

AUBTM-20 bluetooth module

External Commands

IMPORTANT NOTICE

Austar technology (Austar) reserves the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to Austar's terms and conditions of sale supplied at the time of order acknowledgment.

Austar warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with Austar's standard warranty. Testing and other quality control techniques are used to the extent Austar deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

Austar assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using Austar components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

Austar does not warrant or represent that any license, either express or implied, is granted under any Austar patent right, copyright, mask work right, or other Austar intellectual property right relating to any combination, machine, or process in which Austar products or services are used. Information published by Austar regarding third-party products or services does not constitute a license from Austar to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from Austar under the patents or other intellectual property of Austar.

Reproduction of information in Austar data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. Austar is not responsible or liable for such altered documentation.

Resale of Austar products or services with statements different from or beyond the parameters stated by Austar for that product or service voids all express and any implied warranties for the associated Austar product or service and is an unfair and deceptive business practice. Austar is not responsible or liable for any such statements.

Copyright 2006-2007, Austar Technology (Hang Zhou) Co.,Ltd.

REVISION HISTORY

Revision Number	Release Date	Authors	Reason of Revisions	Review Date	Reviewers
00.01	2007/01/24	Tony wang	Initial Draft	2007/01/24	Tom Liu
00.02	2007/06/28	Tom Liu	Add AT+INQ, modify AT+CON	2007/6/24	Tony Wang

DOCUMENT APPROVALS

Role	Person	Signature	Date

TABLE OF CONTENTS

1. INTRODUCTION	1
1.1 BACKGROUND	1
1.2 PURPOSE	1
1.3 SCOPE	1
1.4 CONSTRAINT AND LIMITATION	1
1.5 ACRONYMS AND ABBREVIATIONS	1
1.6 TERMS	1
1.7 DOCUMENT ORGANIZATION	2
1.8 DOCUMENT LOCATION	2
2. PROTOCOL CONTENT	3
2.1 GENERAL	3
2.2 SIGNALING DIAGRAM CONVENTIONS	4
2.3 DIFFERENT MODE OF THE MODULE	4
2.4 MESSAGE FORMAT	5
2.5 MESSAGE DEFINITION	5
2.5.1 <i>Enter Command Mode</i>	5
2.5.2 <i>Enter Remote Command Mode</i>	6
2.5.3 <i>Escape from Command Mode</i>	6
2.5.4 <i>List all the supported commands</i>	7
2.5.5 <i>Reset to default configuration</i>	7
2.5.6 <i>Enter Sniff mode</i>	7
2.5.7 <i>Read the Current Settings</i>	8
2.5.8 <i>Reboot the module</i>	8
2.5.9 <i>Set the mode of the module</i>	9
2.5.10 <i>Read the Mode of the Module</i>	9
2.5.11 <i>Read the Baudrate setting</i>	10
2.5.12 <i>Set the Baudrate</i>	11
2.5.13 <i>Set the RF Power</i>	11
2.5.14 <i>Read the RF Power</i>	12
2.5.15 <i>Read the Local Name</i>	12
2.5.16 <i>Write the Local Name</i>	13
2.5.17 <i>Read the PIN</i>	13
2.5.18 <i>Write the PIN</i>	14
2.5.19 <i>Read the Security Settings</i>	14
2.5.20 <i>Write the Security Settings</i>	15
2.5.21 <i>Read the Discoverable Mode Settings</i>	15
2.5.22 <i>Write the Discoverable Mode Settings</i>	16
2.5.23 <i>Read the Current Active Profile</i>	16
2.5.24 <i>Set the Current Active Profile</i>	17
2.5.25 <i>Read the Current Paired Device</i>	18
2.5.26 <i>Write the PIN Code for a Remote Device</i>	18
2.5.27 <i>Read Service Specific Security Settings</i>	19
2.5.28 <i>Set Service Specific Security Settings</i>	19
2.5.29 <i>Read Local Bluetooth Address</i>	20
2.5.30 <i>Read Class of Device</i>	20
2.5.31 <i>Set Class of Device</i>	21
2.5.32 <i>Read the RSSI for a Bluetooth Address</i>	21
2.5.33 <i>Start an Inquiry Session</i>	22
2.5.34 <i>Connect to a Remote Device</i>	23

2.5.35	<i>Disconnect a Link with a Remote Device</i>	24
2.5.36	<i>Access PIO</i>	24
2.5.37	<i>Access AIO</i>	25
2.5.38	<i>Start an OBEX session</i>	25
2.5.39	<i>Put start</i>	26
2.5.40	<i>Put Next</i>	27
2.5.41	<i>OBEX Session Request</i>	27
2.5.42	<i>Get start</i>	28
2.5.43	<i>Get Next</i>	29
3.	PROCESS DEFINITION	30
4.	REFERENCES	33

1. Introduction

1.1 Background

AUBTM-20 is a Bluetooth v1.2 module with SPP profiles. The module is intended to be integrated into another HOST system which requires Bluetooth functions. The HOST system could send commands to AUBTM-20 through a UART. AUBTM-20 will parse the commands and execute proper functions, e.g. set the maximum transmit power, change the name of the module. And next the module can transmit the data received from the UART with SPP profiles.

1.2 Purpose

The purpose of this document is to define the protocol between the HOST system and AUBTM-13 through the UART connection.

The protocol heavily depends on the AT commands of 3GPP 27.007[1] and 27.005 GSM[1] recommendations. Most of the commands are the same with the GSM standard, with several special commands defined by AUSTAR technology.

1.3 Scope

This document is largely concerned with the following contents:

1. the package structure of protocol.
2. the definition of each command sent by the HOST system and its corresponding function.
3. The definition of each response sent by AUBTM-20 and its corresponding meaning.

1.4 Constraint and Limitation

✧ All operations should stick to local laws and regulations.

