

---

## Getting started with the X-CUBE-CELLULAR cellular connectivity Expansion Package for STM32Cube

### Introduction

The X-CUBE-CELLULAR Expansion Package enables connectivity over cellular networks. The network access technology depends on the cellular modem used. It can be 2G/3G as well as LTE Cat M1 or NB-IoT.

This user manual provides:

- A brief content description of the X-CUBE-CELLULAR Expansion Package for STM32Cube
- Instructions to start the hardware composed of a board based on an STM32 microcontroller associated with a modem, and to exchange data through a cellular network by means of the X-CUBE-CELLULAR firmware



# 1 Acronyms

Table 1 presents the definitions of the acronyms that are relevant for a better contextual understanding of this document.

**Table 1. Acronyms**

Acronym	Definition
API	Application programming interface
APN	Access point name
DB	Data brief (STMicroelectronics document type)
CLI	Command-line interface
eSIM	Embedded SIM
FAQ	Frequently asked questions
FW	Firmware
HTTP	Hypertext transfer protocol
HW	Hardware
IDE	Integrated development environment
MCU	Microcontroller unit
MVNO	Mobile virtual network operator
IP	Internet protocol
PC	Personal computer
SIM	Subscriber identity module
UM	User manual (STMicroelectronics document type)

The X-CUBE-CELLULAR Expansion Package runs on the STM32L4 32-bit microcontrollers based on the Arm® Cortex®-M4 processor.

*Note:* Arm is a registered trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere.



## 2 Overview of available documents

Table 2 lists the main documents useful for getting X-CUBE-CELLULAR started.

**Table 2. References**

ID	Description
[1]	Release note: The main RN is available at the root of the X-CUBE-CELLULAR Expansion Package. The main RN provides a link to the <i>Cellular RN</i> , which contains all information about the current delivery.
[2]	Data brief: <i>Cellular connectivity software expansion for STM32Cube</i> data brief (DB3582). The DB summarizes X-CUBE-CELLULAR features and presents the main license terms associated with the Expansion Package. It is available from STMicroelectronics web site at <a href="http://www.st.com">www.st.com</a> .
[3]	User manual: <i>Getting started with the X-CUBE-CELLULAR cellular connectivity Expansion Package for STM32Cube</i> user manual (UM2426). The UM provides the detailed information needed to use X-CUBE-CELLULAR and adapt it to users' needs. It is available from STMicroelectronics web site at <a href="http://www.st.com">www.st.com</a> .

## 3 Hardware and firmware setup

### 3.1 Connect hardware

To start using [X-CUBE-CELLULAR](#), the user needs:

- A PC
- A Type-A to Micro-B USB cable
- An STM32L4-based master hardware combined to, or associated with a cellular modem (refer to [2] or [3] for possible solutions)

The PC is used to power HW, program FW, and display STM32 FW traces and results onto a serial terminal. The ST-LINK USB driver, available as [STSW-LINK009](#) on STMicroelectronics web site at [www.st.com](http://www.st.com), is mandatory and must be installed on the PC. It is needed to program FW onto the STM32 microcontroller.

Follow the next five steps to connect the HW:

1. If the STM32 microcontroller and modem are on separate boards, connect these boards together.
2. Connect the antenna.
3. All the boards compatible with [X-CUBE-CELLULAR](#) provide a SIM slot, which makes it possible to use a plastic SIM instead of an eSIM. If this option is selected, insert the SIM card.
4. Connect the *STLink* port of the STM32 board with the PC by means of the USB cable.
5. See that a new drive corresponding to the STM32 board is available on the PC, resulting from the previous installation of the ST-LINK USB driver.

### 3.2 Activate the eSIM (optional)

This section applies only to the [P-L496G-CELL01](#) and [P-L496G-CELL02](#) among the HW compatible with [X-CUBE-CELLULAR](#). The other compatible pieces of HW must be used with a plastic SIM.

The [P-L496G-CELL01](#) and [P-L496G-CELL02](#) Discovery packs contain an STMod+ board that embeds an eSIM provisioned with EMnify (MVNO) prepaid data plan profile. Once activated, it allows the user a free amount of 15 Mbytes of data for 90 days. When the data is used or the 90-day period is over, the user can contact EMnify to upgrade his plan. The user can monitor his data consumption and registration status on the EMnify portal.

