UM2389

User manual

Getting started with the STM32Cube function pack for IoT node with Dynamic NFC Tag, environmental and motion sensors

Introduction

FP-SNS-SMARTAG1 is an STM32Cube function pack which allows you to read the motion and environmental sensor data on your IoT node via an NFC enabled reader such as a mobile phone or a tablet. The package supports energy harvesting (enabled by NFC) and battery operated use cases.

This software, together with the suggested combination of STM32 and ST devices can be used, for example, to develop tracking, cold chain, medical, smart sensing, and smart home, city and building applications.

The software runs on an ultra-low power STM32L0 microcontroller and includes drivers for the Dynamic NFC tag and for the motion and environmental sensors.

You can register the NFC Sensor Tag node on the DSH-ASSETRACKING web application for asset tracking that stores and monitors on-board sensor data as well as the geolocalization of the smartphone used to read the IoT node data.

RELATED LINKS

Visit the STM32Cube ecosystem web page on www.st.com for further information

1 FP-SNS-SMARTAG1 software description

1.1 Overview

The key features of the FP-SNS-SMARTAG1 package are:

- Complete firmware to access data from an IoT node with dynamic NFC tag, environmental and motion sensors
- Ultra-low power operations, with the support of energy harvesting and battery operated use cases
- Compatible, in single-shot mode only, with the STNFCSensor application for Android/iOS for reading and displaying sensor data
- Compatible with the STAssetTracking application for Android/iOS for reading data logs from the NFC tag and for sending them to the DSH-ASSETRACKING cloud-based dashboard
- Sample implementation available for the STEVAL-SMARTAG1 evaluation board
- Easy portability across different MCU families, thanks to STM32Cube
- Free, user-friendly license terms

This software tracks temperature, pressure, humidity and vibration values in a fixed time range and sends them via NFC. Using an Android or iOS device the logged data can be monitored/displayed.

The software gathers:

- the temperature, humidity, pressure, and vibration sensor for the HTS221, LPS22HB, LIS2DW12 and ST25DV64K devices for the STEVAL-SMARTAG1 evaluation board running on STM32.

This package is compatible with the STAssetTracking Android (Version 2.2.0 or higher)/iOS (Version 2.1.0 or higher) application and, for one-shot mode only, withST NFC Sensor Android/iOS application (Version 1.3.0 or higher) available on GooglePlay/iTunes stores, to read the information sent via NFC/RFID tag IC protocol.

1.2 Architecture

The STM32Cube function packs leverage the modularity and interoperability of STM32 Nucleo and expansion boards (X-NUCLEO), and STM32Cube and expansion software (X-CUBE), to create function examples, embodying some of the most common use cases for each application area.

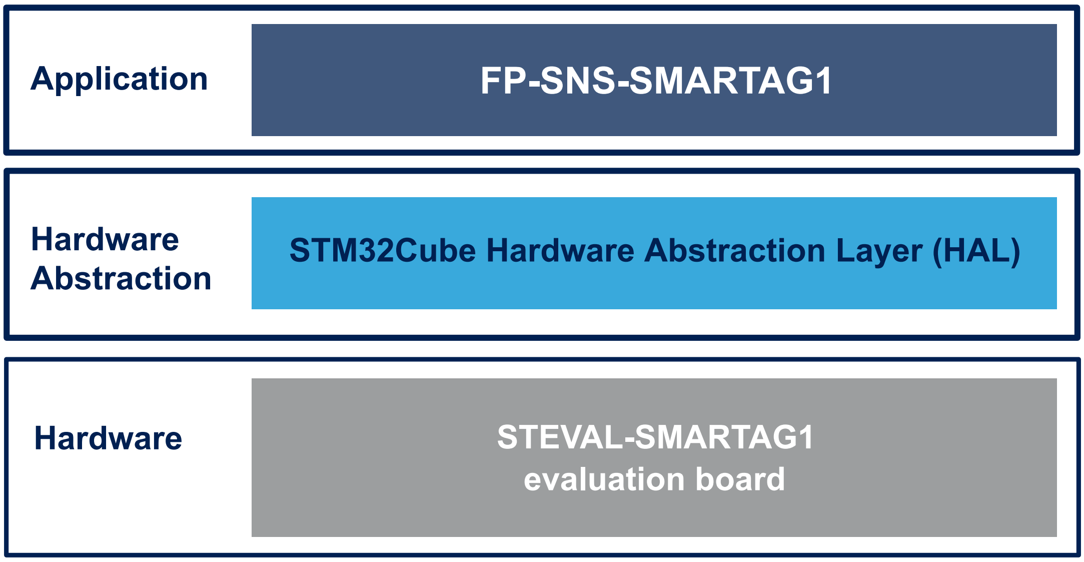
These software function packs are designed to exploit as much as possible the underlying STM32 ODE hardware and software components to best fit the requirements of final users' applications.

Moreover, function packs may include additional libraries and frameworks which do not present the original expansion software packages, thus enabling new functionalities and creating a real and usable system for developers.

To access and use the sensor expansion board, the application software uses:

- **STM32Cube HAL layer**: provides a simple, generic and multi-instance set of generic and extension APIs (application programming interfaces) to interact with the upper layer application, libraries and stacks.
  It is directly based on a generic architecture and allows the layers that are built on it, such as the middleware layer, to implement their functions without requiring the specific hardware configuration for a given microcontroller unit (MCU). This structure improves library code reusability and guarantees easy portability across other devices.
- **Board support package (BSP) layer**: supports the peripherals on the STM32 Nucleo board (except the MCU) with a limited set of APIs providing a programming interface for certain board-specific peripherals like the LED, the user button, etc., and helps determine the specific board version. For the sensor expansion board, it provides the programming interface for various inertial and environmental sensors and support for initializing and reading sensor data.

Figure 1. **FP-SNS-SMARTAG1 software architecture**


Application
FP-SNS-SMARTAG1
Hardware Abstraction
STM32Cube Hardware Abstraction Layer (HAL)
Hardware
STEVAL-SMARTAG1 evaluation board

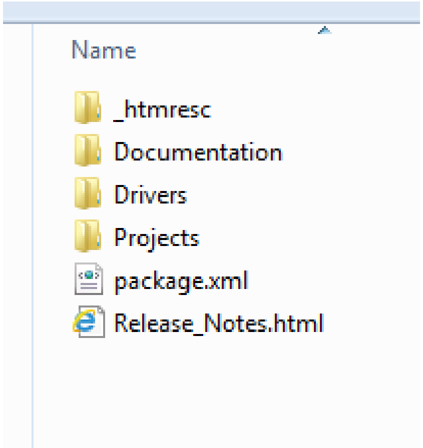
This software is based on the STM32CubeHAL hardware abstraction layer for the STM32 microcontroller. The package extends STM32Cube by providing a board support package (BSP) for the Dynamic NFC/RFID tag IC expansion board based on ST25DV, for the environmental and motion MEMS sensors expansion board, and for the STEVAL-SMARTAG1 evaluation board. The drivers abstract low-level details of the hardware and allow the sample applications to leverage NFC communication and access sensor data in a hardware-independent manner.

The package supports different use cases featuring ultra-low power modes of operation. An advanced, one-shot mode leverages the energy harvesting feature of the dynamic NFC tag to provide enough energy to power the reading of sensor data.

A battery operated data-logger mode enables continuous sensor data reading, which can be viewed on a mobile device using the ST Asset Tracking application (or ST NFC Sensor application for single-shot mode only). Developers can use this package to prototype ultra-low power IoT applications requiring sensor data reading through NFC communication.

1.3 Folder structure

Figure 2. **FP-SNS-SMARTAG1 package folder structure**


Name
_htmresc
Documentation
Drivers
Projects
package.xml
Release_Notes.html

The following folders are included in the software package:

- **Documentation**: contains a compiled HTML file generated from the source code, which details the software components and APIs.

- **Drivers**: contains the HAL drivers, the board-specific drivers for each supported board or hardware platform (including the on-board components), and the CMSIS vendor-independent hardware abstraction layer for the Cortex-M processor series.
- **Projects**: contains a sample application used for reading the motion and environmental sensor data on your IoT node via an NFC enabled reader, such as a mobile phone or a tablet. It is available for the NUCLEO-L053R8 and STEVAL-SMARTAG1 boards and compatible with the IAR Embedded Workbench for ARM, RealView Microcontroller Development Kit (MDK-ARM-STR) and Integrated Development Environment for STM32 (STM32CubeIDE).

1.4 APIs

Detailed user-API technical information with full function and parameter descriptions is available in a compiled HTML file in the package “Documentation” folder.

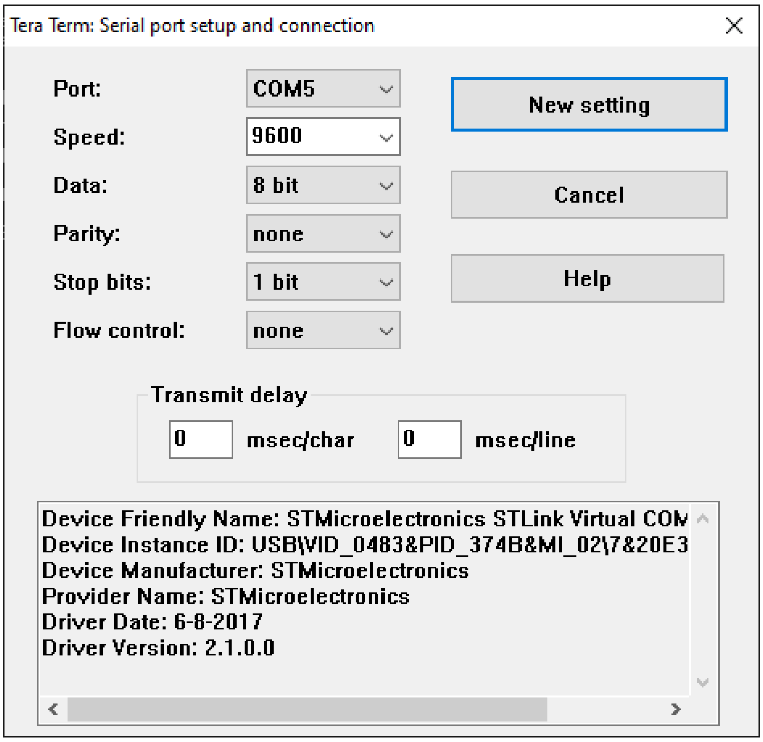
1.5 Sample application description

A sample application is available for the STEVAL-SMARTAG1 evaluation board

Ready to build projects are available for multiple IDEs.

