life.augmented

High-speed datalogging for motor control & sensors

FP-IND-DATALOGMC
Quick Start Guide
STM32Cube Function Pack
Version 1.1 (Jan ’24)

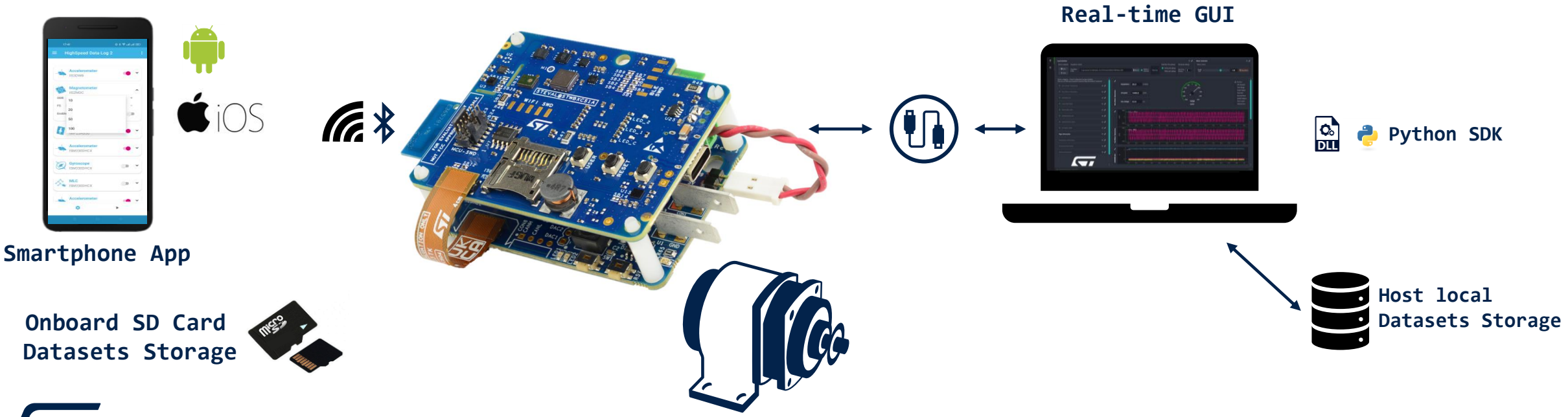
Agenda

life.augmented

1- Application Overview

Datalogging and labeling of heterogeneous data

Comprehensive solution to manage the acquisition of heterogeneous datasets from sensors and motor, including precise timestamps and events tags



FP-IND-DATALOGMC is online at www.st.com

life.augmented

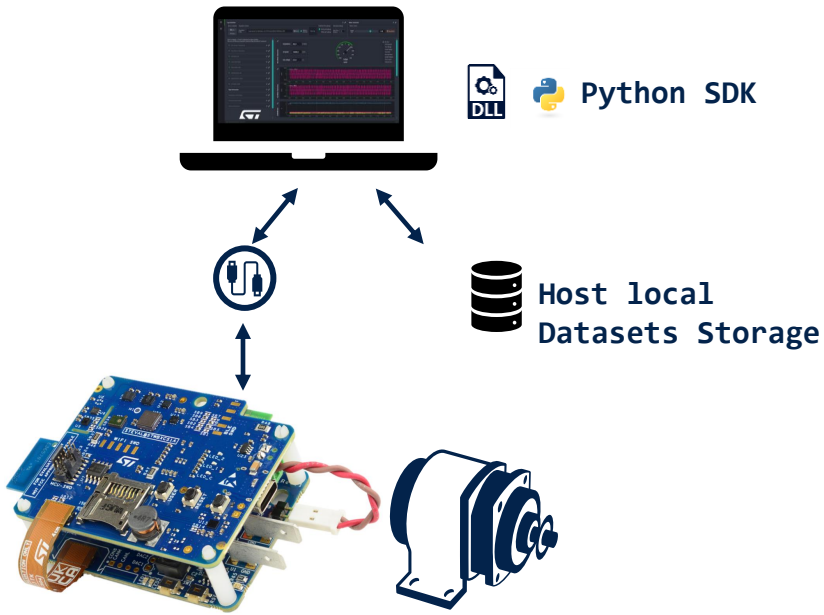
Motor Control High Speed Data logger (*in short: DATALOGMC*)

Application overview

DATALOGMC comes with **two operating modes**:

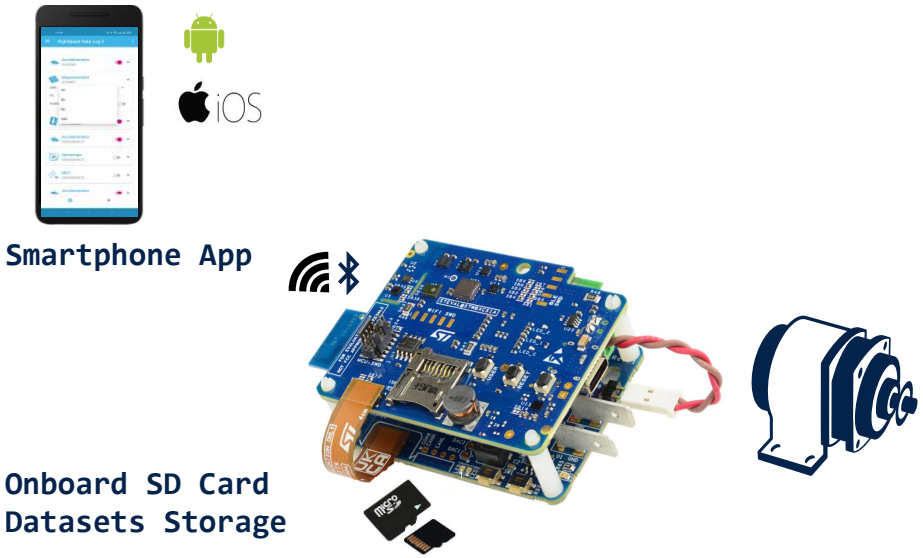
DATALOGMC via USB

It allows user to control the motor, acquire, stream and plot in real-time via Python Application GUI sensors and motor control data.



DATALOGMC via Bluetooth Low Energy (BLE)

It allows user to control the motor, acquire, and save sensors and motor control data to SD Card.



life.augmented

DATALOGMC via USB

Application overview

DATALOGMC via USB allow user to control the motor, acquire, stream and plot in real-time via Python Application GUI sensors and motor control data.

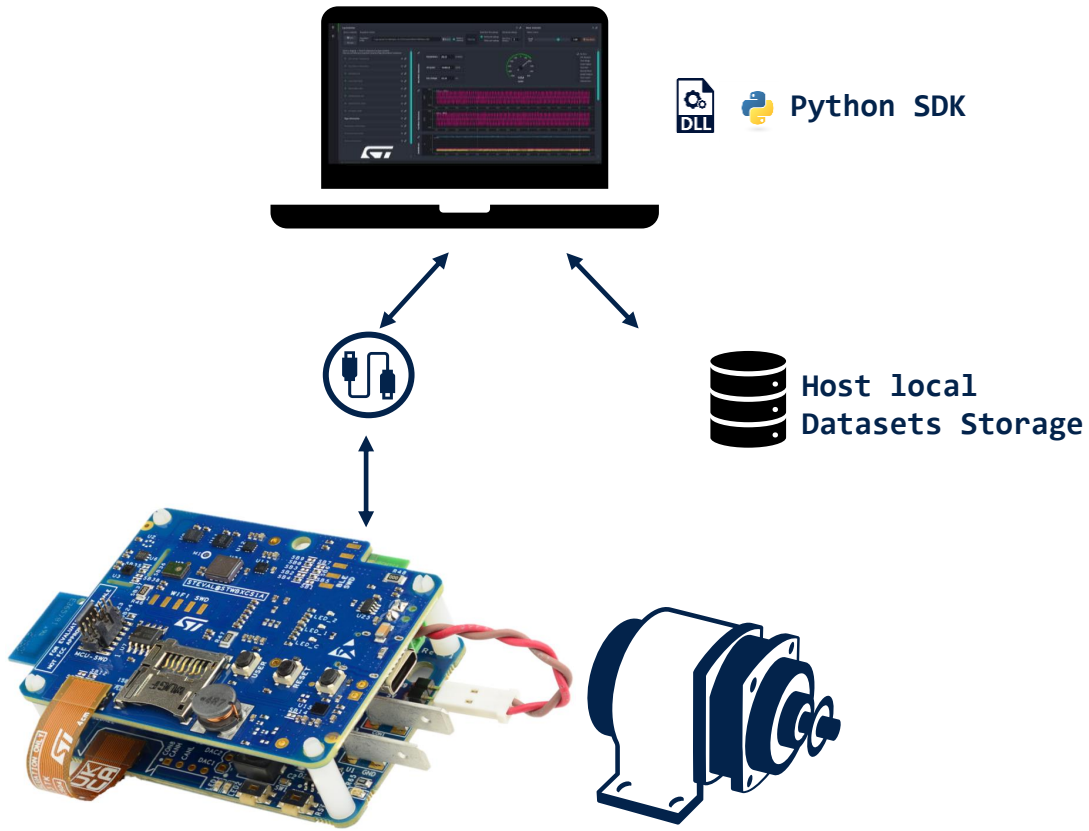
In order to run DATALOGMC via USB you need:

- STEVAL-STWINBX1
- EVLSPIN32G4-ACT
- STEVAL-FLTCB04

  (4cm flex cable included with EVLSPIN32G4-ACT)
- 3-phase brushless motor (250 W max – not included in the kit)
- Power supply (48 V max)
- Laptop/PC with Windows 7, 8 or 10
- USB Type-C cable

For boards programming only:

- STLINK-V3MINIE, STLINK-V3SET, or ST-LINK/V2 + adapter



life.augmented

DATALOGMC via BLE + SD Card

Application overview

DATALOGMC via BLE allow user to control the motor, acquire, and save sensors and motor control data to the SD Card.

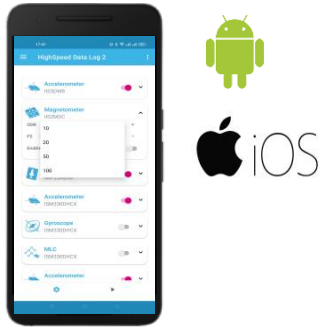
In order to run DATALOGMC via BLE you need:

- STEVAL-STWINBX1
- EVLSPIN32G4-ACT
- STEVAL-FLTCB04

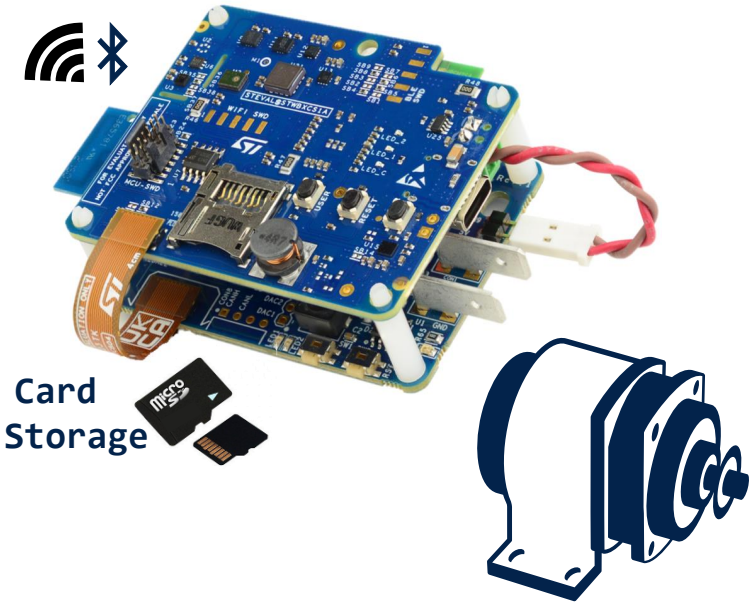
  (4cm flex cable included with EVLSPIN32G4-ACT)
- 3-phase brushless motor (250 W max – not included in the kit)
- Power supply (48 V max)
- STBLESensor App for Android or iOS
- micro-SD card

For boards programming only:

- STLINK-V3MINIE, STLINK-V3SET, or ST-LINK/V2 + adapter



Smartphone App



Onboard SD Card
Datasets Storage

Setup & Application Examples

Software and other prerequisites

- **STM32CubeProgrammer Software**
  - Download and install STM32CubeProgrammer
- **STM32Cube initialization code generator**
  - Download and install STM32CubeMX
- **Integrated Development Environment for STM32**
  - Download and install one among the supported IDEs: STM32CubeIDE, Keil, IAR
- **STM32 Motor Control Software Development Kit**
  - Download and install X-CUBE-MCSDK-6
- **DATALOGMC**
  - Download the FP-IND-DATALOGMC package from www.st.com, copy the .zip file contents into a folder on your PC.
    The package contains binaries and source code with project files (STM32CubeIDE, Keil, IAR)
- **ST BLESensor App**
  - Download and install ST BLESensor App (for both Android and iOS – v5.2 and above)
- **Python3 (>=3.10)**
  - To save, plot and elaborate data, Python utility scripts are available

DATALOGMC is **not** the default firmware on STWIN.box.
To update the firmware, download the function pack or follow the instructions for Fast FOTA

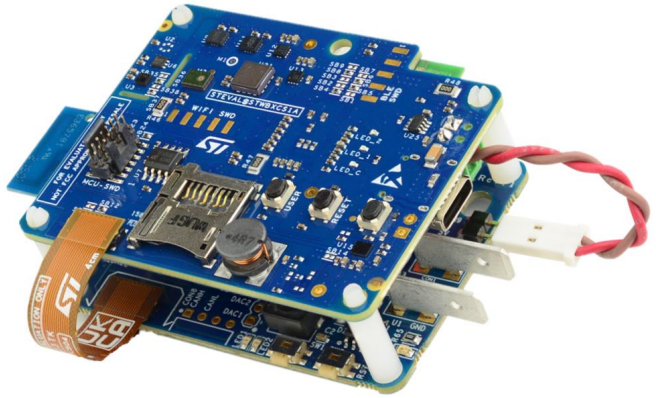
life.augmented

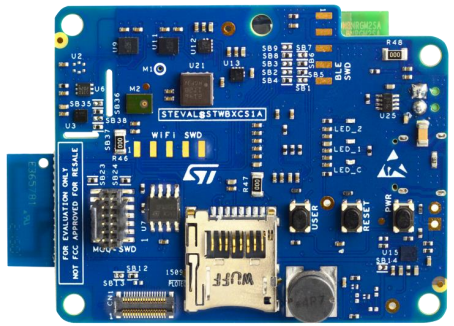
2- Hardware and Software Overview

Hardware Overview

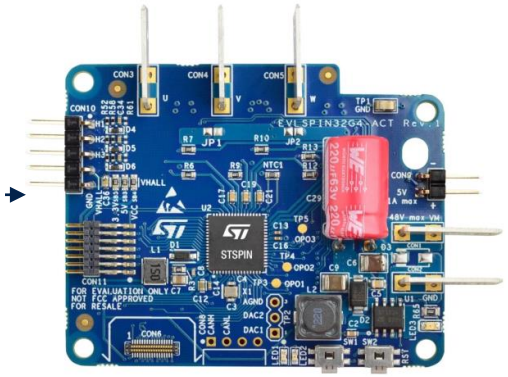
FP-IND-DATALOGMC requires a STEVAL-STWINBX1, an EVLSPIN32G4-ACT, a 3-phase brushless motor, and a power supply.

STEVAL-STWINBX1





EVLSPIN32G4-ACT



3-phase brushless motor

Power supply

STWIN.box development kit - STEVAL-STWINBX1

Hardware Overview

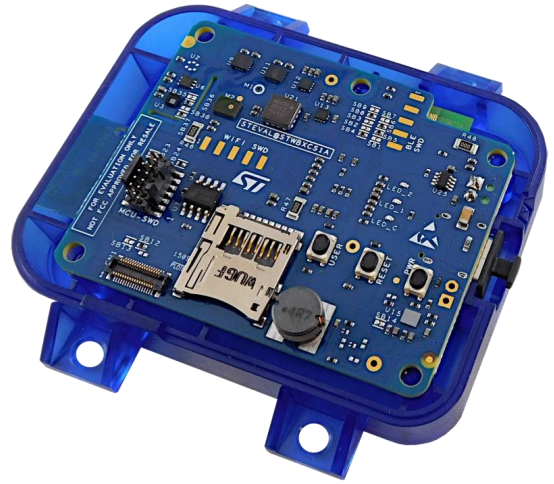
STWIN.box - SensorTile Wireless Industrial Node

The STWIN.box (STEVAL-STWINBX1) is a development kit and reference design that simplifies prototyping and testing of advanced industrial sensing applications in IoT contexts such as condition monitoring and predictive maintenance.
It is an evolution of the original STWIN kit (STEVAL-STWINKT1B) and features a higher mechanical accuracy in the measurement of vibrations, an improved robustness, an updated BoM to reflect the latest and best-in-class MCU and industrial sensors, and an easy-to-use interface for external add-ons.
The STWIN.box kit consists of an STWIN.box core system, a 480mAh LiPo battery, an adapter for the ST-LINK debugger (STEVAL-MKIGIBV4), a plastic case, an adapter board for DIL 24 sensors and a flexible cable.