1.5 Acronyms and Abbreviations

Table 1-1 Acronyms and Definitions

Acronym	Definition

1.6 Terms

Table 1-2 Glossary of Terms

Term	Definition
HOST	The system which integrates AUBTM-13
UART	universal asynchronous receiver/transmitter
Bluetooth	
Profile	
SPP	
Bluetooth address	

AT Commands

Term	Definition
Pin code	

1.7 Document Organization

1.8 Document Location

N/A

2. Protocol Content

2.1 General

For the exchange of the commands and unsolicited results codes, the format, syntax and procedures of 3GPP 27.007 [1] shall be taken as reference. The following rules specifically apply for this protocol.

- Only one command (or unsolicited result code) per command line needs to be expected.
- The HOST, by default, shall not echo the command characters.
- The HOST shall always transmit result codes using verbose format.
- The characters below shall be used for AT commands and result codes formatting:

<cr> corresponds to the carriage return (0/13) as stated in [6]

<lf> corresponds to the line feed (0/10) as stated in [6]

- The format of an AT command from the HOST to the module shall be:

<cr><lf><AT command><cr> <lf>

- The format of the OK code from the module to the HOST shall be:

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

- The format of the generic ERROR code from the module to the HOST shall be:

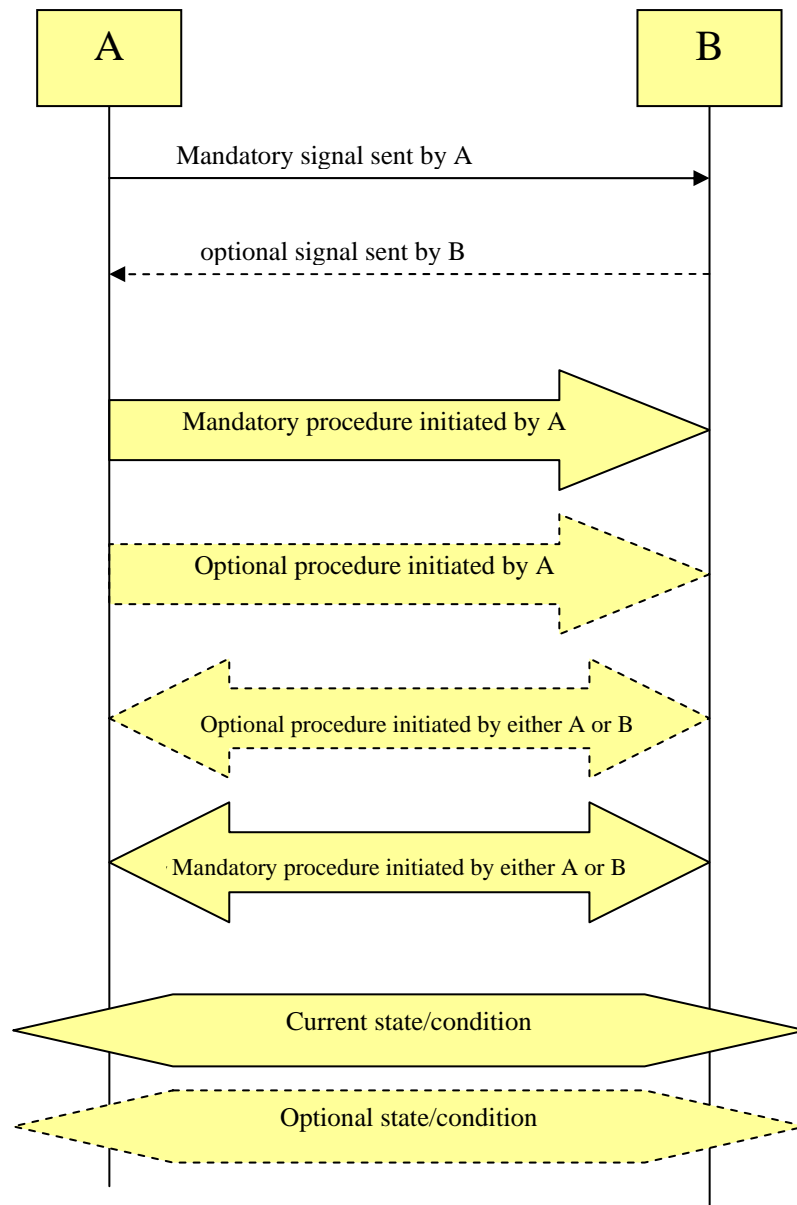
<cr><lf>ERROR<cr><lf>

- The format of an unsolicited result code from the module to the HOST shall be:

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

2.2 Signaling Diagram Conventions

The signaling diagrams in this specification are informative only. Within the diagrams, the following conventions are used to describe procedures:



2.3 Different mode of the module

The module has three different working modes:

- Data and connected mode

In this mode, the SPP connection has been established and all the data from the UART are forwarded to the SPP connection and all the data from the SPP connection are forwarded to the UART.

AT Commands

- Data and command mode

In this mode, the SPP connection has been established. But all the data from the UART would not be forwarded to the SPP connection and all the data from the SPP connection are discarded.

- Command mode

In this mode, no SPP connection has been established. And all the data from the UART would be treated as AT commands.

The module would enter command mode in default after powered up. In this mode the module would not be discovered by any other Bluetooth devices and would only react to the commands from the UART. The host could use commands to make the module discoverable, to find other Bluetooth devices nearby or to connect with one of them.

2.4 Message Format

The GSM 07.07 [1] format and syntax rules shall be taken as the reference for these commands.

2.5 Message Definition

2.5.1 Enter Command Mode

COMMAND	PARAMETER
LLL	

Description:

This command is used to force the module to enter command mode. The module would parse the stream received on the UART and looking for this command during the first 60 seconds after the module is powered up. The module would react to other commands listed in this document only after it received this command.

If the module is in Auto Connect mode and has a SPP link with a remote device, the module would send every byte received on the UART to the remote device before it finally get this command. And it will not send any data to the remote device once it is in command mode.

The command mode would end only after the module receives Escape command.

