specific application requirements the level of reporting can be increased or decreased.

The following separation is possible:

- Report standard events including ACL indicators, which indicate the physical connection status.
- Report standard events only
- Report no events, only UART break indicates lost link, LMX9820A still detects a UART Break to leave “Transparent Mode”
- No reporting the UART and “Transparent Mode” not left on UART Break

The event filter can be configured by the following command:

- “Set Event Filter” (page 173)
- “Get Event Filter” (page 173)

3.1.1.7.1 Report standard events including ACL

In case the NVS setting is set to “report all events”, the LMX9820A reports any status change and ACL link establishment to the host.

The main difference to the other filter settings are the reported ACL indicators.

This reporting can be necessary if the system is actively trying to connect to another device. The ACL indicator includes an error code, which gives information about the reason of a failed connection. (E.g. failed authentication).

NOTE: Please be aware that any kind of established ACL link will be reported to the host. If the device is only waiting for connection, any attempt from another device to connect to the LMX9820A will be reported to the host. For a “slave only” usage, one of the other filtering settings would probably be more useful.

Available ACL Events:

- “ACL Established” (page 163)
- “ACL Terminated” (page 164)

3.1.1.7.2 Report standard events (default)

In this reporting scheme the LMX9820A reports all events and indicators except the ACL indications. This mode is set as default and is basically allowing backwards compatibility to earlier firmware versions.

3.1.1.7.3 Report no events

In case the UART does not send back any event to the host the Event filter has to be set to “Report no events”. The only “event” is the UART Break, which still indicates the loss of the bluetooth link.

This filter setting is useful if the LMX9820A is used as cable replacement in front of a microcontroller, where no status event can be interpreted. But the device will still send a UART Break when “Transparent Mode” is left. In addition the LMX9820A will still recognize a UART Break and leave “Transparent Mode” when detected.

In this setting the pin LSTAT1 can be used as hardware indicator about the link status of the LMX9820A.

3.1.1.7.4 Report no events, UART Break suppressed and ignored

In addition to the level described in Section 3.1.1.7.3, also the UART Break is suppressed and ignored.

This filter setting is useful if the LMX9820A is used as cable replacement in front of a microcontroller, where no status event and even no UART Break can be interpreted. In addition the LMX9820A will not recognize a UART Break, therefore will not leave “Transparent Mode”.

In this setting the pin LSTAT1 can be used as hardware indicator about the link status of the LMX9820A.

3.1.1.8 Default Audio Settings

The Default Audio Settings configure the Audio codec driver as well as the format to be used over bluetooth. Bluetooth by default uses the CVSD coding over the bluetooth link. The LMX9820A by default uses CVSD and no audio codec is selected.

The following audio codecs are supported:

- OKI MSM7717
- Motorola MC145483
- Winbond W681360 on **firmware 6.23 and later**
- PCM slave mode on **firmware 6.23 and later**

The following air formats are supported:

- CVSD
- µ-Law
- A-Law

Available Commands:

- “Set Default Audio Settings” (page 170)
- “Get Default Audio Settings” (page 170)
- “Set PCM Slave Configuration” (page 171) on **firmware 6.23 and later**

3.1.2 LMX9820A Bluetooth Configuration

The Local Bluetooth Configuration includes commands for changing parameters which have influence on if or how the device will answer to requests and how it behaves in different situations.

3.1.2.1 Local Bluetooth Device Address

The BD_Addr is a unique identifier for each bluetooth product. The LMX9820A is delivered by National with a preprogrammed BD_Addr which is stored in a write protected area in the flash and copied to NVS if this is empty.

If necessary this value can be changed to any specific value. A custom value will be reset by the “Restore Factory settings” command.

The commands available are:

- “Read Local Bluetooth Address” (page 157)
- “Change Local Bluetooth Address” (page 157)

Note:
Please be aware that by overwriting this address the uniqueness of the device address cannot be guaranteed anymore.

3.1.2.2 Local Name

The Local Name is transmitted on “Remote Name Requests” from other devices. It just represents a friendly name of the device. Default value is “Serial Port Device”.

- “Read Local Name” (page 156)
- “Write Local Name” (page 156)

3.1.2.3 Class of Device

The Class of Device is based on a numbering scheme of the Bluetooth SIG and is returned on Inquriy requests from other devices. The Class of Device indicates the basic functionality of a device like Mobile Phone, Printer, Headset. This number can be used by the main application to already filter the devices in range for certain functionality.

A complete list of numbers and can be found in the “Bluetooth Assigned Numbers” Document provided by the Bluetooth SIG at https://www.bluetooth.org/foundry/assignnumb/document/baseband.

Some Examples:

- Desktop Computer: 00 01 04
- Handheld PDA: 00 01 14
- Cellular Phone: 70 02 04

Note: the values should be seen as examples.

Default value is 00 00 00. (no specific device)

Commands available:

- "Store Class of Device" (page 158)

3.1.2.4 Operation Mode

The Operation Mode as described in Section 1.6 "Operation Modes" on page 14 has influence on the behaviour of the LMX9820A in different situations.

Commands available:

- "Read Operation Mode" (page 160)
- "Write Operation Mode" (page 160)

3.1.2.5 Fixed Pin

The LMX9820A stores a fixed pin which will be used during pairing processes. The pin is stored in NVS and can be changed by the following commands:

- "Get Fixed Pin" (page 167)
- "Set Fixed Pin" (page 168)

The pin stores the hex value for the ASCII character used as pin. Eg. the pin "1 2 3 4" will be stored as "31 32 33 34".

Default value is in ASCII "0 0 0 0", in hex "30 30 30 30".

NOTE: For the **firmware 6.23 and later**, a dynamic Pin code has been included in the features. it is possible to store a 0 length Pin code, which forces a Pin request event to the host if authentication is requested.

3.2 CONFIGURING THE DEFAULT LINK TIMEOUT

The bluetooth specification defines a specific timeout, which causes the baseband to drop the link if no packages have been received on a link for a specific period of time. In a standard active bluetooth link which is not used by the application to send data at the moment, the master sends out "poll' packages in agreed intervals, to keep the slaves synchronized. The default poll period is 40slots (or 25ms). The slaves acknowledge each package with a "Null" package.

In case those poll packages are not received by the slave or the master does not get the acknowledgement from the slave, both devices will still try to send or receive packages from each other until the "supervision timeout" is reached. After that the link is indicated as lost. This "supervision timeout" is set by default to 20seconds.

A slave will not be able to accept an incoming connection until the link is completely dropped. As it might be useful in certain application to give up a link earlier than 20seconds, the LMX9820A gives the ability to configure the supervision timeout for each link. It can either be configured as default value in NVS or only for the existing SPP link.

NOTE: The Link Supervision Timeout should not be set too low. The value shall also guarantee the quality of service of an existing link. Due to the fact that packages can get lost due to noisy environment or a master might need to share his bandwidth between multiple slaves, the timeout also ensures that the devices don't mark the link as lost if a few packages are not received or acknowledged. E.g. a link timeout of only 1 second could even be too short to guarantee a stable link.

3.2.1 Setting the Default Link Timeout

The default Link Supervision timeout for all devices is 20seconds. As the baseband calculates in slots, the value needs to be stored in number of slots. Therefore the default value used is 0x7D00.

The default link supervision timeout is used for each incoming and outgoing link. The value stored in NVS can be changed anytime and will be active for the next link establishment without Reset.

- "Set Default Link Timeout" (page 117)

- “Get Default Link Timeout” (page 118)

3.2.2 Changing the Link Timeout of an existing link

After a successful link establishment both parties of a link agreed on a specific link timeout. This timeout can be read back or changed anytime by the following commands. In case the local LMX9820A has the role slave, the parameter will only be set locally. In case the LMX9820A is master for the link, the timeout will also be communicated to the connected device, which then will adjust its timeout to that value as well. See also Section 3.2.3 for the difference between master and slave.

- “Set Link Timeout for an existing link” (page 118)
- “Get Link Timeout of an existing link” (page 119)

3.2.3 Difference between Master and slave role for the link timeout

The link timeout is a local parameter stored for each link specific link. Therefore a master, which is connected to several devices can have different link timeouts set for each of those links.

The difference between a master and slave is, that a master reports a change of the link timeout to the slave, which then adjust its own timeout to the reported value. In case the “Set link timeout” command will be sent over UART to a LMX9820A in slave role, the setting will only be active locally and no message will be sent to the master. This means the two devices will have different link timeout settings.

The following figures shall demonstrate the influence of default link timeout and set link timeout, in case they are set on master or slave.

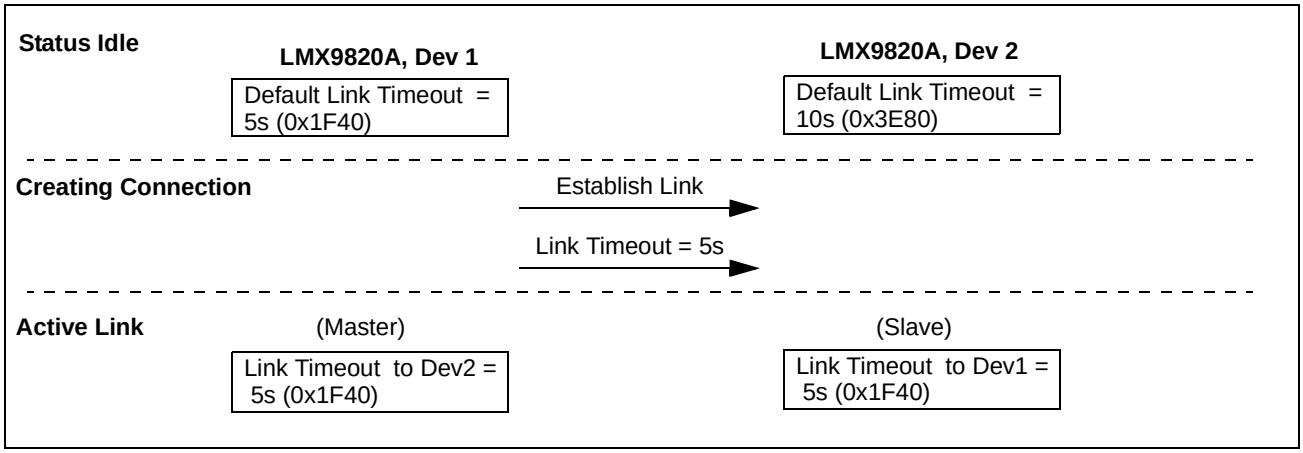


Figure 3-1. Master Default Link Timeout

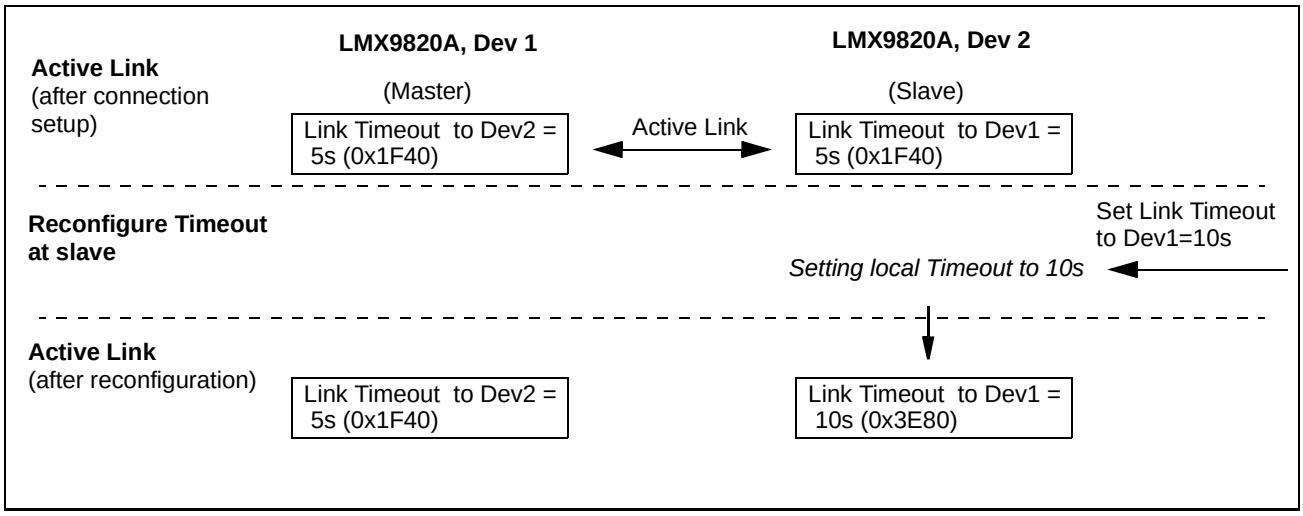


Figure 3-2. Set Link Timeout at Slave

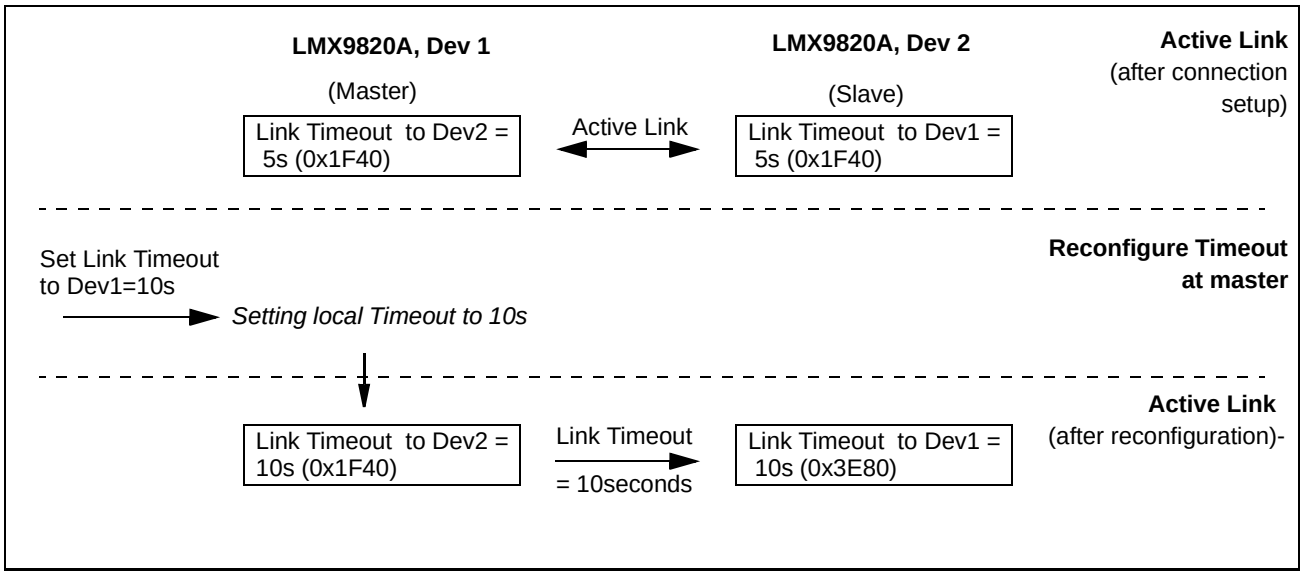


Figure 3-3. Set Link Timeout at Master

3.3 CONFIGURING THE LINK LATENCY

The link latency is part of the Quality of Service bluetooth is able to offer.

The link latency results in the so called “poll period”.

A standard bluetooth link controlled by the master device, polling each of the slaves connected to it in a predefined period. The polling is necessary to keep the slaves synchronized but also to enable them to send data to the master. Since the master has controls the link it is able to send data immediately to the slave to be addressed. The reaction time of a slave device is limited to the poll period agreed with the master or at least on the period the master sends packages to the slave. The default poll period for any Bluetooth link is 40 slots (25ms).

In case the slave needs guaranteed data transmission lower than the default 40 slots, the link latency parameter for this link needs to be reduced.

The LMX9820A offers the ability of configuring the default poll period used for each link. The parameter is stored within the NVS and will be requested for any incoming or outgoing link. Since a master might need to manage several slaves, the parameter has to be seen as request, it can not be 100% guaranteed that exactly this value will be used.

NOTE: Using very small poll periods will heavily increase the power consumption of an active link, since the devices exchange packages in lower intervals.

The parameter can be set and reviewed by the following commands:

- “Set Default Link Latency” (page 161)
- “Get Default Link Latency” (page 162)

3.4 SETTING UP MULTIPLE CONNECTIONS

As already described in Section 2.3, the LMX9820A command interface offers the ability to search for other devices, browse the services and to establish a link to another device.

The LMX9820A bluetooth operation is based on the Serial Port Profile. This profile emulates a serial port over a bluetooth link. As bluetooth is able to handle more than one links, the LMX9820A will also offer multiple communication ports, also called RFComm ports to the host. For each port a separate RFComm instance will be created.

As the LMX9820A has only limited resources on RAM, the maximum number of links is limited to three.

Figure 3-4 on page 41 shows the standard link establishment as already described in Section 2.3 on page 21. This establishment uses the standard configurations for the LMX9820A with one RFcomm port on the device.

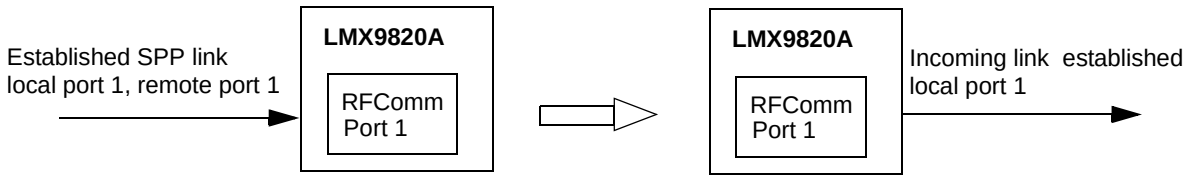


Figure 3-4. Standard point-to-point SPP connection

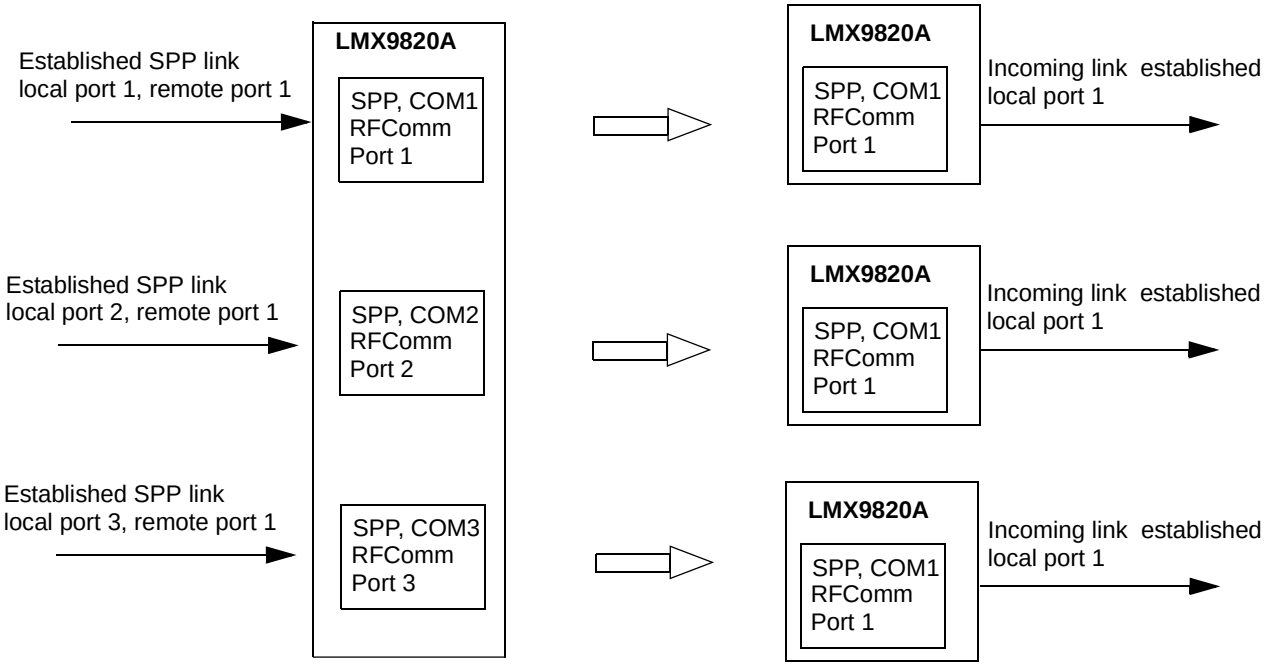


Figure 3-5. Multiple SPP links from LMX9820A

Figure 3-5 on page 41 shows the principle of establishing multiple links with the LMX9820A. The LMX9820A on the left has to initialize 3 RFComm ports, which then can be addressed and bound to a remote device. Any data sent to that specific port are then sent to the device connected to it.

The figure also shows the term “SPP, COM1” which is an indicator for the service database entry stored for this specific port. By default the LMX9820A has configure a Serial Port Profile (SPP) to RFComm port 1. The name, which will be reported to a browsing device is set to “COM1”.

The following sections will guide you through the configuration settings and link setup commands to establish link to three slaves.

3.4.1 Configuration

The LMX9820A by default is configured for point-to-point operation only. Therefore some configuration settings have to be changed before it is possible to establish multiple links.

3.4.1.1 RFComm Ports to open

The Serial Port profile is based on the protocol layer RFCOMM, which offers a serial port emulation to the application interface. Each virtual serial port can be seen as virtual cable between two devices. All data sent to this port will be routed over bluetooth to the remote device.

The LMX9820A can handle up to three RFComm ports simultaneously. Each RFComm port opened creates a buffer instance within the RAM to handle the upcoming data traffic. Because of that by default only RFComm port 1 is activated.

The RFComm ports opened and initialized for operation can be configured by the following command:

- Section 5.2.12 "RFcomm Channels to open" on page 145

The ports in this command are expressed by a 32-bit mask indicating which RFCOMM ports the LMX9820A has to open. Bit 30 and 31 must be set to 0. Bit 0 is RFCOMM port 1 and bit 29 is port 30. The changes take effect as soon as the command has been confirmed.

Examples:

Open RFComm port 1: Set Ports to open to 0x00000001
Open RFComm port 1 and 3: Set Ports to open to 0x00000005
Open RFComm port 1, 2 and 3: Set Ports to open to 0x00000007

So to set up 3 Links, Ports to open has to be configured to 0x00000007.

Table 3-3. Open 3 RFComm Ports

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	52 22 04 00 78
07 - 10	RFComm Ports to open	07 00 00 00
11	End Delimiter	03

3.4.1.2 Configuring Service Database for outgoing connections

The LMX9820A contains a service database which configures the settings for authentication and encryption for that specific port. The configuration includes rules for incoming but also outgoing links.

NOTE: The service database security settings are only used when the device is configured to security level 2 (default). Please see also Section 5.2.16 on page 164.

In principle, the service database is only necessary to be able to offer services to remote devices. Eg. if a remote device creates a SDAP link to the LMX9820A and browses the services, it will only see the entries stored within this database. If for example a second profile needs to be offered, eg. a second SPP called COM2, it needs to be entered into the database. See also Section 5.2.13 on page 146 for details modifying the service database.

By default the following entry is stored within the LMX9820A for RFComm Port 1:

Table 3-4. Default SDP entry for RFComm 1

Parameter	**Description**	**Value**
Entry Index	Index at which the entry can be addressed for enabling or disabling it	0x00
RFComm Port	RFComm Port the settings refer to	0x01
Profile to be offered	The profile to be offered to the remote device. Profiles different from SPP need to be implemented on the host.	SPP
Entry Name	Name which will be shown to the remote device	COM1
Authentication	Defines if authentication is required if port is used for an incoming or outgoing connection. 0x00 No authentication requirements 0x02 Authentication is only required for this profile for incoming connections. 0x20 Authentication is only required for this profile for outgoing connections. 0x22 Authentication is required for this profile for connections in both directions See also Section 5.2.13.3 on page 148.	0x02

Table 3-4. Default SDP entry for RFComm 1

Parameter	Description	Value
Encryption	Defines if encryption is required if port is used for an incoming or outgoing connection. 0x00 No encryption requirements 0x04 Encryption is only required for this profile for incoming connections. 0x40 Encryption is only required for this profile for outgoing connections. 0x44 Encryption is required for this profile for connections in both directions See also Section 5.2.13.3 on page 148	0x04

Table 3-4 shows the default service database entry offered to a remote device. The configuration shows that an incoming link which will address RFComm port 1 will require authentication and encryption. In case the link establishment has not happened before, the LMX9820A will automatically initiate a pin code request to the remote device.

In general, for outgoing connections, there's no service database required for that specific RFComm port. For example, if a link is established from local RFComm port 2 to a remote port and no SDB entry has been made for that port, the default will be that no authentication and no encryption will be required.

To demonstrate the configuration for a secure outgoing connection, RFComm Port 2 shall be entered into the service database as SPP profile, with authentication enabled for outgoing connections.:

Table 3-5. SDB entry with security on outgoing connection

Parameter	Description	Value
Entry Index	Index at which the entry can be addressed for enabling or disabling it	0x01
RFComm Port	RFComm Port the settings refer to	0x02
Profile to be offered	The profile to be offered to the remote device. Profiles different from SPP need to be implemented on the host.	SPP
Entry Name	Name which will be shown to the remote device	COM2
Authentication	0x20 Authentication is only required for this profile for outgoing connections.	0x20
Encryption	0x40 Encryption is only required for this profile for outgoing connections.	0x40

The following can be used to store the additional entry into the service database.

- Section 5.2.13.3 "SDP Store SPP Record" on page 148

The changes take effect immediately after getting the confirmation event.

Table 3-6. Example SDP Store SPP Record

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	52 2B 09 00 86
07	Local RFComm Port	02
08	Authentication setting	20
09	Encryption setting	40
10	Service Name length	05
11 - 15	Service Name	43 4F 4D 32 00 'COM2'
16	End Delimiter	03

3.4.2 Link Establishment

Establishing links to multiple slaves in principle is the same as a standard connection setup already described in Section 2.3 "Setting up a link using the Command Interface" on page 21. The only difference which has to be considered is that the LMX9820A has enough RFComm ports available (opened) and each link is assigned to one specific link.

The link establishment uses the command “Establish SPP Link” to create all links. Each link established will be confirmed by the appropriate confirmation event. The following three tables show the commands which have to be sent to establish the three links. The commands need to be adjusted for the local RFComm Port, the remote BD_ADDR and the remote RFComm Port Number, derived out of the related SDAP request.

Table 3-7. Establish Link on Local RFComm Port 1

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	52 0A 08 00 64
07	**Local RFComm Port**	**01**
08 - 13	Remote BD_Addr	12 34 56 78 9A BC
14	Remote RFComm Port	01 (out of SDAP Request)
15	End Delimiter	03

Table 3-8. Establish Link on Local RFComm Port 2

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	52 0A 08 00 64
07	**Local RFComm Port**	**02**
08 - 13	Remote BD_Addr	34 56 78 9A BC 12
14	Remote RFComm Port	01 (out of SDAP Request)
15	End Delimiter	03

Table 3-9. Establish Link on Local RFComm Port 3

Byte	Parameter	Value
01	Start Delimiter	02
02 - 06	Package Header	52 0A 08 00 64
07	**Local RFComm Port**	**03**
08 - 13	Remote BD_Addr	56 78 9A BC 12 34
14	Remote RFComm Port	01 (out of SDAP Request)
15	End Delimiter	03

3.4.3 Summary

The following tables show a complete example of configuring a device to establish three links to 3 different slaves.

The example is based on the assumption that the other devices are not known and have never been explored before, so all information have to be collected.

3.4.3.1 Device Configuration, preparing for multipoint operation

In order to create more than one link, the device needs to initialize multiple connections. In case, authentication and encryption are necessary for one of the outgoing links an additional service database entry needs to be made. The example shows a new service entry for Local RFComm Port 2 with Authentication and Encryption enabled.

Table 3-10. Initialize 3 RFComm Ports

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,22,04,00,78,07,00,00,00,03	Tx: Cmd: Set Ports To Open, Ports: 07000000
00 / 01	RX	Confirm	02,43,22,01,00,66,00,03	Rx: Event: Set Ports To Open, Status: 00

Table 3-11. Configure additional SDB Entry to enable auth/encr. for outgoing connection

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,2B,09,00,86,02,20,40,05,43, 4F,4D,32,00,03	Tx: Cmd: Store SPP Record, Local Port: 02, Authentication: 20, Encryption: 40, Service Name: COM2.
00 / 01	RX	Confirm	02,43,2B,02,00,70,00,02,03	Rx: Event: Store SPP Record, Status: 00, Identifier: 01

3.4.3.2 Discover Devices

To get the devices BD_Addresses an Inquiry has to be started.

Table 3-12. Device Discovery

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,00,03,00,55,0A,00,00,03	Tx: Cmd: Inquiry, Length: 0A, NumResponces: 00, Mode: 00
00 / 01	RX	Indicator	02,69,01,09,00,73,12,34,56,78,9A, BC,00,00,00,03	Rx: Event: Device Found, BdAddr: 123456789ABC, DeviceClass: 000000
00 / 01	RX	Indicator	02,69,01,09,00,73,34,56,78,9A,BC, 12,00,00,00,03	Rx: Event: Device Found, BdAddr: 3456789ABC12, DeviceClass: 000000
00 / 01	RX	Indicator	02,69,01,09,00,73,56,78,9A,BC,12, 34,00,00,00,03	Rx: Event: Device Found, BdAddr: 56789ABC1234, DeviceClass: 000000
00 / 01	RX	Confirm	02,43,00,01,00,44,00,03	Rx: Event: Inquiry, Status: 00

3.4.3.3 Get Remote RFComm Ports

To get the remote Comports, for each of the links a SDAP Browse has to be done.

Table 3-13. Get Remote RFComm Port of first device

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,32,06,00,8A,**12,34,56,78,9A, BC**,03	Tx: Cmd: SDAP Connect, BdAddr: 123456789ABC
01	RX	Indicator	02,69,50,07,00,C0,**12,34,56,78,9A, BC**,00,03	Rx: Event: ACL Established, BdAddr: 123456789ABC, Status: 00
00 / 01	RX	Confirm	02,43,32,01,00,76,00,03	Rx: Event: SDAP Connect, Status: 00
	TX	Request	02,52,35,02,00,89,**01,11**,03	Tx: Cmd: Service Browse, Browse Group ID: 0111
00 / 01	RX	Confirm	02,43,35,1E,00,96,00,01,02,10,01, 11,**01**,16,42,6C,75,65,74,6F,6F,74, 68,20,53,65,72,69,61,6C,20,50,6F, 72,74,00,03	Rx: Event: Service Browse, Status: 00, Browse Group ID: 0210, Service ID: 0111, PortNo: 01, Service Name: Bluetooth Serial Port.

Table 3-13. Get Remote RFComm Port of first device

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,33,00,00,85,03	Tx: Cmd: SDAP Disconnect
00 / 01	RX	Confirm	02,43,33,01,00,77,00,03	Rx: Event: SDAP Disconnect, Status: 00
00	RX	Indicator	02,69,51,07,00,C1,**12,34,56,78,9A,BC**,16,03	Rx: Event: ACL Terminated, BdAddr: 123456789ABC, Reason: 16

Table 3-14. Get Remote RFComm Port of the second device

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,32,06,00,8A,**34,56,78,9A,BC,12**,03	Tx: Cmd: SDAP Connect, BdAddr: 3456789ABC12
00	RX	Indicator	02,69,50,07,00,C0,**34,56,78,9A,BC,12**,00,03	Rx: Event: ACL Established, BdAddr: 3456789ABC12, Status: 00
00 / 01	RX	Confirm	02,43,32,01,00,76,00,03	Rx: Event: SDAP Connect, Status: 00
00 / 01	RX	Request	02,52,35,02,00,89,01,11,03	Rx: Event: Service Browse, Browse Group ID: 0111
00 / 01	RX	Confirm	02,43,35,1E,00,96,00,01,02,10,01,11,**01**,16,42,6C,75,65,74,6F,6F,74,68,20,53,65,72,69,61,6C,20,50,6F,72,74,00,03	Rx: Event: Service Browse, Status: 00, Browse Group ID: 0210, Service ID: 0111, PortNo: 01, Service Name: Bluetooth Serial Port.
	TX	Request	02,52,33,00,00,85,03	Tx: Cmd: SDAP Disconnect
00 / 01	RX	Confirm	02,43,33,01,00,77,00,03	Rx: Event: SDAP Disconnect, Status: 00
00	RX	Indicator	02,69,51,07,00,C1,**34,56,78,9A,BC,12**,16,03	Rx: Event: ACL Terminated, BdAddr: 56789ABC1234, Reason: 16

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,32,06,00,8A,**56,78,9A,BC,12,34**,03	Tx: Cmd: SDAP Connect, BdAddr: 56789ABC1234
00	RX	Indicator	02,69,50,07,00,C0,**56,78,9A,BC,12,34**,00,03	Rx: Event: ACL Established, BdAddr: 56789ABC1234, Status: 00
00 / 01	RX	Confirm	02,43,32,01,00,76,00,03	Rx: Event: SDAP Connect, Status: 00
00 / 01	RX	Request	02,52,35,02,00,89,01,11,03	Rx: Event: Service Browse, Browse Group ID: 0111
00 / 01	RX	Confirm	02,43,35,1E,00,96,00,01,02,10,01,11,**01**,16,42,6C,75,65,74,6F,6F,74,68,20,53,65,72,69,61,6C,20,50,6F,72,74,00,03	Rx: Event: Service Browse, Status: 00, Browse Group ID: 0210, Service ID: 0111, PortNo: 01, Service Name: Bluetooth Serial Port.
00 / 01	TX	Request	02,52,33,00,00,85,03	Tx: Cmd: SDAP Disconnect
00 / 01	RX	Confirm	02,43,33,01,00,77,00,03	Rx: Event: SDAP Disconnect, Status: 00
00	RX	Indicator	02,69,51,07,00,C1,**56,78,9A,BC,12,34**,16,03	Rx: Event: ACL Terminated, BdAddr: 56789ABC1234, Reason: 16

3.4.3.4 Establish Links

The link establishment is always performed with the same command “Establish Link”, just referring to different local ports.

Table 3-15. Establish Links to multiple slaves

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,0A,08,00,64,**01,12,34,56,78, 9A,BC,01**,03	Tx: Cmd: Establish Link, Local Port: 01, BdAddr: 123456789ABC, Remote Port Number: 01
00 / 01	RX	Confirm	02,43,0A,02,00,4F,**00,01**,03	Rx: Event: Establish Link, Status: 00, Local Port: 01
00	RX	Indicator	02,69,50,07,00,C0,**12,34,56,78,9A, BC,00**,03	Rx: Event: ACL Established, BdAddr: 123456789ABC, Status: 00
00 / 01	RX	Indicator	02,69,3E,04,00,AB,**01,0C,00,00**,03	Rx: Event: Port Status Changed, Local Port: 01, PortStatus: 0C, Break Length: 0000
00 / 01	RX	Indicator	02,69,0B,09,00,7D,**00,12,34,56,78, 9A,BC,01,01**,03	Rx: Event: Link Established, Status: 00, BdAddr: 123456789ABC, Local Port: 01, Remote Port Number: 01
	TX	Request	02,52,0A,08,00,64,**02,34,56,78,9A, BC,12,01**,03	Tx: Cmd: Establish Link, Local Port: 02, BdAddr: 123456789ABC, Remote Port Number: 01
00 / 01	RX	Confirm	02,43,0A,02,00,4F,**00,02**,03	Rx: Event: Establish Link, Status: 00, Local Port: 02
00	RX	Indicator	02,69,50,07,00,C0,**34,56,78,9A,BC ,12,00**,03	Rx: Event: ACL Established, BdAddr: 3456789ABC12, Status: 00
00 / 01	RX	Indicator	02,69,3E,04,00,AB,**02,0C,00,00**,03	Rx: Event: Port Status Changed, Local Port: 02, PortStatus: 0C, Break Length: 0000
00 / 01	RX	Indicator	02,69,0B,09,00,7D,**00,34,56,78,9A ,BC,12,02,01**,03	Rx: Event: Link Established, Status: 00, BdAddr: 3456789ABC12, Local Port: 02, Remote Port Number: 01
	TX	Request	02,52,0A,08,00,64,**03,56,78,9A,BC ,12,34,01**,03	Tx: Cmd: Establish Link, Local Port: 03, BdAddr: 56789ABC1234, Remote Port Number: 01
00 / 01	RX	Confirm	02,43,0A,02,00,4F,00,01,03	Rx: Event: Establish Link, Status: 00, Local Port: 03
00	RX	Indicator	02,69,50,07,00,C0,**56,78,9A,BC,12 ,34,00**,03	Rx: Event: ACL Established, BdAddr: 56789ABC1234, Status: 00
00 / 01	RX	Indicator	02,69,3E,04,00,AB,**03,0C,00,00**,03	Rx: Event: Port Status Changed, Local Port: 03, PortStatus: 0C, Break Length: 0000
00 / 01	RX	Indicator	02,69,0B,09,00,7D,**00,56,78,9A,B C,12,34,03,01,**03	Rx: Event: Link Established, Status: 00, BdAddr: 56789ABC1234, Local Port: 03, Remote Port Number: 01

3.5 DEFAULT CONNECTIONS

The LMX9820A offers a default connection procedure which allows to store several devices into a database-like system within the device. The stored connections are either connected after a reset or by sending the “Connect default connections” command. This allows an easy cable replacement setup but can also be used for automatic multipoint operations.

The successful or failed link establishment will be confirmed for each stored device.

If set to Operation Mode “Automatic” the LMX9820A after reset or boot-up will try to connect to each of those devices 3 times then switch to the next (see Section 1.6 "Operation Modes").

Each connection storage includes the local and remote RFComm port, the BD_Addr, a tranparent flag, which, if set, forces the device to switch to Transparent Mode after link establishment, and a Sniff mode flag, which, if set, will initiate the Sniff mode after link establishment.

Commands available:

- “Store Default Connection” (page 126)
- “Connect to Default Connection” (page 126)
- “Get List of Default Connections” (page 127)
- “Delete Default Connection” (page 128)

NOTE: For multiple connections please make sure that the NVS parameter “Ports to open” (See “Set Ports To Open” (page 145)) is configured correctly to have the appropriate number of RFComm instances initialized.

Eg. For 2 connections set “ports to open” to 0x00000003,
for 3 connections set “ports to open” to 0x00000007.

3.6 POWER MANAGEMENT

The LMX9820A power management is a combination of the firmware and the hardware supported low power modes. Depending on the system activity it decides to switch off as many hardware blocks as possible to reduce the current consumption.

3.6.1 Low Power Modes

The LMX9820A power management can be divided into six modes, which depend on the activity level of the UART interface and the bluetooth radio.

- UART Interface activity
  - — Enabled: UART fully active, RTS and CTS used for flow control (hardware handshaking)
  - — Disabled: UART disabled, RTS/CTS used for Wake up functionality (3.7)
- Bluetooth radio activity
  - — Active Link(s): Bluetooth radio and baseband handling active link, Bluetooth Low Power Modes (3.6.3) can be used to reduce power consumption on radio.
  - — Page/Inquiry Scanning: LMX9820A only scanning, discoverable/connectable for other devices, no active link
  - — No Radio activity: no active link and scanning switched off

Table 3-16. Low Power Modes

UART / Radio	No radio activity	Page/Inquiry scanning	Active Link(s)
Disabled	PM0	PM2	PM4
Enabled	PM1	PM3	PM5

- PM0:
  - — Lowest power consumption, Sleep Mode
  - — UART disabled (using Disable Transport Layer (5.2.9.1)), Wake up functionality (3.7) enabled
  - — Scanning disabled (using Set Scan Mode (5.2.14.6), parameters 0x00,0x00), Device not discoverable/connectable for other devices
  - — No active bluetooth link
- PM1:
  - — UART enabled, device listening to commands (Command Mode)

  - Scanning disabled (using Set Scan Mode (5.2.14.6), parameters 0x00,0x00), Device not discoverable/connectable for other devices
  - No active bluetooth link
- PM2:
  - Typical standby mode
  - UART disabled (using Disable Transport Layer (5.2.9.1)), Wake up functionality (3.7) enabled
  - Scanning enabled (using Set Scan Mode (5.2.14.6)), Device discoverable/connectable for other devices
  - No active bluetooth link, LMX9820A waking up host on incoming link
- PM3:
  - Default mode after boot-up
  - UART enabled, device listening to commands (Command Mode)
  - Scanning enabled (using Set Scan Mode (5.2.14.6)), Device discoverable/connectable for other devices
  - No active bluetooth link
- PM4:
  - UART disabled (using Disable Transport Layer (5.2.9.1)), Wake up functionality (3.7) enabled
  - Scanning has no influence on power management
  - Active bluetooth link, LMX9820A waking up host on another incoming link or incoming data
  - Power consumption can be reduced by Bluetooth Low Power Modes (3.6.3)
- PM5:
  - Full activity mode
  - UART enabled, device listening to commands (Command Mode) or in UART Transparent Mode (1.3)
  - Scanning has no influence on power management
  - Active bluetooth link(s)
  - Power consumption can be reduced by Bluetooth Low Power Modes (3.6.3)

PM0 is lowest power, PM5 can be seen as highest power consumption mode. As indicated, the host is able to influence the power consumption by either switching off scanning using Set Scan Mode (5.2.14.6) or by disabling the UART, using Disable Transport Layer (5.2.9.1) to enable the Wake up functionality (3.7).

Default mode after boot-up is usually PM3, in which the UART is enabled and the device is discoverable and connectable. In case the device is connected from another device or actively establishes a link, it switches to PM5.

In order to save power in a waiting situation, the host might decide to put the device from PM3 to PM2, using the Disable Transport Layer (5.2.9.1) command. The device will still be available for connections and will wake up the host on an incoming link.

If the LMX9820A is already engaged in a link, the host can decide to use Disable Transport Layer (5.2.9.1) to enable Wake up functionality (3.7). This would allow the host to power down and just wait for incoming data even on an active link.

To reach a low power sleep mode (starting from PM3), the host needs to first shut down the radio activity by disabling the scanning using Set Scan Mode (5.2.14.6). This forces the device switching to PM1. After that the host can use Disable Transport Layer (5.2.9.1) to switch off also the UART activity. The LMX9820A will activate the Wake up functionality (3.7) on the UART and go to a very low power mode.

Figure 3-6 on page 50 gives an overview of all possible transitions between the different power modes and what action triggers the change.

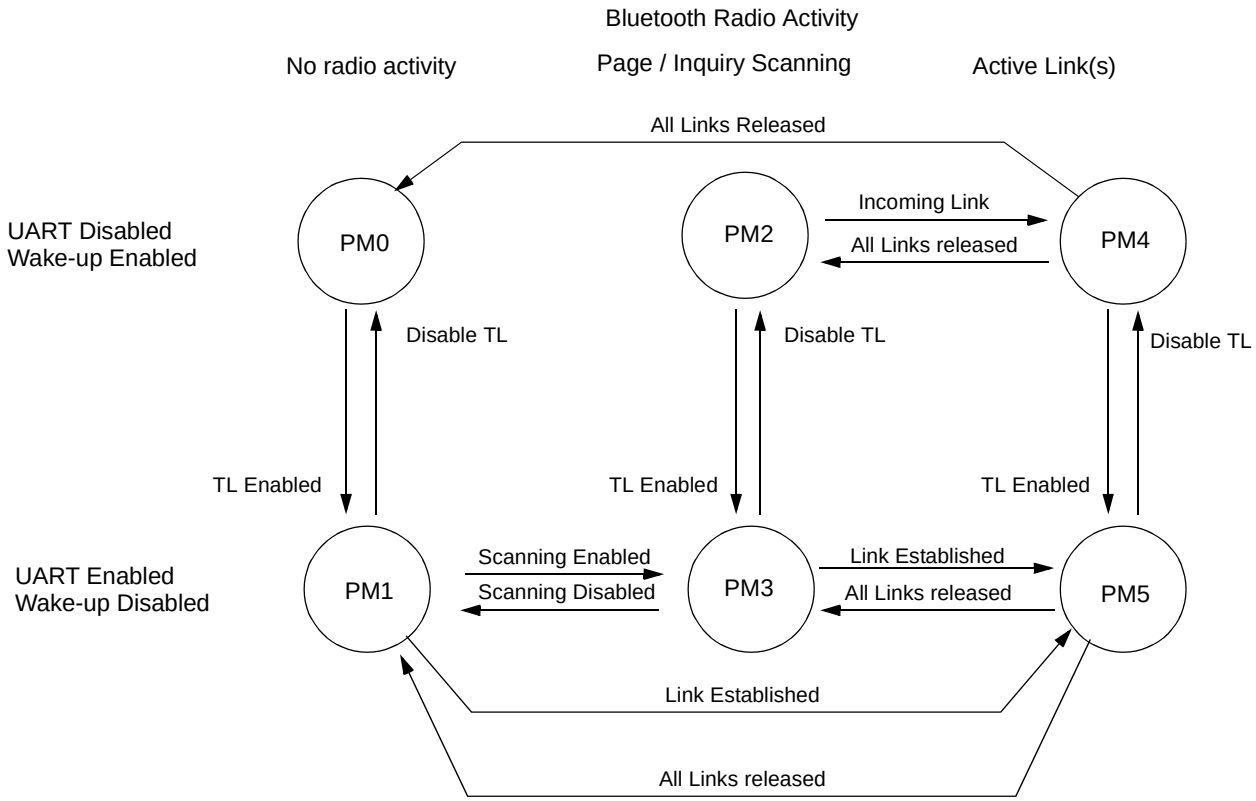


Figure 3-6. Transition between different Hardware Power Modes

3.6.2 Enhanced Power Management

The LMX9820A is able to reduce the power in the waiting state PM2, using an external 32.768khz crystal. If such a crystal is available and connected, the baseband is able to shut down the radio completely between the scanning intervals and with this to drop the power consumption to a few hundred Micro Ampere.

