life.augmented

STEVAL-PDETECT1 Setup

Version 1.3 – Dec ‘25

Agenda

1. Hardware and Software overview
2. Setup & Demo Examples
3. Documents & Related Resources

1 - Hardware and Software overview

life.augmented

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
https://www.st.com/en/evaluation-tools/steval-stwinbx1.html

life.augmented

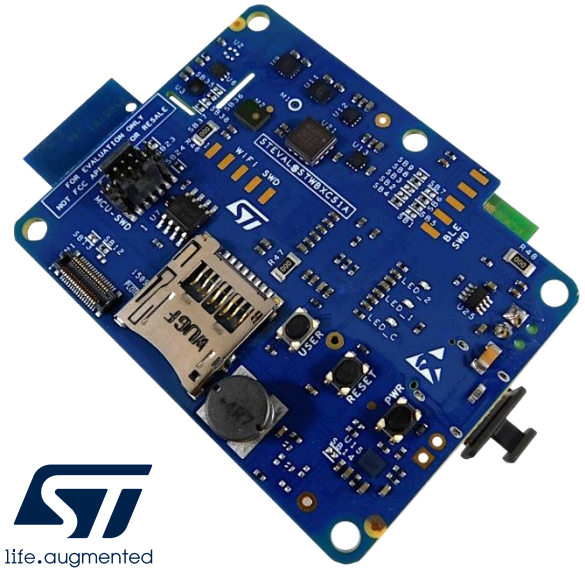
STWIN.box development kit - STEVAL-STWINBX1
Hardware Overview

STWIN.box - SensorTile Wireless Industrial Node

The STEVAL-STWINBX1 development kit includes:

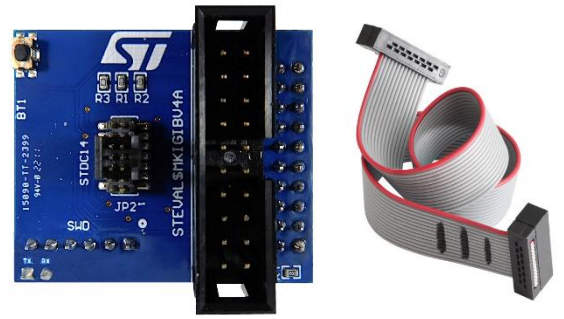
- The STEVAL-STWBXCS1 STWIN.box core system (main board);
- A plastic case with M3 bolts;
- A 480 mAh 3.7 V LiPo battery;
- The STEVAL-MKIGIBV4 ST-LINK adapter with programming cable;
- The STEVAL-C34DIL24 adapter board for DIL24 sensors with the STEVAL-FLTCB01 flexible cable.

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

STWIN.box Modular platform expansion - STEVAL-PDETECT1
Hardware Overview

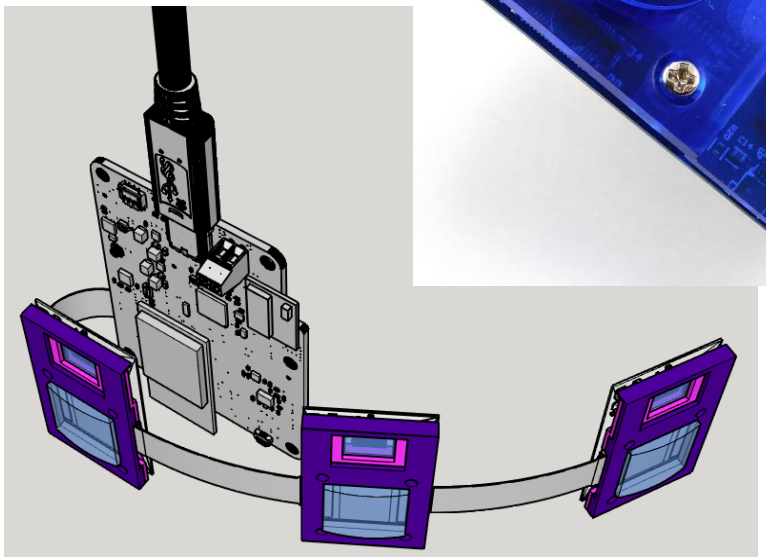
Presence Detection add-on for STWIN.box

Add-on evaluation kit connected to STWIN.box targeting human presence applications.

Key Sensor Products:
- VD6283TX45/1 Ambient Light Sensor
- VL53L8CXV0GC Proximity Sensor
- STHS34PF80 Far Infrared TMOS sensor

The kit includes:
- STEVAL-PDETCS1 board
- TMOS long range Fresnel lens: TMOS63-10
- Plastic case for TMOS63-10 Fresnel lenses
- Time-of-Flight cover glass: IR136C0-IC09-A066
- STEVAL-FLTCB03 flex cable

FP-SNS-DATALOG2 contains PDetect firmware example for STEVAL-STWINBX1 + STEVAL-PDETECT1. It can support up to 3 STEVAL-PDETECT1 connected simultaneously for multi sensors tracking

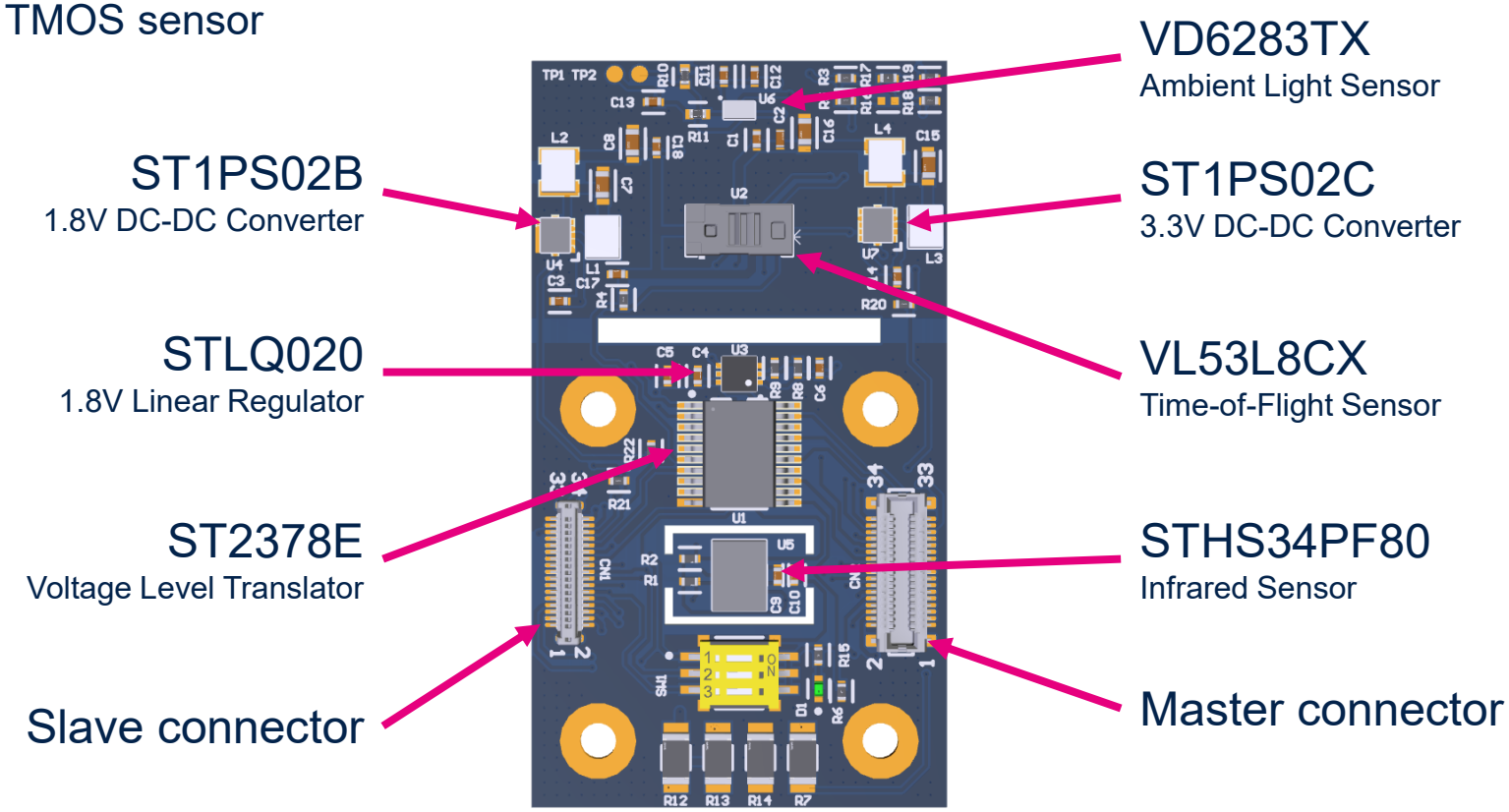


life.augmented

STWIN.box Modular platform expansion - STEVAL-PDETECT1
Hardware Overview

Presence Detection add-on for STWIN.box

- Key Sensor Products:
  - VD6283TX45/1 Ambient Light Sensor
  - VL53L8CXV0GC Proximity Sensor
  - STHS34PF80 Far Infrared TMOS sensor



FP-SNS-DATALOG2
Software Overview

Software Description

The ST High Speed Datalog (FP-SNS-DATALOG2) is a comprehensive multisensor data capture and visualization toolkit, engineered to facilitate the development of embedded data science applications

ST High Speed Datalog is compatible with STDATALOG-PYSDK, a data-centric design and user-friendly Python SDK, and can run with hardware boards that supply real-time data streams, empowering users with full control of the data acquisition process. The included firmware is compatible with the STBLESensor app, which also lets users manage the board and sensor configurations, start/stop data acquisition on a microSD™ card, and control data labeling.