During command mode, the user could issue any commands to configure and operate the module.

After the module leaves the command mode, the module would start the 60 seconds timer again and could enter command mode again if this command is received again.

Response

value	Description
OK	The command is successful
ERROR	The command is failed

Parameter description

Example command:

LLL

Example response

AT Commands

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

2.5.2 Enter Remote Command Mode

COMMAND	PARAMETER
RRR	

Description:

A module could force a remote device to enter command mode and could configure and operate the remote module afterwards, provided that the SPP connection has been established. The host could send any commands listed in this document, except for the 'LLL', to the remote module and all the commands have the same meaning as to the local module.

The remote device would only accept this command during the 60 seconds time frame after powered up.

Response

value	Description
OK	The command is successful
ERROR	The command is failed

Parameter description

Example command:

RRR

Example response

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

2.5.3 Escape from Command Mode

COMMAND	PARAMETER
===	

Description:

This command makes the module to escape the command mode. This command would only be valid after the module has entered command mode.

Response

value	Description
OK	The command is successful
ERROR	The command is failed

Parameter description

Example command:

===

Example response

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

AT Commands

2.5.4 List all the supported commands

COMMAND	PARAMETER
AT	?

Description:

This command will return a list with all the supported commands by this module.

Response

value	Description
OK	The command is successful
ERROR	The command is failed
commands	The name of the commands

Example command:

```
<cr><lf>AT?<cr><lf>
```

Example response

```
<cr><lf>OK<cr><lf>
<cr><lf>AT?<lf>AT+RESET<cr><lf>
```

2.5.5 Reset to default configuration

COMMAND	PARAMETER
AT+F	

Description:

Reset the module to its factory default configuration

Response

value	Description
OK	The command is successful
ERROR	The command is failed

Parameter

Example:

```
<cr><lf>AT+F<cr><lf>
```

Response

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

2.5.6 Enter Sniff mode

COMMAND	PARAMETER
AT+SNIFF	

Description:

Make the module to enter sniff mode immediately.

Response

value	Description
OK	The command is successful
ERROR	The command is failed

AT Commands

Parameter

Example command:

```
<cr><lf>AT+SNIFF<cr><lf>
```

Example response

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

2.5.7 Read the Current Settings

COMMAND	PARAMETER
AT+SETUP	

Description:

This command will return a list of current settings.

Response

value	Description
OK	The command is successful
ERROR	The command is failed

Parameter

Example command:

```
<cr><lf>AT+SETUP<cr><lf>
```

Example response

```
<cr><lf>BAUD:9600,1,0
<cr><lf>PIN:8888
<cr><lf>+PIO:0,0,0,0,0,0,0
<cr><lf>+SNIFF:0
<cr><lf>+MODE:1
<cr><lf>+CON:0
<cr><lf>Firmware ver: v1.0
<cr><lf>+NAME:AUBTM-20
<cr><lf>+BLBDADDR:00:19:A4:01:00:00<cr><lf>
```

2.5.8 Reboot the module

COMMAND	PARAMETER
AT+REBOOT	

Description:

This command would reboot the module immediately

Response

value	Description
OK	The command is successful
ERROR	The command is failed

Parameter

Example command:

AT Commands

<cr><lf>AT+REBOOT<cr><lf>

Example response

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

2.5.9 Set the mode of the module

COMMAND	PARAMETER
AT+MODE=	mode

Description:

The module has three modes: Master, Slave and Auto Connect mode. A module in Master mode could inquiry and search for devices nearby and initiate connection with them. A module in Slave mode would wait for connection request from modules in Master mode. A slave module would not accept commands for inquiry and connect. A module in Auto Connect mode would try to connect to the last connected module, no matter it is working or not. The Bluetooth address of the last connected device would be stored in a register called Connect ID in the module.

Parameter:

mode

value	Parameter Description
0	Slave
1	Master
2	Auto connect mode

Response

value	Description
OK	The command is successful
ERROR	The command is failed

Example command:

<cr><lf>AT+MODE=1<cr><lf>

Example response

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

2.5.10 Read the Mode of the Module

COMMAND	PARAMETER
AT+MODE=?	

Description:

Parameter:

Response

value	Description
OK	The command is successful
ERROR	The command is failed
+MODE:<mode>	

Response Parameter:

mode

AT Commands

value	Parameter Description
0	Slave
1	Master
2	Auto connect mode

Example command:

```
<cr><lf>AT+MODE=?<cr><lf>
```

Example response

```
<cr><lf>+MODE:1<cr><lf>
```

2.5.11 Read the Baudrate setting

COMMAND	PARAMETER
AT+BAUD=?	

Response

value	Description
OK	The command is successful
ERROR	The command is failed
+BAUD: < UART baud rate>,< stop>,< parity>	

Response Parameter

value	Description
UART baud rate	"0"=no change "9600"=9600 baud "19200"=19200 baud "38400"=38400 baud "57600"=57600 baud "115200"=115200 baud "230400"=230400 baud "460800"=460800 baud "921600"= 921600 baud "1382400"=1382400 baud
stop	"0"=no change "1"=one "2"=two
parity	"0"=no change "1"=none "2"=odd "3"=even

Example response

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

```
<cr><lf>+BAUD:9600,1,0<cr><lf>
```

AT Commands

2.5.12 Set the Baudrate

COMMAND	PARAMETER
AT+BAUD=	< UART baud rate>,< stop>,< parity>

Description:

Set the parameter of UART port

Parameter

value	Description
UART baud rate	"0"=no change "9600"=9600 baud "19200"=19200 baud "38400"=38400 baud "57600"=57600 baud "115200"=115200 baud "230400"=230400 baud "460800"=460800 baud "921600"= 921600 baud "1382400"=1382400 baud
stop	"0"=no change "1"=one "2"=two
parity	"0"=no change "1"=none "2"=odd "3"=even

Response

value	Description
OK	The command is successful
ERROR	The command is failed

Example:

<cr><lf>AT+INQ: "1382400", "0", "0"<cr><lf>

Response

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

2.5.13 Set the RF Power

COMMAND	PARAMETER
AT+POWER=	<DefaultPower>,<MaximumPower>

Description:

Set the maximum transmit power and default transmit power of the module, measured in dBm

Command Parameter

Value	Description
DefaultPower	format: x, "x" is from -128 to 127

AT Commands

MaximumPower	format: x, "x" is from -128 to 127
--------------	------------------------------------

Response

value	Description
OK	The command is successful
ERROR	The command is failed

Example:

<cr><lf>AT+POWER: "0", "4"<cr><lf>

Response

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

2.5.14 Read the RF Power

COMMAND	PARAMETER
AT+POWER=?	

