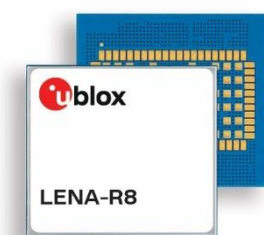




LENA-R8 series

Application development guide

Application note



Abstract

This application note provides detailed technology architecture and examples of how to use AT commands with u-blox LENA-R8 series modules

Document information

Title	LENA-R8 series	
Subtitle	Application development guide	
Document type	Application note	
Document number	UBX- 22038374	
Revision and date	R01	29-May-2023
Disclosure restriction	C1-Public	

This document applies to the following products:

Product name
LENA-R8 series

Contents

Document information	2
Contents	3
1 Introduction	6
2 Cellular technology overview	7
2.1 LTE Cat 1bis	7
3 Application design and development	8
3.1 Initial design decisions	8
3.2 Application stages	9
3.2.1 Persistent configurations	9
3.2.2 Power on/boot	9
3.2.3 Network registration	9
3.2.4 IP acquisition	9
3.2.5 Core application	9
3.2.6 Power-off	9
3.3 Application modes	10
3.3.1 Debug / test mode	10
3.3.2 Production testing	10
3.3.3 Certification mode	10
3.3.4 Firmware update mode	10
4 AT commands response parser	11
4.1 AT command response description	11
4.1.1 Echo	11
4.1.2 Information text response	11
4.1.3 Final result code	12
4.1.4 Unsolicited Result Code	12
4.1.5 Response time	12
4.2 Guidelines for robust AT parser	13
4.2.1 Basic recommendations	13
4.2.2 Multiple AT interfaces	13
4.3 Operational modes of the AT interface	13
5 Local connectivity	15
5.1 Serial interface configuration	15
5.2 USB mode	15
5.2.1 USB Serial ports only	15
5.2.2 RNDIS and serials	16
5.2.3 ECM and serials	16
5.2.4 USB firmware update mode	17
5.3 AT interface on UART	17
5.3.1 Set a fixed baud rate	17
5.4 AT interface on USB	18

5.5	USB network modes	18
5.5.1	Bridge mode	18
5.5.2	Router mode	19
5.6	Power saving	19
5.6.1	USB interface.....	20
5.6.2	UART interface.....	20
5.7	Multiplexer (MUX)	20
5.8	Point-to-point protocol (PPP)	21
5.8.1	Basic setup.....	21
5.8.2	Swap between Data mode and OnLine Command Mode	21
6	User settings persistence	23
6.1	Save user settings	23
6.2	Restore factory configuration	23
7	Cellular modem services	24
7.1	Network registration	24
7.1.1	Band selection	24
7.2	Network attach and PDN connections	24
7.2.1	Initial default bearer	24
7.2.2	Additional bearers	25
7.3	Voice and data services	26
7.3.1	Voice calls.....	26
7.4	Mobility scenarios	26
8	Monitoring module status.....	27
8.1	Retrieve and interpret diagnostic information	27
8.1.1	Diagnostic information via URCS	28
8.1.2	Diagnostic information via polling.....	29
8.2	Full-stack watchdog: how to react to unexpected conditions.....	30
9	Internet protocols	31
9.1	Data security.....	31
9.1.1	Certificates manager +USECMNG.....	31
9.1.2	Profile configuration +USECPRF	31
9.1.3	Complete example.....	33
9.1.4	Troubleshooting secure connection	34
9.2	TCP/UDP internal stack.....	34
9.2.1	Socket connect.....	34
9.2.2	Socket listening	35
9.2.3	Socket write (+USOWR)	35
9.2.4	Socket read (+USORD)	36
9.2.5	Socket close.....	36
9.3	Internet clients	36
9.3.1	HTTP.....	36
9.3.2	FTP.....	38

9.3.3	MQTT	42
9.3.4	MQTT-SN.....	44
10	SIM.....	46
10.1	SIM architecture and behavior	46
10.1.1	SIM card and SIM profiles	46
10.1.2	eSIM/eUICC and remote SIM provisioning	46
10.1.3	SIM subscription	46
10.2	SIM communication.....	46
10.2.1	Commands for restricted access	47
10.2.2	Commands for generic access.....	47
10.2.3	SIM logical channels.....	48
11	FW update	50
11.1	Firmware update Over AT (FOAT)	51
11.1.1	FOAT via +UFWUPD AT command	51
11.2	Firmware Over The Air (FOTA)	52
11.2.1	Firmware download	52
11.2.2	Firmware validation and installation +UFWINSTALL	54
11.3	Flashing.....	55
11.3.1	System setup	55
11.3.2	Short instructions	55
11.4	Impact to device files and settings	56
12	OEM production testing.....	57
13	Migration guides.....	58
13.1	LARA-R6 series to LENA-R8 series.....	58
Appendix	60
A	Glossary	60
Related documentation	61
Revision history	61
Contact	61

1 Introduction


This document provides guidance for developing applications that interface with the u-blox LENA-R8 series modules, including examples of AT command sequences for specific use cases.


[Table 1](#) shows a summary of the documentation available for LENA-R8 series modules.

	Document name	Notes
Application integration	Application development guide app note	This document. Start here!
	EVK-R8 user guide [4]	Starting guide for the LENA-R8 evaluation kit.
	Production and validation test app note [5]	Guidelines of OEM production test and validation test. Contact tech support for this document.
Reference documentation	Data sheet [1]	Use these documents as hardware integration and AT commands API reference.
	System integration manual [2]	
	AT commands manual [3]	
Product release documents	Sample Delivery Note / Information Note	Delivered with every FW release.

Table 1: LENA-R8 documentation overview

The following symbols are used to highlight important information within this document:

 An index finger points out key information pertaining to module integration and performance.

 A warning symbol indicates actions that could negatively impact or damage the module.

2 Cellular technology overview

The LENA-R8 series comprises multi-band and multi-mode modules supporting LTE Cat 1bis with 2G GSM/GPRS fallback, providing a low-cost solution for global and multi-regional coverage.

2.1 LTE Cat 1bis

LTE Cat 1 is the lowest-cost LTE category that has the required speeds to support data streaming and full mobility. It represents the best migration path for legacy 2G and 3G cellular technologies.

LTE Cat 1bis has been developed from LTE Cat 1, aiming to create an even simpler and lower-cost 4G solution with worldwide coverage. Its main characteristics are:

- Same throughput and capabilities of LTE Cat 1.
- Only one RX antenna required as it does not implement RX diversity.
- Lower total solution cost due to simpler hardware.
- RX sensitivity 3 dB lower because of the single RX antenna. This means that LTE Cat 1bis does not ensure the same coverage capabilities at the edge of the cell compared to LTE Cat 1.
- LTE Cat 1bis is not standardized worldwide.

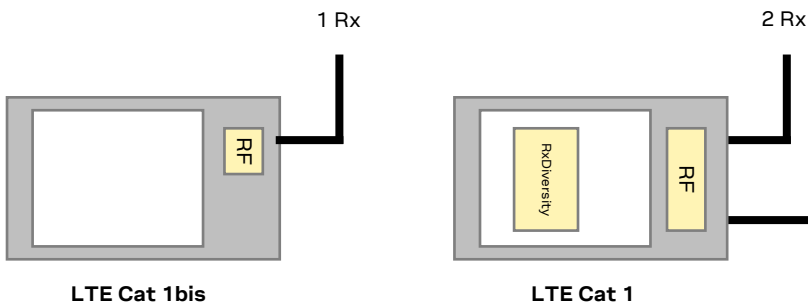


Figure 1: Comparison between LTE Cat 1 bis and LTE Cat 1

3 Application design and development

When designing a host application interfacing with a u-blox cellular module, consider points depicted in [Figure 2](#):

- Choose the module's features that the application needs and the ones that can be disabled.
- Split the application workflow into stages.
- Design the application to work in several modes, reflecting the lifecycle of the product.

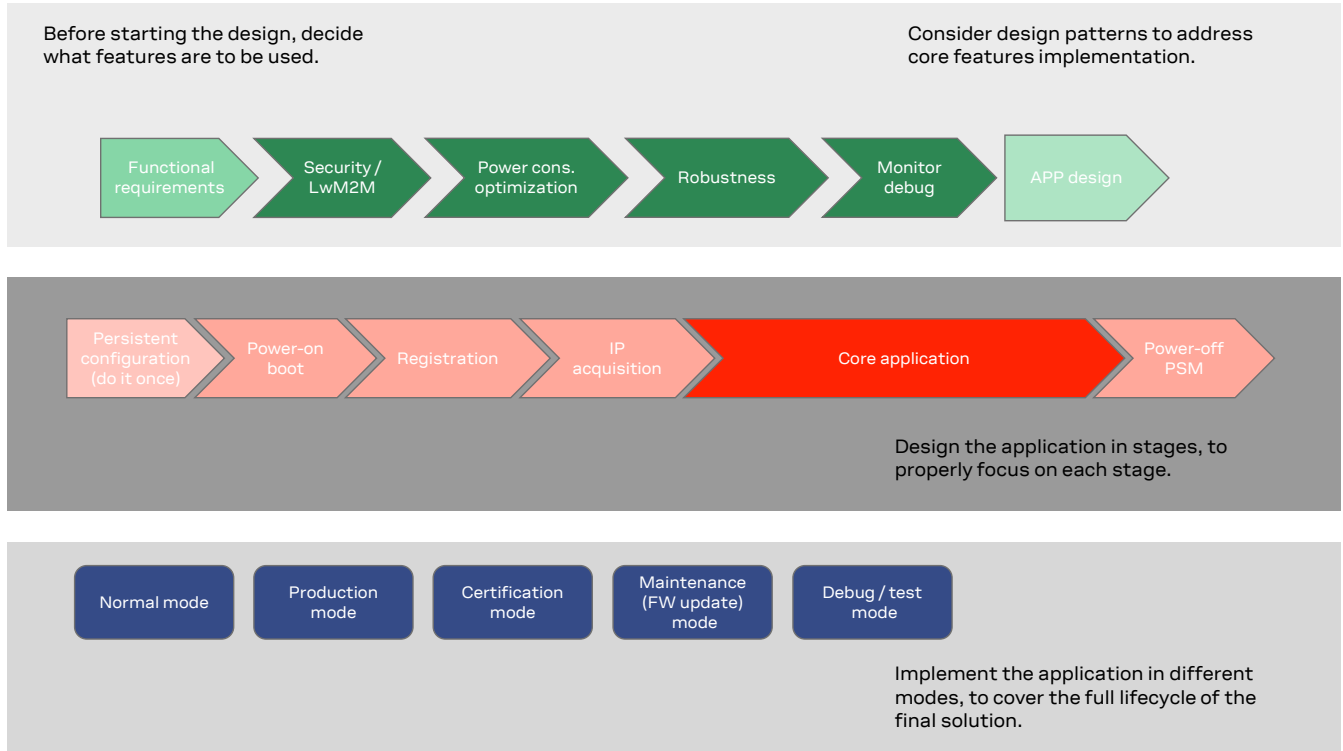


Figure 2: Application design guidelines

3.1 Initial design decisions

If some features will never be used during the application lifetime, they should be disabled or properly configured to minimize their impact on the overall performance. These decisions should be carefully taken at design stage because any later change could cause much effort to adapt and validate the application.

These decisions are:

- Use of power saving
- SW/HW monitor and debug solutions

Power saving features (+UPSVM) shall be configured based on the target power consumption profile.

Debug-ability can be provided via test endpoints, access to the USB interface, and in general a prolific application log containing all AT commands strings exchanged with the module and diagnostic information sent by the module with timing information. Monitoring the module status is a requirement to implement, via HW or SW, recovery procedures as described in [section 8](#).

3.2 Application stages

3.2.1 Persistent configurations

Some module settings are persistent, that is, they are stored in the module non-volatile memory (NVM). Among these APN for Internet connectivity, radio access technology (RAT), and active LTE bands.

The host application shall implement a persistent configuration setting phase, performed once and then at an as-needed basis, where all AT commands related to the required settings are issued.

The summary table on the top of each command section in LENA-R8 series AT commands manual [3] describe if and where the commands settings are stored.

+CCLK						
Modules	All products					
Attributes	Syntax	PIN required	Settings saved	Can be aborted	Response time	Error reference
	partial	No	NVM	No	-	+CME Error

Figure 3: Example of summary table in AT manual

3.2.2 Power on/boot

In general, at each boot the application shall read configurations and make sure they are correct. If not, persistent configurations can be reapplied.

Not all the module configurations are persistent. Therefore, the application, after each module boot, should again set these volatile configurations, e.g., AT+CMEE=2, AT+CMUX=0.

It is very important that the application has a robust mechanism to detect when the module is ready to communicate via AT commands at the power-on. One solution is detecting the greeting message: in LENA-R8 series it is enabled by default and set to "+UUSTATUS: READY".

The application must configure module time. Automatic update of local time with the network time information is the factory-programmed setting (+CTZU: 1), so after LTE attach or GSM registration, the time of the module is updated.

3.2.3 Network registration

For details about the network registration stage, see section 7.1.

3.2.4 IP acquisition

For some details and guidelines on this topic, see section 9.

3.2.5 Core application

For some details and guidelines on this topic, see section 9 and LENA-R8 series AT commands manual [3]. The application shall handle AT commands, responses and unsolicited indications as suggested in section 4. For diagnostic purposes, the application should rely on status AT commands, see section 8. For robustness purposes, the application should implement embedded watchdog procedures, see section 8.28.2.

3.2.6 Power-off

The application might need to switch off the module. Both normal and emergency shutdown are described in the LENA-R8 series system integration manual [2].

3.3 Application modes

An application is usually designed based on the main use cases in actual scenarios, i.e., in “normal mode”.

In addition, the designer shall provide a way to configure the application for more specific contexts, which can have different requirements with respect to normal mode and can help to perform other important steps in the product lifecycle.


3.3.1 Debug / test mode

In general, an application shall always output a log, including the AT commands it issues and their responses, and implement monitoring strategies as described in section 8.

If a problem occurs related to the cellular module and more information is needed, it may be necessary to configure different verbosity levels for the host application and modules log.

In extreme cases, it may be necessary to provide an AT interface passthrough to allow access to diagnostic AT commands.

If the cellular communication is tested against a network simulator, use a suitable test SIM card (usually provided by the network simulator manufacturer). Real SIM card are not suitable for this kind of tests in LTE RAT since authentication and integrity check on module side cannot be disabled on LENA-R8 series modules.

 The +UDCONF=81 AT command is not supported in LENA-R8 series modules.

3.3.2 Production testing

This mode is to be used during the production tests of the end device.

In this scenario the main application is usually inactive, and AT commands can be used to properly configure the module and use its end user testing features. For more details, contact technical support production for LENA-R8 Production & prototype validation guidelines application note [5].

3.3.3 Certification mode

Depending on the certification, such as regulatory, conformance or for MNO type approval, the application might be disabled, and the module externally controlled, e.g., for throughput testing. Specific MNO tests might require the application to be running in normal mode, e.g., remote SIM provisioning, FOTA.

3.3.4 Firmware update mode

A module's firmware update procedure should be implemented, when necessary, either over the air or tethered.

Each update strategy has its requirements and correct implementation, which must be followed to guarantee the success of the operation. For more details, see section 11.

4 AT commands response parser

This section explains how to develop a proper AT parser and how to handle the AT command replies and the URCs.

