

## 1. Introduction

This document describes the Tare function of the FSP200 that redefines the orientation of the sensor. Taring allows the outputs of the FSP200 to be in line with the orientation with which it was mounted into the main device. Commands in this documentation are described in more detail in the SH-2 Reference Manual <sup>[1]</sup> and operation requires SHTTP implementation. The FSP200 Example Code from GitHub (<https://github.com/hcrest/sh2-demo-nucleo>) provides examples application of SH-2 in source code compatible with FSP200. This document assumes that the reader has access to these materials for the reference.

## 2. Tare Overview

There are two types of “Tare” in the FSP200.

- Reorientation Tare: Allows the users to define the system orientation and changes all outputs.
- World (Re-center) Tare: Changes the direction of the world to the direction the user is facing.

There are two ways to tare in the FSP200.

- Commands:
  - Tare Now (Section 6.4.4.1 of SH-2 Reference Manual)
    - TareNow(Z): allows for a temporary tare in Z axis (heading) and performs “World Tare”. This is useful at the start of a game to adjust the current heading.
    - TareNow(XYZ): allows for a temporary tare in X, Y and Z axes and performs “Reorientation Tare”.
  - Persist Tare (Section 6.4.4.2 of SH-2 Reference Manual)
    - Saves the current Tare orientation into the System Orientation configuration set by TareNow command into a permanent record. This is a persistent setting that becomes the default system orientation when the device reboots. This should be applied only to the reorientation tare.
- Configuration Record (Section 5 of this document)
  - System Orientation (FRS Record ID 0x2D3E)
    - Stores the system orientation of the device. Once you know the standard orientation of your sensors in your device, you can use the same orientation for all devices using this record as a convenient starting point. More accurate per-device taring can be done using the Tare commands.
- Please check if there is no System Orientation record before you follow the Tare Example steps below. If there is a previous record, follow the steps in Section 4.2 to clear the record before proceeding.

## 3. Tare Examples

### 3.1. Set Device System Orientation (Reorientation Tare)

In this example, the FSP200 is installed in a device and the system orientation needs to be defined. This can be performed at the factory or by the user at first time setup. This will use a full tare across all 3 axes and save the new orientation into flash using the Persist Tare function.

- Power on the FSP200 and activate the Game Rotation Vector.
- Calibrate the accelerometer by positioning the device in 4-6 unique orientations and ensuring the device is stable for five seconds in each orientation. One way to think about this is to imagine the device is a cube and set it down on each face (upside down, right side up, right side, left side, on its front, and its back).
- Set the device down for five seconds in each orientation so that the gyroscope ZRO can calibrate as well.
- Position the device so that your desired direction is aligned with your device and make sure the device is level.
- Run the Tare Now command in all axes using `sh2_setTareNow (SH2_TARE_X| SH2_TARE_Y| SH2_TARE_Z, SH2_TARE_BASIS_GAMING_ROTATION_VECTOR)` from the example code. Section 6.4.4.1 in SH-2 Reference Manual can provide details of this command instructions.
- Once you confirm the orientation is what you expected, run the Persist Tare command using `sh2_persistTare()` from the example code. This will save the current settings into the System Orientation FRS configuration record (FRS record ID 0x2DE3). Now these settings will be the default orientation upon next device power up.

### 3.2. Set Current Forward Direction (World Tare)

In this example, the user is operating the device at their desk. They want to define the forward orientation before starting a game, so all that needs to be done is tare the Z-axis (yaw). The other axes, which define “level” do not need to be changed from their previously established configuration. This can be done with an on-screen GUI or button on the device.

- Ideally the sensors would be calibrated well by the following steps:
  - Power on the FSP200 and activate the Game Rotation Vector.
  - Set the device down for a few seconds so that the gyroscope ZRO is calibrated as well.
- Position the device into your desired forward direction orientation.
- Run the Tare Now command using `sh2_setTareNow (SH2_TARE_Z, SH2_TARE_BASIS_GAMING_ROTATION_VECTOR)` from the example code. Section 6.4.4.1 in SH-2 Reference Manual can provide details of this command instructions.

- After running this setting, the outputs of the Game Rotation Vector will be pointing to the user's forward direction. This setting will revert to the System Orientation FRS config record (that was set via the last Persist Tare command) the next time the device powers up.

### 3.3. How to Apply System Orientation (Reorientation Tare) to Multiple Devices at Production

It is likely that when using the FSP200 on a production device, the orientation of the sensor will be the same on many units. Rather than taring each individual unit, you can apply the device orientation settings to any number of additional units by using the FRS System Orientation configuration record. General steps for this are listed below.

- Run the full Tare procedure listed in above Section 3.1 (Reorientation Tare) on a "reference" device.
- Read the System Orientation FRS record from the reference device described in the following Section 4.1.
- Copy this FRS record to each device's System Orientation FRS record described in the following Section 4.3.
- Now these devices will have same System Orientation settings on the next reboot.

