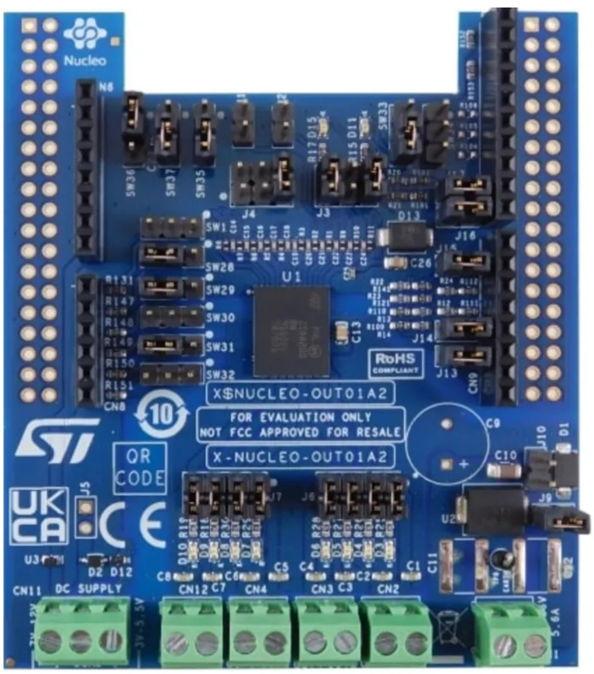
Home » STMicroelectronics » STMicroelectronics UM2207 Expansion Board User Manual

Contents [hide]

life.augmented

STMicroelectronics UM2207 Expansion Board



Introduction

The MotionPM is a middleware library part of the X-CUBE-MEMS1 software and runs on STM32. It provides real-time information about the number of steps and cadence just performed by the user with the device, such as a cell phone. This library is intended to work with ST MEMS only. The algorithm is provided in static library format and is designed to be used on STM32 microcontrollers based on the ARM® Cortex®-M3, ARM Cortex®-M33, ARM® Cortex®-M4, ARM® Cortex®-M7 architecture. It is built on top of STM32Cube software technology that eases portability across different STM32 microcontrollers. The software comes with sample implementation running on X-NUCLEO-IKS4A1 or X-NUCLEO-IKS01A3 expansion board on a NUCLEO-F401RE, NUCLEO-U575ZI-Q or NUCLEO-L152RE development board.

Acronyms and abbreviations

Table 1. **List of acronyms**

Acronym	Description
API	Application programming interface
BSP	Board support package
GUI	Graphical user interface
HAL	Hardware abstraction layer
IDE	Integrated development environment

MotionPM middleware library in X-CUBE-MEMS1 software expansion for STM32Cube

MotionPM overview

The MotionPM library expands the functionality of the X-CUBE-MEMS1 software. The library acquires data from the accelerometer and provides information about the number of steps and cadence just performed by the user with the device. The library is designed for ST MEMS only. Functionality and performance when using other MEMS sensors are not analyzed and can be significantly different from what described in the document. Sample implementation is available for X-NUCLEO-IKS4A1 and X-NUCLEO-IKS01A3 expansion boards, mounted on a NUCLEO-F401RE, NUCLEO-U575ZI-Q or NUCLEO-L152RE development board.

MotionPM library

Technical information fully describing the functions and parameters of the MotionPM APIs can be found in the MotionPM_Package.chm compiled HTML file located in the Documentation folder.

MotionPM library description

The MotionPM pedometer library manages the data acquired from the accelerometer; it features:

- possibility to detect the number of steps and cadence
- recognition based only on accelerometer data
- required accelerometer data sampling frequency is 50 Hz
- resources requirements:
  - Cortex-M3: 8.9 kB of code and 2.5 kB of data memory
  - Cortex-M33: 8.5 kB of code and 2.5 kB of data memory
  - Cortex-M4: 8.8 kB of code and 2.5 kB of data memory
  - Cortex-M7: 8.4 kB of code and 2.5 kB of data memory
- available for ARM® Cortex®-M3, ARM® Cortex®-M33, ARM® Cortex®-M4 and ARM® Cortex®-M7 architectures

