Baracoda AT Commands Description

Document ID	Baracoda_ATset_1-2-4
Version	1.2.4
Date	29-06-2005

Table of Contents

1. INTRODUCTION

The document describes the AT commands supported by the Baracoda SmartModules.

Two ways to configure Baracoda SmartModules

You can either send the commands trough the UART or the Bluetooth. To configure by Bluetooth your application must connect to the module on **SppConfig** service.

Note: All AT commands sends must be terminated by a CR else the command would not be processed by the AT parser. The command can also be terminated by CR + LF.

2. AT COMMANDS

2.1. RS232 COMMANDS

2.1.1 ATB + IPR =?.

This command gets the RS232 parameter settings.

Response: <CR><LF ><**BAUDRATE, STOPBIT,PARITY>**<CR><LF>

BAUDRATE : RS232 Baud rate
STOPBIT : RS232 Stop Bits
PARITY : RS232 Parity

Example:

1. -> ATB+IPR=?
<- 9600,ONE,NONE
- ⇨ *Baud Rate = 9600*
- ⇨ *StopBit = One*
- ⇨ *Parity = None*

2. -> ATB+IPR=?
<- 115200,ONE,EVEN
- ⇨ *Baud Rate = 115200*
- ⇨ *StopBit = One*
- ⇨ *Parity = Even*

2.1.2 ATB + IPR = <BAUDRATE,STOPBIT,PARITY>

This command allows setting the RS232 Baud rate.

Response: <CR><LF>**OK**<CR><LF>

BAUDRATE: RS232 Baud rate
Possible Values:

- 9600
- 19200
- 38400
- 57600
- 115200

STOPBIT: RS232 Stop Bit
Possible Values:

- ONE
- TWO

PARITY: RS232 Parity
Possible Values:

- NONE
- ODD
- EVEN

Note: Only 8bit data length is supported by the firmware.
The values for STOPBIT and PARITY can be either all upper case or all lower case.

Example:

1. -> ATB+IPR=9600, one, none
<- OK
⇨ *Sets the UART Baud Rate to 9600; StopBit to ONE and Parity to NONE.*

2. -> ATB+IPR=9600
<- Error
⇨ *Gives an error since all the parameters are needed.*

2.1.3 ATB + IFC =?.

This command gets the hardware flow control settings.

Response: <CR><LF >Enabled<CR><LF> if the Hardware flow control is enabled
<CR><LF >Disabled<CR><LF> if the Hardware flow control is disabled

Example:

1. -> **ATB+IFC=?**
   <- **Enabled**
   ⇨ *Implies that the Hardware flow control is enabled*

Note: The dongle reboots after the setting.

2.1.4 ATB + IFC =<setting>

This command enables or disables the hardware flow control for the module.

settings can be:
ENABLE : To enable the hardware flow.
DISABLE : To disable the hardware flow.

Response: <CR><LF >OK<CR><LF> if the set operation was successful.
<CR><LF >Error<CR><LF> if the operation was not successful.

Example:

1. -> **ATB+IFC=enable**
   <- **OK**

2.1.5 AT&D =?

This command controls the Data Terminal Ready (DTR) Signal.

Response: : <CR><LF><**n**><CR><LF>

Example:

1. -> AT&D=?
<- O

⇨ *Gives the DTR pin behavior setting, here its set to 0.*

2.1.6 AT&D <[n]>[,M/S].

This command sets the behaviors of the DTR line.

IF n=0 or not specified

If M is specified then

- On DTR=1 – Create connection; switch to data mode.
- On DTR=0 – Disconnect connection; switch to command mode; (Radio off).

If S is specified then

- On DTR=1 – Accept connection; switch to data mode.
- On DTR=0 – Break connection; refuse incoming connection; command mode; (Radio off).

IF n=1

If M is specified then

- On DTR=1 – Create connection and switch to data mode.
- On DTR=0 – Switch to command mode.

If S is specified then

- On DTR=1 – Switch to data mode, if connected else command mode.
- On DTR=0 – Switch to command mode.

In both case connection is kept active.
Note: In this case B+++ cannot be used.

IF n=2

- DTR signal is ignored

If M or S is not specified module behaves as per its current mode (Master or Slave) and AT&D settings.

