



# ComAPI+ API Documentation

30515ST10841A Rev. 4 – 2017-07-20

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

SPECIFICATIONS ARE SUBJECT TO CHANGE WITHOUT NOTICE

## NOTICES LIST

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

# CONTENTS

<b>NOTICES LIST .....</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>
I. License Agreements .....	3
II. Copyrighted Materials .....	3
III. High Risk Materials .....	3
IV. Trademarks .....	3
V. Third Party Rights .....	3
<b>APPLICABILITY TABLE .....</b>	<b>4</b>
<b>CONTENTS .....</b>	<b>5</b>
<b>1. INTRODUCTION .....</b>	<b>7</b>
1.1. Scope .....	7
1.2. Contact Information, Support .....	7
1.3. Text Conventions .....	8
1.4. Commonly Used Terms .....	8
1.5. Related Documents .....	9
<b>2. PRELIMINARY INFORMATION .....</b>	<b>10</b>
<b>3. OVERVIEW .....</b>	<b>11</b>
3.1. Function Table .....	11
3.2. Event Table .....	11
3.3. Status Code Table .....	12
3.4. HCI Packet exchange .....	13
3.5. Flow control .....	14
3.6. Callback Functions .....	15
3.6.1. PComCallback .....	15
<b>4. FUNCTION DESCRIPTION .....</b>	<b>16</b>
4.1. comOpen .....	16
4.2. comClose .....	17
4.3. comWrite .....	18

4.4. comResponse..... 19

5. **DOCUMENT HISTORY** ..... **20**

## 1. INTRODUCTION

### 1.1. Scope

This document describes ComAPI+, a lower layer data communication interface within the Telit Bluetooth stack architecture.

### 1.2. Contact Information, Support

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

- [TS-EMEA@telit.com](mailto:TS-EMEA@telit.com)
  - [TS-AMERICAS@telit.com](mailto:TS-AMERICAS@telit.com)
  - [TS-APAC@telit.com](mailto:TS-APAC@telit.com)
- or
- [TS-SRD@telit.com](mailto:TS-SRD@telit.com) for global Bluetooth support

Alternatively, use:

<http://www.telit.com/support>

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

<http://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.

### 1.3. 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.

### 1.4. Commonly Used Terms

HCI	Host Controller Interface
NOOP	no operation
TBD	To be defined



## 1.5. **Related Documents**

- [1] BlueAPI+ API Documentation, 30515ST10833A
- [2] OSIF API Documentation, 30515ST10842A

## 2. PRELIMINARY INFORMATION

ComAPI+ defines a generic Telit application programming interface used to access data communication devices.

A software adapter or software stack exposing ComAPI+ as its upper layer interface is called a ComAPI+ provider.

The BlueCode+SR uses the ComAPI as simple, hardware independent HCI transport for full HCI packets. Due to this hardware independence, BlueCode+SR is not aware of any HCI chip, target hardware, or other target specifics, so all target specific initialization and configuration must be handled within the ComAPI+ provider. This includes but is not limited to any vendor specific HCI chip initialization and the configuration of communication interface and speed.

For embedded target platforms, proprietary operating systems etc., vendors of embedded devices might be required to implement an appropriate ComAPI+ provider as a data transport between BlueCode+SR and the Bluetooth chip.

As ComAPI+ represents a generic interface, not all functions defined within ComAPI+ source are required by BlueCode+SR.

This document describes ComAPI+ with regard to its relevance within the BlueCode+SR context only and provides the required information for ComAPI+ provider implementers.

