Integration Document
BSXlite Library

BSXlite Integration Document

Document version	1.0
Document release date	July 2021
Document number	BST-BSX-SD001-00
Technical reference code(s)	
Notes	Data and descriptions in this document are subject to change without notice. Product photos and pictures are for illustration purposes only and may differ from the real product appearance.

Table of Contents

1 INTRODUCTION 6
1.1 Who should read this? 6

2 GENERAL DESCRIPTION 7

3 REQUIREMENTS & SPECIFICATIONS FOR INTEGRATION 11
3.1 Input Data Specifications 11
3.2 Output Data Specifications 12

4 INTERFACE USAGE GUIDE 14
4.1 Code Size 14
4.2 Header Files Required 14
4.3 Data types specified in the library 14
4.4 BSXlite Version 15
4.5 Operation Sequence 16
4.5.1 Initialization 16
4.5.2 Signal Processing 16
4.5.3 Set To Default 17

5 INTEGRATION VERIFICATION 18
5.1 Performing calibration for the sensors 18
5.1.1 Acceleration calibration 18
5.1.2 Gyro calibration 18
5.2 Tests for IMU Rotation (Quaternion vector) 18
5.2.1 Yaw rotations 19
5.2.2 Pitch rotations 19
5.2.3 Roll rotations 20
5.3 Tests for orientation (euler angles) vector 20
5.3.1 Yaw rotations 21
5.3.2 Pitch rotations 21
5.3.3 Roll rotations 22

6 APPLICATION EXAMPLE AND HINTS 23

7 LIST OF ABBREVIATIONS 24

8 REFERENCES 25

9 APPENDIX 26
9.1 Overview of Sensors 26
9.1.1 Accelerometer 26
9.1.2 Gyroscope 26

9.2 Axis Remapping example 26
9.2.1 Verifying Axis Remapping 27

List of Tables

1 Features of the Bosch Sensor Fusion Libraries 8
2 Sensor Fusion Solutions from Bosch Sensortec Fusion Libraries 9
3 Fusion Outputs: BSXlite & Bosch Sensortec Sensor Fusion Libraries 10
4 Details of the Physical sensors 11
5 Accuracy Status of Calibrators 12
6 Details of the Output sensors 13
7 Code Size 14
8 Return Values 15
9 Version API details 15
10 Operation Sequence of library 16
11 Initialization API details 16
12 Do step API details 17
13 Set to default API details 17
14 Yaw rotations 19
15 Pitch rotations 19
16 Roll rotations 20
17 Orientation : Yaw rotations 21
18 Orientation : Pitch rotations 21
19 Orientation : Roll rotations 22
20 List fo Abbreviations 24
21 Verifying axis remapping for accelerometer 27
22 Verifying axis remapping for gyroscope 27

List of Figures

1 BSXlite Rotation convention 12
2 BSX default convention : Rotations performed vs Orientation output 20
3 Example for axis remapping 26

1 INTRODUCTION

This manual provides the details of the BSXlite library v1.0.2 (hereafter referred to as BSXlite library) and can be used to understand the interfaces and operation sequence of the BSXlite library. The document covers the below aspects :

- Overview of the BSXlite library
- Pre-requisites and specifications for inputs & outputs
- Explanation of the library interfaces and the operation sequence
- Tests for integration verification

1.1 Who should read this?

This information is intended for users who want to perform integration of BSXlite library on various supported platforms.

2 GENERAL DESCRIPTION

BSXlite library is an IMU fusion (6-DoF) solution which combines a MEMS *3-axis gyroscope* and a MEMS *3-axis accelerometer* to provide a robust orientation information (w.r.t to the Earth's frame) with high static and dynamic accuracy. Orientation is provided as a rotation quaternion (IMU Rotation vector) and also in polar coordinates (IMU Orientation vector as Euler angles).

The BSXlite library is a teaser and a subset of Bosch Sensortec's sensor fusion libraries. The sensor fusion algorithms are designed to provide enhanced user experience in smart devices - smart phones, wearables, hearables. With BSXlite, users can easily integrate and comprehend the performance of the orientation output from Bosch Sensortec's sensor fusion engine. BSXlite is suitable for augumented reality, virtual reality, mobile gaming and other similar applications, where relative orientation is useful.

A list of the outputs and features of Bosch Sensortec's sensor fusion libraries is provided in this section *Key Features of Bosch Sensortec's Sensor Fusion Libraries*.

