BlueMod+S42M
Software User Guide

1VV0301391 Rev. 2 – 2019-02-20

TELIT
TECHNICAL
DOCUMENTATION

[04.2016]

Mod. 0815 2016-08 Rev.1

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+S42M

CONTENTS

1. INTRODUCTION

1.1. Scope

This document describes the usage of the Bluetooth module BlueMod+S42M.

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

Alternatively, use:

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

For detailed information about where you can buy 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+S42M AT Command Reference, 80527AT10839A
- [2] UICP+ UART Interface Control Protocol, 30507ST10756A
- [3] BlueMod+S42M Hardware User Guide, 1VV0301379
- [4] Bluetooth 4.0 Core Specification
- [5] Using the BlueMod+S as a Beacon, 80507NT11471A

2. GENERALS

This document describes the usage of the BlueMod+S42M Bluetooth module featuring firmware version V0.600 or later.

For a detailed description of the commands refer to the *BlueMod+S42M AT Command Reference [1]*.

All referenced documents can be downloaded from:

https://www.telit.com/products/wifi-and-bluetooth-modules/bluemods42m

3. SOFTWARE FEATURES

The BlueMod+S42M peripheral firmware includes the following feature set:

- Peripheral role
- GATT based Terminal I/O service
- User configurable GATT server (up to 20 characteristics)
- One active GATT server connection
- AT command mode
- Easy control over all connection parameters
- Advanced power saving features like UICP and SYSTEMOFF
- Firmware over the air update
- LE secure connections

4. INITIAL CONFIGURATION

In the default configuration (AT+LEADE=0) the BlueMod+S42M advertises the Terminal I/O service immediately after power up and therefore it is able to accept an incoming call request.

If the user wants to configure the BlueMod+S42M with specific settings without interruption by an incoming call, we recommend to disable the automatic advertising by setting AT+LEADE=3. After the configuration is done the advertising can be enabled again (AT+LEADE=0 for advertising Terminal I/O service or AT+LEADE=1 for customized advertising).

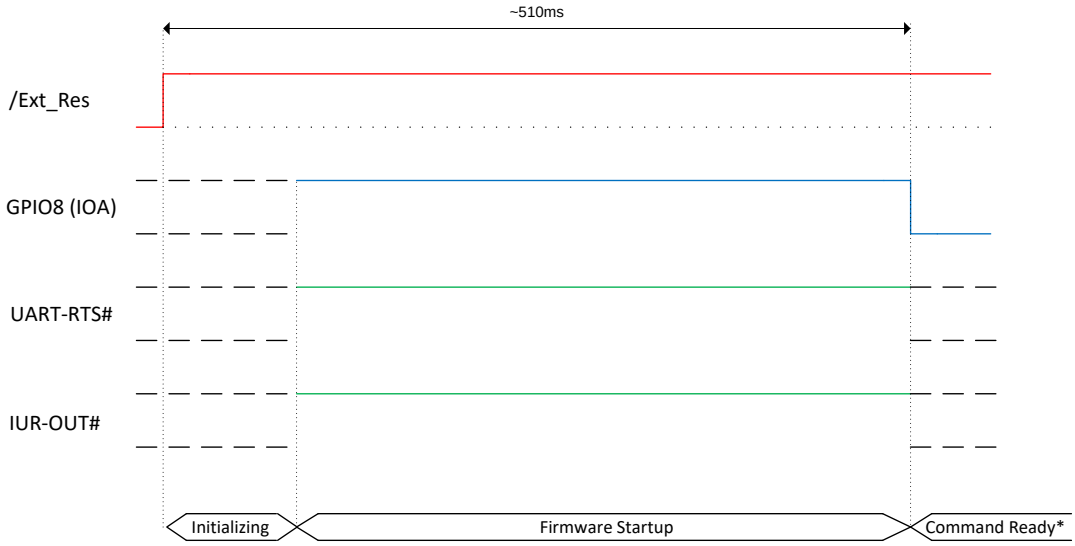
5. STARTUP TIMING

The startup time until the BlueMod+S42M is able to accept link requests or serial data depends on:

- The firmware version
- The usage of the UART Interface Control Protocol (UICP)

For more details about the UICP protocol please refer to the document *UICP+ UART Interface Control Protocol [2]*.

The following diagrams show the startup timing of the BlueMod+S42M based on firmware version V0.600.



(*) The firmware is command ready ~510 ms after the reset has been released and when GPIO8 (IOA) is low.

After GPIO8 gets low the state of the /RTS and /IUR-OUT lines depends on the UICP parameter. When UICP is disabled (AT+UICP=0) both output lines get low, otherwise the UICP function will be started.

6. SECURITY

This chapter describes the security mechanisms of the BlueMod+S42M to control the access to the local Bluetooth devices characteristics. The pairing process is triggered automatically when an access to a characteristic is requested that requires security.

6.1. Pairable and Bondable Mode

In general, we distinguish between pairing and bond. Pairing is the active process to generate a set of encryption keys. The paring can be done with or without user interaction depending of the I/O capabilities. The pairing will result in a bond if the generated data is stored in the bonded device list (AT+BNDLIST).

AT+BPAIRMODE controls if a pairing is performed or not.

Value	Description
0	No pairing (pairing request will be refused)
1	Pairing

The bonded device list is affected by the following commands:

- AT+BNDLIST shows the devices stored in the bonded device list
- AT+BNDSIZE determines the size of the bonded device list and deletes the whole list when modifying the size
- AT+BNDDEL deletes single entries or the whole list
- AT&F1 deletes the bonded device list

If the bonded device list is full and another device is bonded, the least recently used device will be overwritten by the new one. If bonds are not required please set AT+BNDS=0.

6.2. LE Secure Connections

Bluetooth 4.2 supports a new security mechanism called “Secure Connections”.

LE Secure Connection introduces a new method to generate a shared secret (key) in a way that ensures the data integrity and privacy of a connection even in cases where the pairing/ponding procedure was completely tapped with a Bluetooth sniffer if that shared secret is used for authentication and encryption.

Secure connection key generation is applicable for all authentication methods (e.g. just works or passkey entry) while all authentication trigged I/O activity remain the same as for legacy LE security but one new method (display yes/no) is introduced.

Bluetooth 4.2 mandates that LE Secure Connection key generation is used while pairing/bonding if both devices of a given connection support this feature. If one device of a given connection only supports LE legacy security key generation procedures these legacy procedures will be used instead.

From user point of view this negotiation is mostly transparent and backward compatible. The only exceptions are if LE Secure Connection is mandated (AT+LETIO=4) or the new display yes/no (AT+BIOCAP=1) configuration is used.

By configuring AT+LETIO=4 for incoming Terminal I/O connections LE Secure Connection usage is mandated for incoming Terminal I/O connections. In such case Terminal I/O connections from devices that only support LE legacy security are rejected.

By configuring AT+BIOCAP=1 for I/O capabilities “display yes/no”, the “yes/no” functionality is only used for LE Secure Connection procedures.

For LE legacy security, only the “display” functionality is used so the results are the same as for a “display only” configuration.

6.3. Security Levels for Terminal I/O

The behavior of LE Security is configurable using the parameters for I/O capabilities (AT+BIOCAP) and a man in the middle protection (AT+BMITM).

The security level of Terminal I/O is configurable using the parameter AT+LETIO.

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
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

AT+BIOCAP sets the input and output capabilities of the device used for LE Security.

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

AT+BMITM controls the man in the middle (MITM) protection of the device during LE Security.

Value	Description
0	Man in the middle protection disabled (default)
1	Man in the middle protection enabled

LE Security defines the following association models based on the Input/Output (I/O) capabilities of the two devices:

- **Just Works**

This method is used when at least one of the devices does not have display capability of six digits and is not capable of entering six decimal digits using a keyboard or any other means (no I/O).

This method does not provide MITM protection (see 6.4 Connection Example Terminal I/O “Just Works”).

- **Passkey Entry**

This method may be used between a device with a display and a device with numeric keypad entry (such as a keyboard), or two devices with numeric keypad entry (see 6.5 Connection Example Terminal I/O “Passkey Entry”).

In the first case, the display is used to show a six-digit numeric code to the user, who then enters the code on the keypad.

In the second case, the user of each device enters the same six-digit numeric code.

Both cases provide MITM protection.

Possible combinations of I/O capabilities and the possibility of MITM protection are listed in the table below. For each case of the “MITM protection” an example of the serial messages between the BlueMod+S42M and the DTE are listed.

In case the user chooses a scenario where MITM protection is not allowed but one of the communication devices is configured to MITM protection, the pairing is refused.

- **Display Yes/No**

This method may be used between two devices with a display and keys that allow the user to accept or reject a connection.

If the Display Yes/No capability is supported by both devices the displays show a 6-digit numerical code. The user is then requested to compare the codes of both displays. If the codes on both displays are equal the user can accept the connection by pressing the “yes” input of both devices. In case the user presses the “no” input on at least one of the devices the pairing becomes rejected.

In cases where only one device of a connection supports Display Yes/No capabilities, the I/O capabilities of that device will be handled as a “Display only” configuration for passkey entry.

BM+S42M \ Remote device	Display only	Display Yes/No	Keyboard only	No input no output	Display and keyboard
Display only AT+BIOCAP=0	Just Works (both automatic confirmation) *No MITM protection*	Just Works (both automatic confirmation) *No MITM protection*	Passkey entry (one display, one input) *MITM protection* SSPPIN <BT addr> <passkey>	Just Works (both automatic confirmation) *No MITM protection*	Passkey entry (one display, one input) *MITM protection* SSPPIN <BT addr> <passkey>
Display Yes/No AT+BIOCAP=1	Just Works (both automatic confirmation) *No MITM protection*	Just Works (both automatic confirmation) No MITM protection	Passkey entry (one display, one input) *MITM protection* SSPPIN <BT addr> <passkey>	Just Works (both automatic confirmation) *No MITM protection*	Passkey entry (one display, one input) *MITM protection* SSPPIN <BT addr> <passkey>
Keyboard only AT+BIOCAP=2	Passkey entry (one display, one input) *MITM protection* SSPPIN <BT addr> ? AT+BSSPPIN <BT addr>,<passkey>	Passkey entry (one display, one input) *MITM protection* SSPPIN <BT addr> ? AT+BSSPPIN <BT addr>,<passkey>	Passkey entry (both input) *MITM protection* SSPPIN <BT addr> ? AT+BSSPPIN <BT addr>,<passkey>	Just Works (both automatic confirmation) *No MITM protection*	Passkey entry (one display, one input) *MITM protection* SSPPIN <BT addr> ? AT+BSSPPIN <BT addr>,<passkey>
No input no output AT+BIOCAP=3	Just Works (both automatic confirmation) *No MITM protection*	Just Works (both automatic confirmation) *No MITM protection*	Just Works (both automatic confirmation) *No MITM protection*	Just Works (both automatic confirmation) *No MITM protection*	Just Works (both automatic confirmation) *No MITM protection*
Display and keyboard AT+BIOCAP=4	Passkey entry (one display, one input) *MITM protection* SSPPIN <BT addr> ? AT+BSSPPIN <BT addr>,<passkey>	Passkey entry (one display, one input) *MITM protection* SSPPIN <BT addr> ? AT+BSSPPIN <BT addr>,<passkey>	Passkey entry (one display, one input) *MITM protection* SSPPIN <BT addr> <passkey>	Just Works (both automatic confirmation) *No MITM protection*	Passkey entry (one display, one input) *MITM protection* incoming connection: SSPPIN <BT addr> ? AT+BSSPPIN <BT addr>,<passkey> outgoing connection: SSPPIN <BT addr> <passkey>

Green color: BM+S42M output message SSPPIN <BT addr> ? (example)
Blue color: BM+S42M input request AT+BSSPPIN <BT addr> <passkey> (example)

The following flow charts will give an example for the different SSP authentication methods “just works” and “passkey entry” within an incoming call request from a smartphone (iOS or Android) using Telit’s Terminal I/O Utility app in combination with the BlueMod+S42M (see also the connection example in chapter 7.6 Connection Example with Smartphone).