The original binary firmware programmed during the factory process displays information, such as URL and voucher, required for user registration. The user must connect to the provided URL with a browser, create an account, and enter the voucher. A new HW is added to his account (more than one HW can be registered to the same account, in this case the user has a HW list). Once the voucher is entered, the user can select the EMnify button to activate the eSIM provisioned with EMnify profile: the eSIM is activated and allows the connection to cellular networks.

**Caution:** In case the user HW is [P-L496G-CELL01](#) or [P-L496G-CELL02](#), programming a new FW erases factory FW, making the voucher required for registration unavailable and eSIM activation impossible. It is possible to recover the voucher by re-programming factory FW or using an AT command (refer to FAQ in [1]).

To see the voucher, the user simply needs to install and configure a serial terminal on his PC. Tera Term is the tool used for that purpose in the framework of the current document as well as in [3]. Proceed as follows for configuring the serial terminal:

1. Install the Tera Term software on the PC
2. Check that the STM32 board is connected to the PC through the USB cable
3. Start Tera Term and select the proper connection (featuring the *STMicroelectronics* name)
4. Set the parameters
  - a. Terminal
    - **[New line]**
      - **[Receive]:** CR
      - **[Transmit]:** CR
    - **[Local echo]** selected
  - b. Serial
    - **[Baud rate]:** 9600

- **[Data]:** 8 bit
- **[Parity]:** none
- **[Stop]:** 1 bit
- **[Flow control]:** none
- **[Transmit delay]:** 10 ms each

Connect the board to the PC and press the black reset button of the board. A trace appears with the instructions and voucher information.

### 3.3 Select LwIP or modem-socket configuration mode

It is important to note that there are two configuration modes concerning the location of the IP stack:

- Either the IP stack runs on the STM32 microcontroller (LwIP mode)
- Or the IP stack used is located in the modem (modem-socket mode)

Only the LwIP mode is available when the modem does not include an IP stack or no AT command is available to expose the socket interface.

The main interests of the modem-socket mode is to reduce the STM32 memory footprint and make STM32 exchanges with the modem simpler. The main drawback of the modem-socket mode is lower performance. Refer to the *Cellular RN* in [1], which presents STM32 FW size figures (Flash and RAM) for both IP modes.

### 3.4 Program firmware into the STM32 microcontroller

This section explains how to select binary firmware and program it into the STM32 microcontroller.

Binary FW is delivered as part of the [X-CUBE-CELLULAR](#) Expansion Package, for each supported IDE: IAR™, Keil®, System Workbench for STM32 (SW4STM32), or TrueSTUDIO®.

For example, for the [32L496GDISCOVERY](#), the binaries are located in `Projects\STM32L496G_Discovery\Demonstrations\Cellular\Binaries\<XXX>\`, where `<XXX>` indicates the IDE:

- `<XXX> = iar: IAR™`
- `<XXX> = keil: Keil®`
- `<XXX> = sw4: System Workbench for STM32`
- `<XXX> = sw4l: System Workbench for STM32 on Linux®`

The IP mode (LwIP or modem-socket) used in the binary is part of the file name. See for instance two binaries produced with the IAR™ IDE for the BG96 modem:

- `iar_bg96_lwip_v300.bin` for the LwIP mode
- `iar_bg96_socket_v300.bin` for the modem-socket mode

**Note:** *STM32 FW for STM32L496 devices (“L496”) can be used on the [32L496GDISCOVERY](#) Discovery board (with screen) as well as host board of the [P-L496G-CELL01](#) and [P-L496G-CELL02](#) Discovery packs (screenless).*  
When HW and PC are connected by means of the USB cable, the related drive is available on the PC. Drag and drop the chosen FW into that drive. Wait a few seconds that the FW file disappears from the file manager: this indicates that FW is programmed into the STM32 MCU.