## 4. System Orientation Configuration

### 4.1. Get System Orientation Record:

The following code example shows how to get the System Orientation record (FRS ID 0x2D3E) from the device Flash Record System (FRS, See SH-2 Reference Manual Section 4.0<sup>[1]</sup>). As shipped, the FSP200's System Orientation record is empty.

```
uint32_t orientationFRS[4];
uint16_t lenFRS = 4; // Equal to the actual length in words of the specific FRS.
int rc;

memset(orientationFRS, 0, sizeof(orientationFRS));

rc = sh2_getFrs(0x2D3E, orientationFRS, &lenFRS);

if (rc != SH2_OK) {
    printf ("Failed to get System Orientation, rc = %d. \n", rc);
}
```

## 4.2. Clear System Orientation Record:

The following code example shows how to clear the System Orientation record (FRS ID 0x2D3E) from the device Flash Record System.

```
int rc;

uint32_t orientationFRS [4];
rc = sh2_setFrs(0x2D3E, orientationFRS, 0);

if (rc != SH2_OK) {
    printf("Failed to clear System Orientation, rc = %d. \n", rc);
}
```

## 4.3. Set System Orientation Record

The following code example shows how to set the System Orientation record (FRS ID 0x2D3E) in the device Flash Record System.

```
uint32_t orientationFRS[4];
int rc;

// 2's-complement fixed point number with a Q point of 30
orientationFRS[0] = 0x40000000; // x = 1
orientationFRS[1] = 0x00000000; // y = 0
orientationFRS[2] = 0x00000000; // z = 0
orientationFRS[3] = 0x00000000; // w = 0
rc = sh2_setFrs(0x2D3E, orientationFRS, 4);

if (rc != SH2_OK) {
    printf("Failed to set System Orientation, rc = %d. \n", rc);
}
```

You can get the System Orientation configuration from the full Tare procedure in the previous section or use the table in the next section.

## 5. System Orientation Setting

The FSP200 analyzes the data from the sensors. It manages the various sensors, reads data, and provides many outputs to the host. The FSP200 processes motion relative to a frame of reference. It has no knowledge of the orientation of sensors within the device and must be configured to conform to the desired frame of reference of the device. While the FSP200 is not necessarily used within Android systems, the frame of reference defined in the Android specification is used by the FSP200 software. In Android, the device frame of reference is represented as X axis horizontal, positive to the right, Y axis vertical, positive along the face of the device and Z axis positive towards the outside of the front face of the device. The frame of reference is shown in Figure 1.

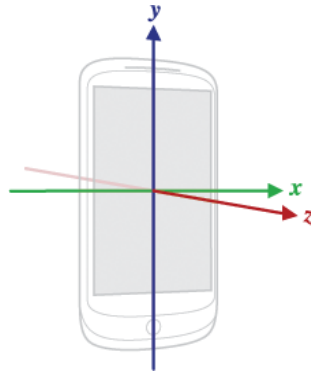


Figure 1: Android Co-ordinate System

The coordinate system for the FSP200 is determined by the placement and orientation of the connected sensor. If the sensor needs to be installed in an orientation different from the default coordinate system, the system orientation record can be modified to align with the desired coordinate system. Please refer to the FSP200 datasheet <sup>[2]</sup> Section 5 for the default orientation. When a sensor (e.g. BMI160) is placed as in Figure 2 and it is different from the device orientation, the System Orientation FRS record (ID 0x2D3E) is used to apply a rotation to the sensor outputs and all the derived outputs. The System Orientation record is a unit quaternion, with each coordinate represented as a 32-bit fixed point number with a Q-point of 30 to represent a fractional number. As in the example, where the sensor was mounted such that its positive Y-axis was aligned opposite to the X-axis of the device it was mounted in, but with its Z-axis aligned correctly, a clockwise rotation of 90° around the Z-axis would be required.

