

Alps Embedded Audio Gateway Beta5.2

Firmware Release Note

Document Version 1.0

Contents

1	Limitation of Liability	4
2	Introduction	4
3	Qualification Status	4
4	Firmware Change History	4
4.1	Changes Relative to Alpha1	4
4.2	Changes Relative to Alpha2	4
4.3	Changes Relative to Alpha3	4
4.4	Changes Relative to Alpha4	5
4.5	Changes Relative to Alpha5	5
4.6	Changes Relative to Alpha6	5
4.7	Changes Relative to Alpha7	5
4.8	Changes Relative to Alpha7.1	5
4.9	Changes Relative to Alpha8	5
4.10	Changes Relative to Beta1	5
4.11	Changes Relative to Beta2	5
4.12	Changes Relative to Beta3	5
4.13	Changes Relative to Beta4	6
4.14	Changes Relative to Beta5.0	6
4.15	Changes Relative to Beta5.1	6
5	Functionality	7
5.1	State Diagram	8
5.2	Serial Port Profile	8
5.3	Headset Profile	9
5.4	Hands-Free Profile	9
6	System Configuration	9
6.1	Connections	9
6.2	Inquiry Scan Activity	10
6.3	Page Scan Activity	10
6.4	Security	10
6.5	Sniff Mode	10
6.6	Park Mode	10
6.7	Hold Mode	10
6.8	Device Local Name	10
6.9	Class Of Device	10
6.10	Timers	11
6.11	RFCOMM	11
6.12	Bluetooth Supported Features	11
7	Host Interface	11
7.1	UART Data Discrimination	11
7.2	UART Physical Specification	12
7.3	UART Logical Specification	13
7.4	UART Configuration	13
8	Security	13
8.1	Security Modes	13
8.2	Security Requests & Responses	14
8.2.1	Link Key	14
8.2.2	PIN Code	14
8.3	Security Database	14
8.3.1	Firmware Initialization	14
8.3.2	Link Key Update	15
9	Test Features	15
9.1	Bluetooth Test Mode	15
9.2	Link Condition	15
9.2.1	Received Signal Strength Indication	15
9.2.2	Link Quality	16
10	Persistent Store	16
10.1	Automatic Defrag	17
10.2	Forced Defrag	17
11	Deep Sleep	17
11.1	Conditions for Deep Sleep	17
12	Programmable Input / Output	18
13	Known Issues / Outstanding Items	18
14	References	18
15	Document Change History	19
15.1	Alpha Version	19
15.2	Beta Version	19
16	Test Results	20
16.1	Power consumption	20

16.1.1	Idle State.....	20
16.1.2	Inquiring and Paging State	20
16.1.3	Discoverable and Connectable State.....	20
16.1.4	RFCOMM Connected State – Master – Deep Sleep Allowed (AT+BSLP=1).....	20
16.1.5	RFCOMM Connected State – Slave – Deep Sleep Allowed (AT+BSLP=1).....	21

1 Limitation of Liability

The firmware described in this document is of Beta grade so it is therefore the responsibility of the user to determine if the firmware is appropriate for production. Primary target usage is evaluation and demonstration.

Alps makes no warranty or representation whatsoever of merchantability or fitness of this firmware for any particular purpose or use. In no event shall Alps be liable for any consequential, incidental or special damages whatsoever arising out of the use of, or inability to use this firmware even if the user has advised Alps of the possibility of such damages. The user assumes any and all risks associated with the use of the firmware and shall indemnify, defend and hold Alps harmless from third party claims arising from use of the firmware.

2 Introduction

This document presents a description of the Alps Embedded Audio Gateway firmware implementing all Bluetooth core protocols up to RFCOMM and profiles associated with a standard Audio Gateway implementation. The supported profiles are discussed in the Functionality section.

The firmware is compatible with specific Alps Bluetooth modules only. Descriptions in this document focus on the Bluetooth implementation and non-Bluetooth related features as opposed to the Bluetooth specification itself. For this reason it is assumed that the reader is familiar with the major functionality of Bluetooth devices.

3 Qualification Status

The Alps Embedded Audio Gateway is layered above pre-qualified components that do not have to be re-qualified when the AG is qualified. Alps can provide assistance in gaining qualification for the Audio Gateway profiles including relevant tests, paperwork and recommended BQB.

4 Firmware Change History

4.1 Changes Relative to Alpha1

- Change local name command *AT+BNAM* so that it accepts any ASCII characters.
- Addition of *+BCUS* result code that contains custom AT Commands sent from the peer Headset or Handsfree device.
- Addition of *AT+BCUS* command to enable host to send custom AT Commands to the peer Headset or Handsfree device.
- All result codes now take the form *<CR><LF>+BXXX<CR><LF>* where *<CR>* is the Carriage Return character and *<LF>* is the Line Feed character.
- AT Commands received from host are parsed as soon as the terminating *<CR>* character is received. Previously, at least 4 characters were needed to kick off the parser.
- PIN code requests while in a state other than pairing (using *AT+BPRS* or *AT+BPRM*) are automatically rejected.
- Addition of feature to allow rejection of PIN code requests using the *AT+BPIN* command.
- Addition of feature to supply link keys to the module for authentication.
- Security database commands added; add and remove peer device.
- *AT+BACN* command for SCO connection attempts now includes packet type setting.
- Removed all references to connection handles from result codes and commands.

4.2 Changes Relative to Alpha2

- Addition of *AT+BEVT* commands to enable/disable event reporting by the firmware. Refer to the AT Command Reference document for more details.
- Result codes *+BSPK* and *+BMIC* changed to *+BVGS* and *+BVGW* respectively to match the corresponding AT Commands.
- *+BVGS* and *+BVGW* result code parameters are now expressed in hexadecimal instead of decimal.
- AT command parser error handling fixed so unexpected reboots due to illegal commands are eliminated.
- Buffering of OPP packets received over the UART implemented in the module to improve interoperability with non-compliant stacks.
- UART echo mechanism fixed so that every command character is echoed.
- OPP service record changed to indicate that all object formats are supported.

4.3 Changes Relative to Alpha3

- Upgraded development platform.
- Fixed connect as master bug where if the SDP search failed, the ACL connection would not be released.
- Fixed *AT+CLIP* parse bug for Hands-Free service.
- Fixed UART → RFCOMM data transfer deadlock bug found at UART baudrates above 115200bps.

4.4 Changes Relative to Alpha4

- Added feature to enable UART configuration from the host using AT Commands.

4.5 Changes Relative to Alpha5

- Changes made to error reporting.
- Event masking command *AT+BEVT* extended to support masking of warnings.