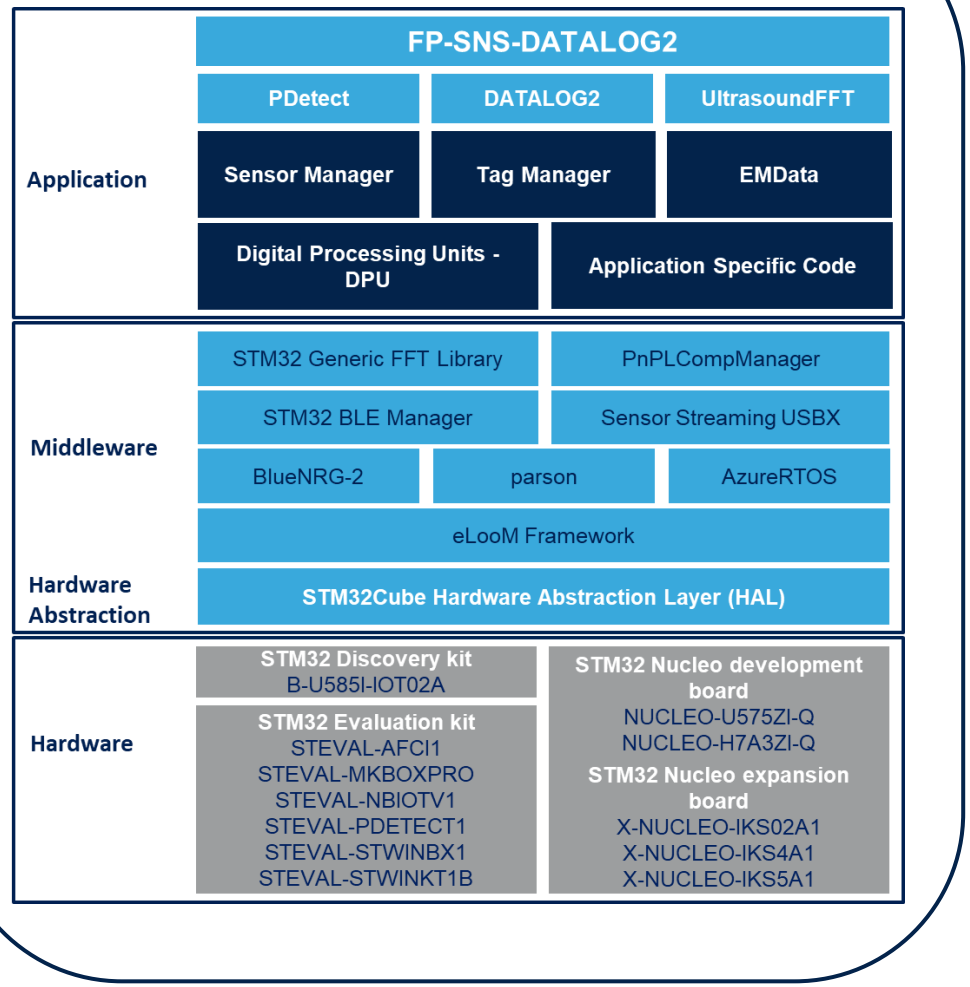
The FP-SNS-DATALOG2 firmware can run on STEVAL-STWINBX1, STEVAL-STWINKT1B, STEVAL-MKBOXPRO, STEVAL-AFCI1, STEVAL-NBIOTV1, B-U585I-IOT02A, and X-NUCLEO-IKS02A1, X-NUCLEO-IKS4A1 and X-NUCLEO-IKS5A1 with NUCLEO-U575ZI-Q or with NUCLEO-H7A3ZI-Q.

ST High Speed Datalog also natively supports STEVAL-PDETECT1, STEVAL-C34KAT1, STEVAL-C34KAT2, STEVAL-C34KPM1, STEVAL-C34DIL24, STEVAL-MKI230KA, STEVAL-MKI245KA, STEVAL-MKI246KA, STEVAL-MKI248KA and SENSEVAL-SCB4XV1 addons for the STEVAL-STWINBX1.

It also supports STEVAL-MKI153V1, STEVAL-MKI223V1K, STEVAL-MKI229A, STEVAL-MKI234KA, STEVAL-MKI240KA, STEVAL-MKI247A and STEVAL-MKI251A add-ons for STEVAL-MKBOXPRO.

The ST High Speed Datalog is part of the ST Edge AI Suite, which is an integrated collection of software tools designed to facilitate the development and deployment of embedded AI applications.

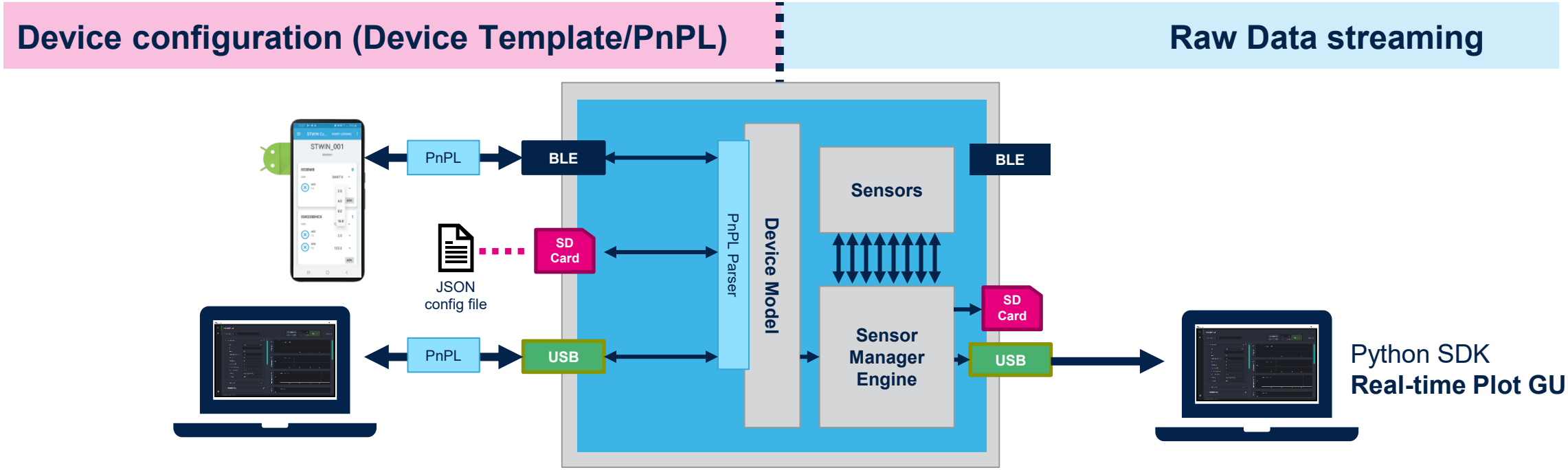
Overall Software Architecture



life.augmented

DATALOG2 demonstration

Optimized STM32 FW Supports streaming of all Sensors at Full data rate



- Based on **AzureRTOS**
  - ThreadX, FileX, USBX
- USBX WCID Streaming class and PC DLL

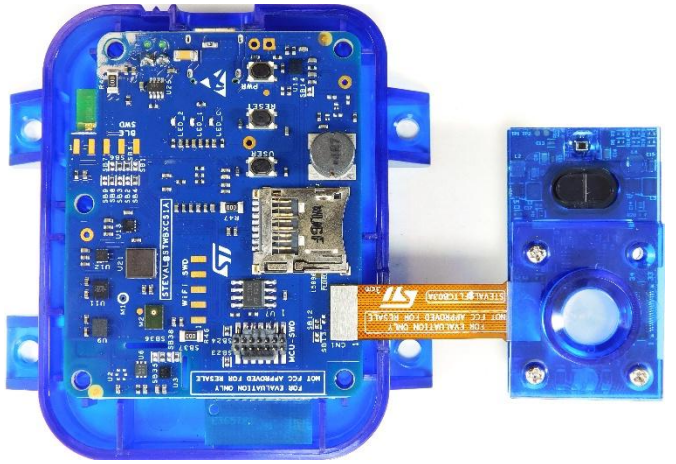
- SD Card
- Full control of acquisition via BLE app
- Control via **PnP-Like** commands

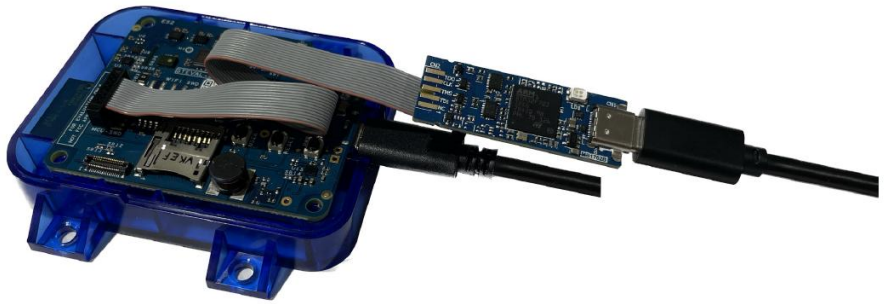
life.augmented

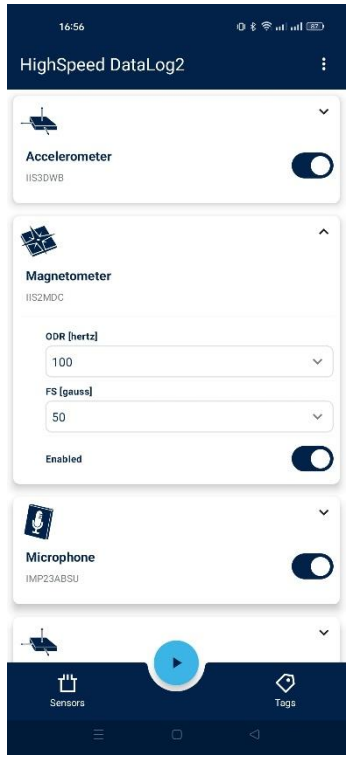
2 - Setup & Demo Examples

HW prerequisites

- 1 STEVAL-STWINBX1
- 1 STEVAL-PDETECT1
- Laptop/PC
- 2 type-C USB cables + 1 STLINK-V3MINIE
  - In alternative it is also possible to directly program the STWIN.box with a single USB-C cable (refer to firmware update – USB)
- 1 micro-SD card
- 1 smartphone with ST BLE Sensor App