MotionPM APIs

The MotionPM library APIs are:

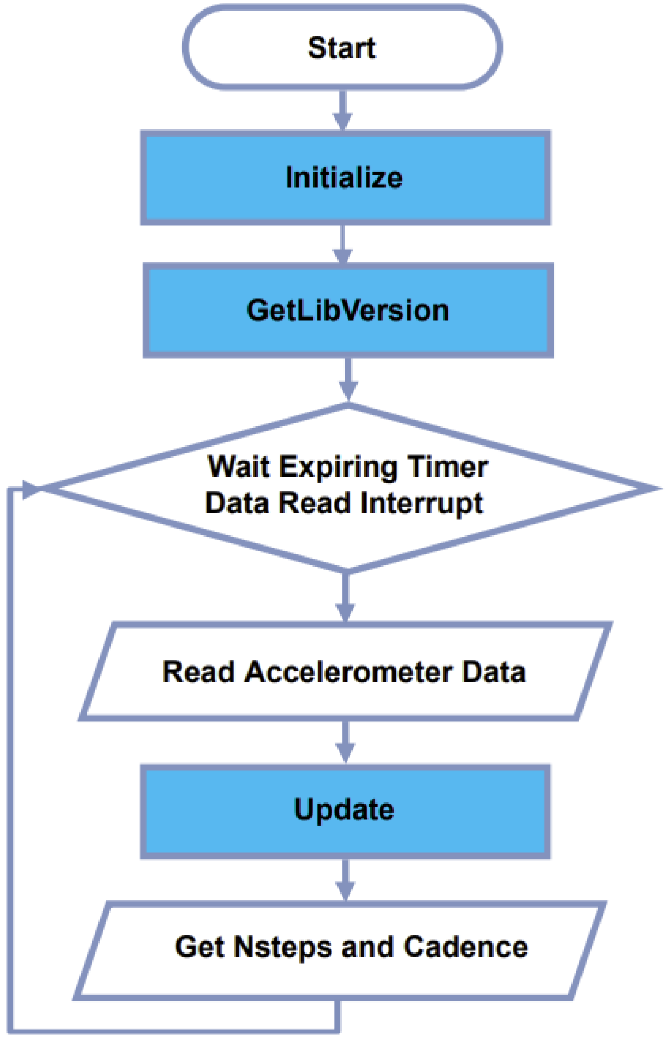
- uint8_t MotionPM_GetLibVersion(char *version)
  - retrieves the library version
  - *version is a pointer to an array of 35 characters
  - returns the number of characters in the version string
- void MotionPM_Initialize(void)
  - performs MotionPM library initialization and setup of the internal mechanism

Note: This function must be called before using the pedometer library and the CRC module in the STM32 microcontroller (in RCC peripheral clock enable register) has to be enabled.

- void MotionPM_Update (MPM_input_t *data_in, MPM_output_t *data_out)
- executes pedometer algorithm
- *data_in parameter is a pointer to a structure with input data
- the parameters for the structure type MPM_input_t are:
  - AccX is the accelerometer sensor value in X axis in g
  - AccY is the accelerometer sensor value in Y axis in g
  - AccZ is the accelerometer sensor value in Z axis in g
- *The data_out parameter is a pointer to a structure with output data
- The parameters for the structure type MPM_output_t are:
  - Nsteps is the number of steps performed by the user
  - Cadence is the user step cadence
- void MotionPM_ResetStepCoun(void)
  - resets the current step count

API flow chart

Figure 1. MotionPM API logic sequence



Demo code

The following demonstration code reads data from the accelerometer sensor and gets the number of steps and cadence.

```
[…]
#define VERSION_STR_LENG 35
[…]

/* Initialization */
char lib_version[VERSION_STR_LENG];

/* Pedometer API initialization function */
MotionPM_Initialize();

/* Optional: Get version */
MotionPM_GetLibVersion(lib_version);

[…]

/* Using Pedometer for wrist algorithm */
Timer_OR_DataRate_Interrupt_Handler()
{
  MPM_input_t data_in;
  MPM_output_t data_out;

  /* Get acceleration X/Y/Z in g */
  MEMS_Read_AccValue(&data_in.AccX, &data_in.AccY, &data_in.AccZ);

  /* Run pedometer algorithm */
  MotionPM_Update(&data_in, &data_out);
}
```

Algorithm performance

The pedometer algorithm only uses data from the accelerometer and runs at a low frequency (50 Hz) to reduce power consumption. It detects and provides real-time information about the number of steps and cadence the user performed with his device.