In this document the following naming conventions are used:

- DCE (Data Communications Equipment) or MT (Mobile Terminal) is the u-blox cellular module
- DTE (Data Terminal Equipment) or TE (Terminal Equipment) is the terminal that sends the command to the module

4.1 AT command response description

A generic AT command execution consists of several steps, showed in the below picture:

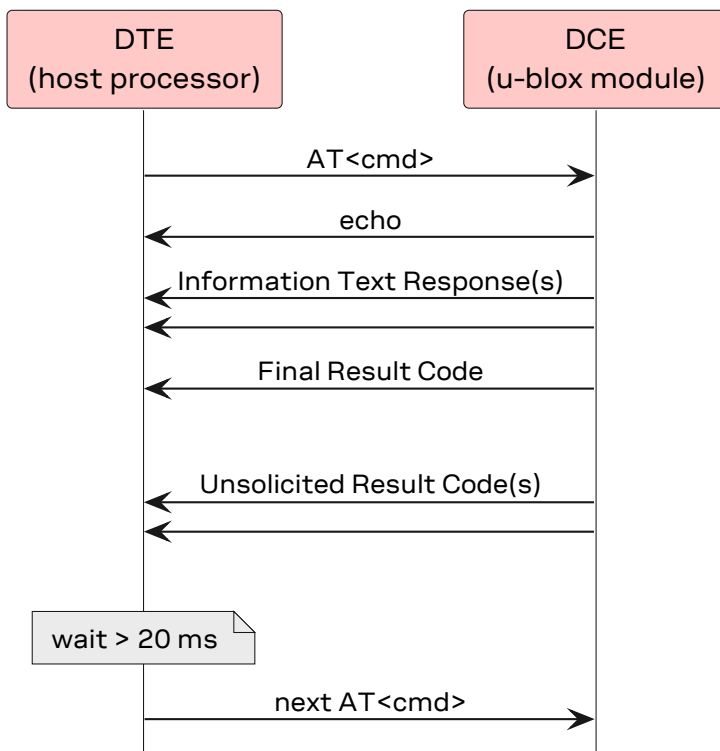


Figure 4: Sequence diagram of a generic AT command

4.1.1 Echo

The echo is a copy of the message received by modem serial interface layer. It is immediately returned to the host, typically in less than few milliseconds.

Echo is enabled by default and can be disabled with ATE command. Typically, it is not useful for a host application, and it is disabled.

4.1.2 Information text response

The information text response is an intermediate message that for some commands provides additional information. Timings in its issuing are very varied and depend on the actions triggered by the command: the longer occur if interactions with the network are required.

4.1.3 Final result code

The final result code is the message issued by the module to provide the final feedback of command execution and confirm the unlock of AT interface.

As long as the final result code is not issued, the module is busy in command execution. It is still capable of receiving new AT command, but one of the two will be discarded depending on the “can be aborted” feature of the first command:

- If the first command is not abortable, the new one will be ignored without providing any feedback to the host processor,
- If the first command can be aborted, it will be dropped, and as LENA-R8 series modules do not support ABORTED final result code, no ABORTED response will be issued, and the module will start the execution of the latest one.

4.1.4 Unsolicited Result Code

An unsolicited result code (URC) is a string message (provided by the DCE) that asynchronously indicates the occurrence of an event that might be related to a previous AT command or to the feature the user is currently using, or to the module’s autonomous activity (for example, due to mobility).

URCs are not issued during command execution (i.e., before the final result code has been provided) or when the module is in data command mode. Typically, URCs are deferred and printed when the AT port returns into command mode; for some classes of URCs different behaviors can be set; see +CMER (mobile termination events), +CNMI (SMS events), +CGEREP (PS events) in AT command manual [3].

4.1.5 Response time

The response time reported, in LENA-R8 series AT commands manual [3], in the summary table (see Figure 3) at the beginning of each command section, represents the maximum time that can be expected in receiving the final result code. It is time that the modem requires to provide a response in the worst scenario, when any step in the procedure to execute the required task fails and repetitions or alternative steps have to be done as per technical specifications. It is usually much higher than the time typically needed to receive a response.

The most representative examples of difference between max response time and typical response time can be found in network commands (e.g., +COPS). Network procedures consider a lot of cases of possible errors, missing answers, etc. In all of these cases, a series of new attempts is defined by technical specifications and this cause a possible delay in accomplish the sub-task.

Thus, in the worst case, when all the attempts to complete each sub-tasks of a procedure fail, the time to receive the final response (likely an error) can be even tens of times longer than the typical time to receive the success answer in an usual scenario.

If the response time for a command is left blank (actually "-"), the response is "immediate": it is issued by the module in less than a few tens of ms. This occurs for those commands that are executed without asynchronous requests to the protocol stack or the internal applications, which usually require time to be answered.

Additional delays to the declared response time have to be considered when using simultaneously different AT ports. In this case an AT command may require a resource that is serving a command received on a different AT port.

Additional delays have to be evaluated, as well, if the command reception and execution involve intermediate communication protocols, e.g., MUX, in this case the communication between host processor and modem are transferred via a specific protocol running on the physical port, that might introduce additional communication delay due to framing and re-transmissions.

4.2 Guidelines for robust AT parser

4.2.1 Basic recommendations

- Always wait for at least 20 ms following a final result code or a URC reception before issuing a new AT command (see [Figure 4](#)). This delay allows the module to issue possible buffered URCs.
- When the module has finished processing an AT command, it will output a final result code (either OK or ERROR) indicating that it is ready to accept a new AT command. The information text responses are issued before the final result code. Change the +CMEE AT command setting to numeric or verbose value (e.g., AT+CMEE=1 or AT+CMEE=2).
- Asynchronous commands, e.g., +UFTPC, return an immediate final result code (IRC) and final result via URC.
- Some AT commands, e.g., +CCLK, return an IRC during command execution.
- Handle the unexpected spaces or line endings. Unexpected 0x20 characters among parameters of the same command response shall be ignored. The same for 0x0D or 0x0A chars between different responses.

4.2.2 Multiple AT interfaces

AT commands executed on different AT capable interfaces are serialized and then executed by the internal AT parser in the arrival order. parallel AT commands execution is not possible.

This behavior can be detected in the following configurations:

- AT commands executed on one serial-over-USB interface and one UART interface
- AT commands executed on two different serial-over-USB interfaces
- AT commands executed on two different MUX virtual channels over UART interface

The only exception to the above rules is when it is defined a system architecture with a PPP dial-up on one port and AT commands on the other one.

The tasks using different AT interfaces have to be aligned and thus avoid triggering actions that could be in collision. For example, avoid triggering a network scan on one AT interface and forcing a network deregistration on another one.

4.3 Operational modes of the AT interface

When implementing the AT parser, it is important to consider that the communication port, whether a virtual serial ports over USB interface, the main UART or a MUX virtual channels, enters different operational modes while processing AT commands.

In command mode, the module (called DCE – data communication equipment) can receive AT commands. Once an AT command is detected on the AT interface, the DCE processes it and may return to command mode by issuing a success or error response.

Special AT commands lead the AT interface into intermediate states where, for example, an SMS payload is expected, or raw/binary data is exchanged (for example, during file transfer), or PPP packets are exchanged. In the latter case, the PPP data mode can be temporarily exited by a special +++ packet or DTR line ON-to-OFF transition and the online command mode (OLCM) state is entered: from this state, which is like the command mode, the DCE can be moved back to PPP data mode via ATO command or can disconnect PPP via ATH command (having previously applied AT+CVHU=0).

[Figure 5](#) depicts the various modes in which the module can operate and shows the actions that cause transitions between the different modes. The transitions triggered by DTR line changes are configurable with the AT&D command, see LENA-R8 series AT commands manual [\[3\]](#).

For more details about the AT command interface settings, see the AT command settings section in the LENA-R8 series AT commands manual [3]

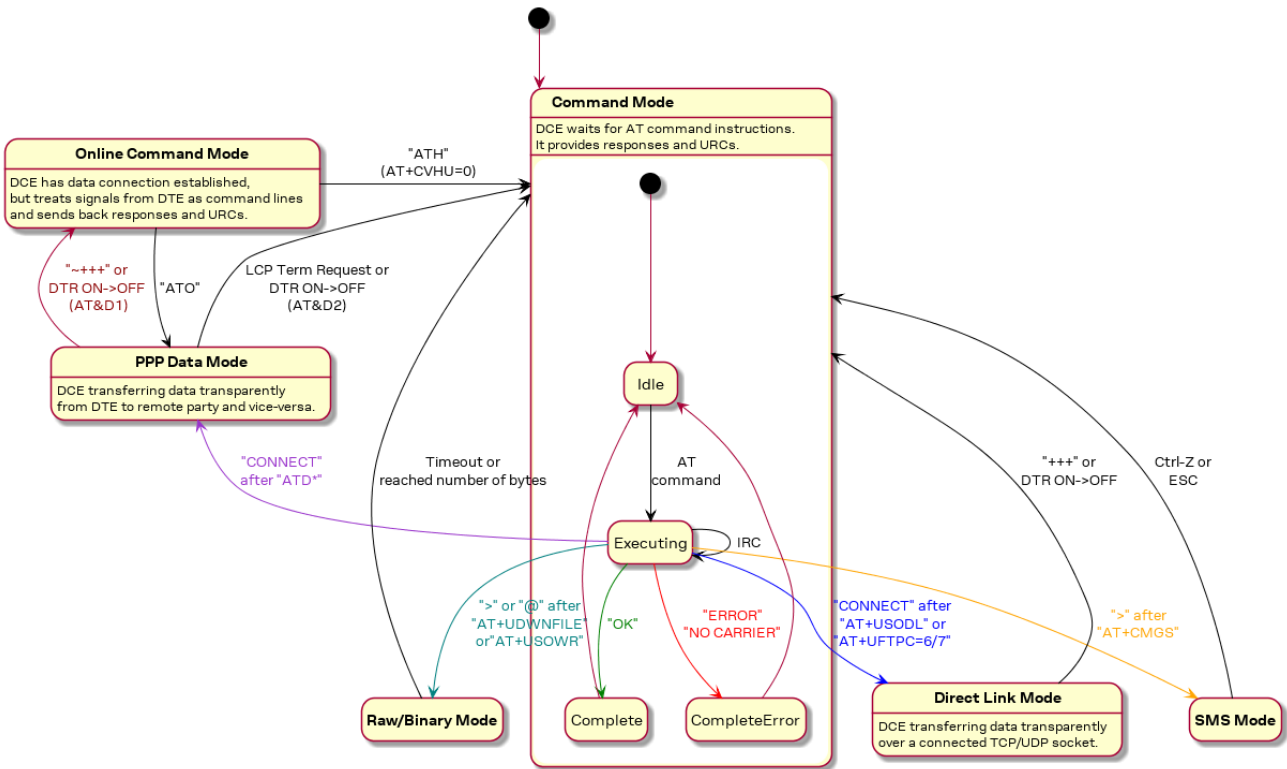


Figure 5: Module operating modes and actions causing mode transitions

5 Local connectivity

5.1 Serial interface configuration

UART and USB interfaces are available simultaneously. UART interface supports AT command interface and PPP data interface. USB interface supports different configurations, that can be selected by using the +SYSNV AT command. Available configurations are:

- serials ports only (default)
- RNDIS and serials ports
- ECM and serials ports

The above configurations are available if the modem input pin **USB_BOOT** (pin 33) is open or shorted to GND. A fourth configuration is enabled if **USB_BOOT** pin is set to high level before boot:

- FW update

See section 11 for further details.

5.2 USB mode

5.2.1 USB Serial ports only

This is the default configuration: it shows eight USB serial ports.

USB identifiers are:

- VID: 0x1782
- PID: 0x4D10

USB port name	Description	notes
Unisoc Usb Serial Port 0	AT interface and PPP data communication	PPP can be activated on only one port per time
Unisoc Usb Serial Port 1	Reserved	
Unisoc Usb Serial Port 2	Reserved	
Unisoc Usb Serial Port 3	Advanced trace log (CP trace)	Captured by ArmTracer tool
Unisoc Usb Serial Port 4	Standard Trace Log (AP trace)	Captured by CoolWatcher tool
Unisoc Usb Serial Port 5	Reserved	
Unisoc Usb Serial Port 6	AT interface and PPP data communication	PPP can be activated on only one port per time
Unisoc Usb Serial Port 7	AT interface and PPP data communication	PPP can be activated on only one port per time

Table 2: Ports in USB serial ports only mode

5.2.1.1 Activation procedure

Command	Response	Description
AT+SYSNV=1, "usbmode", 2	OK	Set USB port in "serial ports only" mode. The configuration will be effective after the next reboot.
AT+CFUN=16	OK	Reboot the module.

Table 3: USB serial ports only mode activation example

5.2.2 RNDIS and serials

When this configuration is applied, the module presents eight serial ports and a RNDIS network adapter.

USB identifiers are:

- VID: 0x1782
- PID: 0x4D11

Device / USB port name	Description	notes
Remote NDIS based Internet Sharing Device	RNDIS interface	
Unisoc Usb Serial Port 0	AT interface and PPP data communication	PPP can be activated on only one port per time
Unisoc Usb Serial Port 1	Reserved	
Unisoc Usb Serial Port 2	Reserved	
Unisoc Usb Serial Port 3	Advanced trace log (CP trace)	Captured by ArmTracer tool
Unisoc Usb Serial Port 4	Standard Trace Log (AP trace)	Captured by CoolWatcher tool
Unisoc Usb Serial Port 5	Reserved	
Unisoc Usb Serial Port 6	AT interface and PPP data communication	PPP can be activated on only one port per time
Unisoc Usb Serial Port 7	AT interface and PPP data communication	PPP can be activated on only one port per time

Table 4: Ports in USB RNDIS mode

5.2.2.1 Activation procedure

Command	Response	Description
AT+SYSNV=1, "usbmode", 3	OK	Set USB port in "RNDIS + serial ports only" mode. The configuration will be effective after the next reboot.
AT+CFUN=16	OK	Reboots the module.

Table 5: USB RNDIS activation example

5.2.3 ECM and serials

When this configuration is applied, the module presents eight serial ports and an ECM network adapter.

USB identifiers are:

- VID: 0x1782
- PID: 0x4D13

Device / USB port name	Description	notes
CDC Ethernet Control Mode (ECM)	ECM interface	
Unisoc Usb Serial Port 0	AT interface and PPP data communication	PPP can be activated on only one port per time
Unisoc Usb Serial Port 1	Reserved	
Unisoc Usb Serial Port 2	Reserved	
Unisoc Usb Serial Port 3	Advanced trace log (CP trace)	Captured by ArmTracer tool
Unisoc Usb Serial Port 4	Standard Trace Log (AP trace)	Captured by CoolWatcher tool
Unisoc Usb Serial Port 5	Reserved	

Device / USB port name	Description	notes
Unisoc Usb Serial Port 6	AT interface and PPP data communication	PPP can be activated on only one port per time
Unisoc Usb Serial Port 7	AT interface and PPP data communication	PPP can be activated on only one port per time

Table 6: USB ECM ports

5.2.3.1 Activation procedure

Command	Response	Description
AT+SYSNV=1, "usbmode", 5	OK	Set USB port in "ECM + serial ports only" mode. The configuration will be effective after the next reboot.
AT+CFUN=16	OK	Reboots the module.

Table 7: USB ECM activation example

5.2.4 USB firmware update mode

If **USB_BOOT** input pin is set to high level before the boot, the modem will run in USB firmware update mode: it remains inactive and just wait for a new firmware download via USB port.

For details of **USB_BOOT** pin, see LENA-R8 series system integration manual [2]. For more details of FW update, see section 11.

