chanhvinh.com

life.augmented

# LSM6DSO32X

Datasheet

## iNEMO 6 DoF inertial module with 32 *g* accelerometer and embedded Machine Learning Core




## Features

- Power consumption: 0.55 mA in combo high-performance mode
- “Always-on" experience with low power consumption for both accelerometer and gyroscope
- Smart FIFO up to 9 kbytes
- Android compliant
- ±4/±8/±16/±32 *g* full scale
- ±125/±250/±500/±1000/±2000 dps full scale
- Analog supply voltage: 1.71 V to 3.6 V
- Independent IO supply (1.62 V)
- Compact footprint: 2.5 mm x 3 mm x 0.83 mm
- SPI / I²C & MIPI I3C$^{SM}$ serial interface with main processor data synchronization
- Advanced pedometer, step detector and step counter
- Significant Motion Detection, tilt detection
- Standard interrupts: free-fall, wakeup, 6D/4D orientation, click and double-click
- Programmable Finite State Machine: accelerometer, gyroscope and external sensors
- Machine Learning Core
- Embedded temperature sensor
- ECOPACK, RoHS and “Green” compliant

| Product status link |
|---|
| LSM6DSO32X |

| Product summary | |
|---|---|
| **Order code** | LSM6DSO32XTR |
| **Temperature range [°C]** | -40 to +85 |
| **Package** | LGA-14L (2.5 x 3.0 x 0.83 mm) |
| **Packing** | Tape & Reel |

**Product label**

SUSTAINABLE TECHNOLOGY

## Applications

- Wearables
- Smart watches
- Sports equipment
- Motion tracking and gesture detection
- Hard-fall detection
- Sensor hub
- Navigation and asset tracking
- IoT and connected devices
- Smart power saving for handheld devices

## Description

The LSM6DSO32X is a system-in-package featuring a 3D digital accelerometer at 32 *g* and a 3D digital gyroscope, boosting power performance at 0.55 mA in high-performance mode and enabling always-on low-power features for an optimal motion experience in wearable, hard-fall detection, navigation and asset tracking applications.

chanhvinh.com



The LSM6DSO32X supports main OS requirements, offering real, virtual and batch sensors with 9 kbytes for dynamic data batching. ST's family of MEMS sensor modules leverages the robust and mature manufacturing processes already used for the production of micromachined accelerometers and gyroscopes. The various sensing elements are manufactured using specialized micromachining processes, while the IC interfaces are developed using CMOS technology that allows the design of a dedicated circuit which is trimmed to better match the characteristics of the sensing element.

The LSM6DSO32X embeds a dedicated core for Machine Learning processing (MLC) and a Finite State Machine (FSM) that provides system flexibility, allowing some algorithms run in the application processor to be moved to the MEMS sensor with the advantage of consistent reduction in power consumption.

The LSM6DSO32X has a full-scale acceleration range of ±4/±8/±16/±32 *g* and an angular rate range of ±125/±250/±500/±1000/±2000 dps.

High robustness to mechanical shock makes the LSM6DSO32X the preferred choice of system designers for the creation and manufacturing of reliable products. The LSM6DSO32X is available in a plastic land grid array (LGA) package.

chanhvinh.com



# 1 Overview

The LSM6DSO32X is a system-in-package featuring a high-performance 3-axis digital accelerometer and 3-axis digital gyroscope.

The LSM6DSO32X delivers best-in-class motion sensing that can detect orientation and gestures in order to empower application developers and consumers with features and capabilities that are more sophisticated than simply orienting their devices to portrait and landscape mode.

The event-detection interrupts enable efficient and reliable motion tracking and contextual awareness, implementing hardware recognition of free-fall events, 6D orientation, click and double-click sensing, activity or inactivity, stationary/motion detection and wakeup events.

The LSM6DSO32X supports main OS requirements, offering real, virtual and batch mode sensors. In addition, the LSM6DSO32X can efficiently run the sensor-related features specified in Android, saving power and enabling faster reaction time. In particular, the LSM6DSO32X has been designed to implement hardware features such as significant motion detection, stationary/motion detection, tilt, pedometer functions, timestamping and to support the data acquisition of an external magnetometer.

The LSM6DSO32X offers hardware flexibility to connect the pins with different mode connections to external sensors to expand functionalities such as adding a sensor hub, etc.

Up to 9 kbytes of FIFO with compression and dynamic allocation of significant data (i.e. external sensors, timestamp, etc.) allows overall power saving of the system.

Like the entire portfolio of MEMS sensor modules, the LSM6DSO32X leverages the robust and mature in-house manufacturing processes already used for the production of micromachined accelerometers and gyroscopes. The various sensing elements are manufactured using specialized micromachining processes, while the IC interfaces are developed using CMOS technology that allows the design of a dedicated circuit which is trimmed to better match the characteristics of the sensing element.

The LSM6DSO32X is available in a small plastic land grid array (LGA) package of 2.5 x 3.0 x 0.83 mm to address ultra-compact solutions.

chanhvinh.com



# 2 Embedded low-power features

The LSM6DSO32X has been designed to be fully compliant with Android, featuring the following on-chip functions:

- 9 kbytes data buffering, data can be compressed two or three times
  - 100% efficiency with flexible configurations and partitioning
  - Possibility to store timestamp
- Event-detection interrupts (fully configurable)
  - Free-fall
  - Wakeup
  - 6D orientation
  - Click and double-click sensing
  - Activity/Inactivity recognition
  - Stationary/Motion detection
- Specific IP blocks with negligible power consumption and high-performance
  - Pedometer functions: step detector and step counters
  - Tilt
  - Significant Motion Detection
  - Finite State Machine (FSM) for accelerometer, gyroscope, and external sensors
  - Machine Learning Core (MLC)
- Sensor hub
  - Up to 6 total sensors: 2 internal (accelerometer and gyroscope) and 4 external sensors

## 2.1 Tilt detection

The tilt function helps to detect activity change and has been implemented in hardware using only the accelerometer to achieve targets of both ultra-low power consumption and robustness during the short duration of dynamic accelerations.

The tilt function is based on a trigger of an event each time the device's tilt changes and can be used with different scenarios, for example:

1. Triggers when phone is in a front pants pocket and the user goes from sitting to standing or standing to sitting;
2. Doesn’t trigger when phone is in a front pants pocket and the user is walking, running or going upstairs.

## 2.2 Significant Motion Detection

The Significant Motion Detection (SMD) function generates an interrupt when a ‘significant motion’, that could be due to a change in user location, is detected. In the LSM6DSO32X device this function has been implemented in hardware using only the accelerometer.

SMD functionality can be used in location-based applications in order to receive a notification indicating when the user is changing location.

## 2.3 Finite State Machine

The LSM6DSO32X can be configured to generate interrupt signals activated by user-defined motion patterns. To do this, up to 16 embedded finite state machines can be programmed independently for motion detection such as glance gestures, absolute wrist tilt, shake and double-shake detection.

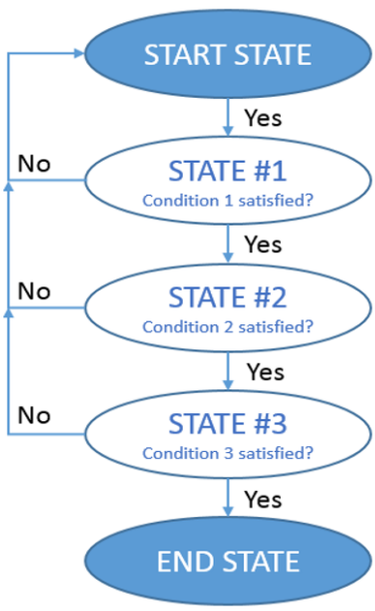
chanhvinh.com



### Definition of Finite State Machine

A state machine is a mathematical abstraction used to design logic connections. It is a behavioral model composed of a finite number of states and transitions between states, similar to a flow chart in which one can inspect the way logic runs when certain conditions are met. The state machine begins with a start state, goes to different states through transitions dependent on the inputs, and can finally end in a specific state (called stop state). The current state is determined by the past states of the system. Figure 1. Generic state machine shows a generic state machine.

**Figure 1. Generic state machine**




### Finite State Machine in the LSM6DSO32X

The LSM6DSO32X works as a combo accelerometer-gyroscope sensor, generating acceleration and angular rate output data. It is also possible to connect an external sensor (magnetometer) by using the Sensor Hub feature (Mode 2). These data can be used as input of up to 16 programs in the embedded Finite State Machine (Figure 2. State machine in the LSM6DSO32X).

All 16 finite state machines are independent: each one has its dedicated memory area and it is independently executed. An interrupt is generated when the end state is reached or when some specific command is performed.

**Figure 2. State machine in the LSM6DSO32X**



chanhvinh.com



## 2.4 Machine Learning Core

The LSM6DSO32X embeds a dedicated core for machine learning processing that provides system flexibility, allowing some algorithms run in the application processor to be moved to the MEMS sensor with the advantage of consistent reduction in power consumption.

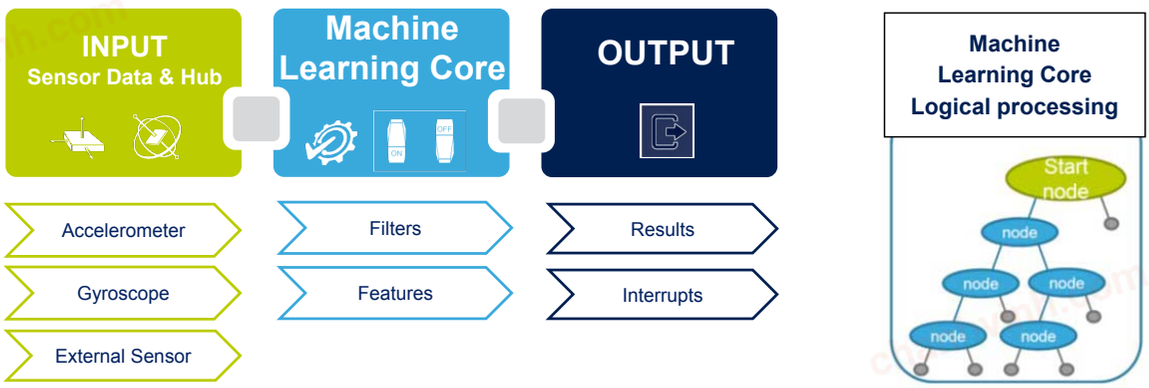
Machine Learning Core logic allows identifying if a data pattern (for example motion, pressure, temperature, magnetic data, etc.) matches a user-defined set of classes. Typical examples of applications could be activity detection like running, walking, driving, etc.

The LSM6DSO32X Machine Learning Core works on data patterns coming from the accelerometer and gyro sensors, but it is also possible to connect and process external sensor data (like magnetometer) by using the Sensor Hub feature (Mode 2).

The input data can be filtered using a dedicated configurable computation block containing filters and features computed in a fixed time window defined by the user.

Machine learning processing is based on logical processing composed of a series of configurable nodes characterized by "if-then-else" conditions where the "feature" values are evaluated against defined thresholds.

**Figure 3. Machine Learning Core in the LSM6DSO32X**




The LSM6DSO32X can be configured to run up to 8 flows simultaneously and independently and every flow can generate up to 16 results. The total number of nodes can be up to 256.

The results of the machine learning processing are available in dedicated output registers readable from the application processor at any time.

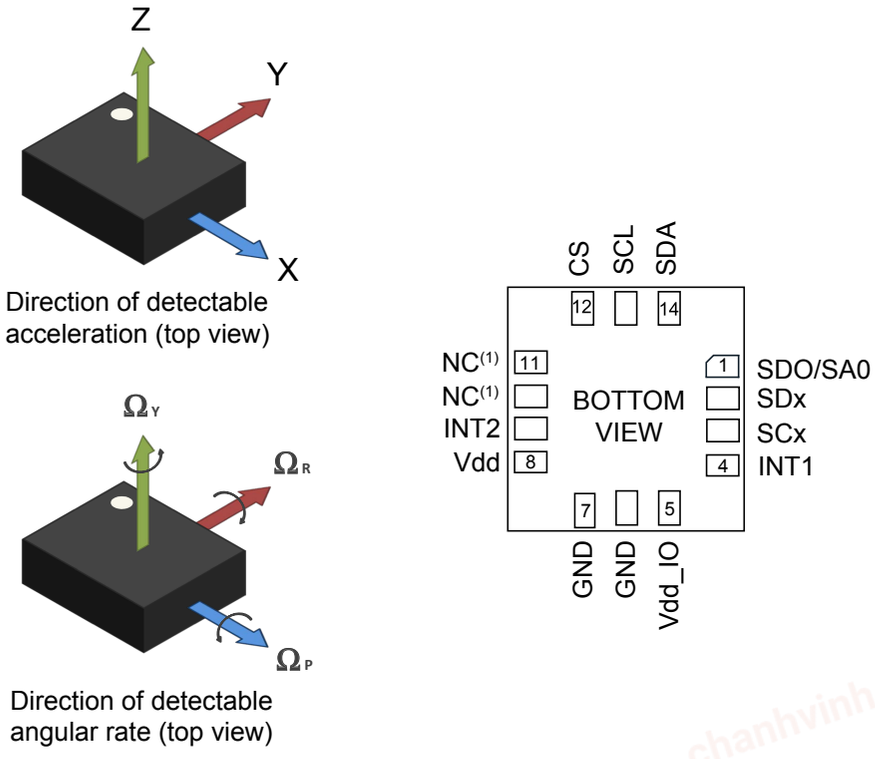
The LSM6DSO32X Machine Learning Core can be configured to generate an interrupt when a change in the result occurs.

chanhvinh.com



# 3 Pin description

Figure 4. **Pin connections**




1. Leave pin electrically unconnected and soldered to PCB.

chanhvinh.com

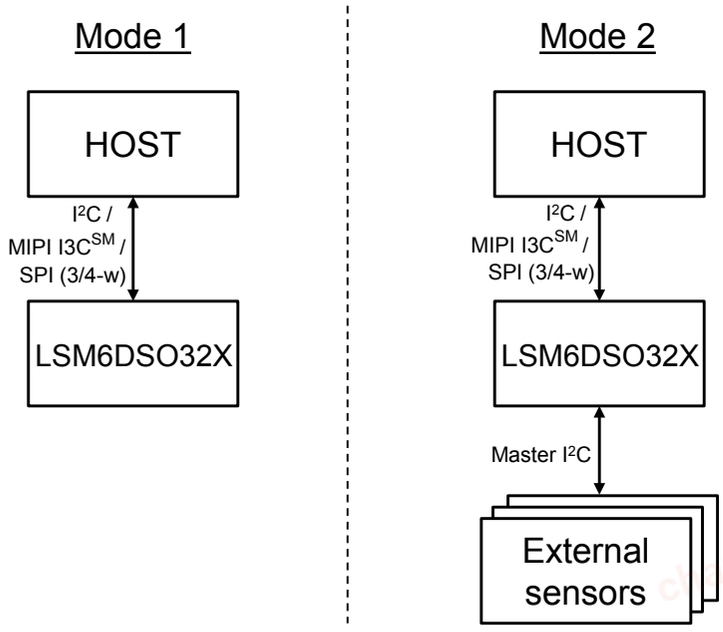


## 3.1 Pin connections

The LSM6DSO32X offers flexibility to connect the pins in order to have two different mode connections and functionalities. In detail:

- **Mode 1**: I²C / MIPI I3C$^{SM}$ slave interface or SPI (3- and 4-wire) serial interface is available;
- **Mode 2**: I²C / MIPI I3C$^{SM}$ slave interface or SPI (3- and 4-wire) serial interface and I²C interface master for external sensor connections are available;

**Figure 5. LSM6DSO32X connection modes**




In the following table each mode is described for the pin connections and function.

chanhvinh.com



**Table 1. Pin description**

| Pin# | Name | Mode 1 function | Mode 2 function |
|---|---|---|---|
| 1 | SDO | SPI 4-wire interface serial data output (SDO) | SPI 4-wire interface serial data output (SDO) |
| | SA0 | I²C least significant bit of the device address (SA0)<br>MIPI I3C$^{SM}$ least significant bit of the static address (SA0) | I²C least significant bit of the device address (SA0)<br>MIPI I3C$^{SM}$ least significant bit of the static address (SA0) |
| 2 | SDx | Connect to Vdd_IO or GND | I²C serial data master (MSDA) |
| 3 | SCx | Connect to Vdd_IO or GND | I²C serial clock master (MSCL) |
| 4 | INT1 | Programmable interrupt 1 / If device is used as MIPI I3C$^{SM}$ pure slave, this pin must be set to '1'. | Programmable interrupt 1 / If device is used as MIPI I3C$^{SM}$ pure slave, this pin must be set to '1'. |
| 5 | Vdd_IO(1) | Power supply for I/O pins | |
| 6 | GND | 0 V supply | |
| 7 | GND | 0 V supply | |
| 8 | Vdd(1) | Power supply | |
| 9 | INT2 | Programmable interrupt 2 (INT2) / Data enable (DEN) | Programmable interrupt 2 (INT2) / Data enable (DEN) /<br>I²C master external synchronization signal (MDRDY) |
| 10 | NC | Connect to Vdd_IO or leave unconnected(2) | Connect to Vdd_IO or leave unconnected(2) |
| 11 | NC | Connect to Vdd_IO or leave unconnected(2) | Connect to Vdd_IO or leave unconnected(2) |
| 12 | CS | I²C/MIPI I3C$^{SM}$/SPI mode selection<br>(1: SPI idle mode / I²C/MIPI I3C$^{SM}$ communication enabled;<br>0: SPI communication mode / I²C/MIPI I3C$^{SM}$ disabled) | I²C/MIPI I3C$^{SM}$/SPI mode selection<br>(1: SPI idle mode / I²C/MIPI I3C$^{SM}$ communication enabled;<br>0: SPI communication mode / I²C/MIPI I3C$^{SM}$ disabled) |
| 13 | SCL | I²C/MIPI I3C$^{SM}$ serial clock (SCL)<br>SPI serial port clock (SPC) | I²C/MIPI I3C$^{SM}$ serial clock (SCL)<br>SPI serial port clock (SPC) |
| 14 | SDA | I²C/MIPI I3C$^{SM}$ serial data (SDA)<br>SPI serial data input (SDI)<br>3-wire interface serial data output (SDO) | I²C/MIPI I3C$^{SM}$ serial data (SDA)<br>SPI serial data input (SDI)<br>3-wire interface serial data output (SDO) |

1. *Recommended 100 nF filter capacitor.*
2. *Leave pin electrically unconnected and soldered to PCB.*

chanhvinh.com



# 4 Module specifications

## 4.1 Mechanical characteristics

@ Vdd = 1.8 V, T = 25 °C, unless otherwise noted.

**Table 2. Mechanical characteristics**

| Symbol | Parameter | Test conditions | Min. | Typ.(1) | Max. | Unit |
|---|---|---|---|---|---|---|
| LA_FS | Linear acceleration measurement range | | | ±4 | | *g* |
| | | | | ±8 | | |
| | | | | ±16 | | |
| | | | | ±32 | | |
| G_FS | Angular rate measurement range | | | ±125 | | dps |
| | | | | ±250 | | |
| | | | | ±500 | | |
| | | | | ±1000 | | |
| | | | | ±2000 | | |
| LA_So | Linear acceleration sensitivity(2) | FS = ±4 *g* | | 0.122 | | mg/LSB |
| | | FS = ±8 *g* | | 0.244 | | |
| | | FS = ±16 *g* | | 0.488 | | |
| | | FS = ±32 *g* | | 0.976 | | |
| G_So | Angular rate sensitivity(2) | FS = ±125 dps | | 4.375 | | mdps/LSB |
| | | FS = ±250 dps | | 8.75 | | |
| | | FS = ±500 dps | | 17.50 | | |
| | | FS = ±1000 dps | | 35 | | |
| | | FS = ±2000 dps | | 70 | | |
| LA_So% | Linear acceleration sensitivity tolerance(3) | at component level | | ±0.5 | | % |
| G_So% | Angular rate sensitivity tolerance(3) | at component level | | ±0.5 | | % |
| LA_SoDr | Linear acceleration sensitivity change vs. temperature(4) | from -40° to +85° | | ±0.007 | | %/°C |
| G_SoDr | Angular rate sensitivity change vs. temperature(4) | from -40° to +85° | | ±0.005 | | %/°C |
| LA_TyOff | Linear acceleration zero-*g* level offset accuracy(5) | | | ±20 | | mg |
| G_TyOff | Angular rate zero-rate level(5) | | | ±0.5 | | dps |
| LA_OffDr | Linear acceleration zero-*g* level change vs. temperature(4) | | | ±0.1 | | mg/ °C |
| G_OffDr | Angular rate typical zero-rate level change vs. temperature(4) | | | ±0.01 | | dps/°C |
| Rn | Rate noise density in high-performance mode(6) | | | 3.8 | | mdps/√Hz |
| RnRMS | Gyroscope RMS noise in normal/low-power mode(7) | | | 75 | | mdps |
| An | Acceleration noise density in high-performance mode(8) | FS = ±4 *g* | | 120 | | µg/√Hz |
| | | FS = ±8 *g* | | 130 | | |
| | | FS = ±16 *g* | | 160 | | |
| | | FS = ±32 *g* | | 220 | | |

chanhvinh.com



| Symbol | Parameter | Test conditions | Min. | Typ.(1) | Max. | Unit |
|---|---|---|---|---|---|---|
| RMS | Acceleration RMS noise in normal/low-power mode(9)(10) | FS = ±4 *g* | | 3.2 | | mg(RMS) |
| | | FS = ±8 *g* | | 3.4 | | |
| | | FS = ±16 *g* | | 4.0 | | |
| | | FS = ±32 *g* | | 5.4 | | |
| | Acceleration RMS noise in ultra-low-power mode(9)(10) | FS = ±4 *g* | | 9.5 | | |
| LA_ODR | Linear acceleration output data rate | | | 1.6(11)<br>12.5<br>26<br>52<br>104<br>208<br>416<br>833<br>1666<br>3332<br>6664 | | Hz |
| G_ODR | Angular rate output data rate | | | 12.5<br>26<br>52<br>104<br>208<br>416<br>833<br>1666<br>3332<br>6664 | | |
| Vst | Linear acceleration self-test output change(12)(13)(14) | | 50 | | 1700 | mg |
| | Angular rate self-test output change(15)(16) | FS = ±250 dps | 20 | | 80 | dps |
| | | FS = ±2000 dps | 150 | | 700 | dps |
| Top | Operating temperature range | | -40 | | +85 | °C |

1. *Typical specifications are not guaranteed.*
2. *Sensitivity values after factory calibration test and trimming.*
3. *Subject to change.*
4. *Measurements are performed in a uniform temperature setup and they are based on characterization data in a limited number of samples. Not measured during final test for production.*
5. *Values after factory calibration test and trimming.*
6. *Gyroscope rate noise density in high-performance mode is independent of the ODR and FS setting.*
7. *Gyroscope RMS noise in normal/low-power mode is independent of the ODR and FS setting.*
8. *Accelerometer noise density in high-performance mode is independent of the ODR.*
9. *Accelerometer RMS noise in normal/low-power/ultra-low-power mode is independent of the ODR.*
10. *Noise RMS related to BW = ODR/2.*
11. *This ODR is available when the accelerometer is in low-power mode.*
12. *The sign of the linear acceleration self-test output change is defined by the STx_XL bits in a dedicated register for all axes.*

chanhvinh.com



13. *The linear acceleration self-test output change is defined with the device in stationary condition as the absolute value of: OUTPUT[LSb] (self-test enabled) - OUTPUT[LSb] (self-test disabled). 1LSb = 0.122 mg at ±4 g full scale.*
14. *Accelerometer self-test limits are full-scale independent.*
15. *The sign of the angular rate self-test output change is defined by the STx_G bits in a dedicated register for all axes.*
16. *The angular rate self-test output change is defined with the device in stationary condition as the absolute value of: OUTPUT[LSb] (self-test enabled) - OUTPUT[LSb] (self-test disabled). 1LSb = 70 mdps at ±2000 dps full scale.*

chanhvinh.com

chanhvinh.com



## 4.2 Electrical characteristics

@ Vdd = 1.8 V, T = 25 °C, unless otherwise noted.

**Table 3. Electrical characteristics**

| Symbol | Parameter | Test conditions | Min. | Typ.(1) | Max. | Unit |
|---|---|---|---|---|---|---|
| Vdd | Supply voltage | | 1.71 | 1.8 | 3.6 | V |
| Vdd_IO | Power supply for I/O | | 1.62 | | 3.6 | V |
| IddHP | Gyroscope and accelerometer current consumption in high-performance mode | | | 0.55 | | mA |
| LA_IddHP | Accelerometer current consumption in high-performance mode | | | 170 | | µA |
| LA_IddLP | Accelerometer current consumption in low-power mode | ODR = 52 Hz<br>ODR = 1.6 Hz | | 26<br>4.5 | | µA |
| LA_IddULP | Accelerometer current consumption in ultra-low-power mode | ODR = 52 Hz<br>ODR = 1.6 Hz | | 9.5<br>4.4 | | µA |
| IddPD | Gyroscope and accelerometer current consumption during power-down | | | 3 | | µA |
| Ton | Turn-on time | | | 35 | | ms |
| $V_{IH}$ | Digital high-level input voltage | | 0.7 * Vdd_IO | | | V |
| $V_{IL}$ | Digital low-level input voltage | | | | 0.3 * Vdd_IO | V |
| $V_{OH}$ | High-level output voltage | $I_{OH}$ = 4 mA(2) | Vdd_IO - 0.2 | | | V |
| $V_{OL}$ | Low-level output voltage | $I_{OL}$ = 4 mA(2) | | | 0.2 | V |
| Top | Operating temperature range | | -40 | | +85 | °C |

1. *Typical specifications are not guaranteed.*
2. *4 mA is the maximum driving capability, i.e. the maximum DC current that can be sourced/sunk by the digital pin in order to guarantee the correct digital output voltage levels $V_{OH}$ and $V_{OL}$.*

## 4.3 Temperature sensor characteristics

@ Vdd = 1.8 V, T = 25 °C unless otherwise noted.

**Table 4. Temperature sensor characteristics**

| Symbol | Parameter | Test condition | Min. | Typ.(1) | Max. | Unit |
|---|---|---|---|---|---|---|
| TODR(2) | Temperature refresh rate | | | 52 | | Hz |
| Toff | Temperature offset(3) | | -15 | | +15 | °C |
| TSen | Temperature sensitivity | | | 256 | | LSB/°C |
| TST | Temperature stabilization time(4) | | | | 500 | µs |
| T_ADC_res | Temperature ADC resolution | | | 16 | | bit |
| Top | Operating temperature range | | -40 | | +85 | °C |

1. *Typical specifications are not guaranteed.*
2. *When the accelerometer is in low-power mode or ultra-low-power mode and the gyroscope part is turned off, the TODR value is equal to the accelerometer ODR.*
3. *The output of the temperature sensor is 0 LSB (typ.) at 25 °C.*
4. *Time from power ON to valid data based on characterization data.*

chanhvinh.com



## 4.4 Communication interface characteristics

### 4.4.1 SPI - serial peripheral interface

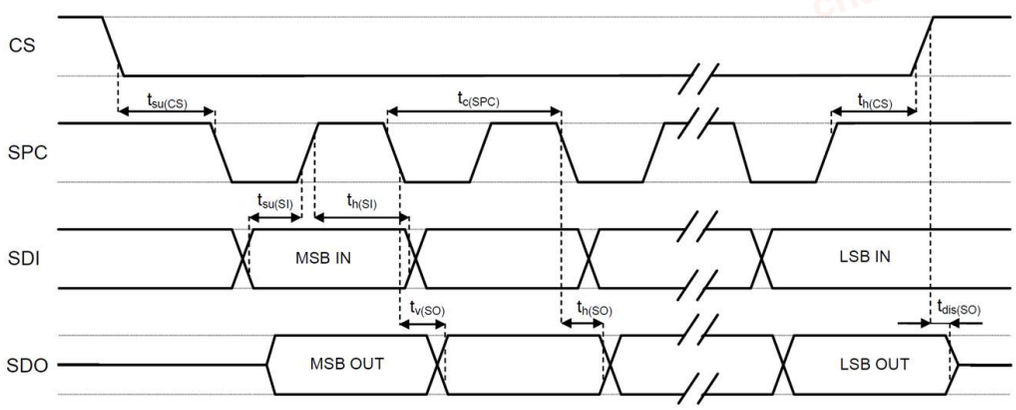
Subject to general operating conditions for Vdd and Top.

**Table 5. SPI slave timing values (in mode 3)**

| Symbol | Parameter | Value[1] | | Unit |
|---|---|---|---|---|
| | | Min | Max | |
| $t_{c(SPC)}$ | SPI clock cycle | 100 | | ns |
| $f_{c(SPC)}$ | SPI clock frequency | | 10 | MHz |
| $t_{su(CS)}$ | CS setup time | 5 | | ns |
| $t_{h(CS)}$ | CS hold time | 20 | | |
| $t_{su(SI)}$ | SDI input setup time | 5 | | |
| $t_{h(SI)}$ | SDI input hold time | 15 | | |
| $t_{v(SO)}$ | SDO valid output time | | 50 | |
| $t_{h(SO)}$ | SDO output hold time | 5 | | |
| $t_{dis(SO)}$ | SDO output disable time | | 50 | |

1. *Values are guaranteed at 10 MHz clock frequency for SPI with both 4 and 3 wires, based on characterization results, not tested in production.*

**Figure 6. SPI slave timing diagram (in mode 3)**




*Note: Measurement points are done at 0.3·Vdd_IO and 0.7·Vdd_IO for both input and output ports.*

chanhvinh.com



### 4.4.2 I²C - inter-IC control interface

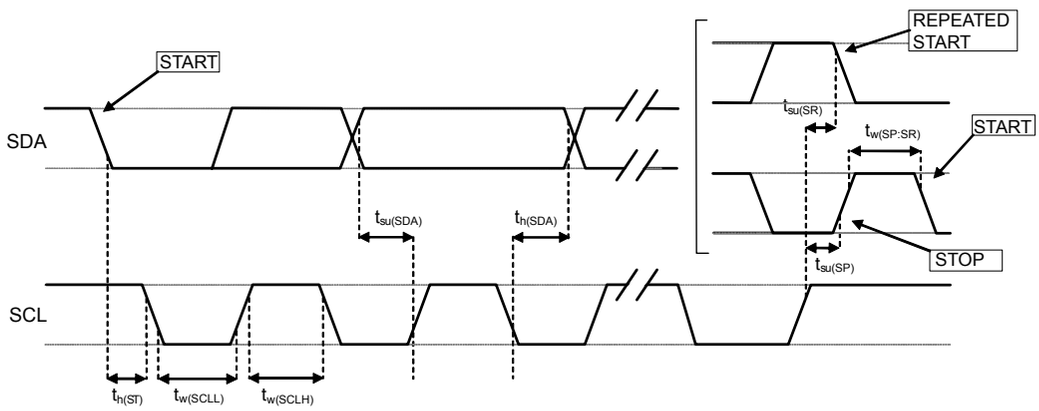
Subject to general operating conditions for Vdd and Top.

**Table 6. I²C slave timing values**

| Symbol | Parameter | I²C fast mode(1)(2) | | I²C fast mode +(1)(2) | | Unit |
|---|---|---|---|---|---|---|
| | | Min | Max | Min | Max | |
| $f_{(SCL)}$ | SCL clock frequency | 0 | 400 | 0 | 1000 | kHz |
| $t_{w(SCLL)}$ | SCL clock low time | 1.3 | | 0.5 | | µs |
| $t_{w(SCLH)}$ | SCL clock high time | 0.6 | | 0.26 | | |
| $t_{su(SDA)}$ | SDA setup time | 100 | | 50 | | ns |
| $t_{h(SDA)}$ | SDA data hold time | 0 | 0.9 | 0 | | µs |
| $t_{h(ST)}$ | START/REPEATED START condition hold time | 0.6 | | 0.26 | | |
| $t_{su(SR)}$ | REPEATED START condition setup time | 0.6 | | 0.26 | | |
| $t_{su(SP)}$ | STOP condition setup time | 0.6 | | 0.26 | | |
| $t_{w(SP:SR)}$ | Bus free time between STOP and START condition | 1.3 | | 0.5 | | |
| | Data valid time | | 0.9 | | 0.45 | |
| | Data valid acknowledge time | | 0.9 | | 0.45 | |
| $C_B$ | Capacitive load for each bus line | | 400 | | 550 | pF |

1. *Data based on standard I²C protocol requirement, not tested in production.*
2. *Data for I²C fast mode and I²C fast mode + have been validated by characterization, not tested in production.*

**Figure 7. I²C slave timing diagram**




*Note: Measurement points are done at 0.3·Vdd_IO and 0.7·Vdd_IO for both ports.*

chanhvinh.com



## 4.5 Absolute maximum ratings

Stresses above those listed as “Absolute maximum ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device under these conditions is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

**Table 7. Absolute maximum ratings**

| Symbol | Ratings | Maximum value | Unit |
|---|---|---|---|
| Vdd / Vdd_IO | Supply voltage | -0.3 to 4.8 | V |
| $T_{STG}$ | Storage temperature range | -40 to +125 | °C |
| Sg | Acceleration *g* for 0.2 ms | 20,000 | *g* |
| ESD | Electrostatic discharge protection (HBM) | 2 | kV |
| Vin | Input voltage on any control pin (including CS, SCL/SPC, SDA/SDI/SDO, SDO/SA0) | -0.3 to Vdd_IO +0.3 | V |

*Note: Supply voltage on any pin should never exceed 4.8 V.*



This device is sensitive to mechanical shock, improper handling can cause permanent damage to the part.