Table 2. **Algorithm elapse time (µs) Cortex-M4, Cortex-M3**

Cortex-M4 STM32F401RE at 84 MHz			Cortex-M3 STM32L152RE at 32 MHz		
Min	Avg	Max	Min	Avg	Max
1	14	747	2	160	9043

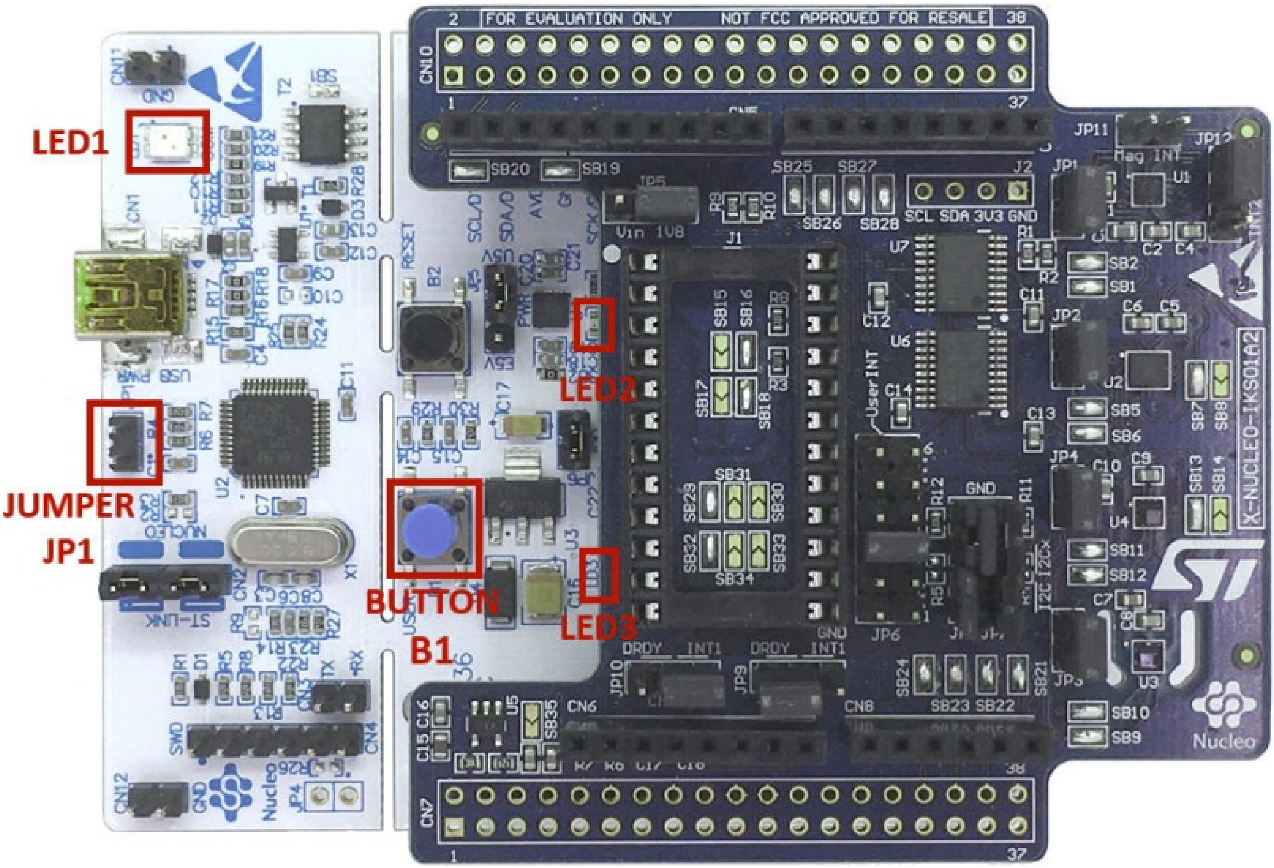
Table 3. **Algorithm elapse time (µs) Cortex-M33 and Cortex-M7**

