

SensorTile.box: Pro Mode

ST MEMS & Sensors training, Italy, November 2019



- Hands-on #0 – Flash a binary
- Hands-on #1 – DataLogExtended example with Unicleo-GUI
- Hands-on #2 – 4D position recognition (FSM example)
- Hands-on #3 – Vibration Monitoring (MLC example)
- Hands-on #4 – 6D position recognition (FSM example)
- Restore the original FW



Hardware and Software check

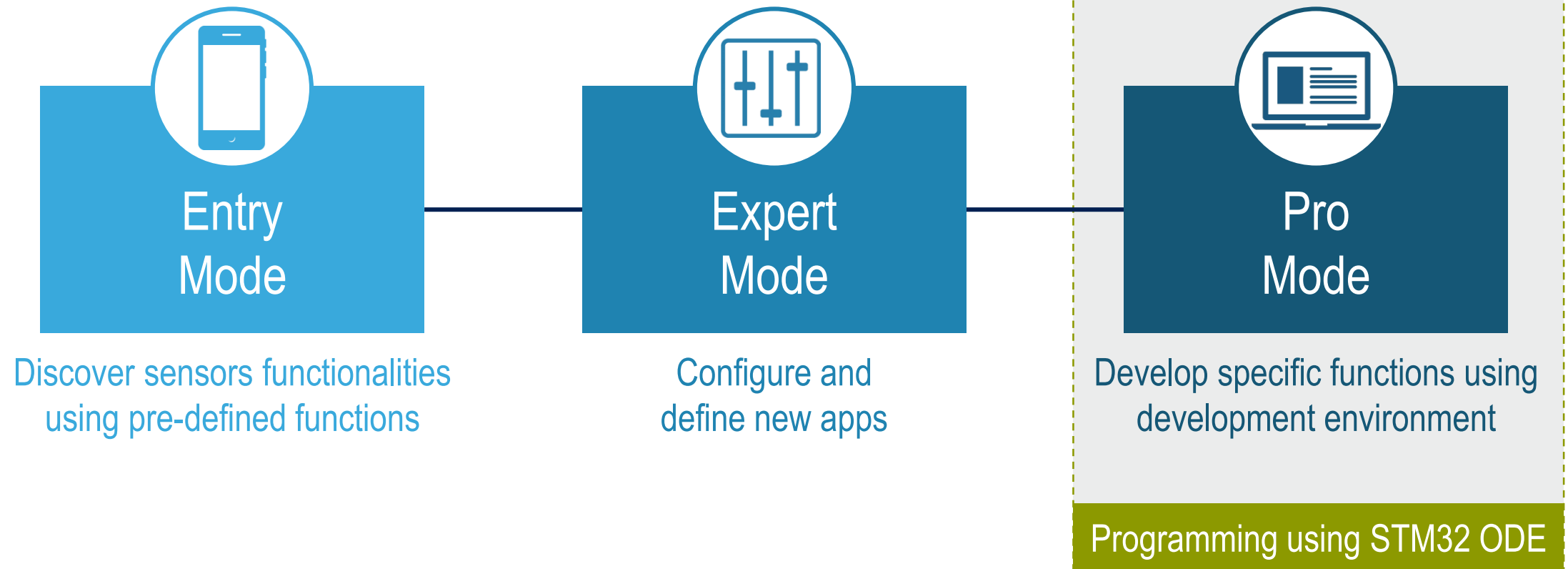
What will we use?

3

Please make sure that you have:

- ✓ Installed **Unicleo-GUI** on your laptop
- ✓ Installed **STM32CubeProg** on your laptop (requires JRE)
- ✓ Installed **Serial terminal** (e.g. **Tera Term**) on your laptop
- ✓ Installed **ST BLE Sensor** app on your smartphone
- ✓ Downloaded **FSM & MLC examples** from GitHub
- ✓ Downloaded **FP-SNS-STBOX1** (FW package)





Programmer

The screenshot shows the STM32CubeProgrammer interface. At the top, it indicates 'Connected' via USB. The main window displays a memory read operation with the following data:

Address	0	4	8	C	ASCII
0x08000000	20000428	0800190D	080018E9	080018EB	(. é
0x08000010	080018ED	080018EF	080018F1	00000000	i ñ
0x08000020	00000000	00000000	00000000	080018F3 ó
0x08000030	080018F5	00000000	080018F7	080018F9	õ ù
0x08000040	08001929	0800192D	08001931	08001935) 1 5
0x08000050	08001939	0800193D	08001941	08001945	9 = A E
0x08000060	08001949	0800194D	08001951	08001955	I M Q U
0x08000070	08001959	0800195D	08001961	08001965	Y] a e
0x08000080	08001969	0800196D	08001971	08001975	i m q u
0x08000090	08001979	0800197D	08001981	08001985	y }
0x080000A0	08001989	0800198D	08001991	08001995
0x080000B0	08001999	0800199D	080019A1	080019A5 j ¥
0x080000C0	080019A9	080019AD	080019B1	080019B5	@ ± μ

The log window at the bottom shows the following messages:

```

15:44:01 : Bank : 0x00
15:44:01 : Address : 0x1ff00000
15:44:01 : Size : 36 Bytes
15:44:01 : Bank : 0x01
15:44:01 : Address : 0x1ff01008
15:44:01 : Size : 28 Bytes
15:44:01 : UPLOADING ...
15:44:01 : Size : 1024 Bytes
15:44:01 : Address : 0x80000000
15:44:01 : Read progress:
15:44:01 : Data read successfully
15:44:01 : Time elapsed during the read operation is: 00:00:00.005
  
```

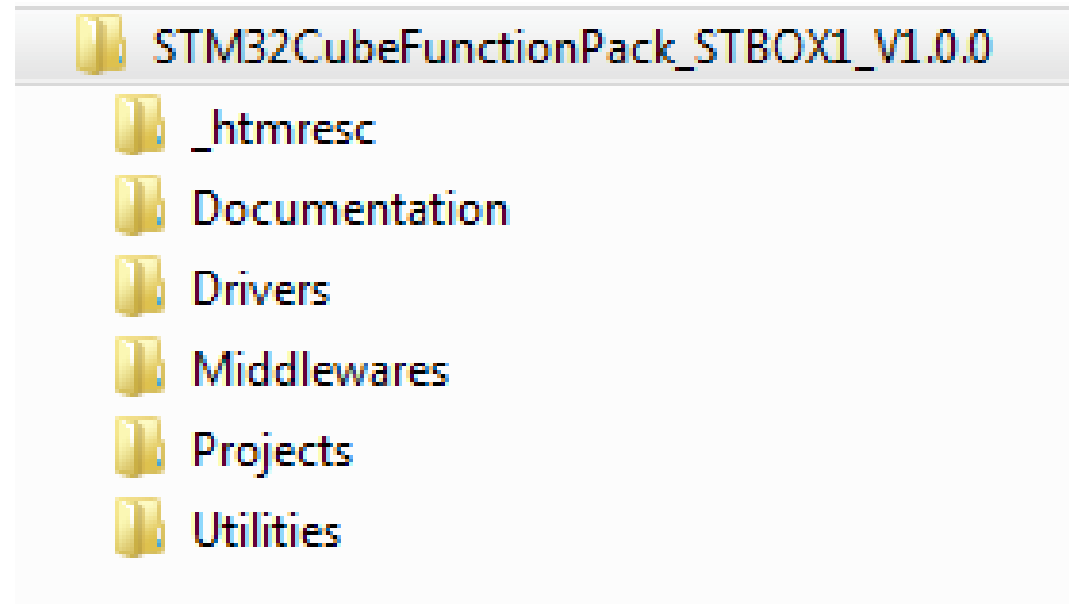
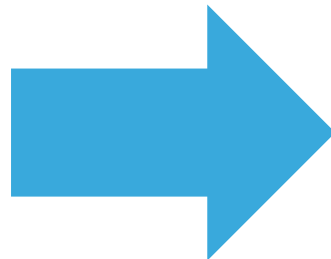
Easy-to-use read/write/verify
Flash, RAM, OTP memory

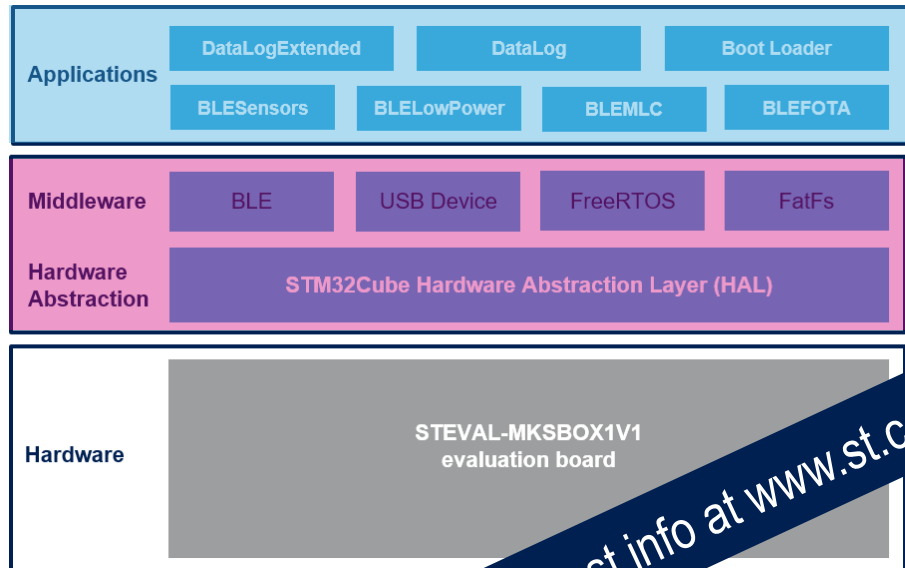
Debug interface (JTAG, SWD)
Bootloader interface (UART, USB)
Over-The-Air (OTA for STM32WB)

Command Line interface
for automation through scripting

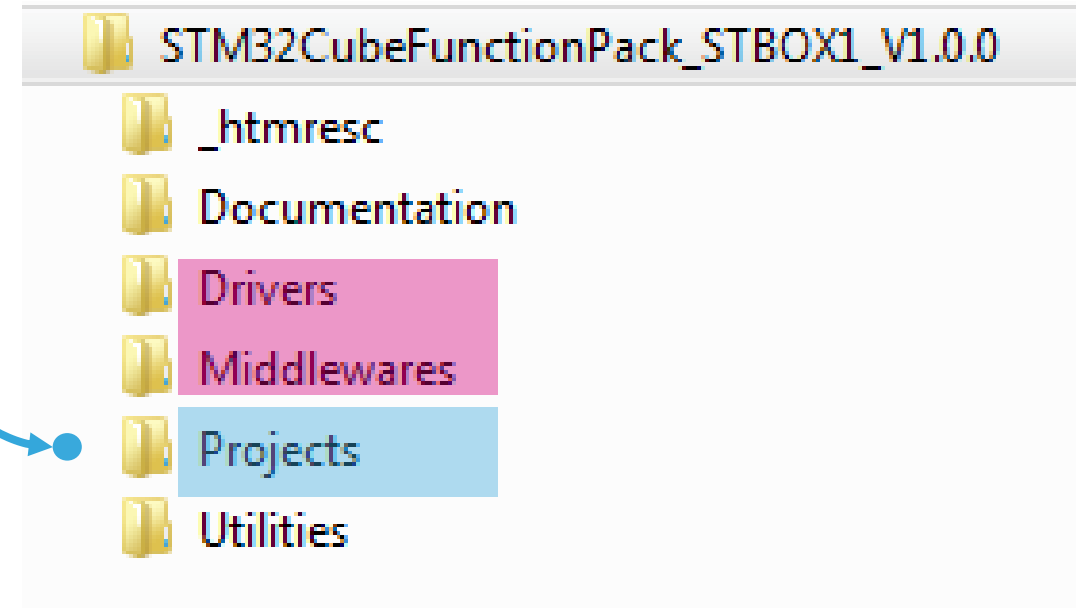


- This is a simple zip of several application examples
- Directory structure is deeply nested and folders have long names
- It is recommended to unzip in **C:\Temp** or similar short paths not to exceed the maximum path length



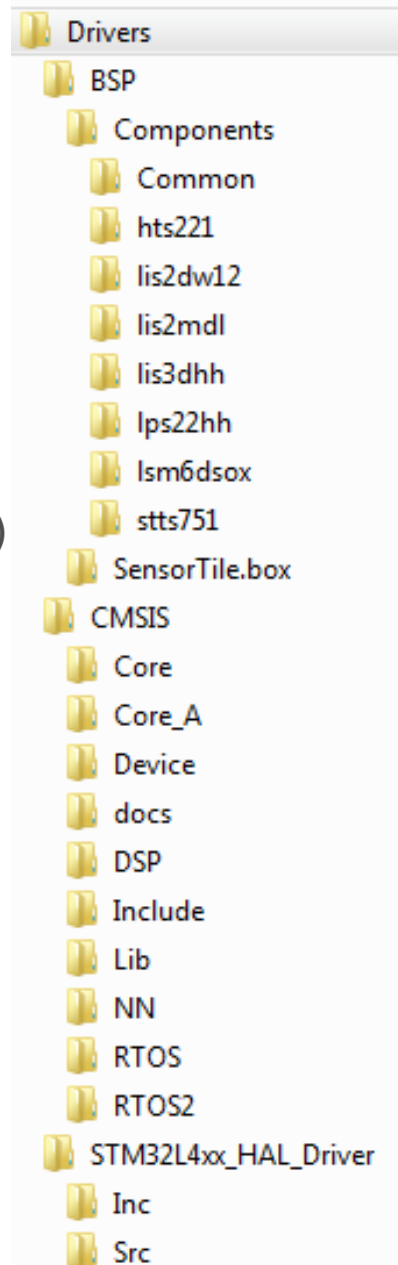


Latest info at www.st.com



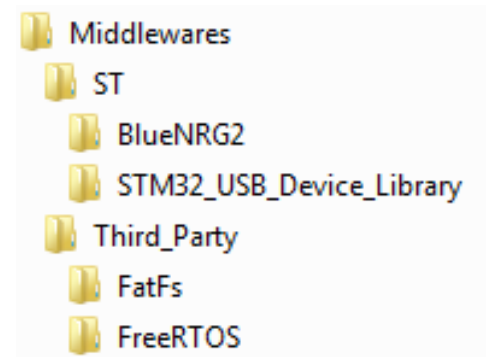
- Drivers

- BSP Board Support Package
 - Components (MEMS)
 - Board (SensorTile.Box)
- CMSIS Cortex MCU Software Interface Std
 - DSP Digital Sig Proc.
 - NN Neural Net
 - RTOS
- HAL Hardware Abstraction Layer

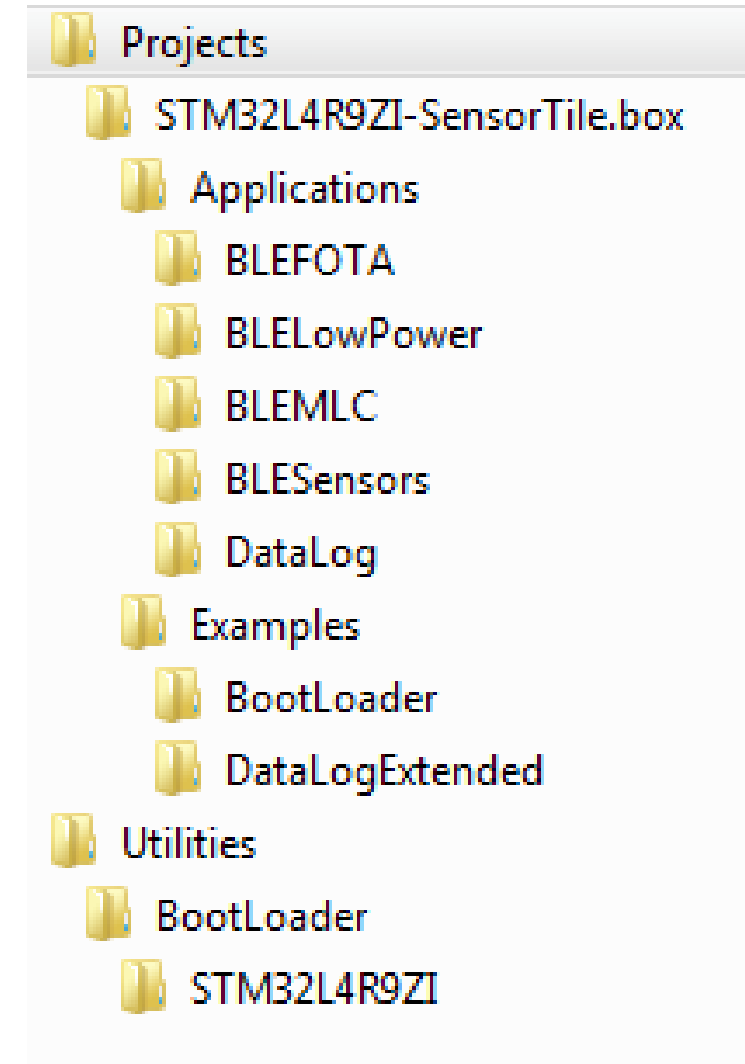


- Middleware

- BLE Bluetooth Low Energy stack
- USB Device library
- FatFS Filing System
- FreeRTOS Real-Time Operating System



- Applications
 - **DataLog** save to SD card at max speed w/RTOS
 - **BLE Sensors** stream to ST BLE Sensor app
 - **BLE LowPower** stream to ST BLE Sensor app w/RTOS
 - **BLEMLC** Machine Learning Core demo
 - **BLEFOTA** Firmware Over-The-Air-Update
- Examples
 - **Bootloader** at 0x0800 000 with BLEFOTA at 0x0800 4000
 - **DataLogExtended** stream to **Unicleo GUI** via USB Virtual COM port
- Utilities
 - **Bootloader** precompiled binary

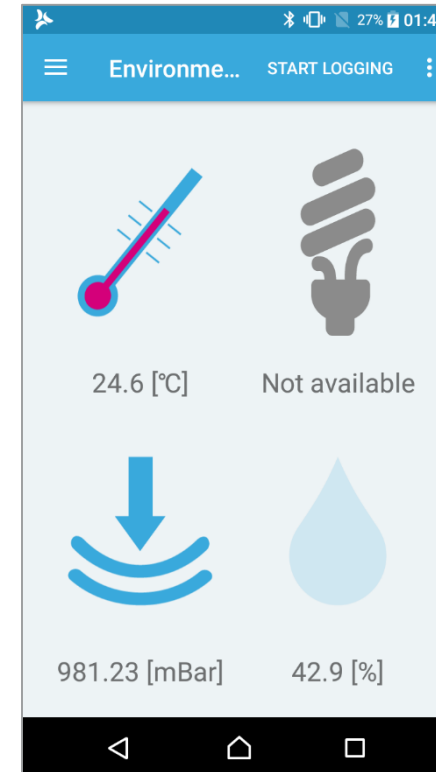


- **BLE Sensors**

- The BLE Sensors sample shows how to easily send inertial and environmental sensor output data of the SensorTile.box to STBLESensor application via Bluetooth.

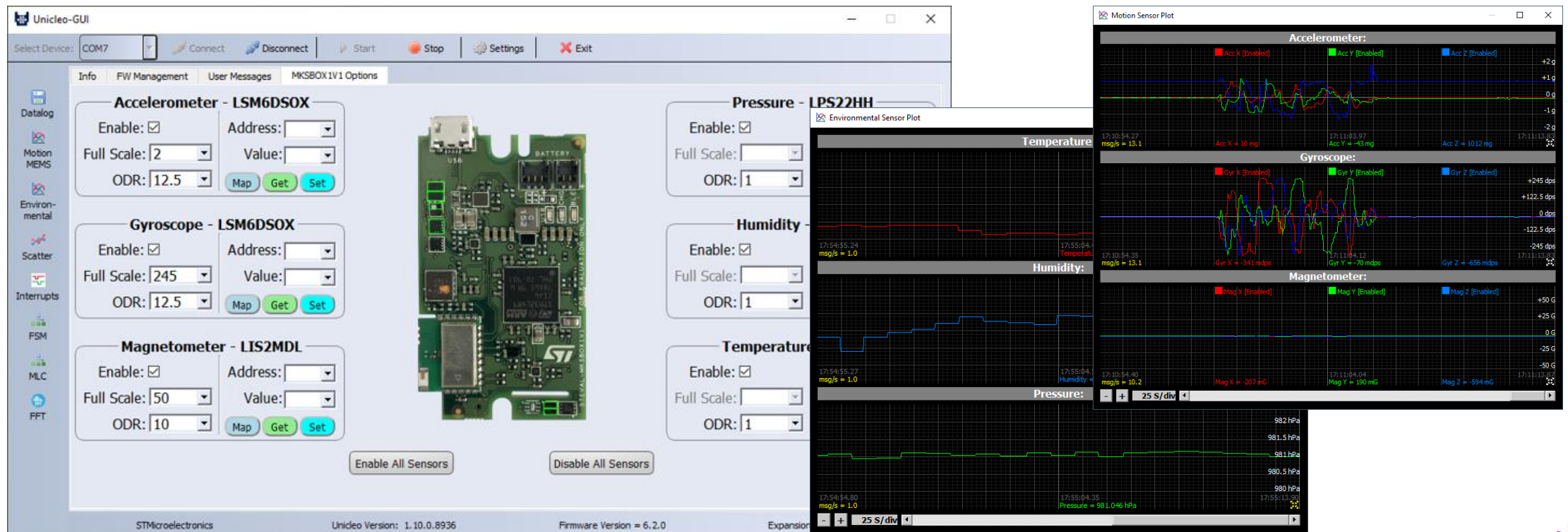
- **BLEMLC**

- The BLEMLC application shows how to program the machine learning core of the LSM6DSOX. In particular, this sample programs the accelerometer to run an activity recognition algorithm or a vibration monitoring algorithm and to send the results to application.



- **DataLogExtended**

- The DataLogExtended sample shows how to dialog with the Unicleo-GUI running on a PC connected via USB to the SensorTile.box, visualizing all the sensor data streamed from the board. It is also possible to program and test MLC and FSM algorithms.