Description:

Read the maximum transmit power and default transmit power of the module, measured in dBm

Response

value	Description
OK	The command is successful
ERROR	The command is failed
+POWER:<DefaultPower>,<MaximumPower>	

Response Parameter

Value	Description
DefaultPower	format: x, "x" is from -128 to 127
MaximumPower	format: x, "x" is from -128 to 127

Example:

<cr><lf>AT+POWER=?<cr><lf>

Response

<cr><lf>+POWER:1,4<cr><lf>

2.5.15 Read the Local Name

COMMAND	PARAMETER
AT+NAME=?	

Response

value	Description
OK	The command is successful
ERROR	The command is failed
+NAME: <name>	

Response Parameter:

AT Commands

value	Description
name	A string contain the user's friendly name of the local module The maxim length of name is 20 characters.

Example:

```
<cr><lf>AT+NAME=?<cr><lf>
```

Response

```
<cr><lf>+NAME:AUBTM-20<cr><lf>
```

2.5.16 Write the Local Name

COMMAND	PARAMETER
AT+NAME=	<name>

Description:

Set The local device's "user friendly" name

Parameter

value	Description
name	A string contain the user's friendly name of the local module The maxim length of name is 20 characters.

Response

value	Description
OK	The command is successful
ERROR	The command is failed

Example:

```
<cr><lf>AT+NAME=AUBTM-20<cr><lf>
```

Response

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

2.5.17 Read the PIN

COMMAND	PARAMETER
AT+PIN=?	

Response

value	Description
OK	The command is successful
ERROR	The command is failed
+PIN:<PIN code>	

Response Parameter

PIN code

value	Description
	The pin code to return when other Bluetooth devices pair with. The maxim length of pin code is 16 digits.

Example:

```
<cr><lf>AT+PIN=?<cr><lf>
```

AT Commands

Response

<cr><lf>+PIN:8888<cr><lf>

2.5.18 Write the PIN

COMMAND	PARAMETER
AT+PIN=	PIN code

Description:

Set Bluetooth PIN code of local module. This is the PIN code that the local machine would send when it receives an authentication request from a remote device.

Command Parameter

PIN code

value	Description
	The pin code to return when other Bluetooth devices pair with. The maximum length of pin code is 16 digits.

Response

value	Description
OK	The command is successful
ERROR	The command is failed

Example:

<cr><lf>AT+PIN=8888<cr><lf>

Response

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

2.5.19 Read the Security Settings

COMMAND	PARAMETER
AT+ BSECMODE =?	

Description:

Response

value	Description
OK	The command is successful
ERROR	The command is failed
+BSECMODE: <mode> , <mode>	

Response Parameter

<mode>

Value	Description
0	Bluetooth Security Mode 1 (Link Level Authentication and Encryption Disabled)
1	Application level security
2	Bluetooth Security Mode 3(Link Level Authentication and Encryption Enabled)

AT Commands

Example:

```
<cr><lf>AT+BSECMODE=?<cr><lf>
```

Response

```
<cr><lf>+BSECMODE:1,1<cr><lf>
```

2.5.20 Write the Security Settings

COMMAND	PARAMETER
AT+ BSECMODE =	<mode>,<mode>

Description:

Set the security mode. By default Security Mode is set to 1

Parameter

<mode>

Value	Description
0	Bluetooth Security Mode 1 (Link Level Authentication and Encryption Disabled)
1	Application level security
2	Bluetooth Security Mode 3(Link Level Authentication and Encryption Enabled)

Response

value	Description
OK	The command is successful
ERROR	The command is failed

Example:

```
<cr><lf>AT+MODE=0,0<cr><lf>
```

Response

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

2.5.21 Read the Discoverable Mode Settings

COMMAND	PARAMETER
AT+ BDISCMODE =?	

Response

value	Description
OK	The command is successful
ERROR	The command is failed
+BDISCMODE: <mode> , <timeout>	

Response Parameter

<mode>

Value	Description
0	Not Discoverable and Not Connectable
1	General Discoverable and Not Connectable
2	Not Discoverable and Connectable

AT Commands

3	General Discoverable and Connectable
<timeout>	
Value	Description
0	Infinite always device is set to the 'mode' as requested in this command
<1 .. 180>	time out Value

Example:

<cr><lf>+BDISCMODE=?<cr><lf>

Response

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

2.5.22 Write the Discoverable Mode Settings

COMMAND	PARAMETER
AT+ BDISCMODE =	<mode>, <timeout>

Description:

Set the device in specified type of discoverable mode for the duration specified in this command
At the expiration of the time period the device is not discoverable and connectable.