Cortex- M33 STM32U575ZI-Q at 160 MHz			Cortex- M7 STM32F767ZI at 96 MHz		
Min	Avg	Max	Min	Avg	Max
<1	25	1317	2	144	7243

Sample application

The MotionPM middleware can be easily manipulated to build user applications; a sample application is provided in the Application folder. It is designed to run on a NUCLEO-F401RE, NUCLEO-U575ZI-Q or NUCLEO-L152RE development board connected to X-NUCLEO-IKS4A1 or X-NUCLEO-IKS01A3 expansion board. The application recognizes the steps and cadence in real-time. Data can be displayed through a GUI.

Figure 2. **STM32 Nucleo: LEDs, button, jumper**



The above figure shows the user button B1 and the three LEDs of the NUCLEO-F401RE board. Once the board is powered, LED LD3 (PWR) turns ON. A USB cable connection is required to monitor real-time data. The board is powered by the PC via USB connection. This working mode allows the user to display detected steps, accelerometer data, time stamp and eventually other sensor data, in real-time, using the MEMS-Studio.

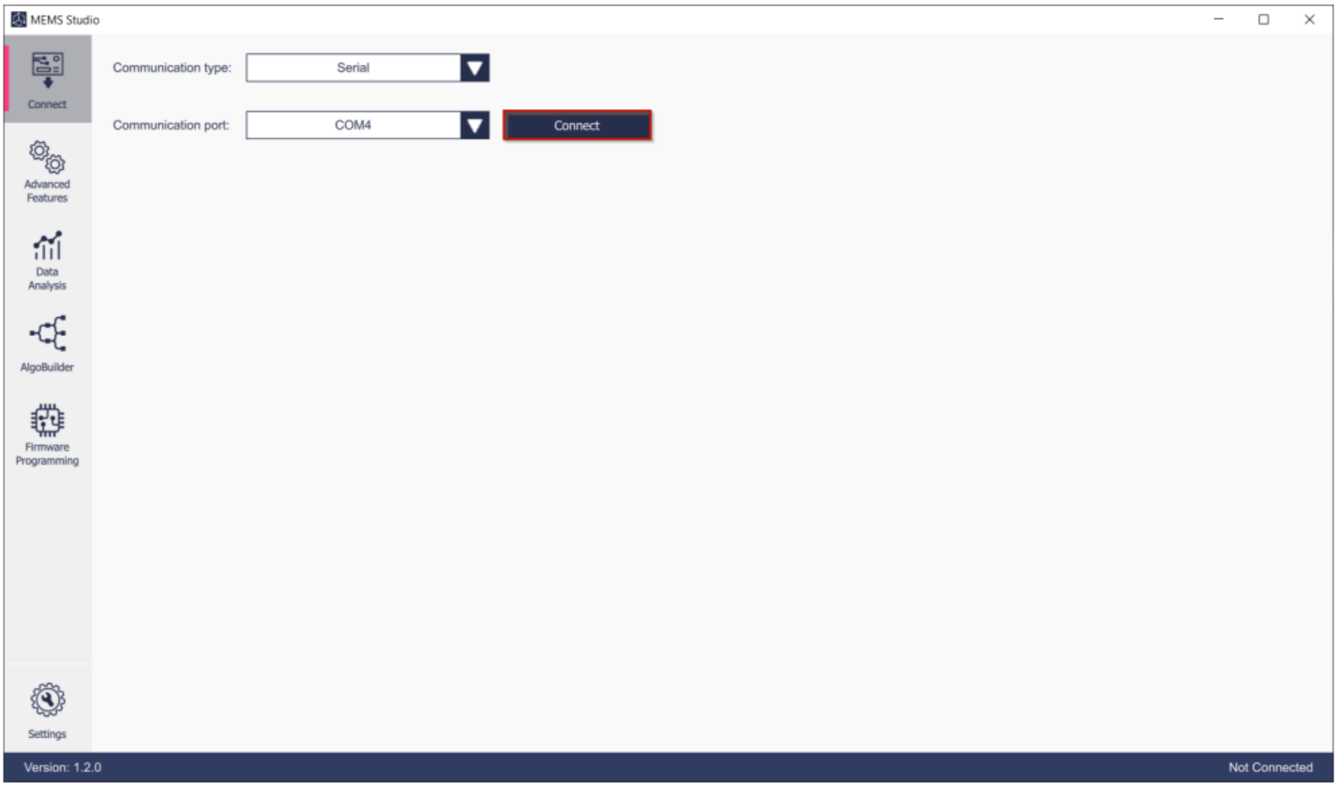
MEMS Studio application

The sample application uses the Windows MEMS-Studio utility, which can be downloaded from www.st.com.

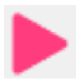
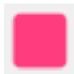
Step 1. Ensure that the necessary drivers are installed and the STM32 Nucleo board with the appropriate expansion board is connected to the PC.

Step 2. Launch the MEMS-Studio application to open the main application window. If an STM32 Nucleo board with supported firmware is connected to the PC, it is automatically detected. Press the [Connect] button to establish connection to the evaluation board.