USB identifiers are:

- VID = 0x0525
- PID = 0xA4A7

Device / USB port name	Description
SPRD U2S Diag	FW update mode

Table 8: USB FW update port

5.3 AT interface on UART

By default, UART rate is set to "autobauding" (+IPR: 0). The modem uses the first AT command to detect the current baud rate. This command will be not executed and will not provide neither echo nor response.

u-blox solution to handle baudrate detection consists of sending a first dummy "AT" without expecting any response. After having waited for 100 ms, AT interface is ready.

In autobauding mode, the greeting message is issued at 115,200 bit/s.

5.3.1 Set a fixed baud rate

Use the +IPR AT command to set a different baud rate for the UART interface in which the command is executed, as shown in following table:

Command	Response	Description
AT+IPR=460800	OK	Set UART speed to fixed value 460,800 bit/s. Cellular module response is sent at the original baud rate value.
After the "OK" final result code, wait for at least 100 ms before issuing a new AT command, to guarantee proper baud rate reconfiguration.		
The new selected baud rate is immediately applied, a device reboot is not needed.		
AT+IPR?	+IPR: 460800	Check the current baud rate.

Command	Response	Description
	OK	

Table 9: Baudrate setting example

5.4 AT interface on USB

AT commands can be issued to the module via the AT enabled ports on the USB interface.

- The USB interface is enabled only if an external voltage detectable as high logic level is applied at the **VUSB_DET** input pin during the switch-on boot sequence of the module.
- AT interface on USB is disabled if input pin **USB_BOOT** (pin 33) is set to high level before boot. See section 11 for further details.

5.5 USB network modes

- Both ECM and RNDIS support bridge and router mode.

5.5.1 Bridge mode

In bridge mode, LENA-R8 acts as a bridge device between the mobile network and the DTE: the IP termination of the data connection is on the DTE network subsystem. For the activated PDP context or EPS bearer, the DTE assigns (i.e., “binds”) the IP address to its USB virtual Ethernet interface and configures its routing rules. Each IP address associated with an active PDP context/EPS bearer is granted to the target by the mobile network, and it shall be retrieved through appropriate AT commands. In bridge mode the PDP context/EPS bearer sets up a bridge between the cellular network and the USB interface: this is defined as bridge PDP context/EPS bearer.

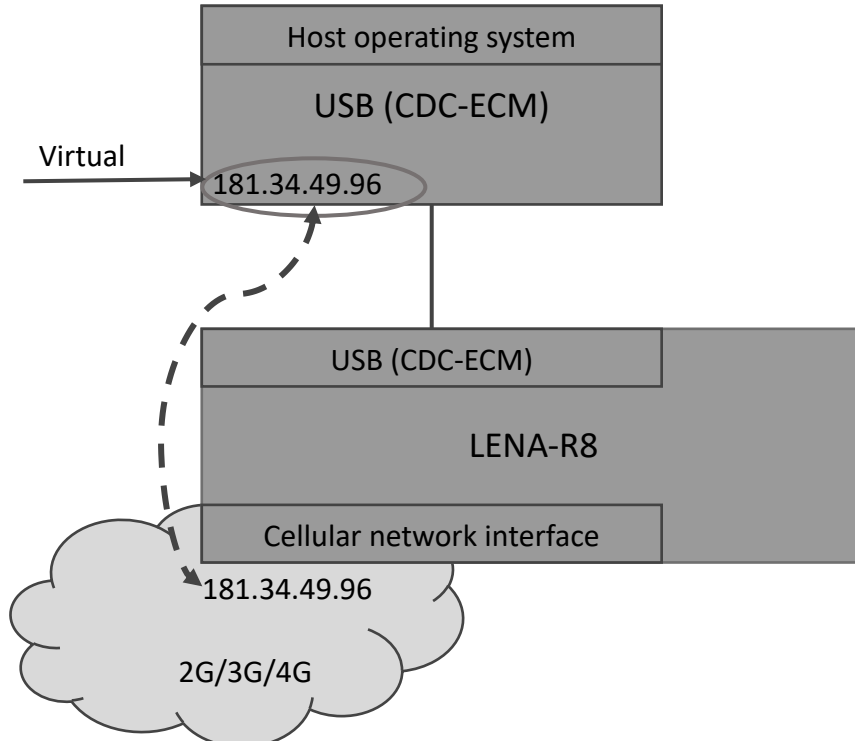


Figure 6: LENA-R8 that acts as a bridge, LENA-R8 network IP address is assigned to DTE

5.5.1.1 Set the module to bridge mode

Command	Response	Description
AT+SYSNV=0,"nat_cfg"	+SYSNV: "nat_cfg", 255 OK	Retrieve the ECM module router/bridge mode configuration
AT+SYSNV=1,"nat_cfg", 0	OK	Set the module to bridge mode configuration
AT+CFUN=16	OK	Reboot the module

Table 10: Bridge mode activation example

5.5.2 Router mode

In the router mode, the IP termination is on the LENA-R8: the module acts as a mobile network router and it can share its data connectivity through a private network over the USB interface. In router mode, the PDP context/EPS bearer are connected to the module IP subsystem: this is defined as router PDP context/EPS bearer.

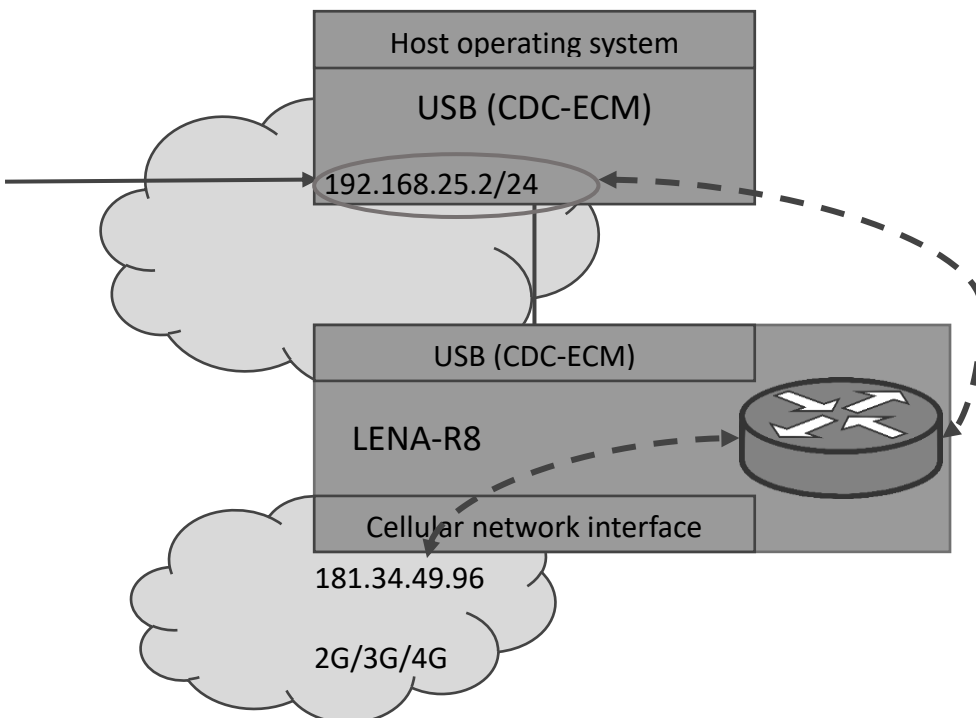


Figure 6: LENA-R8 that acts as a router, LENA-R8 assign to DTE a private IP address

5.5.2.1 Set the module to router mode

Command	Response	Description
AT+SYSNV=0,"nat_cfg"	+SYSNV: "nat_cfg", 0 OK	Retrieve the ECM module router/bridge mode configuration
AT+SYSNV=1,"nat_cfg", 255	OK	Set the module to router mode configuration
AT+CFUN=16	OK	Reboot the module

Table 11: Router mode activation example

5.6 Power saving

The power saving configuration is disabled by default, but it can be enabled and configured using the +UPSV AT command. When the power saving is enabled, the module automatically enters the low power idle mode whenever possible, reducing current consumption. If the module is registered or

attached to a network, the power saving periods are interleaved by wake-up phases in which the module monitors the paging channels, according to 2G/LTE system requirements.

See LENA-R8 AT commands manual [3] for details about +UPSV AT command.

5.6.1 USB interface

The suspend / resume and remote wake up functions are supported by USB interface despite the +UPSV value configured.

5.6.2 UART interface

The power saving mode can be configured in the modes below:

- Disabled (+UPSV: 0):
 - UART interface is always enabled and the module does not enter in power save mode.
- Controlled by UART **RTS** line (+UPSV: 2):
 - If the RTS line state is set to OFF, the low power idle mode is allowed
 - If the RTS line state is set to ON, the module shall exit from low power idle mode
- Controlled by UART **DTR** line (+UPSV: 3):
 - If the DTR line state is set to OFF, the low power idle mode is allowed
 - If the DTR line state is set to ON, the module shall exit from low power idle mode
- Controlled by data received over the UART interface (+UPSV: 4):
 - The module enters low power idle mode whenever possible and it wakes up upon data received over the UART interface. It remains in active mode until the expiration of the defined <Timeout>, and then enters back the low power idle mode.

5.7 Multiplexer (MUX)

LENA-R8 series modules support the multiplexer functionality on the UART physical link as defined in the 3GPP TS 27.010 [7]. This makes it possible to have multiple simultaneous sessions (virtual channels) over the single UART interface.

The following virtual channels are defined:

- Channel 0: multiplexer control.
- Channels 1 to 3: AT commands / data connection.

Multiplexer mode is activated/configured by +CMUX command.

Since its setting are volatile, if rebooted, the modem will start in normal mode: the possible greeting message will be issued on the physical UART and it will be able to receive only on this channel; MUX frame commands will not be decoded.

To improve the robustness of the host application, in case of missing answer to AT commands in MUX mode, the host should try to send an AT command in normal mode in order verify if the modem exited the MUX mode and, if so, restart it.


See IPC – Inter Processor Communication section in LENA-R8 series AT commands manual [3] for further details.


5.8 Point-to-point protocol (PPP)

The module can perform dial-up network (DUN) connections supporting the Point-to-Point Protocol (PPP). The PPP connection is established between the host (e.g., Windows device) and the DCE.

When a data call is initiated by the D* or +CGDATA="PPP",1 AT commands, the module switches to the PPP mode just after the CONNECT intermediate result code.

If a PDN connection is not active on the specific cid, it will be activated.

-  If the network throughput is less than the data sent from the host to the module (which is limited by the radio resources assigned by the network to the transmission in the uplink), then packet data loss may occur, even with hardware flow control enabled. To avoid this issue, do either or both:
 - Reduce the baud rate used on the serial COM port.
 - Slow down data transfer load by adding pauses between data payloads or breaking up their payload and adding delay.

-  LENA-R8 employs a local PPP in which the IP address assigned to the host is private.

5.8.1 Basic setup

The module must be attached to the network and the APN must be properly configured into the PDP context before starting the dial-up.

Command	Response	Description
AT+CGDCONT=1,"IP","internet.wind.biz"	OK	Configure APN for cid 1
AT+CGACT=1,1	OK	Activate PDP context on cid 1
ATD*99***1#	CONNECT	Setup a data communication channel on cid 1
data data data		

Table 12: PPP activation example

5.8.2 Swap between Data mode and OnLine Command Mode

The CONNECT response confirms the module entered the data mode and a transparent communication is established between DCE and remote party.

To exit this mode and return to online command mode, and to send AT commands to the module, there are two options:


- Send escape sequence "+++" on serial link. This command must be sent at least for 2 s later than the last data has been sent or received.
- Toggle DTR from ON to OFF status. This behavior depends on &D settings: see LENA-R8 series AT commands manual [\[3\]](#) for further details.

To return to the data mode, use the ATO commands. To close the connection, use the ATH command.

Command	Response	Description
AT+CGDCONT=1,"IP","internet.wind.biz"	OK	Configure APN for cid 1
AT+CGACT=1,1	OK	Activate PDP context on cid 1
AT+CGDATA="PPP",1	CONNECT	Setup a PPP data communication channel on cid 1
data data data		
		Wait at least for 2s after the last data is sent/received before

Command	Response	Description
		issuing the escape sequence "+++"
+++	OK	Escape sequence to exit the transparent communication (PPP mode). OK response confirms the module returned to online command mode
ATO	CONNECT	Resume a PPP data communication channel
data data data		Wait at least for 2 s
+++	OK	Escape sequence
ATH	OK	Terminate the connection

Table 13: Swap between Data and Online command modes


 An explicit activation of PDP context (AT+CGACT=1,<cid>) is needed to guarantee a correct management of data communication channel resuming (ATO).

6 User settings persistence

6.1 Save user settings

When an AT command is executed in set mode, if the user setting should be persistent between power cycles, the value is automatically stored in NVM, instead of a specific personal profile. No other action is required to the user.

Check LENA-R8 series AT commands manual [3] for further details in appendix to identify the AT commands with persistent settings in NVM and their factory programmed value.

 To avoid flash memory wearing, it is strongly recommended to read the required user setting value and then, if necessary, to save the new user setting, instead of setting the new value directly.

6.2 Restore factory configuration

During the module operation, different files may be stored in the module's file system. Similarly, the NVM is populated with user configuration as well as auxiliary information stored by the module to optimize its operations (e.g., information on the cellular environment).

The host application can restore the module factory configuration via the +UFACTORY AT command. This can be required to recover from an unexpected behavior and restart the module in a controlled configuration, or during the MNO certifications, where the device is tested in various simulated scenarios and the auxiliary information or previous user settings can affect the tests outcome.

Restoring the factory configuration of the module is a two-step process:

1. Set the type of restore to perform, using the +UFACTORY AT command. FS and/or NVM can be restored.
2. Reboot the module.

The +UFACTORY AT command writes a flag in NVM and does not perform any restoring action. This flag is then read at the next reboot, when the +UFACTORY corresponding action is executed. Therefore, it is possible to cancel the +UFACTORY action by issuing the AT+UFACTORY=0,0 command before the reboot.

Table 14 shows the procedure to restore the factory settings.

 The +UFACTORY AT command does not delete SMS stored in the module

Command	Response	Description
Delete all settings and files stored		
AT+UFACTORY=2,1	OK	Set +UFACTORY to delete all files previously stored in the file system and all NVM sectors. No restore is performed so far.
AT+UFACTORY?	+UFACTORY: 2,1 OK	Check the restore action currently set.
AT+CFUN=16	OK	Reboot the module. At next reboot, the restore action previously set is applied.

Table 14: +UFACTORY example

7 Cellular modem services

7.1 Network registration

A LENA-R8 series module by default automatically initiates the registration procedure at power-on. It reads the SIM for the PLMN and starts a network search. Once it finds a suitable cell it can camp on, the registration process starts.

In LTE RAT, an initial default EPS bearer is established during the LTE registration and the UE will be assigned of an IP address. Each network offers connectivity to different IP domains (Internet or private Intranet) which are selected by specifying the APN (Access Point Name) at PDP context activation. APN can be assigned by the network and therefore the initial default bearer; in case of private intranets the assigned APN could not be the expected one and therefore, the user is recommended to double check and set the proper anchor APN as described in section 7.2.1.1.

2G registration is independent from APN and bearer activation. PDP contexts are established on demand, typically by the user, when needed and are not required for registration on the network.

7.1.1 Band selection

Some applications require the used bands are limited to a subset. LENA-R8 supports two separate AT commands to control the bands for the two supported RATs.

- +SETBAND for the 2G bands
- +SETLOCK for the LTE bands

Both commands permit to set and read the band configuration.

The set of a new configuration will be effective only after module reboot. See LENA-R8 series AT command manual [3] for further details.

