AT Command Reference

80507AT10751A, Rev. 06 – 2016-05-24

TELIT
TECHNICAL
DOCUMENTATION

Stollmann is a Telit brand.

BlueMod+S

AT Command Reference

Release r06

Note

This product was developed for the purpose of communication. It is intended solely for our clients for integration into their own technical products after careful examination by experienced technical personnel for its suitability for the intended purpose. The product was not developed for or intended for use in any specific customer application. It may have to be adapted to the specific intended modalities of use or even replaced by other components in order to ensure flawless function in the respective areas of application. Performance data (range, power requirements, footprint, etc.) may depend on the operating environment, the area of application, the configuration, and method of control, as well as on other conditions of use; these may deviate from the technical specifications, the Design Guide specifications, or other product documentation. The actual performance characteristics can be determined only by measurements subsequent to integration in the target environment. Variations in the performance data of mass-produced devices may occur due to individual differences between such devices. Product samples were tested in a reference environment for compliance with the legal requirements applicable to the reference environment. No representation is made regarding the compliance with legal, regulatory, or other requirements in other environments. No representation can be made and no warranty can be assumed regarding the suitability of the product for a specific purpose. Telit reserves the right to make changes to the product without prior notice or to replace the product with a successor model. Of course, any changes to the product for which we have entered into a supply agreement with our customers will be made only if, and only to the extent that, such changes can reasonably be expected to be acceptable to our customers. No general commitment will be made regarding periods of availability; these must be subject to individual agreement.

Trademarks

Table of contents

1 Introduction

This documentation specifies the command interface for the BlueMod+S.

The BlueMod+S supports AT command mode and multiplexing mode.

1.1 AT Command Mode

Each command line consists of a prefix, a body and a terminator.

All command lines begin with the prefix **AT** (ASCII 065, 084) or **at** (ASCII 097, 116).

The body is a string of characters in the ASCII range 032-255. Control characters other than <CR> (carriage return; ASCII 013) and <BS> (back space; ASCII 008) in a command line are ignored.

Note: The control characters are configurable via S registers.

- Carriage return character (CR) *S3 register*
- Line feed character (LF) *S4 register*
- Back space character (BS) *S5 register*

The terminator is <CR>.

There is no distinction between upper-case and lower-case characters. A command line can have a maximum length of 80 characters. If the maximum length is reached any additional character will be discarded. Corrections are made using <BS>. Multiple commands on the same command line are not allowed.

Commands have the following syntax:

Syntax	Description
AT<command>=<value><CR>	Write the value of the command

Responses are sent back to the host and can be any of the following:

Responses	Description
<CR><LF>value<CR><LF>	Read only value (e.g. AT+BOAD)
<CR><LF> list entry 1<CR><LF> list entry 2<CR><LF> … list entry n<CR><LF> <CR><LF>	List value (e.g. AT+BNDLIST)
<CR><LF>OK<CR><LF>	Successful final message
<CR><LF>ERROR<CR><LF>	Error message, command not supported

Note: Exceptions of this syntax are marked seperately.

1.2 Escape Sequence

To enter the command mode during an active data connection the following sequence (escape sequence) can be used:

<delay time[1]>**<+><+><+>**<delay time>

The time interval between each of the three plus signs must not exceed 1 second. The escape sequence remains transparent to the remote device.

Note: The escape sequence character is configurable via S2 register.

1.3 Multiplexing Mode (MUX)

The multiplexing mode is used to handle incoming and outgoing data of different remote endpoints and command data.

Data has to be sent and are received in the following framing (all values in hexadecimal format):

Name	Description	Length	Value
Start	Start of frame	8 bit	CC
Channel ID	Channel identifier	8 bit	00 – FF
Length	Length of data	8 bit	-
Data	Max. 255 bytes data	Min. 0 byte Max. 255 bytes	-

The start byte is used to detect the start of a frame.

The channel ID determines the channel to send data to. This can be the link ID of an active link in case of classic Bluetooth, the data channel of a BLE endpoint or the AT command interface (value FF).

The length field sets the length of the payload to send or to received in bytes.

The data field consists of the payload data to send or receive.

Start of frame, channel ID, length and data are always transmitted in direct, binary form. AT commands have to be sent to the channel ID FF, simply prefixed with start of frame, 0xFF, and length byte. Data received from the AT command interface are marked by channel ID FF. Line editing using backspace is not available in multiplexing mode.

[1] Delay time defined in the S12 register (see page 16)

Examples:

CC 01 0B 31 32 33 34 35 36 37 38 39 30 0D	Send data “1234567890<CR>” to channel 1 via MUX protocol
CC FF 06 41 54 49 39 39 0D	Send AT command “ATI99<CR>” via MUX protocol
CC FF 06 0D 0A 4F 4B 0D 0A	Receive response “<CR><LF>OK<CR><LF>” from AT command interface via MUX protocol

Result messages like RING, CONNECT and NO CARRIER may be sent in multiple frames by the BlueMod+S module. The host controller application needs to collect the data until the closed “<CR><LF>” is received.

If the multiplexing mode is enabled the commands **ATH** and the result messages RING, CONNECT and NO CARRIER includes the additional parameter “channel ID”.

For Terminal I/O connections, the “channel ID” is assigned while connection setup (by messages **RING** for incoming connections and **CONNECT** for outgoing connections) and is valid until the connection is terminated (message **NO CARRIER**).

For customer defined GATT services, the “channel ID” is assigned while service definition for each defined characteristic separately.