You can set up a terminal window for the appropriate UART communication port in order to control the initialization phase, as shown in the figure below.

Note: *To enable this UART functionality on the STEVAL-SMARTAG1 board, you have to recompile the code by uncommenting the line* `//#define SMARTAG_ENABLE_PRINTF` *in the Projects\STM32L031K6-SmarTag1\Examples\<name>\Inc\SMARTAG1_config.h file.*



Port: COM5
Speed: 9600
Data: 8 bit
Parity: none
Stop bits: 1 bit
Flow control: none

Transmit delay: 0 msec/char 0 msec/line

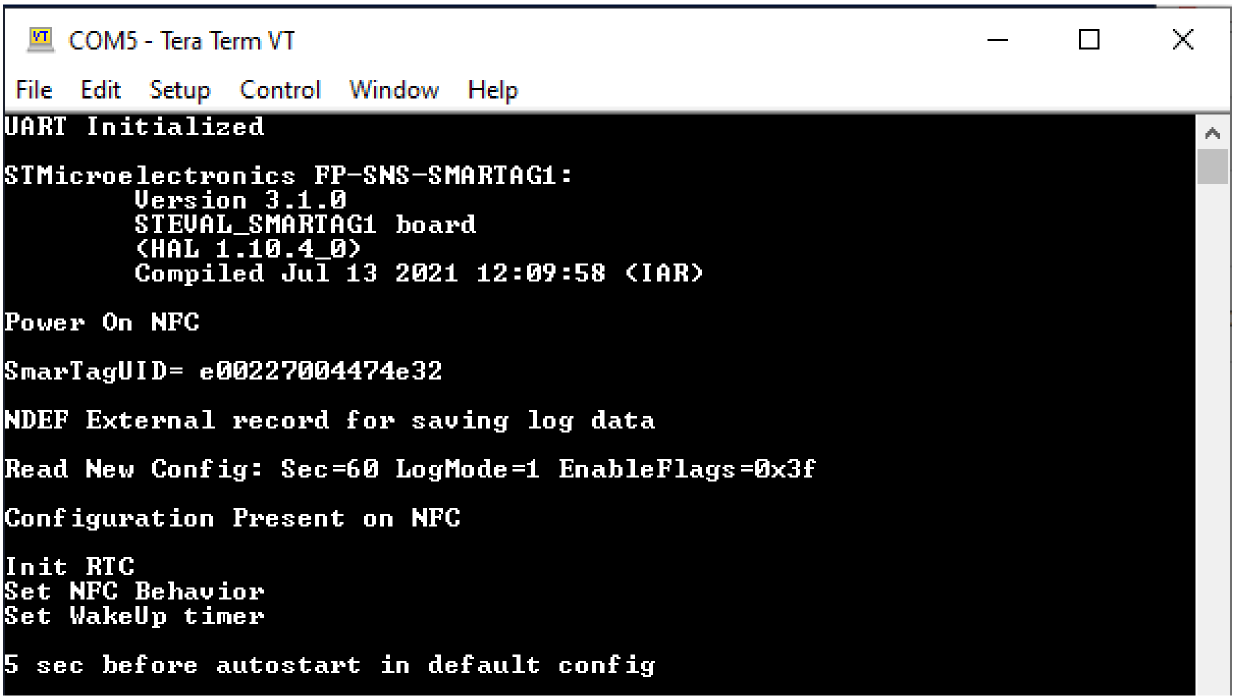
```
Device Friendly Name: STMicroelectronics STLink Virtual COM
Device Instance ID: USB\VID_0483&PID_374B&MI_02\7&20E3
Device Manufacturer: STMicroelectronics
Provider Name: STMicroelectronics
Driver Date: 6-8-2017
Driver Version: 2.1.0.0
```

Figure 3. Tera Term setup

When you first press the reset button, the application:

- starts initializing the UART and I²C interfaces
- shows the SmarTag UID
- shows the default values of the SmarTag config (LogMode, RangeTime, EnableFlags)
- initializes the RTC
- sets the NFC behavior
- sets the wake up timer

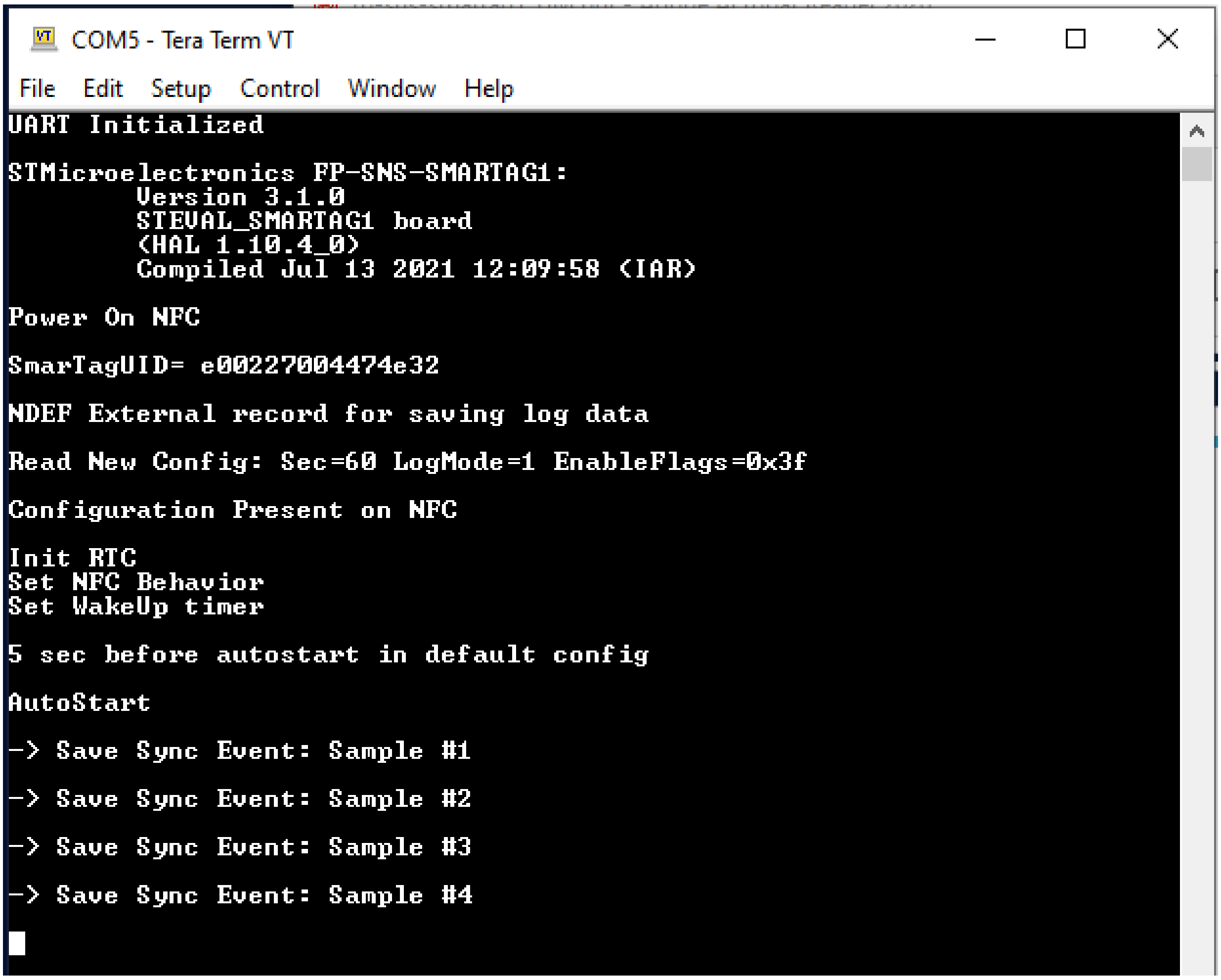
Figure 4. Initialization phase



```
UART Initialized

STMicroelectronics FP-SNS-SMARTAG1:
        Version 3.1.0
        STEVAL_SMARTAG1 board
        (HAL 1.10.4_0)
        Compiled Jul 13 2021 12:09:58 (IAR)

Power On NFC

SmarTagUID= e00227004474e32

NDEF External record for saving log data

Read New Config: Sec=60 LogMode=1 EnableFlags=0x3f

Configuration Present on NFC

Init RTC
Set NFC Behavior
Set WakeUp timer

5 sec before autostart in default config
```

After the auto-start range time, the samples are logged using the default SmarTag config.