4.6 Changes Relative to Alpha6

- Added dynamic registration of service records.
- Added commands to configure inquiry/page scan parameters and sniff mode parameters.

4.7 Changes Relative to Alpha7

- Fixed result code *+BRFC* after connecting to the peer device.
- Fixed send result code *+BRFC* after the data to the UART when connect to the peer device.
- Fixed be not able to send or receive the data on UGXZ4-Flash.

4.8 Changes Relative to Alpha7.1

- Changed inquiry filter using the Class Of Device.
- Added function to configure the initial Modem Status Command.
- Added command to configure the module's security mode.
- Supported to enter the hold mode and to write link policy.
- Disabled UART echo by default.
- Added command to enable device under test mode.

4.9 Changes Relative to Alpha8

- Disabled Master/Slave switch.
- Fixed send a parse ok/error result code after a system error result code when send a command.

4.10 Changes Relative to Beta1

- Not Supported the Fax Profile.
- Fixed *AT+BINQ* bug that the module reboots by issuing this command continuously.
- Fixed *+BRFC* result code parameter for the OPP service.
- Added command to indicate the "call_setup" for the HFP.
- Fixed *AT+BINP* bug that the module reboots by issuing this command continuously when this command is unexpected.
- Changed *AT+BSEC* features that authentication and encryption are always enabled for the DUN service at least.

4.11 Changes Relative to Beta2

- Supported Hands-Free Profile Adopted Version1.0.
- Added command to write the local Class Of Device.
- Added command to change the UART recovery timer.
- Added command to tune the UART for throughput or latency.
- Added command to force defragmentation of the flash ROM.
- Added read commands to obtain the current settings.
- Sniff mode can now be used when a SCO connection exists.
- Enabled Master/Slave switch.

4.12 Changes Relative to Beta3

- For the remote device that has multiple instances of same profile, the function to connect to the specific instance is supported. The commands and result codes related to this function are *AT+BSSV* command, *+BSVR* result code, and *+BSRC* result code.

4.13 Changes Relative to Beta4

- DUN and OPP services removed.
- Support for deep sleep added.
- Polarity of HUM and MUM host interface signals changed to support deep sleep implementation.
- Added commands to control ability of module to enter deep sleep.
- Default Class of Device value changed to reflect currently supported profiles.
- Custom AT commands received from the remote device are now converted before being sent to the host.
- New rules imposed on custom AT commands sent from the host to ensure compatibility with AT command parsers.
- Link Supervision Time Out changed from 5s to 30s.
- Default call hold support list changed to match Bluetooth Hands-Free specification.

4.14 Changes Relative to Beta5.0

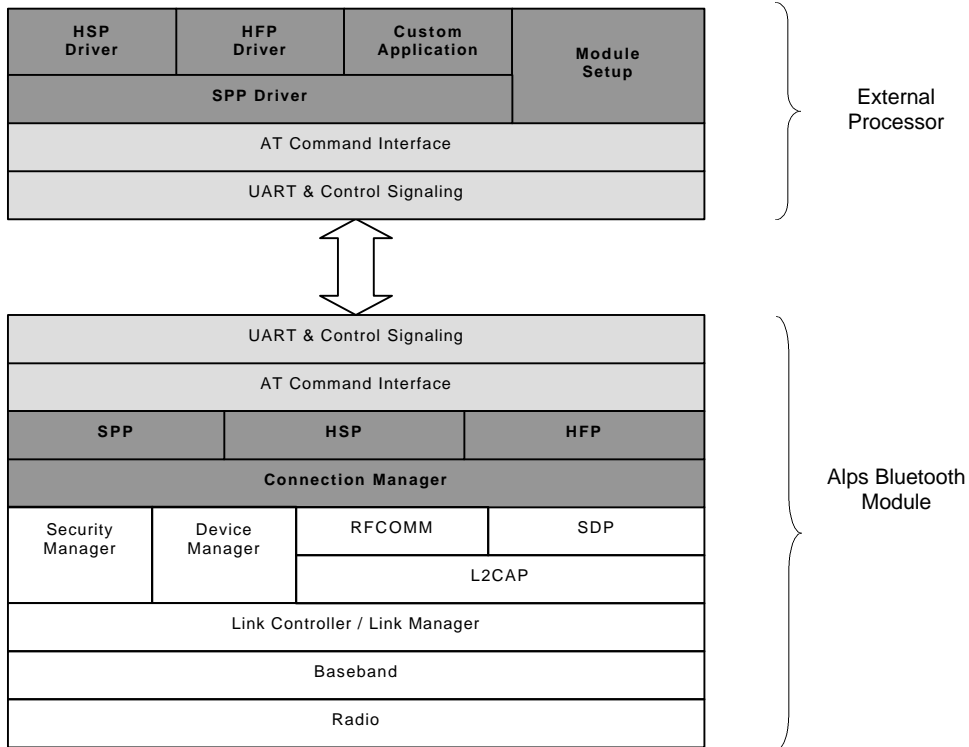
- *AT+BMFS*, *AT+BMTU* and *AT+BTUN* commands removed.
- Remote name request *AT+BRNR* command added.
- Unused timers and handlers removed from firmware application.
- Default baudrate changed from 115.2kbps to 9600bps.
- Default UART recovery timer value changed from 10s to "disabled".

4.15 Changes Relative to Beta5.1

- Default Link Supervision Timeout changed from 30s to 10s.
- Introduced a limit on the number of devices that can be registered with the Security Database.

5 Functionality

The following diagram shows the functionality of the firmware and how it should be integrated with an external host. The firmware supports all of the Bluetooth version 1.1 core protocol layers up to and including RFCOMM in addition to the following Bluetooth profiles; Serial Port Profile (SPP), Headset Profile (HSP) and Hands-Free Profile (HFP).



