UM2334

User manual

Getting started with the X-CUBE-GNSS1 Global Navigation Satellite System software expansion for STM32Cube

Introduction

The X-CUBE-GNSS1 is an expansion software package for STM32Cube.

The software runs on STM32 and includes drivers for the Teseo-LIV3F global navigation satellite system (GNSS) device, middleware for the NMEA protocol support and FreeRTOS for task scheduling to ensure better asynchronous message parsing.

It is built on top of STM32Cube software technology for easy portability across different STM32 microcontrollers.

The software comes with sample implementations for the drivers running on the X-NUCLEO-GNSS1A1 expansion board, when connected to a NUCLEO-F401RE, NUCLEO-L476RG, or NUCLEO-L073RZ board.

The software also includes a sample application for Assisted GNSS provided by the Teseo-LIV3F GNSS device. The application is tailored for the B-L475E-IOT01A Discovery kit for IoT nodes.

1 Acronyms and abbreviations

Table 1. List of acronyms

Acronym	Description
A-GNSS	Assisted global navigation satellite system
BSP	Board support package
BeiDou	China regional navigation satellite system
Galileo	Europe global navigation satellite system
GLONASS	Global navigation satellite system
GNSS	Global navigation satellite system
GPS	Global positioning system
HAL	Hardware abstraction layer
I2C	Inter integrated circuit
IC	Integrated circuit
IP	Internet protocol
NMEA	National Marine Electronics Association - United States standards organization for marine equipment
QZSS	Quasi-zenith satellite system (used in the Asia-Oceania regions)
RTOS	Real-time operating system
TCP	Transmission control protocol
UART	Universal asynchronous receiver-transmitter
USB	Universal serial bus

2 What is STM32Cube?

STM32Cube™ represents the STMicroelectronics initiative to make developers' lives easier by reducing development effort, time and cost. STM32Cube covers the STM32 portfolio.

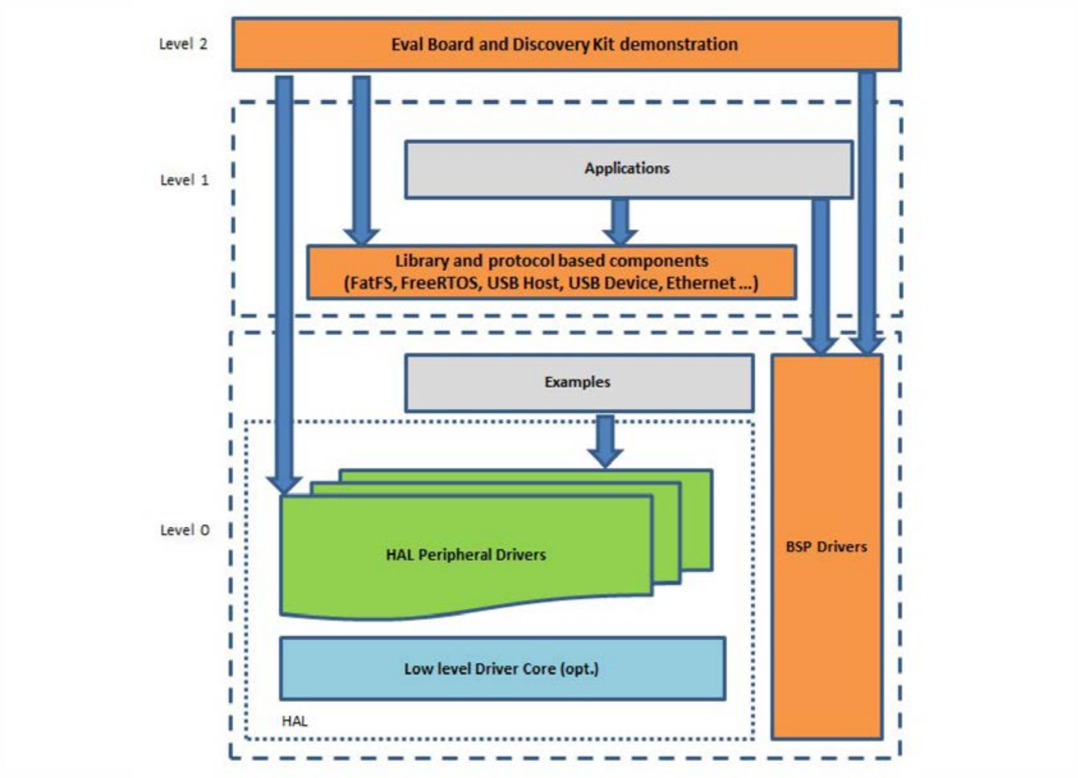
STM32Cube version 1.x includes:

- STM32CubeMX, a graphical software configuration tool that allows the generation of C initialization code using graphical wizards.
- A comprehensive embedded software platform specific to each series (such as the STM32CubeF4 for the STM32F4 series), which includes:
  - the STM32Cube HAL embedded abstraction-layer software, ensuring maximized portability across the STM32 portfolio
  - a consistent set of middleware components such as RTOS, USB, TCP/IP and graphics
  - all embedded software utilities with a full set of examples

2.1 STM32Cube architecture

The STM32Cube firmware solution is built around three independent levels that can easily interact with one another, as described in the diagram below.