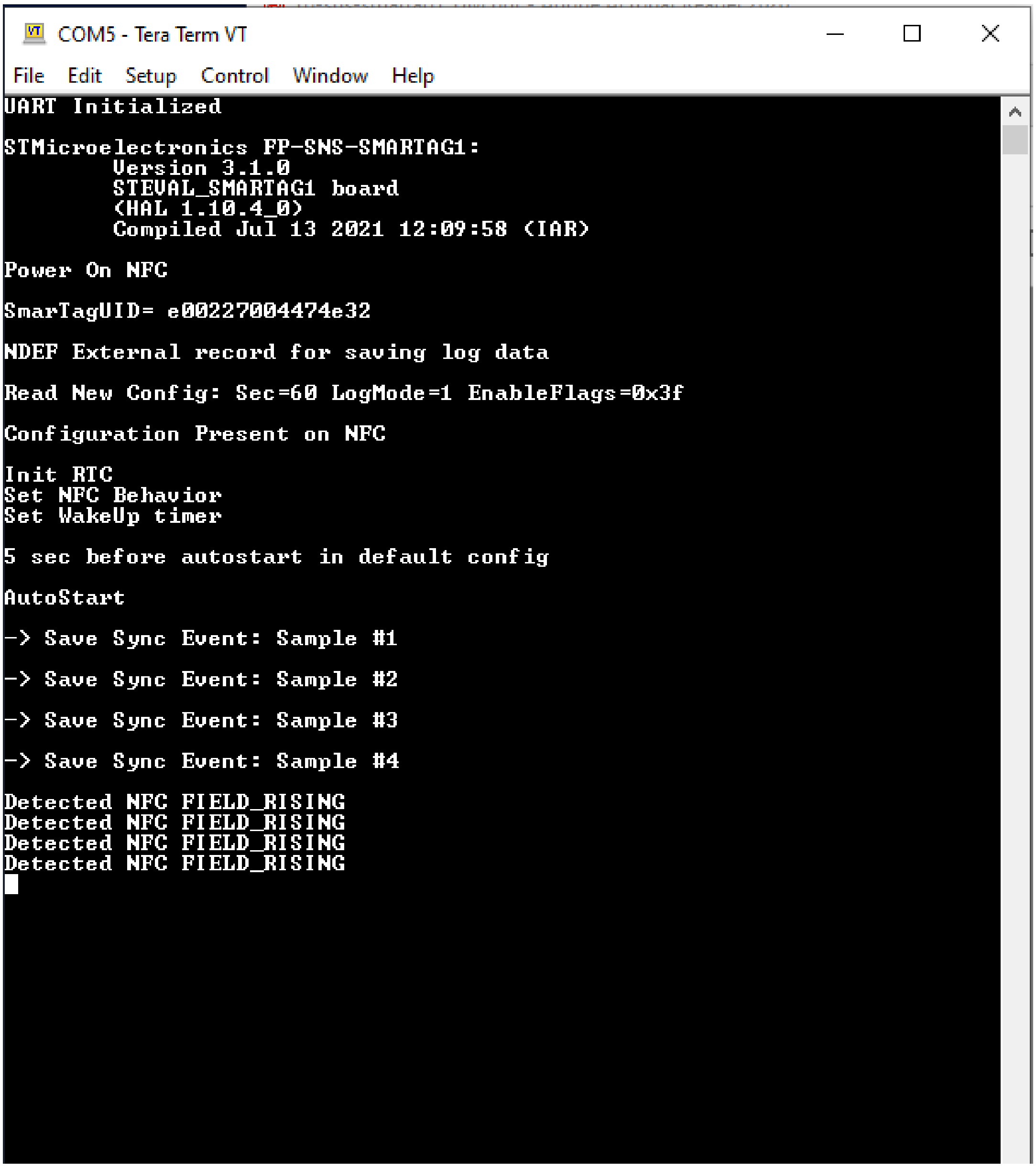
Figure 5. UART console output after auto-start range time



```
UART Initialized

STMicroelectronics FP-SNS-SMARTAG1:
        Version 3.1.0
        STEVAL_SMARTAG1 board
        (HAL 1.10.4_0)
        Compiled Jul 13 2021 12:09:58 (IAR)

Power On NFC

SmarTagUID= e00227004474e32

NDEF External record for saving log data

Read New Config: Sec=60 LogMode=1 EnableFlags=0x3f

Configuration Present on NFC

Init RTC
Set NFC Behavior
Set WakeUp timer

5 sec before autostart in default config

AutoStart

-> Save Sync Event: Sample #1

-> Save Sync Event: Sample #2

-> Save Sync Event: Sample #3

-> Save Sync Event: Sample #4
```

When the smartphone is close to the NFC tag, the message "Detected NFC FIELD_RISING" appears.

Figure 6. UART console output NFC Rising

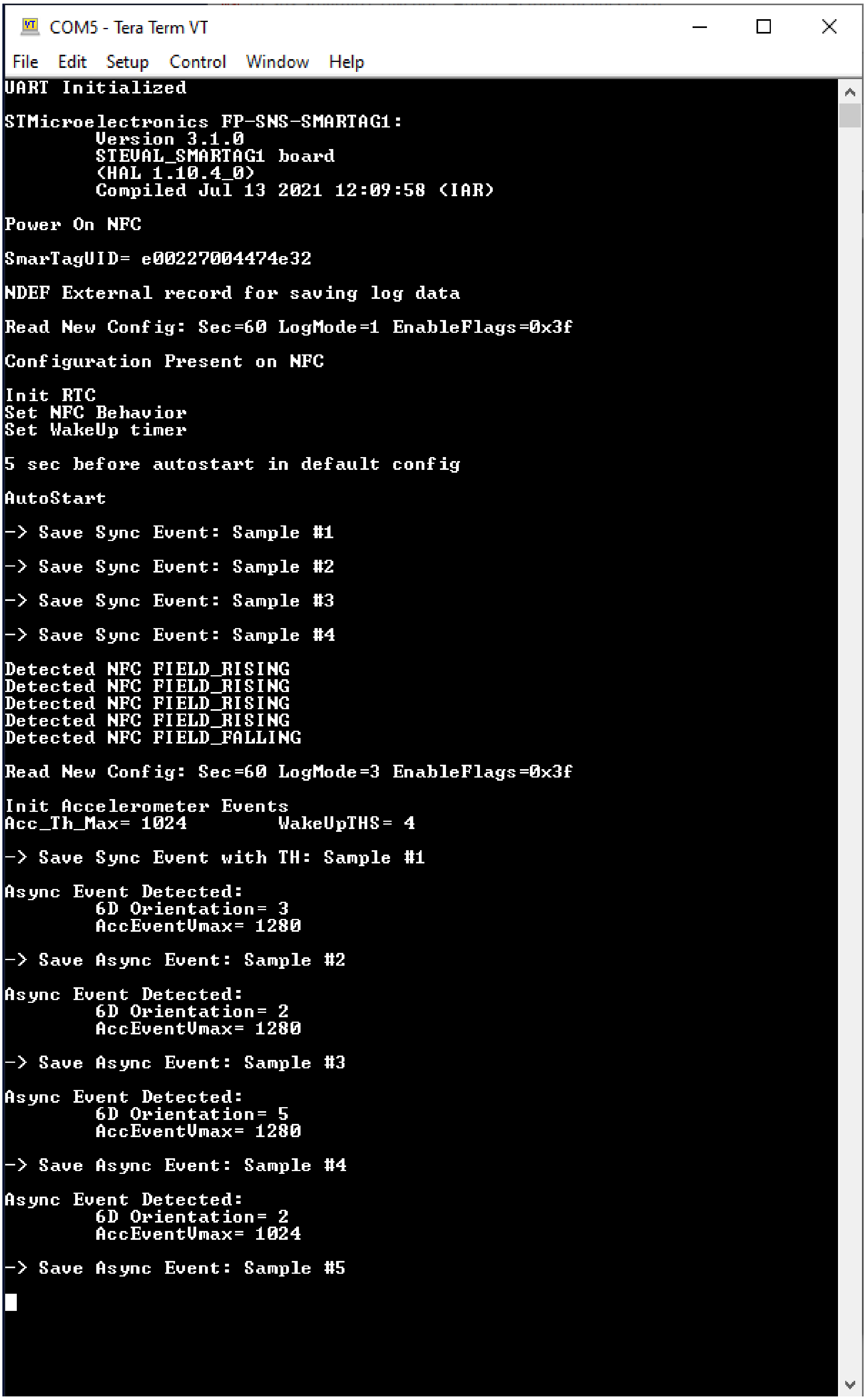


```
UART Initialized

STMicroelectronics FP-SNS-SMARTAG1:
        Version 3.1.0
        STEVAL_SMARTAG1 board
        (HAL 1.10.4_0)
        Compiled Jul 13 2021 12:09:58 (IAR)

Power On NFC

SmarTagUID= e00227004474e32

NDEF External record for saving log data

Read New Config: Sec=60 LogMode=1 EnableFlags=0x3f

Configuration Present on NFC

Init RTC
Set NFC Behavior
Set WakeUp timer

5 sec before autostart in default config

AutoStart

-> Save Sync Event: Sample #1

-> Save Sync Event: Sample #2

-> Save Sync Event: Sample #3

-> Save Sync Event: Sample #4

Detected NFC FIELD_RISING
Detected NFC FIELD_RISING
Detected NFC FIELD_RISING
Detected NFC FIELD_RISING
```

When the smartphone is kept distant from the NFC tag, the message "Detected NFC FIELD FALLING" appears together with the new config if a new one is detected.

In particular, if the new logging mode has a threshold, the messages also highlight the enabled accelerometer events.

Figure 7. UART console output NFC Falling



```
UART Initialized

STMicroelectronics FP-SNS-SMARTAG1:
        Version 3.1.0
        STEVAL_SMARTAG1 board
        (HAL 1.10.4_0)
        Compiled Jul 13 2021 12:09:58 (IAR)

Power On NFC

SmarTagUID= e00227004474e32

NDEF External record for saving log data

Read New Config: Sec=60 LogMode=1 EnableFlags=0x3f

Configuration Present on NFC

Init RTC
Set NFC Behavior
Set WakeUp timer

5 sec before autostart in default config

AutoStart

-> Save Sync Event: Sample #1

-> Save Sync Event: Sample #2

-> Save Sync Event: Sample #3

-> Save Sync Event: Sample #4

Detected NFC FIELD_RISING
Detected NFC FIELD_RISING
Detected NFC FIELD_RISING
Detected NFC FIELD_RISING
Detected NFC FIELD_FALLING

Read New Config: Sec=60 LogMode=3 EnableFlags=0x3f

Init Accelerometer Events
Acc_Th_Max= 1024        WakeUpTHS= 4

-> Save Sync Event with TH: Sample #1

Async Event Detected:
        6D Orientation= 3
        AccEventVmax= 1280

-> Save Async Event: Sample #2

Async Event Detected:
        6D Orientation= 2
        AccEventVmax= 1280

-> Save Async Event: Sample #3

Async Event Detected:
        6D Orientation= 5
        AccEventVmax= 1280

-> Save Async Event: Sample #4

Async Event Detected:
        6D Orientation= 2
        AccEventVmax= 1024

-> Save Async Event: Sample #5
```

1.6 Android and iOS sample client application

The FP-SNS-SMARTAG1 software for STM32Cube is compatible with the ST Asset Tracking Android (Version 2.2.0 or higher)/iOS (Version 2.1.0 or higher) application and, for one-shot mode only, ST NFC Sensor Android/iOS application (Version 1.3.0 or higher) available at their respective GooglePlay/iOS store.

The next sections show some use cases enabled by the Android application.

1.6.1 Settings

When the smartphone is close to the NFC tag, the application is opened as shown in Figure 8. ST Asset Tracking (Android version) setting page 1.