**Note:** *The [X-CUBE-CELLULAR](#) Expansion Package provides all the source files needed to build FW for the various supported IDEs.*

## 4 Run X-CUBE-CELLULAR firmware

### 4.1 Functional content of the X-CUBE-CELLULAR Expansion Package

X-CUBE-CELLULAR provides three applications included in the application folder (`Middlewares\ST\STM32_Cellular\Applications`):

- *Echo Client* sends TCP / UDP data to a remote server; the expected result is to receive the same data back from the server
- *PING Client* allows the user to send a `PING` command and see the result in the terminal on the connected PC
- *HTTP Client* sends data to the cloud (GroveStreams) in order to update a dashboard; it can also receive data from the cloud

### 4.2 Set the serial terminal configuration for X-CUBE-CELLULAR firmware

Start Tera Term, select the proper connection (featuring the *STMicroelectronics* name), and set the parameters (the baud rate is different from the one indicated in [Section 3.2 Activate the eSIM \(optional\)](#)):

- Terminal
  - **[New line]**
    - **[Receive]:** CR
    - **[Transmit]:** CR
  - **[Local echo]** selected
- Serial
  - **[Baud rate]:** 115200
  - **[Data]:** 8 bit
  - **[Parity]:** none
  - **[Stop]:** 1 bit
  - **[Flow control]:** none
  - **[Transmit delay]:** 10 ms each

### 4.3 Start X-CUBE-CELLULAR firmware

Restart the board by pressing the black reset button.

Traces are displayed on the serial terminal, starting with a menu, which offers several options to the user:

- Wait for the timer to trigger FW
- Press any key (except **[1]**, **[2]** and **[3]**) for immediate boot
- Press **[1]** to setup the FW configuration
- Press **[2]** to boot only the modem
- Press **[3]** to display the FAQ

Setting the APN is often needed prior to the boot because it depends on the SIM operator. In such case:

1. First press **[1]** before the end of the timer duration for FW configuration
2. Then press **[2]** for cellular services to set up the SIM parameters
3. Once the setup is done, exit from the menu, which boots the board

Once the board boots, the following is displayed in the Tera Term window:

1. Traces of network registration
  2. Device IP address
  3. Traces of the running *Echo Client* application that starts running and exchanges data with the remote server
- If the process described in this section does not occur as expected, refer to [Section 5 Troubleshooting](#).

## 4.4 **PING client demonstration**

The *PING client* pings any remote server and measures the round-trip delay. It is run by entering in the serial terminal:

- `ping` to ping the predefined remote server (default 8.8.8.8)
- `ping xxx.xxx.xxx.xxx` to ping IPV4 address `xxx.xxx.xxx.xxx` of the remote server

The result of the *PING client* demonstration is displayed on the serial terminal.

## 4.5 **HTTP Client demonstration**

The *HTTP Client* demonstration activates and operates GroveStreams cloud access.

User actions are requested prior to running the *HTTP Client* demonstration:

1. Creation of a GroveStreams account at [grovestreams.com](https://grovestreams.com)
  2. Configuration of STM32 FW accordingly<sup>(1)</sup>
1. *The detailed configuration information is provided in [3].*

Once GroveStreams parameters are set, start the *HTTP Client* demonstration by entering `httpclient on` in the serial terminal. Data are sent to the user's GroveStreams account where they are visible in the user's dashboard. The user can also interact with his HW through the GroveStreams dashboard, turning a LED button on and off.

## 4.6 **Run-time configuration**

During FW execution, the command-line interface (CLI) is entered by pressing the **[return]** key. For example, the `help` command lists all available commands.

The CLI and traces are displayed on the same terminal.

## 5 Troubleshooting

### 5.1 Modem firmware version

The modem factory FW version in the user's HW may not be compliant any longer with the latest X-CUBE-CELLULAR version.

The *Cellular RN* indicates which modem FW version must be used (refer to [1]). The modem FW version must be checked in the serial terminal trace and updated if needed. Three methods are proposed, depending on the three different modem providers:

- For Quectel UG96 and BG96, the material needed is included in the X-CUBE-CELLULAR Expansion Package under the `\Utilities\PC_Software\` directory
- For Sequans, information is available from [www.sequans.com/gm01q-stmod](http://www.sequans.com/gm01q-stmod)

### 5.2 Excessive network attachment duration

Especially for modems that have all bands selected, it is interesting to configure the bands to scan, according to the device location and radio environment.