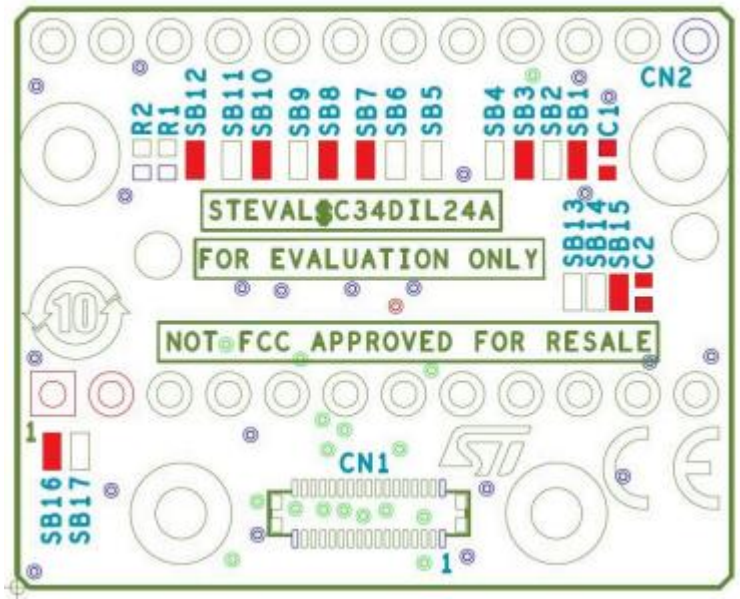


HW prerequisite

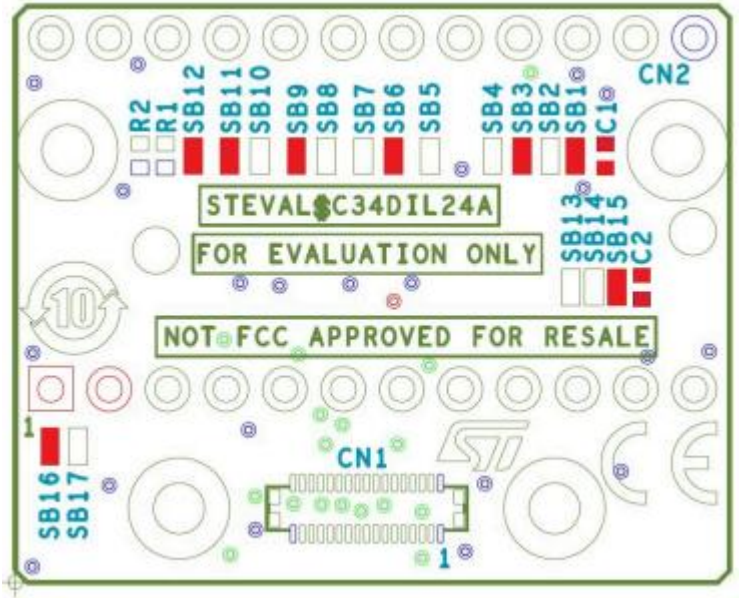
- STEVAL-STWINBX1 is also compatible with external add-ons, like STEVAL-PDETECT1, STEVAL-C34KAT1/2 and STEVAL-C34KPM1
- The included DIL24 adapter (STEVAL-C34DIL24) allows to further expand the capabilities of the STWIN.box
- Notice that hot plug is not supported. Add-ons must be attached to the STEVAL-STWINBX1 before powering it.
- Please be careful when you remove the Flex cable as you may damage it. The safest way to remove it is by pulling it next to the connectors using tweezers.

HW prerequisites

- By default, the DIL24 adapter is configured in SPI mode. Default configuration is valid with STEVAL-MKI230KA and STEVAL-MKI246KA.

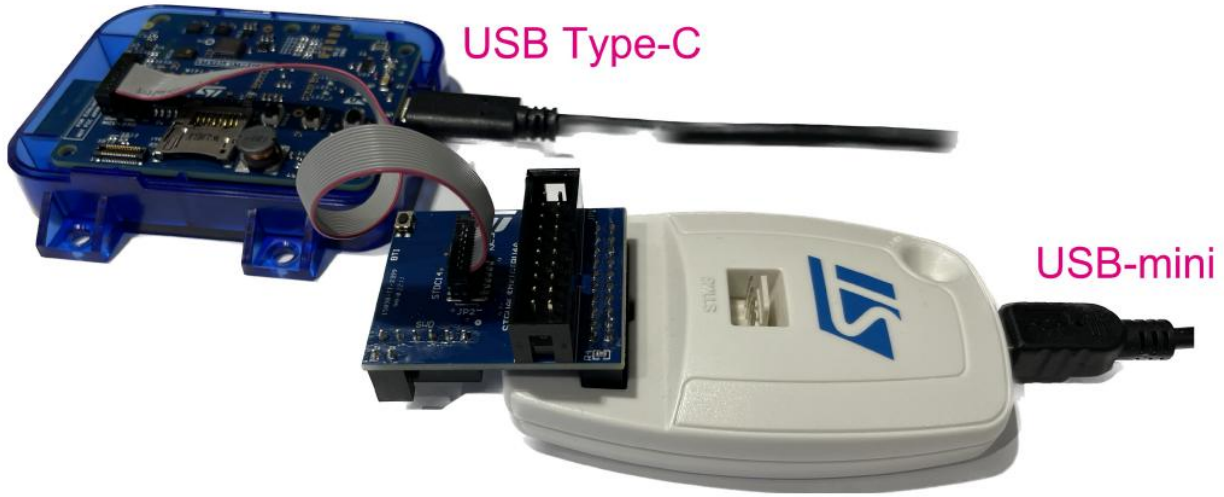


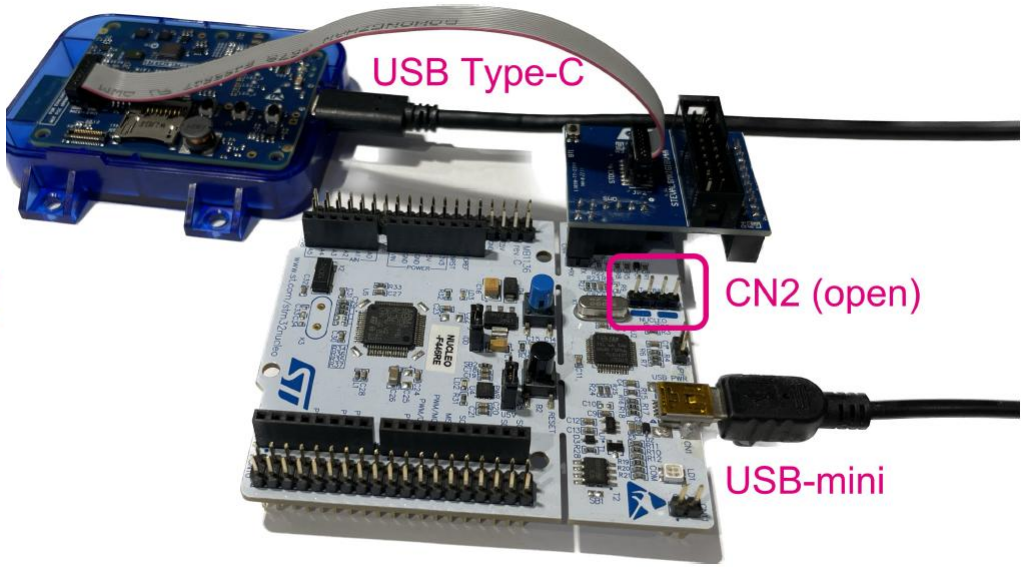
- If you need to use I2C sensors (i.e.: with STEVAL-MKI223V1K or SENSEVAL-SCB4XV1), you need to modify the solder bridges as following:
  - Remove SB7, SB8, SB10
  - Solder SB6, SB9, SB11



Programming options

- It is possible to directly program the STWIN.box with a single USB-C cable (refer to firmware update – USB)
- If you are interested in FW debugging, STWIN.box programming connector is natively compatible with STLINK-V3 debuggers family (STLINK-V3SET or STLINK-V3MINI). STLINKV3 programmers are NOT included in the kit.
- Alternatively, to offer more alternatives, an adapter to ST-Link V2-1 (STM32-Nucleo) or standard JTAG connector is included in the kit.





life.augmented

Software and other prerequisites

- **STM32CubeProgrammer Software**
  - Download and install STM32CubeProgrammer
- **PDetect**
  - Download the FP-SNS-DATALOG2 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 (Keil, IAR, STM32CubeIDE)
- **ST BLE Sensor App**
  - Download and install ST BLE Sensor App (for both Android and iOS – v5.2 and above)
- **Python**
  - To save, plot and elaborate data, Python utility scripts are distributed in a dedicated SDK called STDATALOG-PYSDK

PDetect is **not** the default firmware.
To update the firmware, please follow the instructions for Firmware Update in the next slides

life.augmented

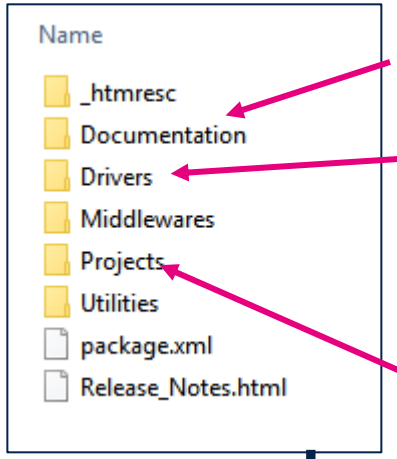
Samples demonstrations

1 **www.st.com**

2 Select:
FP-SNS-DATALOG2

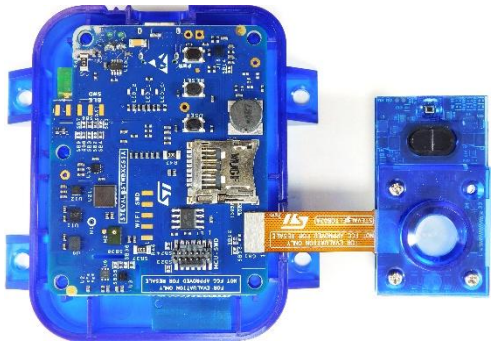
3 Download & unpack

Package structure



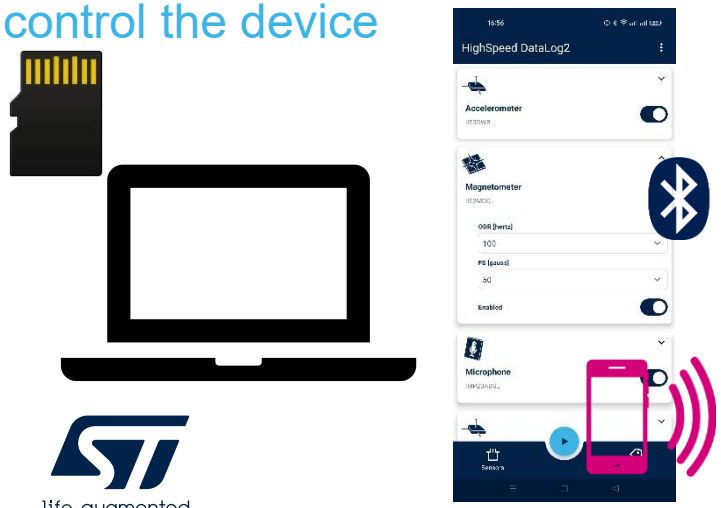
4 Use the pre-compiled binaries or re-compile the code customizing your device configuration

KEIL Tools by ARM · IAR SYSTEMS · STM32 CubeIDE



5

6 Visualize log of sensors data and control the device



life.augmented

Firmware Update – ST BLE Sensor app

PDetect **is not** the default firmware, so it needs to be downloaded on the board by the user.