Key features and fusion outputs are provided in the tables in this section. Additionaly the features supported by BSXlite are also mentioned.

FEATURES	DESCRIPTION	BSXLITE LIBRARY	BST SENSOR FUSION LIBRARIES
Output Data Rates	Outputs available at data sample rates: 1.5625 Hz to 800 Hz Complete set of supported rates are given below : 1.5625, 3.125, 6.25, 12.5, 25, 50, 100, 200, 400, 800 (Hz)	Supports 100 Hz	Supported
Coordinate Systems	Different coordinate systems are supported. By default Android coordinate system is configured. Default coordinate system is East-North-Up (ENU) coordinates	Android	Android / Windows
Sensor Fusion Modes	Different fusion solutions are available according to the sensor combinations. Each mode is suitable for specific usecases. IMU mode : Accel + Gyro fusion E-Compass mode : Accel + Mag fusion NDoF mode : Accel + Mag + Gyro fusion See table *Sensor Fusion Modes* for detailed explanation	Supports IMU Fusion	Supported
Sensor Calibration	Calibration routines for Accelerometer, Gyroscope and Magnetometer	Supports Accel & Gyro	Supported
Calibration Recovery	Recover or reload the previously estimated sensor offsets after a system restart. Advantage : Users do not need to do re-calibration (Given the actual sensor offsets did not change within the physical sensors)	Not supported	Supported
Multi Output Rate Support	Runs each output at independent sample rates as required	Not supported	Supported
Multiple Instances	Enables to create and execute several instances of the library into the same target system	Not supported	Supported
Soft Iron Correction	SIC can be configured to remove Soft Iron effects of magnetometer	Not supported	Supported
Fusion Configuration	Configuration interface to tune and control fusion performance	Not supported	Supported

Table 1: Features of the Bosch Sensor Fusion Libraries

SENSOR FUSION	DESCRIPTION	BSXLITE LIBRARY	BST SENSOR FUSION LIBRARIES
NDoF Fusion	Data fusion of all three MEMS sensors. Provides orientation of the device as a unit rotation quaternion vector. This orientation represents the rotation required to align the device frame to the Earth's frame (ENU by default). The Orientation represented as a Quaternion vector, *quat*, with four unitless components x, y, z, w is defined as below: quat.x = rot.x*sin(θ/2) quat.y = rot.y*sin(θ/2) quat.z = rot.z*sin(θ/2) quat.w = cos(θ/2) where, *rot* represents ENU rotation axes and θ is the rotation angle Along with the rotation vector, a heading accuracy status field is given Indicates the error in heading angle (in radians). The Orientation represented in Euler angles are defined as below: **Heading**(Yaw): Yaw axis (+Z device) has its origin at the center of gravity of the device and is directed towards to the sky (Up). Rotation angle about this axis is called yaw. Heading refers to the magnetic heading, the angle the device makes with it's +Y axis to the Earth's magnetic North. **Pitch**: Pitch axis (+X device) has its origin at the center of gravity and is directed to East(Right). Rotation angle about this axis is Pitch. **Roll**: Roll axis (+Y device) has its origin at the center of gravity and is directed North(Forward). Motion about this axis is called Roll NDoF fusion orientation is used for Navigation applications.	Not Supported	Supported
IMU Fusion	Data fusion of accel and gyro sensors. IMU orientation differs from NDoF orientation with respect to Yaw. As magnetic data is not available, Yaw rotations are derived from gyro data. Hence relative rotation is available in Yaw. The Pitch and Roll angles remains same as in NDoF fusion. Typical applications for IMU fusion are mobile gaming, AR & VR.	Supported	Supported
E-Compass Fusion	Data fusion of accel and mag sensors Magnetic information is used here, instead of gyro. A heading accuracy status is provided to indicate the error in the heading estimate. Heading accuracy is given in radians. E-Compass fusion is useful for outdoor navigation.	Not Supported	Supported