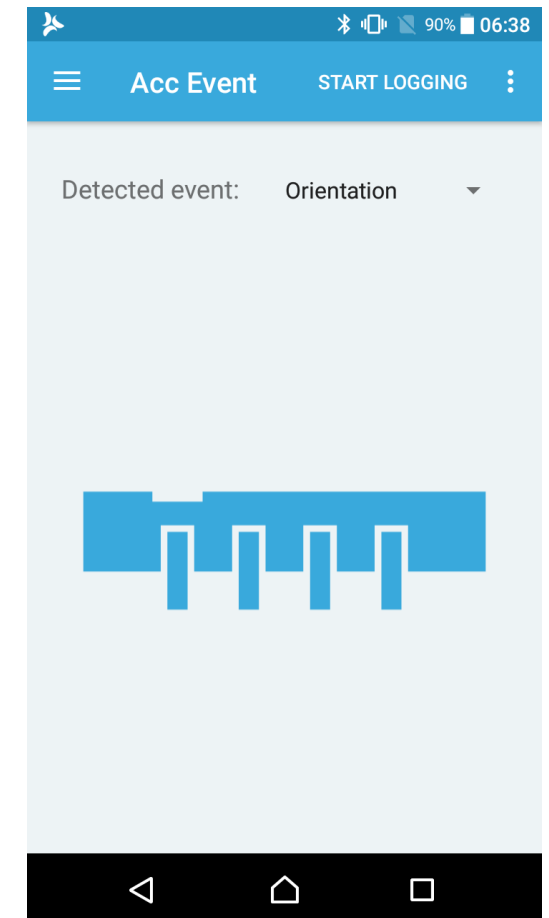


- **BLELowPower**

- The BLELowPower application shows how to use FreeRTOS and **low-power** techniques to send inertial and environmental sensor output, the dB measured by the analog microphone and events of the LSM6DSOX accelerometer (like FreeFall, Single/Double Tap, WakeUp, Tilt and 6D orientation) to STBLESensor application via **Bluetooth**.

- **DataLog**

- The DataLog application shows how to use FreeRTOS to save inertial and environmental sensors and the analog microphone output data of the SensorTile.box to the **SD card** without losing samples (at **max speed**)



Real-Time Operating System (RTOS)

- **BLELowPower** and **DataLog**: RTOS
 - Enables **low-power** operation: microcontroller can sleep when there is no task scheduled for execution (typical task samples sensors and streams over BLE)
 - Enables **max speed**: microcontroller can buffer data read from sensors while write operation on SD card is in progress



- **Bootloader**

- The BootLoader controls the Boot Sequence and allows replacing BLEFOTA firmware with the new BLEFOTA firmware updated Over-The-Air (FOTA) through an Android/iOS device via Bluetooth using the STBLESensor application

- **BLEFOTA**

- The BLEFOTA receives the new firmware from the STBLESensor application and saves it before passing control to BootLoader samples
- It is possible to control the initialization and connection phases and, on boot, it is possible to define the name the board will use for Bluetooth advertising.

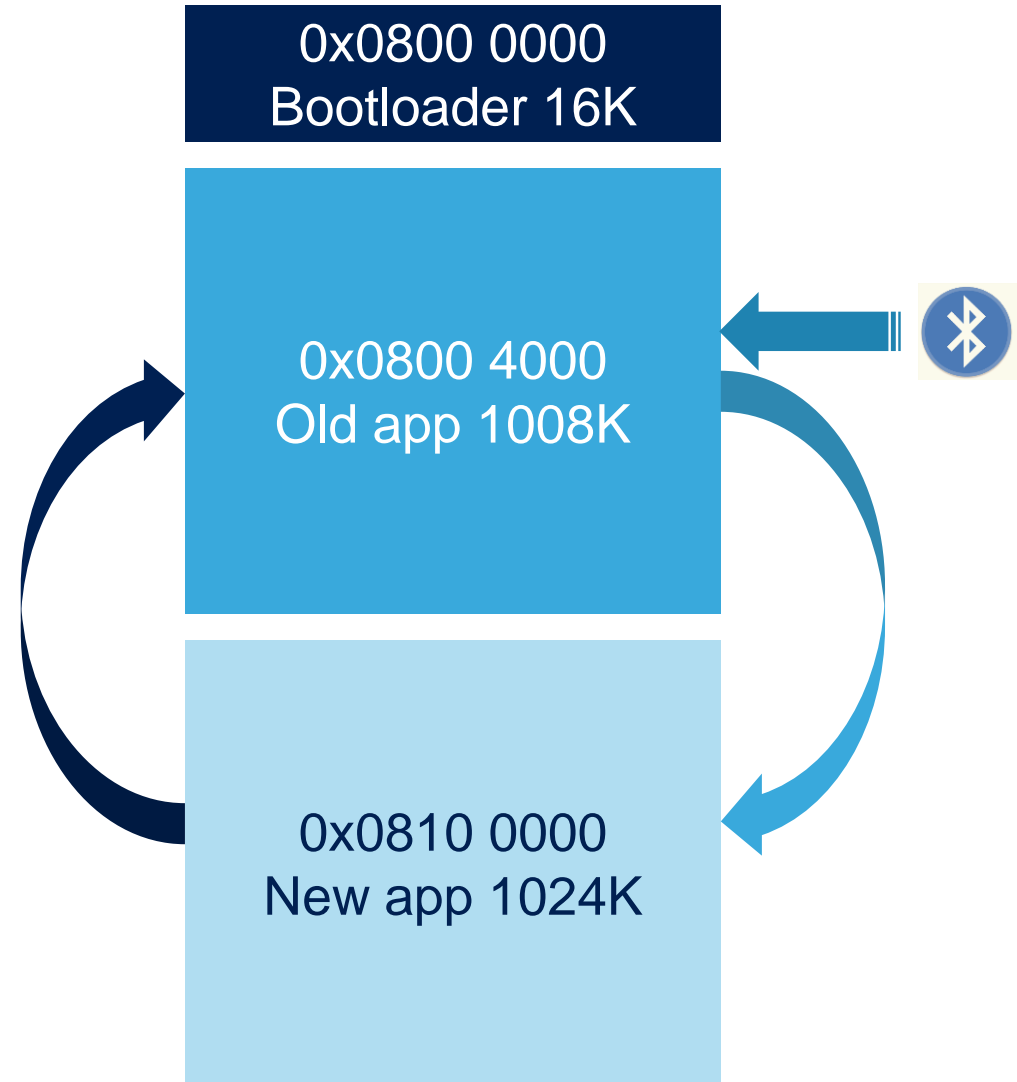
The BootLoader example and the BLEFOTA application work together and they must be loaded together on the STEVAL-MKSBOX1V1 to work properly.



Firmware Over-The-Air (FOTA)

BLEFOTA application and the Bootloader

- 0x0800 0000 Bootloader
 - If the new app is ok, copy it over current app
 - Run current app at 0x0800 0000
- 0x0800 4000 current application (old)
 - Receives the FOTA app over BLE and writes at 0x0810 0000
- 0x0810 0000 FOTA application (new)



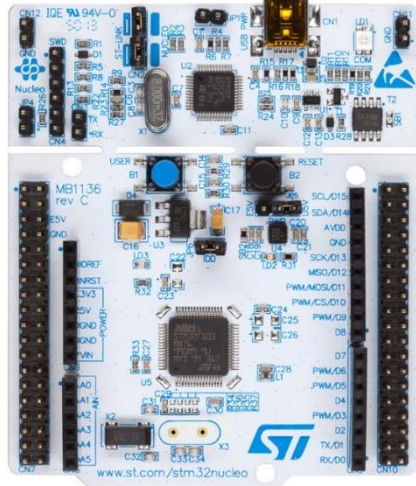
• Unicleo GUI

- Displays data from connected sensors (time plot, scatter plot, 3D plot)
- Saves data to tab separated (TSV) or comma separated (CSV) files
- Program Nucleo with selected MotionXX library
- Reads from and writes registers of sensors on X-Nucleo expansion
- Connects over **USB** or **BLE**
- Supports many platforms
 - Nucleo-L476/F401/L152/L073 with X-Nucleo-IKS01A2/A3 or IKS02A1
 - BlueNRG-Tile
 - **SensorTile.box**



Introduction to Unicleo-GUI

HW Requirements



STM32 Nucleo Board
 X-CUBE-MEMS1
 NUCLEO-F401RE
 NUCLEO-L476RG
 NUCLEO-L152RE
 NUCLEO-L073RZ

+



Expansion Board
 X-NUCLEO-IKS01A2
 X-NUCLEO-IKS01A3

+

OPTIONAL



DIL24 Adapter
 ASM330LHH
 AIS2DW12*
 IIS2DLPC
 IIS2MDC
 ISM303DAC
 ISM330DLC
 LIS2DH12
 LIS2DW12
 LIS2MDL
 LIS3MDL*
 LPS22HH
 LPS33HW
 LSM6DSO
 LSM6DSOX
 LSM6DSR*
 STTS22H*

NUCLEO



STEVAL-STLKT01V1



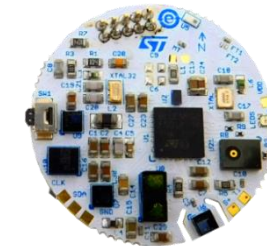
STEVAL-MKSBOX1V1



STEVAL-BCNKT01V1



STEVAL-WESU1



STEVAL-BCN002V1B

BLE DEVICES



Introduction to Unicleo-GUI

Datalog Mode

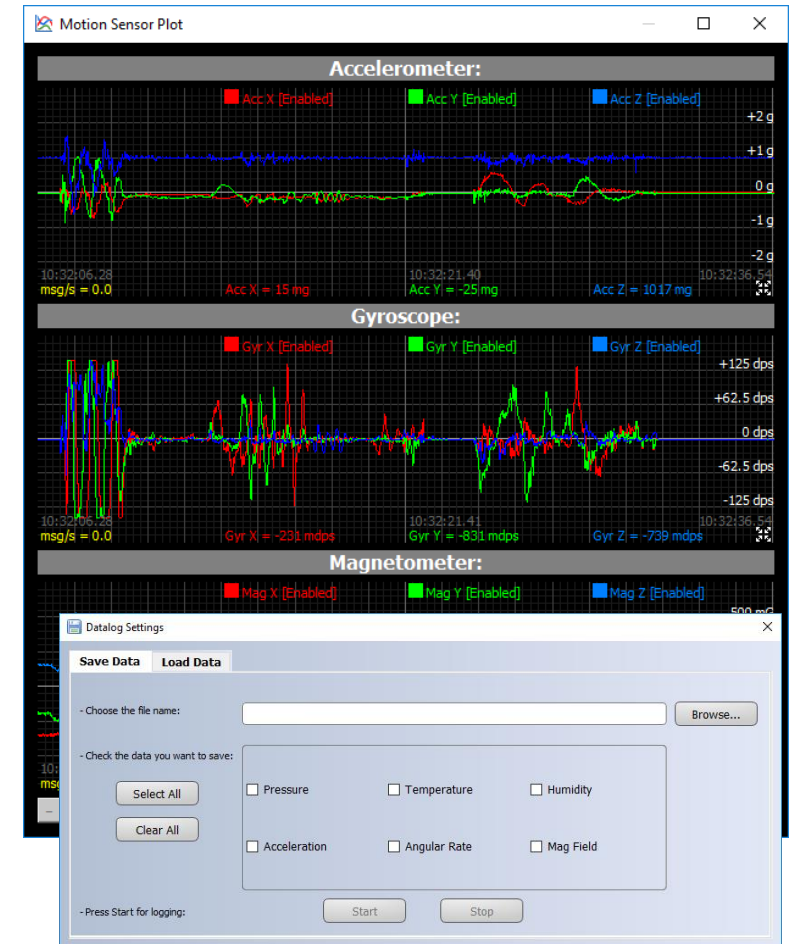
- Configure sensor
- Monitor, display and store sensor data

The Unicleo-GUI interface displays configuration options for several sensors:

- Accelerometer - LSM6DSO:** Enabled, Address: [dropdown], Full Scale: 2, Value: [dropdown], ODR: 104. Buttons: Map, Get, Set.
- Gyroscope - LSM6DSO:** Enabled, Address: [dropdown], Full Scale: 125, Value: [dropdown], ODR: 104. Buttons: Map, Get, Set.
- Magnetometer - LIS2MDL:** Enabled, Address: [dropdown], Full Scale: 50, Value: [dropdown], ODR: 50. Buttons: Map, Get, Set.
- Pressure - LPS22HH:** Enabled, Address: [dropdown], Full Scale: [dropdown], Value: [dropdown], ODR: 1. Buttons: Map, Get, Set.
- Humidity - HTS221:** Enabled, Address: [dropdown], Full Scale: [dropdown], Value: [dropdown], ODR: 1. Buttons: Map, Get, Set.
- Temperature - HTS221:** Enabled, Address: [dropdown], Full Scale: [dropdown], Value: [dropdown], ODR: 1. Buttons: Map, Get, Set.

Buttons: Enable All Sensors, Disable All Sensors.

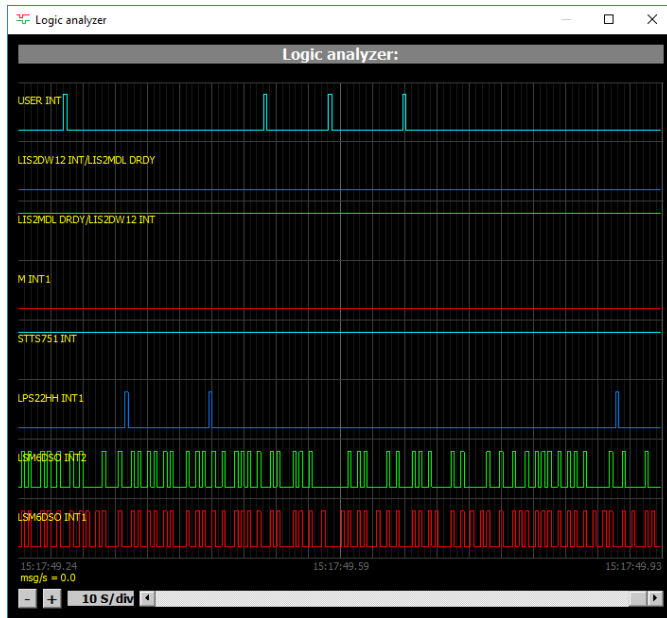
Time	Acc X	Acc Y	Acc Z	Gyr X	Gyr Y	Gyr Z	Mag X	Mag Y	Mag Z	Pressure	Humidity	Temperature
10:32:35.03	16	-25	1018	-253	-826	-787	-409	-157	129			
10:32:35.03	14	-27	1017	-227	-822	-783	-409	-157	129			
10:32:35.04	15	-26	1018	-218	-831	-783	-405	-166	130			
10:32:35.05	16	-27	1019	-214	-848	-791	-409	-162	123			
10:32:35.06	15	-26	1018	-208	-818	-800	-409	-162	123			
10:32:35.07	15	-27	1016	-253	-813	-796	-411	-160	124			
10:32:35.08	15	-27	1016	-253	-813	-796	-411	-160	124			
10:32:35.08	15	-26	1016	-258	-813	-765	-411	-160	124			



Introduction to Unicleo-GUI

Datalog Mode

- Register Map
- Interrupt monitor
- MLC, FSM



Register Map

LSM6DSO LIS2MDL HTS221 LPS22HH

Basic Registers Embedded registers Embedded registers - pages Sensor hub registers

0x01	FUNC_CFG_ACCESS	Get	Set		
0x02	PIN_CTRL	Get	Set		
0x07	FIFO_CTRL1	Get	Set		
0x08	FIFO_CTRL2	Get	Set		
0x09	FIFO_CTRL3	Get	Set		
0x0a	FIFO_CTRL4	Get	Set		
0x0b	COUNTER_BDR_REG1	Get	Set		
0x0c	COUNTER_BDR_REG2	Get	Set		
0x0d	INT1_CTRL	Get	Set		
0x0e	INT2_CTRL	Get	Set		
0x0f	WHO_AM_I	Get			
0x10	CTRL1_XL	Get	Set		
0x11	CTRL2_G	Get	Set		
0x12	CTRL3_C	Get	Set		
0x13	CTRL4_C	Get	Set		
0x14	CTRL5_C	Get	Set		
0x15	CTRL6_C	Get	Set		
0x16	CTRL7_G	Get	Set		
0x17	CTRL8_XL	Get	Set		
0x18	CTRL9_XL	Get	Set		
0x19	CTRL10_C	Get	Set		
0x1a	ALL_INT_SRC	Get			
0x1b	WAKE_UP_SRC	Get			
0x1c	TAP_SRC	Get			
0x1d	D6D_SRC	Get			
0x1e	STATUS_REG/STATUS_SPIAux	Get			
0x20	OUT_TEMP_L	Get			
0x21	OUT_TEMP_H	Get			

MLC

Sensor configuration

Example algorithms

Vibration Monitoring Motion Intensity

Double-click to upload example

Note: Loading example can modify the previous sensor configuration. Change of sensor configuration in "Options" tab after an example algorithm is loaded can influence the example functionality.

All example algorithms were created by Unico software. The Unico can be used in offline mode to develop and test FSM configuration.

MLC Status Registers

0x0d MLC_INT1

0x11 MLC_INT2

0x15 MLC_STATUS

MLC Source Registers

0x70 MLC0_SRC

0x71 MLC1_SRC

0x72 MLC2_SRC

0x73 MLC3_SRC

0x74 MLC4_SRC

0x75 MLC5_SRC

0x76 MLC6_SRC

0x77 MLC7_SRC

FSM Status Registers

Latched Interrupt

Read All Status Registers

0x12 EMB_FUNC_STATUS

0x13 FSM_STATUS_A

0x14 FSM_STATUS_B

0x48 FSM_LONG_COUNTER_L

0x49 FSM_LONG_COUNTER_H



Introduction to Unicleo-GUI

Algorithms Support

- Dedicated window for each algorithm

The image displays five overlapping software windows from the Unicleo-GUI interface:

- Calibrated Magnetometer 3D Plot:** Shows a 3D plot of magnetic field data with a red sphere and a blue ring. Text on the left includes:
 - HI coefficient X: 105 mGauss
 - HI coefficient Y: -84 mGauss
 - HI coefficient Z: -439 mGauss
 - Goodness: **GOOD**
 - SI matrix[0][0]: 1.405
 - SI matrix[0][1]: 0
 - SI matrix[0][2]: 0
 - SI matrix[1][0]: 0
 - SI matrix[1][1]: 1.541
 - SI matrix[1][2]: 0
 - SI matrix[2][0]: 0
 - SI matrix[2][1]: 0
 - SI matrix[2][2]: 1.505
 A 'Clear Points' button is at the bottom.
- E-Compass:** Shows a heading of 171.5 and a goodness of 0.
- Active Time Detection:** A table showing activity logs:

Time	Active Time[s]	State
11:04:36.87	0	Inactive
11:04:42.85	0	Active
11:04:43.84	1	Active
11:04:44.84	2	Active
11:04:45.84	3	Active
11:04:46.84	4	Active
11:04:47.84	5	Active
11:04:48.84	6	Active
11:04:49.83	6	Inactive
11:04:50.83	7	Active
11:04:51.83	8	Active
11:04:52.81	9	Active
11:04:53.81	10	Active

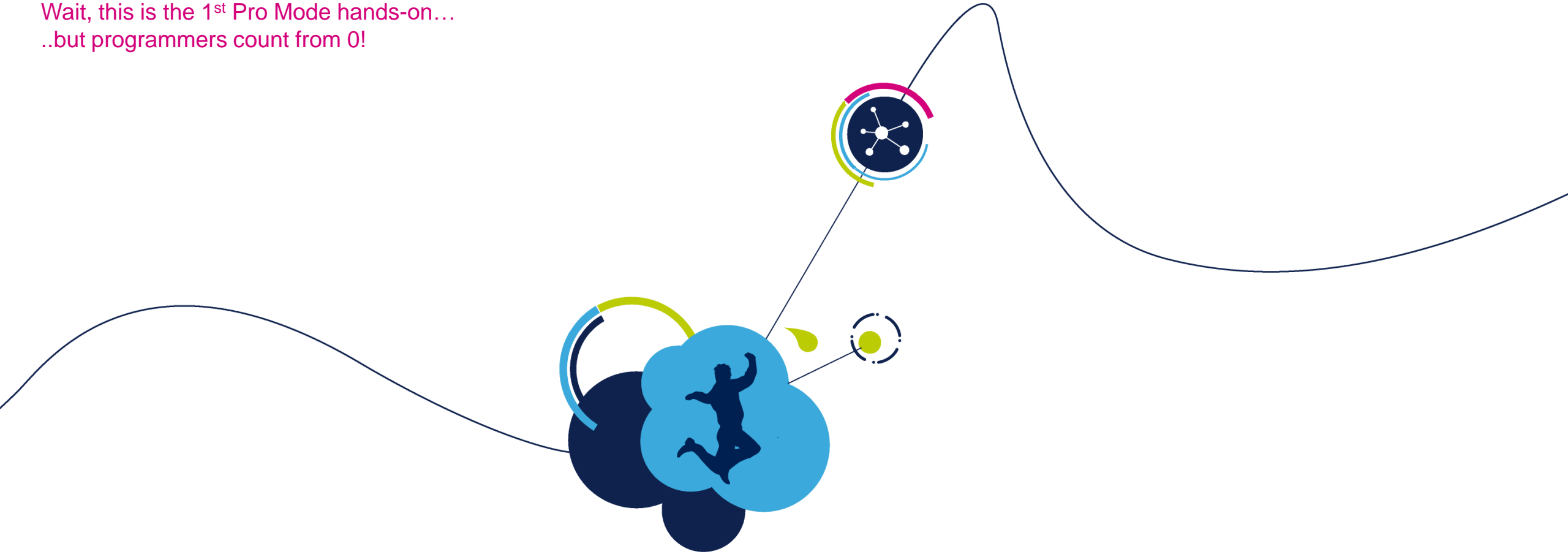
 The 'Current Status' is 'Active', shown with a walking icon. Buttons for 'Download Off-line Data' and 'Save Off-line Data To File' are present.
- Sensor Fusion:** Displays a 3D model of a blue teapot.
- Fitness Activities:** A table of activity counts:

Time	Activity	Counter
09:39:44.09	Biceps Curl	00
09:39:50.07	Biceps Curl	01
09:39:54.07	Biceps Curl	02
09:39:55.96	Squat	00
09:40:01.92	Squat	03
09:40:04.13	Push-up	00
09:40:22.12	Push-up	01

 The 'Current Activity' is 'Push-up' with a count of 01, shown with a push-up icon. Buttons for 'Download Off-line Data' and 'Save Off-line Data To File' are present.



Wait, this is the 1st Pro Mode hands-on...
..but programmers count from 0!