Figure 3. **MEMS-Studio - Connect**

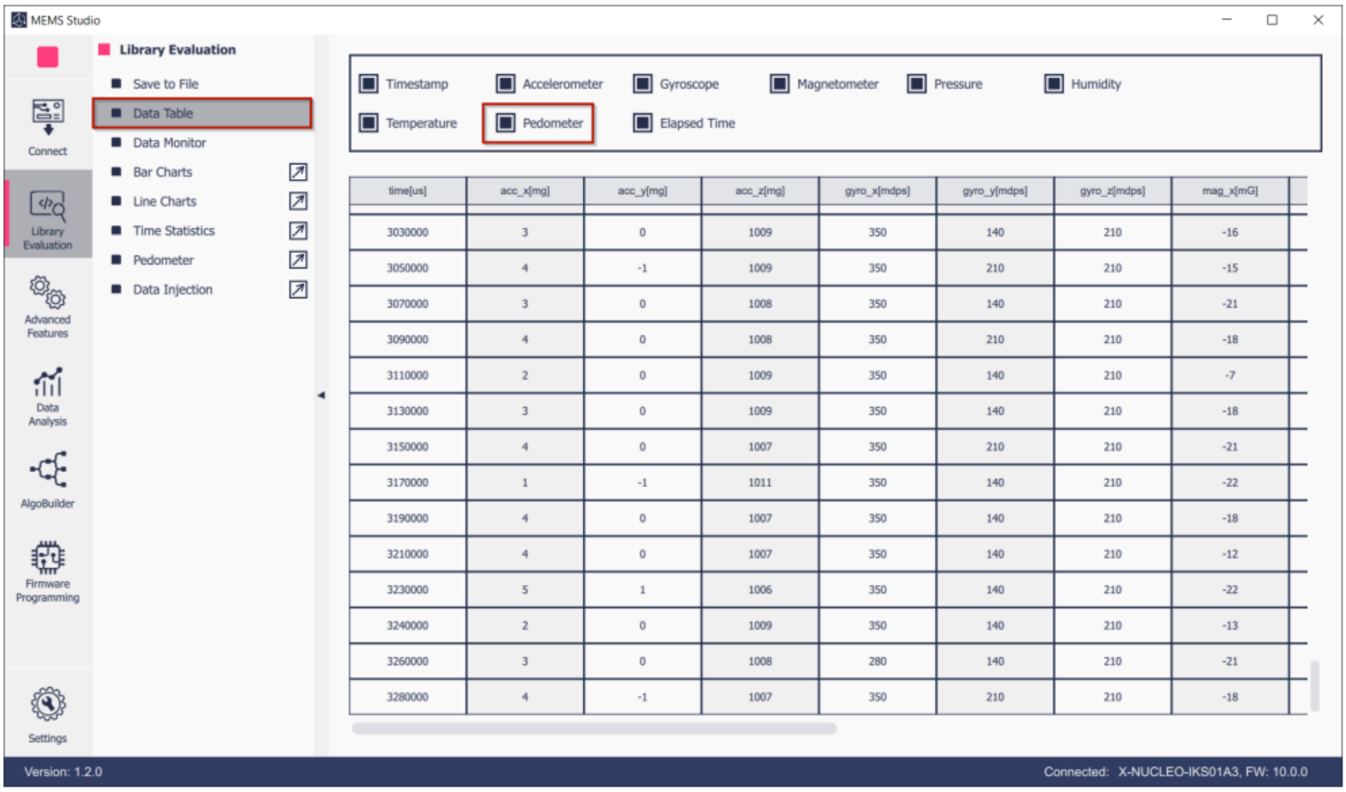


Step 3. When connected to a STM32 Nucleo board with supported firmware [Library Evaluation] tab is opened.

To start and stop data streaming, toggle the appropriate [Start]  or [Stop]  button on the outer vertical tool bar.

The data coming from the connected sensor can be viewed selecting the [Data Table] tab on the inner vertical tool bar.

