



# BlueMod+S42/ADC/LUA AT Command Reference

80512ST10860A Rev. 1 – 2018-08-30

**TELIT**  
**TECHNICAL**  
**DOCUMENTATION**

SPECIFICATIONS ARE SUBJECT TO CHANGE WITHOUT NOTICE

## **NOTICE**

While reasonable efforts have been made to assure the accuracy of this document, Telit assumes no liability resulting from any inaccuracies or omissions in this document, or from use of the information obtained herein. The information in this document has been carefully checked and is believed to be reliable. However, no responsibility is assumed for inaccuracies or omissions. Telit reserves the right to make changes to any products described herein and reserves the right to revise this document and to make changes from time to time in content hereof with no obligation to notify any person of revisions or changes. Telit does not assume any liability arising out of the application or use of any product, software, or circuit described herein; neither does it convey license under its patent rights or the rights of others.

It is possible that this publication may contain references to, or information about Telit products (machines and programs), programming, or services that are not announced in your country. Such references or information must not be construed to mean that Telit intends to announce such Telit products, programming, or services in your country.

## **COPYRIGHTS**

This instruction manual and the Telit products described in this instruction manual may be, include or describe copyrighted Telit material, such as computer programs stored in semiconductor memories or other media. Laws in the Italy and other countries preserve for Telit and its licensors certain exclusive rights for copyrighted material, including the exclusive right to copy, reproduce in any form, distribute and make derivative works of the copyrighted material. Accordingly, any copyrighted material of Telit and its licensors contained herein or in the Telit products described in this instruction manual may not be copied, reproduced, distributed, merged or modified in any manner without the express written permission of Telit. Furthermore, the purchase of Telit products shall not be deemed to grant either directly or by implication, estoppel, or otherwise, any license under the copyrights, patents or patent applications of Telit, as arises by operation of law in the sale of a product.

## **COMPUTER SOFTWARE COPYRIGHTS**

The Telit and 3rd Party supplied Software (SW) products described in this instruction manual may include copyrighted Telit and other 3rd Party supplied computer programs stored in semiconductor memories or other media. Laws in the Italy and other countries preserve for Telit and other 3rd Party supplied SW certain exclusive rights for copyrighted computer programs, including the exclusive right to copy or reproduce in any form the copyrighted computer program. Accordingly, any copyrighted Telit or other 3rd Party supplied SW computer programs contained in the Telit products described in this instruction manual may not be copied (reverse engineered) or reproduced in any manner without the express written permission of Telit or the 3rd Party SW supplier. Furthermore, the purchase of Telit products shall not be deemed to grant either directly or by implication, estoppel, or otherwise, any license under the copyrights, patents or patent applications of Telit or other 3rd Party supplied SW, except for the normal non-exclusive, royalty free license to use that arises by operation of law in the sale of a product.

## USAGE AND DISCLOSURE RESTRICTIONS

### I. License Agreements

The software described in this document is the property of Telit and its licensors. It is furnished by express license agreement only and may be used only in accordance with the terms of such an agreement.

### II. Copyrighted Materials

Software and documentation are copyrighted materials. Making unauthorized copies is prohibited by law. No part of the software or documentation may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, without prior written permission of Telit.

### III. High Risk Materials

Components, units, or third-party products used in the product described herein are NOT fault-tolerant and are NOT designed, manufactured, or intended for use as on-line control equipment in the following hazardous environments requiring fail-safe controls: the operation of Nuclear Facilities, Aircraft Navigation or Aircraft Communication Systems, Air Traffic Control, Life Support, or Weapons Systems (High Risk Activities"). Telit and its supplier(s) specifically disclaim any expressed or implied warranty of fitness for such High Risk Activities.

### IV. Trademarks

TELIT and the Stylized T Logo are registered in Trademark Office. All other product or service names are the property of their respective owners.

### V. Third Party Rights

The software may include Third Party Right software. In this case you agree to comply with all terms and conditions imposed on you in respect of such separate software. In addition to Third Party Terms, the disclaimer of warranty and limitation of liability provisions in this License shall apply to the Third Party Right software.

TELIT HEREBY DISCLAIMS ANY AND ALL WARRANTIES EXPRESS OR IMPLIED FROM ANY THIRD PARTIES REGARDING ANY SEPARATE FILES, ANY THIRD PARTY MATERIALS INCLUDED IN THE SOFTWARE, ANY THIRD PARTY MATERIALS FROM WHICH THE SOFTWARE IS DERIVED (COLLECTIVELY "OTHER CODE"), AND THE USE OF ANY OR ALL THE OTHER CODE IN CONNECTION WITH THE SOFTWARE, INCLUDING (WITHOUT LIMITATION) ANY WARRANTIES OF SATISFACTORY QUALITY OR FITNESS FOR A PARTICULAR PURPOSE.