1.4 Hangup

An active data connection can be disconnected by setting the HANGUP pin (GPIO4) to high level.

2 Commands

The description of the commands is structured into the following parts:

- General commands
- Bluetooth Low Energy (BLE) specific commands
- Generic Attribute Profile (GATT) specific commands

The factory-default values of the commands are marked using the bold letter format.

2.1 General

%B	Baud Rate

AT syntax: **AT%B<value>**

This command determines the baud rate for the UART.

The following standard baud rates are supported:

Value	Description
4	9,600 bps
5	19,200 bps
6	38,400 bps
7	57,600 bps
8	115,200 bps
9	230,400 bps
22	460,800 bps
23	921,600 bps

Additionally to the standard baud rates described above, it is possible to set the following baud rates.

Value	Description
1200	1,200 bps
2400	2,400 bps
4800	4,800 bps
9600	9,600 bps
14400	14,400 bps
19200	19,200 bps
28800	28,800 bps
38400	38,400 bps
57600	57,600 bps
76800	76,800 bps
115200	115,200 bps
230400	230,400 bps
250000	250,000 bps
460800	460,800 bps
921600	921,600 bps
1000000	1,000,000 bps

To set a baud rate write the desired value in the command.

Examples:

AT%B9	Set baud rate to 230,400 bps
AT%B250000	Set baud rate to 250,000 bps

Note: Information regarding the deviation of the real baud rate to the value set can be found in the BlueMod+S Hardware Reference [1].

&F Load Factory Defaults

AT syntax: **AT&F<value>**

The factory-default values will be loaded. For storing values in non-volatile memory, use the **AT&W** command.

Value	Description
0	Set all parameters except **bndlist** to factory defaults
1	Set all parameters to factory defaults

Note: Some restored values require an additional reset to get active (e.g. AT+LETIO, AT+UICP).

&W Store Active Configuration

AT syntax: **AT&W**

The active configuration is stored in non-volatile memory.

A Accept Incoming Call

AT syntax (normal mode): **ATA**
AT syntax (MUX mode): **ATA <channel ID>**

This command is supported for compatibility reasons only. A Bluetooth low energy device has to answer every connection automatically. See also register S0 description.

Examples:

ATA	Accepts connection in normal mode
ATA 0x01	Accepts connection in MUX mode at channel ID 0x01

+BIOCAP SSP I/O Capabilities

AT syntax: **AT+BIOCAP=<value>**

This command sets the input and output capabilities of the device used for SSP.

Value	Description
0	Display only
1	Display Yes/No
2	Keyboard only
3	No input no output
4	Display and keyboard

+BMITM SSP Man in the middle protection

AT syntax: **AT+BMITM=<value>**

This command controls the man in the middle (MITM) protection of the device during SSP. It has to be set in context with **AT+BIOCAP** command. In SSP there are scenarios where MITM protection is not possible.

Value	Description
0	Man in the middle protection disabled
1	Man in the middle protection enabled

In case the user choose a scenario where MITM protection is not possible but one of the communication devices is configured to **AT+BMITM=1** (MITM protection enabled), the pairing is refused.

Possible combinations of I/O capabilities and the possibility of MITM protection are described in the *BlueMod+S User Guide [2].*

+BMUX Activate Multiplexing Mode

Supported since firmware version 1.101

AT syntax: **AT+BMUX=<value>**

This command is used to activate the multiplexing mode protocol.

Value	Description
0	Normal AT mode
1	Non persistent multiplexing mode
2	Persistent multiplexing mode

In the default configuration the device is working in normal AT mode. Setting **AT+BMUX=1** enables the multiplexing mode.

*Note: After receiving "OK" in the response of the **AT+BMUX=1** command all subsequent commands have to be entered in multiplexing frame format.*

The multiplexing mode 1 is not stored persistent. To disable the multiplexing mode the device must be reset.

The multiplexing mode 2 is stored persistent. To disable the multiplexing mode the command AT+BMUX=0 followed by AT&W must be entered. This mode is intended to be used with the AT+SYSTEMOFF command.

In multiplexing mode extended result codes are always active (see chapter 3.3 Extended Result Codes).

+BNAME Local Device Name

AT syntax: **AT+BNAME=<name>**

This command allows the modification of the local device name. The device name is shown on a remote Bluetooth device during device/service discovery. It is limited to 19 characters.

The device name can contain a format element to include the device's own address or parts of it in the name.

In BLE advertising the name is truncated to the first 10 characters.

Format: "%[<s>][<d>]a"

"%"	Identifier start format element
<s>	Character separator on byte order (optional)
<d>	Number (1-12) of digits included in device name (optional, default is 4)
"a"	Identifier end format element

Examples: Device address = "0123456789AB"

AT+BNAME=BM+S %4a	Display on remote end: BM+S 89AB
AT+BNAME=BM+S %4a	Display on remote end: BM+S 89AB
AT+BNAME=BM+S %:3a	Display on remote end: BM+S 9:AB
AT+BNAME=BM+S %3a	Display on remote end: BM+S 9AB
AT+BNAME=BM+S %:12a	Display on remote end: BM+S 01:23:45:67:89:AB

+BNDDEL Delete Bonding Information

AT syntax: **AT+BNDDEL=<value>**

This command deletes the bonding information stored by the BlueMod+S.

Value	Description
Bluetooth address	Delete the bond of the device with specified address from the bonded-device list
*	Delete all bonded devices from the bonded-device list

+BNDLIST Show Bonded Device List

AT syntax: **AT+BNDLIST**

This command shows information about the devices bonded with the BlueMod+S.