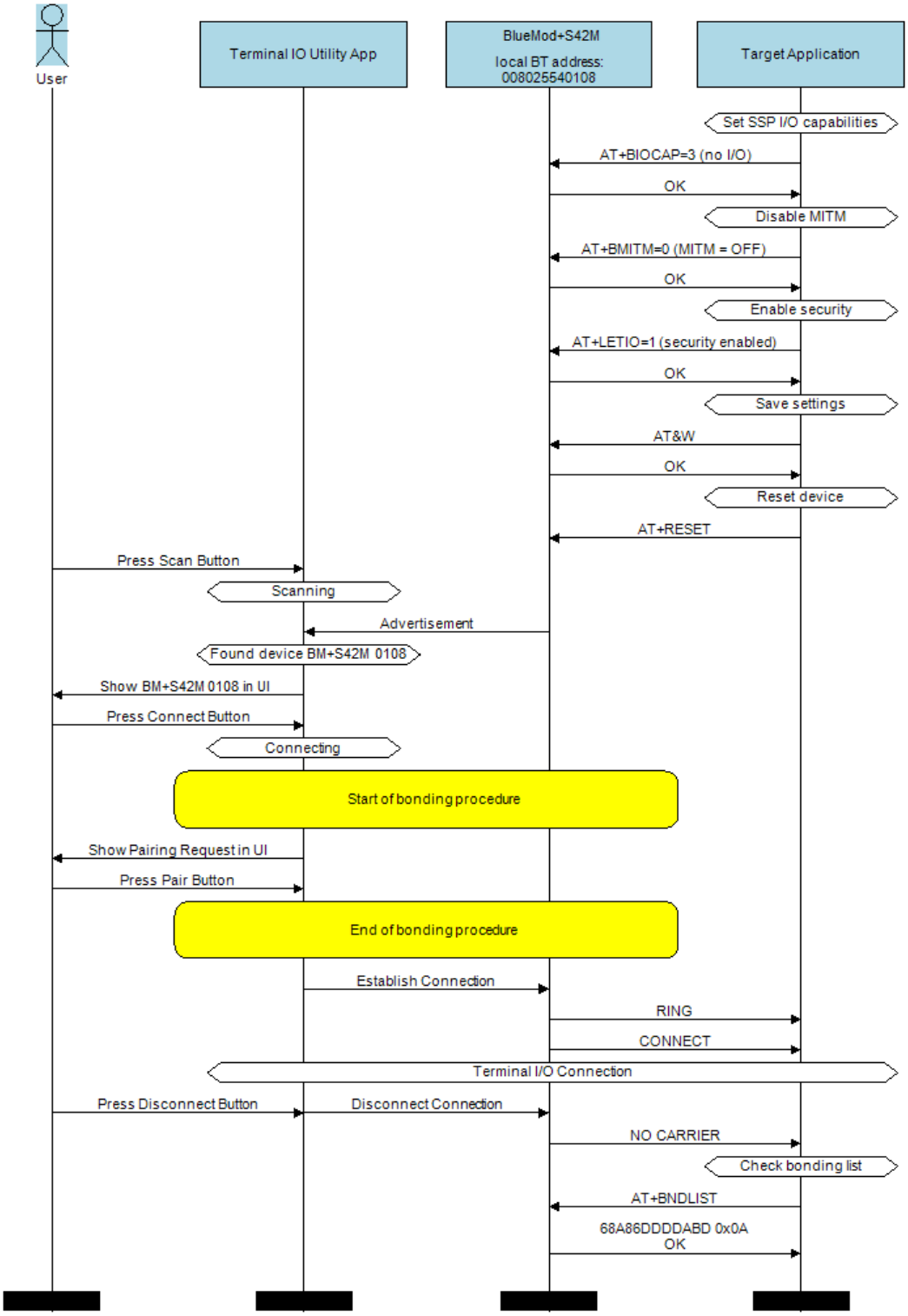
The “*Target Application*” part will simulate the device at the end (DTE) which communicates to the BlueMod+S42M with configuration commands.

The interesting part of the bonding procedure is placed between the yellow boxes “*Start of bonding procedure*” and “*End of bonding procedure*”.

All serial commands between the “*Target Application*” and the “*BlueMod+S42M*” outside of the bonding procedure are used for preparation of LE Security configuration.

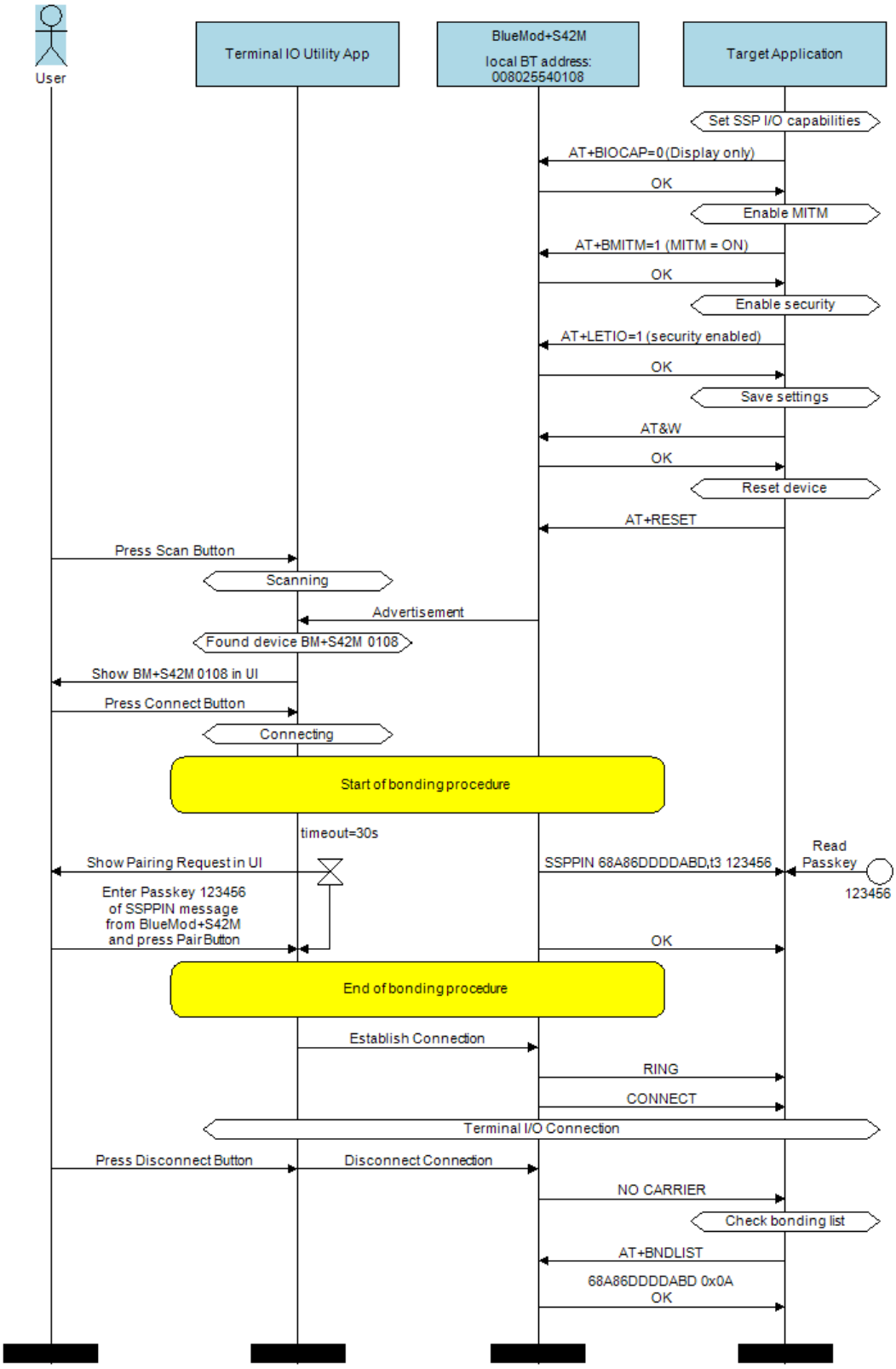
These configuration commands and responses within the flow charts are described in the *BlueMod+S42M AT Command Reference [1]*.

6.4. Connection Example Terminal I/O “Just Works”



6.5. Connection Example Terminal I/O “Passkey Entry”

with I/O capabilities “display only”



7. TERMINAL I/O OVER BLUETOOTH LOW ENERGY

This chapter describes how to setup the BlueMod+S42M to establish a connection using Terminal I/O and Bluetooth Low Energy.

7.1. GAP Role

BlueMod+S42M supports the GAP role peripheral.

A peripheral is a device that advertises by using connectable advertising packets. Searching for Bluetooth Low Energy client devices is not possible.

7.2. Advertising Interval

A device advertises to be visible over the air for other scanning devices. The time between such advertising events is called advertising interval.

To set the advertising interval, use the parameters AT+LEADINTMIN and AT+LEADINTMAX described in the *BlueMod+S42M AT Command Reference [1]*.

7.3. Connection Interval

The connection interval is a delta time that determines the frequency with that the central will transmit and synchronize with a peripheral device during a connection. It can have a huge impact on power consumption and must be set according to the required scenario.

To set the connection interval, use the parameters AT+LECONINTMIN and AT+LECONINTMAX described in the *BlueMod+S42M AT Command Reference [1]*.

7.4. Slave Latency

The slave latency is important for the power consumption of a peripheral device. It sets the number of master connection intervals that the slave can ignore. Setting this value to a high number increases the latency of the device.

To set the slave latency, use the parameter AT+LESLAVELAT described in the *BlueMod+S42M AT Command Reference [1]*.

7.5. Local Device Name

To set the local device name, use the parameter AT+BNAME described in the *BlueMod+S42M AT Command Reference [1]*.

With Bluetooth Low Energy the value of BNAME is used in the LE SCAN_RESP message (advertising) and in the GAP characteristic.

It depends on the scanning device when and which name it displays.

iOS for example takes the name from the advertising in the first step and updates it with the GAP name once it discovers the services from the device.