Table 2: Sensor Fusion Solutions from Bosch Sensortec Fusion Libraries

FUSION OUTPUTS	DESCRIPTION	BSXLITE LIBRARY	BST SENSOR FUSION LIBRARIES
Acceleration Raw	Input acceleration data	No	Yes
Acceleration Offsets	Estimated acceleration offsets with calibration accuracy status	No	Yes
Corrrected Acceleration	Offset corrected acceleration with calibration accuracy status	No	Yes
Angular Rate Raw	Input angular rate data	No	Yes
Angular Rate Offsets	Estimated angular rate offsets with calibration accuracy status	No	Yes
Angular Rate Corrected	Offset corrected gyroscope data with calibration accuracy status	No	Yes
Calibration Accuracy Status	Each of the calibrators provides a calibration accuracy status Please refer Accuracy Status of Calibrators	Yes Accel & Gyro	Yes
Calibration New Data Flag	Each of the calibrators provide a new data flag when a new offset has been estimated	No	Yes
Magnetic Field Raw	Input magnetic field	No	Yes
Magnetic Field Offsets	Estimated offsets of magnetometer signal with calibration accuracy status	No	Yes
Corrected Magnetic Field	Offset corrected magnetic Field with calibration accuracy status	No	Yes
NDoF Rotation Vector	Provides orientation as a unit quaternion vector with heading accuracy status from NDoF fusion	No	Yes
NDoF Orientation	Provides orientation as Euler angles from NDof Fusion	No	Yes
IMU Rotation Vector	Provides orientation as a unit quaternion vector from IMU fusion	Yes	Yes
IMU Orientation	Provides orientation in Euler angles from IMU fusion	Yes	Yes
E-Compass Rotaion vector	Provides orientation as a unit quaternion vector with heading accuracy status from E-Compass fusion	No	Yes
E-Compass Orientation	Provides orientation as Euler angles from E-Compass fusion	No	Yes
Gravity vector	Provides the acceleration due to gravity of the device in all three axes	No	Yes
Linear Acceleration	Provides the linear acceleration of the device in all three axes	No	Yes
Rotation Matrix	Provides rotation information as a rotation matrix	No	Yes

Table 3: Fusion Outputs: BSXlite & Bosch Sensortec Sensor Fusion Libraries

3 REQUIREMENTS & SPECIFICATIONS FOR INTEGRATION

This section will give an overview about the pre-requisites for the fusion library and what has to be verified before integration of the fusion library.

The critical pre-requisites of the library which can affect the library performance are listed below:

- Input data (raw data from different hardware sensors and corresponding timestamps).
- Sensor coordinate system must be remapped to device coordinate system
- The platform should support all standard C/C++ libraries for processing. BSXlite can be integrated into ARM core (cortex3 and above)
- The target system shall provide enough memory resources (RAM and ROM) to allow the fusion library to operate as expected.

3.1 Input Data Specifications

The target system shall feed into the BSX library accelerometer and gyroscope data along with the gyro timestamp in micro seconds. The provided timestamp represents the time at which the raw data was generated by the sensors. If the acceleration and data and angular rate data are coming from different sensors, the signals should be synchronized. The input timestamp to BSXlite should be the timestamp from the gyro signal. The BSXlite only supports an input sensor data rate of 100 Hz.

The 3-axis sensor data must be correctly aligned with the coordinate frame of the target system, which means that the accelerometer and the gyroscope sensor orientation should be inline with the device orientation. (Kindly refer the Appendix for an *example for axis remapping*)

The details of the required input raw data for each of the sensors should be in the below format:

Input Signal	Data Type	Input Unit	Data Rate	Timestamp Unit
Raw Acceleration sensor data	vector_3d_t	$meter/second^2(m/s^2)$	100 Hz	micro seconds
Raw Angular rate sensor data	vector_3d_t	$radians/second(rps)$	100 Hz	micro seconds

Table 4: Details of the Physical sensors

3.2 Output Data Specifications

BSXlite provides four outputs:

1. The IMU Rotation Vector is a rotation quaternion vector. It provides the rotation of the device coordinate system with respect to the ENU coordinate system.