Key Features

- Multi-sensing wireless platform for vibration monitoring and ultrasound detection
- Built around STWIN.box core system board with processing, sensing, connectivity, and expansion capabilities
- Ultra-low power Arm® Cortex®-M33 with FPU and TrustZone at 160 MHz, 2048 kBytes Flash memory (STM32U585AI)
- MicroSD card slot for standalone data logging applications
- On-board Bluetooth® low energy v5.0 wireless technology (BlueNRG-M2), Wi-Fi (EMW3080) and NFC (ST25DV04K)
- Wide range of industrial IoT sensors: Ultra-wide bandwidth (up to 6 kHz), low-noise, 3-axis digital vibration sensor (IIS3DWB), 3D accelerometer + 3D gyro iNEMO inertial measurement unit (ISM330DHCX) with Machine Learning Core, High-performance ultra-low-power 3-axis accelerometer for industrial applications (IIS2DLPC), Ultra-low power 3-axis magnetometer (IIS2MDC), Dual full-scale, 1.26 bar and 4 bar, absolute digital output barometer in full-mold package (ILPS22QS), Low-voltage, ultra low-power, 0.5°C accuracy I²C/SMBus 3.0 temperature sensor (STTS22H), Industrial grade digital MEMS microphone (IMP34DT05), Analog MEMS microphone with frequency response up to 80 kHz (IMP23ABSU)
- Expandable via a 34-pin FPC connector

STEVAL-STWINBX1



Latest info available at
www.st.com/stwinbox

life.augmented

STWIN.box development kit - STEVAL-STWINBX1

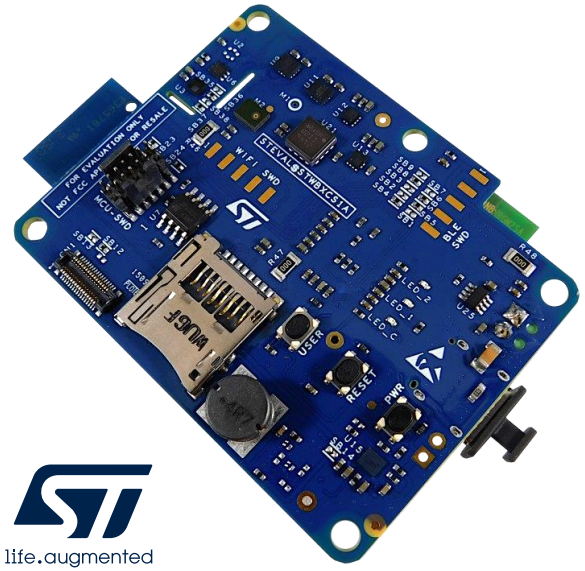
Hardware Overview

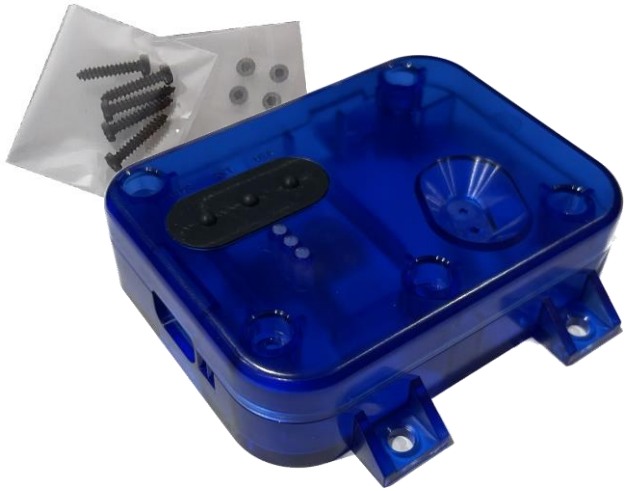
STWIN.box - SensorTile Wireless Industrial Node

The STEVAL-STWINBX1 development kit includes:

- The STEVAL-STWBXCS1 STWIN.box core system (main board);
- A plastic case with M3 bolts;
- A 480 mAh 3.7 V LiPo battery;
- The STEVAL-MKIGIBV4 ST-LINK adapter with programming cable;
- The STEVAL-C34DIL24 adapter board for DIL24 sensors with the STEVAL-FLTCB01 flexible cable.

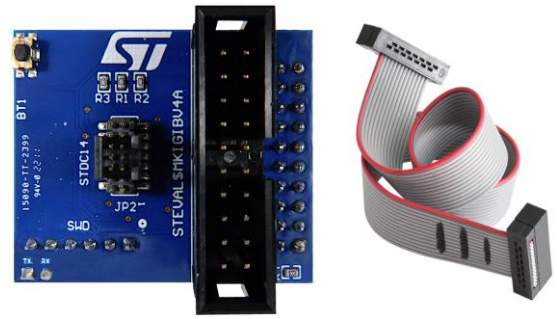
STEVAL-STWBXCS1
STWIN.box Core System





Plastic Case

Battery
LiPo-752535 - 480mAh



STEVAL-MKIGIBV4 + Cable
STLINK Adapter (V2, V2.1)

STEVAL-C34DIL24

STEVAL-FLTCB01
34 pin Flex cable

life.augmented

EVLSPIN32G4-ACT

Hardware Overview

STSPIN32G4 reference design for next generation smart actuators

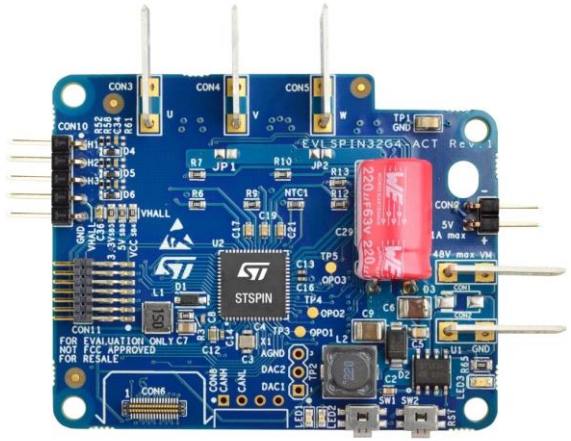
The EVLSPIN32G4-ACT is a reference design for implementing next generation smart actuators, based on the STSPIN32G4, a system-in-package integrating in a 9x9 mm VFQFPN package, a triple high-performance half-bridge gate driver with a rich set of programmable features and a mixed signal STM32G431 microcontroller.

The board is designed to drive three-phase brushless motors up to 5 Arms output current and 48 V supply input delivering a total power of 250 W in a very compact form factor (62 mm x 50 mm). Monitoring is available for the power stage in case of overheating, overvoltage, and overcurrent. The sensing of motor winding currents can be selected between three-shunt or single-shunt topology. The board is ready for FOC and 6-step control algorithms and can run in sensor-less and sensor-based mode using Hall sensors or quadrature encoder.

Key Features

- Power stage based on the STL60N10F7 power MOSFETs with output current up to 5 Arms and protected to overcurrent condition
- Bus voltage from 10 V to 48 V with dedicated monitoring
- STSPIN32G4, high performance three-phase motor controller with embedded STSPIN32G431 MCU
- Triple-shunt or single-shunt differential current sensing using embedded operational amplifiers
- Inputs for speed/position feedback by digital Hall sensors or incremental quadrature encoders
- Predisposition for CAN bus
- NTC sensor for power stage temperature monitoring
- Interface with STWIN.box and external sensor boards

EVLSPIN32G4-ACT



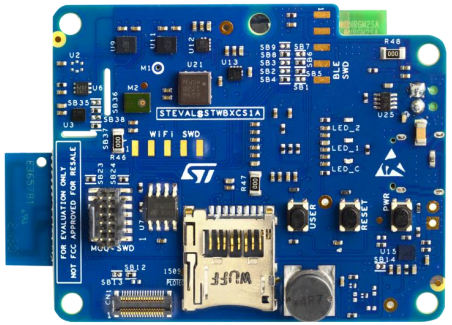
Latest info available at
www.st.com/evlspin32g4-act

life.augmented

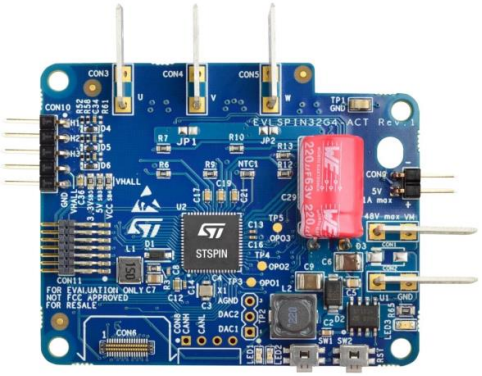
Software Overview

The **DATALOGMC** application requires **two different firmware to run**:

- **FP-IND-DATALOGMC** for STEVAL-STWINBX1 (see DATALOGMC FW)
- **EVLSPIN32G4-ACT firmware** generated by STM32 Motor Control Software Development Kit (MCSDK) (see How to program EVLSPIN32G4-ACT)



Motor Control Protocol



FP-IND-DATALOGMC firmware features:

- Motor Control Protocol controller
- USB and BLE communication
- Sensors acquisition and streaming
- SD Card management

MCSDK firmware features:

- Motor Control Protocol target
- Field Oriented Control

FP-IND-DATALOGMC

Software Overview

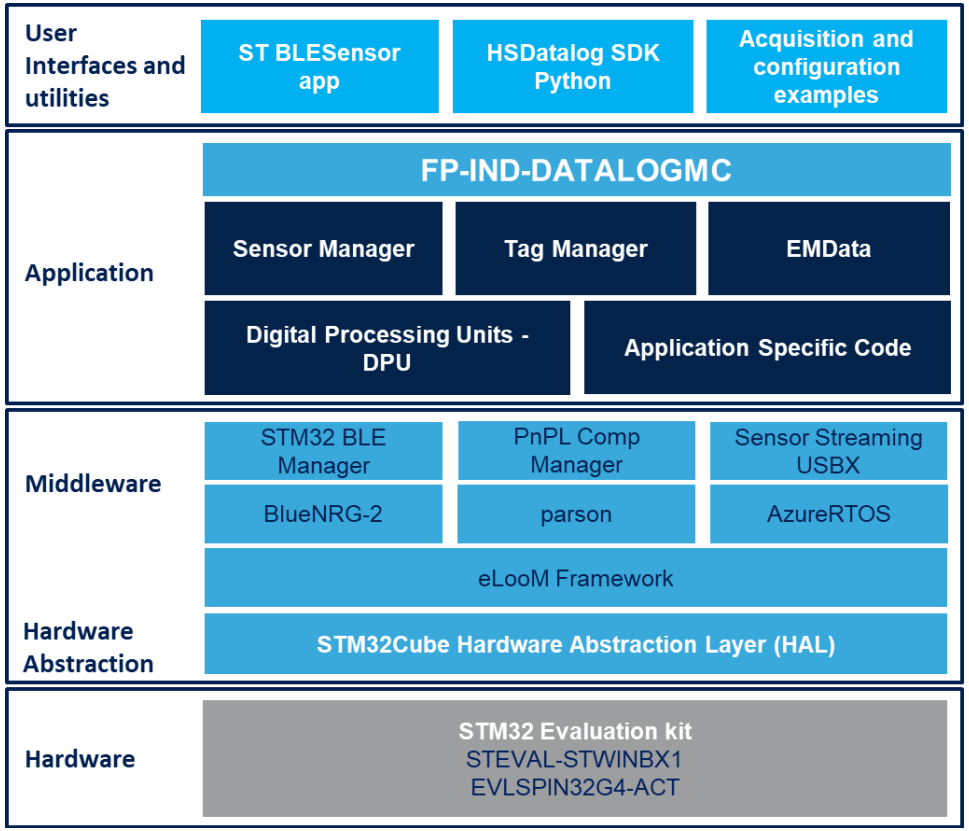
Software Description

The FP-IND-DATALOGMC function pack for STEVAL-STWINBX1 and EVLSPIN32G4-ACT is a powerful integrated toolkit for the next generation of smart actuators. It is derived from a FP-SNS-DATALOG2 function pack, and it allows the collection of heterogeneous data, combining STWIN.box sensor information with STSPIN32G4 motor control data and it provides a comprehensive view of the system's operational conditions. This enables both real-time monitoring and accurate
performance assessment.

Key features

- High data rate (up to 6 Mbit/s) data capture software suite:
  - Simultaneously log motor control telemetries and sensor data.
  - Python real-time control and data analysis
  - Dedicated Python SDK, ready-to-use for integration into any data science design flow
  - Compatible with ST BLESensor app for system setup and real-time control
  - Synchronized timestamping and labeling mechanisms common to all sensors and motor data
- Motor Control Protocol implementation to interact with EVLSPIN32G4-ACT evaluation board, programmed as through MCSDK (X-CUBE-MCSDK-6)
- AzureRTOS: ThreadX, FileX, USBX
- Firmware modular examples based on eLooM (embedded Light object-oriented fraMework for STM32) to enable code reusability at application level
- Free, user-friendly license terms

Overall Software Architecture



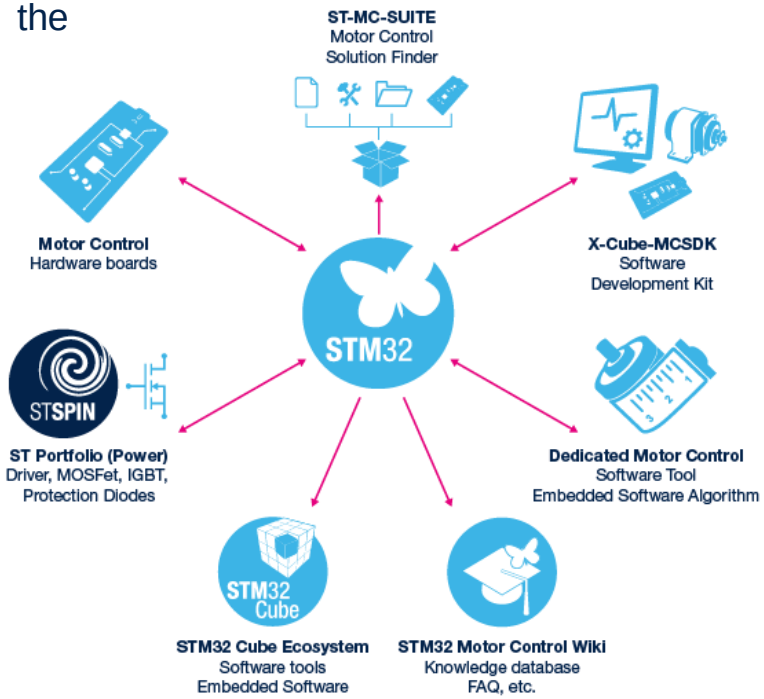
life.augmented

Software Description

STM32 MCSDK (motor control software development kit) firmware includes the permanent magnet synchronous motor (PMSM) firmware library (FOC and 6STEP control) and the STM32 motor control workbench (to configure the firmware library parameters), with its graphical user interface (GUI).
STM32 Motor Control Workbench is a PC software that reduces the design effort and time needed for the firmware configuration.
The user generates a project file through the GUI and initializes the library according to the application needs. Some algorithm variables can be monitored and changed in real time.