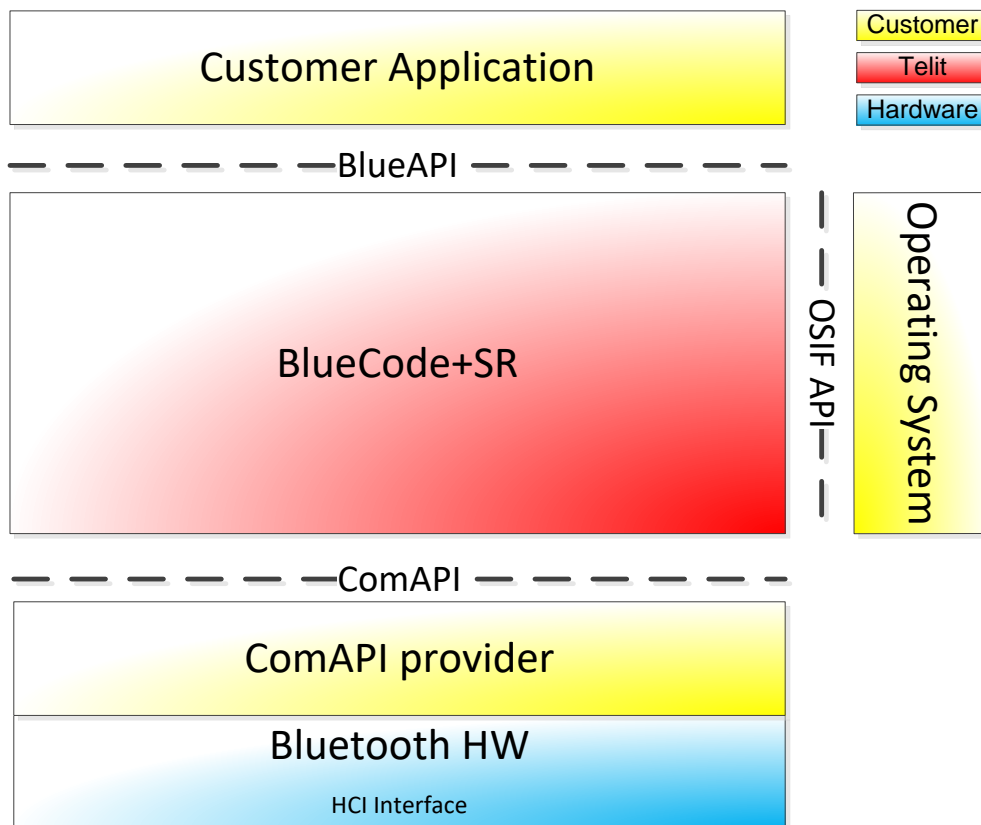


Figure 2-1: Block diagram of lower layer Bluetooth architecture

### 3. OVERVIEW

#### 3.1. Function Table

Function	Description
comOpen	Open a communication device.
comClose	Close a communication device.
comWrite	Write HCI packet via a communication device.
comResponse	Confirm indicated HCI packet

*Table 3-1 API Functions*

#### 3.2. Event Table

Event	Description
comeDataIndication	HCI packet from HCI controller has been received within PComCallback parameter pvBuffer
comeDataTransmitted	HCI packet transmitted; PComCallback parameter pvBuffer contains pointer to application provided write buffer that may be re-used
comeOpenCompleted	A communication device has been opened
comeCloseCompleted	A communication device has been closed
comeError	A communication error has occurred

*Table 3-2 Events*



Further events defined within ComAPI.h are not relevant within the Bluetooth / BlueCode+SR context and are not listed here.

### 3.3. Status Code Table

Each function returns a status code of type COMENTRY that can identify the following events:

Event	Description
comResultSuccess	Operation completed successfully
comResultPending	Operation in process
comResultUnspecified	Operation failed in unspecified manner
comResultNoResources	Operation failed due to out of memory condition
comResultOpenfailed	Lower layer initialization failed
comResultInvalidHandle	The given handle is invalid
comResultInvalidLength	The given length is invalid
comResultInvalidState	Operation failed due to invalid state condition
comResultInvalidParam	A supplied parameter is invalid
comResultTimeout	A synchronization operation failed with timeout

*Table 3-3 Status codes*



Further events defined within ComAPI.h are not relevant within the Bluetooth / BlueCode+SR context and are not listed here.

---

### 3.4. HCI Packet exchange

ComAPI+ is used for the exchange of full HCI packets, so each comWrite call or comeDataIndication event represents a full HCI packet.

There are four kinds of HCI packets that can be exchanged via the ComAPI interface; i.e. HCI Command Packet, HCI Event Packet, HCI ACL Data Packet and HCI Synchronous Data Packet.