NO THIRD PARTY LICENSORS OF OTHER CODE SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND WHETHER MADE UNDER CONTRACT, TORT OR OTHER LEGAL THEORY, ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE OTHER CODE OR THE EXERCISE OF ANY RIGHTS GRANTED UNDER EITHER OR BOTH THIS LICENSE AND THE LEGAL TERMS APPLICABLE TO ANY SEPARATE FILES, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

## APPLICABILITY TABLE

### PRODUCTS

- BLUEMOD+S42/AI/ADC/LUA

## CONTENTS

<b>NOTICE</b>	.....	<b>2</b>
<b>COPYRIGHTS</b>	.....	<b>2</b>
<b>COMPUTER SOFTWARE COPYRIGHTS</b>	.....	<b>2</b>
<b>USAGE AND DISCLOSURE RESTRICTIONS</b>	.....	<b>3</b>
<b>APPLICABILITY TABLE</b>	.....	<b>4</b>
<b>CONTENTS</b>	.....	<b>5</b>
<b>1. INTRODUCTION</b>	.....	<b>6</b>
1.1. Scope	.....	6
1.2. Audience	.....	6
1.3. Contact and Support Information	.....	6
1.4. Text Conventions	.....	7
1.5. Related Documents	.....	8
<b>2. FEATURES</b>	.....	<b>9</b>
<b>3. COMMAND SUMMARY</b>	.....	<b>10</b>
3.1. General	.....	10
3.2. Bluetooth Low Energy	.....	21
3.3. GATT	.....	26
3.3.1. GATT Client Functionality	.....	26
3.3.2. GATT Server Definition	.....	29
3.3.3. GATT Server Data Handling on AT	.....	35
3.3.4. Data Handling with or without Connection	.....	35
<b>4. APPENDIX</b>	.....	<b>36</b>
4.1. Bluetooth Address	.....	36
4.2. Linktype	.....	36
4.3. AT Result Codes	.....	36
4.4. Events	.....	36
<b>5. GLOSSARY AND ACRONYMS</b>	.....	<b>39</b>
<b>6. DOCUMENT HISTORY</b>	.....	<b>40</b>

## 1. INTRODUCTION

### 1.1. Scope

This document specifies the command interface for the BlueMod+S42/ADC/LUA firmware.

### 1.2. Audience

This document is intended for Telit customers, especially system integrators, about to implement Bluetooth modules in their application.

### 1.3. Contact and Support Information

For general contact, technical support services, technical questions and report documentation errors contact Telit Technical Support at:

- [TS-SRD@telit.com](mailto:TS-SRD@telit.com)

Alternatively, use:

<https://www.telit.com/contact-us/>

For detailed information about where you can buy the Telit modules or for recommendations on accessories and components visit:

<https://www.telit.com>

Our aim is to make this guide as helpful as possible. Keep us informed of your comments and suggestions for improvements.

Telit appreciates feedback from the users of our information.

## 1.4. Text Conventions

---



Danger – This information **MUST** be followed or catastrophic equipment failure or bodily injury may occur.

---

---



Caution or Warning – Alerts the user to important points about integrating the module, if these points are not followed, the module and end user equipment may fail or malfunction.

---

---



Tip or Information – Provides advice and suggestions that may be useful when integrating the module.

---

All dates are in ISO 8601 format, i.e. YYYY-MM-DD.

## 1.5. Related Documents

- [1] BlueMod+S42 Hardware User Guide, 1VV0301303
- [2] BlueMod+S42 Lua Software User Guide, 1VV0301471
- [3] BlueMod+S42 Lua API Documentation, 30512ST10861A
- [4] Bluetooth 4.0 Core Specification



## 2. FEATURES

The BlueMod+S42/ADC/LUA supports AT command mode.

In the factory-default configuration the BlueMod+S42/ADC/LUA supports the AT command Mode via an internal Software interface.

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.

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. It is automatically discarded if the input is longer. 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
AT<command>?<CR>	Read the value of the command



Exceptions of this syntax are marked separately.

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

Responses	Description
<CR><LF>value<CR><LF>	Current value
<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

### 3. COMMAND SUMMARY

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.

#### 3.1. General

<b>&amp;F</b>	<b>Load Factory Defaults</b>
---------------	------------------------------

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

The factory-default values will be loaded.

Value	Description
0	Set all parameters except <b>bndlist</b> to factory defaults
1	Set all parameters to factory defaults

When **AT+LEPRIVACY=2** is active (Link Layer Privacy enabled) the **AT&F1** command generates a new Identity Resolving Key (IRK). This key is used to calculate the local random resolvable address and exchanged during bonding. By changing the IRK all previously bonded peer devices are no longer able to resolve the local random resolvable address.