Key features

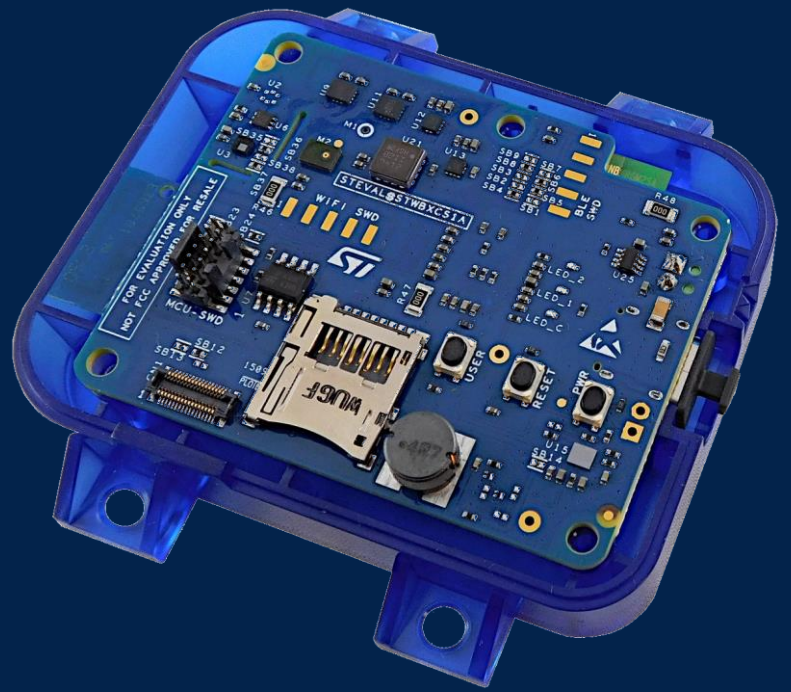
- Single/dual simultaneous field-oriented control (FOC)
- Motor profiler and one-touch tuning for a fast startup of unknown motors
- Simplified firmware architecture based on the STM32Cube HAL/LL libraries
- Current reading topologies supported:
  - 1 shunt resistor
  - 3 shunt resistors
  - 2 ICS (isolated current sensor)
- Speed/position sensors (encoder and Hall) and sensorless operation
- Speed and torque control
- MTPA (maximum torque per ampere), flux weakening, feed forward, and start-on-the-fly
- Full customization and real time communication through Motor Control Protocol



3- Hardware and Software Setup

Setup steps

3.1- How to program STWIN.box (DATALOGMC FW, Controller board)



life.augmented

STEVAL-STWINBX1 FW Setup

1

www.st.com

Select:

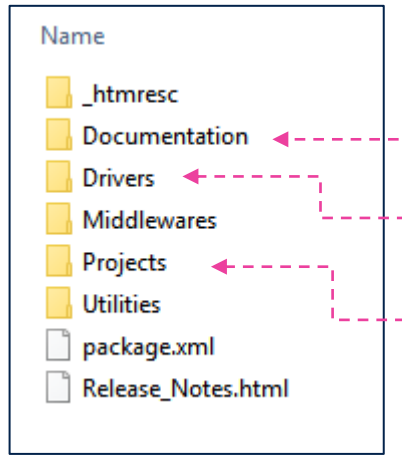
2

FP-IND-DATALOGMC

Download & unpack

3

Package structure



Docs

BSP, HAL drivers

Sample applications

4

5

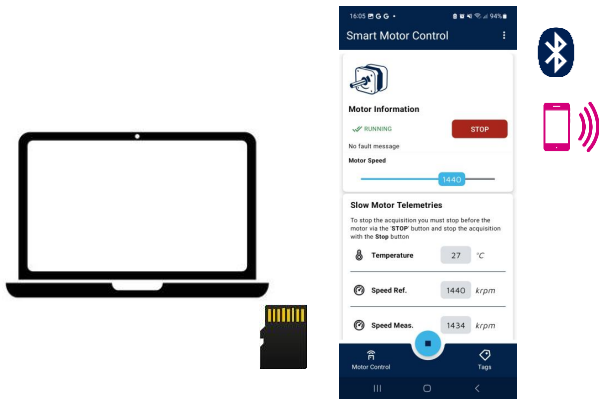
KEIL Tools by ARM

IAR SYSTEMS

STM32 CubeIDE

Use the pre-compiled binaries or re-compile the code customizing your device configuration

6



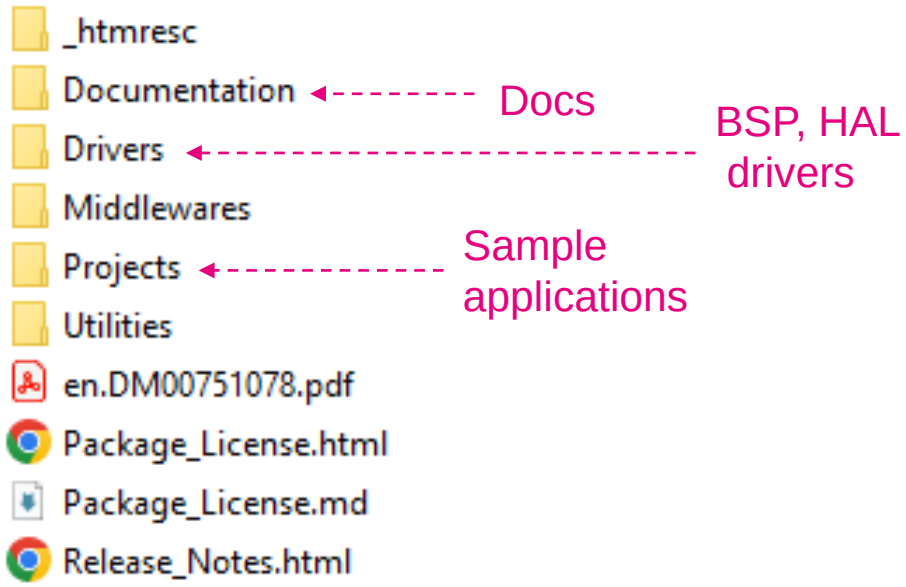
Visualize log of sensors data and control the device

STEVAL-STWINBX1 Firmware Setup

To configure and flash properly the STEVAL-STWINBX1 follow the next steps:

1. Download the FP-IND-DATALOGMC function pack from www.st.com

2. Unpack the function pack folder and navigate into STM32CubeFunctionPack_DATALOGMC_V1.0.0.



life.augmented

STEVAL-STWINBX1 Firmware Setup

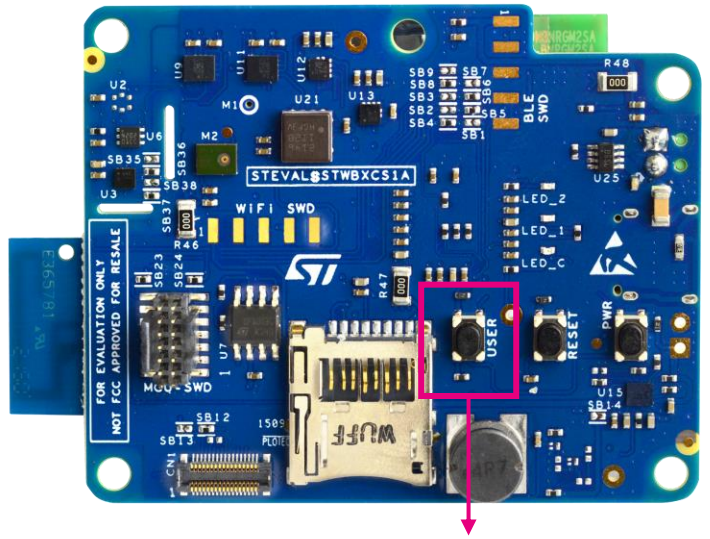
3. User can choose one of the three procedures described below to program the STEVAL-STWINBX1 board with the DATALOGMC firmware:
   - Firmware update via USB
   - Firmware update via STLINK
   - Firmware update via BLE (FOTA)

Firmware update via USB

STEVAL-STWINBX1 can be reprogrammed via USB using the STM32CubeProgrammer "USB mode".

To enter "Firmware upgrade" mode you must follow the procedure below:

- Unplug the core system board.
- Press the USER button.
- While keeping the button pressed, connect the USB cable to the PC.
- Now the board is in DFU mode. Open STM32CubeProgrammer, select the binary located under:
  Projects\STM32U585AI-STWIN.box\Applications\DATALOGMC\Binary
  and download the firmware.



For further details, see **UM2965**

Firmware Update via ST-LINK

To update the firmware via ST-Link follow the procedure below:

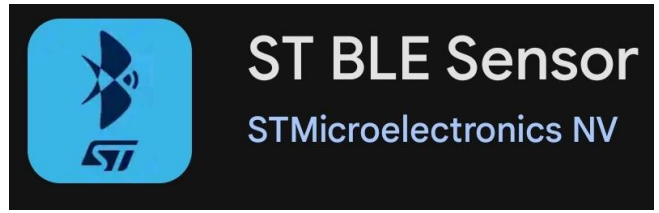
- Connect the STEVAL-STWINBX1 board to any STM32 programmer (here we are using STLINK-V3MINIE, for more configuration see STLINK Setup).
- Connect the STEVAL-STWINBX1 and the programmer to a PC through the proper USB cables.
- Open STM32CubeProgrammer, select the binary file (located under: *Projects\STM32U585AI-STWIN.box\Applications\DATALOGMC\Binary*) and download the firmware.

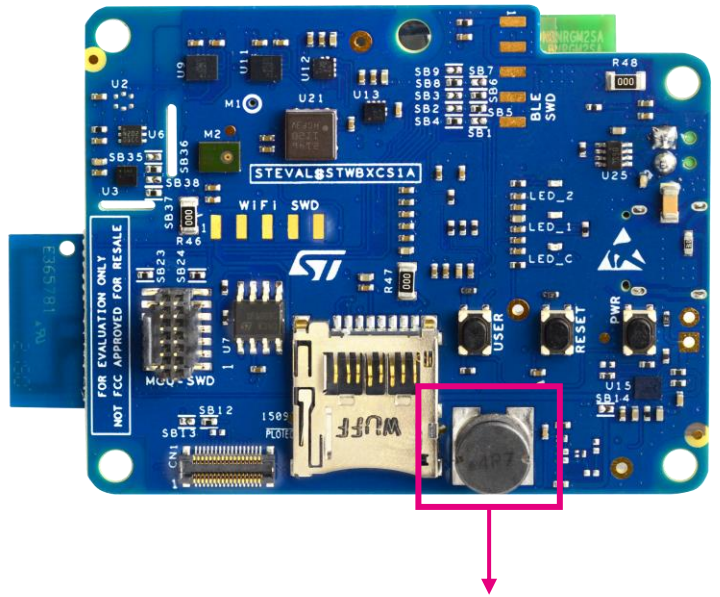
For further details, see **UM2965**

Firmware Update via BLE (FOTA)

The default firmware for STWIN.box enables the Bluetooth pairing via NFC and Firmware On-The-Air upgrade through ST BLESensor app

- Download the app from the Play Store or the Apple Store
- Power on the board by plugging the USB cable
- Turn on the Bluetooth and the NFC on your smartphone
- Place the smartphone on top of the NFC antenna
- The smartphone will read the Bluetooth pairing information and it will automatically load the App.



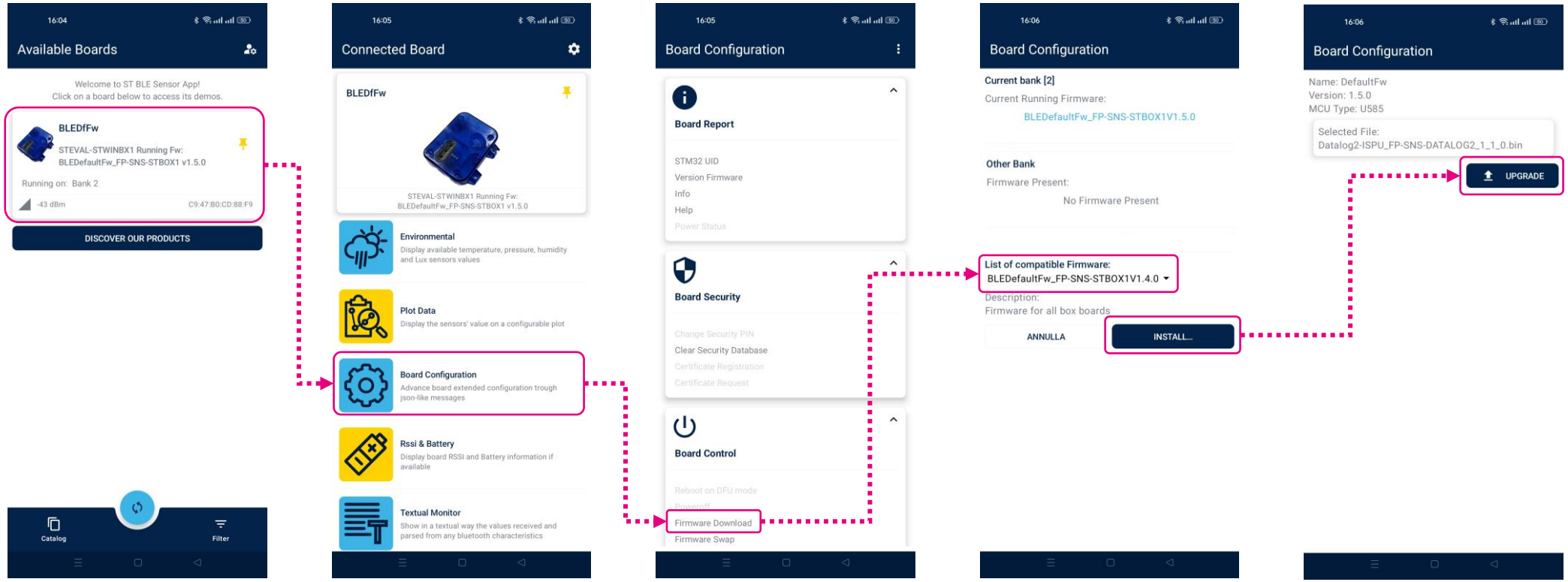


Firmware Update via BLE (FOTA)

- The NFC step is optional, you can also manually open the **ST BLESensor** appn and scan for nearby devices
- The board presents itself as BLEDfFw
- During BLE pairing, if requested, you must insert the following PIN: **123456**
- The application shows the environmental data coming from the board (temperature and pressure)
- At this point, you can choose to upgrade the firmware on the board directly by using the mobile app, by selecting one of the available firmware
- See next slides for details

Firmware Update via BLE (FOTA)

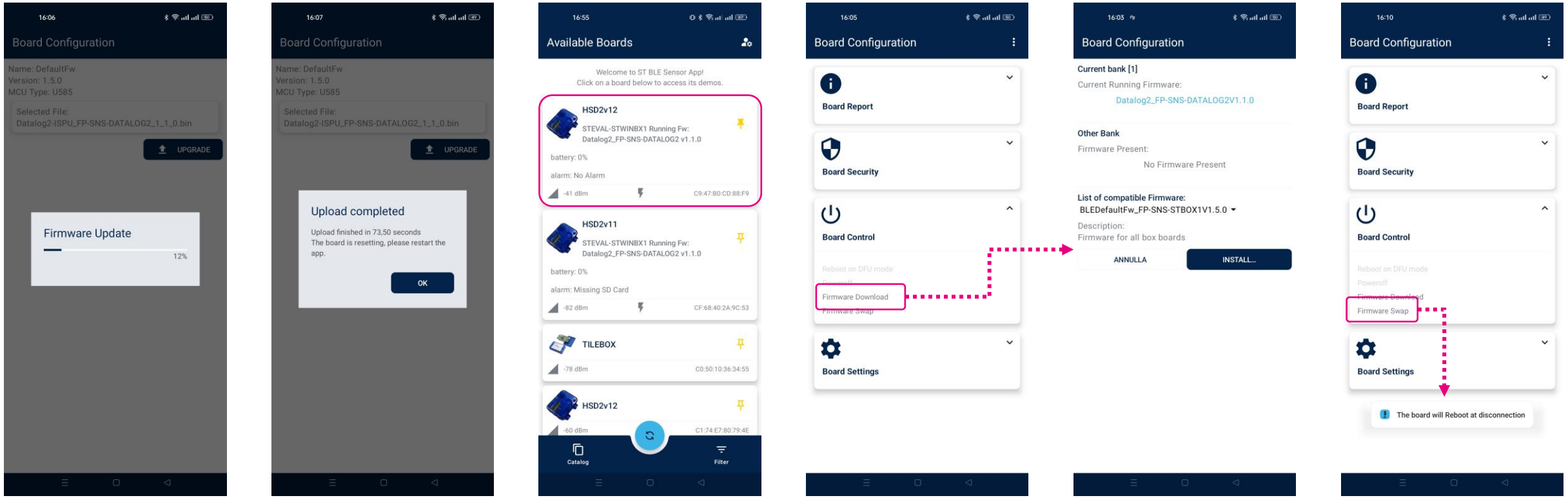
To update the firmware, follow the procedure below:



Firmware Update via BLE (FOTA)

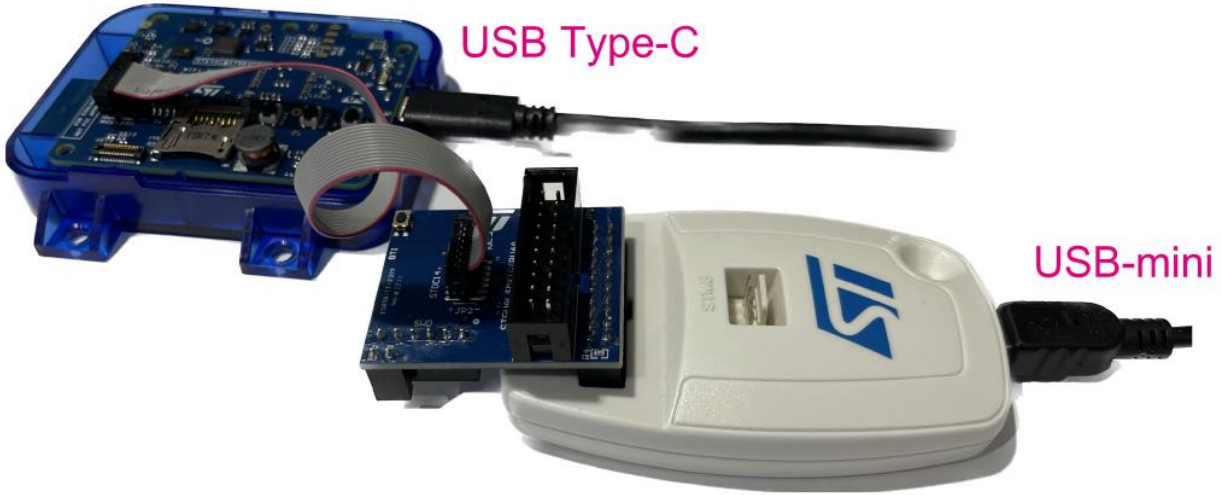
Once the download is finished, the new firmware will restart automatically. To reconnect to BLESensor app (if needed), restart the app.