This device is sensitive to electrostatic discharge (ESD), improper handling can cause permanent damage to the part.

chanhvinh.com



## 4.6 Terminology

### 4.6.1 Sensitivity

Linear acceleration sensitivity can be determined, for example, by applying 1 *g* acceleration to the device. Because the sensor can measure DC accelerations, this can be done easily by pointing the selected axis towards the ground, noting the output value, rotating the sensor 180 degrees (pointing towards the sky) and noting the output value again. By doing so, ±1 *g* acceleration is applied to the sensor. Subtracting the larger output value from the smaller one, and dividing the result by 2, leads to the actual sensitivity of the sensor. This value changes very little over temperature and over time. The sensitivity tolerance describes the range of sensitivities of a large number of sensors (see Table 2).

An angular rate gyroscope is a device that produces a positive-going digital output for counterclockwise rotation around the axis considered. Sensitivity describes the gain of the sensor and can be determined by applying a defined angular velocity to it. This value changes very little over temperature and time (see Table 2).

### 4.6.2 Zero-*g* and zero-rate level

Linear acceleration zero-*g* level offset (TyOff) describes the deviation of an actual output signal from the ideal output signal if no acceleration is present. A sensor in a steady state on a horizontal surface will measure 0 g on both the X-axis and Y-axis, whereas the Z-axis will measure 1 *g*. Ideally, the output is in the middle of the dynamic range of the sensor (content of OUT registers 00h, data expressed as 2's complement number). A deviation from the ideal value in this case is called zero-*g* offset.

Offset is to some extent a result of stress to MEMS sensor and therefore the offset can slightly change after mounting the sensor onto a printed circuit board or exposing it to extensive mechanical stress. Offset changes little over temperature, see "Linear acceleration zero-*g* level change vs. temperature" in Table 2. The zero-*g* level tolerance (TyOff) describes the standard deviation of the range of zero-g levels of a group of sensors.

Zero-rate level describes the actual output signal if there is no angular rate present. The zero-rate level of precise MEMS sensors is, to some extent, a result of stress to the sensor and therefore the zero-rate level can slightly change after mounting the sensor onto a printed circuit board or after exposing it to extensive mechanical stress. This value changes very little over temperature and time (see Table 2).

chanhvinh.com



# 5 Digital interfaces

## 5.1 I²C/SPI interface

The registers embedded inside the LSM6DSO32X may be accessed through both the I²C and SPI serial interfaces. The latter may be SW configured to operate either in 3-wire or 4-wire interface mode. The device is compatible with SPI modes 0 and 3.

The serial interfaces are mapped onto the same pins. To select/exploit the I²C interface, the CS line must be tied high (i.e connected to Vdd_IO).

**Table 8. Serial interface pin description**

| Pin name | Pin description |
|---|---|
| CS | SPI enable<br>I²C/SPI mode selection<br>(1: SPI idle mode / I²C communication enabled;<br>0: SPI communication mode / I²C disabled) |
| SCL/SPC | I²C Serial Clock (SCL)<br>SPI Serial Port Clock (SPC) |
| SDA/SDI/SDO | I²C Serial Data (SDA)<br>SPI Serial Data Input (SDI)<br>3-wire Interface Serial Data Output (SDO) |
| SDO/SA0 | SPI Serial Data Output (SDO)<br>I²C less significant bit of the device address |

### 5.1.1 I²C serial interface

The LSM6DSO32X I²C is a bus slave. The I²C is employed to write the data to the registers, whose content can also be read back.

The relevant I²C terminology is provided in the table below.

**Table 9. I²C terminology**

| Term | Description |
|---|---|
| Transmitter | The device which sends data to the bus |
| Receiver | The device which receives data from the bus |
| Master | The device which initiates a transfer, generates clock signals and terminates a transfer |
| Slave | The device addressed by the master |

There are two signals associated with the I²C bus: the serial clock line (SCL) and the Serial DAta line (SDA). The latter is a bidirectional line used for sending and receiving the data to/from the interface. Both the lines must be connected to Vdd_IO through external pull-up resistors. When the bus is free, both the lines are high.

The I²C interface is implemented with fast mode (400 kHz) I²C standards as well as with the standard mode.

In order to disable the I²C block, (I2C_disable) = 1 must be written in CTRL4_C (13h).

chanhvinh.com



#### 5.1.1.1 I²C operation

The transaction on the bus is started through a START (ST) signal. A START condition is defined as a HIGH to LOW transition on the data line while the SCL line is held HIGH. After this has been transmitted by the master, the bus is considered busy. The next byte of data transmitted after the start condition contains the address of the slave in the first 7 bits and the eighth bit tells whether the master is receiving data from the slave or transmitting data to the slave. When an address is sent, each device in the system compares the first seven bits after a start condition with its address. If they match, the device considers itself addressed by the master.

The Slave ADdress (SAD) associated to the LSM6DSO32X is 110101xb. The SDO/SA0 pin can be used to modify the less significant bit of the device address. If the SDO/SA0 pin is connected to the supply voltage, LSb is '1' (address 1101011b); else if the SDO/SA0 pin is connected to ground, the LSb value is '0' (address 1101010b). This solution permits to connect and address two different inertial modules to the same I²C bus.

Data transfer with acknowledge is mandatory. The transmitter must release the SDA line during the acknowledge pulse. The receiver must then pull the data line LOW so that it remains stable low during the HIGH period of the acknowledge clock pulse. A receiver which has been addressed is obliged to generate an acknowledge after each byte of data received.

The I²C embedded inside the LSM6DSO32X behaves like a slave device and the following protocol must be adhered to. After the start condition (ST) a slave address is sent, once a slave acknowledge (SAK) has been returned, an 8-bit sub-address (SUB) is transmitted. The increment of the address is configured by the CTRL3_C (12h) (IF_INC).

The slave address is completed with a Read/Write bit. If the bit is '1' (Read), a repeated START (SR) condition must be issued after the two sub-address bytes; if the bit is '0' (Write) the master will transmit to the slave with direction unchanged. Table 10 explains how the SAD+Read/Write bit pattern is composed, listing all the possible configurations.

**Table 10. SAD+Read/Write patterns**

| Command | SAD[6:1] | SAD[0] = SA0 | R/W | SAD+R/W |
|---|---|---|---|---|
| Read | 110101 | 0 | 1 | 11010101 (D5h) |
| Write | 110101 | 0 | 0 | 11010100 (D4h) |
| Read | 110101 | 1 | 1 | 11010111 (D7h) |
| Write | 110101 | 1 | 0 | 11010110 (D6h) |

**Table 11. Transfer when master is writing one byte to slave**

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Master | ST | SAD + W | | SUB | | DATA | | SP |
| Slave | | | SAK | | SAK | | SAK | |

**Table 12. Transfer when master is writing multiple bytes to slave**

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Master | ST | SAD + W | | SUB | | DATA | | DATA | | SP |
| Slave | | | SAK | | SAK | | SAK | | SAK | |

**Table 13. Transfer when master is receiving (reading) one byte of data from slave**

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Master | ST | SAD + W | | SUB | | SR | SAD + R | | | NMAK | SP |
| Slave | | | SAK | | SAK | | | SAK | DATA | | |

**Table 14. Transfer when master is receiving (reading) multiple bytes of data from slave**

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Master | ST | SAD+W | | SUB | | SR | SAD+R | | | MAK | | MAK | | NMAK | SP |
| Slave | | | SAK | | SAK | | | SAK | DATA | | DATA | | DATA | | |

chanhvinh.com



Data are transmitted in byte format (DATA). Each data transfer contains 8 bits. The number of bytes transferred per transfer is unlimited. Data is transferred with the Most Significant bit (MSb) first. If a slave receiver doesn't acknowledge the slave address (i.e. it is not able to receive because it is performing some real-time function) the data line must be left HIGH by the slave. The master can then abort the transfer. A LOW to HIGH transition on the SDA line while the SCL line is HIGH is defined as a STOP condition. Each data transfer must be terminated by the generation of a STOP (SP) condition.

In the presented communication format MAK is Master Acknowledge and NMAK is No Master Acknowledge.

chanhvinh.com
chanhvinh.com
chanhvinh.com
chanhvinh.com
chanhvinh.com

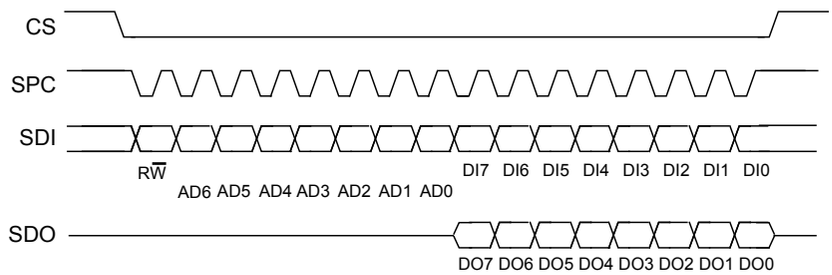
chanhvinh.com



### 5.1.2 SPI bus interface

The LSM6DSO32X SPI is a bus slave. The SPI allows writing and reading the registers of the device.
The serial interface communicates to the application using 4 wires: **CS**, **SPC**, **SDI** and **SDO**.

**Figure 8. Read and write protocol (in mode 3)**




**CS** is the serial port enable and it is controlled by the SPI master. It goes low at the start of the transmission and goes back high at the end. **SPC** is the serial port clock and it is controlled by the SPI master. It is stopped high when **CS** is high (no transmission). **SDI** and **SDO** are, respectively, the serial port data input and output. Those lines are driven at the falling edge of **SPC** and should be captured at the rising edge of **SPC**.

Both the read register and write register commands are completed in 16 clock pulses or in multiples of 8 in case of multiple read/write bytes. Bit duration is the time between two falling edges of **SPC**. The first bit (bit 0) starts at the first falling edge of **SPC** after the falling edge of **CS** while the last bit (bit 15, bit 23, ...) starts at the last falling edge of SPC just before the rising edge of **CS**.

**bit 0**: $R\overline{W}$ bit. When 0, the data DI(7:0) is written into the device. When 1, the data DO(7:0) from the device is read. In latter case, the chip will drive **SDO** at the start of bit 8.

**bit 1-7**: address AD(6:0). This is the address field of the indexed register.

**bit 8-15**: data DI(7:0) (write mode). This is the data that is written into the device (MSb first).

**bit 8-15**: data DO(7:0) (read mode). This is the data that is read from the device (MSb first).

In multiple read/write commands further blocks of 8 clock periods will be added. When the CTRL3_C (12h) (IF_INC) bit is '0', the address used to read/write data remains the same for every block. When the CTRL3_C (12h) (IF_INC) bit is '1', the address used to read/write data is increased at every block.

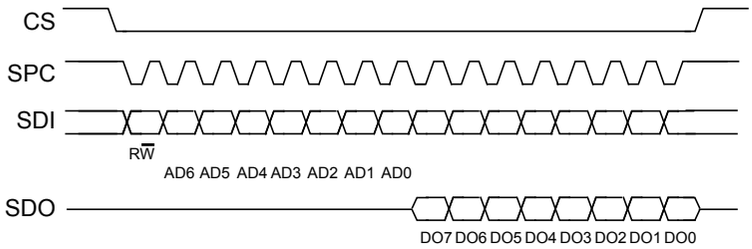
The function and the behavior of **SDI** and **SDO** remain unchanged.

chanhvinh.com



#### 5.1.2.1 SPI read

Figure 9. **SPI read protocol (in mode 3)**




The SPI Read command is performed with 16 clock pulses. A multiple byte read command is performed by adding blocks of 8 clock pulses to the previous one.

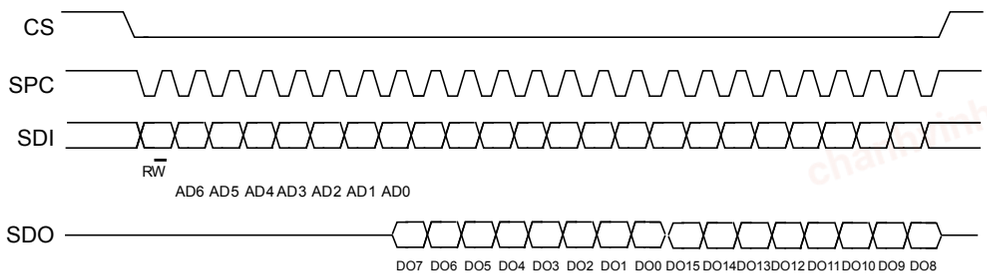
**bit 0**: READ bit. The value is 1.

**bit 1-7**: address AD(6:0). This is the address field of the indexed register.

**bit 8-15**: data DO(7:0) (read mode). This is the data that will be read from the device (MSb first).

**bit 16-...**: data DO(...-8). Further data in multiple byte reads.

Figure 10. **Multiple byte SPI read protocol (2-byte example) (in mode 3)**

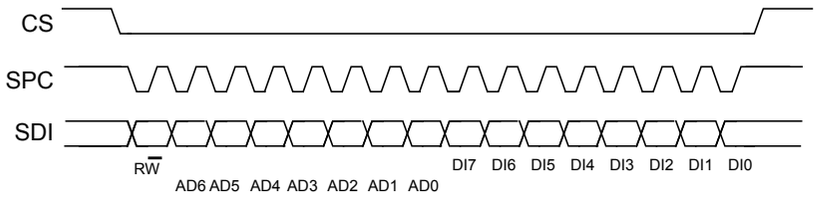


chanhvinh.com



#### 5.1.2.2 SPI write

Figure 11. **SPI write protocol (in mode 3)**




The SPI Write command is performed with 16 clock pulses. A multiple byte write command is performed by adding blocks of 8 clock pulses to the previous one.

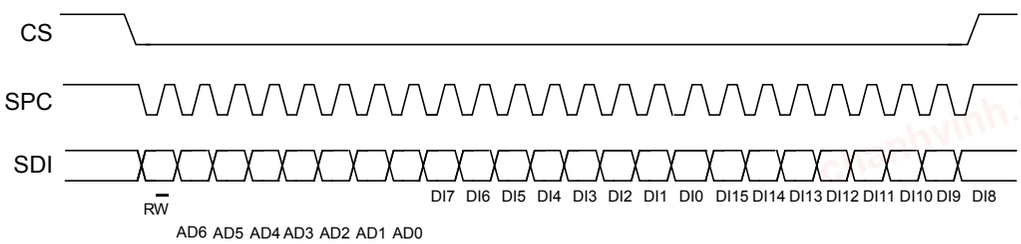
**bit 0**: WRITE bit. The value is 0.

**bit 1 -7**: address AD(6:0). This is the address field of the indexed register.

**bit 8-15**: data DI(7:0) (write mode). This is the data that is written inside the device (MSb first).

**bit 16-...** : data DI(...-8). Further data in multiple byte writes.

Figure 12. **Multiple byte SPI write protocol (2-byte example) (in mode 3)**



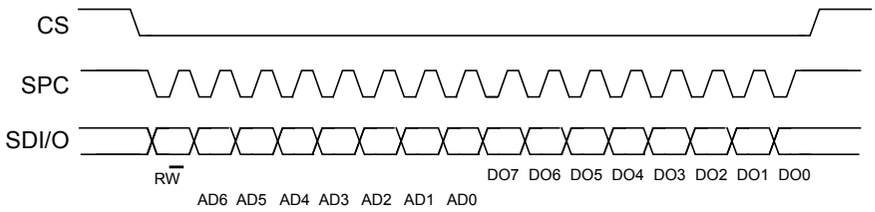
chanhvinh.com



#### 5.1.2.3 *SPI read in 3-wire mode*

3-wire mode is entered by setting the CTRL3_C (12h) (SIM) bit equal to '1' (SPI serial interface mode selection).

**Figure 13. SPI read protocol in 3-wire mode (in mode 3)**




The SPI read command is performed with 16 clock pulses:
**bit 0**: READ bit. The value is 1.
**bit 1-7**: address AD(6:0). This is the address field of the indexed register.
**bit 8-15**: data DO(7:0) (read mode). This is the data that is read from the device (MSb first).
A multiple read command is also available in 3-wire mode.

chanhvinh.com



## 5.2 MIPI I3C$^{SM}$ interface

### 5.2.1 MIPI I3C$^{SM}$ slave interface

The LSM6DSO32X interface includes a MIPI I3C$^{SM}$SDR only slave interface (compliant with release 1.0 of the specification) with MIPI I3C$^{SM}$ SDR embedded features:

- CCC command
- Direct CCC communication (SET and GET)
- Broadcast CCC communication
- Private communications
- Private read and write for single byte
- Multiple read and write
- In-Band Interrupt request

Error Detection and Recovery Methods (S0-S6)

*Note: Refer to Section 5.3 I²C/I3C coexistence in LSM6DSO32X for details concerning the choice of the interface when powering up the device.*

### 5.2.2 MIPI I3C$^{SM}$ CCC supported commands

The list of MIPI I3C$^{SM}$ CCC commands supported by the device is detailed in the following table.

**Table 15. MIPI I3C$^{SM}$ CCC commands**

| Command | Command code | Default | Description |
|---|---|---|---|
| ENTDAA | 0x07 | | DAA procedure |
| SETDASA | 0x87 | | Assign dynamic address using static address 0x6B/0x6A depending on SDO pin |
| ENEC | 0x80 / 0x00 | | Slave activity control (direct and broadcast) |
| DISEC | 0x81/ 0x01 | | Slave activity control (direct and broadcast) |
| ENTAS0 | 0x82 / 0x02 | | Enter activity state (direct and broadcast) |
| ENTAS1 | 0x83 / 0x03 | | Enter activity state (direct and broadcast) |
| ENTAS2 | 0x84 / 0x04 | | Enter activity state (direct and broadcast) |
| ENTAS3 | 0x85 / 0x05 | | Enter activity state (direct and broadcast) |
| SETXTIME | 0x98 / 0x28 | | Timing information exchange |
| GETXTIME | 0x99 | 0x07<br>0x00<br>0x05<br>0x92 | Timing information exchange |
| RSTDAA | 0x86 / 0x06 | | Reset the assigned dynamic address (direct and broadcast) |
| SETMWL | 0x89 / 0x08 | | Define maximum write length during private write (direct and broadcast) |
| SETMRL | 0x8A / 0x09 | | Define maximum read length during private read (direct and broadcast) |
| SETNEWDA | 0x88 | | Change dynamic address |
| GETMWL | 0x8B | 0x00<br>0x08<br>(2 byte) | Get maximum write length during private write |
| GETMRL | 0x8C | 0x00<br>0x10<br>0x09 | Get maximum read length during private read |

chanhvinh.com



| Command | Command code | Default | Description |
|---|---|---|---|
| | | (3 byte) | |
| GETPID | 0x8D | 0x02<br>0x08<br>0x00<br>0x6C<br>0x10<br>0x0B | Device ID register |
| GETBCR | 0x8E | 0x07<br>(1 byte) | Bus characteristics register |
| GETDCR | 0x8F | 0x44 default | MIPI I3C$^{SM}$ device characteristics register |
| GETSTATUS | 0x90 | 0x00<br>0x00<br>(2 byte) | Status register |
| GETMXDS | 0x94 | 0x00<br>0x20<br>(2 byte) | Return max data speed |

chanhvinh.com



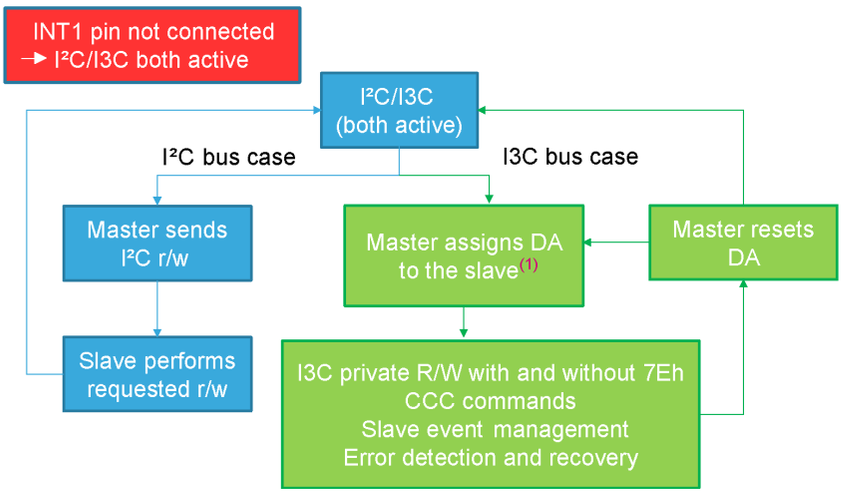
## 5.3 I²C/I3C coexistence in LSM6DSO32X

In the LSM6DSO32X, the SDA and SCL lines are common to both I²C and MIPI I3C$^{SM}$. The I²C bus requires anti-spike filters on the SDA and SCL pins that are not compatible with MIPI I3C$^{SM}$ timing.

The device can be connected to both I²C and MIPI I3C$^{SM}$ or only to the MIPI I3C$^{SM}$ bus depending on the connection of the INT1 pin when the device is powered up:

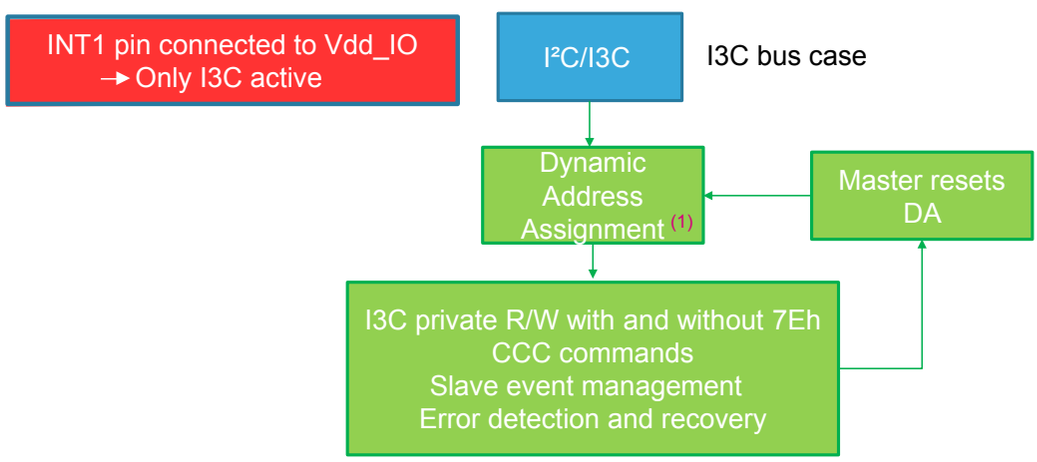
- INT1 pin floating (internal pull-down): I²C / MIPI I3C$^{SM}$ both active, see Figure 14;
- INT1 pin connected to Vdd_IO: only MIPI I3C$^{SM}$ active, see Figure 15.

**Figure 14. I²C and MIPI I3C$^{SM}$ both active (INT1 pin not connected)**




1. *Address assignment (DAA or ENTDA) must be performed with I²C Fast Mode Plus Timing. When the slave is addressed, the I²C slave is disabled and the timing is compatible with MIPI I3C$^{SM}$ specifications.*

**Figure 15. Only MIPI I3C$^{SM}$ active (INT1 pin connected to Vdd_IO)**




1. *When the slave is I3C only, the I²C slave is always disabled. The address can be assigned using MIPI I3C$^{SM}$ SDR timing.*

chanhvinh.com



## 5.4 Master I²C interface

If the LSM6DSO32X is configured in Mode 2, a master I²C line is available. The master serial interface is mapped to the following dedicated pins.

**Table 16. Master I²C pin details**

| Pin name | Pin description |
|---|---|
| MSCL | I²C serial clock master |
| MSDA | I²C serial data master |
| MDRDY | I²C master external synchronization signal |

chanhvinh.com



# 6 Functionality

## 6.1 Operating modes

In the LSM6DSO32X, the accelerometer and the gyroscope can be turned on/off independently of each other and are allowed to have different ODRs and power modes.

The LSM6DSO32X has three operating modes available:

- only accelerometer active and gyroscope in power-down;
- only gyroscope active and accelerometer in power-down;
- both accelerometer and gyroscope sensors active with independent ODR.

The accelerometer is activated from power-down by writing ODR_XL[3:0] in CTRL1_XL (10h) while the gyroscope is activated from power-down by writing ODR_G[3:0] in CTRL2_G (11h). For combo mode the ODRs are totally independent.

## 6.2 Accelerometer power modes

In the LSM6DSO, the accelerometer can be configured in five different operating modes: power-down, ultra-low-power, low-power, normal mode and high-performance mode. The operating mode selected depends on the value of the XL_HM_MODE bit in CTRL6_C (15h). If XL_HM_MODE is set to '0', high-performance mode is valid for all ODRs (from 12.5 Hz up to 6.66 kHz).

To enable the low-power and normal mode, the XL_HM_MODE bit has to be set to '1'. Low-power mode is available for lower ODRs (1.6, 12.5, 26, 52 Hz) while normal mode is available for ODRs equal to 104 and 208 Hz.

### 6.2.1 Accelerometer ultra-low-power mode

The LSM6DSO32X can be configured in ultra-low-power (ULP) mode by setting the XL_ULP_EN bit to 1 in CTRL5_C (14h) register. This mode can be used in accelerometer-only mode (gyroscope sensor must be configured in power-down mode) and for ODR_XL values between 1.6 Hz and 208 Hz.

When ULP mode is intended to be used, the bit XL_HM_MODE must be set to 0.

When ULP mode is switched ON/OFF, the accelerometer must be configured in power-down condition.

The embedded functions based on accelerometer data (free-fall, 6D/4D, tap, double tap, wake-up, activity/inactivity, stationary/motion, step counter, step detection, significant motion, tilt) and the FIFO batching functionality are still supported when ULP mode is enabled.

## 6.3 Gyroscope power modes

In the LSM6DSO, the gyroscope can be configured in four different operating modes: power-down, low-power, normal mode and high-performance mode. The operating mode selected depends on the value of the G_HM_MODE bit in CTRL7_G (16h). If G_HM_MODE is set to '0', high-performance mode is valid for all ODRs (from 12.5 Hz up to 6.66 kHz).

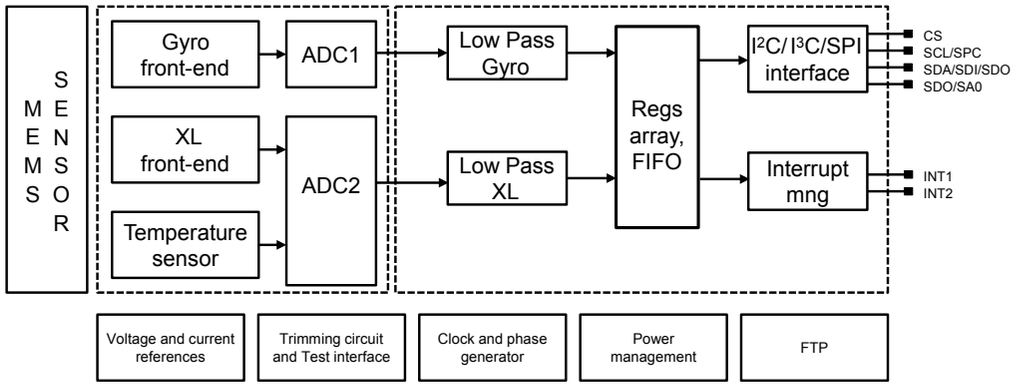
To enable the low-power and normal mode, the G_HM_MODE bit has to be set to '1'. Low-power mode is available for lower ODRs (12.5, 26, 52 Hz) while normal mode is available for ODRs equal to 104 and 208 Hz.

chanhvinh.com



## 6.4 Block diagram of filters

Figure 16. **Block diagram of filters**



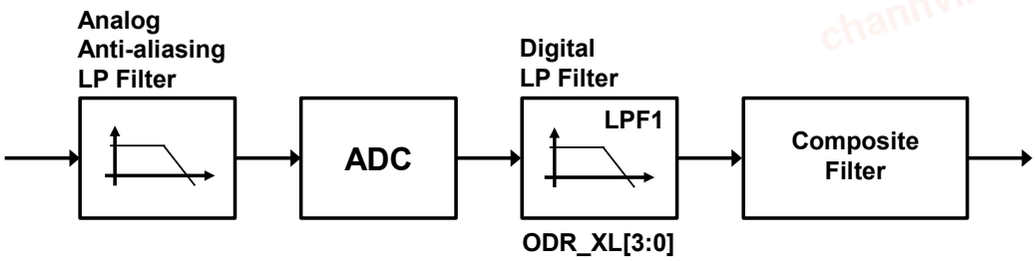

### 6.4.1 Block diagrams of the accelerometer filters

In the LSM6DSO32X, the filtering chain for the accelerometer part is composed of the following:

- Analog filter (anti-aliasing)
- Digital filter (LPF1)
- Composite filter

Details of the block diagram appear in the following figure.

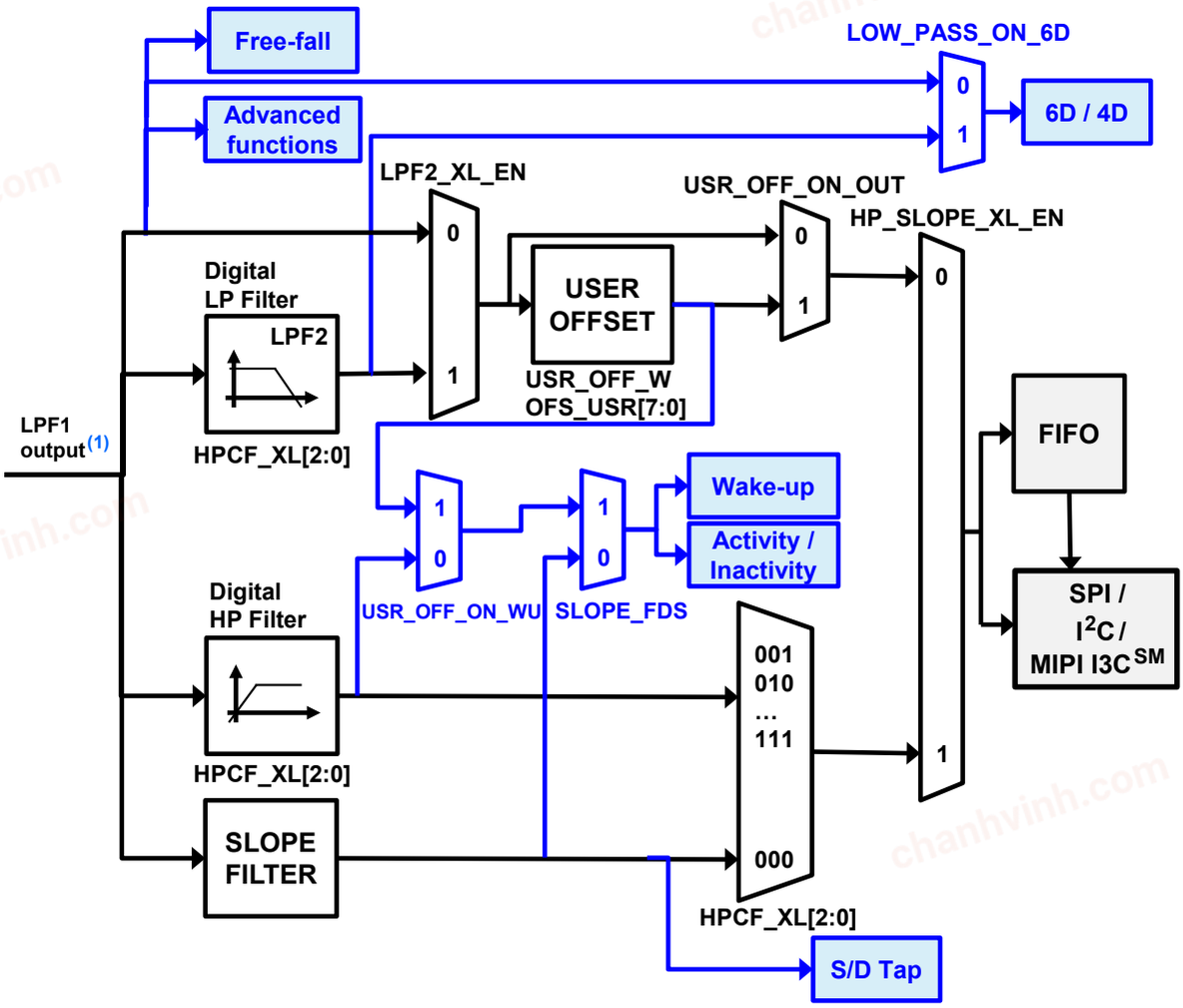
Figure 17. **Accelerometer chain**



chanhvinh.com



Figure 18. **Accelerometer composite filter**




1. *The cutoff value of the LPF1 output is ODR/2 when the accelerometer is in high-performance mode. This value is equal to 700 Hz when the accelerometer is in low-power or normal mode.*

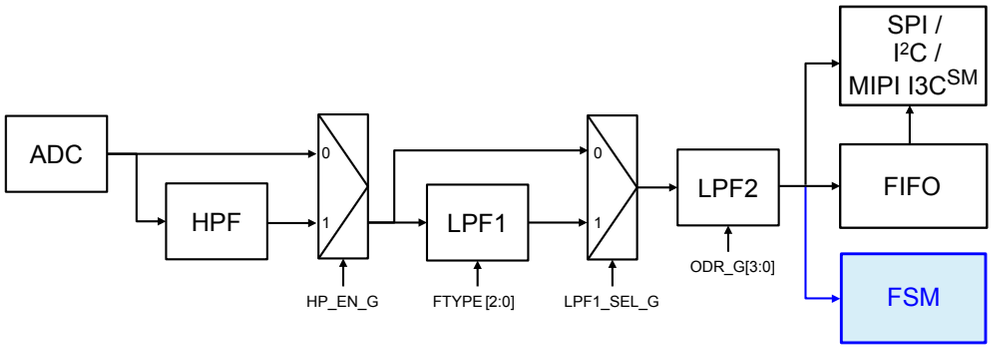
*Note: Advanced functions include pedometer, step detector and step counter, significant motion detection, tilt functions and Finite State Machine.*

chanhvinh.com



### 6.4.2 Block diagrams of the gyroscope filters

Figure 19. **Gyroscope digital chain - Mode 1 and Mode 2**




The digital LPF2 filter cannot be configured by the user and its cutoff frequency depends on the selected gyroscope ODR, as indicated in the following table.