In case of value=1 the non volatile bonding information will be deleted.

---

<b>+BIOCAP</b>	<b>SSP I/O Capabilities</b>
----------------	-----------------------------

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

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

Value	Description	Related commands	Related events
0	Display only	n.a.	SSPPIN
1	Display Yes/No	+BSSPCONF (secure connection)	SSPPIN (LE legacy pairing) SSPCONF (secure connection)
2	Keyboard only	+BSSPPIN	SSPPIN
<b>3</b>	No input no output	n.a.	n.a.
4	Display and keyboard	+BSSPPIN (LE legacy pairing) +BSSPCONF (secure connection)	SSPPIN (LE legacy pairing) SSPCONF (secure connection)

<b>+BNAME</b>	<b>Local Device Name</b>
---------------	--------------------------

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 8 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=LUA %2a	Display on remote end: LUA AB
<b>AT+BNAME=LUA %4a</b>	Display on remote end: LUA 89AB
AT+BNAME=LUA %:3a	Display on remote end: LUA 9:AB
AT+BNAME=LUA %3a	Display on remote end: LUA 9AB
AT+BNAME=LUA %:12a	Display on remote end: LUA 01:23:45:67:89:AB

<b>+BNDDEL</b>	<b>Delete Bonding Information</b>
----------------	-----------------------------------

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

This command deletes the bonding information stored by the BlueMod+S42/ADC/LUA.

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



This command can only be executed while not connected.

---

<b>+BNDLIST</b>	<b>Show Bonded Device List</b>
-----------------	--------------------------------

AT syntax:               **AT+BNDLIST**

This command shows information about the devices bonded with the BlueMod+S42/ADC/LUA. Each entry in the bonded-device list contains the Bluetooth address and the linktype (see chapter 4.2).

Example:

AT+BNDLIST	0080254800DA 0x02 9C04EB06ACA2 0x03 OK
------------	--

<b>+BNDS</b>	<b>Storage Mode for Bonds</b>
--------------	-------------------------------

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 module



By setting AT+BNDS the bonded-device list is deleted internally. 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.

<b>+BNDSIZE</b>	<b>Bonded Devices List Size</b>
-----------------	---------------------------------

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

This command reduces the number of devices (1..4) the bonded-device list can hold. The BlueMod+S42/ADC/LUA can store up to 4 devices. The default size is **4**.



Modification of this parameter will delete all devices in the bonded-device list.

Example:

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

<b>+BOAD</b>	<b>Bluetooth Own Device Address</b>
--------------	-------------------------------------

AT syntax:                   **AT+BOAD**

This command reads the Bluetooth devices' own device address.



This command is read only.

<b>+BPAIRMODE</b>	<b>Configure Pairable Mode</b>
-------------------	--------------------------------

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

This command controls the pairable mode of the BlueMod+S42/ADC/LUA.

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+S42/ADC/LUA advertises TIO as “functional”
1	Pairing allowed, BlueMod+S42/ADC/LUA advertises TIO as “bondable and functional”

<b>+BSSPCONF</b>	<b>Security Pairing Confirmation</b>
------------------	--------------------------------------

AT syntax:               **AT+BSSPCONF <Bluetooth address>,<address type>,<value>**

If a pairing is initiated and LE secure connection is supported, depending on the security settings AT interface generates an event SSPCONF and asks the user for confirmation.

Event: SSPCONF <Bluetooth address>,<address type> <passkey> ?

The user has to confirm the passkey with the above command. If no confirmation is sent by the user within the bonding timeout or in case of active reject, the pairing is rejected with NO CARRIER message.

Value	Description
0	Reject passkey confirmation request
1	Accept passkey confirmation request

Example:

SSPCONF 00802507C08D,t2 794851 ? AT+BSSPCONF 00802507C08D,t2,1  OK	Receive SSP pairing request Send SSP pairing confirmation
---	--

<b>+BSSPPIN</b>	<b>SSP Passkey Response</b>
-----------------	-----------------------------

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

If an authentication is initiated (**AT+LETIO=3 or 4**), 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>,<address type> ?

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

Example:

SSPPIN 00802507C08D,t2 ? AT+BSSPPIN 00802507C08D,t2,314546  OK	Receive SSP passkey request Send SSP passkey response
---	--

## +BSSPDBG SSP Debug Mode

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

This command allows to enable the SSP debug mode of the device. This mode is required to trace the SSP Bluetooth connection by using a sniffer.