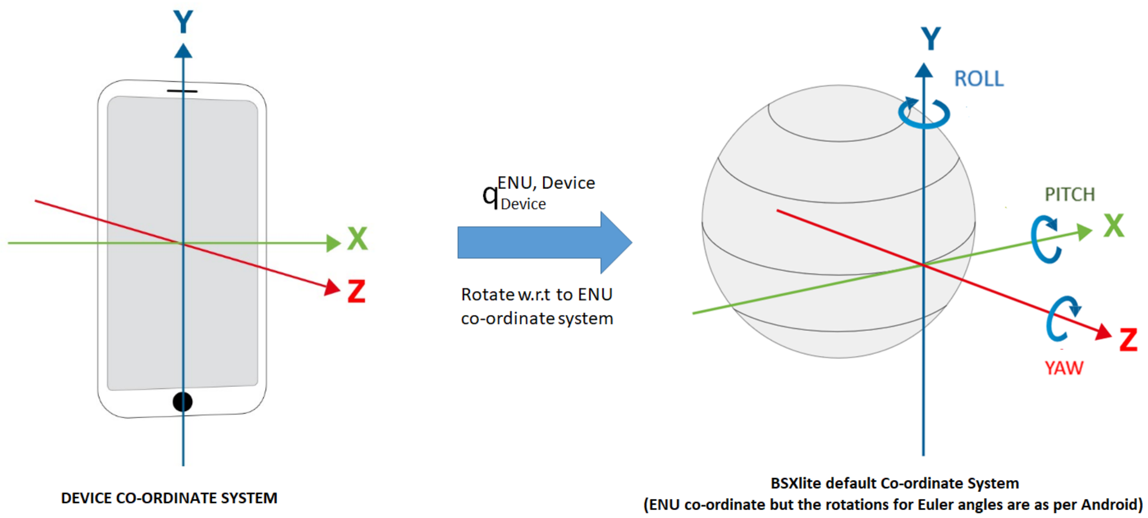

Fig. 1: BSXlite Rotation convention

2. Orientation as Euler angles in [radians] units. The orientaion angles are in the given order: [Heading , Pitch, Roll, Yaw]. In the default BSX coordinate system, Heading and Yaw are equal in both magnitude and direction. The behaviour of the Euler angles are detailed in integration verification section
3. Acceleration calibration accuracy denotes the accuracy of the estimated acceleration offsets. The status can report one of the following values - [0,1,2,3]
4. Gyro calibration accuracy denotes the accuracy of the estimated gyro offsets. The status can report one of the following values - [0,1,2,3]

Calibration accuracy status indicates whether sensor is calibrated for use. When accuracy is 3 for both sensors, you can expect the best possible results from fusion. To calibrate accel, perform the accel calibration as mentioned in Accelerometer Calibration, and for gyroscope, let the device stay stable (Gyroscope Calibration)

Accuracy Level	Description
0	No calibration
1	Low accuracy, calibration process initiated
2	Medium accuracy for the estimated offsets
3	Calibration process complete, highest accuracy for the estimated offsets

Table 5: Accuracy Status of Calibrators

Output signal	**Data Type**	**Units**	**Data Rate**
IMU Rotation vector (Quaternion)	quaternion_t	-	100 Hz
Orientation (Euler angles)	euler_angles_t	$radians$	100 Hz
Acceleration Calibration Accuracy Status	uint8	-	100 Hz
Gyroscope Calibration Accuracy Status	uint8	-	100 Hz

Table 6: Details of the Output sensors

4 INTERFACE USAGE GUIDE

This section provides details of the interfaces of BSXlite and the operation sequence in which the interfaces have to be called. The pre-requisites and details for the usage of interfaces are also mentioned here

4.1 Code Size

The total code size of the compiled BSXlite library is 17688 bytes (compiled for cortex M3 platform)

text	data	bss	dec	hex
16116	0	1572	17688	4518

Table 7: Code Size

The details of the compiler used for code size estimation is given below:

- Compiler: GCC - GNU Tools for ARM Embedded Processors - arm-none-eabi-gcc.exe
- Version: 5.4.1
- Platform: Cortex M3
- Compiler configuration: -std=c99 -mcpu=cortex-m3 -mthumb -Os -fno-common

4.2 Header Files Required

The deliverable of the BSXlite requires the following header file which needs to be included in addition to linking with the BSXlite binary.