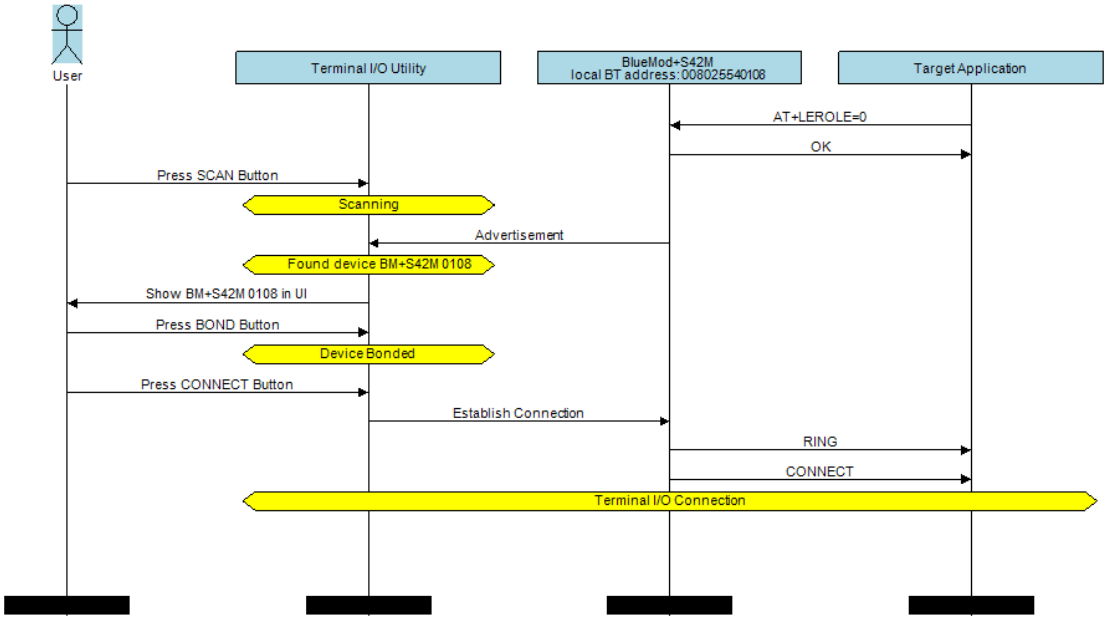
7.6. Connection Example with Smartphone

To establish a Bluetooth Low Energy connection from a smartphone to the BlueMod+S42M the “Terminal IO Utility” App from Telit needs to be installed on the smartphone or Tablet.

The following QR-Codes provide the link to download the “Terminal IO Utility”.

iOS **Logo** **Android**

The Terminal IO Utility App allows the user to connect to Terminal I/O peripheral devices and exchange data providing a simple terminal emulation.



As soon as the connection is established, data can be sent from smartphone to BlueMod+S42M and vice versa.

8. UART INTERFACE CONTROL PROTOCOL (UICP)

8.1. General Protocol Description

Telit UART Interface Control Protocol (UICP) defines a protocol to control the logical state of an UART based interface, thereby peers to switch off local UART devices for power saving (or other) reasons.

The UICP+ is a bi-directional, symmetrical protocol that allows to negotiate UART interface states with a communication partner connected via UART by using of standard UART signal lines.

The UICP+ mechanisms defined here enable the involved peers to negotiate UART interface states by signaling the remote peer it is allowed to enter or exit an UART interface up state.

The UICP+ does not enforce any power saving support of the involved peers but implements mechanisms to allow the save usage of MCU power saving features like UART peripheral switched off.

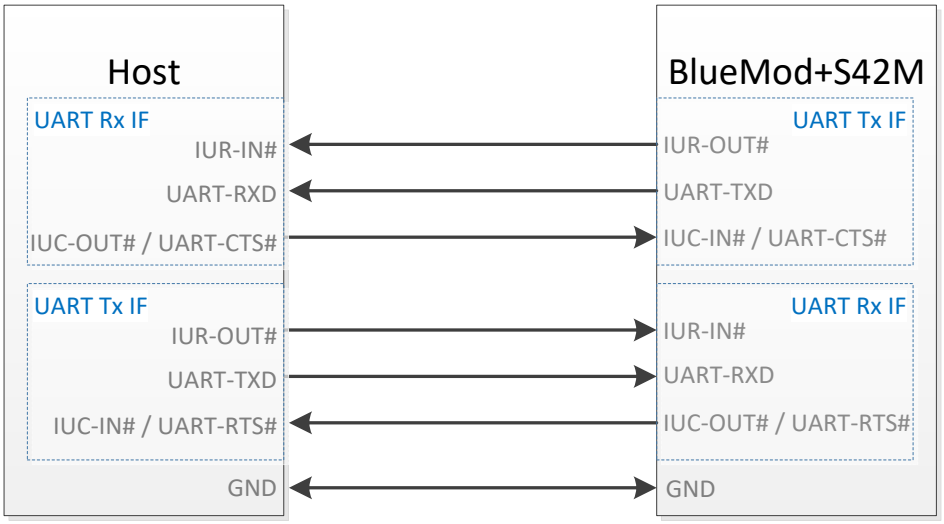
8.2. Requirements of Using UICP on BlueMod+S42M

To make use of UICP, the lines UART-TXD, UART-RXD, UART-RTS# (IUC-OUT#), UART-CTS# (IUC-IN#), IUR-OUT# and IUR-IN# should be connected between BlueMod+S42M and the host and additionally the UICP protocol should be implemented on host site.

A detailed description of implementing UICP is described in the document *UICP+ UART Interface Control Protocol [2]*.

To activate UICP on the BlueMod+S42M the configuration parameter AT+UICP=1 needs to be set.

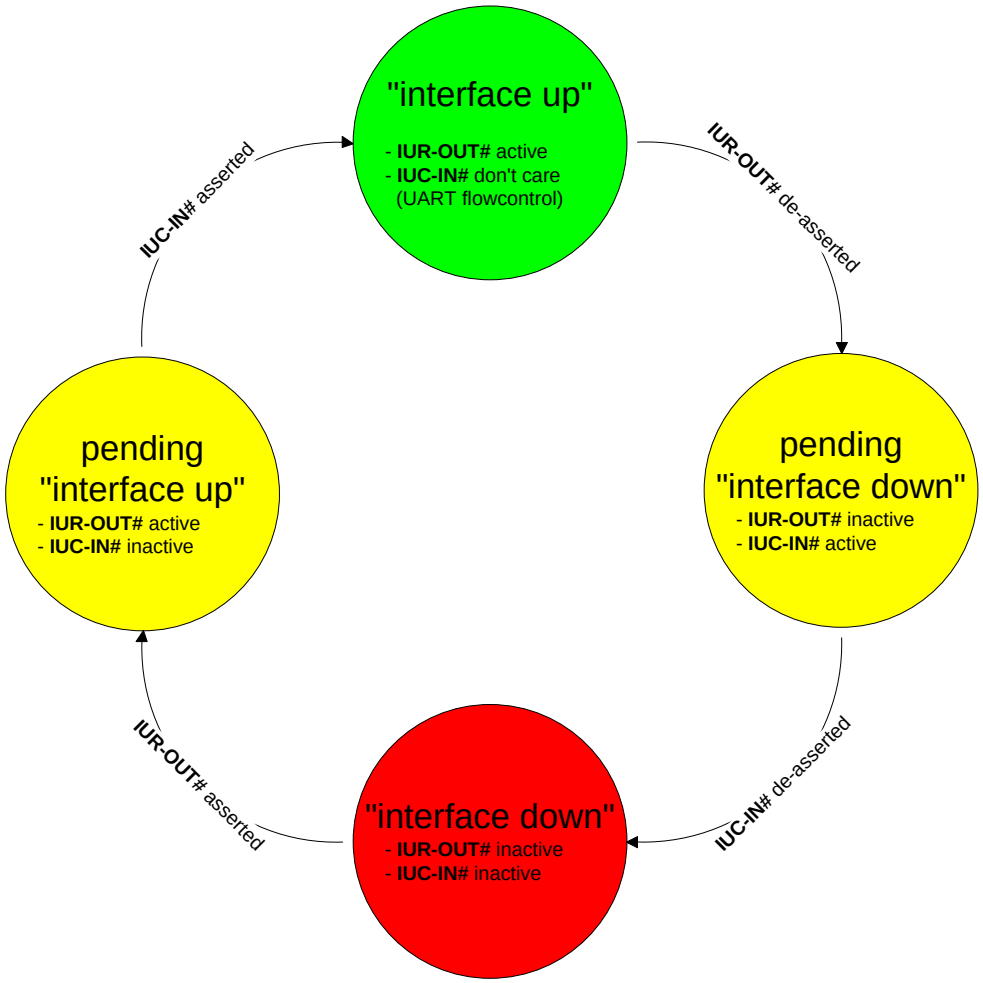
8.3. Connection Example between BlueMod+S42M and Host Controller



Further information about the BlueMod+S42M UART interface is described in the document *BlueMod+S42M Hardware User Guide [3]*.

8.4. UICP Protocol States

The UICP protocol defines four states:



- **interface up**
  normal operation, RTS/CTS hardware flow control is active
- **pending interface down**
  IUR-OUT# is requested to go to “interface down” state
  IUC-IN# is not confirmed
- **interface down**
  IUR-OUT# and IUC-IN# are de-asserted in “interface down” state and can enable MCU power saving
- **pending interface up**
  IUR-OUT# is requested to go to “interface up” state,
  IUC-IN# is not confirmed

All data received before the interface up state has been achieved shall be seen as invalid data and shall be discarded.

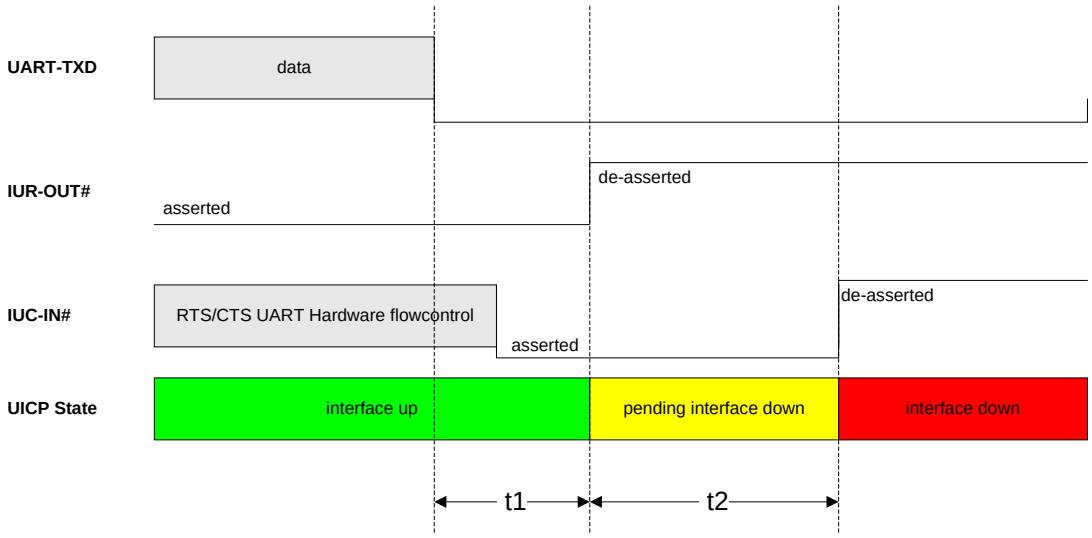
After reset in activated UICP configuration the initial state is “interface down”, in case of non connected host BlueMod+S42M remain in “interface down”.

8.4.1. Drive from “interface up” to “interface down” State

Once a de-asserted IUR-OUT# signal of the initiator is detected by the acceptor, the acceptor shall confirm that signal by de-asserting its IUC-OUT# signal which is connected to the IUC-IN# signal of the initiator.

After the initiator detects a de-asserted IUC-IN# signal both devices go into “interface down” state and can enable MCU power saving mechanisms.