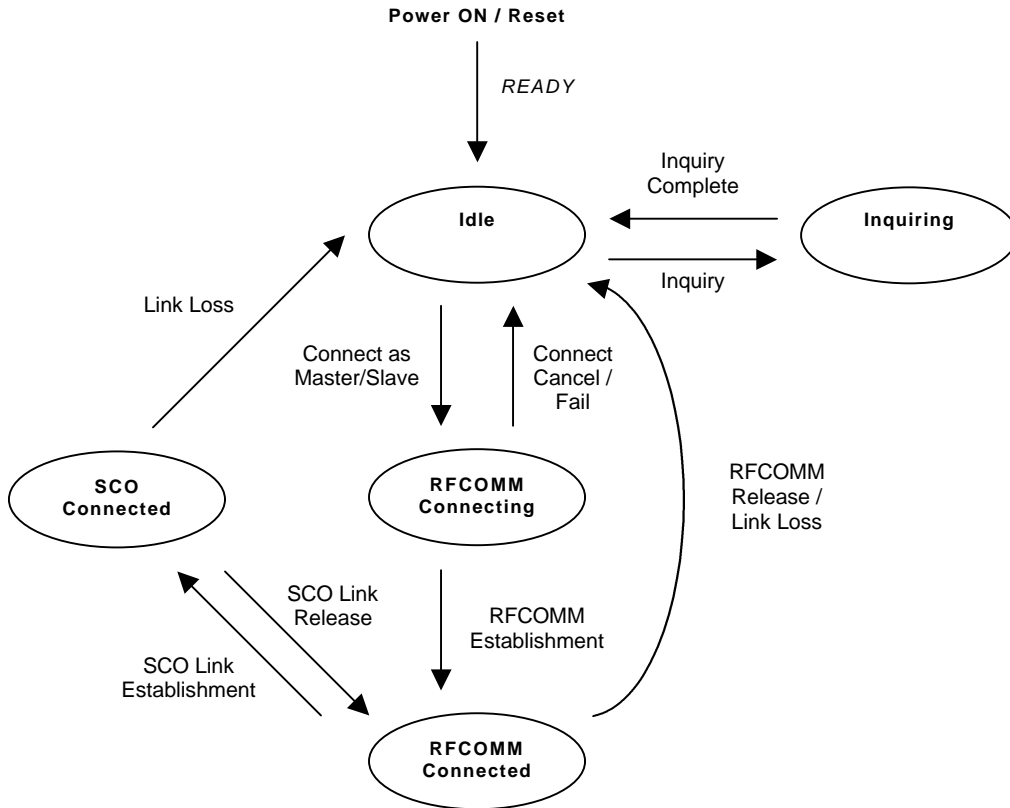
With the exception of the Security Manager and Device Manager, the white blocks in the diagram depict the Bluetooth core entities. The Security Manager allows the module to control the security of the device based upon the Bluetooth Security White Paper and also provides a security database that can be used to manage the security related information of all remote Bluetooth devices encountered. The Device Manager provides an interface similar to the Bluetooth defined Host Controller Interface (HCI) that may be used to communicate with the lower Bluetooth protocol layers. Access to the Bluetooth protocol layers and the Security/Device Managers is made possible through the Connection Manager layer.

The firmware provides the functionality for all of the supported profiles, requiring a simple driver on the external processor to provide a complete implementation. Note that custom applications can be implemented on top of the SPP profile that do not have to conform to any of the Bluetooth specifications.

The interface between the Bluetooth module and the external host processor is UART based. Several additional signals are required to discriminate between data and commands but the operation is relatively simple. Communication between the firmware and the application running on the external processor takes the form of AT Commands, information responses and Result Codes similar to that used by a PC to communicate with a dialup modem.

5.1 State Diagram

The diagram below shows the firmware state transitions and the events that trigger those transitions.



5.2 Serial Port Profile

The SPP functionality has been written against the Bluetooth Profiles Specification, volume 2, v 1.1, February 22nd 2001, Part K:5 that specifies requirements for devices claiming compliance with the Serial Port Profile. The firmware complies with the specification for devices performing the role of Dev A and Dev B. The service record is defined as follows:

Parameter	Default	Configurable	Comments
Type	SerialPort	No	
Attributes defined	ServiceClassIDList ProtocolDescriptorList LanguageBaseIDList ServiceName	No	
Service name	"Serial Port"	No	
RFCOMM server channel	Variable	No	
UUID type	16bit	No	
L2CAP Maximum Transmission Unit (MTU) for SDP connections	48bytes	No	In certain cases the SDP packets will exceed the L2CAP MTU so the remote device must support the SDP continuation flag.
Service record language base	English	No	

5.3 Headset Profile

The HSP functionality has been written against the Bluetooth Profiles Specification, volume 2, v 1.1, February 22nd 2001, Part K:6 that specifies requirements for devices claiming compliance with the Headset Profile. The firmware complies with the specification for devices performing the role of Audio Gateway (AG) only. The service record is defined as follows:

Parameter	Default	Configurable	Comments
Type	HeadsetAudioGateway	No	
Attributes defined	ServiceClassDLList ProtocolDescriptorList BluetoothProfileDescriptorList ServiceName	No	
Service name	"Voice Gateway"	No	
RFCOMM server channel	Variable	No	
UUID type	16bit	No	
L2CAP Maximum Transmission Unit (MTU) for SDP connections	48bytes	No	In certain cases the SDP packets will exceed the L2CAP MTU so the remote device must support the SDP continuation flag.

5.4 Hands-Free Profile

The HFP functionality has been written against the Bluetooth Hands-Free Profile, adopted version 1.0, April 29th 2003 that specifies requirements for devices claiming compliance with the Hands-Free Profile. The firmware complies with the specification for devices performing the role of Audio Gateway (AG) only. The service record is partially configurable and is currently defined as follows:

Parameter	Default	Configurable	Comments
Type	HandsfreeAudioGateway	No	
Attributes defined	ServiceClassDLList ProtocolDescriptorList BluetoothProfileDescriptorList ServiceName Network SupportedFeatures	No	
Service name	"Voice Gateway"	No	
RFCOMM server channel	Variable	No	
UUID type	16bit	No	
L2CAP Maximum Transmission Unit (MTU) for SDP connections	48bytes	No	In certain cases the SDP packets will exceed the L2CAP MTU so the remote device must support the SDP continuation flag.
Network type	-	Yes	AT+BRSR used to set this parameter.
Supported Features	-	Yes	AT+BRSR used to set this parameter.

6 System Configuration

Many of the system parameters are configurable in order to meet the needs of individual applications. Parameters that reside in non-volatile memory must be configured every time the module reboots whereas others that reside in flash ROM only have to be configured once.

6.1 Connections

Parameter	Default	Configurable	Comments
Timeout	None	No	Page Timeout is 5s but connection attempt is continued until this operation is cancelled.
Allowed remote device	Any	No	Connection attempts from all remote devices will be accepted as long as the security requirements are fulfilled.
Link Supervision Timeout	10s	No	Bluetooth connection slaves cannot change the Link Supervision Timeout value. It is the responsibility of the connection master to change the default value if required.

6.2 Inquiry Scan Activity

Parameter	Default	Configurable	Comments
Window	0x0012 (11.25ms)	Yes	AT+BSSP used to configure this parameter.
Interval	0x0800 (1280ms)	Yes	AT+BSSP used to configure this parameter.
Inquiry Access Code	General (0x9E8B33)	No	Inquiring masters must use this access code in order to discover the local device.

Inquiry scan is disabled as long as an active Bluetooth connection exists.