Note: DTR settings for master will **only** work when the module is set on "Auto connect on DTR"

Example:

1. -> AT&D0,M0
   <- OK
   - ⇨ *Sets the DTR setting to 0 and the mode of module to MASTER (auto connect on power up since value 0).*
2. -> AT&D1,M
   <- OK
   - ⇨ *Sets the DTR setting to 1 and the mode of module to MASTER.*
3. -> AT&D2
   <- OK
   - ⇨ *Sets the DTR setting to 2. (This works since second argument is an optional one.*

Note: Here for setting the MASTER auto connect will only work if the DTR setting is set to 0. For example AT&D1,M1 will set the DTR value to 1 and the module in MASTER mode but will not change the Master mode to 1 (that is auto connect in DTR)

2.1.7 AT&S =?.

This command controls the behavior of Data Set Ready (DSR) Signal.

Response: : <CR><LF><**n**><CR><LF>

Example:

1. -> AT&S=?
<- O
⇨ *Gives the DSR pin behavior setting, here its set to 0.*

2.1.8 AT&S <n>

Note: The DSR settings are same for both MASTER and SLAVE.

IF n=0 or not specified

- Then DSR is set to 1 always.

IF n=1

- If DSR=0 (of Bluetooth signal) – the module switches to command mode.
- If DSR=1 (of Bluetooth signal) – the module switches to data mode.

IF n=2

- Handshake Library [Sets DSR pin of slave on receiving the DTR from master and vice-versa]

Example:

1. -> AT&S0
<- OK
⇨ *Sets the DSR setting to 0.*

2. -> AT&S1
<- OK
⇨ *Sets the DSR setting to 1.*

2.1.9 AT&C =?.

This command controls the behavior of Data Carrier Detected (DCD) pin.

Response: : <CR><LF><**n**><CR><LF>

Example:

1. -> AT&C=?
   <- O
   ⇨ *Gives the DCD pin behavior setting, here its set to 0.*

2.1.10 AT&C<n>

Note: The DCD settings are same for both MASTER and SLAVE.

IF n=0 or not specified

- Then DCD pin of the dongle is output pin and always set to 1.

IF n=1

- The DCD pin is set to output and reflects the remote one(that is it reflects the signal received from the Bluetooth connection).

IF n=2

- The DCD pin is set to the input and transmits the DCD signal through Bluetooth connection.

IF n=3

- The DCD pin is set to the input and any changes in the signal are ignored.

Example:

1. -> AT&C0
   <- OK
   ⇨ *Sets the DCD setting to 0.*

2. -> AT&C1
   <- OK
   ⇨ *Sets the DCD setting to 1.*

Note: The module reboots after the setting.

2.1.11 AT + MODE =?

This command returns the current mode of the SmartModule.

Response: <CR><LF><**mode**><CR><LF>

Note: If the mode is MASTER then auto connect setting is also displayed. [Refer 2.1.8 for auto connect settings]

Example:

1. -> **AT+MODE=?**
   <- **SLAVE**
   ⇨ *Gives the module mode, here its set to SLAVE.*

2. -> **AT+MODE=?**
   <- **MASTER 0**
   ⇨ *Gives the module mode, here its set to MASTER 0. Since the MASTER mode has 3 settings; the number after the MASTER string gives that settings. Here 0 means autoconnect on power up.*

2.1.12 AT + MODE =<mode>[,ConMode]

This command sets the mode of the SmartModule.

"mode" parameter can have the value SLAVE or MASTER [case sensitive].

The optional argument is only used when the mode to be set is MASTER. The ConMode variable can have the following values:

- 0 – For auto connect to remote device when power upped.
- 1 – For auto connect to remote device when DTR=1.
- 2 – For auto connect to remote device when serial activity is detected.

When in Master, the Module connects to the address specified by REMOTE BDA or to the last paired device.

Example:

1. -> **AT+MODE=SLAVE**
   <- **OK**
   ⇨ *Command sets the module mode to SLAVE.*

2. -> **AT+MODE=MASTER, 1**
   <- **OK**
   ⇨ *Command sets the module mode to MASTER and the autoconnect mode to AUTOCONNECT ON DTR SIGNAL.*

NOTE: The "SLAVE" and "MASTER" parameters are case sensitive.