Table 17. **Gyroscope LPF2 bandwidth selection**

| Gyroscope ODR [Hz] | LPF2 cutoff [Hz] |
|---|---|
| 12.5 | 4.2 |
| 26 | 8.3 |
| 52 | 16.6 |
| 104 | 33.0 |
| 208 | 66.8 |
| 417 | 135.9 |
| 833 | 295.5 |
| 1667 | 1108.1 |
| 3333 | 1320.7 |
| 6667 | 1441.8 |

Data can be acquired from the output registers and FIFO over the primary I²C/I3C/SPI interface.

chanhvinh.com



## 6.5 FIFO

The presence of a FIFO allows consistent power saving for the system since the host processor does not need continuously poll data from the sensor, but It can wake up only when needed and burst the significant data out from the FIFO.

The LSM6DSO32X embeds 3 kbytes of data in FIFO (up to 9 kbytes with the compression feature enabled) to store the following data:

- Gyroscope
- Accelerometer
- External sensors (up to 4)
- Step counter
- Timestamp
- Temperature

Writing data in the FIFO can be configured to be triggered by the:

- Accelerometer / gyroscope data-ready signal
- Sensor hub data-ready signal
- Step detection signal

The applications have maximum flexibility in choosing the rate of batching for physical sensors with FIFO-dedicated configurations: accelerometer, gyroscope and temperature sensor batch rates can be selected by the user. External sensor writing in FIFO can be triggered by the accelerometer data-ready signal or by an external sensor interrupt. The step counter can be stored in FIFO with associated timestamp each time a step is detected. It is possible to select decimation for timestamp batching in FIFO with a factor of 1, 8, or 32.

The reconstruction of a FIFO stream is a simple task thanks to the FIFO_DATA_OUT_TAG byte that allows recognizing the meaning of a word in FIFO.

FIFO allows correct reconstruction of the timestamp information for each sensor stored in FIFO. If a change in the ODR or BDR (Batch Data Rate) configuration is performed, the application can correctly reconstruct the timestamp and know exactly when the change was applied without disabling FIFO batching. FIFO stores information of the new configuration and timestamp in which the change was applied in the device.

Finally, FIFO embeds a compression algorithm that the user can enable in order to have up to 9 kbytes of data stored in FIFO and take advantage of interface communication length for FIFO flushing and communication power consumption.

The programmable FIFO watermark threshold can be set in FIFO_CTRL1 (07h) and FIFO_CTRL2 (08h) using the WTM[8:0] bits. To monitor the FIFO status, dedicated registers (FIFO_STATUS1 (3Ah), FIFO_STATUS2 (3Bh)) can be read to detect FIFO overrun events, FIFO full status, FIFO empty status, FIFO watermark status and the number of unread samples stored in the FIFO. To generate dedicated interrupts on the INT1 and INT2 pins of these status events, the configuration can be set in INT1_CTRL (0Dh) and INT2_CTRL (0Eh).

The FIFO buffer can be configured according to six different modes:

- Bypass mode
- FIFO mode
- Continuous mode
- Continuous-to-FIFO mode
- Bypass-to-continuous mode
- Bypass-to-FIFO mode

Each mode is selected by the FIFO_MODE_[2:0] bits in the FIFO_CTRL4 (0Ah) register.

### 6.5.1 Bypass mode

In Bypass mode (FIFO_CTRL4 (0Ah)(FIFO_MODE_[2:0] = 000), the FIFO is not operational and it remains empty. Bypass mode is also used to reset the FIFO when in FIFO mode.

chanhvinh.com



### 6.5.2 FIFO mode

In FIFO mode (FIFO_CTRL4 (0Ah)(FIFO_MODE_[2:0] = 001) data from the output channels are stored in the FIFO until it is full.

To reset FIFO content, Bypass mode should be selected by writing FIFO_CTRL4 (0Ah)(FIFO_MODE_[2:0]) to '000'. After this reset command, it is possible to restart FIFO mode by writing FIFO_CTRL4 (0Ah) (FIFO_MODE_[2:0]) to '001'.

The FIFO buffer memorizes up to 9 kbytes of data (with compression enabled) but the depth of the FIFO can be resized by setting the WTM [8:0] bits in FIFO_CTRL1 (07h) and FIFO_CTRL2 (08h). If the STOP_ON_WTM bit in FIFO_CTRL2 (08h) is set to '1', FIFO depth is limited up to the WTM [8:0] bits in FIFO_CTRL1 (07h) and FIFO_CTRL2 (08h).

### 6.5.3 Continuous mode

Continuous mode (FIFO_CTRL4 (0Ah)(FIFO_MODE_[2:0] = 110) provides a continuous FIFO update: as new data arrives, the older data is discarded.

A FIFO threshold flag FIFO_STATUS2 (3Bh)(FIFO_WTM_IA) is asserted when the number of unread samples in FIFO is greater than or equal to FIFO_CTRL1 (07h) and FIFO_CTRL2 (08h)(WTM [8:0]).

It is possible to route the FIFO_WTM_IA flag to the INT1 pin by writing in register INT1_CTRL (0Dh) (INT1_FIFO_TH) = '1' or to the INT2 pin by writing in register INT2_CTRL (0Eh)(INT2_FIFO_TH) = '1'.

A full-flag interrupt can be enabled, INT1_CTRL (0Dh)(INT1_FIFO_FULL) = '1' or INT2_CTRL (0Eh) (INT2_FIFO_FULL) = '1', in order to indicate FIFO saturation and eventually read its content all at once.

If an overrun occurs, at least one of the oldest samples in FIFO has been overwritten and the FIFO_OVR_IA flag in FIFO_STATUS2 (3Bh) is asserted.

In order to empty the FIFO before it is full, it is also possible to pull from FIFO the number of unread samples available inFIFO_STATUS1 (3Ah) and FIFO_STATUS2 (3Bh)(DIFF_FIFO_[9:0]).

### 6.5.4 Continuous-to-FIFO mode

In Continuous-to-FIFO mode (FIFO_CTRL4 (0Ah)(FIFO_MODE_[2:0] = 011), FIFO behavior changes according to the trigger event detected in one of the following interrupt events:

- Single tap
- Double tap
- Wake-up
- Free-fall
- D6D

When the selected trigger bit is equal to '1', FIFO operates in FIFO mode.

When the selected trigger bit is equal to '0', FIFO operates in Continuous mode.

### 6.5.5 Bypass-to-Continuous mode

In Bypass-to-Continuous mode (FIFO_CTRL4 (0Ah)(FIFO_MODE_[2:0] = '100'), data measurement storage inside FIFO operates in Continuous mode when selected triggers are equal to '1', otherwise FIFO content is reset (Bypass mode).

FIFO behavior changes according to the trigger event detected in one of the following interrupt events:

- Single tap
- Double tap
- Wake-up
- Free-fall
- D6D

chanhvinh.com



### 6.5.6 Bypass-to-FIFO mode

In Bypass-to-FIFO mode (FIFO_CTRL4 (0Ah)(FIFO_MODE_[2:0] = '111'), data measurement storage inside FIFO operates in FIFO mode when selected triggers are equal to '1', otherwise FIFO content is reset (Bypass mode).

FIFO behavior changes according to the trigger event detected in one of the following interrupt events:

- Single tap
- Double tap
- Wake-up
- Free-fall
- D6D

### 6.5.7 FIFO reading procedure

The data stored in FIFO are accessible from dedicated registers and each FIFO word is composed of 7 bytes: one tag byte (FIFO_DATA_OUT_TAG (78h), in order to identify the sensor, and 6 bytes of fixed data (FIFO_DATA_OUT registers from (79h) to (7Eh)).

The DIFF_FIFO_[9:0] field in the FIFO_STATUS1 (3Ah) and FIFO_STATUS2 (3Bh) registers contains the number of words (1 byte TAG + 6 bytes DATA) collected in FIFO.

In addition, it is possible to configure a counter of the batch events of accelerometer or gyroscope sensors. The flag COUNTER_BDR_IA in FIFO_STATUS2 (3Bh) alerts that the counter reaches a selectable threshold (CNT_BDR_TH_[10:0] field in COUNTER_BDR_REG1 (0Bh) and COUNTER_BDR_REG2 (0Ch)). This allows triggering the reading of FIFO with the desired latency of one single sensor. The sensor is selectable using the TRIG_COUNTER_BDR bit in COUNTER_BDR_REG1 (0Bh). As for the other FIFO status events, the flag COUNTER_BDR_IA can be routed on the INT1 or INT2 pins by asserting the corresponding bits (INT1_CNT_BDR of INT1_CTRL (0Dh) and INT2_CNT_BDR of INT2_CTRL (0Eh)).

In order to maximize the amount of accelerometer and gyroscope data in FIFO, the user can enable the compression algorithm by setting to 1 both the FIFO_COMPR_EN bit in EMB_FUNC_EN_B (05h) (embedded functions registers bank) and the FIFO_COMPR_RT_EN bit in FIFO_CTRL2 (08h). When compression is enabled, it is also possible to force writing non-compressed data at a selectable rate using the UNCOPTR_RATE_[1:0] field in FIFO_CTRL2 (08h).

Meta information about accelerometer and gyroscope sensor configuration changes can be managed by enabling the ODR_CHG_EN bit in FIFO_CTRL2 (08h).

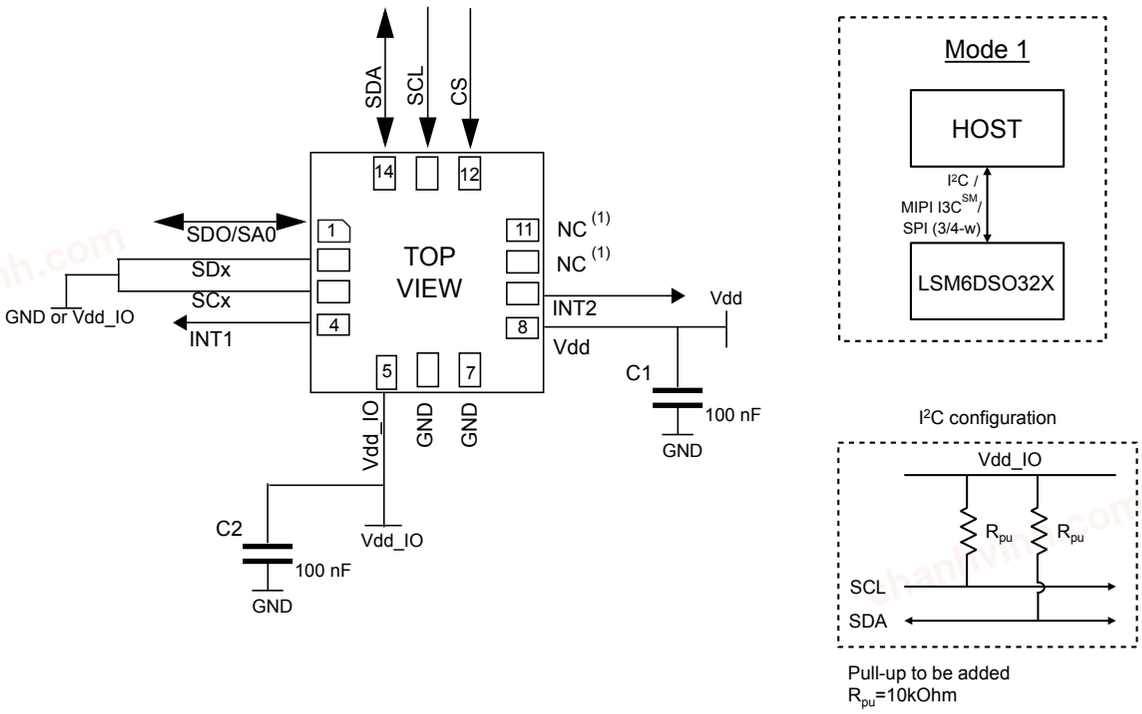
chanhvinh.com



# 7 Application hints

## 7.1 LSM6DSO32X electrical connections in Mode 1

Figure 20. **LSM6DSO32X electrical connections in Mode 1**




1. *Leave pin electrically unconnected and soldered to PCB.*

The device core is supplied through the Vdd line. Power supply decoupling capacitors (C1, C2 = 100 nF ceramic) should be placed as near as possible to the supply pin of the device (common design practice).

The functionality of the device and the measured acceleration/angular rate data is selectable and accessible through the SPI/I²C/MIPI I3C$^{SM}$ interface.

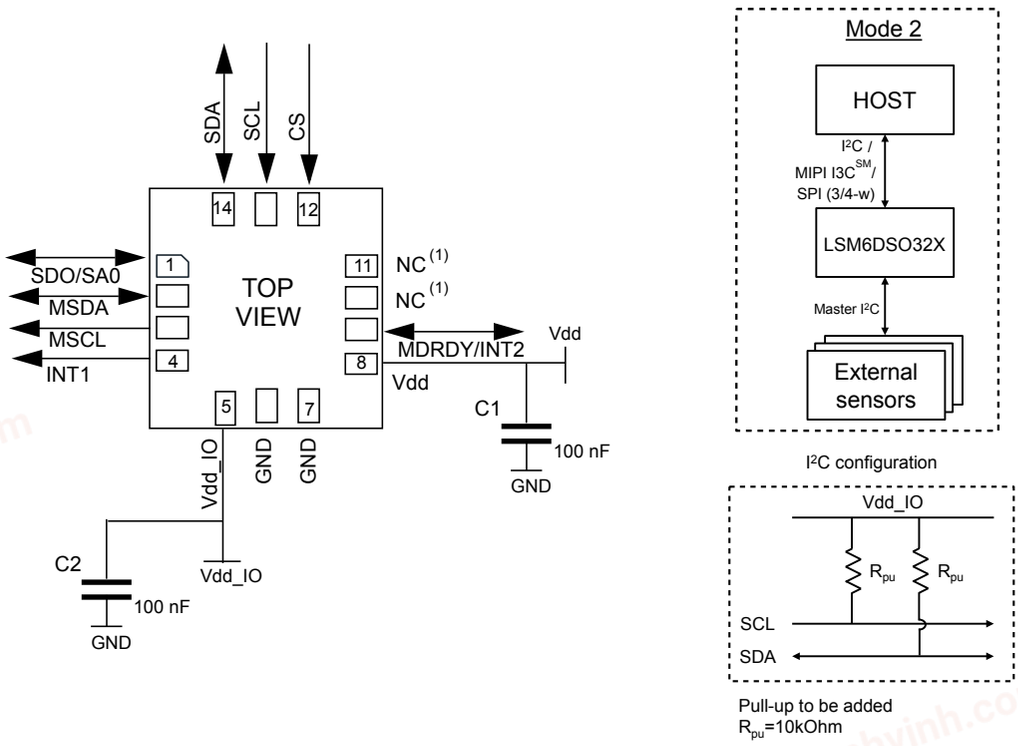
The functions, the threshold and the timing of the two interrupt pins for each sensor can be completely programmed by the user through the SPI/I²C/MIPI I3C$^{SM}$ interface.

chanhvinh.com



## 7.2 LSM6DSO32X electrical connections in Mode 2

Figure 21. **LSM6DSO32X electrical connections in Mode 2**




1. *Leave pin electrically unconnected and soldered to PCB.*

The device core is supplied through the Vdd line. Power supply decoupling capacitors (C1, C2 = 100 nF ceramic) should be placed as near as possible to the supply pin of the device (common design practice).

The functionality of the device and the measured acceleration/angular rate data is selectable and accessible through the SPI/I²C/MIPI I3C$^{SM}$ primary interface.

The functions, the threshold and the timing of the two interrupt pins for each sensor can be completely programmed by the user through the SPI/I²C/MIPI I3C$^{SM}$ primary interface.

chanhvinh.com



**Table 18. Internal pin status**

| Pin # | Name | Mode 1 function | Mode 2 function | Pin status Mode 1 | Pin status Mode 2 |
|---|---|---|---|---|---|
| 1 | SDO | SPI 4-wire interface serial data output (SDO) | SPI 4-wire interface serial data output (SDO) | Default: input without pull-up.<br>Pull-up is enabled if bit SDO_PU_EN = 1 in reg 02h. | Default: input without pull-up.<br>Pull-up is enabled if bit SDO_PU_EN = 1 in reg 02h. |
| | SA0 | I²C least significant bit of the device address (SA0)<br>MIPI I3C$^{SM}$least significant bit of the static address (SA0) | I²C least significant bit of the device address (SA0)<br>MIPI I3C$^{SM}$ least significant bit of the static address (SA0) | | |
| 2 | SDx | Connect to Vdd_IO or GND | I²C serial data master (MSDA) | Default: input without pull-up.<br>Pull-up is enabled if bit SHUB_PU_EN = 1 in reg 14h in sensor hub registers (see Note to enable pull-up). | Default: input without pull-up.<br>Pull-up is enabled if bit SHUB_PU_EN = 1 in reg 14h in sensor hub registers (see Note to enable pull-up). |
| 3 | SCx | Connect to Vdd_IO or GND | I²C serial clock master (MSCL) | Default: input without pull-up.<br>Pull-up is enabled if bit SHUB_PU_EN = 1 in reg 14h in sensor hub registers (see Note to enable pull-up). | Default: input without pull-up.<br>Pull-up is enabled if bit SHUB_PU_EN = 1 in reg 14h in sensor hub registers (see Note to enable pull-up). |
| 4 | INT1 | Programmable interrupt 1 / If device is used as MIPI I3C$^{SM}$ pure slave, this pin must be set to '1'. | Programmable interrupt 1 / If device is used as MIPI I3C$^{SM}$ pure slave, this pin must be set to '1'. | Default: input with pull-down(1) | Default: input with pull-down(1) |
| 5 | Vdd_IO | Power supply for I/O pins | Power supply for I/O pins | | |
| 6 | GND | 0 V supply | 0 V supply | | |
| 7 | GND | 0 V supply | 0 V supply | | |
| 8 | Vdd | Power supply | Power supply | | |
| 9 | INT2 | Programmable interrupt 2 (INT2) / Data enabled (DEN) | Programmable interrupt 2 (INT2) / Data enabled (DEN) / I²C master external synchronization signal (MDRDY) | Default: output forced to ground. | Default: output forced to ground. |
| 10 | NC | Leave unconnected | Leave unconnected | Default: input with pull-up. | Default: input with pull-up. |
| 11 | NC | Connect to Vdd_IO or leave unconnected | Connect to Vdd_IO or leave unconnected | Default: input with pull-up. | Default: input with pull-up. |
| 12 | CS | I²C/SPI mode selection<br>(1:SPI idle mode / I²C communication enabled;<br>0: SPI communication mode / I²C disabled) | I²C/SPI mode selection<br>(1:SPI idle mode / I²C communication enabled;<br>0: SPI communication mode / I²C disabled) | Default: input with pull-up.<br>Pull-up is disabled if bit I2C_disable = 1 in reg 13h and I3C_disable = 1 in reg 18h. | Default: input with pull-up.<br>Pull-up is disabled if bit I2C_disable = 1 in reg 13h and I3C_disable = 1 in reg 18h. |

chanhvinh.com



| Pin # | Name | Mode 1 function | Mode 2 function | Pin status Mode 1 | Pin status Mode 2 |
|---|---|---|---|---|---|
| 13 | SCL | I²C/MIPI I3C$^{SM}$ serial clock (SCL) / SPI serial port clock (SPC) | I²C/MIPI I3C$^{SM}$ serial clock (SCL) / SPI serial port clock (SPC) | Default: input without pull-up | Default: input without pull-up |
| 14 | SDA | I²C/MIPI I3C$^{SM}$ serial data (SDA) / SPI serial data input (SDI) / 3-wire interface serial data output (SDO) | I²C/MIPI I3C$^{SM}$ serial data (SDA) / SPI serial data input (SDI) / 3-wire interface serial data output (SDO) | Default: input without pull-up. | Default: input without pull-up. |

1. *INT1 must be set to '0' or left unconnected during power-on if the I²C/SPI interfaces are used.*

Internal pull-up value is from 30 kΩ to 50 kΩ, depending on Vdd_IO.

*Note: The procedure to enable the pull-up on pins 2 and 3 is as follows:*

*1. Write 40h in register at address 01h (enable access to the sensor hub registers).*

*2. Wite 08h in register at address 14h (enable the pull-up on pins 2 and 3).*

*3. Write 00h in register at address 01h (disable access to the sensor hub registers).*

chanhvinh.com



# 8 Register mapping

The table given below provides a list of the 8/16-bit registers embedded in the device and the corresponding addresses.

**Table 19. Registers address map**

| Name | Type | Register address | | Default | Comment |
|---|---|---|---|---|---|
| | | Hex | Binary | | |
| FUNC_CFG_ACCESS | RW | 01 | 00000001 | 00000000 | |
| PIN_CTRL | RW | 02 | 00000010 | 00111111 | |
| RESERVED | - | 03-06 | | | |
| FIFO_CTRL1 | RW | 07 | 00000111 | 00000000 | |
| FIFO_CTRL2 | RW | 08 | 00001000 | 00000000 | |
| FIFO_CTRL3 | RW | 09 | 00001001 | 00000000 | |
| FIFO_CTRL4 | RW | 0A | 00001010 | 00000000 | |
| COUNTER_BDR_REG1 | RW | 0B | 00001011 | 00000000 | |
| COUNTER_BDR_REG2 | RW | 0C | 00001100 | 00000000 | |
| INT1_CTRL | RW | 0D | 00001101 | 00000000 | |
| INT2_CTRL | RW | 0E | 00001110 | 00000000 | |
| WHO_AM_I | R | 0F | 00001111 | 01101100 | |
| CTRL1_XL | RW | 10 | 00010000 | 00000000 | |
| CTRL2_G | RW | 11 | 00010001 | 00000000 | |
| CTRL3_C | RW | 12 | 00010010 | 00000100 | |
| CTRL4_C | RW | 13 | 00010011 | 00000000 | |
| CTRL5_C | RW | 14 | 00010100 | 00000000 | |
| CTRL6_C | RW | 15 | 00010101 | 00000000 | |
| CTRL7_G | RW | 16 | 00010110 | 00000000 | |
| CTRL8_XL | RW | 17 | 00010111 | 00000000 | |
| CTRL9_XL | RW | 18 | 00011000 | 11100000 | |
| CTRL10_C | RW | 19 | 00011001 | 00000000 | |
| ALL_INT_SRC | R | 1A | 00011010 | output | |
| WAKE_UP_SRC | R | 1B | 00011011 | output | |
| TAP_SRC | R | 1C | 00011100 | output | |
| D6D_SRC | R | 1D | 00011101 | output | |
| STATUS_REG | R | 1E | 00011110 | output | |
| RESERVED | - | 1F | 00011111 | | |
| OUT_TEMP_L | R | 20 | 00100000 | output | |
| OUT_TEMP_H | R | 21 | 00100001 | output | |
| OUTX_L_G | R | 22 | 00100010 | output | |
| OUTX_H_G | R | 23 | 00100011 | output | |
| OUTY_L_G | R | 24 | 00100100 | output | |

chanhvinh.com



| Name | Type | Register address | | Default | Comment |
|---|---|---|---|---|---|
| | | Hex | Binary | | |
| OUTY_H_G | R | 25 | 00100101 | output | |
| OUTZ_L_G | R | 26 | 00100110 | output | |
| OUTZ_H_G | R | 27 | 00100111 | output | |
| OUTX_L_A | R | 28 | 00101000 | output | |
| OUTX_H_A | R | 29 | 00101001 | output | |
| OUTY_L_A | R | 2A | 00101010 | output | |
| OUTY_H_A | R | 2B | 00101011 | output | |
| OUTZ_L_A | R | 2C | 00101100 | output | |
| OUTZ_H_A | R | 2D | 00101101 | output | |
| RESERVED | - | 2E-34 | | | |
| EMB_FUNC_STATUS_MAINPAGE | R | 35 | 00110101 | output | |
| FSM_STATUS_A_MAINPAGE | R | 36 | 00110110 | output | |
| FSM_STATUS_B_MAINPAGE | R | 37 | 00110111 | output | |
| MLC_STATUS_MAINPAGE (38h) | R | 38 | 00111000 | output | |
| STATUS_MASTER_MAINPAGE | R | 39 | 00111001 | output | |
| FIFO_STATUS1 | R | 3A | 00111010 | output | |
| FIFO_STATUS2 | R | 3B | 00111011 | output | |
| RESERVED | - | 3C-3F | | | |
| TIMESTAMP0 | R | 40 | 01000000 | output | |
| TIMESTAMP1 | R | 41 | 01000001 | output | |
| TIMESTAMP2 | R | 42 | 01000010 | output | |
| TIMESTAMP3 | R | 43 | 01000011 | output | |
| RESERVED | - | 44-55 | | | |
| TAP_CFG0 | RW | 56 | 01010110 | 00000000 | |
| TAP_CFG1 | RW | 57 | 01010111 | 00000000 | |
| TAP_CFG2 | RW | 58 | 01011000 | 00000000 | |
| TAP_THS_6D | RW | 59 | 01011001 | 00000000 | |
| INT_DUR2 | RW | 5A | 01011010 | 00000000 | |
| WAKE_UP_THS | RW | 5B | 01011011 | 00000000 | |
| WAKE_UP_DUR | RW | 5C | 01011100 | 00000000 | |
| FREE_FALL | RW | 5D | 01011101 | 00000000 | |
| MD1_CFG | RW | 5E | 01011110 | 00000000 | |
| MD2_CFG | RW | 5F | 01011111 | 00000000 | |
| RESERVED | - | 60-61 | | 00000000 | |
| I3C_BUS_AVB | RW | 62 | 01100010 | 00000000 | |
| INTERNAL_FREQ_FINE | R | 63 | 01100011 | output | |
| RESERVED | - | 64-72 | | | |
| X_OFS_USR | RW | 73 | 01110011 | 00000000 | |
| Y_OFS_USR | RW | 74 | 01110100 | 00000000 | |
| Z_OFS_USR | RW | 75 | 01110101 | 00000000 | |

chanhvinh.com



| Name | Type | Register address | | Default | Comment |
|---|---|---|---|---|---|
| | | Hex | Binary | | |
| RESERVED | - | 76-77 | | | |
| FIFO_DATA_OUT_TAG | R | 78 | 01111000 | output | |
| FIFO_DATA_OUT_X_L | R | 79 | 01111001 | output | |
| FIFO_DATA_OUT_X_H | R | 7A | 01111010 | output | |
| FIFO_DATA_OUT_Y_L | R | 7B | 01111011 | output | |
| FIFO_DATA_OUT_Y_H | R | 7C | 01111100 | output | |
| FIFO_DATA_OUT_Z_L | R | 7D | 01111101 | output | |
| FIFO_DATA_OUT_Z_H | R | 7E | 01111110 | output | |

chanhvinh.com



# 9 Register description

The device contains a set of registers which are used to control its behavior and to retrieve linear acceleration, angular rate and temperature data. The register addresses, made up of 7 bits, are used to identify them and to write the data through the serial interface.

## 9.1 FUNC_CFG_ACCESS (01h)

Enable embedded functions register (r/w)

**Table 20. FUNC_CFG_ACCESS register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| FUNC_CFG_ACCESS | SHUB_REG ACCESS | 0(1) | 0(1) | 0(1) | 0(1) | 0(1) | 0(1) |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 21. FUNC_CFG_ACCESS register description**

| | |
|---|---|
| FUNC_CFG_ACCESS | Enable access to the embedded functions configuration registers.(1) Default value: 0 |
| SHUB_REG_ACCESS | Enable access to the sensor hub (I²C master) registers.(2) Default value: 0 |

1. *Details concerning the embedded functions configuration registers are available in Section 10 Embedded functions register mapping and Section 11 Embedded functions register description.*
2. *Details concerning the sensor hub registers are available in Section 14 Sensor hub register mapping and Section 15 Sensor hub register description.*

## 9.2 PIN_CTRL (02h)

SDO pin pull-up enable/disable register (r/w)

**Table 22. PIN_CTRL register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | SDO_ PU_EN | 1(2) | 1(2) | 1(2) | 1(2) | 1(2) | 1(2) |

1. *This bit must be set to '0' for the correct operation of the device.*
2. *This bit must be set to '1' for the correct operation of the device.*

**Table 23. PIN_CTRL register description**

| | |
|---|---|
| SDO_PU_EN | Enable pull-up on SDO pin<br>(0: SDO pin pull-up disconnected (default); 1: SDO pin with pull-up) |

chanhvinh.com



## 9.3 FIFO_CTRL1 (07h)

FIFO control register 1 (r/w)

**Table 24. FIFO_CTRL1 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| WTM7 | WTM6 | WTM5 | WTM4 | WTM3 | WTM2 | WTM1 | WTM0 |

**Table 25. FIFO_CTRL1 register description**

| | |
|---|---|
| WTM[7:0] | FIFO watermark threshold, in conjunction with WTM8 in FIFO_CTRL2 (08h)<br>1 LSB = 1 sensor (6 bytes) + TAG (1 byte) written in FIFO<br>Watermark flag rises when the number of bytes written in the FIFO is greater than or equal to the threshold level. |

## 9.4 FIFO_CTRL2 (08h)

FIFO control register 2 (r/w)

**Table 26. FIFO_CTRL2 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| STOP_ON_WTM | FIFO_COMPR_RT_EN | 0 | ODRCHG_EN | 0 | UNCOPTR_RATE_1 | UNCOPTR_RATE_0 | WTM8 |

**Table 27. FIFO_CTRL2 register description**

| | |
|---|---|
| STOP_ON_WTM | Sensing chain FIFO stop values memorization at threshold level<br>(0: FIFO depth is not limited (default);<br>1: FIFO depth is limited to threshold level, defined in FIFO_CTRL1 (07h) and FIFO_CTRL2 (08h)) |
| FIFO_COMPR_RT_EN(1) | Enables/Disables compression algorithm runtime |
| ODRCHG_EN | Enables ODR CHANGE virtual sensor to be batched in FIFO |
| UNCOPTR_RATE_[1:0] | This field configures the compression algorithm to write non-compressed data at each rate.<br>(0: Non-compressed data writing is not forced;<br>1: Non-compressed data every 8 batch data rate;<br>2: Non-compressed data every 16 batch data rate;<br>3: Non-compressed data every 32 batch data rate) |
| WTM8 | FIFO watermark threshold, in conjunction with WTM_FIFO[7:0] in FIFO_CTRL1 (07h)<br>1 LSB = 1 sensor (6 bytes) + TAG (1 byte) written in FIFO<br>Watermark flag rises when the number of bytes written in the FIFO is greater than or equal to the threshold level. |

1. *This bit is effective if the FIFO_COMPR_EN bit of EMB_FUNC_EN_B (05h) is set to 1.*

chanhvinh.com



## 9.5 FIFO_CTRL3 (09h)

FIFO control register 3 (r/w)

**Table 28. FIFO_CTRL3 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| BDR_GY_3 | BDR_GY_2 | BDR_GY_1 | BDR_GY_0 | BDR_XL_3 | BDR_XL_2 | BDR_XL_1 | BDR_XL_0 |

**Table 29. FIFO_CTRL3 register description**

| | |
|---|---|
| BDR_GY_[3:0] | Selects Batch Data Rate (write frequency in FIFO) for gyroscope data.<br>(0000: Gyro not batched in FIFO (default);<br>0001: 12.5 Hz;<br>0010: 26 Hz;<br>0011: 52 Hz;<br>0100: 104 Hz;<br>0101: 208 Hz;<br>0110: 417 Hz;<br>0111: 833 Hz;<br>1000: 1667 Hz;<br>1001: 3333 Hz;<br>1010: 6667 Hz;<br>1011: 6.5 Hz;<br>1100-1111: not allowed) |
| BDR_XL_[3:0] | Selects Batch Data Rate (write frequency in FIFO) for accelerometer data.<br>(0000: Accelerometer not batched in FIFO (default);<br>0001: 12.5 Hz;<br>0010: 26 Hz;<br>0011: 52 Hz;<br>0100: 104 Hz;<br>0101: 208 Hz;<br>0110: 417 Hz;<br>0111: 833 Hz;<br>1000: 1667 Hz;<br>1001: 3333 Hz;<br>1010: 6667 Hz;<br>1011: 1.6 Hz;<br>1100-1111: not allowed) |

chanhvinh.com



## 9.6 FIFO_CTRL4 (0Ah)

FIFO control register 4 (r/w)

**Table 30. FIFO_CTRL4 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| DEC_TS_ BATCH_1 | DEC_TS_ BATCH_0 | ODR_T_ BATCH_1 | ODR_T_ BATCH_0 | 0(1) | FIFO_ MODE2 | FIFO_ MODE1 | FIFO_ MODE0 |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 31. FIFO_CTRL4 register description**

| | |
|---|---|
| DEC_TS_BATCH_[1:0] | Selects decimation for timestamp batching in FIFO. Write rate will be the maximum rate between XL and GYRO BDR divided by decimation decoder.<br>(00: Timestamp not batched in FIFO (default);<br>01: Decimation 1: max(BDR_XL[Hz],BDR_GY[Hz]) [Hz];<br>10: Decimation 8: max(BDR_XL[Hz],BDR_GY[Hz])/8 [Hz];<br>11: Decimation 32: max(BDR_XL[Hz],BDR_GY[Hz])/32 [Hz]) |
| ODR_T_BATCH_[1:0] | Selects batch data rate (write frequency in FIFO) for temperature data<br>(00: Temperature not batched in FIFO (default);<br>01: 1.6 Hz;<br>10: 12.5 Hz;<br>11: 52 Hz) |
| FIFO_MODE[2:0] | FIFO mode selection<br>(000: Bypass mode: FIFO disabled;<br>001: FIFO mode: stops collecting data when FIFO is full;<br>010: Reserved;<br>011: Continuous-to-FIFO mode: Continuous mode until trigger is deasserted, then FIFO mode;<br>100: Bypass-to-Continuous mode: Bypass mode until trigger is deasserted, then Continuous mode;<br>101: Reserved;<br>110: Continuous mode: if the FIFO is full, the new sample overwrites the older one;<br>111: Bypass-to-FIFO mode: Bypass mode until trigger is deasserted, then FIFO mode.) |

chanhvinh.com



## 9.7 COUNTER_BDR_REG1 (0Bh)

Counter batch data rate register 1 (r/w)

**Table 32. COUNTER_BDR_REG1 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| dataready_ pulsed | RST_ COUNTER_BDR | TRIG_ COUNTER_BDR | 0(1) | 0(1) | CNT_BDR_ TH_10 | CNT_BDR_ TH_9 | CNT_BDR_ TH_8 |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 33. COUNTER_BDR_REG1 register description**

| | |
|---|---|
| dataready_pulsed | Enables pulsed data-ready mode<br>(0: Data-ready latched mode (returns to 0 only after an interface reading) (default);<br>1: Data-ready pulsed mode (the data ready pulses are 75 µs long) |
| RST_COUNTER_BDR | Resets the internal counter of batch events for a single sensor.<br>This bit is automatically reset to zero if it was set to '1'. |
| TRIG_COUNTER_BDR | Selects the trigger for the internal counter of batch events between XL and gyro.<br>(0: XL batch event;<br>1: GYRO batch event) |
| CNT_BDR_TH_[10:8] | In conjunction with CNT_BDR_TH_[7:0] in COUNTER_BDR_REG2 (0Ch), sets the threshold for the internal counter of batch events. When this counter reaches the threshold, the counter is reset and the COUNTER_BDR_IA flag in FIFO_STATUS2 (3Bh) is set to '1'. |

## 9.8 COUNTER_BDR_REG2 (0Ch)

Counter batch data rate register 2 (r/w)

**Table 34. COUNTER_BDR_REG2 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| CNT_BDR_ TH_7 | CNT_BDR_ TH_6 | CNT_BDR_ TH_5 | CNT_BDR_ TH_4 | CNT_BDR_ TH_3 | CNT_BDR_ TH_2 | CNT_BDR_ TH_1 | CNT_BDR_ TH_0 |

**Table 35. COUNTER_BDR_REG2 register description**

| | |
|---|---|
| CNT_BDR_TH_[7:0] | In conjunction with CNT_BDR_TH_[10:8] in COUNTER_BDR_REG1 (0Bh), sets the threshold for the internal counter of batch events. When this counter reaches the threshold, the counter is reset and the COUNTER_BDR_IA flag in FIFO_STATUS2 (3Bh) is set to '1'. |

chanhvinh.com



## 9.9 INT1_CTRL (0Dh)

INT1 pin control register (r/w)

Each bit in this register enables a signal to be carried over INT1 when the MIPI I3C$^{SM}$ dynamic address is not assigned (I²C or SPI is used). Some bits can be also used to trigger an IBI (In-Band Interrupt) when the MIPI I3C$^{SM}$ interface is used. The output of the pin will be the OR combination of the signals selected here and in MD1_CFG (5Eh).