6.3 Page Scan Activity

Parameter	Default	Configurable	Comments
Window	0x0050 (50ms)	Yes	AT+BSSP used to configure this parameter.
Interval	0x0400 (640ms)	Yes	AT+BSSP used to configure this parameter.

Page scan is disabled as long as an active Bluetooth connection exists.

6.4 Security

Parameter	Default	Configurable	Comments
Security mode	Non-secure	Yes	AT+BSEC used to configure this parameter.
Authentication	Off	Yes	AT+BSEC used to configure this parameter.
Authorization	Off	No	
Encryption	Off	Yes	AT+BSEC used to configure this parameter.
Minimum encryption key length	8bit	No	
Maximum encryption key length	56bit	No	

Further security details can be found in the Security section of this document.

6.5 Sniff Mode

Parameter	Default	Configurable	Comments
Maximum interval	0x0100	Yes	AT+BSNP used to configure this parameter.
Minimum interval	0x0100	Yes	AT+BSNP used to configure this parameter.
Attempt	0x0008	Yes	AT+BSNP used to configure this parameter.
Timeout	0x0008	Yes	AT+BSNP used to configure this parameter.

The peer device must support and enable sniff mode for it to be operable.

6.6 Park Mode

The firmware does not support park mode. Sniff mode may be used instead if low power operation is required.

6.7 Hold Mode

Parameter	Default	Configurable	Comments
Maximum interval	0x0100	Yes	AT+BSHP used to configure this parameter.
Minimum interval	0x0100	Yes	AT+BSHP used to configure this parameter.

The peer device must support and enable hold mode for it to be operable.

6.8 Device Local Name

Parameter	Default	Configurable	Comments
Local name	"Alps AG"	Yes	AT+BNAM used to configure this parameter.

6.9 Class Of Device

Parameter	Default	Configurable	Comments
Class of Device	0x400204	Yes	AT+BSCD used to configure this parameter.

6.10 Timers

Parameter	Default	Configurable	Comments
Watchdog timeout	3000ms	No	
Watchdog period	500ms	No	
UART recovery timeout	Disabled	Yes	AT+BURI used to configure this parameter.

6.11 RFCOMM

Parameter	Default	Configurable	Comments
Flow control	Credit based	No	All Bluetooth version 1.1 devices must support RFCOMM credit based flow control.
Maximum Frame Size (MFS)	320bytes	No	

6.12 Bluetooth Supported Features

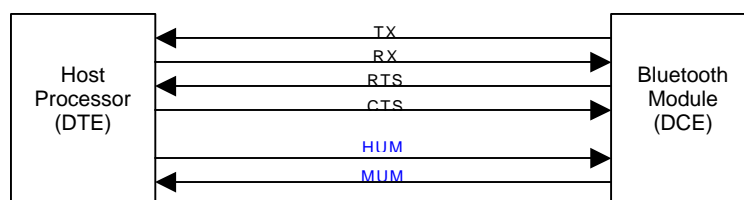
Parameter	Default	Configurable	Comments
3-slot packets	Supported	No	
5-slot packets	Supported	No	
Encryption	Supported	No	
Slot offset	Supported	No	
Timing accuracy	Supported	No	
Role switch	Supported	No	
Hold mode	Supported	No	
Sniff mode	Supported	No	
Park mode	Not supported	No	Use hold or sniff mode for low power operation if required.
RSSI	Supported	No	
Channel quality driven data rate	Supported	No	
SCO link	Supported	No	
HV2 packets	Supported	No	
HV3 packets	Supported	No	
u-law log	Not supported	No	Custom support possible if required.
A-law log	Not supported	No	Custom support possible if required.
CVSD	Supported	No	
Paging scheme	Supported	No	
Power control	Supported	No	
Transparent SCO data	Not supported	No	Custom support possible if required.
Flow control lag	Not supported	No	

7 Host Interface

7.1 UART Data Discrimination

The firmware uses 2 extra signals in addition to the standard UART signals to differentiate data information and command/result code information. The lines are called *HUM* and *MUM* meaning Host UART Mode and Module UART Mode respectively.

When the HUM signal is HI the module will be not be able to enter the deep sleep state or will be woken if it is already in the deep sleep state. It is therefore important that the Hum signal is held HI only when required. Similarly, the MUM signal could be used to wake the host processor if connected to an external interrupt for instance.



Hardware flow control is highly recommended for applications that require *reliable* data transfer. Without hardware flow control, UART buffer overruns are highly probably which result in data loss.

7.2 UART Physical Specification

Data and commands are transferred between the host and the module via the UART. The Bluetooth module takes the role of the Data Circuit terminating Equipment (DCE) and the host processor takes the roles of the Data Terminal Equipment (DTE). The module's default UART settings are specified below.

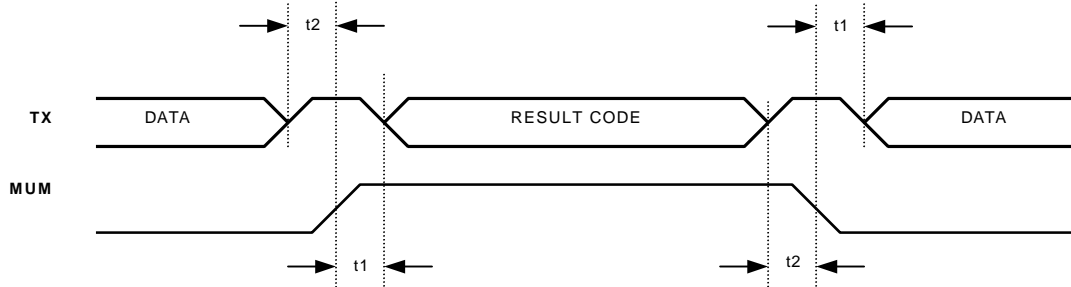
Parameter	Default	Configurable	Comments
Physical Interface	UART	No	
Transport Protocol	None	No	
Baud Rate	9600bps	Yes	AT+BURT used to configure this parameter.
Hardware Flow	On	No	Hardware flow control is essential for applications that require reliable transmission of data.
Data bits	8	No	
Stop bits	1	Yes	AT+BURT used to configure this parameter.
Parity	None	Yes	AT+BURT used to configure this parameter.

While it is not possible for the user to configure the UART flow control, it is possible to do so during module production. However, it is essential for hardware flow control to be enabled for applications that require *reliable* transmission of data. Failure to do so may cause UART buffer overflow on the module resulting in data loss. The UART interface signals used are summarized in the table below. The direction field values should be considered from the Bluetooth module end.