Figure 1. Firmware architecture


Level 2
Eval Board and Discovery Kit demonstration
Level 1
Applications
Library and protocol based components (FatFS, FreeRTOS, USB Host, USB Device, Ethernet ...)
Level 0
Examples
HAL Peripheral Drivers
Low level Driver Core (opt.)
HAL
BSP Drivers

Level 0: This level is divided into three sub-layers:

- Board Support Package (BSP): this layer offers a set of APIs relative to the hardware components in the hardware boards (Audio codec, IO expander, Touchscreen, SRAM driver, LCD drivers. etc…); it is based on modular architecture allowing it to be easily ported on any hardware by just implementing the low level routines. It is composed of two parts:

  - Component: is the driver relative to the external device on the board and not related to the STM32, the component driver provides specific APIs to the external components of the BSP driver, and can be ported on any other board.
  - BSP driver: links the component driver to a specific board and provides a set of easy to use APIs. The API naming convention is BSP_FUNCT_Action(): e.g., BSP_LED_Init(), BSP_LED_On().
- Hardware Abstraction Layer (HAL): this layer provides the low level drivers and the hardware interfacing methods to interact with the upper layers (application, libraries and stacks). It provides generic, multi-instance and function-oriented APIs to help offload user application development time by providing ready to use processes. For example, for the communication peripherals (I²C, UART, etc.) it provides APIs for peripheral initialization and configuration, data transfer management based on polling, interrupt or DMA processes, and communication error management. The HAL Drivers APIs are split in two categories: generic APIs providing common, generic functions to all the STM32 series and extension APIs which provide special, customized functions for a specific family or a specific part number.
- Basic peripheral usage examples: this layer houses the examples built around the STM32 peripherals using the HAL and BSP resources only.

Level 1: This level is divided into two sub-layers:

- Middleware components: set of libraries covering USB Host and Device Libraries, STemWin, FreeRTOS, FatFS, LwIP, and PolarSSL. Horizontal interaction among the components in this layer is performed directly by calling the feature APIs, while vertical interaction with low-level drivers is managed by specific callbacks and static macros implemented in the library system call interface. For example, FatFs implements the disk I/O driver to access a microSD drive or USB Mass Storage Class.
- Examples based on the middleware components: each middleware component comes with one or more examples (or applications) showing how to use it. Integration examples that use several middleware components are provided as well.

Level 2: This level is a single layer with a global, real-time and graphical demonstration based on the middleware service layer, the low level abstraction layer and basic peripheral usage applications for board-based functions.

3 X-CUBE-GNSS1 software expansion for STM32Cube

3.1 Overview

X-CUBE-GNSS1 is a software package that expands the functionality provided by STM32Cube .

The key features of the package are:

- Complete software to build applications using Teseo-LIV3F GNSS device
- Middleware for the NMEA protocol and for Assisted GNSS (A-GNSS) support
- FreeRTOS task scheduling to ensure better asynchronous message parsing
- Easy portability across different MCU families, thanks to STM32Cube
- Sample application to transmit GNSS data to a PC and for A-GNSS support
- Free, user-friendly license terms

The package includes sample applications that the developer can use to start experimenting with the code. A Java application to update the Teseo-LIV3F and the STM32 Nucleo board firmware is included in the package.

3.2 Architecture

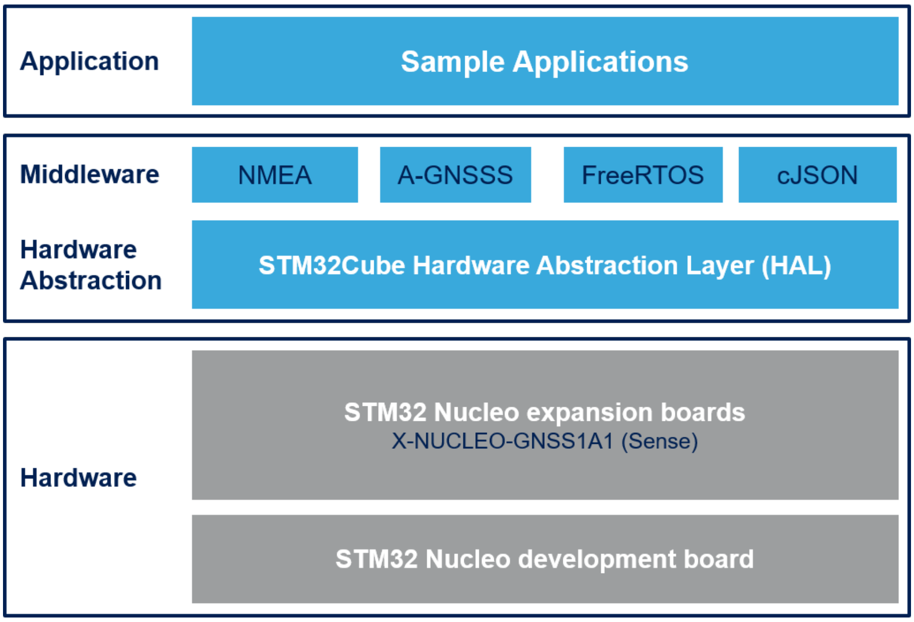
This software is a fully compliant expansion of STM32Cube architecture for the development of applications using the ST Teseo-LIV3F GNSS module.

The software is based on the STM32CubeHAL hardware abstraction layer for the STM32 microcontroller. The package extends STM32Cube by providing a board support package (BSP) for the GNSS expansion board and some middleware components for NMEA protocol and A-GNSS support. The package is also based on FreeRTOS library to exploit the benefits of task scheduling for better asynchronous message parsing.