7.2 Network attach and PDN connections

7.2.1 Initial default bearer

Initial default bearer is mapped on CID 0 and automatically activated by the network. This bearer is reserved for MNO purposes, thus, the user or host application shall configure and explicitly activate additional bearers for their tasks, see section 7.2.2.

Command	Response	Description
AT+CGDCONT?	+CGDCONT: 0,"IP","default-lte", "10.10.243.167" OK	Show the APN assigned and IP address assigned by the network. If no IP is shown, the module is not attached in LTE RAT.
AT+CGACT?	+CGACT: 0,1 OK	Show the <status> = 1 for cid 0 if initial default bearer is active. If not +CGACT will not provide any intermediate response.
AT+COPS?	+COPS: 0,2, "22288",7 OK	Return the MNO id or name where the module is registered on

Table 15: Initial default bearer checking example

7.2.1.1 Change anchor APN

Anchor APN is the APN used for initial default bearer; the EPS bearer automatically activated by the network for LTE registration. Typically, it is configured by the network, but in some cases user or host application could do it, especially when registration in private Intranet is required.

Command	Response	Description
AT+CFUN=0	OK	Detach from the network to change the initial default bearer.
AT+CFGDFTPDN=1,0,"internet.wind.biz"	OK	Set specific or private APN name if needed. Changes shall be applied while the modem is in de-registered state.
AT+CFUN=1	OK	Reattach to network.

Table 16: Anchor APN configuration example

7.2.2 Additional bearers

User applications shall use additional PDP contexts since they are not allowed to use the initial default bearer CID 1. Therefore, if internal sockets or internet clients (HTTP, FTP, PING, MQTT, ...) are required, one or more additional PDP contexts shall be configured and activated.

It is impossible to specify the context ID (<cid>) to use for the internet application if more than one context is active. The first activated context is always used for the data traffic of the embedded internal application. Only DNS and PING functionality are not following this rule and use the last activated context.

Command	Response	Description
AT+CGDCONT?	+CGDCONT: 0,"IP","default-lte","10.10.243.167" OK	Check current settings and active PDP contexts.
AT+CGDCONT=1,"IP","internet.wind.biz"	OK	Configure a PDP context on CID 1
AT+CGDCONT?	+CGDCONT: 0,"IP","default-lte","10.10.170.21" +CGDCONT: 1,"IP","internet.wind.biz","IPv4:0.0.0.0",0,0 OK	Initial default bearer on CID 0 is configured and active. Additional bearer on CID 1 is configured but not active yet (there is not IP address assigned)
AT+CGACT=1,1	OK	Activate PDP context on CID 1
AT+CGDCONT?	+CGDCONT: 0,"IP","default-lte","10.10.170.21" +CGDCONT: 1,"IP","internet.wind.biz","10.15.164.174",0,0 OK	Both PDP context are active
AT+CGDCONT=2,"IP","internet.wind"	OK	
AT+CGACT=1,2	OK	Activate PDP context on CID 2
AT+CGDCONT?;+CGACT?	+CGDCONT: 0,"IP","default-lte","10.10.170.21" +CGDCONT: 1,"IP","internet.wind.biz","10.15.164.174",0,0 +CGDCONT: 2,"IP","internet.wind","10.21.139.211",0,0 +CGACT: 1,1 +CGACT: 2,1 OK	
AT+CGACT=0,2	OK	Deactivate PDP context on CID 2

Table 17: Additional bearers management example

7.3 Voice and data services

LENA-R8 module do not support voice calls since the audio interface is not implemented. However, voice calls capabilities are supported from the network point of view: the device is capable to send and receive voice calls, but no audio can be sent or received.

In LTE, voice calls are supported via the IP-based IMS protocol (VoLTE). LENA-R8 reserves CID 11 for IMS contexts.

In 2G, voice calls work instead in Circuit Switched, a service implemented in the network completely independent from data transfer (Packet Switched service). CS calls and PS data transfer cannot run in parallel, due to GSM/GPRS technology limitations. Data is suspended while speech calls are performed. SMS can be routed both via CS and PS services, but this is managed by the network.

7.3.1 Voice calls

Command	Response	Description
AT+CLIP=1	OK	Enable Calling line identification presentation
	RING +CLIP: "+390402529400",145,,,,0 RING RING	"RING" URCs notify an incoming call. After the 1 st RING, the +CLIP URC provides the caller number
ATA	CONNECT	Answer the call. "CONNECT" confirms that voice call is active, but no audio can be listened or sent since local audio interface is not implemented.
ATH	OK NO CARRIER	Terminate the call

Table 18: Voice call example

7.4 Mobility scenarios

Devices used in non-static installations can move out of range of the currently serving cell. This means entering another cell of the RPLMN (registered PLMN) or of a different PLMN or moving into an area where there is no cellular coverage or no roaming agreement for the device.

If the module loses the synchronization with the serving cell but finds another cell to camp on, any PDP context and open sockets will be kept. This holds in particular for seamless change of serving cell with cell reselection or handover procedures, even changing RAT.

If mobility implies crossing national borders or simply changing the PLMN, it is likely that the new PLMN will force the module to reattach; if the selected cell belongs to a legacy RAT, it is necessary to re-establish the PDP context in order to restore the services requiring cellular connectivity.

If in roaming conditions there is a PLMN with higher priority than the currently selected PLMN, the module periodically performs a PLMN scan when in RRC idle state.

8 Monitoring module status

8.1 Retrieve and interpret diagnostic information

It is recommended to track the module status in the host application. Such diagnostic information allows detecting specific scenarios and implementing proper handling in the host application.

The module status can be returned by AT command responses and unsolicited result codes (URCs). Depending on the host application architecture, URCs, periodic polling, or both, can be used. URCs provide the most updated information, some of which is not available via polling. For AT commands that enable URCs, they might also return the same information when polled, as indicated in the next paragraphs.

Some commands store the setting of the URC reporting in NVM, so they are referred to as persistent settings.

Some AT commands provide a choice on how to handle the URCs when the AT interface is busy. For all other AT commands, URC is issued at the return into command mode, as explained in LENA-R8 series AT command manual [3], in the “URCs presentation deferring” section.

URCs and AT command responses are presented with their generic syntax because parameters names are explanatory. For precise meaning, see the LENA-R8 series AT command manual [3].

For reference AT command sequences of reading module settings and debugging, see LENA-R8 scripts in u-blox m-center scripts examples on GitHub [11].

8.1.1 Diagnostic information via URCs

Command	URC	Description	Can be polled
AT+CREG=3	+CREG: <stat>[, [<lac>], [<ci>] [, [<ActStatus>], <cause_type>, <reject_cause>]]]	Enable registration status URC for non-EPS services (e.g. SMS) and reject cause in case of unavailability of such services.	Yes
AT+CEREG=5	+CEREG: <stat>[, [<tac>], [<ci>], [<ActT>], [<cause_t ype>], [<reject_cause>], [<Assigned_Active_Time>], [<Assigned_Periodic_TAU>]]]]]	Enable registration status URC for EPS services, reject cause in case of unavailability of such services and PSM related timers if granted by the NW.	Yes
AT+CGEREP=2,1	+CGEV: NW PDN DEACT <cid>	Report all registration and PDN connectivity status events. The first parameter of the AT command is set to 2 indicate that URC will not be discarded in case the AT interface is busy.	No
AT+CTZR=2	+CTZE: <tz>, <dst>[, <time>]	Enable reporting of changes in local time (+CTZU: 1 is the factory default setting, so +CCLK will be automatically updated. Usually time and time zone info is provided by the NW at LTE attach only.	No
AT+CMER=3,0,0,2	+CIEV: <descr>, <value>	Report variations in some indicators like roaming, SIM indication (provided that the feature is configured via +UGPIOC), signal level.	No
AT+CNMI=2,1	+CMTI: <mem>, <index>	Report the index in current selected memory (can be the factory default ME or SIM, see AT+CPSM?) where the Mobile terminated SMS has been stored. The first parameter set to 2 indicates that URC are buffered in case of busy AT interface. The setting is persistent (can be saved in profiles via AT&W).	No

Table 19: Diagnostic information via URCs

8.1.2 Diagnostic information via polling

Command	Response (omitting OK)	Description	Suggested usage
ATI0I9	LENA-R8001M10-00C-00 02.00,A01.22	Return the model and FW identification, useful for tracking purposes.	Module initialization
AT+CGSN	353760970000040	Return the IMEI.	Module initialization
AT+CIMI AT+CCID	001010123456789 +CCID: 89860000502000180722	Return the SIM identities for tracking purposes. They can change after a SIM refresh event triggered by MNO	Module initialization
AT+CEER	+CEER: "EMM cause",15,"No suitable cells in tracking area"	It returns the last error cause (as received by the network) that led to the failure of a registration or PDN connectivity establishment.	When detecting registration or connectivity problems
AT+COPS?	+COPS: <mode>[,<format>,<oper>[,<Act>]]	It returns the current registered PLMN in the configured format, RAT is fixed to 7 (LTE Cat-M1).	Periodically (30 s) and when detecting problems
AT+CGDCONT?	+CGDCONT: <cid>,<PDP_type>,<APN>,<PDP_addr>,<d_comp>,<h_comp>[,<IPv4AddrAlloc>,<request_type>,<P-CSCF_discovery>,<IM_CN_Signalling_Flag_Ind>[,<NSLPI>[,<secure_PCO>[,<IPv4_MTU_discovery>[,<Local_Addr_Ind>]]]]]	It returns the active and defined EPS bearers with the APN used and the IP type and addresses.	Periodically (30 s) and when detecting problems
AT+CPOL?	+CPOL: <index1>,<format>,<oper1>[,<GSM_Act1>,<GSM_Compact_Act1>,<UTRAN_Act1>[,<E-UTRAN_Act>]]	It returns the high priority PLMN list, which might change at SIM refresh.	Module initialization and SIM refresh, only in debug/test mode
AT+CPSMS?	+CPSMS: <mode>,[<Requested_Periodic_RAU>],[<Requested_GPRS_READY_timer>],[<Requested_Periodic_TAU>],[<Requested_Active_Time>]	It returns the PSM settings as required to the NW. As setting is persistent, if the NW does not grant PSM, the module will propose the PSM timers at every attach and registration attempt.	Module initialization

Table 20: Diagnostic information via polling

8.2 Full-stack watchdog: how to react to unexpected conditions

The application shall properly handle communication or connectivity problems when using LENA-R8 in the cellular mobile environment.

When a problem at a specific level is encountered, the corresponding countermeasure shall be tried and, if not resolving the issue, proceed with the solutions for the lower levels. [Table 21](#) shows a full-stack watchdog for monitoring LENA-R8 series modules.

Level	Problem	Countermeasure	AT commands / actions	Notes
Socket/dial-up	Cannot send / receive data	Close and re-open socket	AT+USOCL=<socket_id> AT+USOCR=<protocol> AT+USOCO=...	
		Disconnect and re-connect dial-up	Send +++ / Move DTR ATD*99***<cid>#	
IP/PDP	Cannot get an IP address; cannot establish dial-up	Detach/re-attach	AT+CFUN=0 AT+CFUN=1	Consider possible restrictions to multiple active <cid>s for APN.
		Deactivate/re-activate context (<cid> != 1)	AT+CGACT=0, <cid> AT+CGACT=1, <cid>	
Network registration	Cannot register	Detach/re-attach	AT+CFUN=0 AT+CFUN=1	AT+CFUN=0 / 1 is faster than AT+COPS=2 / 0
RF	Cannot register	Disable/re-enable RF functionality	AT+CFUN=0 AT+CFUN=1	AT+CFUN=4 is NVM persistent, better to use AT+CFUN=0
Module FW	Cannot register	Soft reset	AT+CFUN=16	
AT interface	No response from module	HW switch off	GPIO power on	See LENA-R8 series system integration manual [2] for details and alternatives

Table 21: Full-stack watchdog example

9 Internet protocols

9.1 Data security

9.1.1 Certificates manager +USECMNG

The +USECMNG AT command manages SSL/TLS certificates and private keys. Particularly, the command can:

- Import certificates and private keys
- List and retrieve information of imported certificates and private keys
- Remove certificates and private keys
- Calculate MD5 hash for imported certificate or private key

For additional details on this AT command, the number and the format of the certificates, and the private keys accepted, see the AT commands manual [3].

The SSL/TLS connection with server and/or mutual authentication can be performed using the following key size:

- for Rivest-Shamir-Adleman (RSA) keys at least 2048 bits
 - for Elliptic Curve Digital Signature Algorithm (ECDSA) keys at least 192 bits
- The same limitation is also applied to the keys used in the generation of certificates.

The following example shows the use of the +USECMNG AT command to perform a mutual authentication, using certification authority (CA) certificate, client certificate, and client private key.

Command	Response	Description
AT+USECMNG=1,0,"ca_cert","ca_certificate.crt"	+USECMNG: 1,0,"ca_cert", "d10137cee624fcee624418db5eaa" OK	Import the CA certificate from the "ca_cert.crt" file stored on the file system.
AT+USECMNG=1,1,"client_cert","client_certificate.crt"	+USECMNG: 1,1,"client_certificate", "b137ce137ce5edd6723d8b13" OK	Import the client certificate from the "client_cert.crt" file stored on the file system.
AT+USECMNG=1,2,"client_key","client_key.key"	+USECMNG: 1,2,"client_key", "087ab34c9aa03fbce5edd6723d8b8e05" OK	Import the client private key from the "client_key.key" file stored on the file system.
AT+USECMNG=3	CA, "ca_cert", "An MQTT broker", "2032/10/18 08:23:32" CC, "client_cert", "A client certificate", "2032/06/22 12:34:48" PK, "client_key" OK	List all imported certificates or private keys.

Table 22: +USECMNG examples

The import of the PKCS8 encrypted private key is not supported.

9.1.2 Profile configuration +USECPRF

The +USECPRF AT command allows the configuration of USECMNG (u-blox SECURITY MaNaGement) profiles used for an SSL/TLS connection.

The command manages security profiles for the configuration of the following SSL/TLS connections properties:

- Certificate validation level
- Minimum SSL/TLS version
- Cipher suites to be proposed: legacy, IANA nomenclature, list of cipher suites
- Certificate to be used for server and mutual authentication
- Expected server hostname, when using certificate validation level 1, 2 or 3
- Password for the client private key if it is password protected
- Pre-shared key used for connection
- Server name indication (SNI)
- Server certificate pinning
- Pre-shared key

For additional details on this AT command and related configurations, see AT commands manual [3].



Command	Response	Description
AT+USECPRF=0	OK	Reset (set to factory-programmed value) all the parameters of security profile #0.  We recommend issuing the reset as the first command to erase all previously stored values.
AT+USECPRF=0,0,1	OK	Enable certificate validation without URL integrity check for profile #0. The server certificate will be verified with a specific trusted certificate or with each of the imported trusted root certificates.
AT+USECPRF=0,2,3	OK	Select legacy cipher suite for profile #0.
AT+USECPRF=0,3,"ca_cert"	OK	Select trusted root certificate internal name for profile #0.
AT+USECPRF=0,5,"client_cert"	OK	Select trusted client certificate internal name for profile #0.
AT+USECPRF=0,6,"client_key"	OK	Select trusted client key internal name for profile #0.
AT+USECPRF=0,10,"<SNI_address>"	OK	Configure the server's name indication.  Some servers require this configuration to correctly perform the secure connection.