Hands-on #0

Flash a binary

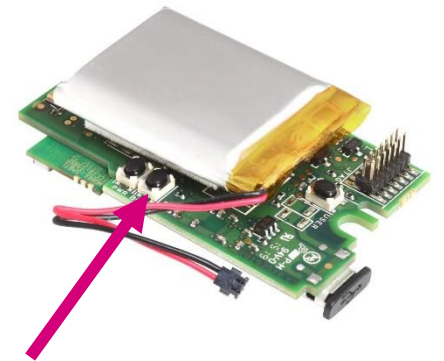


Flash a binary

Introduction

22

- The binary must be loaded using DFU (Direct Firmware Upgrade over USB)
- These are the steps:
 1. Run ST BLE Sensor app to enter DFU mode
 - a) Connect to your device, open the debug console and issue the “DFU” command
 - b) Hold BOOT button during power up (e.g. when connecting to a laptop with a micro USB cable while the battery is disconnected)
 2. Run STM32CubeProg to program the device
 - Select USB and push “Connect”, Mass Erase then Program new binary



BOOT button



Flash a binary

Enter DFU mode

1. Press the button
Connect to a device

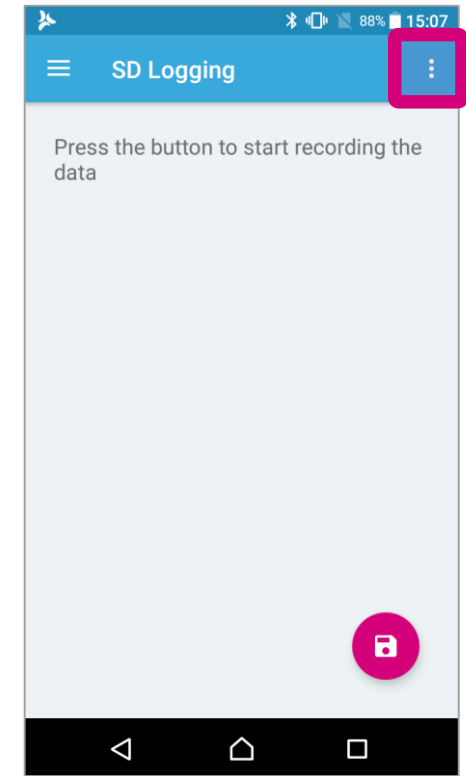
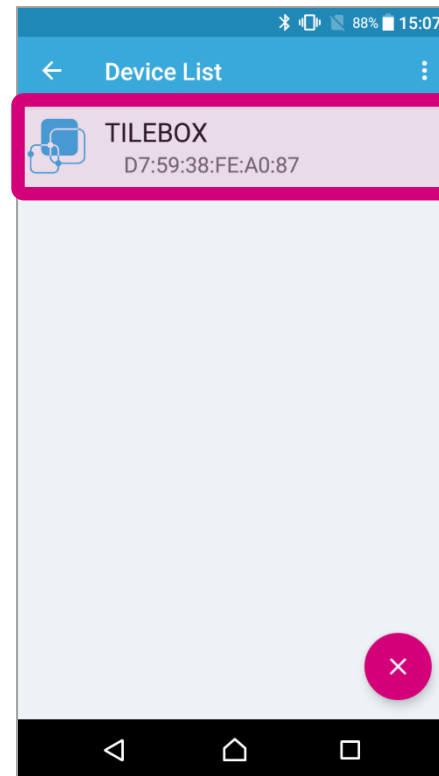
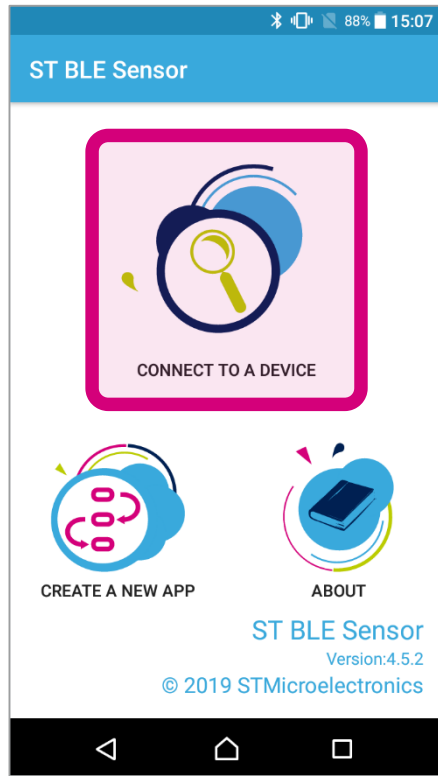
2. Select your
SensorTile.Box

3. Touch the button in
the top right corner

Run the **ST BLE Sensor** app

or

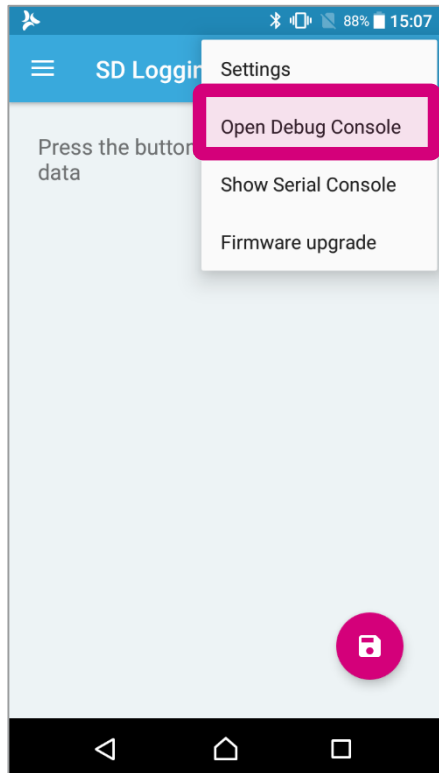
Go back to the main screen of the **ST BLE Sensor** app



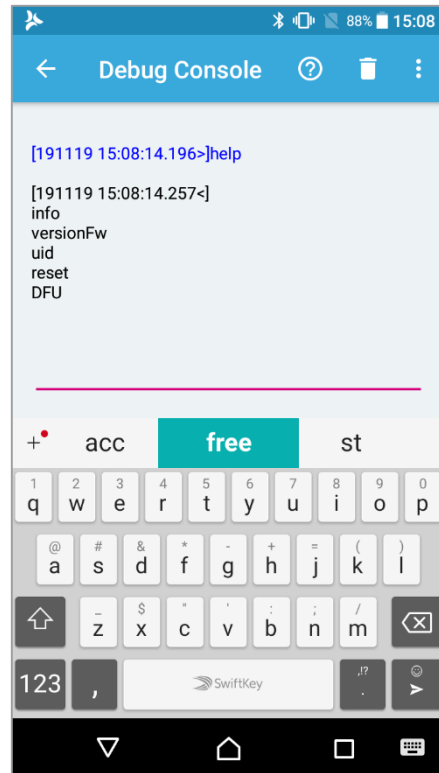
Flash a binary

Enter DFU mode

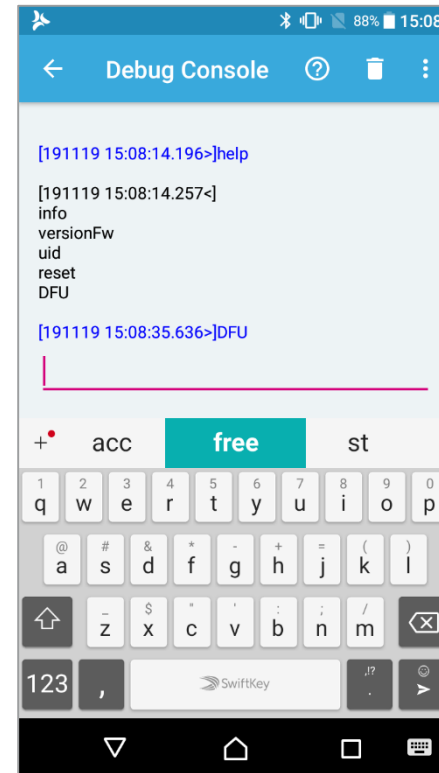
1. Select **Open Debug Console**



2. Type **“help”** to see list of commands



3. Type **“DFU”** (all uppercase)



Your device is now in **Direct Firmware Upgrade** mode



1. Run
STM32CubeProgrammer



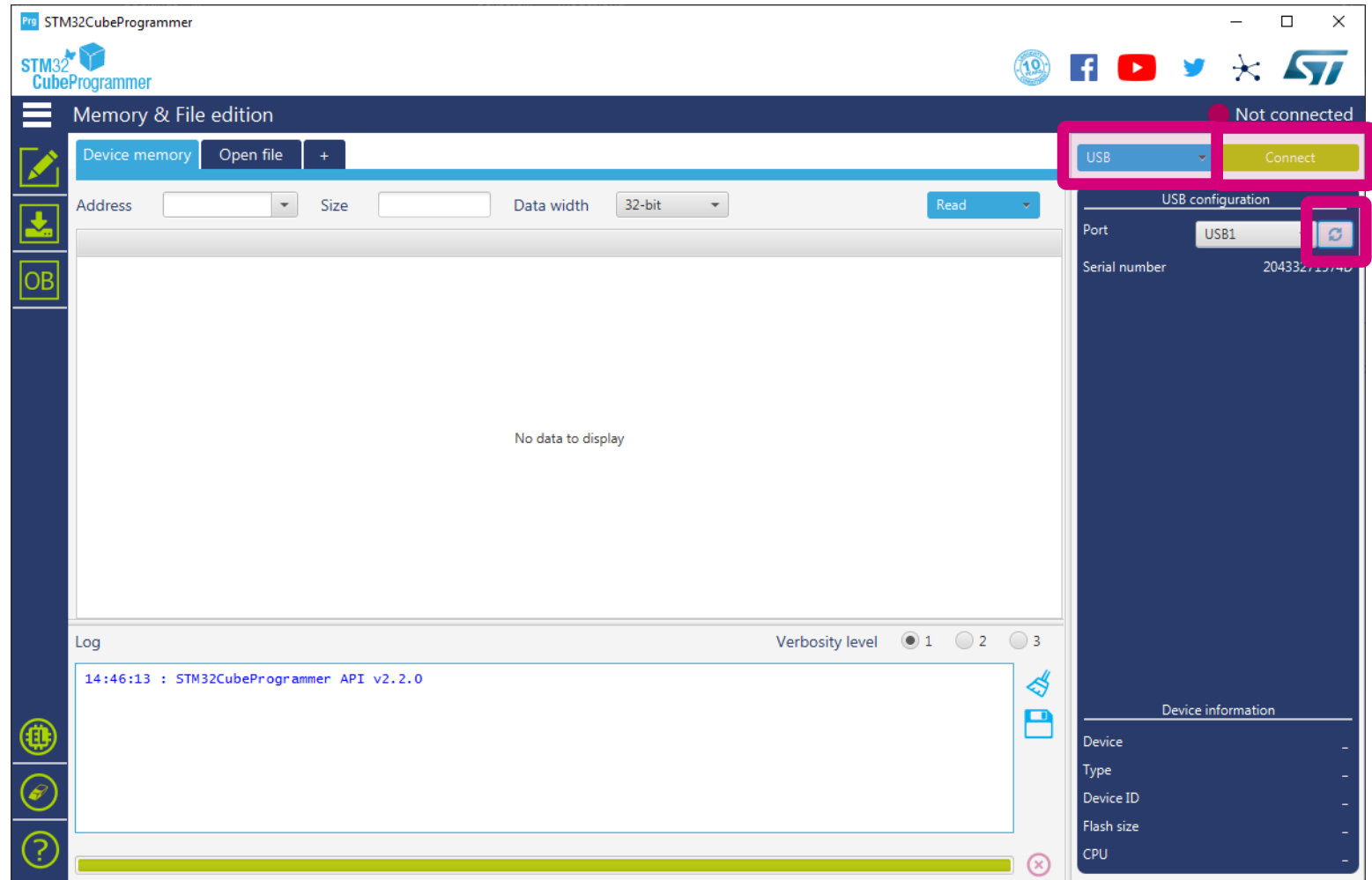
2. Wait for STM32CubeProgrammer
to open



Flash a binary

26

1. Select **USB** from drop-down menu
2. Click **Refresh**
3. Click **Connect**



Flash a binary

27

1. Check that Data is **read successfully**
2. Select **Erasing & Programming** from the menu on the left



STM32CubeProgrammer

Memory & File edition

Device memory | Open file | +

Address: 0x08000000 | Size: 0x400 | Data width: 32-bit | Read

Address	0	4	8	C	ASCII
0x08000000	20000428	0800190D	080018E9	080018EB	(. é . . . ë . . .
0x08000010	080018ED	080018EF	080018F1	00000000	í . . . ï . . . ñ
0x08000020	00000000	00000000	00000000	080018F3 ó
0x08000030	080018F5	00000000	080018F7	080018F9	õ + ù . . .
0x08000040	08001929	0800192D	08001931	08001935) . . . - . . . 1 . . . 5 . . .
0x08000050	08001939	0800193D	08001941	08001945	9 . . . = . . . A . . . E . . .
0x08000060	08001949	0800194D	08001951	08001955	I . . . M . . . Q . . . U . . .
0x08000070	08001959	0800195D	08001961	08001965	Y . . .] . . . a . . . e . . .
0x08000080	08001969	0800196D	08001971	08001975	i . . . m . . . q . . . u . . .
0x08000090	08001979	0800197D	08001981	08001985	y . . . }
0x080000A0	08001989	0800198D	08001991	08001995
0x080000B0	08001999	0800199D	080019A1	080019A5 j . . . ¥
0x080000C0	080019A9	080019AD	080019B1	080019B5	© ± . . . μ

Log

15:25:02 : Bank : 0x00
15:25:02 : Address : 0x1ff00000
15:25:02 : Size : 36 Bytes
15:25:02 : Bank : 0x01
15:25:02 : Address : 0x1ff01008
15:25:02 : Size : 28 Bytes
15:25:02 : UPLOADING . . .
15:25:02 : Size : 1024 Bytes
15:25:02 : Address : 0x8000000
15:25:02 : Read progress:
15:25:02 : Data read successfully
15:25:02 : Read operation is: 00:00:00.006

Verboosity level: 1 2 3

USB configuration: USB1, Disconnect, Serial number: 20433271574D

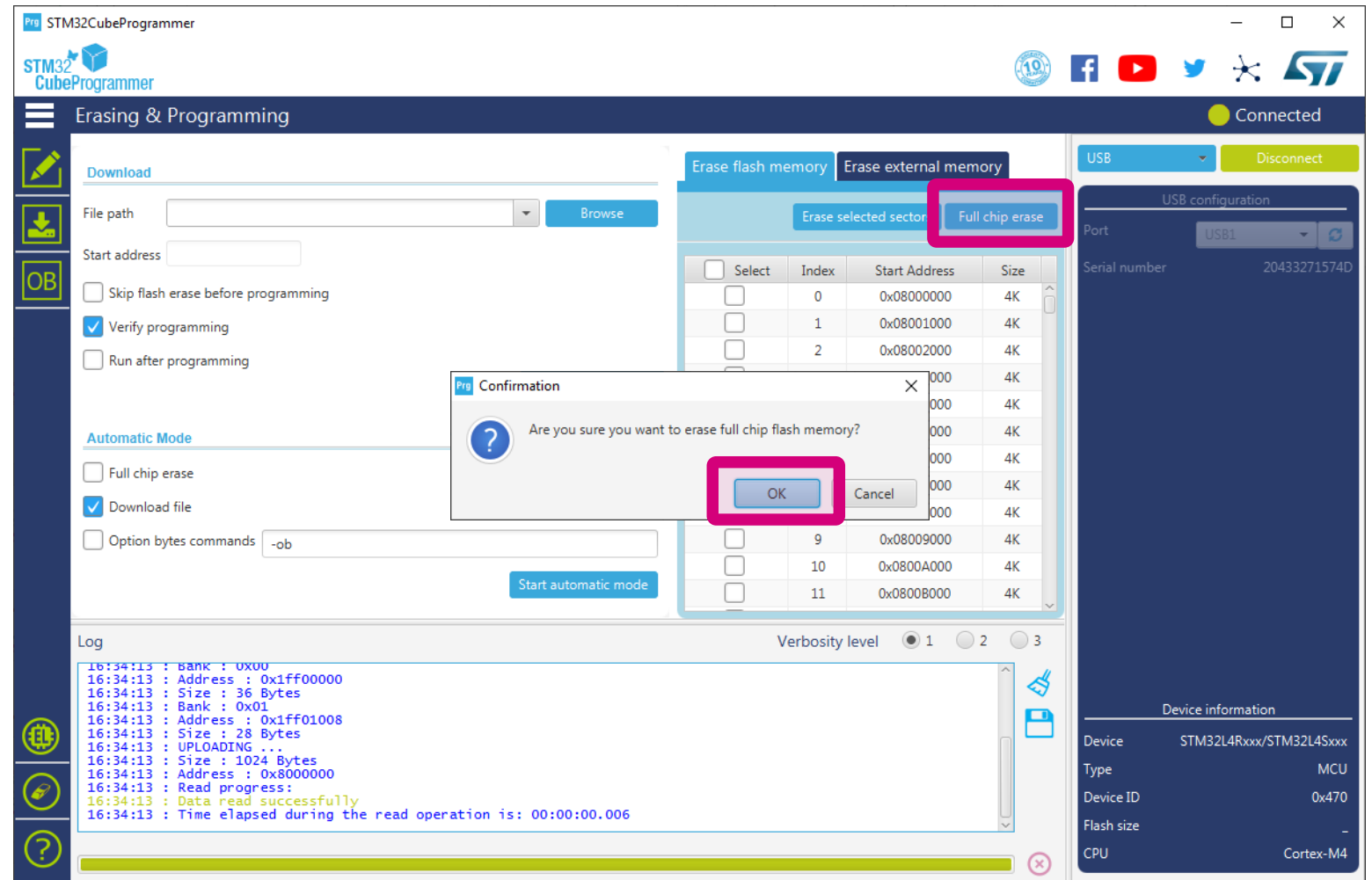
Device information: Device: STM32L4Rxxx/STM32L4Sxxx, Type: MCU, Device ID: 0x470, Flash size: -, CPU: Cortex-M4



Flash a binary

28

1. Select **Full chip erase**
2. Click **OK**



Flash a binary

29

1. Wait until Full chip erase is completed
2. Click **OK**

The screenshot shows the STM32CubeProgrammer interface during an erasing operation. A message dialog box is displayed in the center, with the text: "Mass erase command correctly executed. Note: if there's any flash protection, it will not be erased." The "OK" button in the dialog is highlighted with a pink rectangle. The background interface shows the "Erasing & Programming" tab, with the "Erase flash memory" sub-tab active. A table lists sectors to be erased:

Select	Index	Start Address	Size
<input type="checkbox"/>	0	0x08000000	4K
<input type="checkbox"/>	1	0x08001000	4K
<input type="checkbox"/>	2	0x08002000	4K
<input type="checkbox"/>	3	0x08003000	4K
<input type="checkbox"/>	4	0x08004000	4K
<input type="checkbox"/>	5	0x08005000	4K
<input type="checkbox"/>	6	0x08006000	4K
<input type="checkbox"/>	7	0x08007000	4K
<input type="checkbox"/>	8	0x08008000	4K
<input type="checkbox"/>	9	0x08009000	4K
<input type="checkbox"/>	10	0x0800A000	4K
<input type="checkbox"/>	11	0x0800B000	4K

The log window at the bottom shows the following messages:

```
16:34:13 : read progress:
16:34:13 : Data read successfully
16:34:13 : Time elapsed during the read operation is: 00:00:00.006
16:34:55 : MASS ERASE ...
16:34:55 : Mass erase command correctly executed. Note: if there's any flash protection, it will not be erased.
16:34:55 : UPLOADING ...
16:34:55 : Size : 1024 Bytes
16:34:55 : Address : 0x8000000
16:34:55 : Read progress:
16:34:55 : Data read successfully
16:34:55 : Time elapsed during the read operation is: 00:00:00.004
```



1. Click **Browse** and navigate to **DataLogExtended.bin** from the Function Pack
...\\STM32CubeFunctionPack_STBOX1_V1.0.0\\Projects\\STM32L4...\\Examples\\DataLogExtended\\Binary\\DataLogExtended.bin
2. Check that **Start address** is set to 0x08000000
3. Check Verify programming (optional)
4. Click **Start Programming**

The screenshot displays the STM32CubeProgrammer interface. The 'Erasing & Programming' tab is active. In the 'Download' section, the 'File path' is set to 'C:\STM32CubeFunctionPack_STBOX1_V1.0.0\Projects\STM32L4...\\Examples\\DataLogExtended\\Binary\\DataLogExtended.bin', the 'Start address' is '0x08000000', and the 'Verify programming' checkbox is checked. The 'Start Programming' button is highlighted. The 'Automatic Mode' section has 'Download file' checked. The 'Log' window shows a successful data read operation.

