

STM32Cube function pack for high speed datalogging and ultrasound processing

User Interfaces and utilities	cli_example C++	HSDataLog SDK Python	Acquisition and configuration examples
Application	FP-SNS-DATALOG2		
	DATALOG2		UltrasoundFFT
	Digital Processing Units - DPU	EMData	PnPL Manager
	Sensor Manager	Tag Manager	Application Specific Code
Middleware	BlueNRG-2	Sensor Streaming USBX	STM32 BLE Manager
	STM32 Generic FFT Library	parson	AzureRTOS
	eLooM Framework		
Hardware Abstraction	STM32Cube Hardware Abstraction Layer (HAL)		
Hardware	STEWAL-STWINBX1 evaluation kit		STEWAL-MKBOXPRO evaluation kit



Features

- High-rate (up to 6 Mbit/s) data capture software suite:
 - Python and C++ real-time control and data analysis
 - Dedicated Python SDK, ready-to-use for integration into any data science design flow
 - Compatible with [STBLESensor](#) app for system setup and real-time control
 - Able to configure and enable [ISM330DHCX](#) and [LSM6DSV16X](#) machine learning core unit and [ISM330IS](#) intelligent sensor processing unit (ISPU)
 - Synchronized timestamping and labeling mechanisms common to all sensors
- Generic FFT library middleware to enable frequency domain analysis for any kind of sensor through fast Fourier transform (with programmable size, overlapping, and windowing)
- AzureRTOS: ThreadX, FileX, USBX
- Easy portability across different MCU families, thanks to STM32Cube
- 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

Product summary

Software example for	
STEWAL-STWINBX1 and STEWAL-MKBOXPRO	FP-SNS-DATALOG2
STWIN.box - SensorTile Wireless Industrial Node Development Kit	STEWAL-STWINBX1
Sensortile.box PRO	STEWAL-MKBOXPRO
Firmware runs on:	STM32U585AI16Q
FW development environments	- Keil - IAR Embedded Workbench - STM32CubeIDE
Other utilities	STBLESensor mobile Android/iOS app
Applications	Condition Monitoring / Predictive Maintenance Sensing

Description

The [FP-SNS-DATALOG2](#) function pack for the [STEWAL-STWINBX1](#) and [STEWAL-MKBOXPRO](#) represents an evolution of [FP-SNS-DATALOG1](#). It provides a comprehensive solution to save data from any combination of sensors and microphones configured up to the maximum sampling rate.

It also natively supports [STEWAL-C34KAT1](#), [STEWAL-C34DIL24](#) and [STEWAL-MKI230KA](#) add-ons for the [STEWAL-STWINBX1](#).

[FP-SNS-DATALOG2](#) is based on application-level modules (SensorManager, PnPLManager, DPU Digital Processing Units) that a final user can reuse and easily extend to build its custom application. These application modules adopt state-of-the-art design patterns and support natively low-power modes.

To enable this solution, the function pack has been built on top of eLooM, an embedded light object oriented framework for STM32 applications specifically designed for embedded low power applications powered by STM32.

The DATALOG2 application allows storing sensor data onto a microSD™ card (secure digital high capacity - SDHC) formatted with the FAT32 file system, or stream to a PC via USB (WinUSB class) using the companion host software (cli_example) provided for Windows and Linux. It can also configure the [ISM330DHCX](#) and the [LSM6DSV16X](#) machine learning core unit (MLC) and the [ISM330IS](#) intelligent sensor processing unit (ISPU) to read the output of the selected algorithm.

This application allows downloading the selected configuration from a JSON file to the board as well as starting and controlling data acquisition. The same set of commands can be operated from a host via Command Line Interface or via Bluetooth using the [STBLESensor](#) app (available for Android and iOS from v4.17 and above), which lets you manage the board and sensor configurations, start/stop data acquisition on SD card, and control data labeling.

To read and process sensor data acquired using [FP-SNS-DATALOG2](#), a few easy-to-use scripts in Python are provided within the software package.

The scripts have been developed and tested with Python 3.10.

The package also includes an example of ultrasound condition monitoring (UltrasoundFFT) for [STEVAL-STWINBX1](#) that calculates the FFT of the [IMP23ABSU](#) analog microphone signal and streams the result to a PC GUI via USB. The microphone sampling rate is set by default to 192 kHz whereas the microphone bandwidth is up to 80 kHz.

The software is available also on [GitHub](#), where the users can signal bugs and propose new ideas through [\[Issues\]](#) and [\[Pull Requests\]](#) tabs.

Revision history

Table 1. Document revision history

Date	Revision	Changes
24-Jan-2023	1	Initial release.
01-Mar-2023	2	Updated Section Description .
03-Apr-2023	3	Updated Title, Features and Description in cover page. Updated Product summary.

IMPORTANT NOTICE – READ CAREFULLY

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

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

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

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

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

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

© 2023 STMicroelectronics – All rights reserved