**Table 36. INT1_CTRL register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| DEN_DRDY _flag | INT1_ CNT_BDR | INT1_ FIFO_FULL | INT1_ FIFO_OVR | INT1_ FIFO_TH | INT1_ BOOT | INT1_ DRDY_G | INT1_ DRDY_XL |

**Table 37. INT1_CTRL register description**

| | |
|---|---|
| DEN_DRDY_flag | Sends DEN_DRDY (DEN stamped on Sensor Data flag) to INT1 pin |
| INT1_CNT_BDR | Enables COUNTER_BDR_IA interrupt on INT1 |
| INT1_FIFO_FULL | Enables FIFO full flag interrupt on INT1 pin. It can be also used to trigger an IBI when the MIPI I3C$^{SM}$ interface is used. |
| INT1_FIFO_OVR | Enables FIFO overrun interrupt on INT1 pin. It can be also used to trigger an IBI when the MIPI I3C$^{SM}$ interface is used. |
| INT1_FIFO_TH | Enables FIFO threshold interrupt on INT1 pin. It can be also used to trigger an IBI when the MIPI I3C$^{SM}$ interface is used. |
| INT1_BOOT | Enables boot status on INT1 pin |
| INT1_DRDY_G | Enables gyroscope data-ready interrupt on INT1 pin. It can be also used to trigger an IBI when the MIPI I3C$^{SM}$ interface is used. |
| INT1_DRDY_XL | Enables accelerometer data-ready interrupt on INT1 pin. It can be also used to trigger an IBI when the MIPI I3C$^{SM}$ interface is used. |

chanhvinh.com



## 9.10 INT2_CTRL (0Eh)

INT2 pin control register (r/w)

Each bit in this register enables a signal to be carried over INT2 when the MIPI I3C$^{SM}$ dynamic address is not assigned (I²C or SPI is used). Some bits can be also used to trigger an IBI when the MIPI I3C$^{SM}$ interface is used. The output of the pin will be the OR combination of the signals selected here and in MD2_CFG (5Fh).

**Table 38. INT2_CTRL register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | INT2_ CNT_BDR | INT2_ FIFO_FULL | INT2_ FIFO_OVR | INT2_ FIFO_TH | INT2_ DRDY_TEMP | INT2_ DRDY_G | INT2_ DRDY_XL |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 39. INT2_CTRL register description**

| | |
|---|---|
| INT2_CNT_BDR | Enables COUNTER_BDR_IA interrupt on INT2 |
| INT2_FIFO_FULL | Enables FIFO full flag interrupt on INT2 pin |
| INT2_FIFO_OVR | Enables FIFO overrun interrupt on INT2 pin |
| INT2_FIFO_TH | Enables FIFO threshold interrupt on INT2 pin |
| INT2_DRDY_TEMP | Enables temperature sensor data-ready interrupt on INT2 pin. It can be also used to trigger an IBI when the MIPI I3C$^{SM}$ interface is used and INT2_ON_INT1 = ‘1’ in CTRL4_C (13h). |
| INT2_DRDY_G | Gyroscope data-ready interrupt on INT2 pin |
| INT2_DRDY_XL | Accelerometer data-ready interrupt on INT2 pin |

## 9.11 WHO_AM_I (0Fh)

WHO_AM_I register (r). This is a read-only register. Its value is fixed at 6Ch.

**Table 40. WhoAmI register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 |

chanhvinh.com



## 9.12 CTRL1_XL (10h)

Accelerometer control register 1 (r/w)

**Table 41. CTRL1_XL register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ODR_XL3 | ODR_XL2 | ODR_XL1 | ODR_XL0 | FS1_XL | FS0_XL | LPF2_XL_EN | 0(1) |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 42. CTRL1_XL register description**

| | |
|---|---|
| ODR_XL[3:0] | Accelerometer ODR selection (see Table 43) |
| FS[1:0]_XL | Accelerometer full-scale selection (see Table 44) |
| LPF2_XL_EN | Accelerometer high-resolution selection<br>(0: output from first stage digital filtering selected (default);<br>1: output from LPF2 second filtering stage selected) |

**Table 43. Accelerometer ODR register setting**

| ODR_XL3 | ODR_XL2 | ODR_XL1 | ODR_XL0 | ODR selection [Hz] when XL_HM_MODE = 1 in CTRL6_C (15h) | ODR selection [Hz] when XL_HM_MODE = 0 in CTRL6_C (15h) |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | Power-down | Power-down |
| 1 | 0 | 1 | 1 | 1.6 Hz (low power only) | 12.5 Hz (high performance) |
| 0 | 0 | 0 | 1 | 12.5 Hz (low power) | 12.5 Hz (high performance) |
| 0 | 0 | 1 | 0 | 26 Hz (low power) | 26 Hz (high performance) |
| 0 | 0 | 1 | 1 | 52 Hz (low power) | 52 Hz (high performance) |
| 0 | 1 | 0 | 0 | 104 Hz (normal mode) | 104 Hz (high performance) |
| 0 | 1 | 0 | 1 | 208 Hz (normal mode) | 208 Hz (high performance) |
| 0 | 1 | 1 | 0 | 416 Hz (high performance) | 416 Hz (high performance) |
| 0 | 1 | 1 | 1 | 833 Hz (high performance) | 833 Hz (high performance) |
| 1 | 0 | 0 | 0 | 1.66 kHz (high performance) | 1.66 kHz (high performance) |
| 1 | 0 | 0 | 1 | 3.33 kHz (high performance) | 3.33 kHz (high performance) |
| 1 | 0 | 1 | 0 | 6.66 kHz (high performance) | 6.66 kHz (high performance) |
| 1 | 1 | x | x | Not allowed | Not allowed |

**Table 44. Accelerometer full-scale selection**

| FS[1:0]_XL | Accelerometer full-scale |
|---|---|
| 00 (default) | ±4 *g* |
| 01 | ±32 *g* |
| 10 | ±8 *g* |
| 11 | ±16 *g* |

chanhvinh.com



## 9.13 CTRL2_G (11h)

Gyroscope control register 2 (r/w)

**Table 45. CTRL2_G register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ODR_G3 | ODR_G2 | ODR_G1 | ODR_G0 | FS1_G | FS0_G | FS_125 | 0(1) |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 46. CTRL2_G register description**

| | |
|---|---|
| ODR_G[3:0] | Gyroscope output data rate selection. Default value: 0000<br>(Refer to Table 47) |
| FS[1:0]_G | Gyroscope chain full-scale selection<br>(00: ±250 dps;<br>01: ±500 dps;<br>10: ±1000 dps;<br>11: ±2000 dps) |
| FS_125 | Selects gyro chain full-scale ±125 dps<br>(0: FS selected through bits FS[1:0]_G;<br>1: FS set to ±125 dps) |

**Table 47. Gyroscope ODR configuration setting**

| ODR_G3 | ODR_G2 | ODR_G1 | ODR_G0 | ODR [Hz] when G_HM_MODE = 1 in CTRL7_G (16h) | ODR [Hz] when G_HM_MODE = 0 in CTRL7_G (16h) |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | Power down | Power down |
| 0 | 0 | 0 | 1 | 12.5 Hz (low power) | 12.5 Hz (high performance) |
| 0 | 0 | 1 | 0 | 26 Hz (low power) | 26 Hz (high performance) |
| 0 | 0 | 1 | 1 | 52 Hz (low power) | 52 Hz (high performance) |
| 0 | 1 | 0 | 0 | 104 Hz (normal mode) | 104 Hz (high performance) |
| 0 | 1 | 0 | 1 | 208 Hz (normal mode) | 208 Hz (high performance) |
| 0 | 1 | 1 | 0 | 416 Hz (high performance) | 416 Hz (high performance) |
| 0 | 1 | 1 | 1 | 833 Hz (high performance) | 833 Hz (high performance) |
| 1 | 0 | 0 | 0 | 1.66 kHz (high performance) | 1.66 kHz (high performance) |
| 1 | 0 | 0 | 1 | 3.33 kHz (high performance | 3.33 kHz (high performance) |
| 1 | 0 | 1 | 0 | 6.66 kHz (high performance | 6.66 kHz (high performance) |
| 1 | 0 | 1 | 1 | Not available | Not available |

chanhvinh.com



## 9.14 CTRL3_C (12h)

Control register 3 (r/w)

**Table 48. CTRL3_C register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| BOOT | BDU | H_LACTIVE | PP_OD | SIM | IF_INC | 0(1) | SW_RESET |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 49. CTRL3_C register description**

| | |
|---|---|
| BOOT | Reboots memory content. Default value: 0<br>(0: normal mode; 1: reboot memory content)<br>This bit is automatically cleared. |
| BDU | Block Data Update. Default value: 0<br>(0: continuous update;<br>1: output registers are not updated until MSB and LSB have been read) |
| H_LACTIVE | Interrupt activation level. Default value: 0<br>(0: interrupt output pins active high; 1: interrupt output pins active low) |
| PP_OD | Push-pull/open-drain selection on INT1 and INT2 pins. Default value: 0<br>(0: push-pull mode; 1: open-drain mode) |
| SIM | SPI Serial Interface Mode selection. Default value: 0<br>(0: 4-wire interface; 1: 3-wire interface) |
| IF_INC | Register address automatically incremented during a multiple byte access with a serial interface (I²C or SPI). Default value: 1<br>(0: disabled; 1: enabled) |
| SW_RESET | Software reset. Default value: 0<br>(0: normal mode; 1: reset device)<br>This bit is automatically cleared. |

chanhvinh.com



## 9.15 CTRL4_C (13h)

Control register 4 (r/w)

**Table 50. CTRL4_C register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | SLEEP_G | INT2_on_INT1 | 0(1) | DRDY_MASK | I2C_disable | LPF1_SEL_G | 0(1) |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 51. CTRL4_C register description**

| | |
|---|---|
| SLEEP_G | Enables gyroscope Sleep mode. Default value:0<br>(0: disabled; 1: enabled) |
| INT2_on_INT1 | All interrupt signals available on INT1 pin enable. Default value: 0<br>(0: interrupt signals divided between INT1 and INT2 pins;<br>1: all interrupt signals in logic or on INT1 pin) |
| DRDY_MASK | Enables data available<br>(0: disabled;<br>1: mask DRDY on pin (both XL & Gyro) until filter settling ends (XL and Gyro independently masked). |
| I2C_disable | Disables I²C interface. Default value: 0<br>(0: SPI, I²C and MIPI I3C$^{SM}$ interfaces enabled (default); 1: I²C interface disabled) |
| LPF1_SEL_G | Enables gyroscope digital LPF1; the bandwidth can be selected through FTYPE[2:0] in CTRL6_C (15h).<br>(0: disabled; 1: enabled) |

chanhvinh.com



## 9.16 CTRL5_C (14h)

Control register 5 (r/w)

**Table 52. CTRL5_C register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| XL_ULP_EN | ROUNDING1 | ROUNDING0 | 0(1) | ST1_G | ST0_G | ST1_XL | ST0_XL |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 53. CTRL5_C register description**

| | |
|---|---|
| XL_ULP_EN | Accelerometer ultra-low-power mode enable(1). Default value: 0<br>(0: Ultra-low-power mode disabled; 1: Ultra-low-power mode enabled) |
| ROUNDING[1:0] | Circular burst-mode (rounding) read from the output registers. Default value: 00<br>(00: no rounding;<br>01: accelerometer only;<br>10: gyroscope only;<br>11: gyroscope + accelerometer) |
| ST[1:0]_G | Angular rate sensor self-test enable. Default value: 00<br>(00: Self-test disabled; Other: refer to Table 54) |
| ST[1:0]_XL | Linear acceleration sensor self-test enable. Default value: 00<br>(00: Self-test disabled; Other: refer to Table 55) |

1. *Further details about the accelerometer ultra-low-power mode are provided in Section 6.2.1 Accelerometer ultra-low-power mode.*

**Table 54. Angular rate sensor self-test mode selection**

| ST1_G | ST0_G | Self-test mode |
|---|---|---|
| 0 | 0 | Normal mode |
| 0 | 1 | Positive sign self-test |
| 1 | 0 | Not allowed |
| 1 | 1 | Negative sign self-test |

**Table 55. Linear acceleration sensor self-test mode selection**

| ST1_XL | ST0_XL | Self-test mode |
|---|---|---|
| 0 | 0 | Normal mode |
| 0 | 1 | Positive sign self-test |
| 1 | 0 | Negative sign self-test |
| 1 | 1 | Not allowed |

chanhvinh.com



## 9.17 CTRL6_C (15h)

Control register 6 (r/w)

**Table 56. CTRL6_C register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| TRIG_EN | LVL1_EN | LVL2_EN | XL_HM _MODE | USR_ OFF_W | FTYPE_2 | FTYPE_1 | FTYPE_0 |

**Table 57. CTRL6_C register description**

| | |
|---|---|
| TRIG_EN | DEN data edge-sensitive trigger enable. Refer to Table 58. |
| LVL1_EN | DEN data level-sensitive trigger enable. Refer to Table 58. |
| LVL2_EN | DEN level-sensitive latched enable. Refer toTable 58. |
| XL_HM_MODE | High-performance operating mode disable for accelerometer. Default value: 0<br>(0: high-performance operating mode enabled;<br>1: high-performance operating mode disabled) |
| USR_OFF_W | Weight of XL user offset bits of registers X_OFS_USR (73h), Y_OFS_USR (74h), Z_OFS_USR (75h)<br>(0: $2^{-10}$*g*/LSB;<br>1: $2^{-6}$*g*/LSB) |
| FTYPE[2:0] | Gyroscope's low-pass filter (LPF1) bandwidth selection<br>Table 59 shows the selectable bandwidth values. |

**Table 58. Trigger mode selection**

| TRIG_EN, LVL1_EN, LVL2_EN | Trigger mode |
|---|---|
| 100 | Edge-sensitive trigger mode is selected |
| 010 | Level-sensitive trigger mode is selected |
| 011 | Level-sensitive latched mode is selected |
| 110 | Level-sensitive FIFO enable mode is selected |

**Table 59. Gyroscope LPF1 bandwidth selection**

| FTYPE [2:0] | 12.5 Hz | 26 Hz | 52 Hz | 104 Hz | 208 Hz | 416 Hz | 833 Hz | 1.67 kHz | 3.33 kHz | 6.67 kHz |
|---|---|---|---|---|---|---|---|---|---|---|
| 000 | 4.2 | 8.3 | 16.6 | 33.0 | 67.0 | 136.6 | 239.2 | 304.2 | 328.5 | 335.5 |
| 001 | 4.2 | 8.3 | 16.6 | 33.0 | 67.0 | 130.5 | 192.4 | 220.7 | 229.6 | 232.0 |
| 010 | 4.2 | 8.3 | 16.6 | 33.0 | 67.0 | 120.3 | 154.2 | 166.6 | 170.1 | 171.1 |
| 011 | 4.2 | 8.3 | 16.6 | 33.0 | 67.0 | 137.1 | 281.8 | 453.2 | 559.2 | 609.0 |
| 100 | 4.2 | 8.3 | 16.7 | 33.0 | 62.4 | 86.7 | 96.6 | 99.6 | 100.4 | 100.6 |
| 101 | 4.2 | 8.3 | 16.8 | 31.0 | 43.2 | 48.0 | 49.4 | 49.8 | 49.9 | 49.9 |
| 110 | 4.1 | 7.8 | 13.4 | 19.0 | 23.1 | 24.6 | 25.0 | 25.1 | 25.1 | 25.1 |
| 111 | 3.9 | 6.7 | 9.7 | 11.5 | 12.2 | 12.4 | 12.5 | 12.5 | 12.5 | 12.5 |

chanhvinh.com



## 9.18 CTRL7_G (16h)

Control register 7 (r/w)

**Table 60. CTRL7_G register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| G_HM_MODE | HP_EN_G | HPM1_G | HPM0_G | 0(1) | 0(1) | USR_OFF_ON_OUT | 0(1) |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 61. CTRL7_G register description**

| | |
|---|---|
| G_HM_MODE | Disables high-performance operating mode for gyroscope. Default value: 0<br>(0: high-performance operating mode enabled;<br>1: high-performance operating mode disabled) |
| HP_EN_G | Enables gyroscope digital high-pass filter. The filter is enabled only if the gyro is in HP mode. Default value: 0<br>(0: HPF disabled; 1: HPF enabled) |
| HPM_G[1:0] | Gyroscope digital HP filter cutoff selection. Default: 00<br>(00: 16 mHz;<br>01: 65 mHz;<br>10: 260 mHz;<br>11: 1.04 Hz) |
| USR_OFF_ON_OUT | Enables accelerometer user offset correction block; it's valid for the low-pass path - see Figure 18. Accelerometer composite filter. Default value: 0<br>(0: accelerometer user offset correction block bypassed;<br>1: accelerometer user offset correction block enabled) |

chanhvinh.com



## 9.19 CTRL8_XL (17h)

Control register 8 (r/w)

**Table 62. CTRL8_XL register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| HPCF_XL_2 | HPCF_XL_1 | HPCF_XL_0 | HP_REF_MODE_XL | FASTSETTL_MODE_XL | HP_SLOPE_XL_EN | 0(1) | LOW_PASS_ON_6D |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 63. CTRL8_XL register description**

| | |
|---|---|
| HPCF_XL_[2:0] | Accelerometer LPF2 and HP filter configuration and cutoff setting. Refer to Table 64. |
| HP_REF_MODE_XL | Enables accelerometer high-pass filter reference mode (valid for high-pass path - HP_SLOPE_XL_EN bit must be '1'). Default value: 0(1)<br>(0: disabled, 1: enabled) |
| FASTSETTL_MODE_XL | Enables accelerometer LPF2 and HPF fast-settling mode. The filter sets the second samples after writing this bit. Active only during device exit from power- down mode. Default value: 0<br>(0: disabled, 1: enabled) |
| HP_SLOPE_XL_EN | Accelerometer slope filter / high-pass filter selection. Refer to Figure 22. Accelerometer block diagram. |
| LOW_PASS_ON_6D | LPF2 on 6D function selection. Refer to Figure 22. Default value: 0<br>(0: ODR/2 low-pass filtered data sent to 6D interrupt function;<br>1: LPF2 output data sent to 6D interrupt function) |

1. *When enabled, the first output data have to be discarded.*

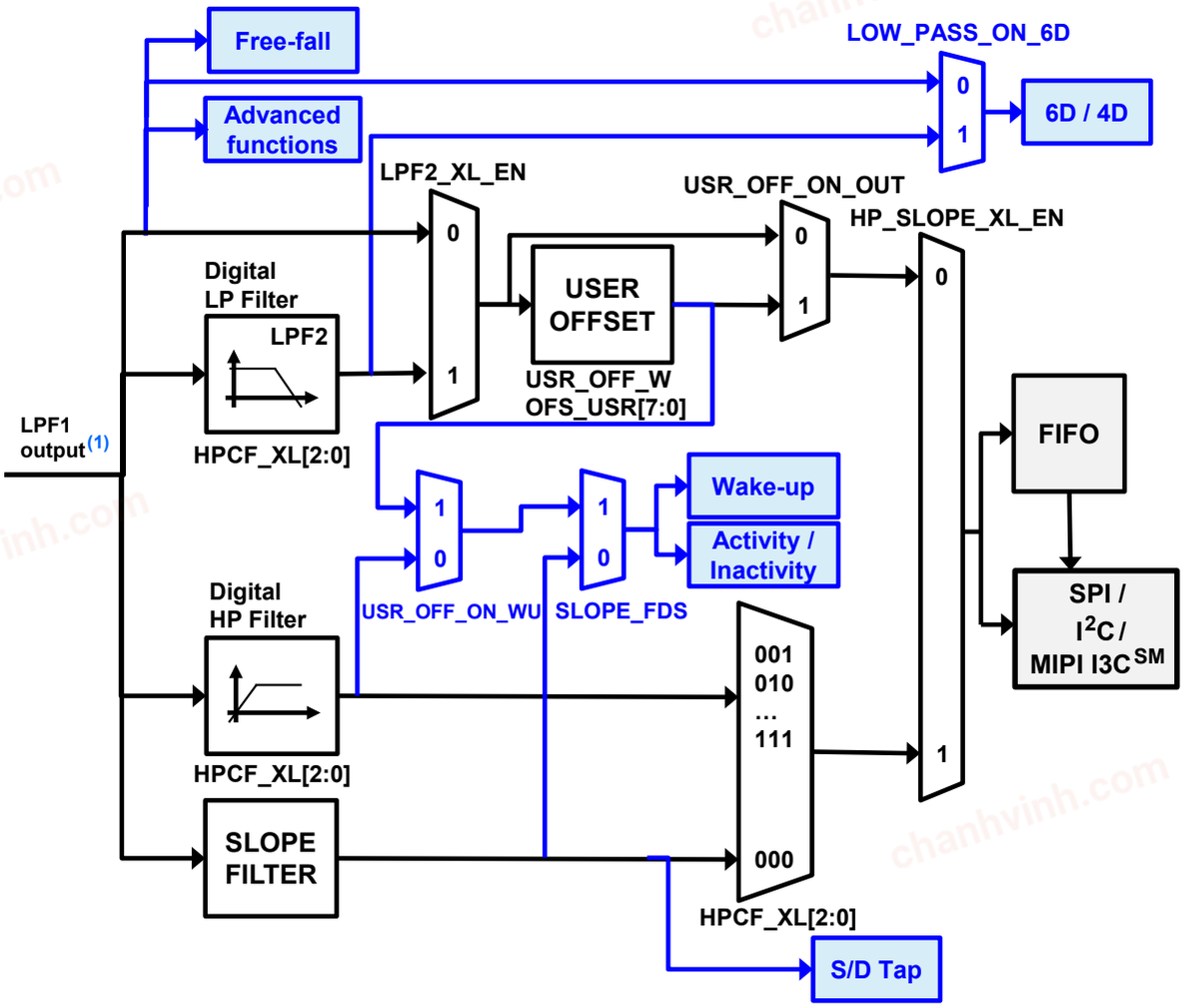
**Table 64. Accelerometer bandwidth configurations**

| Filter type | HP_SLOPE_XL_EN | LPF2_XL_EN | HPCF_XL_[2:0] | Bandwidth |
|---|---|---|---|---|
| Low pass | 0 | 0 | - | ODR/2 |
| | | 1 | 000 | ODR/4 |
| | | | 001 | ODR/10 |
| | | | 010 | ODR/20 |
| | | | 011 | ODR/45 |
| | | | 100 | ODR/100 |
| | | | 101 | ODR/200 |
| | | | 110 | ODR/400 |
| | | | 111 | ODR/800 |
| High pass | 1 | - | 000 | SLOPE (ODR/4) |
| | | | 001 | ODR/10 |
| | | | 010 | ODR/20 |
| | | | 011 | ODR/45 |
| | | | 100 | ODR/100 |
| | | | 101 | ODR/200 |
| | | | 110 | ODR/400 |
| | | | 111 | ODR/800 |

chanhvinh.com



Figure 22. **Accelerometer block diagram**




1. *The cutoff value of the LPF1 output is ODR/2 when the accelerometer is in high-performance mode. This value is equal to 700 Hz when the accelerometer is in low-power or normal mode.*

chanhvinh.com



## 9.20 CTRL9_XL (18h)

Control register 9 (r/w)

**Table 65. CTRL9_XL register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| DEN_X | DEN_Y | DEN_Z | DEN_XL_G | DEN_XL_EN | DEN_LH | I3C_disable | 0(1) |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 66. CTRL9_XL register description**

| | |
|---|---|
| DEN_X | DEN value stored in LSB of X-axis. Default value: 1<br>(0: DEN not stored in X-axis LSB; 1: DEN stored in X-axis LSB) |
| DEN_Y | DEN value stored in LSB of Y-axis. Default value: 1<br>(0: DEN not stored in Y-axis LSB; 1: DEN stored in Y-axis LSB) |
| DEN_Z | DEN value stored in LSB of Z-axis. Default value: 1<br>(0: DEN not stored in Z-axis LSB; 1: DEN stored in Z-axis LSB) |
| DEN_XL_G | DEN stamping sensor selection. Default value: 0<br>(0: DEN pin info stamped in the gyroscope axis selected by bits [7:5];<br>1: DEN pin info stamped in the accelerometer axis selected by bits [7:5]) |
| DEN_XL_EN | Extends DEN functionality to accelerometer sensor. Default value: 0<br>(0: disabled; 1: enabled) |
| DEN_LH | DEN active level configuration. Default value: 0<br>(0: active low; 1: active high) |
| I3C_disable | Disables MIPI I3C$^{SM}$ communication protocol(1)<br>(0: SPI, I²C, MIPI I3C$^{SM}$ interfaces enabled (default);<br>1: MIPI I3C$^{SM}$ interface disabled) |

1. *It is recommended to set this bit to '1' during the initial device configuration phase, when the I3C interface is not used.*

chanhvinh.com



## 9.21 CTRL10_C (19h)

Control register 10 (r/w)

**Table 67. CTRL10_C register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | 0(1) | TIMESTAMP_EN | 0(1) | 0(1) | 0(1) | 0(1) | 0(1) |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 68. CTRL10_C register description**

| | |
|---|---|
| TIMESTAMP_EN | Enables timestamp counter. default value: 0<br>(0: disabled; 1: enabled)<br>The counter is readable in TIMESTAMP0 (40h), TIMESTAMP1 (41h), TIMESTAMP2 (42h), and TIMESTAMP3 (43h). |

## 9.22 ALL_INT_SRC (1Ah)

Source register for all interrupts (r)

**Table 69. ALL_INT_SRC register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| TIMESTAMP_ENDCOUNT | 0 | SLEEP_CHANGE_IA | D6D_IA | DOUBLE_TAP | SINGLE_TAP | WU_IA | FF_IA |

**Table 70. ALL_INT_SRC register description**

| | |
|---|---|
| TIMESTAMP_ENDCOUNT | Alerts timestamp overflow within 6.4 ms |
| SLEEP_CHANGE_IA | Detects change event in activity/inactivity status. Default value: 0<br>(0: change status not detected; 1: change status detected) |
| D6D_IA | Interrupt active for change in position of portrait, landscape, face-up, face-down. Default value: 0<br>(0: change in position not detected; 1: change in position detected) |
| DOUBLE_TAP | Double-tap event status. Default value: 0<br>(0:event not detected, 1: event detected) |
| SINGLE_TAP | Single-tap event status. Default value:0<br>(0: event not detected, 1: event detected) |
| WU_IA | Wake-up event status. Default value: 0<br>(0: event not detected, 1: event detected) |
| FF_IA | Free-fall event status. Default value: 0<br>(0: event not detected, 1: event detected) |

chanhvinh.com



## 9.23 WAKE_UP_SRC (1Bh)

Wake-up interrupt source register (r)

**Table 71. WAKE_UP_SRC register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | SLEEP_ CHANGE_IA | FF_IA | SLEEP_ STATE | WU_IA | X_WU | Y_WU | Z_WU |

**Table 72. WAKE_UP_SRC register description**

| | |
|---|---|
| SLEEP_ CHANGE_IA | Detects change event in activity/inactivity status. Default value: 0<br>(0: change status not detected; 1: change status detected) |
| FF_IA | Free-fall event detection status. Default value: 0<br>(0: free-fall event not detected; 1: free-fall event detected) |
| SLEEP_STATE | Sleep status bit. Default value: 0<br>(0: Activity status; 1: Inactivity status) |
| WU_IA | Wakeup event detection status. Default value: 0<br>(0: wakeup event not detected; 1: wakeup event detected.) |
| X_WU | Wakeup event detection status on X-axis. Default value: 0<br>(0: wakeup event on X-axis not detected; 1: wakeup event on X-axis detected) |
| Y_WU | Wakeup event detection status on Y-axis. Default value: 0<br>(0: wakeup event on Y-axis not detected; 1: wakeup event on Y-axis detected) |
| Z_WU | Wakeup event detection status on Z-axis. Default value: 0<br>(0: wakeup event on Z-axis not detected; 1: wakeup event on Z-axis detected) |

chanhvinh.com



## 9.24 TAP_SRC (1Ch)

Tap source register (r)

**Table 73. TAP_SRC register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | TAP_IA | SINGLE_TAP | DOUBLE_TAP | TAP_SIGN | X_TAP | Y_TAP | Z_TAP |

**Table 74. TAP_SRC register description**

| | |
|---|---|
| TAP_IA | Tap event detection status. Default: 0<br>(0: tap event not detected; 1: tap event detected) |
| SINGLE_TAP | Single-tap event status. Default value: 0<br>(0: single tap event not detected; 1: single tap event detected) |
| DOUBLE_TAP | Double-tap event detection status. Default value: 0<br>(0: double-tap event not detected; 1: double-tap event detected.) |
| TAP_SIGN | Sign of acceleration detected by tap event. Default: 0<br>(0: positive sign of acceleration detected by tap event;<br>1: negative sign of acceleration detected by tap event) |
| X_TAP | Tap event detection status on X-axis. Default value: 0<br>(0: tap event on X-axis not detected; 1: tap event on X-axis detected) |
| Y_TAP | Tap event detection status on Y-axis. Default value: 0<br>(0: tap event on Y-axis not detected; 1: tap event on Y-axis detected) |
| Z_TAP | Tap event detection status on Z-axis. Default value: 0<br>(0: tap event on Z-axis not detected; 1: tap event on Z-axis detected) |

chanhvinh.com



## 9.25 D6D_SRC (1Dh)

Portrait, landscape, face-up and face-down source register (r)

**Table 75. D6D_SRC register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| DEN_DRDY | D6D_IA | ZH | ZL | YH | YL | XH | XL |

**Table 76. D6D_SRC register description**

| | |
|---|---|
| DEN_DRDY | DEN data-ready signal. It is set high when data output is related to the data coming from a DEN active condition.(1) |
| D6D_IA | Interrupt active for change position portrait, landscape, face-up, face-down. Default value: 0<br>(0: change position not detected; 1: change position detected) |
| ZH | Z-axis high event (over threshold). Default value: 0<br>(0: event not detected; 1: event (over threshold) detected) |
| ZL | Z-axis low event (under threshold). Default value: 0<br>(0: event not detected; 1: event (under threshold) detected) |
| YH | Y-axis high event (over threshold). Default value: 0<br>(0: event not detected; 1: event (over-threshold) detected) |
| YL | Y-axis low event (under threshold). Default value: 0<br>(0: event not detected; 1: event (under threshold) detected) |
| XH | X-axis high event (over threshold). Default value: 0<br>(0: event not detected; 1: event (over threshold) detected) |
| XL | X-axis low event (under threshold). Default value: 0<br>(0: event not detected; 1: event (under threshold) detected) |

1. *The DEN data-ready signal can be latched or pulsed depending on the value of the dataready_pulsed bit of the COUNTER_BDR_REG1 (0Bh) register.*

chanhvinh.com



## 9.26 STATUS_REG (1Eh)

Status register (r)

**Table 77. STATUS_REG register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | TDA | GDA | XLDA |

**Table 78. STATUS_REG register description**

| | |
|---|---|
| TDA | Temperature new data available. Default: 0<br>(0: no set of data is available at temperature sensor output;<br>1: a new set of data is available at temperature sensor output) |
| GDA | Gyroscope new data available. Default value: 0<br>(0: no set of data available at gyroscope output;<br>1: a new set of data is available at gyroscope output) |
| XLDA | Accelerometer new data available. Default value: 0<br>(0: no set of data available at accelerometer output;<br>1: a new set of data is available at accelerometer output) |

chanhvinh.com



## 9.27 OUT_TEMP_L (20h), OUT_TEMP_H (21h)

Temperature data output register (r). L and H registers together express a 16-bit word in two's complement.