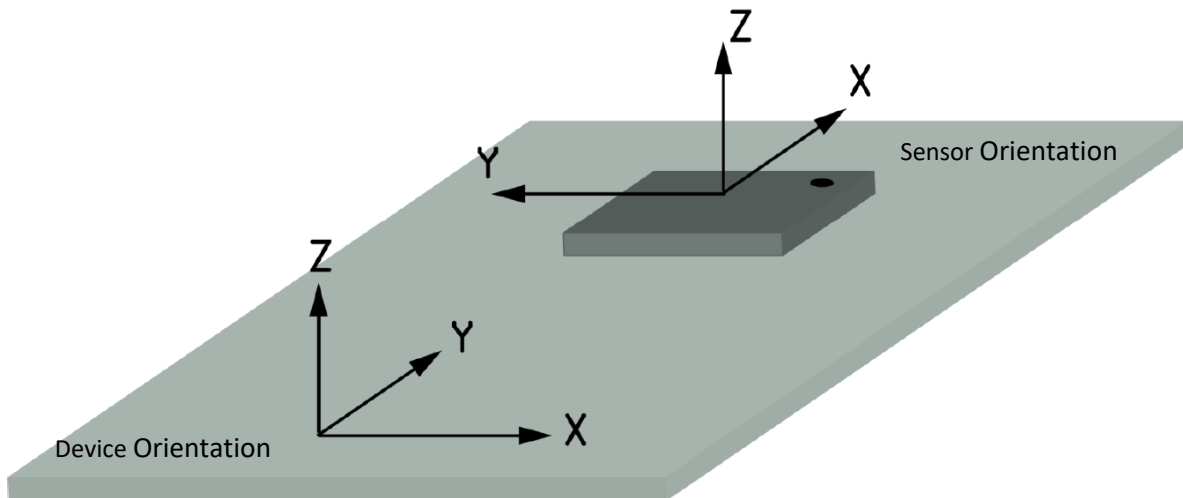


Figure 2: Example System Orientation on the Device

This rotation would be represented by a quaternion of  $(\frac{\sqrt{2}}{2}, 0, 0, \frac{\sqrt{2}}{2})$ .

Examples of mappings for sensor mounting are provided below to clarify the use of the FRS record. Assuming the device is placed in this configuration, the Z-axis is Up, the X-axis is East and the Y-axis is North and the sensor is positioned with the X-axis North, the Y-axis West and the Z-axis Up, then the rotation highlighted below will cause the sensor outputs to be aligned to the desired system outputs.

Sensor Axis			Mapping quaternion			
X	Y	Z	Qw	Qx	Qy	Qz
East	North	Up	1	0	0	0
North	West	Up	$(\sqrt{2})/2$	0	0	$(\sqrt{2})/2$
West	South	Up	0	0	0	1
South	East	Up	$(\sqrt{2})/2$	0	0	$-(\sqrt{2})/2$
East	South	Down	0	0	-1	0
North	East	Down	0	$-(\sqrt{2})/2$	$-(\sqrt{2})/2$	0
West	North	Down	0	-1	0	0
South	West	Down	0	$-(\sqrt{2})/2$	$(\sqrt{2})/2$	0
Up	South	East	0	0	$-(\sqrt{2})/2$	$(\sqrt{2})/2$
North	Up	East	1/2	-1/2	-1/2	1/2
Down	North	East	$(\sqrt{2})/2$	$-(\sqrt{2})/2$	0	0
South	Down	East	1/2	-1/2	1/2	-1/2
Up	North	West	$-(\sqrt{2})/2$	$-(\sqrt{2})/2$	0	0
North	Down	West	-1/2	-1/2	-1/2	-1/2
Down	South	West	0	0	$-(\sqrt{2})/2$	$-(\sqrt{2})/2$
South	Up	West	1/2	1/2	-1/2	-1/2
Up	East	North	-1/2	-1/2	-1/2	1/2
West	Up	North	0	$-(\sqrt{2})/2$	0	$(\sqrt{2})/2$
Down	West	North	1/2	-1/2	1/2	1/2
East	Down	North	$-(\sqrt{2})/2$	0	$-(\sqrt{2})/2$	0
Up	West	South	1/2	1/2	-1/2	1/2
West	Down	South	0	$-(\sqrt{2})/2$	0	$-(\sqrt{2})/2$
Down	East	South	1/2	-1/2	-1/2	-1/2
East	Up	South	$(\sqrt{2})/2$	0	$-(\sqrt{2})/2$	0

Table 1: System Orientation for Multiples of 90 Degree Rotations

## References

1. 1000-3625 SH-2 Reference Manual, Hillcrest Labs
2. 1000-4121 FSP200 Datasheet, Hillcrest Labs

## Notices

Information furnished by Hillcrest Laboratories, Inc. (Hillcrest) is believed to be accurate and reliable. However, Hillcrest assumes no responsibility for its use, nor for any infringement of patents or other rights of third parties that may result from its use.

Hillcrest reserves the right to make changes, corrections, modifications or improvements to this document at any time without notice. Information in this document supersedes and replaces all information previously supplied. Hillcrest makes no warranties, express or implied, regarding the information contained in this document.

Information in this document is provided solely to enable the use of Hillcrest products. "Typical" parameters provided by Hillcrest are not guaranteed, and can vary between applications and over time.

The product is designed primarily for use in consumer electronics and has not been evaluated for use in products or systems where failure or malfunction may result in personal injury, death, severe property damage or environmental damage, such as aerospace, life saving, life sustaining or military applications. Hillcrest assumes no liability for any claims or damages arising from information contained in this document, or from the use of products and services detailed herein. This exclusion includes, but is not limited to, claims or damages based on the infringement of patents, trademarks, copyrights and/or any other intellectual property rights. The product is provided by Hillcrest "As Is." Hillcrest makes no representations or warranties regarding the product express or implied, including without limitation any implied warranties as to title, noninfringement, merchantability, or fitness for a particular purpose.

Freespace is a registered trademark of Hillcrest Laboratories, Inc. The Hillcrest Labs logo is a trademark of Hillcrest Laboratories, Inc. All other trademarks and copyrights are the property of their respective owners.

Third party notices are available on our website at: <http://hillcrestlabs.com/legal/third-party-attributions/>