The software layers used by the application software to access and use the GNSS expansion board are:

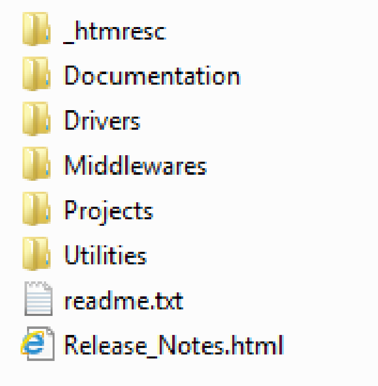
- **STM32Cube HAL layer**: consists of simple, generic and multi-instance APIs (application programming interfaces) which interact with the upper layer applications, libraries and stacks. These generic and extension APIs are based on a common framework so that overlying layers like middleware can function without requiring specific microcontroller unit (MCU) hardware information. This structure improves library code reusability and guarantees easy portability across other devices.
- **Board support package (BSP) layer**: provides software support for the STM32 Nucleo board peripherals, excluding the MCU. These specific APIs provide a programming interface for certain board specific peripherals like LEDs, user buttons, etc., and can also be used to fetch individual board version information. It also provides support for initializing, configuring and reading data.

Figure 2. X-CUBE-GNSS1 software architecture



3.3 Folder structure

Figure 3. X-CUBE-GNSS1 package folder structure


_htmresc
Documentation
Drivers
Middlewares
Projects
Utilities
readme.txt
Release_Notes.html

The following folders are included in the software package:

- **Documentation**: contains a compiled HTML file generated from the source code, detailing the software components and APIs.
- **Drivers**: contains the HAL drivers, the board specific drivers for each supported board or hardware platform (including the on-board components) and the CMSIS vendor-independent hardware abstraction layer for the ARM® Cortex®-M processor series.
- **Middlewares**: contains a library for the NMEA protocol support (e.g., for NMEA message parsing); FreeRTOS library for task scheduling management; A-GNSS library for Assisted GNSS support; cJSON library for JSON message parsing.
- **Projects**: contains the sample applications used to access the GNSS data provided for the NUCLEO-F401RE, NUCLEO-L476RG, NUCLEO-L073RZ platforms and a sample application for A-GNSS provided for the B-L475E-IOT01A Discovery kit for IoT node. All applications are available under three development environments: IAR Embedded Workbench for ARM (IAR-EWARM), RealView Microcontroller Development Kit (MDK-ARM) and System Workbench for STM32 (SW4STM32).
- **Utilities**: contains a “PC_Software” folder with a Java utility to update the Teseo-LIV3F device firmware.

3.4 APIs

Detailed technical information about the APIs available to the user can be found in a compiled HTML file located inside the “Documentation” folder of the software package where all the functions and parameters are fully described.

3.5 GetPos sample application

An example application using the X-NUCLEO-GNSS1A1 expansion board with a NUCLEO-F401RE or NUCLEO-L476RG board is included in the package (in the folder ROOT_DIR\Projects\Multi\Applications\GetPos). Ready to be built projects are available for multiple IDEs.

In this application, real-time GNSS data received by the Teseo-LIV3F device can be displayed through a serial connection and a serial terminal on a PC.

The GetPos application is built on top of the FreeRTOS support introducing a task (consumer), to parse the messages (enqueued in a shared queue) coming from the Teseo-LIV3F device, and a task (listener) to parse commands coming from the serial terminal.

The sample application also shows three advanced features supported by the Teseo-LIV3F device:

- **Geofencing**: allows the Teseo-LIV3F receiver to raise an NMEA message when the resolved GNSS position is close to enter or exit from a specific circle.
- **Odometer**: provides information on the traveled distance using only the resolved GNSS position.
- **Data Logging**: allows the Teseo-LIV3F receiver to save the resolved GNSS position on the local Flash memory to be retrieved on demand from the host.

The Teseo-LIV3F device sends the received GNSS data via UART (or I²C) to the STM32 microcontroller on the STM32 Nucleo board according to the NMEA 0183 Version 4.0 protocol.

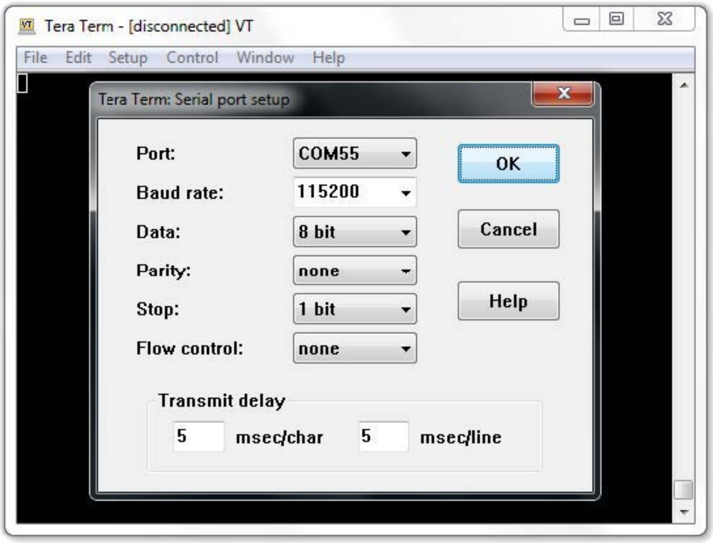
At compile time, the user can choose the interface to receive data (setting the macro configUSE_I2C in file gnss_app_cfg.h) and also enable a specific feature (setting the corresponding macro in the file gnss_app_cfg.h).