Value	Description
0	SSP Debug mode off
1	SSP Debug mode on



SSP debug mode is for tracing purposes only and shall be deactivated for normal operation. Otherwise the connection may be unsecure.

## 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: **ATH**

This command disconnects the existing Bluetooth connection.

	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. "LUA %4a")
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 Telit"
7	Returns "OK"
8	Returns "ERROR"
9	Returns "OK"
77	Returns bootloader version
99	Returns the firmware creation date

<b>+DFUMODE</b>	Device Firmware Update Mode
-----------------	-----------------------------

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 <b>(default)</b>
2	Device firmware update over the air (OTA)

<b>+DFUNAME</b>	Over the Air Update Name
-----------------	--------------------------

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 activate the device name it is necessary to store the settings and perform a reset or use the command **AT+DFUSTART**.

Name	Description
<b>BM+S_DFU</b>	Device firmware update name is "BM+S_DFU" ( <b>default</b> )

<b>+DFUSTART</b>	<b>Start Bootloader</b>
------------------	-------------------------

AT syntax:                   **AT+DFUSTART**

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

<b>+PNPPID</b>	<b>PnP Product ID</b>
----------------	-----------------------

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

This command sets the product ID provided in the device information service (DIS). The format is a 16 bit hex value. The default value is 0xB017 (Telit product ID for BlueMod+S42/ADC/LUA firmware).

<b>+PNPPVER</b>	<b>PnP Product Version</b>
-----------------	----------------------------

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

This command sets the product version provided in the device information service (DIS). The format is a 16 bit hex value. The default value is the version number of the particular Telit BlueMod+S42/ADC/LUA firmware, e.g. 0x1000 for firmware version 1.000.



After setting the vendor ID (AT+PNPVID) to a different value than the default 0x008F the user has to set his own product version (otherwise the value 0x0100 will be used).

---

<b>+PNPVID</b>	<b>PnP Vendor ID</b>
----------------	----------------------

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

This command sets the vendor ID provided in the device information service (DIS). The format is a 16 bit hex value. The default value is 0x008F (Telit vendor ID).

<b>+PNPVSRC</b>	<b>PnP Vendor ID Source</b>
-----------------	-----------------------------

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

This command sets the vendor ID source provided in the device information service (DIS).

Value	Description
1	Bluetooth SIG assigned company ID
2	USB assigned company ID

<b>Q</b>	<b>Suppress Results</b>
----------	-------------------------

AT syntax:               **ATQ<value>**

This command allows/suppresses result codes and messages.

Value	Description
<b>0 or empty</b>	Enable result messages after command input
1	Suppress result messages after command input

<b>+RESET</b>	<b>Reset Device</b>
---------------	---------------------

AT syntax:               **AT+RESET**

This command resets the whole functionality of the BlueMod+S42/ADC/LUA by a forced hardware reset (like power off/on).



No OK response will be sent before the device performs a reset.

---

<b>+RFMAXTXPWR</b>	<b>Maximum Output Power</b>
--------------------	-----------------------------

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
3	3 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
-----------------	---

<b>+SYSTEMOFF</b>	<b>Enter System Off Mode</b>
-------------------	------------------------------

AT syntax:               **AT+SYSTEMOFF**

This command sets the module into low power mode during the time the module is not used.

When setting AT+SYSTEMOFF (or AT+SYSTEMOFF=1) the module will wake-up on GPIO activity. To achieve the lowest power consumption set AT+SYSTEMOFF=2. In this case the module will wake-up by RESET signal.

Value	Description
1	Wake-up by GPIO
2	Wake-up by RESET signal

Possible use cases and a usage example are described in the *BlueMod+S42 Lua Software User Guide [2]*.

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

### 3.2. Bluetooth Low Energy

#### +LEFIXPIN Fix PIN for Pairing Procedure

AT syntax: **AT+LEFIXPIN=fixpin**

This command specifies a 6 digit SSP passkey, to be used for the security procedure. If this value has a length of 0 (no digit specified in command) **AT+LEFIXPIN=** a randomly generated PIN is used instead. To use this feature see also commands **AT+LETIO**, **AT+BIOCAP** and Hayes event SSPPIN.

For further information, see *BlueMod+S42 Lua Software User Guide [2]*.

Parameter	Description
fixpin	6 digits pin value (digit 0...9 only), default is zero length pin ""

#### +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
$n=20\dots10240$	Use maximum advertising interval of n ms
<b>1280</b>	Use maximum advertising interval of 1280 ms



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+S42/ADC/LUA. It is just provided for compatibility reasons. The used advertising interval is set by **AT+LEADINTMAX** parameter.



The command “AT+LECONPARAM” used with “?” shows the active connection parameters used by the connection defined by connHnd. A LECONPARAM event is generated containing the active connection parameters.