Signal Name	Meaning	Mandatory	Direction	Function
TX	Transmit	Yes	Output	Data transmit
RX	Receive	Yes	Input	Data receive
RTS	Ready To Send	No	Output	Used by module to flow control host.
CTS	Clear To Send	No	Input	Used by host to flow control module.
HUM	Host UART Mode	Yes	Input	Determines if information from host is command (logic 1) or data (logic 0).
MUM	Module UART Mode	Yes	Output	Determines if information from module is result code (logic 1) or data (logic 0).

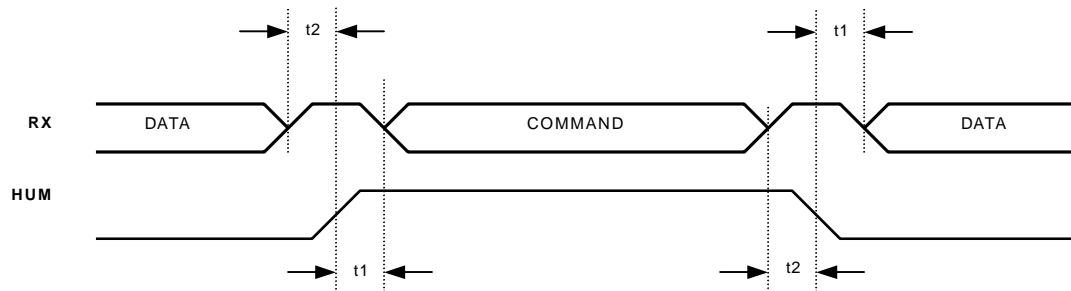
The diagram below gives the timing for data (user data and result code) transmitted from the Bluetooth module to the host processor. Timing is independent of the UART baudrate and other settings. The data setup time and hold times are defined as follows:

Setup time t_1 : min 50ms
Hold time t_2 : min 50ms



The diagram below gives the timing for data (user data and command) transmitted from the host to the Bluetooth module. The Bluetooth module should not expect more than one command within one HUM pulse. Timing is independent of the UART baudrate and other settings. The data setup time and hold times are defined as follows:

Setup time t_1 : min 50ms
Hold time t_2 : min 50ms



7.3 UART Logical Specification

The firmware is controlled by way of AT commands similar to those used to control Hayes compatible modems. In response to the AT commands the module generates information responses and result codes. All commands, responses and codes are represented using printable ASCII byte codes. *Ref[2]* gives details of the commands and codes available.

When a Bluetooth connection has been established for the purpose of data transfer, using SPP for instance, user data should be considered prone to errors. This means that an error recovery mechanism may be required at the transfer application level. The Zmodem file transfer protocol is one such application that uses an error recovery mechanism.

7.4 UART Configuration

It is possible to configure the module UART settings using the *AT+BURT* command. The new UART settings become effective as soon as the command is parsed and the *OK* result code is sent to the host (using the old UART settings). The new UART settings are stored in flash ROM to be used every time the module is started from that point onwards. The UART parameters are boundary checked but it is still possible to configure the UART in such a way that communication becomes impossible with the host.

To counteract such a problem the firmware contains a mechanism whereby the original default UART settings can be restored and communication made possible again. Similar to the self-configuration feature, when the module boots a UART recovery timer is started if it has been enabled. If no data is sent from the host to the module via the UART before the timer expires, the original default UART settings will be restored. When the UART settings have been recovered, it will be indicated to the host using the *+BURT* unsolicited result code at the recovered UART settings of course. The UART recovery timer is configured using the *AT+BURI* command where a parameter of zero will disable the feature. For added protection, if the UART recovery timer has been disabled, it will not be possible to configure the UART settings using the *AT+BURT* command.

The procedure to configure the UART settings is outlined in the following table:

Command	Result Code	UART Setting	Comment
	READY	115200 8-N-1	Module ready to receive AT commands
AT+BURT=3B0,0,0		115200 8-N-1	Configure UART settings
	OK	115200 8-N-1	Command parsed successfully
+BURT:3B0,0,0		230400 8-N-1	New UART settings

8 Security

The Bluetooth specification defines authentication and encryption security procedures that can be used to create secure communication between devices. Authentication is based on a secret link key that is shared between two devices. If the link key does not exist it can be generated using a pairing procedure, otherwise known as bonding, in which the devices will exchange PIN codes. Once the link key has been generated it should be stored in non-volatile memory to be used whenever authentication is required.

Note that certain devices can be configured to be non-pairable, which means that PIN code exchanges are not accepted and hence new link keys cannot be generated. This is a feature that enhances the security of device by giving the user control over when the device can be paired and when it cannot.

8.1 Security Modes

The firmware can be configured to implement several different Security Modes in order to meet the specific requirements of the host application. The available modes and the AT commands used to configure them are shown in the table below.

Security Mode	Required AT Commands	Comments
Non-secure	AT+BSEC=0	The local device does not initiate any security procedures but will forward any security related requests from the remote device to the host.
Link level	AT+BSEC=1	The local device initiates authentication at the link level. That is, authentication using either PIN codes or link keys must occur for link establishment to complete successfully.
Link level with encryption	AT+BSEC=2	This is the same as link level security except that encryption is also requested when ACL connections are established. This provides the greatest form of secure communication.
Non-pairable	AT+BSEC=0 AT+BLNK= AT+BPIN=	The local device does not initiate any security procedures. All link key and PIN code requests from unknown devices should be rejected

8.2 Security Requests & Responses

8.2.1 Link Key

When a link key is required by the firmware for authentication, it will send a `+BLNK` result code to the host. The host must respond to all such link key requests using the `AT+BLNK` command.

Accept

If the link key is available and the host is willing to accept security procedures from the specified remote device it should respond with the appropriate link key. This is a 32byte value and should be specified just like it was received in the `+BPRC` result code.

Reject

If the link key is not available or the host is not willing to accept the request it can reject using the `AT+BLNK=` command where a zero length link key is specified. Depending on the security configuration of the remote device, a PIN code request may follow the link key rejection.

8.2.2 PIN Code

The firmware cannot automatically handle PIN code requests and therefore sends all such request to the host using the `+BPIN` result code to which the host must respond using the `AT+BPIN` command.

Accept

To accept the request the host must specify a valid non-zero length PIN. If the PIN supplied by the remote device matches a new link key will be created and sent to the host using a `+BPRC` result code.

Reject

If the host wishes to reject the PIN code request it should respond with a zero length PIN code as in `AT+BPIN=`. If the current security mode is "non-secure" this will equate to a non-pairable state.

8.3 Security Database

The firmware has a security database that can be used to automate part of the authentication procedure for convenience. When a remote device's details (Bluetooth address and link key) are added to the database, the firmware will automatically handle all link key requests (`+BLNK`) for that particular device, freeing the host from this task. PIN code requests (`+BPIN`) are sent to the host as normal.