**Table 79. OUT_TEMP_L register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Temp7 | Temp6 | Temp5 | Temp4 | Temp3 | Temp2 | Temp1 | Temp0 |

**Table 80. OUT_TEMP_H register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Temp15 | Temp14 | Temp13 | Temp12 | Temp11 | Temp10 | Temp9 | Temp8 |

**Table 81. OUT_TEMP register description**

| | |
|---|---|
| Temp[15:0] | Temperature sensor output data<br>The value is expressed as two's complement sign extended on the MSB. |

## 9.28 OUTX_L_G (22h) and OUTX_H_G (23h)

Angular rate sensor pitch axis (X) angular rate output register (r). The value is expressed as a 16-bit word in two's complement.

**Table 82. OUTX_L_G register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |

**Table 83. OUTX_H_G register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 |

**Table 84. OUTX_H_G register description**

| | |
|---|---|
| D[15:0] | Pitch axis (X) angular rate value |

chanhvinh.com



## 9.29 OUTY_L_G (24h) and OUTY_H_G (25h)

Angular rate sensor roll axis (Y) angular rate output register (r). The value is expressed as a 16-bit word in two's complement.

**Table 85. OUTY_L_G register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |

**Table 86. OUTY_H_G register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 |

**Table 87. OUTY_H_G register description**

| | |
|---|---|
| D[15:0] | Roll axis (Y) angular rate value |

## 9.30 OUTZ_L_G (26h) and OUTZ_H_G (27h)

Angular rate sensor yaw axis (Z) angular rate output register (r). The value is expressed as a 16-bit word in two's complement.

**Table 88. OUTZ_L_G register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |

**Table 89. OUTZ_H_G register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 |

**Table 90. OUTZ_H_G register description**

| | |
|---|---|
| D[15:0] | Yaw axis (Z) angular rate value |

chanhvinh.com



## 9.31 OUTX_L_A (28h) and OUTX_H_A (29h)

Linear acceleration sensor X-axis output register (r). The value is expressed as a 16-bit word in two's complement.

**Table 91. OUTX_L_A register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |

**Table 92. OUTX_H_A register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 |

**Table 93. OUTX_H_A register description**

| | |
|---|---|
| D[15:0] | X-axis linear acceleration value. |

## 9.32 OUTY_L_A (2Ah) and OUTY_H_A (2Bh)

Linear acceleration sensor Y-axis output register (r). The value is expressed as a 16-bit word in two's complement.

**Table 94. OUTY_L_A register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |

**Table 95. OUTY_H_A register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 |

**Table 96. OUTY_H_A register description**

| | |
|---|---|
| D[15:0] | Y-axis linear acceleration value |

chanhvinh.com



## 9.33 OUTZ_L_A (2Ch) and OUTZ_H_A (2Dh)

Linear acceleration sensor Z-axis output register (r). The value is expressed as a 16-bit word in two's complement.

**Table 97. OUTZ_L_A register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |

**Table 98. OUTZ_H_A register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 |

**Table 99. OUTZ_H_A register description**

| | |
|---|---|
| D[15:0] | Z-axis linear acceleration value |

## 9.34 EMB_FUNC_STATUS_MAINPAGE (35h)

Embedded function status register (r)

**Table 100. EMB_FUNC_STATUS_MAINPAGE register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| IS_FSM_LC | 0 | IS_SIGMOT | IS_TILT | IS_STEP_DET | 0 | 0 | 0 |

**Table 101. EMB_FUNC_STATUS_MAINPAGE register description**

| | |
|---|---|
| IS_FSM_LC | Interrupt status bit for FSM long counter timeout interrupt event.<br>(1: interrupt detected; 0: no interrupt) |
| IS_SIGMOT | Interrupt status bit for significant motion detection<br>(1: interrupt detected; 0: no interrupt) |
| IS_TILT | Interrupt status bit for tilt detection<br>(1: interrupt detected; 0: no interrupt) |
| IS_STEP_DET | Interrupt status bit for step detection<br>(1: interrupt detected; 0: no interrupt) |

chanhvinh.com



## 9.35 FSM_STATUS_A_MAINPAGE (36h)

Finite State Machine status register (r)

**Table 102. FSM_STATUS_A_MAINPAGE register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| IS_FSM8 | IS_FSM7 | IS_FSM6 | IS_FSM5 | IS_FSM4 | IS_FSM3 | IS_FSM2 | IS_FSM1 |

**Table 103. FSM_STATUS_A_MAINPAGE register description**

| | |
|---|---|
| IS_FSM8 | Interrupt status bit for FSM8 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM7 | Interrupt status bit for FSM7 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM6 | Interrupt status bit for FSM6 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM5 | Interrupt status bit for FSM5 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM4 | Interrupt status bit for FSM4 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM3 | Interrupt status bit for FSM3 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM2 | Interrupt status bit for FSM2 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM1 | Interrupt status bit for FSM1 interrupt event. (1: interrupt detected; 0: no interrupt) |

## 9.36 FSM_STATUS_B_MAINPAGE (37h)

Finite State Machine status register (r)

**Table 104. FSM_STATUS_B_MAINPAGE register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| IS_FSM16 | IS_FSM15 | IS_FSM14 | IS_FSM13 | IS_FSM12 | IS_FSM11 | IS_FSM10 | IS_FSM9 |

**Table 105. FSM_STATUS_B_MAINPAGE register description**

| | |
|---|---|
| IS_FSM16 | Interrupt status bit for FSM16 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM15 | Interrupt status bit for FSM15 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM14 | Interrupt status bit for FSM14 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM13 | Interrupt status bit for FSM13 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM12 | Interrupt status bit for FSM12 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM11 | Interrupt status bit for FSM11 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM10 | Interrupt status bit for FSM10 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM9 | Interrupt status bit for FSM9 interrupt event. (1: interrupt detected; 0: no interrupt) |

chanhvinh.com



## 9.37 MLC_STATUS_MAINPAGE (38h)

Machine Learning Core status register (r)

**Table 106. MLC_STATUS _MAINPAGE register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| IS_MLC8 | IS_MLC7 | IS_MLC6 | IS_MLC5 | IS_MLC4 | IS_MLC3 | IS_MLC2 | IS_MLC1 |

**Table 107. MLC_STATUS_MAINPAGE register description**

| | |
|---|---|
| IS_MLC8 | Interrupt status bit for MLC8 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_MLC7 | Interrupt status bit for MLC7 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_MLC6 | Interrupt status bit for MLC6 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_MLC5 | Interrupt status bit for MLC5 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_MLC4 | Interrupt status bit for MLC4 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_MLC3 | Interrupt status bit for MLC3 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_MLC2 | Interrupt status bit for MLC2 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_MLC1 | Interrupt status bit for MLC1 interrupt event. (1: interrupt detected; 0: no interrupt) |

chanhvinh.com



## 9.38 STATUS_MASTER_MAINPAGE (39h)

Sensor hub source register (r)

**Table 108. STATUS_MASTER_MAINPAGE register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| WR_ONCE_DONE | SLAVE3_NACK | SLAVE2_NACK | SLAVE1_NACK | SLAVE0_NACK | 0 | 0 | SENS_HUB_ENDOP |

**Table 109. STATUS_MASTER_MAINPAGE register description**

| | |
|---|---|
| WR_ONCE_DONE | When the bit WRITE_ONCE in MASTER_CONFIG (14h) is configured as 1, this bit is set to 1 when the write operation on slave 0 has been performed and completed. Default value: 0 |
| SLAVE3_NACK | This bit is set to 1 if Not acknowledge occurs on slave 3 communication. Default value: 0 |
| SLAVE2_NACK | This bit is set to 1 if Not acknowledge occurs on slave 2 communication. Default value: 0 |
| SLAVE1_NACK | This bit is set to 1 if Not acknowledge occurs on slave 1 communication. Default value: 0 |
| SLAVE0_NACK | This bit is set to 1 if Not acknowledge occurs on slave 0 communication. Default value: 0 |
| SENS_HUB_ENDOP | Sensor hub communication status. Default value: 0<br>(0: sensor hub communication not concluded;<br>1: sensor hub communication concluded) |

## 9.39 FIFO_STATUS1 (3Ah)

FIFO status register 1 (r)

**Table 110. FIFO_STATUS1 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| DIFF_FIFO_7 | DIFF_FIFO_6 | DIFF_FIFO_5 | DIFF_FIFO_4 | DIFF_FIFO_3 | DIFF_FIFO_2 | DIFF_FIFO_1 | DIFF_FIFO_0 |

**Table 111. FIFO_STATUS1 register description**

| | |
|---|---|
| DIFF_FIFO_[7:0] | Number of unread sensor data (TAG + 6 bytes) stored in FIFO<br>In conjunction with DIFF_FIFO[9:8] in FIFO_STATUS2 (3Bh). |

chanhvinh.com



## 9.40 FIFO_STATUS2 (3Bh)

FIFO status register 2 (r)

**Table 112. FIFO_STATUS2 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| FIFO_ WTM_IA | FIFO_ OVR_IA | FIFO_ FULL_IA | COUNTER _BDR_IA | FIFO_OVR_ LATCHED | 0 | DIFF_ FIFO_9 | DIFF_ FIFO_8 |

**Table 113. FIFO_STATUS2 register description**

| | |
|---|---|
| FIFO_WTM_IA | FIFO watermark status. Default value: 0<br>(0: FIFO filling is lower than WTM;<br>1: FIFO filling is equal to or greater than WTM)<br>Watermark is set through bits WTM[8:0] in FIFO_CTRL2 (08h) and FIFO_CTRL1 (07h). |
| FIFO_OVR_IA | FIFO overrun status. Default value: 0<br>(0: FIFO is not completely filled; 1: FIFO is completely filled) |
| FIFO_FULL_IA | Smart FIFO full status. Default value: 0<br>(0: FIFO is not full; 1: FIFO will be full at the next ODR) |
| COUNTER_BDR_IA | Counter BDR reaches the CNT_BDR_TH_[10:0] threshold set in COUNTER_BDR_REG1 (0Bh) and COUNTER_BDR_REG2 (0Ch). Default value: 0<br>This bit is reset when these registers are read. |
| FIFO_OVR_LATCHED | Latched FIFO overrun status. Default value: 0<br>This bit is reset when this register is read. |
| DIFF_FIFO_[9:8] | Number of unread sensor data (TAG + 6 bytes) stored in FIFO. Default value: 00<br>In conjunction with DIFF_FIFO[7:0] in FIFO_STATUS1 (3Ah) |

chanhvinh.com



## 9.41 TIMESTAMP0 (40h), TIMESTAMP1 (41h), TIMESTAMP2 (42h), and TIMESTAMP3 (43h)

Timestamp first data output register (r). The value is expressed as a 32-bit word and the bit resolution is 25 µs.

**Table 114. TIMESTAMP output registers**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| D31 | D30 | D29 | D28 | D27 | D26 | D25 | D24 |
| D23 | D22 | D21 | D20 | D19 | D18 | D17 | D16 |
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 |
| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |

**Table 115. TIMESTAMP output register description**

| | |
|---|---|
| D[31:0] | Timestamp output registers: 1LSB = 25 µs |

chanhvinh.com



## 9.42 TAP_CFG0 (56h)

Activity/inactivity functions, configuration of filtering, and tap recognition functions (r/w)

**Table 116. TAP_CFG0 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | INT_CLR_ON_READ | SLEEP_STATUS_ON_INT | SLOPE_FDS | TAP_X_EN | TAP_Y_EN | TAP_Z_EN | LIR |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 117. TAP_CFG0 register description**

| | |
|---|---|
| INT_CLR_ON_READ | This bit allows immediately clearing the latched interrupts of an event detection upon the read of the corresponding status register. It must be set to 1 together with LIR. Default value: 0<br>(0: latched interrupt signal cleared at the end of the ODR period;<br>1: latched interrupt signal immediately cleared) |
| SLEEP_STATUS_ON_INT | Activity/inactivity interrupt mode configuration.<br>If INT1_SLEEP_CHANGE or INT2_SLEEP_CHANGE bits are enabled, drives the sleep status or sleep change on the INT pins. Default value: 0<br>(0: sleep change notification on INT pins; 1: sleep status reported on INT pins) |
| SLOPE_FDS | HPF or SLOPE filter selection on wake-up and Activity/Inactivity functions. Default value: 0<br>(0: SLOPE filter applied; 1: HPF applied) |
| TAP_X_EN | Enable X direction in tap recognition. Default value: 0<br>(0: X direction disabled; 1: X direction enabled) |
| TAP_Y_EN | Enable Y direction in tap recognition. Default value: 0<br>(0: Y direction disabled; 1: Y direction enabled) |
| TAP_Z_EN | Enable Z direction in tap recognition. Default value: 0<br>(0: Z direction disabled; 1: Z direction enabled) |
| LIR | Latched Interrupt. Default value: 0<br>(0: interrupt request not latched; 1: interrupt request latched) |

chanhvinh.com



## 9.43 TAP_CFG1 (57h)

Tap configuration register (r/w)

**Table 118. TAP_CFG1 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| TAP_ PRIORITY_2 | TAP_ PRIORITY_1 | TAP_ PRIORITY_0 | TAP_THS_X_4 | TAP_THS_X_3 | TAP_THS_X_2 | TAP_THS_X_1 | TAP_THS_X_0 |

**Table 119. TAP_CFG1 register description**

| | |
|---|---|
| TAP_PRIORITY_[2:0] | Selection of axis priority for TAP detection (see Table 120) |
| TAP_THS_X_[4:0] | X-axis tap recognition threshold. Default value: 0<br>1 LSB = FS_XL / ($2^5$) |

**Table 120. TAP priority decoding**

| TAP_PRIORITY_[2:0] | Max. priority | Mid. priority | Min. priority |
|---|---|---|---|
| 000 | X | Y | Z |
| 001 | Y | X | Z |
| 010 | X | Z | Y |
| 011 | Z | Y | X |
| 100 | X | Y | Z |
| 101 | Y | Z | X |
| 110 | Z | X | Y |
| 111 | Z | Y | X |

## 9.44 TAP_CFG2 (58h)

Enables interrupt and inactivity functions, and tap recognition functions (r/w)

**Table 121. TAP_CFG2 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| INTERRUPTS_ ENABLE | INACT_EN1 | INACT_EN0 | TAP_THS_Y_4 | TAP_THS_Y_3 | TAP_THS_Y_2 | TAP_THS_Y_1 | TAP_THS_Y_0 |

**Table 122. TAP_CFG2 register description**

| | |
|---|---|
| INTERRUPTS_ ENABLE | Enable basic interrupts (6D/4D, free-fall, wake-up, tap, inactivity). Default value: 0<br>(0: interrupt disabled; 1: interrupt enabled) |
| INACT_EN[1:0] | Enable activity/inactivity (sleep) function. Default value: 00<br>(00: stationary/motion-only interrupts generated, XL and gyro do not change;<br>01: sets accelerometer ODR to 12.5 Hz (low-power mode), gyro does not change;<br>10: sets accelerometer ODR to 12.5 Hz (low-power mode), gyro to sleep mode;<br>11: sets accelerometer ODR to 12.5 Hz (low-power mode), gyro to power-down mode) |
| TAP_THS_Y_[4:0] | Y-axis tap recognition threshold. Default value: 0<br>1 LSB = FS_XL / ($2^5$) |

chanhvinh.com



## 9.45 TAP_THS_6D (59h)

Portrait/landscape orientation and tap function threshold register (r/w)

**Table 123. TAP_THS_6D register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| D4D_EN | SIXD_THS1 | SIXD_THS0 | TAP_THS_Z_4 | TAP_THS_Z_3 | TAP_THS_Z_2 | TAP_THS_Z_1 | TAP_THS_Z_0 |

**Table 124. TAP_THS_6D register description**

| | |
|---|---|
| D4D_EN | 4D orientation detection enable. Z-axis position detection is disabled. Default value: 0<br>(0: disabled; 1: enabled) |
| SIXD_THS[1:0] | Threshold for 4D/6D function. Default value: 00<br>For details, refer to Table 125. |
| TAP_THS_Z_[4:0] | Z-axis recognition threshold. Default value: 0<br>1 LSB = FS_XL / ($2^5$) |

**Table 125. Threshold for D4D/D6D function**

| SIXD_THS[1:0] | Threshold value |
|---|---|
| 00 | 68 degrees |
| 01 | 47 degrees |
| 10 | Reserved |
| 11 | Reserved |

chanhvinh.com



## 9.46 INT_DUR2 (5Ah)

Tap recognition function setting register (r/w)

**Table 126. INT_DUR2 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| DUR3 | DUR2 | DUR1 | DUR0 | QUIET1 | QUIET0 | SHOCK1 | SHOCK0 |

**Table 127. INT_DUR2 register description**

| | |
|---|---|
| DUR[3:0] | Duration of maximum time gap for double tap recognition. Default: 0000<br>When double tap recognition is enabled, this register expresses the maximum time between two consecutive detected taps to determine a double tap event. The default value of these bits is 0000b which corresponds to 16/ODR_XL time. If the DUR[3:0] bits are set to a different value, 1LSB corresponds to 32/ODR_XL time. |
| QUIET[1:0] | Expected quiet time after a tap detection. Default value: 00<br>Quiet time is the time after the first detected tap in which there must not be any overthreshold event. The default value of these bits is 00b which corresponds to 2/ODR_XL time. If the QUIET[1:0] bits are set to a different value, 1LSB corresponds to 4/ODR_XL time. |
| SHOCK[1:0] | Maximum duration of overthreshold event. Default value: 00<br>Maximum duration is the maximum time of an overthreshold signal detection to be recognized as a tap event. The default value of these bits is 00b which corresponds to 4/ODR_XL time. If the SHOCK[1:0] bits are set to a different value, 1LSB corresponds to 8/ODR_XL time. |

## 9.47 WAKE_UP_THS (5Bh)

Single/double-tap selection and wake-up configuration (r/w)

**Table 128. WAKE_UP_THS register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| SINGLE_ DOUBLE_TAP | USR_OFF_ ON_WU | WK_THS5 | WK_THS4 | WK_THS3 | WK_THS2 | WK_THS1 | WK_THS0 |

**Table 129. WAKE_UP_THS register description**

| | |
|---|---|
| SINGLE_DOUBLE_TAP | Single/double-tap event enable. Default value: 0<br>(0: only single-tap event enabled; 1: both single and double-tap events enabled) |
| USR_OFF_ON_WU | Drives the low-pass filtered data with user offset correction (instead of high-pass filtered data) to the wakeup function. |
| WK_THS[5:0] | Threshold for wakeup: 1 LSB weight depends on WAKE_THS_W in WAKE_UP_DUR (5Ch).<br>Default value: 000000 |

chanhvinh.com



## 9.48 WAKE_UP_DUR (5Ch)

Free-fall, wakeup and sleep mode functions duration setting register (r/w)

**Table 130. WAKE_UP_DUR register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| FF_DUR5 | WAKE_DUR1 | WAKE_DUR0 | WAKE_THS_W | SLEEP_DUR3 | SLEEP_DUR2 | SLEEP_DUR1 | SLEEP_DUR0 |

**Table 131. WAKE_UP_DUR register description**

| | |
|---|---|
| FF_DUR5 | Free fall duration event. Default: 0<br>For the complete configuration of the free-fall duration, refer to FF_DUR[4:0] in FREE_FALL (5Dh) configuration.<br>1 LSB = 1 ODR_time |
| WAKE_DUR[1:0] | Wake up duration event. Default: 00<br>1 LSB = 1 ODR_time |
| WAKE_THS_W | Weight of 1 LSB of wakeup threshold. Default: 0<br>(0: 1 LSB = FS_XL / ($2^6$);<br>1: 1 LSB = FS_XL / ($2^8$) ) |
| SLEEP_DUR[3:0] | Duration to go in sleep mode. Default value: 0000 (this corresponds to 16 ODR)<br>1 LSB = 512 ODR |

chanhvinh.com



## 9.49 FREE_FALL (5Dh)

Free-fall function duration setting register (r/w)

**Table 132. FREE_FALL register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| FF_DUR4 | FF_DUR3 | FF_DUR2 | FF_DUR1 | FF_DUR0 | FF_THS2 | FF_THS1 | FF_THS0 |

**Table 133. FREE_FALL register description**

| | |
|---|---|
| FF_DUR[4:0] | Free-fall duration event. Default: 0<br>For the complete configuration of the free-fall duration, refer to FF_DUR5 in WAKE_UP_DUR (5Ch) configuration. |
| FF_THS[2:0] | Free-fall threshold setting. Default: 000<br>For details refer to Table 134. |

**Table 134. Threshold for free-fall function**

| FF_THS[2:0] | Threshold value |
|---|---|
| 000 | 312 mg |
| 001 | 438 mg |
| 010 | 500 mg |
| 011 | Reserved |
| 100 | Reserved |
| 101 | Reserved |
| 110 | Reserved |
| 111 | Reserved |

chanhvinh.com



## 9.50 MD1_CFG (5Eh)

Functions routing on INT1 register (r/w)

**Table 135. MD1_CFG register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| INT1_SLEEP_CHANGE | INT1_SINGLE_TAP | INT1_WU | INT1_FF | INT1_DOUBLE_TAP | INT1_6D | INT1_EMB_FUNC | INT1_SHUB |

**Table 136. MD1_CFG register description**

| | |
|---|---|
| INT1_SLEEP_CHANGE[1] | Routing of activity/inactivity recognition event on INT1. Default: 0<br>(0: routing of activity/inactivity event on INT1 disabled;<br>1: routing of activity/inactivity event on INT1 enabled) |
| INT1_SINGLE_TAP | Routing of single-tap recognition event on INT1. Default: 0<br>(0: routing of single-tap event on INT1 disabled;<br>1: routing of single-tap event on INT1 enabled) |
| INT1_WU | Routing of wakeup event on INT1. Default value: 0<br>(0: routing of wakeup event on INT1 disabled;<br>1: routing of wakeup event on INT1 enabled) |
| INT1_FF | Routing of free-fall event on INT1. Default value: 0<br>(0: routing of free-fall event on INT1 disabled;<br>1: routing of free-fall event on INT1 enabled) |
| INT1_DOUBLE_TAP | Routing of tap event on INT1. Default value: 0<br>(0: routing of double-tap event on INT1 disabled;<br>1: routing of double-tap event on INT1 enabled) |
| INT1_6D | Routing of 6D event on INT1. Default value: 0<br>(0: routing of 6D event on INT1 disabled;<br>1: routing of 6D event on INT1 enabled) |
| INT1_EMB_FUNC | Routing of embedded functions event on INT1. Default value: 0<br>(0: routing of embedded functions event on INT1 disabled;<br>1: routing of embedded functions event on INT1 enabled) |
| INT1_SHUB | Routing of sensor hub communication concluded event on INT1. Default value: 0<br>(0: routing of sensor hub communication concluded event on INT1 disabled;<br>1: routing of sensor hub communication concluded event on INT1 enabled) |

1. *Activity/Inactivity interrupt mode (sleep change or sleep status) depends on the SLEEP_STATUS_ON_INT bit in TAP_CFG0 (56h) register.*

chanhvinh.com



## 9.51 MD2_CFG (5Fh)

Functions routing on INT2 register (r/w)

**Table 137. MD2_CFG register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| INT2_SLEEP_CHANGE | INT2_SINGLE_TAP | INT2_WU | INT2_FF | INT2_DOUBLE_TAP | INT2_6D | INT2_EMB_FUNC | INT2_TIMESTAMP |

**Table 138. MD2_CFG register description**

| | |
|---|---|
| INT2_SLEEP_CHANGE[1] | Routing of activity/inactivity recognition event on INT2. Default: 0<br>(0: routing of activity/inactivity event on INT2 disabled;<br>1: routing of activity/inactivity event on INT2 enabled) |
| INT2_SINGLE_TAP | Single-tap recognition routing on INT2. Default: 0<br>(0: routing of single-tap event on INT2 disabled;<br>1: routing of single-tap event on INT2 enabled) |
| INT2_WU | Routing of wakeup event on INT2. Default value: 0<br>(0: routing of wakeup event on INT2 disabled;<br>1: routing of wake-up event on INT2 enabled) |
| INT2_FF | Routing of free-fall event on INT2. Default value: 0<br>(0: routing of free-fall event on INT2 disabled;<br>1: routing of free-fall event on INT2 enabled) |
| INT2_DOUBLE_TAP | Routing of tap event on INT2. Default value: 0<br>(0: routing of double-tap event on INT2 disabled;<br>1: routing of double-tap event on INT2 enabled) |
| INT2_6D | Routing of 6D event on INT2. Default value: 0<br>(0: routing of 6D event on INT2 disabled;<br>1: routing of 6D event on INT2 enabled) |
| INT2_EMB_FUNC | Routing of embedded functions event on INT2. Default value: 0<br>(0: routing of embedded functions event on INT2 disabled;<br>1: routing of embedded functions event on INT2 enabled) |
| INT2_TIMESTAMP | Enables routing on INT2 pin of the alert for timestamp overflow within 6.4 ms. |

1. *Activity/Inactivity interrupt mode (sleep change or sleep status) depends on the SLEEP_STATUS_ON_INT bit in TAP_CFG0 (56h) register.*

chanhvinh.com



## 9.52 I3C_BUS_AVB (62h)

I3C_BUS_AVB register (r/w)

**Table 139. I3C_BUS_AVB register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | 0(1) | 0(1) | I3C_Bus_Avb_Sel1 | I3C_Bus_Avb_Sel0 | 0(1) | 0(1) | PD_DIS_INT1 |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 140. I3C_BUS_AVB register description**

| | |
|---|---|
| PD_DIS_INT1 | This bit allows disabling the INT1 pull-down.<br>(0: Pull-down on INT1 enabled (pull-down is effectively connected only when no interrupts are routed to the INT1 pin or when I3C dynamic address is assigned);<br>1: Pull-down on INT1 disabled (pull-down not connected) |
| I3C_Bus_Avb_Sel[1:0] | These bits are used to select the bus available time when I3C IBI is used. Default value: 00<br>(00: bus available time equal to 50 µsec (default);<br>01: bus available time equal to 2 µsec;<br>10: bus available time equal to 1 msec;<br>11: bus available time equal to 25 msec) |

## 9.53 INTERNAL_FREQ_FINE (63h)

Internal frequency register (r)

**Table 141. INTERNAL_FREQ_FINE register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| FREQ_FINE7 | FREQ_FINE6 | FREQ_FINE5 | FREQ_FINE4 | FREQ_FINE3 | FREQ_FINE2 | FREQ_FINE1 | FREQ_FINE0 |

**Table 142. INTERNAL_FREQ_FINE register description**

| | |
|---|---|
| FREQ_FINE[7:0] | Difference in percentage of the effective ODR (and Timestamp Rate) with respect to the typical. Step: 0.15%. 8-bit format, 2's complement. |

## 9.54 X_OFS_USR (73h)

Accelerometer X-axis user offset correction (r/w). The offset value set in the X_OFS_USR offset register is internally subtracted from the acceleration value measured on the X-axis.

**Table 143. X_OFS_USR register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| X_OFS_USR_7 | X_OFS_USR_6 | X_OFS_USR_5 | X_OFS_USR_4 | X_OFS_USR_3 | X_OFS_USR_2 | X_OFS_USR_1 | X_OFS_USR_0 |

**Table 144. X_OFS_USR register description**

| | |
|---|---|
| X_OFS_USR_[7:0] | Accelerometer X-axis user offset correction expressed in two's complement, weight depends on USR_OFF_W in CTRL6_C (15h). The value must be in the range [-127 127]. |

chanhvinh.com



## 9.55 Y_OFS_USR (74h)

Accelerometer Y-axis user offset correction (r/w). The offset value set in the Y_OFS_USR offset register is internally subtracted from the acceleration value measured on the Y-axis.

**Table 145. Y_OFS_USR register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Y_OFS_USR_7 | Y_OFS_USR_6 | Y_OFS_USR_5 | Y_OFS_USR_4 | Y_OFS_USR_3 | Y_OFS_USR_2 | Y_OFS_USR_1 | Y_OFS_USR_0 |

**Table 146. Y_OFS_USR register description**

| | |
|---|---|
| Y_OFS_USR_[7:0] | Accelerometer Y-axis user offset calibration expressed in 2's complement, weight depends on USR_OFF_W in CTRL6_C (15h). The value must be in the range [-127, +127]. |

## 9.56 Z_OFS_USR (75h)

Accelerometer Z-axis user offset correction (r/w). The offset value set in the Z_OFS_USR offset register is internally subtracted from the acceleration value measured on the Z-axis.

**Table 147. Z_OFS_USR register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Z_OFS_USR_7 | Z_OFS_USR_6 | Z_OFS_USR_5 | Z_OFS_USR_4 | Z_OFS_USR_3 | Z_OFS_USR_2 | Z_OFS_USR_1 | Z_OFS_USR_0 |

**Table 148. Z_OFS_USR register description**

| | |
|---|---|
| Z_OFS_USR_[7:0] | Accelerometer Z-axis user offset calibration expressed in 2's complement, weight depends on USR_OFF_W in CTRL6_C (15h). The value must be in the range [-127, +127]. |

chanhvinh.com



## 9.57 FIFO_DATA_OUT_TAG (78h)

FIFO tag register (r)

**Table 149. FIFO_DATA_OUT_TAG register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| TAG_ SENSOR_4 | TAG_ SENSOR_3 | TAG_ SENSOR_2 | TAG_ SENSOR_1 | TAG_ SENSOR_0 | TAG_CNT_1 | TAG_CNT_0 | TAG_ PARITY |

**Table 150. FIFO_DATA_OUT_TAG register description**

| | |
|---|---|
| TAG_SENSOR_[4:0] | FIFO tag: identifies the sensor in:<br>FIFO_DATA_OUT_X_L (79h) and FIFO_DATA_OUT_X_H (7Ah), FIFO_DATA_OUT_Y_L (7Bh) and FIFO_DATA_OUT_Y_H (7Ch), and FIFO_DATA_OUT_Z_L (7Dh) and FIFO_DATA_OUT_Z_H (7Eh)<br>For details, refer to Table 151 |
| TAG_CNT_[1:0] | 2-bit counter which identifies sensor time slot |
| TAG_PARITY | Parity check of TAG content |

**Table 151. FIFO tag**

| TAG_SENSOR_[4:0] | Sensor name |
|---|---|
| 0x01 | Gyroscope NC |
| 0x02 | Accelerometer NC |
| 0x03 | Temperature |
| 0x04 | Timestamp |
| 0x05 | CFG_Change |
| 0x06 | Accelerometer NC_T_2 |
| 0x07 | Accelerometer NC_T_1 |
| 0x08 | Accelerometer 2xC |
| 0x09 | Accelerometer 3xC |
| 0x0A | Gyroscope NC_T_2 |
| 0x0B | Gyroscope NC_T_1 |
| 0x0C | Gyroscope 2xC |
| 0x0D | Gyroscope 3xC |
| 0x0E | Sensor Hub Slave 0 |
| 0x0F | Sensor Hub Slave 1 |
| 0x10 | Sensor Hub Slave 2 |
| 0x11 | Sensor Hub Slave 3 |
| 0x12 | Step Counter |
| 0x19 | Sensor Hub Nack |

chanhvinh.com



## 9.58 FIFO_DATA_OUT_X_L (79h) and FIFO_DATA_OUT_X_H (7Ah)

FIFO data output X (r)

**Table 152. FIFO_DATA_OUT_X_H and FIFO_DATA_OUT_X_L registers**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 |
| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |

**Table 153. FIFO_DATA_OUT_X_H and FIFO_DATA_OUT_X_L register description**

| | |
|---|---|
| D[15:0] | FIFO X-axis output |

## 9.59 FIFO_DATA_OUT_Y_L (7Bh) and FIFO_DATA_OUT_Y_H (7Ch)

FIFO data output Y (r)

**Table 154. FIFO_DATA_OUT_Y_H and FIFO_DATA_OUT_Y_L registers**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 |
| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |

**Table 155. FIFO_DATA_OUT_Y_H and FIFO_DATA_OUT_Y_L register description**

| | |
|---|---|
| D[15:0] | FIFO Y-axis output |

## 9.60 FIFO_DATA_OUT_Z_L (7Dh) and FIFO_DATA_OUT_Z_H (7Eh)

FIFO data output Z (r)

**Table 156. FIFO_DATA_OUT_Z_H and FIFO_DATA_OUT_Z_L registers**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 |
| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |

**Table 157. FIFO_DATA_OUT_Z_H and FIFO_DATA_OUT_Z_L register description**

| | |
|---|---|
| D[15:0] | FIFO Z-axis output |

chanhvinh.com



# 10 Embedded functions register mapping

The table given below provides a list of the registers for the embedded functions available in the device and the corresponding addresses. Embedded functions registers are accessible when FUNC_CFG_EN is set to '1' in FUNC_CFG_ACCESS (01h).