3.5.1 GetPos usage procedure

The following instructions show how to use the GetPos sample application to return information relating to a GNSS position.

Step 1. Open a serial connection between the STM32 Nucleo and the X-NUCLEO-GNSS1A1 board and your PC with the correct COM port for your PC and the following remaining parameters:

Figure 4. GetPos: TeraTerm serial COM port setup



Tera Term - [disconnected] VT

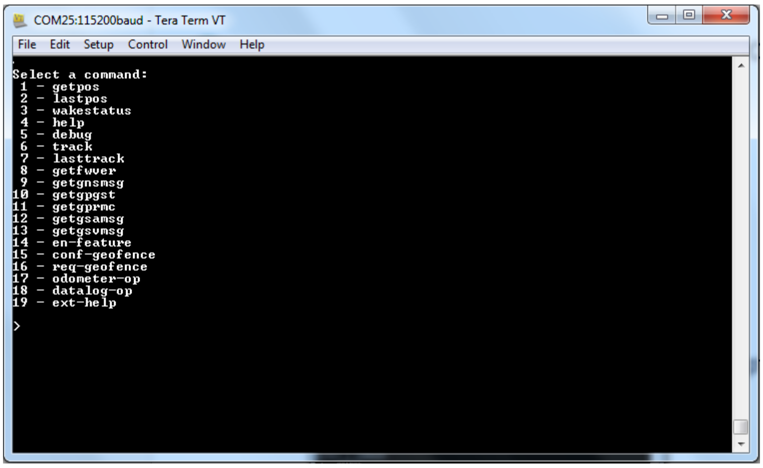
Tera Term: Serial port setup

Port: COM55
Baud rate: 115200
Data: 8 bit
Parity: none
Stop: 1 bit
Flow control: none

Transmit delay
5 msec/char
5 msec/line

Step 2. Push the Reset button to show the menu options in the terminal window.

Figure 5. GetPos: menu



COM25:115200baud - Tera Term VT
File Edit Setup Control Window Help

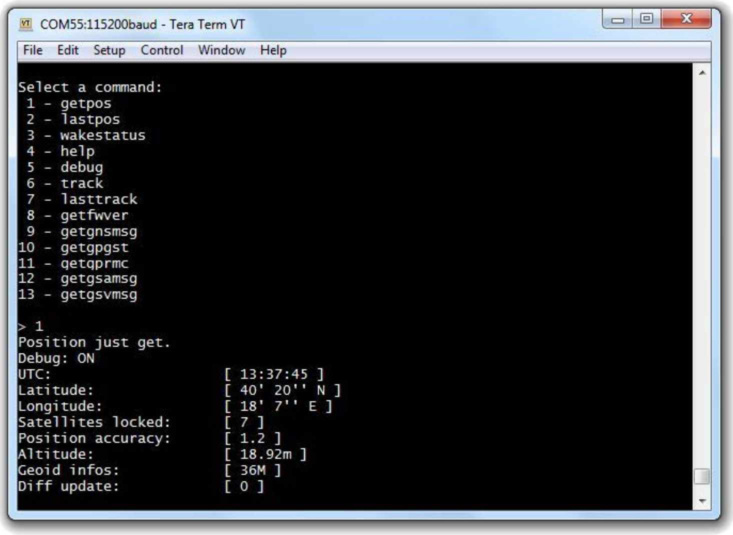
```
Select a command:
 1 - getpos
 2 - lastpos
 3 - wakestatus
 4 - help
 5 - debug
 6 - track
 7 - lasttrack
 8 - getfwver
 9 - getgnsmsg
10 - getgpgst
11 - getgprmc
12 - getgsamsg
13 - getgsvmsg
14 - en-feature
15 - conf-geofence
16 - req-geofence
17 - odometer-op
18 - datalog-op
19 - ext-help

>
```

Step 3. Select from the different options to obtain human readable information related to the acquired GNSS position, the satellites in view, the active satellites and so on.

The figure below relates to menu selection option 1

Figure 6. GetPos: menu option 1



COM55:115200baud - Tera Term VT
File Edit Setup Control Window Help

```
Select a command:
 1 - getpos
 2 - lastpos
 3 - wakestatus
 4 - help
 5 - debug
 6 - track
 7 - lasttrack
 8 - getfwver
 9 - getgnsmsg
10 - getgpgst
11 - getgprmc
12 - getgsamsg
13 - getgsvmsg

> 1
Position just get.
Debug: ON
UTC:                     [ 13:37:45 ]
Latitude:                [ 40' 20'' N ]
Longitude:               [ 18' 7'' E ]
Satellites locked:       [ 7 ]
Position accuracy:       [ 1.2 ]
Altitude:                [ 18.92m ]
Geoid infos:             [ 36M ]
Diff update:             [ 0 ]
```

3.5.2 SimOSGetPos

SimOsGetPos is a particular case of the GetPos application described in this section, included in the package folder ROOT_DIR\Projects\Multi\Applications\SimOSGetPos.

It shows how real-time GNSS data ($GPGGA) received by the Teseo-LIV3F device can be displayed, through a serial connection and a serial terminal, on a PC.

This application is tailored for STM32 Nucleo L0 family and excludes the FreeRTOS support.