During MCU power saving, the MCU can switch off the UART but shall be able to detect an IUR# assert.



t1 >= 100 ms (see this chapter)

t2 < 1 s

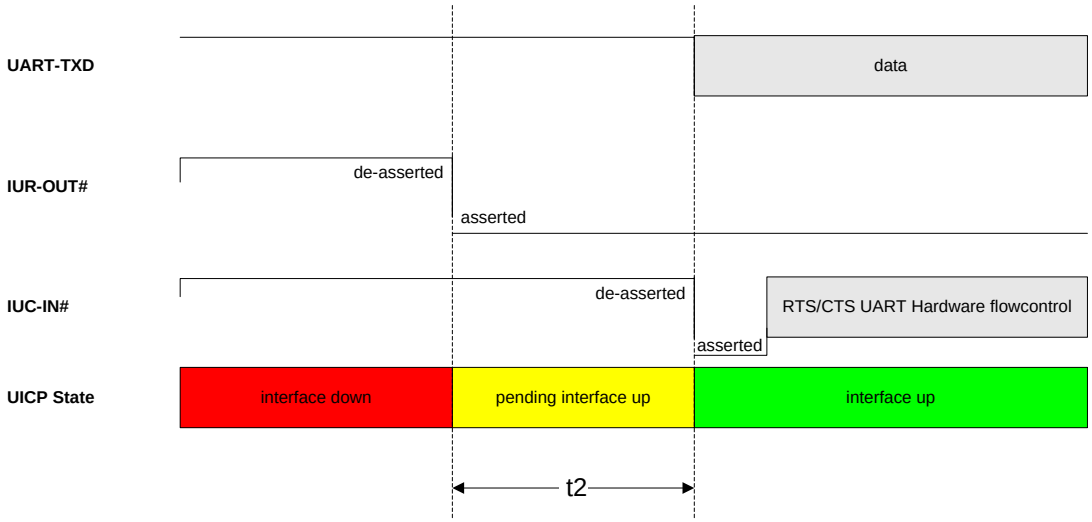
8.4.2. Drive from “interface down” to “interface up” State

To initiate the state change from “interface down” state to “interface up” state the initiator shall assert the IUR-OUT# signal.

The acceptor confirms the IUR-IN# signal with asserting its IUC-OUT# signal which is connected to the IUC-IN# signal of the initiator.

Once the acceptor detects the assert of the IUR-OUT# signal from the initiator, it can disable MCU power saving mechanisms but shall ensure the UART is ready to receive data before it confirms asserting its IUC-OUT# signal which is connected to the IUC-IN# signal of the initiator.

Once the initiator detects the assert of the IUC-IN# signal of the acceptor, the in initiator can send data to the acceptor.



8.5. Example of UICP Usage

The following examples shows the state change between the BlueMod+S42M and the host.

The scenario here might be that both devices use the “interface down” state to drive the MCU into some kind of power saving mode that allows to “wake up” the MCU with external GPIO signals.

8.5.1. State Change from “interface up” to “interface down”

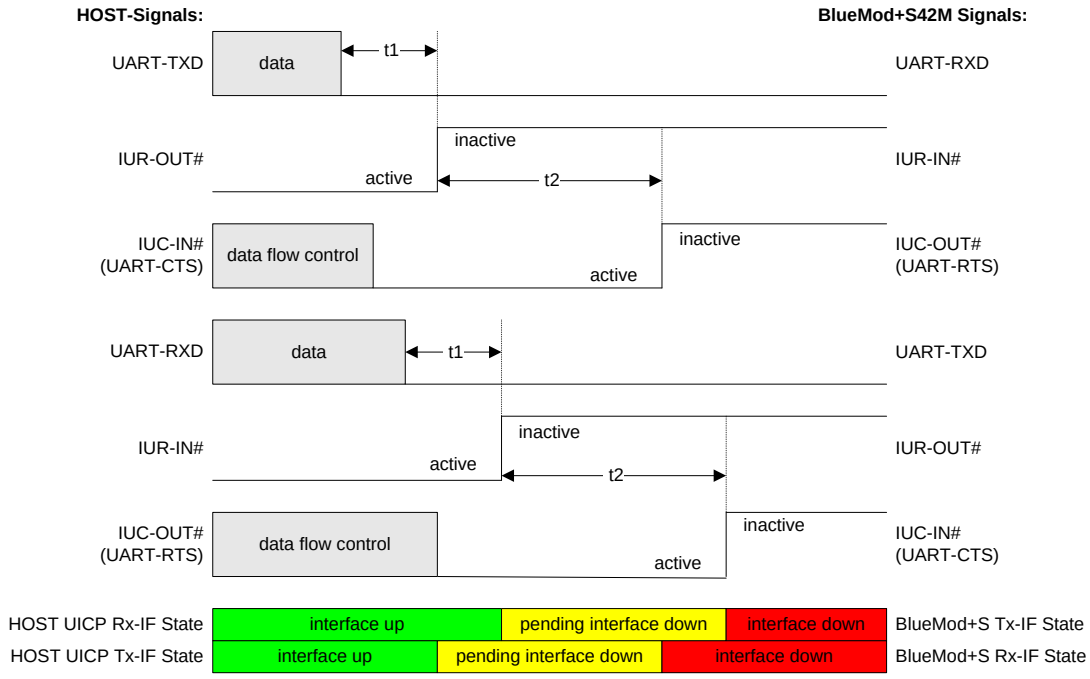
Host and BlueMod+S42M are in the state “interface up” and exchange bidirectional data. After the host has send all data and is idle for **t1** in its Tx direction it signals the BlueMod+S42M it is allowed to go to “interface down” state by de-asserting

IUR-OUT# signal.

Parallel to that UICP signaling from host to BlueMod+S42M the BlueMod+S42M has send all data as well and is idle for **t1** in its Tx direction, so it signals the host it is allowed to go to “interface down” state by de-asserting IUR-OUT# signal.

The host and the BlueMod+S42M each wait for a maximum time **t2** to detect the de-asserted IUC-IN# signal. After receiving this input change via the IUC-IN# signal both devices may change from state “pending interface down” to state “interface down”.

Both UICP signaling sequences proceed in parallel until host and BlueMod+S42M interfaces are in “interface down” state.



8.5.2. State Change from “interface down” to “interface up”

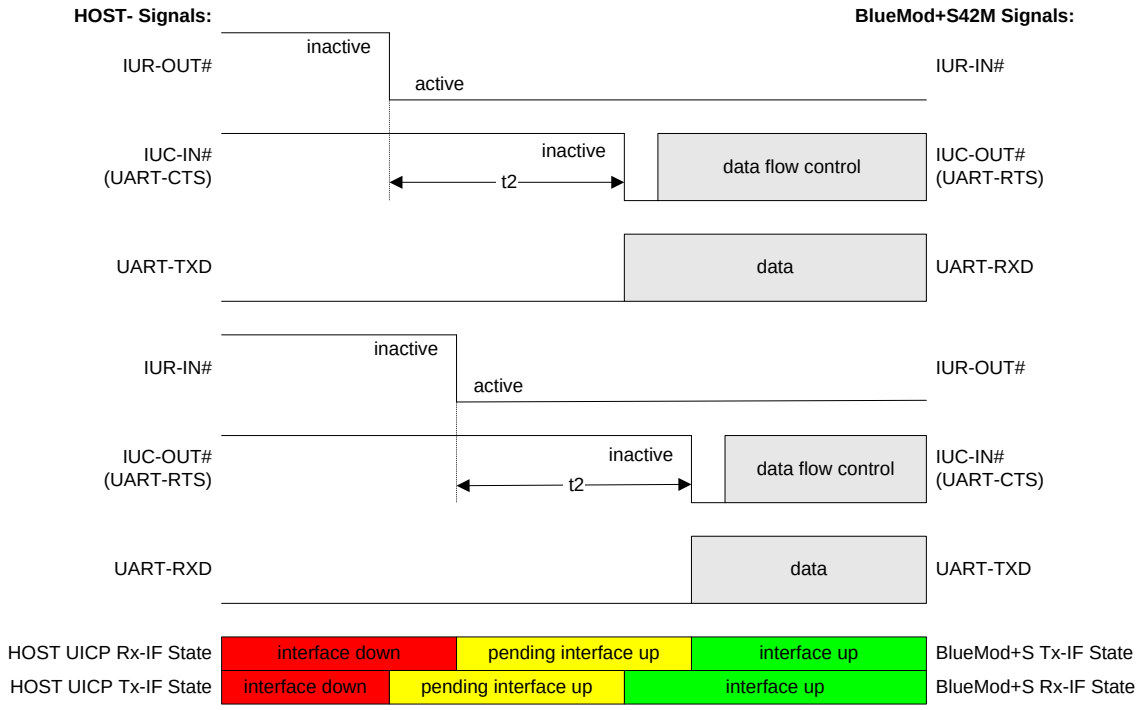
Host and BlueMod+S42M are in the state “Interface down” and may have the MCU into some kind of power saving states.

The host wants to send data to the BlueMod+S42M and asserts its IUR-OUT# signal.

Parallel to that UICP signaling from host to BlueMod+S42M the BlueMod+S42M wants to send data to the host and asserts its IUR-OUT# signal as well.

The host and the BlueMod+S42M each wait for a maximum time **t2** to detect the assertion via the IUC-IN# signal. After receiving this input change of IUC-IN# both devices may assume that the interface of the remote device changed from state “pending interface up” to state “interface up”.

Both UICP signaling sequences proceed in parallel until host and BlueMod+S42M interfaces are in “interface up” state and data can be exchanged bidirectional.



9. GATT SERVER CONFIGURATION

This chapter describes the preparation of an own GATT server by using the integrated AT commands via the serial UART interface. With these commands a user can setup his own GATT services or profiles in the BlueMod+S42M and define its own advertising data and timing. The services can exist in addition to the Terminal I/O service or stand-alone.

The integrated Terminal I/O service acts as a UART based cable replacement. Activating this service will automatically set predefined advertising data. A detailed description of the Terminal I/O service is listed chapter 7 Terminal I/O over Bluetooth Low Energy.

The exchanged data of a defined characteristic is enclosed within an AT command or response message.

9.1. Configuring the Advertising

When using the integrated Terminal I/O profile it is not possible to include an additional UUID within the existing Terminal I/O advertising.

If the user wants to set the UUID of a service other than Terminal I/O in the advertising, the Terminal I/O advertising has to be disabled by setting AT+LEADE=3.

The complete set of advertising data has to be set with the commands: AT+LEADPAR, AT+LEADDATA, AT+LESCDATA.

The advertising starts, as soon as it is enabled by AT+LEADE=1.

All referenced services shall be defined before enabling advertising, otherwise the device sends advertising without providing the appropriate services for a short time.

9.1.1. List of available UUIDs

A list of reserved 16bit UUIDs is available onto the web site of the Bluetooth SIG https://www.bluetooth.org.

9.1.2. Beacon Mode

To configure the BlueMod+S42M as a Beacon or iBeacon please follow the instructions in the document *Using the BlueMod+S as a Beacon [5]*.