HCI Command Packets can only be sent to the Bluetooth Host Controller, HCI Event Packets can only be sent from the Bluetooth Host Controller, and HCI ACL/Synchronous Data Packets can be sent both to and from the Bluetooth Host Controller.

HCI does not provide the ability to differentiate the four HCI packet types.

Therefore, if the HCI packets are exchanged via the ComAPI+ interface, a HCI packet indicator has to be added to each HCI packet according to the following Table:

HCI Packet type	HCI packet indicator
HCI command packet	0x01
HCI ACL data packet	0x02
HCI synchronous data packet	0x03
HCI event packet	0x04

*Table 3-4 HCI packet indicators*

The HCI packet indicator shall be set in the first byte of the packet exchanged followed by the HCI packet data. Over the ComAPI+ interface, only HCI packet indicators followed by HCI packet data are allowed.



This mechanism follows the definition of the HCI UART interface as specified by the Bluetooth Core specification. Due to that, a Bluetooth Host controller (HCI Chip) that communicates via UART to the ComAPI provider will already implement this HCI packet indicator mechanism in its signaling so no additional action is required for the ComAPI provider implementation.

---

### 3.5. Flow control

BlueCode+SR provides flow control functionality to an Application by allowing that Application to delay the acknowledge of data received via BlueCode+SR.

In the other direction, BlueCode+SR handles back pressure applied by the remote peer by delaying the acknowledge of data received from the Application.

Due to the fact that this sliding window flow control mechanism of BlueCode+SR is channel and connection based, it cannot be adopted to the ComAPI interface since it routes all channels/connection to one single communication path.

Instead, the ComAPI interface is designed to handle back pressure by signaling and handling `comResultNoResources` status in cases where HCI packets cannot be forwarded due to flow control situations.

Example:

A Bluetooth connection is established, but for whatever reason the HCI Controller is not able to handle new HCI packets via UART from the ComAPI provider and applies UART flow control.

The ComAPI provider should still be able to receive some additional HCI packets via the ComAPI interface by queuing them internally until the UART flow control is released by the HCI controller, but on a certain point such internal Queue is full so the ComAPI provider must apply flow control to the BlueCode+SR stack.

To do that, the ComAPI provider can return a `comResultNoResources` status in cases where the BlueCode+SR stack tries to forward a HCI packet via the `comWrite` function to the ComAPI provider but the ComAPI provider is not able to store or forward them properly.

The BlueCode+SR stack will then internally queue any HCI packet to be send until its internal Queue is full.

In such cases the BlueCode+SR stack will delay acknowledges to the application for messages received from the application, so the application using the BlueCode+SR stack for communication becomes aware of the flow control situation and can react accordingly.

As soon as the HCI controller releases the UART flow control, the ComAPI provider shall continue to forward the HCI packets queued internally and signal a `comeDataTransmitted` event for each HCI packet forwarded to the HCI Controller successfully.

If the BlueCode+SR receives such a `comeDataTransmitted` event, it will resume forwarding HCI packets queued internally to the ComAPI provider and resuming the acknowledgement for messages received from the application as soon as it is appropriate, so the application gets notified if a flow control situation is resolved.

For the opposite direction, an equal mechanism is implemented where the BlueCode+SR stack refuses to receive HCI packets from the ComAPI provider in cases where no more HCI frames can be handled due to internal or by the application initiated flow control mechanisms.

In such cases, the ComAPI provider shall queue the HCI packets and/or apply flow control to the HCI controller until the BlueCode+SR stack uses the `comResponse` function to signal that a received HCI packet is handles successfully and new HCI packets can be handled.

## 3.6. Callback Functions

### 3.6.1. PComCallback

Callback function prototype for communication event signaling.

```
typedef TComResult (WINAPI * PComCallback) (
    void*      pvContext,
    TComEvent  event,
    uint32_t   uStatus,
    void*      pvBuffer,
    uint32_t   uLength
);
```

Parameter	Type	Function
pvContext	void*	Application-defined context value.
event	TComEvent	Signaled event type.
uStatus	uint32_t	Status value.
pvBuffer	void*	Pointer to event type dependent argument buffer.
		comeDataIndication: Pointer to HCI packet received from HCI controller.
		comeDataTransmitted: Pointer to write buffer ready for re-use.
uLength	uint32_t	Number of bytes in argument buffer

Return value:

- comResultSuccess in case of successful operation.
- One of the remaining return status codes (see 3.3) otherwise.