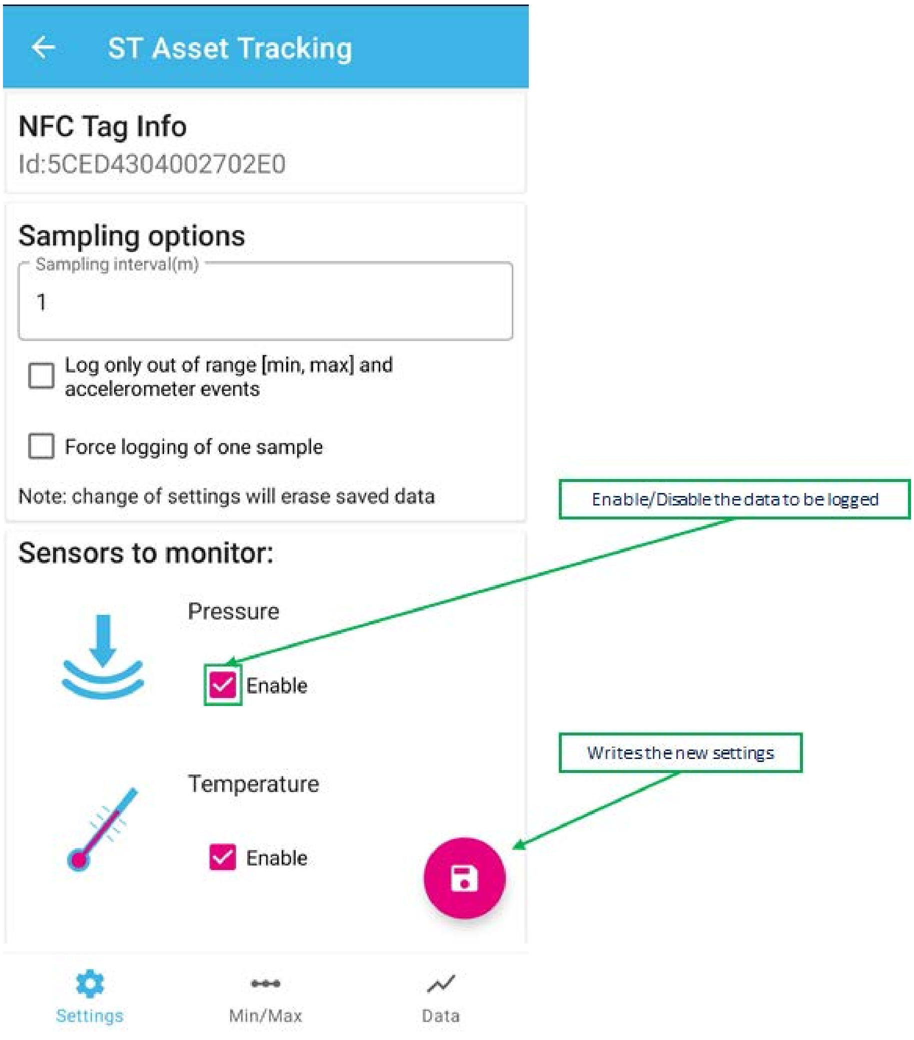
From this page, you can select the data to be logged (pressure, temperature, humidity and vibration) as well as the time interval.

After saving the settings, when the smartphone is distant from the NFC tag, the logging starts in default mode.

You can choose other two different logging modes:

1. **Log only out of range [min, max] and accelerometer events**:
   a. sync events: the selected data are logged only if a minimum or maximum threshold value is reached.
   b. async events: logging only if a wakeup or change of orientation events occur (if selected, as shown in Figure 9. ST Asset Tracking (Android version) setting page 2)
2. **Force logging of one sample**: the current value of the selected data is logged; then, the data logging restarts in the previous mode.

Figure 8. ST Asset Tracking (Android version) setting page 1



ST Asset Tracking

NFC Tag Info
Id:5CED4304002702E0

Sampling options
Sampling interval(m)
1

□ Log only out of range [min, max] and accelerometer events

□ Force logging of one sample

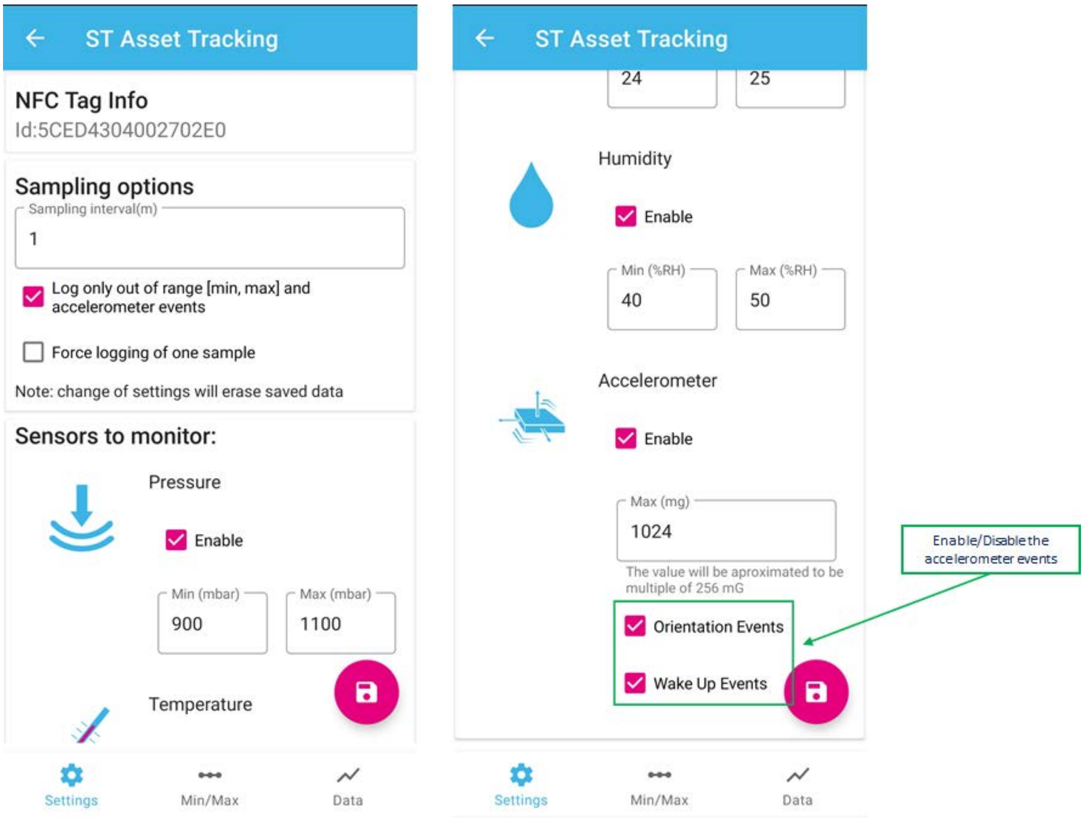
Note: change of settings will erase saved data

Sensors to monitor:

Pressure
☑ Enable

Temperature
☑ Enable

Figure 9. ST Asset Tracking (Android version) setting page 2



ST Asset Tracking

NFC Tag Info
Id:5CED4304002702E0

Sampling options

Sampling interval(m): 1

☑ Log only out of range [min, max] and accelerometer events

☐ Force logging of one sample

Note: change of settings will erase saved data

Sensors to monitor:

Pressure

☑ Enable

Min (mbar): 900

Max (mbar): 1100

Temperature

ST Asset Tracking

24

25

Humidity

☑ Enable

Min (%RH): 40

Max (%RH): 50

Accelerometer

☑ Enable

Max (mg): 1024

The value will be aproximated to be multiple of 256 mG

☑ Orientation Events

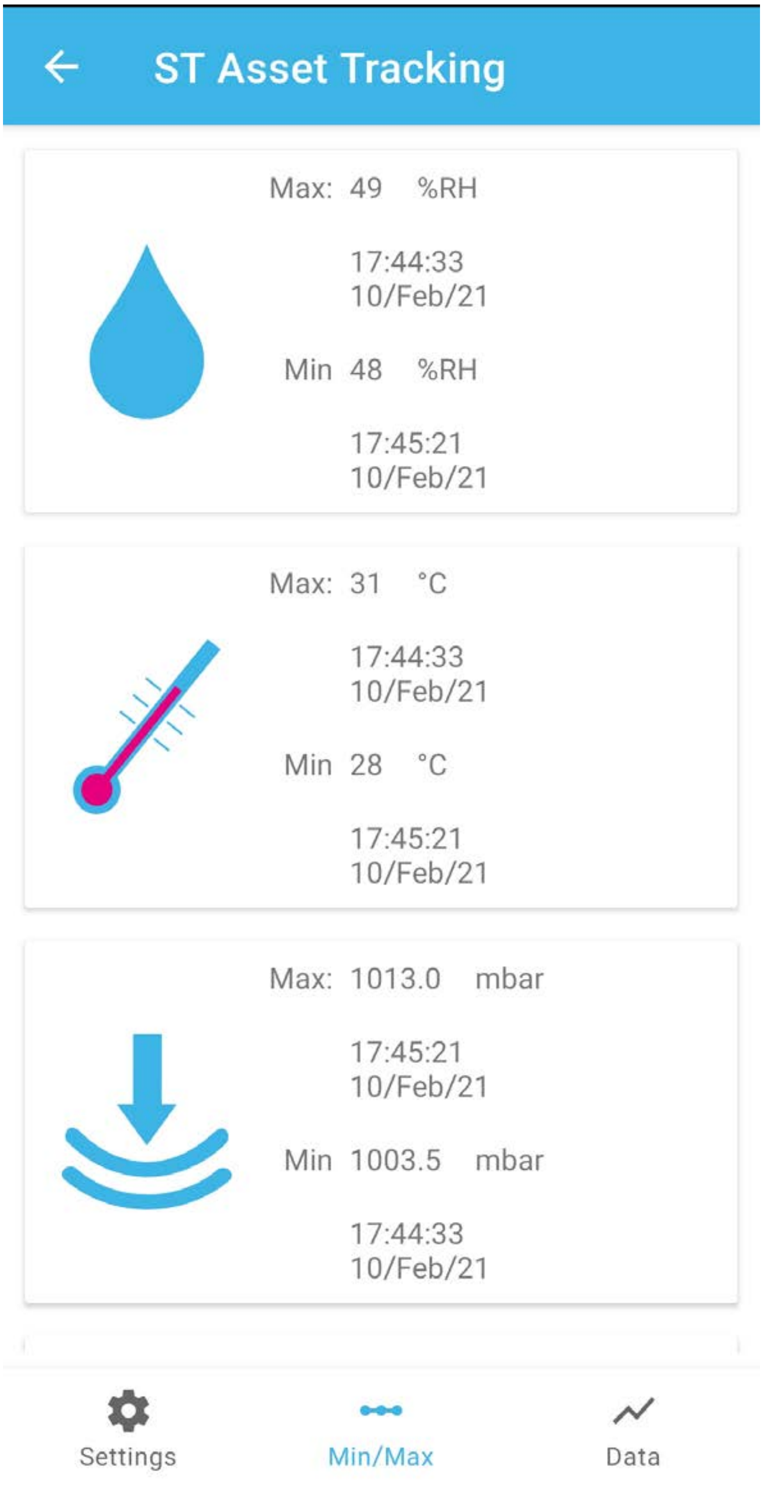
☑ Wake Up Events

Enable/Disable the accelerometer events

1.6.2 Min./Max.

This page shows the maximum and minimum values obtained during the logging of the selected data.