- **bsxlite_interface.h**

4.3 Data types specified in the library

- Module instance definition - bsxlite_instance_t

```
typedef size_t bsxlite_instance_t;
```

- Generic data struct to hold input sensor data- vector_3d_t

```
typedef struct
{
float x;
float y;
float z;
}vector_3d_t;
```

- Quaternion structure - quaternion_t

```
typedef struct
{
float x;
float y;
float z;
float w;
```

```
}quaternion_t;
```

▶ Euler angle structure - euler_angles_t

```
typedef struct
{
float heading;
float pitch;
float roll;
float yaw;
}euler_angles_t;
```

▶ Fusion output structure

```
typedef struct
{
    quaternion_t rotation_vector;
    euler_angles_t orientation;
    uint8_t accel_calibration_status;
    uint8_t gyro_calibration_status;
}bsxlite_out_t;
```

▶ Module return type definition - bsxlite_return_t

```
typedef int16_t bsxlite_return_t;
```

Return Value	Identifier	Description
0	BSXLITE_OK	Status OK
-254	BSXLITE_E_FATAL	Fatal error Triggered by exceptions such as a null pointer to library
-5	BSXLITE_E_DOSTEPS_TSINTRADIFFOUTOFRANGE	When the time difference between two consecutive timestamp values is >10% than the sensor output data rate. In case of the BSXlite: Sensors ODR = 100Hz –> T = 1/ODR = 0.01s –> 10% = 1 ms
2	BSXLITE_I_DOSTEPS_NOOUTPUTSRETURNABLE	Not enough memory available to allocate the BSXlite output

Table 8: Return Values

4.4 BSXlite Version

To retrieve the current version of the BSXlite , the API **bsxlite_get_version** has to be called. The description of the API is provided below:

```
void bsxlite_get_version(bsxlite_version * version);
```

Parameter	Description
version	returns the library version information

Table 9: Version API details

4.5 Operation Sequence

This section provides the details of the interface functions that have to be called for a typical operation of the BSXlite.

Operation	BSXlite function
Initialize and set to default the internal parameters of the BSXlite engine	bsxlite_init
Run the BSXlite fusion algorithm	bsxlite_do_step
Set/Reset internal parameters of the BSXlite engine	bsxlite_set_to_default

Table 10: Operation Sequence of library

4.5.1 Initialization

Initialization of the BSXlite is performed by calling **bsxlite_init**. Calling this function resets all the library parameters to default. The init interface initializes the BSXlite and returns its instance. This instance pointer should be provided as input to all other BSXlite functions.

The description of the API is provided below:

```
bsxlite_return_t bsxlite_init(bsxlite_instance_t *instance)
```

Parameter	Description
instance	instance of Bsxlite

Table 11: Initialization API details

4.5.2 Signal Processing

Processing of input sensor data to get outputs from BSXlite can be done using the function **bsxlite_do_step**.

The description of the API is provided below

```
bsxlite_return_t bsxlite_do_step(const bsxlite_instance_t *instance, const int32_t
    w_time_stamp, const vector_3d_t * accel_in, const vector_3d_t * gyro_in, bsxlite_out *
    output_data);
```

Parameter	Data Type	Description
instance	bsxlite_instance_t	instance of BSXlite
w_time_stamp	int32	angular rate time stamp in [micro s] (micro seconds)
acc_input	vector_3d_t	input acceleration data <x y z> in $[m/s^2]$
gyro_input	vector_3d_t	input angular rate data <x y z> in [radians/second]
output_data	bsxlite_out_t	output_data consists of the below signals **IMU Rotation Vector** : Quaternion vector <x, y, z, w> Data type : quaternion_t **Orientation** : Euler angles <heading, pitch, roll, yaw> Data type : euler_angles_t Units : [radians] **Accel calibration status** Data type : uint8 Dimension : [1] **Gyro calibration status** Data type : : uint8 Dimension : [1]

Table 12: Do step API details

4.5.3 Set To Default

Properties of the BSXlite library can be reset (or set to default values) by calling **bsxlite_set_to_default**. The description of the API is provided below:

```
bsxlite_return_t bsxlite_set_to_default(const bsxlite_instance_t *instance);
```

Parameter	Description
instance	instance of Bsxlite

Table 13: Set to default API details

5 INTEGRATION VERIFICATION

This section details about the tests which can be performed to verify successful integration. Perform calibration movements for accel signal and gyro signal for optimal performance and before testing.

5.1 Performing calibration for the sensors

What is Calibration? Data from the physical sensors may show deviation from the ideal response due to inherent errors and the exposure to different environmental conditions. In order to compensate the offset present in the raw signal, calibration routines are present in the library for each sensor. Thus the raw sensor signal can be corrected and the corrected sensor signal can be used for optimized fusion by the BSXlite library. Calibration accuracy status for accel and gyro are provided as output from BSXlite to indicate whether the raw sensor data is calibrated or not.