Each entry in the **bonded-device list** contains the Bluetooth address and the linktype (see chapter 3.1).

Example:

AT+BNDLIST	0080254800DA 0x02 9C04EB06ACA2 0x0A OK

+BNDS Storage Mode for Bonds

AT syntax: **AT+BNDS=<mode>**

This command controls the storage mode for bonding information (link keys).

Mode	Description
0	Bonds persists for the duration of the authenticated connection
1	Bonds are permanently stored in the NVRAM of the BlueMod+S

*Note: By setting **AT+BNDS** to 0 the bonded-device list is deleted internally.*

Note: The bonding information is stored in the module flash. If your application does not need to store bonds switch this parameter to 0, to protect the module flash from unnecessary clear and write operations. Every flash has a limited number of clear cycles.

+BNDSIZE Bonded Devices List Size

AT syntax: **AT+BNDSIZE=<value>**

This command reduces the number of entries (1…4) the bonded-device list can hold.

The BlueMod+S can store up to 4 devices. The default size is **4**.

Note: Modification of this parameter will delete all entries in the bonded-device list.

Example:

AT+BNDSIZE=1	Limit the number of bonded devices to 1

+BOAD Bluetooth Own Device Address

AT syntax: **AT+BOAD**

This command reads the Bluetooth devices' own device address.

Note: This command is read only.

+BPAIRMODE Configure Pairable Mode

Supported since firmware version 1.101

AT syntax: **AT+BPAIRMODE=<mode>**

This command controls the pairable mode of the BlueMod+S.

When set to “0” the module is only connectable for clients stored in the local bondlist. New pairing requests will be rejected.

Mode	Description
0	No pairing allowed, BlueMod+S advertises TIO as “functional”
1	Pairing allowed, BlueMod+S advertises TIO as “bondable and functional”

Note: This command restricts the access only to security enabled characteristics.

- *For Terminal I/O this means AT+LETIO has to be set to 1.*
- *For the GATT Server the permission parameters of the command AT+LEATTRIB have to be set to a value that requires authentication (e.g. “AT+LEATTRIB=charval,uuid=2A19,perm=0022,len=1” with “perm” for read with authentication permitted and write with authentication permitted).*

+BSSPPIN SSP Passkey Response

AT syntax: **AT+BSSPPIN <Bluetooth address>,<SSP passkey>**

If a SSP is initiated and MITM is active (**AT+BMITM=1**), depending on the I/O capabilities (**AT+BIOCAP**) the AT interface generates an event SSPPIN and asks the user for the SSP passkey.

Event: SSPPIN <Bluetooth address> ?

The user has to answer this request with the SSP passkey displayed on the remote device.

Example:

SSPPIN 00802507C08D ? AT+BSSPPIN 00802507C08D,314546 OK RING CONNECT	Receive SSP passkey request Send SSP passkey response

+DFUMODE Device Firmware Update Mode

Supported since firmware version 2.000 and bootloader version 2.000

AT syntax: **AT+DFUMODE=<value>**

This command sets the device firmware update mode.

To activate the mode it is necessary to store the settings and perform a reset or use the command AT+DFUSTART.

Value	Description
1	Device firmware update over serial interface
2	Device firmware update over the air (OTA)

+DFUNAME Over The Air Update Name

Supported since firmware version 2.000 and bootloader version 2.000

AT syntax: **AT+DFUNAME=<name>**

This command sets the device name for the over the air firmware update mode.

The name is limited to 8 characters.

To store the modified name persistent use the AT&W command.

Name	Description
BM+S_DFU	Device firmware update name is “BM+S_DFU”

+DFUSTART Start Bootloader

Supported since firmware version 2.000 and bootloader version 2.000

AT syntax: **AT+DFUSTART**

This command sets the device into the configured firmware update mode. The command times out after 2 minutes.

E Local Echo

AT syntax: **ATE<value>**

This command selects the local echo in command mode.

Value	Description
0 or empty	No local echo
1	Local echo on in command phase

H Disconnect

AT syntax (normal mode): **ATH**
AT syntax (MUX mode): **ATH <channel ID>**

This command disconnects the existing Bluetooth connection.

Examples:

ATH	Disconnects connection in normal mode
ATH 0x01	Disconnects connection in MUX mode at channel ID 0x01

I Display Version Information

AT syntax: **ATI<value>**

Displays different information about version number and settings.

Value	Description
0 or empty	Returns the device name (e.g. “BlueMod+S“)
1	Returns “0”
2	Returns “OK”
3	Returns the version string: “V1.xyz”
4	Returns the manufacturers name: “Stollmann E+V GmbH”
5	Returns “ERROR”
6	Returns the copyright string: “(c) Copyright Stollmann E+V GmbH”
7	Returns “OK”
8	Returns “ERROR”
9	Returns “OK”
77	Returns bootloader version
99	Returns the firmware creation date

+IOACFG Config of Pin IOA

AT syntax: **AT+IOACFG=<value>**

This command configures the function of the IOA pin (GPIO8). GPIO8 can be configured as output signal “Device Ready". The signal is low active, a low level shows the device ready to process commands and establish Bluetooth connections after startup.

Value	Description
0	Input: Analog (no function, lowest power consumption)
1	Output: Signal “Device Ready”

+IOBCFG Config of Pin IOB

Supported since firmware version 2.000

AT syntax: **AT+IOBCFG=<value>**

This command configures the function of the IOB pin (GPIO3). GPIO3 can be configured as output signal "Data Carrier Detect" (DCD). The signal is low active. A low level signals that the device is connected on Terminal I/O level to a remote device.