The usage of the 32.768 khz is controlled by the Enhanced PowerManagement (PMM) bit in NVS. By default PMM is disabled, so the 32.768kHz is not used even if connected. The PMM bit can be enabled by the Write NVS (5.2.18.16) command, configuring bit 1 of the system parameter at address 0x037A to 0.

3.6.3 Bluetooth Low Power Modes

The Bluetooth standard offers several different Low Power Modes to reduce the “active” time for the transceiver. The following modes are currently defined within the bluetooth specification:

- Sniff Mode
  - — Master and Slave device arrange certain interval to talk to each other. In between the “Sniff Slots” the slave does not listen to Master transmissions and is able to switch off the radio.
  - — Slave is still an active member of the piconet.
  - — Datatransmissions reduced to the Sniff slots.
- Hold Mode
  - — Master and Slave device agree to stop transmission for a specific time.
  - — Slave is still an active member of the piconet
  - — Slave and Master are not able to exchange data within the hold time.
- Park Mode
  - — Slave is not part of active piconet anymore
  - — Slave kept synchronized by “Beacons”
  - — Slave has to be unparked before data can be transmitted between devices again.

The LMX9820A supports all Low Power Modes.

One important parameter for switching to a Low Power Mode is the “Link Policy”. The Link policy defines, which modes are accepted for this specific link.

3.6.3.1 Link Policy

The Link Policy defines which Low Power Modes are allowed for one specific link. The settings indicate, which modes are allowed on the local device for this specific link. To use one of the features, both devices in the link have to allow the mode.

The following features can be switched on or off:

- Master/Slave Switch
- Sniff Mode
- Hold Mode
- Park Mode

3.6.3.1.1 Default Link Policy Setting

The LMX9820A includes the parameter “Default Link Policy” in NVS, which automatically is set for all incoming and outgoing links.

On default, the link policy is configured to accept Master/Slave switch as well as all Low Power Modes.

The Default Link Policy can be checked and changed by the following commands:

- “Set Default Link Policy” (page 128)
- “Get Default Link Policy” (page 129)

3.6.3.1.2 Setting Link Policy in an active link

In an active link, each of the settings can be switched on or off by the “Set Link Policy” Command.

On default the LMX9820A will use the “Default Link Policy” stored in NVS. The Link Policy of an active link can be checked and changed by the following commands:

- “Set Link Policy” (page 130)
- “Get Link Policy” (page 130)

3.6.3.2 Sniff Mode

The Sniff Mode allows to reduce the transmission slots to a specific interval, defined by parameters sent with the command. In standard operation, the slave is listening continuously to the master. This causes high power consumption on the slave. By setting a Sniff interval, the Slave does not expect packages from the Master for a specific time and therefore can switch of the radio receiver to reduce power or is able to actively talk to other devices.

The “Enter Sniff Mode” command has the following parameters:

- BD_Addr
  - The BD_Addr of the remote device
- Maximal Sniff Interval
  - The maximum interval of slots the two devices are not exchanging packages.
  - Range: 0x0006 to 0x1000 (3.725ms to 2.56s)
- Minimum Sniff Interval
  - The minimum interval of slots the two devices are not exchanging packages
  - Range: 0x0006 to 0x1000 (3.725ms to 2.56s)
- Sniff Attempts
  - Number of slots the slave has to listen to incoming packages from the master, beginning at the sniff slot.
  - Range: 0x0001 to 0x07FF
- Sniff Timeout
  - Number of slots the slave has to listen even if he still receives packages for itself.
  - Range: 0x0000 to 0x0028

As it could be possible that master or slave has to manage more than one links, the parameters give a range of sniff slots in which the “real” sniff timing will be.

The devices will agree on the parameter and confirm the mode change by an indicator (“Power Save Mode Changed” (page 134)).

After this the Sniff Mode will be active until it is released by the “Exit Sniff Mode” command.

Commands available:

- “Enter Sniff Mode” (page 131)
- “Exit Sniff Mode” (page 132)

EXAMPLE:

Calculation of Sniff parameters:

The communication should be reduced to send 1 package each 300ms. Minimum should be 50ms.

a) Calculation of Sniff interval

1 time slot: 625µs.

300ms / 625µs = 480 slots = 1E0 hex

=> maximum Sniff interval: 01E0 hex

50ms / 625µs = 80 slots = 50 hex

minimum Sniff Interval: 0050 hex

b) Sniff Attempts and Timeout

The slave starts listening at the sniff slots for Nsniff attempt consecutive receive slots unless a packet is received. After every reception of a packet, the slave continues listening at the subsequent Nsniff timeout or remaining of the receive slots, whichever is greater.

For Nsniff timeout > 0 the slave continues listening as long as it receives packets.

Note that Nsniff attempt =1 and Nsniff timeout =0 cause the slave to listen only at the first sniff slot, irrespective of packets received from the master.

Note that Nsniff attempt = 0 is not allowed.

For the example the following parameters have been successfully tested:

Sniff attempts: 5 slots

Sniff timeout: 3 slot

This means the slave listens for a minimum of 5 slots for packets from the master. After a received package it will listen for three more slots.

3.6.3.3 Hold Mode

The Hold Mode provides the ability to stop package transmission between two devices for one specific time range.

After that time they start normal transmission again.

The Hold Mode uses the following parameters:

- BD_Addr
  — The BD_Addr of the remote device
- Maximal Hold Interval
  — The maximum number of slots the two devices are not exchanging packages.
- Minimum Hold Interval
  — The minimum number of slots the two devices are not exchanging packages

The Hold Mode will be confirmed on entering and on leaving the Mode with the “Power Save Mode Changed” Indicator.

Commands available:

- “Enter Hold Mode” (page 133)

Please see Section 3.6.3.2 "Sniff Mode" for the calculation of the Hold mode parameters.

3.6.3.4 Park Mode

Park Mode enables devices to completely disconnect from each other and just keep synchronized. In that case a Master has all resources available for setting up other links or managing its current piconet. If necessary, the link to the parked slave can be re-established by a special unpark procedure.

The following parameters are used for Park Mode:

- BD_Addr
  — The BD_Addr of the remote device
- Maximal Beacon Interval
  — Acceptable longest length between beacons.
- Minimum Beacon Interval
  — Shortest length between beacons.

As the clocks of Master and slave have to be kept synchronous, the maximal beacon interval should be not to high, otherwise a reestablishment could fail.

Available commands:

- “Enter Park Mode” (page 132)
- “Exit Park Mode” (page 133)

NOTE: Since broadcast packages are not acknowledged and they are not seen as data traffic, a parked linked will be dropped after the supervision timeout. To keep a parked device connected, it has to be unparked and parked within the supervision timeout. See also Section 3.2 “Configuring the Default Link Timeout” on page 38 for details on the supervision timeout.

3.7 WAKE UP FUNCTIONALITY

In certain applications the LMX9820A will be used most of the time in a waiting status, meaning it is waiting for beeing connected or listening to commands. To reduce power consumption of the system, the LMX9820A supports a specific Wake up functionality.

The LMX9820A supports to disable the UART transport layer and switch off the command interpreter and all hardware components not needed for the current operation. The interface can be reactivated again by either side by using hardware pins.

The LMX9820A uses either the RTS signal or the HOST_WU pin to wake up the host. The RTS / CTS signals are connected in a NULL-Modem fashion, meaning that RTS on the Host is connected to CTS on the LMX9820A and vice versa.

Therefore the host would need to be able to monitor its CTS input or has to use a separate hardware pin. In case the LMX9820A has to be triggered by the host, the RTS pin is used as the hardware Wake-Up signal.

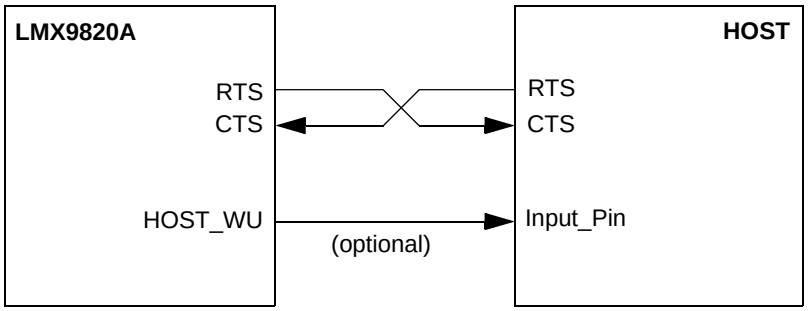


Figure 3-7. Host Wake Up hardware connections

3.7.1 Disabling the transport layer

The transport layer can be disabled in any operation state. The LMX9820A will try to wake up the host as soon as the interface would be needed again.

To disable the transport layer the following command has to be used:

- “Disable Transport Layer” (page 135)

The command will be confirmed with the standard “Disable Transport Layer” Confirm event. After the command has been sent, the LMX9820A will switch the functionality of the RTS and CTS pins from normal hardware handshake to the wake-up functionality.

Both the Host and the LMX9820A shall set RTS=0 since they may be in a sleep mode and thus both are "Not Ready to Receive". The HW Wake-Up signal is then defined as a rising edge on the CTS input i.e. a device wakes up the other device by asserting its own "Ready to Receive" output (i.e. setting RTS=1).

If the LMX9820A redefines the CTS input from "flow-control" input to "wake-up" input there will be a short period of time during which the signalling is ambiguous. To avoid this, delays are introduced as illustrated in Figure 3-8.

To guarantee no loss of data the UART shiftes out the last byte of the confirm event and the Host redefines its RTS output when it has received the last byte of the “Disable transport layer event”.

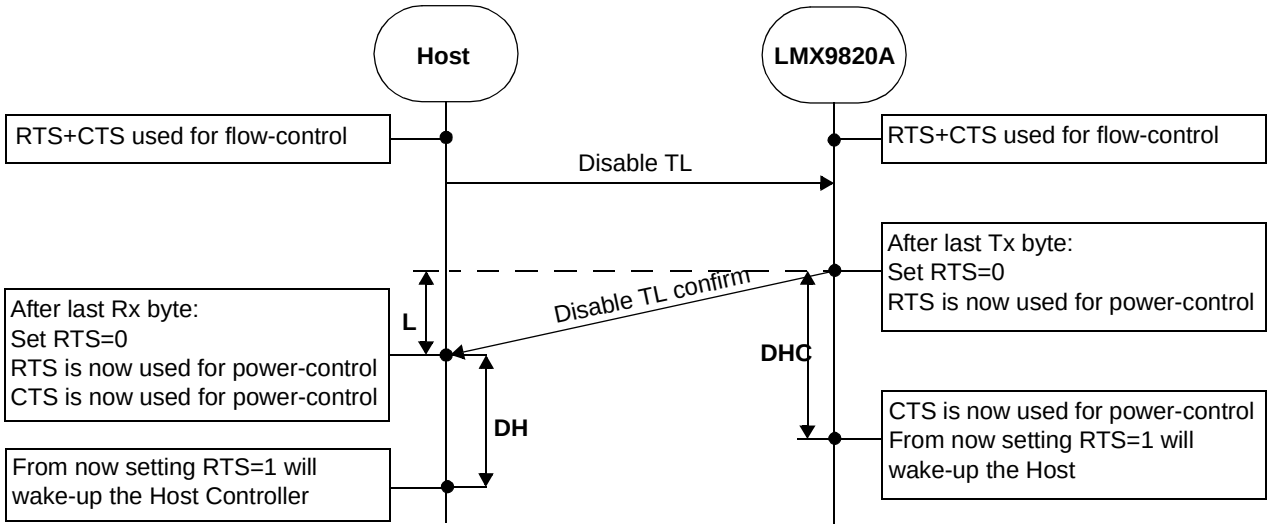


Figure 3-8. Disable Transport Layer Timing

L is the time period in which the RTS/CTS signalling is ambiguous.

DHC is the time period the LMX9820A must delay redefining CTS to Wake-Up input.

DH is the time period the Host must wait before attempting to send a Wake-Up signal to the LMX9820A by setting RTS = 1.

In order to make the mechanism work the following relations must be true:

Lmax ≥ **L** ≥ **Lmin**

DHC ≥ **Lmax**

DH ≥ **DHC - Lmin**

3.7.2 LMX9820A wakes up host

If the LMX9820A needs to send data to the host it must first make sure that the UART transport layer is enabled. If UART is disabled the LMX9820A assumes that the host is sleeping and starts the wakeup by setting RTS to 1. To be able to react on that Wakeup, the host has to monitor the CTS pin.

On receiving the CTS rising edge the host has to wake up its UART interface and switches the RTS/CTS pin functionality back to normal hardware handshake. As soon as RTS is 1, the LMX9820A will confirm the wakeup by sending the TL Enabled Event and start sending the pending events that triggered the wake-up.

Please see Figure 3-9 for the complete process.

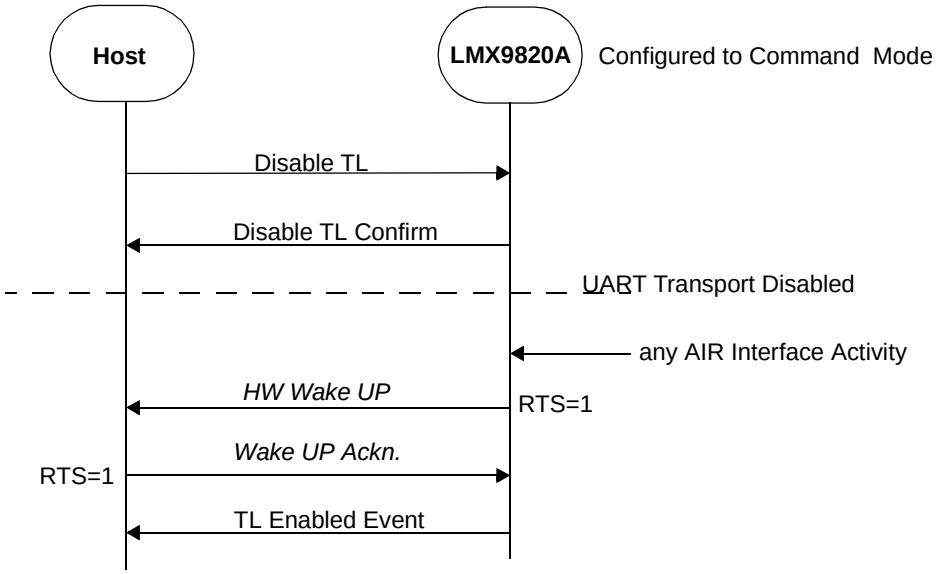


Figure 3-9. LMX9820A waking up the host in command mode

Note:

Even though the LMX9820A signals RTS=1 (i.e. "Ready to Receive") the Host may not send any data or commands to the LMX9820A before it has received the “Transport Layer Enabled” event.

The event will always be received as soon as the LMX9820A re-enables the UART interface again. One exception is if the device is in “Automatic Idle Mode”. If the LMX9820A is operating in default automatic idle mode waiting for beeing connected and the transport layer was switched off, it will not send the TL Enabled Event. As soon as it gets connected the LMX9820A will request the wake up and directly send the “Link Established” after successful connection establishment.

Please see Figure 3-10 for that specific case.

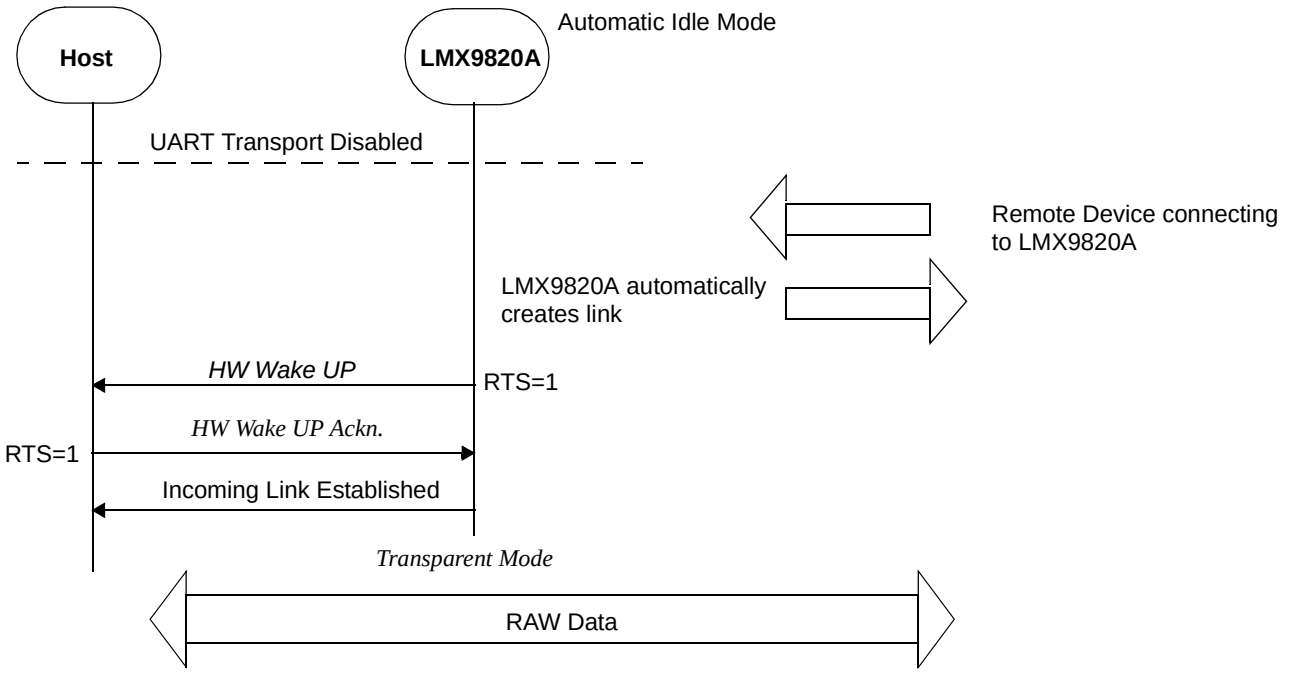


Figure 3-10. No TL enabled event after Automatic Idle

3.7.3 Host wakes up LMX9820A

If the host needs to send data or commands to the LMX9820A it must first make sure that the UART transport layer is enabled. If UART is disabled the host must assume that the host is sleeping and starts the wakeup by setting RTS to 1. The LMX9820A will wake up to the rising edge of its CTS pin.

When the LMX9820A detects the Wake-Up signal it activates the UART HW and acknowledges the Wake-Up signal by sending a “Transport Layer Enable” event. When the Host has received the “Transport Layer Enable” event, the LMX9820A is ready to receive commands.

This process is shown in the following figure

.

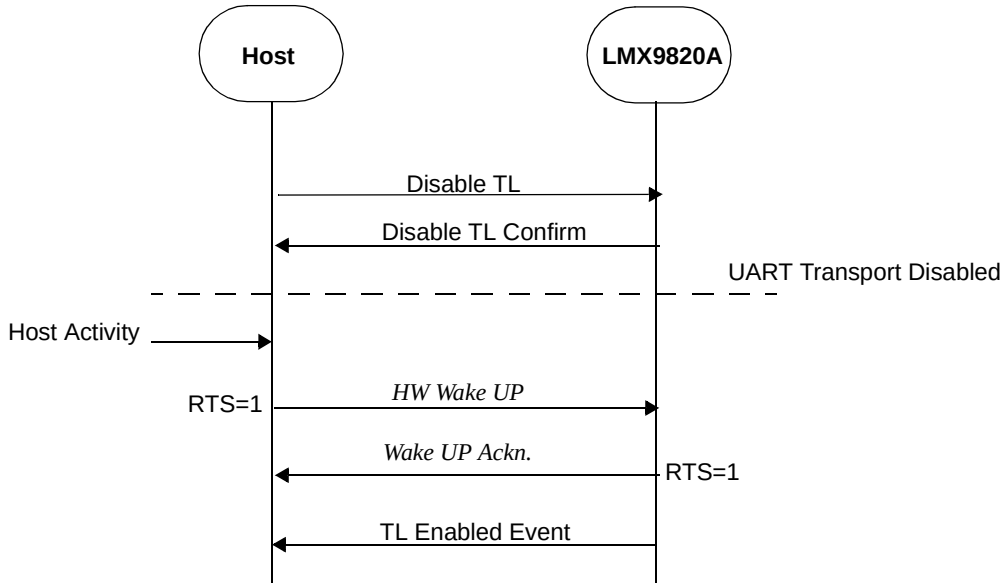


Figure 3-11. Host waking up LMX9820A

NOTE:

Even though the LMX9820A signals RTS=1 (i.e. "Ready to Receive") the Host may not send any commands to the LMX9820A before receiving the Transport Layer Enabled event.

3.8 ESTABLISH AUDIO LINKS

3.8.1 Bluetooth Background

A standard bluetooth link consists of asynchronous connectionless (ACL) links, meaning, data are sent by request from the master to slave. Since the slave has to wait for the master polls to be able to transmit data, these ACL links are not suitable for audio links. For this, the bluetooth specification defines the synchronous connection-oriented (SCO) links., which are used to transport realtime audio data. On SCO links, master and slave communicate on dedicated reserved slots. The frequency on which the master and slave will exchange packages is defined by the package type.

SCO package types:

- HV3:
  - 30bytes per package
  - no FEC
  - Master and slave exchange data every 6 slots; link consumes about 33% of the complete bluetooth bandwidth
  - mostly used since it leaves most flexibility for other bluetooth links.
- HV2:
  - 20bytes per package
  - 2/3 FEC
  - Master and slave exchange data every 4 slots; link consumes 50% of the complete bluetooth bandwidth
- HV1:
  - 10bytes per package
  - 1/3 FEC
  - Master and slave exchange data every 2nd slot; link consumes 100% of the complete bluetooth bandwidth, no other link possible

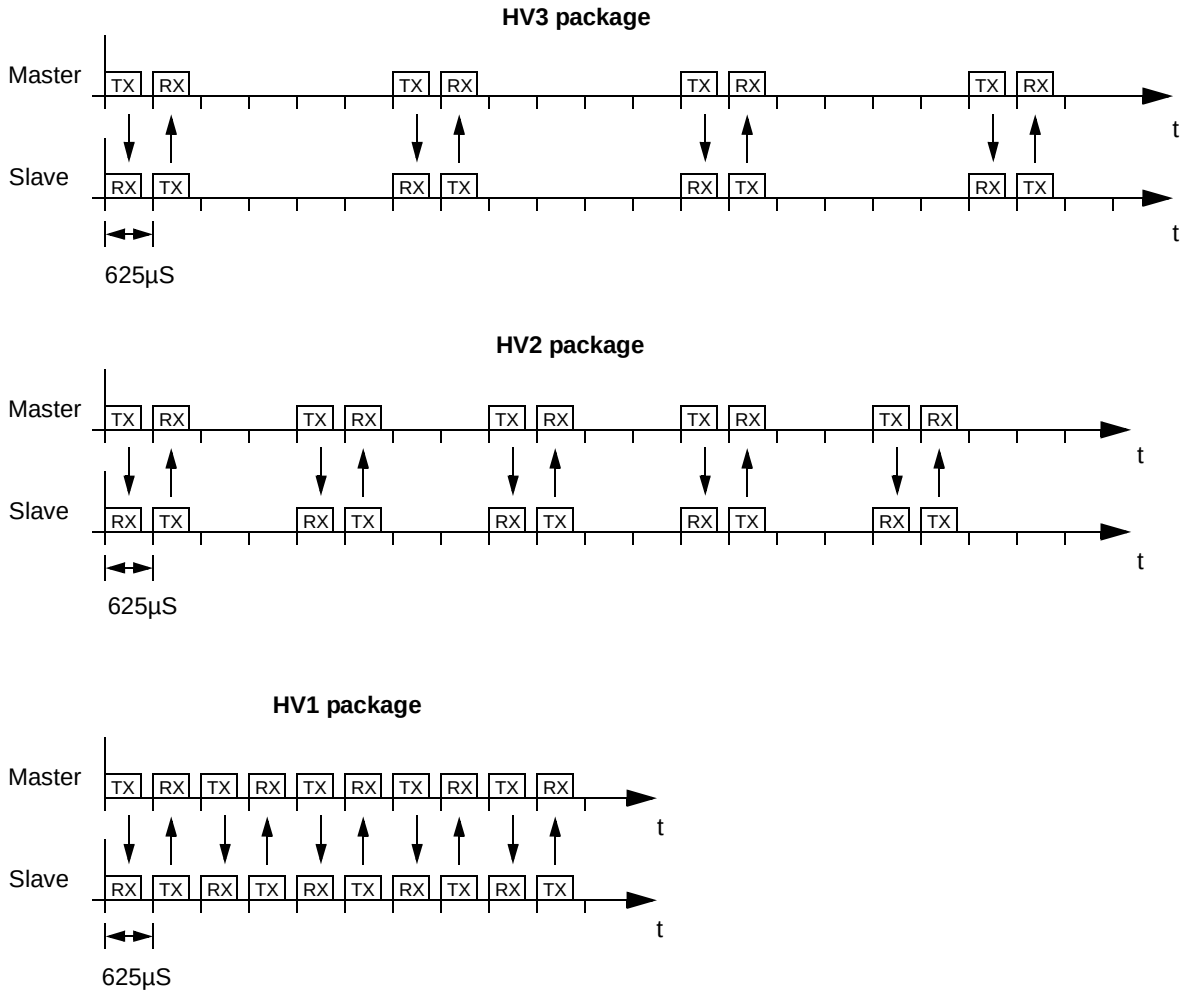


Figure 3-12. SCO link bandwidth requirements

3.8.2 Routing of audio data

As mentioned in Section 1.7 “Audio support” on page 18, the LMX9820A offers the Advanced audio interface (AAI), also called PCM interface, in order to connect an external PCM codec. The codec is used to convert the analog microphone signals into the digital PCM stream, which is then directly routed to an existing bluetooth SCO link. The same way, any incoming SCO data will be directly routed to the PCM interface to be converted into analog loudspeaker signals by the external codec.

On Firmware version 6.23 and later, the LMX9820A offers the PCM Slave codec setting which bypass the use of an external codec.The module in this configuration will use the PCM Slave settings stored in NVS.

Each SCO link is based on a previously established ACL link. The ACL link is used for the standard SPP profile. Because of this it is possible to have simultaneous data and voice transmissions. See Figure 3-13 on page 59 for the split between SCO and ACL/SPP data routing.

NOTE: Simultaneous data transmission is only possible if a SCO package type different from HV1 is used. As indicated in Figure 3-12, the HV1 package requires 100% of the bluetooth bandwidth, therefore no other transmission is possible without losing HV1 packages.

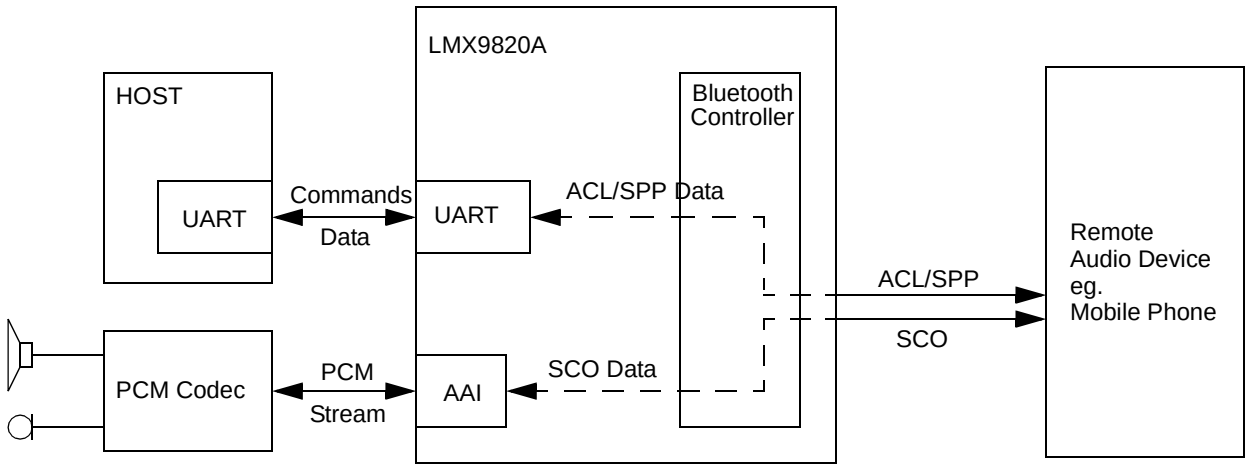


Figure 3-13. SCO and ACL/SPP routing within LMX9820A

3.8.3 Audio Link commands

The LMX9820A includes all functionality to establish and accept SCO links and routing the data to the PCM using the settings configured in the NVS. The settings for audio codec and air format can either be configured by default or for after link establishment

3.8.3.1 PCM codec configuration

In order to route the information of the SCO link to the PCM codec in the right format and timing, the firmware needs to know the codec and air format to be used. Those parameters can be configured after link establishment by the following commands:

- "Set audio settings" on page 121
- "Get audio settings" on page 122

This command defines which format should be used at the PCM interface (codec driver or PCM Slave for firmware 6.23) as well as the format used on the bluetooth link. This command only sets the parameters for the existing link.

The settings can also be stored in NVS as default for future incoming or outgoing SCO link establishments by the following command:

- "Set Default Audio Settings" on page 170
- "Get Default Audio Settings" on page 170

Please see also Figure 3-14 on page 60 for the influence of both parameters to the blocks of the LMX9820A. The codec setting configures the AAI interface timing to be able to interact with the external codec. The air format configures which format will be used on the bluetooth link. Both parameters influence the configuration of the internal CVSD codec, which will take care of the conversion to the appropriate formats.

In case PCM Slave is used, these settings have to be stored separately. Please refer to Section 3.8.3.1.1 “Using PCM Slave” on page 62 for details.

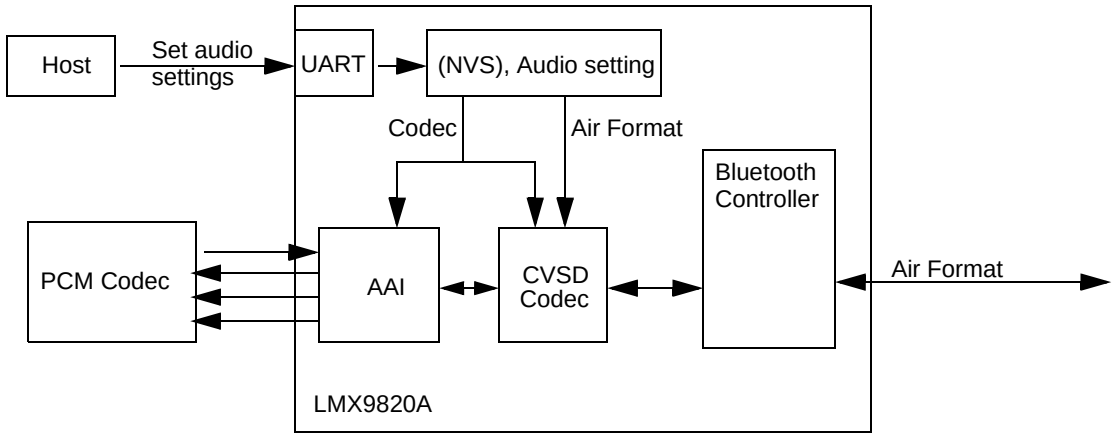


Figure 3-14. Influence of audio settings to LMX9820A blocks and formats

3.8.3.1.1 Using a specific codec driver

The LMX9820A includes two specific drivers, which are optimized for the Motorola MC145483/Winbond W681360 codec or the OKI MSM7717 / Winbond W681310 codec.

The interface settings depend on the external clock used for the LMX9820A. The following parameters have to be considered to interface with the AAI in that configurations:

- Bit Clock (Pin SCLK)
- Frame Clock (Pin SFS)
- Data format (13-bit linear, 8-log PCM a-Law)

In general these settings define the signalling on STD and SRD pins for the data. Figure 3-15 shows the principle for the relation between frame sync signal and a data slot. The frame sync signal is typically raised in a frequency of 8kHz. The data are sampled at the bit clock provided by the LMX9820A at pin SCLK.

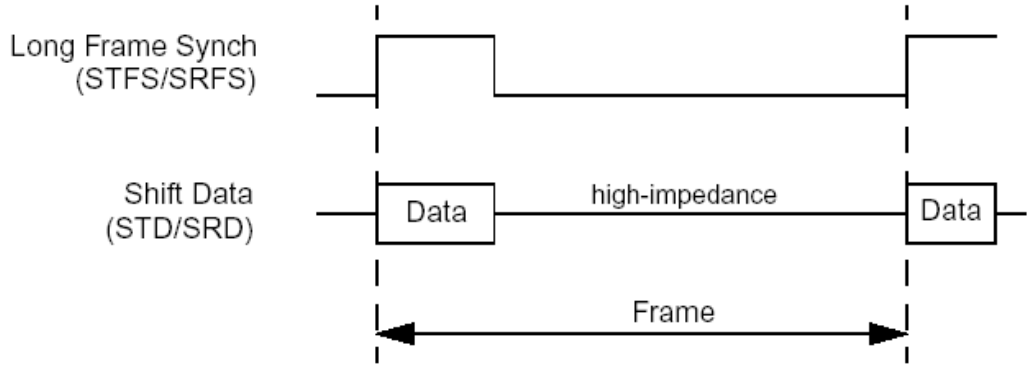


Figure 3-15. AAI generic data framing

Example:

Motorola MC145483 at 13Mhz external clock.

- Bit Clock:520kHz

- Frame Clock:8kHz
- AAI Frame Sync Pulse: Short

The Motorola codec supports 3-bit audio control at the end of each data word. This information is used by the codec to control the volume. The codec requires a 13-bit linear data format. Figure 3-16 on page 61 shows the generic setup of the 16-bit data word for the Motorola codec.

Figure 3-17 on page 61 gives the overview and the relation between the actual SFS signal and the data slot. Figure 3-18 on page 62 finally shows the real transmission, with actual bit clock and the 13-bit data word.

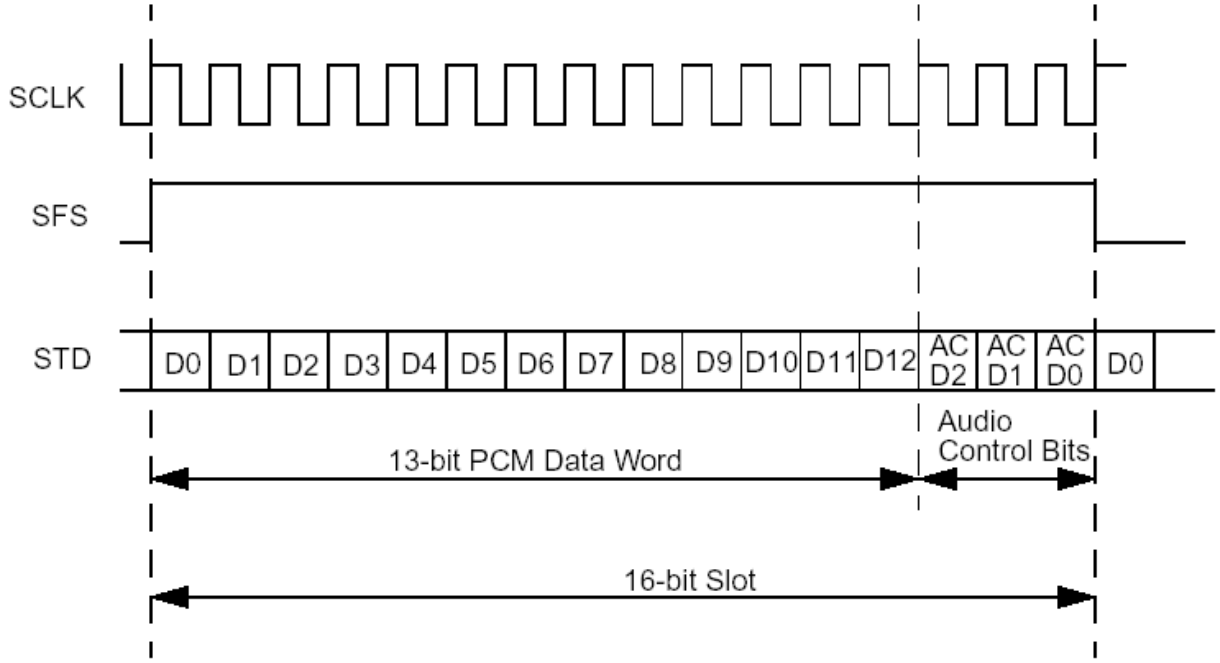


Figure 3-16. Theoretical AAI Interface stream at Motorola settings

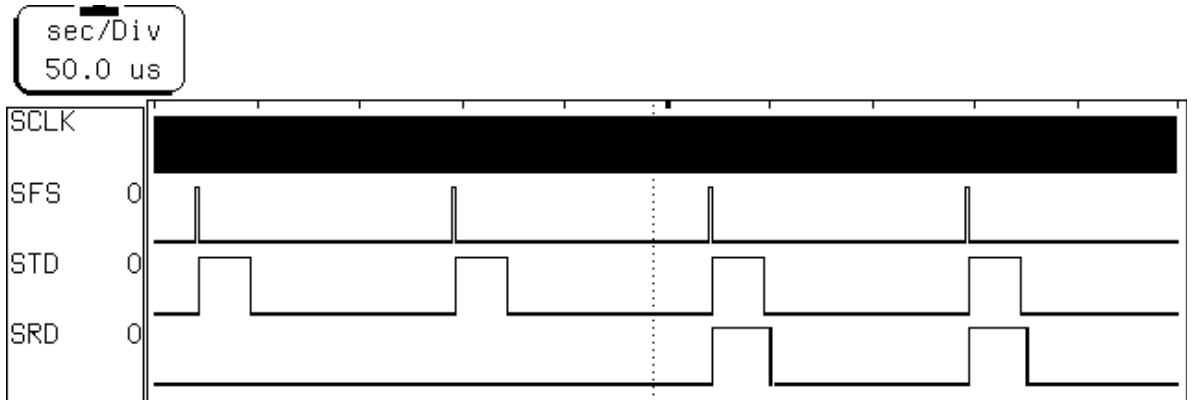


Figure 3-17. AAI stream overview for Motorola codec

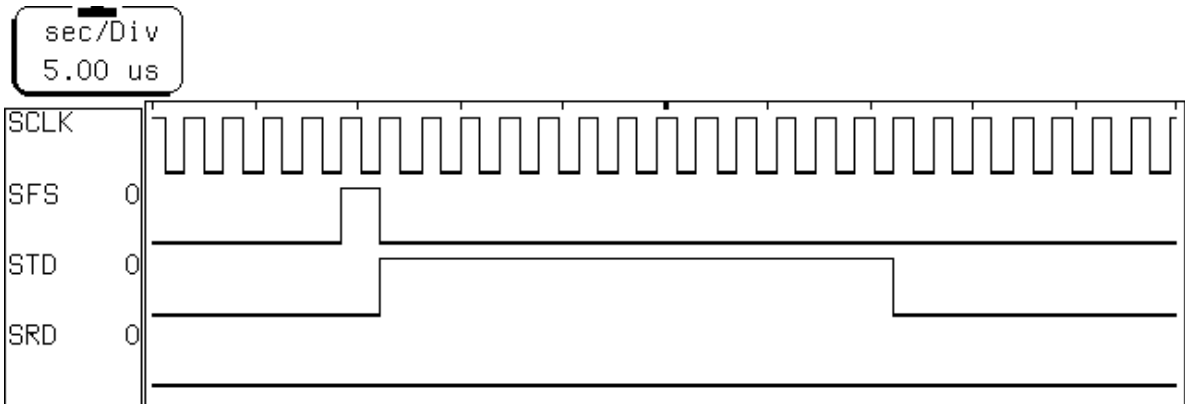


Figure 3-18. AAI slot usage for Motorola with audio control set to 0xFF

3.8.3.1.1 Using PCM Slave

Since the LMX9820A is also intended to be used in PDA like devices, the audio interface allows to be driven from an external controller or DSP. This means, the LMX9820A will sample the data from or to the DSP on the Bit clock and Frame clock provided by the DSP.

In order to be able to adjust the internal timing, the LMX9820A audio interface needs to be configured to the specific settings. The settings should be chosen so that the actual frame sync is configured to 8Khz. The bit-clock should not exceed the range of 128khz to 1024khz.

The following parameters need to be configured in order to get a correct PCM stream setup. The settings have to be configured by the command.

a.) Slot Selection
The AAI supports up to four simultaneous slots of which the LMX9820A will extract and send the data. This allows a DSP to drive multiple codecs the same time, differentiating the codecs on the slot. In a single connection slot 1 should be used

b.) Number of slots per frame
This configures how many slots should be transmitted per frame. Maximum is four.

c.) PCM data format
The data format used. The on-chip codec supports

- 8-bit A-law
- 8-bit u-law
- 13/14/15/16 bit linear

d.) Frame sync length
Defines, if a short or a long frame sync signal is used.

Short frame sync pulse:
A short frame sync pulse has the length of one data bit. Using short fame sync pulses, the transfer of the first data bit or the first slot begins at the first positive edge of the shift clock after the frame sync pulse (negative edge).

Long frame sync pulse:
For 8 bit data the frame sync generated will be 6 bits long and for 16 bit data the frame sync can be configured to 13, 14, 15 or 16 bits long. When receiving frame syncs it should be active on the first bit of data and stay active for a least two bits and it needs to go low for at least one clock cycle prior to starting a new frame. Using long frame sync pulses, the transfer of the first word (first slot) begins at the first positive edge of the shift clock after the positive edge of frame sync pulse.

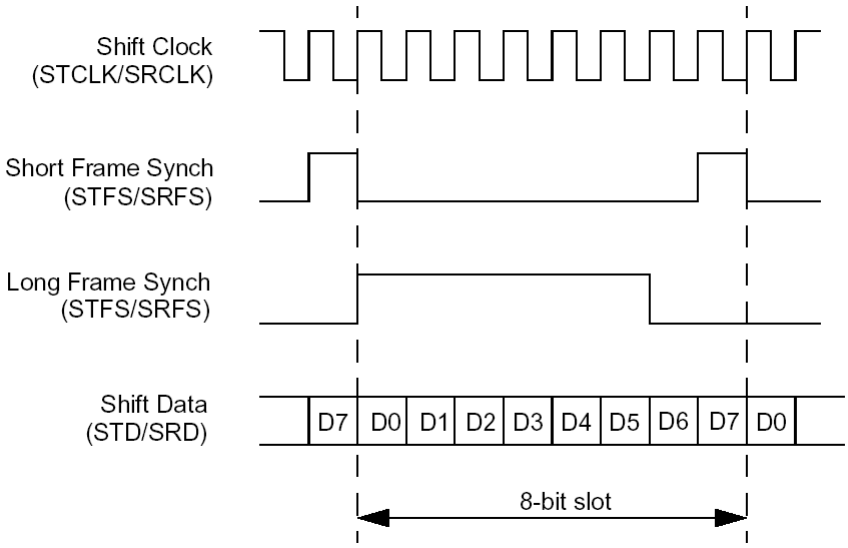


Figure 3-19. Difference between long and short frame sync signal

e.) Data word length
The LMX9820A supports data word length of 8 or 16bit.

f.) Frame sync polarity
The Frame pulse signal can either be used normal or inverted.