The default firmware for STWIN.box enables Bluetooth pairing via **NFC** and fast firmware over-the-air upgrade through the **ST BLE Sensor app**.

By turning on Bluetooth and NFC on the smartphone and placing the smartphone on top of the NFC antenna of the STWIN.box, the smartphone reads the Bluetooth pairing information and automatically loads the ST BLE Sensor.

In alternative, just open manually the application. During BLE pairing, if requested, insert the following PIN: **123456.**

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.

Firmware Update – ST-LINK

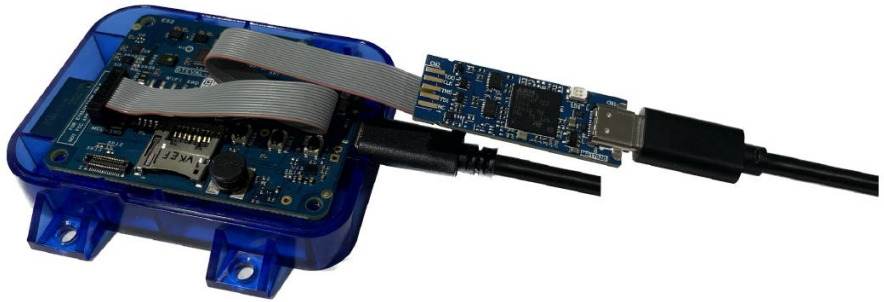
An alternative way is to use the **pre-compiled binary** provided in the package (i.e.: *Projects\STM32U585AI-STWIN.box\Applications\PDetect\Binary)*

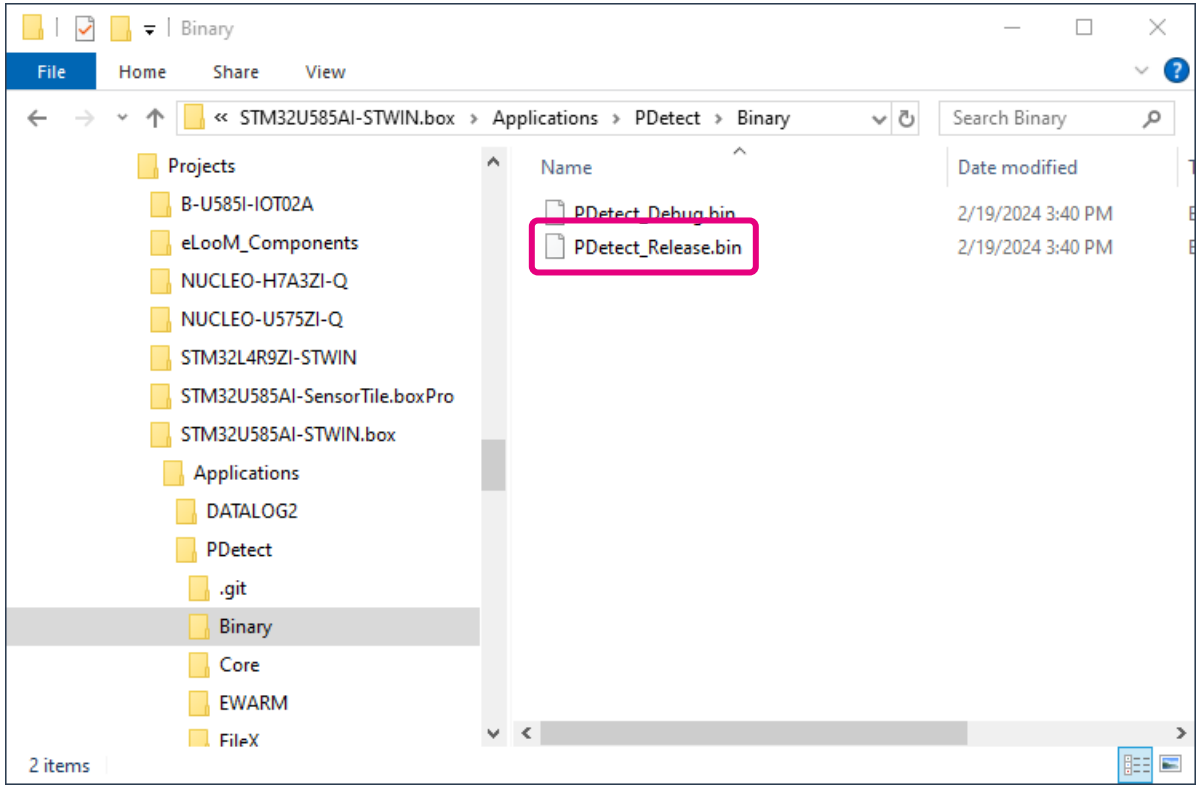
To update the firmware:

- Connect the board to the preferred programmer (here we are using STLINK-V3MINIE).
- Connect both the boards to a PC through the proper USB cables.
- Open STM32CubeProgrammer, select the proper binary file and download the firmware.

For further details, see **UM2965** for

STWIN.box





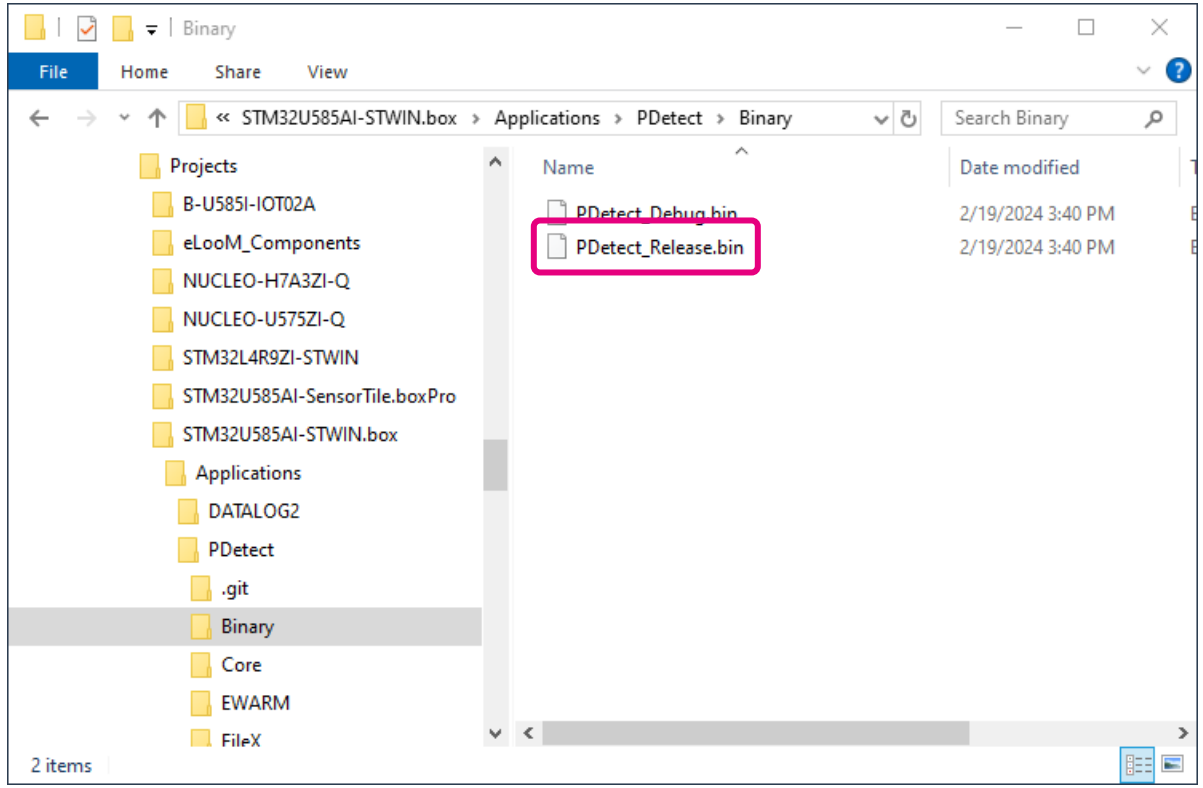
life.augmented

Firmware Update – USB

STEVAL-STWINBX1 can also 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 USR button in STWIN.box
- While keeping the button pressed, connect the USB cable to the PC.
- Now the board is in DFU mode. Open STM32CubeProgrammer, select the proper binary file and download the firmware.



For further details, see **UM2965** for STWIN.box

2.1 – PDetect Demonstration

2.1.1 - USB sensor data streaming Real Time Plot

From HSDPython_SDK to STDATALOG-PYSDK

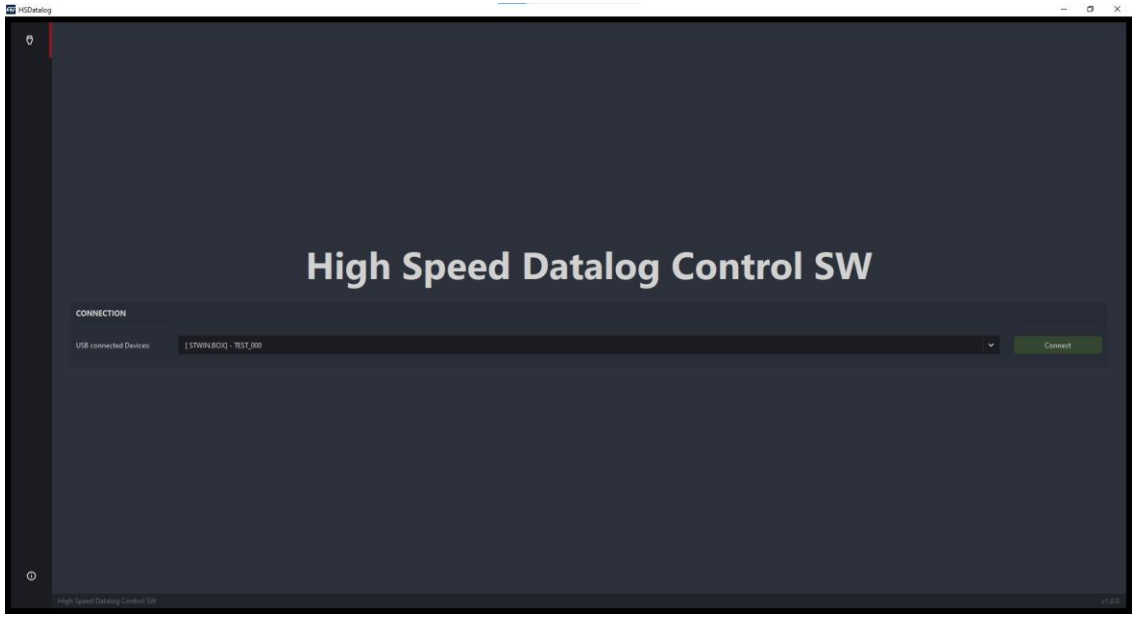
- From FP-SNS-DATALOG2 v3.0.0, HSDPython_SDK has been expanded and moved to a separate software product: STDATALOG-PYSDK
- The STDATALOG-PYSDK is a comprehensive Python framework designed to facilitate the capture, processing, and visualization of data from a wide range of sources, including sensors, algorithms, simulated signals, and telemetry from actuators.
- It is compatible with all firmware examples available in FP-SNS-DATALOG2, FP-IND-DATALOGMC, and FP-SNS-DATALOG1
- The python software development kit (SDK) for data logging has been developed using Python 3.13.
  - It is compatible with Python 3.10, 3.11, and 3.12 also.
  - Python must be already properly installed on the user's machine before installing and running STDATALOG-PYSDK.
  - *STDATALOG-PYSDK* requires different Python modules. The package is distributed with *installers* that solve all the required dependencies
- The next slides show how to use the *stdatalog_GUI.py* script available in the SDK. Please see the full documentation available on the STDATALOG-PYSDK landing page to install the SDK properly on your machine