Value	Description
0	Input: Analog (no function, lowest power consumption)
1	Output: Signal “DCD”
2 *)	Output: Low level if a lower layer connection is active Output: High level if no lower layer connection is active

*) *Supported since firmware version 2.006*

+LOAD Load Stored Parameter Setting

AT syntax: **AT+LOAD**

This command loads all parameters stored in non-volatile RAM.

O — Return to Online State

AT syntax: **ATO**

If the BlueMod+S is in command mode after issuing an escape sequence while a connection is active, ATO returns the BlueMod+S to data mode.

Q — Suppress Results

AT syntax: **ATQ<value>**

This command allows/suppresses result codes and messages.

Value	Description
0 or empty	Enable result messages after command input
1	Suppress result messages after command input

+RESET — Reset Device

AT syntax: **AT+RESET**

This command resets the whole functionality of the BlueMod+S by a forced hardware reset (like power off/on).

+RFMAXTXPWR — Maximum Output Power

AT syntax: **AT+RFMAXTXPWR=<value>**

This command sets the maximum output power of the Bluetooth radio of the device. A changed value becomes active immediately.

Value	Description
-128	Use factory default maximum output power of 4 dBm
4	4 dBm
0	0 dBm
-4	-4 dBm
-8	-8 dBm
-12	-12 dBm
-16	-16 dBm
-20	-20 dBm
-30	-30 dBm

All other values in the range of -127 to 127 could be set with this command as well, but the equal or next lower value from the power table will be set internally. Furthermore the value will be set to a value amongst minimum and maximum output power value of the device.

Example:

AT+RFMAXTXPWR=0	The maximum output power will be set to 0 dBm

S — AT S Register

AT syntax: **ATSx=<value>**

This command configures the S register settings.

Mode	Description
Sx?	Show the current setting of register Sx
Sx=1	Set register Sx to 1

AT command S register set (all values in decimal format):

Register	Value	Description
S0	**1**	Immediate call acceptance *Note: Setting of S0 only allow value 1 for BLE. In case of BLE connections always one RING is signalised and automatic call acceptance is set.*
S2	**43**	Character for escape sequence
S3	**13**	Carriage-return character
S4	**10**	Line-feed character
S5	**08**	Backspace character
S12	**100**	Delay time by using the escape sequence in 10 ms increments

+SYSTEMOFF — Enter System Off Mode

Supported since firmware version 2.000

AT syntax: **AT+SYSTEMOFF**

This command sets the module into low power mode during the time the module is not used. The module will restart on GPIO activity. The host can use the IOA pin to monitor the system status.

Possible use cases and an usage example are described in the *BlueMod+S User Guide [2].*

+UICP Set UART Interface Control Protocol

AT syntax: **AT+UICP=<mode>**

This command sets the mode of the UART Interface Control Protocol (UICP).

To activate UICP, it is necessary to store the settings and perform a reset.

Mode	Description
0	UICP off
1	UICP on

The valid sequence to e.g. activate UICP is:

AT+UICP=1

AT&W

AT+RESET

V Result Message Format

AT syntax: **ATV<value>**

This command determines the format of the result messages.

Value	Description
0 or empty	Result message is presented numerically (followed by <CR>)
1	Result message is presented as text

W Extended Result Codes

AT syntax: **ATW<value>**

This command enables/disables extended result codes.

Value	Description
0 or empty	Result message is presented without extended result codes
1	Result message is presented with extended result codes (include error causes)

2.2 Bluetooth Low Energy

+LEADINTMAX Maximum Advertising Interval

AT syntax: **AT+LEADINTMAX=<value>**

This command configures the maximum advertising interval (in milliseconds) for a Bluetooth Low Energy Peripheral.

Value	Description
n=20…10240	Use maximum advertising interval of *n* ms
1280	Use maximum advertising interval of 1280 ms

*Note: Make sure that the value of **AT+LEADINTMAX** is higher or equal the value of **AT+LEADINTMIN**.*

+LEADINTMIN Minimum Advertising Interval

AT syntax: **AT+LEADINTMIN=<value>**

This command is not used in the BlueMod+S. It is just provided for compatibility reasons. The used advertising interval is set by AT+LEADINTMAX parameter.

+LECONINTMAX Maximum Connection Interval

AT syntax: **AT+LECONINTMAX=<value>**

This command configures the maximum connection interval for a Bluetooth Low Energy connection. The unit is in 1.25 milliseconds timeslots.

Value	Description
n=6…3200	Use maximum connection interval of *n* * 1.25 ms
32	Use maximum connection interval of 40 ms

*Note: Make sure that the value of **AT+LECONINTMAX** is higher or equal the value of **AT+LECONINTMIN**.*

+LECONINTMIN Minimum Connection Interval

AT syntax: **AT+LECONINTMIN=<value>**

This command configures the minimum connection interval for a Bluetooth Low Energy connection. The unit is in 1.25 milliseconds timeslots.

Value	Description
n=6…3200	Use minimum connection interval of *n* * 1.25 ms
16	Use minimum connection interval of 20 ms

*Note: Make sure that the value of **AT+LECONINTMAX** is higher or equal the value of **AT+LECONINTMIN**.*

+LEROLE Bluetooth Low Energy Device Role

AT syntax: **AT+LEROLE=<value>**

This command configures the Bluetooth Low Energy role of the device.