Figure 10. ST Asset Tracking (Android version) maximum and minimum value page



ST Asset Tracking

Max: 49 %RH

17:44:33
10/Feb/21

Min 48 %RH

17:45:21
10/Feb/21

Max: 31 °C

17:44:33
10/Feb/21

Min 28 °C

17:45:21
10/Feb/21

Max: 1013.0 mbar

17:45:21
10/Feb/21

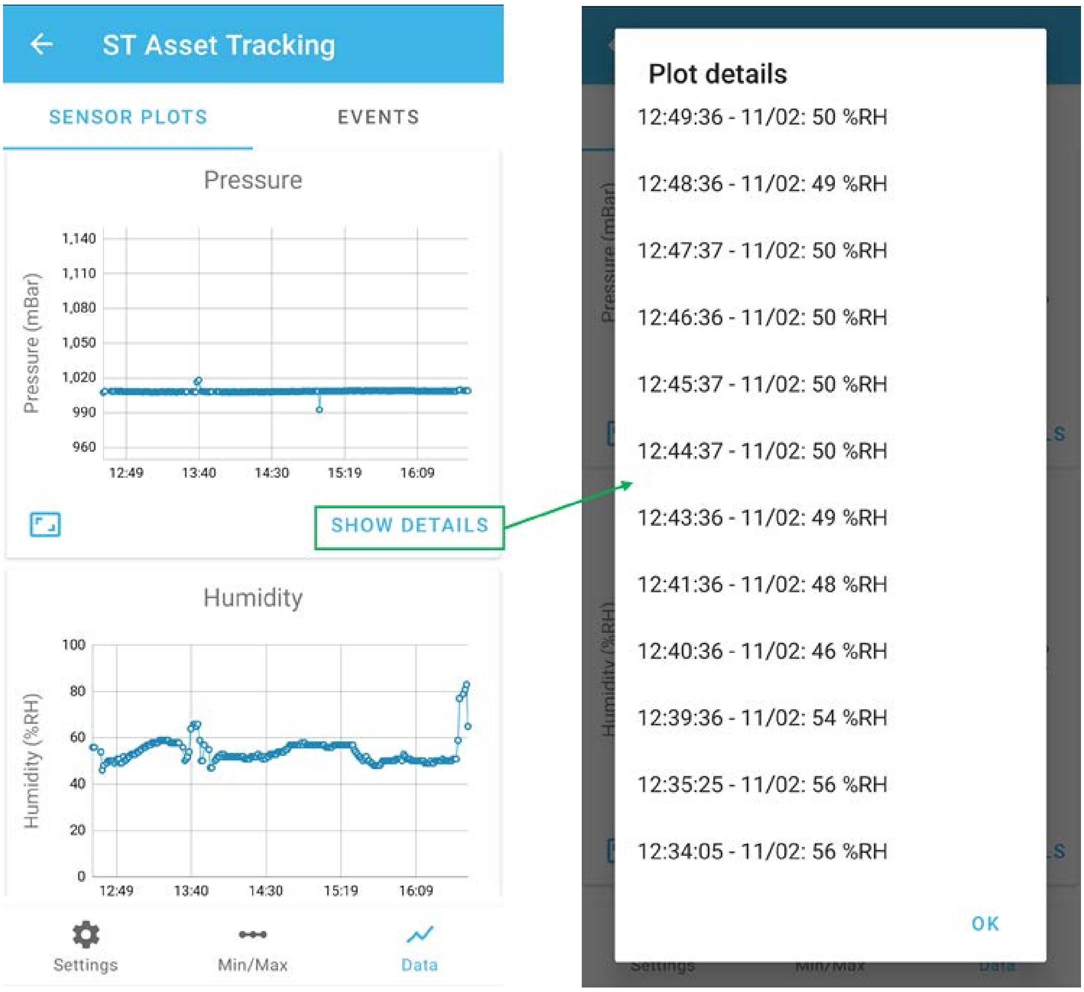
Min 1003.5 mbar

17:44:33
10/Feb/21

1.6.3 Data

This page shows the plots of the data selected for the logging, when the sensor plot feature is selected.

Figure 11. ST Asset Tracking (Android version) chart sensor plot page



12:49:36 - 11/02: 50 %RH

12:48:36 - 11/02: 49 %RH

12:47:37 - 11/02: 50 %RH

12:46:36 - 11/02: 50 %RH

12:45:37 - 11/02: 50 %RH

12:44:37 - 11/02: 50 %RH

12:43:36 - 11/02: 49 %RH

12:41:36 - 11/02: 48 %RH

12:40:36 - 11/02: 46 %RH

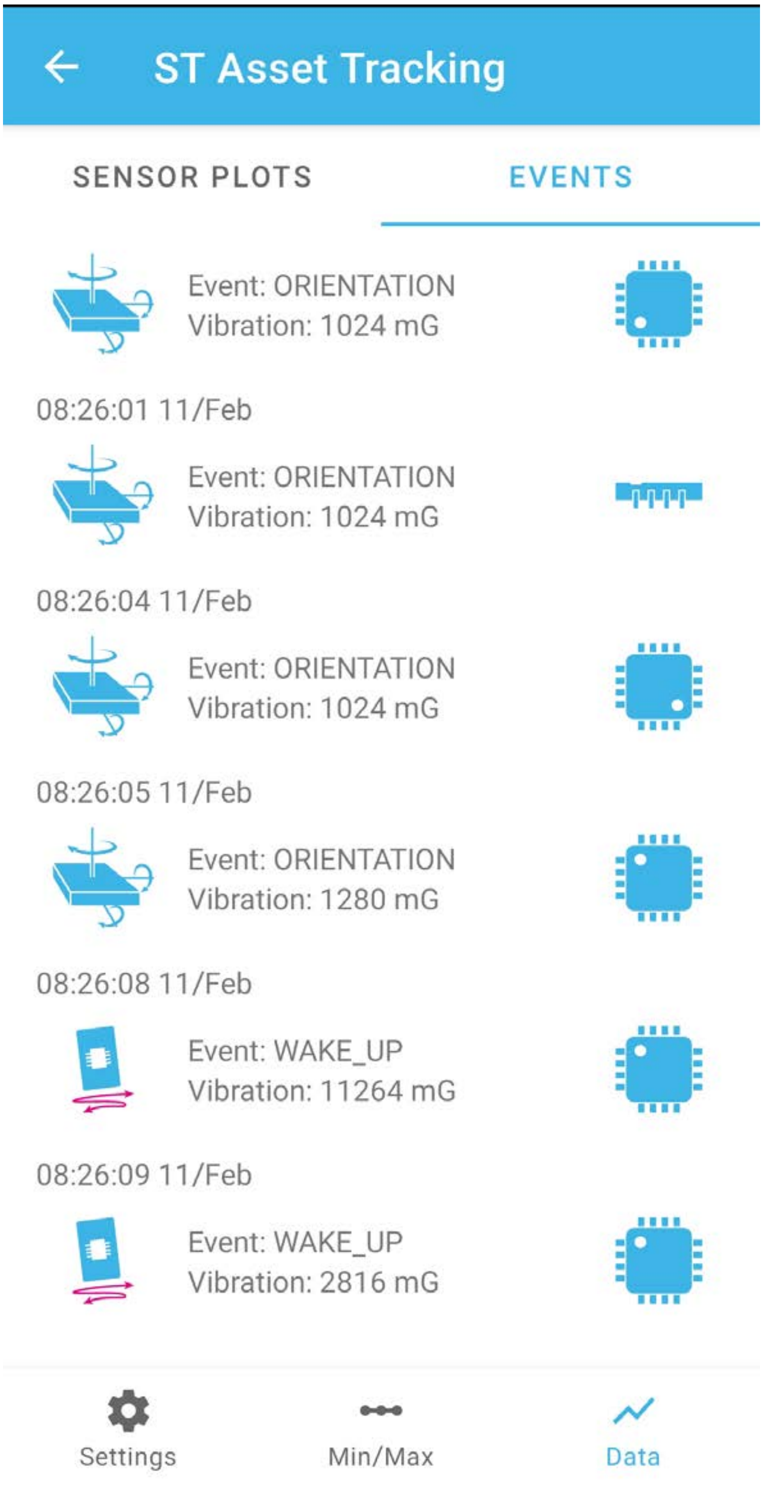
12:39:36 - 11/02: 54 %RH

12:35:25 - 11/02: 56 %RH

12:34:05 - 11/02: 56 %RH

If the "Log with Threshold" has been chosen and Async is selected, the accelerometer events are shown.

Figure 12. ST Asset Tracking (Android version) chart async page



ST Asset Tracking

SENSOR PLOTS EVENTS

Event: ORIENTATION
Vibration: 1024 mG

08:26:01 11/Feb

Event: ORIENTATION
Vibration: 1024 mG

08:26:04 11/Feb

Event: ORIENTATION
Vibration: 1024 mG

08:26:05 11/Feb

Event: ORIENTATION
Vibration: 1280 mG

08:26:08 11/Feb

Event: WAKE_UP
Vibration: 11264 mG

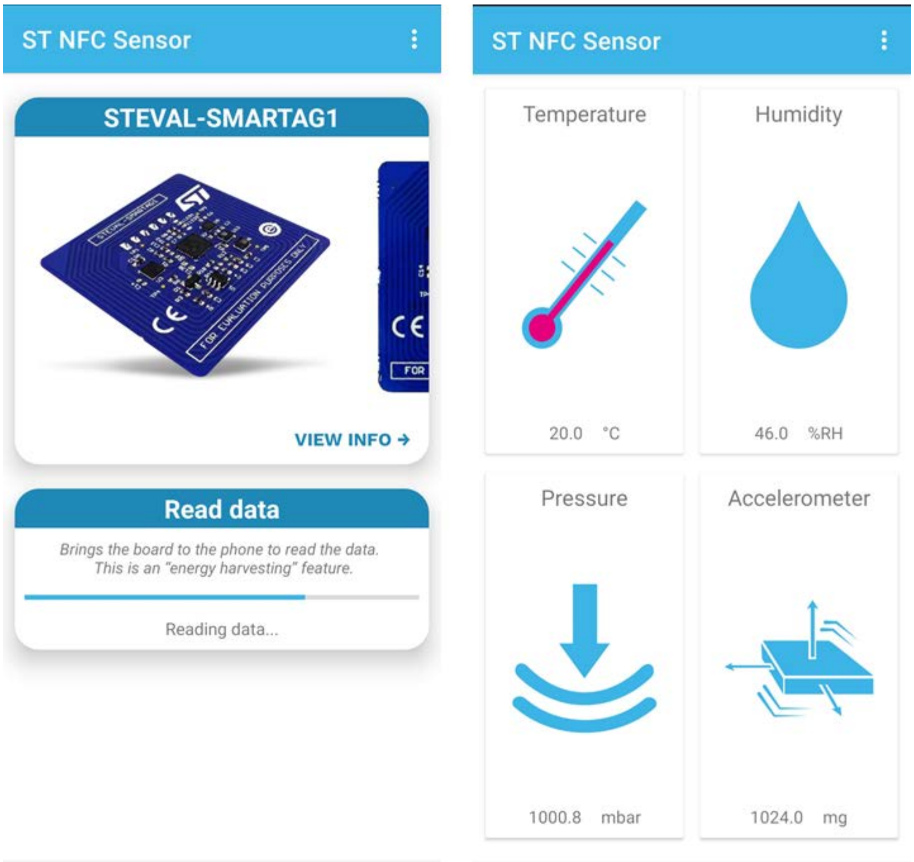
08:26:09 11/Feb

Event: WAKE_UP
Vibration: 2816 mG

1.6.4 Single shot

In case the battery is not inserted, ST NFC Sensor application reads and display the current value of the data in energy harvesting mode from the tag.