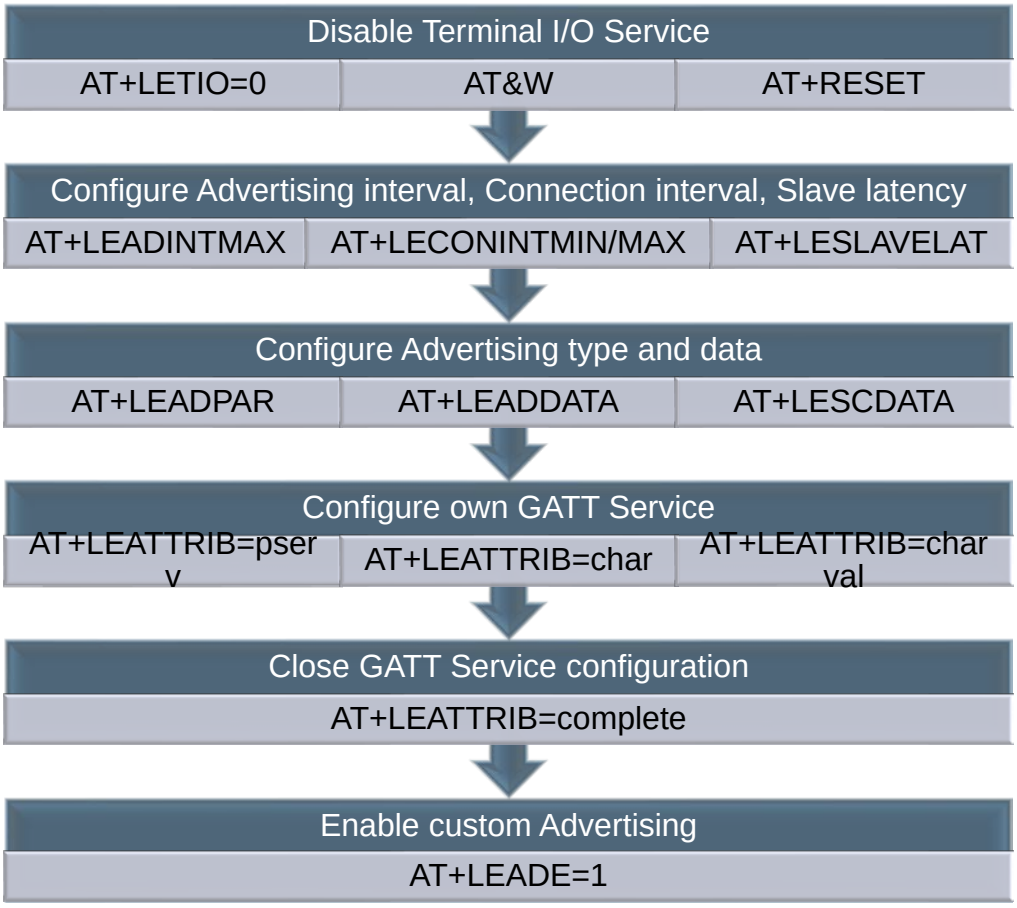
9.2. Configure own GATT Service

This part described the required steps to create an own GATT service.

If the integrated Terminal I/O service shall keep enabled the custom specific advertising type and data is not required. In this case the custom advertising may not activated.

For a detailed description of the GATT configuration commands refer to the *BlueMod+S42M AT Command Reference [1]*.

The following flow chart lists the different steps to configure the module with a custom specific GATT server.



9.3. GATT Service Example

To demonstrate the setup of an own GATT service the following example lists all required steps based on the “Health Thermometer Monitor” (HTM) profile.

9.3.1. GATT Service Initialization

This GATT service demo creates one GATT service with the following identities:

Health Thermometer Service UUID: 0x1809
See also: https://www.bluetooth.com/specifications/gatt/services

The implemented Health Thermometer Service includes the mandatory Service Characteristics Requirements (Temperature Measurement).
Compare to: HTP / Health Thermometer Profile
https://www.bluetooth.com/specifications/adopted-specifications

The used timing values for the advertising timing and connection interval are taken random. Specified values can be taken from the HTM Profile specification.

Host Controller	BlueMod+S42M
AT+LETIO=0	OK
AT&W	OK
AT+RESET	OK
AT+LEADINTMAX=3500	OK
AT+LEADCONINTMAX=500	OK
AT+LEADCONINTMIN=200	OK
AT+LESLAVELAT=0	OK
AT+LEADPAR=ADVTYPE=0	OK
AT+LEADDATA=020106030209180C0954454D5045524154555245	OK
AT+LESCDATA=03020918	OK
AT+LEATTRIB=pserv,uuid=1809	OK
AT+LEATTRIB=char,prop=20	OK
AT+LEATTRIB=charval,uuid=2A1C,perm=0001,len=20	0x20 OK
AT+LEATTRIB=complete	OK
AT+LEADE=1	OK

9.3.2. Detection of an Established GATT Connection

When the module is configured with a custom GATT server an incoming connection is silently accepted. There is no UART based message from the module to report an active GATT connection like it is available for the Terminal I/O connection.

A low layer GATT connection can be detected with the GPIO3 “IOB”. Use the command AT+IOBCFG=2 to enable this option. For a detailed description of these command refer to the *BlueMod+S42M AT Command Reference [1]*.

9.3.3. Exchange Data Between Host and Connected Destination

To demonstrate HTM specific data the following data packets from the host can be sent to simulate a real temperature value.

- Temperature definition: 39.40°C
- Position: finger
- Date/Time: 05.December 2012, 11:52:42 A.M.

Data	Value	Description
0x 06		Temperature measurement flags
0x 0F 64 (00 FE)	39,40 °C	Temperature measurement value in Celsius
0x 07 DC	2012	Date/Time base: year
0x 0C	12 / December	Date/Time base: month
0x 05	05	Date/Time base: day
0x 0B	11 / AM	Date/Time base: hours
0x 34	52 minutes	Date/Time base: minutes
0x 2A	42 seconds	Date/Time base: seconds
0x 04	finger	Temperature type

The data packet needs to be sent with the following AT command to the defined “channel”:

AT+LESRVDATA=0x20,06640F00FEDC070C050B342A0400000000

9.4. Test own Configured GATT Service

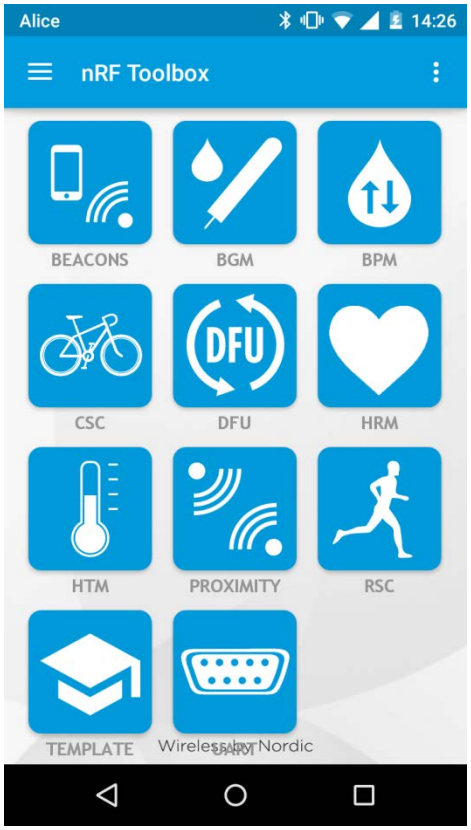
In order to test the own created GATT service there are a lot of applications available.

We will demonstrate the functionality of the own created Health Thermometer Monitor (HTM) GATT service with two applications:

- Nordic nRF Toolbox using HTM device
  This application will only connect to the BlueMod+S42M device and display the received measured value.
- Nordic nRF Master Control Panel
  This application allows detailed information of the configured BlueMod+S42M device (advertising data, services and characteristic information)

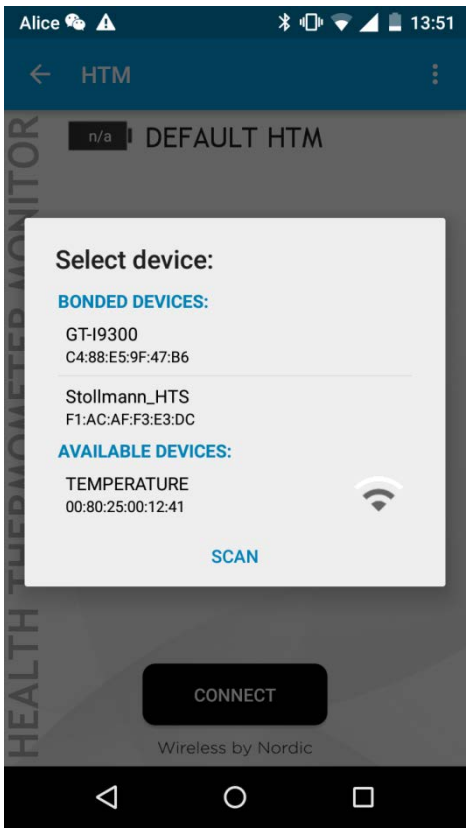
9.4.1. Connect from Smartphone with Nordic nRF Toolbox

- Start the nRF Toolbox application and select the “HTM” icon to search for Health Thermometer Monitor devices.



- Start scanning
- The BlueMod+S42M is displayed as “TEMPERATURE” because the local name of the advertising data is set to this identifier

```
AT+LEADDATA=02010603020918
0C0954454D5045524154455245
```

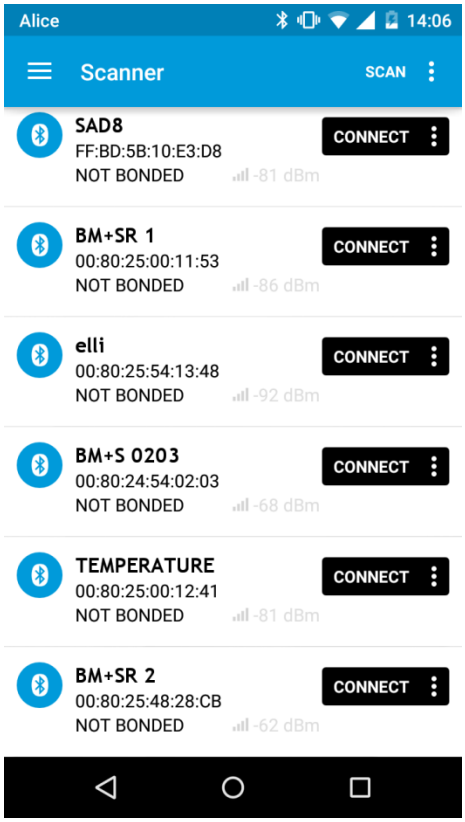


- Press the “CONNECT” button to establish the connection to the BlueMod+S42M device.
- To read a value on the smartphone side it is required to send a data packet from the host controller to the BlueMod+S42M device.