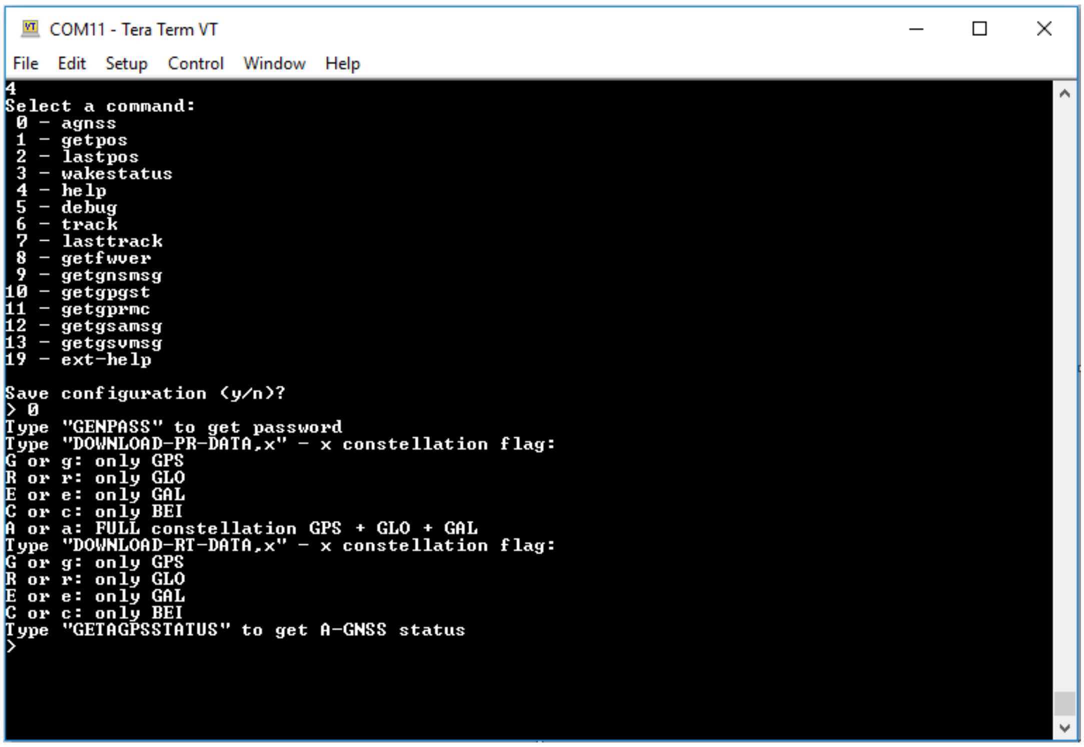
3.5.3 A_GetPos

A_GetPos is an extension of the GetPos application including the Assisted GNSS feature (both predictive and real-time) supported by the Teseo-LIV3F GNSS device.

The application has to access the Internet and is customized for B-L475E-IOT01A Discovery kit for IoT node providing the Wi-Fi connectivity (for further details on Assisted GNSS, refer to AN5160 and Section 3 of UM2399 freely available at www.st.com).

This application can be found in the package folder ROOT_DIR\Projects\Multi\Applications\A_GetPos

Figure 7. A_GetPos menu option 0 and related sub-options



```
4
Select a command:
 0 - agnss
 1 - getpos
 2 - lastpos
 3 - wakestatus
 4 - help
 5 - debug
 6 - track
 7 - lasttrack
 8 - getfwver
 9 - getgnsmsg
10 - getgpgst
11 - getgprmc
12 - getgsamsg
13 - getgsvmsg
19 - ext-help

Save configuration (y/n)?
> 0
Type "GENPASS" to get password
Type "DOWNLOAD-PR-DATA,x" - x constellation flag:
G or g: only GPS
R or r: only GLO
E or e: only GAL
C or c: only BEI
A or a: FULL constellation GPS + GLO + GAL
Type "DOWNLOAD-RT-DATA,x" - x constellation flag:
G or g: only GPS
R or r: only GLO
E or e: only GAL
C or c: only BEI
Type "GETAGPSSTATUS" to get A-GNSS status
>
```

3.6 Virtual_COM_Port sample application

Virtual_COM_Port is the application to be loaded in order to use the ST TESEO-SUITE PC software tool (available at , together with a **Quick Training Guide** on how to use it) for managing, configuring and evaluating the Teseo GNSS device.

It is included in the package folder *ROOT_DIR\Projects\Multi\Applications\Virtual_COM_Port*. Ready-to-build projects are available for multiple IDEs.

Using this application, the real-time GNSS data received by the Teseo-LIV3F device are displayed through a serial connection and a serial terminal on a PC.

3.7 GNSS FW Upgrader utility

The X-CUBE-GNSS1 software expansion for STM32Cube contains a Java utility to upgrade the Teseo-LIV3F GNSS device firmware to the latest available version. This utility is available in the ROOT_DIR\Utilities \PC_Software folder.