Table 23: +USECPRF AT command examples

9.1.2.1 Cipher suites

A cipher suite is a set of algorithms and protocols used in the SSL/TLS handshake to negotiate the security setting for the secure connection. The cipher suite for the TLS protocol mainly consists of:

- Key Exchange Algorithm: determines the way symmetric keys are exchanged (RSA, DH, ECDH, DHE, ECDHE, PSK).
- Authentication/ Digital Signature Algorithm: determines how server authentication and client authentication (if required) are performed (RSA, ECDSA, DSA, etc.).
- Bulk Data Encryption: determines which symmetric key algorithm is used to encrypt the actual data (AES, CHACHA20, Camellia, ARIA, etc.). The Bulk Data Encryption is defined by an algorithm, his strength, and operating mode (block cipher mode or stream cipher mode).
- Message Authentication Code (MAC) algorithm: Determines the method that the connection should use to perform data integrity checks (SHA, SHA-256, SHA-384, POLY1305, etc.). Hash-Based Message Authentication Code (HMAC) is used.

A cipher suite is defined by a string representing a named combination of the algorithms and protocol:

$$\text{TLS}_{\{ \text{Key Exchange} \}}_{\{ \text{Authentication/Digital Signature} \}}_{\text{WITH}}_{\{ \text{Bulk Data Encryption} \}}_{\{ \text{Message Authentication Code} \}}$$

As an example, for the TLS 1.0, TLS 1.1, and TLS 1.2 protocols, the following paragraph shows each part of the cipher suite string **TLS_RSA_WITH_AES_256_CBC_SHA**:

- Key Exchange Algorithm: **RSA**.
- Bulk Data Encryption: **AES_256_CBC**.
- Message Authentication Code (MAC) Algorithm: **SHA**.

The Authenticated Encryption with Associated Data (AEAD) bulk ciphers can perform authentication and encryption of the message. For the AEAD bulk ciphers in the string representation the Bulk Data Encryption part and Message Authentication Code part are merged.

If the remote server does not support one of these cipher suites selected in the security profile settings, the handshake fails, and module will be unable to connect to the server.

9.1.3 Complete example

Command	Response	Description
Step 1: Import a trusted root certificate using the byte stream similar to the +UDWNFILE AT command		
AT+USECMNG=0,0,"ThawteCA",1516 >		Start the data transfer using the stream of byte. Unlike the example in section 9.1.1, here the certificate is transferred as a byte stream and is not stored in the LENA-R8 file system.
-----BEGIN CERTIFICATE----- MIIEDCCAwigAwIBAgIQNE7VVyDV7ex J9C/OjVaMaA== -----END CERTIFICATE-----	+USECMNG: 1,0,"ThawteCA", "8ccadc0b22cef5be72ac411a 11a8d812" OK	Input PEM formatted trusted root certificate data bytes. Output MD5 hash string of the stored trusted root certificate DER.
Step 2: List all available certificates and private key		
AT+USECMNG=3	CA, "ThawteCA", "thawte Primary Root CA", "2036/07/17" OK	List all available certificates and private keys.
Step 3: Set the security profile 2 validation level to a trusted root		
AT+USECPRF=2,0,1	OK	Security profile 2 has the validation level set to a trusted root.
Step 4: Set the security profile 2 trusted root certificate to the CA certificate imported as "ThawteCA"		
AT+USECPRF=2,3,"ThawteCA"	OK	Security profile 2 will use the CA certificate imported as "ThawteCA" for server certificate validation.
Step 5: Use the configured USECMNG profile 2 with the UHTTP application		
AT+UHTTP=0,1,"www.ssl_tls_test_ server.com"	OK	Configure the UHTTP server name.
AT+UHTTP=0,6,1,2	OK	Enable the SSL/TLS for the UHTTP profile #0 and specify the SSL/TLS security profile #2.
AT+UHTTPC=0,1,"/", "https.resp"	OK	Execute the HTTP GET command.
	+UUHTTPCR: 0,1,1	HTTP GET URC response.

Table 24: HTTPS example

Due to the significant memory fingerprint of an SSL/TLS connection, the number of concurrent SSL/TLS connections is limited. The +USECMNG AT command and the underlying SSL/TLS infrastructure allows 4 concurrent SSL/TLS connections (i.e., 4 HTTPS requests or 2 HTTPS and 2 FTPS requests).




9.1.4 Troubleshooting secure connection

This section provides a list of recommendations to configure the secure SSL/TLS connection between cellular modules and server.




If the application is unable to complete a secure connection, we recommend application designer to review all the following items so to properly configure the TLS session.

- Decide the certification validation level required for your system and configure the module accordingly with the `<op_code>=0` of the `+USECPRF` AT command.
- Install the SSL/TLS CA certificate based on server TLS certificate chain, by the `+USECMNG` and `+USECPRF` AT commands.
- Check the SSL/TLS protocol version required at the server and configure the module accordingly with the `<op_code>=1` of the `+USECPRF` AT command.
- Make sure that cipher suite required by the destination server is present in the list of cipher suites available by default in the u-blox module. Alternatively, configure it with the `<op_code>=2` of the `+USECPRF` AT command.
- If mutual authentication is adopted, properly configure the module with the specific device certificates and keys by the `+USECMNG` and `+USECPRF` AT commands.
- Finally, ensure the SNI and the expected server “host name” are properly configured and aligned with the destination server. This can be achieved with the `<op_code>=10` and `<op_code>=4`, of the `+USECPRF` AT command.

9.2 TCP/UDP internal stack

-  Verify that the module is registered with the network and a PS data connection is activated. Make sure to follow the steps in section 7.2 before using the AT commands in this section.
-  For UDP it is highly recommended to use `+USOST` and `+USORF` AT commands instead of `+USOCO`, `+USOWR` and `+USORD` AT commands.
-  The use of `+USOST` and `+USORF` AT commands is recommended without the use of the `+USOCO` AT command. The `+USOCO` AT command is compatible only with `+USORD` and `+USOWR` AT commands.

9.2.1 Socket connect

Command	Response	Description
<code>AT+USOCR=6</code>	<code>+USOCR: 0</code> OK	TCP socket creation. In this example socket #0 is created. The information text response returns the created socket identifier (in this case #0). If a new socket is created (without closing the already existent), a new socket identifier will be returned.  Created socket is by default mapped to CID 1.  It is possible to manually configure a mapping between the embedded socket and another PDP context by specifying the IP type and CID. E.g., with the following AT command: <code>AT+USOCR=<protocol>[,<local_port>[,<IP_type>e][,<cid>]]].</code>
<code>AT+USOCR=17</code>	<code>+USOCR: 1</code> OK	Create another socket (in this case the socket is UDP, and its identifier is 1).  Created socket is by default mapped to the CID 1.
<code>AT+USOCL=1</code>	OK	Close socket #1. The socket #1 is free.

Command	Response	Description
AT+UDNSRN=0,"ftp.u-blox.com"	+UDNSRN: "195.34.89.241" OK	DNS resolution of the URL "ftp.u-blox.com".
AT+USOCO=0,"195.34.89.241",7	OK	Connect socket #0 to port 7 of a remote host with IP address 195.34.89.241. The connection is now uniquely associated to the socket. The socket is now ready for read/write operations.
AT+USOCO=0,"195.34.89.241",7	ERROR +UUSOCL: 0	If the connection is not successfully performed, an error result code is returned and the socket used for the connection attempt is <u>closed</u> . The notification is provided by +UUSOCL URC.

Table 25: Socket connection example

9.2.2 Socket listening



Command	Response	Description
AT+USOCR=6	+USOCR: 0 OK	TCP socket creation with ID #0.
AT+USOLI=0,1099	OK	Set the socket in listening mode on port 1099.  The ability to reach the opened port on the server depends also on the network operator. Some network operators do not allow incoming connection on opened TCP/UDP port.
	+UUSOLI: 1,"151.9.34.66",39912,0,"151.9.34.74",1099	When a connection request arrives from a remote host, a new socket is created with the first integer identifier available. In this example the socket ID is #1. The +UUSOLI URC indicates: <ul style="list-style-type: none"> • 1: the new socket created. Incoming data from the established connection will be received on this socket. Data to be sent must be written into this socket • 151.9.34.66: IP of the remote server • 39912: service port • 0: listening socket. It is the socket identifier specified with the +USOLI AT command • 151.9.34.74: module IP address • 1099: listening port assigned to the connection. Configured with the +USOLI AT command Socket #1 is now ready for reading/writing data.
	+UUSORD: 1,18	18 bytes of incoming data over the previously established connection.  The incoming data will always be sent on the related socket.

Table 26: Socket in listening mode example

9.2.3 Socket write (+USOWR)


Command	Response	Description
AT+USOWR=0,2,"12"	+USOWR: 0,2 OK	Write 2 data bytes data on socket #0. If the final result code is returned, then the data is sent to a lower level of the protocol stack. This is not an acknowledgment from the remote host where the data bytes were sent.  Some characters are not allowed in base syntax mode. For the allowed characters, see the AT commands manual [3].

Table 27: Writing on socket example

9.2.4 Socket read (+USORD)

Command	Response	Description
	+UUSORD: 0,2	The remote server sends 2 data bytes on socket #0. A URC is returned indicating the socket on which the data is received, and the total amount of data received.
AT+USORD=0,2	+USORD: 0,2,"ax" OK	Read data. The data is between quotation marks.

Table 28: Reading from socket example

9.2.5 Socket close



Command	Response	Description
AT+USOCL=0	OK	The socket is closed by the module (socket #0).  No +UUSOCL URC returned.

Table 29: Closure of a socket

9.3 Internet clients

 Verify that the module is registered with the network and a PS data connection is activated. Follow the steps in section 7.2 before using the AT commands in this section.

Internet embedded client that are available in the u-blox LENA-R8 cellular mode are HTTP, FTP, MQTT, and MQTT-SN.


The following sections provide examples by use case. For a complete list of AT commands and details see the LENA-R8 AT commands manual [3].

9.3.1 HTTP

9.3.1.1 Default and minimal configuration

This section shows an example of the u-blox proprietary +UHTTP and +UHTTPC AT commands. These commands are used for sending requests to a remote HTTP server, receiving the server responses, and transparently storing them in the file system.

The supported HTTP methods are HEAD, GET, DELETE, PUT, POST file, and POST data.

Command	Response	Description
AT+CMEE=2	OK	Set verbose error result codes.
AT+UHTTP=0	OK	Reset the HTTP profile #0.
AT+UHTTP=0,1,"httpbin.org"	OK	Set the server domain name and port.
AT+UHTTP=0,5,80	OK	 HTTP server name (e.g., "httpbin.org"). The factory-programmed value is an empty text string.
AT+UDNSRN=0,"httpbin.org"	+UDNSRN: "54.72.52.58" OK	DNS resolution of httpbin.org.
AT+UHTTPC=0,0,"/", "head.ffi"	OK +UUHTTPCPC: 0,0,1	HEAD request of the default page and store the result into the "head.ffi" file on the local file system of the module. The +UUHTTPCPC URC notifies the success/failure of the operation (in this example: success).
AT+UHTTPC=0,1,"/", "get.ffi"	OK +UUHTTPCPC: 0,1,1	GET request of the default page and store the result into the "get.ffi" file on the local file system of the module. The +UUHTTPCPC URC notifies the

Command	Response	Description
		success/failure of the operation (in this example: success).
AT+UHTTTPC=0,5,"/post","post.fff", "name_post=MyName&age_post=30",0	OK +UUHTTTPCR: 0,5,1	POST request sending data using content-type application/x-www-form-urlencoded. The result is saved in the "post.fff" file on the local file system of the module. The +UUHTTTPCR notifies the success/failure of the operation (in this example: success).
AT+UHTTTP=0,2,"test_user"	OK	Set the authentication for the HTTP server: HTTP server username.
AT+UHTTTP=0,3,"P455w0rd"	OK	HTTP server password.
AT+UHTTTP=0,4,1	OK	HTTP server authentication method (basic authentication).
AT+UHTTTPC=0,1," /basic-auth/test_user/P455w0rd","get_auth.fff"	OK +UUHTTTPCR: 0,1,1	GET request returning information on authenticated user. The page requires basic authentication. The result is saved in "get_auth.fff" file on the local file system of the module. The +UUHTTTPCR URC notifies the success/failure of the operation (in this example: success).

Table 30: HTTP configuration example

9.3.1.2 HTTP POST

Command	Response	Description
AT+CMEE=2	OK	Set the verbose error result codes.
AT+UDWNFILE="postdata.txt",11	>hello world OK	Write some data in the file to send.
AT+URDFILE="postdata.txt"	+URDFILE: postdata.txt,11 ,"hello world" OK	Optionally check whether the data is present.
AT+UHTTTP=0	OK	Reset the HTTP profile #0.
AT+UHTTTP=0,1,"httpbin.org"	OK	Set up a connection to an echo server (httpbin.org) that checks, and echoes post commands.
AT+UHTTTP=0,5,80	OK	Set the port of the HTTP request to 80
AT+UHTTTPC=0,4,"/post","result.txt", "postdata.txt",1	OK +UUHTTTPCR: 0,4,1	Submit a post command in text format and store the answer in result.txt.
AT+URDFILE="result.txt"	+URDFILE: result.txt,498, "HTTP/1.1 200 OK Content-Type: application/json Date: Tue, 15 Jan 2013 16:06:11 GMT Server: gunicorn/0.16.1 Content-Length: 345 Connection: Close { "headers": { "Content-Length": "11", "Host": "httpbin.org", "Content-Type": "text/plain", "User-Agent": "UBlox Leon G200/1.0 (N7/HTTP 1.0)", "Connection": "keep- alive" },	Check the server's reply.

Command	Response	Description
	<pre>"args": {}, "data": "hello world", "url": "http://httpbin.org/post" , "files": {}, "json": null, "form": {}, "origin": "10.82.21.198" }" OK"</pre>	

Table 31: HTTP POST example

9.3.1.3 Secure HTTP (HTTPS)

Configure a secure manager profile before starting a secure HTTP. See section 9.1 for further details.

The following example describes how to configure the secure HTTP. Only the secure manager profile must be configured, the other HTTP commands will behave as in the case of unencrypted session.




Command	Response	Description
AT+UHTTP=0,6,1	OK	Enable secure HTTP. HTTPS (SSL encryption) enabled.  The port number is not set automatically. Usually for HTTPS the port 443 (standard value for HTTPS) is used. As previously reported, the port number can be changed using AT+UHTTP=0,5,<port_number> command.

Table 32: HTTPS enabling example

9.3.2 FTP

9.3.2.1 Default and minimal configuration

Command	Response	Description
AT+UFTP=1,"ftp.u-blox.com"	OK	Parameter configuration for FTP server connection. These parameters will be set: <ul style="list-style-type: none"> FTP server hostname FTP username FTP password FTP connection mode (PASSIVE connection). Most FTP servers prefer the PASSIVE mode due to security issues.
AT+UFTP=2,"anonymous"	OK	
AT+UFTP=3,"password"	OK	
AT+UFTP=6,1	OK	
AT+UDNSRN=0,"ftp.u-blox.com"	+UDNSRN: "195.34.89.241" OK	Hostname resolution.
AT+UFTPC=1	OK +UUFTPCR: 1,1	Connect to the server and manage the FTP connection using the +UFTPC AT command. Let's start connecting to the server. The +UUFTPCR URC provides the FTP command result (the last parameter provides the result, 1 if is successfully performed).
AT+UFTPC=13	OK +UUFTPCD: 13,194,"-rw-r--r-- 1 ftp ftp 1037 Aug 5 09:45 dat_000"	Request the file list on the server. The +UUFTPCD URC provides the FTP data.