Select	Index	Start Address	Size
<input type="checkbox"/>	0	0x08000000	4K
<input type="checkbox"/>	1	0x08001000	4K
<input type="checkbox"/>	2	0x08002000	4K
<input type="checkbox"/>	3	0x08003000	4K
<input type="checkbox"/>	4	0x08004000	4K
<input type="checkbox"/>	5	0x08005000	4K
<input type="checkbox"/>	6	0x08006000	4K
<input type="checkbox"/>	7	0x08007000	4K
<input type="checkbox"/>	8	0x08008000	4K
<input type="checkbox"/>	9	0x08009000	4K
<input type="checkbox"/>	10	0x0800A000	4K
<input type="checkbox"/>	11	0x0800B000	4K

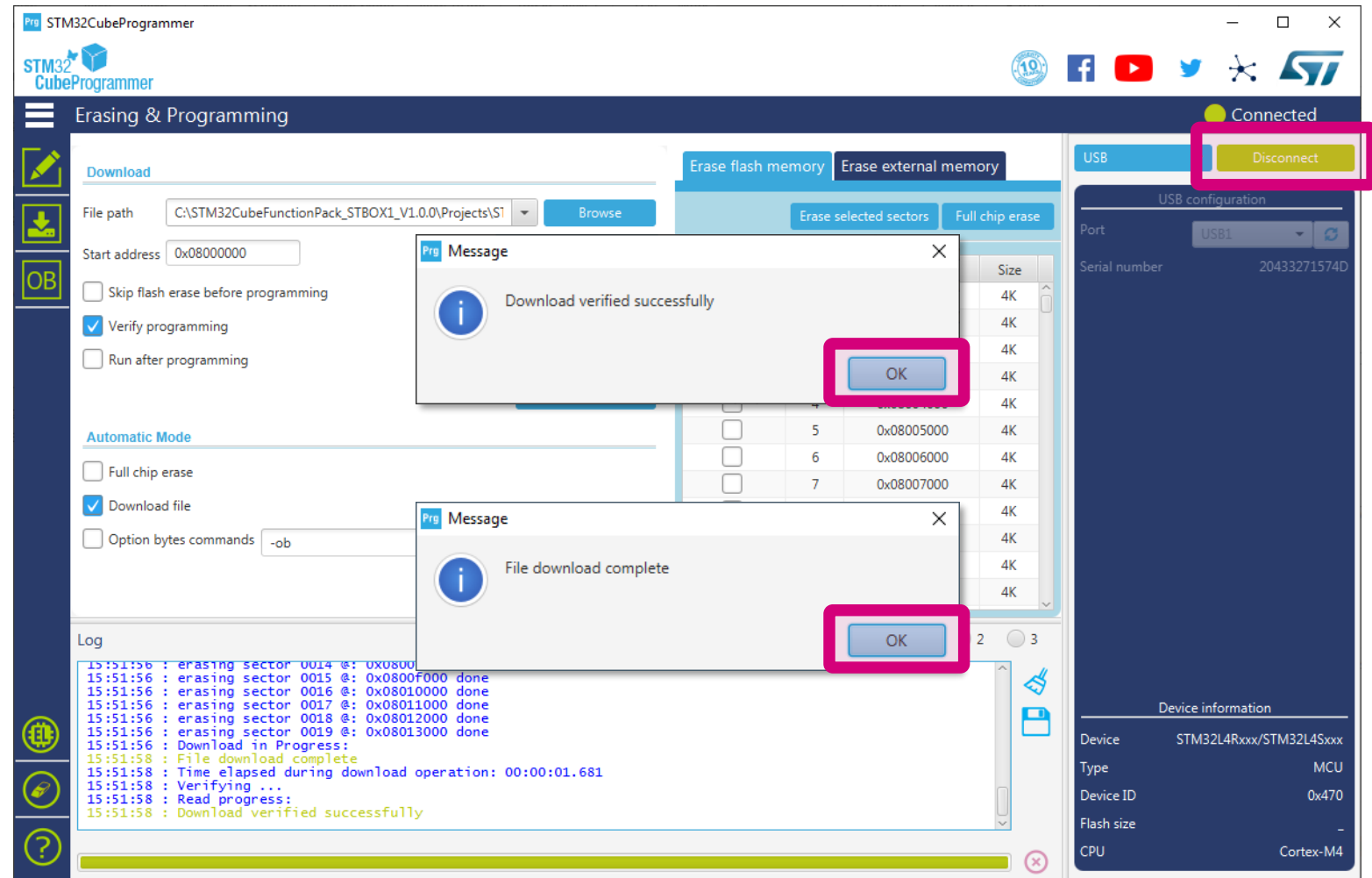
```
Log
15:25:02 : Bank : 0x00
15:25:02 : Address : 0x1ff00000
15:25:02 : Size : 36 Bytes
15:25:02 : Bank : 0x01
15:25:02 : Address : 0x1ff01008
15:25:02 : Size : 28 Bytes
15:25:02 : UPLOADING ...
15:25:02 : Size : 1024 Bytes
15:25:02 : Address : 0x8000000
15:25:02 : Read progress:
15:25:02 : Data read successfully
15:25:02 : Time elapsed during the read operation is: 00:00:00.006
```

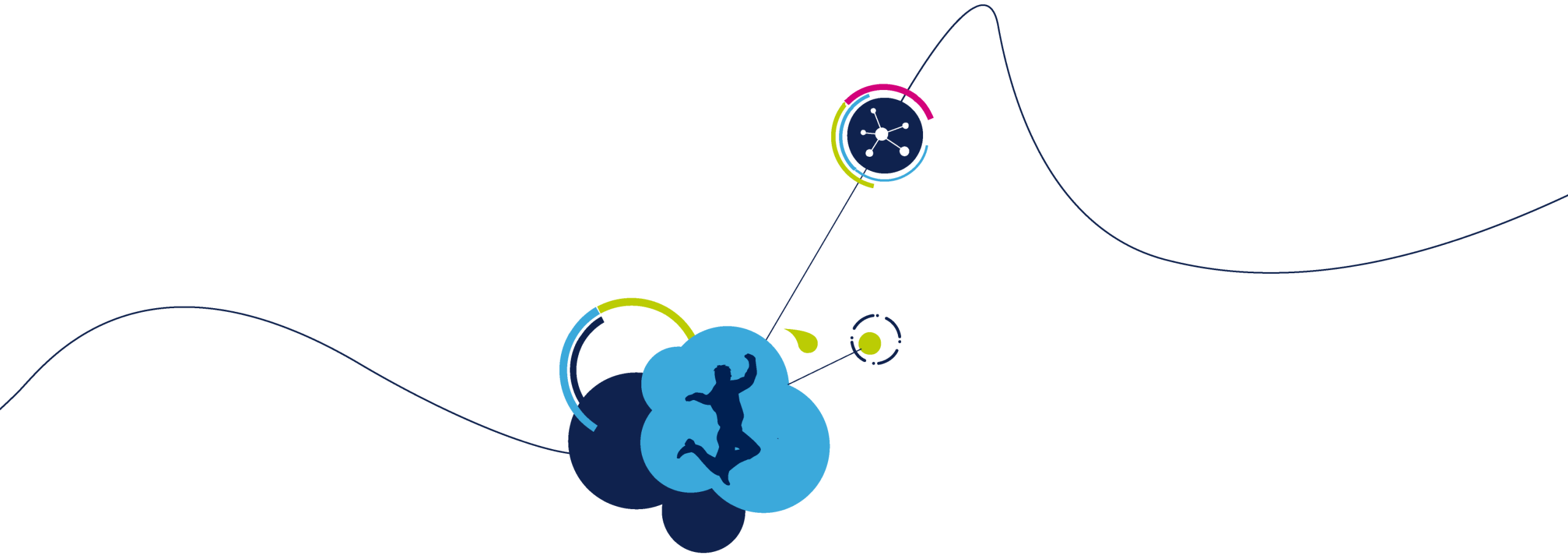


Flash a binary

31

1. Wait until **Download verified successfully** appears, then click **OK**
2. Click **OK** on the **File download complete**
3. Click **Disconnect**
4. Unplug and plug the device to reset





Hands-on #1

DataLogExtended example with Unicleo-GUI

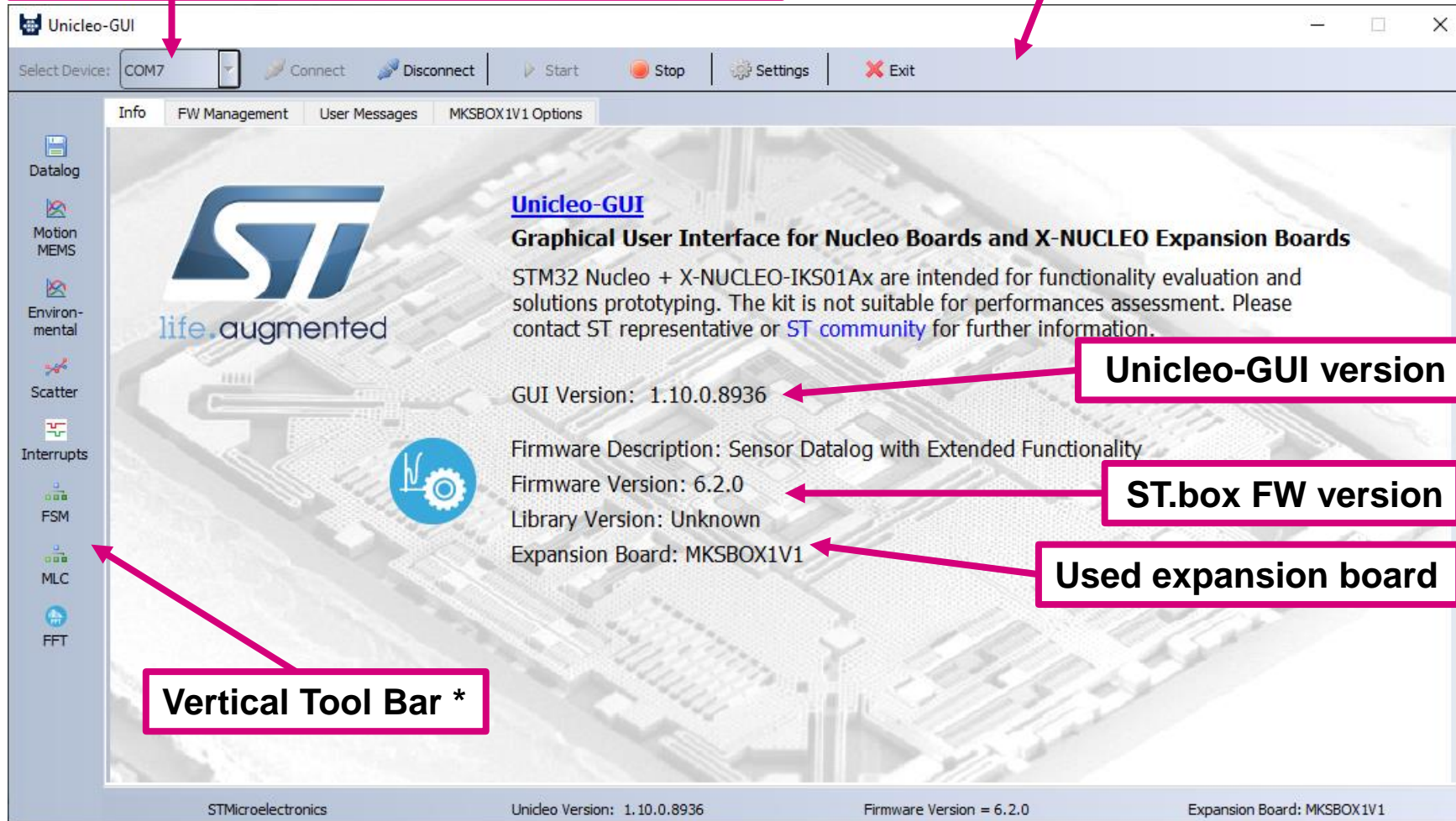


DataLogExtended with Unicleo-GUI

Main Window

COM Port selection
(only COM ports with manufacturer set to STM and Nucleo boards with appropriate firmware)

Control Tool Bar



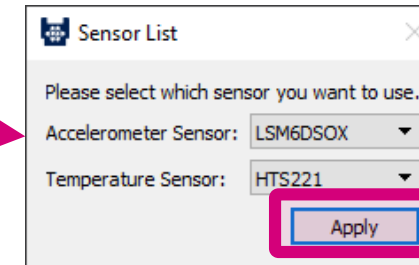
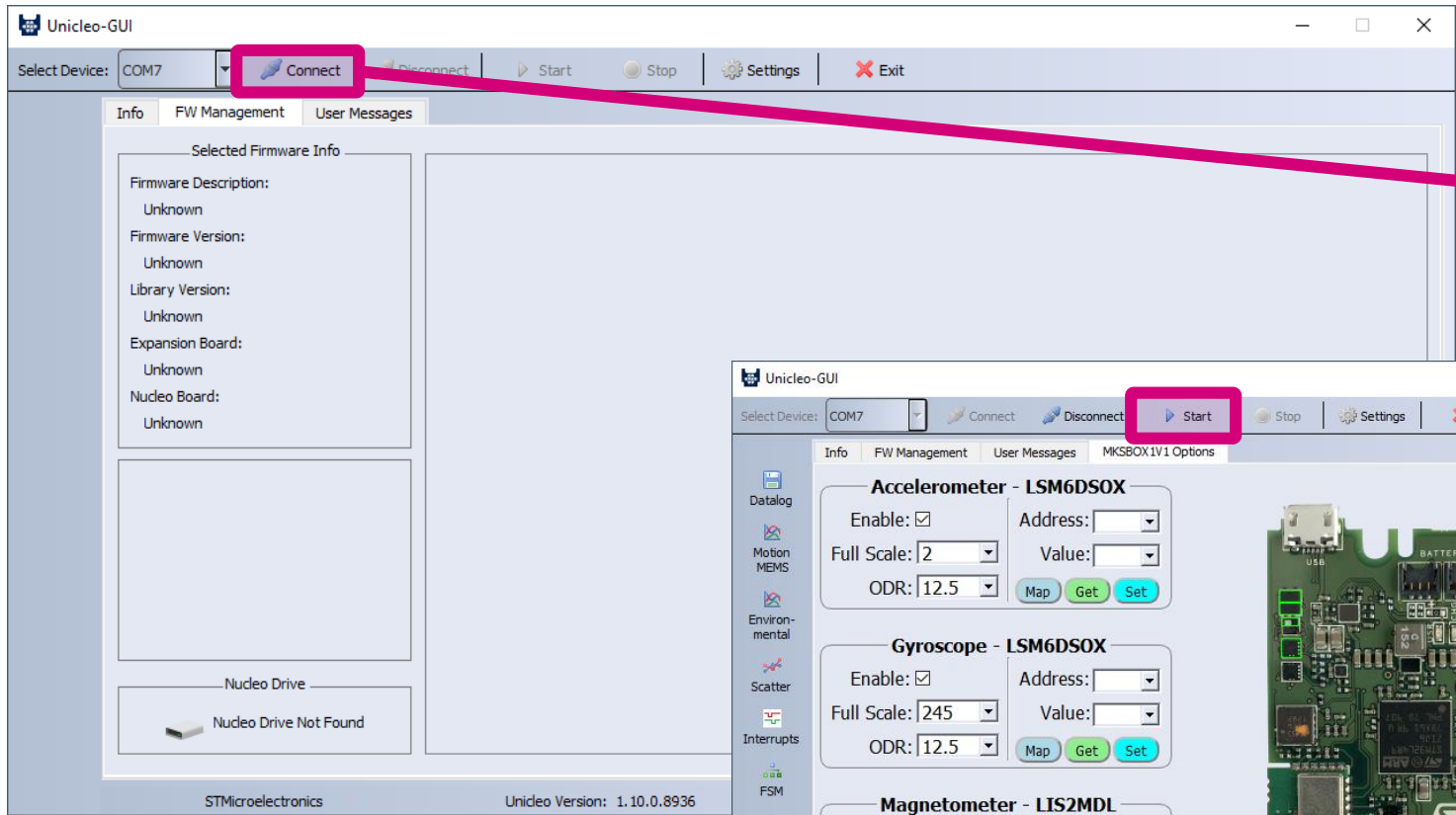
* Functions are added to the vertical toolbar based on firmware functionality



DataLogExtended with Unicleo-GUI

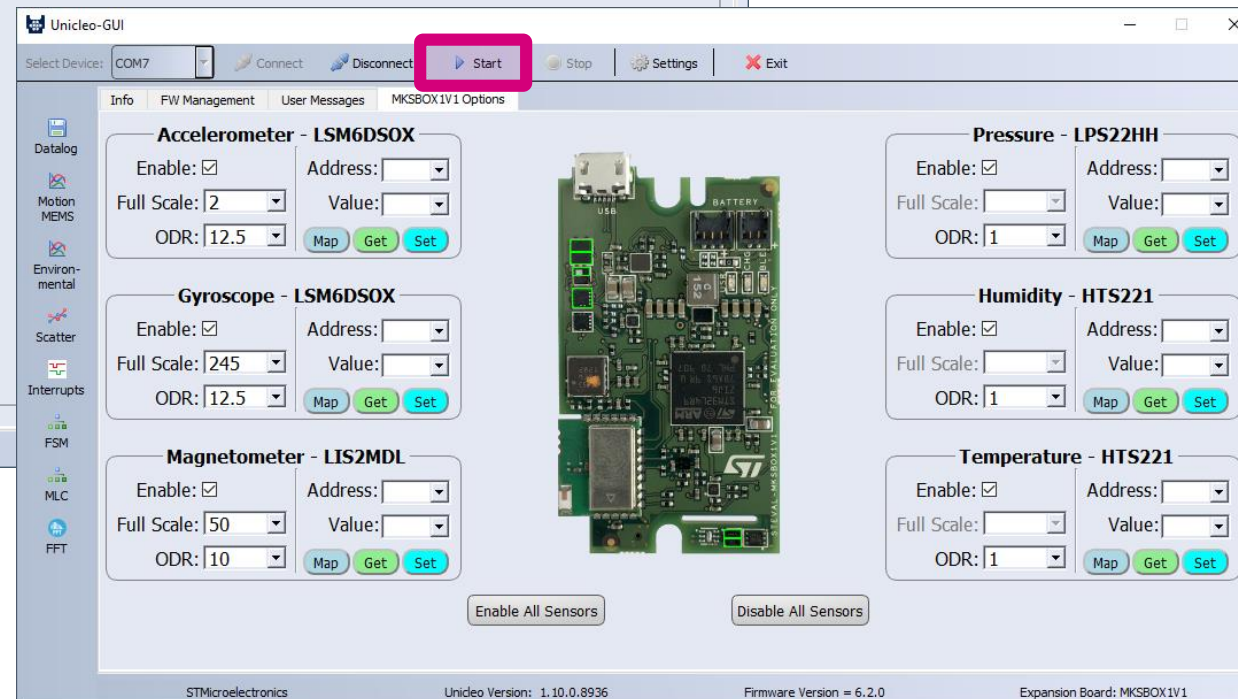
Connecting

1. Click **Connect**



2. Click **Apply**

3. Click **Start**



DataLogExtended with Unicleo-GUI

Expansion Board Options Tab

Unicleo-GUI

Select Device: COM7 | Connect | Disconnect | Start | Stop | Settings | Exit

Info | FW Management | User Messages | MKSBOX1V1 Options

Accelerometer - LSM6DSOX
Enable: | Address:
Full Scale: | Value:
ODR: | Map | Get | Set

Gyroscope - LSM6DSOX
Enable: | Address:
Full Scale: | Value:
ODR: | Map | Get | Set

Magnetometer - LIS2MDL
Enable: | Address:
Full Scale: | Value:
ODR: | Map | Get | Set

Pressure - LPS22HH
Enable: | Address:
Full Scale: | Value:
ODR: | Map | Get | Set

Humidity - HTS221
Enable: | Address:
Full Scale: | Value:
ODR: | Map | Get | Set

Temperature - HTS221
Enable: | Address:
Full Scale: | Value:
ODR: | Map | Get | Set

Enable All Sensors | Disable All Sensors

Firmware Version = 6.2.0 | Expansion Board: MKSBOX1V1

Enable/Disable sensor
Set Full Scale
Set ODR

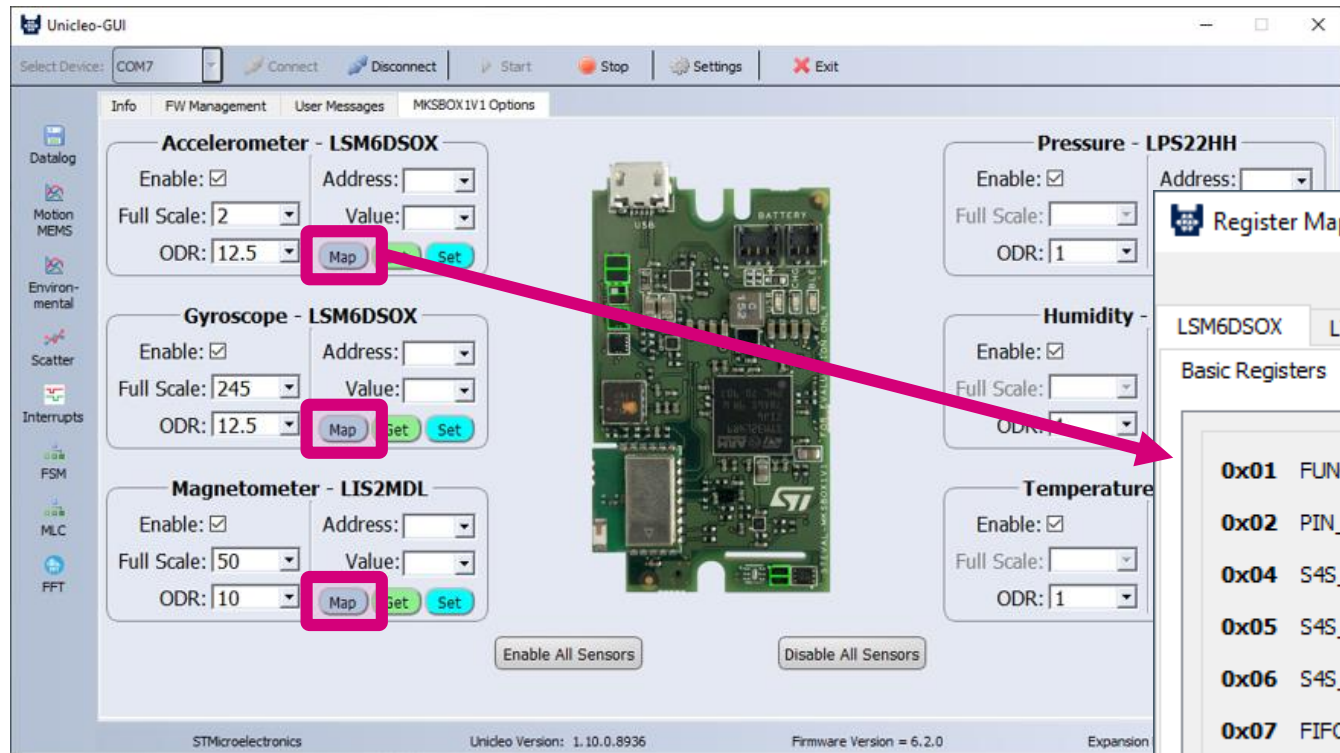
Read from register
Write to register



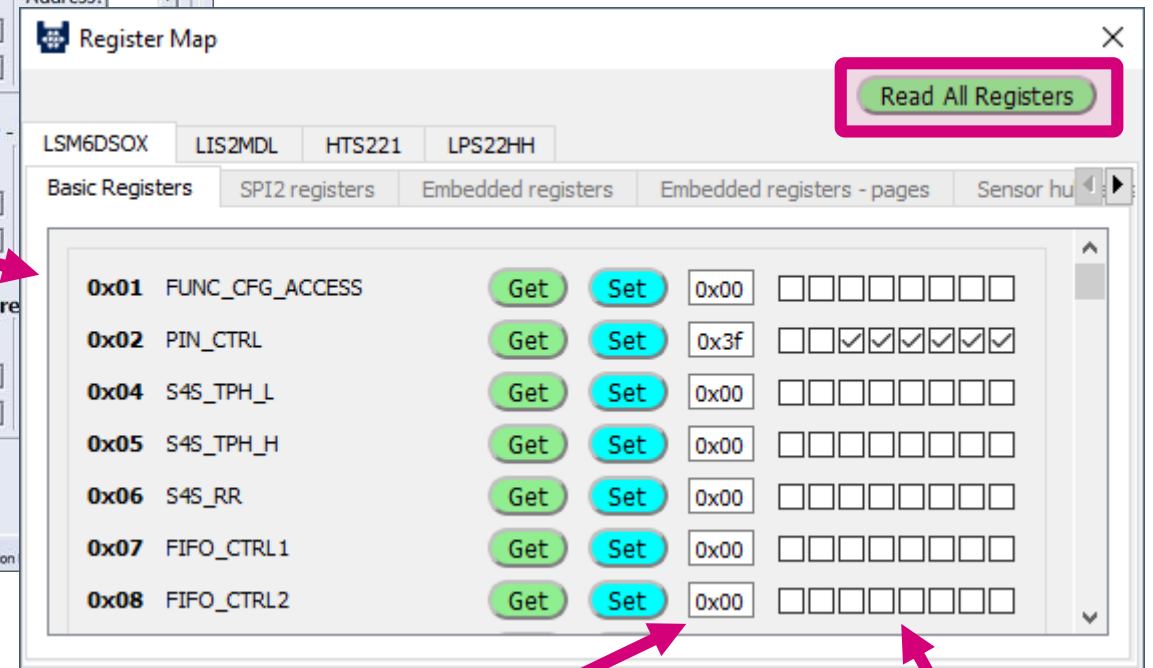
DataLogExtended with Unicleo-GUI

Register map

1. Click **MAP**



2. Read current values of all registers



Binary value of register

Hexadecimal value of register



DataLogExtended with Unicleo-GUI

Register map

- There are two ways how to read a register's description directly in Unicleo-GUI:
 - Move your mouse over a row of the register
 - Right-click on the register address

The screenshot displays the Unicleo-GUI interface for the Register Map of an LSM6DSOX sensor. The interface includes a 'Register Map' window with tabs for 'Basic Registers', 'SPI2 registers', 'Embedded registers', and 'Embedded registers - pages'. A 'Read All Registers' button is visible in the top right corner. The main window shows a list of registers with their addresses, names, and bit fields. Two registers are highlighted with pink boxes to illustrate the methods described in the text:

- Register 0x02 PIN_CTRL:** A pink box highlights the register name and its bit fields. A tooltip is visible over this register, providing a detailed description: "PIN_CTRL [7] OIS_PU_DIS: Disable pull-up on both OCS_Aux and SDO_Aux pins. Default value: 0 (0: OCS_Aux and SDO_Aux pins with pull-up; 1: OCS_Aux and SDO_Aux pins pull-up disconnected) [6] SDO_PU_EN: Enable pull-up on SDO pin (0: SDO pin pull-up disconnected (default); 1: SDO pin with pull-up)".
- Register 0x04 S4S_TPH_L:** A pink box highlights the register address. A tooltip is visible over this address, providing a detailed description: "S4S_TPH_L [7] TPH_H_SEL: Chooses if the TPH formula must be taken into account. [6:0] TPH_L[6:0]: S4S time frame expressed in number of samples as described in the equation below. If TPH_H_SEL=0 and TPH_L[6:0] = d0, S4S is disabled. Right Mouse Button click on address label will display this help permanently till next click."