## 4. FUNCTION DESCRIPTION

### 4.1. comOpen

Open a communication device.

The event comeOpenCompleted shall be signaled after the device opening procedure has been completed.

Any vendor or target specific HCI Chip or hardware initialization procedures (e.g. open and configure HCI UART, transfer of HCI Chip ROM patches, device specific Bluetooth address and any calibration data to the HCI chip) shall be completed by the ComAPI implementation before the comeOpenCompleted event is signaled.

If the comeOpenCompleted event is received by the BlueCode+SR stack, the stack expect the communication device and HCI Chip to be fully configured and functional.

BlueCode+SR will call comClose() by Application request (see 4.2) with the handle received in phCom in order to close a successfully opened communication device and release all allocated resources.

COMENTRY

```
comOpen (
    char*      pszDeviceName,
    char*      pszParameter,
    PComCallback callback,
    void*      pvContext,
    HCOM*      phCom
);
```

Parameter	Type	Function
pszDeviceName	char*	Name of communication device to open.
pszParameter	char*	Device specific parameter string, or NULL.
callback	PComCallback	Application-defined event callback function receiving information about communication events (see 3.2, 3.6.1).
pvContext	void*	Application-defined context returned within callback function.
phCom	HANDLE*	Pointer receiving interface handle on successful interface creation.

Return value:

- comResultSuccess in case of successful operation.
- One of the remaining return status codes (see 3.3) otherwise.



The parameter pszDeviceName and pszParameter are not used within the BlueCode+SR context and shall be ignored by the ComAPI implementation.



## 4.2. comClose

Close a communication device that was previously opened with comOpen().

This function is used within the actual BlueCode+SR context to switch the radio off on Application request (RadioModeSetReq(RadioOff)).

```
COMENTRY
comClose (
    HCOM*          phCom
);
```

Parameter	Type	Function
phCom	HANDLE	Communication device handle created via comOpen().

Return value:

- comResultSuccess.

### 4.3. comWrite

Send HCI packet to HCI controller.

The event `comeDataTransmitted` shall be signaled after the HCI packet has been sent and the buffer provided in `pBuffer` may be freed or re-used.



The event `comeDataTransmitted` shall not be generated from within the `comWrite` function but asynchronous after the `comWrite` function terminated.

```
COMENTRY
comWrite
(
    HCOM      hCom,
    uint8_t*  pBuffer,
    uint32_t  uBufferLength
);
```

Parameter	Type	Function
<code>hCom</code>	<code>HCOM</code>	Communication device handle created via <code>comOpen()</code> .
<code>pBuffer</code>	<code>uint8_t*</code>	Pointer to buffer containing the packet to be written.
<code>uBufferLength</code>	<code>uint32_t</code>	Number of bytes to write.

Return value:

- `comResultSuccess` in case of successful operation.
- One of the remaining return status codes (see 3.3) otherwise.

#### 4.4. **comResponse**

Confirm indicated RX HCI packet.

This function will be called for each RX HCI packet previously signaled via `comeDataIndication` to acknowledge the HCI packet. The buffer provided in `pvBuffer` may be freed or re-used.

```
COMENTRY
comResponse (
    HCOM hCom,
    uint8_t* pvBuffer
);
```

Parameter	Type	Function
<code>hCom</code>	HCOM	Communication device handle created via <code>comOpen()</code> .
<code>pvBuffer</code>	<code>void*</code>	pointer to a buffer containing the packet to be confirmed.

Return value:

- `comResultSuccess` in case of successful operation.
- One of the remaining return status codes (see 3.3) otherwise.

## 5. DOCUMENT HISTORY

---

Revision	Date	Changes
1	2013-08-26	First issue
2	2013-11-15	Removed property functions and defined exchanged data as full HCI packets Introduced flow control chapter Introduced HCI packet exchange chapter
3	2015-06-08	Fixed ComClose() description to define radio-off functionality
4	2017-07-20	First Telit issue

---



# 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