Command	Response	Description
	<pre>-rw-r--r-- 1 ftp ftp 21041 Aug 5 09:12 data.zip -rw-r--r-- 1 ftp ftp 12 Aug 5 09:42 xlog.zip " +UUFTPCR: 13,1</pre>	
AT+UFTPC=10,"uploads"	<pre>OK +UUFTPCR: 10,1</pre>	Directory creation on the FTP server.
AT+UFTPC=13	<pre>OK +UUFTPCD: 13,258,"-rw-r-- r-- 1 ftp ftp 1037 Aug 5 09:45 dat_000 -rw-r--r-- 1 ftp ftp 21041 Aug 5 09:12 data.zip drwxr-xr-x 2 ftp ftp 4096 Aug 5 09:48 uploads -rw-r--r-- 1 ftp ftp 12 Aug 5 09:42 xlog.zip " +UUFTPCR: 13,1</pre>	Request again the file list.
		Change directory to directory name "uploads".
AT+UFTPC=8,"uploads"	<pre>OK +UUFTPCR: 8,1</pre>	 Use AT+UFTPC=8,".." to return back in the parent directory.
AT+UFTPC=5,"gps_positions","gps_positions"	<pre>OK +UUFTPCR: 5,1</pre>	Upload a file from the module to the FTP server from the local file system of the module (in this example filename "gps_positions").
AT+UFTPC=5,"gps_positions","gps_positions",250	<pre>OK +UUFTPCR: 5,1</pre>	Restart the upload file from the module to FTP server from the local module file system (in this example filename "gps_positions"), starting from byte 250.  The FTP server should support the REST command to support these functionalities. The server should write the file starting from byte indicated.
AT+UFTPC=13	<pre>OK +UUFTPCD: 13,70,"-rw-r-- r-- 1 ftp ftp 176673 Aug 5 10:03 gps_positions" +UUFTPCR: 13,1</pre>	File list and information request.
AT+UFTPC=14	<pre>OK +UUFTPCD: 14,15,"gps_posi tions" +UUFTPCR: 14,1</pre>	File list request.
AT+UFTPC=8,".."	<pre>OK +UUFTPCR: 8,1</pre>	Return to the parent directory.
AT+UFTPC=4,"data.zip","data.zip"	<pre>OK +UUFTPCR: 4,1</pre>	Download a file from the FTP server to the local file system of the module.
AT+UFTPC=4,"data.zip","data.zip",1	<pre>OK +UUFTPCR: 4,1</pre>	Restart the file download from the FTP server to the local module file system from the latest byte saved on the file system (this is automatically calculated). The data received is written after the latest byte available on the file system.

Command	Response	Description
AT+UFTPC=0	OK +UUFTPCR: 0,1	FTP server disconnection.

Table 33: FTP configuration example

9.3.2.2 Direct link

9.3.2.2.1 Retrieve a file from FTP server

Command	Response	Description
AT+UFTPC=1, "ftp.u-blox.com"	OK	Parameter configuration for FTP server connection. These parameters will be set: <ul style="list-style-type: none"> FTP server hostname
AT+UFTPC=2, "anonymous"	OK	<ul style="list-style-type: none"> FTP username
AT+UFTPC=3, "password"	OK	<ul style="list-style-type: none"> FTP password
AT+UFTPC=6, 1	OK	<ul style="list-style-type: none"> FTP connection mode (PASSIVE connection)
AT+UDNSRN=0, "ftp.u-blox.com"	+UDNSRN: "195.34.89.241" OK	Hostname resolution.
AT+UFTPC=1	OK +UUFTPCR: 1,1	Connect to the server and manage the FTP connection using the +UFTPC AT command. The +UUFTPCR URC is returned when the connection is established.
AT+UFTPC=6, "file_to_retrieve"	CONNECT	Send to the FTP server a RETRIEVE file request for file_to_retrieve. The CONNECT intermediate result code means the direct link mode is activated: the data received from FTP connection will be redirected to the serial port.
AT+UFTPC=6, "file_to_retrieve", 2 50	CONNECT	Restart a RETRIEVE file request for file_to_retrieve file from byte 250. The CONNECT intermediate result code means the direct link mode activation: the data received from FTP connection is redirected to the serial port. The data reception begins with the byte indicated.
+++	DISCONNECT OK	⚠ When the file has entirely been retrieved the module does not exit from the direct link mode. It is necessary to manually exit using the "+++" escape sequence.
	+UUFTPCR: 6,1	The +UUFTPCR URC notifies how the retrieve operation has been concluded (1 means success).

Table 34: FTP file retrieving example

9.3.2.2.2 Aborting retrieve file request

Command	Response	Description
+++	DISCONNECT OK	If entering "+++" escape sequence before the requested file has been entirely retrieved from FTP server, the module exits from the direct link.
	+UUFTPCR: 6,0	The +UUFTPCR URC notifies that the retrieve operation has not been concluded successfully (0 means fail).

9.3.2.2.3 Store a file on FTP server

Command	Response	Description
AT+UFTPC=1, "ftp.u-blox.com"	OK	Parameter configuration for FTP server connection. These parameters will be set: <ul style="list-style-type: none"> FTP server hostname;





Command	Response	Description
AT+UFTP=2, "anonymous"	OK	<ul style="list-style-type: none"> FTP username;
AT+UFTP=3, "password"	OK	<ul style="list-style-type: none"> FTP password;
AT+UFTP=6, 1	OK	<ul style="list-style-type: none"> FTP connection mode (PASSIVE connection).
AT+UDNSRN=0, "ftp.u-blox.com"	+UDNSRN: "195.34.89.241" OK	Hostname resolution. Connect to the server and manage the FTP. Connection using the +UFTPC command. Let's start connecting to the server.
AT+UFTPC=1	OK +UUFTPCR: 1, 1	The +UUFTPCR URC is returned when the connection is established.
AT+UFTPC=7, "file_to_store"	CONNECT	Send to FTP server a STORE file request for <code>file_to_store</code> . The CONNECT intermediate result code means the direct link mode is activated: the data sent through the serial port will be redirected to the FTP server through the FTP connection.
AT+UFTPC=7, "file_to_store", 250	CONNECT	Restart the STORE file request for <code>file_to_store</code> from byte 250. The CONNECT intermediate result code means the direct link mode activation: the data sent through the serial port is redirected to the FTP server through the FTP connection. The data is written on the FTP server starting from byte indicated.  The FTP server should support REST command to support this functionality.
+++	DISCONNECT OK +UUFTPCR: 7, 1	When the data upload is completed use the "+++" escape sequence for exiting from the direct link mode. The +UUFTPCR URC notifies if the STORE operation has been concluded successfully.

Table 35: FTP file uploading example

9.3.2.2.4 About "+++" escape sequence usage

To switch from the data mode to the command mode, the application shall send a proper escape sequence to the module.

The escape sequence "+++" is detected when it is received by the module in a single separate frame of 3 bytes in length. This will happen if the host waits 2 seconds after all data has been transmitted before issuing the "+++" string.

-  In case the host application needs to send "+++" as the final part of the payload, an additional byte should be added to avoid false detection.
-  In case flow control is activated by the module (e.g., when data is transmitted over a congested or low throughput network), there is the risk that the escape sequence is queued in the host connectivity buffers and delivered to the module in frames containing data payload. To avoid missed detection of the escape sequence, it is suggested to send the "+++" string when the COM port has CTS asserted/flow control disabled.
-  The module does not recognize the escape sequence "+++" if a delay bigger than 500 ms is placed between the three "+" characters.

9.3.2.3 Using secure option

Command	Response	Description
AT+UFTP=0, "123.213.132.231"	OK	Parameters configuration for the FTP server connection in secure mode: <ul style="list-style-type: none"> • FTP server address • FTP username • FTP password • FTP SSL encryption control channel enabled • FTP SSL encryption data channel enabled
AT+UFTP=2, "myname"	OK	
AT+UFTP=3, "mypwd"	OK	
AT+UFTP=8, 1	OK	
AT+UFTP=12, 1	OK	
AT+UFTPC=1		FTP login.
	OK	Connect to the server and manage the FTP connection using the +UFTPC AT command. Let's start connecting to the server.
	+UUFTPCR: 1, 1	The +UUFTPCR URC provides the FTP command result (the second parameter provides the result, 1 if is successfully performed).
		Some operators may not accept a secure FTP connection:
AT+UFTPC=1	OK	The URC provides the FTP command result: the second parameter is 0, an error has occurred.
	+UUFTPCR: 1, 0	
AT+UFTPER	+UFTPER: 8, 63	Retrieving of error class and code: <ul style="list-style-type: none"> • Error class 8: "Wrong FTP API usage" • Error code 63: "Cannot set secure socket"
	OK	

Table 36: FP secure option example

- When the FTP client is using a secure connection, only the explicit mode is supported (ftpes://). Moreover, in the explicit mode, the secure connection will be established after the FTP connection (before login) on the same port of the control channel.
- When the FTP client is using a secure connection, the FTPS server may request that the session data of the control channel connection should be reused to establish secure connection on the data channel. In this case, the session resumption feature for the FTPS client shall be configured via <op_code>: 13 of the +USECPRF AT command.

9.3.3 MQTT

9.3.3.1 Default and minimal configuration

The configuration required to start a MQTT session depends on the broker (server) configuration, the most important of which is the MQTT remote server information. Use the broker configuration to correctly set up the module before starting a session.

Command	Response	Description
AT+CMEE=2	OK	Set verbose error result codes.
AT+UMQTT?	+UMQTT: 0, "357862090033897" +UMQTT: 2, "", 1883 +UMQTT: 3, "", 1883 +UMQTT: 4, "" +UMQTT: 6, 0 +UMQTT: 7, 0 +UMQTT: 8, "" +UMQTT: 9, 0, "" +UMQTT: 10, 0 +UMQTT: 11, 0	Read the current profile configuration. All the reported values can be modified; see the AT commands manual [3] for a detailed description. The default client id value is the IMEI of the module because it guarantees the uniqueness of the client to the server.

Command	Response	Description
	OK	
AT+UMQTT=2, "192.168.105.30", 1883	OK	Set the remote MQTT server's IP address and port. Alternatively, the server's name can be set with the AT+UMQTT=3 command.

Table 37: MQTT configuration example

9.3.3.2 Start and end a MQTT session

See the previous section to configure the MQTT profile before starting a connection.

Command	Response	Description
AT+UMQTTC=1	OK	Connect to the broker.
	+UUMQTTC: 1,1	The MQTT session request is successfully performed. The MQTT session can start. The +UUMQTTC URC provides the result of the requested action from the MQTT broker
AT+UMQTTC=0	OK	Disconnect from the broker, end of the MQTT session.
	+UUMQTTC: 0,1	The disconnection is successfully performed.

Table 38: Starting / ending MQTT session

9.3.3.3 Subscribe to a topic and publish a message to the same topic

The following example is a demonstration of the main functionalities that can be performed with the AT commands. In this MQTT session the module subscribes to a topic, publishes a message to the topic and receives the published message (since it is subscribed to topic of the published message).

Command	Response	Description
AT+UMQTTC=4,0,"module/lights"	OK	Subscribe to a topic.
	+UUMQTTC: 4,1,0,"module/light"	The broker granted QoS level is 0.
AT+UMQTTC=2,0,0,0,"module/lights","light_1 is red"	OK	Publish "light_1 is red" message to the "module/lights" topic with requested QoS level and retain value set to 0.
	+UUMQTTC: 2,1	
	+UUMQTTC: 6,1	Notification of the received publish message.
AT+UMQTTC=6,1	+UMQTTC: 6,0,27,13,"module/lights",14,"light_1 is red"	Read the received publish message.
	OK	
AT+UMQTTC=5,"module/lights"	OK	Unsubscribe from the previously subscribed topic.
	+UUMQTTC: 5,1	

Table 39: MQTT subscribe and publish examples

9.3.3.4 Secure MQTT

Configure a secure manager profile before starting a secure MQTT session (using the TLS encryption protocol). For more details, see section 9.1.

The following example shows how to configure the MQTT profile before starting a secure session with the broker. Only the secure manager profile and the remote port must be configured; the other MQTT commands will behave as in the case of unencrypted session.

Command	Response	Description
AT+UMQTT=11,1,2	OK	Enable the secure MQTT option using the USECMNG profile 2.

Command	Response	Description
AT+UMQTT=2, "192.168.105.30", 8883	OK	Set the remote MQTT broker IP address and port. The default port for secure MQTT is 8883.
AT+UMQTTC=1	OK +UUMQTTC: 1,1	Connect to the broker and start a secure MQTT session.

Table 40: Secure MQTT example

9.3.4 MQTT-SN

9.3.4.1 Default and minimal configuration

The configuration required to start a MQTT-SN session depends on the gateway configuration, most importantly, the MQTT-SN remote server information. Before starting a session, be sure to correctly set up the module with the gateway configuration.

Command	Response	Description
AT+CMEE=2	OK	Set verbose error result codes.
AT+UMQTTSN?	+UMQTTSN: 0, "357862090033897" +UMQTTSN: 1, "", 1883 +UMQTTSN: 2, "", 1883 +UMQTTSN: 4, 0 +UMQTTSN: 5, 0 +UMQTTSN: 6, "" +UMQTTSN: 7, 0, "" +UMQTTSN: 8, 0 OK	Read the current profile configuration. All the reported values can be modified; see the AT commands manual [3] for a detailed description. The default client id value is the IMEI of the module because it guarantees the uniqueness of the client to the server.
AT+UMQTTSN=2, "192.168.105.30", 10000	OK	Set the IP address and port of the remote MQTTSN gateway. Alternatively, the gateway's server name can be set with the AT+UMQTTSN=1 command.

Table 41: MQTT-SN configuration example

MQTT-SN secure functionality (option 9 of the AT+UMQTTSN command) is not supported in this product.

9.3.4.2 Subscribe to a normal topic

Example of MQTT-SN session subscription to a topic.

Command	Response	Description
AT+UMQTTSNC=5,0,0, "room/temperature"	OK +UUMQTTSNC: 5,1,0,1	Subscribe to a normal topic (0) with requested QoS level set to 0. The gateway granted QoS level is 0 and the topic ID for "room/temperature" is 1.

Table 42: MQTT-SN subscribe example

9.3.4.3 Publish and read a message to a topic

In this MQTT-SN session the module publishes a message to the topic and receives the published message (assuming it is subscribed to topic of the published message).

Command	Response	Description
AT+UMQTTSNC=4,0,0,0, "1", "20 degrees Celsius"	OK +UUMQTTSNC: 4,1 +UUMQTTSNC: 9,1	Publish the "20 degrees Celsius" message to the topic ID 1 with requested QoS level and retain value set to 0. Notification of the received publish message.

Command	Response	Description
AT+UMQTTSN=9,1	+UMQTTSN: 9,1,0,19,1,"1",18,"20 degrees Celsius" OK	Read the received publish message.

Table 43: MQTT-SN subscribe and publish examples

9.3.4.4 MQTT Anywhere

MQTT Anywhere is a u-blox IoT communication SIM-based LPWA service that can operate around the world without the need for specific cellular agreements with multiple MNOs.

This service uses the MQTT-SN protocol, and it is directly integrated into u-blox products. Additionally, devices are authenticated via the hardware [IoT SIM card](#), ensuring that the user traffic is never exposed to the public internet. Device payloads can be enriched and transformed using the Data Flow Manager within u-blox Thingstream, which also provides integration with virtually any 3rd party enterprise system or IoT platform.

When using a Thingstream SIM card ([IoT SIM card](#)), be aware that there are two different APNs available:

- APN 'tsudp' allows only connectivity to Thingstream MQTT Anywhere server and it is mandatory to access this service;
- APN 'tsiot' allows generic data traffic.

Additional details on this topic are available on the product [webpage](#).