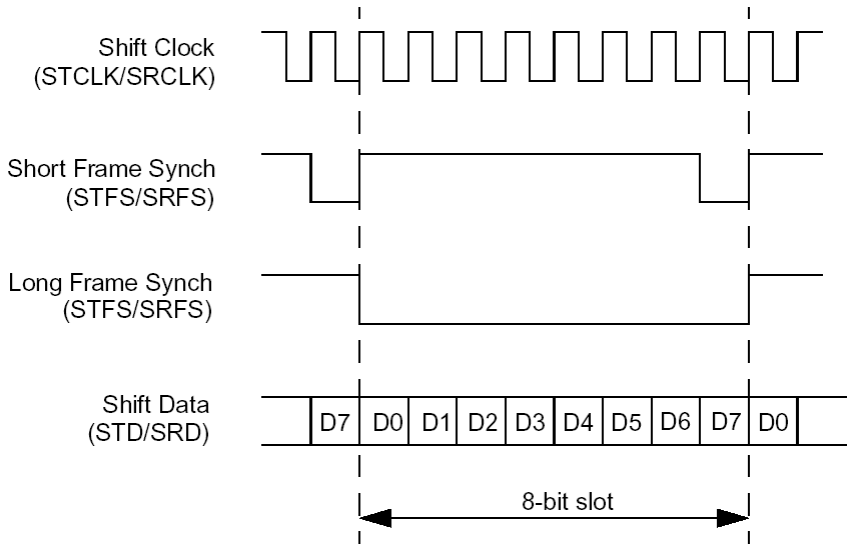


Figure 3-20. Inverted Frame sync pulses

3.8.3.2 SCO Link Establishment

In general a SCO link can only be established on top of an existing ACL link. So in order to establish a SCO link to another device, it first has to establish a standard SPP Link, using the Establish SPP link command. Once this has been set up, a SCO link can be created to that device, using the following command.

- "Establish SCO Link" on page 119

The command requires the BD_Addr as well as the SCO link package type (HV1, HV2 or HV3) to be used.

3.8.3.3 Accept incoming SCO Links

The LMX9820A will automatically accept incoming SCO link requests and confirm it with the indicator as follows. The firmware will use the default audio settings stored in NVS for the PCM codec driver and Bluetooth air format.

- "SCO Link Established Indicator" on page 120

3.8.3.4 Changing SCO package type

Once a link has been established the devices are able to switch the package type in order to optimize their bandwidth or audio quality requirements. The commands will require the BD_Addr as well as the new packet type to be used.

- "Change SCO Packet Type" on page 123

In case the remote device changed the package type, it will be indicated to the host by the following indicator.

- "Change SCO Link Packet Type Indicator" on page 123

3.8.3.5 Configuring Volume and Microphone level

The LMX9820A commandset offers commands to configure the volume of the loudspeaker output on the PCM codec or to Mute the microphone. The commands can only be executed on existing links.

Please see details in the command section:

- "Mute" on page 124
- "Set Volume" on page 124
- "Get Volume" on page 125

3.8.3.6 Releasing a SCO link

Releasing a SCO link only requires to send the following release command to the LMX9820A. The command doesn't require any other parameter to be set, since the LMX9820A can only support one SCO link at a time anyway.

- "Release SCO Link" on page 120

As final confirmation that the SCO link is released, it will be indicated by the following message.

- "SCO Link Released Indicator" on page 121

3.8.4 Example

To establish an audio link to other devices, normally the guidelines for a headset or handsfree profile have to be followed. Because of this please refer to Section 4.0 "Profile Support" for details, how to handle audio links.

4.0 Profile Support

The LMX9820A is a full bluetooth node including the Generic Access profile (GAP), Service Discovery Profile (SDAP) and Serial Port Profile (SPP). In addition to those standard profiles, additional profiles can be implemented on host, using the LMX9820A as gateway to transport the higher layer protocols. A typical example for such a protocol is the AT-Command-set, which is used for Dial-up Networking (DUN) or also audio profiles like Headset (HSP) or Handsfree profiles (HFP).

The following information are based on the bluetooth specification 1.1, volume 2 ([3]).

4.1 MULTIPLE SERVICE ENTRIES

The LMX9820A includes a configurable database, which includes the services to be offered to remote devices. These services, which also include the dedicated RFCOMM port number, will be offered to SDAP requests.

Section 2.3 "Setting up a link using the Command Interface" on page 21 described, how to establish a link using the LMX9820A command interface. This also includes the SDAP requests, which are necessary to get the available services and the dedicated RFComm port numbers.

4.1.1 The SDAP Service Database

By default the LMX9820A will offer one SPP profile to remote devices, which is dedicated to RFComm Port 1. The LMX9820A supports up to 16 database entries. See Table 4-1 for the default service database configuration.

Table 4-1. Default Service Database Configuration

Index	Enabled	Local RFComm Port	Type	Name	Authentication	Encryption	Profile Specifics
00	1	01	SPP	COM1	0x02	0x04	none
01							
0F							

Once a remote device connects to the LMX9820A, the firmware will first compare the RFCOMM port connected to the service database. In case no entry is found, the default settings as shown at index 0x00 in Table 4-1 will be used. In case an appropriate entry for the addressed RFCOMM port exists, the LMX9820A will check the authentication and encryption settings and initiate the necessary procedures.

Independent of the stored profile, the LMX9820A will not do any additional profile specific procedures beyond the SPP link establishment including authentication and encryption. In case other profiles like DUN or OPP have been stored, the host has to monitor the RFCOMM port number which has been connected (Incoming Link Established Indicator (5.2.3.7)) and has to react with the appropriate profile procedures.

4.1.2 Configuring the service database

The service database can be configured by a few simple commands. Each of the following commands will add a specific entry to the database. Once the command has been sent, the new entry will automatically stored to the first unused index.

The index number will be confirmed within the appropriate command confirmation events.

- “SDP Store SPP Record” (page 148)
- “SDP Store DUN Record” (page 149)
- “SDP Store FAX Record” (page 150)
- “SDP Store OPP Record” (page 151)
- “SDP Store FTP Record” (page 152)
- “SDP Store IrMCSync Record” (page 153)
- “Store Generic SDP Record” (page 154)

The index can be used to enable/disable the different entries during operation.

- “SDP Enable SDP Record” (page 147)

The entries can not be deleted separately. The database will only be deleted or reset to default by the following commands.

- “SDP Delete All SDP Records” (page 147)
- “Restore Factory Settings” (page 174)

4.1.3 Example

As mentioned before the LMX9820A will only use the RFCOMM port and the Authentication and Encryption parameters of the stored profiles. In case profiles other than SPP shall be implemented the LMX9820A will only handle the SPP link establishment, the remaining profile specific actions like AT commands for DUN need to be implemented on the host.

For this the host needs to know, which profile has been connected.

The following sections describe an example for the configuration of the LMX9820A service database to offer 3 services and how the host will be notified on the addressed profile.

In this example, the LMX9820A shall be used as bluetooth node within a mobile phone, offering the following services

- Serial Port Profile, SPP
- Dial Up Networking, DUN
- Headset Audio Gateway, HSAG

4.1.3.1 Configuring the service database

The first step in enabling the phone for these services is to add the service record to the service database. For this the “SDP Store DUN Record” and the “Store Generic SDP Record” need to be used.

For the setup the default SPP entry as set by Restore Factory Settings (5.2.18.8) is assumed and will not be modified. Therefore only the DUN and the HSP profile entries need to be made. Table 4-2 shows the two commands to be sent to the LMX9820A.

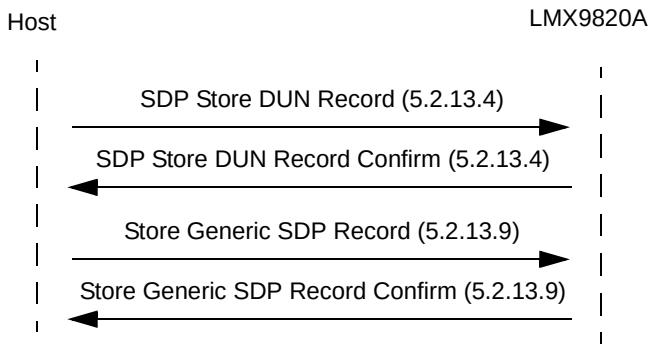


Figure 4-1. Configuring the service database

Table 4-2. Configuring the service database

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,2C,08,00,86,02,02,04,04,44, 55,4E,00,03	Tx: Cmd: Store DUN Record, Local Port: 02, Authentication: 02, Encryption: 04, Service Name: DUN.
00 / 01	RX	Confirm	02,43,2C,02,00,71,00,01,03	Rx: Event: Store DUN Record, Status: 00, Identifier: 01

Table 4-2. Configuring the service database

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,31,58,00,DB,03,22,44,53,00,00,00,0A,00,00,00,00,01,00,35,06,19,12,11,19,03,12,04,00,35,0C,35,03,19,00,01,35,05,19,03,00,08,03,05,00,35,03,19,02,10,06,00,35,09,09,6E,65,09,6A,00,09,00,01,09,00,35,08,35,06,19,08,11,09,00,01,00,01,25,0E,56,6F,69,63,65,20,47,61,74,65,77,61,79,00,03	Tx: Cmd: Store SDP Record, Local Port: 03, Authentication: 22, Encryption: 44, SdpRecord: 00000A000000000100350619121119031204003 50C3503190001350519030008030500350319021006003509096E65096A0009000109003508350619081109000100 01250E566F6963652047617465776179 00
00 / 01	RX	Confirm	Rx(RAW): 02,43,31,02,00,76,00,02,03	Rx: Event: Store SDP Record, Status: 00, Identifier: 02

After these commands the service database will be filled as described in Table 4-3.

NOTE: This example shows each profile connected to a different RFCOMM port. This makes it easy for the host to differentiate between the different profiles. As the maximum amount of RFCOMM ports to be opened on the LMX9820A is 3, it is also possible to connect one RFCOMM port to multiple profiles. In this case the application on the host needs to be able to differentiate between the different profiles on application level.

Table 4-3. Example Service Database Configuration

Index	Enabled	Local RFComm Port	Type	Name	Authentication	Encryption	Profile Specifics
00	1	01	SPP	COM1	0x02	0x04	none
01	1	02	DUN	DUN	0x02	0x04	none
02	1	03	HSAG	Voice Gateway	0x22	0x44	none

4.1.3.2 Configuring RFCOMM ports to open

The LMX9820A can handle up to three RFComm ports simultaneously. Each RFComm port opened creates a buffer instance within the RAM to handle the upcoming data traffic. Because of that by default only RFComm port 1 is activated.

The RFComm ports opened and initialized for operation can be configured by the following command:

- Section 5.2.12 "RFcomm Channels to open" on page 145

The ports in this command are expressed by a 32-bit mask indicating which RFCOMM ports the LMX9820A has to open. Bit 30 and 31 must be set to 0. Bit 0 is RFCOMM port 1 and bit 29 is port 30. The changes take effect as soon as the command has been confirmed.

Examples:

Open RFComm port 1: Set Ports to open to 0x00000001
Open RFComm port 1 and 3: Set Ports to open to 0x00000005
Open RFComm port 1, 2 and 3: Set Ports to open to 0x00000007

In this example, ports 1 to 3 would need to be opened, so the parameter has to be configured to 0x00000007.

NOTE: The LMX9820A is optimized for a maximum of 3 open RFCOMM ports. Any additional port can reduce the ability to establish ACL links.

Table 4-4. Open the necessary RFCOMM Ports

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,22,04,00,78,07,00,00,00,03	Tx: Cmd: Set Ports To Open, Ports: 07000000
00 / 01	RX	Confirm	02,43,22,01,00,66,00,03	Rx: Event: Set Ports To Open, Status: 00

4.1.3.3 Handling incoming links

As described in Section 4.1.3.1 on page 66, for incoming links the LMX9820A will check the service database for configurations on authentication and encryption and, if appropriate, serve profile specific attributes. Once the link is established the LMX9820A will indicate the incoming link and RFCOMM port number to the host with the standard command Incoming Link Established Indicator (5.2.3.7). Afterwards the host needs to decide according to the RFCOMM port number addressed, which service has been contacted and which profile specific procedure are necessary.

Figure 4-2 shows a typical flow, in which the remote device selects the DUN profile out of a service database.

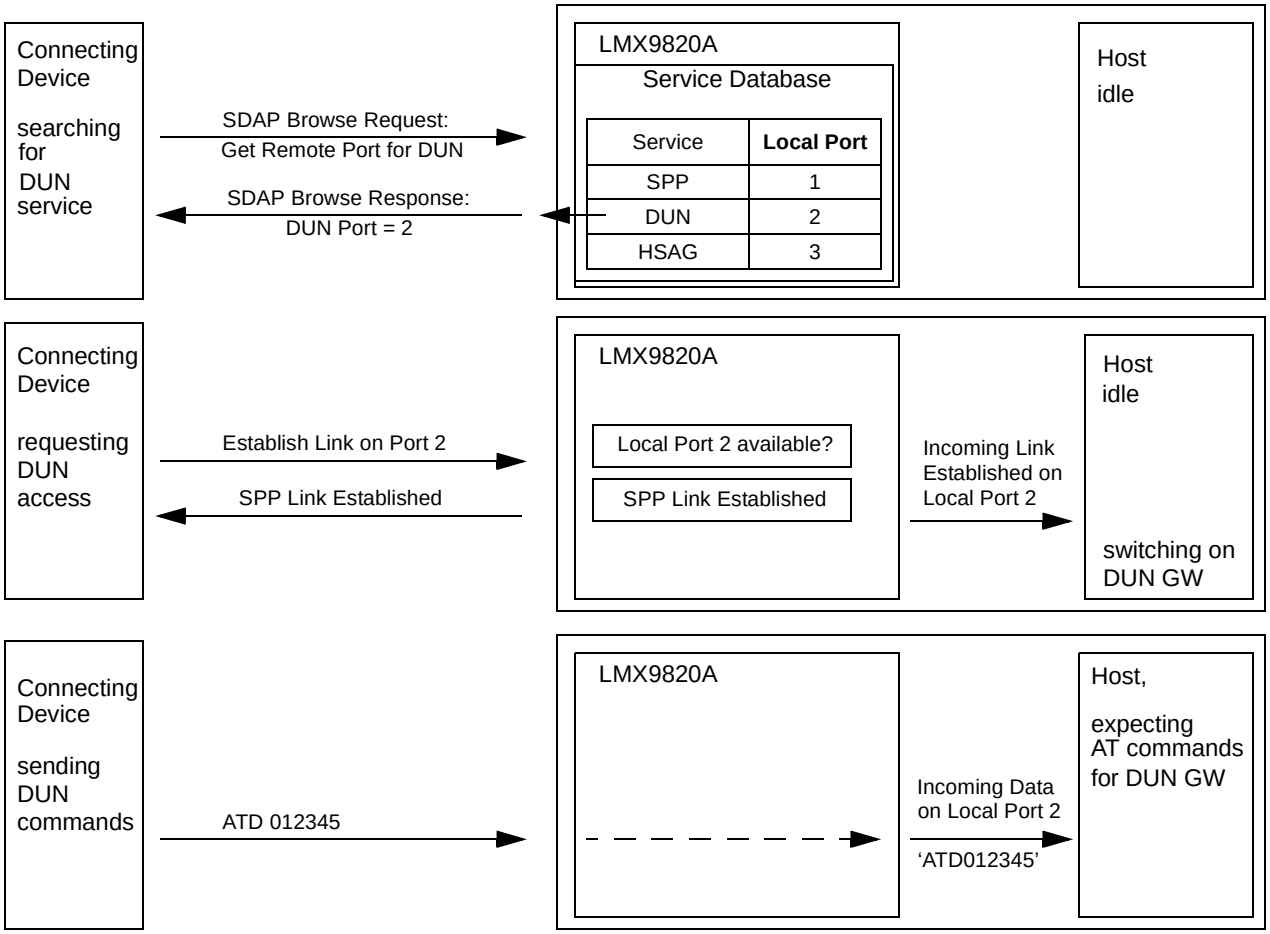


Figure 4-2. Example differentiate different profiles

The link establishment will be handled as standard SPP link. As first action, the connecting device needs to ask for the RFCOMM port it has to use for the required service. So in this case, the connecting device will first create a SDAP connecting and send a service browse request including the UUID for the desired service. The LMX9820A will answer with RFCOMM port number stored for DUN in NVS, in this example port 2.

Afterwards the connecting device requests a standard SPP link to the port number extracted from the SDAP browse. In case the RFCOMM port is available the LMX9820A accepts the incoming link, considering the settings within the service database.

Finally the LMX9820A indicates the successful link establishment by sending the "Incoming Link Established Indicator" (page 114), to the host. The event includes also the port number which has been connected.

According to the port number reported, the host needs to decide which profile procedures will have to be initiated.

Table 4-5. Host notification for incoming link on RFCOMM port 2

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
00 / 01	RX	Indicator	02,69,0C,07,00,7C,12,34,56,78,9A ,BC,**02**,03	Rx: Event: Incoming Link Established, BdAddr: 123456789ABC, Local Port: **02**

NOTE: As shown in Figure 4-2 the host will not be notified about the profile connected, it will only be notified about the RFCOMM port the remote port has established a link to. Because of this, the host needs to know the mapping between RFCOMM port and profile. This is necessary in case these profiles shall be available simultaneously.

Since most of the SPP based profiles are based on AT commands, the differentiation between the profiles could also happen on application level, meaning all profiles use the same RFCOMM port. In this case, the host will not have to store the mapping between profile and RFCOMM port but would have to have a generic AT Commandset application, supporting all different profiles. The drawback of this is that all remaining profiles will be blocked as soon as one profile is in use.

4.2 SERIAL PORT PROFILE (SPP)

The SPP profile the basic profile to be used for standard cable replacement. The SPP profile is fully included within the LMX9820A. The SPP profile is basically covered by the SPP Link establishment command. Please see the examples in Section 2.5 "Examples" on page 32.

4.3 DIAL-UP NETWORKING (DUN)

The Dial-up Networking Profile defines the protocols and procedures that shall be used by devices implementing the usage model called ‘Internet Bridge’. The most common examples of such devices are modems and cellular phones.

The following roles are defined for this profile:

Gateway (GW) – This is the device that provides access to the public network. Typical devices acting as gateways are cellular phones and modems.

Data Terminal (DT) – This is the device that uses the dial-up services of the gateway. Typical devices acting as data terminals are laptops and desktop PCs.

In the rest of this document, these terms are only used to designate these roles.

The scenarios covered by this profile are the following:

- Usage of a GW by a DT as a wireless modem for connecting to a dial-up internet access server or using other dial-up services
- Usage of a GW by a DT to receive data calls

The LMX9820A is capable of acting as both gateway or data terminal. In case of acting as DT, no further configurations are necessary. Acting as GW the LMX9820A needs to be configured to offer the DUN networking service to the DT, meaning an additional Service Record has to be stored within the device.

Both configuration scenarios are explained in more detail within the following sections.

4.3.1 AT Commandset

Dial up networking is based on a serial connection between two devices. The profile itself is handled by AT commands. So in order to create a DUN profile a standard SPP link has to be established, which then is used to transport the AT commands. For example using the LMX9820A to dial out with a mobile phone it first establishes a standard SPP link as done with SPP profile and then sending “ATDT0123456” as data to dial the specific number.

The AT commandset together with the SPP Profile form the Dial up networking profile.

Please refer to [3] to get the full of AT Commands to be supported by GW or DT devices. The commands are also listed in Section 5.4 “AT Commands” on page 185.

4.3.2 Acting as DUN data terminal

Using the LMX9820A as data terminal (DT) means, it uses the DUN service of another device, eg. a mobile phone, to dial out. The host needs to send the standard AT commands to establish the SPP link as described in Section 2.3 “Setting up a link using the Command Interface” on page 21. There are no additional configurations necessary on the LMX9820A.

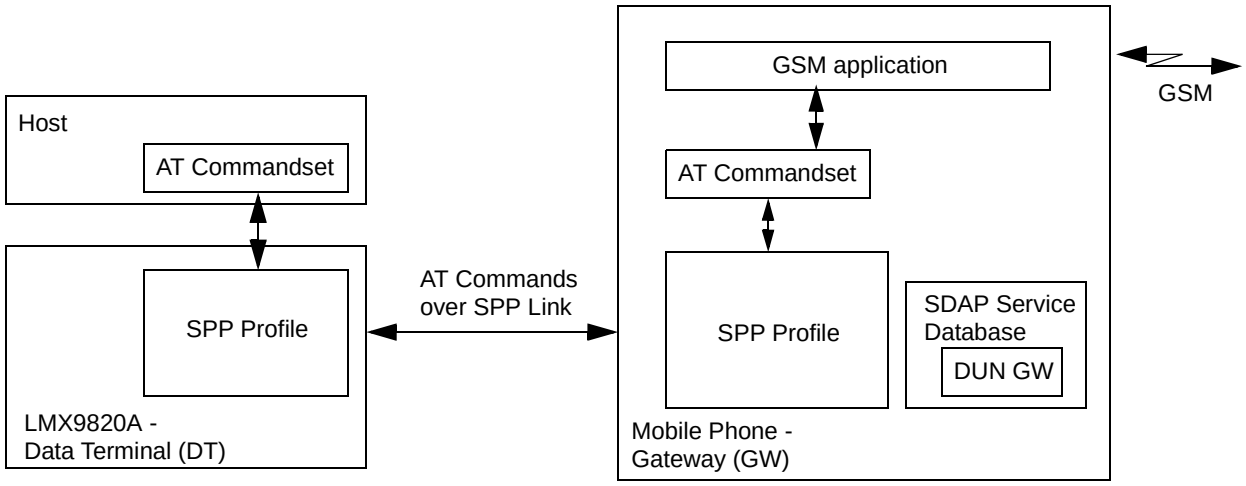


Figure 4-3. Software Layer implementations for DUN

The only difference for the link setup is that the DT has to connect to the DUN profile of the remote device instead of connecting to the SPP profile. To reach that, the correct remote RFComm port has to be used within the Establish Link (5.2.3.1) command. Once the correct RFComm port has been used, the remote device knows automatically that DUN has been addressed.

The following chapters give a step by step description of connecting to the DUN service of a mobile phone.

4.3.2.1 Searching for the device

The Inquiry is a standard procedure which is not related to any profile. With this the BD_Addr of the remote device will be known. The remote device needs to have inquiry scan switched on.

4.3.2.2 Getting the RFCOMM port for DUN on the remote device

As described in Section 2.3.3 “SDAP Service Browse for SPP” on page 24, the remote RFComm port number for the service can be found out by using the Service Browse command. The command includes the parameter UUID, which defines the service to be looked for. The UUID for Dial Up Networking profile is 0x1103. See Figure 4-4 for the command flow, Table 4-6 shows the detailed parameters to be used for the commands.

Table 4-6. Get Remote RFComm Port for DUN profile of the remote device

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,32,06,00,8A,12,34,56,78,9A, BC,03	Tx: Cmd: SDAP Connect, BdAddr: 123456789ABC
00	RX	Indicator	02,69,50,07,00,C0,12,34,56,78,9A, BC,00,03	Rx: Event: ACL Established, BdAddr: 123456789ABC, Status: 00
00 / 01	RX	Confirm	02,43,32,01,00,76,00,03	Rx: Event: SDAP Connect, Status: 00
00 / 01	RX	Request	02,52,35,02,00,89,**03,11**,03	Rx: Event: Service Browse, Browse Group ID: **0311**
00 / 01	RX	Confirm	02,43,35,1B,00,93,00,01,02,10,03, 11,**03**,13,44,69,61,6C,2D,75,70,20, 4E,65,74,77,6F,72,6B,69,6E,67,00, 03	Rx: Event: Service Browse, Status: 00, Browse Group ID: 0210, Service ID: 0311, PortNo: **03**, Service Name: Dial-up Networking.
	TX	Request	02,52,33,00,00,85,03	Tx: Cmd: SDAP Disconnect

Table 4-6. Get Remote RFComm Port for DUN profile of the remote device

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
00 / 01	RX	Confirm	02,43,33,01,00,77,00,03	Rx: Event: SDAP Disconnect, Status: 00
00	RX	Indicator	02,69,51,07,00,C1,12,34,56,78,9A, BC,16,03	Rx: Event: ACL Terminated, BdAddr: 123456789ABC, Reason: 16

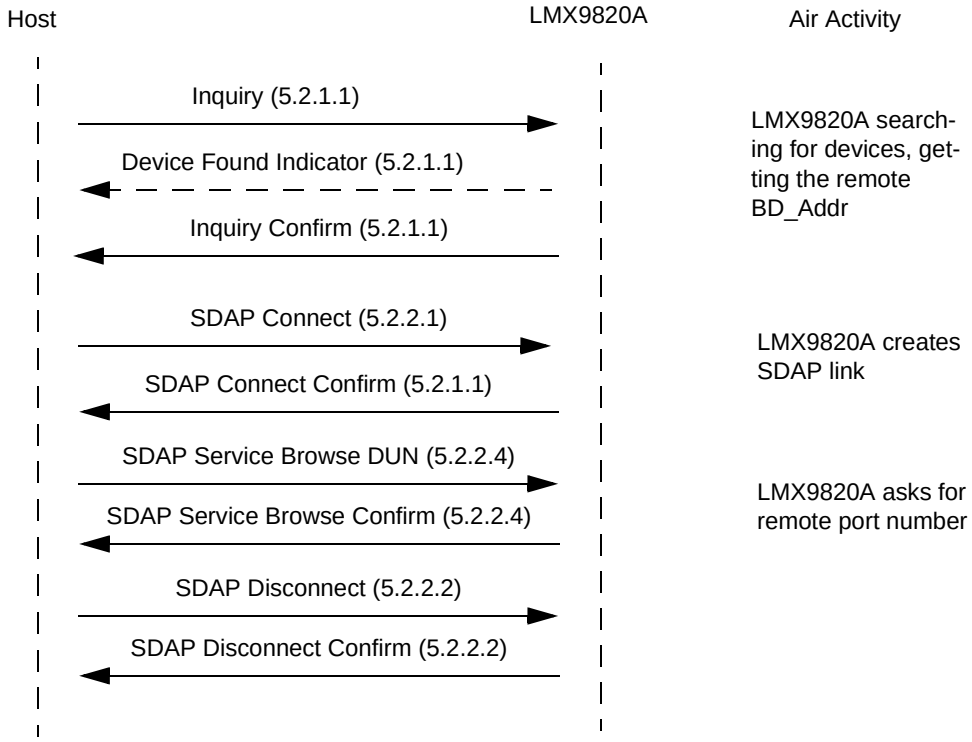


Figure 4-4. Command Flow: Requesting the RFComm Port Number from the remote device

4.3.2.3 Establishing a Link to a Dial Up Networking service

The link establishment to a DUN profile is the same as to create a link to a standard SPP port. The remote device will recognize the profile to be used by the RFComm port, to which the SPP link has been created to. For this example, out of the procedure explained in Section 4.3.2.2, the RFComm port to be addressed for DUN is 0x03.

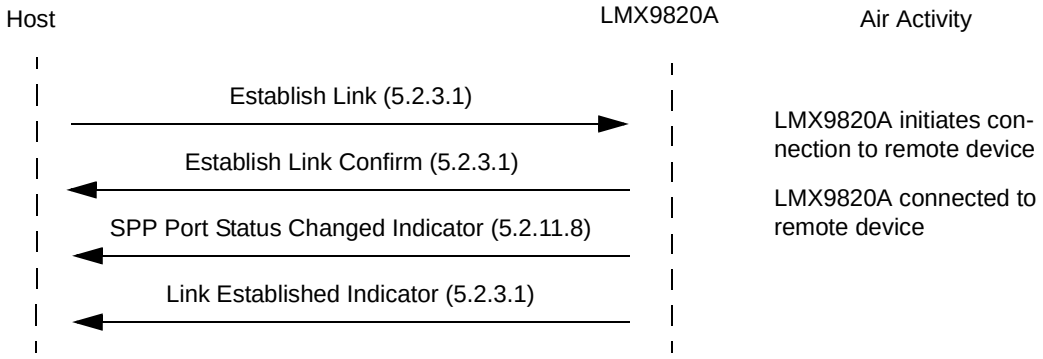


Figure 4-5. Command Flow: Establish a link to a DUN service of the remote device

Table 4-7. Establish Link to DUN service on remote device

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,0A,08,00,64,01,12,34,56,78, 9A,BC,03,03	Tx: Cmd: Establish Link, Local Port: 01, BdAddr: 123456789ABC, Remote Port Number: 03
00 / 01	RX	Confirm	02,43,0A,02,00,4F,00**,01**,03	Rx: Event: Establish Link, Status: 00, Local Port: **01**
00	RX	Indicator	02,69,50,07,00,C0,12,34,56,78,9A, BC,**00**,03	Rx: Event: ACL Established, BdAddr: 123456789ABC, Status: **00**
00 / 01	RX	Indicator	02,69,3E,04,00,AB,01,0C,00,00,03	Rx: Event: Port Status Changed, Local Port: 01, PortStatus: 0C, Break Length: 0000
00 / 01	RX	Indicator	02,69,0B,09,00,7D,00,12,34,56,78, 9A,BC,01,**03**,03	Rx: Event: Link Established, Status: 00, BdAddr: 123456789ABC, Local Port: 01, Remote Port Number: **03**

4.3.2.4 Sending AT Commands

Once the SPP link is established the GW will activate it's AT commandset, expecting and accepting a specific AT commands as described in [3]. A full list of commands to be supported by the GW can also be found in Section 5.4.1 on page 185.

AT commands are basically ASCII characters to be sent to the remote device. The commands additionally make use of Carriage Return (0x0D) and Line Feed (0x0A).

So the commands from DT to GW will be thus:

AT<cmd>=<value><cr>

If the command is processed successfully, the resulting response from the GW to the DT is:

<cr><lf>OK<cr><lf>

If the command is not processed successfully, the resulting response from the GW to the DT is:

<cr><lf>ERROR<cr><lf>

The AT commands can be sent by either using the SPP Send Data (5.2.3.3) command or by first switching to transparent mode and then send the commands directly.

4.3.2.4.1 Using the command interface to send and receive AT commands

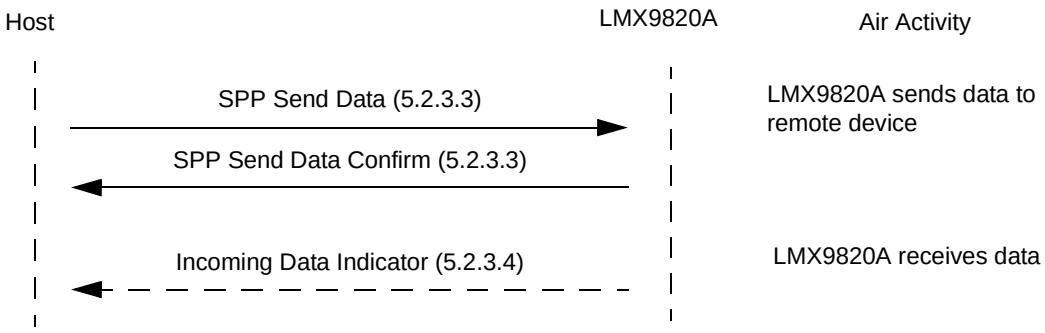


Figure 4-6. Flow Command: Sending AT Commands

Table 4-8. Send init command "AT" and dialing command "ATDT01234567890"

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,0F,07,00,68,01,04,00,41,54, 0D,0A,03	Tx: Cmd: Send Data, Local Port: 01, Payload Data: 41540D0A (AT)
00 / 01	RX	Confirm	02,43,0F,02,00,54,00,01,03	Rx: Event: Send Data, Status: 00, Local Port: 01
00 / 01	RX	Indicator	02,69,10,09,00,82,01,06,00,0D,0A, 4F,4B,0D,0A,03	Rx: Event: Incoming Data, Local Port: 01, Received Data: 0D0A4F4B0D0A *(OK)*
	TX	Request	02,52,0F,13,00,74,01,10,00,41,54, 44,54,30,31,32,33,34,35,36,37,38, 39,30,0D,03	Tx: Cmd: Send Data, Local Port: 01, Payload Data: 4154445430313233343536373839300D *(ATDT01234567890)*
00 / 01	RX	Confirm	02,43,0F,02,00,54,00,01,03	Rx: Event: Send Data, Status: 00, Local Port: 01

4.3.2.4.2 Using transparent mode to send and receive AT commands

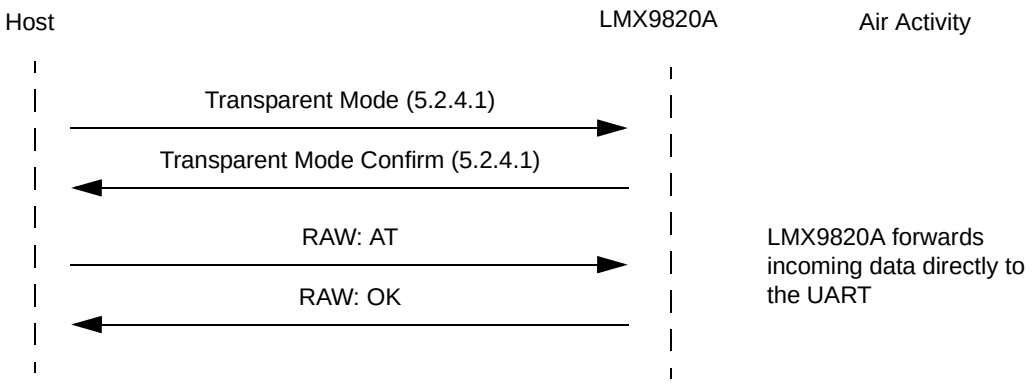


Figure 4-7. Sending AT Commands

Table 4-9. Send init command "AT" and dialing command "ATDT01234567890"

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,11,01,00,64,01,03	Tx: Cmd: Transparent Mode, Local Port: 01
00 / 01	RX	Confirm	02,43,11,02,00,56,00,01,03	Rx: Event: Transparent Mode, Status: 00, Local Port: 01
	TX	RAW Data	41,54,0D,0A	Tx(RAW): 41,54,0D *(AT)*
00 / 01 / 02 / 03	RX	RAW Data	0D,0A,4F,4B,0D,0A	Rx(RAW): 0D,0A,4F,4B,0D,0A *(OK)*
	TX	RAW Data	41,54,44,54,30,31,32,33,34,35,36, 37,38,39,30,0D	Tx(RAW): 02,52,0F,13,00,74,01,10,00,41,54,44,54,30,31,32,33,34,35,36,37,38,39,30,0D *(ATDT01234567890)*

4.3.3 Acting as DUN gateway

A DUN gateway offers the ability to access another network, e.g. GSM, routing the data between the data from the bluetooth link to the other network. The LMX9820A can be configured to offer a DUN networking service to other devices, eg as in a mobile phone or LAN access point. The host is able to differentiate the requested DUN service by checking the local port, to which has been connected to. Please see Section 4.1.3.3 on page 68 for details how to differentiate between different profiles.

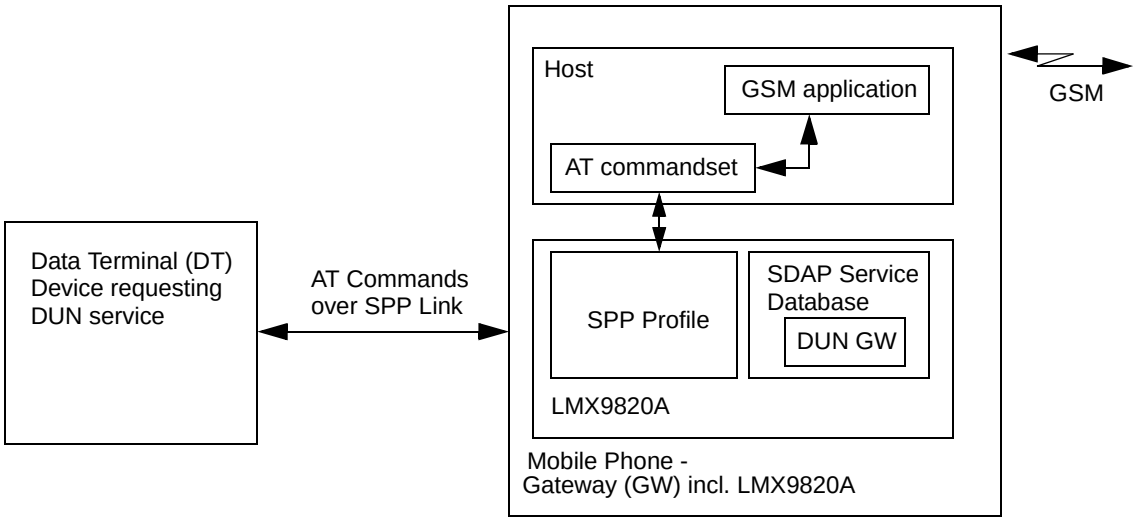


Figure 4-8. Using LMX9820A in a DUN GW application

4.3.3.1 Configuring the Service Database

In order to offer an additional DUN service to other devices a new entry has been made into the service database. For this the SDP Store DUN Record (5.2.13.4) can be used. The following example shows how to add the DUN entry to the default SDB database including only one SPP entry. .

Table 4-10. Adding the DUN service to the Service Database

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,2C,08,00,86,02,02,04,04,44, 55,4E,00,03	Tx: Cmd: Store DUN Record, Local Port: **02**, Authentication: 02, Encryption: 04, Service Name: DUN.
00 / 01	RX	Confirm	02,43,2C,02,00,71,00,01,03	Rx: Event: Store DUN Record, Status: 00, Identifier: 01

After this command the Service database has the following content:

Table 4-11. Service Database for COM1 and DUN

Index	Enabled	Local RFComm Port	Type	Name	Authentication	Encryption	Profile Specifics
00	1	01	SPP	COM1	0x02	0x04	none
01	1	02	DUN	DUN	0x02	0x04	none

4.3.3.2 Adapt RFCOMM ports to open

As both services shall be connectable independently, two separate RFCOMM ports need to be used. For this the Set Ports To Open (5.2.12.1) command should be used. .

Table 4-12. Set RFCOMM ports to open

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,22,04,00,78,03,00,00,00,03	Tx: Cmd: Set Ports To Open, Ports: **03000000**
00 / 01	RX	Confirm	02,43,22,01,00,66,00,03	Rx: Event: Set Ports To Open, Status: 00

4.3.3.3 Routing AT Commands

The DUN GW profile is based on the standard AT commandset as listed in Section 5.4.1 “DUN GW” on page 185.

Once an incoming link has been established to the appropriate RFCOMM port, the host needs to be able react to these AT commands. The LMX9820A will only route the AT commands from the host to the remote device.

Please refer to Section 4.1.3.3 “Handling incoming links” on page 68 for how to manage incoming connections. The AT commands need to be sent and recognized as described in Section 4.3.2.4 “Sending AT Commands” on page 72.

4.4 HEADSET PROFILE

The following description is based on [3]. Please refer to that document for complete description of the profile.

This Headset profile defines the protocols and procedures that shall be used by devices implementing the usage model called ‘Ultimate Headset’. The most common examples of such devices are headsets, personal computers, and cellular phones.

The headset can be wirelessly connected for the purposes of acting as the device’s audio input and output mechanism, providing full duplex audio.

The following roles are defined for this profile:

Audio Gateway (HSAG) – This is the device that is the gateway of the audio, both for input and output. Typical devices acting as Audio Gateways are cellular phones and personal computer.

Headset (HS) – This is the device acting as the Audio Gateway’s remote audio input and output mechanism.

The profile specifies guidelines for incoming and outgoing connections for both profile roles, covering the following scenarios:

- AG initiated connection establishment
- HS initiated connection establishment
- Audio connection release
  — HS initiated
  — HSAG initiated
- Audio connection transfer
  — Transfer from HSAG to HS
  — Transfer from HS to HSAG
- Remote Volume Control

The following sections describe, how to use the LMX9820A as HSAG or HS.

4.4.1 AT Commandset and Results

The command line termination character shall be carriage return (IA5 0/13). The response formatting character shall be line feed (IA5 0/10). The HSAG shall not echo command characters (Opposite to default recommendation by ITU V.250). The HSAG shall transmit result codes, using the verbose (rather than numeric) format.

The format for a command from the HS to the HSAG is thus:

AT<cmd>=<value><cr>

If the command is processed successfully, the resulting response from the HSAG to the HS is:

<cr><lf>OK<cr><lf>

If the command is not processed successfully, the resulting response from the HSAG to the HS is:

<cr><lf>ERROR<cr><lf>

The format for an unsolicited result code (such as RING) from the HSAG to the HS is:

<cr><lf><result code><cr><lf>

The headset profile uses a subset of AT commands and result codes from existing standards.

Please see ... for the complete list of commands necessary for the headset profile.

4.4.2 Acting as Headset

The headset is defined as the device, providing the input and output for audio data, eg. microphone and speaker. The LMX9820A is able to connect an external codec via the onchip advanced audio interface (AAI). Please see Figure 1-8 "PCM Codec connection block diagram" on page 18 how to connect the codec to the AAI.

The HS side of the headset profile is mostly controlled by Audio gateway. While the SPP link establishment can be initiated from both sides, the SCO link establishment will always be initiated by the HSAG. The LMX9820A will automatically accept incoming SCO links and report the successful establishment to the host.

The only configuration necessary is to create the service database record and to configure the default audio settings for the PCM codec driver.

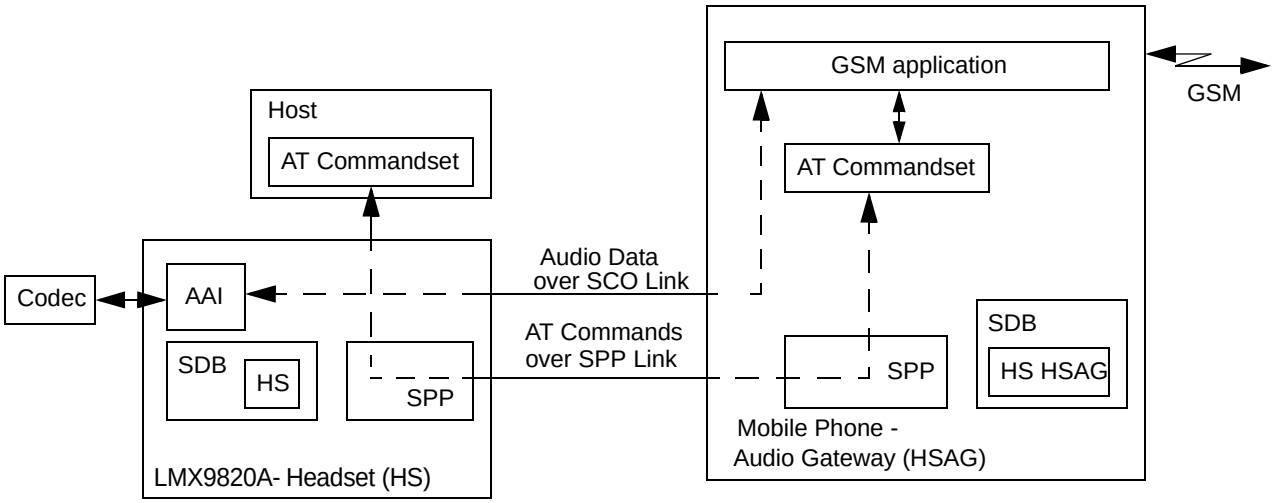


Figure 4-9. Headset Data and Audio stream

4.4.2.1 Configuring the audio path

In general the LMX9820A can be seen as gateway between the UART interface and the bluetooth link. The audio link is one additional interface for the LMX9820A to be routed. Please see also Figure 3-13 on page 59 for the two routing options.

The LMX9820A supports different configurations for the PCM codec interface and the air interface. The bluetooth specification defines the following formats to be used over the bluetooth link:

- CVSD (default)
- µ-Law
- A-Law

As indicated, the default and mostly used coding format used over a bluetooth link is CVSD. Since the headset most likely will always use the same settings, it is recommended to set the parameters to a default in the NVS, using the Set Default Audio Settings (5.2.18.1) command.

Table 4-13. Set Default Audio Settings to OKI codec and CVSD

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,5B,02,00,AF,**02**,**00**,03	Tx: Cmd: Set Default Audio Settings, Codec Type: **02**, Air Format: **00**
00 / 01	RX	Confirm	02,43,5B,01,00,9F,00,03	Rx: Event: Set Default Audio Settings, Status: 00

4.4.2.2 Configuring the Service Database for Headset

In order to offer an additional HS service to other devices a new entry has been made into the service database. For this the Store Generic SDP Record (5.2.13.9) needs to be used. This command is a generic command to generate any possible profile within the device. To be able to do so, significant knowledge of the internal SDB structure would be required. In order to use this command, the “Simply Blue Commander” software needs to be used, to create this command.

The generation of profile entries is supported in Simply Blue Commander versions later than 1.3.0.3. Please follow the following figures to create a headset service database entry:

1) Select “Definitions/Create Service Record” within the Simply Blue Commander menu.