**Table 158. Register address map - embedded functions**

| Name | Type | Register address | | Default | Comment |
|---|---|---|---|---|---|
| | | Hex | Binary | | |
| PAGE_SEL | r/w | 02 | 00000010 | 00000001 | |
| EMB_FUNC_EN_A | r/w | 04 | 00000100 | 00000000 | |
| EMB_FUNC_EN_B | r/w | 05 | 00000101 | 00000000 | |
| PAGE_ADDRESS | r/w | 08 | 00001000 | 00000000 | |
| PAGE_VALUE | r/w | 09 | 00001001 | 00000000 | |
| EMB_FUNC_INT1 | r/w | 0A | 00001010 | 00000000 | |
| FSM_INT1_A | r/w | 0B | 00001011 | 00000000 | |
| FSM_INT1_B | r/w | 0C | 00001100 | 00000000 | |
| MLC_INT1 | r/w | 0D | 00001101 | 00000000 | |
| EMB_FUNC_INT2 | r/w | 0E | 00001110 | 00000000 | |
| FSM_INT2_A | r/w | 0F | 00001111 | 00000000 | |
| FSM_INT2_B | r/w | 10 | 00010000 | 00000000 | |
| MLC_INT2 | r/w | 11 | 00010001 | 00000000 | |
| EMB_FUNC_STATUS | r | 12 | 00010010 | output | |
| FSM_STATUS_A | r | 13 | 00010011 | output | |
| FSM_STATUS_B | r | 14 | 00010100 | output | |
| MLC_STATUS (15h) | r | 15 | 00010101 | output | |
| PAGE_RW | r/w | 17 | 00010111 | 00000000 | |
| RESERVED | - | 18-43 | | | |
| EMB_FUNC_FIFO_CFG | r/w | 44 | 01000100 | 00000000 | |
| FSM_ENABLE_A | r/w | 46 | 01000110 | 00000000 | |
| FSM_ENABLE_B | r/w | 47 | 01000111 | 00000000 | |
| FSM_LONG_COUNTER_L | r/w | 48 | 01001000 | 00000000 | |
| FSM_LONG_COUNTER_H | r/w | 49 | 01001001 | 00000000 | |
| FSM_LONG_COUNTER_CLEAR | r/w | 4A | 01001010 | 00000000 | |
| FSM_OUTS1 | r | 4C | 01001100 | output | |
| FSM_OUTS2 | r | 4D | 01001101 | output | |
| FSM_OUTS3 | r | 4E | 01001110 | output | |
| FSM_OUTS4 | r | 4F | 01001111 | output | |
| FSM_OUTS5 | r | 50 | 01010000 | output | |
| FSM_OUTS6 | r | 51 | 01010001 | output | |
| FSM_OUTS7 | r | 52 | 01010010 | output | |
| FSM_OUTS8 | r | 53 | 01010011 | output | |

chanhvinh.com



| Name | Type | Register address | | Default | Comment |
|---|---|---|---|---|---|
| | | Hex | Binary | | |
| FSM_OUTS9 | r | 54 | 01010100 | output | |
| FSM_OUTS10 | r | 55 | 01010101 | output | |
| FSM_OUTS11 | r | 56 | 01010110 | output | |
| FSM_OUTS12 | r | 57 | 01010111 | output | |
| FSM_OUTS13 | r | 58 | 01011000 | output | |
| FSM_OUTS14 | r | 59 | 01011001 | output | |
| FSM_OUTS15 | r | 5A | 01011010 | output | |
| FSM_OUTS16 | r | 5B | 01011011 | output | |
| RESERVED | - | 5E | 01011110 | | |
| EMB_FUNC_ODR_CFG_B | r/w | 5F | 01011111 | 01001011 | |
| EMB_FUNC_ODR_CFG_C | r/w | 60 | 01100000 | 00010101 | |
| STEP_COUNTER_L | r | 62 | 01100010 | output | |
| STEP_COUNTER_H | r | 63 | 01100011 | output | |
| EMB_FUNC_SRC | r/w | 64 | 01100100 | output | |
| EMB_FUNC_INIT_A | r/w | 66 | 01100110 | 00000000 | |
| EMB_FUNC_INIT_B | r/w | 67 | 01100111 | 00000000 | |
| MLC0_SRC | r | 70 | 01110000 | output | |
| MLC1_SRC | r | 71 | 01110001 | output | |
| MLC2_SRC | r | 72 | 01110010 | output | |
| MLC3_SRC | r | 73 | 01110011 | output | |
| MLC4_SRC | r | 74 | 01110100 | output | |
| MLC5_SRC | r | 75 | 01110101 | output | |
| MLC6_SRC | r | 76 | 01110110 | output | |
| MLC7_SRC | r | 77 | 01110111 | output | |

Registers marked as Reserved must not be changed. Writing to those registers may cause permanent damage to the device.

The content of the registers that are loaded at boot should not be changed. They contain the factory calibration values. Their content is automatically restored when the device is powered up.

chanhvinh.com



# 11 Embedded functions register description

## 11.1 PAGE_SEL (02h)

Enable advanced features dedicated page (r/w)

**Table 159. PAGE_SEL register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| PAGE_SEL3 | PAGE_SEL2 | PAGE_SEL1 | PAGE_SEL0 | 0(1) | 0(1) | 0(1) | 1(2) |

1. *This bit must be set to '0' for the correct operation of the device.*
2. *This bit must be set to '1' for the correct operation of the device.*

**Table 160. PAGE_SEL register description**

| | |
|---|---|
| PAGE_SEL[3:0] | Select the advanced features dedicated page. Default value: 0000 |

## 11.2 EMB_FUNC_EN_A (04h)

Embedded functions enable register (r/w)

**Table 161. EMB_FUNC_EN_A register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | 0(1) | SIGN_ MOTION_EN | TILT_EN | PEDO_EN | 0(1) | 0(1) | 0(1) |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 162. EMB_FUNC_EN_A register description**

| | |
|---|---|
| SIGN_MOTION_EN | Enable significant motion detection function. Default value: 0<br>(0: significant motion detection function disabled;<br>1: significant motion detection function enabled) |
| TILT_EN | Enable tilt calculation. Default value: 0<br>(0: tilt algorithm disabled;<br>1: tilt algorithm enabled) |
| PEDO_EN | Enable pedometer algorithm. Default value: 0<br>(0: pedometer algorithm disabled;<br>1: pedometer algorithm enabled) |

chanhvinh.com



## 11.3 EMB_FUNC_EN_B (05h)

Embedded functions enable register (r/w)

**Table 163. EMB_FUNC_EN_B register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | 0(1) | 0(1) | MLC_EN | FIFO_ COMPR_EN | 0(1) | 0(1) | FSM_EN |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 164. EMB_FUNC_EN_B register description**

| | |
|---|---|
| MLC_EN | Enable Machine Learning Core feature. Default value: 0<br>(0: Machine Learning Core feature disabled;<br>1: Machine Learning Core feature enabled) |
| FIFO_COMPR_EN(1) | Enable FIFO compression feature. Default value: 0<br>(0: FIFO compression feature disabled;<br>1: FIFO compression feature enabled) |
| FSM_EN | Enable Finite State Machine (FSM) feature. Default value: 0<br>(0: FSM feature disabled; 1: FSM feature enabled) |

1. *This bit is effective if the FIFO_COMPR_RT_EN bit of FIFO_CTRL2 (08h) is set to 1.*

## 11.4 PAGE_ADDRESS (08h)

Page address register (r/w)

**Table 165. PAGE_ADDRESS register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| PAGE_ ADDR7 | PAGE_ ADDR6 | PAGE_ ADDR5 | PAGE_ ADDR4 | PAGE_ ADDR3 | PAGE_ ADDR2 | PAGE_ ADDR1 | PAGE_ ADDR0 |

**Table 166. PAGE_ADDRESS register description**

| | |
|---|---|
| PAGE_ADDR[7:0] | After setting the bit PAGE_WRITE / PAGE_READ in register PAGE_RW (17h), this register is used to set the address of the register to be written/read in the advanced features page selected through the bits PAGE_SEL[3:0] in register PAGE_SEL (02h). |

chanhvinh.com



## 11.5 PAGE_VALUE (09h)

Page value register (r/w)

**Table 167. PAGE_VALUE register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| PAGE_ VALUE7 | PAGE_ VALUE6 | PAGE_ VALUE5 | PAGE_ VALUE4 | PAGE_ VALUE3 | PAGE_ VALUE2 | PAGE_ VALUE1 | PAGE_ VALUE0 |

**Table 168. PAGE_VALUE register description**

| | |
|---|---|
| PAGE_VALUE[7:0] | These bits are used to write (if the bit PAGE_WRITE = 1 in register PAGE_RW (17h)) or read (if the bit PAGE_READ = 1 in register PAGE_RW (17h)) the data at the address PAGE_ADDR[7:0] of the selected advanced features page. |

## 11.6 EMB_FUNC_INT1 (0Ah)

INT1 pin control register (r/w)

Each bit in this register enables a signal to be carried over INT1. The pin's output will supply the OR combination of the selected signals.

**Table 169. EMB_FUNC_INT1 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| INT1_ FSM_LC | 0(1) | INT1_ SIG_MOT | INT1_TILT | INT1_STEP_ DETECTOR | 0(1) | 0(1) | 0(1) |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 170. EMB_FUNC_INT1 register description**

| | |
|---|---|
| INT1_FSM_LC(1) | Routing of FSM long counter timeout interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_SIG_MOT(1) | Routing of significant motion event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_TILT(1) | Routing of tilt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_STEP_DETECTOR(1) | Routing of pedometer step recognition event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |

1. *This bit is effective if the INT1_EMB_FUNC bit of MD1_CFG (5Eh) is set to 1.*

chanhvinh.com



## 11.7 FSM_INT1_A (0Bh)

INT1 pin control register (r/w)

Each bit in this register enables a signal to be carried over INT1. The pin's output will supply the OR combination of the selected signals.

**Table 171. FSM_INT1_A register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| INT1_FSM8 | INT1_FSM7 | INT1_FSM6 | INT1_FSM5 | INT1_FSM4 | INT1_FSM3 | INT1_FSM2 | INT1_FSM1 |

**Table 172. FSM_INT1_A register description**

| | |
|---|---|
| INT1_FSM8(1) | Routing of FSM8 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_FSM7(1) | Routing of FSM7 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_FSM6(1) | Routing of FSM6 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_FSM5(1) | Routing of FSM5 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_FSM4(1) | Routing of FSM4 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_FSM3(1) | Routing of FSM3 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_FSM2(1) | Routing of FSM2 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_FSM1(1) | Routing of FSM1 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |

1. *This bit is effective if the INT1_EMB_FUNC bit of MD1_CFG (5Eh) is set to 1.*

chanhvinh.com



## 11.8 FSM_INT1_B (0Ch)

INT1 pin control register (r/w)

Each bit in this register enables a signal to be carried over INT1. The pin's output will supply the OR combination of the selected signals.

**Table 173. FSM_INT1_B register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| INT1_FSM16 | INT1_FSM15 | INT1_FSM14 | INT1_FSM13 | INT1_FSM12 | INT1_FSM11 | INT1_FSM10 | INT1_FSM9 |

**Table 174. FSM_INT1_B register description**

| | |
|---|---|
| INT1_FSM16(1) | Routing of FSM16 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_FSM15(1) | Routing of FSM15 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_FSM14(1) | Routing of FSM14 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_FSM13(1) | Routing of FSM13 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_FSM12(1) | Routing of FSM12 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_FSM11(1) | Routing of FSM11 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_FSM10(1) | Routing of FSM10 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_FSM9(1) | Routing of FSM9 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |

1. *This bit is effective if the INT1_EMB_FUNC bit of MD1_CFG (5Eh) is set to 1.*

chanhvinh.com



## 11.9 MLC_INT1 (0Dh)

INT1 pin control register (r/w)

Each bit in this register enables a signal to be carried over INT1. The pin's output will supply the OR combination of the selected signals.

**Table 175. MLC_INT1 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| INT1_MLC8 | INT1_MLC7 | INT1_MLC6 | INT1_MLC5 | INT1_MLC4 | INT1_MLC3 | INT1_MLC2 | INT1_MLC1 |

**Table 176. MLC_INT1 register description**

| | |
|---|---|
| INT1_MLC8 | Routing of MLC8 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_MLC7 | Routing of MLC7 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_MLC6 | Routing of MLC6 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_MLC5 | Routing of MLC5 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_MLC4 | Routing of MLC4 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_MLC3 | Routing of MLC3 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_MLC2 | Routing of MLC2 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |
| INT1_MLC1 | Routing of MLC1 interrupt event on INT1. Default value: 0<br>(0: routing on INT1 disabled; 1: routing on INT1 enabled) |

chanhvinh.com



## 11.10 EMB_FUNC_INT2 (0Eh)

INT2 pin control register (r/w)

Each bit in this register enables a signal to be carried over INT2. The pin's output will supply the OR combination of the selected signals.

**Table 177. EMB_FUNC_INT2 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| INT2_ FSM_LC | 0(1) | INT2_ SIG_MOT | INT2_TILT | INT2_STEP_ DETECTOR | 0(1) | 0(1) | 0(1) |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 178. EMB_FUNC_INT2 register description**

| | |
|---|---|
| INT2_FSM_LC(1) | Routing of FSM long counter timeout interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_SIG_MOT(1) | Routing of significant motion event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_TILT(1) | Routing of tilt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_STEP_DETECTOR(1) | Routing of pedometer step recognition event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |

1. *This bit is effective if the INT2_EMB_FUNC bit of MD2_CFG (5Fh) is set to 1.*

chanhvinh.com



## 11.11 FSM_INT2_A (0Fh)

INT2 pin control register (r/w)

Each bit in this register enables a signal to be carried over INT2. The pin's output will supply the OR combination of the selected signals.

**Table 179. FSM_INT2_A register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| INT2_FSM8 | INT2_FSM7 | INT2_FSM6 | INT2_FSM5 | INT2_FSM4 | INT2_FSM3 | INT2_FSM2 | INT2_FSM1 |

**Table 180. FSM_INT2_A register description**

| | |
|---|---|
| INT2_FSM8(1) | Routing of FSM8 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_FSM7(1) | Routing of FSM7 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_FSM6(1) | Routing of FSM6 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_FSM5(1) | Routing of FSM5 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_FSM4(1) | Routing of FSM4 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_FSM3(1) | Routing of FSM3 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_FSM2(1) | Routing of FSM2 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_FSM1(1) | Routing of FSM1 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |

1. *This bit is effective if the INT2_EMB_FUNC bit of MD2_CFG (5Fh) is set to 1.*

chanhvinh.com



## 11.12 FSM_INT2_B (10h)

INT2 pin control register (r/w)

Each bit in this register enables a signal to be carried over INT2. The pin's output will supply the OR combination of the selected signals.

**Table 181. FSM_INT2_B register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| INT2_FSM16 | INT2_FSM15 | INT2_FSM14 | INT2_FSM13 | INT2_FSM12 | INT2_FSM11 | INT2_FSM10 | INT2_FSM9 |

**Table 182. FSM_INT2_B register description**

| | |
|---|---|
| INT2_FSM16(1) | Routing of FSM16 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_FSM15(1) | Routing of FSM15 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_FSM14(1) | Routing of FSM14 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_FSM13(1) | Routing of FSM13 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_FSM12(1) | Routing of FSM12 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_FSM11(1) | Routing of FSM11 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_FSM10(1) | Routing of FSM10 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_FSM9(1) | Routing of FSM9 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |

1. *This bit is effective if the INT2_EMB_FUNC bit of MD2_CFG (5Fh) is set to 1.*

chanhvinh.com



## 11.13 MLC_INT2 (11h)

INT2 pin control register (r/w)

Each bit in this register enables a signal to be carried over INT2. The pin's output will supply the OR combination of the selected signals.

**Table 183. MLC_INT2 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| INT2_MLC8 | INT2_MLC7 | INT2_MLC6 | INT2_MLC5 | INT2_MLC4 | INT2_MLC3 | INT2_MLC2 | INT2_MLC1 |

**Table 184. MLC_INT2 register description**

| | |
|---|---|
| INT2_MLC8 | Routing of MLC8 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_MLC7 | Routing of MLC7 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_MLC6 | Routing of MLC6 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_MLC5 | Routing of MLC5 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_MLC4 | Routing of MLC4 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_MLC3 | Routing of MLC3 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_MLC2 | Routing of MLC2 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |
| INT2_MLC1 | Routing of MLC1 interrupt event on INT2. Default value: 0<br>(0: routing on INT2 disabled; 1: routing on INT2 enabled) |

## 11.14 EMB_FUNC_STATUS (12h)

Embedded function status register (r)

**Table 185. EMB_FUNC_STATUS register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| IS_<br>FSM_LC | 0 | IS_<br>SIGMOT | IS_TILT | IS_<br>STEP_DET | 0 | 0 | 0 |

**Table 186. EMB_FUNC_STATUS register description**

| | |
|---|---|
| IS_FSM_LC | Interrupt status bit for FSM long counter timeout interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_SIGMOT | Interrupt status bit for significant motion detection. (1: interrupt detected; 0: no interrupt) |
| IS_TILT | Interrupt status bit for tilt detection. (1: interrupt detected; 0: no interrupt) |
| IS_STEP_DET | Interrupt status bit for step detection. (1: interrupt detected; 0: no interrupt) |

chanhvinh.com



## 11.15 FSM_STATUS_A (13h)

Finite State Machine status register (r)

**Table 187. FSM_STATUS_A register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| IS_FSM8 | IS_FSM7 | IS_FSM6 | IS_FSM5 | IS_FSM4 | IS_FSM3 | IS_FSM2 | IS_FSM1 |

**Table 188. FSM_STATUS_A register description**

| | |
|---|---|
| IS_FSM8 | Interrupt status bit for FSM8 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM7 | Interrupt status bit for FSM7 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM6 | Interrupt status bit for FSM6 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM5 | Interrupt status bit for FSM5 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM4 | Interrupt status bit for FSM4 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM3 | Interrupt status bit for FSM3 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM2 | Interrupt status bit for FSM2 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM1 | Interrupt status bit for FSM1 interrupt event. (1: interrupt detected; 0: no interrupt) |

## 11.16 FSM_STATUS_B (14h)

Finite State Machine status register (r)

**Table 189. FSM_STATUS_B register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| IS_FSM16 | IS_FSM15 | IS_FSM14 | IS_FSM13 | IS_FSM12 | IS_FSM11 | IS_FSM10 | IS_FSM9 |

**Table 190. FSM_STATUS_B register description**

| | |
|---|---|
| IS_FSM16 | Interrupt status bit for FSM16 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM15 | Interrupt status bit for FSM15 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM14 | Interrupt status bit for FSM14 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM13 | Interrupt status bit for FSM13 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM12 | Interrupt status bit for FSM12 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM11 | Interrupt status bit for FSM11 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM10 | Interrupt status bit for FSM10 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_FSM9 | Interrupt status bit for FSM9 interrupt event. (1: interrupt detected; 0: no interrupt) |

chanhvinh.com



## 11.17 MLC_STATUS (15h)

Machine Learning Core status register (r).

**Table 191. MLC_STATUS register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| IS_MLC8 | IS_MLC7 | IS_MLC6 | IS_MLC5 | IS_MLC4 | IS_MLC3 | IS_MLC | IS_MLC1 |

**Table 192. MLC_STATUS register description**

| | |
|---|---|
| IS_MLC8 | Interrupt status bit for MLC8 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_MLC7 | Interrupt status bit for MLC7 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_MLC6 | Interrupt status bit for MLC6 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_MLC5 | Interrupt status bit for MLC5 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_MLC4 | Interrupt status bit for MLC4 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_MLC3 | Interrupt status bit for MLC3 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_MLC2 | Interrupt status bit for MLC2 interrupt event. (1: interrupt detected; 0: no interrupt) |
| IS_MLC1 | Interrupt status bit for MLC1 interrupt event. (1: interrupt detected; 0: no interrupt) |

## 11.18 PAGE_RW (17h)

Enable read and write mode of advanced features dedicated page (r/w)

**Table 193. PAGE_RW register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| EMB_ FUNC_LIR | PAGE_ WRITE | PAGE_ READ | 0(1) | 0(1) | 0(1) | 0(1) | 0(1) |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 194. PAGE_RW register description**

| | |
|---|---|
| EMB_FUNC_LIR | Latched Interrupt mode for embedded functions. Default value: 0<br>(0: embedded functions interrupt request not latched;<br>1: embedded functions interrupt request latched) |
| PAGE_WRITE | Enable writes to the selected advanced features dedicated page. (1) Default value: 0<br>(1: enable; 0: disable) |
| PAGE_READ | Enable reads from the selected advanced features dedicated page.(1) Default value: 0<br>(1: enable; 0: disable) |

1. *Page selected by PAGE_SEL[3:0] in PAGE_SEL (02h) register.*

chanhvinh.com



## 11.19 EMB_FUNC_FIFO_CFG (44h)

Embedded functions batching configuration register (r/w)

**Table 195. EMB_FUNC_FIFO_CFG register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | PEDO_FIFO_EN | 0(1) | 0(1) | 0(1) | 0(1) | 0(1) | 0(1) |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 196. EMB_FUNC_FIFO_CFG register description**

| | |
|---|---|
| PEDO_FIFO_EN | Enable FIFO batching of step counter values. Default value: 0 |

## 11.20 FSM_ENABLE_A (46h)

FSM enable register (r/w)

**Table 197. FSM_ENABLE_A register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| FSM8_EN | FSM7_EN | FSM6_EN | FSM5_EN | FSM4_EN | FSM3_EN | FSM2_EN | FSM1_EN |

**Table 198. FSM_ENABLE_A register description**

| | |
|---|---|
| FSM8_EN | FSM8 enable. Default value: 0 (0: FSM8 disabled; 1: FSM8 enabled) |
| FSM7_EN | FSM7 enable. Default value: 0 (0: FSM7 disabled; 1: FSM7 enabled) |
| FSM6_EN | FSM6 enable. Default value: 0 (0: FSM6 disabled; 1: FSM6 enabled) |
| FSM5_EN | FSM5 enable. Default value: 0 (0: FSM5 disabled; 1: FSM5 enabled) |
| FSM4_EN | FSM4 enable. Default value: 0 (0: FSM4 disabled; 1: FSM4 enabled) |
| FSM3_EN | FSM3 enable. Default value: 0 (0: FSM3 disabled; 1: FSM3 enabled) |
| FSM2_EN | FSM2 enable. Default value: 0 (0: FSM2 disabled; 1: FSM2 enabled) |
| FSM1_EN | FSM1 enable. Default value: 0 (0: FSM1 disabled; 1: FSM1 enabled) |

## 11.21 FSM_ENABLE_B (47h)

FSM enable register (r/w)

**Table 199. FSM_ENABLE_B register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| FSM16_EN | FSM15_EN | FSM14_EN | FSM13_EN | FSM12_EN | FSM11_EN | FSM10_EN | FSM9_EN |

**Table 200. FSM_ENABLE_B register description**

| | |
|---|---|
| FSM16_EN | FSM16 enable. Default value: 0 (0: FSM16 disabled; 1: FSM16 enabled) |
| FSM15_EN | FSM15 enable. Default value: 0 (0: FSM15 disabled; 1: FSM15 enabled) |
| FSM14_EN | FSM14 enable. Default value: 0 (0: FSM14 disabled; 1: FSM14 enabled) |
| FSM13_EN | FSM13 enable. Default value: 0 (0: FSM13 disabled; 1: FSM13 enabled) |
| FSM12_EN | FSM12 enable. Default value: 0 (0: FSM12 disabled; 1: FSM12 enabled) |
| FSM11_EN | FSM11 enable. Default value: 0 (0: FSM11 disabled; 1: FSM11 enabled) |
| FSM10_EN | FSM10 enable. Default value: 0 (0: FSM10 disabled; 1: FSM10 enabled) |
| FSM9_EN | FSM9 enable. Default value: 0 (0: FSM9 disabled; 1: FSM9 enabled) |

chanhvinh.com



## 11.22 FSM_LONG_COUNTER_L (48h) and FSM_LONG_COUNTER_H (49h)

FSM long counter status register (r/w)

Long counter value is an unsigned integer value (16-bit format); this value can be reset using the LC_CLEAR bit in FSM_LONG_COUNTER_CLEAR (4Ah) register.

**Table 201. FSM_LONG_COUNTER_L register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| FSM_LC_7 | FSM_LC_6 | FSM_LC_5 | FSM_LC_4 | FSM_LC_3 | FSM_LC_2 | FSM_LC_1 | FSM_LC_0 |

**Table 202. FSM_LONG_COUNTER_L register description**

| | |
|---|---|
| FSM_LC_[7:0] | Long counter current value (LSbyte). Default value: 00000000 |

**Table 203. FSM_LONG_COUNTER_H register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| FSM_LC_15 | FSM_LC_14 | FSM_LC_13 | FSM_LC_12 | FSM_LC_11 | FSM_LC_10 | FSM_LC_9 | FSM_LC_8 |

**Table 204. FSM_LONG_COUNTER_H register description**

| | |
|---|---|
| FSM_LC_[15:8] | Long counter current value (MSbyte). Default value: 00000000 |

## 11.23 FSM_LONG_COUNTER_CLEAR (4Ah)

FSM long counter reset register (r/w)

**Table 205. FSM_LONG_COUNTER_CLEAR register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | 0(1) | 0(1) | 0(1) | 0(1) | 0(1) | FSM_LC_CLEARED | FSM_LC_CLEAR |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 206. FSM_LONG_COUNTER_CLEAR register description**

| | |
|---|---|
| FSM_LC_CLEARED | This read-only bit is automatically set to 1 when the long counter reset is done. Default value: 0 |
| FSM_LC_CLEAR | Clear FSM long counter value. Default value: 0 |

chanhvinh.com



## 11.24 FSM_OUTS1 (4Ch)

FSM1 output register (r)

**Table 207. FSM_OUTS1 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| P_X | N_X | P_Y | N_Y | P_Z | N_Z | P_V | N_V |

**Table 208. FSM_OUTS1 register description**

| | |
|---|---|
| P_X | FSM1 output: positive event detected on the X-axis. (0: event not detected; 1: event detected) |
| N_X | FSM1 output: negative event detected on the X-axis. (0: event not detected; 1: event detected) |
| P_Y | FSM1 output: positive event detected on the Y-axis. (0: event not detected; 1: event detected) |
| N_Y | FSM1 output: negative event detected on the Y-axis. (0: event not detected; 1: event detected) |
| P_Z | FSM1 output: positive event detected on the Z-axis. (0: event not detected; 1: event detected) |
| N_Z | FSM1 output: negative event detected on the Z-axis. (0: event not detected; 1: event detected) |
| P_V | FSM1 output: positive event detected on the vector. (0: event not detected; 1: event detected) |
| N_V | FSM1 output: negative event detected on the vector. (0: event not detected; 1: event detected) |

## 11.25 FSM_OUTS2 (4Dh)

FSM2 output register (r)

**Table 209. FSM_OUTS2 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| P_X | N_X | P_Y | N_Y | P_Z | N_Z | P_V | N_V |

**Table 210. FSM_OUTS2 register description**

| | |
|---|---|
| P_X | FSM2 output: positive event detected on the X-axis. (0: event not detected; 1: event detected) |
| N_X | FSM2 output: negative event detected on the X-axis. (0: event not detected; 1: event detected) |
| P_Y | FSM2 output: positive event detected on the Y-axis. (0: event not detected; 1: event detected) |
| N_Y | FSM2 output: negative event detected on the Y-axis. (0: event not detected; 1: event detected) |
| P_Z | FSM2 output: positive event detected on the Z-axis. (0: event not detected; 1: event detected) |
| N_Z | FSM2 output: negative event detected on the Z-axis. (0: event not detected; 1: event detected) |
| P_V | FSM2 output: positive event detected on the vector. (0: event not detected; 1: event detected) |
| N_V | FSM2 output: negative event detected on the vector. (0: event not detected; 1: event detected) |

chanhvinh.com



## 11.26 FSM_OUTS3 (4Eh)

FSM3 output register (r)

**Table 211. FSM_OUTS3 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| P_X | N_X | P_Y | N_Y | P_Z | N_Z | P_V | N_V |

**Table 212. FSM_OUTS3 register description**

| | |
|---|---|
| P_X | FSM3 output: positive event detected on the X-axis. (0: event not detected; 1: event detected) |
| N_X | FSM3 output: negative event detected on the X-axis. (0: event not detected; 1: event detected) |
| P_Y | FSM3 output: positive event detected on the Y-axis. (0: event not detected; 1: event detected) |
| N_Y | FSM3 output: negative event detected on the Y-axis. (0: event not detected; 1: event detected) |
| P_Z | FSM3 output: positive event detected on the Z-axis. (0: event not detected; 1: event detected) |
| N_Z | FSM3 output: negative event detected on the Z-axis. (0: event not detected; 1: event detected) |
| P_V | FSM3 output: positive event detected on the vector. (0: event not detected; 1: event detected) |
| N_V | FSM3 output: negative event detected on the vector. (0: event not detected; 1: event detected) |

## 11.27 FSM_OUTS4 (4Fh)

FSM4 output register (r)

**Table 213. FSM_OUTS4 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| P_X | N_X | P_Y | N_Y | P_Z | N_Z | P_V | N_V |

**Table 214. FSM_OUTS4 register description**

| | |
|---|---|
| P_X | FSM4 output: positive event detected on the X-axis. (0: event not detected; 1: event detected) |
| N_X | FSM4 output: negative event detected on the X-axis. (0: event not detected; 1: event detected) |
| P_Y | FSM4 output: positive event detected on the Y-axis. (0: event not detected; 1: event detected) |
| N_Y | FSM4 output: negative event detected on the Y-axis. (0: event not detected; 1: event detected) |
| P_Z | FSM4 output: positive event detected on the Z-axis. (0: event not detected; 1: event detected) |
| N_Z | FSM4 output: negative event detected on the Z-axis. (0: event not detected; 1: event detected) |
| P_V | FSM4 output: positive event detected on the vector. (0: event not detected; 1: event detected) |
| N_V | FSM4 output: negative event detected on the vector. (0: event not detected; 1: event detected) |

chanhvinh.com



## 11.28 FSM_OUTS5 (50h)

FSM5 output register (r)

Table 215. **FSM_OUTS5 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| P_X | N_X | P_Y | N_Y | P_Z | N_Z | P_V | N_V |

Table 216. **FSM_OUTS5 register description**

| | |
|---|---|
| P_X | FSM5 output: positive event detected on the X-axis. (0: event not detected; 1: event detected) |
| N_X | FSM5 output: negative event detected on the X-axis. (0: event not detected; 1: event detected) |
| P_Y | FSM5 output: positive event detected on the Y-axis. (0: event not detected; 1: event detected) |
| N_Y | FSM5 output: negative event detected on the Y-axis. (0: event not detected; 1: event detected) |
| P_Z | FSM5 output: positive event detected on the Z-axis. (0: event not detected; 1: event detected) |
| N_Z | FSM5 output: negative event detected on the Z-axis. (0: event not detected; 1: event detected) |
| P_V | FSM5 output: positive event detected on the vector. (0: event not detected; 1: event detected) |
| N_V | FSM5 output: negative event detected on the vector. (0: event not detected; 1: event detected) |

## 11.29 FSM_OUTS6 (51h)

FSM6 output register (r)

Table 217. **FSM_OUTS6 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| P_X | N_X | P_Y | N_Y | P_Z | N_Z | P_V | N_V |

Table 218. **FSM_OUTS6 register description**

| | |
|---|---|
| P_X | FSM6 output: positive event detected on the X-axis. (0: event not detected; 1: event detected) |
| N_X | FSM6 output: negative event detected on the X-axis. (0: event not detected; 1: event detected) |
| P_Y | FSM6 output: positive event detected on the Y-axis. (0: event not detected; 1: event detected) |
| N_Y | FSM6 output: negative event detected on the Y-axis. (0: event not detected; 1: event detected) |
| P_Z | FSM6 output: positive event detected on the Z-axis. (0: event not detected; 1: event detected) |
| N_Z | FSM6 output: negative event detected on the Z-axis. (0: event not detected; 1: event detected) |
| P_V | FSM6 output: positive event detected on the vector. (0: event not detected; 1: event detected) |
| N_V | FSM6 output: negative event detected on the vector. (0: event not detected; 1: event detected) |

chanhvinh.com



## 11.30 FSM_OUTS7 (52h)

FSM7 output register (r)

Table 219. **FSM_OUTS7 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| P_X | N_X | P_Y | N_Y | P_Z | N_Z | P_V | N_V |

Table 220. **FSM_OUTS7 register description**

| | |
|---|---|
| P_X | FSM7 output: positive event detected on the X-axis. (0: event not detected; 1: event detected) |
| N_X | FSM7 output: negative event detected on the X-axis. (0: event not detected; 1: event detected) |
| P_Y | FSM7 output: positive event detected on the Y-axis. (0: event not detected; 1: event detected) |
| N_Y | FSM7 output: negative event detected on the Y-axis. (0: event not detected; 1: event detected) |
| P_Z | FSM7 output: positive event detected on the Z-axis. (0: event not detected; 1: event detected) |
| N_Z | FSM7 output: negative event detected on the Z-axis. (0: event not detected; 1: event detected) |
| P_V | FSM7 output: positive event detected on the vector. (0: event not detected; 1: event detected) |
| N_V | FSM7 output: negative event detected on the vector. (0: event not detected; 1: event detected) |

## 11.31 FSM_OUTS8 (53h)

FSM8 output register (r)

Table 221. **FSM_OUTS8 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| P_X | N_X | P_Y | N_Y | P_Z | N_Z | P_V | N_V |