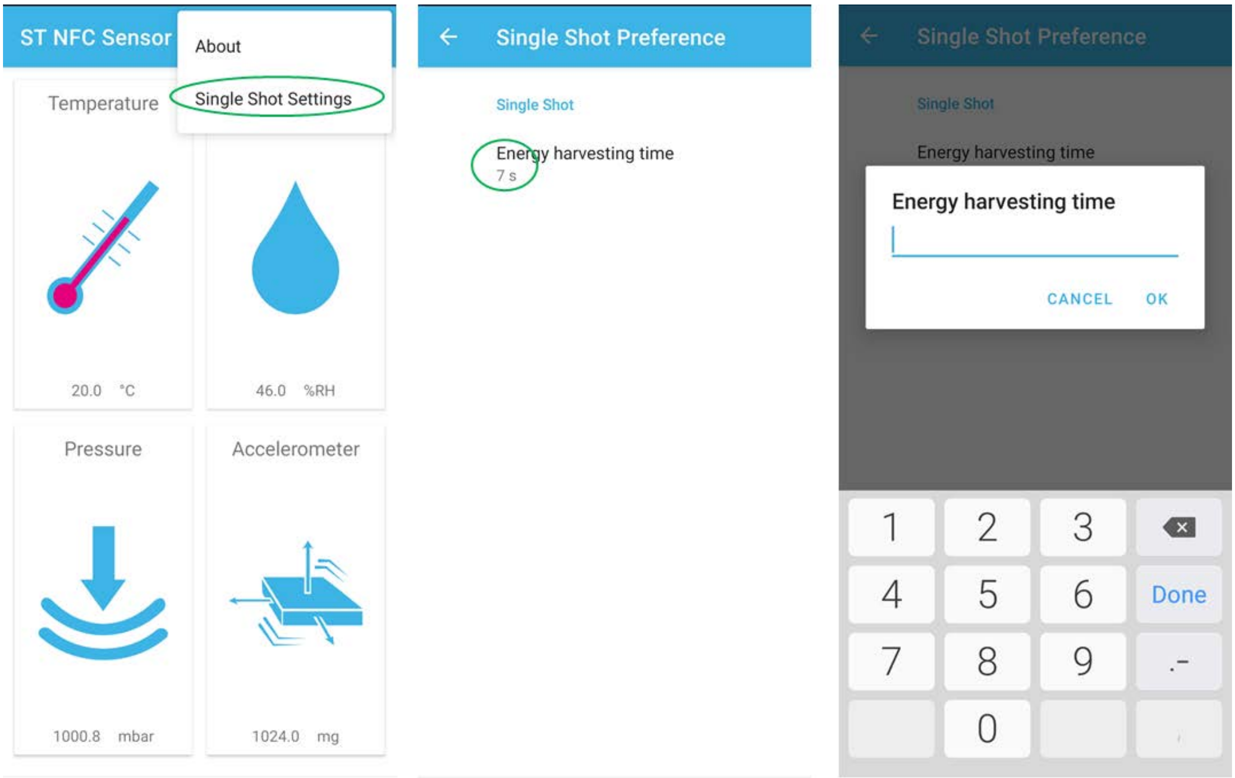
Figure 13. ST NFC Sensor (Android version) single shot page



Brings the board to the phone to read the data. This is an "energy harvesting" feature.

The energy harvesting time can be changed by setting the single shot.

Figure 14. ST NFC Sensor (Android version) single shot - energy harvesting time setting



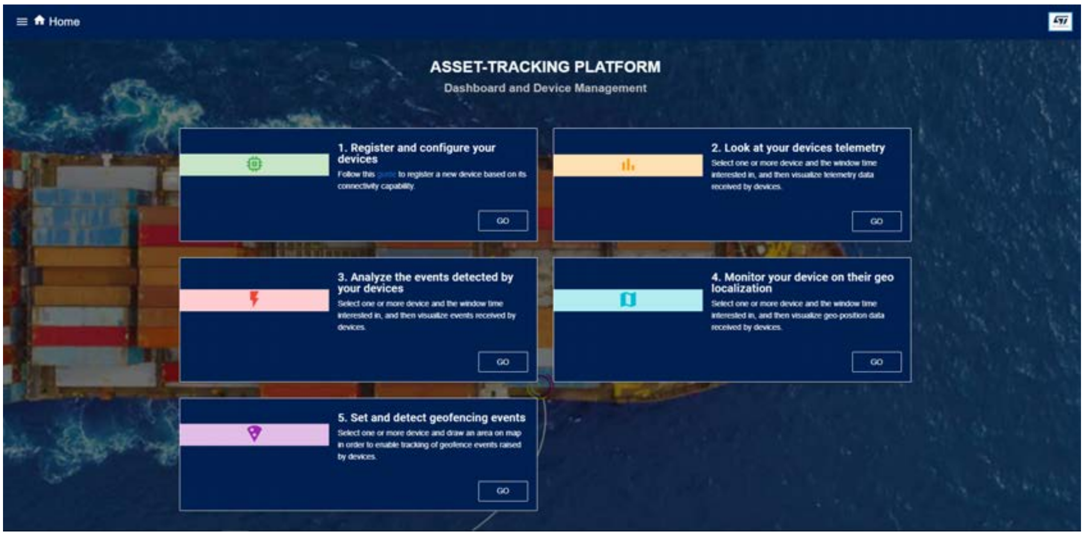
Energy harvesting time
7 s

Energy harvesting time

2 Using the Asset Tracking web dashboard

Step 1. Go to DSH asset tracking web dashoboard.

Figure 15. **DSH-ASSETRACKING dashboard homepage**



ASSET-TRACKING PLATFORM

Dashboard and Device Management

1. Register and configure your devices
Follow this guide to register a new device based on its connectivity capability.

2. Look at your devices telemetry
Select one or more device and the window time interested in, and then visualize telemetry data received by devices.

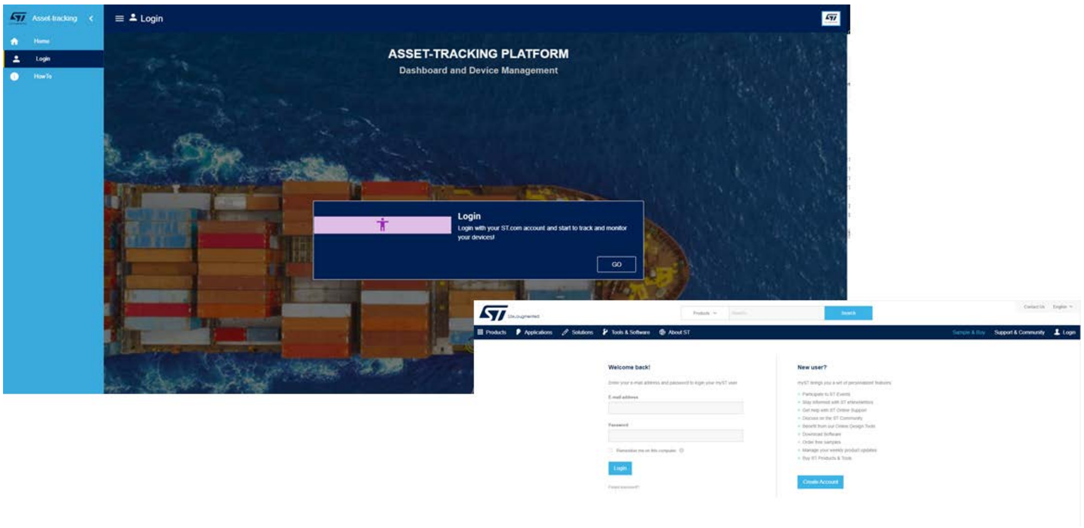
3. Analyze the events detected by your devices
Select one or more device and the window time interested in, and then visualize events received by devices.

4. Monitor your device on their geo localization
Select one or more device and the window time interested in, and then visualize geo-position data received by devices.

5. Set and detect geofencing events
Select one or more device and draw an area on map in order to enable tracking of geofence events raised by devices.

Step 2. Select login, click [**GO**] button and provide your username and password.

Figure 16. **DSH-ASSETRACKING dashboard - login**



ASSET-TRACKING PLATFORM

Dashboard and Device Management

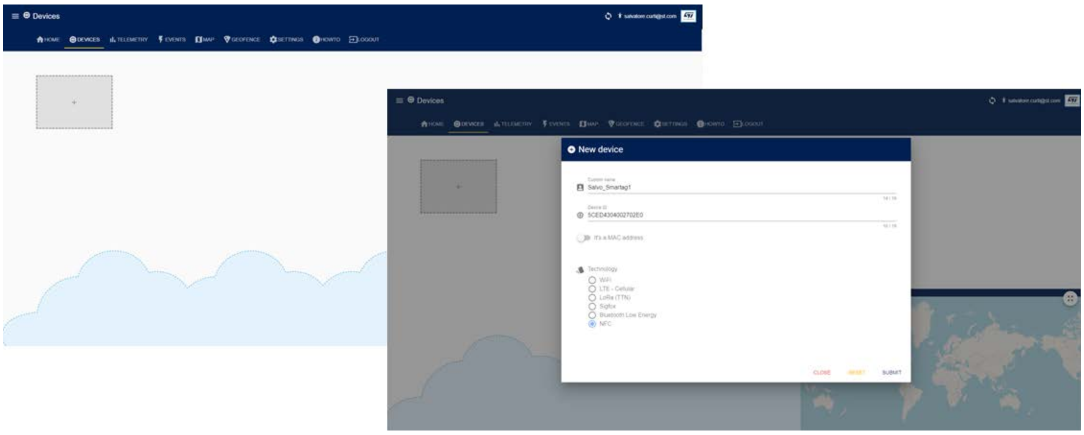
Login
Login with your ST.com account and start to track and monitor your devices!