Parameter	Description
connHnd	Set to zero
connIntMin	6...3200 minimum connection interval in steps of 1.25 ms (mandatory for peripheral)
connIntMax	6...3200 maximum connection interval in steps of 1.25 ms
slaveLat	0...499 connection intervals
connTimeout	Optional connection supervision timeout in steps of 10 ms. Will be calculated internally if not specified. Has to be calculated according to Bluetooth core spec.

The new connection parameters are signaled by event LECONPARAM.

<b>+LEPRIVACY</b>	<b>Enable Link Layer Privacy</b>
-------------------	----------------------------------

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

This command controls the signaling of LERESOLVED events and the use of Link Layer Privacy for new BLE connections.

Value	Description
0	No LERESOLVED events are signaled during pairing. The local device uses its public address for all Bluetooth roles.
1	When pairing with a peer device that is using a resolvable random address a LERESOLVED event is signaled.
2	Same as 1. The local device uses a random resolvable address for all Bluetooth roles. This address is changed every 15 minutes.

<b>+LESLAVELAT</b>	<b>Slave Latency</b>
--------------------	----------------------

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

<b>+LETIO</b>	<b>Enable Terminal I/O Service</b>
---------------	------------------------------------

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

This command controls the Terminal I/O service. If set to 0 the Terminal I/O service is disabled.

Value	Description
0	Terminal I/O service disabled (no advertising, no characteristics)
1	Terminal I/O service enabled, security is required with encryption (no MITM)
2	Terminal I/O service enabled, no security (authentication or encryption) required
3	Terminal I/O service enabled, authenticated pairing with encryption (MITM required)
4	Terminal I/O service enabled, authenticated LE Secure connections pairing with encryption (MITM required, LE secure connections required)

<b>+LEADDATA</b>	<b>Setup Advertise Data for Customized Advertising</b>
------------------	--

AT syntax:               **AT+LEADDATA=<value<sub>1</sub>> .. <value<sub>n</sub>>**

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

*Value<sub>k</sub>* represents an octet in hexadecimal format,  $k \leq 31$ .

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)* [4].

Example 1: Advertise support for battery service

AT+LEADDATA=02010603020F18	Set flags + UUID of battery service
----------------------------	-------------------------------------



## Example 2: Advertise custom specific data

<pre>AT+LEADDATA= 0201060AFF8F0014B10101000100</pre>	<pre>020106 (AD Flags) - 0x02 – data length 2 bytes - 0x01 – AD Flags - 0x06 - LE General Discoverable Mode + BR/EDR   Not Supported  0AFF8F00 (data container) - 0x0A – data length 10 bytes - 0xFF – Manufacturer Specific Data - 0x008F – Telit Company ID   (see Bluetooth assigned numbers)  14B10101000100 - Custom specific binary data</pre>
--	--

<b>+LEADE</b>	<b>Enable Customized Advertising</b>
---------------	--------------------------------------

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 TIO 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

<b>+LEADPAR</b>	<b>Setup Parameters for Customized Advertising</b>
-----------------	--

AT syntax: **AT+LEADPAR=par<sub>1</sub>=<value<sub>1</sub>>[, .. [,par<sub>n</sub>=<value<sub>n</sub>>]]**

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

par <sub>n</sub>	value <sub>n</sub>
advtype	Type of advertising: <b>0: undirected (default)</b>
Optional	2: scannable
Coding: decimal.	3: non-connectable

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

- advtype = 0 - advertising type “undirected”



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

AT syntax: **AT+LESCDATA=<value<sub>1</sub>> .. <value<sub>n</sub>>**

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

Value<sub>k</sub> 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)* [4].

Example:

AT+LESCDATA=03020F18	0x03 - data length 3 bytes 0x02 - incomplete list of 16-bit service class UUIDs 0x180F - UUID for battery service (see <a href="https://www.bluetooth.com/specifications/gatt/services">https://www.bluetooth.com/specifications/gatt/services</a> )
----------------------	---

### 3.3. GATT

#### 3.3.1. GATT Client Functionality

The BlueMod+S42\ADC\LUA does not support full GATT Client functionality but only a subset that allows to scan for advertisement and scan response data of remote devices.

It does not support GATT service discovery or connection setup.

<b>+LEBUUIDSET</b>	<b>Set 128 bit Base UUID</b>
--------------------	------------------------------

AT syntax:               **AT+LEBUUIDSET=ux**

This command configures base UUIDs needed by the stack to identify 128 bit UUIDs correct. 128 bit UUIDs are module internally treated as 16 bit UUIDs with a defined base UUID. The Telit defined base UUID for Terminal IO V2 is 0000xxxx000010008000008025000000 with xxx as the variable 16 bit UUID part.