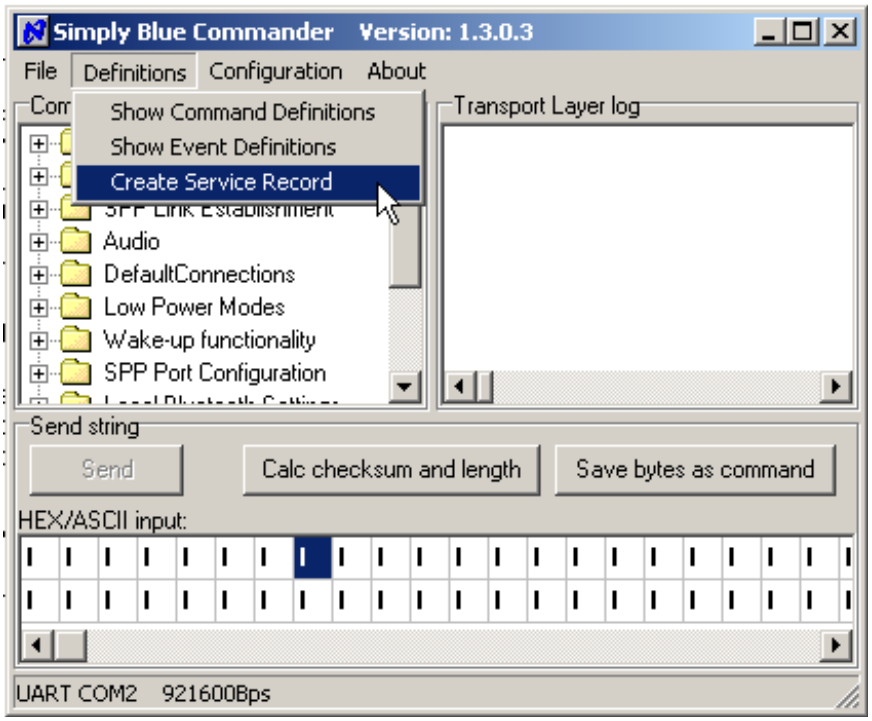


Figure 4-10. Opening “Create Service Record” Dialogbox

2) Select the service record required, in this case “Headset”

Figure 4-11. Selecting the profile

3) Select the settings desired for the headset profile and choose the correct RFCOMM port. As this example device shall still be able to accept a second standard data links as well, RFCOMM port 2 should be used. Finally confirm the dialog with pressing “Create”.

Figure 4-12. Configuring the profile settings for headset

4) The “Create” will fill the Hex/ASCII line of the Simply Blue Commander with the string necessary to send to the LMX9820A. Afterwards just close the dialog with the “Close” Button..

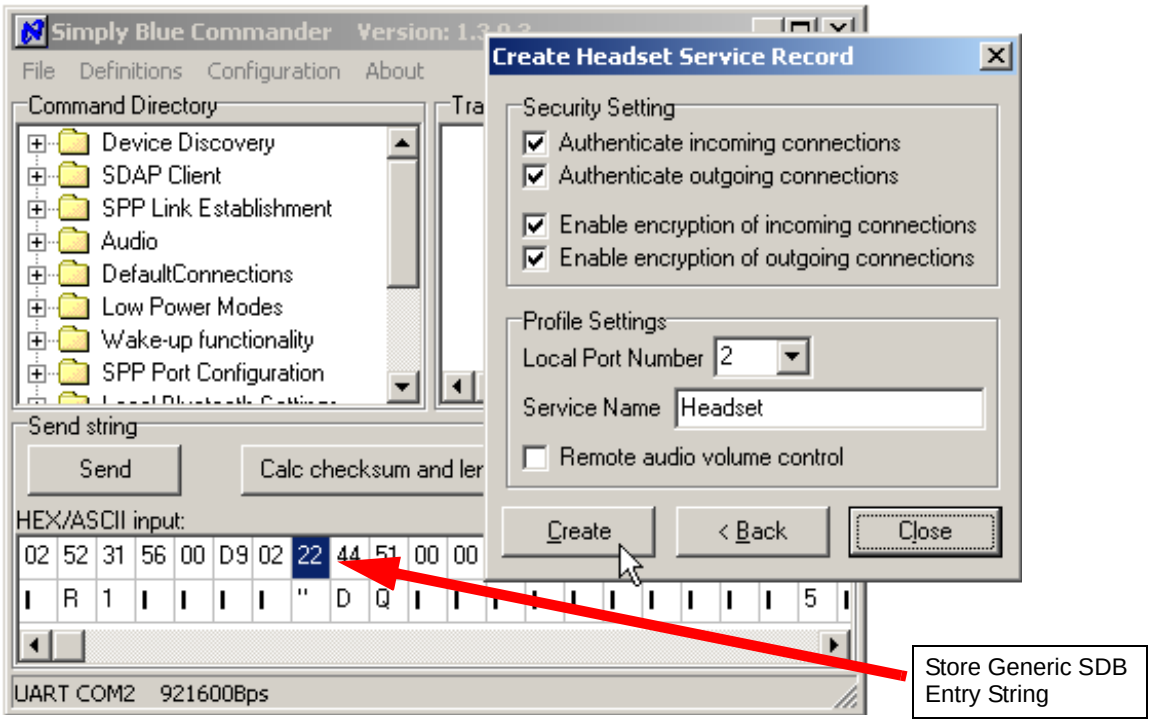


Figure 4-13. Creating the command string for the headset entry

5) The string can be sent directly to the board or stored as command within the directory. Once sent it can also be copied out of the log entry and copied into the development code when switching of the interpretation option.

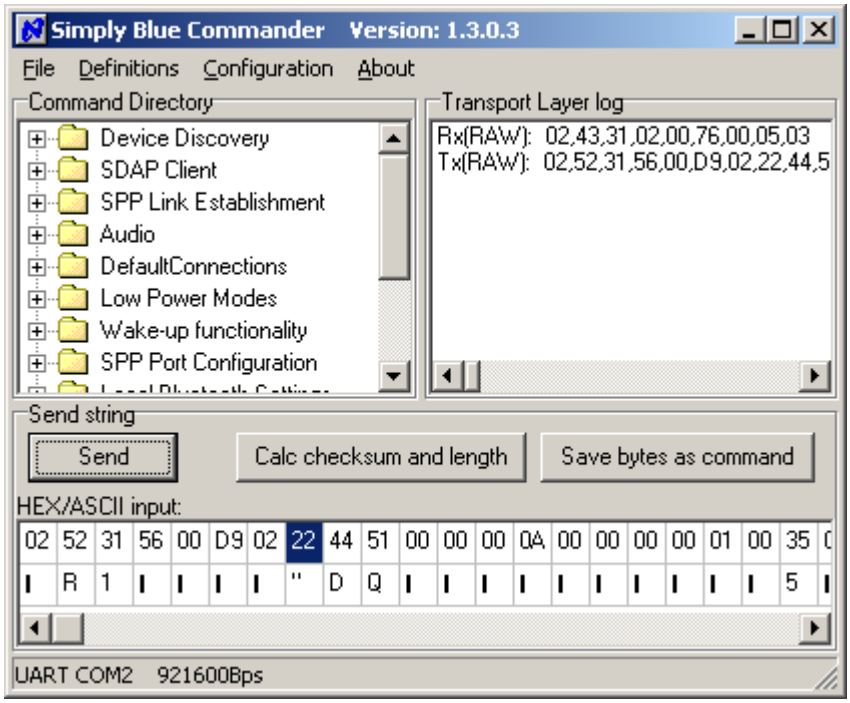


Figure 4-14. Send the command to the LMX9820A

Table 4-14 shows the command string beeing the result out of the steps just described.

Table 4-14. Adding the headset service to the Service Database

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,31,56,00,D9,**02**,**22**,**44**,51,00,00,00,0A,00,00,00,00,01,00,35,06,19,08,11,19,03,12,04,00,35,0C,35,03,19,00,01,35,05,19,03,00,08,02,05,00,35,03,19,02,10,06,00,35,09,09,6E,65,09,6A,00,09,00,01,09,00,35,08,35,06,19,08,11,09,00,01,00,01,25,08,48,65,61,64,73,65,74,00,02,03,28,00,03	Tx: Cmd: Store SDP Record, Local Port: **02**, Authentication: **22**, Encryption: **44**, SdpRecord: 00000A000000001003506190811190312040035 0C350319000135051903000802050035031902100600350909 6E65096A0009000109003508350619081109000100125084865616473657400020328 00
00 / 01	RX	Confirm	02,43,31,02,00,76,00,**01**,03	Rx: Event: Store SDP Record, Status: 00, Identifier: **01**

After this command the Service database has the following content:

Table 4-15. Service Database for COM1 and DUN

Index	Enabled	Local RFComm Port	Type	Name	Authentication	Encryption	Profile Specifics
00	1	01	SPP	COM1	0x02	0x04	none
01	1	02	HS	Headset	0x22	0x44	none

4.4.2.3 Adapt RFCOMM ports to open

For a simpler differentiation on the host the services have been set to two separate RFCOMM port. In order to make both ports available for connection, the Set Ports To Open (5.2.12.1) command should be used to make the appropriate configuration. .

Table 4-16. Set RFCOMM ports to open

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,22,04,00,78,03,00,00,00,03	Tx: Cmd: Set Ports To Open, Ports: **03000000**
00 / 01	RX	Confirm	02,43,22,01,00,66,00,03	Rx: Event: Set Ports To Open, Status: 00

4.4.2.4 Store Class of Device

During the Inquiry procedure the searching device will already receive the class of device. Please see also Section 3.1.2.3 "Class of Device" on page 37 for more information on that parameter. In order to be recognized it is beneficial to already indicate the audio capability within the class of device. The following table shows the command how to set the “Class of Device” using the Store Class of Device (5.2.14.5) command. The class is set to 0x220404, which reflects the class for a headset..

Table 4-17. Store Class of Device

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,28,03,00,7D,**04**,**04**,**22**,03	Tx: Cmd: Store Class of Device, Class of Device: **040422**
00 / 01	RX	Confirm	02,43,22,01,00,66,00,03	Rx: Event: Store Class of Device, Status: 00

4.4.2.5 Connection handling as headset

The Headset profile is based on some specific AT commands also listed in Section 5.4.2 on page 186.

In a headset application the link will either be initiated from the HSAG or the HS itself. In both cases the HSAG will control the SCO link establishment. The HS will only react on incoming AT commands and maybe send the request establishment or release.

The LMX9820A accepts incoming SCO links or SPP links automatically. It just indicates the successful establishment to the host. Therefore, for the HS implementation no further LMX9820A specific commands besides the already described configuration are required. The headset profile itself is controlled by a few AT commands.

The following chapters give the generic guideline of how to act as headset within the different scenarios.

4.4.2.5.1 Incoming audio connection

Figure 4-15 on page 81shows the flow diagram of a headset link initiated by the HSAG as defined in [3].

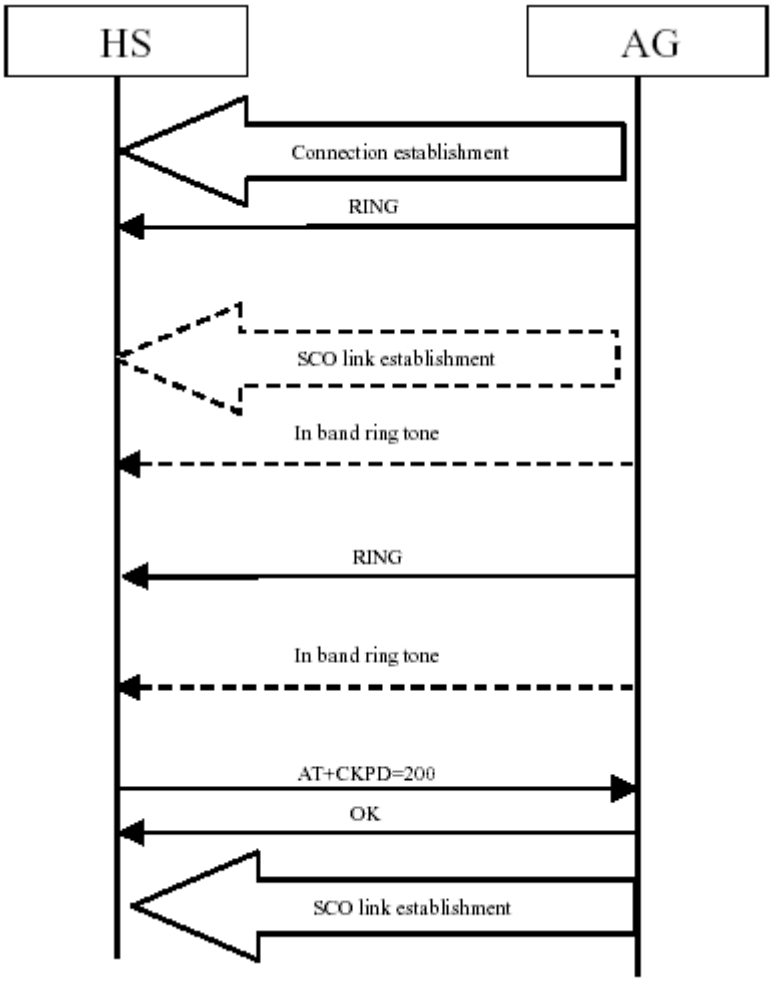


Figure 4-15. "Incoming audio connection establishment" as defined in bluetooth specification

As the specification figure already shows, the HS only needs to accept the incoming SPP and audio link and to indicate a button press by sending the "AT+CKPD=200" command. In addition the HS application needs to be able to recognize the RING command and the OK result code.

Please see Figure 4-16 which shows the headset part of the profile for the incoming audio connection 'translated' to the LMX9820A interface.

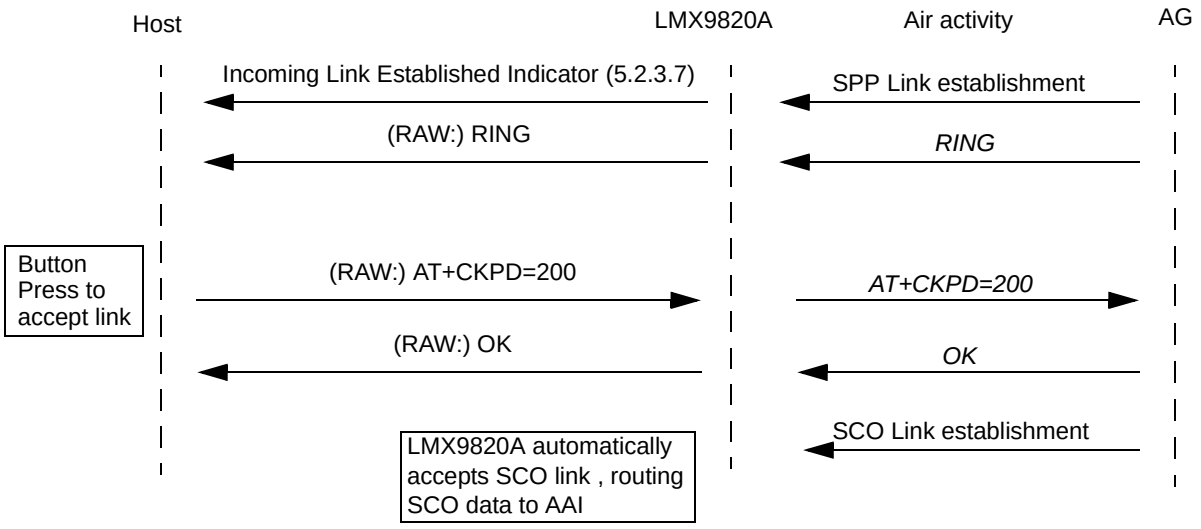


Figure 4-16. Incoming audio connection with LMX9820A as HS

Figure 4-16 shows the flow based on automatic mode, in which the LMX9820A will automatically switch to transparent mode on the UART. Please see the following table for the detailed description of the UART traffic between host and LMX9820A.

Table 4-18. Incoming audio connection with LMX9820A as HS

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
00 / 01	RX	Indicator	02,69,50,07,00,C0,B0,CF,22,17,00, 08,00,03	Rx: Event: Incoming Link Established, BdAddr:123456789ABC, Local Port: 02
00 / 01 / 02 / 03	RX	RAW Data	0D,0A,52,49,4E,47,0D,0A	Rx(RAW): 0D,0A,52,49,4E,47,0D,0A *(RING)*
00 / 01 / 02 / 03	TX	RAW Data	41,54,2B,43,4B,50,44,3D,32,30,30, 0D	Tx(RAW): 41,54,2B,43,4B,50,44,3D,32,30,30,0D *(AT+CKPD=200)*
00 / 01 / 02 / 03	RX	RAW Data	0D,0A,4F,4B,0D,0A	Rx(RAW): 0D,0A,4F,4B,0D,0A *(OK)*

After the final OK is received the audio link is established.

4.4.2.5.2 Headset initiated connection establishment

The Headset profile also defines the scenario in which the headset initiates the link the HSAG. See Figure 4-17 on page 83 for the profile specification of the Bluetooth SIG.

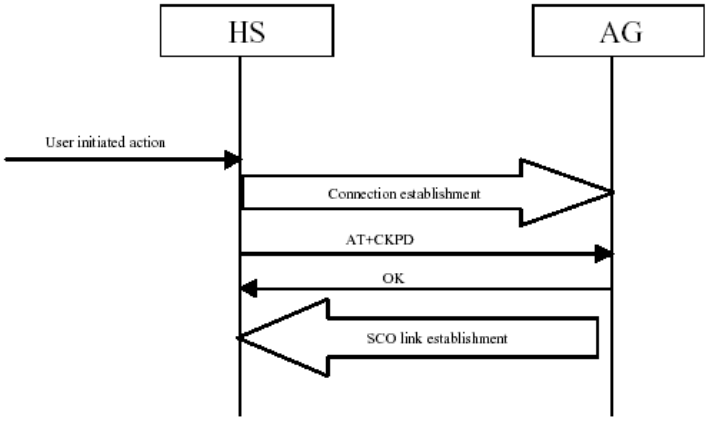


Figure 4-17. Outgoing audio connection establishment as defined by the Bluetooth SIG

As Figure 4-17 indicates, the headset basically only requests a SPP link and sends the AT+CKPD command. The rest is controlled by the HSAG. Therefore the link establishment is limited to the SPP link establishment and AT commands.

In order to establish a link to the HSAG, the headset needs to know the RFCOMM port number it has to connect on the remote device. Usually, these information are requested on first time of connection only, so only the SPP Link establish command will be necessary for the establishment.

Table 4-19 gives an indication of the necessary SDAP request. The most important parameter is the UUID for the Headset Audio Gateway, which is 0x1112.

Figure 4-18 and Table 4-20 show the final profile flow necessary to establish a connection to the HSAG with the LMX9820A.

Table 4-19. Requesting the RFCOMM port number for HSAG from the remote device

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,32,06,00,8A,12,34,56,78,9A, BC,03	Tx: Cmd: SDAP Connect, BdAddr: 123456789ABC
00	RX	Indicator	02,69,50,07,00,C0,12,34,56,78,9A, BC,00,03	Rx: Event: ACL Established, BdAddr: 123456789ABC, Status: 00
00 / 01	RX	Confirm	02,43,32,01,00,76,00,03	Rx: Event: SDAP Connect, Status: 00
	TX	Request	02,52,35,02,00,89,12,11,03	Tx: Cmd: Service Browse, Browse Group ID: 0811
00 / 01	RX	Confirm	02,43,35,16,00,8E,00,01,02,10,**12**, **11**,**03**,0E,56,6F,69,63,65,20,67,61, 74,65,77,61,79,00,03	Rx: Event: Service Browse, Status: 00, Browse Group ID: 0210, Service ID: **1211**, PortNo: **03**, Service Name: Voice gateway.
	TX	Request	02,52,33,00,00,85,03	Tx: Cmd: SDAP Disconnect
00 / 01	RX	Confirm	02,43,33,01,00,77,00,03	Rx: Event: SDAP Disconnect, Status: 00
00	RX	Indicator	02,69,51,07,00,C1,12,34,56,78,9A, BC,13,03	Rx: Event: ACL Terminated, BdAddr: 123456789ABC, Reason: 13

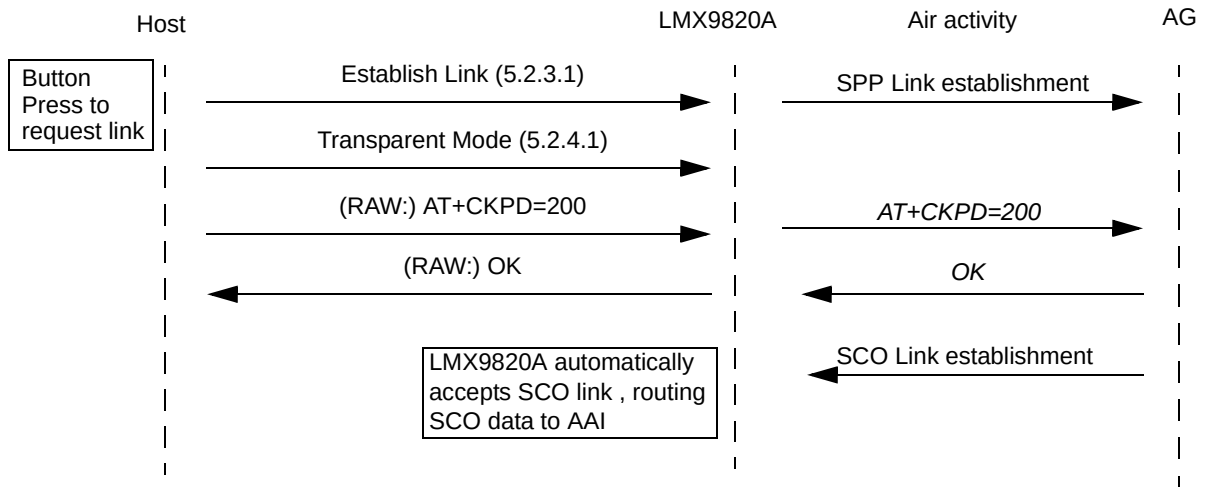


Figure 4-18. Outgoing link establishment from the LMX9820A as HS

Table 4-20. Outgoing link establishment from the LMX9820A as HS

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,0A,08,00,64,01,12,34,56,78, 9A,BC,**03**,03	Tx: Cmd: Establish Link, Local Port: 01, BdAddr: 123456789ABC, Remote Port Number: **03**
00 / 01	RX	Confirm	02,43,0A,02,00,4F,00,01,03	Rx: Event: Establish Link, Status: 00, Local Port: 01
00	RX	Indicator	02,69,50,07,00,C0,12,34,56,78,9A, BC,00,03	Rx: Event: ACL Established, BdAddr: 123456789ABC, Status: 00
00 / 01	RX	Indicator	02,69,0B,09,00,7D,00,12,34,56,78, 9A,BC,01,**03**,03	Rx: Event: Link Established, Status: 00, BdAddr: 123456789ABC, Local Port: 01, Remote Port Number: **03**
00 / 01	RX	Indicator	02,69,3E,04,00,AB,01,8C,00,00,03	Rx: Event: Port Status Changed, Local Port: 01, PortStatus: 8C, Break Length: 0000
	TX	Request	02,52,11,01,00,64,01,03	Tx: Cmd: Transparent Mode, Local Port: 01
00 / 01	RX	Confirm	02,43,11,02,00,56,00,01,03	Rx: Event: Transparent Mode, Status: 00, Local Port: 01
	TX	RAW Data	41,54,2B,43,4B,50,44,3D,32,30,30, 0D	Tx(RAW): 41,54,2B,43,4B,50,44,3D,32,30,30,0D *(AT+CKPD=200)*
00 / 01 / 02 / 03	RX	RAW Data	0D,0A,4F,4B,0D,0A	Rx(RAW): 0D,0A,4F,4B,0D,0A *(OK)*

4.4.2.5.3 Audio connection release from the HS

The connection release is based on the same command as the connection confirmation, AT+CKPD=200. In case the user wants to release the link from the HS to the HSAG, it presses the same button again, which will send the same AT command. The HSAG will confirm by an OK and release the link.

The LMX9820A confirms the released link by the standard SPP Link Released Indicator (5.2.3.6). See following the Bluetooth specification description for the realization with LMX9820A. In case the HSAG releases the link the LMX9820A will send the same event.

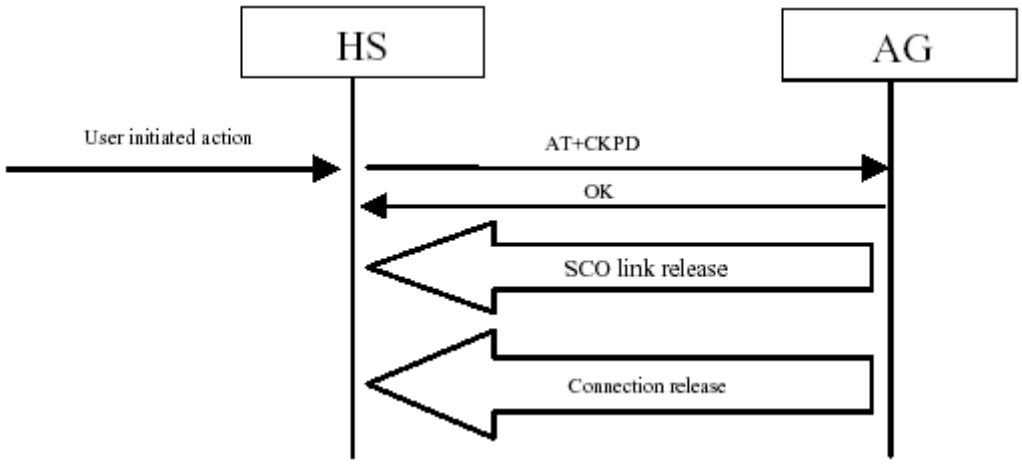


Figure 4-19. Audio link release by the HS

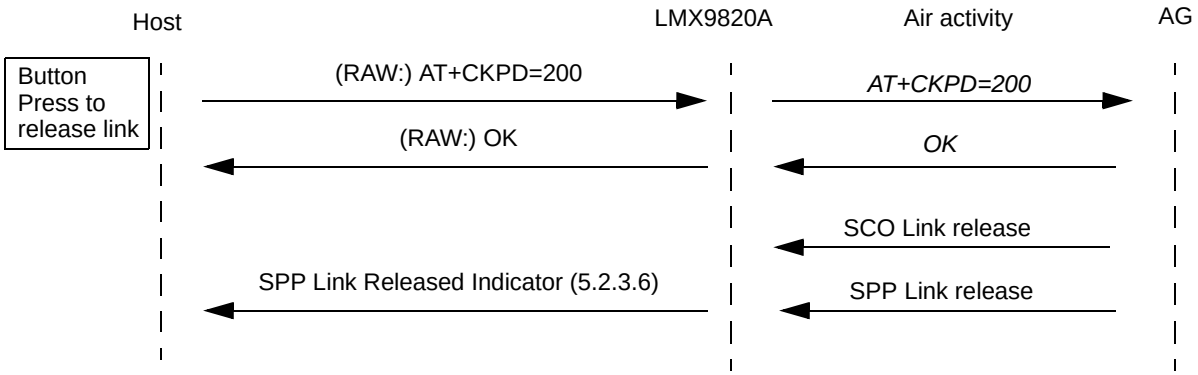


Figure 4-20. Audio link release from the LMX9820A as HS

Table 4-21. Audio link release from the LMX9820A as HS

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	RAW Data	41,54,2B,43,4B,50,44,3D,32,30,30, 0D	Tx(RAW): 41,54,2B,43,4B,50,44,3D,32,30,30,0D
00 / 01 / 02 / 03	RX	RAW Data	0D,0A,4F,4B,0D,0A	Rx(RAW): 0D,0A,4F,4B,0D,0A
00 / 01 / 02	RX	UART BREAK		Rx(RAW): 00
00 / 01	RX	Indicator	02,69,11,02,00,7C,01,00,03	Rx: Event: Transparent Mode, Local Port: 01, Mode: 00
00 / 01	RX	Indicator	02,69,0E,02,00,79,01,01,03	Rx: Event: Link Released, Reason: 01, Local Port: 01
00	RX	Indicator	02,69,51,07,00,C1,12,34,56,78,9A, BC,13,03	Rx: Event: ACL Terminated, BdAddr: 123456789ABC, Reason: 13

4.4.2.5.4 Audio connection release from the HSAG

In case the HSAG releases the audio link, no further AT commands are required. The HSAG just releases the SPP and SCO link. The host will be notified by the standard SPP Link Released Indicator (5.2.3.6) and SCO Link Released Indicator (5.2.6.4).

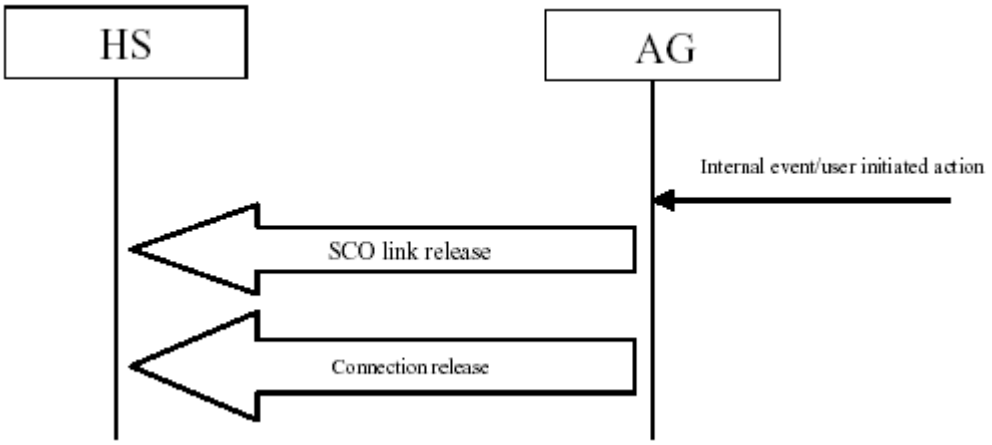


Figure 4-21. Audio link release by the HSAG

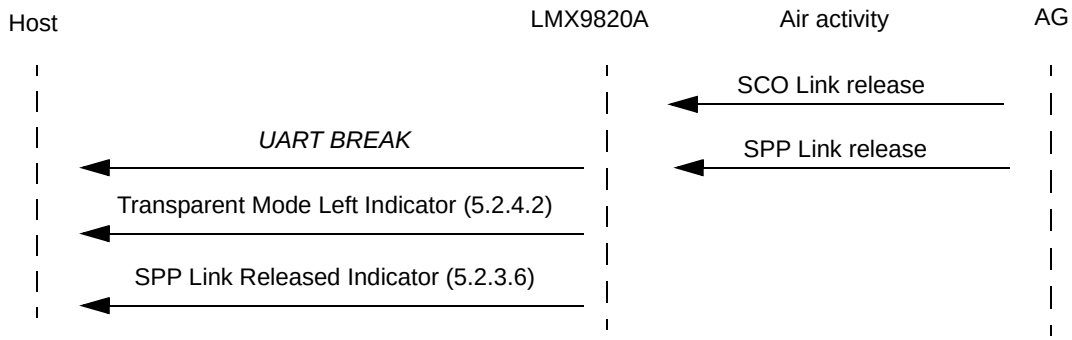


Figure 4-22. Audio link release by the HSAG

Table 4-22. Audio link release from the LMX9820A as HS

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
00 / 01 / 02	RX	UART BREAK		Rx(RAW): 00
00 / 01	RX	Indicator	02,69,11,02,00,7C,01,00,03	Rx: Event: Transparent Mode, Local Port: 01, Mode: 00
00 / 01	RX	Indicator	02,69,0E,02,00,79,01,01,03	Rx: Event: Link Released, Reason: 01, Local Port: 01
00	RX	Indicator	02,69,51,07,00,C1,12,34,56,78,9A, BC,13,03	Rx: Event: ACL Terminated, BdAddr: 123456789ABC, Reason: 13

4.4.3 Acting as Audio Gateway

The audio gateway is defined as a device, handling audio data and using the HS as analog front end for both for input and output. Typical devices acting as Audio Gateways are cellular phones and personal computer. The LMX9820A can be used in an audio gateway by creating the required data link to the headset and routing all data from the Advanced Audio Interface (AAI) to the bluetooth link. The audio interface needs to be connected to the host PCM interface. Timing and electrical specification for the interface are derived out of the driver configuration set in NVS. Bitclock and frameclock settings for different codec options are described in [1]. Please see Figure 4-9 "Headset Data and Audio stream" on page 76 how to implement the LMX9820A in a audio gateway application.

The Audio Gateway (AG) controls the interoperation with the headset in terms of establishing and releasing the required SCO link. While the SPP link establishment can be initiated from both sides, the SCO link establishment will always be initiated by the HSAG. The LMX9820A offers dedicated commands to control the audio link.

In order to be able to create a link to a headset, it is required to create the service database record and to configure the default audio settings for the PCM codec driver.

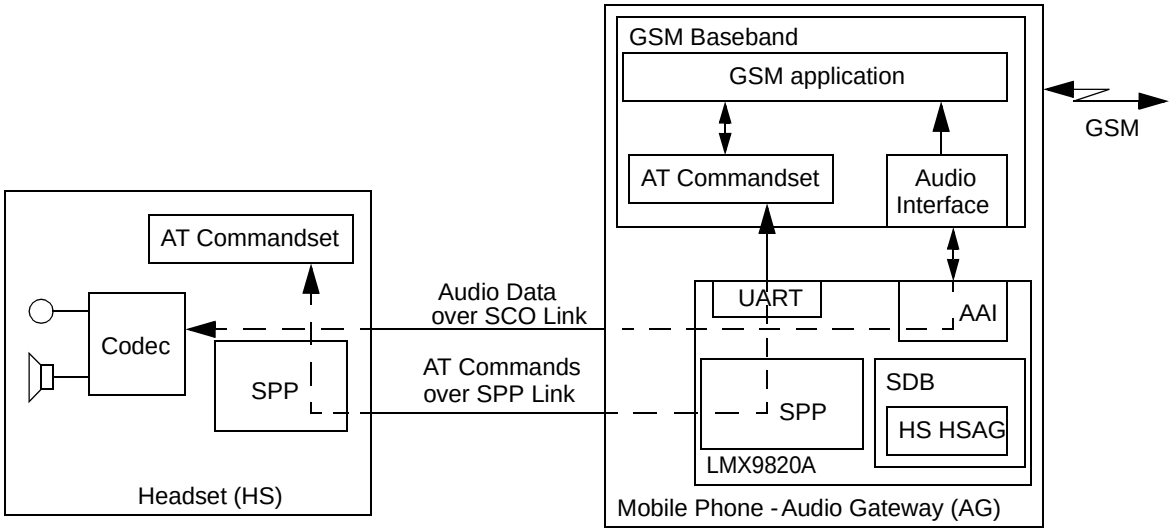


Figure 4-23. Audio Gateway Data and Audio stream using LMX9820A in the HSAG

4.4.3.1 Configuring the audio path

In general the LMX9820A can be seen as gateway between the UART interface and the bluetooth link. The audio link is one additional interface for the LMX9820A to be routed. Please see also Figure 3-13 on page 59 for the two routing options.

The LMX9820A supports different configurations for the PCM codec interface and the air interface. The bluetooth specification defines the following formats to be used over the bluetooth link:

- CVSD (default)
- µ-Law
- A-Law

As indicated, the default and mostly used coding format used over a bluetooth link is CVSD. Since the headset most likely will always use the same settings, it is recommended to set the parameters to a default in the NVS, using the Set Default Audio Settings (5.2.18.1) command. Bitclock and frameclock information for the different codecs are listed in [1].

Table 4-23. Set Default Audio Settings to Motorola codec and CVSD

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,5B,02,00,AF,**02**,**01**,03	Tx: Cmd: Set Default Audio Settings, Codec Type: **02**, Air Format: **01**
00 / 01	RX	Confirm	02,43,5B,01,00,9F,00,03	Rx: Event: Set Default Audio Settings, Status: 00

4.4.3.2 Configuring the Service Database for Audio Gateway

In order to offer an additional HSAG service to other devices a new entry has been made into the service database. For this the Store Generic SDP Record (5.2.13.9) needs to be used. This command is a generic command to generate any possible profile within the device. To be able to do so, significant knowledge of the internal SDB structure would be required. In order to use this command, the “Simply Blue Commander” software needs to be used, to create this command.

The generation of profile entries is supported in Simply Blue Commander versions later than 1.3.0.3. Please follow the following figures to create a headset service database entry:

1) Select “Definitions/Create Service Record” within the Simply Blue Commander menu.

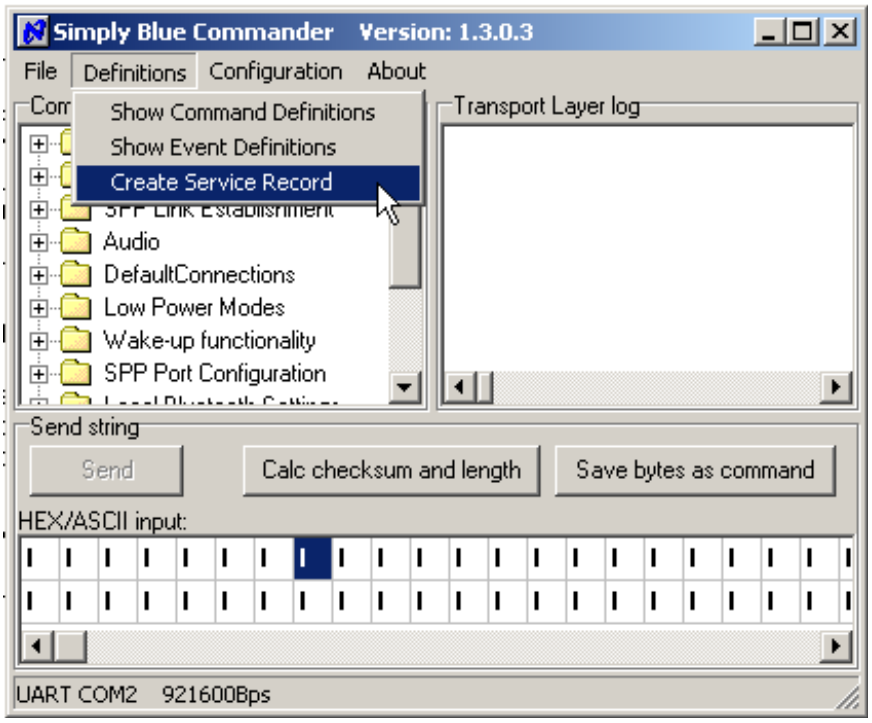


Figure 4-24. Opening “Create Service Record” Dialogbox

2) Select the service record required, in this case “Headset Audio Gateway”

Figure 4-25. Selecting the profile

3) Select the settings desired for the HSAG profile and choose the correct RFCOMM port. As this example device shall still be able to accept a second standard data links as well, RFCOMM port 2 should be used. Finally confirm the dialog with pressing “Create”.

Figure 4-26. Configuring the profile settings for HSAG

4) The “Create” will fill the Hex/ASCII line of the Simply Blue Commander with the string necessary to send to the LMX9820A. Afterwards just close the dialog with the “Close” Button..

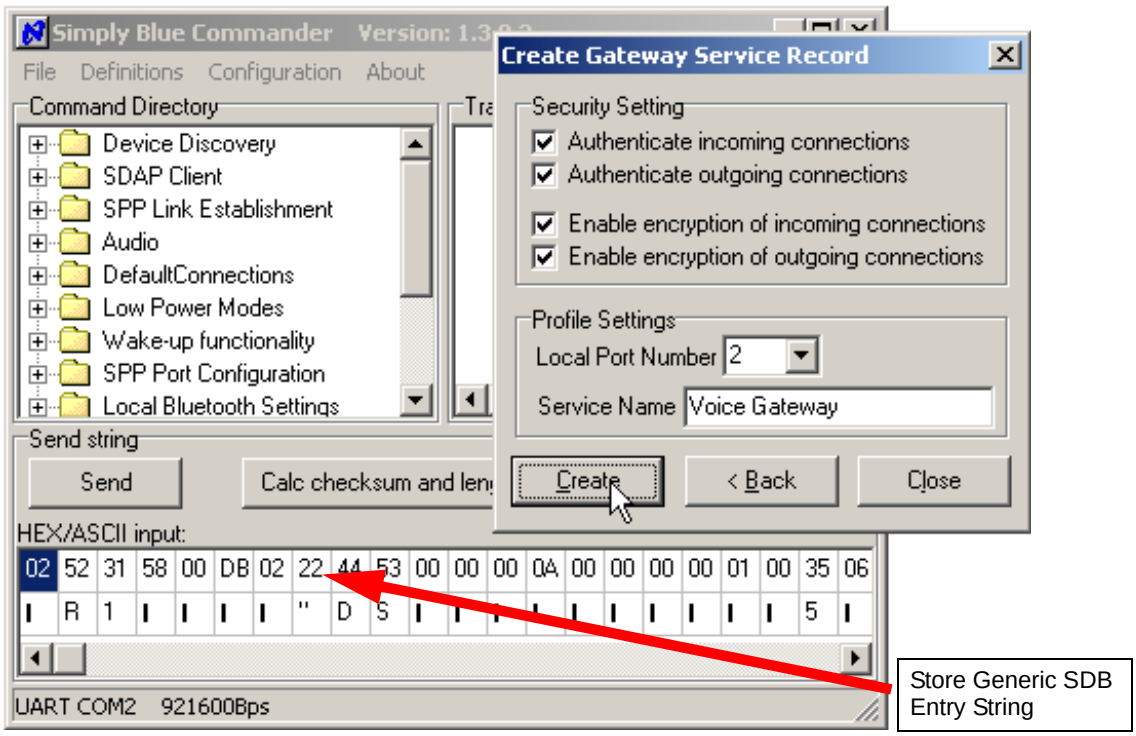


Figure 4-27. Creating the command string for the audio gateway entry

5) The string can be sent directly to the board or stored as command within the directory. Once sent it can also be copied out of the log entry and copied into the development code when switching of the interpretation option.

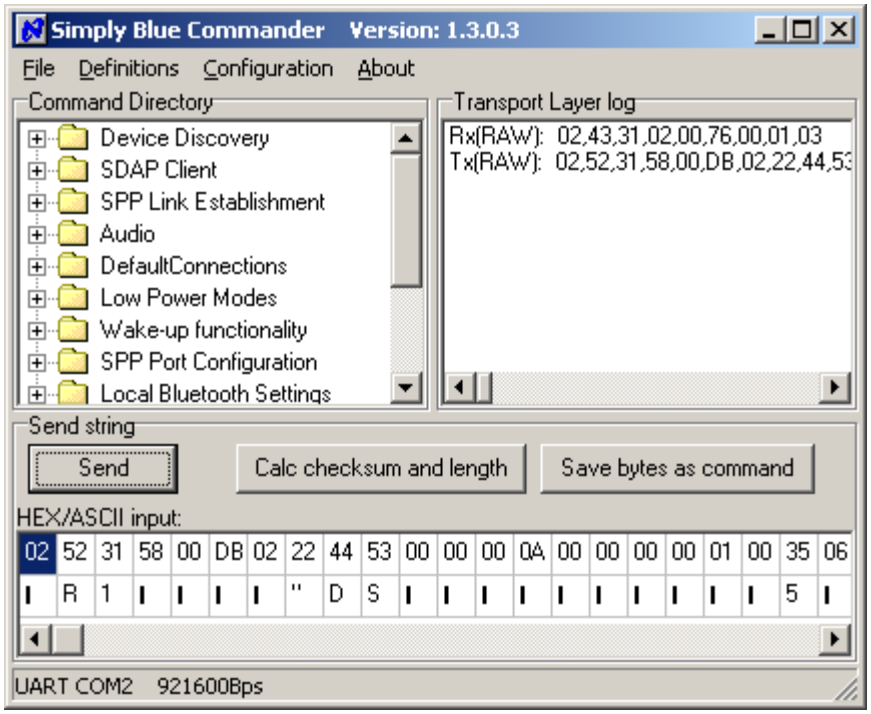


Figure 4-28. Send the command to the LMX9820A

Table 4-14 shows the command string beeing the result out of the steps just described.

Table 4-24. Adding the HSAG service to the Service Database

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,31,58,00,DB,**02**,**22**,**44**,53,00,00,00,0A,00,00,00,00,01,00,35,06,19,12,11,19,03,12,04,00,35,0C,35,03,19,00,01,35,05,19,03,00,08,02,05,00,35,03,19,02,10,06,00,35,09,09,6E,65,09,6A,00,09,00,01,09,00,35,08,35,06,19,08,11,09,00,01,00,01,25,0E,56,6F,69,63,65,20,47,61,74,65,77,61,79,00,03	Tx: Cmd: Store SDP Record, Local Port: **02**, Authentication: **22**, Encryption: **44**, SdpRecord: 00000A000000001003506191211190312040035 0C3503190001350519030008020500350319021006003509096E65096A0009000109003508350619081109000100 01250E566F6963652047617465776179 00
00 / 01	RX	Confirm	02,43,31,02,00,76,00,**01**,03	Rx: Event: Store SDP Record, Status: 00, Identifier: **01**

After this command the Service database has the following content:

Table 4-25. Service Database for COM1 and HSAG

Index	Enabled	Local RFComm Port	Type	Name	Authentication	Encryption	Profile Specifics
00	1	01	SPP	COM1	0x02	0x04	none
01	1	02	HSAG	Voice Gateway	0x22	0x44	none

4.4.3.3 Adapt RFCOMM ports to open

For a simpler differentiation on the host the services have been set to two separate RFCOMM port. In order to make both ports available for connection, the Set Ports To Open (5.2.12.1) command should be used to make the appropriate configuration. .