Value	Description
0	Set device role to “Peripheral”

When set to “Peripheral” (default), the device advertises and accepts incoming BLE connections. Searching for Bluetooth Low Energy devices with **AT+LESCAN** command is not possible.

+LESLAVELAT Slave Latency

AT syntax: **AT+LESLAVELAT=<value>**

This command configures the slave latency (in connection intervals) for a Bluetooth Low Energy connection.

Value	Description
n=0…499	Use a slave latency of *n* connection intervals
0	Use no slave latency

+LETIO Enable Terminal I/O Service

AT syntax: **AT+LETIO=<value>**

This command controls the Terminal I/O service. If set to 0 the Terminal I/O service is disabled. To activate the change, it is necessary to store the settings (**AT&W**) and perform a reset (**AT+RESET**).

Value	Description
0	Terminal I/O service disabled (no advertising, no characteristics)
1	Terminal I/O service enabled, security is required
2	Terminal I/O service enabled, no security required

The valid sequence to change the setting is:

AT+LETIO=1

AT&W

AT+RESET

+LEADDATA Setup Advertise Data for Customized Advertising

Supported since firmware version 1.101

AT syntax: **AT+LEADDATA=<value$_1$> .. <value$_n$>**

This command is used to setup the advertise data for a customized advertising.

Value$_k$ represents an octet in hexadecimal format, k <= 31.

The coding of the data is according to the *Bluetooth 4.0 Core Specification / Vol. 3, Part C, Chapter 11 and 18 (Length/Type/Value coding) [3]*.

Example:

41 54 2B 4C 45 41 44 44 41 54 41 3D 30 32 30 31 30 36 30 33 30 32 30 46 31 38 0D *(human readable: AT+LEADDATA=02010603020F18)*	Set flags + UUID of battery service

+LEADE Enable Customized Advertising

AT syntax: **AT+LEADE=<value>**

This command controls the advertising behavior.

With **AT+LEADE=0** only the build in Terminal I/O service is advertised.

With **AT+LEADE=1** only the customized advertising value is advertised.

With **AT+LEADE=3** the module stops all advertising. With disabled advertising the client (e.g. iPhone) is not able to discover the device or to connect to the device. This should only be done when the service is not in use to save battery power.

With disabled internal TI/O due to **AT+LETIO=0**, the values 0 and 3 show the same behavior. There will be no advertising and no connection.

Value	Description
0	Customized advertising disabled, internal TIO advertising enabled
1	Customized advertising enabled, internal TIO advertising disabled
2	Reserved for future use
3	Advertising off, customized advertising disabled, internal TIO advertising disabled

+LEADPAR Setup Parameters for Customized Advertising

Supported since firmware version 1.101

AT syntax: **AT+LEADPAR=par$_1$=<value$_1$>[, .. [,par$_n$=<value$_n$>]]**

This command is used to setup parameters for a customized advertising.

par$_n$	**value$_n$**
advtype Optional Coding: decimal.	Type of advertising: **0: undirected (default)** 2: scannable 3: non-connectable

This command is optional, if not submitted these defaults apply:

- advtype = 0 - advertising type “undirected”

*Note: The values for the minimum and maximum of the advertising interval may be set with the **AT+LEADINTMIN** and **AT+LEADINTMAX** commands.*

Example:

AT+LEADPAR=ADVTYPE=0	Set type of advertising “undirected”

+LESCDATA Setup Scan Response Data for Customized Advertising

Supported since firmware version 1.101

AT syntax: **AT+LESCDATA=<value$_1$> .. <value$_n$>**

This command is used to setup the scan response data for a customized advertising.

Value$_k$ represents an octet in hexadecimal format, k <= 31.

The coding of the data is according to the *Bluetooth 4.0 Core Specification / Vol. 3, Part C, Chapter 11 and 18 (Length/Type/Value coding) [3]*.

Example:

41 54 2B 4C 45 53 43 44 41 54 41 3D 30 33 30 32 30 46 31 38 0D *(human readable: AT+LESCDATA=03020F18)*	Set UUID of battery service

2.3 GATT

All commands described in this chapter can only be used in multiplexing mode (**AT+BMUX=1**).

2.3.1 GATT Server Definition

+LEATTRIB Define Attributes for a Service

Supported since firmware version 1.101

AT syntax: **AT+LEATTRIB=<type>[,par$_1$=<value$_1$>[, .. ,par$_n$=<value$_n$>]..]]**

This command is used to define attributes for one or more services in the GATT server. The maximum number of services and characteristics depends on the used features. Every combination results in a different number of possible service and characteristic combinations. The maximum number of possible characteristics is limited to 20. This restriction results from the limited size of the internal definition

array. If the space is completely used the command “AT+LEATTRIB=charval,…” returns “ERROR”.

The GAP and GATT services that each GATT server must expose are built-in services in the BlueMod+S and thus shall not be defined by the application!