Parameter

<mode>

Value	Description
0	Not Discoverable and Not Connectable
1	General Discoverable and Not Connectable
2	Not Discoverable and Connectable
3	General Discoverable and Connectable

<timeout>

Value	Description
0	Infinite always device is set to the 'mode' as requested in this command
<1 .. 180>	time out Value
> 180	Any value larger then 180 is considered as 180 seconds

Default value for Discovery Mode is 1
i.e. always discoverable and connectable And Discovery Time is 180s

Response

value	Description
OK	The command is successful
ERROR	The command is failed

Example:

<cr><lf>AT+BDISCMODE=1, 45<cr><lf>

Response

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

2.5.23 Read the Current Active Profile

COMMAND	PARAMETER
---------	-----------

AT Commands

AT+ BPROFILE =?	
-----------------	--

Response

value	Description
OK	The command is successful
ERROR	The command is failed
+BPROFILE: <profile>	

Response Parameter

<profile>

Value	Description
0	No Active Profile
1	Serial Port Profile
2	Head-Set Profile (Future)
3	Hands-Free Profile (Future)

Example:

```
<cr><lf>AT+BPROFILE=?<cr><lf>
```

Response

```
<cr><lf>+BPROFILE:1<cr><lf>
```

2.5.24 Set the Current Active Profile

COMMAND	PARAMETER
AT+ BPROFILE=	<profile>

Description:

This command is used to choose the "Current Active Profile". This command enables the user application to define the current context in which the application decides to work. If "No Active Profile" is enabled, profile specific commands mentioned in the subsequent sections will not be processed. If "No Active Profile" is enabled and peer initiated profile connections are received by the application, the user application should change the context using this command to use the features on that profile connection.

Parameter

<profile>

Value	Description
0	No Active Profile
1	Serial Port Profile
2	Head-Set Profile (Future)
3	Hands-Free Profile (Future)

Response

value	Description
OK	The command is successful
ERROR	The command is failed

Example:

```
<cr><lf>AT+BPROFILE=1<cr><lf>
```

Response

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

AT Commands

2.5.25 Read the Current Paired Device

COMMAND	PARAMETER
AT+ BPAIR =?	

Response

value	Description
OK	The command is successful
ERROR	The command is failed
+BPAIR: <BD_ADDR>	

Response Parameter

BD_ADDR

Value	Description
	format: xx:xx:xx:xx:xx:xx, "x" is from 0 to F

Example:

```
<cr><lf>AT+BPAIR=?<cr><lf>
```

Response

```
<cr><lf>+BPAIR: 01:2a:12:01:00:1f<cr><lf>
```

2.5.26 Write the PIN Code for a Remote Device

COMMAND	PARAMETER
AT+BPAIR =	<Pin Code> , <BD_ADDR>

Description:

This command will store the remote device PIN Code. The stored PIN code will be used during connection setup and pairing with peer.

The information of a maximum of 5 devices can be stored in the database.

Note that this list consists of devices which are not paired.

Parameter

pin code

Value	Description
	The pin code to store. The maxim length of pin code is 16 digits.

BD_ADDR

Value	Description
	format: xx:xx:xx:xx:xx:xx, "x" is from 0 to F

Response

value	Description
OK	The command is successful
ERROR	The command is failed

Example:

```
<cr><lf>AT+BPAIR=1290,01:2a:12:01:00:1f<cr><lf>
```

Response

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

AT Commands

2.5.27 Read Service Specific Security Settings

COMMAND	PARAMETER
AT+ BSERVICEATTR =?	

Response

value	Description
OK	The command is successful
ERROR	The command is failed
+BSERVICEATTR:<service_type>,<authentication>,<encryption>	

Response Parameter

service_type

Value	Description
1	Serial Port Profile
2	Head-Set Profile
3	Hands-Free Profile

Authentication

Value	Description
0	disable
1	enable

Encryption

Value	Description
0	disable
1	enable

Example:

```
<cr><lf>AT+BSERVICEATTR=?<cr><lf>
```

Response

```
<cr><lf>+BSERVICEATTR:1,0,1<cr><lf>
```

2.5.28 Set Service Specific Security Settings

COMMAND	PARAMETER
AT+ BSERVICEATTR =	<service_type>,<authentication>,<encryption>

Description:

This command is used to register security requirements for specific services.

The values registered through this command will be used only when device is operated in Security Mode 2.

Enabling encryption on Bluetooth link mandates authentication. This command will not allow enabling the encryption of service types with authentication being disabled.

Parameter

service_type

Value	Description
1	Serial Port Profile
2	Head-Set Profile
3	Hands-Free Profile

Authentication

AT Commands

Value	Description
0	disable
1	enable

Encryption

Value	Description
0	disable
1	enable

Default settings for all services is
Authentication – disabled
Encryption - disabled

Response

value	Description
OK	The command is successful
ERROR	The command is failed

Example:

```
<cr><lf>AT+BSERVICEATTR=1,0,1<cr><lf>
```

Response

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

2.5.29 Read Local Bluetooth Address

COMMAND	PARAMETER
AT+ BLBDADDR	

Description:

Get the local device BD ADDR.

Response

value	Description
OK	The command is successful
ERROR	The command is failed
+ BLBDADDR: <Bluetooth address>	

Parameter

Example:

```
<cr><lf>AT+BLBDADDR<cr><lf>
```

Response

```
<cr><lf>+BLBDADDR:00:12:34:56:78:AB<cr><lf>
```

2.5.30 Read Class of Device

COMMAND	PARAMETER
AT+BCOD =?	

Response

AT Commands

value	Description
OK	The command is successful
ERROR	The command is failed
+BCOD: <COD bitmap>	

Response Parameter

<cod bitmap>

Value	Description
0	should be set depending on the end product device type as desired by the manufacturer. The value for 'cod bitmap' should be in Hex format.

Response

OK
<cod bitmap>

2.5.31 Set Class of Device

COMMAND	PARAMETER
AT+BCOD=	<cod bitmap>

Description:

This command is used to set the Bluetooth device type.

Parameter

<cod bitmap>

Value	Description
0	should be set depending on the end product device type as desired by the manufacturer. The value for 'cod bitmap' should be in Hex format.