2.2. GENERAL COMMANDS

2.2.1 AT

Response: <CR><LF>**OK**<CR><LF>

2.2.2 ATI

This command returns information related to SmartModule depending on the num argument.

If num = 0

Response: <CR><LF>***<manufacture identification >***<CR><LF>
<CR><LF>***<model identification>***<CR><LF>

Combination of AT+GMI & AT+GMM

If num = 3

Response: <CR><LF>***<firmware version>*** **<**<CR><LF>

Similar to AT+GMR

Else

Response: <CR><LF>**OK**<CR><LF>

2.2.3 AT + GMI.

This command returns manufacturer identification information.

Response: <CR><LF>***<manufacture identification>***<CR><LF>

2.2.4 AT + GMM.

This command returns models identification.

Response: <CR><LF>***<model identification>***<CR><LF>

2.2.5 AT + GSN.

This command returns local Bluetooth address.

Response: <CR><LF>**<BDA>**<CR><LF>

2.2.6 ATB + BDA

This command returns local Bluetooth address.

Response: <CR><LF>***<BDA>***<CR><LF>

Note: This command is exactly same as AT+GSN

2.2.7 AT + GMR.

This command returns the revised software version.

Response: <CR><LF>***<firmware version>***<CR><LF>

Note: The command is same as ATI3

2.2.8 ATB + RST.

This command is used to reset the factory settings.

Note: The command does not return anything just resets the settings and reboots.

2.3. DIAL COMMANDS

2.3.1 ATBD =<BDA>[,Cn]

This command is used for dialing a remote device specified by BDA.

Cn is an optional argument which can be UUID of remote device or the channel number.
-if Cn bet 1 and 20 => the argument is channel no.
-if Cn bet 1100 and 1114 => argument is UUID and channel is set to 0.

List of Bluetooth services (profiles/UUIDs):

Code Text Meaning
1101 SerialPort Serial port, serial data link without any restriction
1102 LANAccessUsingPPP LAN Access with PPP protocol
1103 DialupNetworking Dial Up Networking to establish switched connections to the ISDN or PSTN
1104 IrMCSync
1105 OBEXObjectPush OBEX Object Push
1106 OBEXFileTransfer OBEX Filetransfer
1107 IrMCSyncCommand
1108 *Headset access via Bluetooth*
1109 Cordless Telephony
1100 Intercom
1111 Fax
1112 *HeadsetAudioGateway Headset Gateway for audio signals*
1113 WAP
1114 WAP_CLIENT

Services
1 - SPP
2 - DUN
3 – SPPCONFIG

Note: If Cn argument is not specified then by default Cn is 1 (SPP)

Example:

1. -> ATBD=0010c617bdbd

There is no response from the command it just tries to connect the remote device whose BDA is as above.

2. -> ATBD=0010c617bdbd,3

The command will connect to the service 3 (SPPCONFIG) of the remote device whose BDA is as above.

2.3.2 ATBH

This command is used to disconnect the connection already created.

Response: <CR><LF>**Ok**<CR><LF> if disconnects
<CR><LF>**Error**<CR><LF> if error occurs

2.3.3 ATBO

This command is used to enter in data mode (Online mode/non AT mode).

Response: <CR><LF>**Ok**<CR><LF>
<CR><LF>**Error**<CR><LF> => If it cannot switch.

Note: Switching will only work if there is a carrier present i.e. a Bluetooth connection with remote device.

2.3.4 B+++

This command is used to enter in command mode (Offline mode/AT mode).

Response: <CR><LF>**Ok**<CR><LF>
<CR><LF>**Error**<CR><LF> - If it cannot switch

Note:
- The delay between the characters of B+++ should not be more then 1 sec else the command will not work. Also the command must be terminated by Carriage return.

- The command cannot be used when the DTR setting of the module is set to 1.

2.4. SPECIFIC COMMANDS TO BLUETOOTH

2.4.1 ATB + CPIN =?

This command returns the pin code of the SmartModule.

Response: <**Pincode>**

Note: Max length of the Pincode can be 16.