DataLogExtended with Unicleo-GUI

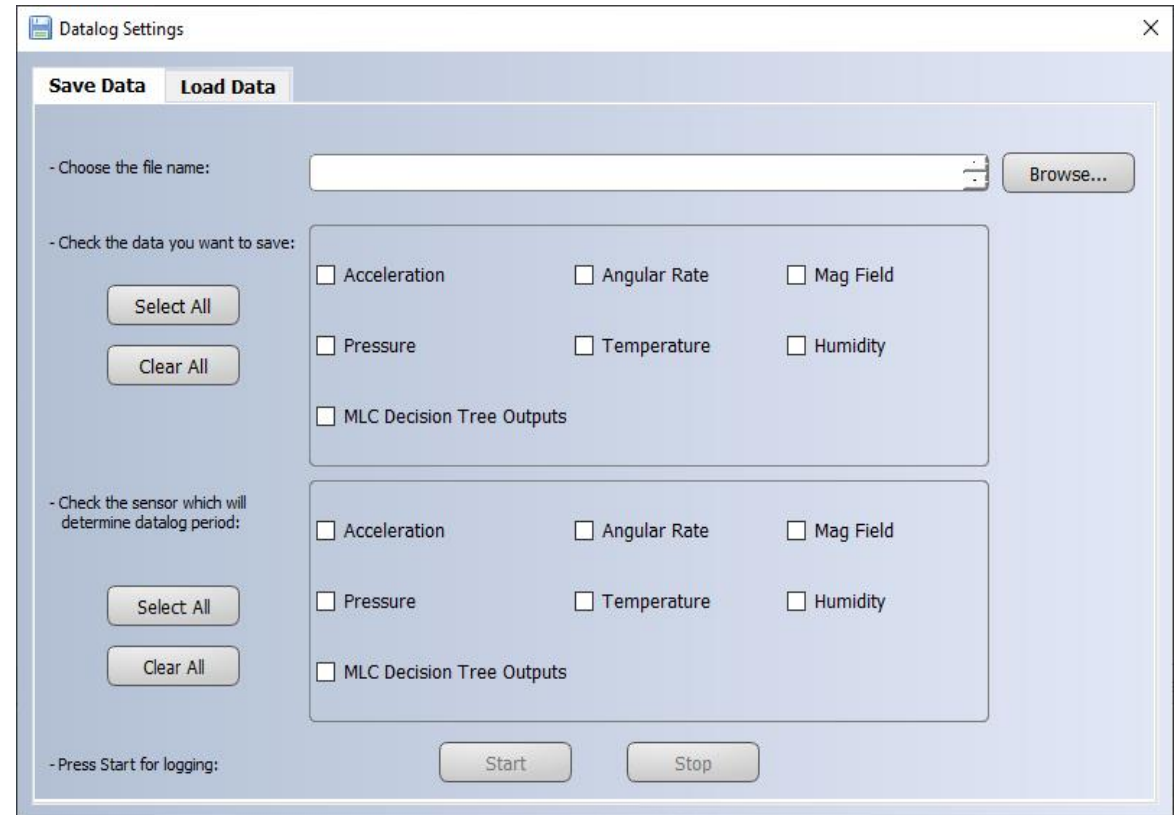
Datalog – Save/Load Options

- **Save options**

- CSV (comma separated values) or TSV (tab separate values) format
- Possible to select which data should be saved
- Selection of values is adjusted according to used firmware
- File name is automatically generated
- Dedicated directory for datalogs

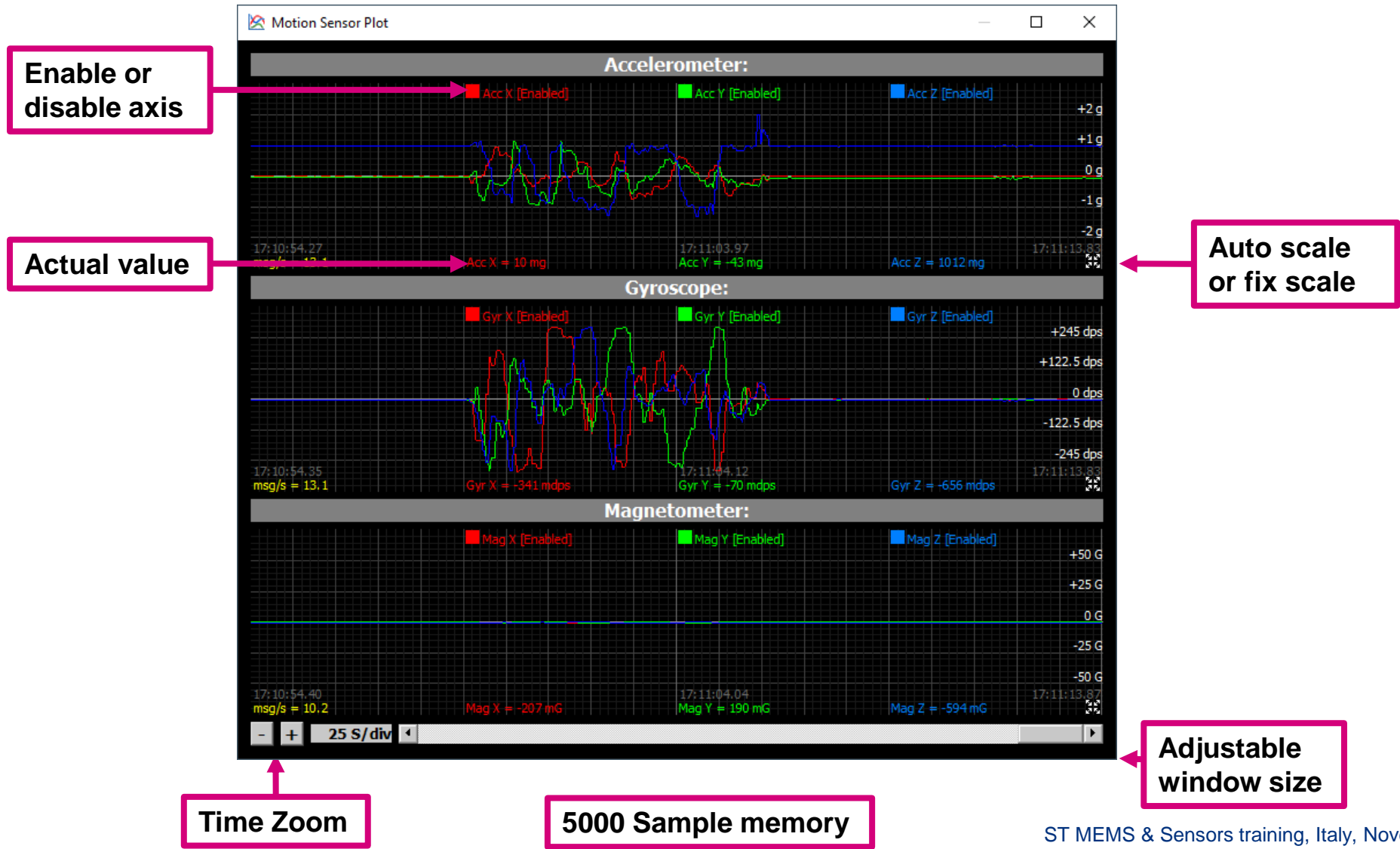
- **Load options**

- Possible to load and display previously stored data



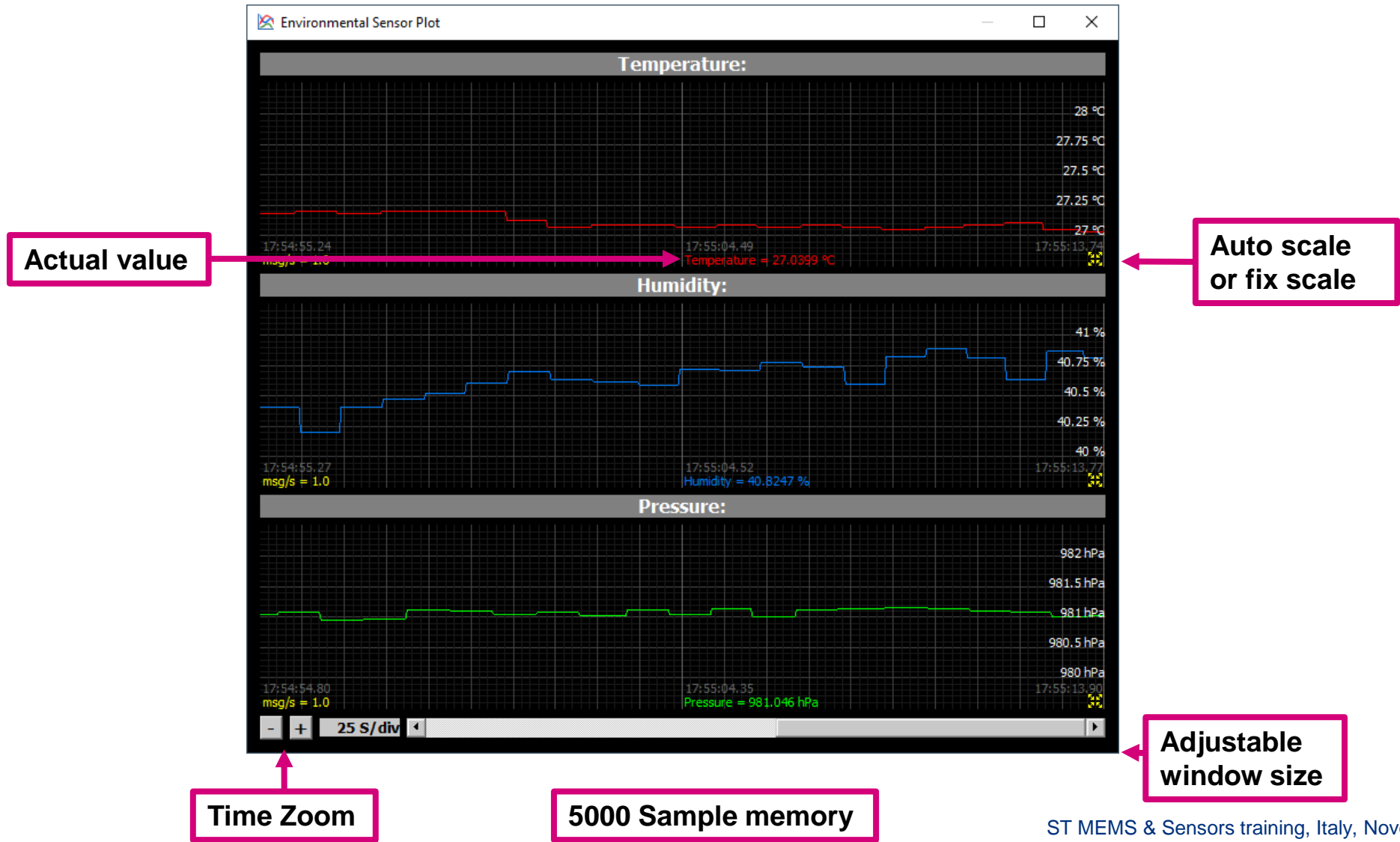
DataLogExtended with Unicleo-GUI

Plot – Motion Sensor



DataLogExtended with Unicleo-GUI

Plot – Environmental Sensor

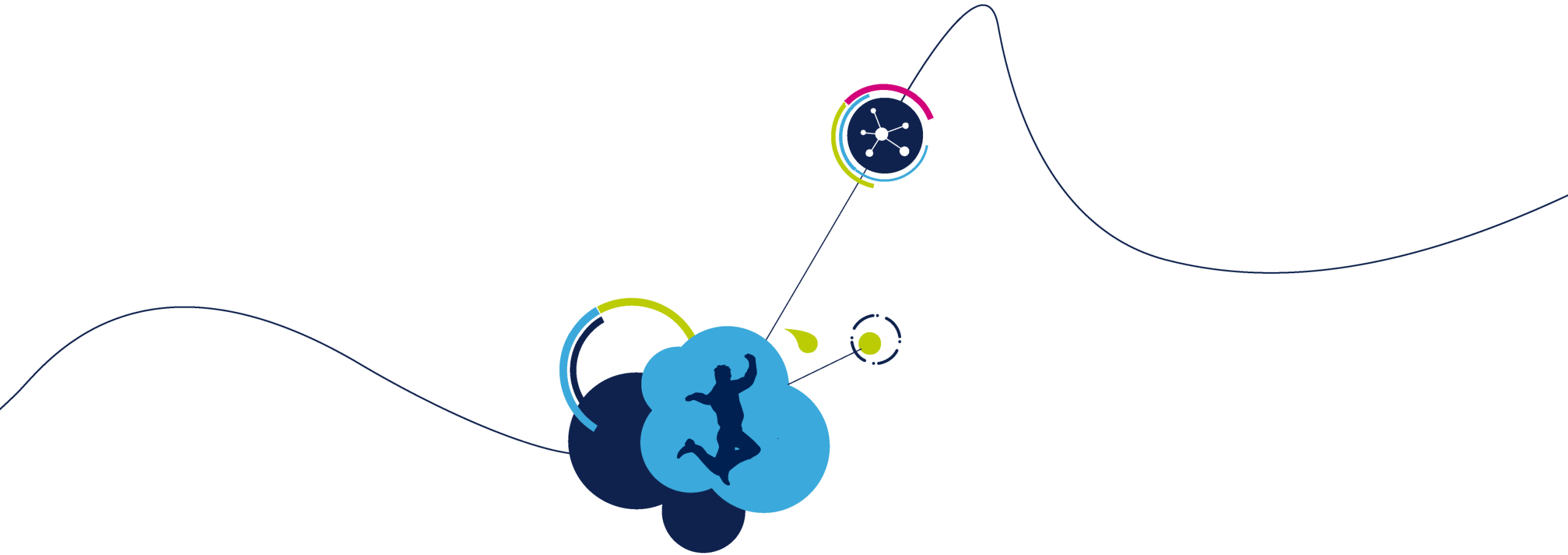


DataLogExtended with Unicleo-GUI

DataLogExtended example

See output data of all sensors

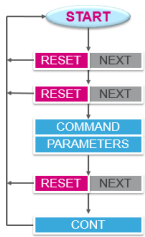




Hands-on #2

4D position recognition (FSM example)





Finite State Machine

FSM in ST sensors

Description

- The Finite State Machine is a series of state parameters and variables that allows running of low power programs
- FSM features
 - Runs up to 16 independent programs simultaneously
 - FSM can process axl, gyro data and external sensor like magnetometer or interface with MLC
 - User-programmable thresholds, hysteresis, timers and axis masks
 - Input from several sensors
- Development tools, documentation and support

Applications

- Smart Sensors and IoT
- Gesture recognition
- Movement detection

ST sensors

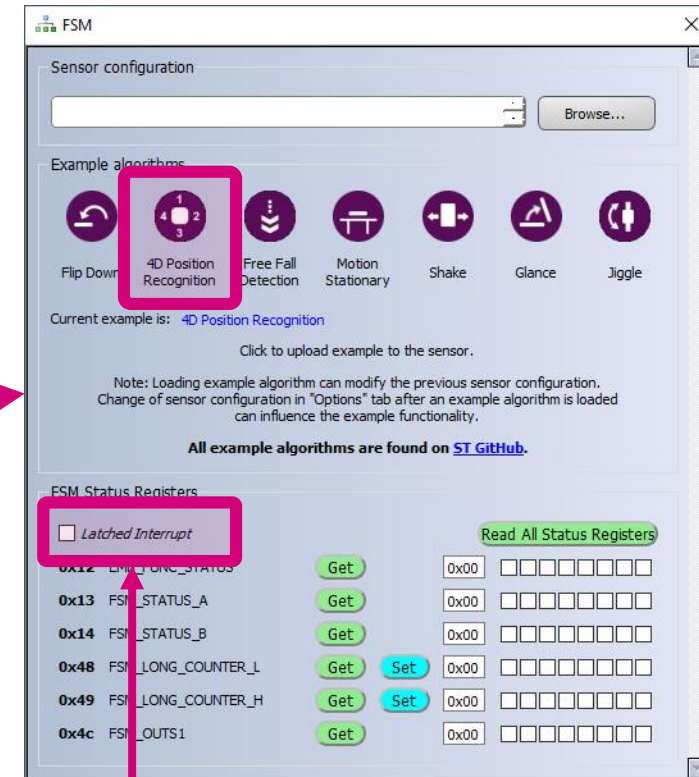
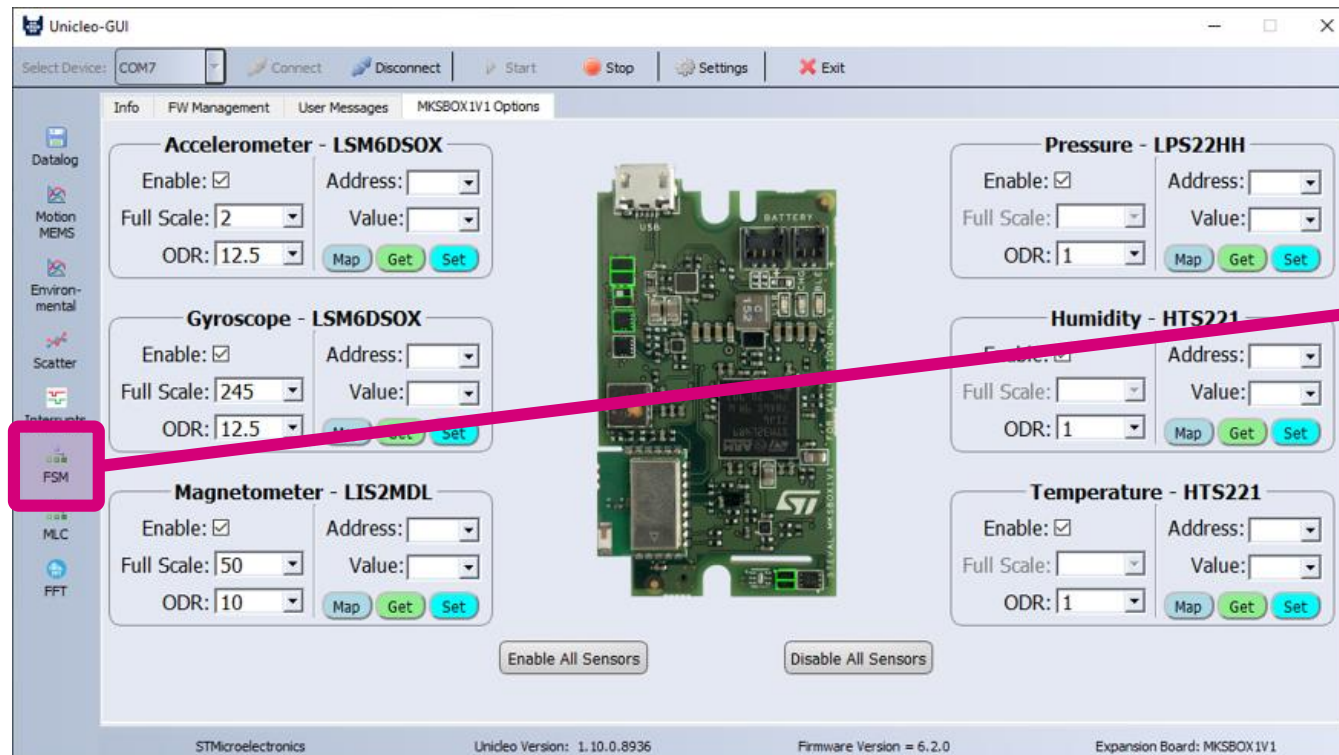
- LSM6DSO (6axis IMU)
- LSM6DSR (6axis IMU)
- ISM330DHCX (6axis IMU)