Table 4-26. Set RFCOMM ports to open

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,22,04,00,78,03,00,00,00,03	Tx: Cmd: Set Ports To Open, Ports: **03000000**
00 / 01	RX	Confirm	02,43,22,01,00,66,00,03	Rx: Event: Set Ports To Open, Status: 00

4.4.3.4 Store Class of Device

During the Inquiry procedure the searching device will receive the class of device. Please see also Section 3.1.2.3 "Class of Device" on page 37 for more information on that parameter. In order to be recognized as audio device it is beneficial to indicate the audio capability within the class of device. The following table shows the command how to set the “Class of Device” using the Store Class of Device (5.2.14.5) command. The class is set to 522204, which reflects the class for a headset..

Table 4-27. Store Class of Device

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,28,03,00,7D,**04**,**22,52**,03	Tx: Cmd: Store Class of Device, Class of Device: **042252**
00 / 01	RX	Confirm	02,43,22,01,00,66,00,03	Rx: Event: Store Class of Device, Status: 00

4.4.3.5 Connection handling as audio gateway

The Headset profile is based on some specific AT commands also listed in Section 5.4.2 “Headset Profile” on page 186.

In a headset application the link will either be initiated from the HSAG or the HS itself. In both cases the HSAG will control the SCO link establishment. The HS will only react on incoming AT commands and maybe send the request establishment or release.

The LMX9820A offers specific commands to establish and release SPP and SCO links. The first connection establishment is based on the standard SPP link. Once this is established the HSAG indicates the incoming call by sending the RING At command over the SPP link. Once the HS responds with the appropriate AT command, the HSAG needs to establish the SCO connection. In some implementations it is also possible to establish the SCO right after the SPP link to support inband ringtones.

The following chapters give the generic guideline of how to act as HSAG within the different scenarios.

4.4.3.5.1 Requesting the RFCOMM port from the Headset

Table xx gives an indication of the necessary SDAP request. The most important parameter is the UUID for the Headset Audio Gateway, which is 0x0811.

Table 4-28. Requesting the RFCOMM port number for HSAG from the remote device

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,32,06,00,8A,12,34,56,78,9A, BC,03	Tx: Cmd: SDAP Connect, BdAddr: 123456789ABC
00	RX	Indicator	02,69,50,07,00,C0,12,34,56,78,9A, BC,00,03	Rx: Event: ACL Established, BdAddr: 123456789ABC, Status: 00
00 / 01	RX	Confirm	02,43,32,01,00,76,00,03	Rx: Event: SDAP Connect, Status: 00
	TX	Request	02,52,35,02,00,89,12,11,03	Tx: Cmd: Service Browse, Browse Group ID: 0811
00 / 01	RX	Confirm	02,43,35,16,00,8E,00,01,02,10,**08**, **11**,**07**,0E,56,6F,69,63,65,20,67,61, 74,65,77,61,79,00,03	Rx: Event: Service Browse, Status: 00, Browse Group ID: 0210, Service ID: **0811**, PortNo: **07**, Service Name: Voice gateway.
	TX	Request	02,52,33,00,00,85,03	Tx: Cmd: SDAP Disconnect
00 / 01	RX	Confirm	02,43,33,01,00,77,00,03	Rx: Event: SDAP Disconnect, Status: 00
00	RX	Indicator	02,69,51,07,00,C1,12,34,56,78,9A, BC,13,03	Rx: Event: ACL Terminated, BdAddr: 123456789ABC, Reason: 13

4.4.3.5.2 Outgoing audio connection

Figure 4-15 on page 81 shows the flow diagram of a HSAG link initiated by the HSAG as defined in [3].

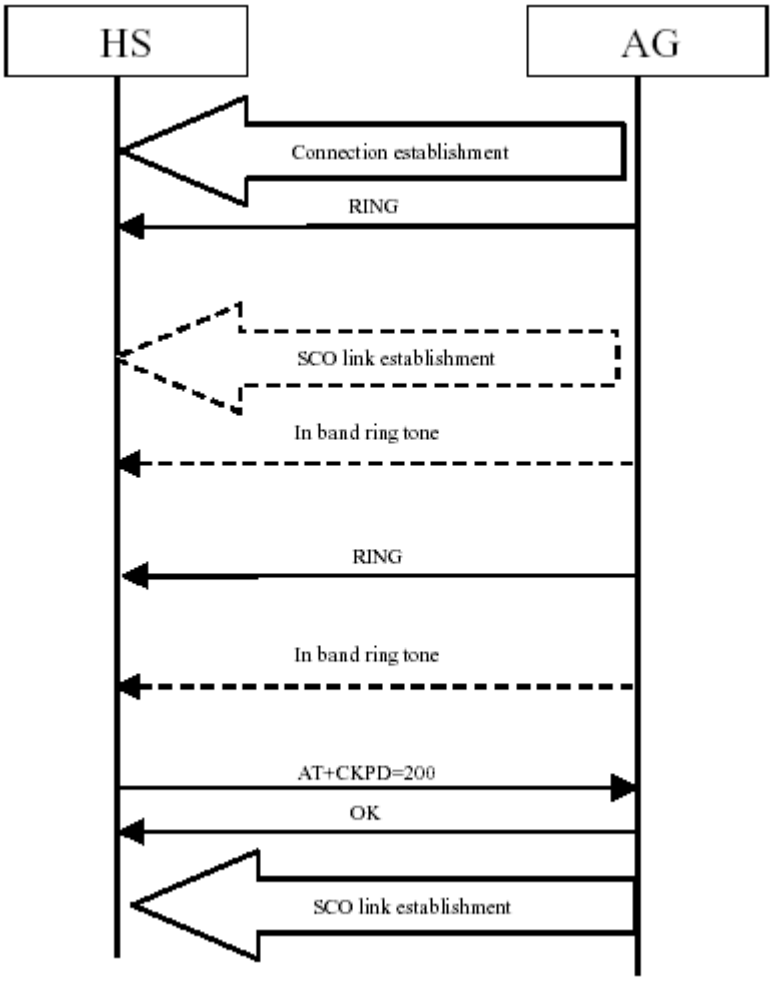


Figure 4-29. "AG initiated audio connection establishment" as defined in [3]

As the specification figure shows, the link is initiated by the HSAG, e.g. in case of an incoming call on a mobile phone. The HSAG will send the "RING" command to force an audio or visual signalling on the headset. The HSAG keeps on ringing until the HS user accepts the incoming link by pressing a button. On this the HS application will send the "AT+CKPD=200" command. The HSAG needs to confirm the successful reception of the command with the "OK" and finally sends the SCO establishment command.

Please see Figure 4-16 which shows the HSAG part of the profile for the outgoing audio connection 'translated' to the LMX9820A command interface.

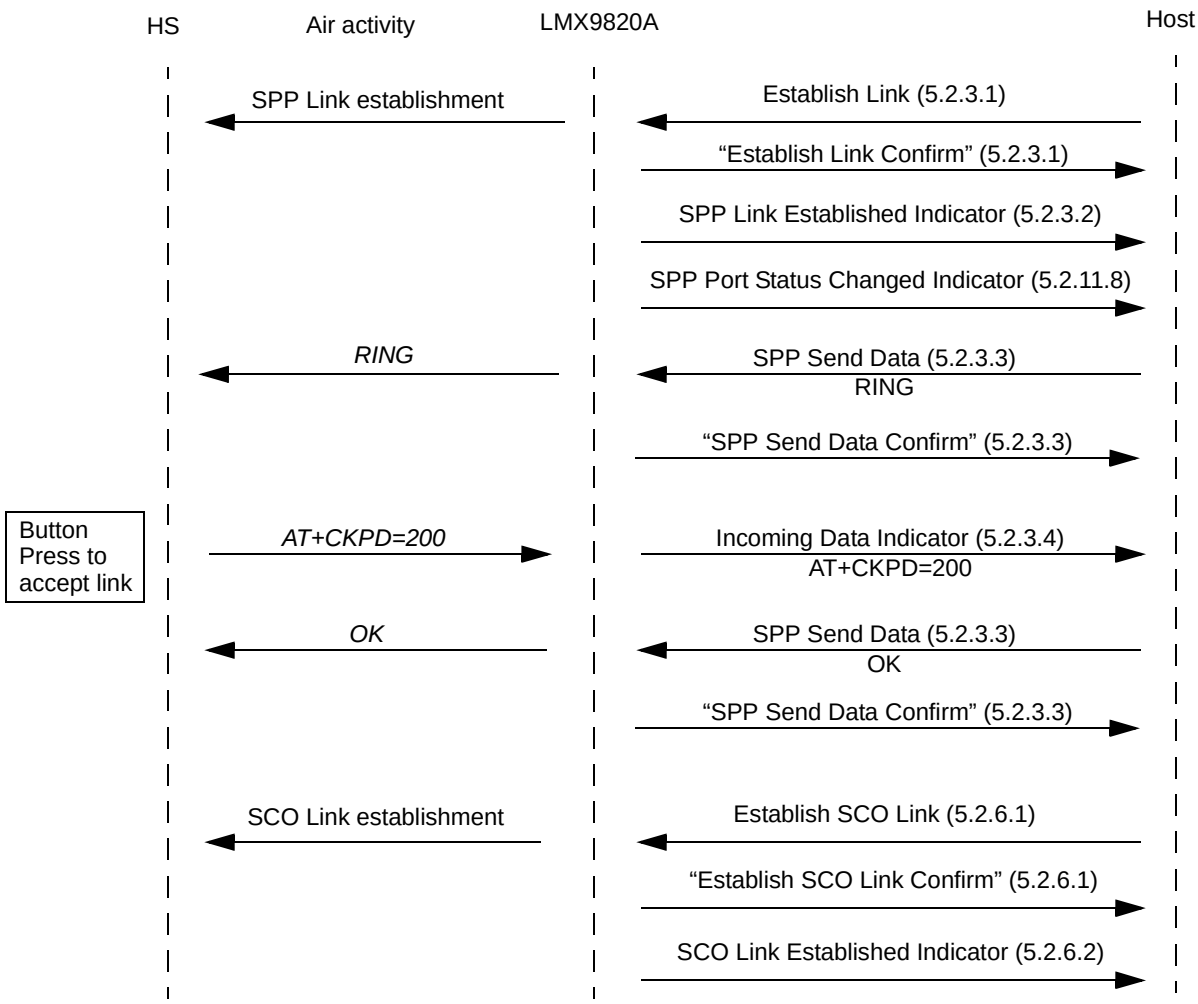


Figure 4-30. Outgoing audio connection with LMX9820A as HSAG

Please see the following table for the detailed description of the UART traffic between host and LMX9820A.

Table 4-29. Outgoing audio connection with LMX9820A as HSAG

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,0A,08,00,64,**02**,12,34,56,78, 9A.BC,**07**,03	Tx: Cmd: Establish Link, Local Port: **02**, BdAddr: 123456789ABC, Remote Port Number: **07**
00 / 01	RX	Confirm	02,43,0A,02,00,4F,00,**02**,03	Rx: Event: Establish Link, Status: 00, Local Port: **02**
00	RX	Indicator	02,69,50,07,00,C0,12,34,56,78,9A. BC,00,03	Rx: Event: ACL Established, BdAddr: 123456789ABC, Status: 00
00 / 01	RX	Indicator	02,69,0B,09,00,7D,00,12,34,56,78, 9A.BC,**02**,**07**,03	Rx: Event: Link Established, Status: 00, BdAddr: 123456789ABC, Local Port: **02**, Remote Port Number: **07**
00 / 01	RX	Indicator	02,69,3E,04,00,AB,02,8C,00,00,03	Rx: Event: Port Status Changed, Local Port: 02, PortStatus: 8C, Break Length: 0000
	TX	Request	02,52,0F,0B,00,6C,02,08,00,0D,0A ,52,49,4E,47,0D,0A,03	Tx: Cmd: Send Data, Local Port: 02, Payload Data: 0D0A52494E470D0A *(RING)*

Table 4-29. Outgoing audio connection with LMX9820A as HSAG

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
00 / 01	RX	Confirm	02,43,0F,02,00,54,00,02,03	Rx: Event: Send Data, Status: 00, Local Port: 02
00 / 01	RX	Indicator	02,69,10,0F,00,88,02,0C,00,41,54, 2B,43,4B,50,44,3D,32,30,30,0D,03	Rx: Event: Incoming Data, Local Port: 02, Received Data: 41542B434B50443D3230300D *(AT+CKPD=200)*
	TX	Request	02,52,0F,09,00,6A,02,06,00,0D,0A, 4F,4B,0D,0A,03	Tx: Cmd: Send Data, Local Port: 02, Payload Data: 0D0A4F4B0D0A *(OK)*
00 / 01	RX	Confirm	02,43,0F,02,00,54,00,02,03	Rx: Event: Send Data, Status: 00, Local Port: 02
	TX	Request	02,52,5D,08,00,B7,12,34,56,78,9A. BC,80,00,03	Tx: Cmd: Establish SCO Link, BdAddr: 123456789ABC, Packet Type: 8000
00 / 01	RX	Confirm	02,43,5D,07,00,A7,00,12,34,56,78, 9A.BC,03	Rx: Event: Establish SCO Link, Status: 00, BdAddr: 123456789ABC
00 / 01	RX	Indicator	02,69,5D,07,00,CD,00,12,34,56,78 ,9A.BC,03	Rx: Event: SCO Link Established, Status: 00, BdAddr: 123456789ABC

In some applications, the headset might send additional commands for volume control or status. Please refer to Section 5.4.2 on page 186 for the complete list of AT commands required by the headset profile.

4.4.3.5.3 Incoming audio connection from Headset

The Headset profile also defines the scenario in which the headset initiates the link the HSAG.

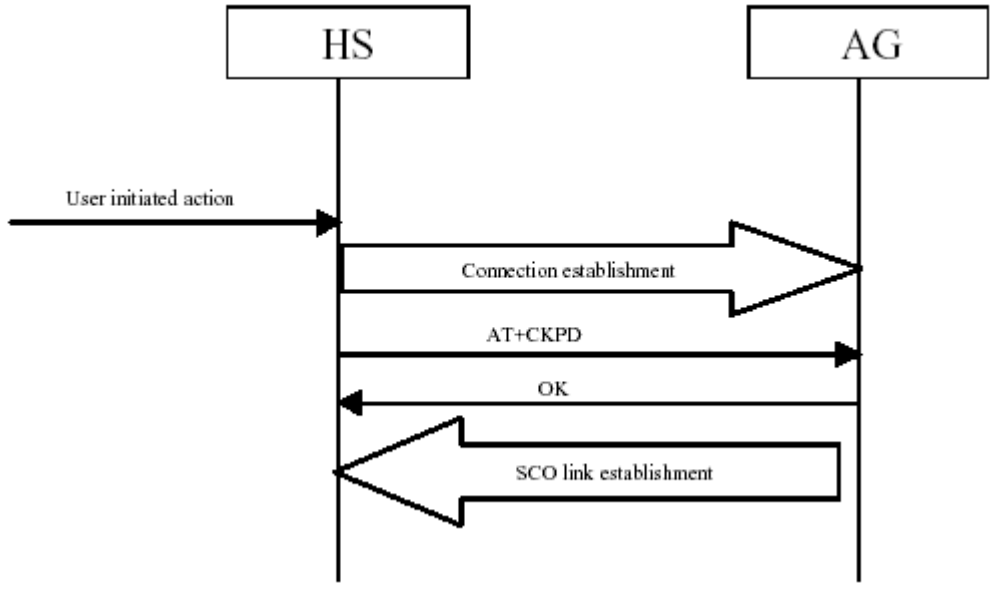


Figure 4-31. Incoming audio connection on HSAG as defined in [3]

As Figure 4-17 indicates, the headset requests a SPP link and sends the AT+CKPD=200 command. The HSAG needs to accept the incoming link and afterwards establish the SCO as already described for an outgoing link.

The headset will contact the HSAG on the RFCOMM port, the HSAG has been assigned to.

Figure 4-32 and Table 4-30 show the profile flow necessary to accept an incoming link from a headset using the LMX9820A in an HSAG application. The LMX9820A in this example is using the default setting, meaning configured for automatic mode. Because of this, it will switch to transparent mode as soon as the incoming SPP link has been established. Therefore a UART BREAK is sent before the SCO links can be sent.

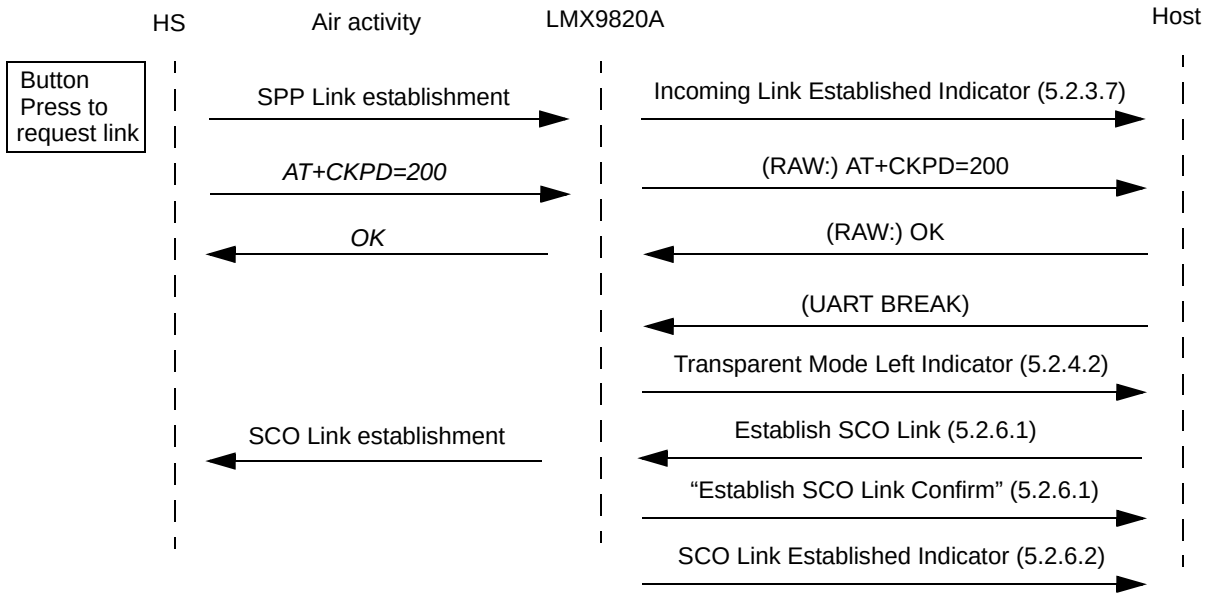


Figure 4-32. Incoming HS connection as HSAG

Table 4-30. Incoming headset connection

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,69,0C,07,00,7C,12,34,56,78,9A ,BC,02,03	Rx: Event: Incoming Link Established, BdAddr: 123456789ABC, Local Port: 02
00 / 01 / 02 / 03	RX	RAW Data	41,54,2B,43,4B,50,44,3D,32,30,30, 0D	Rx(RAW): 41,54,2B,43,4B,50,44,3D,32,30,30,0D
	TX	RAW Data	0D,0A,4F,4B,0D,0A	Tx(RAW): 0D,0A,4F,4B,0D,0A
	TX	UART BREAK		Rx(RAW): 00
00 / 01	RX	Indicator	02,69,11,02,00,7C,02,00,03	Rx: Event: Transparent Mode, Local Port: 02, Mode: 00
	TX	Request	02,52,5D,08,00,B7,12,34,56,78,9A. BC,80,00,03	Tx: Cmd: Establish SCO Link, BdAddr: 123456789ABC, Packet Type: 8000
00 / 01	RX	Confirm	02,43,5D,07,00,A7,00,12,34,56,78, 9A.BC,03	Rx: Event: Establish SCO Link, Status: 00, BdAddr: 123456789ABC
00 / 01	RX	Indicator	02,69,5D,07,00,CD,00,12,34,56,78 ,9A.BC,03	Rx: Event: SCO Link Established, Status: 00, BdAddr: 123456789ABC

4.4.3.5.4 Audio connection release from the HS

The connection release is based on the same command as the connection confirmation, AT+CKPD=200. In case the user wants to release the link from the HS to the HSAG, it presses the same button again, which will send the same AT command. The HSAG will confirm by an OK and release the link.

The LMX9820A confirms the released link by the standard SPP Link Released Indicator (5.2.3.6). See following the Bluetooth specification description for the realization with LMX9820A. In case the HSAG releases the link the LMX9820A will send the same event.

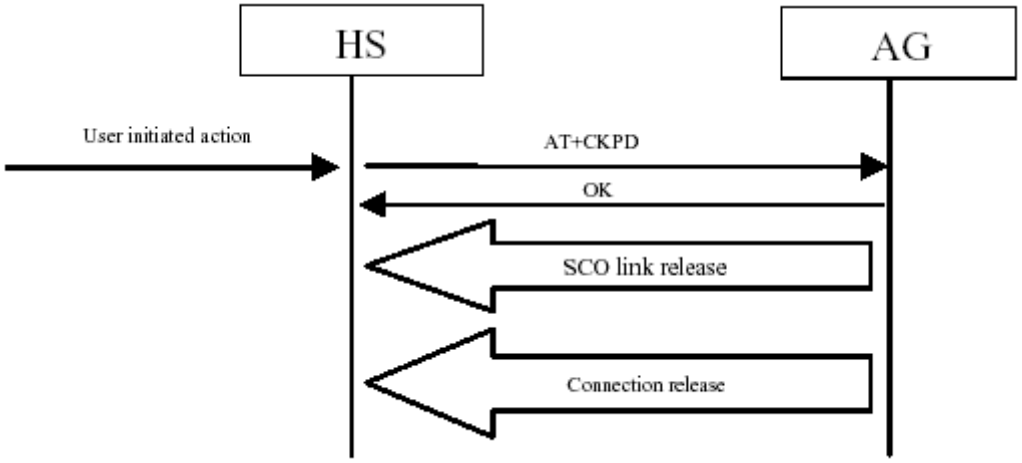


Figure 4-33. Audio link release by the HS

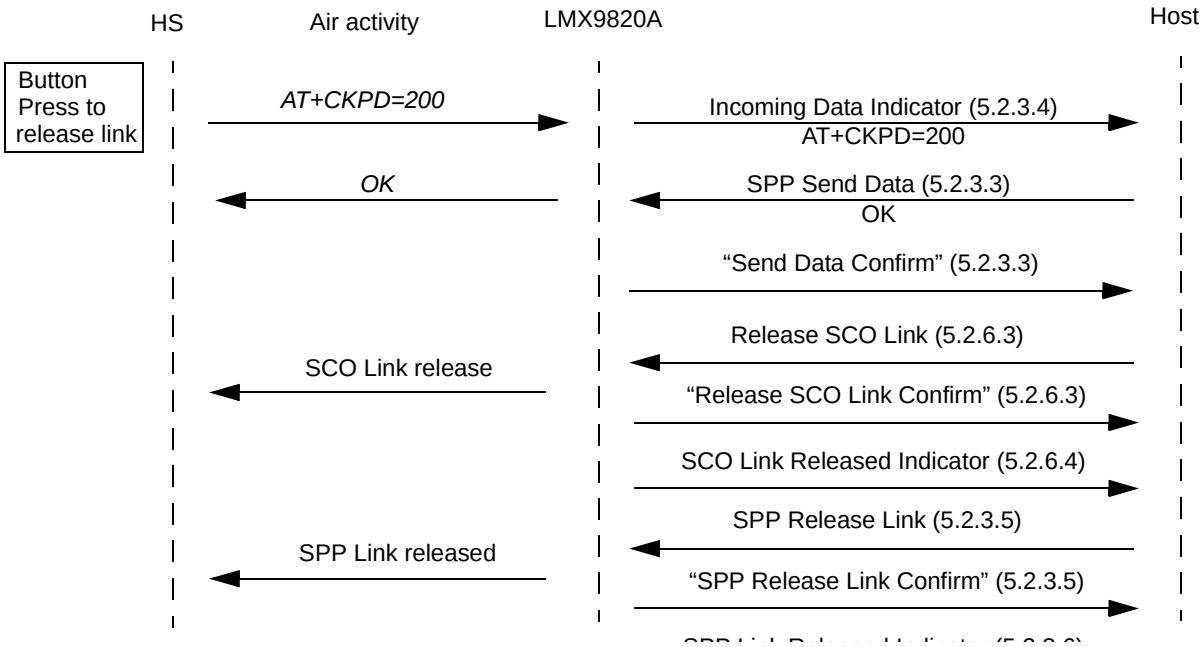


Figure 4-34. Audio link release by the HS

Table 4-31. Audio link release by the HS

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
00 / 01	RX	Indicator	02,69,10,0F,00,88,02,0C,00,41,54, 2B,43,4B,50,44,3D,32,30,30,0D03	Rx: Event: Incoming Data, Local Port: 02, Received Data: 41542B434B50443D3230300D
	TX	Request	02,52,0F,09,00,6A,02,06,00,0D,0A, 4F,4B,0D,0A,03	Tx: Cmd: Send Data, Local Port: 02, Payload Data: 0D0A4F4B0D0A
00 / 01	RX	Confirm	02,43,0F,02,00,54,00,02,03	Rx: Event: Send Data, Status: 00, Local Port: 02
	TX	Request	02,52,5E,00,00,B0,03	Tx: Cmd: Release SCO Link

Table 4-31. Audio link release by the HS

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
00 / 01	RX	Confirm	02,43,5E,01,00,A2,00,03	Rx: Event: Release SCO Link, Status: 00
00 / 01	RX	Indicator	02,69,5E,02,00,C9,00,00,03	Rx: Event: SCO Link Released, Status: 00, HCI Reason: 00
	TX	Request	02,52,0D,01,00,60,02,03	Tx: Cmd: Release Link, Local Port: 02
00 / 01	RX	Confirm	02,43,0D,02,00,52,00,02,03	Rx: Event: Release Link, Status: 00, LocalPort: 02
00 / 01	RX	Indicator	02,69,0E,02,00,79,00,02,03	Rx: Event: Link Released, Reason: 00, Local Port: 02
00 / 01	RX	Indicator	02,69,51,07,00,C1,69,9A,01,A4,07, 00,16,03	Rx: Event: ACL Terminated, BdAddr: 699A01A40700, Reason: 16

4.4.3.5.5 Audio connection release initiated by the HSAG

The release of the audio connection uses the standard LMX9820A commandset, no further AT command is required.

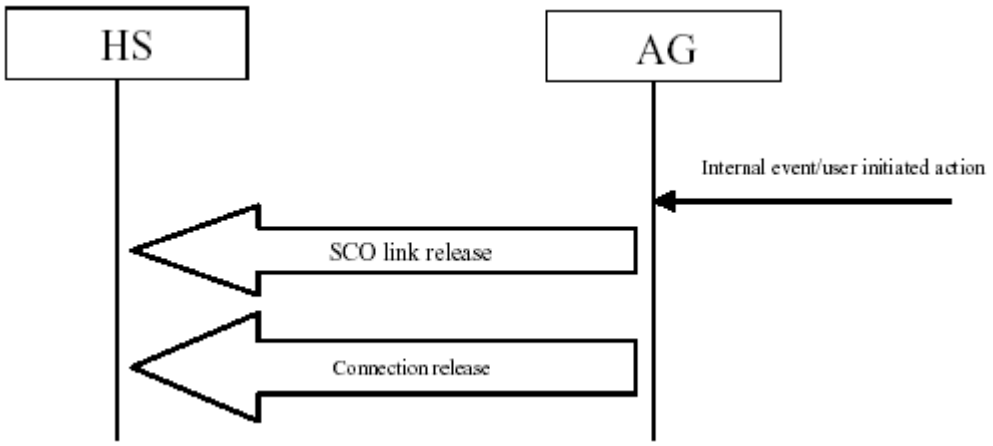


Figure 4-35. Audio link release initiated by the HSAG

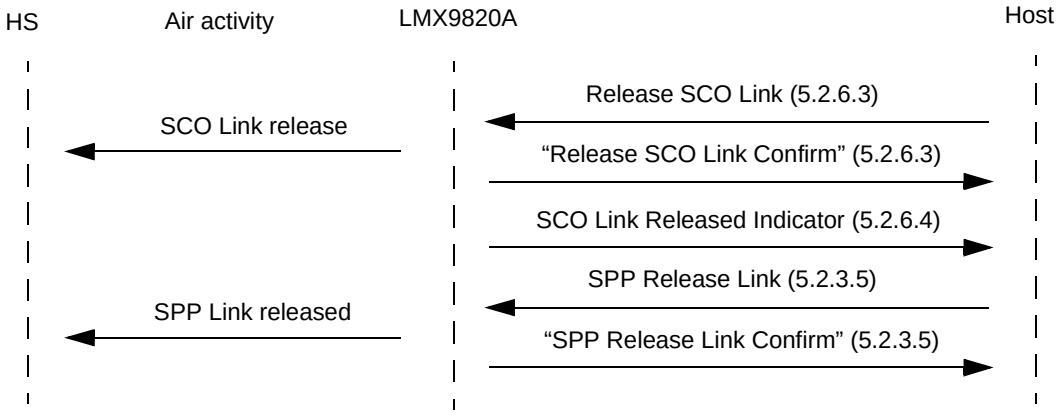


Figure 4-36. Audio link release initiated by the HSAG

Table 4-32. Audio link release by the HSAG

Filter	Direction	What	Hex Code	Interpreted by Simply Blue Commander
	TX	Request	02,52,5E,00,00,B0,03	Tx: Cmd: Release SCO Link
00 / 01	RX	Confirm	02,43,5E,01,00,A2,00,03	Rx: Event: Release SCO Link, Status: 00
00 / 01	RX	Indicator	02,69,5E,02,00,C9,00,00,03	Rx: Event: SCO Link Released, Status: 00, HCI Reason: 00
	TX	Request	02,52,0D,01,00,60,02,03	Tx: Cmd: Release Link, Local Port: 02
00 / 01	RX	Confirm	02,43,0D,02,00,52,00,02,03	Rx: Event: Release Link, Status: 00, LocalPort: 02
00 / 01	RX	Indicator	02,69,0E,02,00,79,00,02,03	Rx: Event: Link Released, Reason: 00, Local Port: 02
00 / 01	RX	Indicator	02,69,51,07,00,C1,69,9A,01,A4,07, 00,16,03	Rx: Event: ACL Terminated, BdAddr: 699A01A40700, Reason: 16

5.0 LMX9820A Command Interface

5.1 UART PROTOCOL PRINCIPLES

The LMX9820A can be controller with simple commands on the UART interface. The commands have to be sent within a special package format. The following sections describe the format of the command set packages.

5.1.1 Framing

The connection is considered "Error free". But for packet recognition and synchronization, some framing is used.
All packets sent in both directions are constructed after the following model:

Table 5-1. Package Framing

Start delimiter	Packet Type identification	Op code	Data length	Check-sum	Packet Data	End delimiter
1 byte	1 byte	1 byte	2 bytes	1 byte	<Data length> bytes	1 byte
	|---------- Checksum ----------|					

5.1.2 Start delimiter

The start delimiter indicates the LMX9820A the beginning of a new package. The "STX" char is used as start delimiter.

STX = 0x02

5.1.3 Packet type identification

This byte identifies the type of packet. The following types are valid:

Table 5-2. Packet Type Identification

Code	Packet Type	Description
0x52 'R'	Request (REQ)	A request sent to the Bluetooth module. All request are answered by exactly one confirm.
0x43 'C'	Confirm (CFM)	The Bluetooth modules confirm to a request. All request are answered by exactly one confirm.
0x69 'i'	Indication (IND)	Information sent from the Bluetooth module, that is not a direct confirm to a request.
0x72 'r'	Response (RES)	An optional response to an indication. This is used to respond to some type of indication messaged.

All other values are reserved.

5.1.4 Opcode

The opcode is a command specifier. Each command is represented by this one byte identifier.

Table 5-3. Opcode Values

Opcode	Value
GAP_INQUIRY	0x00
GAP_DEVICE_FOUND	0x01
GAP_REMOTE_DEVICE_NAME	0x02
GAP_READ_LOCAL_NAME	0x03
GAP_WRITE_LOCAL_NAME	0x04
GAP_READ_LOCAL_BDA	0x05
GAP_SET_SCANMODE	0x06
GAP_GET_FIXED_PIN	0x16

Table 5-3. Opcode Values

GAP_SET_FIXED_PIN	0x17
GAP_GET_PIN	0x75
GAP_GET_SECURITY_MODE	0x18
GAP_SET_SECURITY_MODE	0x19
GAP_REMOVE_PAIRING	0x1B
GAP_LIST_PAIRED_DEVICES	0x1C
GAP_ENTER_SNIFF_MODE	0x21
GAP_EXIT_SNIFF_MODE	0x37
GAP_ENTER_PARK_MODE	0x38
GAP_EXIT_PARK_MODE	0x39
GAP_ENTER_HOLD_MODE	0x3A
GAP_SET_LINK_POLICY	0x3B
GAP_GET_LINK_POLICY	0x3C
GAP_POWER_SAVE_MODE_CHANGED	0x3D
GAP_ACL_ESTABLISHED	0x50
GAP_ACL_TERMINATED	0x51
GAP_SET_AUDIO_CONFIG	0x59
GAP_GET_AUDIO_CONFIG	0x5A
GAP_ESTABLISH_SCO_LINK	0x5D
GAP_RELEASE_SCO_LINK	0x5E
GAP_MUTE_MIC	0x5F
GAP_SET_VOLUME	0x60
GAP_GET_VOLUME	0x61
GAP_CHANGE_SCO_PACKET_TYPE	0x62
SPP_SET_PORT_CONFIG	0x07
SPP_GET_PORT_CONFIG	0x08
SPP_PORT_CONFIG_CHANGED	0x09
SPP_ESTABLISH_LINK	0x0A
SPP_LINK_ESTABLISHED	0x0B
SPP_INCOMMING_LINK_ESTABLISHED	0x0C
SPP_RELEASE_LINK	0x0D
SPP_LINK_RELEASED	0x0E
SPP_SEND_DATA	0x0F
SPP_INCOMING_DATA	0x10
SPP_TRANSPARENT_MODE	0x11
SPP_CONNECT_DEFAULT_CON	0x12
SPP_STORE_DEFAULT_CON	0x13
SPP_GET_LIST_DEFAULT_CON	0x14
SPP_DELETE_DEFAULT_CON	0x15
SPP_SET_LINK_TIMEOUT	0x57
SPP_GET_LINK_TIMEOUT	0x58

Table 5-3. Opcode Values

SPP_PORT_STATUS_CHANGED	0x3E
SPP_GET_PORT_STATUS	0x40
SPP_PORT_SET_DTR	0x41
SPP_PORT_SET_RTS	0x42
SPP_PORT_BREAK	0x43
SPP_PORT_OVERRUN_ERROR	0x44
SPP_PORT_PARITY_ERROR	0x45
SPP_PORT_FRAMING_ERROR	0x46
SDAP_CONNECT	0x32
SDAP_DISCONNECT	0x33
SDAP_CONNECTION_LOST	0x34
SDAP_SERVICE_BROWSE	0x35
SDAP_SERVICE_SEARCH	0x36
SDAP_SERVICE_REQUEST	0x1E
SDAP_ATTRIBUTE_REQUEST	0x3F
CHANGE_LOCAL_BDADDRESS	0x27
CHANGE_NVS_UART_SPEED	0x23
CHANGE_UART_SETTINGS	0x48
SET_PORTS_TO_OPEN	0x22
GET_PORTS_TO_OPEN	0x1F
RESTORE_FACTORY_SETTINGS	0x1A
STORE_CLASS_OF_DEVICE	0x28
FORCE_MASTER_ROLE	0x1D
READ_OPERATION_MODE	0x49
WRITE_OPERATION_MODE	0x4A
SET_DEFAULT_LINK_POLICY	0x4C
GET_DEFAULT_LINK_POLICY	0x4D
SET_EVENT_FILTER	0x4E
GET_EVENT_FILTER	0x4F
SET_DEFAULT_LINK_TIMEOUT	0x55
GET_DEFAULT_LINK_TIMEOUT	0x56
SET_DEFAULT_AUDIO_CONFIG	0x5B
GET_DEFAULT_AUDIO_CONFIG	0x5C
SET_DEFAULT_LINK_LATENCY	0x63
GET_DEFAULT_LINK_LATENCY	0x64
SET_PCM_SLAVE_CONFIG	0x74
SET_DEFAULT_SNIFF_MODE_PARAMETERS	0x76
GET_DEFAULT_SNIFF_MODE_PARAMETERS	0x77
ENABLE_SDP_RECORD	0x29
DELETE_SDP_RECORDS	0x2A

Table 5-3. Opcode Values

STORE_SPP_RECORD	0x2B
STORE_DUN_RECORD	0x2C
STORE_FAX_RECORD	0x2D
STORE_OPP_RECORD	0x2E
STORE_FTP_RECORD	0x2F
STORE_SYNC_RECORD	0x30
STORE_SDP_RECORD	0x31
RESET	0x26
LMX9820A_READY	0x25
TEST_MODE	0x24
FIRMWARE_UPGRADE	0x47
READ_RSSI	0x20
RF_TEST_MODE	0x4B
DISABLE_TL	0x52
TL_ENABLED	0x53
HCI_COMMAND	0x65
READ_NVS	0x72
WRITE_NVS	0x73

5.1.5 Data length

Number of bytes in the "Packet data" area. The maximum size is 333 bytes.

5.1.6 Packet data

The data fields hold binary data; hence both 0x02 (=STX) and 0x03 (=ETX) are allowed as data.

5.1.7 Checksum

This is a simple Block Check Character (BCC) checksum of the bytes from "Packet type" to, and including, “data length”. The BCC checksum is calculated as the low byte of the sum of all bytes.

E.g. if the sum of all bytes are 0x3724, the checksum is 0x24.

5.1.8 End delimiter

The "ETX" char is used as end delimiter.
ETX = 0x03

5.1.9 Retransmission

The connections is considered "Error free", hence no need for implementing timeouts and retransmissions.

5.1.10 Flow control

A transparent data-mode is supported for RFCOMM communication. When using this transparent mode, full hardware handshake is needed.

When not in transparent mode, the protocol principle of REQ-CFM, limits the need of buffer capacity. As IND's can come out of REQ-CFM sequence, and is unconfirmed, the user device has to be able to read these data fast enough / have enough buffer capacity.

5.1.11 Byte Order

The byte order of the protocol is Little Endian, if nothing else is specified.

5.2 COMMAND SET

The LMX9820A implements a complete command set for bluetooth operation and local configuration.

The command set is based on a request/confirm scheme meaning any command will be confirmed by an appropriate event including the same opcode.

5.2.1 Searching for remote devices

The first step to establish a link to another device is to discover the devices in range. The discovering process in called “Inquiry”.

5.2.1.1 Inquiry

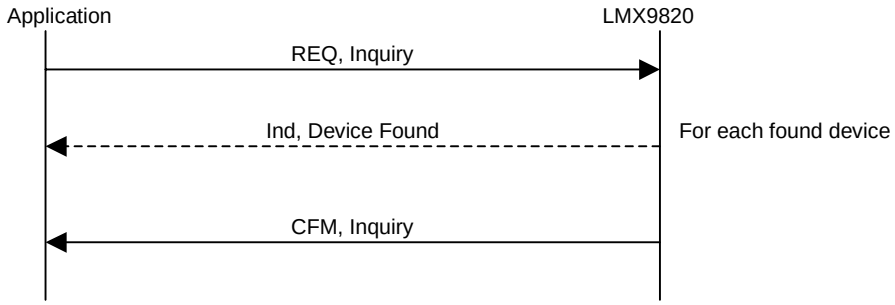


Figure 5-1. Inquiry Command Flow

Table 5-4. Inquiry Command

Description	**Initiates a search for other Bluetooth devices.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	GAP_INQUIRY	
DataLength	3	
Data	Length 1 byte	Duration of inquiry Range: 0x01 -0x30 (1.28s - 61.44s)
	NumResponses 1 byte	Maximum number of responses Range: 0x00 - 0xFF 0x00 = Unlimited number of responses.
	Mode 1 Byte	General Inquiry 0x00 Limited Inquiry 0x01

Table 5-5. Inquiry Confirm

Description	**Confirms that the search for other Bluetooth devices is complete.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	GAP_INQUIRY	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_DURATION_OUT_OF_RANGE ERROR_INVALID_MODE ERROR_INVALID_NO_OF_PARAMETERS

Table 5-6. Device Found Indicator

Description	Indicates that a device has been found.	
Firmware	5.xx	
PacketType	IND	
Opcode	GAP_DEVICE_FOUND	
DataLength	9	
Data	BdAddr 6 bytes	Bluetooth device address of the found device.
	DeviceClass 3 byte	Class of the found device.

5.2.1.2 Get Friendly Name of the Remote Device

As seen the Device Found Indicator only delivers the BD_Addr and the Class of Device of Remote Devices. To get the friendly name of the device a separate command has to be used.

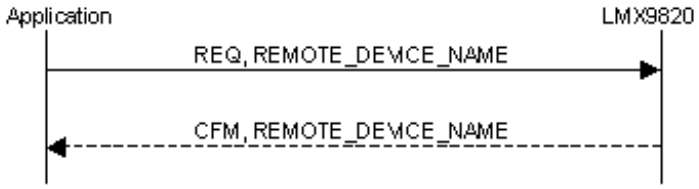


Figure 5-2. Get Remote Device Name Flow

Table 5-7. Get Remote Device Name Command

Description	Request the user-friendly name from a known remote Bluetooth device.	
Firmware	5.xx	
PacketType	REQ	
Opcode	GAP_REMOTE_DEVICE_NAME	
DataLength	6	
Data	BdAddr 6 byte	Bluetooth device address for the remote device

Table 5-8. Get Remote Device Name Confirm

Description	Confirm to the request above.	
Firmware	5.xx	
PacketType	CFM	
Opcode	GAP_REMOTE_DEVICE_NAME	
DataLength	8+ NameLength if ok, otherwise 8	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_TIMEOUT
	BdAddr 6 byte	Bluetooth device address for the remote device
	NameLength 1 byte	Number of bytes in device name

Table 5-8. Get Remote Device Name Confirm

	DeviceName	The user-friendly name of the remote device.
	Length bytes	NULL terminated. Maximum length is 30 bytes.

5.2.2 SDAP Client Commands

Establishing a link to another device requires that devices BD_Addr but also the RFComm Port Number, the profile to connect to is registered at. If remote Com Port is unknown, it can be requested by a service discovery request using the the SDAP client.

A SDAP request can only be done via an establish SDAP link. So the Command flow would be as following:

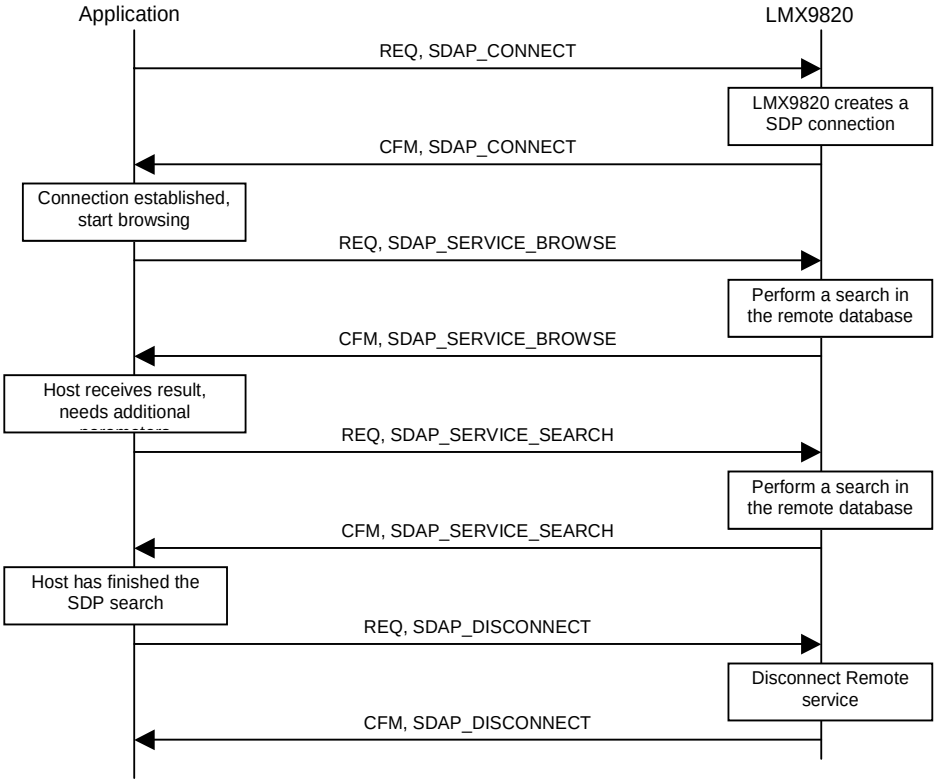


Figure 5-3. Requesting the services of a remote device

5.2.2.1 SDAP Connect

The SDAP Connect Request forces the LMX9820A to create a SDAP link to another device. This command is required for further SDAP Service Requests

Table 5-9. SDAP Connect Request

Description	**Creates an SDP connection to a remote device. Only one SDP connection can be active.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	SDAP_CONNECT	
DataLength	6	
Data	BdAddr 6 bytes	The Bluetooth address of the remote device to connect to.