The presence of parameters *par*$_1$,..,*par*$_n$ depends on the value of <*type*>:

type	**par$_1$**	**par$_2$**	**par$_3$**	**par$_4$** (1)
pserv Mandatory	uuid=<*16/128bit UUID*> Mandatory Coding: hexadecimal.			
char Mandatory	prop=<*properties*> Mandatory Coding: hexadecimal. *properties* may have the bitmask values *Read*, *Write Without Response*, *Write*, *Notify* and *Indicate* set. Note: internally the controller generates an additional Client Characteristic Configuration Descriptor (CCCD) attribute with permissions „readable and writable without authentication or authorization“ if bits *Notify* or *Indicate* are set.			
charval Mandatory	uuid=<*16/128bit UUID*> Mandatory Coding: hexadecimal.	perm=<*permissions*> Mandatory Coding: hexadecimal. 16-bit value that decodes the access permissions and authentication requirements.	len=<*length*> Mandatory Coding: decimal. Maximum supported length is 20 bytes. A length of 0 means a variable length of the characteristic (up	val=<*value*> Optional Coding: hexadecimal. Pre-defined characteristic value that will be loaded when the service set is activated.

type	par_1	par_2	par_3	par_4 (1)
			to 20 bytes)	
chardcccd Optional	perm=<*permissions*> Mandatory Coding: hexadecimal. This command is needed only if a CCCD shall be generated with permissions other than „readable and writable without authentication or authorization“. See note in *type*=char description.			
chardusrd Optional	usrd=<*user description*> Mandatory Coding: UTF-8 string. User description string. Note: - Since *Extended Properties* (see *type*=char, parameter *properties* description) are not supported the remote peer may not write to this characteristic descriptor. - The number of characters is limited to 13. Internally the character sequence is terminated by a zero byte resulting in an ASCII-Z string.	perm=<*permissions*> Optional Coding: hexadecimal. 16-bit value that decodes the access permissions and authentication requirements. If this parameter is omitted the default *readable without authentication or authorization* applies.		
complete Mandatory	No parameter. Used to signal that all attribute definitions have been sent to the controller.			

(1) par_4 supported since firmware version 2.000

The characteristic properties are coded as a hexadecimal bitmask as defined in *Bluetooth Core Spec 4.0 Volume3 Part G Chapter 3.3.1.1 [3]*.

Value	Properties
02	Read
04	Write without response
08	Write
10	Notify
20	Indicate

The values can be combined, for example read & notify result in 12.

The attribute permissions (parameter perm=<*permissions*>) are bit coded in a 16 bit hexadecimal value.

Bit	Value	Function
0,1	0 1 2 3	Read not permitted Read permitted Read with authentication permitted Read with authentication and MITM protection permitted
2..3	-	Reserved
4,5	0 1 2 3	Write not permitted Write permitted Write with authentication permitted Write with authentication and MITM protection permitted
6..15	-	Reserved

Example: Value for read only is 0001.

The complete service/s is/are defined through repeated submissions of the **AT+LEATTRIB** command (see example below).

The **AT+LEATTRIB** commands must be submitted in a specific order:

Definition of first service:

AT+LEATTRIB=pserv, …

Definition of first characteristic of first service:

AT+LEATTRIB=char, …

… optional characteristic descriptors (AT+LEATTRIB=chardxxx) …

AT+LEATTRIB=charval, …

Definition of second characteristic of first service:

AT+LEATTRIB=char, …

… optional characteristic descriptors (AT+LEATTRIB=chardxxx) …

AT+LEATTRIB=charval, …

…

Definition of second service:

AT+LEATTRIB=pserv, …

Definition of first characteristic of second service:

AT+LEATTRIB=char, …

… optional characteristic descriptors (AT+LEATTRIB=chardxxx) …

AT+LEATTRIB=charval, …

…

…

Completion of service and characteristics definition:

AT+LEATTRIB=complete

Upon successful execution of the **AT+LEATTRIB** command with *type=charval* the device returns the data channel ID which is associated to the characteristic value.

Format: <CR><LF>0x<*channel*><CR><LF>

Data belonging to the characteristic value is exchanged over multiplexer data channel ID *channel*. The channel value is coded in hexadecimal digits.

Example:

The example below shows the Battery Service. Battery Service is a simple service which exposes the battery charging level as single characteristic value.

Command	Response	Description
CC FF 1C 41 54 2B 4C 45 41 54 54 52 49 42 3D 70 73 65 72 76 2C 75 75 69 64 3D 31 38 30 46 0D *(human readable: AT+LEATTRIB=pserv,uuid=180F)* CC FF 19 41 54 2B 4C 45 41 54 54 52 49 42 3D 63 68 61 72 2C 70 72 6F 70 3D 31 32 0D *(human readable: AT+LEATTRIB=char,prop=12)*	CC FF 06 0D 0A 4F 4B 0D 0A *(OK)* CC FF 06 0D 0A 4F 4B 0D 0A *(OK)*	Declares the properties of the battery level value
CC FF 2E 41 54 2B 4C 45 41 54 54 52 49 42 3D 63 68 61 72 76 61 6C 2C 75 75 69 64 3D 32 41 31 39 2C 70 65 72 6D 3D 30 30 30 31 2C 6C 65 6E 3D 31 0D *(human readable: AT+LEATTRIB=charval,uuid=2A19,perm=0001, len=1)*	CC FF 08 0D 0A 30 78 30 32 0D 0A CC FF 06 0D 0A 4F 4B 0D 0A *(0x02 OK)*	Declares the battery level value (one byte in the range 0,..,100). Battery level values are exchanged over multiplexer channel 2
CC FF 15 41 54 2B 4C 45 41 54 54 52 49 42 3D 63 6F 6D 70 6C 65 74 65 0D *(human readable: AT+LEATTRIB=complete)*	CC FF 06 0D 0A 4F 4B 0D 0A *(OK)*	Completes the service definition sequence

+LESRVSETOPEN Open a Service Set for Definition

Supported since firmware version 2.000

AT syntax: **AT+LESRVSETOPEN=<value>**

This command allows to define service sets. At the moment only 1 service set is supported.