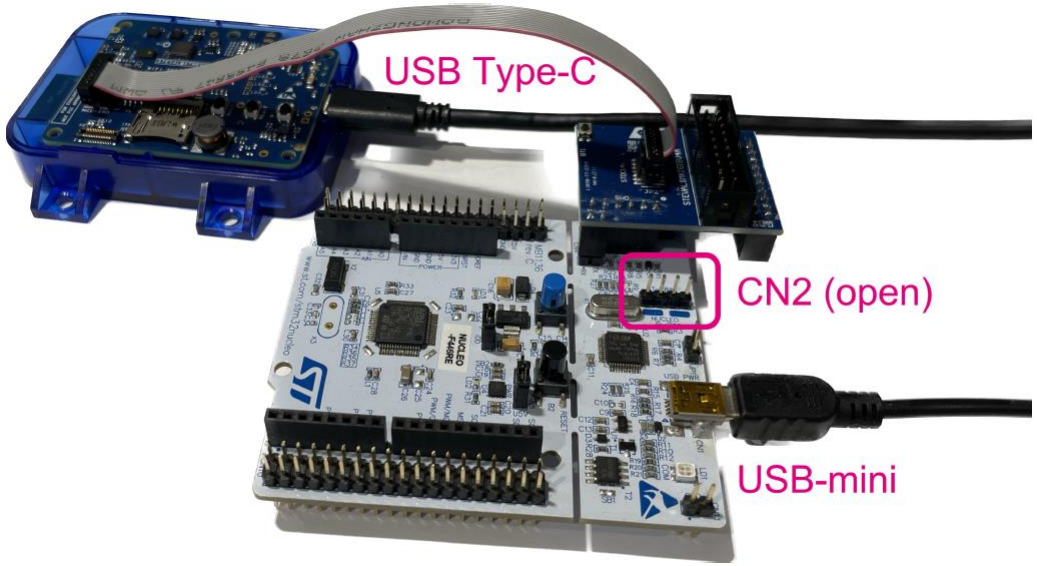
In Board Configuration tab you can also swap between 2 firmware already loaded into the STWIN.box flash, download a new firmware or upgrade the current on with the latest available.



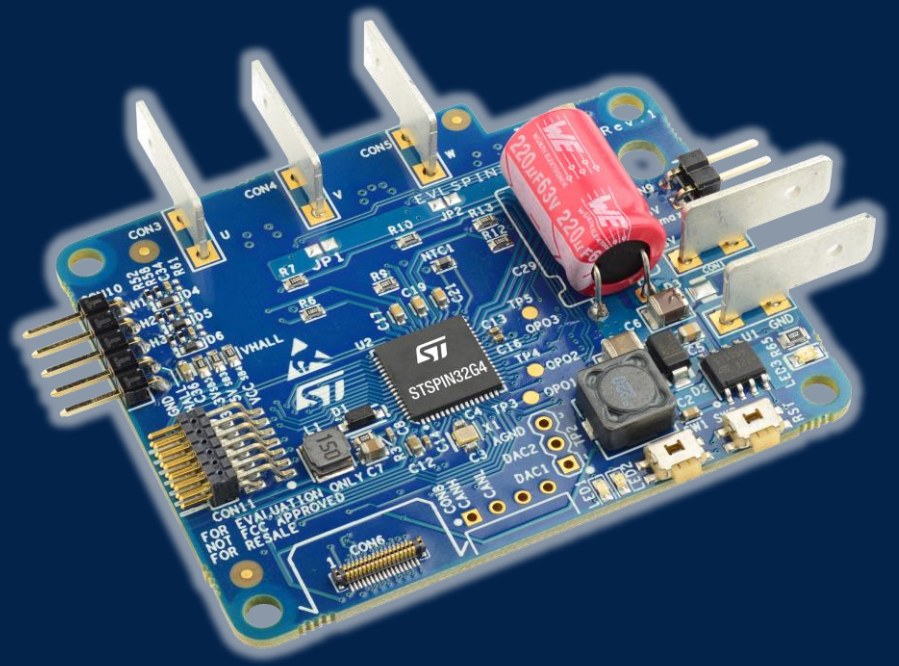
STLINK Setup

- STWIN.box programming connector is natively compatible with STLINK-V3 debuggers family (STLINK-V3SET or STLINK-V3MINIE). STLINK-V3 programmers are NOT included in the kit.
- Alternatively, in order to offer more alternatives, an adapter to STM32 Nucleo-64 boards (ST-LINK/V2-1) or ARM standard JTAG connector (STLINK/V2) is included in the kit.





3.2- How to program EVLSPIN32G4-ACT



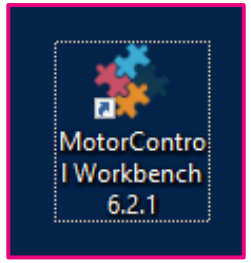
life.augmented

EVLSPIN32G4-ACT Firmware Setup

To configure and flash properly the EVLSPIN32G4-ACT follow the next steps:

1. Download and install STM32 Motor Control Software Development Kit
   (Require STM32CubeMX and one among the supported IDEs: STM32CubeIDE, Keil, IAR)