To set a base UUID the 16 bit UUID part could have any legal 16 bit value. It is internally ignored for this command. If the internal base UUID table is full the command reports ERROR, otherwise it reports OK.

The UUID list shall be saved permanent with **AT&W**.

Parameter	Description
ux	With x= 128 bit base UUID

<b>+LEBUUIDLIST</b>	<b>Show Base UUID List</b>
---------------------	----------------------------

AT syntax:               **AT+LEBUUIDLIST**

This command shows information about the configured 128 bit base UUIDs within the **Error! Unknown document property name..**

The variable 16 bit UUID part is marked with the character 'x'.

Example:

AT+LEBUUIDLIST	0000xxxx000010008000008025000000 0000xxxx111100002222008033330000 OK
----------------	--

<b>+LESCAN</b>	<b>Search Bluetooth Low Energy Devices</b>
----------------	--

AT syntax:               **AT+LESCAN**  
**AT+LESCAN=GATT[,<rssix>][,RAW]**  
**AT+LESCAN=<ux>[,<rssix>][,RAW]**  
**AT+LESCAN=<brad>[,<tx>][,<rssix>][,RAW]**

With this command an automatic search for all discoverable Bluetooth Low Energy devices will be initiated. The discovery will last for a time defined by command **AT+LESCANDURATION**.

Parameter	Description
rssix	Filter for devices with RSSI value same or stronger x
brad	Filter for Bluetooth remote device address (12 hex digits)
tx	x is the remote Bluetooth address type see chapter LinkType. If not specified a public address is assumed
ux	With x=UUID of a service (4 or 32 hex digits)
GATT	Show all found devices
RAW	Hexdump of advertising and scan response data without duplicate filtering

Any character input while the **Error! Unknown document property name.** is searching will abort the search procedure.

The resulting list depends on the used command parameters.

As a result, a list will be output containing the Bluetooth addresses of the visible devices in range, the advertisement type, the RSSI, the Bluetooth friendly name, the TX level, manufacturer specific data and all UUIDs contained in the advertising and scan response data, if available, of the remote device. Please note that more AD types could be decoded in future releases.

Bluetooth address, RSSI and TYPE are always provided. All other values like Bluetooth friendly name, TX level, manufacturer specific data and UUID are optional and depends of the advertising data of the discovered device. UUIDs can be 16-bit or 128-bit values.

The output is filtered to show each unique advertising packet only once. If the internal filter table is full, all new advertising packets are shown unfiltered. The output in RAW mode is also unfiltered.

The optional parameter <rssix> (8-bit signed value) can be used to perform a search only for devices with a rssi value higher than the provided value. E.g. **AT+LESCAN=GATT,rxssi-50** will show all devices with a rssi value same or higher than -50dBm. This means -45dBm devices are shown, -55dBm devices are filtered.

The optional parameter <brad> (12 hex digits) can be used to perform a search for a device with the specified Bluetooth address. The optional parameter <tx> specifies the type of Bluetooth low energy address.

The optional parameter <ux> (16-bit or 128-bit uuid value) can be used to perform a search for devices which advertises a specific service.

To show devices supporting Terminal I/O only, the UUID FEFB shall be used (**AT+LESCAN=uFEFB**).

If the "RAW" parameter is given, the output will not contain decoded AD type data. Instead it will contain the Bluetooth address of the visible device in range, the RSSI, the advertisement type and the complete advertise or scan response data from the remote device. The data is displayed as an ascii coded byte steam in hexadecimal values.

There will be no duplicate filtering for advertising packets. All received packets (advertise or scan response) will be printed as soon they are received. If UUID filtering is on

(**AT+LESCAN=ux,RAW**), the output will be printed after receiving the complete advertising data. In active scan mode these are the advertising and scan response packets.

Examples:

AT+LESCAN	008025497826,t2 RSSI:-62 TYPE:CONN NAME:BM+SR 7 TX:4 MNF:8F0009B0011000 UUID:53544D544552494F5345525631303030 UUID:FEFB  OK
AT+LESCAN=GATT,RAW	0080254800DD,t2 RSSI:-77 TYPE:CONN DATA:02010608FF8F0009B0011000 0080254800DD RSSI:-79 TYPE:SCANRSP DATA:110730303031565245534F495245544D54530302FBFE080 86A75657267656E  OK

<b>+LESCANDURATION</b>	<b>Duration for +LESCAN</b>
------------------------	-----------------------------

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

This command configures how long the **Error! Unknown document property name.** is searching for discoverable Bluetooth Low Energy devices when the command **AT+LESCAN** is used.