In addition to the MQTT-SN basic settings, the MQTT Anywhere service required the configuration of a unique client ID and the clean session. See an example of these configurations in the table below.

Command	Response	Description
AT+UMQTTSN=0,"identity:xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"	OK	To ensure a unique value, the identity value of the SN thing needs to be used as the client ID. The identity value can be found on the management console page of u-blox Thingstream platform.
AT+UMQTTSN=8,600	OK	Set the connection duration (in seconds).
AT+UMQTTSN=10,1	OK	Set the MQTT-SN clean session.

Table 44: MQTT Anywhere example

A complete example of the 'MQTT Anywhere' configuration can be seen on the IoT Communication-as-a-Service guide [webpage](#).

10 SIM

10.1 SIM architecture and behavior

10.1.1 SIM card and SIM profiles

The SIM card contains, besides the authentication information which is at the basis of the security of the cellular communications, registration parameters and MNO preferences that might affect the PLMN selection procedure. Overall, the main files available in a SIM card are called a “SIM profile”. Most SIM cards have single MNO profiles which are preinstalled and contain files like IMSI, EHPLMN list, PPLMN lists. Some MNOs sell global SIM cards whose MCC is 901 and whose actual MNO is the first entry in the operator preferred PLMN list.

Some SIM cards need to be activated at first usage, so they require an exchange of data with the MNO SIM OTA server. Other SIM cards need to be provisioned and, again, a data session with the SIM OTA server is performed at first startup. Finally, some SIM cards might contain several profiles (so called multi-IMSI SIM) and swap between them based on the UE registration conditions, which are reported by the cellular module to the UICC applet mounted on the SIM card via SIM toolkit events and commands.

10.1.2 eSIM/eUICC and remote SIM provisioning

The SIM can be a chip soldered in the customer design: in this case the SIM cannot be removed or changed, hence it shall have a generic SIM profile to access the cellular network.

Such eSIMs are called M2M SIM cards and can have IP connectivity throughout the world, on one or more PDP contexts, possibly with a static IP address assigned at LTE attach.

The eUICC might have a default subscriber profile, that might be updated over the air by the MNO to a different profile (e.g., a different subscriber identity IMSI).

Support of the SIM toolkit feature and the BIP protocol is required to comply to GSMA SGP.02 “Remote Provisioning Architecture for Embedded UICC” technical specification.

10.1.3 SIM subscription

There is no way to understand if a USIM is enabled to LTE: LTE subscription is stored in the HLR (Home Location Register) of the home network. If LTE registration succeeds, the SIM is LTE enabled. Subscription restriction might apply in roaming or in specific geographic areas.

Usually if the LTE registration has failed with the reject cause #15, “No Suitable cells in Tracking area”, it might mean that the SIM is not LTE enabled. The host application can issue the +CEER and AT+CEREG=3 command to retrieve the reject cause. The reject cause #19, “ESM failure”, might indicate a wrong APN on the initial default bearer (cid 0).

It is advisable to check the reject cause to avoid triggering too many SW or HW resets of the module, which might cause network unfriendly behaviors.

10.2 SIM communication

Module and SIM card communicate through a serial interface. The module automatically starts a communication with the SIM at boot for cellular protocol stack operations. The host application can interact with applets and services residing in the SIM card using a set of AT commands. Based on capabilities these commands can be divided in two groups:

- commands for restricted access
- commands for generic access

10.2.1 Commands for restricted access

Commands for restricted access are a set of high-level commands that allow simple but limited interactions with the SIM and its contents. The handling of all the steps required by communication protocol used by module - SIM interface is managed internally by the module, and therefore not a concern for the host application.

This subset includes the +CRSM AT commands.

Table 45 reports some examples of commands for restricted access.

Command	Response	Description
Read IMSI (International Mobile Subscriber Identity)		
AT+CRSM=176,28423,0,0,9	+CRSM: 144,0,"082922888505109849" OK	Read the IMSI using the Restricted SIM access command. <ul style="list-style-type: none"> 176 is the read command for EF in binary format. 28423 is the ID for EF_IMSI, the SIM elementary file where IMSI is stored by SIM manufacturer. The <response> parameter in +CRSM answer contains few prefix bytes followed by the IMSI value represented with nibbles swapped in pairs as can be verified by comparing the response to +CIMI
AT+CIMI	222885850018994 OK	Read the IMSI. The response is the readable serial number. This example is for reference only, as an alternative AT command to read IMSI.
LOCI management		
AT+CRSM=176,28542,0,0,11	+CRSM: 144,0,"5C36967C22F2885FC90000" OK	Read EF_LOCI, the SIM elementary file where information about last CS location is stored
AT+CRSM=214,28542,0,0,11,"FFFFFFF FFFFFFFFFFFFFFF"	+CRSM: 144,0,"" OK	Erase EF_LOCI content. This command is useful to start a registration procedure from scratch. For this scope, the below EF shall have to be erased as well: EF_PSLOCI (28531) EF_EPSLOCI (28643)
Read emergency call codes		
AT+CRSM=178,28599,1,4,16	+CRSM: 144,0,"11F2FF456D657267656E6379FF FFF17" OK	Read the list of Emergency Call numbers stored in the SIM by MNO. 178 is the read command for EF in record format.

Table 45: examples of +CRSM commands

10.2.2 Commands for generic access

Commands for generic access are a subset of low-level commands that allow the direct control of messages sent to the SIM and received from it; the full knowledge of APDU protocol syntax and procedures is therefore needed.

This subset includes the +CSIM AT command.

10.2.3 SIM logical channels

The exchange of messages (APDU) between SIM card and module occurs through "logical channels" that work on the physical SIM serial interface.

10.2.3.1 Basic logical channel

At module boot, the "basic logical channel" (logical channel 0) is automatically opened, and it is used for cellular protocol stack operations.

This channel is owned by the module, and the host application is not allowed to close it. For the same reason, internal module commands have the priority, and AT commands that do not coordinate (e.g., +CSIM) will not disrupt the module functionality but might be disturbed by the module.

10.2.3.2 Supplementary logical channels

To allow interactions with applets and services residing in the SIM card, cellular SIM cards support supplementary logical channels (up to 3).

These channels shall be explicitly activated by the module and the SIM card assigns them a progressive number from 1 to 3. The user or host application can manage logical channels using proper AT commands. It is recommended to use an independent supplementary channel for each different applet/service and to close them at the end for reuse.

Table 46 shows the procedure to read ICCID using the +CSIM AT commands.

Command	Response	Description
Generic SIM access		
AT+CSIM=10,"0070000000"	+CSIM: 4,6C01 OK	Request to open a new supplementary logical channel. The command syntax is not accepted because '01' value is expected for Luicc The behavior differs from other u-blox modules but anyway conformant to ETSI specs [8].
AT+CSIM=10,"0070000001"	+CSIM: 6,"019000" OK	Request to Open a new supplementary logical channel. The syntax is similar to above example, but Luicc field has been set to '01' as expected by LENA-R8. As per ISO/IEC 7816-4, <response> parameter contains the number assigned to the opened channel: "01". The action result "9000" means 'Command successfully executed'.
AT+CSIM=14,"01A40004023F00"	+CSIM: 86,"62278202782183023F00A50780017 1830242868A01058B032F0601C60C9001 6083010183010A83010C9000" OK	SELECT Master File 0x3F00
AT+CSIM=14,"01A40004022FE2"	+CSIM: 54,"62178202412183022FE28A01058B0 32F06088002000A8801109000" OK	SELECT EF_ICCID 0x2FE2
AT+CSIM=10,"01B00000FF"	+CSIM: 24,"989388435801009549F49000" OK	READ the content of selected file (10 bytes). The content is returned with nibbles swapped in pairs as can be verified by comparing the response to +CCID. The action result "9000" means 'Command successfully executed'

Command	Response	Description
AT+CCID	+CCID: 8939883485100059944F OK	
AT+CSIM=10, "0070800100"	+CSIM: 4, "9000" OK	CLOSE channel

Table 46: Examples of +CSIM commands

11 FW update

The device firmware (FW) management is a key feature for devices integrating a cellular module. It shall be considered during the application design to evaluate which method will have to be supported.

u-blox module supports below methods for firmware update:

- **FOAT** (Firmware Over AT) / +UFWUPD: it is based on transferring the firmware image from host file system to the module via AT interface (USB or UART), using the Xmodem or Xmodem-1k protocol.
- **FOTA** (Firmware Over The Air) / +UFWINSTALL: it is based on firmware delta package stored in the module file system. This package contains code differences between installed firmware and the target one.
- **Flashing**: it does not use the standard AT interface, but a dedicated serial interface by means of a specific software tool for Windows.

All methods are composed by three different stages:

- **Download**: it consists of storing the new firmware image on the module file system. Depending on the method this phase can be triggered locally by host application / user or remotely on request to u-blox.
- **Validation**: it consists of checking the file content and its signature. Depending on the method this phase can start automatically or need a further command.
- **Installation**: it is the phase during which the new firmware is finally installed. In all methods it automatically run after validation.

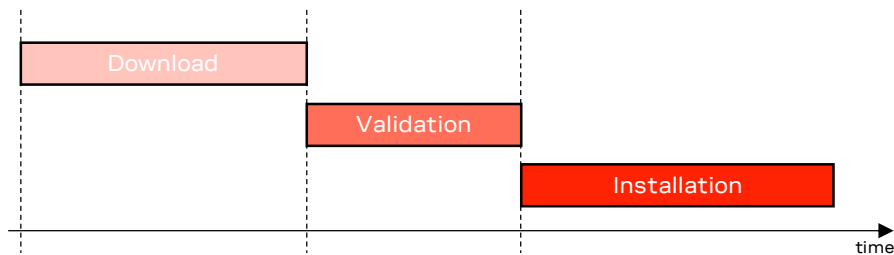


Figure 7: FW update stages

	FOAT (applied with +UFWUPD)	FOTA (applied with +UFWINSTALL)	Flashing
Starting FW image storage	Host FS	Module FS	Windows © PC
Starting FW type	Delta package	Delta package	Full image
Available ports	USB (in Normal mode) UART	USB (in Normal mode) UART	USB (in USB boot mode)
Download trigger	+UFWUPD (via serial port)	+UFTPC (via FTP) +UHTTTPC (via HTTP) +UDWNFILE (via serial port)	ResearDownload tool
Validation trigger	automatic	+UFWINSTALL	automatic
Installation trigger	automatic	automatic	automatic

Table 47: Comparison of FW update modes features

11.1 Firmware update Over AT (FOAT)


Firmware update Over AT (FOAT) is a firmware installation method on transferring the content of delta package from host file system to the module via AT interface (USB or UART), using the Xmodem or Xmodem-1k protocol.

As soon as the download is completed, the procedure automatically proceeds with validation and installation phases.

11.1.1 FOAT via +UFWUPD AT command

On receiving the +UFWUPD AT command, the module:

- Resets, restarts, and then switches to Firmware Update Mode
- Configures the serial port at the new baud rate (if the baud rate specified in the +UFWUPD AT command is different from the one previously used)
- Sends the **+UFWUPD: ONGOING** IRC to the host terminal via the AT interface, followed by up to three “C” (0x43) characters and up to ten <NACK> (0x15) characters. The first three “C” characters are sent with a 3 s timeout after the last one, next <NACK> characters are sent with a 10 s timeout after the last one. The total timeout is 109 seconds. This is the timeout within which the host terminal must send TX data

 If data is sent while the “C” character is coming, the protocol uses the CRC method to detect transmission errors, otherwise the standard CHECKSUM method is used.


Then it is possible to send the FOAT update file via the Xmodem protocol using a terminal emulator tool at the selected serial interface and selected baud rate without flow control (e.g., HyperTerminal with these settings: frame format 8N1, no flow control, baud rate configurable, power saving disabled). The update file will be downloaded into the module file system.

After the download ends, the +UFWPREVAL URCS display the progress indication for the update file validation. The progression of the validation is incremental, but the increment can be different from 1. The +UFWPREVAL: 100 URC may not be issued, and the module can start the installation phase issuing the +UFWUPD URC.

After the end of the update file validation:

- If the validation fails, the procedure will be suspended and a FOAT error code will be issued together with the +UFWUPD URC. The module exits from the update procedure mode and returns to the normal mode since the firmware is unchanged and usable.
- If the validation is successfully performed, the firmware installation procedure will start, notified by the +UFWUPD URCS.

During the update operations, the +UFWUPD URCS display the progress indication and the operation result on the serial interface set via the +UFWUPD AT command. The progression of the installation is incremental, but the increment can be different from 1.

 Both IRC and any further final result codes are sent at the new baud rate on the serial interface selected. Only a syntax error in the +UFWUPD AT command triggers an error result code at the original baud rate.

During the update process, the module cannot be used to make calls, even emergency calls.

When the firmware update is completed, a URC will notify the final result of the operation. See the LENA-R8 AT commands manual [3] for the list of possible final result codes.

At the end of a successful installation, the module boots up (see example of +UFWUPD procedure in [Table 48](#)).

Command	Response	Description
AT+UFWUPD=1,115200,,1	> +UFWUPD: ONGOING CCC	+UFWUPD trigger the procedure: the module switches to Firmware Update Mode and issue +UFWUPD: ONGOING IRC, followed by up to three "C" characters. If data is sent earlier than the third "C", the protocol uses the CRC method to detect transmission errors, otherwise the standard CHECKSUM method is used.
	+UFWPREVAL:0 +UFWPREVAL:1 ... +UFWPREVAL:100	+UFWINSTALL trigger the procedure. The validation procedure starts. +UFWPREVAL URCS with index from 0 to 100 show the progress.
	+UFWUPD:0 +UFWUPD:100	As soon as the validation is end, the installation procedure automatically starts. +UFWUPD IRCs with index from 0 to 100 show the progress. After + UFWUPD:100, the module takes few minutes to complete the procedure.
	(Module reboot)	The module automatically reboots to start with the new FW version
	+UFWINSTALL: 128	+UFWINSTALL: 128 confirms the end of installation procedure.


Table 48: +UFWUPD example

11.2 Firmware Over The Air (FOTA)

Firmware Over The Air (FOTA) is a firmware installation method based on a delta package stored in the module file system. This package contains code differences between installed firmware and the target one.

Firmware delta package contains code differences between installed firmware and the target one. It has to be saved in FOAT partition and named "updatePackage.bin"; to do it, use one of the methods described in section [11.2.1](#)

When delta package file is available in the file system, the installation can be triggered by issuing +UFWINSTALL AT command, see section [11.2.2](#) for details about this phase.

 The starting release of delta package shall correspond to the current firmware running on the module.

11.2.1 Firmware download

11.2.1.1 Firmware download via FTP

Firmware for LENA-R8 series modules can be downloaded using standard FTP. This section goes through the AT commands required to download a firmware delta file from an FTP server.

The host needs to first configure an FTP profile with the server parameters in order to start the FW download. After the firmware update has been downloaded, install the new firmware using the +UFWINSTALL AT command.

11.2.1.1.1 FTP service configuration +UFTP

Before starting a firmware download via FTP the host needs to first configure the FTP profile with the FTP server and other parameters.

For a complete description of the FTP profile configuration and examples, see the LENA-R8 series AT commands manual [3]

11.2.1.1.2 FTP command +UFTPC

The AT+UFTPC=100 command is used to trigger a firmware delta file download from an FTP server. The downloaded file is automatically labeled with the “FOAT” tag and saved in a special folder with the “updatePackage.bin” name. These path and file name are required by the +UFWINSTALL AT command in order to correctly perform the FW installation.