Welcome back!
Enter your e-mail address and password to login your myST user
E-mail address
Password
□ Remember me on this computer
Forgot password?

New user?
myST brings you a set of personalized features
- Participate to ST Events
- Stay informed with ST eNewsletters
- Get help with ST Online Support
- Discuss on the ST Community
- Benefit from our Online Design Tools
- Download Software
- Order free samples
- Manage your weekly product updates
- Buy ST Products & Tools

Step 3. Add the new device.

Figure 17. DSH-ASSETRACKING dashboard - adding device



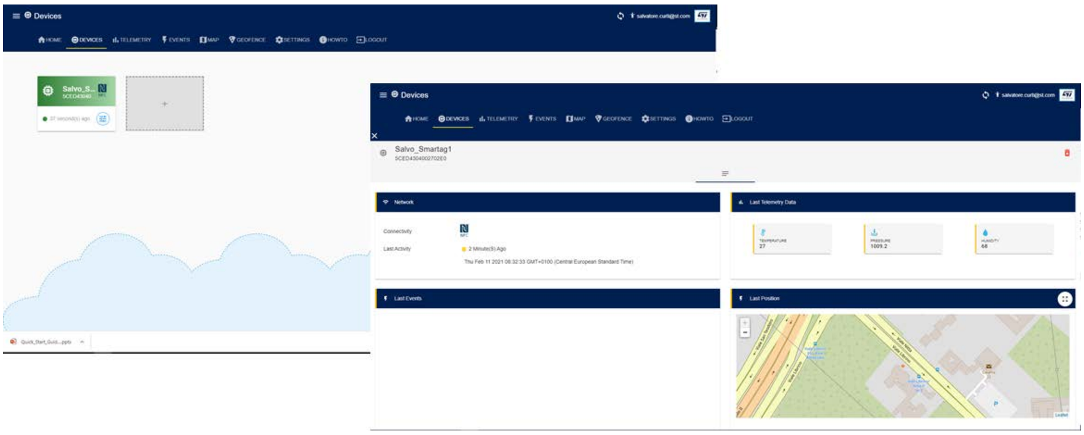
Custom name: Salvo_Smartag1

Device ID: 5CED4304002702E0

○ It's a MAC address

Technology
- ○ WiFi
- ○ LTE - Cellular
- ○ LoRa (TTN)
- ○ Sigfox
- ○ Bluetooth Low Energy
- ● NFC

Step 4. Select the device you want to monitor.

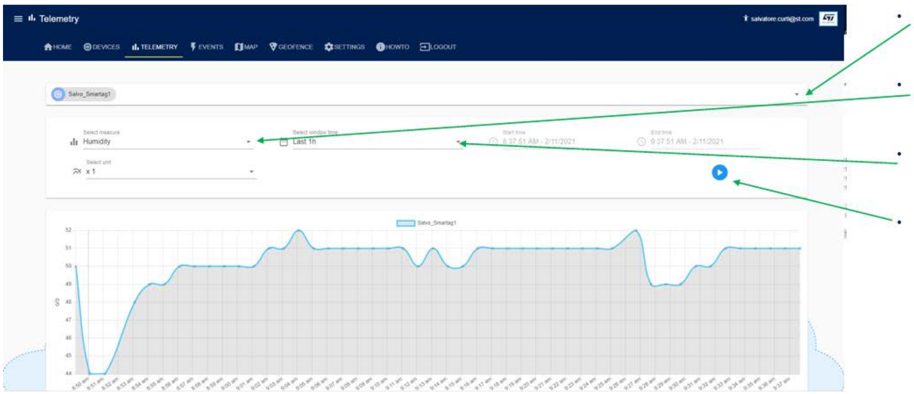


Salvo_Smartag1
5CED4304002702E0

Connectivity: NFC
Last Activity: 2 Minute(S) Ago
Thu Feb 11 2021 08:32:33 GMT+0100 (Central European Standard Time)

TEMPERATURE 27
PRESSURE 1009.2
HUMIDITY 68

Figure 18. **DSH-ASSETRACKING dashboard - selecting the device**


Telemetry
HOME DEVICES TELEMETRY EVENTS MAP GEOFENCE SETTINGS HOWTO LOGOUT
salvatore.curti@st.com
Salvo_Smartag1
52
51
50
49
48
47
46
45
44

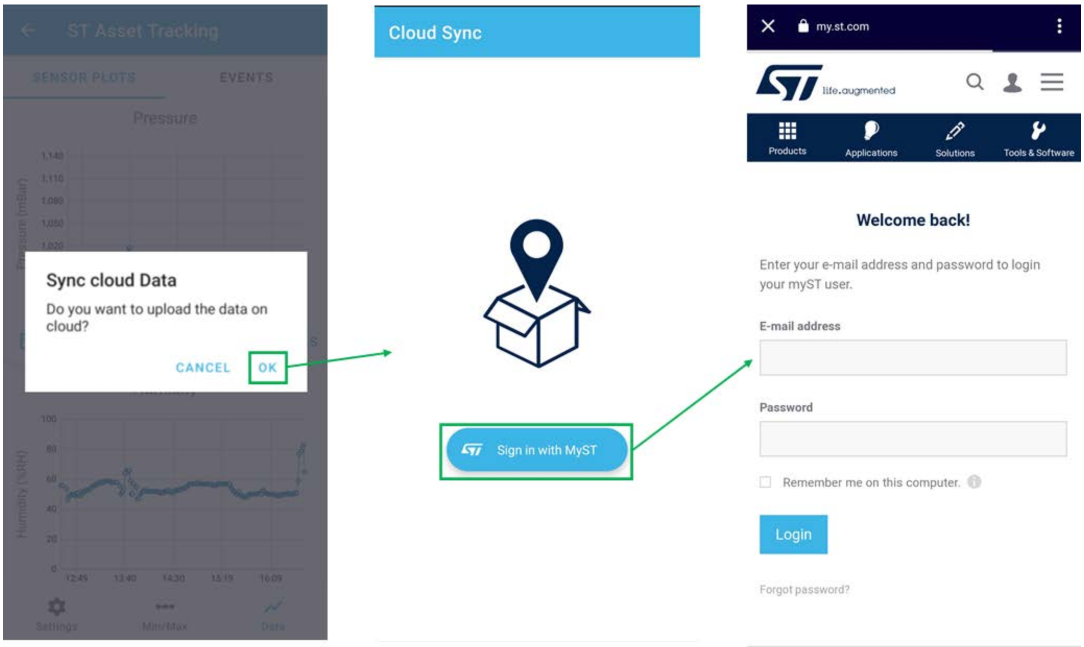
Salvo_Smartag1
Select measure: Humidity
Select window time: Last 1h
Start time: 8:37:51 AM - 2/11/2021
End time: 9:37:51 AM - 2/11/2021
Select unit: x 1

- Select the device you want to monitor.
- Select desired telemetry data to display
- Choose the amount of data to display
- Select Play Button

Figure 19. **DSH-ASSETRACKING dashboard - selecting telemetry**

To upload data on the cloud, you can also register the device from the ST Asset Tracking application.

Figure 20. Uploading data the cloud through ST Asset Tracking application



Sync cloud Data

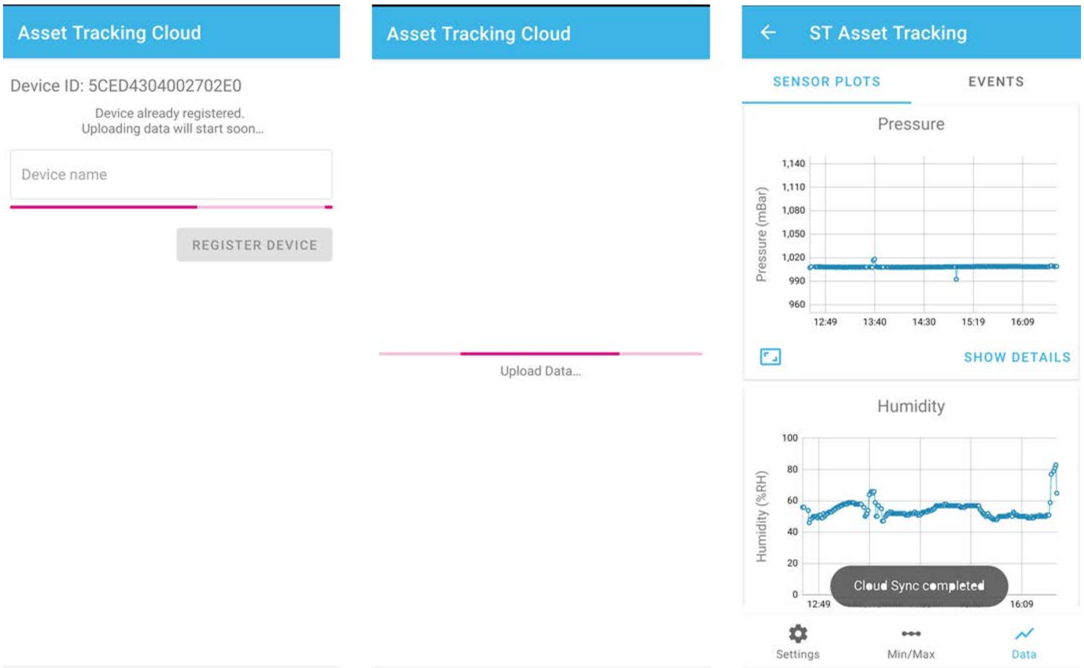
Do you want to upload the data on cloud?

Welcome back!

Enter your e-mail address and password to login your myST user.

□ Remember me on this computer.

Figure 21. ST Asset Tracking application - data uploaded to the cloud and related plots



Device ID: 5CED4304002702E0

Device already registered.
Uploading data will start soon...

Upload Data...