life.augmented

Execute *stdatalog_GUI.py*

- FP-SNS-DATALOG2 provides an example to manage data from up to 3 STEVAL-PDETECT boards in daisy chain.
- *stdatalog_GUI.py* is the reference Python script within the STDATALOG-PYSDK, developed in Python 3.13 on Windows, Linux and macOS environments.
  - *STDATALOG-PYSDK* requires different Python modules. The package is distributed with *installers* that solve all the required dependencies
  - Please see the full documentation available on the STDATALOG-PYSDK landing page to install the SDK properly on your machine

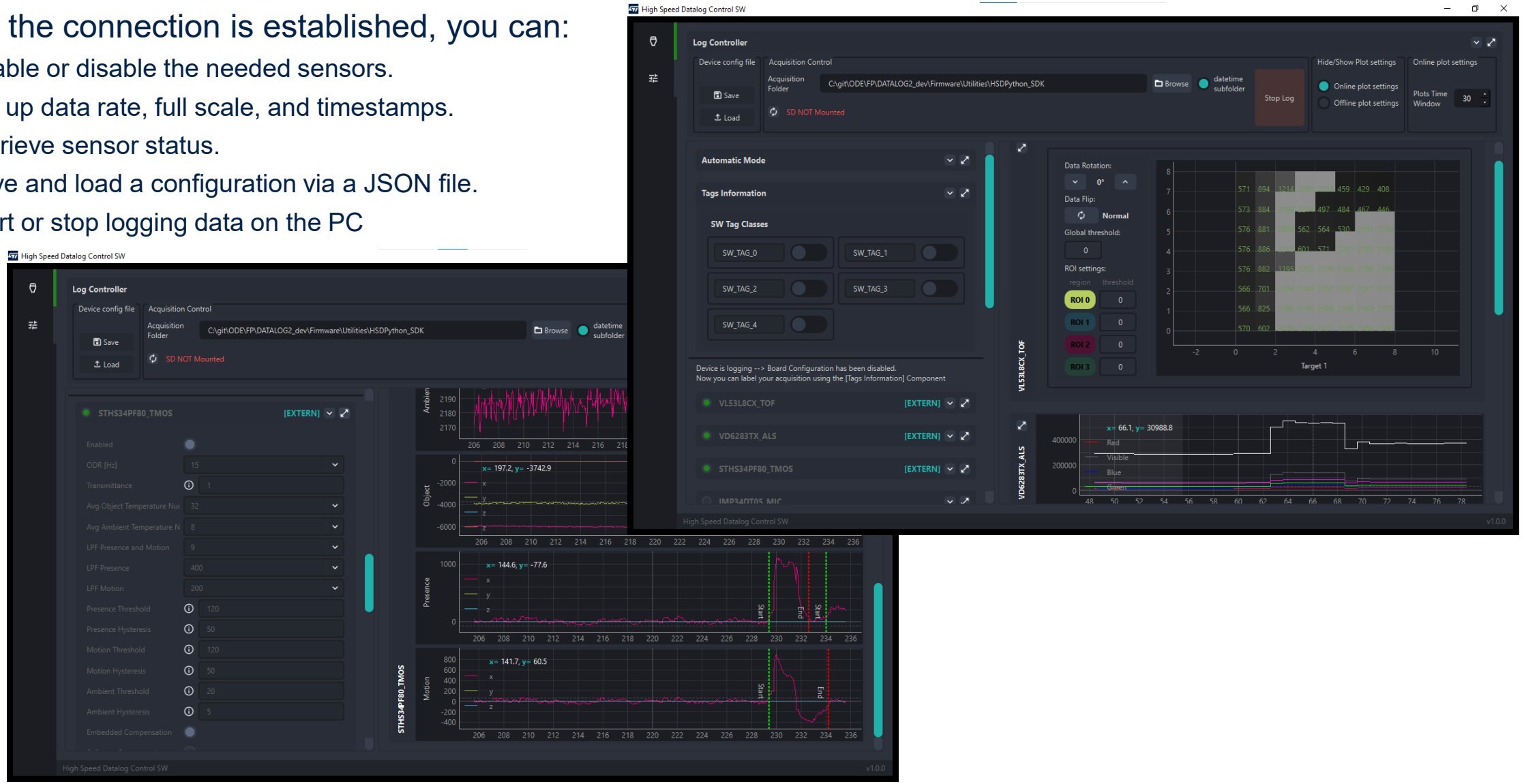
- Once the board is connected via USB and the Python environment has been properly updated, you can launch the real-time plot by executing *stdatalog_GUI.py* available in *stdatalog_examples\gui_applications\stdatalog\GUI*.
  - Depending on your local setup, to execute the script, you can open a command shell there and run *python stdatalog_GUI.py*.
- Click on the Connect button to allow the connection between the board and the PC.