11.2.1.1.3 Firmware download via FTP example

Table 49 **Error! Reference source not found.** reports an example of firmware download via FTP.

Command	Response	Description
AT+UFTP=1, "ftp.firmware.com"	OK	Configure server name.
AT+UFTP=2, "username"	OK	Set username.
AT+UFTP=3, "password"	OK	Set password.
AT+UFTP=6, 1	OK	FTP mode: passive.
AT+UFTPC=1	OK	FTP login request.
	+UUFTPCR: 1, 1	FTP successfully logged
AT+UFTPC=100, "/fw/delta"	OK	Start FTP download.
	+UUFTPCR: 100, 1, "d557104b7a29965b9e3dbcf38c56501b"	URC file transfer complete reporting the md5 checksum.
AT+ULSTFILE=0, "FOAT"	+ULSTFILE: "updatePackage.bin" OK	List the delta files on file system. The “FOAT” tag is used to store firmware delta files.

Table 49: Firmware download via FTP example

11.2.1.2 Firmware download via HTTP

This section goes through the AT commands required to download a firmware delta file from a server via HTTP.

The host needs to first configure a HTTP profile with the server parameters to start the firmware download. After the firmware delta file has been downloaded, install the new firmware using the +UFWINSTALL AT command; for more details, see section 6.

11.2.1.2.1 HTTP profile configuration +UHTTP

Before starting a firmware download via HTTP, the host needs to first configure the HTTP application profile parameters.

For a complete description of the HTTP profile configuration, see the LENA-R8 series AT commands manual [3].

11.2.1.2.2 HTTP command +UHHTPC

The AT+UHHTPC=100 command is used to trigger a firmware package download from a server via HTTP. The downloaded file is automatically labeled with the “FOAT” tag and is saved in a special folder with the “updatePackage.bin” name. These path and file name are required by the +UFWINSTALL AT command in order to correctly perform the FW installation.

11.2.1.2.3 Firmware download via HTTP example

Table 50 reports an example of firmware download via HTTP.

Command	Response	Description
AT+UHTTP=0,0,"125.24.51.133"	OK	Configure the server IP address
AT+UHTTP=0,2,"username"	OK	Set the username
AT+UHTTP=0,3,"password"	OK	Set the password
AT+UHTTFC=0,100,"/fw/delta"	OK	Get the FOTA update file
	+UUHTTFCR: 0,100,1,200,"60bfec89060a2901148ad1c2beae214b"	URC file transfer complete reporting the md5 checksum
AT+ULSTFILE=0,"FOAT"	+ULSTFILE: "updatePackage.bin" OK	List the delta files on file system. The "FOAT" tag is used to store firmware delta files.

Table 50: Firmware download via HTTP example

11.2.1.3 Firmware download via serial interface

It is possible to download a firmware delta file to the LENA-R8 file system from a host processor connected to the module via serial interface (UART or USB). Once the delta file is on the host processor's file system, an application can use the +UDWNFILE AT command to transfer the delta file to the module, specifying the "FOAT" tag so the module will know this is a firmware delta.

For a complete description of the +UDWNFILE AT command, see the LENA-R8 series AT commands manual [3].

11.2.1.3.1 Firmware download via serial interface example

Table 51 reports an example of firmware download via UART (+UDWNFILE).

Command	Response	Description
AT+UDELFILE="updatePackage.bin"	OK	Delete a possible existent delta file.
AT+UDWNFILE="updatePackage.bin", 74268,"FOAT" > <delta file contents>	OK	Download the delta file to LENA-R8 file system via UART.
AT+ULSTFILE=0,"FOAT"	+ULSTFILE: "updatePackage.bin" OK	List the delta files on file system. The "FOAT" tag is used to store firmware delta files.

Table 51: Firmware download via serial interface example

11.2.2 Firmware validation and installation +UFWINSTALL

FOAT firmware update example is the AT command to trigger the validation of delta package stored in module file system and its installation.

Command	Response	Description
AT+UFWINSTALL=1,115200,,1	+UFWPREVAL:0 +UFWPREVAL:1 ... +UFWPREVAL:100	+UFWINSTALL trigger the procedure. The validation procedure starts. +UFWPREVAL URCS with index from 0 to 100 show the progress.
	+UFWINSTALL:0 +UFWINSTALL:100 (wait ~2 min)	As soon as the validation is end, the installation procedure automatically starts. +UFWINSTALL URCS with index from 0 to 100 show the progress. After +UFWINSTALL:100, the module takes few minutes to complete the procedure.
	+UFWINSTALL: 128	+UFWINSTALL: 128 confirms the end of installation procedure.

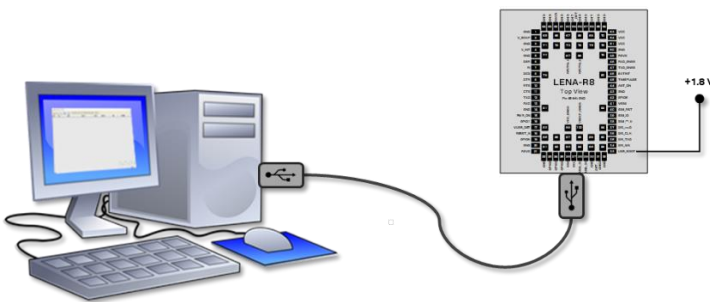
Command	Response	Description
	(Module reboot)	The module automatically reboots to start with the new FW version
	+UUSTATUS: READY	Default greeting message

Table 52: +UFWINSTALL AT command example

11.3 Flashing

The flashing is a firmware update method that does not use AT interface, but a dedicated serial interface by means of a specific software tool for Windows, ResearchDownloadUsb.

To activate this specific serial interface, the “USB_BOOT” input pin has to be set to high status before the modem boot. Doing so the module will be forced to start in USB boot mode for firmware update, waiting for a firmware download. See [2] for details about “USB_BOOT” pin placement and characteristics; see [4] for instructions on how to manage USB_BOOT pin in EVK-R8.


Figure 8: Setup for flashing

As soon as ResearchDownloadUsb recognizes the USB device for flashing, it autonomously starts the update.

11.3.1 System setup

11.3.1.1 USB drivers for flashing

The flashing process requires specific USB driver has been installed on PC.

Download the driver installation package from u-blox LENA-R8 series webpage [9]. The package contains drivers for both possible modes: «Normal boot» (typical usage) and «USB boot» (needed for USB FW flashing).

11.3.1.1.1 Installation instructions

- Download the driver archive, unzip it and run the executable file
- Configure the module in USB FW update boot mode by setting the USB_BOOT pin to the proper status
- Turn on the module and wait for USB device recognition

11.3.1.2 Software tool

ResearchDownloadUsb is the tool for flashing LENA-R8 modules.

Download it from u-blox LENA-R8 series webpage [9]. No installation is needed, just unzip the package and run the exe file.

11.3.2 Short instructions

1. Configure the system setup in USB FW update boot mode; see u-blox LENA-R8 series system integration manual [2] or u-blox EVK-R8 user guide [4]

2. Run ResearchDownload.exe tool
3. Click button and select the proper LENA-R8 firmware image file (*.pac file)
4. Click button
5. Switch on LENA-R8 module
6. Wait until Status column shows “Finish” and verify Progress column shows “Passed”
7. Click button
8. Configure the system setup in normal boot mode
9. Reboot the module

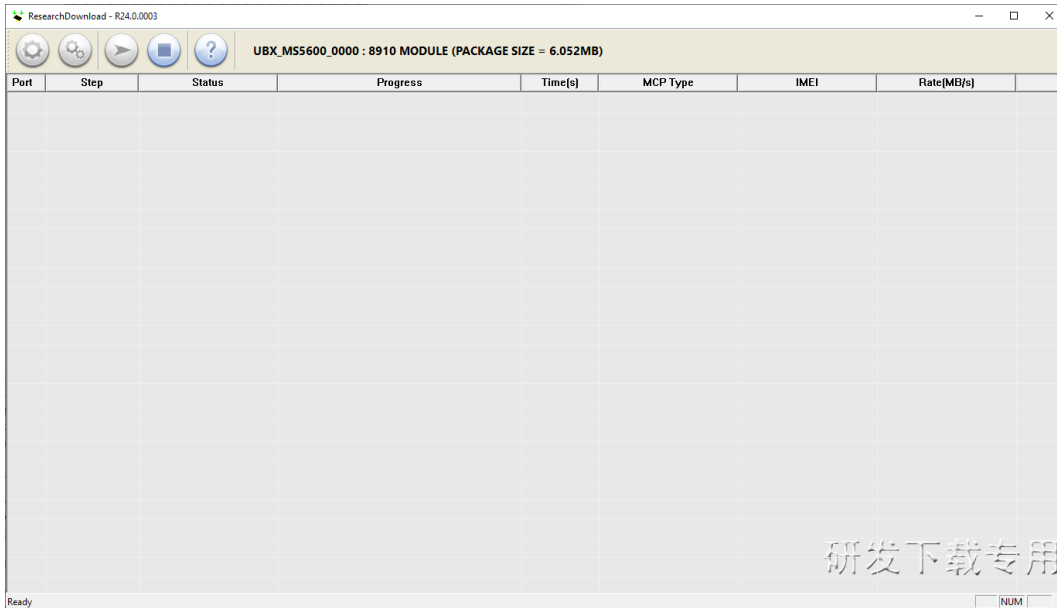


Figure 9: Flashing tool screenshot

11.4 Impact to device files and settings

This firmware update methods do not restore any factory configuration.

To set the default modem settings after a firmware update, use the +UFACTORY AT command, as described in section 6.

Item	FOAT (applied with +UFWUPD)	FOTA (applied with +UFWINSTALL)	Flashing
"USER" tagged files stored in user file system	Preserved	Preserved	Preserved
MNO profiles	Preserved	Preserved	Preserved
MNO list	Preserved	Preserved	Preserved
User NVM settings	Preserved	Preserved	Preserved
User certificate and private keys	Preserved	Preserved	Preserved
RF Calibration Data	Preserved	Preserved	Preserved
SMS	Preserved	Preserved	Preserved

Table 53: Impact of FW update

12 OEM production testing

Guidelines for production testing and prototype validation for applications using LENA-R8 modules are provided in LENA-R8 production and prototype validation guide [\[5\]](#).

13 Migration guides

13.1 LARA-R6 series to LENA-R8 series

This list of software changes between LARA-R6 and LENA-R8 series modules may help in migrating a host application between the two modules. For details of the AT commands below, compare LARA-L6 / LARA-R6 AT Commands Manual [10] and LENA-R8 AT commands manual [3].

Command or feature	LARA-R6 series	LENA-R8 series
3GPP Release	Rel. 10 for LTE	Release 13 LTE
Local connectivity peripherals	+USIO, +UARTCONF	n/a
+UARTCONF	Available for AUX UART configuration	n/a
AT command execution and URC	AT command processing blocked if AT command ongoing on a different interface, URC issues on all ports (+UURCONF to configure)	AT commands ongoing on one interface does not block other AT interfaces.
+CFUN syntax	+CFUN=19 allows to disable UICC driver	In +CFUN: 0, AT+COPS=0 AT command forces the modem in +CFUN: 1 status.
+UAUTHREQ authentication parameters configuration	AT+UAUTHREQ=...<password>,<username>	AT+UAUTHREQ=...<username>,<password>
MNO profile selection	+UMNOPROF Default MNO profile: 90	n/a
+URAT syntax	AT+URAT=<1stAcT>[,<2ndAcT>[,<3rdAcT>]]	AT+URAT=<SelectedAcT>[,<PreferredAcT>[,<2ndPreferredAcT>]]
Band configuration	+UBANDMASK	+SETBAND for 2G bands +SETLOCK for LTE bands
CS/PS usage setting	+USVCDOMAIN	n/a
Cell deep scan	+UCFSCAN	n/a
IMS cid		IMS cid (11): activation is not notified by URC deactivation is notified by the +CGEV URC.
Internet applications/Sockets	+CGACT and optionally +CGDCONT: Always issue the AT+CGACT=1,<cid> command on used context <cid> even if the +CGDCONT AT command returns that the context is still active with a valid IP address. This is highly recommended to activate the PS data connection avoiding possible conflicts between applications.	online data mode resuming: explicitly activate PDP context with +CGACT to guarantee a proper resume with ATO; see 0.
PDP contexts usage	Internet app are allowed to address different PDP contexts	Internet app uses PDP context based on their order in activation; see 0 Table 16: Anchor APN configuration example Additional bearers.
DNS configuration	+UDNSCFG allows independent DNS configuration for each cid	+CDNSCFG, common setting for each cid. Shall be used after every changing in PDP context activation
PDN sharing between dial-up and Internet applications/Socket	Supported, use +UEMBPF to reserve port range for embedded IP clients	n/a
PPP	DTR (virtual and physical) line must be asserted to start a PPP connection	
Device management	LwM2M	n/a
FOTA	uFOTA via LwM2M, FTP, HTTP	FOTA via FTP, HTTP

Command or feature	LARA-R6 series	LENA-R8 series
Cell and network diagnostic	+UCGED, +VZWRSPR/+VZWRSRQ (in +UMNOPROF: 3)	n/a
Call hang up	+CHUP	ATH, +CHUP
+UPSV (options)	+UPSV: (0,1,3)	+UPSV: (0,2,3,4) +UPSV:2 cannot be supported at the same time of the "General Purpose I/O" function and/or the "GNSS data ready" function over the GPIO3 line
Fast dormancy	+CNMPD	n/a

Table 54: Migration LARA-R6 series to LENA-R8 series guide


Appendix

A Glossary

Abbreviation	Definition
ASCII	American Standard Code for Information Interchange
ARM	Arm (Advanced RISC Machines) Holdings
AEC	Automotive Electronics Council
BBR	Battery Backed RAM
BER	Bit Error Rate
CPU	Central Processing Unit
UTC	Coordinated Universal Time
DCE	Data Circuit-terminating Equipment* / Data Communication Equipment*
DTE	Data Terminal Equipment
DC	Direct Current
DRX	Discontinuous Reception
DDC	Display Data Channel
DL	Down Link (Reception)
XYZ	This table can be automatically constructed using the custom macro.

Related documentation

- [1] u-blox LENA-R8 series data sheet, [UBX-22003110](#)
- [2] u-blox LENA-R8 series system integration manual, [UBX-22015376](#)
- [3] u-blox LENA-R8 series AT commands manual, [UBX-22016905](#)
- [4] u-blox EVK-R8 – User guide, [UBX-22018774](#)
- [5] u-blox LENA-R8 series Production and prototype testing guidelines Application note, [UBX-22040091](#)
- [6] 3GPP TS 27.010 V3.4.0 - Terminal Equipment to User Equipment (TE-UE) multiplexer protocol (Release 1999)
- [7] u-blox Mux implementation in cellular modules application note, [UBX-13001887](#)
- [8] ETSI TS 102 221 - Smart Cards; UICC-Terminal interface; Physical and logical characteristics
- [9] u-blox LENA-R8 series webpage, www.u-blox.com/en/product/lena-r8-series
- [10] u-blox LARA-L6 / LARA-R6 series AT Commands Manual, [UBX-21046719](#)
- [11] Github m-center repository, github.com/u-blox/m-center
- [12] ubxlib: u-blox host library, <https://www.u-blox.com/en/product/ubxlib>

 For product change notifications and regular updates of u-blox documentation, register on our website, www.u-blox.com.

Revision history

Revision	Date	Name	Comments
R01	29-May-2023	mrod	Initial release

Contact

u-blox AG

Address: Zürcherstrasse 68
8800 Thalwil
Switzerland

For further support and contact information, visit us at www.u-blox.com/support.