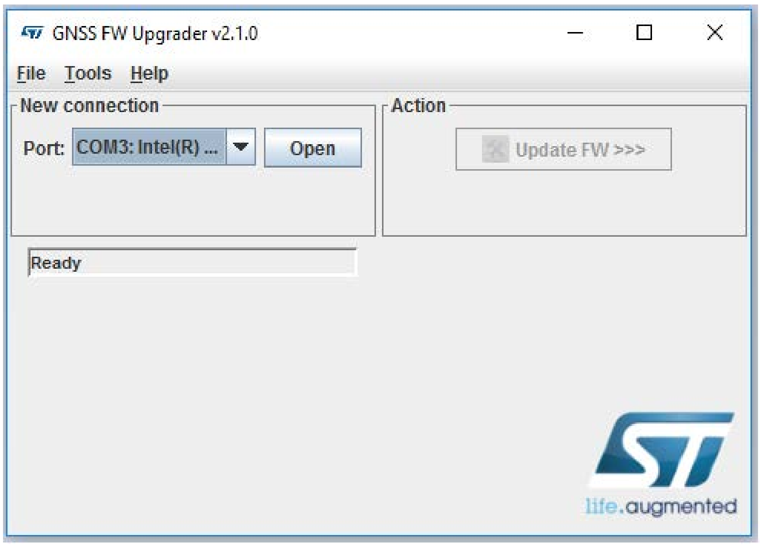
Step 1. Connect the X-NUCLEO-GNSS1A1 expansion board to the STM32 Nucleo development board.

Step 2. Copy the FW_Updater firmware to the Nucleo drive.
You will find the firmware in the ROOT_DIR\Projects\Multi\Applications\FW_Updater folder on your PC.

Step 3. Run the GNSS Upgrader utility

Step 4. Establish connection with the STM32 Nucleo board by selecting the serial port and clicking the Open button.

Figure 8. **GNSS FW Upgrader: Open button**



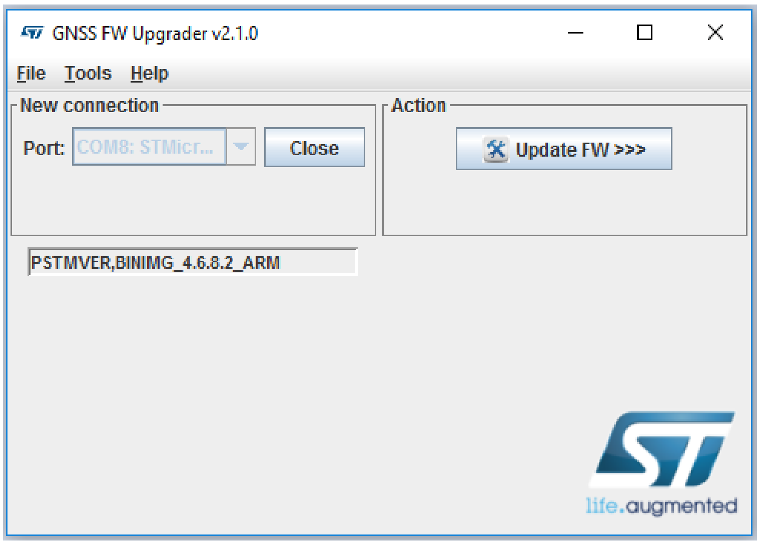
Ready

After a few seconds, the Teseo-LIV3F current firmware version is displayed in the GUI

A pop-up window indicates whether the firmware is the latest available version.

Step 5. If the current firmware is not the latest version, click on the **Update FW >>>** button to start the upgrade process.

Figure 9. **GNSS FW Upgrader: upgrade process**



PSTMVER,BINIMG_4.6.8.2_ARM

Step 6. Wait for the upgrade to complete

A new pop-up should now confirm that the firmware is up to date.

The GUI will show the new firmware version.

4 System setup guide

4.1 Hardware description

4.1.1 STM32 Nucleo platform

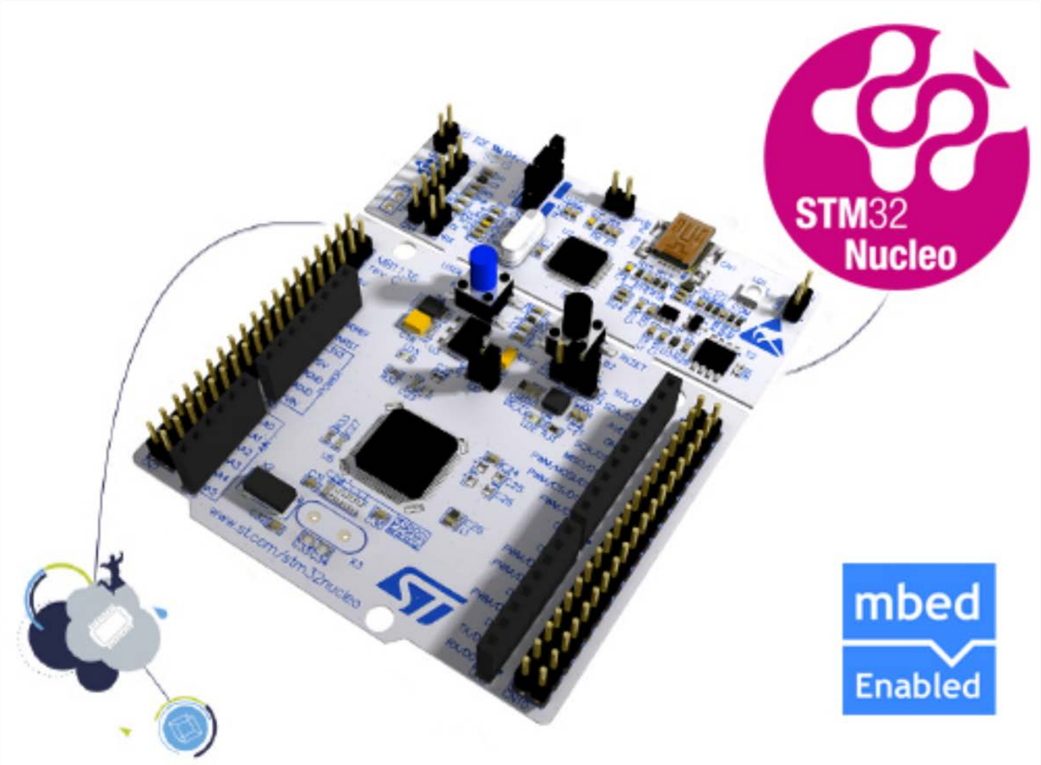
STM32 Nucleo development boards provide an affordable and flexible way for users to test solutions and build prototypes with any STM32 microcontroller line.