2. Run **MotorControl Workbench**

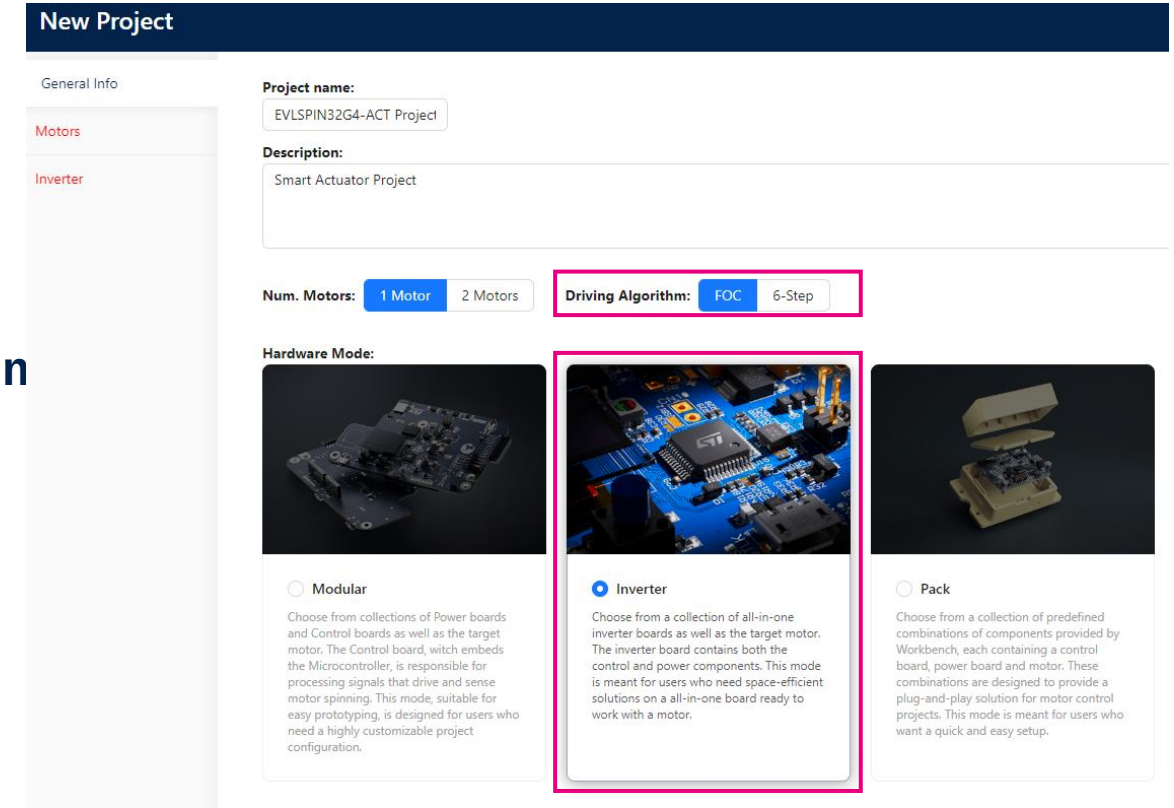


3. Select **New Project**

For further details, see Getting started with STM32 Motor control SDK6.0

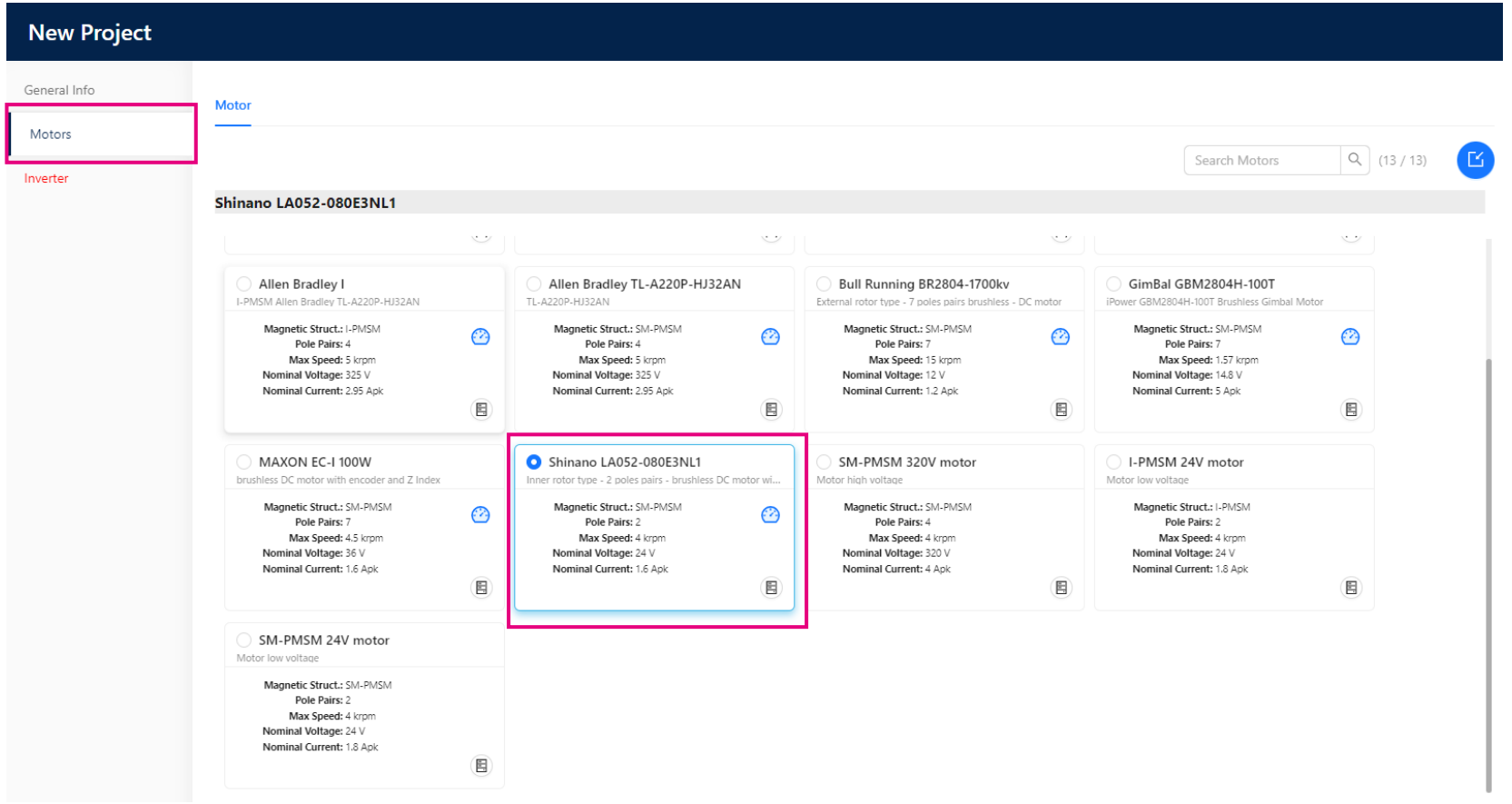
EVLSPIN32G4-ACT Firmware Setup

4. In **general Info** tab select **FOC** as **Driving Algorithm** and **Inverter** as **Hardware** and click **Next**



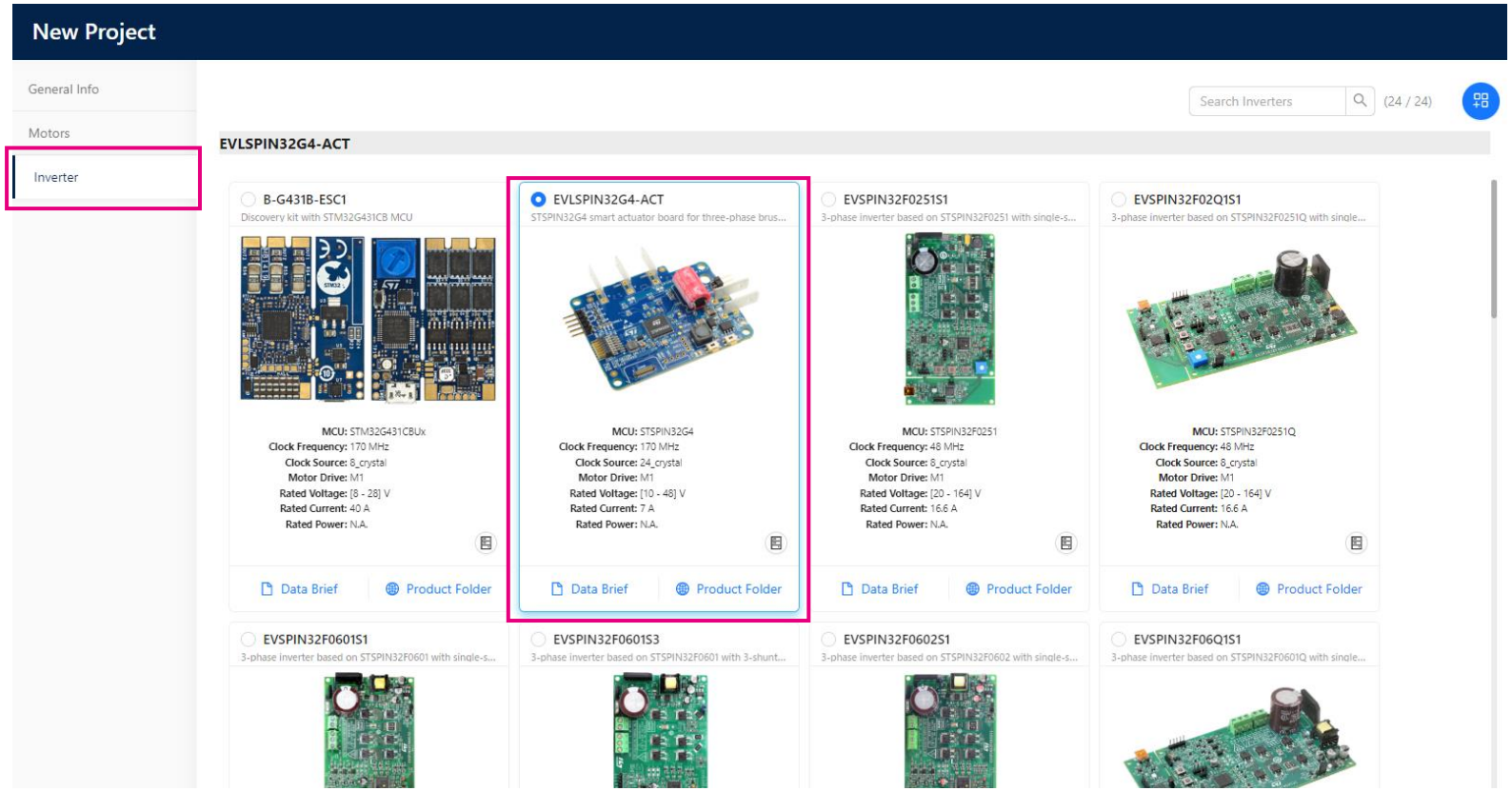
EVLSPIN32G4-ACT Firmware Setup

5. Choose your motor in **Motors** tab, and click **Next**



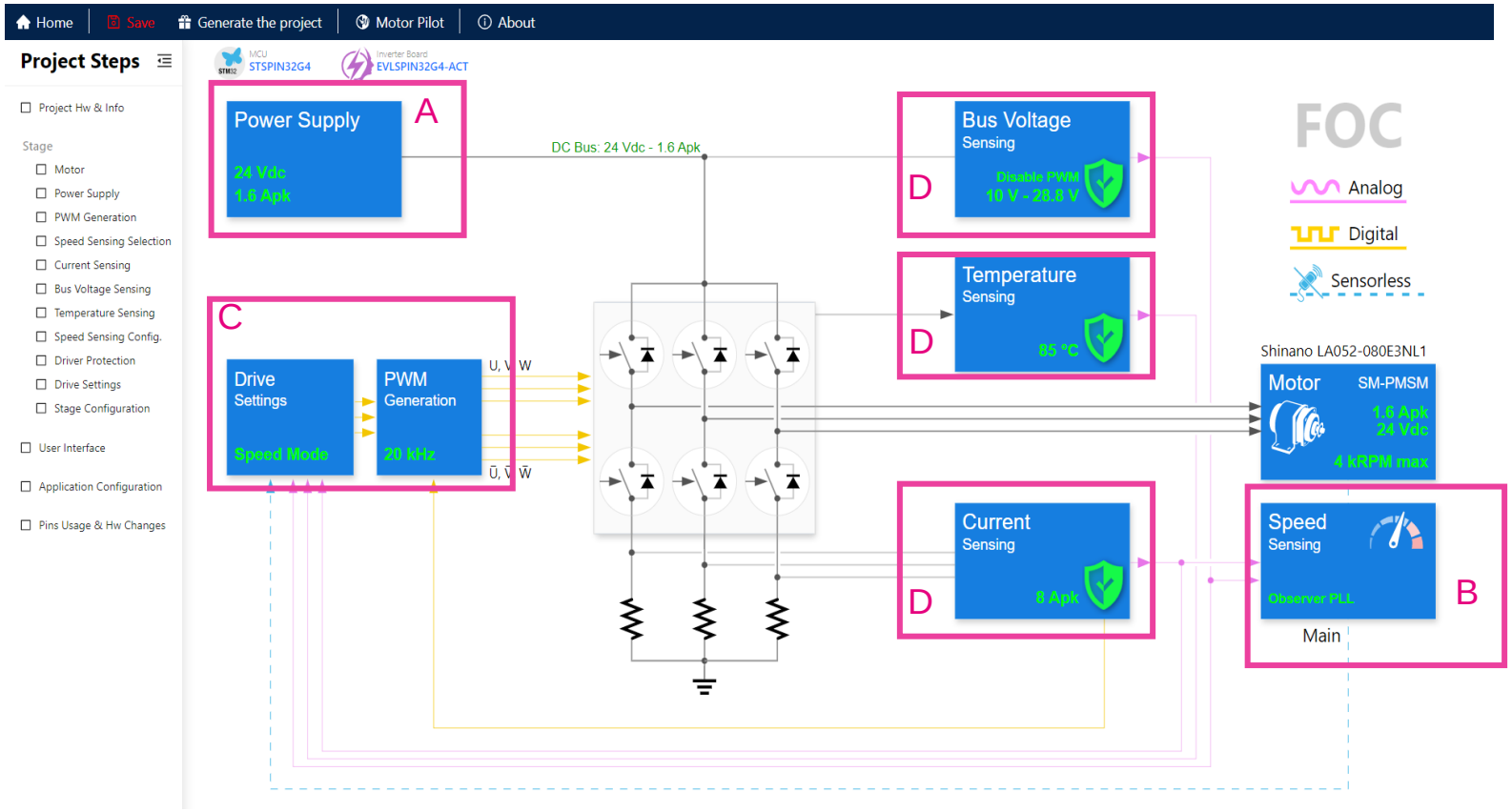
EVLSPIN32G4-ACT Firmware Setup

6. Choose **EVLSPIN32G4-ACT** in the **Inverter** tab and click **Ok**



EVLSPIN32G4-ACT Firmware Setup

7. **Customize** your project:
   - A. Supply voltage
   - B. Speed and position feedback
   - C. Drive settings and PWM frequency
   - D. Protection thresholds and behavior

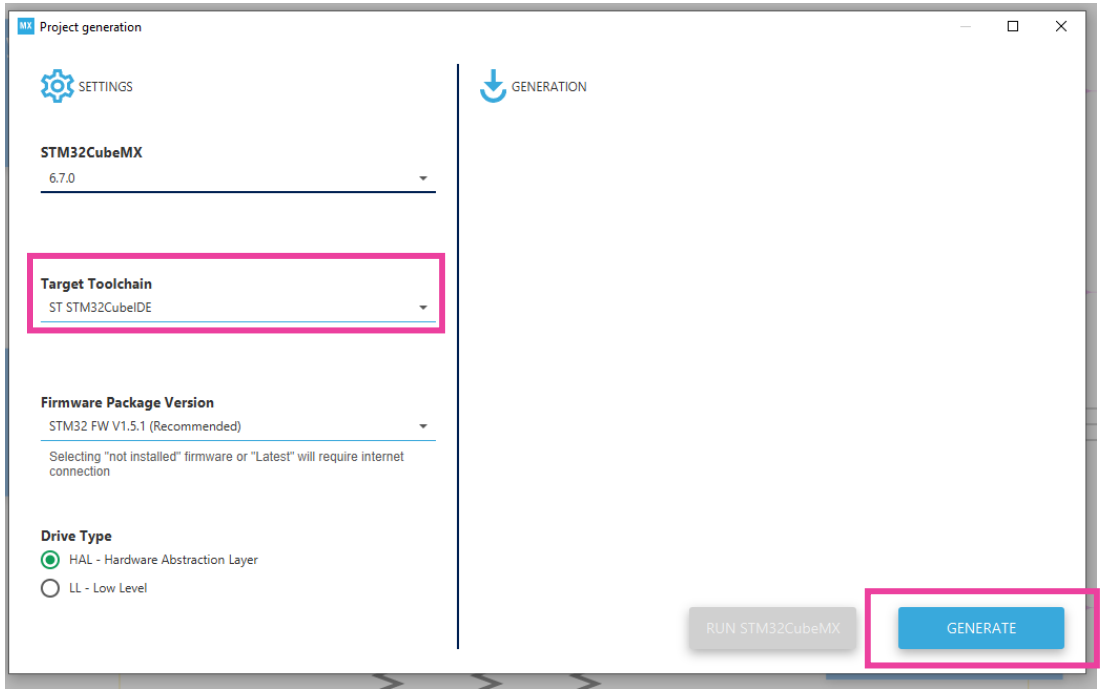


8. Click **Save** and **Generate the project**

EVLSPIN32G4-ACT Firmware Setup

9. Select your IDE in **Target Toolchain** and click **Generate** to create the project source code
(Supported IDEs: STM32CubeIDE, Keil, IAR)

Note: *If STM32CubeG4 libraries are not installed yet, the code generation procedure could ask you to download and install them.*

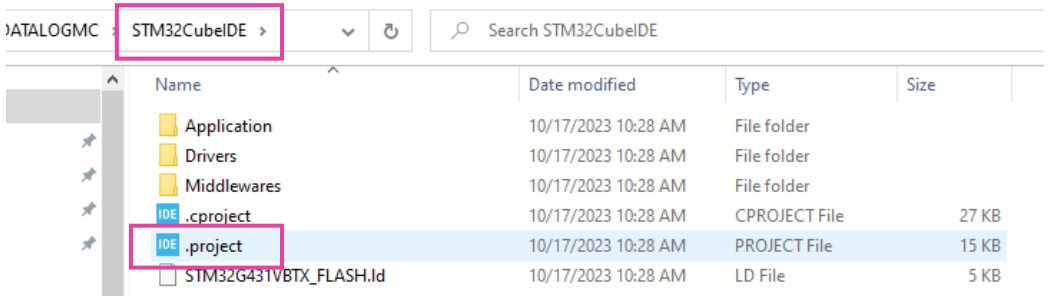


EVLSPIN32G4-ACT Firmware Setup

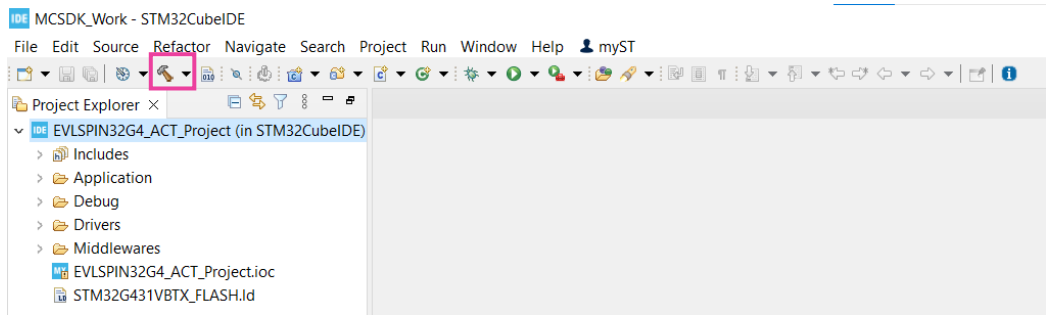
10. Click **OPEN FOLDER** button

EVLSPIN32G4-ACT Firmware Setup

11. Open the **STM32CubeIDE** folder and add the generated project to the workspace



12. Build the project to **generate the binary**



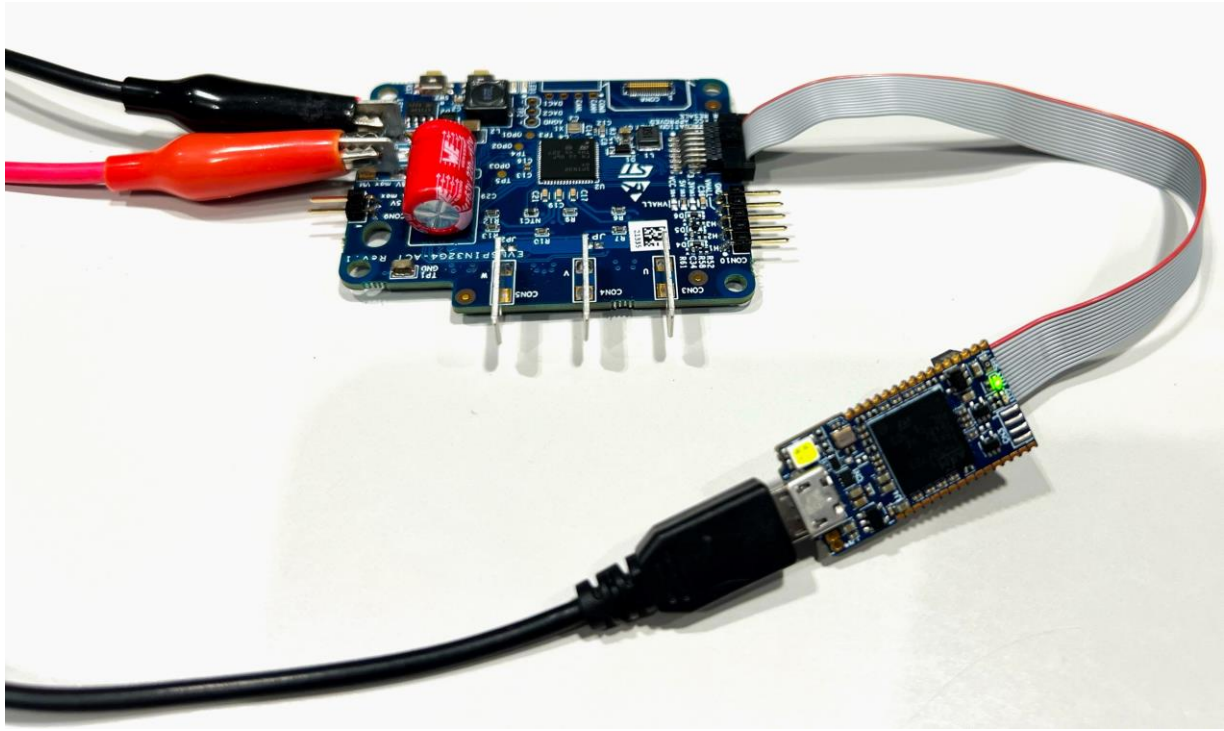
Note: *If you have selected a different toolchain, please refer to its documentation to open and compile the generated project*

life.augmented

EVLSPIN32G4-ACT Firmware Setup

12. Power on the EVLSPIN32G4-ACT (see Hardware setup), **connect** the **programmer** as illustrated in the figure and **download the binary**.
    The board is now ready to be connected to the STWIN.box.
    For more detail visit: Getting started with the EVLSPIN32G4-ACT

13. In case the STLink-V3 is not available, it's possible to use the STLink-V2 adapter as explained for the STWIN.box



life.augmented

3.3- Hardware Setup

Setup

EVLSPIN32G4-ACT plus STEVAL-STWINBX1

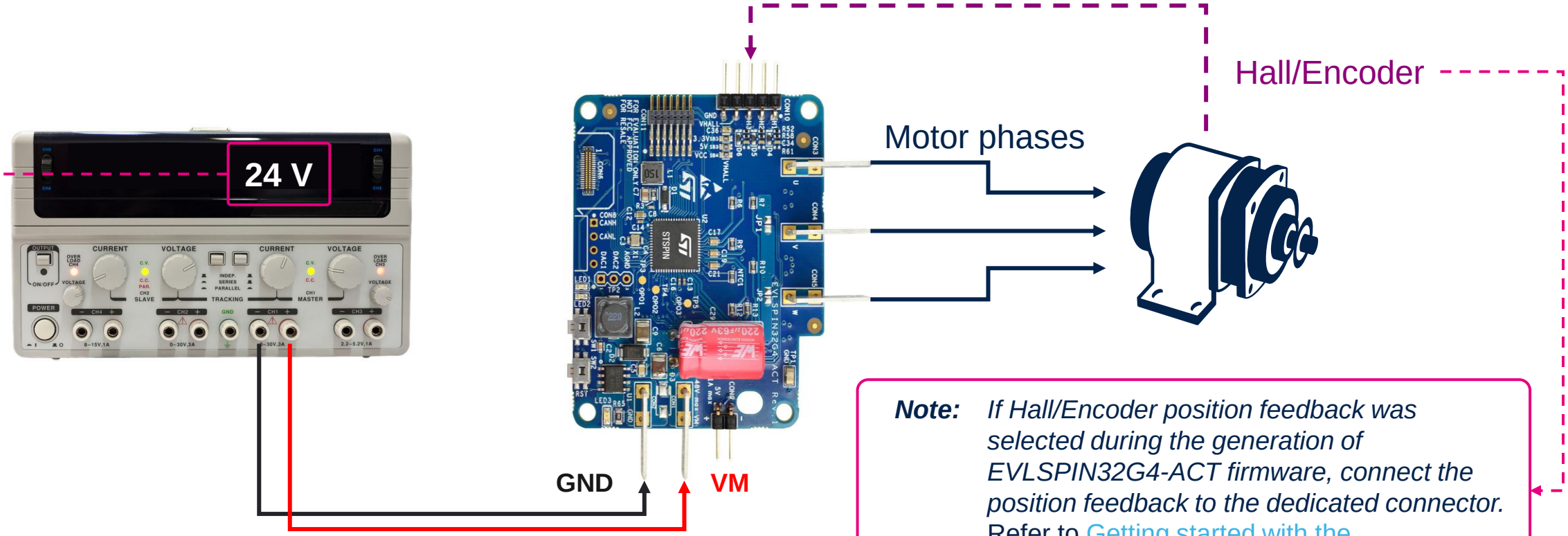
Connect the two boards using the **flex** cable included with EVLSPIN32G4-ACT (STEVAL-FLTCB04) as shown in the figure.

User has the option to supply the STEVAL-STWINBX1 board through the EVLSPIN32G4-ACT by connecting the **red-brown cable** provided in the EVLSPIN32G4-ACT package.
In alternative, STEVAL-STWINBX1 can be supply via USB.

Setup

EVLSPIN32G4-ACT plus STEVAL-STWINBX1

Connect power supply and motor to the EVLSPIN32G4-ACT.