Figure 4. MEMS-Studio - Library Evaluation - Data Table

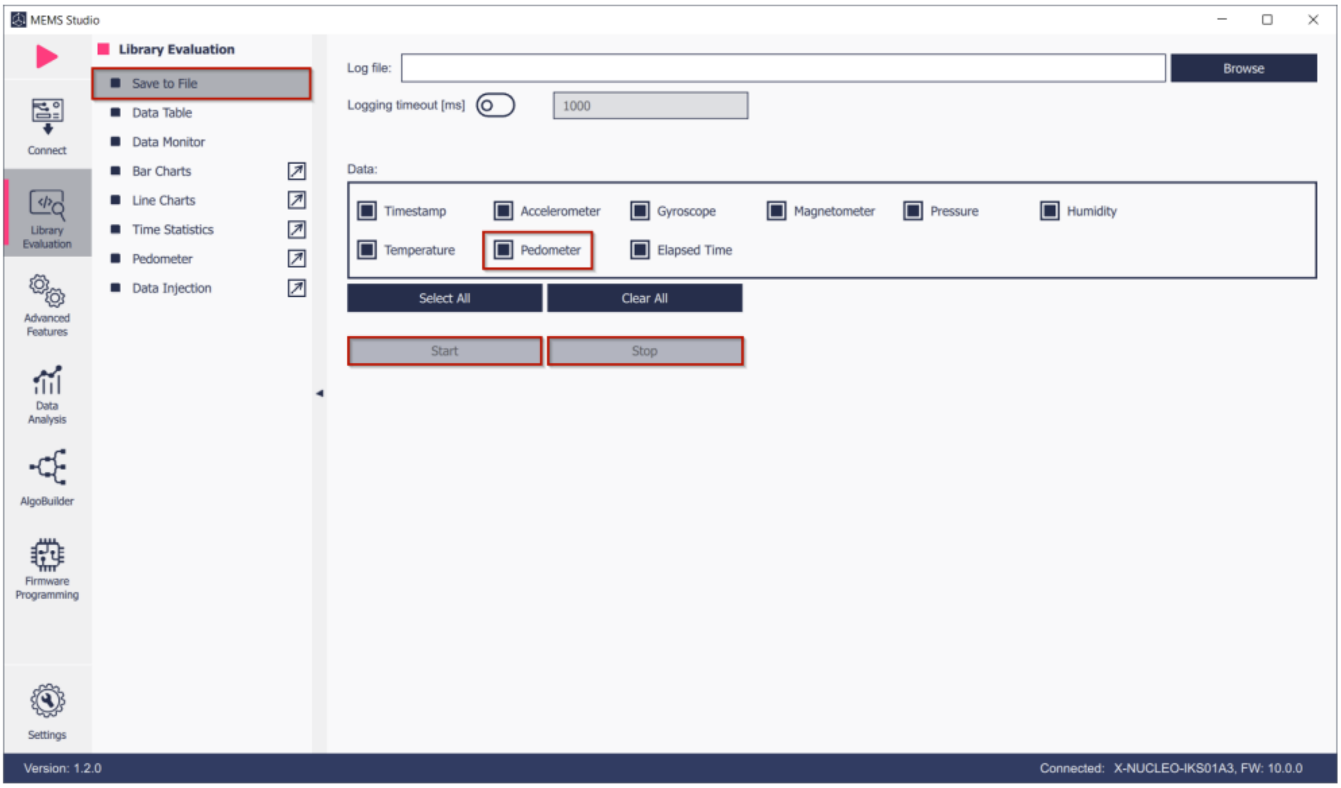


Step 4. Click on the [Pedometer] to open the dedicated application window.

Figure 5. MEMS-Studio - Library Evaluation - Pedometer