How to do Calibration? Calibration movements have to be performed to facilitate the calibration logic inside the library. In order to calibrate the accelerometer raw data a sequence of movements is required to be done with the device. Whereas gyro calibration is done by keeping the device stable. Detailed descriptions are provided below. A calibration accuracy status of 3 indicates successful calibration and offsets has been estimated for correcting raw sensor data.

5.1.1 Acceleration calibration

A detailed description of how to perform the 6 axes accel calibration movements are provided below (rotations to be performed in clockwise direction):

1. Keep the device flat on a surface (+Z, acc_x = 0 m/s^2, acc_y = 0 m/s^2, acc_z = 9.81 m/s^2) and keep it stable for 2 seconds
2. Rotate 90 degrees in Pitch axis (+Y) and keep it stable for 2 seconds
3. Then rotate 180 degrees in Pitch (-Y), and keep it stable for 2 seconds
4. Go back to initial position (as in Step 1), then rotate 90 degrees in Roll (+X), and keep it stable for 2 second
5. Then rotate 180 degrees in Roll (-X) and keep it stable for 2 seconds
6. Keep device upside down (-Z), and keep it stable for 2 seconds
7. The calibration accuracy status should become 3, indicating that the calibration has been completed and the best offset estimation has been determined

5.1.2 Gyro calibration

To calibrate gyroscope signal, keep the device stable for approx. 12 seconds. Gyroscope calibration is complete when calibration accuracy status becomes 3.

5.2 Tests for IMU Rotation (Quaternion vector)

Test cases and expected results for the IMU rotation quaternion vector are tabulated below. The rotations are to be performed in clockwise direction in each axis.

5.2.1 Yaw rotations

S. No	Test Angle Yaw (degrees)	Quat X	Quat Y	Quat Z	Quat W
1	0	0	0	0	1
2	90	0	0	$\frac{-1}{\sqrt{2}}$	$\frac{1}{\sqrt{2}}$
3	180	0	0	1	0
4	270	0	0	$\frac{1}{\sqrt{2}}$	$\frac{1}{\sqrt{2}}$

Table 14: Yaw rotations

5.2.2 Pitch rotations

S. No	Test Angle Pitch (degrees)	Quat X	Quat Y	Quat Z	Quat W
1	0	0	0	0	1
2	90	$\frac{-1}{\sqrt{2}}$	0	0	$\frac{1}{\sqrt{2}}$
3	180	1 or -1	0	0	0
4	270	$\frac{1}{\sqrt{2}}$	0	0	$\frac{1}{\sqrt{2}}$

Table 15: Pitch rotations

5.2.3 Roll rotations

S. No	Test Angle Roll (degrees)	Quat X	Quat Y	Quat Z	Quat W
1	0	0	0	0	1
2	90	0	$\frac{-1}{\sqrt{2}}$	0	$\frac{1}{\sqrt{2}}$
3	180	0	1 or -1	0	0
4	270	0	$\frac{1}{\sqrt{2}}$	0	$\frac{1}{\sqrt{2}}$

Table 16: Roll rotations

5.3 Tests for orientation (euler angles) vector

Below a graphical representation of the orientation output angles when rotations from 0 to 360 degrees are applied in different axes is provided below. The rotations are performed in clockwise direction in each corresponding axis.