Value	Description
0	Sets duration time to infinite
1..300	Sets duration time between 1 seconds and 300 seconds ( <b>default=10</b> )

### 3.3.2. GATT Server Definition

<b>+LEATTRIB</b>	<b>Define Attributes for a Service</b>
------------------	--

AT syntax:                   **AT+LEATTRIB=<type>[,par<sub>1</sub>=<value<sub>1</sub>>[, .. ,par<sub>n</sub>=<value<sub>n</sub>>]..]**

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  $par_1, \dots, par_n$  depends on the value of  $\langle type \rangle$ :

type	par <sub>1</sub>	par <sub>2</sub>	par <sub>3</sub>	par <sub>4</sub> <sup>(1)</sup>
pserv	uuid=<16/128bit UUID>			
Mandatory	Mandatory  Coding: hexadecimal.			
char	prop=<properties>			
Mandatory	Mandatory  Coding: hexadecimal.  <i>properties</i> may have the bitmask values <i>Read</i> , <i>Write Without Response</i> , <i>Write</i> , <i>Notify</i> and <i>Indicate</i> 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 <i>Notify</i> or <i>Indicate</i> are set.			
charval	uuid=<16/128bit UUID>	perm=<permissions>	len=<length>	val=<value>
Mandatory	Mandatory  Coding: hexadecimal.	Mandatory  Coding: hexadecimal.  16-bit value that decodes the access permissions and authentication requirements.	Mandatory  Coding: decimal.  Maximum supported length is 20 bytes. A length of 0 means a variable length of the characteristic (up to 20 bytes)	Optional  Coding: hexadecimal.  Pre-defined characteristic value that will be loaded when the service set is activated.

type	par1	par2	par3	par4 <sup>(1)</sup>
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 <i>type=char</i> description.  CCCDs are required to be readable without authentication and required to be writable.			
chardusrd  Optional	usrd=<user description>  Mandatory  Coding: UTF-8 string.  User description string.  Note: - Since <i>Extended Properties</i> (see <i>type=char</i> , parameter <i>properties</i> 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.  User Descriptions are required to be readable and required to be not writeable.  If this parameter is omitted the default <i>readable without authentication or authorization</i> applies.		
complete  Mandatory	No parameter.  Used to signal that all attribute definitions have been sent to the controller.			

<sup>(1)</sup> *par4* 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 [4]*.

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
0,1,2	0	Read not permitted
	1	Read permitted
	2	Read with authentication permitted
	3	Read with authentication and MITM protection permitted
	4	Read with authentication, LE Secure Connection and MITM protection permitted
3	-	Reserved
4,5,6	0	Write not permitted
	1	Write permitted
	2	Write with authentication permitted
	3	Write with authentication and MITM protection permitted
	4	Write with authentication, LE Secure Connection and MITM protection permitted
7..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 with the AT+LESRVDATA command / LESRVDATA event.

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
AT+LEATTRIB=pserv,uuid=180F	OK	Declares the properties of the battery level value
AT+LEATTRIB=char,prop=12	OK	
AT+LEATTRIB=charval,uuid=2A19,perm=0001,len=1	0x20 OK	Declares the battery level value (one byte in the range 0,...,100). Battery level values are exchanged over channel 0x20
AT+LEATTRIB=complete	OK	Completes the service definition sequence

## +LESRVCCDS

## Behavior of CCCD Value Store

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

This command allows to define the behavior of the firmware regarding stored Client Characteristic Configuration Descriptor (CCCD) values.

Value	Properties
0	CCCD values set from bonded peer devices are not stored in non-volatile memory. After reconnect a peer device need to set the CCCD bits again to enable Notifications and Indications.
1	CCCD values set from bonded peer devices are stored in non-volatile memory. After reconnect the GATT Server restores the security level that was required for setting the CCCD before sending Notifications and Indications.

### 3.3.3. GATT Server Data Handling on AT

<b>+LESRVDATA</b>	<b>GATT Server Data Exchange</b>
-------------------	----------------------------------

AT syntax:                   **AT+LESRVDATA=<channel>,<hexdata>**

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 over the corresponding channel. 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.

Parameter	Description
channel	Channel ID from AT+LEATTRIB command
hexdata	ASCII coded byte stream as hexadecimal values e.g. 017aFF for a three byte value

#### 3.3.3.1. Error Handling

Since there is no end-to-end flow control, the GATT server cannot guarantee successful data transfer for notifications and writes without response and will silently discard the data.

#### 3.3.4. 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.

## 4. APPENDIX

### 4.1. Bluetooth Address

The BlueMod+S42/ADC/LUA supports public and random Bluetooth addresses. The differentiation between the address types is done using the parameter “t2” for public addresses and “t3” for random addresses.