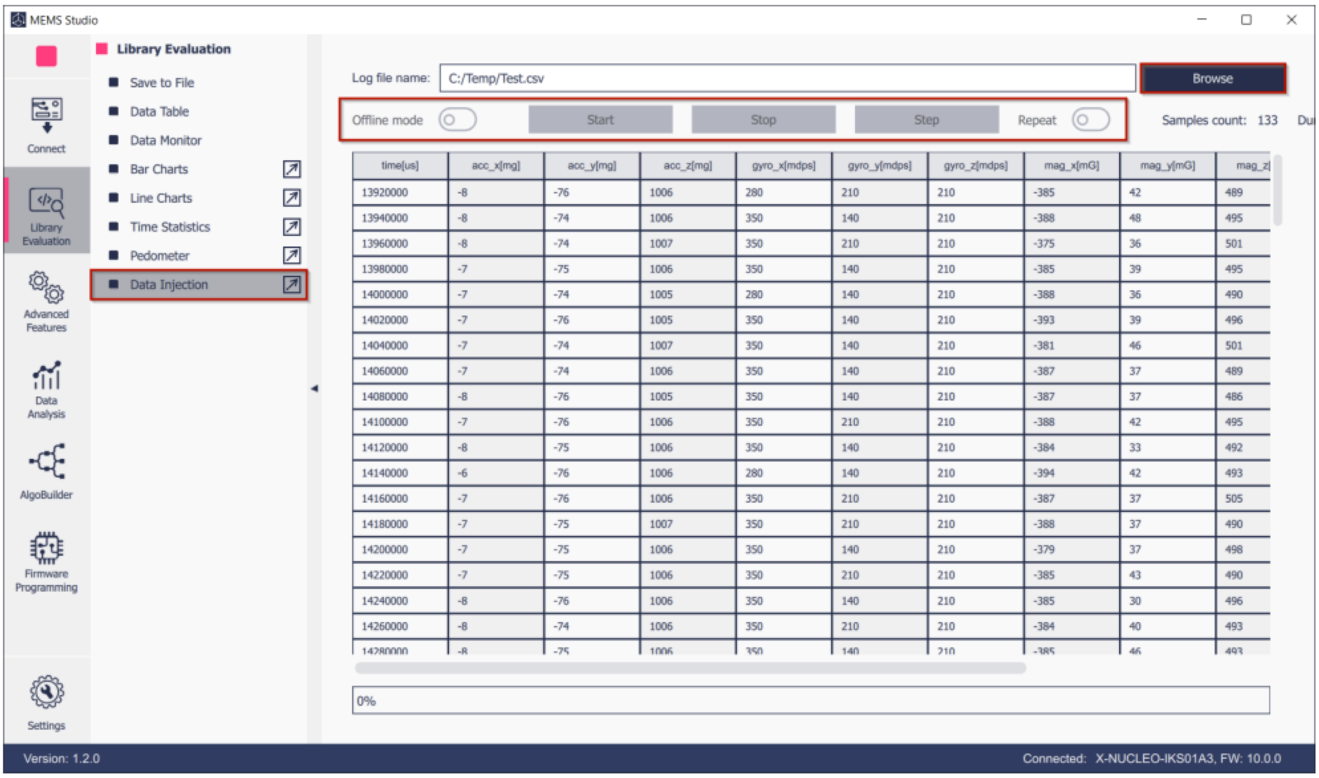
Step 5. Click on the [Save To File] to open the datalogging configuration window. Select the sensor and pedometer data to be saved in the file. You can start or stop saving by clicking on the corresponding button.