Remote devices are automatically added to the database when a new link key is created, for instance, when a pairing procedure completes. Otherwise the host must manually add the devices using the `AT+BSDA` command. In either case, the device can be removed from the database using the `AT+BSDD` command. This will be necessary if the link key of a particular device changes or the device has for some reason become untrusted.

Since the security database is implemented in volatile memory (RAM) all database entries will be lost when the module is reset or powered down. It is therefore necessary for the host to save a copy of the Bluetooth address and associated link key for each of its trusted devices. The host will normally add each device to the security database as part of the firmware initialization sequence.

The firmware imposes a limit on the number of devices that can be manually added to the security database using the `AT+BSDA` command. The value is currently set to **10** devices. If a device is removed from the database, another can be added in its place free of charge. If more than the maximum number of devices must be supported, the host should respond to the link key requests manually.

Here follows some common examples of security database usage:

8.3.1 Firmware Initialization

The host adds 3 separate devices to the security database so that link key requests will be handled automatically by the firmware. The link keys will have been received from the module in a previous session, contained in the `+BPRC` result code.

```
← +BINF:852,2,C7,123456,My Audio Gateway
← READY
→ AT+BSDA=2,C7,111111,12345678901234567890123456789012
← OK
→ AT+BSDA=2,C7,222222,12345678901234567890123456789012
← OK
→ AT+BSDA=2,C7,333333,12345678901234567890123456789012
← OK
...
```

8.3.2 Link Key Update

The host adds a device to the security database but at some point during the session, the link key is changed perhaps due to a request from the peer device. The host then removes the device from the database and re-adds it with the updated link key value.

```

< +BINF:852,2,C7,123456,My Audio Gateway
< READY
-> AT+BSDA=2,C7,111111,12345678901234567890123456789012
< OK
...
< +BPIN:2,C7,111111
-> AT+BPIN=0000
< OK
< +BPRC:2,C7,111111,01234567890123456789012345678901
-> AT+BSDD=2,C7,111111
< OK
-> AT+BSDA=2,C7,111111,01234567890123456789012345678901
< OK
...

```

9 Test Features

In order to aid functional verification and certification testing, the firmware supports various test features. The available test features are described in the following sections.

9.1 Bluetooth Test Mode

The Bluetooth specification defines a test mode for the support of testing the Bluetooth transmitter and receiver. It is intended mainly for certification/compliance testing of the radio and baseband layer but may be used for regulatory approval or in-production testing also. Refer to Bluetooth Core Specification, volume 1, v 1.1, February 22nd 2001, Part I:1 for further details of the Bluetooth test mode.

The firmware supports the Bluetooth test mode through the `AT+BDUT` command. When this command is issued the firmware will set the inquiry/page scan parameters to their Bluetooth defaults and then enable Device Under Test (DUT) mode. It is the responsibility of the external processor to issue the `AT+BSLV` command to make the module connectable and discoverable. The command flow is as shown below:

Command	Result Code	Comment
	READY	Module ready to receive AT commands
AT+BDUT	OK	Change scan parameters and enable Device Under Test mode
	OK	Command parsed successfully
AT+BSLV	OK	Enable slave mode
	OK	Command parsed successfully

The external processor should not issue any other commands when DUT mode is enabled. Normal operation will resume when the module is reset.

9.2 Link Condition

By reading the Received Signal Strength Indication (RSSI) and the Link Quality, the condition of an active Bluetooth connection can be verified. The commands to read the values can only be sent to the module when an active connection exists. The meanings of the values returned by the module are specified in the following sections.

9.2.1 Received Signal Strength Indication

It is possible to read the RSSI for the currently active Bluetooth connection using the `AT+BRSI` command. This command provokes a `+BRSI` result code that returns a *signed* 8bit integer giving values between -128 and $+127$. The Bluetooth specification gives the following definitions:

- If the RSSI is within the Golden Receiver Range, the return value is zero.
- If the RSSI is below the Golden Receiver Range lower limit, the return value is a negative value.
- If the RSSI is above the Golden Receiver Range upper limit, the return value is a positive value.

The Golden Receiver Range is the target signal strength at the receiver. If the RSSI reading rises above the Golden Range upper limit, the return parameter will increase one unit for approximately every dB it rises, i.e. if the signal is 15dB above the golden range then the RSSI value will return +15 (or something close). The value will limit somewhere between +20 and +30: the exact figure depends on the module design. If the RSSI drops below the Golden Range lower limit, the module cannot measure accurately enough to indicate exactly how far the incoming signal strength is below the limit. Instead, the value will vary between -1 and -10 based on how many of the last few packets were below the limit. The measurement will normally limit at -10 or recover to 0 very quickly, without spending much time at the intervening values.

Note that the RSSI return value may be ideal (zero) when either the devices are far apart transmitting at maximum power with RSSI at the bottom of the Golden Range, or very close but transmitting at minimum power with RSSI at the top of the Golden Range. This means that in a power-controlled link the RSSI cannot be used to determine the distance between two Bluetooth devices.

9.2.2 Link Quality

It is possible to read the link quality of the currently active Bluetooth connection using the `AT+BQAL` command. This command provokes a `+BQAL` result code that returns an *unsigned* 8bit integer giving values between 0 and +255. The link quality value is directly related to the Bit Error Rate (BER) with a scale as follows:

Link Quality	BER (%)	Comments
255	0.000	BER resolution between 255 and 215 is 0.0025%.
254	0.0025	
253	0.0050	
...
215	0.1000	BER resolution between 215 and 89 is 0.0800%.
214	0.1800	0.1800
213	0.2600	0.2600
...
89	10.1800	BER resolution between 89 and 0 is 0.6400%.
88	10.8200	
87	11.4600	
...
0	67.1400	

Generally speaking, a link with a BER of between 0% and 0.1% is workable. A link with a BER above 1% will give poor results. The scale below 215 is not fully characterized since results in this region are not stable and often indicate that a link is dropping more packets than it is receiving.

10 Persistent Store

The Bluetooth module contains flash ROM that is used to persistently store, amongst other things, configuration data for the firmware. The block of flash ROM allocated for the configuration data is called Persistent Store (PS). Through the use of AT commands it is possible for an external processor to change various firmware configuration values and store these in PS to be used as the default values every time the module is powered up or is reset.

The flash ROM imposes certain limitations that affect the way in which new parameter values are stored. Individual entries cannot be erased or overwritten so when a parameter is changed and stored in PS the old value is marked as unused and the new value is appended to the end of the PS memory block. This means that as changes are made to the parameters, the amount of memory available for PS decreases. There is a finite amount of memory allocated for PS so if many changes are carried out, it will fill up and further changes will not be possible.