Power Supply
24 Vdc
1.5 Apk

Note: *The power supply voltage depends on the motor used. Is the one properly selected during the generation of EVLSPIN32G4-ACT firmware.*

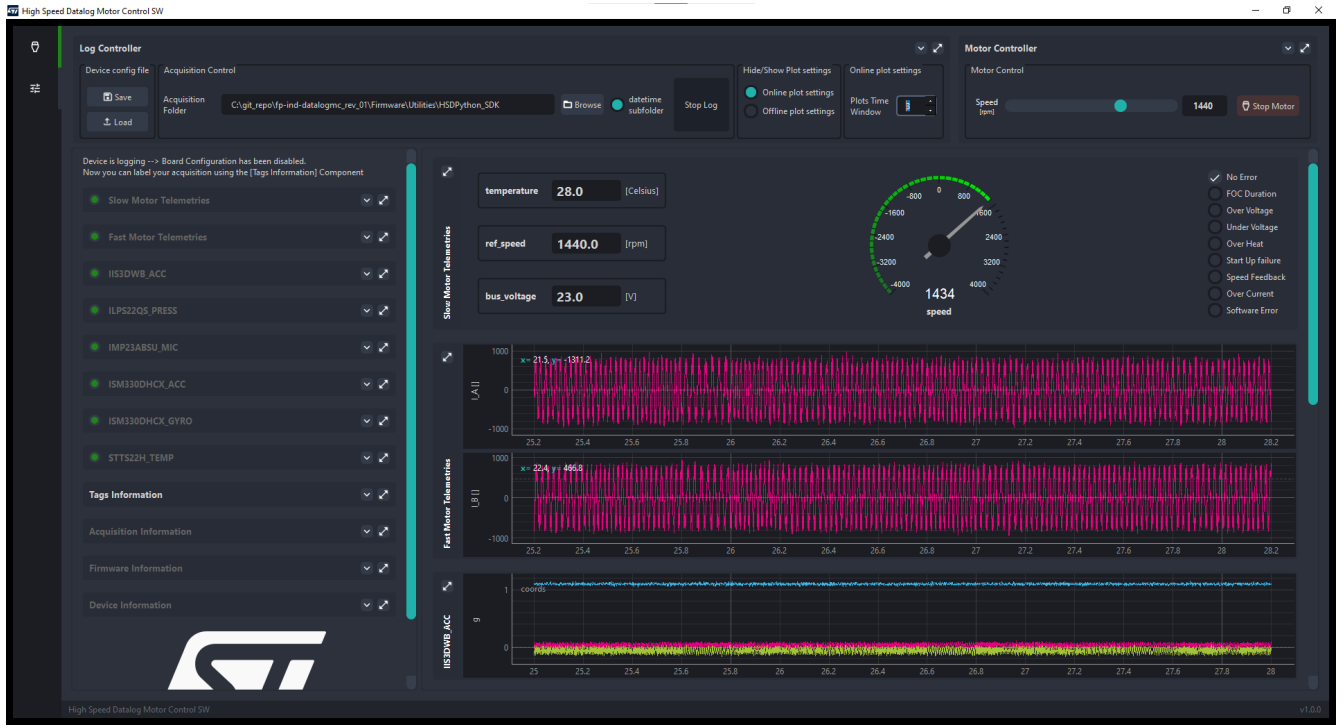
Note: *If Hall/Encoder position feedback was selected during the generation of EVLSPIN32G4-ACT firmware, connect the position feedback to the dedicated connector.* Refer to Getting started with the EVLSPIN32G4-ACT for more details.

4- DATALOGMC Demonstration

4.1- USB data streaming Real Time Plot

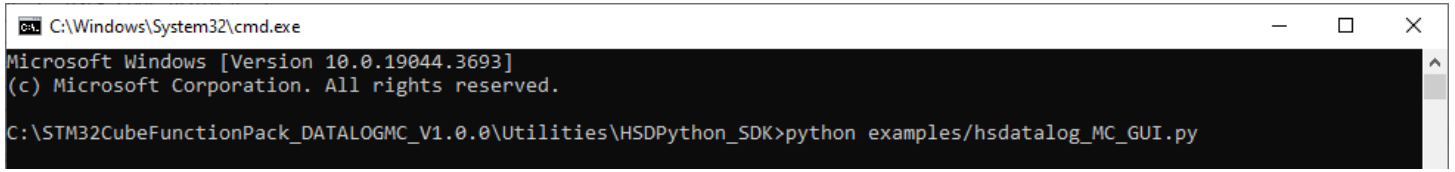
Python Application for real-time plot

- DATALOGMC provides a Python example for **real time control and plot of Sensor and Motor control data** such as control current, voltage, speed, faults.
- *examples/hsdatalog_MC_GUI.py* works within the HSDPython_SDK, developed in Python 3.10 on Windows and Linux environments.
  - *HSDPython_SDK* requires different Python modules, distributed together with FP-INC-DATALOGMC. By installing them through *the provided installers*, all the required dependencies are automatically solved – see HSDPython_SDK to fully setup your Python environment



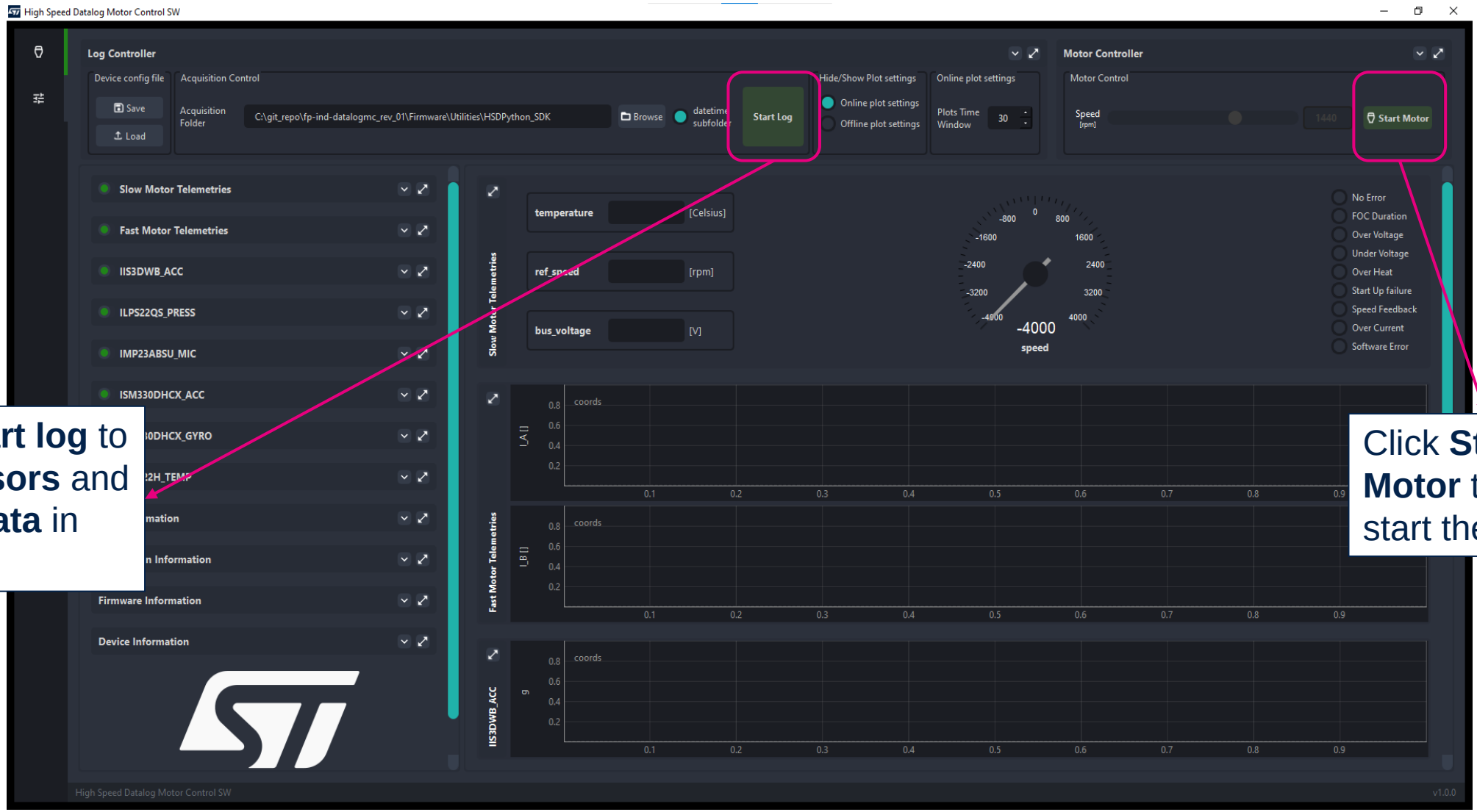
Execute *hsdatalog_MC_GUI.py*

- Once the two boards have been setup and configured and the Python environment has been properly updated, connect the STWIN.box board via USB and launch the real time plot by just executing ***hsdatalog_MC_GUI.py*** available in ***Utilities/HSDPython_SDK/examples.***



- Click on ***Connect*** button to allow the connection between the board and the PC

Execute *hsdatalog_MC_GUI.py*

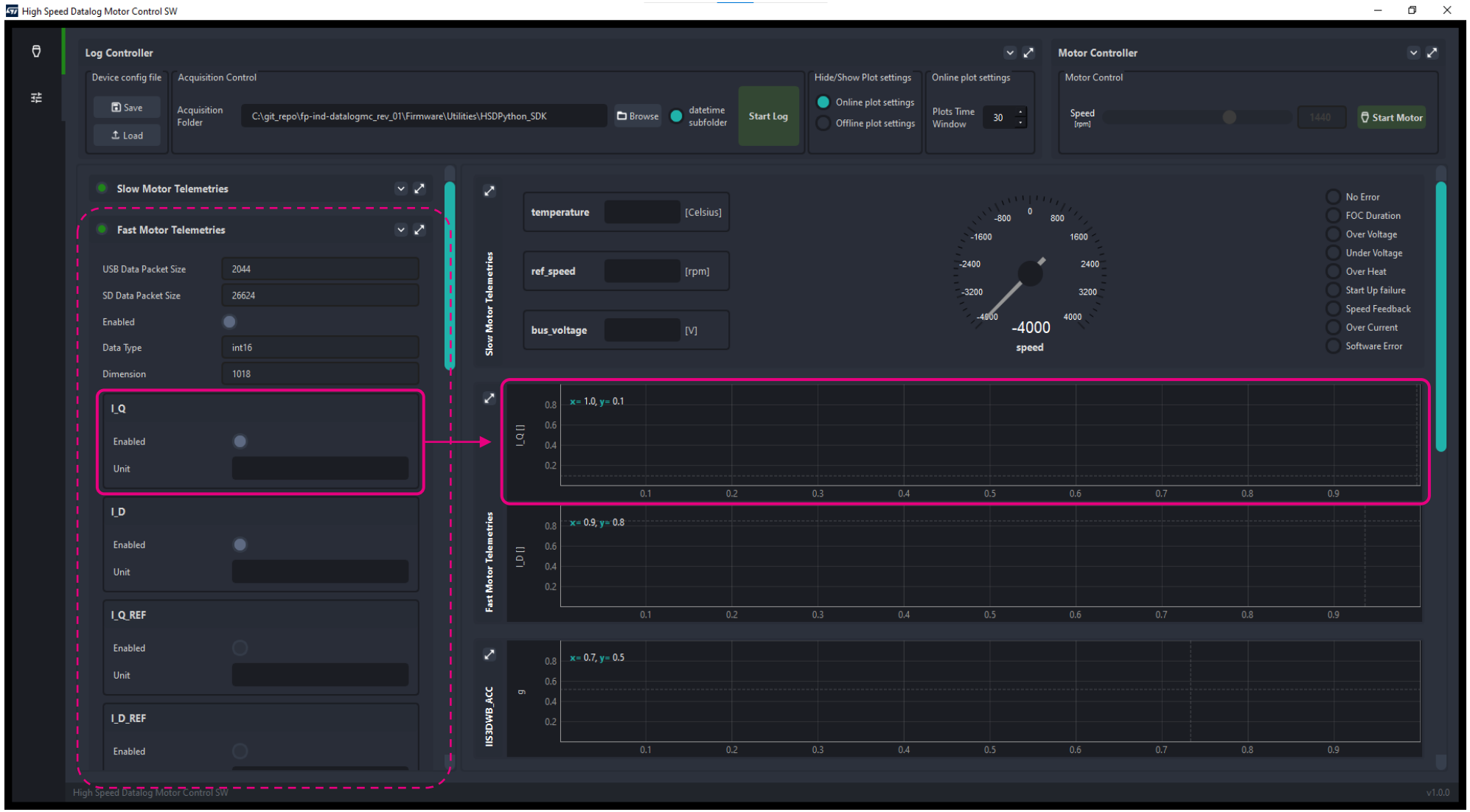


Click **Start log** to see **sensors** and **motor data** in real-time

Click **Start Motor** to start the motor

Note: User must start the motor in order to watch motor data telemetries.

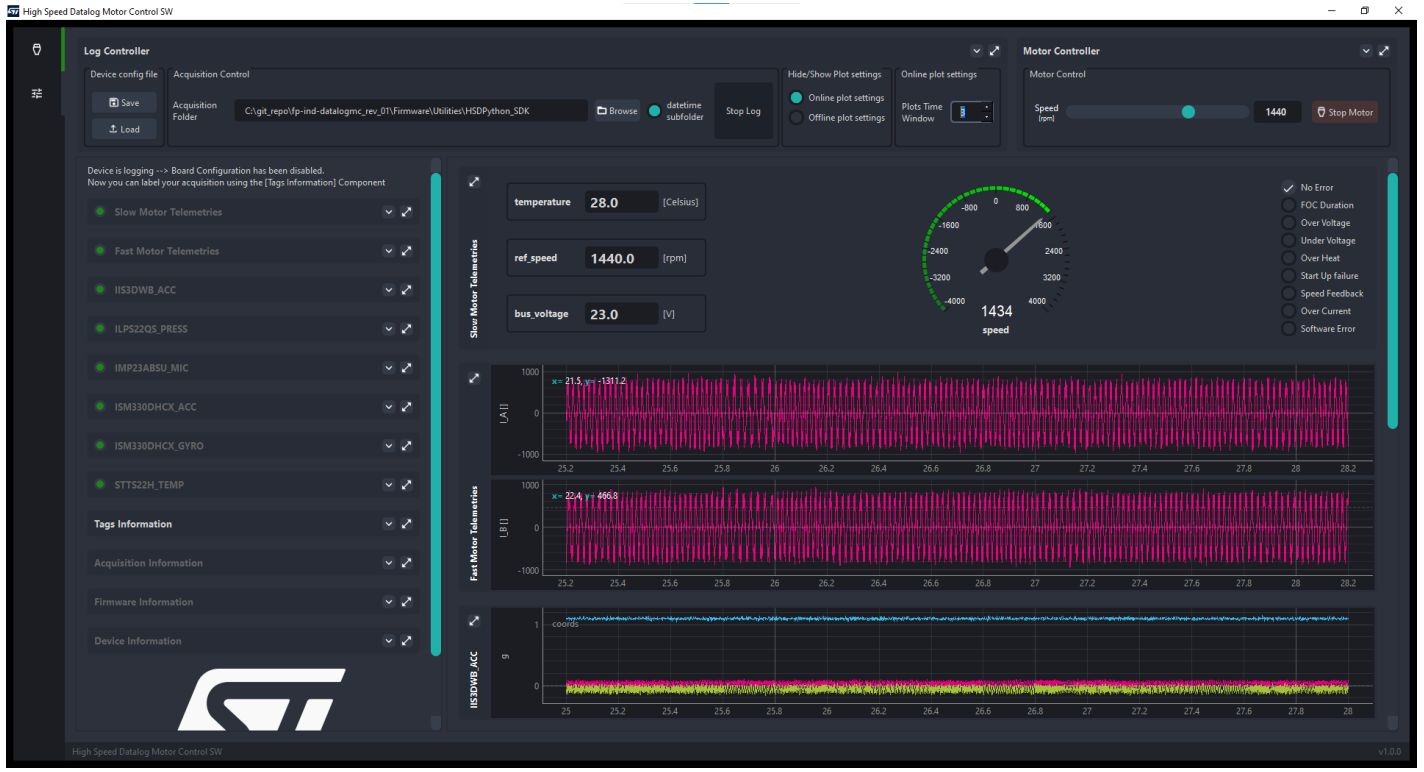
Execute *hsdatalog_MC_GUI.py*



Note: User can select maximun four **Fast Motor Telemetries**

hsdatalog_MC_GUI.py

- **hsdatalog_MC_GUI.py** allow you to:
  - Start/Stop the motor via motor control widget
  - Set Motor velocity via motor control widget
  - Configure fast and slow motor telemetries
  - Enable/disable the needed sensors
  - Setup data rate, full scale, timestamps
  - Retrieve sensor status
  - Save and load a configuration via a JSON file
  - Start/stop logging data on the PC
  - Tag current acquisition with your label (see next slide)
- Once clicked on *Start Log* button, data are live plotted and the application will create a YYYYMMDD_HH_MM_SS (i.e., 20230128_16_33_00) folder containing the raw data and the JSON configuration file.