Figure 6. **MEMS-Studio - Library Evaluation - Save To File**



Step 6. Data Injection mode can be used to send the previously acquired data to the library and receive the result. Select the [Data Injection] tab on the vertical tool bar to open the dedicated view for this functionality.

Figure 7. **MEMS-Studio - Library Evaluation - Data Injection**



Step 7. Click on the [Browse] button to select the file with the previously captured data in

CSV format.

The data will be loaded into the table in the current view.

Other buttons will become active. You can click on:

- [Offline Mode] button to switch the firmware offline mode on/off (mode utilizing the previously captured data).
- [Start]/[Stop]/[Step]/[Repeat] buttons to control the data feed from MEMS-Studio to the library.

References

All of the following resources are freely available on www.st.com.

1. UM1859: Getting started with the X-CUBE-MEMS1 motion MEMS and environmental sensor software expansion for STM32Cube
2. UM1724: STM32 Nucleo-64 boards (MB1136)
3. UM3233: Getting started with MEMS-Studio

Revision history

Table 4. Document revision history

Table 4. Document revision history

Date	Version	Changes
06-Jun-2017	1	Initial release.
06-Feb-2018	2	Added references to NUCLEO-L152RE development board and Table 2. Elapsed time (µs) algorithm..
20-Mar-2018	3	Updated Introduction and Section 2.1 MotionPM overview.
20-Feb-2019	4	Updated Section 2.2.5: Algorithm performance. Added X-NUCLEO-IKS01A3 expansion board compatibility information.
02-Jun-2025	5	UpdatedSection Introduction, Section 2.1: MotionPM overview, Section 2.2.1: MotionPM library description, Section 2.2.2: MotionPM APIs, Section 2.2.4: Demo code, Section 2.2.5: Algorithm performance, Section 2.3: Sample application, Section 2.4: MEMS Studio application.

IMPORTANT NOTICE – READ CAREFULLY

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.

Frequently Asked Questions

- **Q: Can the MotionPM library be used with non-ST MEMS sensors?**
  - A: No, the library is designed for ST MEMS only. Using other MEMS sensors may result in different functionality and performance.
- **Q: How to initialize the MotionPM library?**
  - A: You need to call the `MotionPM_Initialize()` function before using the pedometer library. Ensure that the CRCmodule in the STM32 microcontroller is enabled in the RCC peripheral clock enable register.

Documents / Resources

	STMicroelectronics UM2207 Expansion Board [pdf] User Manual X-NUCLEO-IKS4A1, X-NUCLEO-IKS01A3, NUCLEO-F401RE, NUCLEO-U575ZI-Q, NUCLEO-L152RE, UM2207 Expansion Board, UM2207, Expa nsion Board

References

- User Manual

STMicroelectronics

Expansion Board, NUCLEO-F401RE, NUCLEO-L152RE, NUCLEO-U575ZI-Q, STMicroelectronics, UM2207, UM2207 Expansion Board, X-NUCLEO-IKS01A3, X-NUCLEO-IKS4A1

Leave a comment

Your email address will not be published. Required fields are marked *

Comment *

Name

Email

Website

Save my name, email, and website in this browser for the next time I comment.

Post Comment

Search:

e.g. whirlpool wrf535swhz

Search