life.augmented

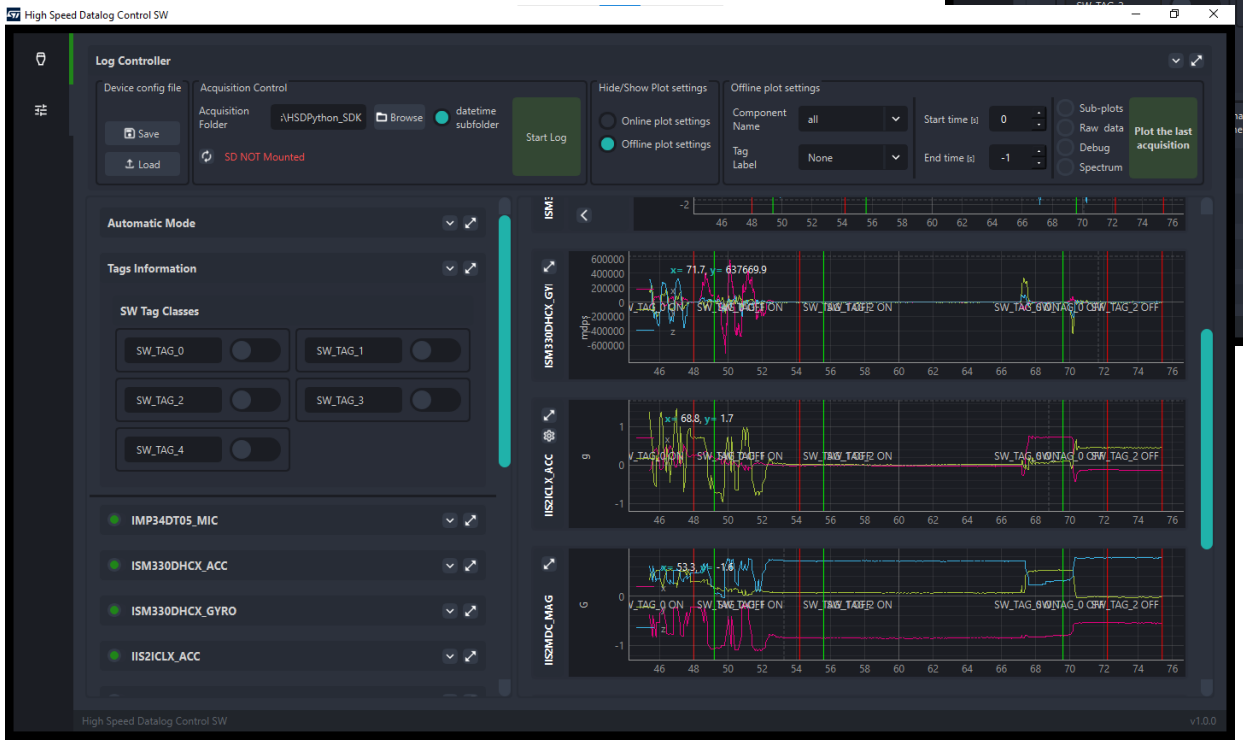
stdatalog_GUI.py

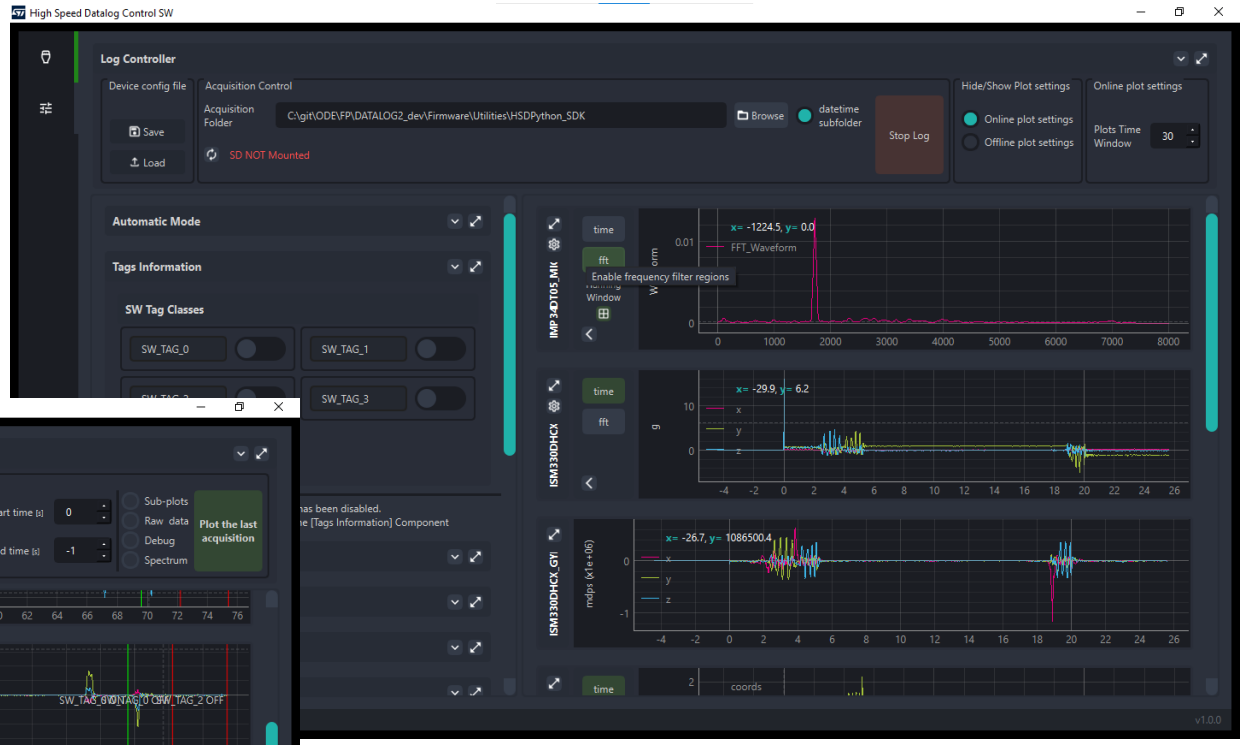
- Once the connection is established, you can:
  - Enable or disable the needed sensors.
  - Set up data rate, full scale, and timestamps.
  - Retrieve sensor status.
  - Save and load a configuration via a JSON file.
  - Start or stop logging data on the PC



stdatalog_GUI.py

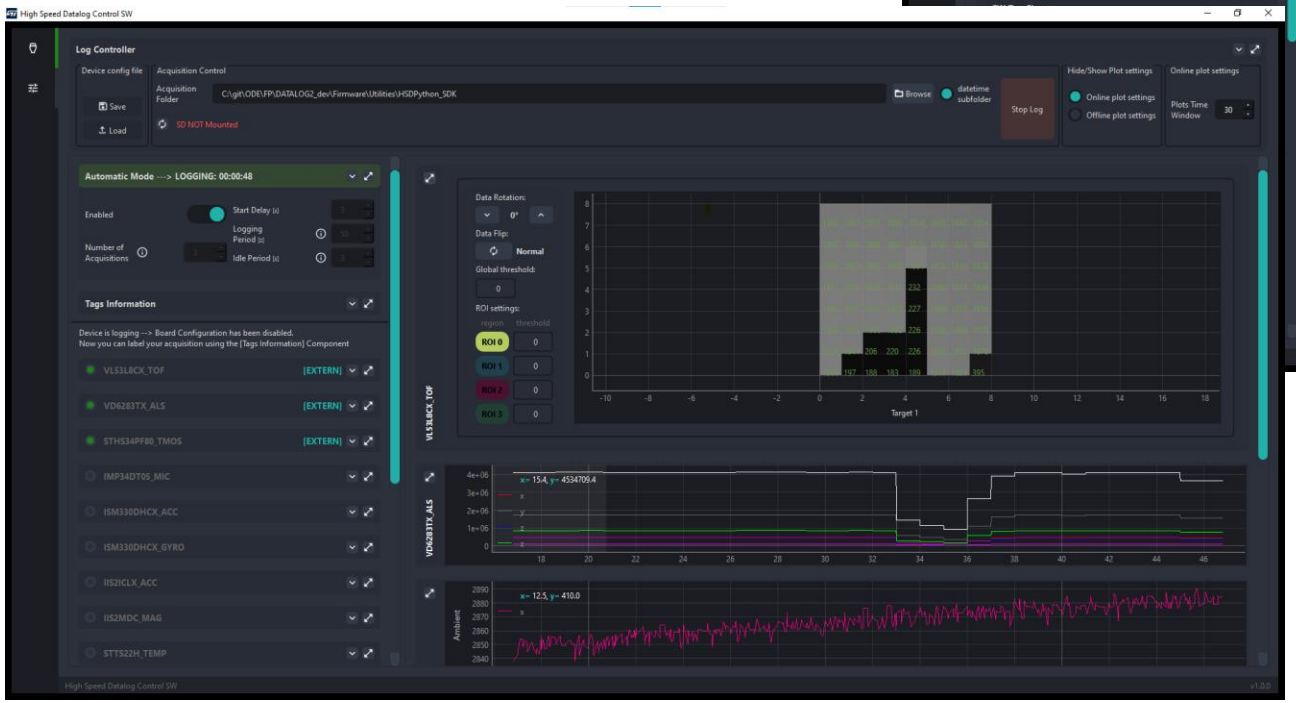
- Real Time Plot GUI also allows you to:
  - Set up tag classes and handle data tagging and labeling of an ongoing acquisition.
  - Enable FFT calculation for accelerometers and microphones.
  - Set up the acquisition name and description.





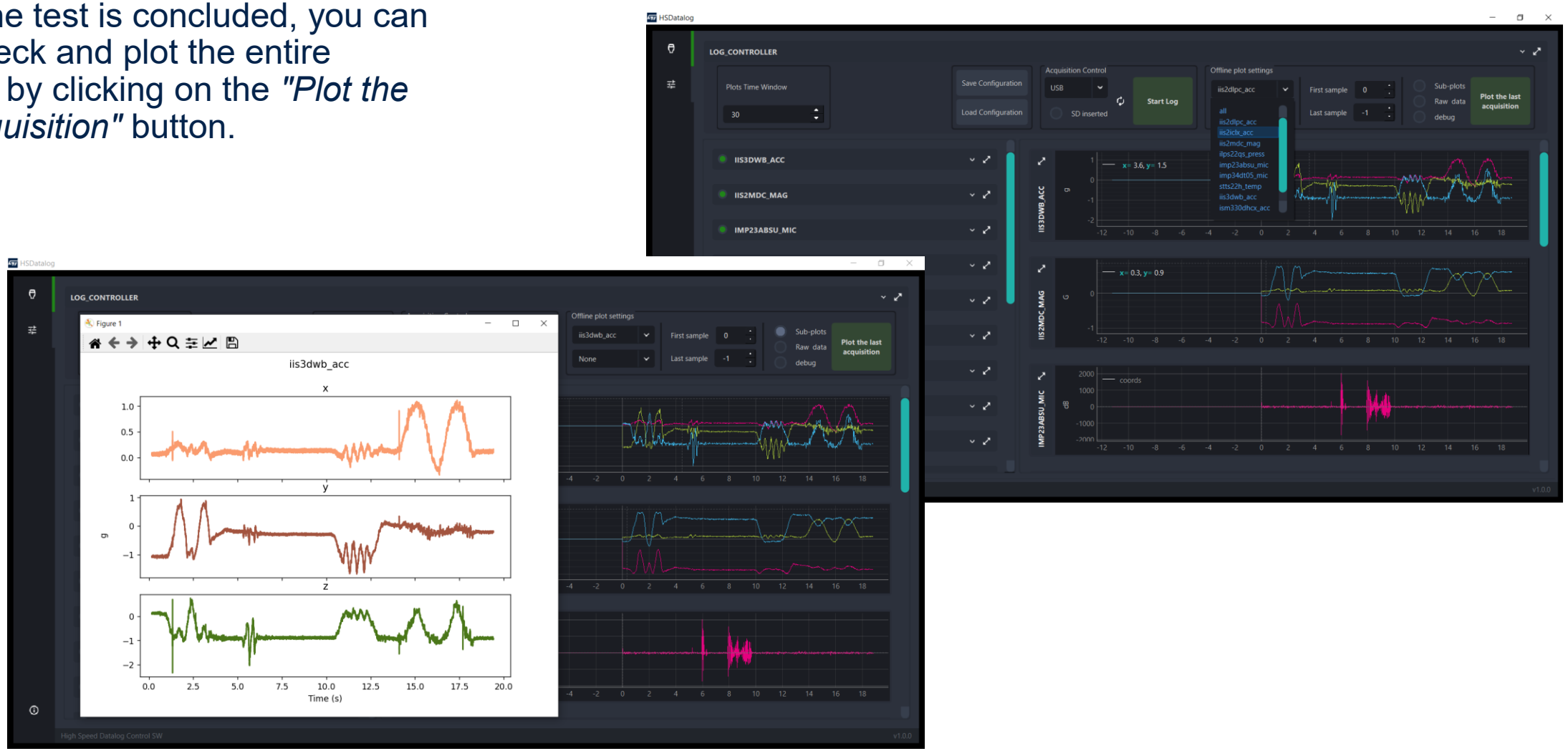
stdatalog_GUI.py

- Furthermore, you can set up Automode to start the datalog operations or to pause all executions for specific periods.
  - Set up the desired values in the Automatic Mode widget
  - See Automode section for further details



stdatalog_GUI.py

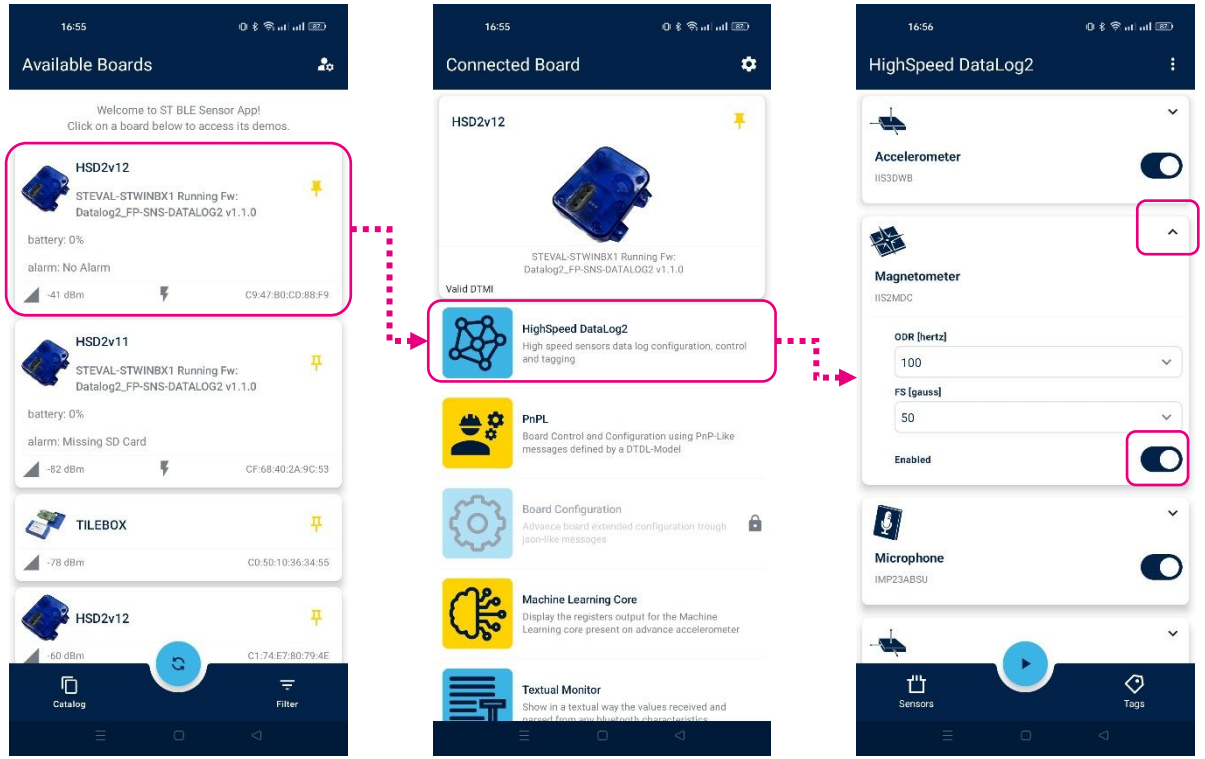
- Once the test is concluded, you can also check and plot the entire dataset by clicking on the *"Plot the last acquisition"* button.