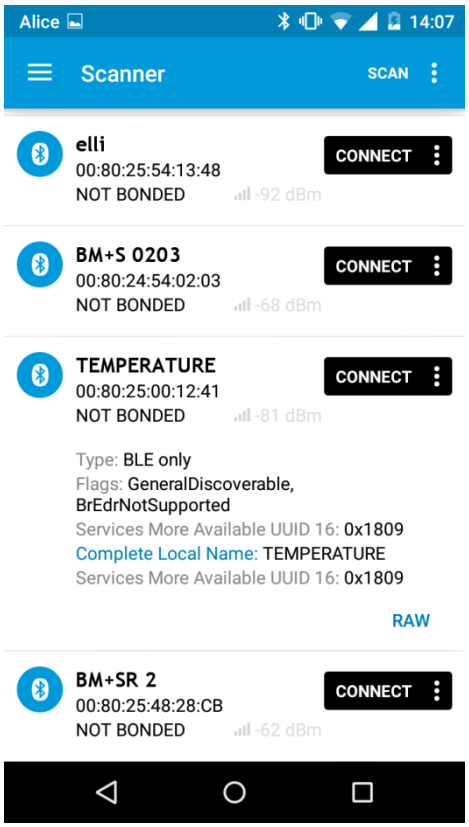
9.4.2. Connect from Smartphone with Nordic nRF Master Control Panel

- Start Nordic Master Control Panel application.
- This application will list all connectable devices.
- The BlueMod+S42M is displayed as “TEMPERATURE” because the local name of the advertising data is set to this identifier

```
AT+LEADDATA=02010603020918
0C0954454D5045524154455245
```

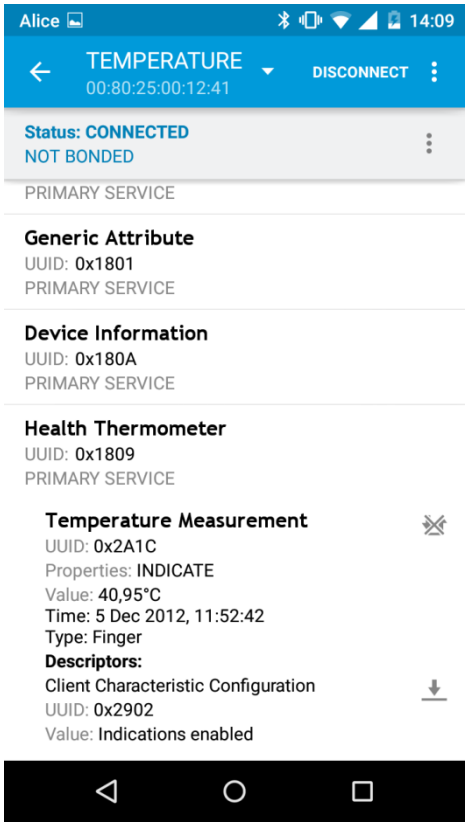


- Reading the details of the selected device “TEMPERATURE” will display the configured advertising data.



- After connecting to the BlueMod+S42M device the primary services will be listed in the application.
- To read a value on the smartphone side it is required to send a data packet from the host controller to the BlueMod+S42M device.
- This data content is listed as value in the appropriate service.

(For detailed information about the data content please contact Bluetooth SIG, https://www.bluetooth.org.)



10. SYSTEM OFF MODE

The BlueMod+S42M supports the possibility to set the module into low power mode during the time the module is not used with the AT+SYSTEMOFF command.

The BlueMod+S42M will restart on activity at the GPIO input lines UART-RTS#, IUR-IN# or GPIO[4].

The host controller can use the IOA pin (GPIO[8]) to monitor the system status. Please also verify the configuration of the AT+IOACFG parameter.

It is also possible to monitor the UART flow control line UART-RTS#.

10.1. Using System OFF Mode for Terminal I/O

The following example will list the communication between the host controller and the BlueMod+S42M using the integrated Terminal I/O profile.

To set the BlueMod+S42M into the low power mode the host controller needs to send the AT+SYSTEMOFF command.

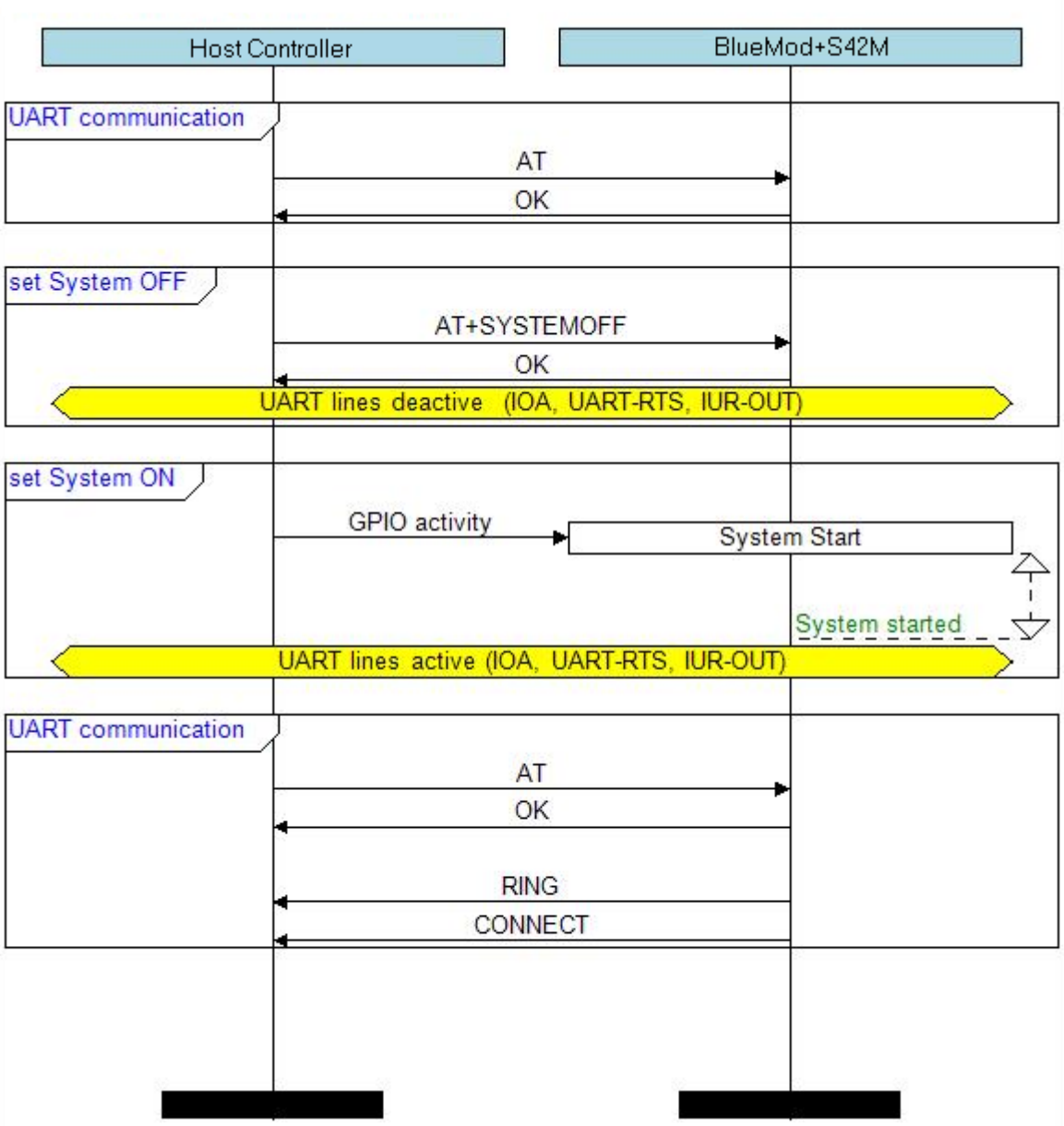
The BlueMod+S42M will respond “OK” before changing into low power mode.

To activate the BlueMod+S42M from low power mode the host controller needs to activate one of the following GPIO lines: UART-RTS#, IUR-IN#, GPIO[4]

The module detects the GPIO change and starts the firmware.

After the firmware is started the host can continue the UART communication.

An incoming call is reported with RING and CONNECT.


Host Controller
BlueMod+S42M
UART communication
AT
OK
set System OFF
AT+SYSTEMOFF
OK
UART lines deactive (IOA, UART-RTS, IUR-OUT)
set System ON
GPIO activity
System Start
System started
UART lines active (IOA, UART-RTS, IUR-OUT)
UART communication
AT
OK
RING
CONNECT

11. FIRMWARE UPDATE

This section describes the step-by step procedure of updating the firmware on a BlueMod +S42M via the local UART interface. New firmware update can add new features, as well as provide a way to fix bugs, and protects you from security vulnerabilities.

A firmware update can be performed using the BM+S42M Updater tool or Over the air.

11.1. BM+S42M Updater

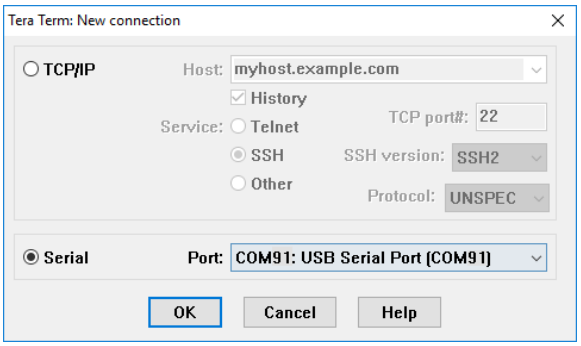
The BM+S42M Updater is a Windows™ program that contains the firmware and uses a PC with a serial port for the update. The file name of the executable program consists of version and patch information.

11.1.1. Prerequisites

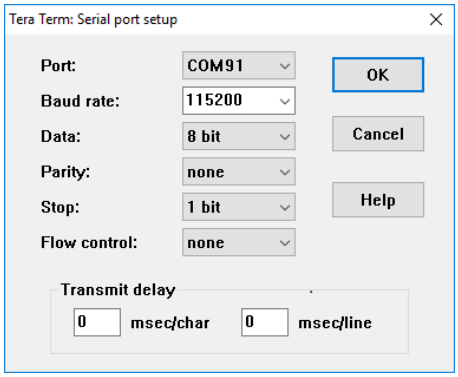
- A PC running on Windows® operating system.
- Download and extract the **Telit_BlueModS42M_v1.2.1_FW_Update** package from the Telit- Download Zone to your local drive.
- A serial communication interface such as Tera Term.

11.1.2. Procedure

1. Connect the module to the PC using the USB cable.
2. Open Tera Term on your PC.
3. Select the **Serial** radio button and select **Port** from the drop-down list, and then click **OK.**



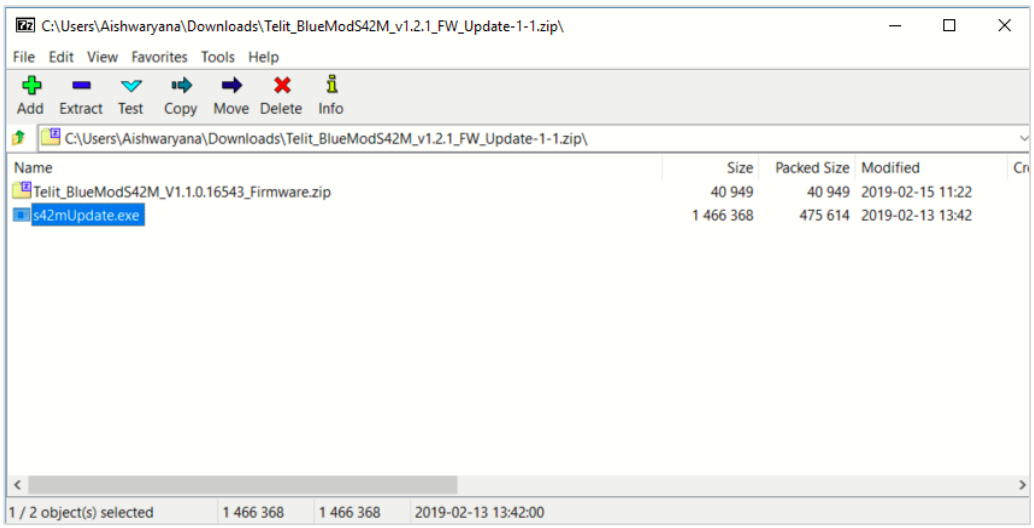
4. Select **Setup>Serial port.** The **Serial port setup** window appears.
5. Configure the setting as shown below to change the device into AT command mode.
   - Baud rate: 115200
   - Data: 8 bit
   - Parity: None
   - Stop: 1 bit
   - Flow Control: none (select “hardware” option during the data flow)



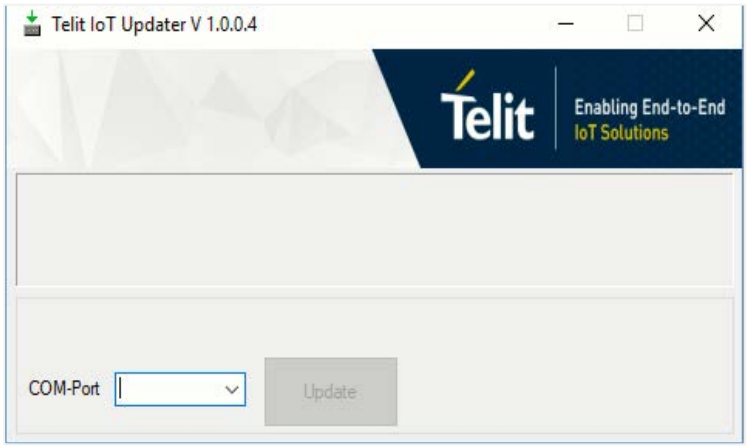
6. Once connection is established, issue the following command to display the associated firmware version as shown below.