The AT+LESRVSETOPEN command is only accepted when no service was previously activated with the AT+LESRVSETACT command or defined with AT+LEATTRIB.

The service set content is defined with the AT+LEATTRIB command, can be persistently stored with the AT+LESRVSETSAVE command.

The activation depends on the parameter AT+LESRVBOOTMODE. If AT+LESRVBOOTMODE is set to 0 the service set is activated with the AT+LESRVSETACT command.

If the AT+LESRVSETOPEN command is used to define a service set that exists already in flash memory the existing service is deleted from flash.

If the AT+LESRVSETOPEN command is submitted with parameter value = 0 a test mode is entered: services and characteristics can be defined without storing these in the flash memory. Each AT+LEATTRIB command is mapped to an operation on the Nordic SoftDevice API and the command parameters are immediately checked for integrity. The behavior is different for a non-zero parameter value: the parameters of the AT+LEATTRIB command are written to flash without preceding check for integrity. The check is made when the service set is activated with the AT+LESRVSETACT command. The reason is due to the fact the SoftDevice cannot remove registered services, defining more than one service set would require a target reset before a new service can be defined.

AT+LESRVSETOPEN with parameter value = 0 can be used to “develop” resp. “debug” a new service. Once it is found to be OK it can be then re-defined and stored with the validated command sequence for a non-zero service set.

Value	Description
n=0	Select dummy service set
n=1	Select service set n for new service definitions

+LESRVSETSAVE Save a Service Set Definition

Supported since firmware version 2.000

AT syntax: **AT+LESRVSETSAVE**

This command persistently stores the service set definition that was previously addressed with the AT+LESRVSETOPEN command and returns a 16 bit checksum for the activated set as human readable 0xXXXX string.

The checksum is calculated “live” at save time and can be used as data consistency indicator while later activation of a set since the checksum of the set activation is also calculated live while activation and shall result in the same value in case the set data is not corrupted due to FLASH or RAM failure.

+LESRVSETACT Activates a Service Set

Supported since firmware version 2.000

AT syntax: **AT+LESRVSETACT=<value>**

This command allows to activate a service set stored in the NRF51822_FW.

In case any service set is already activated, the command will respond with “ERROR”.

In case the definitions of the activated services are detected as not consistent at GATT level, the command will respond with “ERROR”.

In case the definitions of the activated services are successfully enabled, the command will return a 16 bit checksum for the activated set as human readable 0xXXXX string followed by “OK”.

That checksum is calculated “live” at activation time and can be used as data consistency indicator while later activation of a set since the checksum of the set save operation was also calculated live while storage and shall result in the same value in case the set data is not corrupted due to FLASH or RAM failure.

Value	Description
n=1	Select service set n for activation. The NRF51822_FW will make all service information that are defined for the activated service set available via BLE and create one channel ID for each characteristic value of the service set definition

+LESRVBOOTSET Service Set for Boot Activation

Supported since firmware version 2.000

AT syntax: **AT+LESRVBOOTSET=<value>**

This command allows to define the service set used by AT+LESRVBOOTMODE command. At the moment only one service set 1 is supported.

Value	Description
1	Service set is loaded during system startup depending on parameter AT+LESRVBOOTMODE

+LESRVBOOTMODE Boot Behavior of Stored GATT Service Set

Supported since firmware version 2.000

AT syntax: **AT+LESRVBOOTMODE=<value>**

This command allows to define the behavior of the module during system startup concerning the stored service set.

If you use the AT+SYSTEMOFF command it is recommended to use value=2 to avoid unnecessary output during system startup.

Value	Description
0	Service set is not loaded during system startup. Use AT+LESRVSETACT command to activate service set
1	Service set is loaded during system startup. The channel ID/UUID and the CRC are sent on the serial port
2	Service set is loaded silently during system startup. No output on the serial port

2.3.2 GATT Server Data Handling on MUX Channel

MUX channels are created during GATT server definition using the **AT+LEATTRIB** command. After "**AT+LEATTRIB=complete**", the GATT server is ready to be used.

All data for defined characteristics is sent in one MUX frame on the corresponding channel so a MUX frame has to have the defined data length for the characteristic addressed. If a characteristic has a length of 4, 4 bytes of data has to be sent. The only exception is a defined length of 0, which means a variable length characteristic. All data sizes between 1 and 20 are allowed.

By default the value of a characteristic is set to 0. It is recommended that the application sets initial data values for all characteristics immediately after registration of all characteristics.

2.3.2.1 Data Handling with or without Connection

All data which is set during a connection is directly sent to the remote side if characteristic definition supports indication or notification. If indications or notifications are not supported only the local server value is updated.

All data which is set without a connection is only updated in the local server.

Only the last value written to a characteristic is stored in the local server.

With every new connection all not signaled data in the server is sent over the link if possible.

2.3.2.2 Error Handling

All MUX frames with a wrong channel id or a wrong data size are silently discarded.

There is no definition for an error response at the moment in MUX protocol.

3 Appendix

3.1 Linktype

Linktype	Meaning
0x02	Bluetooth low energy using public address
0x03	Bluetooth low energy using random address
0x0A	Bluetooth low energy using resolved address

3.2 AT Result Codes

Result codes (numerical and verbose):

Numeric	Text	Meaning
0	OK	Command completed
1	CONNECT	Connection established
2	RING	Indicates an incoming call (link request received)
3	NO CARRIER	Connection disconnected
4	ERROR	Illegal command or error that cannot be indicated otherwise

3.3 Extended Result Codes

Extended result codes (numerical and verbose) are available after activation with **ATW1** command.