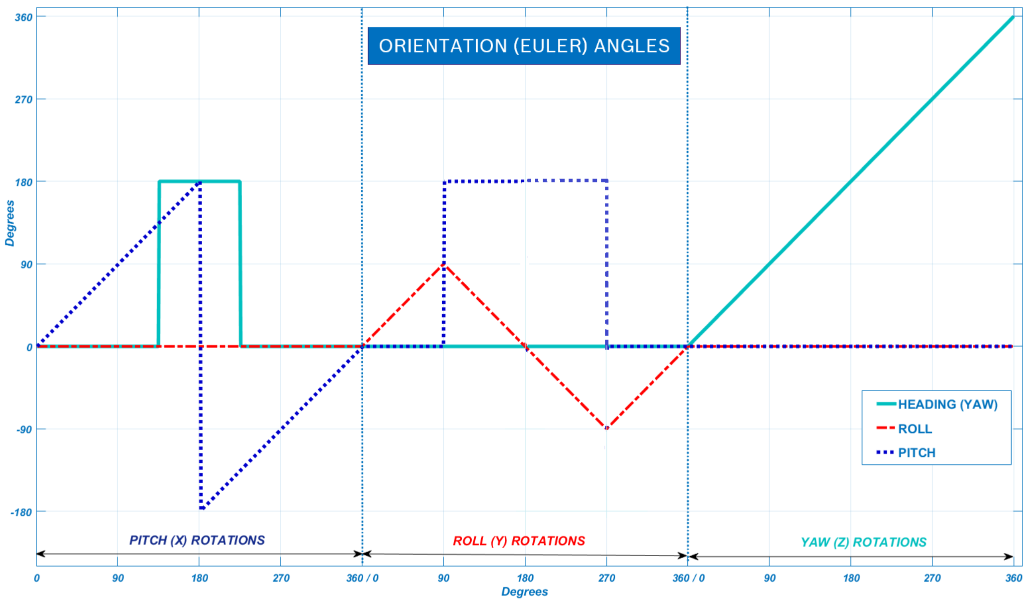

Fig. 2: BSX default convention : Rotations performed vs Orientation output

- The euler angles in BSXlite by default are defined to have range as below:
  1. For the roll angle(i.e. rotation around Y axis), the range is [-90,+90] degrees.
  2. For pitch angle (i.e. rotation around X axis), the range is [-180, 180] degrees. Therefore for pitch angle the value -180 and +180 degree represents same orientation.

3. For yaw angle (i.e. rotation about Z axis), the range is [0 360] degrees. The toggling of yaw angle between the value 0 degree and 360 degree is expected as both value represents same orientation.

- In BSXlite default convention, when Pitch angle crosses +/-135 degrees, heading jumps by 180 degrees in accordance with the switch in direction for heading
- In BSXlite default convention, roll rotation range is [-90, +90]. To indicate rotations exceeding 90 degrees in roll, the pitch is switched by +/-180 degrees. This is required as the X-axis gets reversed in direction with roll rotation exceeding 90 degrees

Expected results for orientation vector (in degrees) are tabulated below. The orientation output provided by the BSX engine is given in radians. Refer the tables in this section after converting to degrees. The angles should not have a deviation exceeding +/- 5 degrees from the actual expected value. The tables below show the ideal expected value.

5.3.1 Yaw rotations

S. No	Test Angle (Yaw)	Yaw (Heading)	Pitch	Roll
1	0	0	0	0
2	90	90	0	0
3	180	180	0	0
4	270	270	0	0

Table 17: Orientation : Yaw rotations

5.3.2 Pitch rotations

S. No	Test Angle (Pitch)	Yaw (Heading)	Pitch	Roll
1	0	0	0	0
2	90	0	90	0
3	180	180	180 or -180	0
4	270	0	-90	0

Table 18: Orientation : Pitch rotations

5.3.3 Roll rotations

S. No	Test Angle (Roll)	Yaw (Heading)	Pitch	Roll
1	0	0	0	0
2	90	0	0 or 180/-180	90
3	180	0	180 or -180	0
4	270	0	0 or 180/-180	-90

Table 19: Orientation : Roll rotations

6 APPLICATION EXAMPLE AND HINTS

With respect to any examples or hints given herein, any typical values stated herein and/or any information regarding the application of the device, Bosch Sensortec hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights or copyrights of any third party. The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics. They are provided for illustrative purposes only and no evaluation regarding infringement of intellectual property rights or copyrights or regarding functionality, performance or error has been made.

7 LIST OF ABBREVIATIONS

Abbreviation	Description
BST	Bosch Sensortec GMBH
IMU	Inertial Measurement Unit
6-Dof	6 Degrees of Freedom
NDof, 9-DOF	Nine Degrees Of Freedom
ENU	East North Up Coordinate System
MEMS	Micro Electro Mechanical Systems

Table 20: List fo Abbreviations

8 REFERENCES

1. Bosch Sensortec Website http://www.bosch-sensortec.com
2. Android Sensors Overview http://developer.android.com/guide/topics/sensors/sensors_overview.html

9 APPENDIX

9.1 Overview of Sensors

This section will provide information about the sensors which are used and also provides basic functionalities of the sensors. The BSXlite solution utilizes only inertial sensors, accelerometer and gyroscope sensors.

9.1.1 Accelerometer

An accelerometer is an electromechanical device that measures the changes in acceleration. The measured acceleration signal is made up of the component of gravity (static acceleration) and linear acceleration in each of the axes. The direction of the gravitational force with respect to the device is used to calculate the orientation of the device. We assume a value of $1g = 9.80665m/s^2$.