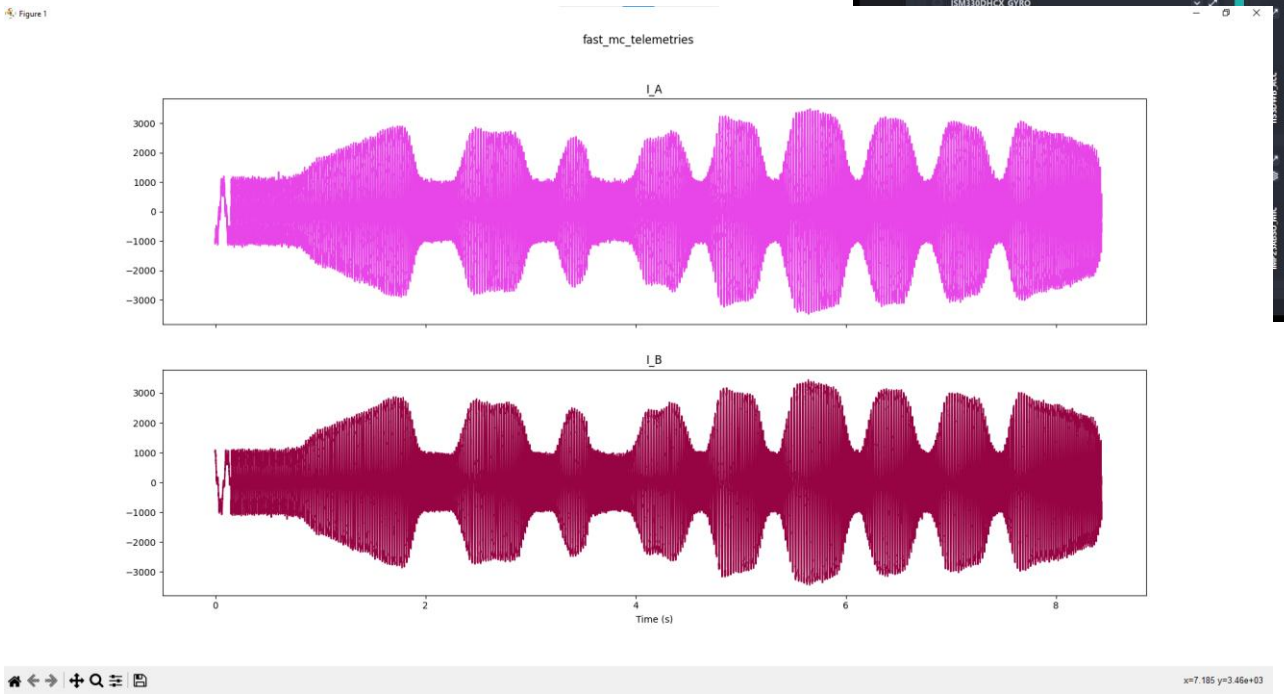


hsdatalog_MC_GUI.py

- Tag your acquisition with **TAGS INFO**:
  - choose which tag classes will be used for the next acquisition
  - handle data tagging and labelling of an ongoing acquisition
  - set up the acquisition name and description

hsdatalog_MC_GUI.py

- Once concluded the test, you can also check and plot the entire dataset by clicking on ***Plot the last acquisition*** button

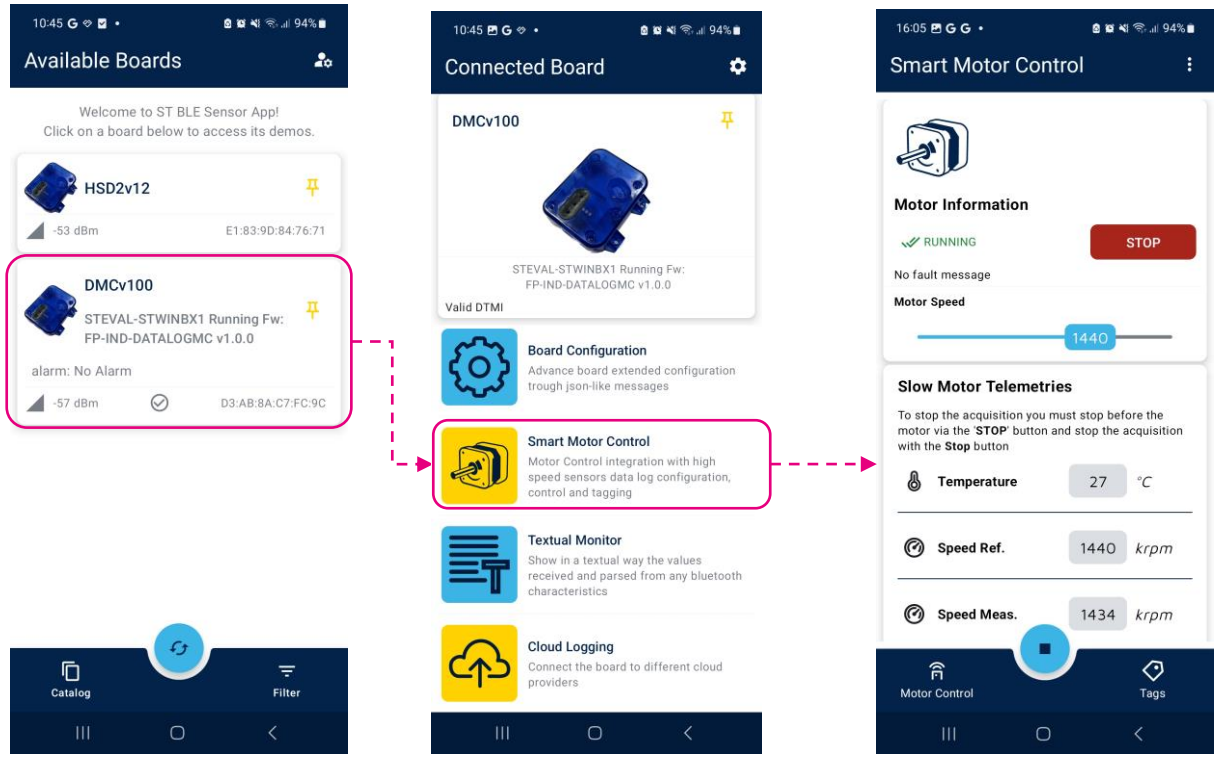


4.2- Data logging on SD card, configuration with BLESensor App

ST BLESensor App: Smart Motor Control tab

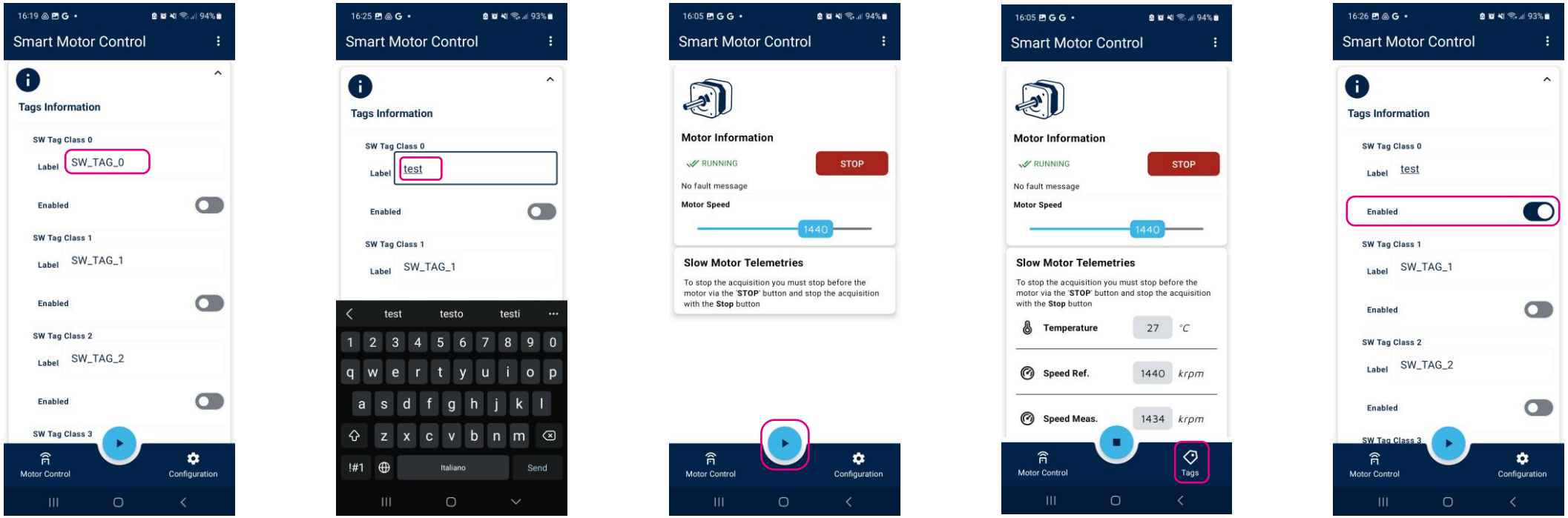
- DATALOGMC application can be controlled via Bluetooth using the ST BLE Sensor app (for both Android and iOS – **v5.2** and above) which lets you manage start/stop motor, set motor velocity, slow and fast telemetries configurations, sensor configurations, start/stop data acquisition on SD card and control data labelling.

- Once connected, the main window allow you to:
  1. Start the motor
  2. Configure motor telemetries and sensors
  3. Start the log



Acquisition settings and control

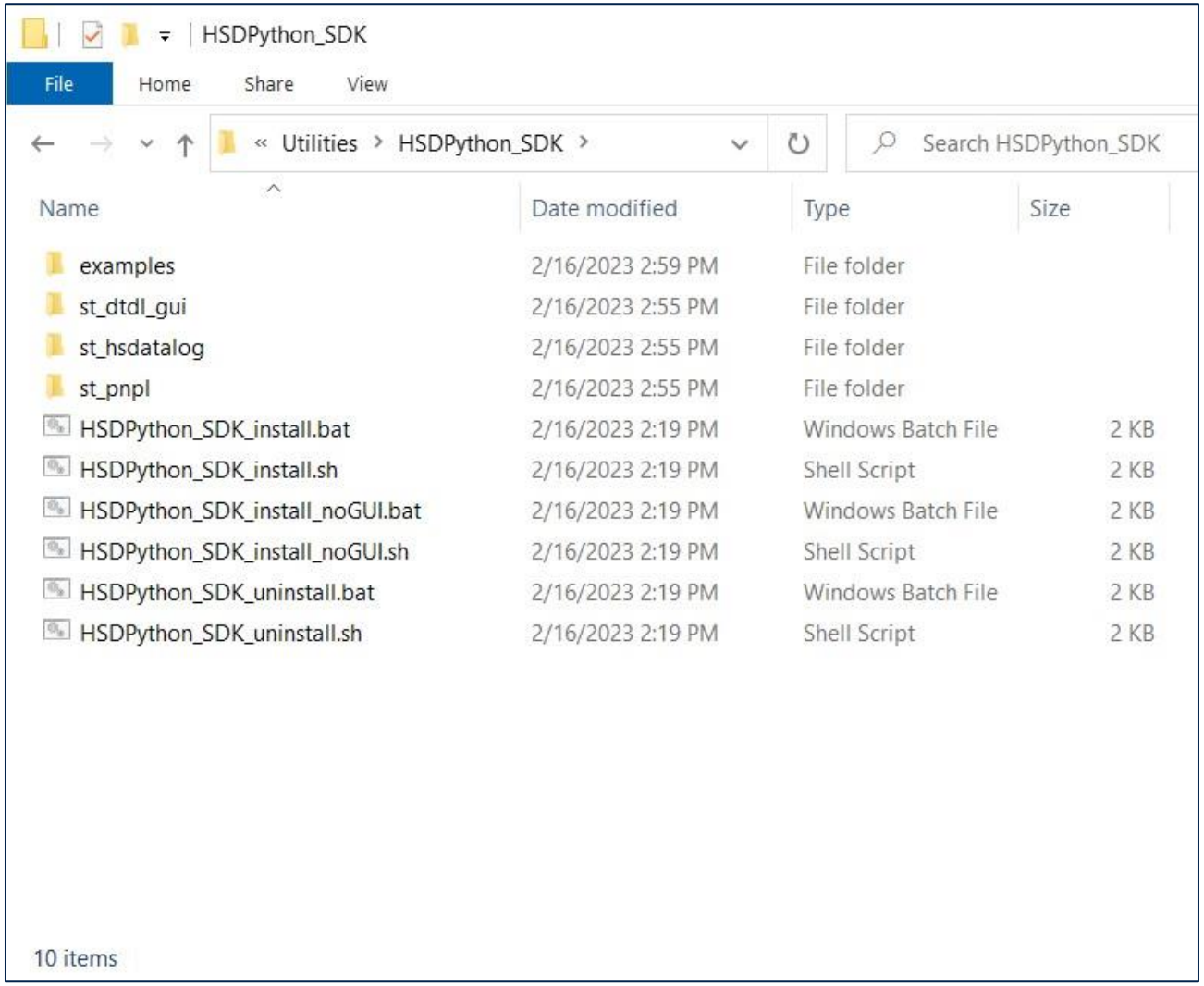
- By clicking to the **tags** button you can switch to the acquisition settings and control tab to:
  - choose which tag classes will be used for the next acquisition
  - handle data tagging and labelling of an ongoing acquisition
- A YYYYMMDD_HH_MM_SS (i.e., 20200128_16_33_00) folder containing the raw data and the JSON configuration file will be created into the SD card



4.3- HSDPython_SDK

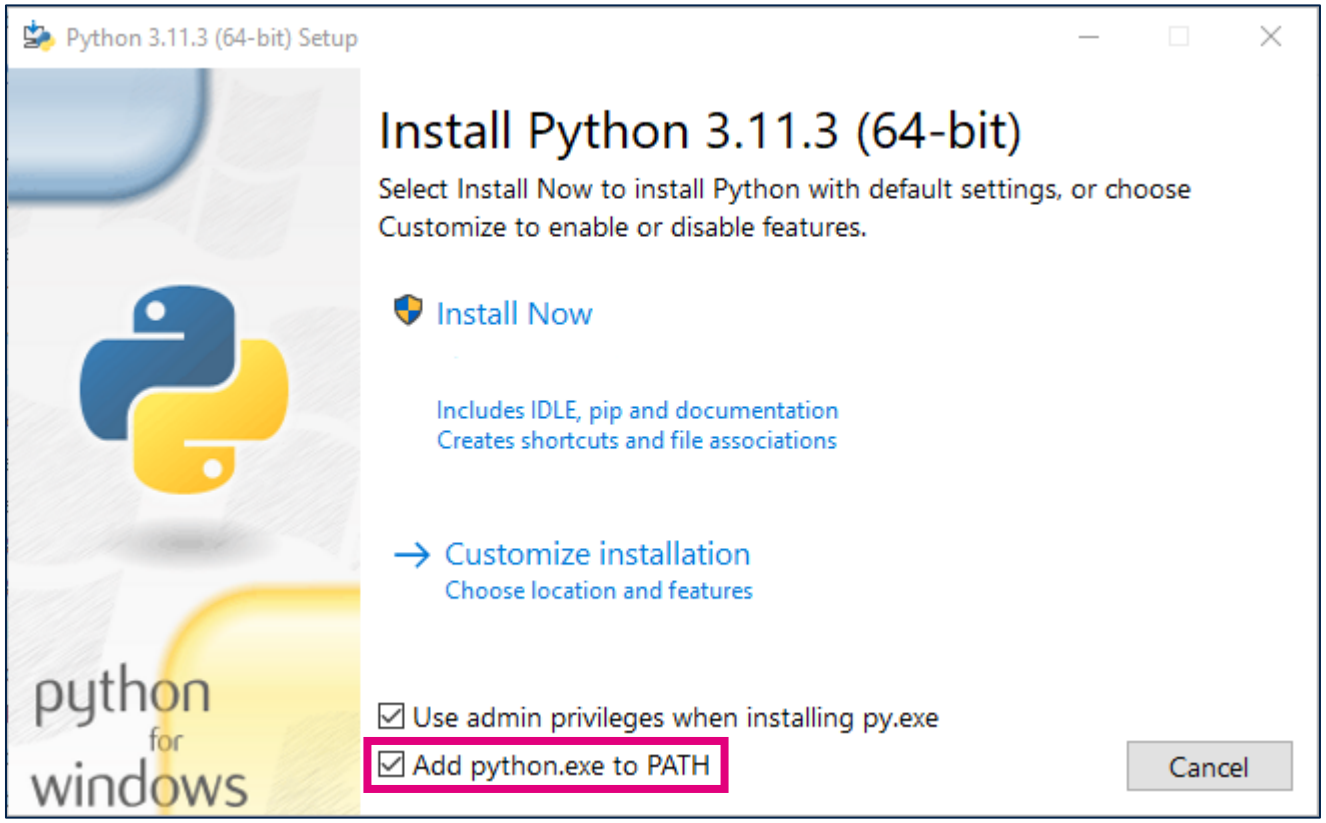
HSDPython_SDK

- FP-IND-DATALOGMC comes with a dedicated Python SDK, ready-to-use for integration into any data science design flow.
- It is distributed in FP-SNS-DATALOG2 and FP-IND-DATALOGMC and can handle data acquired from both packages.
- HSDPython_SDK has been developed in Python 3.10
- The SDK contains many Python scripts, examples and Jupiter notebooks that can be used to log and elaborate data
- The scripts take advantage for the API provided by the st_dtdl_gui, st_hsdatalog and st_pnpl Python modules.