DataLogExtended with Unicleo-GUI

FSM example: 4D position recognition

1. Click on **FSM**
2. Select an example by clicking the desired icon



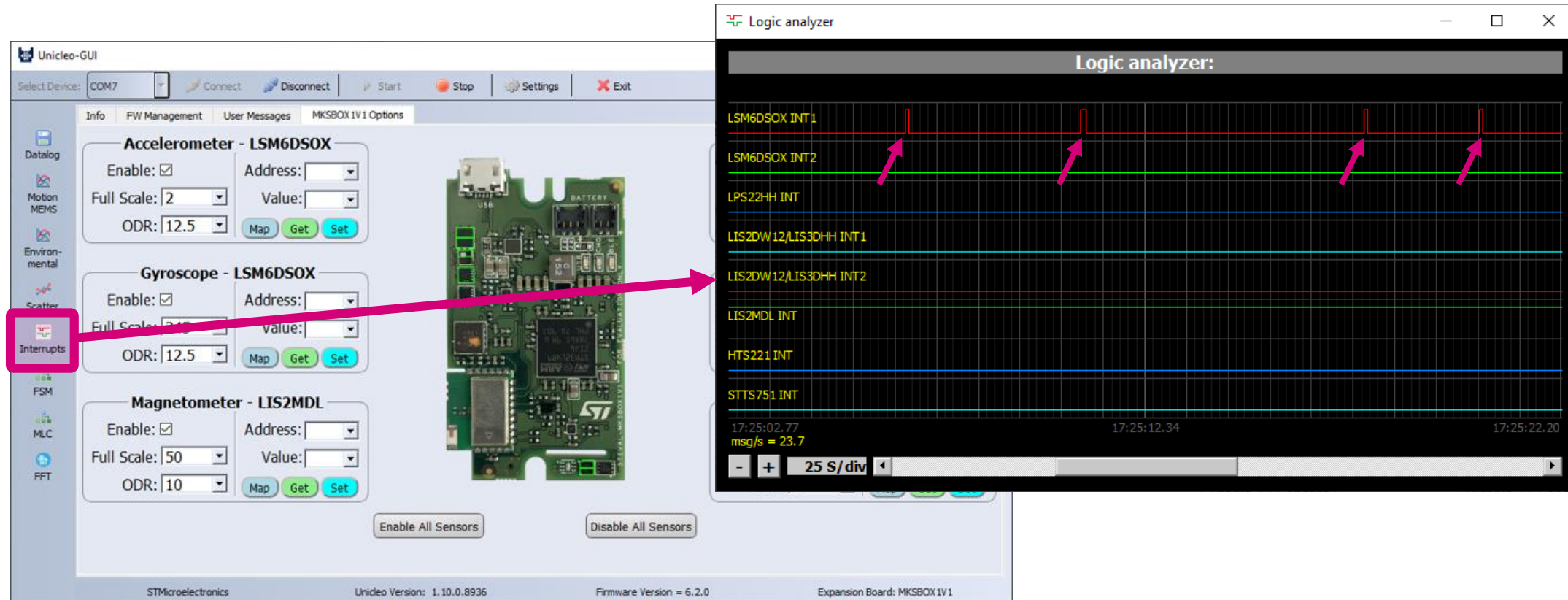
You can use latched or pulsed Interrupt mode



DataLogExtended with Unicleo-GUI

FSM example: 4D position recognition

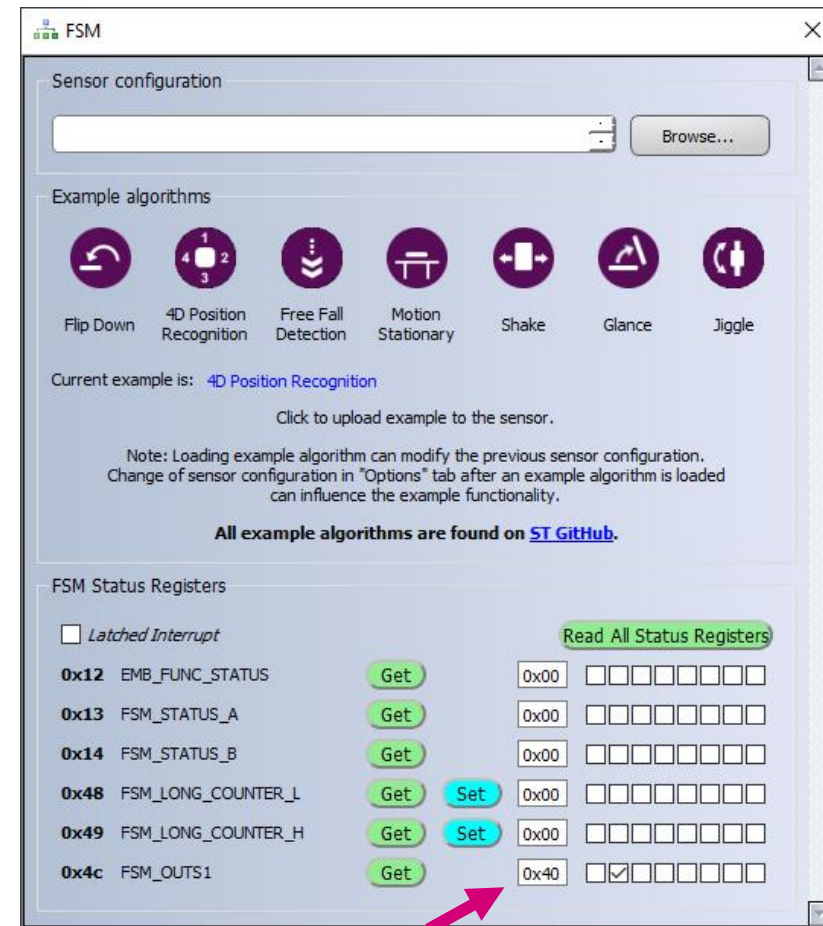
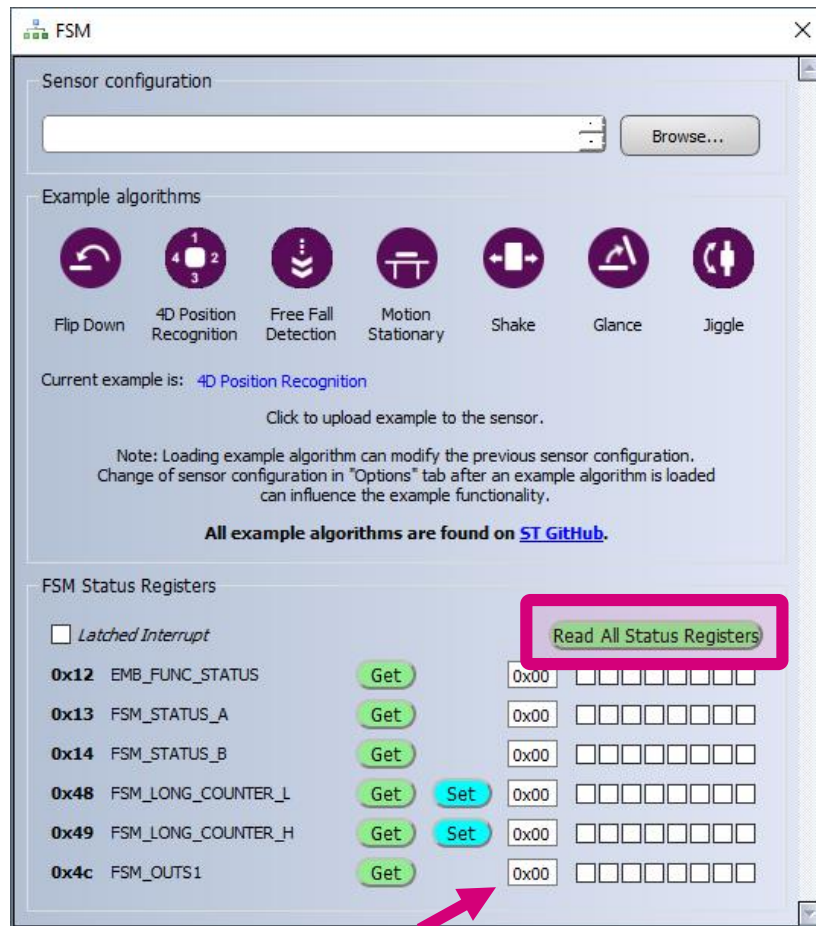
3. Perform desired action and check in **Interrupts** graph

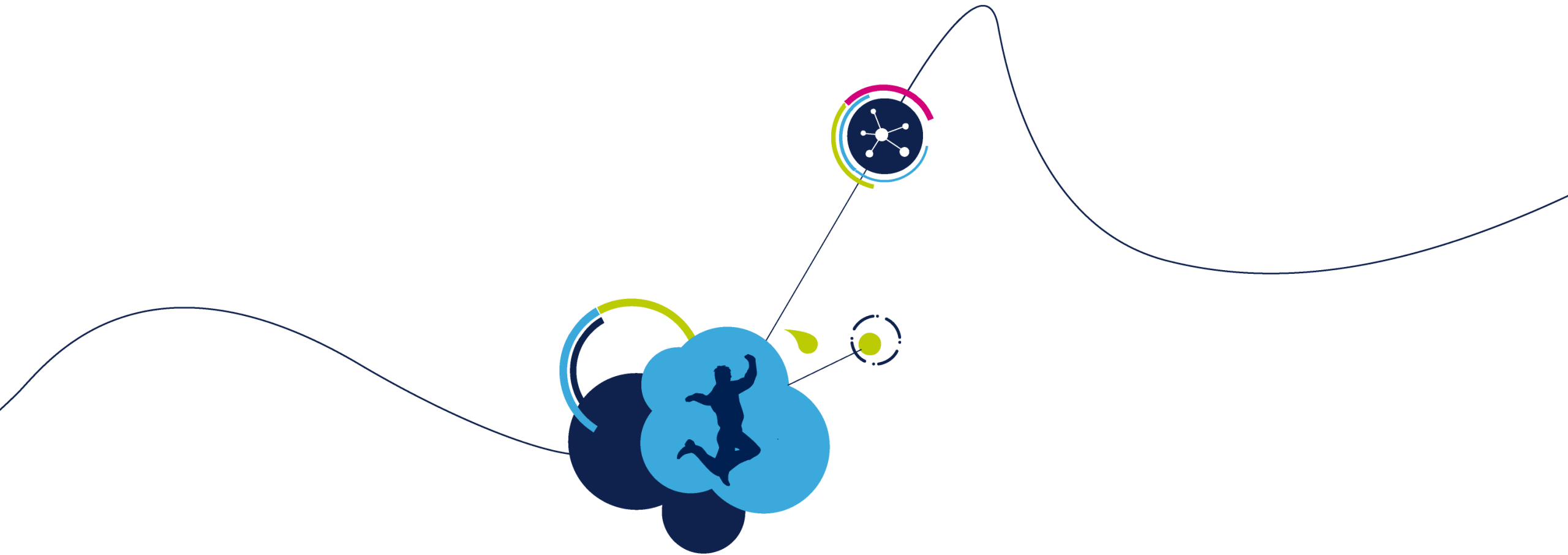


DataLogExtended with Unicleo-GUI

FSM example: 4D position recognition

4. Click **Read All Status Registers** to see how the FSM status registers (mainly the FSM output register **FSM_OUTS1**) are changed when desired action is performed

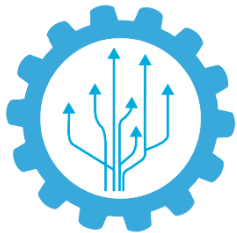




Hands-on #3

Vibration Monitoring (MLC example)





Machine Learning

MLC in ST sensors

48

Description

- MLC is programmable logic able to identify if a pattern matches an activity in a user defined set of classes: e.g. running, walking, driving, etc.
- MLC features:
 - Decision Tree logic embedded in sensor
 - Up to 8 decision trees can be configured to run simultaneously
 - Machine Learning approach
 - Input from several sensors
- Development tools, documentation and support

Applications

- Smart Sensors and IoT
- Activity tracking
- Pattern recognition

ST sensors

- LSM6DSOX (6axis IMU)
- ISM330DHCX (6axis IMU)
- AMS330LHHX (6axis IMU)

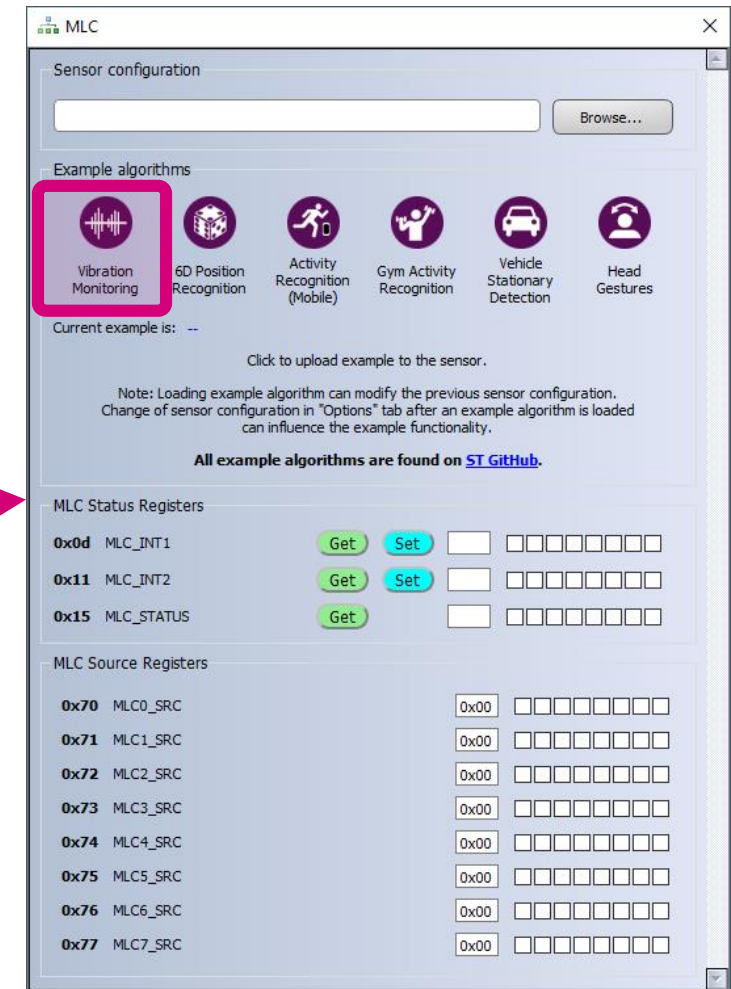
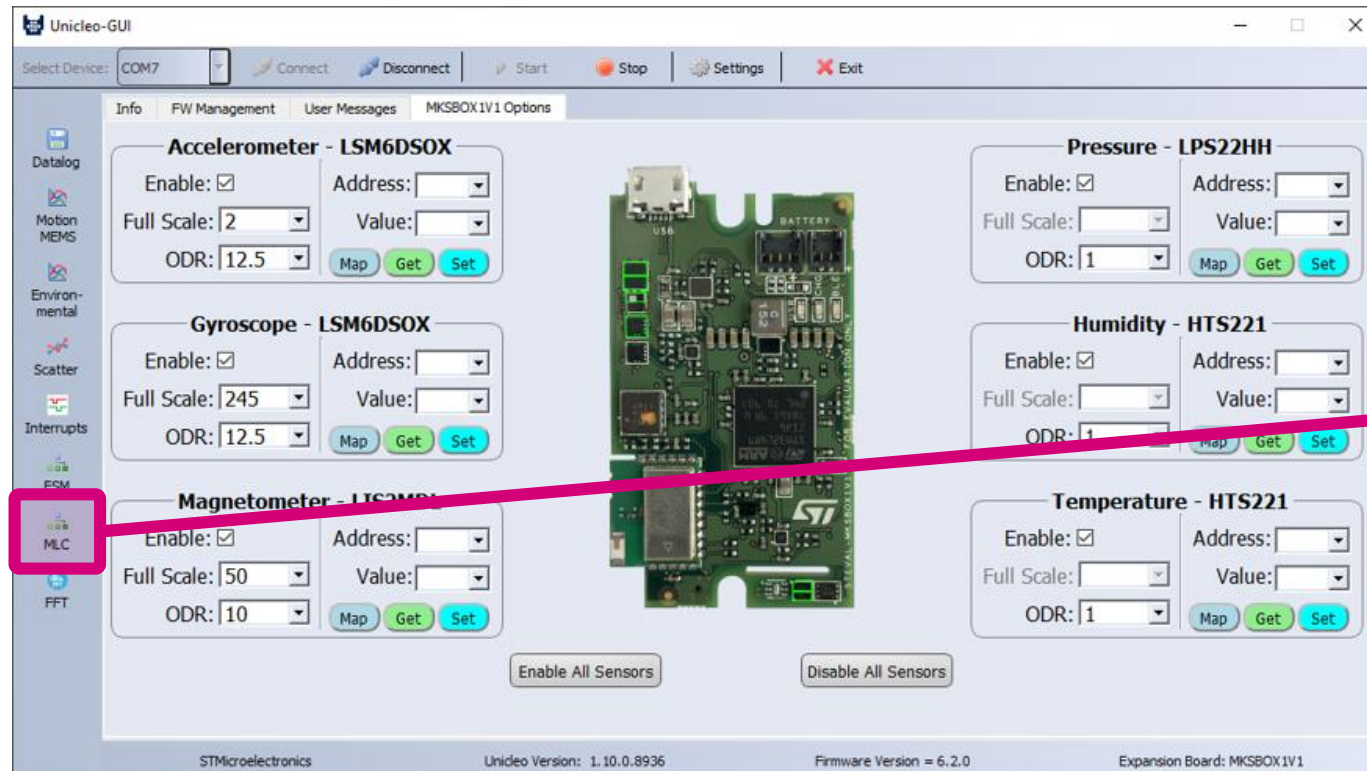
Get it [here!](#)



DataLogExtended with Unicleo-GUI

MLC example: Vibration Monitoring

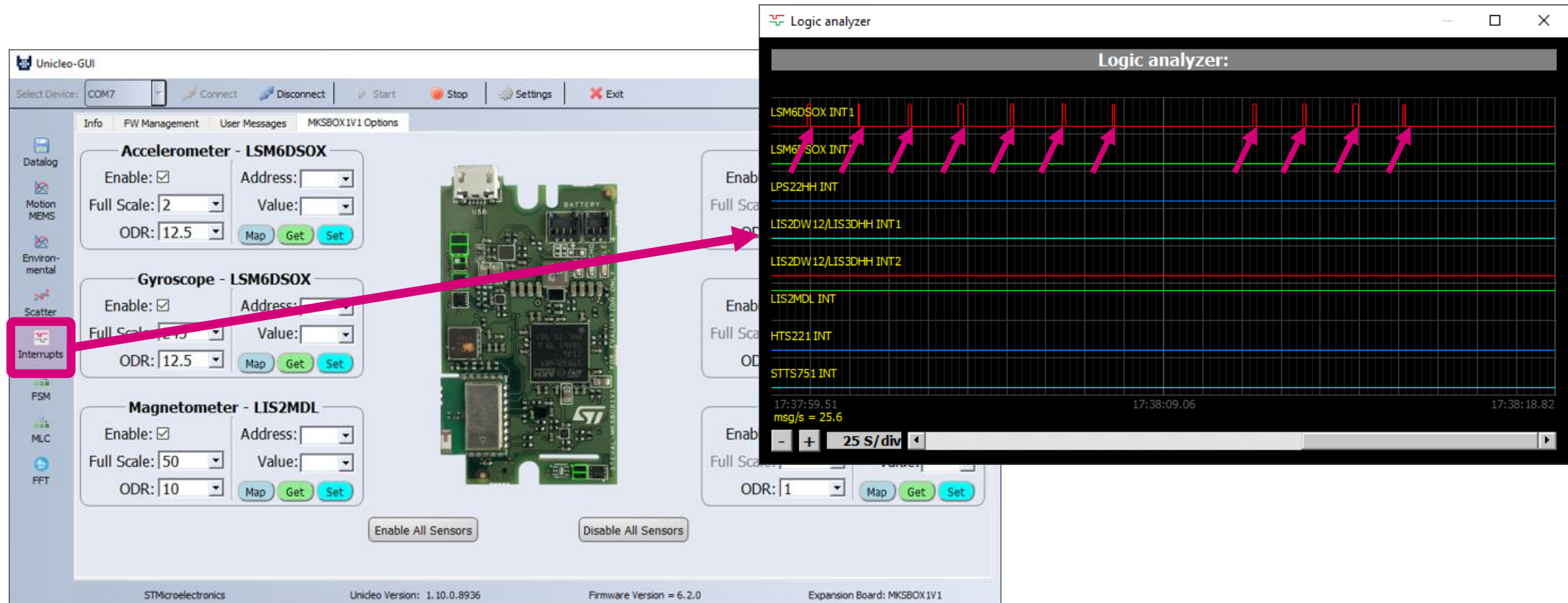
1. Click on **MLC**
2. Select an example by clicking the desired icon



DataLogExtended with Unicleo-GUI

MLC example: Vibration Monitoring

3. Perform desired action and check in **Interrupts** graph



DataLogExtended with Unicleo-GUI

MLC example: Vibration Monitoring

4. See how the MLC Source Registers are changed when desired action is performed

MLC Status Registers

0x0d	MLC_INT1	Get	Set	<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0x11	MLC_INT2	Get	Set	<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0x15	MLC_STATUS	Get		<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