   ATi99

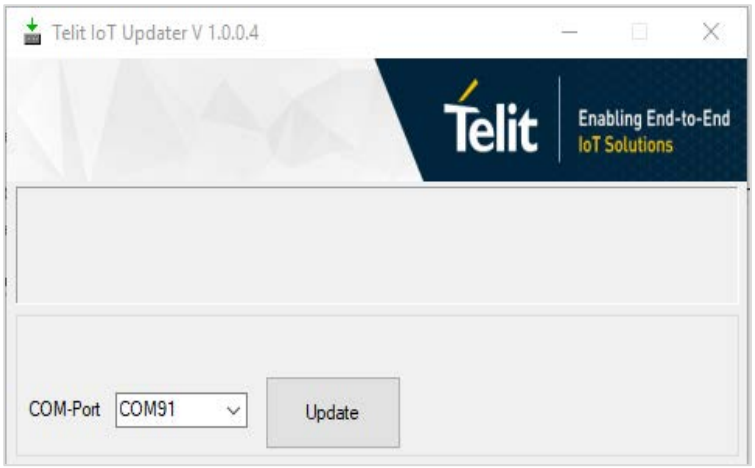
7. Now, disable the COM Port in Tera term and navigate to the extracted **Telit_BlueModS42M_v1.2.1_FW_Update** package.



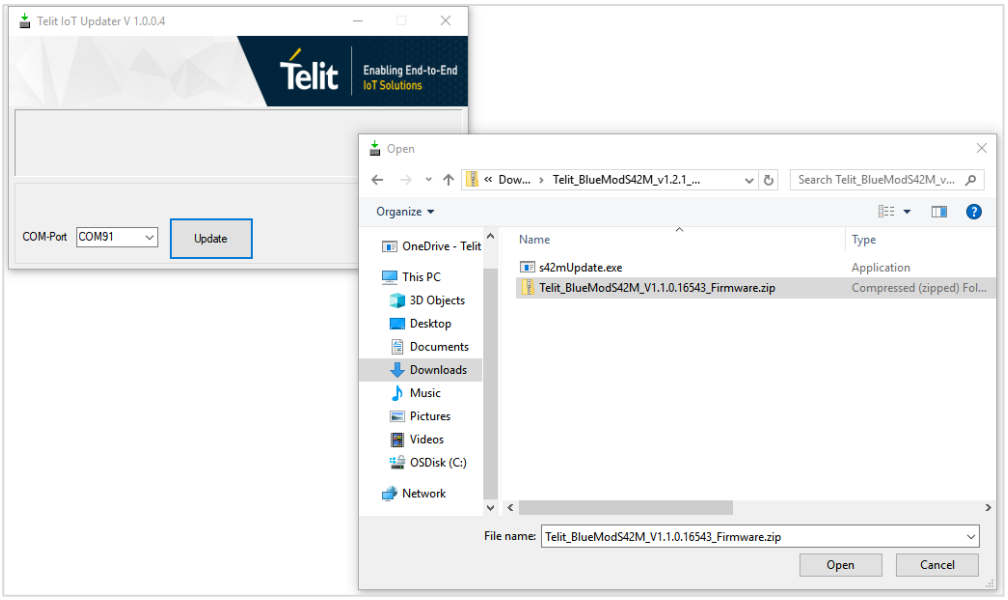
8. Double-click on the **“s42mUpdate.exe”** file to launch the **Telit IoT Updater 1.0.0.4**.



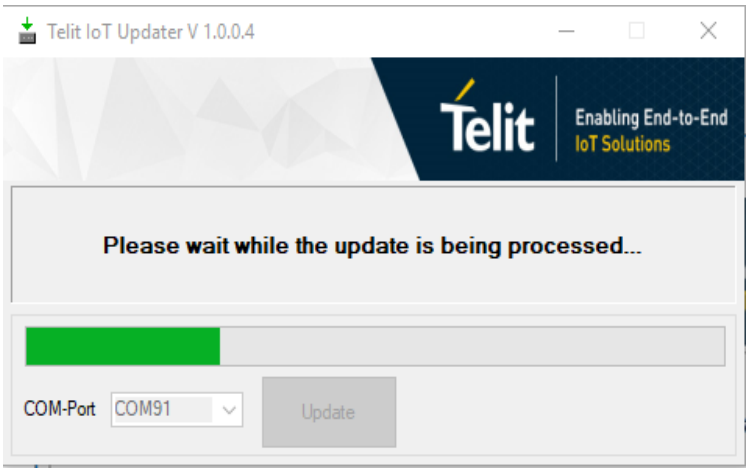
9. Select the associated **COM-Port** from the drop-down list and click **Update** to navigate to the folder where you have extracted the **Telit_BlueModS42M_v1.2.1_FW_Update** package.



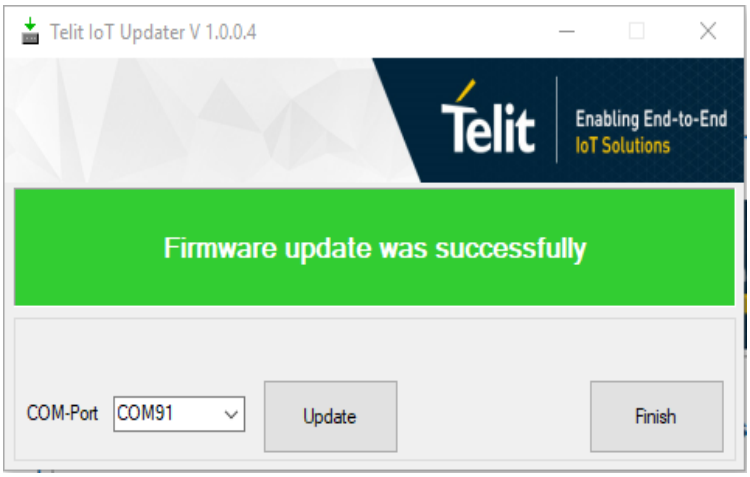
10. Select the **Telit_BlueModS42M_V1.1.0.16543_Firmware.zip** file as shown below and click **Open** to upload the file.



11. A progress bar on Telit IoT Updater will indicate how much **%** of the file is uploaded.



12. After a successful update, click **Finish**.



13. To verify the firmware version updated on the module, reset the BlueMod+S42M device, enable the COM port in Tera Term and issue the following command:

    `ATi99`

    The device will show the updated version as follows:

11.2. Firmware Update Over the Air (OTA)

The BlueMod+S42M supports firmware update over the air. The firmware update over the air can be performed using the Realtek App. Currently, OTA is supported only in Android devices.

11.2.1. Firmware Update OTA using Realtek App on Android Devices

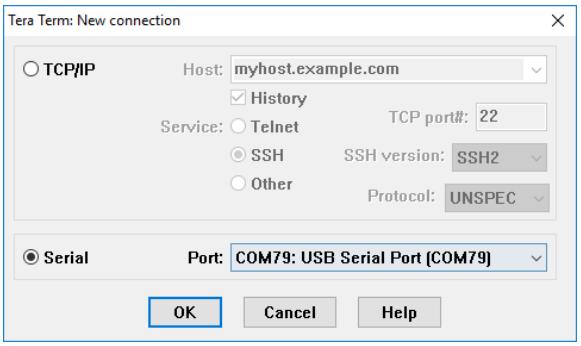
The following section provides a step-by-step procedure to perform an OTA firmware update using a Realtek app on an Android device.

11.2.1.1. Prerequisites

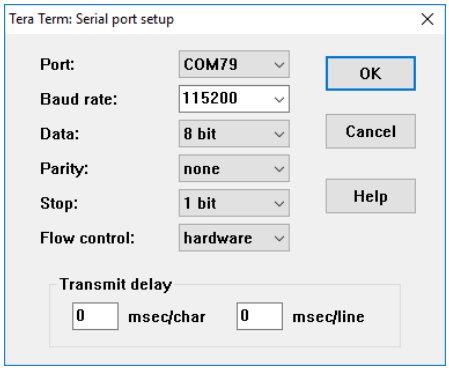
- A PC running on Windows® operating system.
- Bluetooth enabled Smartphone running Android 4.3 or later
- Download and extract **Telit_BlueModS42M_V1_2_1_Firmware_Update_OTA** package from the Telit-Download Zone and transfer**(.bin)** into your mobile device.
- Download and install the **Realtek app(.apk)** from the Telit-Download Zone on your mobile device.
- A serial communication interface such as Tera Term.

11.2.1.2. Procedure

1. Connect the module to the PC using the USB cable.
2. Open Tera Term on your PC.
3. Select the **Serial** radio button and select **Port** from the drop-down list, and then click **OK.**

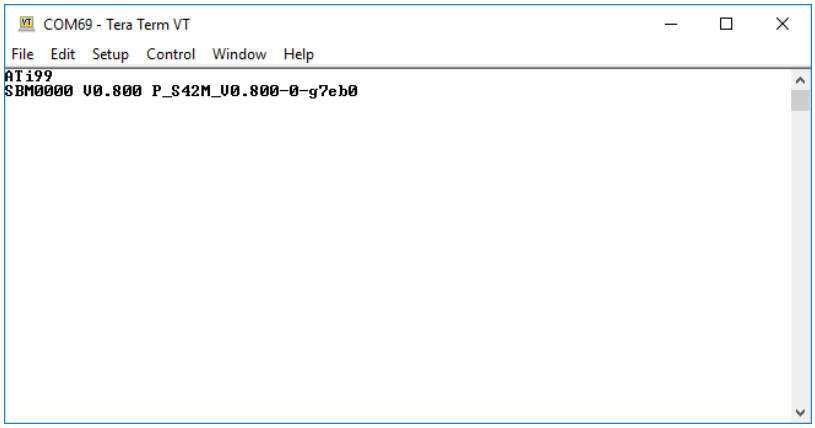


4. Select **Setup>Serial port.** The **Serial port setup** window appears.
5. Configure the setting as shown below to change the device into AT command mode.
   - Baud rate: 115200
   - Data: 8 bit
   - Parity: None
   - Stop: 1 bit
   - Flow Control: hardware



6. Once connection is established, issue the following command to display the associated firmware version as shown below.