When no M1 / NB1 network is available, the 2G fallback can last several minutes. It is then recommended to reduce the technology (only 2G or M1 and 2G), and also reduce the scanned bands.

To do this, boot FW, select choice [2] from the menu (refer to Section 4.3 Start X-CUBE-CELLULAR firmware), and then use AT commands to setup the modem.

As a result, observe the cellular technology (LTE Cat M, NB-IoT, GSM) and the bands used from the trace at boot time as illustrated by the example below:

```

BG96:>>>> BG96 mode and bands configuration <<<<<
BG96:LTE Cat.M1 band active (scan rank = 1)
BG96:Cat.M1 BANDS config = 0x01000
BG96:CatM1_B13
BG96:GSM band active (scan rank = 2)
BG96:GSM BANDS config = 0xf
BG96:GSM_900
BG96:GSM_1800
BG96:GSM_850
BG96:GSM_1900
BG96:LTE Cat.NB1 band active (scan rank = 3)
BG96:Cat.NB1 BANDS config = 0x08
BG96:CatNB1_B4
BG96:>>>> ..... <<<<<

```

AT commands trace on and modem config get provide the current activated bands.

As a matter of example, for configuring first with the B13 band in M1, then all bands in GSM, and finally the B4 band in NB1, type the [return] key to get the prompt and enter the following commands:

```

modem config iotopmode ALL
modem config nwscanmode AUTO
modem config gsmband any
modem config m1band B13
modem config nblband B4
modem config scanseq M1_GSM_NB1
modem config send

```

Once this is done, reboot the STM32 board by pressing its black reset button.

- Note:**
1. *The modem command displays the possible commands for modem management, such as `modem help`*
  2. *After the last command is sent (`modem config send`), the modem is programmed and saved. Performing the configuration at each boot is not needed*
  3. *The new parameters are effective only after reboot*
  4. *The user can restore the default parameters by using command `modem config send alone`*



## 6 Deep dive in X-CUBE-CELLULAR

---

Refer to [\[3\]](#) to know more about the possibilities of the X-CUBE-CELLULAR Expansion Package.

## Revision history

**Table 3. Document revision history**

Date	Version	Changes
22-May-2019	1	Initial release.

## Contents

<b>1</b>	<b>Acronyms</b>	<b>2</b>
<b>2</b>	<b>Overview of available documents</b>	<b>3</b>
<b>3</b>	<b>Hardware and firmware setup</b>	<b>4</b>
<b>3.1</b>	Connect hardware	4
<b>3.2</b>	Activate the eSIM (optional)	4
<b>3.3</b>	Select LwIP or modem-socket configuration mode	5
<b>3.4</b>	Program firmware into the STM32 microcontroller	5
<b>4</b>	<b>Run X-CUBE-CELLULAR firmware</b>	<b>6</b>
<b>4.1</b>	Functional content of the X-CUBE-CELLULAR Expansion Package	6
<b>4.2</b>	Set the serial terminal configuration for X-CUBE-CELLULAR firmware	6
<b>4.3</b>	Start X-CUBE-CELLULAR firmware	6
<b>4.4</b>	<i>PING client</i> demonstration	6
<b>4.5</b>	<i>HTTP Client</i> demonstration	7
<b>4.6</b>	Run-time configuration	7
<b>5</b>	<b>Troubleshooting</b>	<b>8</b>
<b>5.1</b>	Modem firmware version	8
<b>5.2</b>	Excessive network attachment duration	8
<b>6</b>	<b>Deep dive in X-CUBE-CELLULAR</b>	<b>9</b>
	<b>Revision history</b>	<b>10</b>
	<b>Contents</b>	<b>11</b>
	<b>List of tables</b>	<b>12</b>

## List of tables

<b>Table 1.</b>	Acronyms .....	2
<b>Table 2.</b>	References .....	3
<b>Table 3.</b>	Document revision history .....	10

**IMPORTANT NOTICE – PLEASE READ CAREFULLY**

STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST’s terms and conditions of sale in place at the time of order acknowledgement.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of Purchasers’ products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. For additional information about ST trademarks, please refer to [www.st.com/trademarks](http://www.st.com/trademarks). All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.

© 2019 STMicroelectronics – All rights reserved