Response

value	Description
OK	The command is successful
ERROR	The command is failed

Example:

<cr><lf>AT+BCOD=520204<cr><lf>

Response

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

2.5.32 Read the RSSI for a Bluetooth Address

COMMAND	PARAMETER
AT+RSSI=	<Bluetooth address>

Description:

This command will read the value for the difference between the measured Received Signal Strength Indication (RSSI) and the limits of the Golden Receive Power Range for a connection to another Bluetooth device. Any positive RSSI value returned by this command indicates how many dB the RSSI is above the upper limit, any negative value indicates how many dB the RSSI is below the lower limit. The value zero indicates that the RSSI is inside the Golden Receive Power Range.

AT Commands

The RSSI measurement compares the received signal power with two threshold levels, which define the Golden Receive Power Range. The lower threshold level corresponds to a received power between -56 dBm and 6 dB above the actual sensitivity of the receiver. The upper threshold level is 20 dB above the lower threshold level to an accuracy of +/- 6 dB.

Parameter

<Bluetooth address>

Value	Description
	format: xx:xx:xx:xx:xx:xx, "x" is from 0 to F

Response

value	Description
OK	The command is successful
ERROR	The command is failed
+RSSI:<Bluetooth address>,<RSSI>	

Response Parameter:

<Bluetooth address>

Value	Description
	format: xx:xx:xx:xx:xx:xx, "x" is from 0 to F

< RSSI >

Value	Description
-128 ≤ N ≤ 127	Size: 1 Octet (signed integer) Units: dB

Example:

```
<cr><lf>AT+RSSI=00:12:39:00:34:35 <cr><lf>
```

Response

```
<cr><lf>+RSSI:00:12:39:00:34:35, 110<cr><lf>
```

2.5.33 Start an Inquiry Session

COMMAND	PARAMETER
AT+INQ	

Description:

This command will cause the module to enter Inquiry Mode. Inquiry Mode is used to discover other nearby Bluetooth devices. An "OK" is sent from the module to the Host when the this command has been started by the module. When the Inquiry process is completed, the module will send an +INQ event to the Host indicating that the Inquiry has finished. The event parameters of +INQ event will have a result from the Inquiry process, which reports the address and name of Bluetooth devices that responded.

Response

value	Description
OK	The command is successful
ERROR	The command is failed
+INQ:<Bluetooth address>,<name>[,<Bluetooth address>,<name>]	

Response Parameter

<Bluetooth address>

Value	Description
-------	-------------

AT Commands

	format: xx:xx:xx:xx:xx:xx, "x" is from 0 to F
<name>	
Value	Description
name	A string contain the user's friendly name of the local module The maxim length of name is 20 characters.

Example:

<cr><lf>AT+INQ<cr><lf>

Response

<cr><lf>+INQ:00:12:39:00:34:35,"Tom's Mobile",00:13:45:46:99:23,"Jerry"<cr><lf>

2.5.34 Connect to a Remote Device

COMMAND	PARAMETER
AT+CON=	<Bluetooth address>,<profile>

Description:

This command will cause the module to create an connection to the Bluetooth device with the Bluetooth Address specified by the command parameters. The command also tells the module which profile to connect.

Parameter

<Bluetooth address>

Value	Description
	format: xx:xx:xx:xx:xx:xx, "x" is from 0 to F

<profile>

Value	Description
0	handsfree
1	headset
2	spp
3	opp

Response

value	Description
OK	The command is sent successful
ERROR	The command is failed
+CON:<n>	Return the result of the request for connection

Parameter

<n>

Value	Description
0	The connection has not been established
1	The connection has been established

Example:

<cr><lf>AT+CON=00:12:39:00:34:35 ,1<cr><lf>

Response

<cr><lf>+CON:1<cr><lf>

AT Commands

2.5.35 Disconnect a Link with a Remote Device

COMMAND	PARAMETER
AT+DCON=	<profile>,<Bluetooth address>

Description:

This command will cause the module to disconnect with the Bluetooth device with the Bluetooth Address specified by the command parameters. The command also tells the module which profile to disconnect.

Parameter

<Bluetooth address>

Value	Description
	format: xx:xx:xx:xx:xx:xx, "x" is from 0 to F

<profile>

Value	Description
0	handsfree
1	headset
2	spp
3	opp

Response

value	Description
OK	The command is sent successful
ERROR	The command has failed
+DCON:<n>	Return the result of the disconnect request

Parameter

<n>

Value	Description
0	The request has failed
1	The connection has been disconnected successfully

Example:

```
<cr><lf>AT+DCON=1,00:12:39:00:34:35 <cr><lf>
```

Response

```
<cr><lf>+DCON:1<cr><lf>
```

2.5.36 Access PIO

COMMAND	PARAMETER
AT+PIO=	<handle>,<pio>,<value>

Description:

Set the PIO port

Parameter:

< handle >

Value	Description
SET	set the value of pio
GET	get the value of pio

<pio>:

AT Commands

Value	Description
0 to 6	set the value of pio

<value>

Value	Description
0 or 1	set the value of pio

Response

When the handle is "SET"

value	Description
OK	The command is successful
ERROR	The command is failed

When the handle is "GET"

value	Description
OK	The command is successful
ERROR	The command is failed
+PIO=<value>	

Response Parameter:

<value>

Value	Description
0 or 1	the value of pio

Example:

```
<cr><lf>AT+ PIO: "SET", "0", "1"<cr><lf>
```

2.5.37 Access AIO

COMMAND	PARAMETER
AT+AIO=	<aio>

Description:

Read the AIO input

Parameter:

<aio>:

Value	Description
0 or 1	Analog channel

Response

Example:

2.5.38 Start an OBEX session

COMMAND	PARAMETER
AT+OPPCON=	<Bluetooth address>,<packSize>

Description:

Host uses this command to start a session with an OBEX server. This command would also negotiate with the server about the maxim packet size. The final packet size would be smaller one of the client and the server.

AT Commands

Parameter:

<Bluetooth Address>:

Value	Description
	format: xx:xx:xx:xx:xx:xx, "x" is from 0 to F

<packlength>:

Value	Description
	The length of each packet required by the client.