MLC Source Registers

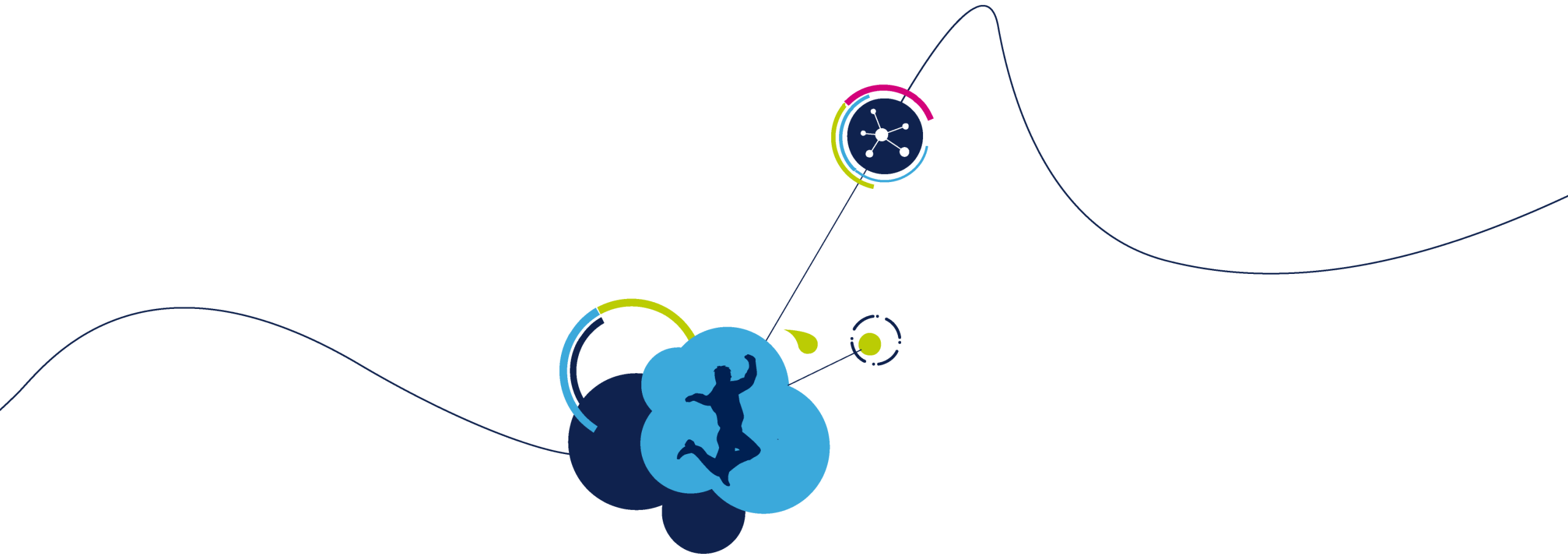
0x70	MLC0_SRC	<input type="text" value="0x00"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0x71	MLC1_SRC	<input type="text" value="0x00"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0x72	MLC2_SRC	<input type="text" value="0x00"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0x73	MLC3_SRC	<input type="text" value="0x00"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0x74	MLC4_SRC	<input type="text" value="0x00"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0x75	MLC5_SRC	<input type="text" value="0x00"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0x76	MLC6_SRC	<input type="text" value="0x00"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0x77	MLC7_SRC	<input type="text" value="0x00"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

MLC Status Registers

0x0d	MLC_INT1	Get	Set	<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0x11	MLC_INT2	Get	Set	<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0x15	MLC_STATUS	Get		<input type="text"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

MLC Source Registers

0x70	MLC0_SRC	<input type="text" value="0x02"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
0x71	MLC1_SRC	<input type="text" value="0x00"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0x72	MLC2_SRC	<input type="text" value="0x00"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0x73	MLC3_SRC	<input type="text" value="0x00"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0x74	MLC4_SRC	<input type="text" value="0x00"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0x75	MLC5_SRC	<input type="text" value="0x00"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0x76	MLC6_SRC	<input type="text" value="0x00"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0x77	MLC7_SRC	<input type="text" value="0x00"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>



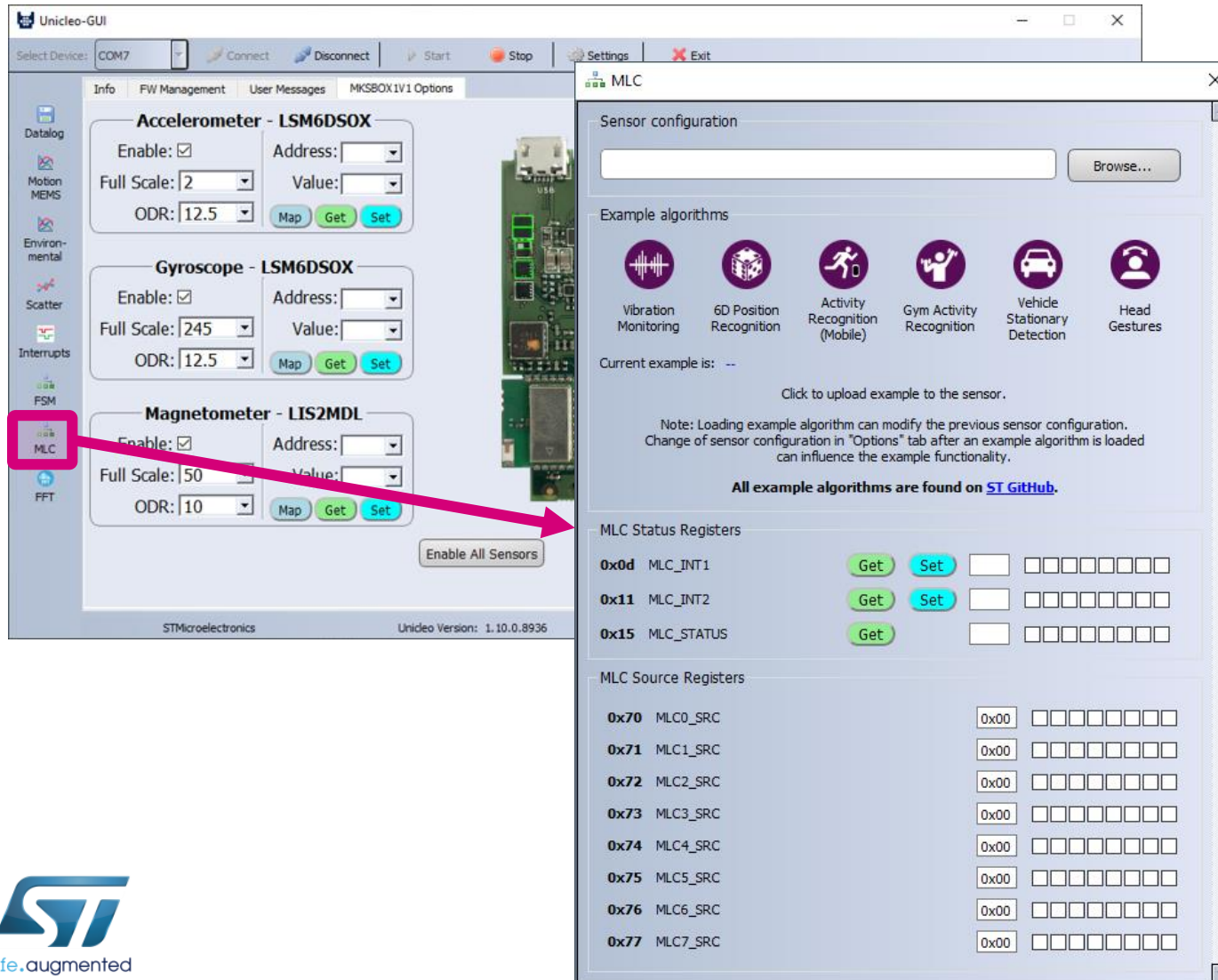
Hands-on #4

6D position recognition (customized MLC example)



DataLogExtended with Unicleo-GUI

customized MLC example: 6D orientation detection



You can also use with Unicleo-GUI **your own algorithm** (created in Unico)!

We will now demonstrate how to upload external configuration (.ucf) file with using an example algorithm from **ST GitHub**...

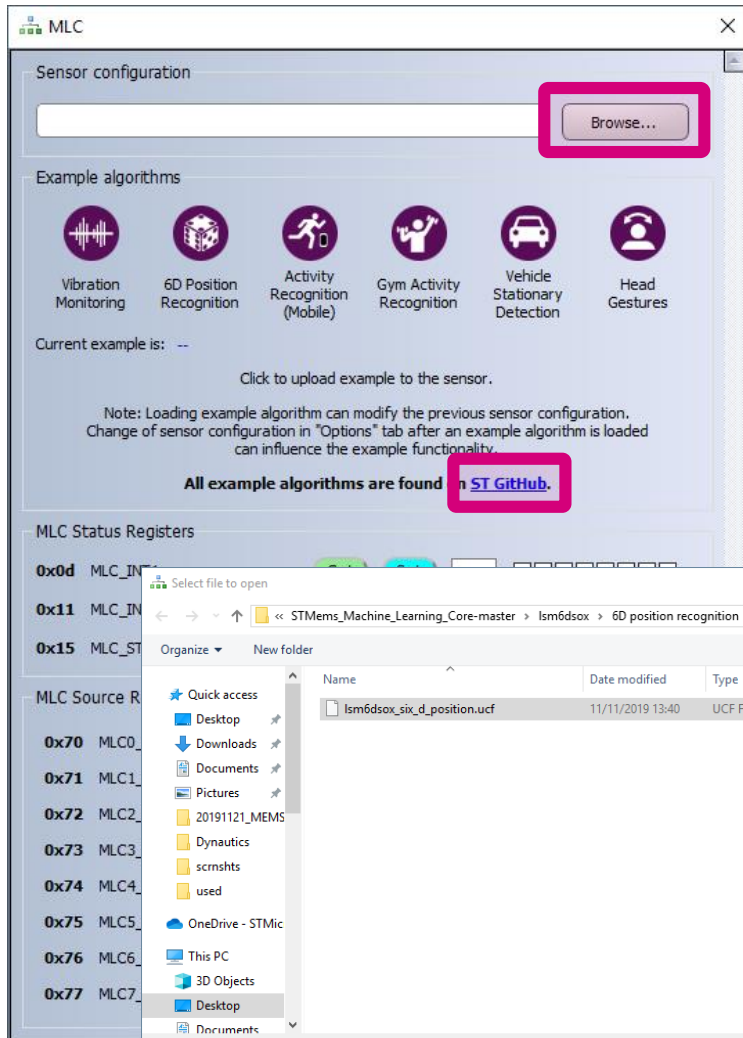
1. Click **MLC**



DataLogExtended with Unicleo-GUI

customized MLC example: 6D orientation detection

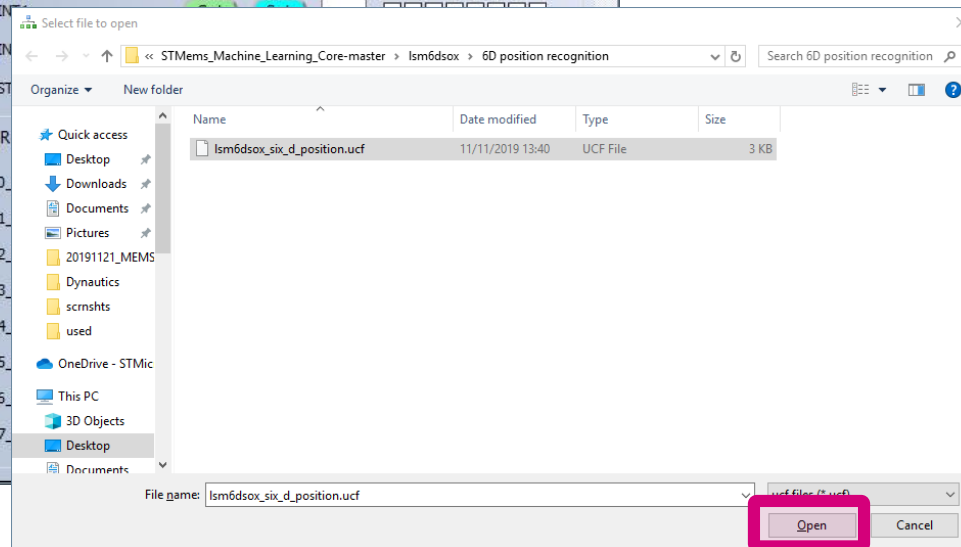
54



2. Click **Browse** and navigate to:

...\\STMems_Machine_Learning_Core-master\\lsm6dsox\\6D position recognition\\lsm6dsox_six_d_position.ucf

3. Click **Open**



Example algorithms can be downloaded from ST GitHub -> click the [ST GitHub](#) (with link) in the MLC window



DataLogExtended with Unicleo-GUI

customized MLC example: 6D orientation detection



4. Check that the current example is **lsm6dsox_six_d_position.ucf**

5. Perform desired action and check in **Interrupts** graph and source registers

Unicleo-GUI

Select Device: COM7 Connect Disconnect Start Stop Settings Exit

Info FW Management User Messages MKSBOX1V1 Options

Accelerometer - LSM6DSOX

Enable: Address: [] Full Scale: [2] Value: [] ODR: [12.5] Map Get Set

Gyroscope - LSM6DSOX

Enable: Address: [] Full Scale: [] Value: [] ODR: [12.5] Map Get Set

Magnetometer - LIS2MDL

Enable: Address: [] Full Scale: [50] Value: [] ODR: [10] Map Get Set

Interrupts

Logic analyzer

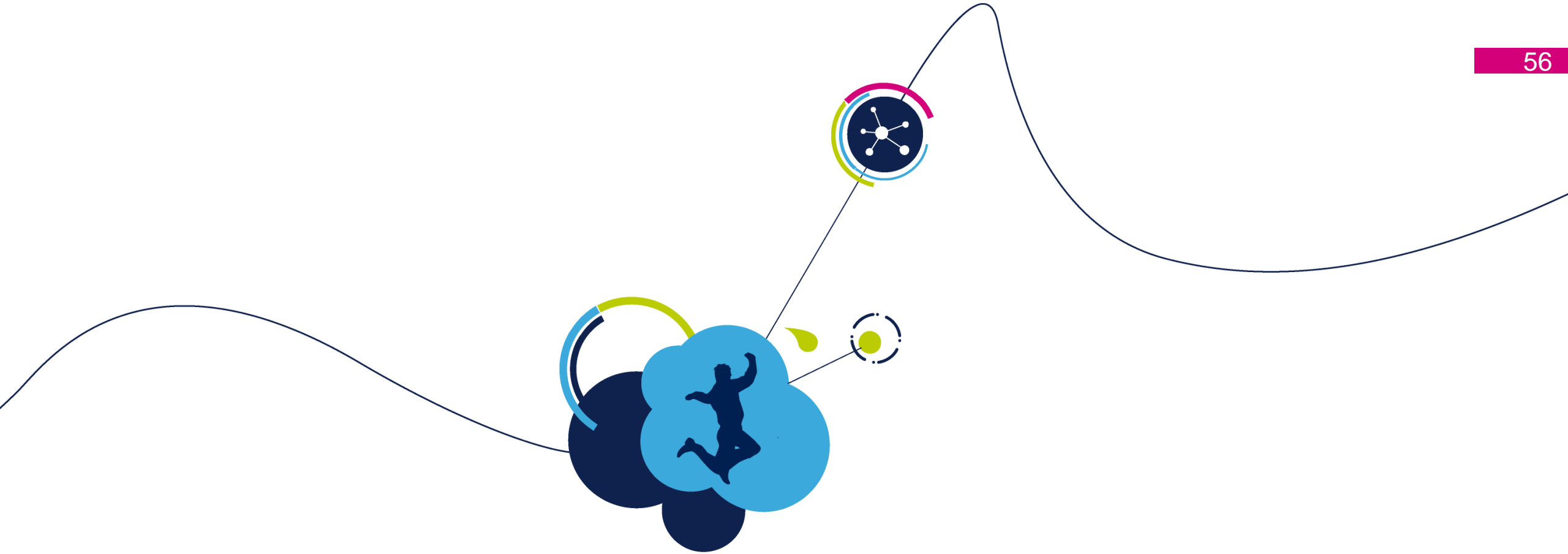
Logic analyzer:

- LSM6DSOX INT1
- LSM6DSOX INT2
- LPS22HH INT
- LIS2DW12/LIS3DHH INT1
- LIS2DW12/LIS3DHH INT2
- LIS2MDL INT
- HTS221 INT
- STTS751 INT

10:24:02.40 msg/s = 39.0

40 S/div





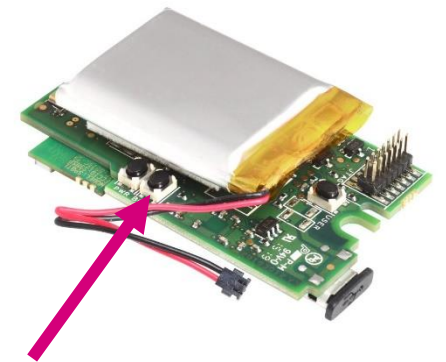
Restore the original FW



Restore the original FW

Introduction

- The binary must be loaded using DFU (Direct Firmware Upgrade over USB)
- These are the steps:
 1. Run ST BLE Sensor app to enter DFU mode
 - a) Connect to your device, open the debug console and issue the “DFU” command
 - b) Hold **BOOT** button during power up (e.g. when connecting to a laptop with a micro USB cable while the battery is disconnected)
 2. Run STM32CubeProg to program the device
 - Select USB and push “Connect”, Mass Erase then Program new binary



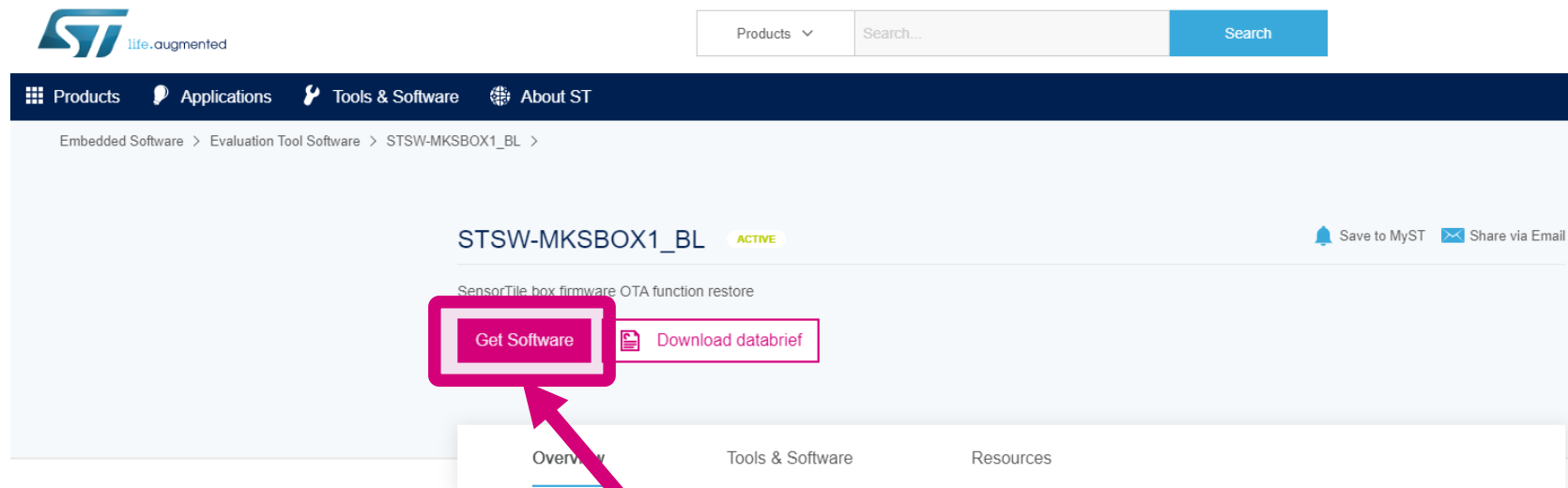
BOOT button



Restore the original FW

1. Download **STSW-MKSBOX1_BL** - SensorTile.box firmware OTA function restore

<https://www.st.com/en/embedded-software/stsw-mksbox1-bl.html>

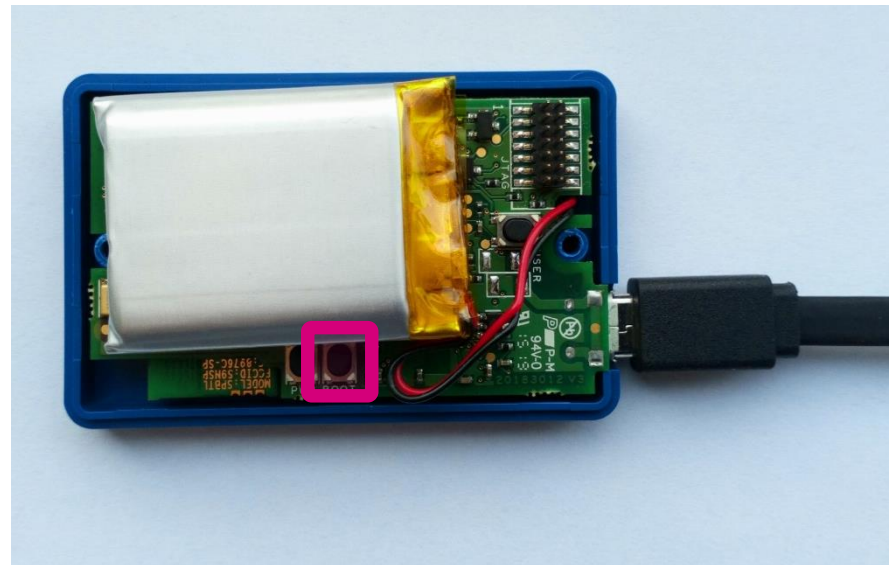


Click on **Get Software, Accept** and log in to your myST account or fill in your credentials



Restore the original FW

2. Press the **BOOT button** on SensorTile.Box, hold it and connect SensorTile.Box to your PC with micro USB cable. The device will enter the **DFU mode**.