Table 5-10. SDAP Connect Confirm

Description	Confirms the request above	
Firmware	5.xx	
PacketType	CFM	
Opcode	SDAP_CONNECT	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_CONNECTION_FAILED

5.2.2.2 SDAP Disconnect

The SDAP link has to be disconnected after finishing the service browse/search.r

Table 5-11. Disconnect Request

Description	This command disconnects the active SDP connection.
Firmware	5.xx
PacketType	REQ
Opcode	SDAP_DISCONNECT
DataLength	0

Table 5-12. Disconnect Confirm

Description	Confirms the request above	
Firmware	5.xx	
PacketType	CFM	
Opcode	SDAP_DISCONNECT	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS

5.2.2.3 SDAP Connection Lost Indicator

This indicator appears after an unexpected loss of the SDAP link.

Table 5-13. Connection Lost Indicator

Description	Notification sent to the application when a loss of the SDP connection is detected.
Firmware	5.xx
PacketType	IND
Opcode	SDAP_CONNECTION_LOST
DataLength	0

5.2.2.4 SDAP Service Browse

The SDAP Service Browse can be used to get the RFComm Portnumbers of all or only specific Service Classes. The search mechanism is based on the 16bit-UUID for the services. The actual list of UUIDs can be found within the “Bluetooth Assigned Numbers” Document of the Bluetooth SIG.

Table 5-14. Example UUIDs for Service Classes

Service Class	UUID	Description
PublicBrowseGroup	0x1002	Returns the list of all registered services
SPP	0x1101	Serial Port Profile
DUN	0x1103	Dial-Up Networking Profile

Table 5-15. SDAP Service Browse Request

Description	This command is used to browse the service record of the remote device.	
Firmware	5.xx	
PacketType	REQ	
Opcode	SDAP_SERVICE_BROWSE	
DataLength	2	
Data	BrowseGroupID 2 bytes	The requested browse group (16 bit UUID). The UUID has to be byte swapped within the command, e.g. to search for SPP entries the full command is 02 52 35 02 00 89 **01 11** 03

Table 5-16. SDAP Service Browse Confirm

Description	Confirms the request above		
Firmware	5.xx		
PacketType	CFM		
Opcode	SDAP_SERVICE_BROWSE		
DataLength	2+NoOfServices*(6 + NameLength)		
Data	Status 1 byte		ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_RESULT_TOO_LARGE ERROR_TRUNCATED_ANSWER
	NoOfServices 1 byte		Number of services found on remote device.
	For each service		
		BrowseGroupID 2 bytes	The browse group UUID that the service belongs to.
		ServiceID 2 bytes	The service UUID.
		PortNr 1 byte	RFCOMM port number. The port which has to be used for link establishment to that service.
		NameLength 1 byte	The number of bytes in the service name
		ServiceName NameLength bytes	The name of the service.

5.2.2.5 SDAP Service Search

The SDAP Service Search command offers the ability to search for specific attributes for a service. The attributes are defined within the SDP Specification.

Table 5-17. SDAP Service Search Request

Description	**This command is used to search for services in the service record of the remote device.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	SDAP_SERVICE_SEARCH	
DataLength	2 + 2*SearchPatternLength + 2*AttributesLength	
Data	SearchPatternLength 1 bytes	Number of 16-bit UUID's in Search pattern List. **Note**: Must be less than 86 elements.
	SearchPattern	List of the requested services. The search pattern list is a list of 16-bit UUID's of the requested services.
	AttributesLength 1 byte	Number of 16-bit UUID's in attributeId list. **Note**: Must be less than 86 elements.
	Attributes	List of requested attributes for the requested services. The attribute list is a list of 16-bit UUID's for the requested attributes.

Table 5-18. SDAP Service Search Confirm

Description	**Confirms the request above**	
Firmware	5.xx	
PacketType	CFM	
Opcode	SDAP_SERVICE_SEARCH	
DataLength	3 + Length	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_UNKNOWN_ERROR ERROR_RESULT_TOO_LARGE
	Length 2 byte	Length of the result of the search. Maximum 330 bytes
	Result Length bytes	Result of the search

5.2.2.6 SDAP Service Request

Each Service Entry has a unique number called "service record handle". This command is used to get the record handle for stored entries for specific UUIDs.

Table 5-19. SDAP Service Request

Description	**This command is used the service record handles, from a remote device, for the given services in the search pattern.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	SDAP_SERVICE_REQUEST	
DataLength	1 +2* SearchPatternLength	
Data	SearchPatternLength 1 bytes	Number of 16-bit UUID's in Search pattern List. **Note**: Must be less than 86 elements.

Table 5-19. SDAP Service Request

	SearchPattern < 2* SearchPattern-Length > bytes	List of the requested services. The search pattern list is a list of 16-bit UUID's of the requested services

Table 5-20. SDAP Service Request Confirm

Description	Confirms the request above	
Firmware	5.xx	
PacketType	CFM	
Opcode	SDAP_SERVICE_REQUEST	
DataLength	3 + 4*Length	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_UNKNOWN_ERROR ERROR_RESULT_TOO_LARGE
	Length 2 byte	Number of 32-bit service record handles returned from remote device.
	Result <4*Length> bytes	The received 32-bit service record handles from the remote device.

5.2.2.7 SDAP Attribute Request

Instead of browsing the whole list of services within the remote database it is also possible to search only for specific attributes within a chosen entry. This command is based on the attribute "connection handle" of that specific entry.

Connection handles can be retrieved by the "SDAP Service Search" Command. (see Section 5.2.2.5 on page 108)

Table 5-21. SDAP Attribute Request

Description	This command is used to get the given attributes for a given service record handle.	
Firmware	5.xx	
PacketType	REQ	
Opcode	SDAP_ATTRIBUTE_REQUEST	
DataLength	5 + 2*AttributesLength	
Data	Handle 4 bytes	The 32-bit service record handle returned for a given service by SDAP_ATTRIBUTE_REQUEST.
	AttributesLength 1 byte	Number of 16-bit UUID's in attributeId list. **Note**: Must be less than 86 elements.
	Attributes < 2 * AttributesLen-gth> bytes	List of requested attributes for the requested services. The attribute list is a list of 16-bit UUID's for the requested attributes.

Table 5-22. SDAP Attribute Request Confirm

Description	Confirms the request above
Firmware	5.xx
PacketType	CFM
Opcode	SDAP_ATTRIBUTE_REQUEST
DataLength	3 + Length

Table 5-22. SDAP Attribute Request Confirm

Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_UNKNOWN_ERROR ERROR_RESULT_TOO_LARGE
	Length 2 byte	Length of the result of the search. Maximum 330 bytes.
	Result	Result of the search

5.2.3 SPP Link Establishment

This section describes the basic functionality of creating a full SPP link to a remote device. Basically only one single command is needed to create the connection. The command “Establish Link” requires the BD_Addr and the RFComm port of the remote device, determined out of the Inquiry and the SDAP connection. The command and event flow can be found within the following flowchart.

The command will first be confirmed by a standard confirmation package. Afterwards the LMX9820A will start to page and try to connect to the remote device. The SPP_Link_Establishment indicator returns an error code reporting the success of the link establishment.

The flow also shows the procedure of sending data to the remote device using the “Send Data” Command. Incoming data are indicated by the “Incoming data event”.

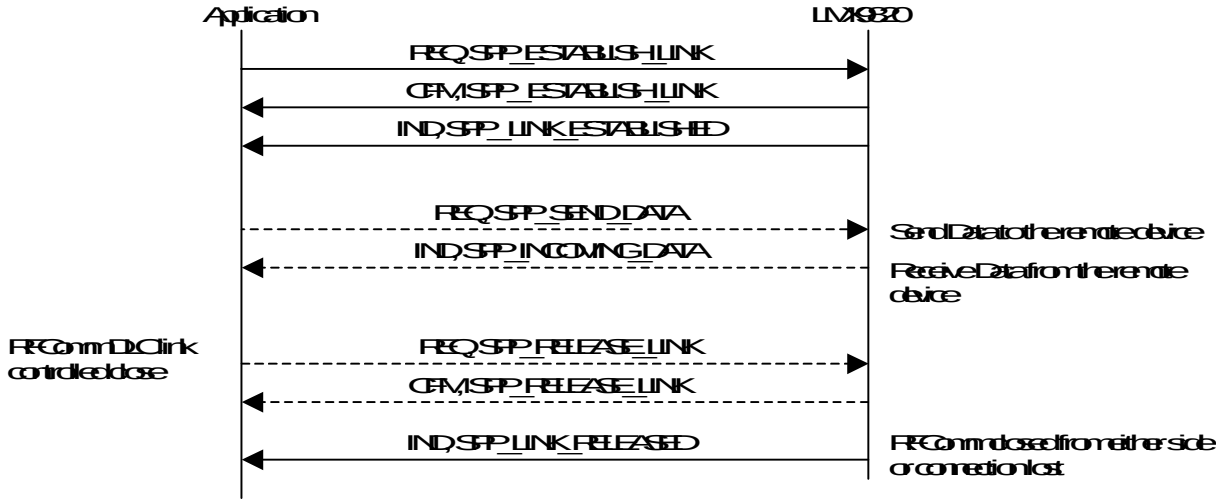


Figure 5-4. SPP Link Handling Flow

5.2.3.1 Establish Link

The Establish Link command is the major command to establish a link to a remote device. To create a link the BD_Addr and the RFComm Channel on the remote device is required.

Table 5-23. Establish Link Request

Description	Establish a DLC link to remote Bluetooth device
Firmware	5.xx
PacketType	REQ

Table 5-23. Establish Link Request

Opcode	SPP_ESTABLISH_LINK	
DataLength	8	
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30
	BdAddr 6 byte	Bluetooth device address for the remote device
	RemotePortNumber 1 byte	Remote device RFCOMM port number. (Must be found using SDAP)

Table 5-24. Establish Link Confirm

Description	**Confirm that the DLC link establishment is initiated. Note: This confirm does NOT indicate link establishment, only that link establishment is in progress. When link establishment response is received from the core, a SPP_LINK_ESTABLISHED indication is sent from the LMX9820A.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	SPP_ESTABLISH_LINK	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_SPP_PORT_BUSY ERROR_SPP_PORT_NOT_OPEN ERROR_SPP_INVALID_PORT ERROR_SPP_AUTOMATIC_CONNECTIONS_PROGRESSING
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

5.2.3.2 SPP Link Established Indicator

Table 5-25. SPP Link Established Indicator

Description	**Indication of establishment of a locally requested DLC link.**	
Firmware	5.xx	
PacketType	IND	
Opcode	SPP_LINK_ESTABLISHED	
DataLength	9	
Data	Status 1 byte	Refer to “RFCOMM Error Codes” (page 184).
	BdAddr 6 byte	Bluetooth device address for the remote device.
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30.
	RemotePortNumber 1 byte	Remote device RFCOMM port number.

5.2.3.3 SPP Send Data

If not switched to transparent, data have to be sent to a remote device using this command. The local RFComm Port is used to address the remote device.

Table 5-26. SPP Send Data

Description	**Send data on a SPP link to remote Bluetooth device**	
Firmware	5.xx	
PacketType	REQ	
Opcode	SPP_SEND_DATA	
DataLength	3 + <PayloadSize>	
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30
	PayloadSize 2 bytes	Number of data bytes to send. Valid range is 1 to 330 bytes.
	PayloadData <PayloadSize> bytes	The data to send.

Description	**Confirm to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	SPP_SEND_DATA	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_LIMIT ERROR_UNABLE_TO_SEND ERROR_CURRENTLY_NO_BUFFER ERROR_NO_CONNECTION ERROR_SPP_INVALID_PORT ERROR_SPP_PORT_NOT_OPEN
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

5.2.3.4 Incoming Data Indicator

Table 5-27. SPP Incoming Data Indicator

Description	**Incoming data on a DLC link, from a remote Bluetooth device**	
Firmware	5.xx	
PacketType	IND	
Opcode	SPP_INCOMING_DATA	
DataLength	3 + <PayloadSize>	
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30
	PayloadSize 2 bytes	Number of data bytes to send. Valid range is 1 to 330 bytes.

Table 5-27. SPP Incoming Data Indicator

	PayloadData <PayloadSize> bytes	The data to send.

5.2.3.5 SPP Release Link

Table 5-28. SPP Release Link Request

Description	**Release a DLC link to remote Bluetooth device**	
Firmware	5.xx	
PacketType	REQ	
Opcode	SPP_RELEASE_LINK	
DataLength	1	
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

Table 5-29. SPP Release Link Confirm

Description	**Confirm that the release is initiated. When the release is complete, a SPP_LINK_RELEASED indication is sent from the LMX9820A.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	SPP_RELEASE_LINK	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_NO_CONNECTION ERROR_SPP_INVALID_PORT ERROR_SPP_PORT_NOT_OPEN
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

5.2.3.6 SPP Link Released Indicator

Table 5-30. SPP Link Released Indicator

Description	**Indicates that a DLC link is released. The link release may have been initiated locally or remote, or could be caused by a loss of link (disturbance, dead device, ….)**	
Firmware	5.xx	
PacketType	IND	
Opcode	SPP_LINK_RELEASED	
DataLength	2	
Data	Reason 1 byte	Refer to “RFCOMM Release Reasons” (page 184)
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

5.2.3.7 Incoming Link Established Indicator

In case a remote device creates a link to the LMX9820A, the device will indicate the successful link establishment by sending by the “Incoming Link Established Indicator”. The packet includes the BD_Addr of the remote device and the local RFComm Port it connected to.

Table 5-31. Incoming Link Established Indicator

Description	Indication of establishment of a remotely requested DLC link.	
Firmware	5.xx	
PacketType	IND	
Opcode	SPP_ INCOMING_LINK_ESTABLISHED	
DataLength	7	
Data	BdAddr 6 byte	Bluetooth device address for the remote device
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

5.2.4 Transparent Mode

Transparent Mode offers the ability to switch off the Command Interface on the LMX9820A and use it as a cable replacement. This means data can be sent over a bluetooth link just by routing them to the LMX9820A without any package framing.

Transparent mode can only be enabled if one SPP link is established. Transparent mode can not be used if a device has two or more active SPP links.

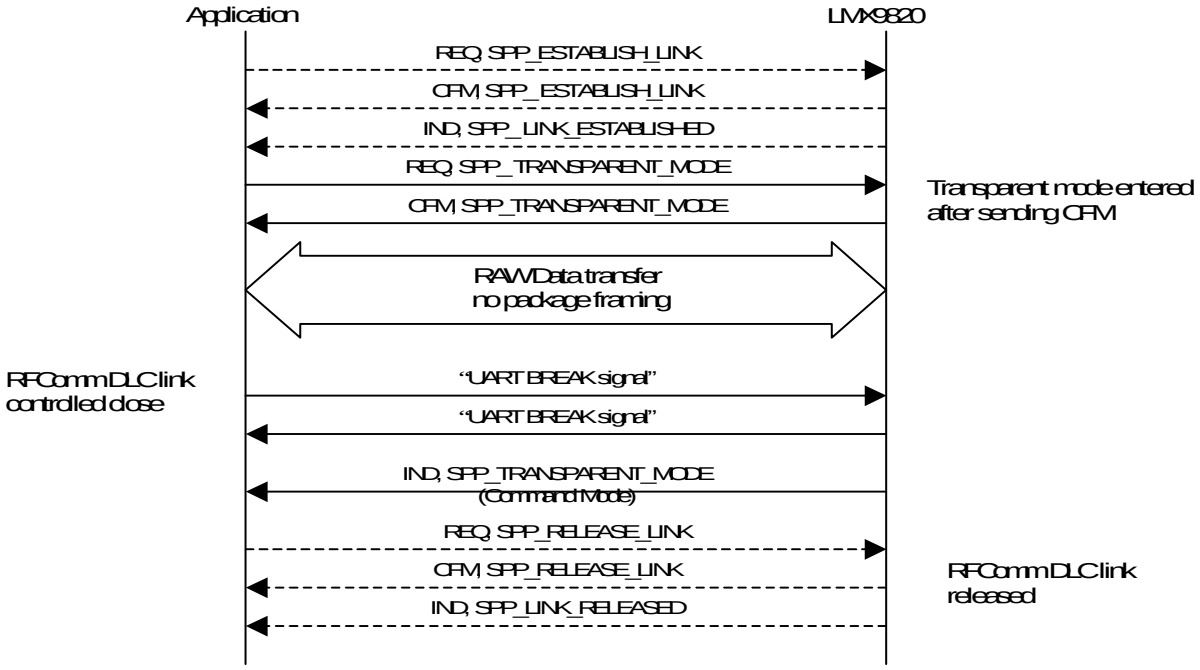


Figure 5-5. Transparent Mode; initiated by the Transparent Mode Request

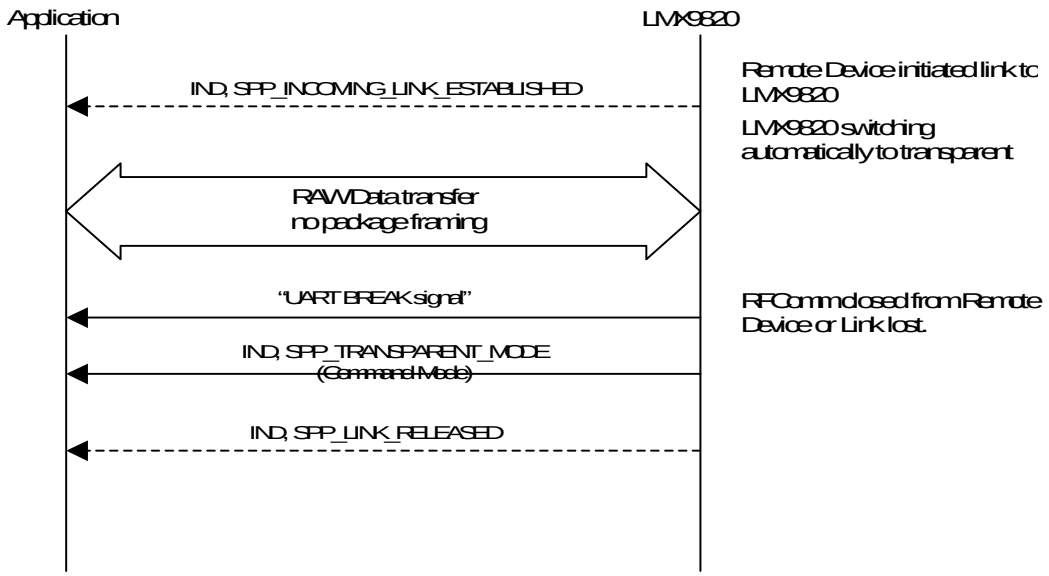


Figure 5-6. Transparent Mode; automatically activated in automatic slave operation

5.2.4.1 Transparent Mode

Table 5-32. Transparent Mode Request

Description	**Switch to transparent mode on a SPP link to remote Bluetooth device**	
Firmware	5.xx	
PacketType	REQ	
Opcode	SPP_TRANSPARENT_MODE	
DataLength	1	
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

Table 5-33. Transparent Mode Confirm

Description	**Confirm to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	SPP_TRANSPARENT_MODE	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_SPP_PORT_NOT_OPEN ERROR_SPP_INVALID_PORT ERROR_SPP_MULTIPLE_CONNECTIONS ERROR_NO_CONNECTION
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

5.2.4.2 Transparent Mode Left Indicator

Table 5-34. Transparent Mode Left Indicator

Description	**Indication from Simply Blue that transparent mode is left.**	
Firmware	5.xx	
PacketType	IND	
Opcode	SPP_TRANSPARENT_MODE	
DataLength	2	
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30
	Mode 1 byte	Command Mode = 0 Transparent Mode = 1

5.2.5 Link Timeout

The Link Timeout commands report and set the Supervision timeout setting for an established link. The link timeout which is set during link establishment on all incoming and outgoing links is read out of the NVS. The timeout can either be changed on an existing link or for each link by configuring the default setting in NVS.

See also Section 3.2 “Configuring the Default Link Timeout” on page 38 for a detailed description of the feature.

5.2.5.1 Set Default Link Timeout

Table 5-35. Set Default Link Timeout Request

Description	**This command is used to change the default link supervision timeout. The default link supervision timeout is set during connection setup. The default link supervision timeout setting is stored in NVS.**	
Firmware	5.15 or higher	
PacketType	REQ	
Opcode	SET_DEFAULT_LINK_TIMEOUT	
DataLength	2	
Data	LinkTimeout 2 byte	The link supervision timeout in slots (0,625ms). The default value stored in the NVS after a factory reset is 20s. 0x0000: No link supervision timeout (the timer is disabled) 0x0190-0xFFFF: Valid timeout range (in slots) 0x7D00: The default value (20s)

Table 5-36. Set Default Link Timeout Confirm

Description	Response to the request above.	
Firmware	5.15 or higher	
PacketType	CFM	
Opcode	SET_DEFAULT_LINK_TIMEOUT	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_ILLEGAL_LINK_TIMEOUT

5.2.5.2 Get Default Link Timeout

Table 5-37. Get Default Link Timeout Request

Description	This command is used to read the default link supervision timeout setting from NVS.	
Firmware	5.15 or higher	
PacketType	REQ	
Opcode	GET_DEFAULT_LINK_TIMEOUT	
DataLength	0	

Table 5-38. Get Default Link Timeout Confirm

Description	Response to the request above.	
Firmware	5.15 or higher	
PacketType	CFM	
Opcode	GET_DEFAULT_LINK_TIMEOUT	
DataLength	2	
Data	LinkTimeout 2 byte	The link supervision timeout in slots. 0x0000: No link supervision timeout (the timer is disabled) 0x0190-0xFFFF: Valid timeout range (in slots)

5.2.5.3 Set Link Timeout for an existing link

Table 5-39. Set Link Timeout request

Description	This command is used to change the current ACL link supervision timeout. This command will affect all physical links established to the specified device.	
Firmware	5.15 or higher	
PacketType	REQ	
Opcode	SPP_SET_LINK_TIMEOUT	
DataLength	8	
Data	BdAddr 6 bytes	The Bluetooth address of the remote device of the ACL link for which the Link Supervision Timeout is changed.
	LinkTimeout 2 byte	The link supervision timeout in slots (0,625ms). 0x0000: No link supervision timeout (the timer is disabled) 0x0190-0xFFFF: Valid timeout range (in slots) 0x7D00: The default value (20s)

Table 5-40. Set Link Timeout Confirm

Description	Response to the request above.
Firmware	5.15 or higher
PacketType	CFM
Opcode	SPP_SET_LINK_TIMEOUT
DataLength	7

Table 5-40. Set Link Timeout Confirm

Data	Status 1 byte	ERROR_OK ERROR_UNSPECIFIED_ERROR ERROR_INVALID_NO_OF_PARAMETERS ERROR_NO_CONNECTION ERROR_ILLEGAL_LINK_TIMEOUT
	BdAddr 6 bytes	The Bluetooth address of the remote device of the ACL link for which the Link Supervision Timeout is changed.

5.2.5.4 Get Link Timeout of an existing link

Table 5-41. Get Link Timeout request

Description	**This command is used to get the current link supervision timeout setting for the given ACL link.**	
Firmware	5.15 or higher	
PacketType	REQ	
Opcode	SPP_GET_LINK_TIMEOUT	
DataLength	6	
Data	BdAddr 6 bytes	The Bluetooth address of the remote device of the ACL link for which the Link Supervision Timeout is requested.

Table 5-42. Get Link Timeout confirm

Description	**Response to the request above.**	
Firmware	5.15 or higher	
PacketType	CFM	
Opcode	SPP_GET_LINK_TIMEOUT	
DataLength	9	
Data	Status 1 byte	ERROR_OK ERROR_UNSPECIFIED_ERROR ERROR_INVALID_NO_OF_PARAMETERS ERROR_NO_CONNECTION
	BdAddr 6 bytes	The Bluetooth address of the remote device of the ACL link for which the Link Supervision Timeout is requested.
	LinkTimeout 2 byte	The link supervision timeout in slots (0,625ms). 0x0000: No link supervision timeout (the timer is disabled) 0x0190-0xFFFF: Valid timeout range (in slots)

5.2.6 Establish Audio (SCO) Link

5.2.6.1 Establish SCO Link

Table 5-43. Establish SCO Link Command

Description	**This command is used to establish a SCO link to a remote Bluetooth device. An ACL link must be established before a SCO link can be established.**
Firmware	6.xx

Table 5-43. Establish SCO Link Command

PacketType	REQ	
Opcode	GAP_ESTABLISH_SCO_LINK	
DataLength	8	
Data	BdAddr 6 bytes	The Bluetooth address of the remote device to which the SCO link is established.
	PacketType 2 bytes	0x0020: HV1 0x0040: HV2 0x0080: HV3 The packet types can be combined / or'ed together in order to enable multiple packet types.

Table 5-44. Establish SCO Link Confirm

Description	**Response to the request above.**	
Firmware	6.xx	
PacketType	CFM	
Opcode	GAP_ESTABLISH_SCO_LINK	
DataLength	7	
Data	Status 1 byte	ERROR_OK ERROR_UNSPECIFIED_ERROR
	BdAddr 6 bytes	The Bluetooth address of the remote device.

5.2.6.2 SCO Link Established Indicator

Table 5-45. SCO Link Established Indicator

Description	**Indicates that a SCO link is established. Either the local device or the remote device may have initiated the link establishment.**	
Firmware	6.xx	
PacketType	IND	
Opcode	GAP_ESTABLISH_SCO_LINK	
DataLength	7	
Data	Status 1 byte	ERROR_OK ERROR_UNSPECIFIED_ERROR ERROR_NO_CONNECTION ERROR_CONNECTION_FAILED
	BdAddr 6 bytes	The Bluetooth address of the remote.

5.2.6.3 Release SCO Link

Table 5-46. Release SCO Link Command

Description	**This command is used to disconnect a SCO link.**
Firmware	6.xx
PacketType	REQ

Table 5-46. Release SCO Link Command

Opcode	GAP_RELEASE_SCO_LINK
DataLength	0

Table 5-47. Release SCO Link Confirm

Description	**Response to the request above.**	
Firmware	6.xx	
PacketType	CFM	
Opcode	GAP_RELEASE_SCO_LINK	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_NO_CONNECTION ERROR_INVALID_NO_OF_PARAMETERS

5.2.6.4 SCO Link Released Indicator

Table 5-48. SCO Link Released Indicator

Description	**Indicates that a SCO link is released. The link release may have been initiated by the local device, the remote device or it could be caused by a loss of link (disturbance, dead device, etc.)**	
Firmware	6.xx	
PacketType	IND	
Opcode	GAP_RELEASE_SCO_LINK	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_UNKNOWN_ERROR ERROR_NO_CONNECTION
	HciReason 1 byte	The HCI release reason. The release reason indicated by the remote application.

5.2.6.5 Set audio settings

Table 5-49. Set audio settings command

Description	**This command is used to set the audio settings the device shall use for the next SCO (audio) link established.**	
Firmware	6.xx	
PacketType	REQ	
Opcode	GAP_SET_AUDIO_CONFIG	
DataLength	2	
Data	CodecType 1 byte	0x00: No codec available 0x01: Motorola MC145483 0x02: OKI MSM7717 0x03: PCM Slave 0x04: Winbond W681360 0x05-0xFF: Reserved.

Table 5-49. Set audio settings command

	AirFormat 1 byte	0x00: CVSD 0x01: µ-law 0x02: A-law 0x03-0xFF: Reserved.

Table 5-50. Set audio settings confirm

Description	Response to the request above.	
Firmware	6.xx	
PacketType	CFM	
Opcode	GAP_SET_AUDIO_CONFIG	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_ILLEGAL_AUDIO_CODEC_TYPE ERROR_ILLEGAL_AUDIO_AIR_FORMAT

5.2.6.6 Get audio settings

Table 5-51. Get audio settings command

Description	This command is used to get the audio settings the device shall use for the next SCO (audio) link established.
Firmware	6.xx
PacketType	REQ
Opcode	GAP_GET_AUDIO_CONFIG
DataLength	0

Table 5-52. Get audio settings confirm

Description	Response to the request above.	
Firmware	6.xx	
PacketType	CFM	
Opcode	GAP_GET_AUDIO_CONFIG	
DataLength	3	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS
	CodecType 1 byte	0x00: No codec available 0x01: Motorola MC145483 0x02: OKI MSM7717 0x03: PCM Slave 0x04: Winbond W681360 0x05-0xFF: Reserved.
	AirFormat 1 byte	0x00: CVSD 0x01: µ-law 0x02: A-law 0x03-0xFF: Reserved.

5.2.6.7 Change SCO Packet Type

Table 5-53. Change SCO Packet Type Command

Description	**This command is used to change the packet type used/enabled for the SCO link.**	
Firmware	6.xx	
PacketType	REQ	
Opcode	GAP_CHANGE_SCO_PACKET_TYPE	
DataLength	2	
Data	PacketType 2 bytes	0x0020: HV1 0x0040: HV2 0x0080: HV3 The packet types can be combined / or'ed together in order to enable multiple packet types.

Table 5-54. Change SCO Packet Type Confirm

Description	**Response to the request above.**	
Firmware	6.xx	
PacketType	CFM	
Opcode	GAP_CHANGE_SCO_PACKET_TYPE	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_UNSPECIFIED_ERROR

5.2.6.8 Change SCO Link Packet Type Indicator

Table 5-55. Change SCO Link Packet Type Indicator

Description	**Indicates that the packet type of the SCO link has changed. Either the local device or the remote device may have initiated the change of the packet type of the SCO link.**	
Firmware	6.xx	
PacketType	IND	
Opcode	GAP_CHANGE_SCO_PACKET_TYPE	
DataLength	3	
Data	Status 1 byte	ERROR_OK ERROR_UNSPECIFIED_ERROR ERROR_NO_CONNECTION ERROR_CONNECTION_FAILED
	PacketType 2 bytes	0x0020: HV1 0x0040: HV2 0x0080: HV3 The packet types can be combined / or'ed together in order to enable multiple packet types.

5.2.6.9 Mute

Table 5-56. Mute Command

Description	**This command is used to mute the microphone.**	
Firmware	6.10 or later	
PacketType	REQ	
Opcode	GAP_MUTE_MIC	
DataLength	1	
Data	Mute 1 byte	0x01: Mute 0x00: Un mute

Table 5-57. Mute Confirm

Description	**Response to the request above.**	
Firmware	6.10 or later	
PacketType	CFM	
Opcode	GAP_SET_VOLUME	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS

5.2.6.10 Set Volume

Table 5-58. Set Volume Command

Description	**This command is used to set the speaker volume.**	
Firmware	6.10 or later	
PacketType	REQ	
Opcode	GAP_SET_VOLUME	
DataLength	1	
Data	Volume 1 byte1	Volume control is only supported by the Motorola MC145483 codec. The following values are defined for MC145483: 0x00: Mute 0x01 - 0x1F: Step 1 (MIN) 0x20 - 0x3F: Step 2 0x40 - 0x5F: Step 3 0x60 - 0x7F: Step 4 0x80 - 0x9F: Step 5 0xA0 - 0xBF: Step 6 0xC0 - 0xDF: Step 7 0xE0 - 0xFF: Step 8 (MAX) The following values are defined for MSM7717: 0x00: Mute 0x01 - 0xFF: Max volume

Table 5-59. Set Volume Confirm

Description	Response to the request above.	
Firmware	6.10 or later	
PacketType	CFM	
Opcode	GAP_SET_VOLUME	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_SET_VOLUME_FAILED

5.2.6.11 Get Volume

Table 5-60. Get Volume Command

Description	This command is used to get the speaker volume.
Firmware	6.10 or later
PacketType	REQ
Opcode	GAP_GET_VOLUME
DataLength	0

Table 5-61. Get Volume Confirm

Description	Response to the request above.	
Firmware	6.10 or later	
PacketType	CFM	
Opcode	GAP_GET_VOLUME	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS
	Volume 1 byte	The actual volume used.

5.2.7 Default Connections

A special feature for link setup is the storage of default connections. The LMX9820A can store up to three connections in its non-volatile data memory. If the “Operation Mode” parameter in NVS is set to automatic, the LMX9820A tries to connect to every device stored within the memory. After three attempts the device will give up and try the next one or stay in idle mode.

The connections can also be established during runtime by the “Establish default connection” command.

The connections are stored in a table like structure. Each connection is entered with an index number. This index is used to that device within the Establish or Delete commands. The transparent option can only be set if only 1 connection is stored.

Table 5-62. Example Default Connection Table

Index	Local Port	BD_Addr	Remote Port	Transparent	Sniff Mode
00	01	12 34 56 78 90 12	01	no	no
01	02	98 76 54 32 10 01	03	no	no

5.2.7.1 Store Default Connection

Table 5-63. Store Default Connection Confirm Request

Description	**Stores a default connection in NVS.**	
Firmware	5.xx, 6.23 or later for Sniff Mode flag	
PacketType	REQ	
Opcode	SPP_STORE_DEFAULT_CON	
DataLength	10	
Data	Index 1 byte	Index in the default connection storage. Range 0x00 to 0x02.
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30
	RemotePort 1 byte	RFCOMM port number on remote device
	RemoteBdAddress 6 bytes	The BdAddress of the remote device
	OptionFlag 1 byte	Different options available for transparent and Sniff mode. 0x00 Transparent mode off, Normal mode. 0x01 Transparent mode on (only for point-to-point), Normal mode. 0x02 Transparent mode off, Sniff mode enabled. 0x03 Transparent mode on, Sniff mode enabled.

Table 5-64. Store Default Connection Confirm

Description	**Confirm to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	SPP_STORE_DEFAULT_CON	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_LIMIT ERROR_SPP_INVALID_PORT ERROR_SPP_MULTIPLE_TRANSPARENT ERROR_SPP_PORT_BUSY

5.2.7.2 Connect to Default Connection

Table 5-65. Connect to Default Connection Request

Description	**Connects the LMX9820A to a stored connection. Either a single or all connections can be established**
Firmware	5.xx
PacketType	REQ
Opcode	SPP_CONNECT_DEFAULT_CON
DataLength	1

Table 5-65. Connect to Default Connection Request

Data	Index 1 byte	Index in the default connection storage. Range 0x00 to 0x02. If index is set to 0xFF, all default connections will be established.

Table 5-66. Store Default Connection Confirm

Description	**Confirm to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	SPP_ CONNECT _DEFAULT_CON	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_SPP_AUTOMATIC_CONNECTIONS_PROGRESSING ERROR_LIMIT ERROR_SPP_DEFAULT_CONNECTION_NOT_STORED

5.2.7.3 Get List of Default Connections

Table 5-67. Get List of Default Connections Request

Description	**Request a list of the default connections stored in NVS**	
Firmware	5.xx	
PacketType	REQ	
Opcode	SPP_GET_LIST_DEFAULT_CON	
DataLength	1	
Data	Index 1 byte	Index in the default connection storage. Range 0x00 to 0x02.

Table 5-68. Get List of Default Connections Confirm

Description	**Confirm to the request above.**		
Firmware	5.xx		
PacketType	CFM		
Opcode	SPP_GET_LIST_DEFAULT_CON		
DataLength	11		
Data	Status 1 byte	ERROR_OK ERROR_LIMIT	
	Index 1 byte	Index in the default connection storage. Range 0x00 to 0x02.	
	EntryStatus 1 byte	Bitfield: 0000000x	0: Entry is not stored. 1: Entry is stored.
		Bitfield: 000000x0	0: Command mode. 1: Transparent mode.
		Bitfield: 00000x00	0: Normal mode 1: Sniff mode
		Bitfield: xxxxx000	Reserved.

Table 5-68. Get List of Default Connections Confirm

	RemoteBdAddress 6 bytes	The BdAddress of the remote device
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30
	RemotePort 1 byte	RFCOMM port number on remote device

5.2.7.4 Delete Default Connection

Table 5-69. Delete Default Connections Request

Description	**Deletes a stored default connection in NVS**	
Firmware	5.xx	
PacketType	REQ	
Opcode	SPP_DELETE_DEFAULT_CON	
DataLength	1	
Data	Index 1 byte	Index in the default connection storage. Range 0x00 to 0x02.

Table 5-70. Delete Default Connections Confirm

Description	**Confirm to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	SPP_DELETE_DEFAULT_CON	
DataLength	2	
Data	Index 1 byte	Index in the default connection storage. Range 0x00 to 0x02.
	Status 1 byte	0x00 OK

5.2.8 Bluetooth Low Power Modes

A bluetooth link is a based on a physically synchronized connection, which means that the devices can only communicate after successful synchronization. For this, each package also includes some synchronization information. Also a specific polling scheme is in place to keep synchronization if no traffic is necessary.

As the slave has to actively listen to packages from the master, there are different methods to decrease the necessary active receive slots on devices.

The ability to switch to those specific modes is controlled by the Link Policy. To make sure both devices support the low power mode requested, Link Policy can be set first. It will only be successful if both sides support it.

5.2.8.1 Set Default Link Policy

Table 5-71. Set Default Link Policy Command

Description	**This command is used to change the default link policy. The default link policy is set during connection setup. The default link policy setting is stored in NVS.**
Firmware	5.05 or later

Table 5-71. Set Default Link Policy Command

PacketType	REQ	
Opcode	GAP_SET_DEFAULT_LINK_POLICY	
DataLength	2	
Data	LinkPolicy 2 byte	Bitfield: 0x0001 = Master-slave switch allowed 0x0002 = Hold mode allowed 0x0004 = Sniff mode allowed 0x0008 = Park mode allowed

Table 5-72. Set Default Link Policy Confirm

Description	**Response to the request above.**	
Firmware	5.05 or later	
PacketType	CFM	
Opcode	GAP_SET_DEFAULT_LINK_POLICY	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_ILLEGAL_LINK_POLICY

5.2.8.2 Get Default Link Policy

Table 5-73. Get Default Link Policy Command

Description	**This command is used to read the default link policy setting from NVS.**
Firmware	5.05 or later
PacketType	REQ
Opcode	GET_DEFAULT_LINK_POLICY
DataLength	0

Table 5-74. Get Default Link Policy Confirm

Description	**Response to the request above.**	
Firmware	5.05 or later	
PacketType	CFM	
Opcode	GET_DEFAULT_LINK_POLICY	
DataLength	2	
Data	LinkPolicy 2 byte	Bitfield: 0x0001 = Master-slave switch allowed 0x0002 = Hold mode allowed 0x0004 = Sniff mode allowed 0x0008 = Park mode allowed

5.2.8.3 Set Link Policy

Table 5-75. Set Link Policy Request

Description	This command is used to change the current link policy setting for the given link.	
Firmware	5.xx	
PacketType	REQ	
Opcode	GAP_SET_LINK_POLICY	
DataLength	8	
Data	BdAddr 6 bytes	The Bluetooth address of the remote device of which to change the link policy settings for the link.
	LinkPolicy 2 byte	Bitfield: 0x0001 = Master-slave switch allowed 0x0002 = Hold mode allowed 0x0004 = Sniff mode allowed 0x0008 = Park mode allowed

Table 5-76. Set Link Policy Confirm

Description	Response to the request above.	
Firmware	5.xx	
PacketType	CFM	
Opcode	GAP_SET_LINK_POLICY	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_UNSPECIFIED_ERROR ERROR_INVALID_NO_OF_PARAMETERS ERROR_NO_CONNECTION ERROR_ILLEGAL_LINK_POLICY

5.2.8.4 Get Link Policy

Table 5-77. Get Link Policy Request

Description	This command is used to get the current link policy setting for the given link.	
Firmware	5.xx	
PacketType	REQ	
Opcode	GAP_GET_LINK_POLICY	
DataLength	6	
Data	BdAddr 6 bytes	The Bluetooth address of the remote device of which to get the current link policy settings for the link.

Table 5-78. Get Link Policy Confirm

Description	Response to the request above.
Firmware	5.xx
PacketType	CFM
Opcode	GAP_GET_LINK_POLICY

Table 5-78. Get Link Policy Confirm

DataLength	3	
Data	Status 1 byte	ERROR_OK ERROR_UNSPECIFIED_ERROR ERROR_INVALID_NO_OF_PARAMETERS ERROR_NO_CONNECTION
	LinkPolicy 2 byte	Bitfield: 0x0001 = Master-slave switch allowed 0x0002 = Hold mode allowed 0x0004 = Sniff mode allowed 0x0008 = Park mode allowed

5.2.8.5 Enter Sniff Mode

Command to enter the sniff mode. The command includes the maximum and minimum value for the sniff interval. After sending the command, Master and slave will calculate a reasonable sniff time and will switch into Sniff mode.

Table 5-79. Enter Sniff Mode Request

Description	**This command is used to request sniff mode on a given link with user specified parameters.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	GAP_ENTER_SNIFF_MODE	
DataLength	14	
Data	BdAddr 6 bytes	The Bluetooth address of the remote device of which to put the link in sniff mode.
	SniffMaxInterval 2 bytes	Maximum sniff interval in slots.
	SniffMinInterval 2 bytes	Minimum sniff interval in slots
	SniffAttempt 2 bytes	Number of slots the slave must listen, beginning at the sniff slot, even if it does not receive a packet with its own AM.
	SniffTimeout 2 bytes	Number of additional slots the slave must listen if it continues to receive only packets with its own AM address.

Table 5-80. Enter Sniff Mode Confirm

Description	**Response to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	GAP_ENTER_SNIFF_MODE	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_UNSPECIFIED_ERROR ERROR_INVALID_NO_OF_PARAMETERS ERROR_NO_CONNECTION

5.2.8.6 Exit Sniff Mode

Table 5-81. Exit Sniff Mode Request

Description	This command is used to exit a current sniff mode on a given link.	
Firmware	5.xx	
PacketType	REQ	
Opcode	GAP_EXIT_SNIFF_MODE	
DataLength	6	
Data	BdAddr 6 bytes	The Bluetooth address of the remote device of which to exit the current sniff mode.

Table 5-82. Exit Sniff Mode Confirm

Description	Response to the request above.	
Firmware	5.xx	
PacketType	CFM	
Opcode	GAP_EXIT_SNIFF_MODE	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_UNSPECIFIED_ERROR ERROR_INVALID_NO_OF_PARAMETERS ERROR_NO_CONNECTION

5.2.8.7 Enter Park Mode

In Park Mode the slave will lose its active member address and will not longer be part of the piconet. It will be kept synchronized by beacons within the specified interval range.

Table 5-83. Enter Park Mode Request

Description	This command is used to request park mode on a given link with user specified parameters.	
Firmware	5.xx	
PacketType	REQ	
Opcode	GAP_ENTER_PARK_MODE	
DataLength	10	
Data	BdAddr 6 bytes	The Bluetooth address of the remote device of which to put the link in park mode.
	BeaconMaxInterval 2 bytes	Acceptable longest length of the interval between beacons.
	BeaconMinInterval 2 bytes	Acceptable shortest length of the interval between beacons.

Table 5-84. Enter Park Mode Confirm

Description	Response to the request above.
Firmware	5.xx

Table 5-84. Enter Park Mode Confirm

PacketType	CFM	
Opcode	GAP_ENTER_PARK_MODE	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_UNSPECIFIED_ERROR ERROR_INVALID_NO_OF_PARAMETERS ERROR_NO_CONNECTION

5.2.8.8 Exit Park Mode

This commands forces the devices getting the parked slave back as active member of the piconet.

Table 5-85. Exit Park Mode Request

Description	**This command is used to exit a current park mode on a given link.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	GAP_EXIT_PARK_MODE	
DataLength	6	
Data	BdAddr 6 bytes	The Bluetooth address of the remote device of which to exit the current park mode.

Table 5-86. Enter Park Mode Confirm

Description	**Response to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	GAP_ENTER_PARK_MODE	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_UNSPECIFIED_ERROR ERROR_INVALID_NO_OF_PARAMETERS ERROR_NO_CONNECTION

5.2.8.9 Enter Hold Mode

Table 5-87. Enter Hold Mode Request

Description	**This command is used to request Hold mode on a given link with user specified parameters.**
Firmware	5.xx
PacketType	REQ
Opcode	GAP_ENTER_HOLD_MODE
DataLength	10

Table 5-87. Enter Hold Mode Request

Data	BdAddr 6 bytes	The Bluetooth address of the remote device of which to put the link in Hold mode.
	HoldMaxInterval 2 bytes	Maximum length of the Hold interval for which the Host may actually enter into the hold mode after negotiation with the remote device.
	HoldMinInterval 2 bytes	minimum length of the Hold interval for which the Host may actually enter into the hold mode after the negotiation with the remote device.