Table 222. **FSM_OUTS8 register description**

| | |
|---|---|
| P_X | FSM8 output: positive event detected on the X-axis. (0: event not detected; 1: event detected) |
| N_X | FSM8 output: negative event detected on the X-axis. (0: event not detected; 1: event detected) |
| P_Y | FSM8 output: positive event detected on the Y-axis. (0: event not detected; 1: event detected) |
| N_Y | FSM8 output: negative event detected on the Y-axis. (0: event not detected; 1: event detected) |
| P_Z | FSM8 output: positive event detected on the Z-axis. (0: event not detected; 1: event detected) |
| N_Z | FSM8 output: negative event detected on the Z-axis. (0: event not detected; 1: event detected) |
| P_V | FSM8 output: positive event detected on the vector. (0: event not detected; 1: event detected) |
| N_V | FSM8 output: negative event detected on the vector. (0: event not detected; 1: event detected) |

chanhvinh.com



## 11.32 FSM_OUTS9 (54h)

FSM9 output register (r)

Table 223. **FSM_OUTS9 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| P_X | N_X | P_Y | N_Y | P_Z | N_Z | P_V | N_V |

Table 224. **FSM_OUTS9 register description**

| | |
|---|---|
| P_X | FSM9 output: positive event detected on the X-axis. (0: event not detected; 1: event detected) |
| N_X | FSM9 output: negative event detected on the X-axis. (0: event not detected; 1: event detected) |
| P_Y | FSM9 output: positive event detected on the Y-axis. (0: event not detected; 1: event detected) |
| N_Y | FSM9 output: negative event detected on the Y-axis. (0: event not detected; 1: event detected) |
| P_Z | FSM9 output: positive event detected on the Z-axis. (0: event not detected; 1: event detected) |
| N_Z | FSM9 output: negative event detected on the Z-axis. (0: event not detected; 1: event detected) |
| P_V | FSM9 output: positive event detected on the vector. (0: event not detected; 1: event detected) |
| N_V | FSM9 output: negative event detected on the vector. (0: event not detected; 1: event detected) |

## 11.33 FSM_OUTS10 (55h)

FSM10 output register (r)

Table 225. **FSM_OUTS10 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| P_X | N_X | P_Y | N_Y | P_Z | N_Z | P_V | N_V |

Table 226. **FSM_OUTS10 register description**

| | |
|---|---|
| P_X | FSM10 output: positive event detected on the X-axis. (0: event not detected; 1: event detected) |
| N_X | FSM10 output: negative event detected on the X-axis. (0: event not detected; 1: event detected) |
| P_Y | FSM10 output: positive event detected on the Y-axis. (0: event not detected; 1: event detected) |
| N_Y | FSM10 output: negative event detected on the Y-axis. (0: event not detected; 1: event detected) |
| P_Z | FSM10 output: positive event detected on the Z-axis. (0: event not detected; 1: event detected) |
| N_Z | FSM10 output: negative event detected on the Z-axis. (0: event not detected; 1: event detected) |
| P_V | FSM10 output: positive event detected on the vector. (0: event not detected; 1: event detected) |
| N_V | FSM10 output: negative event detected on the vector. (0: event not detected; 1: event detected) |

chanhvinh.com



## 11.34 FSM_OUTS11 (56h)

FSM11 output register (r)

Table 227. **FSM_OUTS11 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| P_X | N_X | P_Y | N_Y | P_Z | N_Z | P_V | N_V |

Table 228. **FSM_OUTS11 register description**

| | |
|---|---|
| P_X | FSM11 output: positive event detected on the X-axis. (0: event not detected; 1: event detected) |
| N_X | FSM11 output: negative event detected on the X-axis. (0: event not detected; 1: event detected) |
| P_Y | FSM11 output: positive event detected on the Y-axis. (0: event not detected; 1: event detected) |
| N_Y | FSM11 output: negative event detected on the Y-axis. (0: event not detected; 1: event detected) |
| P_Z | FSM11 output: positive event detected on the Z-axis. (0: event not detected; 1: event detected) |
| N_Z | FSM11 output: negative event detected on the Z-axis. (0: event not detected; 1: event detected) |
| P_V | FSM11 output: positive event detected on the vector. (0: event not detected; 1: event detected) |
| N_V | FSM11 output: negative event detected on the vector. (0: event not detected; 1: event detected) |

## 11.35 FSM_OUTS12 (57h)

FSM12 output register (r)

Table 229. **FSM_OUTS12 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| P_X | N_X | P_Y | N_Y | P_Z | N_Z | P_V | N_V |

Table 230. **FSM_OUTS12 register description**

| | |
|---|---|
| P_X | FSM12 output: positive event detected on the X-axis. (0: event not detected; 1: event detected) |
| N_X | FSM12 output: negative event detected on the X-axis. (0: event not detected; 1: event detected) |
| P_Y | FSM12 output: positive event detected on the Y-axis. (0: event not detected; 1: event detected) |
| N_Y | FSM12 output: negative event detected on the Y-axis. (0: event not detected; 1: event detected) |
| P_Z | FSM12 output: positive event detected on the Z-axis. (0: event not detected; 1: event detected) |
| N_Z | FSM12 output: negative event detected on the Z-axis. (0: event not detected; 1: event detected) |
| P_V | FSM12 output: positive event detected on the vector. (0: event not detected; 1: event detected) |
| N_V | FSM12 output: negative event detected on the vector. (0: event not detected; 1: event detected) |

chanhvinh.com



## 11.36 FSM_OUTS13 (58h)

FSM13 output register (r)

**Table 231. FSM_OUTS13 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| P_X | N_X | P_Y | N_Y | P_Z | N_Z | P_V | N_V |

**Table 232. FSM_OUTS13 register description**

| | |
|---|---|
| P_X | FSM13 output: positive event detected on the X-axis. (0: event not detected; 1: event detected) |
| N_X | FSM13 output: negative event detected on the X-axis. (0: event not detected; 1: event detected) |
| P_Y | FSM13 output: positive event detected on the Y-axis. (0: event not detected; 1: event detected) |
| N_Y | FSM13 output: negative event detected on the Y-axis. (0: event not detected; 1: event detected) |
| P_Z | FSM13 output: positive event detected on the Z-axis. (0: event not detected; 1: event detected) |
| N_Z | FSM13 output: negative event detected on the Z-axis. (0: event not detected; 1: event detected) |
| P_V | FSM13 output: positive event detected on the vector. (0: event not detected; 1: event detected) |
| N_V | FSM13 output: negative event detected on the vector. (0: event not detected; 1: event detected) |

## 11.37 FSM_OUTS14 (59h)

FSM14 output register (r)

**Table 233. FSM_OUTS14 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| P_X | N_X | P_Y | N_Y | P_Z | N_Z | P_V | N_V |

**Table 234. FSM_OUTS14 register description**

| | |
|---|---|
| P_X | FSM14 output: positive event detected on the X-axis. (0: event not detected; 1: event detected) |
| N_X | FSM14 output: negative event detected on the X-axis. (0: event not detected; 1: event detected) |
| P_Y | FSM14 output: positive event detected on the Y-axis. (0: event not detected; 1: event detected) |
| N_Y | FSM14 output: negative event detected on the Y-axis. (0: event not detected; 1: event detected) |
| P_Z | FSM14 output: positive event detected on the Z-axis. (0: event not detected; 1: event detected) |
| N_Z | FSM14 output: negative event detected on the Z-axis. (0: event not detected; 1: event detected) |
| P_V | FSM14 output: positive event detected on the vector. (0: event not detected; 1: event detected) |
| N_V | FSM14 output: negative event detected on the vector. (0: event not detected; 1: event detected) |

chanhvinh.com



## 11.38 FSM_OUTS15 (5Ah)

FSM15 output register (r)

**Table 235. FSM_OUTS15 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| P_X | N_X | P_Y | N_Y | P_Z | N_Z | P_V | N_V |

**Table 236. FSM_OUTS15 register description**

| | |
|---|---|
| P_X | FSM15 output: positive event detected on the X-axis. (0: event not detected; 1: event detected) |
| N_X | FSM15 output: negative event detected on the X-axis. (0: event not detected; 1: event detected) |
| P_Y | FSM15 output: positive event detected on the Y-axis. (0: event not detected; 1: event detected) |
| N_Y | FSM15 output: negative event detected on the Y-axis. (0: event not detected; 1: event detected) |
| P_Z | FSM15 output: positive event detected on the Z-axis. (0: event not detected; 1: event detected) |
| N_Z | FSM15 output: negative event detected on the Z-axis. (0: event not detected; 1: event detected) |
| P_V | FSM15 output: positive event detected on the vector. (0: event not detected; 1: event detected) |
| N_V | FSM15 output: negative event detected on the vector. (0: event not detected; 1: event detected) |

## 11.39 FSM_OUTS16 (5Bh)

FSM16 output register (r)

**Table 237. FSM_OUTS16 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| P_X | N_X | P_Y | N_Y | P_Z | N_Z | P_V | N_V |

**Table 238. FSM_OUTS16 register description**

| | |
|---|---|
| P_X | FSM16 output: positive event detected on the X-axis. (0: event not detected; 1: event detected) |
| N_X | FSM16 output: negative event detected on the X-axis. (0: event not detected; 1: event detected) |
| P_Y | FSM16 output: positive event detected on the Y-axis. (0: event not detected; 1: event detected) |
| N_Y | FSM16 output: negative event detected on the Y-axis. (0: event not detected; 1: event detected) |
| P_Z | FSM16 output: positive event detected on the Z-axis. (0: event not detected; 1: event detected) |
| N_Z | FSM16 output: negative event detected on the Z-axis. (0: event not detected; 1: event detected) |
| P_V | FSM16 output: positive event detected on the vector. (0: event not detected; 1: event detected) |
| N_V | FSM16 output: negative event detected on the vector. (0: event not detected; 1: event detected) |

chanhvinh.com



## 11.40 EMB_FUNC_ODR_CFG_B (5Fh)

Finite State Machine output data rate configuration register (r/w)

**Table 239. EMB_FUNC_ODR_CFG_B register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | 1(2) | 0(1) | FSM_ODR1 | FSM_ODR0 | 0(1) | 1(2) | 1(2) |

1. *This bit must be set to '0' for the correct operation of the device.*
2. *This bit must be set to '1' for the correct operation of the device.*

**Table 240. EMB_FUNC_ODR_CFG_B register description**

| | |
|---|---|
| FSM_ODR[1:0] | Finite State Machine ODR configuration:<br>(00: 12.5 Hz;<br>01: 26 Hz (default);<br>10: 52 Hz;<br>11: 104 Hz) |

## 11.41 EMB_FUNC_ODR_CFG_C (60h)

Machine Learning Core output data rate configuration register (r/w)

**Table 241. EMB_FUNC_ODR_CFG_C register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | 0(1) | MLC_ODR1 | MLC_ODR0 | 0(1) | 1(2) | 0(1) | 1(2) |

1. *This bit must be set to '0' for the correct operation of the device.*
2. *This bit must be set to '1' for the correct operation of the device.*

**Table 242. EMB_FUNC_ODR_CFG_C register description**

| | |
|---|---|
| MLC_ODR[1:0] | Machine Learning Core ODR configuration:<br>(00: 12.5 Hz;<br>01: 26 Hz (default);<br>10: 52 Hz;<br>11: 104 Hz) |

chanhvinh.com



## 11.42 STEP_COUNTER_L (62h) and STEP_COUNTER_H (63h)

Step counter output register (r)

**Table 243. STEP_COUNTER_L register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| STEP_7 | STEP_6 | STEP_5 | STEP_4 | STEP_3 | STEP_2 | STEP_1 | STEP_0 |

**Table 244. STEP_COUNTER_L register description**

| | |
|---|---|
| STEP_[7:0] | Step counter output (LSbyte) |

**Table 245. STEP_COUNTER_H register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| STEP_15 | STEP_14 | STEP_13 | STEP_12 | STEP_11 | STEP_10 | STEP_9 | STEP_8 |

**Table 246. STEP_COUNTER_H register description**

| | |
|---|---|
| STEP_[15:8] | Step counter output (MSbyte) |

## 11.43 EMB_FUNC_SRC (64h)

Embedded function source register (r)

**Table 247. EMB_FUNC_SRC register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| PEDO_RST_STEP | 0 | STEP_DETECTED | STEP_COUNT_DELTA_IA | STEP_OVERFLOW | STEPCOUNTER_BIT_SET | 0 | 0 |

**Table 248. EMB_FUNC_SRC register description**

| | |
|---|---|
| PEDO_RST_STEP | Reset pedometer step counter. Read/write bit.<br>(0: disabled; 1: enabled) |
| STEP_DETECTED | Step detector event detection status. Read-only bit.<br>(0: step detection event not detected; 1: step detection event detected) |
| STEP_COUNT_DELTA_IA | Pedometer step recognition on delta time status. Read-only bit.<br>(0: no step recognized during delta time; 1: at least one step recognized during delta time) |
| STEP_OVERFLOW | Step counter overflow status. Read-only bit.<br>(0: step counter value < $2^{16}$; 1: step counter value reached $2^{16}$) |
| STEPCOUNTER_BIT_SET | This bit is equal to 1 when the step count is increased. If a timer period is programmed in PEDO_SC_DELTAT_L (D0h) and PEDO_SC_DELTAT_H (D1h) embedded advanced features (page 1) registers, this bit is kept to 0. Read-only bit. |

chanhvinh.com



## 11.44 EMB_FUNC_INIT_A (66h)

Embedded functions initialization register (r/w)

**Table 249. EMB_FUNC_INIT_A register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | 0(1) | SIG_MOT_INIT | TILT_INIT | STEP_DET_INIT | 0(1) | 0(1) | 0(1) |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 250. EMB_FUNC_INIT_A register description**

| | |
|---|---|
| SIG_MOT_INIT | Significant motion detection algorithm initialization request. Default value: 0 |
| TILT_INIT | Tilt algorithm initialization request. Default value: 0 |
| STEP_DET_INIT | Pedometer step counter/detector algorithm initialization request. Default value: 0 |

## 11.45 EMB_FUNC_INIT_B (67h)

Embedded functions initialization register (r/w)

**Table 251. EMB_FUNC_INIT_B register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | 0(1) | 0(1) | MLC_INIT | FIFO_COMPR_INIT | 0(1) | 0(1) | FSM_INIT |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 252. EMB_FUNC_INIT_B register description**

| | |
|---|---|
| MLC_INIT | Machine Learning Core initialization request. Default value: 0 |
| FIFO_COMPR_INIT | FIFO compression feature initialization request. Default value: 0 |
| FSM_INIT | FSM initialization request. Default value: 0 |

## 11.46 MLC0_SRC (70h)

Machine Learning Core source register (r)

**Table 253. MLC0_SRC register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MLC0_SRC_7 | MLC0_SRC_6 | MLC0_SRC_5 | MLC0_SRC_4 | MLC0_SRC_3 | MLC0_SRC_2 | MLC0_SRC_1 | MLC0_SRC_0 |

**Table 254. MLC0_SRC register description**

| | |
|---|---|
| MLC0_SRC_[7:0] | Output value of MLC0 decision tree |

chanhvinh.com



## 11.47 MLC1_SRC (71h)

Machine Learning Core source register (r)

**Table 255. MLC1_SRC register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MLC1_SRC_7 | MLC1_SRC_6 | MLC1_SRC_5 | MLC1_SRC_4 | MLC1_SRC_3 | MLCS1_SRC_2 | MLC1_SRC_1 | MLC1_SRC_0 |

**Table 256. MLC1_SRC register description**

| | |
|---|---|
| MLC1_SRC_[7:0] | Output value of MLC1 decision tree |

## 11.48 MLC2_SRC (72h)

Machine Learning Core source register (r)

**Table 257. MLC2_SRC register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MLC2_SRC_7 | MLC2_SRC_6 | MLC2_SRC_5 | MLC2_SRC_4 | MLC2_SRC_3 | MLC2_SRC_2 | MLC2_SRC_1 | MLC2_SRC_0 |

**Table 258. MLC2_SRC register description**

| | |
|---|---|
| MLC2_SRC_[7:0] | Output value of MLC2 decision tree |

## 11.49 MLC3_SRC (73h)

Machine Learning Core source register (r)

**Table 259. MLC3_SRC register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MLC3_SRC_7 | MLC3_SRC_6 | MLC3_SRC_5 | MLC3_SRC_4 | MLC3_SRC_3 | MLC3_SRC_2 | MLC3_SRC_1 | MLC3_SRC_0 |

**Table 260. MLC3_SRC register description**

| | |
|---|---|
| MLC3_SRC_[7:0] | Output value of MLC3 decision tree |

chanhvinh.com



## 11.50 MLC4_SRC (74h)

Machine Learning Core source register (r)

**Table 261. MLC4_SRC register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MLC4_SRC_7 | MLC4_SRC_6 | MLC4_SRC_5 | MLC4_SRC_4 | MLC4_SRC_3 | MLC4_SRC_2 | MLC4_SRC_1 | MLC4_SRC_0 |

**Table 262. MLC4_SRC register description**

| | |
|---|---|
| MLC4_SRC_[7:0] | Output value of MLC4 decision tree |

## 11.51 MLC5_SRC (75h)

Machine Learning Core source register (r)

**Table 263. MLC5_SRC register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MLC5_SRC_7 | MLC5_SRC_6 | MLC5_SRC_5 | MLC5_SRC_4 | MLC5_SRC_3 | MLC5_SRC_2 | MLC5_SRC_1 | MLC5_SRC_0 |

**Table 264. MLC5_SRC register description**

| | |
|---|---|
| MLC5_SRC_[7:0] | Output value of MLC5 decision tree |

## 11.52 MLC6_SRC (76h)

Machine Learning Core source register (r)

**Table 265. MLC6_SRC register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MLC6_SRC_7 | MLC6_SRC_6 | MLC6_SRC_5 | MLC6_SRC_4 | MLC6_SRC_3 | MLC6_SRC_2 | MLC6_SRC_1 | MLC6_SRC_0 |

**Table 266. MLC6_SRC register description**

| | |
|---|---|
| MLC6_SRC_[7:0] | Output value of MLC6 decision tree |

## 11.53 MLC7_SRC (77h)

Machine Learning Core source register (r)

**Table 267. MLC7_SRC register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MLC7_SRC_7 | MLC7_SRC_6 | MLC7_SRC_5 | MLC7_SRC_4 | MLC7_SRC_3 | MLC7_SRC_2 | MLC7_SRC_1 | MLC7_SRC_0 |

**Table 268. MLC7_SRC register description**

| | |
|---|---|
| MLC7_SRC_[7:0] | Output value of MLC7 decision tree |

chanhvinh.com



# 12 Embedded advanced features pages

The table given below provides a list of the registers for the embedded advanced features page 0. These registers are accessible when PAGE_SEL[3:0] are set to 0000 in PAGE_SEL (02h).

**Table 269. Register address map - embedded advanced features page 0**

| Name | Type | Register address | | Default | Comment |
|---|---|---|---|---|---|
| | | Hex | Binary | | |
| MAG_SENSITIVITY_L | r/w | BA | 10111010 | 00100100 | |
| MAG_SENSITIVITY_H | r/w | BB | 10111011 | 00010110 | |
| MAG_OFFX_L | r/w | C0 | 11000000 | 00000000 | |
| MAG_OFFX_H | r/w | C1 | 11000001 | 00000000 | |
| MAG_OFFY_L | r/w | C2 | 11000010 | 00000000 | |
| MAG_OFFY_H | r/w | C3 | 11000011 | 00000000 | |
| MAG_OFFZ_L | r/w | C4 | 11000100 | 00000000 | |
| MAG_OFFZ_H | r/w | C5 | 11000101 | 00000000 | |
| MAG_SI_XX_L | r/w | C6 | 11000110 | 00000000 | |
| MAG_SI_XX_H | r/w | C7 | 11000111 | 00111100 | |
| MAG_SI_XY_L | r/w | C8 | 11001000 | 00000000 | |
| MAG_SI_XY_H | r/w | C9 | 11001001 | 00000000 | |
| MAG_SI_XZ_L | r/w | CA | 11001010 | 00000000 | |
| MAG_SI_XZ_H | r/w | CB | 11001011 | 00000000 | |
| MAG_SI_YY_L | r/w | CC | 11001100 | 00000000 | |
| MAG_SI_YY_H | r/w | CD | 11001101 | 00111100 | |
| MAG_SI_YZ_L | r/w | CE | 11001110 | 00000000 | |
| MAG_SI_YZ_H | r/w | CF | 11001111 | 00000000 | |
| MAG_SI_ZZ_L | r/w | D0 | 11010000 | 00000000 | |
| MAG_SI_ZZ_H | r/w | D1 | 11010001 | 00111100 | |
| MAG_CFG_A | r/w | D4 | 11010100 | 00000101 | |
| MAG_CFG_B | r/w | D5 | 11010101 | 00000010 | |

chanhvinh.com



The following table provides a list of the registers for the embedded advanced features page 1. These registers are accessible when PAGE_SEL[3:0] are set to 0001 in PAGE_SEL (02h).

**Table 270. Register address map - embedded advanced features page 1**

| Name | Type | Register address | | Default | Comment |
|---|---|---|---|---|---|
| | | Hex | Binary | | |
| FSM_LC_TIMEOUT_L | r/w | 7A | 01111010 | 00000000 | |
| FSM_LC_TIMEOUT_H | r/w | 7B | 01111011 | 00000000 | |
| FSM_PROGRAMS | r/w | 7C | 01111100 | 00000000 | |
| FSM_START_ADD_L | r/w | 7E | 01111110 | 00000000 | |
| FSM_START_ADD_H | r/w | 7F | 01111111 | 00000000 | |
| PEDO_CMD_REG | r/w | 83 | 10000011 | 00000000 | |
| PEDO_DEB_STEPS_CONF | r/w | 84 | 10000100 | 00001010 | |
| PEDO_SC_DELTAT_L | r/w | D0 | 11010000 | 00000000 | |
| PEDO_SC_DELTAT_H | r/w | D1 | 11010001 | 00000000 | |
| MLC_MAG_SENSITIVITY_L | r/w | E8 | 11101000 | 0000000 | |
| MLC_MAG_SENSITIVITY_H | r/w | E9 | 11101001 | 00111100 | |

Registers marked as Reserved must not be changed. Writing to those registers may cause permanent damage to the device.

The content of the registers that are loaded at boot should not be changed. They contain the factory calibration values. Their content is automatically restored when the device is powered up.

**Write procedure example:**

Example: write value 06h register at address 84h (PEDO_DEB_STEPS_CONF) in Page 1

1. Write bit FUNC_CFG_EN = 1 in FUNC_CFG_ACCESS (01h) // Enable access to embedded functions registers
2. Write bit PAGE_WRITE = 1 in PAGE_RW (17h) register // Select write operation mode
3. Write 0001 in PAGE_SEL[3:0] field of register PAGE_SEL (02h) // Select page 1
4. Write 84h in PAGE_ADDR register (08h) // Set address
5. Write 06h in PAGE_DATA register (09h) // Set value to be written
6. Write bit PAGE_WRITE = 0 in PAGE_RW (17h) register // Write operation disabled
7. Write bit FUNC_CFG_EN = 0 in FUNC_CFG_ACCESS (01h) // Disable access to embedded functions registers

**Read procedure example:**

Example: read value of register at address 84h (PEDO_DEB_STEPS_CONF) in Page 1

1. Write bit FUNC_CFG_EN = 1 in FUNC_CFG_ACCESS (01h) // Enable access to embedded functions registers
2. Write bit PAGE_READ = 1 in PAGE_RW (17h) register // Select read operation mode
3. Write 0001 in PAGE_SEL[3:0] field of register PAGE_SEL (02h) // Select page 1
4. Write 84h in PAGE_ADDR register (08h) // Set address
5. Read value of PAGE_DATA register (09h) // Get register value
6. Write bit PAGE_READ = 0 in PAGE_RW (17h) register // Read operation disabled
7. Write bit FUNC_CFG_EN = 0 in FUNC_CFG_ACCESS (01h) // Disable access to embedded functions registers

*Note: Steps 1 and 2 of both procedures are intended to be performed at the beginning of the procedure. Steps 6 and 7 of both procedures are intended to be performed at the end of the procedure. If the procedure involves multiple operations, only steps 3, 4 and 5 must be repeated for each operation. If, in particular, the multiple operations involve consecutive registers, only step 5 can be performed.*

chanhvinh.com



# 13 Embedded advanced features register description

## 13.1 Page 0 - Embedded advanced features registers

### 13.1.1 MAG_SENSITIVITY_L (BAh) and MAG_SENSITIVITY_H (BBh)

External magnetometer sensitivity value register for the Finite State Machine (r/w)

This register corresponds to the LSB-to-gauss conversion value of the external magnetometer sensor. The register value is expressed as half-precision floating-point format: SEEEEEFFFFFFFFFF (S: 1 sign bit; E: 5 exponent bits; F: 10 fraction bits).

Default value of MAG_SENS[15:0] is 0x1624, corresponding to 0.0015 gauss/LSB.

**Table 271. MAG_SENSITIVITY_L register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_ SENS_7 | MAG_ SENS_6 | MAG_ SENS_5 | MAG_ SENS_4 | MAG_ SENS_3 | MAG_ SENS_2 | MAG_ SENS_1 | MAG_ SENS_0 |

**Table 272. MAG_SENSITIVITY_L register description**

| | |
|---|---|
| MAG_SENS_[7:0] | External magnetometer sensitivity (LSbyte). Default value: 00100100 |

**Table 273. MAG_SENSITIVITY_H register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_ SENS_15 | MAG_ SENS_14 | MAG_ SENS_13 | MAG_ SENS_12 | MAG_ SENS_11 | MAG_ SENS_10 | MAG_ SENS_9 | MAG_ SENS_8 |

**Table 274. MAG_SENSITIVITY_H register description**

| | |
|---|---|
| MAG_SENS_[15:8] | External magnetometer sensitivity (MSbyte). Default value: 00010110 |

chanhvinh.com



### 13.1.2 MAG_OFFX_L (C0h) and MAG_OFFX_H (C1h)

Offset for X-axis hard-iron compensation register (r/w)

The value is expressed as half-precision floating-point format: SEEEEEFFFFFFFFFF (S: 1 sign bit; E: 5 exponent bits; F: 10 fraction bits).

**Table 275. MAG_OFFX_L register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_OFFX_7 | MAG_OFFX_6 | MAG_OFFX_5 | MAG_OFFX_4 | MAG_OFFX_3 | MAG_OFFX_2 | MAG_OFFX_1 | MAG_OFFX_0 |

**Table 276. MAG_OFFX_L register description**

| | |
|---|---|
| MAG_OFFX_[7:0] | Offset for X-axis hard-iron compensation (LSbyte). Default value: 00000000 |

**Table 277. MAG_OFFX_H register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_OFFX_15 | MAG_OFFX_14 | MAG_OFFX_13 | MAG_OFFX_12 | MAG_OFFX_11 | MAG_OFFX_10 | MAG_OFFX_9 | MAG_OFFX_8 |

**Table 278. MAG_OFFX_H register description**

| | |
|---|---|
| MAG_OFFX_[15:8] | Offset for X-axis hard-iron compensation (MSbyte). Default value: 00000000 |

### 13.1.3 MAG_OFFY_L (C2h) and MAG_OFFY_H (C3h)

Offset for Y-axis hard-iron compensation register (r/w)

The value is expressed as half-precision floating-point format: SEEEEEFFFFFFFFFF (S: 1 sign bit; E: 5 exponent bits; F: 10 fraction bits).

**Table 279. MAG_OFFY_L register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_OFFY_7 | MAG_OFFY_6 | MAG_OFFY_5 | MAG_OFFY_4 | MAG_OFFY_3 | MAG_OFFY_2 | MAG_OFFY_1 | MAG_OFFY_0 |

**Table 280. MAG_OFFY_L register description**

| | |
|---|---|
| MAG_OFFY_[7:0] | Offset for Y-axis hard-iron compensation (LSbyte). Default value: 00000000 |

**Table 281. MAG_OFFY_H register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_OFFY_15 | MAG_OFFY_14 | MAG_OFFY_13 | MAG_OFFY_12 | MAG_OFFY_11 | MAG_OFFY_10 | MAG_OFFY_9 | MAG_OFFY_8 |

**Table 282. MAG_OFFY_H register description**

| | |
|---|---|
| MAG_OFFY_[15:8] | Offset for Y-axis hard-iron compensation (MSbyte). Default value: 00000000 |

chanhvinh.com



### 13.1.4 MAG_OFFZ_L (C4h) and MAG_OFFZ_H (C5h)

Offset for Z-axis hard-iron compensation register (r/w)

The value is expressed as half-precision floating-point format: SEEEEEFFFFFFFFFF (S: 1 sign bit; E: 5 exponent bits; F: 10 fraction bits).

**Table 283. MAG_OFFZ_L register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_OFFZ_7 | MAG_OFFZ_6 | MAG_OFFZ_5 | MAG_OFFZ_4 | MAG_OFFZ_3 | MAG_OFFZ_2 | MAG_OFFZ_1 | MAG_OFFZ_0 |

**Table 284. MAG_OFFZ_L register description**

| | |
|---|---|
| MAG_OFFZ_[7:0] | Offset for Z-axis hard-iron compensation (LSbyte). Default value: 00000000 |

**Table 285. MAG_OFFZ_H register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_OFFZ_15 | MAG_OFFZ_14 | MAG_OFFZ_13 | MAG_OFFZ_12 | MAG_OFFZ_11 | MAG_OFFZ_10 | MAG_OFFZ_9 | MAG_OFFZ_8 |

**Table 286. MAG_OFFZ_H register description**

| | |
|---|---|
| MAG_OFFZ_[15:8] | Offset for Z-axis hard-iron compensation (MSbyte). Default value: 00000000 |

### 13.1.5 MAG_SI_XX_L (C6h) and MAG_SI_XX_H (C7h)

Soft-iron (3x3 symmetric) matrix correction register (r/w)

The value is expressed as half-precision floating-point format: SEEEEEFFFFFFFFFF (S: 1 sign bit; E: 5 exponent bits; F: 10 fraction bits).

**Table 287. MAG_SI_XX_L register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_SI_XX_7 | MAG_SI_XX_6 | MAG_SI_XX_5 | MAG_SI_XX_4 | MAG_SI_XX_3 | MAG_SI_XX_2 | MAG_SI_XX_1 | MAG_SI_XX_0 |

**Table 288. MAG_SI_XX_L register description**

| | |
|---|---|
| MAG_SI_XX_[7:0] | Soft-iron correction row1 col1 coefficient (LSbyte). Default value: 00000000 |

**Table 289. MAG_SI_XX_H register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_SI_XX_15 | MAG_SI_XX_14 | MAG_SI_XX_13 | MAG_SI_XX_12 | MAG_SI_XX_11 | MAG_SI_XX_10 | MAG_SI_XX_9 | MAG_SI_XX_8 |

**Table 290. MAG_SI_XX_H register description**

| | |
|---|---|
| MAG_SI_XX_[15:8] | Soft-iron correction row1 col1 coefficient (MSbyte). Default value: 00111100 |

chanhvinh.com



### 13.1.6 MAG_SI_XY_L (C8h) and MAG_SI_XY_H (C9h)

Soft-iron (3x3 symmetric) matrix correction register (r/w)

The value is expressed as half-precision floating-point format: SEEEEEFFFFFFFFFF (S: 1 sign bit; E: 5 exponent bits; F: 10 fraction bits).

**Table 291. MAG_SI_XY_L register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_SI_XY_7 | MAG_SI_XY_6 | MAG_SI_XY_5 | MAG_SI_XY_4 | MAG_SI_XY_3 | MAG_SI_XY_2 | MAG_SI_XY_1 | MAG_SI_XY_0 |

**Table 292. MAG_SI_XY_L register description**

| | |
|---|---|
| MAG_SI_XY_[7:0] | Soft-iron correction row1 col2 (and row2 col1) coefficient (LSbyte). Default value: 00000000 |

**Table 293. MAG_SI_XY_H register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_SI_XY_15 | MAG_SI_XY_14 | MAG_SI_XY_13 | MAG_SI_XY_12 | MAG_SI_XY_11 | MAG_SI_XY_10 | MAG_SI_XY_9 | MAG_SI_XY_8 |

**Table 294. MAG_SI_XY_H register description**

| | |
|---|---|
| MAG_SI_XY_[15:8] | Soft-iron correction row1 col2 (and row2 col1) coefficient (MSbyte). Default value: 00000000 |

### 13.1.7 MAG_SI_XZ_L (CAh) and MAG_SI_XZ_H (CBh)

Soft-iron (3x3 symmetric) matrix correction register (r/w)

The value is expressed as half-precision floating-point format: SEEEEEFFFFFFFFFF (S: 1 sign bit; E: 5 exponent bits; F: 10 fraction bits).

**Table 295. MAG_SI_XZ_L register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_SI_XZ_7 | MAG_SI_XZ_6 | MAG_SI_XZ_5 | MAG_SI_XZ_4 | MAG_SI_XZ_3 | MAG_SI_XZ_2 | MAG_SI_XZ_1 | MAG_SI_XZ_0 |

**Table 296. MAG_SI_XZ_L register description**

| | |
|---|---|
| MAG_SI_XZ_[7:0] | Soft-iron correction row1 col3 (and row3 col1) coefficient (LSbyte). Default value: 00000000 |

**Table 297. MAG_SI_XZ_H register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_SI_XZ_15 | MAG_SI_XZ_14 | MAG_SI_XZ_13 | MAG_SI_XZ_12 | MAG_SI_XZ_11 | MAG_SI_XZ_10 | MAG_SI_XZ_9 | MAG_SI_XZ_8 |

**Table 298. MAG_SI_XZ_H register description**

| | |
|---|---|
| MAG_SI_XZ_[15:8] | Soft-iron correction row1 col3 (and row3 col1) coefficient (MSbyte). Default value: 00000000 |

chanhvinh.com



### 13.1.8 MAG_SI_YY_L (CCh) and MAG_SI_YY_H (CDh)

Soft-iron (3x3 symmetric) matrix correction register (r/w)

The value is expressed as half-precision floating-point format: SEEEEEFFFFFFFFFF (S: 1 sign bit; E: 5 exponent bits; F: 10 fraction bits).