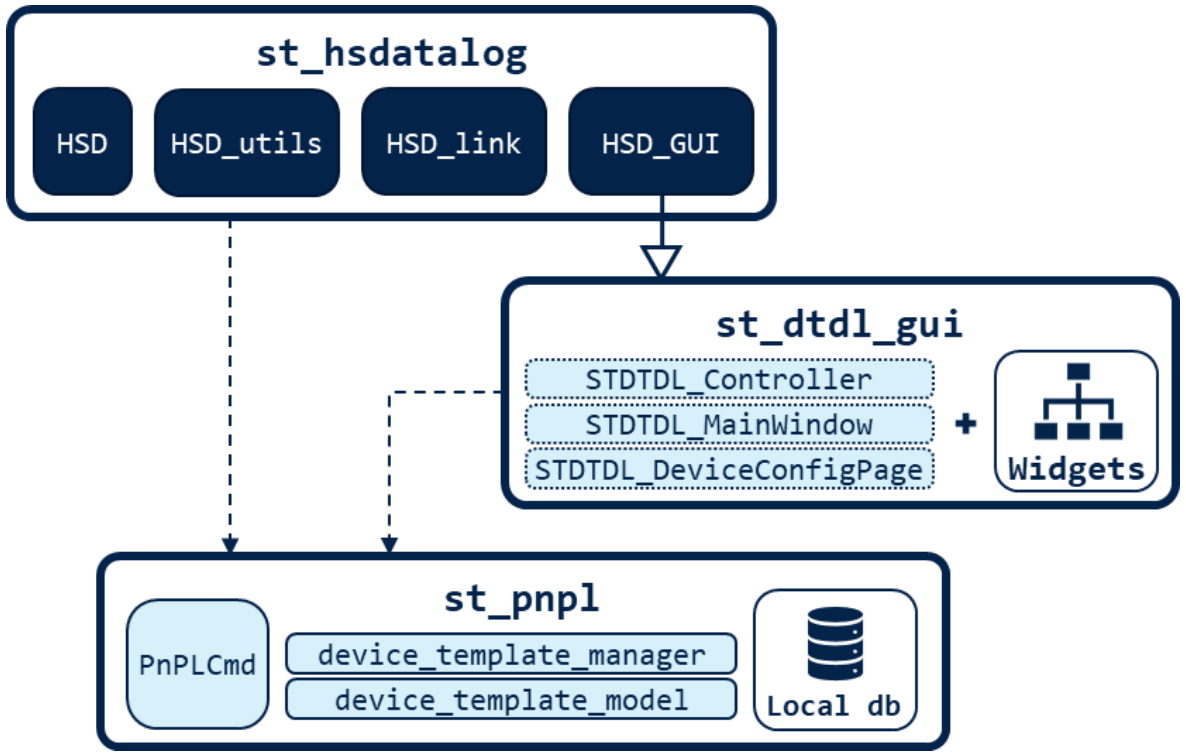
HSDPython_SDK

- Before using HSDPython_SDK, Python3 (>=3.10) must be properly installed on your machine.
- The following steps are valid for a Windows machine. Similar approach can be followed on other OS as well.
  - Download the installer from python.org and launch it
  - Select **Add python.exe flag** and click **Install Now**. Administrator privileges are needed.
  - Once the setup is complete, you can use Python on your machine



HSDPython_SDK installation

- The three modules are distributed as Python wheels
- Launch *HSDPython_SDK_install.bat* (Windows) or *HSDPython_SDK_install.sh* (Linux)
- The SDK modules and their dependencies will be installed in your Python environment
- For Linux users, further steps are needed.
  - A step-by-step procedure is described in detail in the readme_linux file.

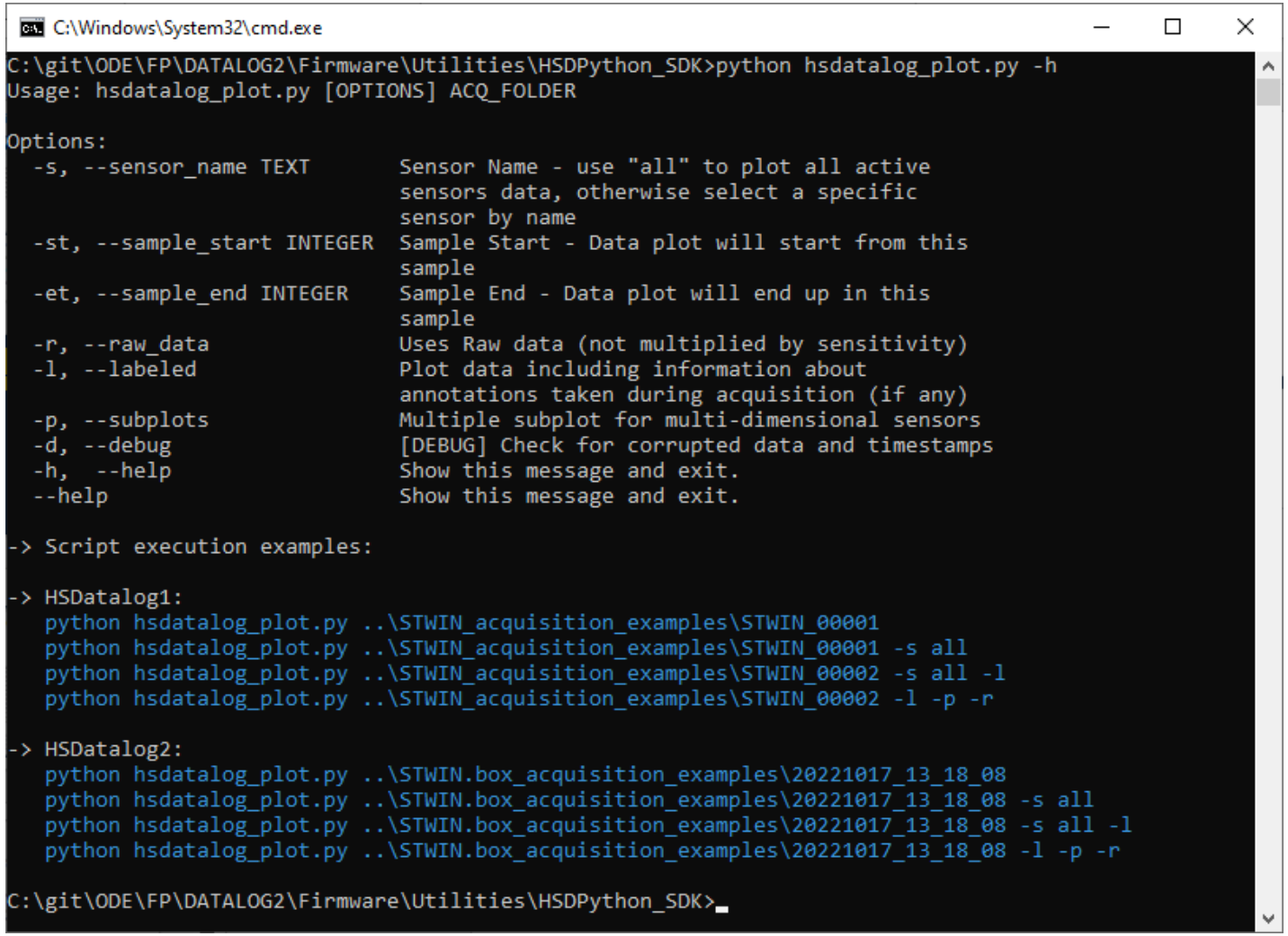


HSDPython_SDK scripts

- The SDK can be used to develop a custom project either by importing the provided modules in a new application or by modifying one of the available scripts
- Here the list of scripts available in the examples folder:
  - *hsdatalog_data_export.py* can convert data into CSV or TSV files.
  - *hsdatalog_data_export_by_tags.py* can be used for tagged acquisition to convert data into different files, one for each tag used.
  - *hsdatalog_dataframes.py* can save data as pandas dataframe for further processing needs.
  - *hsdatalog_MC_GUI.py* provides an example for real time control and plot.
  - *hsdatalog_plot.py* can plot the desired data.
  - *hsdatalog_to_nanoedge.py* can prepare data to be imported into NanoEdge AI Studio solution.
  - *hsdatalog_to_unico.py* can prepare data to be imported into Unico-GUI.
  - *hsdatalog_to_wav.py* can convert audio data into a wave file.

HSDPython_SDK scripts

- You can execute the scripts in your preferred Python environment
  - i.e.: use the command *python hsdatalog_plot.py*
- Discover the complete list of parameters for each script by executing with the *-h* option
  - i.e.: *python hsdatalog_ploy.py -h*



life.augmented

5- Documents & Related Resources

Documents & Related Resources

All documents are available in the DESIGN tab of the related products webpage

FP-IND-DATALOGMC:

- **DB5152:** STM32Cube function pack for high-speed datalogging of sensors data and motor control telemetries – **databrief**
- **UM:** Getting started with the STM32Cube function pack for high-speed datalogging of sensors data and motor control telemetries– **user manual**
- **Software setup file**

STEVAL-STWINBX1:

- **Gerber files, BOM, Schematic**
- **DB4598**: STWIN.box - SensorTile Wireless Industrial Node Development Kit – **databrief**
- **UM2965**: Getting started with the STEVAL-STWINBX1 SensorTile wireless industrial node development kit – **user manual**

EVLSPIN32G4-ACT:

- **Gerber files, BOM, Schematic**
- **DB5035**: STSPIN32G4 reference design for next generation smart actuators – **databrief**
- **UM3168**: Getting started with the EVLSPIN32G4-ACT – **user manual**

life.augmented

Consult www.st.com for the complete list

6- STM32 Open Development Environment: Overview

life.augmented

STM32 ODE Ecosystem

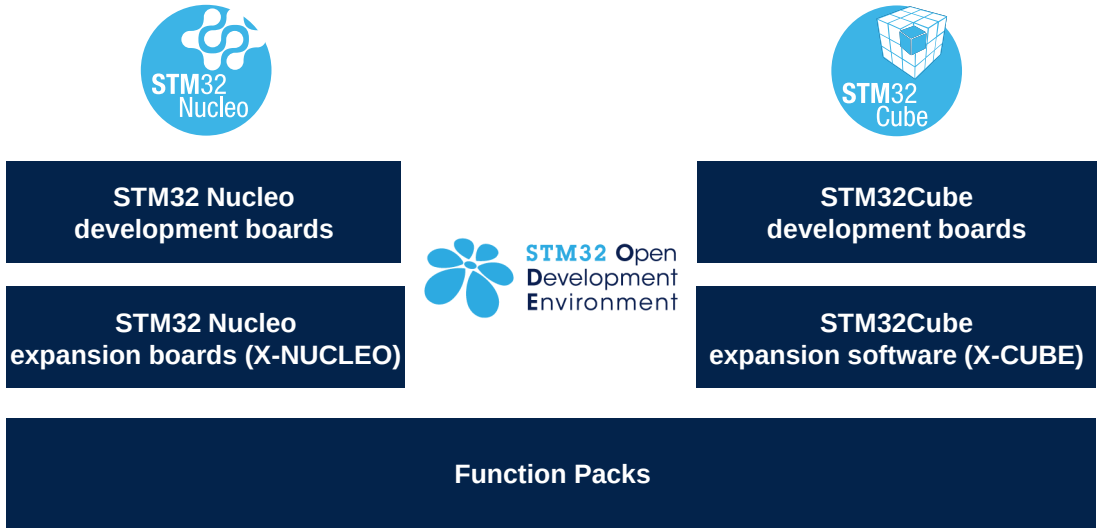
FAST, AFFORDABLE PROTOTYPING AND DEVELOPMENT

The STM32 Open Development Environment (ODE) is an **open**, **flexible**, **easy** and **affordable** way to develop innovative devices and applications based on the STM32 32-bit microcontroller family combined with other state-of-the-art ST components connected via expansion boards. It enables fast prototyping with leading-edge components that can quickly be transformed into final designs.

The STM32 ODE includes the following five elements:

- STM32 Nucleo development boards. A comprehensive range of affordable development boards for all STM32 microcontroller series, with unlimited unified expansion capability, and with integrated debugger/programmer
- STM32 Nucleo expansion boards. Boards with additional functionality to add sensing, control, connectivity, power, audio or other functions as needed. The expansion boards are plugged on top of the STM32 Nucleo development boards. More complex functionalities can be achieved by stacking additional expansion boards
- STM32Cube software. A set of free-of-charge tools and embedded software bricks to enable fast and easy development on the STM32, including a Hardware Abstraction Layer, middleware and the STM32CubeMX PC-based configurator and code generator
- STM32Cube expansion software. Expansion software provided free of charge for use with STM32 Nucleo expansion boards, and compatible with the STM32Cube software framework
- STM32Cube Function Packs. Set of function examples for some of the most common application cases built by leveraging the modularity and interoperability of STM32 Nucleo development boards and expansions, with STM32Cube software and expansions.

The STM32 Open Development Environment is compatible with a number of IDEs including IAR EWARM, Keil MDK, mbed and GCC-based environments.



life.augmented

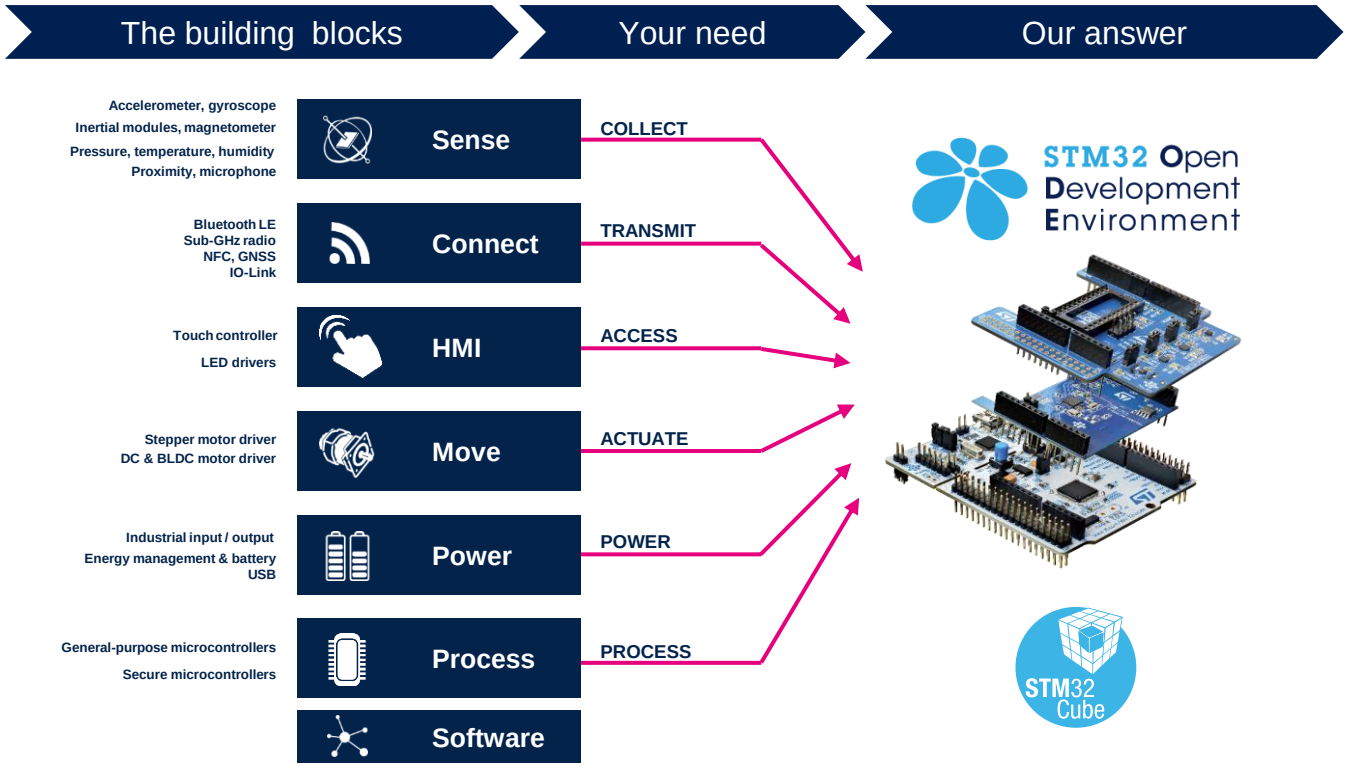
STM32 Open Development Environment: all that you need

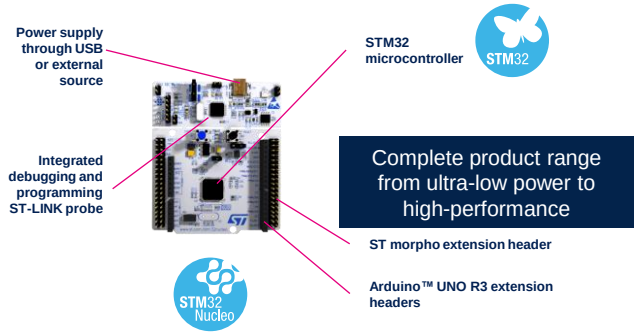
The combination of a broad range of expandable boards based on leading-edge commercial products and modular software, from driver to application level, enables fast prototyping of ideas that can be smoothly transformed into final designs.

To start your design:

- Choose the appropriate STM32 Nucleo development board (MCU) and expansion (X-NUCLEO) boards (sensors, connectivity, audio, motor control etc.) for the functionality you need
- Select your development environment (IAR EWARM, Keil MDK, and GCC-based IDEs) and use the free STM32Cube tools and software.
- Download all the necessary software to run the functionality on the selected STM32 Nucleo expansion boards.
- Compile your design and upload it to the STM32 Nucleo development board.
- Then start developing and testing your application.

Software developed on the STM32 Open Development Environment prototyping hardware can be directly used in an advanced prototyping board or in and end product design using the same commercial ST components, or components from the same family as those found on the STM32 Nucleo boards.





life.augmented

Thank you

life.augmented