Restore the original FW

1. Run
STM32CubeProgrammer



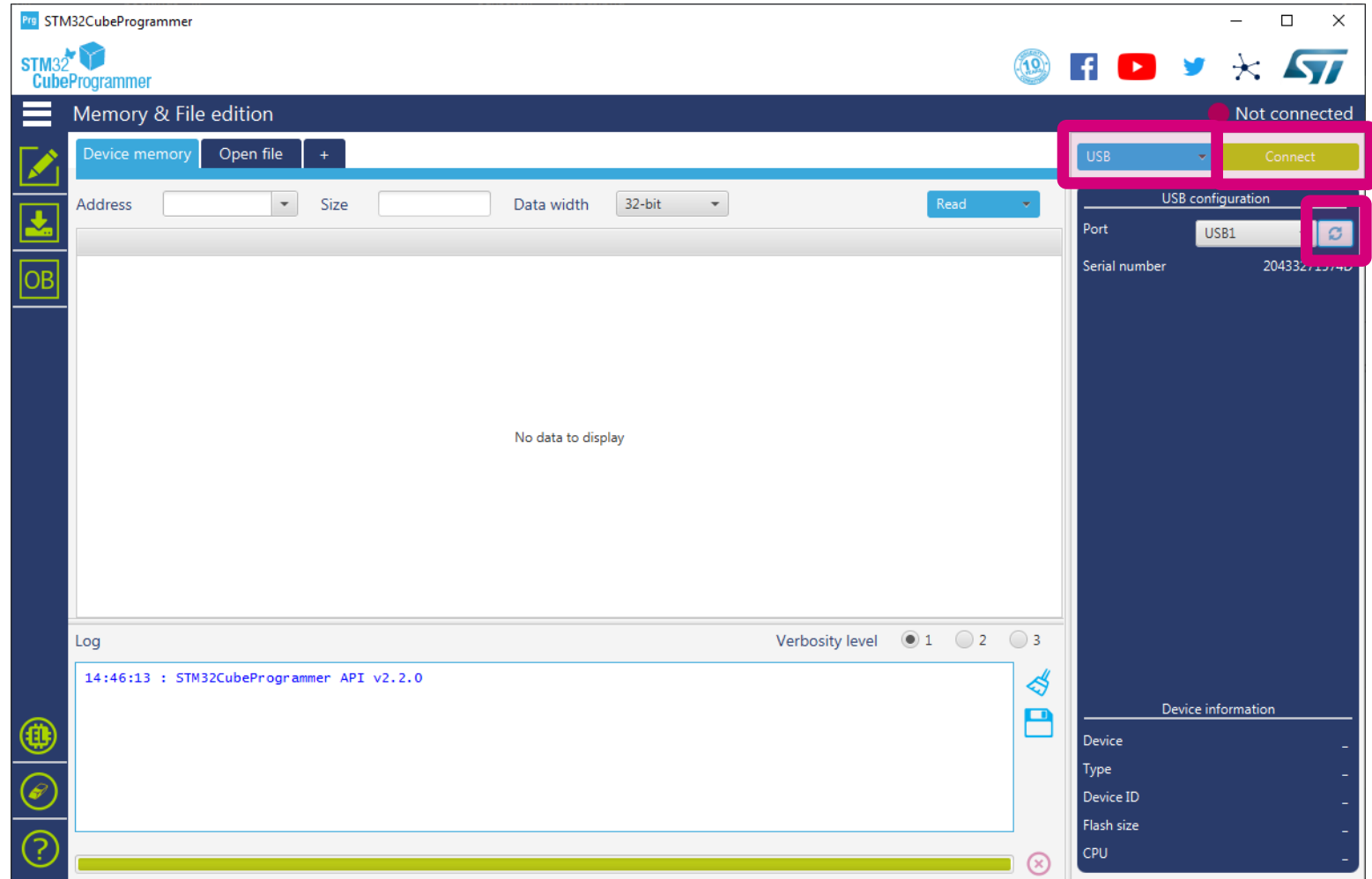
2. Wait for STM32CubeProgrammer
to open



Restore the original FW

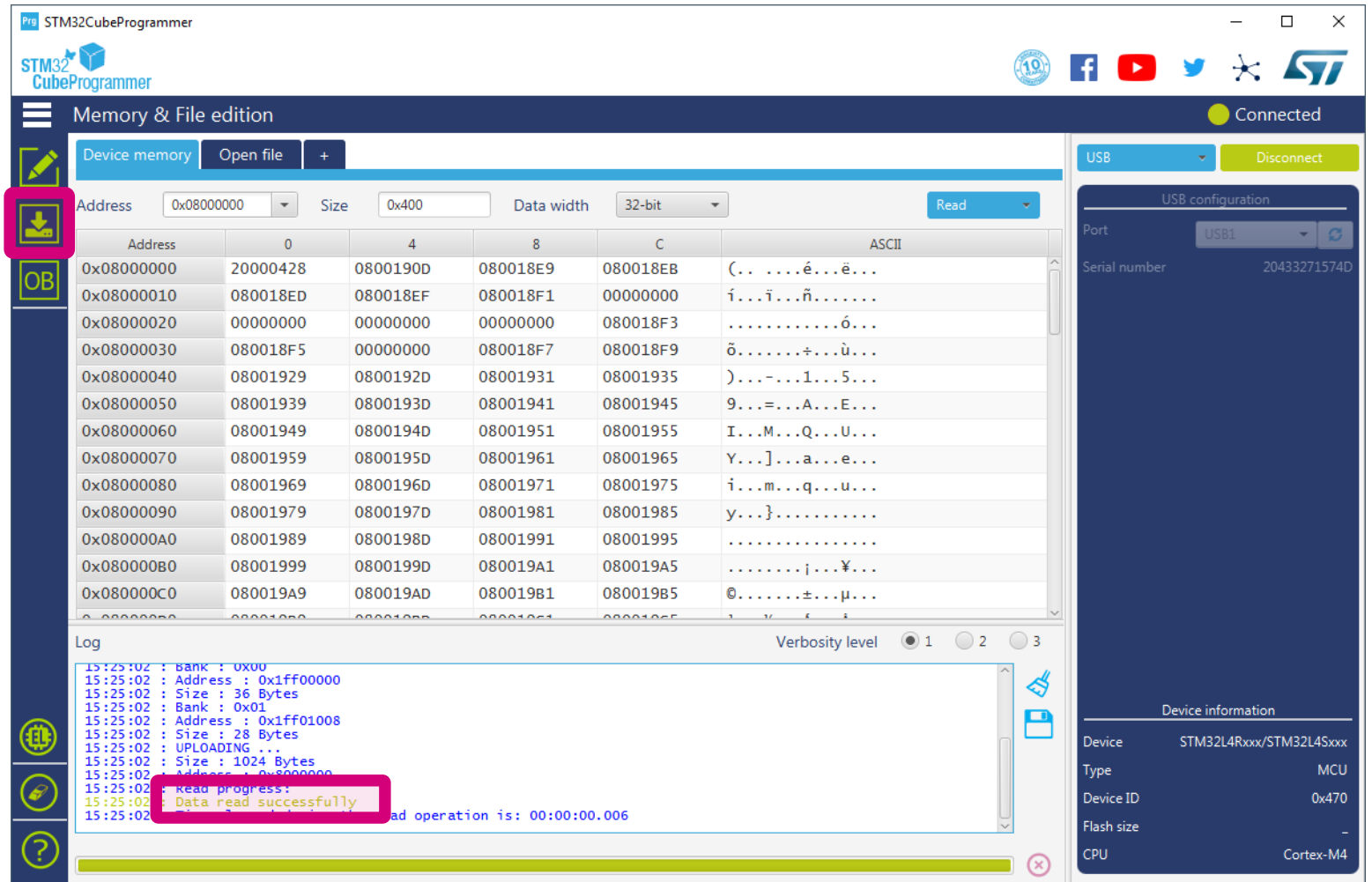
61

1. Select **USB** from drop-down menu
2. Click **Refresh**
3. Click **Connect**



Restore the original FW

1. Check that Data is **read successfully**
2. Select **Erasing & Programming** from the menu on the left



STM32CubeProgrammer

Memory & File edition

Device memory | Open file | +

Address: 0x08000000 | Size: 0x400 | Data width: 32-bit | Read

Address	0	4	8	C	ASCII
0x08000000	20000428	0800190D	080018E9	080018EB	(. é ë
0x08000010	080018ED	080018EF	080018F1	00000000	í ï ñ
0x08000020	00000000	00000000	00000000	080018F3 ó
0x08000030	080018F5	00000000	080018F7	080018F9	õ + ù
0x08000040	08001929	0800192D	08001931	08001935) 1 5
0x08000050	08001939	0800193D	08001941	08001945	9 = A E
0x08000060	08001949	0800194D	08001951	08001955	I M Q U
0x08000070	08001959	0800195D	08001961	08001965	Y] a e
0x08000080	08001969	0800196D	08001971	08001975	i m q u
0x08000090	08001979	0800197D	08001981	08001985	y }
0x080000A0	08001989	0800198D	08001991	08001995
0x080000B0	08001999	0800199D	080019A1	080019A5 j ¥
0x080000C0	080019A9	080019AD	080019B1	080019B5	@ ± μ

Log

Verbosity level: 1 2 3

```
15:25:02 : Bank : 0x00
15:25:02 : Address : 0x1ff00000
15:25:02 : Size : 36 Bytes
15:25:02 : Bank : 0x01
15:25:02 : Address : 0x1ff01008
15:25:02 : Size : 28 Bytes
15:25:02 : UPLOADING ...
15:25:02 : Size : 1024 Bytes
15:25:02 : Address : 0x8000000
15:25:02 : Read progress:
15:25:02 : Data read successfully
15:25:02 : Read operation is: 00:00:00.006
```

USB configuration

Port: USB1 | Disconnect

Serial number: 20433271574D

Device information

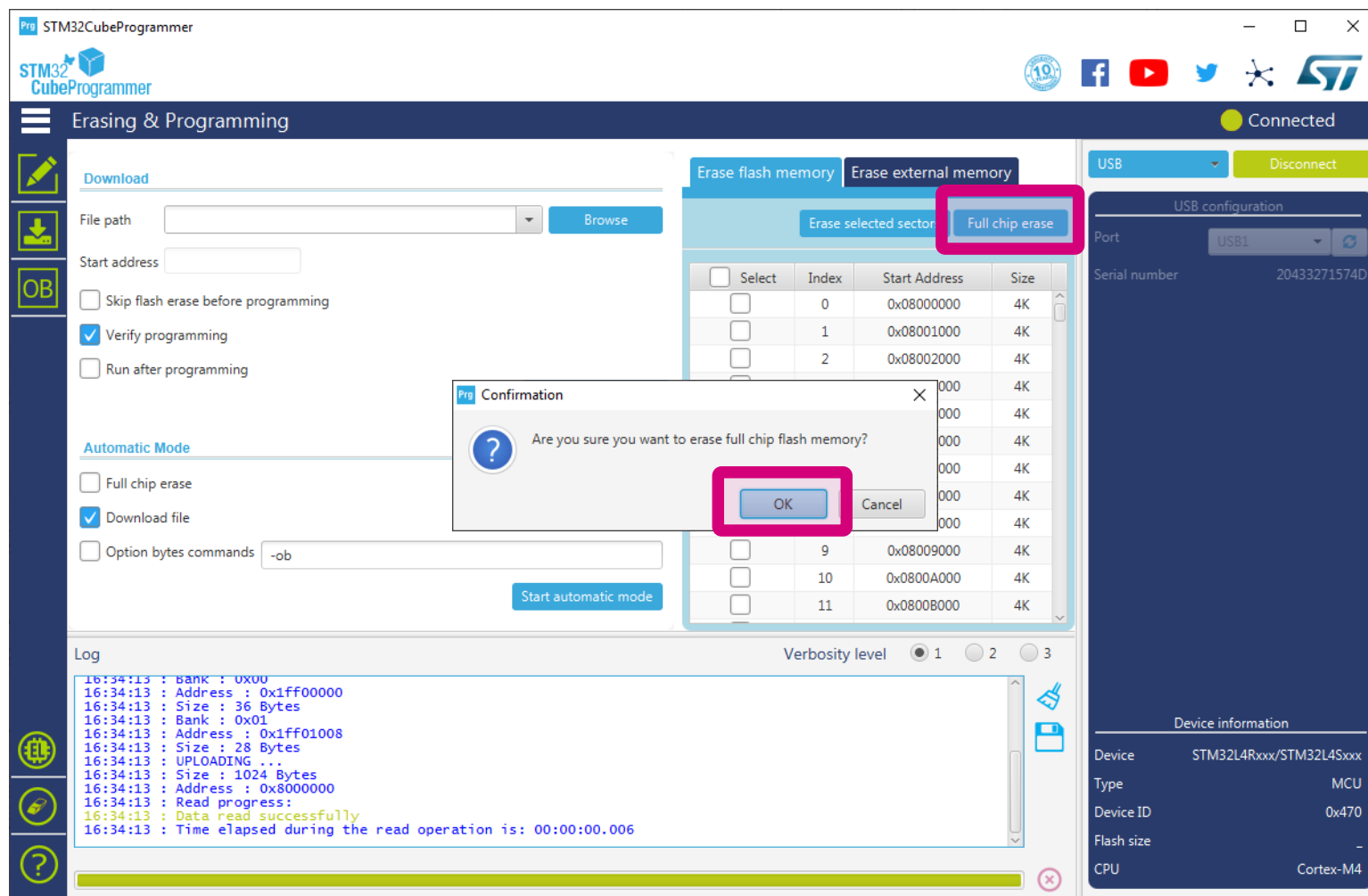
Device: STM32L4Rxxx/STM32L4Sxxx
Type: MCU
Device ID: 0x470
Flash size: -
CPU: Cortex-M4



Restore the original FW

63

1. Select **Full chip erase**
2. Click **OK**



Restore the original FW

64

1. Wait until Full chip erase is completed
2. Click **OK**

The screenshot shows the STM32CubeProgrammer software interface. The main window is titled "Erasing & Programming" and has several tabs: "Download", "Erase flash memory", and "Erase external memory". The "Erase flash memory" tab is active, showing a table of sectors to be erased. A message dialog box is overlaid on the interface, with the "OK" button highlighted in pink. The log window at the bottom shows the following text:

```
Log  
16:34:13 : read progress:  
16:34:13 : Data read successfully  
16:34:13 : Time elapsed during the read operation is: 00:00:00.006  
16:34:55 : MASS ERASE ...  
16:34:55 : Mass erase command correctly executed. Note: if there's any flash protection, it will not be erased.  
16:34:55 : UPLOADING ...  
16:34:55 : Size : 1024 Bytes  
16:34:55 : Address : 0x8000000  
16:34:55 : Read progress:  
16:34:55 : Data read successfully  
16:34:55 : Time elapsed during the read operation is: 00:00:00.004
```



Restore the original FW

65

1. Click **Browse** and navigate to **BLEFOTA_BL_1.0.0.bin**
2. Check that **Start address** is set to **0x08000000**
3. Check **Verify programming** (optional)
4. Click **Start Programming**

The screenshot displays the STM32CubeProgrammer interface. The 'Erasing & Programming' window is active, showing the 'Download' section with the file path 'C:\STM32CubeFunctionPack_STBOX1_V1.0.0\Projects\ST1' and the start address '0x08000000'. The 'Verify programming' checkbox is checked. The 'Start Programming' button is highlighted. The 'Automatic Mode' section has 'Download file' checked. The 'Log' window shows a successful data read operation.

Select	Index	Start Address	Size
<input type="checkbox"/>	0	0x08000000	4K
<input type="checkbox"/>	1	0x08001000	4K
<input type="checkbox"/>	2	0x08002000	4K
<input type="checkbox"/>	3	0x08003000	4K
<input type="checkbox"/>	4	0x08004000	4K
<input type="checkbox"/>	5	0x08005000	4K
<input type="checkbox"/>	6	0x08006000	4K
<input type="checkbox"/>	7	0x08007000	4K
<input type="checkbox"/>	8	0x08008000	4K
<input type="checkbox"/>	9	0x08009000	4K
<input type="checkbox"/>	10	0x0800A000	4K
<input type="checkbox"/>	11	0x0800B000	4K

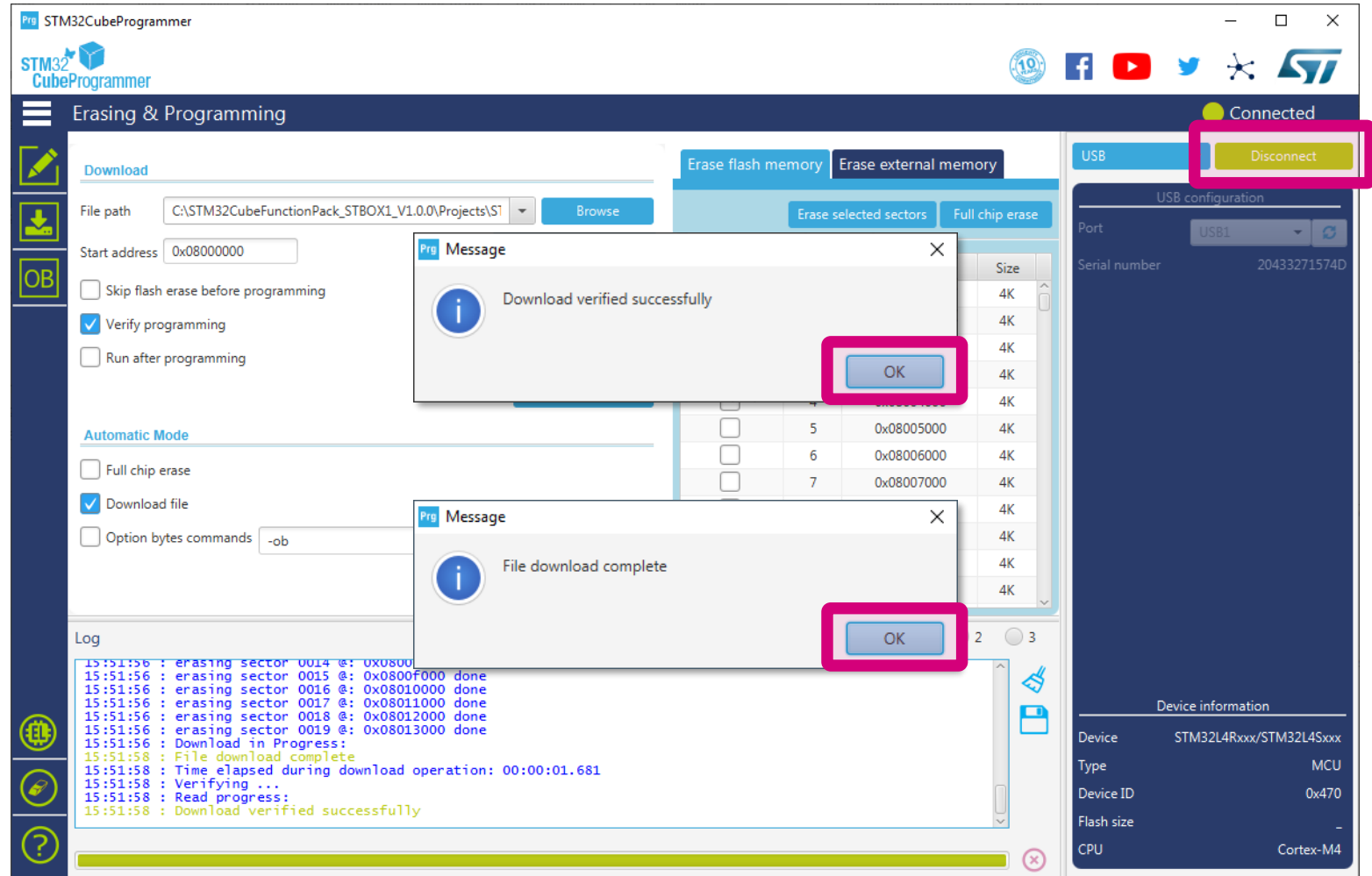
```
Log
15:25:02 : Bank : 0x00
15:25:02 : Address : 0x1ff00000
15:25:02 : Size : 36 Bytes
15:25:02 : Bank : 0x01
15:25:02 : Address : 0x1ff01008
15:25:02 : Size : 28 Bytes
15:25:02 : UPLOADING ...
15:25:02 : Size : 1024 Bytes
15:25:02 : Address : 0x8000000
15:25:02 : Read progress:
15:25:02 : Data read successfully
15:25:02 : Time elapsed during the read operation is: 00:00:00.006
```



Restore the original FW

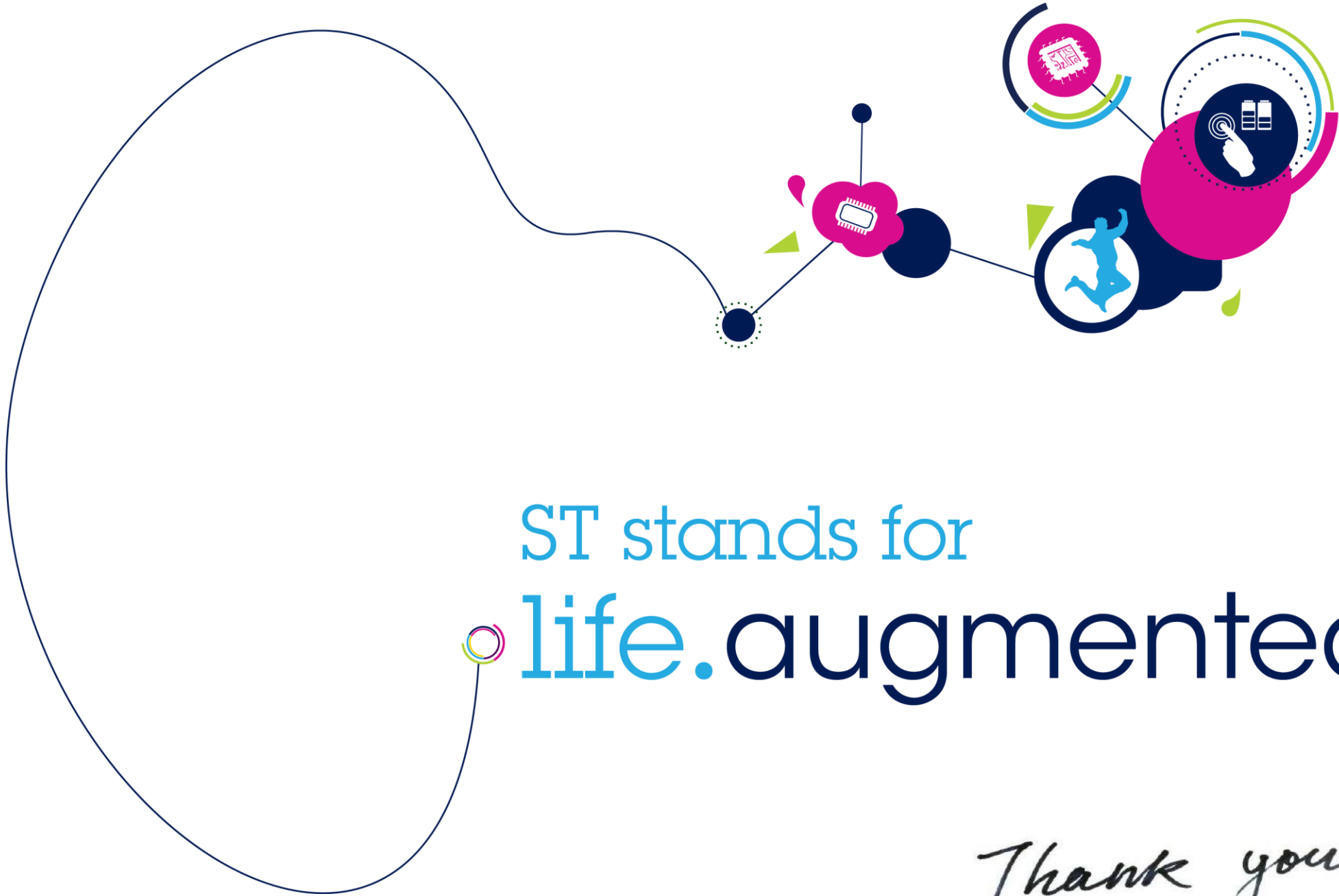
66

1. Wait until **Download verified successfully** appears, then click **OK**
2. Click **OK** on the **File download complete**
3. Click **Disconnect**
4. Unplug and plug the device to reset



Now, your SensorTile.box is flashed with the original firmware it comes with from the ST store!





ST stands for
life.augmented

Thank you!