Table 5-88. Enter Hold Mode Confirm

Description	**Response to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	GAP_ENTER_HOLD_MODE	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_UNSPECIFIED_ERROR ERROR_INVALID_NO_OF_PARAMETERS ERROR_NO_CONNECTION

5.2.8.10 Power Save Mode Changed

In case the remote device changed the Power Mode for that link, this event will be returned by the LMX9820A.

Table 5-89. Power Save Mode Changed Indicator

Description	**This indication is sent to the host when changes the power save mode on a link occur.**	
Firmware	5.xx	
PacketType	IND	
Opcode	GAP_POWER_SAVE_MODE_CHANGED	
DataLength	8	
Data	Status 1 byte	ERROR_OK ERROR_ATTEMPT_FAILED ERROR_UNSPECIFIED_ERROR
	BdAddr 6 bytes	The Bluetooth address of the remote device for which the power save mode has changed on the link.
	Mode 1 byte	0x00 = Active mode (Left power save mode) 0x01 = Hold mode (Hold mode entered) 0x02 = Sniff mode (Sniff mode entered) 0x03 = Park mode (Park mode entered)

5.2.9 Wake-Up Functionality

Wake-up functionality allows to stop the communication between host and LMX9820A in order to save power on both devices. The wake-up itself is done via the RTS/CTS pins or the Host_WU pin. For a detailed description of the hardware wake-up please see also Section 3.7 “Wake up functionality” on page 53.

The communication between host and LMX9820A can be stopped by sending the LMX9820A the Disable Transport Layer command. The command can be sent any time. The LMX9820A will continue its current operation status and try to wake up the host as soon as the interface is needed.

5.2.9.1 Disable Transport Layer

Table 5-90. Disable Transport Layer Command

Description	**This Command disables the transport layer and thereby allowing powersaving.**	
Firmware	5.10 or later	
PacketType	REQ	
Opcode	DISABLE_TL	
DataLength	0	

Table 5-91. Disable Transport Layer Confirm

Description	**Response to the request above.**	
Firmware	5.10 or later	
PacketType	CFM	
Opcode	DISABLE_TL	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_COMMAND_DISALLOWED

5.2.9.2 Transport layer enabled indicator

Table 5-92. Transport Layer enabled indicator

Description	**This indication is sent when the transport layer is re-enabled. Transport layer is re-enabled by pulling RTS signal of the host or an event is ready to be sent from the Simply Blue to the host, normally due to an air-interface event.**	
Firmware	5.10 or later	
PacketType	IND	
Opcode	TL_ENABLED	
DataLength	1	
Data	Status 1 byte	ERROR_OK

5.2.10 SPP Port Configuration

An active SPP link appears as a virtual serial port connection between two devices. As any other serial connection it has different settings for that “virtual” serial port. The following settings enable the host to change specific port settings on that virtual port. The reference for all commands is the local RFcomm port the link has been set up with.

5.2.10.1 SPP Set Port Configuration

Table 5-93. SPP Set Port Config Request

Description	**Write the configuration for the SPP port. This is "virtual" settings for the air connection, not settings for the LMX9820A serial port. Note: The baudrate in this configuration has no impact on the throughput on the bluetooth link itself or the LMX9820A. It is not used by the RFCOMM layer.**
Firmware	5.xx
PacketType	REQ

Table 5-93. SPP Set Port Config Request

Opcode	SPP_SET_PORT_CONFIG			
DataLength	6			
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30		
	BaudRate 1 byte	Baudrate 0x00 = 2400 baud 0x01 = 4800 baud 0x02 = 7200 baud 0x03 = 9600 baud 0x04 = 19200 baud 0x05 = 38400 baud 0x06 = 57600 baud 0x07 = 115200 baud 0x08 = 230400 baud **Note: The baudrate in this configuration has no impact on the throughput on the bluetooth link itself or the LMX9820A. It is not used by the RFCOMM layer. The parameter might be used as informative parameter on the application level.**		
	Portsettings 1 byte	Bitfield: 000000XX	Number of databits	00=5 bits 01=6 bits 02=7 bits 03=8 bits
		Bitfield: 00000X00	Number of stopbits	0 = 1 1 = 1.5
		Bitfield: 0000X000	Parity	0 = No parity 1 = Parity
		Bitfield: 00XX0000	ParityType	00=ODD 01=EVEN 02=MARK 03=SPACE
		Bitfield: XX000000	Reserved	
	FlowControl 1 byte	Bitfield: 0000000X	XonXoffOnInput	0=Disable 1=Enable
		Bitfield: 000000X0	XonXoffOnOutput	0=Disable 1=Enable
		Bitfield: 00000X00	RtrOnInput	0=Disable 1=Enable
		Bitfield: 0000X000	RtrOnOutput	0=Disable 1=Enable
		Bitfield: 000X0000	RtcOnInput	0=Disable 1=Enable
		Bitfield: 00X00000	RtcOnOutput	0=Disable 1=Enable
		Bitfield: XX000000	Reserved	
	XonChar 1 byte	Char used for Xon, if Xon/Xoff flowcontrol is used.		
	XoffChar 1 byte	Char used for Xoff, if Xon/Xoff flowcontrol is used.		

Table 5-94. SPP Set Port Configuration Confirm

Description	**Confirm to the request above.**

Table 5-94. SPP Set Port Configuration Confirm

Firmware	5.xx	
PacketType	CFM	
Opcode	SPP_SET_PORT_CONFIG	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_UNSPECIFIED_ERROR ERROR_UNSPECIFIED_ERROR ERROR_SPP_INVALID_PORT ERROR_SPP_PORT_NOT_OPEN ERROR_UART_SPEED_OUT_OF_RANGE
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

5.2.10.2 SPP Get Port Configuration

Table 5-95. SPP Get Port Configuration Request

Description	**Read the configuration for the SPP port. This is "virtual" settings for the air connection, not settings for the LMX9820A serial port.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	SPP_GET_PORT_CONFIG	
DataLength	1	
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

Table 5-96. SPP Get Port Configuration Confirm

Description	**Confirm to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	SPP_GET_PORT_CONFIG	
DataLength	7	
Data	Status 1 byte	ERROR_OK ERROR_UNSPECIFIED_ERROR ERROR_SPP_INVALID_PORT ERROR_SPP_PORT_NOT_OPEN
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30
	BaudRate 1 byte	Baudrate 0x00 = 2400 baud 0x01 = 4800 baud 0x02 = 7200 baud 0x03 = 9600 baud 0x04 = 19200 baud 0x05 = 38400 baud 0x06 = 57600 baud 0x07 = 115200 baud 0x08 = 230400 baud

Table 5-96. SPP Get Port Configuration Confirm

	Portsettings 1 byte	Bitfield: 000000XX	Number of databits	00=5 bits 01=6 bits 02=7 bits 03=8 bits
		Bitfield: 00000X00	Number of stopbits	0 = 1 1 = 1.5
		Bitfield: 0000X000	Parity	0 = No parity 1 = Parity
		Bitfield: 00XX0000	ParityType	00=ODD 01=EVEN 02=MARK 03=SPACE
		Bitfield: XX000000	Reserved	
	FlowControl 1 byte	Bitfield: 0000000X	XonXoffOnInput	0=Disable 1=Enable
		Bitfield: 000000X0	XonXoffOnOutput	0=Disable 1=Enable
		Bitfield: 00000X00	RtrOnInput	0=Disable 1=Enable
		Bitfield: 0000X000	RtrOnOutput	0=Disable 1=Enable
		Bitfield: 000X0000	RtcOnInput	0=Disable 1=Enable
		Bitfield: 00X00000	RtcOnOutput	0=Disable 1=Enable
		Bitfield: XX000000	Reserved	
	XonChar 1 byte	Char used for Xon, if Xon/Xoff flowcontrol is used.		
	XoffChar 1 byte	Char used for Xoff, if Xon/Xoff flowcontrol is used.		

5.2.10.3 SPP Port Configuration Changed Indicator

Table 5-97. SPP Port Configuration Changed Indicator

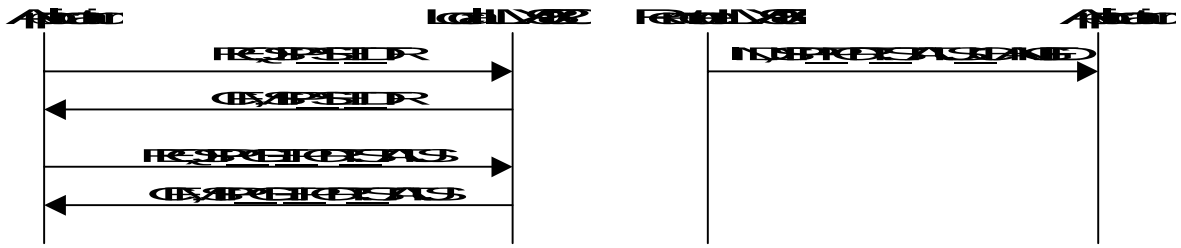
Description	**Send from the LMX9820A when remote device has changed the port configuration.**	
Firmware	5.xx	
PacketType	IND	
Opcode	SPP_PORT_CONFIG_CHANGED	
DataLength	6	
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

Table 5-97. SPP Port Configuration Changed Indicator

	BaudRate 1 byte	Baudrate 0x00 = 2400 baud 0x01 = 4800 baud 0x02 = 7200 baud 0x03 = 9600 baud 0x04 = 19200 baud 0x05 = 38400 baud 0x06 = 57600 baud 0x07 = 115200 baud 0x08 = 230400 baud		
	Portsettings 1 byte	Bitfield: 000000XX	Number of databits	00=5 bits 01=6 bits 02=7 bits 03=8 bits
		Bitfield: 00000X00	Number of stopbits	0 = 1 1 = 1.5
		Bitfield: 0000X000	Parity	0 = No parity 1 = Parity
		Bitfield: 00XX0000	ParityType	00=ODD 01=EVEN 02=MARK 03=SPACE
		Bitfield: XX000000	Reserved	
	FlowControl 1 byte	Bitfield: 0000000X	XonXoffOnInput	0=Disable 1=Enable
		Bitfield: 000000X0	XonXoffOnOutput	0=Disable 1=Enable
		Bitfield: 00000X00	RtrOnInput	0=Disable 1=Enable
		Bitfield: 0000X000	RtrOnOutput	0=Disable 1=Enable
		Bitfield: 000X0000	RtcOnInput	0=Disable 1=Enable
		Bitfield: 00X00000	RtcOnOutput	0=Disable 1=Enable
		Bitfield: XX000000	Reserved	
	XonChar 1 byte	Char used for Xon, if Xon/Xoff flowcontrol is used.		
	XoffChar 1 byte	Char used for Xoff, if Xon/Xoff flowcontrol is used.		

5.2.11 SPP Port Status

An active SPP link allows signalling of modem status and line status over the bluetooth link. The following commands and events describe how to change or get the status of those line parameters.



5.2.11.1 SPP Get Port Status

Description	Get the current state of the modem status and line status. This command resets the value (to 0) of the following members of the port status: OverrunError ParityError FramingError BreakLength The value of DSR and CTS are only changed when new values are received from the remote device!	
Firmware	5.xx	
PacketType	REQ	
Opcode	SPP_GET_PORT_STATUS	
DataLength	1	
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

Table 5-98. SPP Get Port Status Confirm

Description	Confirm to the request above.			
Firmware	5.xx			
PacketType	CFM			
Opcode	SPP_GET_PORT_STATUS			
DataLength	5			
Data	Status 1 byte	ERROR_OK ERROR_UNSPECIFIED_ERROR ERROR_SPP_INVALID_PORT ERROR_SPP_PORT_NOT_OPEN		
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30		
	PortStatus 1 byte	Bitfield: 0000000X	DTR	0 = Low 1 = High
		Bitfield: 000000X0	RTS	0 = Low 1 = High
		Bitfield: 00000X00	DSR	0 = Low 1 = High
		Bitfield: 0000X000	CTS	0 = Low 1 = High

Table 5-98. SPP Get Port Status Confirm

		Bitfield: 000X0000	Overrun Error	0 = No Error 1 = Overrun Error
		Bitfield: 00X00000	Parity Error	0 = No Error 1 = Parity Error
		Bitfield: 0X000000	Framing Error	0 = No Error 1 = Framing Error
		Bitfield:X0000000	DLC established	0 = No DLC 1 = DLC is available
	Break Length 2 bytes	The length in ms of the detected break. The value 0 is used to indicate that no break has been detected.		

5.2.11.2 SPP Port Set DTR

Table 5-99. SPP Port Set DTR Request

Description	**This command sets the state of the DTR bit. Since RFCOMM acts as a "null modem" where DTR and DSR are connected, the remote device will see this as a change of the state of the DSR signal.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	SPP_PORT_SET_DTR	
DataLength	2	
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30
	State	0: False. 1: True.

Table 5-100. SPP Port Set DTR Confirm

Description	**Confirm to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	SPP_PORT_SET_DTR	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_ILLEGAL_STATE_VALUE ERROR_SPP_INVALID_PORT ERROR_SPP_PORT_NOT_OPEN
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

5.2.11.3 SPP Port Set RTS

Table 5-101. SPP Port Set RTS Request

Description	**This command sets the state of the RTS bit. Since RFCOMM acts as a "null modem" where RTS and CTS are connected, the remote device will see this as a change of the state of the CTS signal.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	SPP_PORT_SET_RTS	
DataLength	2	
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30
	State	0: False. 1: True.

Table 5-102. SPP Port Set RTS Confirm

Description	**Confirm to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	SPP_PORT_SET_RTS	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_ILLEGAL_STATE_VALUE ERROR_SPP_INVALID_PORT ERROR_SPP_PORT_NOT_OPEN
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

5.2.11.4 SPP Port Set BREAK

Table 5-103. SPP Set Port Break Request

Description	**This command indicates that the host has detected a break.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	SPP_PORT_BREAK	
DataLength	2	
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30
	BreakLength	The length of the break in ms.

Table 5-104. SPP Set Port Break Confirm

Description	**Confirm to the request above.**
Firmware	5.xx

Table 5-104. SPP Set Port Break Confirm

PacketType	CFM	
Opcode	SPP_PORT_BREAK	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_SPP_INVALID_PORT ERROR_SPP_PORT_NOT_OPEN
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

5.2.11.5 Set Overrun Error

Table 5-105. SPP Set Overrun Error Request

Description	**This command is used to indicate that the host has detected an overrun error. This command is only used if the "application" is an UART / UART driver.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	SPP_PORT_OVERRUN_ERROR	
DataLength	1	
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

Table 5-106. SPP Port Overrun Error Confirm

Description	**Confirm to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	SPP_PORT_OVERRUN_ERROR	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_SPP_INVALID_PORT ERROR_SPP_PORT_NOT_OPEN
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

5.2.11.6 SPP Set Parity Error

Table 5-107. SPP Port Parity Error Request

Description	**This command is used to indicate that the host has detected a parity error. This command is only used if the "application" is an UART / UART driver.**
Firmware	5.xx
PacketType	REQ
Opcode	SPP_PORT_PARITY_ERROR
DataLength	1

Table 5-107. SPP Port Parity Error Request

Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

Table 5-108. SPP Port Parity Error Confirm

Description	**Confirm to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	SPP_PORT_PARITY_ERROR	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_SPP_INVALID_PORT ERROR_SPP_PORT_NOT_OPEN
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

5.2.11.7 SPP Set Framing Error

Table 5-109. SPP Port Framing Error Request

Description	**This command is used to indicate that the host has detected a framing error. This command is only used if the "application" is an UART / UART driver.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	SPP_PORT_FRAMING_ERROR	
DataLength	1	
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

Table 5-110. SPP Port Framing Errors Confirm

Description	**Confirm to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	SPP_PORT_FRAMING_ERROR	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_SPP_INVALID_PORT ERROR_SPP_PORT_NOT_OPEN
	LocalPort 1 byte	Local RFCOMM port number. Range 1-30

5.2.11.8 SPP Port Status Changed Indicator

Table 5-111. SPP Port Status Changed Indicator

Description	**Send from the LMX9820A when remote device has changed the port.**

Table 5-111. SPP Port Status Changed Indicator

Firmware	5.xx			
PacketType	IND			
Opcode	SPP_PORT_STATUS_CHANGED			
DataLength	4			
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30		
	PortStatus 1 byte	Bitfield: 0000000X	DTR	0 = Low 1 = High
		Bitfield: 000000X0	RTS	0 = Low 1 = High
		Bitfield: 00000X00	DSR	0 = Low 1 = High
		Bitfield: 0000X000	CTS	0 = Low 1 = High
		Bitfield: 000X0000	Overrun Error	0 = No Error 1 = Overrun Error
		Bitfield: 00X00000	Parity Error	0 = No Error 1 = Parity Error
		Bitfield: 0X000000	Framing Error	0 = No Error 1 = Framing Error
		Bitfield:X0000000	DLC established	0 = No DLC 1 = DLC is available
	Break Length 2 bytes	The length in ms of the detected break. The value 0 is used to indicate that no break has been detected.		

5.2.12 RFcomm Channels to open

Each Service within the Service Database is registered to a specific RFComm channel. The configuration, which ports will be opened and initialized can be configured with the command Set Ports to Open.

If a RFComm has not been opened, it is not connectable from outside or can not be used for setting up a link.

5.2.12.1 Set Ports To Open

Table 5-112. SDP Set Ports to Open Request

Description	**This command will change which RFCOMM ports the LMX9820A will open both at start-up and runtime.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	SET_PORTS_TO_OPEN	
DataLength	4	
Data	PORTS 4 Bytes	This field is a 32-bit mask indicating which RFCOMM ports the LMX9820A has to open. Bit 30 and 31 must be set to 0. Bit 0 is RFCOMM port 1 and bit 29 is port 30 e.g. if this field has the value 0x00000007, port 1 to 3 will be opened. All other ports will be closed if open.

Table 5-113. SDP Set Ports to Open Confirm

Description	Confirms the request above	
Firmware	5.xx	
PacketType	CFM	
Opcode	PORTS_TO_OPEN	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_PORT ERROR_INVALID_NO_OF_PARAMETERS

5.2.12.2 Get Ports To Open

Table 5-114. SDP Get Ports to open Request

Description	This command will get the value of which RFCOMM ports the LMX9820A will open both at start-up and runtime.	
Firmware	5.xx	
PacketType	REQ	
Opcode	GET_PORTS_TO_OPEN	
DataLength	0	
Data		

Table 5-115. SDP Get Ports to open Confirm

Description	Confirms the request above	
Firmware	5.xx	
PacketType	CFM	
Opcode	PORTS_TO_OPEN	
DataLength	5	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS
	PORTS 4 Bytes	This field is a 32-bit mask indicating which RFCOMM ports the LMX9820A has to open. Bit 30 and 31 must be set to 0. Bit 0 is RFCOMM port 1 and bit 29 is port 30 e.g. if this field has the value 0x00000007, port 1 to 3 will be opened.

5.2.13 Local Service Database Configuration

The LMX9820A allows the modification of the Local Service Discovery Database. On default, the service database contains one entry configured at RFComm port 1 for a Serial Port Profile, Authentication and Encryption enabled.

If the application needs to open a second connection to another device, the dedicated service has to be registered into the database and the registered RFComm port has to be opened (see Section 5.2.12.1). The service entry in general includes information about the name of the service, which appears on a remote device after browsing, the port number and security settings.

Each registered service entry itself can be enabled or disabled. This allows to have different services registered to one specific com port enabling and disabling them by needs.

In addition the command set allows to add besides SPP also DUN, FAX, OPP, FTP and SYNC profiles by one command. Each with the profile specific settings needed.

The service records are stored in a database like system within the NVS, so they are still available after reset. The storage of an entry is confirmed by the LMX9820A with a specific record identifier. This identifier is needed for the Enabling or Disabling command to address those specific entries.

5.2.13.1 SDP Enable SDP Record

Table 5-116. SDP Enable SDP Record Request

Description	This command is used to enable/disable stored SDP records in the LMX9820A.	
Firmware	5.xx	
PacketType	REQ	
Opcode	ENABLE_SDP_RECORD	
DataLength	2	
Data	State 1 byte	The new state of the SDP record. 0x00 Disable the record. 0x01 Enable the record.
	Identifier 1 byte	The identifier of the service record to address. This will be received when the record was stored in the LMX9820A.

Table 5-117. SDP Enable SDP Record Confirm

Description	Confirms the request above	
Firmware	5.xx	
PacketType	CFM	
Opcode	ENABLE_SDP_RECORD	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_ILLEGAL_STATE_VALUE ERROR_IDENTIFIER_OUT_OF_RANGE ERROR_RECORD_ALREADY_IN_SELECTED_STATE ERROR_IDENTIFIER_NOT_IN_USE ERROR_INVALID_NO_OF_PARAMETERS
	Identifier 1 byte	The identifier received when the record was stored in the LMX9820A.

5.2.13.2 SDP Delete All SDP Records

Table 5-118. SDP Delete SDP Record Request

Description	This command is used to delete all stored SDP records in the LMX9820A.	
Firmware	5.xx	
PacketType	REQ	
Opcode	DELETE_SDP_RECORDS	
DataLength	0	
Data	None	

Table 5-119. SDP Delete SDP Record Confirm

Description	Confirms the request above	
Firmware	5.xx	
PacketType	CFM	
Opcode	DELETE_SDP_RECORDS	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS

5.2.13.3 SDP Store SPP Record

Table 5-120. SDP Store SPP Record Request

Description	This command will create a new SPP record in the local SDP database, stored in NVS.	
Firmware	5.xx	
PacketType	REQ	
Opcode	STORE_SPP_RECORD	
DataLength	4 + NameLength	
Data	LocalPort 1 byte	The local RFCOMM port used by this service. Range 1 to 30.
	Authentication 1 byte	0x00 No authentication requirements 0x02 Authentication is only required for this profile for incoming connections. 0x20 Authentication is only required for this profile for outgoing connections. 0x22 Authentication is required for this profile for connections in both directions
	Encryption 1 byte	0x00 No encryption requirements 0x04 Encryption is only required for this profile for incoming connections. 0x40 Encryption is only required for this profile for outgoing connections. 0x44 Encryption is required for this profile for connections in both directions
	NameLength 1 byte	The number of bytes in the service name including the NULL termination.
	ServiceName <NameLength> bytes	A text string containing a human-readable name for the service. Must be NULL terminated and must maximum contain 15 bytes. The string is placed at language base 0x0100.

Table 5-121. SDP Store SPP Record Confirm

Description	Confirms the request above
Firmware	5.xx
PacketType	CFM
Opcode	STORE_SPP_RECORD
DataLength	2

Table 5-121. SDP Store SPP Record Confirm

Data	Status 1 byte	ERROR_OK ERROR_INVALID_PORT ERROR_NAME_TOO_LONG ERROR_INVALID_AUTHENTICATION_VALUE ERROR_INVALID_ENCRYPTION_VALUE ERROR_MAXIMUM_NO_OF_SERVICE_RECORDS_REACHED ERROR_WRITING_TO_NVS
	Identifier 1 byte	A unique identifier, which must be used when dynamic enabling/disabling the record.

5.2.13.4 SDP Store DUN Record

Table 5-122. SDP Store DUN Record Request

Description	**This command will create a new DUN record in the local SDP database, stored in NVS.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	STORE_DUN_RECORD	
DataLength	4 + <Length>	
Data	LocalPort 1 byte	The local RFCOMM port used by this service. Range 1 to 30.
	Authentication 1 byte	0x00 No authentication requirements 0x02 Authentication is only required for this profile for incoming connections. 0x20 Authentication is only required for this profile for outgoing connections. 0x22 Authentication is required for this profile for connections in both directions
	Encryption 1 byte	0x00 No encryption requirements 0x04 Encryption is only required for this profile for incoming connections. 0x40 Encryption is only required for this profile for outgoing connections. 0x44 Encryption is required for this profile for connections in both directions
	NameLength 1 byte	The number of bytes in the service name including the NULL termination.
	ServiceName <NameLength> Bytes	A text string containing a human-readable name for the service. Must be NULL terminated and must maximum contain 15 bytes. The string is placed at language base 0x0100.

Table 5-123. SDP Store DUN Record Confirm

Description	**Confirms the request above**
Firmware	5.xx
PacketType	CFM
Opcode	STORE_DUN_RECORD
DataLength	2

Table 5-123. SDP Store DUN Record Confirm

Data	Status 1 byte	ERROR_OK ERROR_INVALID_PORT ERROR_NAME_TOO_LONG ERROR_INVALID_AUTHENTICATION_VALUE ERROR_INVALID_ENCRYPTION_VALUE ERROR_MAXIMUM_NO_OF_SERVICE_RECORDS_REACHED ERROR_WRITING_TO_NVS
	Identifier 1 byte	A unique identifier, which must be used when dynamic enabling/disabling the record.

5.2.13.5 SDP Store FAX Record

Table 5-124. SDP Store Fax Record Request

Description	**This command will create a new FAX record in the local SDP database, stored in NVS.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	STORE_FAX_RECORD	
DataLength	7 + <NameLength>	
Data	LocalPort 1 byte	The local RFCOMM port used by this service. Range 1 to 30.
	FaxClass1 1 byte	Fax Class 1 Support. 0x00 = False 0x01 = True
	FaxClass20 1 byte	Fax Class 2.0 Support. 0x00 = False 0x01 = True
	FaxClass2 1 byte	Fax Class 2 Support. 0x00 = False 0x01 = True
	Authentication 1 byte	0x00 No authentication requirements 0x02 Authentication is only required for this profile for incoming connections. 0x20 Authentication is only required for this profile for outgoing connections. 0x22 Authentication is required for this profile for connections in both directions
	Encryption 1 byte	0x00 No encryption requirements 0x04 Encryption is only required for this profile for incoming connections. 0x40 Encryption is only required for this profile for outgoing connections. 0x44 Encryption is required for this profile for connections in both directions
	NameLength 1 byte	The number of bytes in the service name including the NULL termination.

Table 5-124. SDP Store Fax Record Request

	ServiceName <NameLength> Bytes	A text string containing a human-readable name for the service. Must be NULL terminated. The string is placed at language base 0x0100.

Table 5-125. SDP Store FAX Record Confirm

Description	Confirms the request above	
Firmware	5.xx	
PacketType	CFM	
Opcode	STORE_FAX_RECORD	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_PORT ERROR_NAME_TOO_LONG ERROR_INVALID_AUTHENTICATION_VALUE ERROR_INVALID_ENCRYPTION_VALUE ERROR_MAXIMUM_NO_OF_SERVICE_RECORDS_REACHED ERROR_WRITING_TO_NVS ERROR_INVALID_SUPPORTED_FAXCLASS_VALUE
	Identifier 1 byte	A unique identifier, which must be used when dynamic enabling/disabling the record.

5.2.13.6 SDP Store OPP Record

Table 5-126. SDP Store OPP Record Request

Description	This command will create a new OPP record in the local SDP database, stored in NVS.	
Firmware	5.xx	
PacketType	REQ	
Opcode	STORE_OPP_RECORD	
DataLength	5 + <NameLength> + <NoOfFormats>	
Data	LocalPort 1 byte	The local RFCOMM port used by this service. Range 1 to 30.
	Authentication 1 byte	0x00 No authentication requirements 0x02 Authentication is only required for this profile for incoming connections. 0x20 Authentication is only required for this profile for outgoing connections. 0x22 Authentication is required for this profile for connections in both directions
	Encryption 1 byte	0x00 No encryption requirements 0x04 Encryption is only required for this profile for incoming connections. 0x40 Encryption is only required for this profile for outgoing connections. 0x44 Encryption is required for this profile for connections in both directions

Table 5-126. SDP Store OPP Record Request

	NoOfFormats 1 byte	The number of bytes in the supported format field. Range 1 to 7.
	SupportedFormats <NoOfFormats> bytes	A list of supported formats, see profile specification for further details.
	NameLength 1 byte	The number of bytes in the service name including the NULL termination.
	ServiceName <NameLength> Bytes	A text string containing a human-readable name for the service. Must be NULL terminated. The string is placed at language base 0x0100.

Table 5-127. SDP Store OPP Record Confirm

Description	**Confirms the request above**	
Firmware	5.xx	
PacketType	CFM	
Opcode	STORE_OPP_RECORD	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_PORT ERROR_NAME_TOO_LONG ERROR_INVALID_AUTHENTICATION_VALUE ERROR_INVALID_ENCRYPTION_VALUE ERROR_MAXIMUM_NO_OF_SERVICE_RECORDS_REACHED ERROR_WRITING_TO_NVS ERROR_TOO_ MANY_SUPPORTED_FORMATS
	Identifier 1 byte	A unique identifier, which must be used when dynamic enabling/disabling the record.

5.2.13.7 SDP Store FTP Record

Table 5-128. SDP Store FTP Record Request

Description	**This command will create a new FTP record in the local SDP database, stored in NVS.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	STORE_FTP_RECORD	
DataLength	4 + <NameLength>	
Data	LocalPort 1 byte	The local RFCOMM port used by this service. Range 1 to 30.
	Authentication 1 byte	0x00 No authentication requirements 0x02 Authentication is only required for this profile for incoming connections. 0x20 Authentication is only required for this profile for outgoing connections. 0x22 Authentication is required for this profile for connections in both directions

Table 5-128. SDP Store FTP Record Request

	Encryption 1 byte	0x00 No encryption requirements 0x04 Encryption is only required for this profile for incoming connections. 0x40 Encryption is only required for this profile for outgoing connections. 0x44 Encryption is required for this profile for connections in both directions
	NameLength 1 byte	The number of bytes in the service name including the NULL termination.
	ServiceName <Length> Bytes	A text string containing a human-readable name for the service. Must be NULL terminated. The string is placed at language base 0x0100.

Table 5-129. SDP Store FTP Record Confirm

Description	**Confirms the request above**	
Firmware	5.xx	
PacketType	CFM	
Opcode	STORE_FTP_RECORD	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_PORT ERROR_NAME_TOO_LONG ERROR_INVALID_AUTHENTICATION_VALUE ERROR_INVALID_ENCRYPTION_VALUE ERROR_MAXIMUM_NO_OF_SERVICE_RECORDS_REACHED ERROR_WRITING_TO_NVS
	Identifier 1 byte	A unique identifier, which must be used when dynamic enabling/disabling the record.

5.2.13.8 SDP Store IrMCSync Record

Table 5-130. SDP Store SYNC Record Request

Description	**This command will create a new IrMCSync record in the local SDP database, stored in NVS.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	STORE_SYNC_RECORD	
DataLength	5+ <NameLength> + <NoOfDataStores>	
Data	LocalPort 1 byte	The local RFCOMM port used by this service. Range 1 to 30.

Table 5-130. SDP Store SYNC Record Request

	Authentication 1 byte	0x00 No authentication requirements 0x02 Authentication is only required for this profile for incoming connections. 0x20 Authentication is only required for this profile for outgoing connections. 0x22 Authentication is required for this profile for connections in both directions
	Encryption 1 byte	0x00 No encryption requirements 0x04 Encryption is only required for this profile for incoming connections. 0x40 Encryption is only required for this profile for outgoing connections. 0x44 Encryption is required for this profile for connections in both directions
	NoOfDataStores 1 byte	The number of bytes in the supported data stores. Range 1 to 5
	SupportedDataStores <NoOfDataStores> bytes	A list of supported data stores, see profile specification for further details.
	NameLength 1 byte	The number of bytes in the service name including the NULL termination.
	ServiceName <NameLength> Bytes	A text string containing a human-readable name for the service. Must be NULL terminated. The string is placed at language base 0x0100.

Table 5-131. SDP Store SYNC Record Confirm

Description	Confirms the request above	
Firmware	5.xx	
PacketType	CFM	
Opcode	STORE_SYNC_RECORD	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_PORT ERROR_NAME_TOO_LONG ERROR_INVALID_AUTHENTICATION_VALUE ERROR_INVALID_ENCRYPTION_VALUE ERROR_MAXIMUM_NO_OF_SERVICE_RECORDS_REACHED ERROR_WRITING_TO_NVS ERROR_TOO_ MANY_DATASTORES
	Identifier 1 byte	A unique identifier, which must be used when dynamic enabling/disabling the record.

5.2.13.9 Store Generic SDP Record

The generic SDP record command can be used to create any SDP record. In order to use this command, it is necessary to have detailed knowledge about the SDP protocol. For Simply Blue a special dialog has been created in Simply Blue Commander version 1.3.0.3 or later, to generate the appropriate key. Please refer to the “Simply Blue Commander” application.

Table 5-132. Store Generic SDP Record

Description	**This command will create a new service record in the local SDP database, store in the NVS.**	
Firmware	6.xx	
PacketType	REQ	
Opcode	STORE_SDP_RECORD	
DataLength	5 + <SdpRecordLength>	
Data	LocalPort 1 byte	The local RFCOMM port used by this service. Range 1 to 30.
	Authentication 1 byte	0x00 No authentication requirements. 0x02 Authentication is only required for this profile for incoming connections. 0x20 Authentication is only required for this profile for outgoing connections. 0x22 Authentication is required for this profile for connections in both directions
	Encryption 1 byte	0x00 No encryption requirements 0x04 Encryption is only required for this profile for incoming connections. 0x40 Encryption is only required for this profile for outgoing connections. 0x44 Encryption is required for this profile for connections in both directions
	SdpRecordLength 2 bytes	The number of bytes in the SDP record.
	SdpRecord <SdpRecord-Length> Bytes	The SDP record data formatted as: < uint16:AttributeId, SdpDataElement:AttributeData, uint16:AttributeId, SdpDataElement:AttributeData, …>.

Table 5-133. Store Generic SDP Record Confirm

Description	**Confirms the request above**	
Firmware	6.xx	
PacketType	CFM	
Opcode	STORE_SDP_RECORD	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_SDP_RECORD ERROR_MAXIMUM_NO_OF_SERVICE_RECORDS_REACHED ERROR_WRITING_TO_NVS
	Identifier 1 byte	A unique identifier, which must be used when dynamic enabling/disabling the record.

5.2.14 Local Bluetooth Settings

The LMX9820A uses the NVS Data memory to store all parameters specific for the local device. All bluetooth settings are checked during runtime and read out directly from this memory area.

5.2.14.1 Read Local Name

Table 5-134. Read Local Name Request

Description	Request the user-friendly name for the local Bluetooth device.	
Firmware	5.xx	
PacketType	REQ	
Opcode	GAP_READ_LOCAL_NAME	
DataLength	0	
Data	None	

Table 5-135. Read Local Name Confirm

Description	Confirm to the request above.	
Firmware	5.xx	
PacketType	CFM	
Opcode	GAP_READ_LOCAL_NAME	
DataLength	2 + NameLength	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_UNKNOWN_ERROR
	NameLength 1 byte	Number of bytes in device name
	DeviceName NameLength bytes	The user-friendly name of the local device. The string is NULL terminated. Max length is 30 bytes.

5.2.14.2 Write Local Name

Table 5-136. Write Local Name Request

Description	Change the user-friendly name for the local Bluetooth device. The name is stored in NVS	
Firmware	5.xx	
PacketType	REQ	
Opcode	GAP_WRITE_LOCAL_NAME	
DataLength	1+ NameLength	
Data	NameLength 1 byte	Number of bytes in device name
	DeviceName Length bytes	The user-friendly name of the local device. (String must be NULL terminated). Max length is 30 bytes.

Table 5-137. Write Local Name Confirm

Description	Confirm to the request above.
Firmware	5.xx
PacketType	CFM
Opcode	GAP_WRITE_LOCAL_NAME

Table 5-137. Write Local Name Confirm

DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_NAME_TOO_LONG ERROR_INVALID_NO_OF_PARAMETERS

5.2.14.3 Read Local Bluetooth Address

Table 5-138. Read Local BD_Addr Request

Description	**Read the Bluetooth device address of the local Bluetooth device.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	GAP_READ_LOCAL_BDA	
DataLength	0	
Data	None	

Table 5-139. Read Local BD_Addr Confirm

Description	**Confirm to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	GAP_READ_LOCAL_BDA	
DataLength	7	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_UNKNOWN_ERROR
	BdAddr 6 byte	Bluetooth device address for the local device

5.2.14.4 Change Local Bluetooth Address

Table 5-140. Change Local BD_Address Request

Description	**The LMX9820A will change the local BdAddress with the one specified in this command. If the new BdAddress has the value 0xFFFFFFFFFFFF, the LMX9820A will use the default Bluetooth address specified by NSC and return ERROR_RESET_TO_NSC_BDADDRESS. A delay of at least 1 sec should be inserted between this command and the reset operation.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	CHANGE_LOCAL_BDADDRESS	
DataLength	6	
Data	BdAddress 6 Bytes	The new Bluetooth address for the local device. The address is stored in NVS.

Table 5-141. Change Local BD_Addr Confirm

Description	**Confirms the request above**

Table 5-141. Change Local BD_Addr Confirm

Firmware	5.xx	
PacketType	CFM	
Opcode	CHANGE_LOCAL_BDADDRESS	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_RESET_TO_NSC_BDADDRESS ERROR_INVALID_NO_OF_PARAMETERS

5.2.14.5 Store Class of Device

Table 5-142. Store Class of Device Request

Description	**This command will store the class of device for the LMX9820A in NVS. The proper value for the class of device parameter is specified by the Bluetooth SIG.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	STORE_CLASS_OF_DEVICE	
DataLength	3	
Data	ClassOfDevice 3 bytes	The class of device value to be stored in NVS.

Table 5-143. Store Class of Device Confirm

Description	**Confirms the request above**	
Firmware	5.xx	
PacketType	CFM	
Opcode	STORE_CLASS_OF_DEVICE	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS

5.2.14.6 Set Scan Mode

Table 5-144. Set Scan Mode Request

Description	**Change the Bluetooth scan mode. Automatic limited discoverable mode automatically toggles between general and limited inquiry scanning. This mode defined by the Bluetooth GAP profile specification, refer to part K.1, section 6.2.1. The automatic Limited discoverable mode times out after 60 sec. At this point the LMX9820A sends the GAP_SET_SCANMODE indication and resets page and inquiry scan settings to the value that was stored before the automatic limited discoverable mode was entered. The Connectability mode and discoverability modes are stored in NVS and restored during startup. An exception are the limited discoverable mode and automatic limited discoverable modes are selected, in this case neither connectability mode or discoverability mode are stored in NVS.**
Firmware	5.xx
PacketType	REQ
Opcode	GAP_SET_SCANMODE

Table 5-144. Set Scan Mode Request

DataLength	2	
Data	Connectability 1 byte	0 = Not connectable 1 = Connectable
	Discoverability 1 byte	0 = Non discoverable 1 = General discoverable 2 = Limited discoverable 3 = Automatic limited discoverable mode (see Bluetooth GAP profile)

Table 5-145. Set Scan Mode Confirm

Description	**Confirm to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	GAP_SET_SCANMODE	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_CONNECTABILITY_PARAMETER ERROR_INVALID_DISCOVERABILITY_PARAMETER ERROR_INVALID_NO_OF_PARAMETERS ERROR_UNKNOWN_ERROR

5.2.14.7 Automatic Limited Discoverable Mode Ended

Table 5-146. Automatic limited discoverable ended Indicator

Description	**Indication send from the device, when automatic limited discoverable mode has ended.**	
Firmware	5.xx	
PacketType	IND	
Opcode	GAP_SET_SCANMODE	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_UNKNOWN_ERROR

5.2.14.8 Force Master Role

The Force Master Role is initiated only after boot up or reset. The role is stored as parameter within NVS.

If Force Master is activated the LMX9820A tries to switch its role to master if connected from another device. For this the LMX9820A is sending a Master/Slave switch request after link establishment to the remote device. If the switch is successful, the link setup will be continued and the LMX9820A is open for other incoming connections.
If the switch fails, the link will be dropped.

This functionality allows Access Point like applications on the LMX9820A.

Table 5-147. Force Master Role Request

Description	**This command will change the preferred role of the LMX9820A e.g. to force master role at any connections. The LMX9820A must be reset to let the changes take effect. A delay of at least 1 sec should be inserted between this command and the reset operation.**
Firmware	5.xx
PacketType	REQ

Table 5-147. Force Master Role Request

Opcode	FORCE_MASTER_ROLE	
DataLength	1	
Data	Role 1 bytes	0x00 Don't care about role 0x01 Force master role at connection setup

Table 5-148. Force Master Role Confirm

Description	**Confirms the request above**	
Firmware	5.xx	
PacketType	CFM	
Opcode	FORCE_MASTER_ROLE	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_ROLE ERROR_INVALID_NO_OF_PARAMETERS

5.2.14.9 Read Operation Mode

The Operation Mode of the LMX9820A controls the behaviour in different situations. Please see Section 1.6 for details.

Table 5-149.

Description	**This command will read out the current operation mode.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	READ_OPERATION_MODE	
DataLength	0	
Data	None	

Table 5-150. Read Operation Mode Confirm

Description	**Confirms the request above**	
Firmware	5.xx, 6.23 or later for Automatic Sniff Mode	
PacketType	CFM	
Opcode	READ_OPERATION_MODE	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS
	Mode 1 Byte	0x00 Command Mode 0x01 Automatic Mode 0x02 Automatic Sniff Mode

5.2.14.10 Write Operation Mode

Table 5-151. Write Operation Mode Request

Description	**This command will change the operation mode stored in NVS. The new setting will take effect after a reset.**

Table 5-151. Write Operation Mode Request

Firmware	5.xx, 6.23 or later for Automatic Sniff Mode	
PacketType	REQ	
Opcode	WRITE_OPERATION_MODE	
DataLength	1	
Data	Mode 1 Byte	0x00 Command Mode 0x01 Automatic Mode 0x02 Automatic Sniff Mode

Table 5-152. Write Operation Mode Confirm

Description	**Confirms the request above**	
Firmware	5.xx	
PacketType	CFM	
Opcode	WRITE_OPERATION_MODE	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_INVALID_MODE

5.2.14.11 Set Default Link Latency

Table 5-153. Set Default Link Latency Request

Description	**This command is used to change the default SPP link latency. The default link latency is set during SPP connection setup. The default link latency setting is stored in NVS. The link latency is used to calculate a poll interval for the ACL link. The calculated poll interval may be bigger than the latency specified because of e.g. an SCO link or another ACL link.**	
Firmware	6.16 or later	
PacketType	REQ	
Opcode	SET_DEFAULT_LINK_LATENCY	
DataLength	2	
Data	Latency 2 byte	The link latency in slots. 0: No link latency requirement (default) 2-400: Valid link latency

Table 5-154. Set Default Link Latency Confirm

Description	**Response to the request above.**	
Firmware	6.16 or later	
PacketType	CFM	
Opcode	SET_DEFAULT_LINK_LATENCY	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_LIMIT

5.2.14.12 Get Default Link Latency

Table 5-155. Get Default Link Latency Request

Description	This command is used to read the default link latency setting from NVS.
Firmware	6.16 or later
PacketType	REQ
Opcode	GET_DEFAULT_LINK_LATENCY
DataLength	0

Table 5-156. Get Default Link Latency Confirm

Description	Response to the request above.	
Firmware	6.16 or later	
PacketType	CFM	
Opcode	GET_DEFAULT_LINK_LATENCY	
DataLength	2	
Data	Latency 2 byte	The link latency in slots. 0: No link latency requirement (default) 2-400: Valid link latency

5.2.14.13 Set Default Sniff Mode Parameters

Table 5-157. Set Default Sniff Mode Parameters Request

Description	This command is used to change the default Sniff mode parameters stored in the NVS. These parameters are used in Automatic Sniff Mode operation mode and when Sniff mode is enabled for a default connection stored in the NVS.	
Firmware	6.23 or later	
PacketType	REQ	
Opcode	SET_DEFAULT_SNIFF_MODE_PARAMETERS	
DataLength	2	
Data	SniffMaxInterval 2 bytes	Time = N * 0.625 msec Range for N: 0x0006 to 0x1000. Only even values are valid Time Range: 1.25 msec to 2.56 sec
	SniffMinInterval 2 bytes	Time = N * 0.625 msec Range for N: 0x0006 to 0x1000. Only even values are valid Time Range: 1.25 msec to 2.56 sec
	SniffAttempt 2 bytes	Number of Baseband receive slots for sniff attempt. Length = N* 1.25 msec Range for N: 0x0001 – SniffMinInterval/2
	SniffTimeout 2 bytes	Number of Baseband receive slots for sniff timeout. Length = N * 1.25 msec Range for N: 0x0000 to 0x0028 Time Range: 0 msec - 40.9 Seconds

Table 5-158. Set Default Sniff Mode Parameters Confirm

Description	Response to the request above
Firmware	6.23 or later

Table 5-158. Set Default Sniff Mode Parameters Confirm

Description	Response to the request above	
PacketType	CFM	
Opcode	SET_DEFAULT_SNIFF_MODE_PARAMETERS	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_LIMIT

5.2.14.14 Get Default Sniff Mode Parameters

Table 5-159. Get Default Sniff Mode Parameters Request

Description	This command is used to read the default Sniff mode parameters from the NVS.
Firmware	6.23 or later
PacketType	REQ
Opcode	GET_DEFAULT_SNIFF_MODE_PARAMETERS
DataLength	0

Table 5-160. Get Default Sniff Mode Parameters Confirm

Description	Response to the request above.	
Firmware	6.23 or later	
PacketType	CFM	
Opcode	GET_DEFAULT_SNIFF_MODE_PARAMETERS	
DataLength	2	
Data	SniffMaxInterval 2 bytes	Time = N * 0.625 msec Range for N: 0x0006 to 0x1000. Only even values are valid Time Range: 1.25 msec to 2.56 sec
	SniffMinInterval 2 bytes	Time = N * 0.625 msec Range for N: 0x0006 to 0x1000. Only even values are valid Time Range: 1.25 msec to 2.56 sec
	SniffAttempt 2 bytes	Number of Baseband receive slots for sniff attempt. Length = N* 1.25 msec Range for N: 0x0001 – SniffMinInterval/2
	SniffTimeout 2 bytes	Number of Baseband receive slots for sniff timeout. Length = N * 1.25 msec Range for N: 0x0000 to 0x0028 Time Range: 0 msec - 40.9 Seconds

5.2.15 ACL Indications

If the Event Filter is set to “Report all events”, the LMX9820A indicates any established and terminated ACL link to the host. This feature enables the user to monitor the bluetooth physical interface. In case the establishment failed the indicators report the reason for terminating or not establishing the link. The ACL error events can be found inTable 5-207 "ACL Error Codes" on page 181.