2.1.2 – Data logging on SD card, configuration with BLESensor App

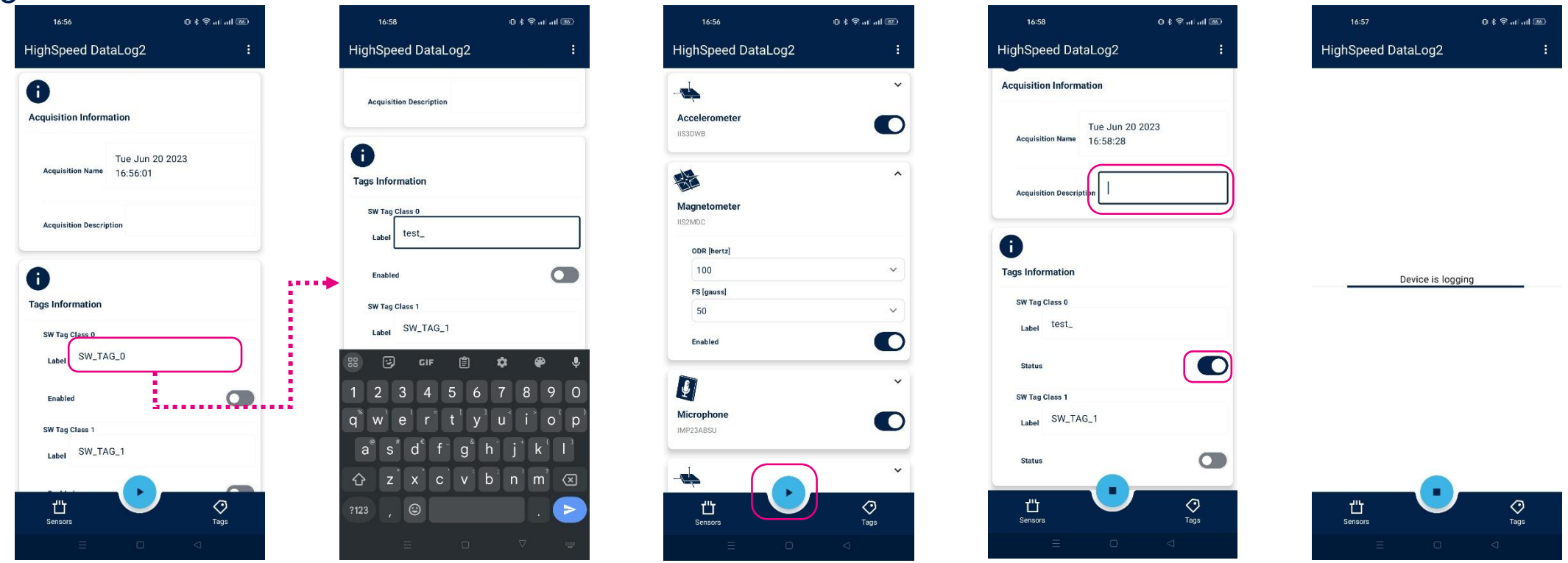
ST BLESensor App: DATALOG2 tab

- PDetect 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 the board and sensor configurations, start/stop data acquisition on SD card and control data labelling.
- Once connected, you can configure the device by:
  - enabling/disabling a specific sensor
  - changing sensor parameters



Acquisition settings and control

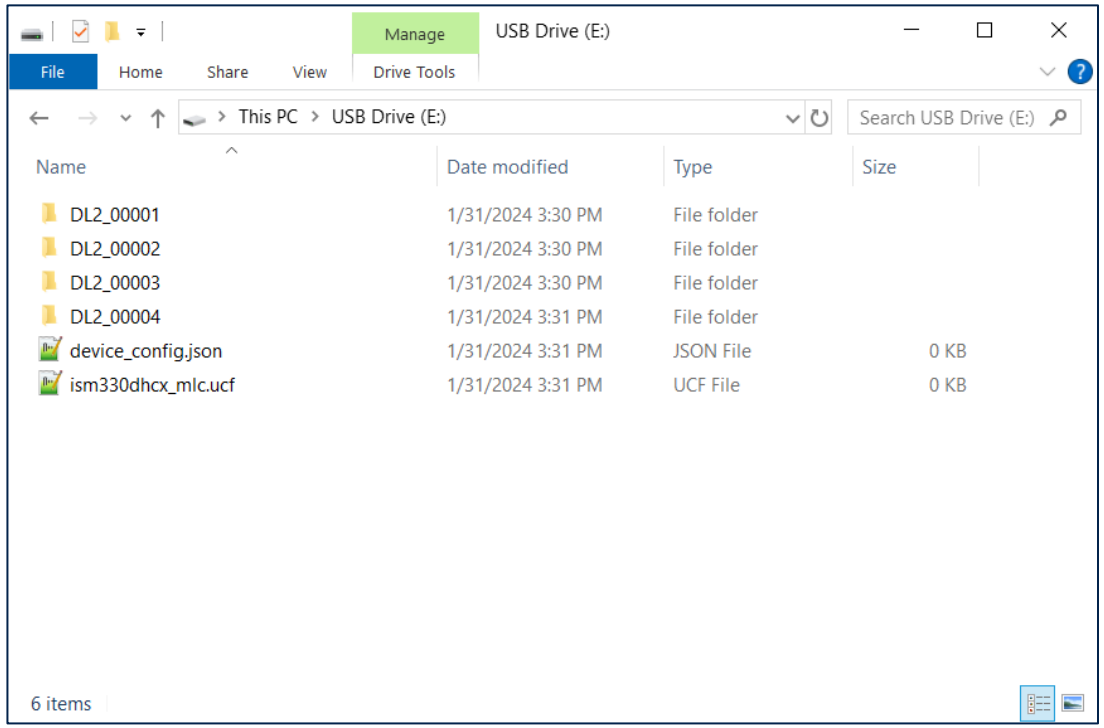
- By clicking the tags button, you can switch to the acquisition settings and control tab to:
  - start and stop an acquisition (to an SD card)
  - choose which tag classes will be used for the next acquisition
  - handle data tagging and labeling of an ongoing acquisition
  - set up the acquisition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



2.1.3 – Data logging on SD card, standalone mode

Start an acquisition in standalone mode

- PDetect can work also standalone, saving selected sensor data at the highest possible rate into the SD card
- PDetect can read a custom sensors configuration from the SD card root folder. To do so, you can simply save a JSON configuration file in the root folder of the SD card



- Once the firmware is flashed on the board:
  - Insert the SD card
  - If the board is battery-powered and switched off, press PWR button to switch on the board. Press the RESET button
  - If the SD card is not inserted properly, the orange LED will be switched off. Otherwise, the orange LED will be switched on
  - If a JSON configuration file is present in the root folder of the SD card, the custom sensor configuration is loaded from the file itself
  - Press the USR button to start saving data. During datalog stage you will see the green LED blinking at 4Hz
  - To stop the data acquisition, press again the USR button. During idle stage you will see the green LED blinking at 1 Hz

Automode

- PDetect also features the automode, which can be initiated automatically at the device power-up or reset
- This mode can be used to start the datalog operations or to pause all the executions for a specific period by putting the sensor node in the "idle" phase.
- Automode allows automatically saving data on the SD card, generating different acquisitions folders. It can be useful to automate long acquisition setups, avoid datasets that are too large and reduce SD card errors by avoiding data loss through autosaving.

```
"automode": {
    "enabled": true,
    "nof_acquisitions": 7,
    "start_delay_s": 5,
    "logging_period_s": 30,
    "idle_period_s": 5,
    "c_type": 2
}
```

Automode

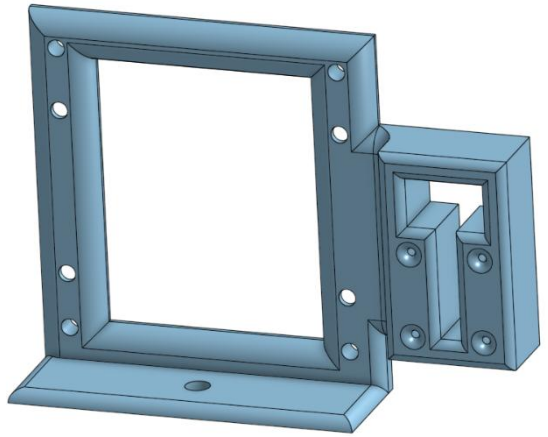
- As for the standalone mode, to enable the automode you must set up the automode component properly in the device_config.json:
  - **enabled**: if true, the automode starts after the reset and node initialization. If false, automode is not executed.
  - **nof_acquisitions**: gives the number of times the automode is executed; 0 indicates an infinite loop and it is the default value.
  - **start_delay_s**: indicates the initial delay in seconds applied after reset and before the first execution phase starts. Minimum valid value is 1.
  - **logging_period_s**: specifies the duration in seconds of the datalog phase. Minimum valid value is 1.
  - **idle_period_s**: specifies the duration in seconds of the idle phase. Minimum valid value is 1.
  - **c_type**: describes the component type (0 “sensor”, 1 “algorithm”, 2 “other”).
- Then place it in the root folder of the SD card.
- If the board is battery-powered and switched off, press the PWR button to switch on the board. Press the RESET button.