In multiplexing mode extended result codes are always active.

3.3.1 CONNECT

For the AT result code CONNECT the following extended result codes are available:

Syntax normal mode: CONNECT <<Bdaddr> <linktype>>
Syntax MUX mode: CONNECT <<Bdaddr> <linktype> <channel ID>>

Examples:

Normal mode	Numerical (ATV0)	1 <00802501D11A 0x02>
	Verbose (ATV1)	CONNECT <00802501D11A 0x02>
MUX mode	Numerical (ATV0)	1 <00802501D11A 0x02 0x01>
	Verbose (ATV1)	CONNECT <00802501D11A 0x02 0x01>

3.3.2 RING

For the AT result code RING the following extended result codes are available:

Syntax normal mode: RING <<Bdaddr> <linktype>>
Syntax MUX mode: RING <<Bdaddr> <linktype> <channel ID>>

Examples:

Normal mode	Numerical (ATV0)	2 <00802501D11A 0x02>
	Verbose (ATV1)	RING <00802501D11A 0x02>
MUX mode	Numerical (ATV0)	2 <00802501D11A 0x02 0x01>
	Verbose (ATV1)	RING <00802501D11A 0x02 0x01>

3.3.3 NO CARRIER

For the AT result code NO CARRIER the following extended result codes are available:

Syntax normal mode: NO CARRIER <<Bluetooth release/error code>>
Syntax MUX mode; NO CARRIER <<Bluetooth release/error code> <channel ID>>

The following table shows the release/error codes:

Error code	Meaning
0x0000	Success
0x0001	Accept
0x0002	Reject
0x0003	Resource error
0x0004	Invalid parameter
0x0005	Invalid state
0x0006	Connection disconnect
0x0007	Connection paused
0x0008	Connection lost
0x0009	Authentication failed
0x000A	Flowcontrol violation
0x000B	Init timeout
0x000C	Init out of sync
0x000D	Init hardware failure
0x000E	Lower layer error
0x00FD	Unspecified
0x00FE	Not supported

Examples:

Normal mode	Numerical (ATV0)	3 <0006>
	Verbose (ATV1)	NO CARRIER <0006>
MUX mode	Numerical (ATV0)	3 <0006 0x01>
	Verbose (ATV1)	NO CARRIER <0006 0x01>

3.4 Events

Text	Description
SSPPIN <Bdaddr> ?	SSP passkey request (see AT+BSSPPIN command)

4 Related Documents

[1] BlueMod+S Hardware Reference

[2] BlueMod+S User Guide

[3] Bluetooth 4.0 Core Specification

[4] BlueMod+S - BlueMod+SR Delta Reference

5 History

Version	Release Date	By	Change description
r01	06.03.2014	or	Initial version
r02d01	11.06.2014	or	Add custom gatt server and advertising commands. Add bootloader version support. Decrease BNDLIST size to 4 Remove ATS30
r02d02	05.08.2014	ta	Add +BPAIRMODE command
r02	27.08.2014	or	Modified gatt server parameter Added note for MUX mode results Added note for AT&F More detailed description for returned channel format Correct characteristic default value description Remove note from +LEATTRIB table Revised description of +LEADPAR
r03	17.02.2015	ta	Added chapter 2.3.2 GATT Server Data Handling on MUX Channel Added variable length of charval Added firmware dependency for new implemented commands
r04	17.06.2015	ta	Added new commands for firmware V2.000: AT+LESRVBOOTSET AT+LESRVBOOTMODE AT+LESRVSETOPEN AT+LESRVSETSAVE AT+LESRVSETACT AT+IOBCFG AT+SYSTEMOFF AT+DFUMODE AT+DFUNAME AT+DFUSTART Extended commands for firmware V2.000: AT+BMUX AT+LEATTRIB Added a second note to AT+BNDS Added supported output power values
r05	28.04.2016	ta	Added new value “2” of +IOBCFG command Added responses of list commands Corrected extended result codes syntax Corrected description of +DFUNAME command
r06	24.05.2016	bg	Telit cover page added

Telit Wireless Solutions GmbH
Mendelssohnstraße 15 D
22761 Hamburg
Germany

Phone: +49 (0)40 890 88-0
Fax: +49 (0)40 890 88-444
E-mail: ts-srd@telit.com
www.telit.com

SUPPORT INQUIRIES

Link to **www.telit.com** and contact our technical support team for any questions related to technical issues.

www.telit.com

Telit Communications S.p.A.
Via Stazione di Prosecco, 5/B
I-34010 Sgonico (Trieste), Italy

Telit IoT Platforms LLC
5300 Broken Sound Blvd, Suite 150
Boca Raton, FL 33487, USA

Telit Wireless Solutions Inc.
3131 RDU Center Drive, Suite 135
Morrisville, NC 27560, USA

Telit Wireless Solutions Co., Ltd.
8th Fl., Shinyoung Securities Bld.
6, Gukjegeumyung-ro8-gil, Yeongdeungpo-gu
Seoul, 150-884, Korea

Telit Wireless Solutions Ltd.
10 Habarzel St.
Tel Aviv 69710, Israel

Telit Wireless Solutions
Technologia e Servicos Ltda
Avenida Paulista, 1776, Room 10.C
01310-921 São Paulo, Brazil

[04.2016]