5.2.15.1 ACL Established

Table 5-161. ACL Established Indicator

Description	This indication is sent to the host when an ACL link is established.

Table 5-161. ACL Established Indicator

Firmware	5.05 or later	
PacketType	IND	
Opcode	GAP_ACL_ESTABLISHED	
DataLength	7	
Data	BdAddr 6 bytes	The Bluetooth address of the remote device.
	Status 1 byte	See “ACL Error Codes” (page 180)

5.2.15.2 ACL Terminated

Table 5-162. ACL Terminated Indicator

Description	**This indication is sent to the host when an ACL link is terminated.**	
Firmware	5.05 or later	
PacketType	IND	
Opcode	GAP_ACL_TERMINATED	
DataLength	8	
Data	BdAddr 6 bytes	The Bluetooth address of the remote device.
	Status 1 byte	See “ACL Error Codes” (page 180)

5.2.16 Bluetooth Security

Bluetooth security is part of the Generic Access Profile GAP. It is controlled by:

- Security Mode
  - — Security Mode 1:
    - – No Security, the device never will ask for authentication or pairing.
  - — Security Mode 2:
    - – The level of security (Authorization, Authentication, Encryption) is determined by the setting in the service database entries. Each entry can have different security requirements.
    - – no authentication necessary for SDAP links
  - — Security Mode 3:
    - – Authentication already necessary on Link Manager level: SDAP links already require authentication (Service requests)
    - – Device always asks for authentication
- Service Database Entry (only for Security Mode 2)
  - — Each entry can specify the settings for authentication and encryption

The LMX9820A on default is in Security Mode 2.

5.2.16.1 Get Security Mode

Table 5-163. Get Security Mode Request

Description	**Reads the current security mode of the Bluetooth device.**
Firmware	5.xx
PacketType	REQ

Table 5-163. Get Security Mode Request

Opcode	GAP_GET_SECURITY_MODE	
DataLength	0	
Data	None	

Table 5-164. Get Security Mode Confirm

Description	**Confirm to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	GAP_GET_SECURITY_MODE	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS
	Mode 1 byte	The current Bluetooth security mode. 0x01 Security mode 1 0x02 Security mode 2 0x03 Security mode 3 0x83 Security mode 3 with link level encryption

5.2.16.2 Set Security Mode

Table 5-165. Set Security Mode Request

Description	**Changes the current security mode of the Bluetooth device. The security mode is stored in NVS and restored during power up.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	GAP_SET_SECURITY_MODE	
DataLength	1	
Data	Mode 1 byte	The current Bluetooth security mode. 0x01 Security mode 1 0x02 Security mode 2 0x03 Security mode 3 0x83 Security mode 3 with link level encryption

Table 5-166. Set Security Mode Confirm

Description	**Confirm to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	GAP_SET_SECURITY_MODE	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_INVALID_SECURITY_MODE

5.2.17 Pairing

The pairing process is part of the authentication procedure. If a local or remote service asks for authentication during link establishment, the authentication process will check if a link key already exists between the two devices. If not, the Link Manager initiates the pairing process. Within this, the two devices exchange a PIN code and create a secure link key which will be stored in each device.

During next link setup, the authentication routine takes the existing link key and proceeds without this pairing procedure.

The LMX9820A has a fixed PinCode which can be changed with the command “Change fixed Pin”. This pin is used during any pairing procedure. On firmware revision 6.23 and later, the Pin can be stored with a 0 length, which will forces a Pin request event to the host if authentication is required. The application needs to respond with the appropriate pin in the Pin request.

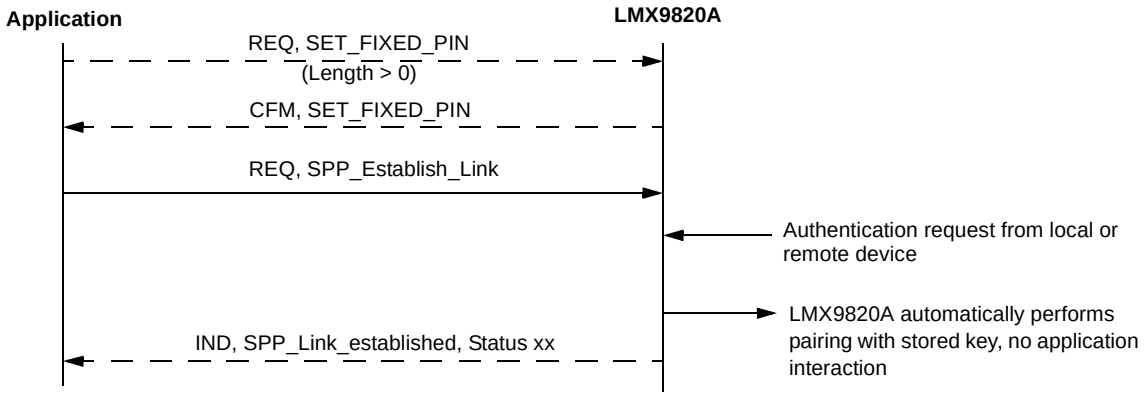


Figure 5-7. Authentication procedure with fixed pin

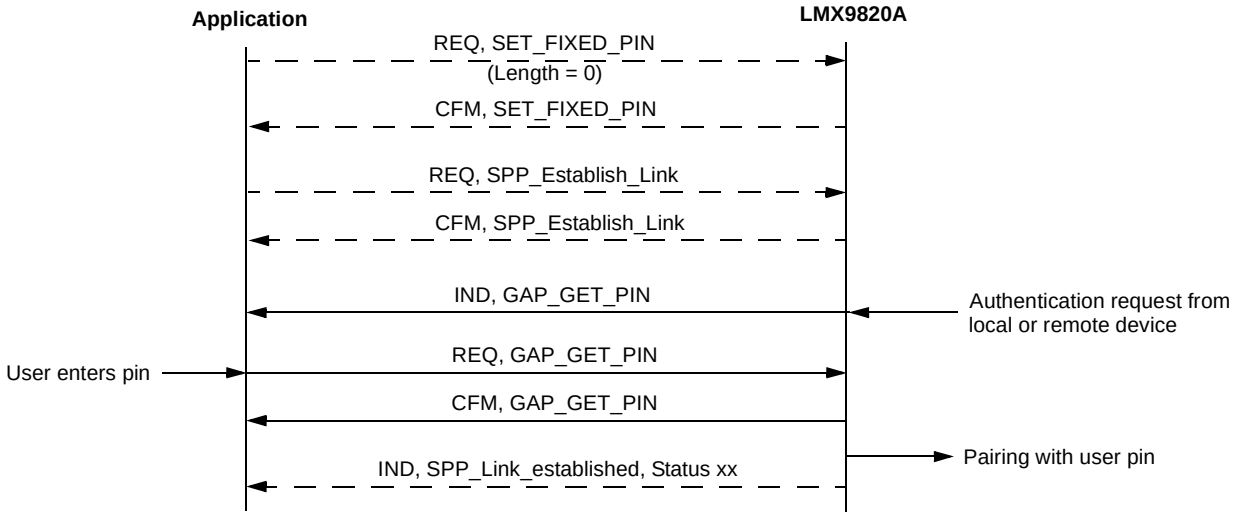


Figure 5-8. Authentication procedure with Pin request

5.2.17.1 Remove Pairing

Table 5-167. Remove Pairing Command

Description	Remove pairing with a remote device.
Firmware	5.xx

Table 5-167. Remove Pairing Command

PacketType	REQ	
Opcode	GAP_REMOVE_PAIRING	
DataLength	6	
	BdAddress 6 byte	Remove pairing to the BdAddress.

Table 5-168. Remove Paired Device Confirm

Description	**Response to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	GAP_REMOVE_PAIRING	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_LINKKEY_DOES_NOT_EXISTS

5.2.17.2 List Paired Devices

Table 5-169. List Paired Devices Command

Description	**Request a list of paired devices from NVS**
Firmware	5.xx
PacketType	REQ
Opcode	GAP_LIST_PAIRED_DEVICES
DataLength	0

Table 5-170. List paired devices Confirm

Description	**Response to the request above.**	
Firmware	5.xx	
PacketType	CFM	
Opcode	LIST_PAIRED_DEVICES	
DataLength	2 +6 * DeviceCount	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS
	DeviceCount 1 byte	Number of devices in the list of paired devices If 0 the device is not paired to any other devices. The maximum number of paired devices is 7.
	BdAddresses 6 byte * DeviceCount	The list of paired devices

5.2.17.3 Get Fixed Pin

Table 5-171. Get Fixed Pin Request

Description	**Reads the current fixed pin code from NVS**

Table 5-171. Get Fixed Pin Request

Firmware	5.xx
PacketType	REQ
Opcode	GAP_GET_FIXED_PIN
DataLength	0

Table 5-172. Get Fixed Pin Confirm

Description	Response to the request above.	
Firmware	5.xx	
PacketType	CFM	
Opcode	GAP_GET_ FIXED _PIN	
DataLength	2+Pinlength	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS
	Pinlength 1 byte	Length of pin code, in bytes. Range: 0x01-0x10
	Pincode Pinlength bytes	PIN code used when the two Bluetooth devices are paired. The maximum length of a PIN code is 128 bits (16 bytes).

5.2.17.4 Set Fixed Pin

Table 5-173. Set Fixed Pin Request

Description	Stores a new fixed pin code in NVS	
Firmware	5.xx	
PacketType	REQ	
Opcode	GAP_SET_FIXED_PIN	
DataLength	1+ Pinlength	
	Pinlength 1 byte	Length of pin code, in bytes. Range: 0x01-0x10
	Pincode Pinlength bytes	PIN code used when the two Bluetooth devices are paired. The maximum length of a PIN code is 128 bits (16 bytes).

Table 5-174. Set Fixed Pin Confirm

Description	Response to the request above.	
Firmware	5.xx	
PacketType	CFM	
Opcode	GAP_SET_ FIXED _PIN	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_PINCODE_TOO_LONG

5.2.17.5 Pin Request

The PIN request procedure will only be used, in case the Pincode length stored in NVS is 0. See also Figure 5-8 on page 166.

5.2.17.5.1 Pin request indicator

Table 5-175. Pin Request Indicator

Description	**This event is used to inform the Host when a PIN code is requested during authentication of an ACL link. This event is only generated if the length of the fixed Pin parameter stored in the NVS is set to 0.**	
Firmware	6.23 or later	
PacketType	IND	
Opcode	GAP_GET_PIN	
DataLength	6	
Data	BdAddr 6 byte	The Bluetooth device address of the remote device.

5.2.17.5.1 Pin request response

Table 5-176. Pin Request Command

Description	**This command is used to send a PIN code to the LMX9820A as response to a GAP_GET_PIN indication.**	
Firmware	6.23 or later	
PacketType	REQ	
Opcode	GAP_GET_PIN	
DataLength	7 + Pinlength	
Data	BdAddr 6 byte	Length of pin code, in bytes. Range: 0x00 - 0x10 0x00 indicates that the Host does not allow the authentication of the ACL link.
	Pincode Pinlength bytes	PIN code used when the two Bluetooth devices are paired. The maximum length of a PIN code is 128 bits (16 bytes).

Table 5-177. Pin Request Command Confirm

Description	**Response to the request above**	
Firmware	6.23 or later	
PacketType	CFM	
Opcode	GAP_GET_PIN	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_PINCODE_LENGTH

5.2.18 Local Hardware Configuration

The LMX9820A offers a set of commands to configure general hardware specific parameters.

5.2.18.1 Set Default Audio Settings

Table 5-178. Set Default Audio Settings Command

Description	This command is used to set the default audio settings stored in NVS.	
Firmware	6.xx	
PacketType	REQ	
Opcode	SET_DEFAULT_AUDIO_CONFIG	
DataLength	2	
Data	CodecType 1 byte	0x00: No codec available 0x01: Motorola MC145483 0x02: OKI MSM7717 0x03: PCM Slave 0x04: Winbond W681360 0x05-0xFF: Reserved.
	AirFormat 1 byte	0x00: CVSD 0x01: µ-law 0x02: A-law 0x03-0xFF: Reserved.

Table 5-179. Set Default Audio Settings Confirm

Description	Response to the request above.	
Firmware	6.xx	
PacketType	CFM	
Opcode	SET_DEFAULT_AUDIO_CONFIG	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_ILLEGAL_AUDIO_CODEC_TYPE ERROR_ILLEGAL_AUDIO_AIR_FORMAT

5.2.18.2 Get Default Audio Settings

Table 5-180. Get Default Audio Settings Command

Description	This command is used to get the default audio settings stored in NVS.
Firmware	6.xx
PacketType	REQ
Opcode	GET_DEFAULT_AUDIO_CONFIG
DataLength	0

Table 5-181. Get Default Audio Settings Confirm

Description	Response to the request above.
Firmware	6.xx
PacketType	CFM
Opcode	GET_DEFAULT_AUDIO_CONFIG

Table 5-181. Get Default Audio Settings Confirm

DataLength	3	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS
	CodecType 1 byte	0x00: No codec available 0x01: Motorola MC145483 0x02: OKI MSM7717 0x03: PCM Slave 0x04: Winbond W681360 0x05-0xFF: Reserved.
	AirFormat 1 byte	0x00: CVSD 0x01: µ-law 0x02: A-law 0x03-0xFF: Reserved.

5.2.18.3 Set PCM Slave Configuration

Table 5-182. Set PCM Slave Configuration Request

Description	**This command is used to set the PCM slave configuration in the NVS.**
Firmware	6.23 and later
PacketType	REQ
Opcode	SET_PCM_SLAVE_CONFIG
DataLength	3

Table 5-182. Set PCM Slave Configuration Request

Description	This command is used to set the PCM slave configuration in the NVS.	
Data	PcmSlaveConfig 2 bytes	This 16-bit value (LSB first) is used to store the PCM format configuration for the PCM generic slave. BIT0-1, Slot selection 00: use slot 0 01: use slot 1 10: use slot 2 11: use slot 3 BIT2-3: Number of slots per frame 00: 1 slot 01: 2 slots 10: 3 slots 11: 4 slots BIT4-6: PCM data format 000: Reserved 001: 8 bit A-law 010: 8 bit u-law 011: 13 bit linear 100: 14 bit linear 101: 15 bit linear 110: 16 bit linear 111: Reserved BIT7:Frame sync length 0: short frame sync 1: long frame sync BIT8: Data word length 0: 8-bit data word length 1: 16-bit data word length BIT9: Frame sync polarity 0: use inverted frame sync 1: use normal frame sync BIT10-15: Unused, set to 0
	Fcprs 1 byte	This value is an unsigned integer indicating the frame clock prescaler for generic PCM slave. The ratio between the bit clock and the frame clock must be written into the ACCR.FCPRS register for the generic PCM slave to operate correctly. The following equation must be true: **bit_clock/(Fcprs + 1) = frame_clock**. Example: Example: bit clock = 480000, frame sync rate = 8000, Fcprs must be set to 59 since 480000/(59 + 1) = 8000

Table 5-183. Set PCM Slave Configuration Confirm

Description	Response to the request above	
Firmware	6.23 and later	
PacketType	CFM	
Opcode	SET_PCM_SLAVE_CONFIG	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_INVALID_CODEC_SETTING

5.2.18.4 Set Event Filter

Table 5-184. Set Event Filter Command

Description	This command is used to set the event filter. The setting is stored in NVS.	
Firmware	5.05 or later Filter Level 0x03: 6.10 or later	
PacketType	REQ	
Opcode	SET_EVENT_FILTER	
DataLength	1	
Data	Filter 1 byte	0x00: All events reported 0x01: No ACL Link Indicators (default) 0x02: No events reported 0x03: No events generated and no UART break generated when device leaves transparent mode

Table 5-185. Set Event Filter Confirm

Description	Response to the request above.	
Firmware	5.05 or later	
PacketType	CFM	
Opcode	SET_EVENT_FILTER	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS ERROR_LIMIT

5.2.18.5 Get Event Filter

Table 5-186. Get Event Filter Command

Description	This command reads the current event filter setting.
Firmware	5.05 or later
PacketType	REQ
Opcode	GET_EVENT_FILTER
DataLength	0

Table 5-187. Get Event Filter Confirm

Description	Response to the request above.	
Firmware	5.05 or later	
PacketType	CFM	
Opcode	GET_EVENT_FILTER	
DataLength	1	
Data	Filter 1 byte	0x00: All events reported 0x01: No ACL Link Indicators (default) 0x02: No events reported 0x03: No events generated and no UART break generated when device leaves transparent mode

5.2.18.6 Reset

Table 5-188. Reset Request

Description	**This command will perform a soft reset of the LMX9820A. The LMX9820A will send a LMX9820A_READY indication when it has performed the reset. Note: A delay of at least 1 sec should be inserted between this NVS write commands and the reset operation.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	RESET	
DataLength	0	
Data	None	

5.2.18.7 LMX9820A Ready

Table 5-189. Simply Blue Ready Indicator

Description	**The LMX9820A will send this indication to the host when the device is fully initialized and ready to receive commands from the host.**	
Firmware	5.xx	
PacketType	IND	
Opcode	DEVICE_READY	
DataLength	1+Length	
Data	Length 1 byte	Number of bytes in software version string
	version <Length> bytes	ASCII string containing the software version. e.g. "0200" indicating that the software version is version 2.00.

5.2.18.8 Restore Factory Settings

Table 5-190. Restore Factory Settings Request

Description	**This command will restore the LMX9820A configuration in NVS to factory settings except the Bluetooth address of the device, which can be restored otherwise. The factory settings are similar to all default settings listed in Table 1-1 "LMX9820A System Parameters" on page 10. The LMX9820A needs to be restarted in order to let the changes take effect. No commands that write to NVS should be sent to the device in between RESTORE_FACTORY_SETTINGS REQ and CFM. When CFM is sent all the factory settings are completed.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	RESTORE_FACTORY_SETTINGS	
DataLength	0	
Data	none	

Table 5-191.

Description	**Confirms the request above**
Firmware	5.xx

Table 5-191.

PacketType	CFM	
Opcode	RESTORE_FACTORY_SETTINGS	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS

5.2.18.9 Change NVS UART Speed

The UART Speed is only active if the ISEL pin configuration of the LMX9820A is set to “Check NVS settings”. Changed UART speeds get active after reset.

Table 5-192. Change UART Speed Request

Description	**This command will change the UART speed stored in NVS. The new UART speed will be used after a reset. A delay of at least 1 sec should be inserted between this command and the reset operation.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	CHANGE_NVS_UART_SPEED	
DataLength	1	
Data	UartSpeed 1 Byte	The UART speed to be stored in NVS 0x00 = 2400 0x01 = 4800 0x02 = 7200 0x03 = 9600 0x04 = 19200 0x05 = 38400 0x06 = 57600 0x07 = 115200 0x08 = 230400 0x09 = 460800 0x0A = 921600

Table 5-193. Change UART Speed Confirm

Description	**Confirms the request above**	
Firmware	5.xx	
PacketType	CFM	
Opcode	CHANGE_NVS_UART_SPEED	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_UART_SPEED_OUT_OF_RANGE ERROR_INVALID_NO_OF_PARAMETERS

5.2.18.10 Change UART Settings

UART Settings will be stored in NVS and are valid for ALL UART speeds at LMX9820A, except for ISEL pin selected 9600baud. That special settings always uses No parity and One Stop bit.

Table 5-194. Change UART Settings Request

Description	**This command will change the UART settings stored in NVS. The new UART settings will be used after a reset. A delay of at least 1 sec should be inserted between this command and the reset operation.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	CHANGE_UART_SETTINGS	
DataLength	2	
Data	ParityBit 1 Byte	0x00 None 0x01 Even 0x02 Odd
	StopBits 1 Byte	0x00 One Stop bit 0x01 Two Stop bits

Table 5-195. Change UART Settings Confirm

Description	**Confirms the request above**	
Firmware	5.xx	
PacketType	CFM	
Opcode	CHANGE_UART_SETTINGS	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_PARITY_BIT_OUT_OF_RANGE, ERROR_STOP_BITS_OUT_OF_RANGE, ERROR_INVALID_NO_OF_PARAMETERS

5.2.18.11 Bluetooth Test Mode

Bluetooth Qualification requires specific test modes to prove the functionality and quality of the bluetooth device.

The Test Mode Command offers the ability to enable either the bluetooth specified “Device Under Test” Mode and also a UART Loopback mode.

This modes can only be left by a reset.

Table 5-196. Initiate Bluetooth Test Mode Request

Description	**This command will activate the test mode. To exit the test mode, a reset of the device must be performed. If local loopback mode is activated all data send to the device though the UART, are send back to the host. The device can only leave local loopback mode by performing a hardware reset.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	TEST_MODE	
DataLength	1	
Data	Mode 1 byte	The Bluetooth test mode to enter: 0x01 Enable Bluetooth test mode 0x02 Enable local loopback mode

Table 5-197. Initiate Bluetooth Test Mode Confirm

Description	Confirms the request above	
Firmware	5.xx	
PacketType	CFM	
Opcode	TEST_MODE	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_ILLEGAL_TESTMODE ERROR_UNKNOWN_ERROR ERROR_INVALID_NO_OF_PARAMETERS

5.2.18.12 Initiate RF Test Mode

In addition to the standard Bluetooth Test Mode the LMX9820A offers a special RF Test Mode, switching the transmitter into a continuous transmit mode. This is needed for Bluetooth qualification as well as regulatory testings for FCC and ETSI and will be needed for country specific qualification.

This mode can only be left with a hardware reset.

Table 5-198. Initiate RF Test Mode Request

Description	This command will activate the RF test mode. To exit the test mode, a reset of the device must be performed. The transmitter test must be stopped before a new test with changed parameters can be started.	
Firmware	5.xx	
PacketType	REQ	
Opcode	RF_TEST_MODE	
DataLength	13	
Data	Test 1 byte	Test Scenario: 0: Stop transmit 1: Burst transmit, take payload from this HCI Command 2: Burst transmit, use PRBS-9 sequence
	Channel 1 byte	Channel number 0 – 78 0: 2402 MHz 78: 2480 MHz 255 (0xFF): Hopping in connection state
	PaCtrl 1 byte	Reserved for future use

Table 5-198. Initiate RF Test Mode Request

	ModulationCtrl 1 byte	Modulation Control: 0: No modulation 1: Access code only (68 us TX data every 1250 us – note 1) 2: Bluetooth DH1 packet with defined payload (note 2) 3: Bluetooth DM1 packet (17 bytes) 4: Bluetooth DH1 packet (27 bytes) 5: Invalid 6: Invalid 7: Bluetooth HV3 packet (30 bytes) 8: Invalid 9: Bluetooth AUX1 packet (29 bytes) 10: Bluetooth DM3 packet (121 bytes) 11: Bluetooth DH3 packet (183 bytes) 12: Invalid 13: Invalid 14: Bluetooth DM5 packet (224 bytes) 15: Bluetooth DH5 packet (339 bytes)
	ModulationPattern 8 bytes	Defines Access Code modulation if ModulationCtrl = 1, else ignored.
	ModulationPayload 1 byte	Defines one byte of payload repeated through packet if Test = 1, else ignored

Table 5-199. Initiate RF Test Mode Confirm

Description	**Confirms the request above**	
Firmware	5.xx	
PacketType	CFM	
Opcode	RF_TEST_MODE	
DataLength	1	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS

5.2.18.13 Read RSSI

Table 5-200. Read RSSI Request

Description	**This command will read out the current RSSI value for given link a remote device.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	READ_RSSI	
DataLength	6	
Data	BdAddress 6 byte	The link to the bdaddress for which to read out the current RSSI value.

Table 5-201. Read RSSI Confirm

Description	**Confirms the request above**

Table 5-201. Read RSSI Confirm

Firmware	5.xx	
PacketType	CFM	
Opcode	READ_RSSI	
DataLength	2	
Data	Status 1 byte	ERROR_OK ERROR_UNKNOWN_ERROR ERROR_INVALID_NO_OF_PARAMETERS
	RSSI 1 byte	The RSSI value Range: -128 < = N < = 127 Units: dB

5.2.18.14 Firmware Upgrade

The LMX9820A offers a special code to update the firmware of the UART interface. The ISP (In-System-Programming) Code is located in a write protected area of the memory.

ISP can either be activated by setting the pin ENV0 to 0 and reset, or by sending the Firmware Upgrade command.

As the chip will leave the standard command interface after that command, it is not confirmed.

Table 5-202. Firmware Upgrade Request

Description	**This command is used to start the ISP code for firmware upgrade.**	
Firmware	5.xx	
PacketType	REQ	
Opcode	FIRMWARE_UPGRADE	
DataLength	0	
Data		

5.2.18.15 Read NVS

Table 5-203. Read NVS Request

Description	**This command can be used to read from the EEPROM (NVS)**	
Firmware	6.23 or later	
PacketType	REQ	
Opcode	READ_NVS	
DataLength	3	
Data	Address 2 bytes	The address to read from
	Length 1 byte	The number of bytes that should be read

Table 5-204. Read NVS Confirm

Description	**Confirm the request above**
Firmware	6.23 or later
PacketType	CFM
Opcode	READ_NVS

Table 5-204. Read NVS Confirm

Description	Confirm the request above	
DataLength	4 + Length	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS
	Address 2 bytes	The address read from
	Length 1 byte	The number of bytes that has been read
	Data <Length> bytes	The data read

5.2.18.16 Write NVS

Table 5-205. Write NVS Request

Description	This command can be used to write data to the NVS	
Firmware	6.23 or later	
PacketType	REQ	
Opcode	WRITE_NVS	
DataLength	3 + Length	
Data	Address 2 bytes	The address to write to
	Length 1 byte	The number of bytes that should be written
	Data <Length> bytes	The data

Table 5-206. Write NVS Confirm

Description	Confirms the request above	
Firmware	6.23 or later	
PacketType	CFM	
Opcode	WRITE_NVS	
DataLength	4	
Data	Status 1 byte	ERROR_OK ERROR_INVALID_NO_OF_PARAMETERS
	Address 2 bytes	The address written to
	Length 1 byte	The number of bytes that has been written

5.3 ERROR CODES

5.3.1 ACL Error Codes

The following table is copied out Bluetooth Specification 1.1 Part H6.1.

Table 5-207. ACL Error Codes

Error Code	Description
0x01	Unknown HCI Command.
0x02	No Connection.
0x03	Hardware Failure.
0x04	Page Timeout.
0x05	Authentication Failure.
0x06	Key Missing.
0x07	Memory Full.
0x08	Connection Timeout.
0x09	Max Number Of Connections.
0x0A	Max Number Of SCO Connections To A Device.
0x0B	ACL connection already exists.
0x0C	Command Disallowed.
0x0D	Host Rejected due to limited resources.
0x0E	Host Rejected due to security reasons.
0x0F	Host Rejected due to remote device is only a personal device.
0x10	Host Timeout.
0x11	Unsupported Feature or Parameter Value.
0x12	Invalid HCI Command Parameters.
0x13	Other End Terminated Connection: User Ended Connection.
0x14	Other End Terminated Connection: Low Resources.
0x15	Other End Terminated Connection: About to Power Off.
0x16	Connection Terminated by Local Host.
0x17	Repeated Attempts.
0x18	Pairing Not Allowed.
0x19	Unknown LMP PDU.
0x1A	Unsupported Remote Feature.
0x1B	SCO Offset Rejected.
0x1C	SCO Interval Rejected.
0x1D	SCO Air Mode Rejected.
0x1E	Invalid LMP Parameters.
0x1F	Unspecified Error.
0x20	Unsupported LMP Parameter Value.
0x21	Role Change Not Allowed
0x22	LMP Response Timeout
0x23	LMP Error Transaction Collision
0x24	LMP PDU Not Allowed
0x25	Encryption Mode Not Acceptable
0x26	Unit Key Used
0x27	QoS is Not Supported
0x28	Instant Passed

Table 5-207. ACL Error Codes

0x29	Pairing with Unit Key Not Supported
0x2A-0xFF	Reserved for Future Use.

5.3.2 Generic error codes

Table 5-208. Generic Error Codes

Error code	Macro	Description
0x00	ERROR_OK	No error.
0x01	ERROR_INVALID_NO_OF_PARAMETERS	The number of bytes in the request does not correspond to the protocol specification
0x02	ERROR_DURATION_OUT_OF_RANGE	The given duration value is not valid according to the specification.
0x03	ERROR_INVALID_MODE	The selected mode is not valid according to the specification
0x04	ERROR_TIMEOUT	A timeout occurred.
0x05	ERROR_UNKNOWN_ERROR	An unknown error occurred.
0x06	ERROR_NAME_TOO_LONG	The number of bytes in the name string is longer than the maximum specified value.
0x07	ERROR_INVALID_DISCOVERABILITY_PARAMETER	The given discoverability parameter does not contain a valid value according to the specification.
0x08	ERROR_INVALID_CONNECTABILITY_PARAMETER	The given connectability parameter does not contain a valid value according to the specification.
0x09	ERROR_INVALID_SECURITY_MODE	The given security mode is not a valid Bluetooth security mode.
0x0a	ERROR_LINKKEY_DOES_NOT_EXISTS	No link key exists for the given Bluetooth address
0x0b	ERROR_CONNECTION_FAILED	The connection setup failed due to unknown reasons.
0x0c	ERROR_TRUNCATED_ANSWER	The returned number of services is too large to be handled by the LMX9820A. The answer is truncated
0x0d	ERROR_RESULT_TOO_LARGE	The SDP result from the remote device is too large to be handled by the LMX9820A due to ram limitations
0x0e	ERROR_NOT_POSSIBLE_TO_ENTER_TESTMODE	It is currently not possible to enter the selected test mode.
0x0f	ERROR_ILLEGAL_TESTMODE	The given test mode is not a valid test mode according to the specification
0x10	ERROR_RESET_TO_NSC_BDADDRESS	The LMX9820A will change the Bluetooth address to the NSC address.
0x11	ERROR_UART_SPEED_OUT_OF_RANGE	The selected UART speed value is not valid according to the specification.
0x12	ERROR_INVALID_PORT	The given port value is larger than the maximum specified value.
0x13	ERROR_ILLEGAL_STATE_VALUE	The given state value is not a valid state according to the specification
0x14	ERROR_IDENTIFIER_OUT_OF_RANGE	The given identifier is larger than the maximum specified value.
0x15	ERROR_RECORD_ALREADY_IN_SELECTED_STATE	The service record is already enabled/disabled.
0x16	ERROR_INVALID_AUTHENTICATION_VALUE	The given authentication value is not a valid value according to the specification.

Table 5-208. Generic Error Codes

0x17	ERROR_INVALID_ENCRYPTION_VALUE	The given encryption value is not a valid value according to the specification.
0x18	ERROR_MAXIMUM_NO_OF_SERVICE_RECORDS_REACHED	The maximum number of service records, which the LMX9820A is able to store, is reached.
0x19	ERROR_WRITING_TO_NVS	An error occurred while writing to flash. The service record may not be stored.
0x1a	ERROR_INVALID_ROLE	The given role value is not a valid value according to the specification.
0x1b	ERROR_LIMIT	Limits exceeded (Parameter(s) violates limits).
0x1c	ERROR_UNEXPECTED	Unexpected at this moment.
0x1d	ERROR_UNABLE_TO_SEND	Could not send at this moment, no reason specified.
0x1e	ERROR_CURRENTLY_NO_BUFFER	Currently no room in buffer, try again later.
0x1f	ERROR_NO_CONNECTION	Trying to use an inexistent connection.
0x20	ERROR_SPP_INVALID_PORT	Port number out of range.
0x21	ERROR_SPP_PORT_NOT_OPEN	Port is closed.
0x22	ERROR_SPP_PORT_BUSY	Connection establishment on a PORT that has a connection.
0x23	ERROR_SPP_MULTIPLE_CONNECTIONS	Transparent mode attempted while more than 1 connection active.
0x24	ERROR_SPP_MULTIPLE_TRANSPARENT	Trying to store a default connection when a transparent default connection is already stored, or trying to store a transparent default connection when another connection is already stored.
0x25	ERROR_SPP_DEFAULT_CONNECTION_NOT_STORED	Trying to connection to a default connection, which is not stored.
0x26	ERROR_SPP_AUTOMATIC_CONNECTIONS_PROGRESSING	Trying to start connecting to default connections when default connection establishment is already progressing.
0x27	ERROR_UNSPECIFIED_ERROR	Other error.
0x28	ERROR_IDENTIFIER_NOT_IN_USE	Trying to enable a SDP record which is not stored. Wrong identifier.
0x29	ERROR_INVALID_SUPPORTED_FAXCLASS_VALUE	Faxclass parameter must be 0 or 1.
0x2a	ERROR_TOO_MANY_SUPPORTED_FORMATS	The given number of supported formats excesses the specified maximum number of supported formats.
0x2b	ERROR_TOO_MANY_DATASTORES	The given number of data stores excess the specified maximum number of data stores.
0x2C	ERROR_ATTEMPT_FAILED	Attempt to change low power mode failed
0x2D	ERROR_ILLEGAL_LINK_POLICY	The given link policy value is out of range
0x2E	ERROR_PINCODE_LENGTH	The pin code length field is zero or too large.
0x2F	ERROR_PARITY_BIT_OUT_OF_RANGE	The given parity check is out of range
0x30	ERROR_STOP_BITS_OUT_OF_RANGE	The given number of stop bits is out of range
0x31	ERROR_ILLEGAL_LINK_TIMEOUT	The given link timeout value is out of range
0x32	ERROR_COMMAND_DISALLOWED	The command is not allowed.
0x33	ERROR_ILLEGAL_AUDIO_CODEC_TYPE	The given Audio CODEC type is out of range.
0x34	ERROR_ILLEGAL_AUDIO_AIR_FORMAT	The given Audio Air format is out of range.
0x35	ERROR_SDP_RECORD_TOO_LONG	The SDP record is too long.

Table 5-208. Generic Error Codes

0x36	ERROR_SDP_FAILED_TO_CREATE_RECORD	The SDP server failed to create the SDP record.
0x37	ERROR_SET_VOLUME_FAILED	The selected codec does not support volume control.
0x38	ERROR_ILLEGAL_PACKET_TYPE	The packet type specified in the request is not valid.

5.3.3 RFCOMM Error Codes

Table 5-209. RFCOMM Error Codes

Error code	Macro	Description
0x00	RFCS_NO_ERROR	No error
0x01	RFCS_INVALID_DLC	The DLC does not exist
0x02	RFCS_INVALID_PORT	The port does not exist
0x03	RFCS_DLC_ESTABLISH_FAILED	The DLC establishment failed
0x04	RFCS_ACCESS_REJECTED	SECM did not authorize access to the requested service (DLC)
0x05	RFCS_INVALID_CONNECTION	There does not exist a DLC/L2CAP connection to the device
0xFF	RFCS_UNSPECIFIED_ERROR	Not used

5.3.4 RFCOMM Release Reasons

Table 5-210. RFComm Release Reasons

Error code	Macro	Description
0x00	RFCR_DLC_DISC_LOCAL_DEVICE	The local device has disconnected the DLC.
0x01	RFCR_DLC_DISC_REMOTE_DEVICE	The remote device has disconnected the DLC.
0x02	RFCR_DLC_DISC_ACL_FAILURE	ACL link failure/ link supervision timeout.
0x03	RFCR_DLC_DISC_LOWER_LAYER	Lower layer (e.g. L2CAP) has disconnected the DLC.

5.4 AT COMMANDS

The list of AT Commands is derived out of the “Bluetooth Profiles Book V1.1” from the Bluetooth SIG. All commands shall be implemented as described in International Telecommunication Union, “ITU-T Recommendation V.250”

5.4.1 DUN GW

5.4.1.1 Required commands

Table 5-211. Required AT Commands for DUN GW

Name	Description
&C	Circuit 109 (Received line signal detector) Behavior
&D	Circuit 108 (Data terminal ready) Behavior
&F	Set to Factory-defined Configuration
+GCAP	Request Complete Capabilities List
+GMI	Request Manufacturer Identification
+GMM	Request Model Identification
+GMR	Request Revision Identification
A	Answer
D	Dial
E	Command Echo
H	Hook Control
L	Monitor Speaker Loudness
M	Monitor Speaker Mode
O	Return to Online Data State
P	Select Pulse Dialling
Q	Result Code Suppression
S0	Automatic Answer
S10	Automatic Disconnect Delay
S3	Command Line Termination Character
S4	Response Formatting Character
S5	Command Line Editing Character
S6	Pause Before Blind Dialling
S7	Connection Completion Timeout
S8	Comma Dial Modifier Time
T	Select Tone Dialling
V	DCE Response Format
X	Result Code Selection and Call Progress Monitoring Control
Z	Reset To Default Configuration

5.4.1.2 Required Result Codes

Table 5-212. Result Codes for DUN

Name	Description
&C	Circuit 109 (Received line signal detector) Behavior

Table 5-212. Result Codes for DUN

Name	Description
OK	Acknowledges execution of a command.
CONNECT	Connection has been established.
RING	The DCE has detected an incoming call signal from the network.
NO CARRIER	The connection has been terminated, or the attempt to establish a connection failed.
ERROR	Error.
NO DIALTONE	No dial-tone detected.
BUSY	Busy signal detected.

5.4.2 Headset Profile

The following AT commands are defined for the headset profile.

5.4.2.1 Headset AT Command

Table 5-213. AT Commands for Headset profile

AT Capability	Syntax	Description	Values
RING		The Incoming call indication of ITU V.250, Section 6.3.4	
Microphone gain	+VGM=<gain>	Unsolicited result code issued by the HSAG to set the microphone gain of the HS. <gain> is a decimal numeric constant, relating to a particular (implementation-dependent) volume level controlled by the HS.	<gain>: 0-15
Speaker gain	+VGS=<gain>	Unsolicited result code issued by the HSAG to set the speaker gain of the HS. <gain> is a decimal numeric constant, relating to a particular (implementation-dependent) volume level controlled by the HS.	<gain>: 0-15
Microphone gain level report	+VGM=<gain>	Command issued by the HS to report the current microphone gain level setting to the HSAG. <gain> is a decimal numeric constant, relating to a particular (implementation-dependent) volume level controlled by the HS	<gain>: 0-15
Speaker gain level indication report	+VGS=<gain>	Command issued by the HS to report the current speaker gain level setting to the HSAG. <gain> is a decimal numeric constant, relating to a particular (implementation-dependent) volume level controlled by the HS	<gain>: 0-15
Headset button press	+CKPD=200	Command issued by the HS to indicate that the button has been pressed.	

5.4.2.2 Required Result Codes

Table 5-214. Required result codes for Headset implementations

Name	Description
OK	Acknowledges execution of a command.
RING	The DCE has detected an incoming call signal from the network.
ERROR	Error.

6.0 Bibliography

[1] National Semiconductor: LMX9820A Datasheet, Rev 1.2, May 2006

[2] Bluetooth SIG: Specification of the Bluetooth System 1.1, Volume 1 / Core, Version 1.1, February 22 2001

[3] Bluetooth SIG: Specification of the Bluetooth System 1.1, Volume 2 / Profiles, Version 1.1, February 22 2001

7.0 Revision History

This is a report of the revision/creation process of the LMX9820/LMX9820A Bluetooth Serial Port Module - Software Users Guide. Any revisions (i.e., additions, deletions, parameter corrections, etc.) are recorded in the table(s) below.

Revision # (PDF Date)	Revisions / Comments
1.0 (05/16/03)	Initial release.
1.1 (06/20/03)	• Updated Table 0-1 — Changed SW Version to 5.05 — Changed SBC Version to 1.2.0.1 • Updated Table 1-1 "LMX9820A System Parameters" — Added Event Filter — Added Default Link Policy Settings • Added Section "NOTE: The Command “Firmware Upgrade” sets the empty flag on flash memory. This means the LMX9820A needs to be programmed after that command, ended by the ISP “GO” command. Otherwise the flag will stay marked as empty and the module will not come up to Bluetooth software after reset." • Updated Section "The LMX9820A power management is a combination of the firmware and the hardware supported low power modes. Depending on the system activity it decides to switch off as many hardware blocks as possible to reduce the current consumption." — Changed Content — Added Section 3.6.3.1.1 "Default Link Policy Setting" • Added the following commands in Section 5.2 — Get Event Filter — Set Event Filter — Get Default Link Policy — Set Default Link Policy — ACL Indications • Added Section 5.3.1 "ACL Error Codes"
1.2 (08/19/03)	• Added Table 2-21 "Log File of Incoming Link as automatic slave" • Added Table 2-22 "Log File of a Released Link as Automatic Slave" • Updated Section 2.2 "Using the automatic slave operation" • Updated Section 3.6.3.2 "Sniff Mode" • Updated Section 3.6.3.3 "Hold Mode" • Updated Table 5-3 "Opcode Values"
1.3 (09/20/03)	• Several updates on references • Added Section Section 2.5 “Examples” on page 32
1.4 (11/25/03)	• Added Section 2.4.2 “Leaving transparent mode with UART BREAK” on page 31 • Added Section 3.7 “Wake up functionality” on page 53 • Added Section 5.2.9 “Wake-Up Functionality” on page 134 • Updated all commands in Section 5.0 "LMX9820A Command Interface" with Firmware version
1.5 (03/25/04)	• Added NVS Initialization Byte in Table 1-1 "LMX9820A System Parameters" on page 10 • Added Default Supervision Timeout in Table 1-1 "LMX9820A System Parameters" on page 10

Revision # (PDF Date)	Revisions / Comments
1.6 (10/04/04)	• Added Default Audio Settings in Table 1-1 "LMX9820A System Parameters" on page 10 • Added Section 1.7 "Audio support" on page 18 • Modified Event Byte documentation in Section 2.5 "Examples" on page 32, corrected event levels numbers from previous versions • Modified Event Byte documentation in Section 3.4.3 "Summary" on page 44, corrected event levels numbers from previous versions • Added Section 3.8 "Establish audio Links" on page 57 • Added Section 4.0 "Profile Support" on page 65 • Added Section 5.2.6 "Establish Audio (SCO) Link" on page 119 • Added Section 5.2.14.11 "Set Default Link Latency" on page 161 • Added Section 5.2.14.12 "Get Default Link Latency" on page 162
1.6.2 (02/08/06)	• Fixed UART parity settings (Odd and Even) in NVS and command.
1.6.3 (06/20/06)	• Added all new features supported by firmware 6.23 revision: • Added Automatic Sniff mode, Set and Get Default Sniff mode commands. • Modified Figure 1-4 "Operation Flow after boot-up or Reset" to integrate sniff mode. • Modified default connection with Sniff mode capability • Updated commands “Write Operation Mode”, “Read Operation Mode”, “Store Default Connection”, “Get Event Filter” • Added PCM slave settings. Added “Set PCM Slave Configuration” section and commands, added “Using a specific codec driver” and “Using PCM Slave” sections. • Added “Read NVS”, “Write NVS” commands. • Added Pin code request and event. Added drawings in section “Pairing”, added “Pin Request” and event commands.
1.6.4 (10/02/06)	• Modified NVS map in the section System Parameter to reflect the changes from firmware 6.23 • Added hardware section and low power modes in section Power Management