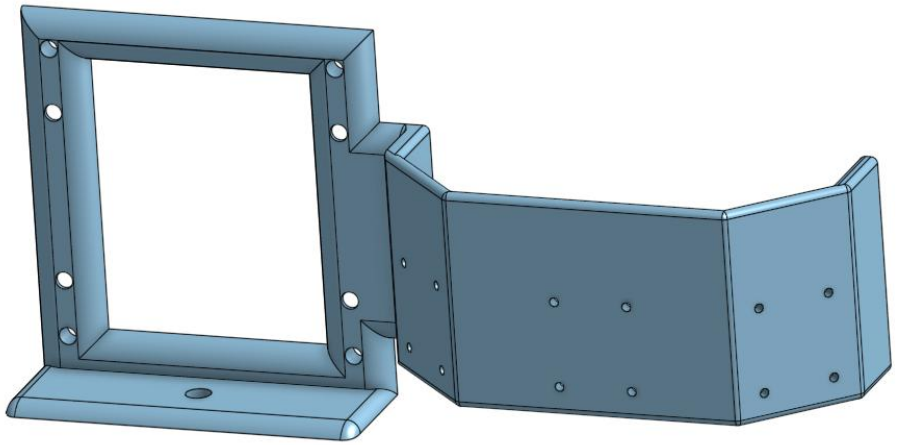
```
"automode": {
    "enabled": true,
    "nof_acquisitions": 7,
    "start_delay_s": 5,
    "logging_period_s": 30,
    "idle_period_s": 5,
    "c_type": 2
}
```

2.2 – STEVAL-PDETECT1 Mounting Holders

STEVAL-PDETECT1 Mounting Holders

- Example 3D CAD Holders are available on st.com-STEVAL-PDETECT1 page under CAD Resources
- Holder examples available for mounting either 1 STEVAL-PDETECT1 board or 3 STEVAL-PDETECT1 boards along with the STEVAL-STWINBX1
- You can use the holder to mount the sensors onto various positions, such as on the wall, ceiling, desk, and tripod directly for application evaluation + testing

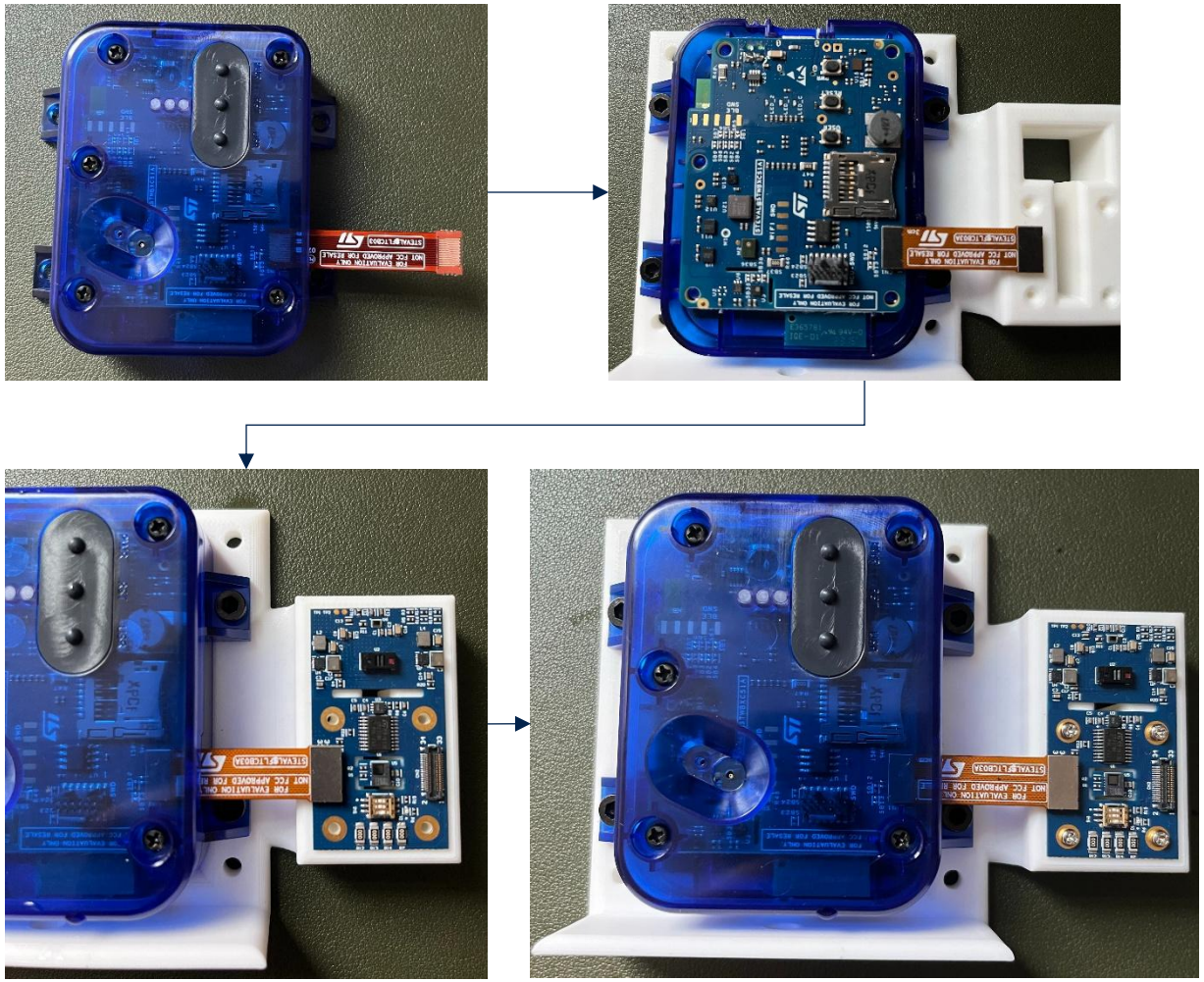




life.augmented

1 STEVAL-PDETECT1 Holder Installation Guide

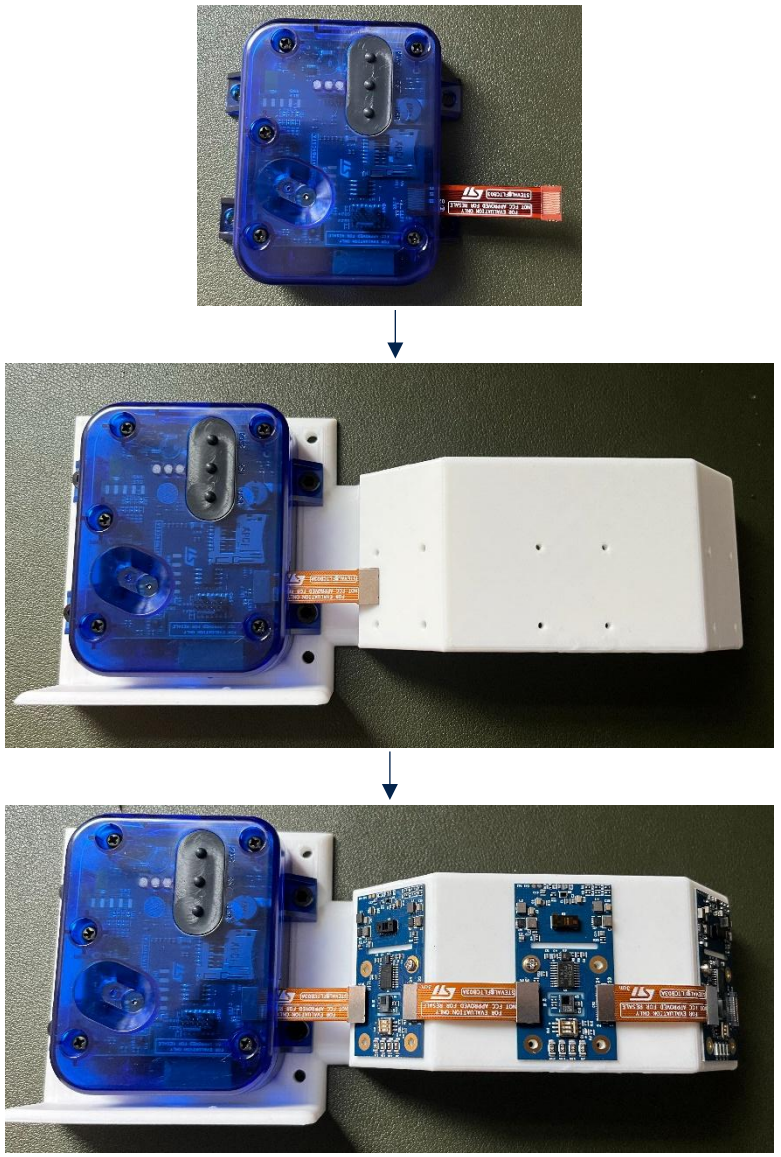
1. Securely mount the STEVAL-STWINBX1 onto the case using 5 provided screws and attach the short ribbon cable in the STEVAL-PDETECT1
2. Mount the casing onto the holder using M4 – 8mm Screws + Bolt
3. Connect STEVAL-PDETECT1 board to the STWINBX1 with the flex cable
4. Mount the STEVAL-PDETECT1 board onto the holder using 4 screws provided in the kit



life.augmented

3 STEVAL-PDETECT1 Holder Installation Guide

1. Securely mount the STEVAL-STWINBX1 onto the case using 5 provided screws and attach the short ribbon cable in the STEVAL-PDETECT1
2. Mount the casing onto the holder using M4 – 8mm Screws + Bolt
3. Connect 3 STEVAL-PDETECT1 board to the STWINBX1 with provided flex ribbon cables
4. Mount the 3 STEVAL-PDETECT1 boards onto the holder using screws provided in the evaluation kit



3 - Documents & Related Resources

Documents & Related Resources

FP-SNS-DATALOG2:

- **DB4865:** STM32Cube function pack for high speed datalogging and ultrasound processing– **databrief**
- **UM3106:** Getting started with the STM32Cube function pack for high speed datalogging and ultrasound processing – **user manual**

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**

STEVAL-PDETECT1:

- **Gerber files, BOM, Schematic**
- **DB5165:** Presence detection add-on for STWIN.box – **databrief**
- **UM3320:** Getting started with the STEVAL-PDETECT1 Presence Detection evaluation board – **user manual**

STDATALOG-PYSDK:

- **DB5446**: Python software development kit (SDK) for data logging: complete toolkit with extensive examples for developers – **databrief**

Visit www.st.com for the complete list

Thank you

life.augmented