2.4.2 ATB + CPIN =<pin>

This command set the pin code of the SmartModule. Max Pin length=16

Response: <CR><LF>**Ok**<CR><LF> If it succeeds
<CR><LF>**Error**<CR><LF> If it fails

2.4.3 ATB + ENC =?.

This command returns the encryption value of the SmartModule.

Response: <CR><LF>**Enabled**<CR><LF> If encryption is enabled
<CR><LF>**Disabled**<CR><LF> If encryption is disabled

2.4.4 ATB + ENC =<encMode>

This command set the encryption mode of the SmartModule.

encMode parameter can take the following values:

Enable : To enable the encryption.
Disable : To disable the encryption.

Response: <CR><LF>**Ok**<CR><LF> If it succeeds
<CR><LF>**Error**<CR><LF> If it fails

2.4.5 ATB + INQ = maxDevice, TimeOut[,nameNeeded].

This command is used to do the Bluetooth inquiry of devices with settings as specified by the parameters.

maxDevice: Maximum number of devices that are to be responded.
timeOut: Timeout for inquiry in seconds.
nameNeeded: Optional field which when specified displays the names of the devices also.

Note: This command will only work if issued through UART and will return nothing if issued from Bluetooth.

Example:

1. -> ATB+INQ=10,15
   <- ***Device BDAs.....***
   <- .....
   <- OK
   ⇨ *The response is the device BDA's that are found*

2. -> ATB+INQ=10,15,1
   <- ***Device BDAs and Name.....***
   <- .....
   <- OK
   ⇨ *The response is the device BDA's along with the name of the devices found.*

Note: It is recommended that you inquire less number of devices since inquiring more devices will over flow the internal buffer and fail the inquiry !

2.4.6 ATB + BTNAME =?.

This command retrieves the module name.

Response: <CR><LF>**<*Module Name*>**<CR><LF>

2.4.7 ATB + BTNAME=<name>

This command set the name of the SmartModule. (Names up to 248 Bytes) (No default value, depends of the last name programmed)

Response: <CR><LF>**Ok**<CR><LF> If it succeeds.
<CR><LF>**Error**<CR><LF> If it fails.

2.4.8 ATB + SNIFF=? .

This commands the sniff value settings of the SmartModule.

Response: <CR><LF>**<sniff settings>**<CR><LF> If it succeeds

<sniff settings> : First 4 characters are f the Hexadecimal value for 'max interval' representation in ASCII.
Next 4 characters are f the Hexadecimal value for 'min interval' representation in ASCII.
Next 4 characters are f the Hexadecimal value for 'attempt' representation in ASCII.
Next 4 characters are f the Hexadecimal value for time out representation in ASCII.

2.4.9 ATB + PSNIFF=<num>

This command is used to set the sniff parameters to a predefined set of sniff values.

If num=0

In this mode the dongle has maximum throughput and high power consumption. It's no Sniff Mode. The values for parameters are:

Max Interval = 0x0000
Min Interval = 0x0000
Attempt = 0x0000
Timeout = 0x0000

If num=1

In this mode the dongle has medium throughput and low power consumption. The values for parameters are:

Max Interval = 0x00F0
Min Interval = 0x0050
Attempt = 0x0008
Timeout = 0x0008

If num=2

In this mode the dongle has low throughput and very low power consumption. The values for parameters are:

Max Interval = 0x0320

Min Interval = 0x0050
Attempt = 0x0008
Timeout = 0x0008

If num=3

In this mode the dongle has maximum throughput and low power consumption if there is no data else high power consumption. The values for parameters are:

Max Interval = 0x00F0
Min Interval = 0x0050
Attempt = 0x0008
Timeout = 0x0048

Response: <CR><LF>**Ok**<CR><LF>

1. -> ATB+PSNIFF=1
<- OK
⇨ *The sniff parameters will be set to the predefined values 1(see above).*

2.4.10 ATB + SNIFF=<maxInterval, minInterval, attempt, timeout>

This command is used to set the user defined sniff values.

Response: <CR><LF>**Ok**<CR><LF> If it succeeds in setting the sniff values.

Note:
- The maxInterval must be less then 5sec. i.e. < 0x1F40.
- Also minInterval < maxInterval.