For advanced signal fusion, high performance of the accelerometer is crucial, since its output contributes to the static accuracy of the orientation estimation. A high performance can be determined by checking the non-linearity of the measurement, e.g. bias and cross-axis sensitivity including their temperature dependency, hysteresis effects due to direction of a rotation, cross-coupling among different axis, rotation of the MEMS within the package, noise, electrical signal coupling, etc.

9.1.2 Gyroscope

In principle, a gyroscope is a device for measuring changes of the orientation, based on the principle of the angular momentum. MEMS gyroscopes are used for measuring the rate of rotation in space (roll, pitch and yaw), also referred to as 'angular rate'. The unit of measurement is rad/s.

A Gyroscope is immune with respect to linear accelerations and distortions affecting other motion measurement devices such as accelerometers and magnetometers. Hence, advanced fusion algorithms taking into account the angular rate can be used to be robust against magnetic distortion, achieve a fast update of outputs and provide accurate estimation of a device orientation while exposing the device dynamic motion.

9.2 Axis Remapping example

Samples of the raw motion signals provided to the BSXlite shall be vectors with components x, y, z. The samples of the raw signals received from the motion sensors should also exhibit the same order, x, y, z. If not they should be remapped accordingly. An example for axis remapping is given below, the host device orientation is defined by (x_d, y_d, z_d) and the accelerometer orientation is defined by (x_a, y_a, z_a).

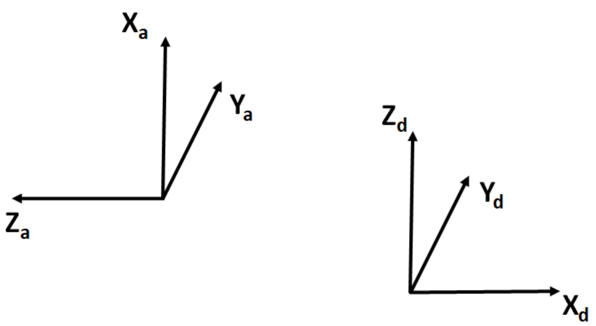

Fig. 3: Example for axis remapping

The remapping for the above example is to be done as:

$$acceleration\_x_d = -raw\_acceleration\_z_a$$
$$acceleration\_y_d = raw\_acceleration\_y_a$$
$$acceleration\_z_d = raw\_acceleration\_x_a$$

9.2.1 Verifying Axis Remapping

The below scenarios can be checked to verify whether the axis remapping for accelerometer and gyroscope with respect to the host device orientation frame was successful.

S.No	Test Scenario	Expected Result
1	X-axis perpendicular to the ground, pointing upwards	Input raw acceleration should be X: +9.81 m/s^2, Y= 0 m/s^2, Z = 0 m/s^2
2	Y-axis perpendicular to the ground, pointing upwards	Input raw acceleration should be X: 0 m/s^2, Y= +9.81 m/s^2, Z = 0 m/s^2
3	Z-axis perpendicular to the ground, pointing upwards	Input raw acceleration should be X: 0 m/s^2, Y= 0 m/s^2, Z = +9.81 m/s^2

Table 21: Verifying axis remapping for accelerometer

S.No	Test Scenario	Expected Result
1	Clockwise rotation around X-axis	Prominent negative signal observed in raw angular rate values in X-component alone
2	Counterclockwise rotation around X-axis	Prominent positive signal observed in raw angular rate values in X-component alone
3	Clockwise rotation around Y-axis	Prominent negative signal observed in raw angular rate values in Y-component alone
2	Counterclockwise rotation around Y-axis	Prominent positive signal observed in raw angular rate values in Y-component alone
5	Clockwise rotation around Z-axis	Prominent negative signals observed in raw angular rate values in Z-component alone
6	Counterclockwise rotation around Z-axis	Prominent positive signal observed in raw angular rate values in Z-component alone

Table 22: Verifying axis remapping for gyroscope

Bosch Sensortec GmbH
Gerhard-Kindler-Straße 9
72770 Reutlingen / Germany

www.bosch-sensortec.com
Tel.: +49 7121 3535-900
email: contact@bosch-sensortec.com

Modifications reserved | Printed in Germany
Preliminary - specifications subject to change without notice
Document number: BST-BSX-SD001-00
Version 1.0