Cloud Sync completed

RELATED LINKS

Visit the DSH-ASSETRACKING homepage on www.st.com for further information

3 System setup guide

3.1 Hardware description

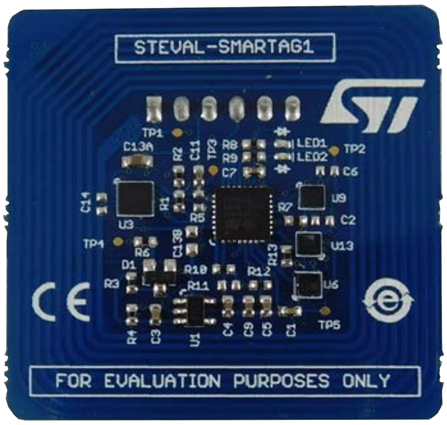
3.1.1 STEVAL-SMARTAG1 evaluation board

This smart and flexible NFC Tracker evaluation board with sensors includes a comprehensive software library and a sample application to monitor and log sensor data over NFC from an Android or iOS device.

The ultra-low power sensor node evaluation board mounts an ST25DV NFC Tag, an STM32L0 ARM Cortex M0+, environment sensors (temperature, humidity and pressure) and motion (accelerometer) sensor.

The evaluation board features NFC harvesting to supply power and a battery cradle for a CR2032 battery.

Figure 22. **STEVAL-SMARTAG1 evaluation board**



The STEVAL-SMARTAG1 evalaution board key features are:

- ST25DV64K dynamic NFC tag solution based on 64K-bit (8K-Byte) EEPROM and with I^2C interface, Fast Transfer Mode and Energy Harvesting features
- STM32L031K6 ultra-low-power ARM Cortex-M0+ MCU running at 32 MHz with 32-Kbytes Flash and 8-Kbytes RAM
- LIS2DW12 ultra-low-power high-performance three-axis linear accelerometer
- LPS22HB ultra-compact piezo-resistive absolute pressure sensor which functions as a digital output barometer: 260-1260 hPa
- HTS221 capacitive digital sensor for relative humidity and temperature
- STLQ015 low drop linear regulator power management
- CR2032 Battery powered (not included)
- STM32Cube function pack (FP-SNS-SMARTAG1)
- Android (Google Play) and iOS demo apps (ST Asset Tracking and ST NFC Sensor)
- Suitable for the following applications:
  - Internet of Things
  - Supply Chain and Cold-Chain Management
  - Smart building, home and city
  - Retail and apparel
  - Smart packaging
  - Medical and pharmaceutical
  - Batteryless sensing
  - Smart agriculture (soil control, animal tracking, etc.)

3.2 Software description

The following software components are needed to set up a suitable development environment to create applications for the STM32 Nucleo equipped with the sensors and NFC tag expansion boards and for the STEVAL-SMARTAG1 evaluation board:

- FP-SNS-SMARTAG1: complete firmware to access data from an IoT node with dynamic NFC tag, environmental and motion sensors. The package provides easy portability across different MCU families, thanks to STM32Cube. The FP-SNS-SMARTAG1 firmware and related documentation is available on www.st.com.
- Development tool-chain and compiler. The STM32Cube expansion software supports the following environments:
  - IAR Embedded Workbench for ARM® toolchain + ST-LINK
  - RealView Microcontroller Development Kit (MDK-ARM-STR) toolchain + ST-LINK
  - Integrated Development Environment for STM32 (STM32CubeIDE) + ST-LINK

3.3 Hardware and software setup

3.3.1 Hardware setup

The following hardware components are required:

- One STEVAL-SMARTAG1 evaluation board
- One ST-LINK/V2-1 debugger/programmer integrated in the STM32 Nucleo board
- One USB type A to Mini-B USB cable to connect the STM32 Nucleo to the PC
- One CR 2032 battery

3.3.2 Software setup

This section describes how to set up different hardware components before writing and executing an application on the STEVAL-SMARTAG1 evaluation board.

3.3.2.1 Development tool-chains and compilers

Select one of the Integrated Development Environments supported by the STM32Cube expansion software and follow the system requirements and setup information provided by the selected IDE provider.

3.3.3 System setup guide

3.3.3.1 STEVAL-SMARTAG1 evaluation board setup

The ST-LINK/V2-1 debugger/programmer integrated in the STM32 Nucleo board must be used to program the STEVAL-SMARTAG1 evaluation board. The developer can download the relevant version of the ST-LINK/V2-1 USB driver by clicking STSW-LINK008 or STSW-LINK009.

Step 1. Connect the STEVAL-SMARTAG1 to the ST-LINK/V2-1 debugger/programmer integrated on the STM32 Nucleo board via the SWD connector to start programming.

Step 2. Connect your STM32 Nucleo board to the STEVAL-SMARTAG1, after ensuring CN2 jumpers are OFF.

Important:
Pay attention to the polarity of the connectors: pin 1 can be identified by a little circle on the PCB silkscreen (on the STM32 Nucleo and the STEVAL-SMARTAG1 boards).

Figure 23. STEVAL-SMARTAG1 evaluation board connected to ST-LINK/V2-1 via SWD

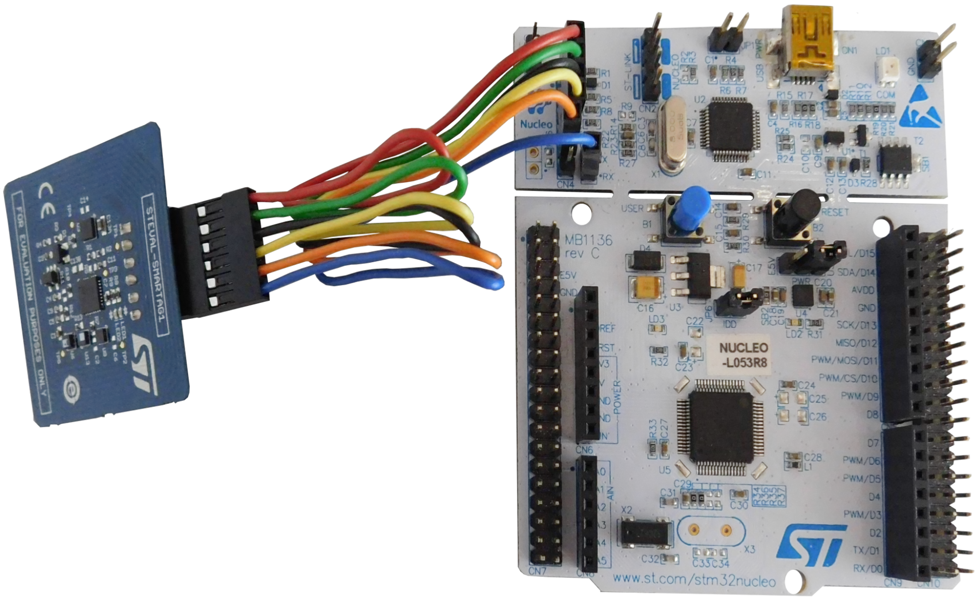
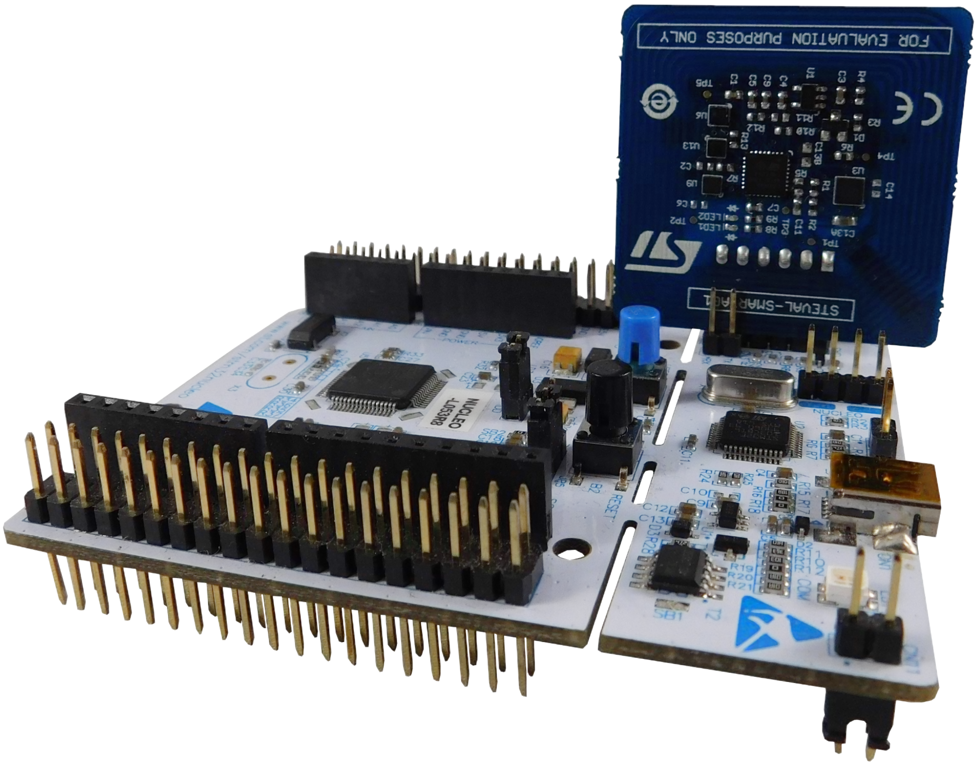


Figure 24. STEVAL-SMARTAG1 evaluation board directly connected to ST-LINK/V2-1



Revision history

Table 1. Document revision history

Date	Version	Changes
26-Mar-2018	1	Initial release.
13-Nov-2019	2	Updated Section 1.1 Overview, Section 1.2 Architecture, Section 1.3 Folder structure, Section 1.5 Sample application description, Section 2.2 Software description, Figure 21. STEVALSMARTAG1 evaluation board connected to ST-LINK/V2-1 via SWD and Figure 22. STEVALSMARTAG1 evaluation board directly connected to ST-LINK/V2-1. Added Section 2.1.4 X-NUCLEO-IKS01A3 expansion board.
15-Feb-2021	3	Updated Section 1.1 Overview, Section 1.2 Architecture, Section 1.6 Android and iOS sample client application, Section 1.6.1 Settings, Section 1.6.2 Min./Max., Section 1.6.3 Data and Section 1.6.4 Single shot. Added Section 2 Using the Asset Tracking web dashboard.
15-Jul-2021	4	Updated Section 1.1 Overview, Section 1.2 Architecture, Section 1.5 Sample application description, Section 3.3.1 Hardware setup and Section 3.3.2 Software setup. Removed Section 3.1.1 STM32 Nucleo, Section 3.1.2 X-NUCLEO-NFC04A1 expansion board, Section 3.1.3 X-NUCLEO-IKS01A2 expansion board, Section 3.1.4 X-NUCLEO-IKS01A3 expansion board, Section 3.3.3.1 STM32 Nucleo and expansion boards setup, and Section 3.3.3.3 Important additional hardware information.

Contents

List of figures

IMPORTANT NOTICE – PLEASE READ CAREFULLY