When the module powers up or is reset, it reads all necessary configuration data from the PS. This can be a time consuming process and will therefore account for a large percentage of the firmware boot time. It follows that as the amount of PS utilized increases, so does the boot time.

The following AT commands can cause changes to be made to the PS:

Command	Description
AT+BNAM=<name>	Change the Local Name
AT+BURT=<rate>,<stop>,<parity>	Change UART settings
AT+BURI=<time>	Change UART recover timer

In order to get around the problems associated with PS, the firmware will attempt to remove all of the redundant data from the PS leaving only the newest values of each parameter. This clean-up process is known as defrag. Note that the boot time when defrag is performed is substantially longer than usual. The firmware can automatically defrag the PS or an external processor can request the firmware to perform the defrag explicitly using an AT command.

10.1 Automatic Defrag

When the PS utilization reaches 70%, the firmware will automatically perform the defrag when it next boots up. It is possible for the PS to reach 100% if changes are made to the PS without the resetting the module or powering it down and then up again. Excluding when a defrag occurs, the longest boot time will be experienced when the PS utilization is just below the 70% threshold.

It is important to note that the module power supply must remain constant during the defrag process in order to maintain the integrity of the contents of PS.

10.2 Forced Defrag

It is not possible to directly configure the 70% threshold figure used for automatic defrag. However, an external processor can read the current utilization from the firmware and force a defrag based on that value. The *AT+BDFG?* Command is used to read the current PS utilization defined as a percentage of the total PS available. The *AT+BDFG* command is used to force the defrag itself. In order for the defrag to occur the module must reboot and this handled automatically by the firmware.

It is important to note that the module power supply must remain constant during the defrag process in order to maintain the integrity of the contents of PS.

An example command sequence follows where the external processor has determined that a threshold of 50% is necessary for the intended application.

Command	Result Code	Comment
	READY	Module ready to receive AT commands
AT+BDFG?	OK	Read current PS utilization command parsed successfully
	+BDFG:54	Current PS utilization is approximately 54%
AT+BDFG	OK	Forces defrag command parsed successfully.
	READY	Defrag complete, module ready to receive AT commands

11 Deep Sleep

The module has the ability to enter a deep sleep state in order to reduce power consumption and it will do so automatically whenever possible, however, it is not reasonable to have this functionality enabled at all times. The module's UART is unresponsive in the deep sleep state and therefore cannot be used to transfer data reliably. This does not pose a problem when sending commands to the module because the HUM signal wakes the module prior to the command arriving over the UART. The table below shows the states in which the module is allowed to enter deep sleep. Additionally, the host must allow the module to enter deep by issuing the *AT+BSLP* command with the appropriate parameter.

Firmware state	Allow sleep	Comments
Idle	Yes	HUM signal used to wake the module prior to command transfer.
Inquiring	No	Benefits of deep sleep during inquiry are insignificant.
Connecting	Yes	HUM signal use to wake the module prior to command transfer. Note that deep sleep is most effective when connecting as slave.
Connected	Yes	Enabled for HFP and HSP connections only.
SCO Connected	Yes	Enabled for HFP and HSP connections only.

11.1 Conditions for Deep Sleep

Several conditions must be met in order for the module to enter deep sleep. The conditions include both hardware connected to the module and the configuration of the module itself.

- Module must be allowed to enter deep sleep (refer to table above).
- SPI terminals must be left unconnected.
- UART hardware flow control signal CTS must be LO.
- No data must be transferred to the module over the UART.
- All PIO that are configured as inputs must be LO.

12 Programmable Input / Output

HUM Host UART Mode
This input PIO is used to determine if the information received from the host is user data or AT commands. A logic '0' indicates data and a logic '1' indicates AT command.

MUM Module UART Mode
This output PIO is used to determine if the information sent to the host is user data or a result code. A logic '0' indicates data and a logic '1' indicates result code.

UGXZ2 (Version2 Class2 SMD)

Port Name	Pin	Direction	Name	Function
PIO[2] / PORT4	10	Input	HUM	Determines the type of information received via UART.
PIO[4] / PORT3	9	Output	MUM	Determines the type of information transmitted via UART.

UGPZ1 (Version2 Class1 FIT)

Port Name	Pin	Direction	Name	Function
PIO[2] / Port2	4	Input	HUM	Determines the type of information received via UART.
PIO[7] / LED/FlashPort2	19	Output	MUM	Determines the type of information transmitted via UART.

13 Known Issues / Outstanding Items

- The firmware always interprets the FC bit of the modem status as zero in MSC messages received from the peer Bluetooth device. MSC messages are always transmitted from the module as expected.
- Issuing inquires in quick succession may cause the module to enter an unknown state. Allow at least 100ms between successive inquiry requests with the *AT+BINQ* command.
- Under certain circumstances, data sent to the host via the UART may be initially under-clocked if the module is just waking from the deep sleep state.
- Rapid temperature changes may cause link loss when 2 devices are connected, the connection has been placed in sniff mode and deep sleep has been enabled on the local Bluetooth device.

14 References

In the following references xxx is the firmware version and yy is the document version.

Ref[1]	AG_xxx_USER_yy.pdf	Alps AG Users Guide
Ref[2]	AG_xxx_AT_yy.pdf	Alps AG AT Command Reference List

15 Document Change History

15.1 Alpha Version

Version	Section	Details
Alpha4 1.1	Host Interface – Physical	Corrected signal directions in table and in diagram.
Alpha4 1.2	All	Formatting changes for addition of Contents section.
Alpha5 1.0	Host Interface	Created one main section and added sub-section detailing the user UART configuration feature.
Alpha6 1.0	Programmable I/O	Corrected mistake in MUM description. Input → Output.
Alpha6 1.0	UART Physical Specification	Added detailed timing charts and descriptions. Removed previous waveform diagram.
Alpha6 1.0	Host Interface	Changed sub-section ordering.
Alpha7 1.0	Functionality	Changed service attributes as dynamic registration is implemented.
Alpha7 1.0	Functionality	Updated profile descriptions to make it clear that DUN and FAX profiles are hosted i.e. applications reside on host processor.