The Arduino™ connectivity support and ST morpho connectors make it easy to expand the functionality of the STM32 Nucleo open development platform with a wide range of specialized expansion boards to choose from.

The STM32 Nucleo board does not require separate probes as it integrates the ST-LINK/V2-1 debugger/ programmer.

The STM32 Nucleo board comes with the comprehensive STM32 software HAL library together with various packaged software examples.

Figure 10. STM32 Nucleo board



Information regarding the STM32 Nucleo board is available at www.st.com/stm32nucleo

4.1.2 X-NUCLEO-GNSS1A1 expansion board

The X-NUCLEO-GNSS1A1 expansion board is based on the Teseo-LIV3F tiny GNSS module.

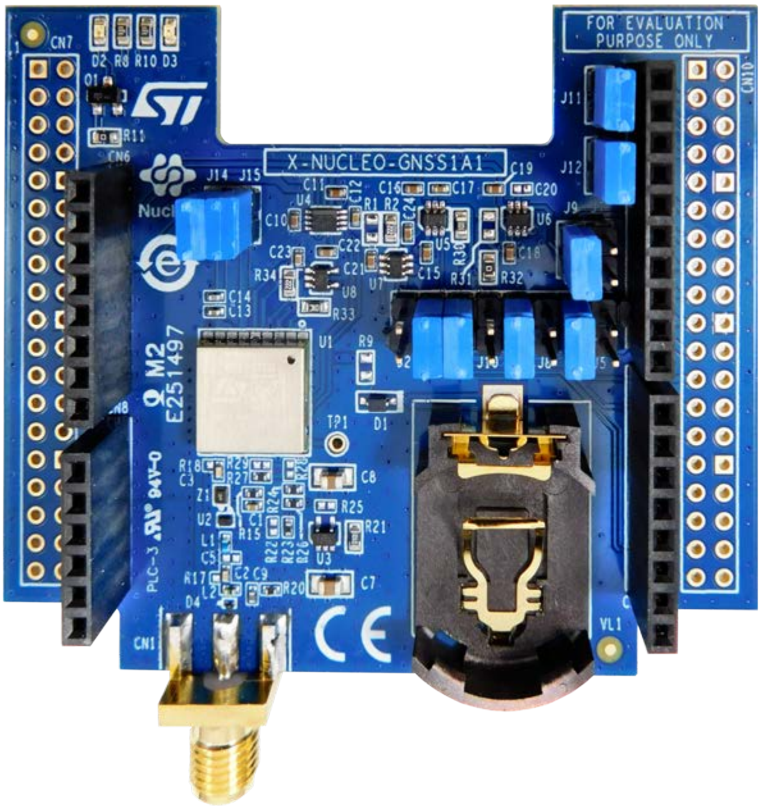
It represents an affordable, easy-to-use, global navigation satellite system (GNSS) module, embedding a TeseoIII single die standalone positioning receiver IC, usable in different configurations in your STM32 Nucleo project.

The Teseo-LIV3F is a compact (9.7x10.1 mm) module that provides superior accuracy thanks to the on-board 26 MHz temperature compensated crystal oscillator (TCXO) and a reduced time-to-first fix (TTFF) with its dedicated 32 KHz real-time clock (RTC) oscillator.

The Teseo-LIV3F module runs the GNSS firmware (X-CUBE-GNSS1) to perform all GNSS operations including acquisition, tracking, navigation and data output without external memory support.

The X-NUCLEO-GNSS1A1 expansion board is compatible with the Arduino™ UNO R3 connector and the ST morpho connector, so it can be plugged to the STM32 Nucleo development board and stacked with additional STM32 Nucleo expansion boards.

Figure 11. **X-NUCLEO-GNSS1A1 expansion board**



4.2 Hardware setup

The following hardware components are needed:

1. One STM32 Nucleo development platform (suggested order code: NUCLEO-F401RE, NUCLEO-L476RG or NUCLEO-L073RZ)
2. One Discovery kit for IoT node (order code: B-L475E-IOT01A)
3. One GNSS expansion board (order code: X-NUCLEO-GNSS1A1)
4. One GPS/GLONASS/Beidou antenna to be connected to the X-NUCLEO-GNSS1A1 (bundled with the GNSS expansion board)
5. One USB type A to Mini-B USB cable to connect the STM32 Nucleo to the PC

For the X-NUCLEO-GNSS1A1 proper operations, the following jumper settings must be used:

- J2 open
- J3 closed
- J4 closed
- J5 open
- J6 closed
- J7 closed
- J8 open
- J9 closed
- J10 open
- J11 closed
- J12 closed
- J13 closed

- J14 closed
- J15 closed

4.3 Software setup

The following software components are required to set up a suitable development environment for creating applications for the STM32 Nucleo equipped with the X-NUCLEO-GNSS1A1 expansion board:

- X-CUBE-GNSS1: an expansion for STM32Cube dedicated to GNSS applications development. The X-CUBE-GNSS1 firmware and related documentation is available on www.st.com.
- Development tool-chain and Compiler: the STM32Cube expansion software supports the three following environments:
  - IAR Embedded Workbench for ARM® (IAR-EWARM) toolchain + ST-LINK
  - RealView Microcontroller Development Kit (MDK-ARM-STM32) toolchain + ST-LINK
  - System Workbench for STM32 (SW4STM32) + ST-LINK

4.4 System setup guide

This section describes how to setup different hardware parts before developing and executing an application on the STM32 Nucleo board with the GNSS expansion board.