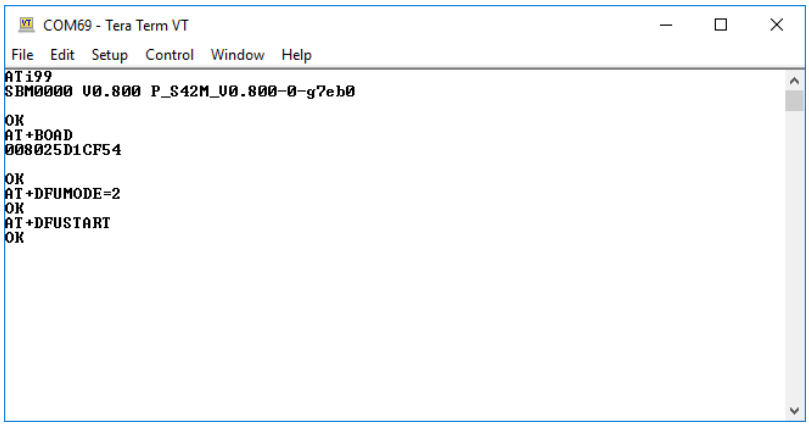
   ATi99



7. To perform OTA update, issue the following AT commands.

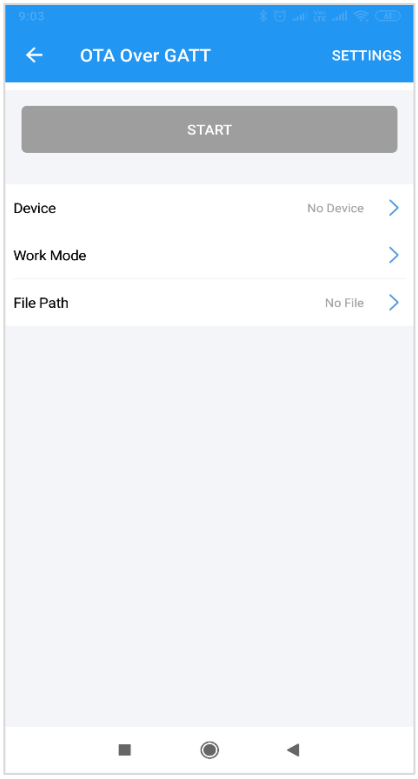
   AT+DFUMODE=2

   AT+DFUSTART

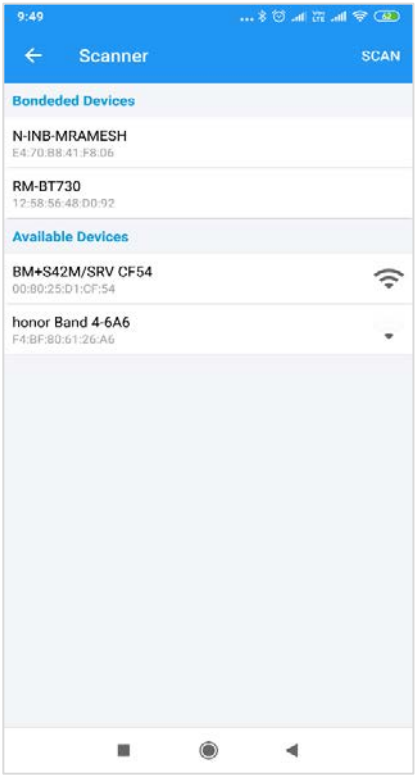


8. Now, launch the Realtek app on your mobile device. On the main screen, under **GATT** option, tap **FUNCTION TEST.**

9. Tap **Device,** to begin search for your devices.

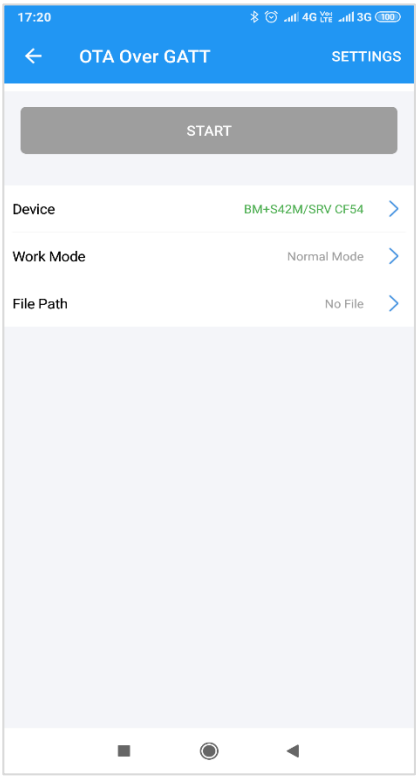


10. Once your device is found, tap on the device name to connect.

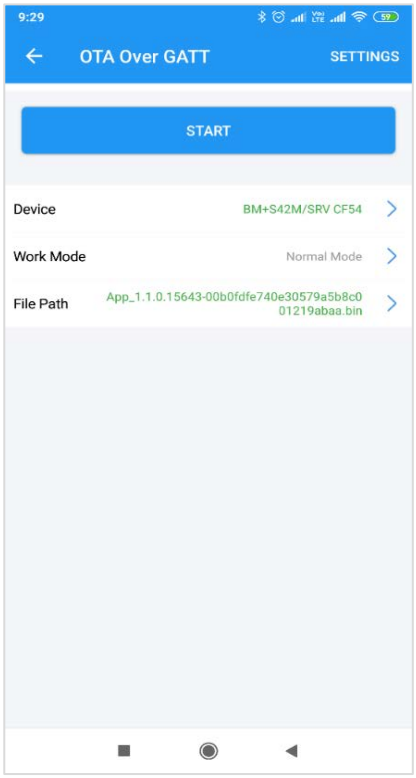


11. Atfer a successful device connection, the following screen is displayed.

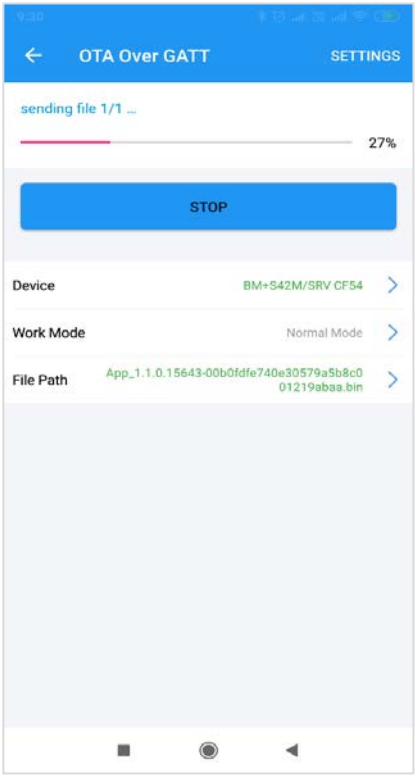
    Note: By default, the **Work Mode** is selected as **Normal Mode**.



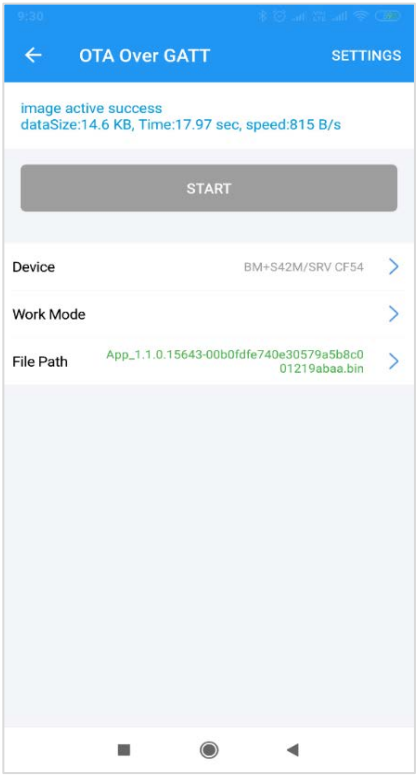
12. Next, tap **File Path** to search and select the associated **.bin** file which was previously copied to the mobile, and then press the **START** button.



13. A progress bar will indicate how much **%** of the file is uploaded.



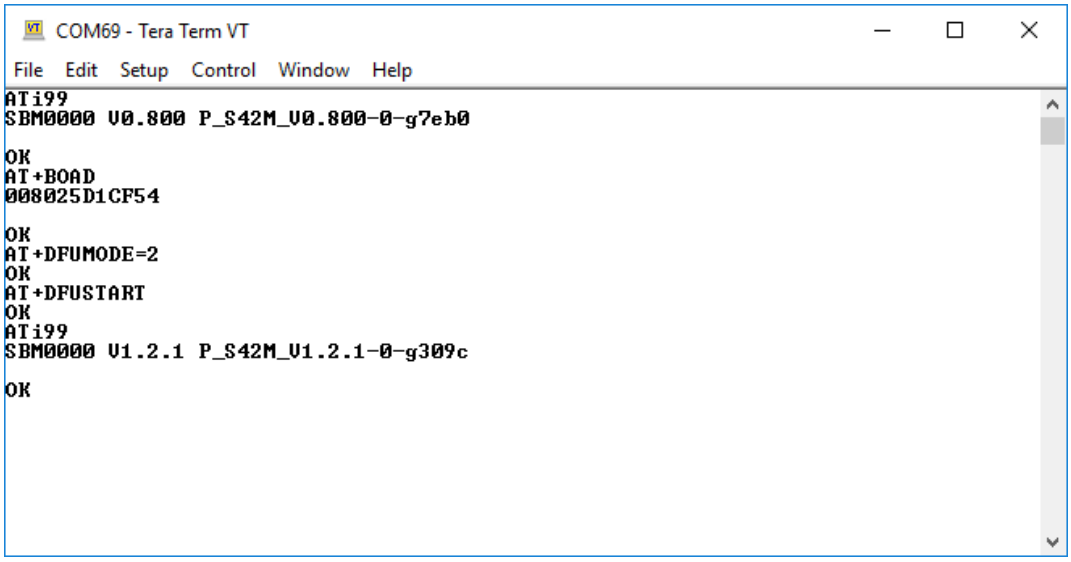
14. Once the file is sent successfully, the following message is displayed.



15. To check the firmware version updated on the module, reset the BlueMod+S42 device and issue the following AT Command

    `ATi99`

    The device will show the updated version as follows:



12. GLOSSARY AND ACRONYMS

AT	Attention Command
GAP	Generic Access Profile
GATT	Generic Attribute Profile
SSP	Secure Simple Pairing
UART	Universal Asynchronous Receiver/Transmitter
UICP	UART Interface Control Protocol
UUID	Universal Unique Identifier
OTA	Over The Air

13. DOCUMENT HISTORY

Revision	Date	Changes
0	2017-11-16	First issue
1	2019-02-15	Added step-by step procedure for performing a serial firmware update under section 11.1BM+S42M Updater.
2	2019-02-20	Added step-by step procedure for performing a firmware Update Over the Air under section 11.2Firmware Update Over the Air (OTA)

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

Mod. 0815 2016-08 Rev.1

[04.2016]