15.2 Beta Version

Version	Section	Details
Beta2 1.0	Functionality	Removed Fax Profile Hosting sub-section.
Beta2 1.0	Host Interface	Removed UART Configuration sub-section.
Beta2 1.1	RSSI Return Value	Added new section.
Beta2 1.1	Link Quality Return Value	Added new section.
Beta2 1.2	Functionality	Changed descriptions of Hands-Free Profile sub-section.
Beta2 1.3	State Diagram	Add new section.
Beta3 1.0	Functionality	Changed description of Hands-Free Profile sub-section due to supported HFP adopted version 1.0.
Beta3 1.0	Functionality	Described service records more detail.
Beta3 1.0	Functionality	Integrated State Diagram section into Test Features section.
Beta3 1.0	System Configuration	Added new section.
Beta3 1.0	Host Interface	Added the UART recovery timer description.
Beta3 1.0	Security Policy	Added new section.
Beta3 1.0	Test Features	Added new section.
Beta3 1.0	Test Features	Integrated RSSI Return Value and Link Quality Return Value sections into Test Features section.
Beta3 1.0	Persistent Store	Added new section.
Beta3 1.0	Programmable Input / Output	Added the pin assignment for UGPZ1 module.
Beta3 1.0	Test Results	Added new section.
Beta4 1.0	Test Results	Updated Test Result section.
Beta5.0 1.0	All	Removed references to DUN and OPP.
Beta5.0 1.0	UART Physical Specification	Changed HUM and MUM descriptions and diagrams to suit the new specification
Beta5.0 v1.0	Deep Sleep	Added new section
Beta5.0 v1.0	UART Data Discrimination	Added regarding HUM/MUM and relationship with deep sleep.
Beta5.0 v1.0	Test Results	Removed test results pending completion.
Beta5.1 v1.0	Profile Sections	MTU description updated to indicate that it cannot be configured.
Beta5.1 v1.0	Timers	UART recovery timer default value changed to "Disabled"
Beta5.1 v1.0	RFCOMM	MFS description changed to indicate that it cannot be configured.
Beta5.1 v1.0	UART Physical Specification	Default baudrate changed from 115.2kbps to 9600bps.
Beta5.1 v1.0	Known Issues / Outstanding Items	Updated list to reflect completed items.
Beta5.2 v1.0	System Configuration	Link Supervision Timeout value changed.
Beta5.2 v1.0	Security Policy	"Security Policy" section is now named "Security Modes".
Beta5.2 v1.0	Security	New main section with several new sub sections.

16 Test Results

16.1 Power consumption

Module serial number Z2XB03A
 Module power supply 3.3V
 Radio power table 01ff e200 03ff e700 09ff ec00 12ff f100 1bff f600 24ff fb00 2eff 0000
 PIO circuits PIO circuits (LED, switch) driven by separate power supply
 Oscilloscope TDS7054
 Setup File -
 UART Baudrate 115.2kbps
 Application Alps Embedded Audio Gateway Beta5.0 (25TH May 2004)

16.1.1 Idle State

Extra Details	Current Consumption [mA]				
	Max	High	Low	Min	Mean
Host communications inactive	17.50	17.28	0.21	0.03	0.80
Host communications active	18.19	18.01	0.93	0.75	1.56

16.1.2 Inquiring and Paging State

State	Current Consumption [mA]				
	Max	High	Low	Min	Mean
Inquiring	51.41	50.95	50.07	49.37	50.60
Paging	51.32	51.24	50.05	49.37	50.63

16.1.3 Discoverable and Connectable State

Inquiry Scan [ms]		Page Scan [ms]		Current Consumption [mA]				
Interval	Window	Interval	Window	Max	High	Low	Min	Mean
2560	11.25	0	0	63.70	63.50	0.37	0.24	1.59
160	160	0	0	63.93	63.45	15.74	15.28	62.08
0	0	2560	11.25	63.87	63.42	0.39	0.26	1.67
0	0	160	160	64.34	63.74	37.08	36.58	63.44
2560	11.25	2560	11.25	64.06	63.37	0.42	0.29	2.51
160	80	160	80	64.03	63.74	15.82	14.91	61.93
1280	11.25	640	50	64.01	63.62	0.42	0.30	8.19

16.1.4 RFCOMM Connected State – Master – Deep Sleep Allowed (AT+BSLP=1)

Sniff Parameters (ms)			Hold Parameters	Extra Details		Current Consumption [mA]				
Interval	Attempt	Timeout	Interval (ms)	Data	Audio	Max	High	Low	Min	Mean
-	-	-	-	No	No	57.66	51.29	0.29	0.14	5.81
-	-	-	-	No	HV1	63.16	62.50	51.08	41.45	52.25
-	-	-	-	No	HV2	63.59	62.93	16.31	16.02	35.83
-	-	-	-	No	HV3	64.02	63.39	16.29	1.95	33.39
160	9.375	9.375	-	No	No	56.55	48.27	0.44	0.10	2.94
160	9.375	9.375	-	No	HV1	63.23	62.67	51.38	41.93	52.52
160	9.375	9.375	-	No	HV2	63.58	42.62	16.59	15.92	35.91
160	9.375	9.375	-	No	HV3	63.63	63.01	16.15	1.89	28.23
1000	9.375	9.375	-	No	No	58.14	41.94	0.45	0.10	1.23
1000	9.375	9.375	-	No	HV1	63.02	62.71	50.96	41.34	52.29
1000	9.375	9.375	-	No	HV2	63.73	63.25	16.19	15.90	35.83
1000	9.375	9.375	-	No	HV3	63.74	54.21	16.08	2.09	27.54
-	-	-	4000	No	No	15.80	15.58	0.38	0.22	0.81

16.1.5 RFCOMM Connected State – Slave – Deep Sleep Allowed (AT+BSLP=1)

Sniff Parameters (ms)			Hold Parameters	Extra Details		Current Consumption [mA]				
Interval	Attempt	Timeout	Interval (ms)	Data	Audio	Max	High	Low	Min	Mean
-	-	-	-	No	No	56.14	53.88	20.01	17.27	29.32
-	-	-	-	No	HV1	62.49	57.82	42.17	40.73	52.06
-	-	-	-	No	HV2	63.21	54.96	17.31	17.03	42.08
-	-	-	-	No	HV3	63.27	42.39	17.72	17.07	34.26
160	9.375	9.375	-	No	No	56.53	41.65	0.25	0.14	5.27
160	9.375	9.375	-	No	HV1	62.29	54.34	41.63	39.68	51.66
160	9.375	9.375	-	No	HV2	61.21	58.73	16.76	15.38	34.53
160	9.375	9.375	-	No	HV3	62.73	54.90	16.42	14.59	30.14
1000	9.375	9.375	-	No	No	63.36	62.98	0.37	0.24	1.79
1000	9.375	9.375	-	No	HV1	62.16	54.41	41.51	39.85	51.62
1000	9.375	9.375	-	No	HV2	60.30	56.96	15.89	15.44	34.30
1000	9.375	9.375	-	No	HV3	62.59	53.89	16.82	15.00	28.86
-	-	-	4000	No	No	15.75	15.75	0.40	0.24	0.82