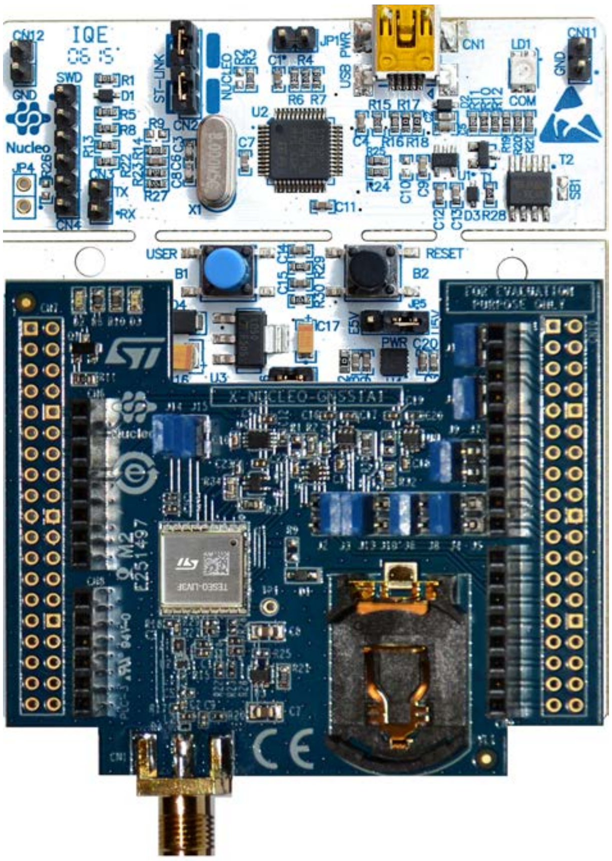
4.4.1 STM32 Nucleo and GNSS expansion board setup

The STM32 Nucleo board integrates the ST-LINK/V2-1 debugger/programmer.

You can download the appropriate ST-LINK/V2-1 USB driver (STSW-LINK009) from www.st.com.

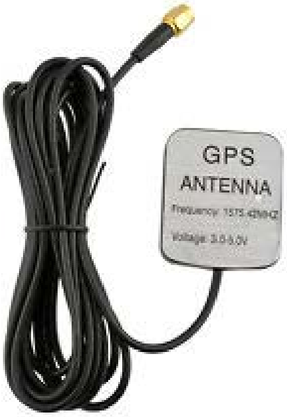
The GNSS expansion board X-NUCLEO-GNSS1A1 is easily connected to the STM32 Nucleo development through the Arduino UNO R3 extension connector and can interface with the external STM32 microcontroller on the STM32 Nucleo board either via UART or Inter-Integrated Circuit (I²C) channels.

Figure 12. X-NUCLEO-GNSS1A1 expansion board connected to an STM32 Nucleo board



The GPS/GLONASS/Beidou antenna included with the X-NUCLEO-GNSS1A1 expansion board must be connected to the antenna connector on the expansion board.

Figure 13. **GPS/GLONASS/Beidou antenna**



4.4.2 GNSS Firmware Upgrader utility setup

The GNSS FW Upgrader Java utility included in the software package is a graphical user interface that can be used to upgrade the Teseo-LIV3F on the X-NUCLEO-GNSS1A1 to the latest firmware version.

It has no particular minimum requirements, you can use any Linux, Microsoft or OSX PC with:

- Java Runtime Environment (JRE) 6+
- 1 x USB port

To use the Firmware Updater Java utility, make sure you have correctly set up your hardware and software.

The utility can be launched by simply clicking twice on the FWUPG.jar file, located in the “Utilities\PC_Software \FirmwareUpdaterTool” folder.

Revision history

Table 2. Document revision history

Date	Version	Changes
06-Dec-2017	1	Initial release.
25-Jan-2018	2	Updated Figure 5 GetPos: menu. Added Section 3.6 Virtual_COM_Port sample application.
11-May-2018	3	Updated Introduction, Section 3.1 Overview, Section 3.2 Architecture, Section 3.3 Folder structure, Section 3.5 GetPos sample application. Added Section 3.5.2 SimOSGetPos.
15-Oct-2018	4	Updated Introduction, Table 1. List of acronyms, Section 3.1 Overview, Figure 2. X-CUBE-GNSS1 software architecture, Section 3.3 Folder structure, Figure 8. GNSS FW Upgrader: Open button, Figure 9. GNSS FW Upgrader: upgrade process, Figure 11. X-NUCLEO-GNSS1A1 expansion board and Figure 12. X-NUCLEO-GNSS1A1 expansion board connected to an STM32 Nucleo board. Added Section 3.5.3 A_GetPos.

Contents

List of tables

List of figures