A Bluetooth address value itself is a special byte array variant. There are two valid representations.

The Bluetooth addresses “008025540203” and **00:80:25:54:02:03** are equivalent.

E.g.            public address:            00:80:25:54:02:03,t2 or 008025540203,t2  
                   random address:            F1:B9:EB:41:D8:1E,t3 or F1B9EB41D81E,t3

### 4.2. Linktype

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

### 4.3. AT Result Codes

Result codes (numerical and verbose):

Numeric	Text	Meaning
0	OK	Command completed
4	ERROR	Illegal command or error that cannot be indicated otherwise

### 4.4. Events

**SSPCONF** **SSP Passkey Confirmation**

Syntax:                    **SSPCONF Bdaddr,tx Passkey ?**

With this event the module requests the confirmation of the passkey displayed on both devices.

Parameter	Description
Bdaddr	Remote Bluetooth address
tx	x is the remote Bluetooth address type (see chapter Bluetooth Address)
Passkey	Passkey to be acknowledged on local side (see command <b>AT+LETIO=3</b> )

<b>SSPPIN</b>	<b>SSP Passkey Request</b>
---------------	----------------------------

Syntax: **SSPPIN Bdaddr,tx ?**

With this event the module requests the entry of the passkey displayed on the remote device.

Parameter	Description
Bdaddr	Remote Bluetooth address
tx	x is the remote Bluetooth address type (see chapter Bluetooth Address)

<b>SSPPIN</b>	<b>SSP Passkey Display</b>
---------------	----------------------------

Syntax: **SSPPIN Bdaddr,tx Passkey**

With this event the module shows the passkey to be entered on the remote device.

Parameter	Description
Bdaddr	Remote Bluetooth address
tx	x is the remote Bluetooth address type (see chapter Bluetooth Address)
Passkey	Passkey to be entered on remote side

<b>LECONPARAM</b>	<b>Connection Parameters Updated</b>
-------------------	--------------------------------------

Syntax: **LECONPARAM:connHnd,connInt,slaveLat,connTimeout**

With this event the user is informed about the active connection parameters of the peripheral link if any. This is the answer to a "AT+LECONPARAM=0 command.

Parameter	Description
connHnd	Is always zero.
connInt	Actual connection interval in steps of 1.25 ms
slaveLat	Actual slave latency in connection intervals
connTimeout	Actual connection supervision timeout in steps of 10 ms

<b>LESRVDATA</b>	<b>GATT Server Data Exchange</b>
------------------	----------------------------------

Syntax:                   **LESRVDATA:<channel>,<data>**

With this event the user is informed about new data on a GATT server characteristic.

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

Parameter	Description
channel	Channel ID from AT+LEATTRIB command
hexdata	ASCII coded byte stream as hexadecimal values e.g. 017aFF for a three byte value

<b>LERESOLVED</b>	<b>Bluetooth Address Resolved</b>
-------------------	-----------------------------------

Syntax:                   **LERESOLVED:<privacy-bd>,<privacy-bd-type>,<public-bd>,<public-bd-type>**

With this event the user is informed during pairing about a relation between the currently used random resolvable address of a peer device and its public address stored in the bond database.

Parameter	Description
<b>privacy-bd</b>	Current privacy address used by the peer device
<b>privacy-bd-type</b>	Privacy address type (currently only 0x03)
<b>public-bd</b>	Public address of the peer device
<b>public-bd-type</b>	Public address type of the peer device

## 5. GLOSSARY AND ACRONYMS

ADC	Analog Digital Converter
AT	Attention Command
GAP	Generic Access Profile
GATT	Generic Attribute Profile
IRK	Identity Resolving Key
SSP	Secure Simple Pairing
TWI	Two Wire Interface
UART	Universal Asynchronous Receiver/Transmitter
UICP	UART Interface Control Protocol
UUID	Universal Unique Identifier

## 6. DOCUMENT HISTORY

---

Revision	Date	Changes
0	2017-10-06	First issue
1	2018-08-30	Added new commands +DFUMODE, +DFUNAME, +DFUSTART





# SUPPORT INQUIRIES

Link to [www.telit.com](http://www.telit.com) and contact our technical support team for any questions related to technical issues.

[www.telit.com](http://www.telit.com)



---

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

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

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

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

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

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

---

Telit reserves all rights to this document and the information contained herein. Products, names, logos and designs described herein may in whole or in part be subject to intellectual property rights. The information contained herein is provided "as is". No warranty of any kind, either express or implied, is made in relation to the accuracy, reliability, fitness for a particular purpose or content of this document. This document may be revised by Telit at any time. For most recent documents, please visit [www.telit.com](http://www.telit.com)

Copyright © 2016, Telit