Response

Value	Description
OK	The command is correct
ERROR	The command is incorrect
+OPPCON:	< packetSize>

Response Parameter:

<packetSize>

Value	Description
	the packetSize negotiated by the OBEX server and client. The host would disassemble the object into packet of <packetSize> length.

Example:

2.5.39 Put start

COMMAND	PARAMETER
AT+PUSHF=	<name>,<Totallength>,<length>,<end>

Description:

Host uses this command to start to push an object to an OBEX server. The host should start to send the binary code of the object after getting the OK message from the module.

Parameter:

<name>:

Value	Description
	The name of the object

<Totallength>:

Value	Description
	The total length of the object in bytes.

<length>:

Value	Description
	The amount of bytes which will be sent after this command. Note: This may be smaller then the packet size if this is the last packet

<end>:

Value	Description
0 or 1	0: There is still more to send 1: This is the last packet

Response

Value	Description
OK	The command is correct
ERROR	The command is incorrect

Response Parameter:

AT Commands

Example:

2.5.40 Put Next

COMMAND	PARAMETER
AT+PUSHN=	<length>,<end>

Description:

Host uses this command to send the rest part of the object.

Parameter:

<length>:

Value	Description
	The amount of bytes which will be sent after this command.

<end>:

Value	Description
0 or 1	0: There is still more to send 1: This is the last packet

<length>:

Value	Description
	The amount of bytes which will be sent after this command. Note: This may be smaller then the packet size if this is the last packet

<end>:

Value	Description
0 or 1	0: There is still more to send 1: This is the last packet

Response

Value	Description
OK	
ERROR	

Example:

2.5.41 OBEX Session Request

COMMAND	PARAMETER
AT+OPPREQ=	<Bluetooth address>,<packSize>

Description:

Host uses this command to notify that an OBEX client wants to start a session with the host. This command would also negotiate with the server about the maxim packet size. The final packet size would be smaller one of the client and the server.

Parameter:

<Bluetooth Address>:

Value	Description
	format: xx:xx:xx:xx:xx:xx, "x" is from 0 to F

<packlength>:

Value	Description
	The length of each packet required by the client.

AT Commands

Response

Value	Description
OK	The command is correct
ERROR	The command is incorrect
+OPPREQ:	< packetSize>

Response Parameter:

<packetSize>

Value	Description
	the packetSize negotiated by the OBEX server and client. The host would disassemble the object into packet of <packetSize> length.

Example:

2.5.42 Get start

COMMAND	PARAMETER
AT+PULLF=	<name>,<Totallength>,<length>,<end>

Description:

When an OBEX client wants to push an object to the module, the module would send this message to the host. The host should return OK if it wants to accept this transfer, ERROR if not. The module would, after receiving the OK response, to send the object in binary mode. The amount of byte to be sent is specified by the parameter Totallength.

Parameter:

<name>:

Value	Description
	The name of the object

<Totallength>:

Value	Description
	The total length of the object in bytes.

<length>:

Value	Description
	The amount of bytes which will be sent after this command. Note: This may be smaller then the packet size if this is the last packet

<end>:

Value	Description
0 or 1	0: There is still more to send 1: This is the last packet

Response

Value	Description
OK	
ERROR	

Example:

AT Commands

2.5.43 Get Next

COMMAND	PARAMETER
AT+PULLN=	<length>,<end>

Description:

The host uses this command to send the next packet

Parameter:

<length>:

Value	Description
	The amount of bytes which will be sent after this command. Note: This may be smaller than the packet size if this is the last packet

<end>:

Value	Description
0 or 1	0: There is still more to send 1: This is the last packet

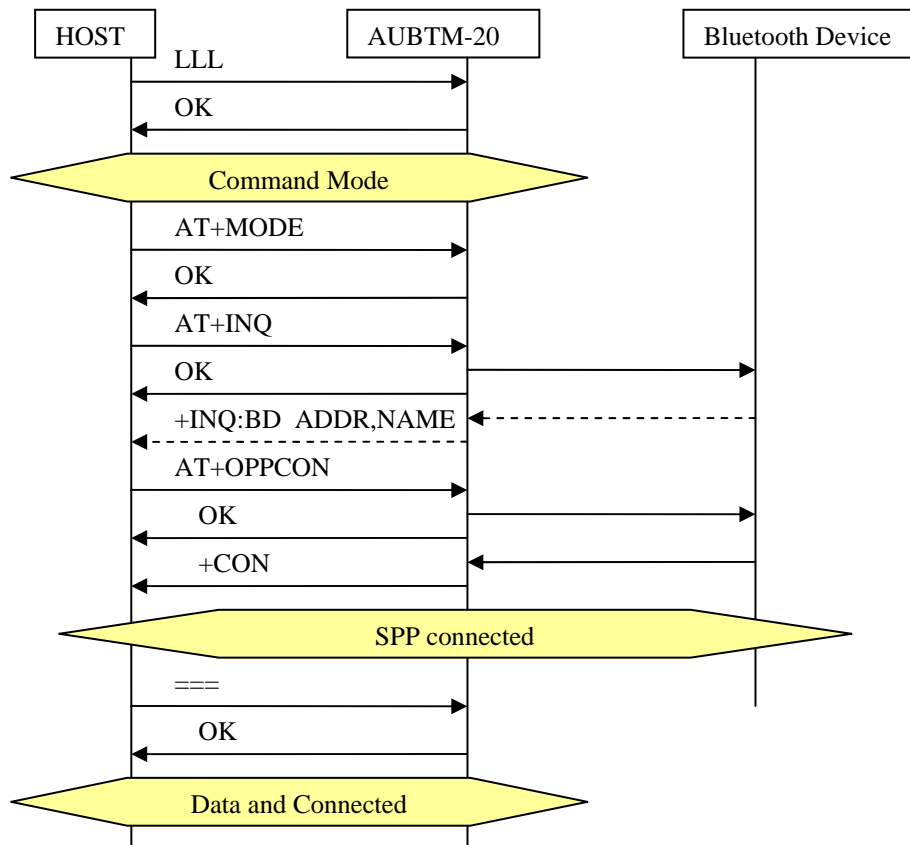
Response

Value	Description
OK	
ERROR	

Example:

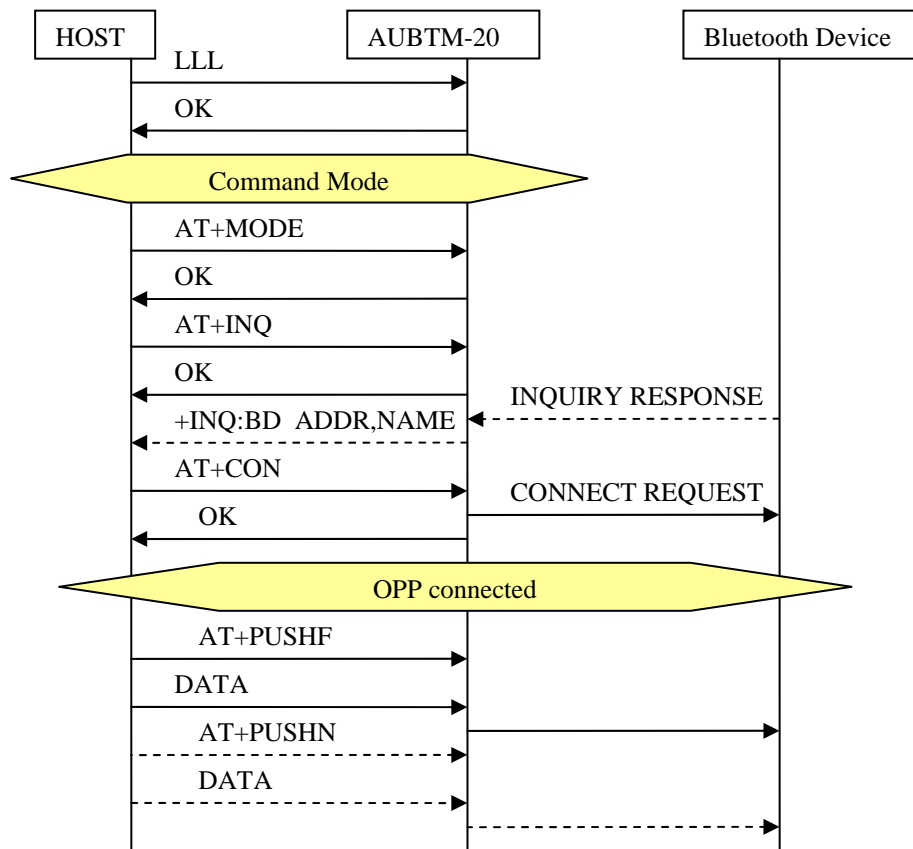
3. Process Definition

A typical process of SPP session would be:



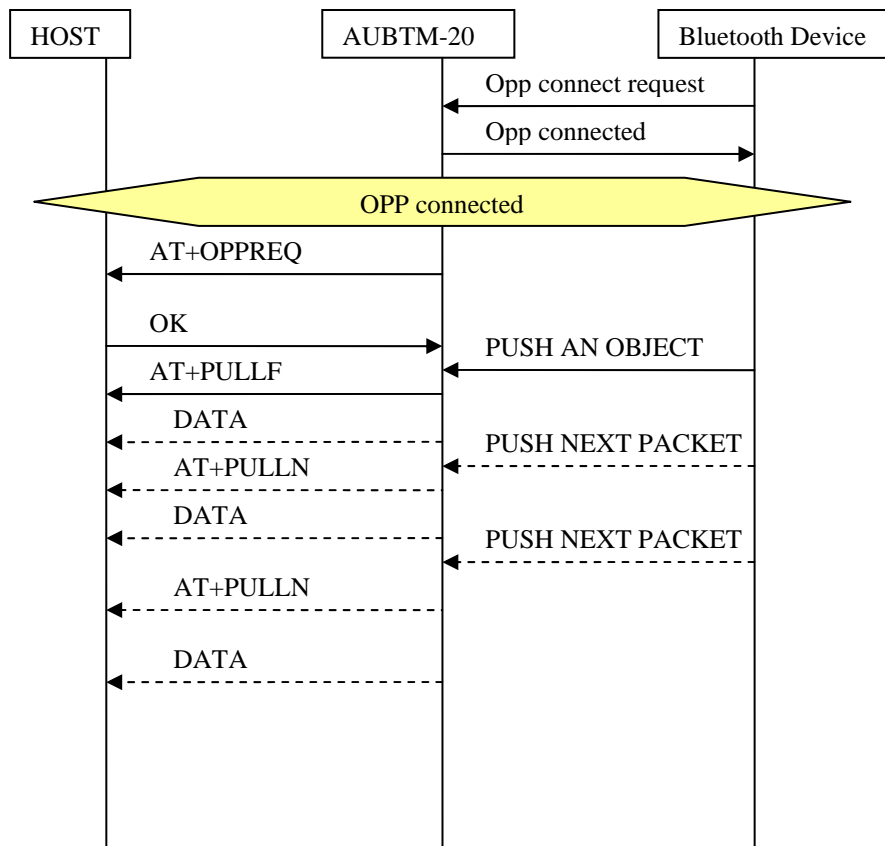
AT Commands

A typical OPP client session would be like this



AT Commands

A typical OPP server session would be like this



4. References

- [1] 3GPP 27.007 v6.8.0 now supersedes and replaces ETS 300 916, "Digital cellular telecommunications system (Phase 2+); AT command set for GSM Mobile Equipment (ME) (GSM 07.07 version 7.5.0)" <http://www.3gpp.org/ftp/Specs/html-info/27007.htm>
- [2] "GSM 02.30 (version 7.1.0): Digital cellular telecommunications system (Phase 2+); Man-Machine Interface (MMI) of the Mobile Station (MS)"

End of Document