**Table 299. MAG_SI_YY_L register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_SI_YY_7 | MAG_SI_YY_6 | MAG_SI_YY_5 | MAG_SI_YY_4 | MAG_SI_YY_3 | MAG_SI_YY_2 | MAG_SI_YY_1 | MAG_SI_YY_0 |

**Table 300. MAG_SI_YY_L register description**

| | |
|---|---|
| MAG_SI_YY_[7:0] | Soft-iron correction row2 col2 coefficient (LSbyte). Default value: 00000000 |

**Table 301. MAG_SI_YY_H register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_SI_YY_15 | MAG_SI_YY_14 | MAG_SI_YY_13 | MAG_SI_YY_12 | MAG_SI_YY_11 | MAG_SI_YY_10 | MAG_SI_YY_9 | MAG_SI_YY_8 |

**Table 302. MAG_SI_YY_H register description**

| | |
|---|---|
| MAG_SI_YY_[15:8] | Soft-iron correction row2 col2 coefficient (MSbyte). Default value: 00111100 |

### 13.1.9 MAG_SI_YZ_L (CEh) and MAG_SI_YZ_H (CFh)

Soft-iron (3x3 symmetric) matrix correction register (r/w)

The value is expressed as half-precision floating-point format: SEEEEEFFFFFFFFFF (S: 1 sign bit; E: 5 exponent bits; F: 10 fraction bits).

**Table 303. MAG_SI_YZ_L register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_SI_YZ_7 | MAG_SI_YZ_6 | MAG_SI_YZ_5 | MAG_SI_YZ_4 | MAG_SI_YZ_3 | MAG_SI_YZ_2 | MAG_SI_YZ_1 | MAG_SI_YZ_0 |

**Table 304. MAG_SI_YZ_L register description**

| | |
|---|---|
| MAG_SI_YZ_[7:0] | Soft-iron correction row2 col3 (and row3 col2) coefficient (LSbyte). Default value: 00000000 |

**Table 305. MAG_SI_YZ_H register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_SI_YZ_15 | MAG_SI_YZ_14 | MAG_SI_YZ_13 | MAG_SI_YZ_12 | MAG_SI_YZ_11 | MAG_SI_YZ_10 | MAG_SI_YZ_9 | MAG_SI_YZ_8 |

**Table 306. MAG_SI_YZ_H register description**

| | |
|---|---|
| MAG_SI_YZ_[15:8] | Soft-iron correction row2 col3 (and row3 col2) coefficient (MSbyte). Default value: 00000000 |

chanhvinh.com



### 13.1.10 MAG_SI_ZZ_L (D0h) and MAG_SI_ZZ_H (D1h)

Soft-iron (3x3 symmetric) matrix correction register (r/w)

The value is expressed as half-precision floating-point format: SEEEEEFFFFFFFFFF (S: 1 sign bit; E: 5 exponent bits; F: 10 fraction bits).

**Table 307. MAG_SI_ZZ_L register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_SI_ZZ_7 | MAG_SI_ZZ_6 | MAG_SI_ZZ_5 | MAG_SI_ZZ_4 | MAG_SI_ZZ_3 | MAG_SI_ZZ_2 | MAG_SI_ZZ_1 | MAG_SI_ZZ_0 |

**Table 308. MAG_SI_ZZ_L register description**

| | |
|---|---|
| MAG_SI_ZZ_[7:0] | Soft-iron correction row3 col3 coefficient (LSbyte). Default value: 00000000 |

**Table 309. MAG_SI_ZZ_H register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MAG_SI_ZZ_15 | MAG_SI_ZZ_14 | MAG_SI_ZZ_13 | MAG_SI_ZZ_12 | MAG_SI_ZZ_11 | MAG_SI_ZZ_10 | MAG_SI_ZZ_9 | MAG_SI_ZZ_8 |

**Table 310. MAG_SI_ZZ_H register description**

| | |
|---|---|
| MAG_SI_ZZ_[15:8] | Soft-iron correction row3 col3 coefficient (MSbyte). Default value: 00111100 |

chanhvinh.com



### 13.1.11 MAG_CFG_A (D4h)

External magnetometer coordinates (Y and Z axes) rotation register (r/w)

**Table 311. MAG_CFG_A register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | MAG_Y_ AXIS2 | MAG_Y_ AXIS1 | MAG_Y_ AXIS0 | 0(1) | MAG_Z_ AXIS2 | MAG_Z_ AXIS1 | MAG_Z_ AXIS0 |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 312. MAG_CFG_A description**

| | |
|---|---|
| MAG_Y_AXIS[2:0] | Magnetometer Y-axis coordinates rotation (to be aligned to accelerometer/gyroscope axes orientation)<br>(000: Y = Y; (default)<br>001: Y = -Y;<br>010: Y = X;<br>011: Y = -X;<br>100: Y = -Z;<br>101: Y = Z;<br>Others: Y = Y) |
| MAG_Z_AXIS[2:0] | Magnetometer Z-axis coordinates rotation (to be aligned to accelerometer/gyroscope axes orientation)<br>(000: Z = Y;<br>001: Z = -Y;<br>010: Z = X;<br>011: Z = -X;<br>100: Z = -Z;<br>101: Z = Z; (default)<br>Others: Z = Y) |

### 13.1.12 MAG_CFG_B (D5h)

External magnetometer coordinates (X-axis) rotation register (r/w).

**Table 313. MAG_CFG_B register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | 0(1) | 0(1) | 0(1) | 0(1) | MAG_X_ AXIS2 | MAG_X_ AXIS1 | MAG_X_ AXIS0 |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 314. MAG_CFG_B description**

| | |
|---|---|
| MAG_X_AXIS[2:0] | Magnetometer X-axis coordinates rotation (to be aligned to accelerometer/gyroscope axes orientation)<br>(000: X = Y;<br>001: X = -Y;<br>010: X = X; (default)<br>011: X = -X;<br>100: X = -Z;<br>101: X = Z;<br>Others: X = Y) |

chanhvinh.com



## 13.2 Page 1 - Embedded advanced features registers

### 13.2.1 FSM_LC_TIMEOUT_L (7Ah) and FSM_LC_TIMEOUT_H (7Bh)

FSM long counter timeout register (r/w)

The long counter timeout value is an unsigned integer value (16-bit format). When the long counter value reaches this value, the FSM generates an interrupt.

**Table 315. FSM_LC_TIMEOUT_L register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| FSM_LC_ TIMEOUT7 | FSM_LC_ TIMEOUT6 | FSM_LC_ TIMEOUT5 | FSM_LC_ TIMEOUT4 | FSM_LC_ TIMEOUT3 | FSM_LC_ TIMEOUT2 | FSM_LC_ TIMEOUT1 | FSM_LC_ TIMEOUT0 |

**Table 316. FSM_LC_TIMEOUT_L register description**

| | |
|---|---|
| FSM_LC_TIMEOUT[7:0] | FSM long counter timeout value (LSbyte). Default value: 00000000 |

**Table 317. FSM_LC_TIMEOUT_H register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| FSM_LC_ TIMEOUT15 | FSM_LC_ TIMEOUT14 | FSM_LC_ TIMEOUT13 | FSM_LC_ TIMEOUT12 | FSM_LC_ TIMEOUT11 | FSM_LC_ TIMEOUT10 | FSM_LC_ TIMEOUT9 | FSM_LC_ TIMEOUT8 |

**Table 318. FSM_LC_TIMEOUT_H register description**

| | |
|---|---|
| FSM_LC_TIMEOUT[15:8] | FSM long counter timeout value (MSbyte). Default value: 00000000 |

### 13.2.2 FSM_PROGRAMS (7Ch)

FSM number of programs register (r/w)

**Table 319. FSM_PROGRAMS register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| FSM_N_ PROG7 | FSM_N_ PROG6 | FSM_N_ PROG5 | FSM_N_ PROG4 | FSM_N_ PROG3 | FSM_N_ PROG2 | FSM_N_ PROG1 | FSM_N_ PROG0 |

**Table 320. FSM_PROGRAMS register description**

| | |
|---|---|
| FSM_N_PROG[7:0] | Number of FSM programs; must be less than or equal to 16. Default value: 00000000 |

chanhvinh.com



### 13.2.3 FSM_START_ADD_L (7Eh) and FSM_START_ADD_H (7Fh)

FSM start address register (r/w). First available address is 0x033C.

**Table 321. FSM_START_ADD_L register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| FSM_START7 | FSM_START6 | FSM_START5 | FSM_START4 | FSM_START3 | FSM_START2 | FSM_START1 | FSM_START0 |

**Table 322. FSM_START_ADD_L register description**

| | |
|---|---|
| FSM_START[7:0] | FSM start address value (LSbyte). Default value: 00000000 |

**Table 323. FSM_START_ADD_H register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| FSM_START15 | FSM_START14 | FSM_START13 | FSM_START12 | FSM_START11 | FSM_START10 | FSM_START9 | FSM_START8 |

**Table 324. FSM_START_ADD_H register description**

| | |
|---|---|
| FSM_START[15:8] | FSM start address value (MSbyte). Default value: 00000000 |

### 13.2.4 PEDO_CMD_REG (83h)

Pedometer configuration register (r/w)

**Table 325. PEDO_CMD_REG register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | 0(1) | 0(1) | 0(1) | CARRY_COUNT_EN | FP_REJECTION_EN | 0(1) | AD_DET_EN |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 326. PEDO_CMD_REG register description**

| | |
|---|---|
| CARRY_COUNT_EN | Set when user wants to generate interrupt only on count overflow event. |
| FP_REJECTION_EN(1) | Enables the false-positive rejection feature. |
| AD_DET_EN(2) | Enables the advanced detection feature. |

1. *This bit is effective if the MLC_EN bit of EMB_FUNC_EN_B (05h) is set to 1.*
2. *This bit is effective if both the FP_REJECTION_EN bit in PEDO_CMD_REG (83h) register and the MLC_EN bit of EMB_FUNC_EN_B (05h) are set to 1.*

chanhvinh.com



### 13.2.5 PEDO_DEB_STEPS_CONF (84h)

Pedometer debounce configuration register (r/w)

Table 327. **PEDO_DEB_STEPS_CONF register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| DEB_STEP7 | DEB_STEP6 | DEB_STEP5 | DEB_STEP4 | DEB_STEP3 | DEB_STEP2 | DEB_STEP1 | DEB_STEP0 |

Table 328. **PEDO_DEB_STEPS_CONF register description**

| | |
|---|---|
| DEB_STEP[7:0] | Debounce threshold. Minimum number of steps to increment the step counter (debounce).<br>Default value: 00001010 |

### 13.2.6 PEDO_SC_DELTAT_L (D0h) and PEDO_SC_DELTAT_H (D1h)

Time period register for step detection on delta time (r/w)

Table 329. **PEDO_SC_DELTAT_L register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| PD_SC_7 | PD_SC_6 | PD_SC_5 | PD_SC_4 | PD_SC_3 | PD_SC_2 | PD_SC_1 | PD_SC_0 |

Table 330. **PEDO_SC_DELTAT_H register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| PD_SC_15 | PD_SC_14 | PD_SC_13 | PD_SC_12 | PD_SC_11 | PD_SC_10 | PD_SC_9 | PD_SC_8 |

Table 331. **PEDO_SC_DELTAT_H/L register description**

| | |
|---|---|
| PD_SC_[15:0] | Time period value (1LSB = 6.4 ms) |

### 13.2.7 MLC_MAG_SENSITIVITY_L (E8h) and MLC_MAG_SENSITIVITY_H (E9h)

External magnetometer sensitivity value register for the Machine Learning Core (r/w).

This register corresponds to the LSB-to-gauss conversion value of the external magnetometer sensor. The register value is expressed as half-precision floating-point format: SEEEEEFFFFFFFFFF (S: 1 sign bit; E: 5 exponent bits; F: 10 fraction bits). Default value of MLC_MAG_S_[15:0] is 0x3C00, corresponding to 1 gauss/LSB.

Table 332. **MLC_MAG_SENSITIVITY_L register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MLC_MAG_S_7 | MLC_MAG_S_6 | MLC_MAG_S_5 | MLC_MAG_S_4 | MLC_MAG_S_3 | MLC_MAG_S_2 | MLC_MAG_S_1 | MLC_MAG_S_0 |

Table 333. **MLC_ MAG_SENSITIVITY_L register description**

| | |
|---|---|
| MLC_MAG_S_[7:0] | External magnetometer sensitivity (LSbyte). Default value: 00000000 |

Table 334. **MLC_MAG_SENSITIVITY_H register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MLC_MAG_S_15 | MLC_MAG_S_14 | MLC_MAG_S_13 | MLC_MAG_S_12 | MLC_MAG_S_11 | MLC_MAG_S_10 | MLC_MAG_S_9 | MLC_MAG_S_8 |

Table 335. **MLC_ MAG_SENSITIVITY_H register description**

| | |
|---|---|
| MLC_MAG_S_[15:8] | External magnetometer sensitivity (MSbyte). Default value: 00000000 |

chanhvinh.com



# 14 Sensor hub register mapping

The table given below provides a list of the registers for the sensor hub functions available in the device and the corresponding addresses. The sensor hub registers are accessible when bit SHUB_REG_ACCESS is set to '1' in FUNC_CFG_ACCESS (01h).

**Table 336. Register address map - sensor hub registers**

| Name | Type | Register address | | Default | Comment |
|---|---|---|---|---|---|
| | | Hex | Binary | | |
| SENSOR_HUB_1 | r | 02 | 00000010 | output | |
| SENSOR_HUB_2 | r | 03 | 00000011 | output | |
| SENSOR_HUB_3 | r | 04 | 00000100 | output | |
| SENSOR_HUB_4 | r | 05 | 00000101 | output | |
| SENSOR_HUB_5 | r | 06 | 00000110 | output | |
| SENSOR_HUB_6 | r | 07 | 00000111 | output | |
| SENSOR_HUB_7 | r | 08 | 00001000 | output | |
| SENSOR_HUB_8 | r | 09 | 00001001 | output | |
| SENSOR_HUB_9 | r | 0A | 00001010 | output | |
| SENSOR_HUB_10 | r | 0B | 00001011 | output | |
| SENSOR_HUB_11 | r | 0C | 00001100 | output | |
| SENSOR_HUB_12 | r | 0D | 00001101 | output | |
| SENSOR_HUB_13 | r | 0E | 00001110 | output | |
| SENSOR_HUB_14 | r | 0F | 00001111 | output | |
| SENSOR_HUB_15 | r | 10 | 00010000 | output | |
| SENSOR_HUB_16 | r | 11 | 00010001 | output | |
| SENSOR_HUB_17 | r | 12 | 00010010 | output | |
| SENSOR_HUB_18 | r | 13 | 00010011 | output | |
| MASTER_CONFIG | rw | 14 | 00010100 | 00000000 | |
| SLV0_ADD | rw | 15 | 00010101 | 00000000 | |
| SLV0_SUBADD | rw | 16 | 00010110 | 00000000 | |
| SLV0_CONFIG | rw | 17 | 0001 0111 | 00000000 | |
| SLV1_ADD | rw | 18 | 00011000 | 00000000 | |
| SLV1_SUBADD | rw | 19 | 00011001 | 00000000 | |
| SLV1_CONFIG | rw | 1A | 00011010 | 00000000 | |
| SLV2_ADD | rw | 1B | 00011011 | 00000000 | |
| SLV2_SUBADD | rw | 1C | 00011100 | 00000000 | |
| SLV2_CONFIG | rw | 1D | 00011101 | 00000000 | |
| SLV3_ADD | rw | 1E | 00011110 | 00000000 | |
| SLV3_SUBADD | rw | 1F | 00011111 | 00000000 | |
| SLV3_CONFIG | rw | 20 | 00100000 | 00000000 | |
| DATAWRITE_SLV0 | rw | 21 | 00100001 | 00000000 | |
| STATUS_MASTER | r | 22 | 00100010 | output | |

Registers marked as Reserved must not be changed. Writing to those registers may cause permanent damage to the device.

The content of the registers that are loaded at boot should not be changed. They contain the factory calibration values. Their content is automatically restored when the device is powered up.

chanhvinh.com



# 15 Sensor hub register description

## 15.1 SENSOR_HUB_1 (02h)

Sensor hub output register (r)

First byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 337. SENSOR_HUB_1 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub1_7 | Sensor Hub1_6 | Sensor Hub1_5 | Sensor Hub1_4 | Sensor Hub1_3 | Sensor Hub1_2 | Sensor Hub1_1 | Sensor Hub1_0 |

**Table 338. SENSOR_HUB_1 register description**

| | |
|---|---|
| SensorHub1_[7:0] | First byte associated to external sensors |

## 15.2 SENSOR_HUB_2 (03h)

Sensor hub output register (r)

Second byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 339. SENSOR_HUB_2 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub2_7 | Sensor Hub2_6 | Sensor Hub2_5 | Sensor Hub2_4 | Sensor Hub2_3 | Sensor Hub2_2 | Sensor Hub2_1 | Sensor Hub2_0 |

**Table 340. SENSOR_HUB_2 register description**

| | |
|---|---|
| SensorHub2_[7:0] | Second byte associated to external sensors |

## 15.3 SENSOR_HUB_3 (04h)

Sensor hub output register (r)

Third byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 341. SENSOR_HUB_3 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub3_7 | Sensor Hub3_6 | Sensor Hub3_5 | Sensor Hub3_4 | Sensor Hub3_3 | Sensor Hub3_2 | Sensor Hub3_1 | Sensor Hub3_0 |

**Table 342. SENSOR_HUB_3 register description**

| | |
|---|---|
| SensorHub3_[7:0] | Third byte associated to external sensors |

chanhvinh.com



## 15.4 SENSOR_HUB_4 (05h)

Sensor hub output register (r)

Fourth byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 343. SENSOR_HUB_4 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub4_7 | Sensor Hub4_6 | Sensor Hub4_5 | Sensor Hub4_4 | Sensor Hub4_3 | Sensor Hub4_2 | Sensor Hub4_1 | Sensor Hub4_0 |

**Table 344. SENSOR_HUB_4 register description**

| | |
|---|---|
| SensorHub4_[7:0] | Fourth byte associated to external sensors |

## 15.5 SENSOR_HUB_5 (06h)

Sensor hub output register (r)

Fifth byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 345. SENSOR_HUB_5 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub5_7 | Sensor Hub5_6 | Sensor Hub5_5 | Sensor Hub5_4 | Sensor Hub5_3 | Sensor Hub5_2 | Sensor Hub5_1 | Sensor Hub5_0 |

**Table 346. SENSOR_HUB_5 register description**

| | |
|---|---|
| SensorHub5_[7:0] | Fifth byte associated to external sensors |

## 15.6 SENSOR_HUB_6 (07h)

Sensor hub output register (r)

Sixth byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 347. SENSOR_HUB_6 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub6_7 | Sensor Hub6_6 | Sensor Hub6_5 | Sensor Hub6_4 | Sensor Hub6_3 | Sensor Hub6_2 | Sensor Hub6_1 | Sensor Hub6_0 |

**Table 348. SENSOR_HUB_6 register description**

| | |
|---|---|
| SensorHub6_[7:0] | Sixth byte associated to external sensors |

chanhvinh.com



## 15.7 SENSOR_HUB_7 (08h)

Sensor hub output register (r)

Seventh byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 349. SENSOR_HUB_7 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub7_7 | Sensor Hub7_6 | Sensor Hub7_5 | Sensor Hub7_4 | Sensor Hub7_3 | Sensor Hub7_2 | Sensor Hub7_1 | Sensor Hub7_0 |

**Table 350. SENSOR_HUB_7 register description**

| | |
|---|---|
| SensorHub7_[7:0] | Seventh byte associated to external sensors |

## 15.8 SENSOR_HUB_8 (09h)

Sensor hub output register (r)

Eighth byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 351. SENSOR_HUB_8 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub8_7 | Sensor Hub8_6 | Sensor Hub8_5 | Sensor Hub8_4 | Sensor Hub8_3 | Sensor Hub8_2 | Sensor Hub8_1 | Sensor Hub8_0 |

**Table 352. SENSOR_HUB_8 register description**

| | |
|---|---|
| SensorHub8_[7:0] | Eighth byte associated to external sensors |

## 15.9 SENSOR_HUB_9 (0Ah)

Sensor hub output register (r)

Ninth byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 353. SENSOR_HUB_9 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub9_7 | Sensor Hub9_6 | Sensor Hub9_5 | Sensor Hub9_4 | Sensor Hub9_3 | Sensor Hub9_2 | Sensor Hub9_1 | Sensor Hub9_0 |

**Table 354. SENSOR_HUB_9 register description**

| | |
|---|---|
| SensorHub9_[7:0] | Ninth byte associated to external sensors |

chanhvinh.com



## 15.10 SENSOR_HUB_10 (0Bh)

Sensor hub output register (r)

Tenth byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 355. SENSOR_HUB_10 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub10_7 | Sensor Hub10_6 | Sensor Hub10_5 | Sensor Hub10_4 | Sensor Hub10_3 | Sensor Hub10_2 | Sensor Hub10_1 | Sensor Hub10_0 |

**Table 356. SENSOR_HUB_10 register description**

| | |
|---|---|
| SensorHub_10[7:0] | Tenth byte associated to external sensors |

## 15.11 SENSOR_HUB_11 (0Ch)

Sensor hub output register (r)

Eleventh byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 357. SENSOR_HUB_11 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub11_7 | Sensor Hub11_6 | Sensor Hub11_5 | Sensor Hub11_4 | Sensor Hub11_3 | Sensor Hub11_2 | Sensor Hub11_1 | Sensor Hub11_0 |

**Table 358. SENSOR_HUB_11 register description**

| | |
|---|---|
| SensorHub11_[7:0] | Eleventh byte associated to external sensors |

## 15.12 SENSOR_HUB_12 (0Dh)

Sensor hub output register (r)

Twelfth byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 359. SENSOR_HUB_12 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub12_7 | Sensor Hub12_6 | Sensor Hub12_5 | Sensor Hub12_4 | Sensor Hub12_3 | Sensor Hub12_2 | Sensor Hub12_1 | Sensor Hub12_0 |

**Table 360. SENSOR_HUB_12 register description**

| | |
|---|---|
| SensorHub_12[7:0] | Twelfth byte associated to external sensors |

chanhvinh.com



## 15.13 SENSOR_HUB_13 (0Eh)

Sensor hub output register (r)

Thirteenth byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 361. SENSOR_HUB_13 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub13_7 | Sensor Hub13_6 | Sensor Hub13_5 | Sensor Hub13_4 | Sensor Hub13_3 | Sensor Hub13_2 | Sensor Hub13_1 | Sensor Hub13_0 |

**Table 362. SENSOR_HUB_13 register description**

| | |
|---|---|
| SensorHub13_[7:0] | Thirteenth byte associated to external sensors |

## 15.14 SENSOR_HUB_14 (0Fh)

Sensor hub output register (r)

Fourteenth byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 363. SENSOR_HUB_14 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub14_7 | Sensor Hub14_6 | Sensor Hub14_5 | Sensor Hub14_4 | Sensor Hub14_3 | Sensor Hub14_2 | Sensor Hub14_1 | Sensor Hub14_0 |

**Table 364. SENSOR_HUB_14 register description**

| | |
|---|---|
| SensorHub14_[7:0] | Fourteenth byte associated to external sensors |

## 15.15 SENSOR_HUB_15 (10h)

Sensor hub output register (r)

Fifteenth byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 365. SENSOR_HUB_15 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub15_7 | Sensor Hub15_6 | Sensor Hub15_5 | Sensor Hub15_4 | Sensor Hub15_3 | Sensor Hub15_2 | Sensor Hub15_1 | Sensor Hub15_0 |

**Table 366. SENSOR_HUB_15 register description**

| | |
|---|---|
| SensorHub15_[7:0] | Fifteenth byte associated to external sensors |

chanhvinh.com



## 15.16 SENSOR_HUB_16 (11h)

Sensor hub output register (r)

Sixteenth byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 367. SENSOR_HUB_16 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub16_7 | Sensor Hub16_6 | Sensor Hub16_5 | Sensor Hub16_4 | Sensor Hub16_3 | Sensor Hub16_2 | Sensor Hub16_1 | Sensor Hub16_0 |

**Table 368. SENSOR_HUB_16 register description**

| | |
|---|---|
| SensorHub16_[7:0] | Sixteenth byte associated to external sensors |

## 15.17 SENSOR_HUB_17 (12h)

Sensor hub output register (r)

Seventeenth byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 369. SENSOR_HUB_17 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub17_7 | Sensor Hub17_6 | Sensor Hub17_5 | Sensor Hub17_4 | Sensor Hub17_3 | Sensor Hub17_2 | Sensor Hub17_1 | Sensor Hub17_7 |

**Table 370. SENSOR_HUB_17 register description**

| | |
|---|---|
| SensorHub17_[7:0] | Seventeenth byte associated to external sensors |

## 15.18 SENSOR_HUB_18 (13h)

Sensor hub output register (r)

Eighteenth byte associated to external sensors. The content of the register is consistent with the SLAVEx_CONFIG number of read operation configurations (for external sensors from x = 0 to x = 3).

**Table 371. SENSOR_HUB_17 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Sensor Hub18_7 | Sensor Hub18_6 | Sensor Hub18_5 | Sensor Hub18_4 | Sensor Hub18_3 | Sensor Hub18_2 | Sensor Hub18_1 | Sensor Hub18_0 |

**Table 372. SENSOR_HUB_17 register description**

| | |
|---|---|
| SensorHub18_[7:0] | Eighteenth byte associated to external sensors |

chanhvinh.com



## 15.19 MASTER_CONFIG (14h)

Master configuration register (r/w)

**Table 373. MASTER_CONFIG register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| RST_MASTER_REGS | WRITE_ONCE | START_CONFIG | PASS_THROUGH_MODE | SHUB_PU_EN | MASTER_ON | AUX_SENS_ON1 | AUX_SENS_ON0 |

**Table 374. MASTER_CONFIG register description**

| | |
|---|---|
| RST_MASTER_REGS | Reset Master logic and output registers. Must be set to '1' and then set it to '0'. Default value: 0 |
| WRITE_ONCE | Slave 0 write operation is performed only at the first sensor hub cycle. Default value: 0<br>(0: write operation for each sensor hub cycle;<br>1: write operation only for the first sensor hub cycle) |
| START_CONFIG | Sensor hub trigger signal selection. Default value: 0<br>(0: sensor hub trigger signal is the accelerometer/gyro data-ready;<br>1: sensor hub trigger signal external from INT2 pin) |
| PASS_THROUGH_MODE | I²C interface pass-through. Default value: 0<br>(0: pass-through disabled;<br>1: pass-through enabled, main I²C line is short-circuited with the auxiliary line) |
| SHUB_PU_EN | Master I²C pull-up enable. Default value: 0<br>(0: internal pull-up on auxiliary I²C line disabled;<br>1: internal pull-up on auxiliary I²C line enabled) |
| MASTER_ON | Sensor hub I²C master enable. Default: 0<br>(0: master I²C of sensor hub disabled; 1: master I²C of sensor hub enabled) |
| AUX_SENS_ON[1:0] | Number of external sensors to be read by the sensor hub.<br>(00: one sensor (default);<br>01: two sensors;<br>10: three sensors;<br>11: four sensors) |

## 15.20 SLV0_ADD (15h)

I²C slave address of the first external sensor (Sensor 1) register (r/w).

**Table 375. SLV0_ADD register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| slave0_add6 | slave0_add5 | slave0_add4 | slave0_add3 | slave0_add2 | slave0_add1 | slave0_add0 | rw_0 |

**Table 376. SLV_ADD register description**

| | |
|---|---|
| slave0_add[6:0] | I²C slave address of Sensor1 that can be read by the sensor hub. Default value: 0000000 |
| rw_0 | Read/write operation on Sensor 1. Default value: 0<br>(0: write operation; 1: read operation) |

chanhvinh.com



## 15.21 SLV0_SUBADD (16h)

Address of register on the first external sensor (Sensor 1) register (r/w)

**Table 377. SLV0_SUBADD register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| slave0_ reg7 | slave0_ reg6 | slave0_ reg5 | slave0_ reg4 | slave0_ reg3 | slave0_ reg2 | slave0_ reg1 | slave0_ reg0 |

**Table 378. SLV0_SUBADD register description**

| | |
|---|---|
| slave0_reg[7:0] | Address of register on Sensor1 that has to be read/written according to the rw_0 bit value in SLV0_ADD (15h). Default value: 00000000 |

## 15.22 SLAVE0_CONFIG (17h)

First external sensor (Sensor1) configuration and sensor hub settings register (r/w)

**Table 379. SLAVE0_CONFIG register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| SHUB_ ODR_1 | SHUB_ ODR_0 | 0(1) | 0(1) | BATCH_EXT _SENS_0_EN | Slave0_ numop2 | Slave0_ numop1 | Slave0_ numop0 |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 380. SLAVE0_CONFIG register description**

| | |
|---|---|
| SHUB_ODR_[1:0] | Rate at which the master communicates. Default value: 00<br>(00: 104 Hz (or at the maximum ODR between the accelerometer and gyro if it is less than 104 Hz);<br>01: 52 Hz (or at the maximum ODR between the accelerometer and gyro if it is less than 52 Hz);<br>10: 26 Hz (or at the maximum ODR between the accelerometer and gyro if it is less than 26 Hz);<br>11: 12.5 Hz (or at the maximum ODR between the accelerometer and gyro if it is less than 12.5 Hz) |
| BATCH_EXT_ SENS_0_EN | Enable FIFO data batching of first slave. Default value: 0 |
| Slave0_numop[2:0] | Number of read operations on Sensor 1. Default value: 000 |

## 15.23 SLV1_ADD (18h)

I²C slave address of the second external sensor (Sensor 2) register (r/w)

**Table 381. SLV1_ADD register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Slave1_ add6 | Slave1_ add5 | Slave1_ add4 | Slave1_ add3 | Slave1_ add2 | Slave1_ add1 | Slave1_ add0 | r_1 |

**Table 382. SLV1_ADD register description**

| | |
|---|---|
| Slave1_add[6:0] | I²C slave address of Sensor 2 that can be read by the sensor hub. Default value: 0000000 |
| r_1 | Read operation on Sensor 2 enable. Default value: 0<br>(0: read operation disabled; 1: read operation enabled) |

chanhvinh.com



## 15.24 SLV1_SUBADD (19h)

Address of register on the second external sensor (Sensor 2) register (r/w)

**Table 383. SLV1_SUBADD register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Slave1_ reg7 | Slave1_ reg6 | Slave1_ reg5 | Slave1_ reg4 | Slave1_ reg3 | Slave1_ reg2 | Slave1_ reg1 | Slave1_ reg0 |

**Table 384. SLV1_SUBADD register description**

| | |
|---|---|
| Slave1_reg[7:0] | Address of register on Sensor 2 that has to be read/written according to the r_1 bit value in SLV1_ADD (18h). |

## 15.25 SLAVE1_CONFIG (1Ah)

Second external sensor (Sensor 2) configuration register (r/w)

**Table 385. SLAVE1_CONFIG register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | 0(1) | 0(1) | 0(1) | BATCH_EXT_ SENS_1_EN | Slave1_ numop2 | Slave1_ numop1 | Slave1_ numop0 |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 386. SLAVE1_CONFIG register description**

| | |
|---|---|
| BATCH_EXT_SENS_1_EN | Enable FIFO data batching of second slave. Default value: 0 |
| Slave1_numop[2:0] | Number of read operations on Sensor 2. Default value: 000 |

## 15.26 SLV2_ADD (1Bh)

I²C slave address of the third external sensor (Sensor 3) register (r/w)

**Table 387. SLV2_ADD register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Slave2_ add6 | Slave2_ add5 | Slave2_ add4 | Slave2_ add3 | Slave2_ add2 | Slave2_ add1 | Slave2_ add0 | r_2 |

**Table 388. SLV2_ADD register description**

| | |
|---|---|
| Slave2_add[6:0] | I²C slave address of Sensor 3 that can be read by the sensor hub. |
| r_2 | Read operation on Sensor 3 enable. Default value: 0<br>(0: read operation disabled; 1: read operation enabled) |

chanhvinh.com



## 15.27 SLV2_SUBADD (1Ch)

Address of register on the third external sensor (Sensor 3) register (r/w)

**Table 389. SLV2_SUBADD register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Slave2_ reg7 | Slave2_ reg6 | Slave2_ reg5 | Slave2_ reg4 | Slave2_ reg3 | Slave2_ reg2 | Slave2_ reg1 | Slave2_ reg0 |

**Table 390. SLV2_SUBADD register description**

| | |
|---|---|
| Slave2_reg[7:0] | Address of register on Sensor 3 that has to be read/written according to the r_2 bit value in SLV2_ADD (1Bh). |

## 15.28 SLAVE2_CONFIG (1Dh)

Third external sensor (Sensor 3) configuration register (r/w)

**Table 391. SLAVE2_CONFIG register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | 0(1) | 0(1) | 0(1) | BATCH_EXT _SENS_2_EN | Slave2_ numop2 | Slave2_ numop1 | Slave2_ numop0 |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 392. SLAVE2_CONFIG register description**

| | |
|---|---|
| BATCH_EXT_SENS_2_EN | Enable FIFO data batching of third slave. Default value: 0 |
| Slave2_numop[2:0] | Number of read operations on Sensor 3. Default value: 000 |

## 15.29 SLV3_ADD (1Eh)

I²C slave address of the fourth external sensor (Sensor 4) register (r/w)

**Table 393. SLV3_ADD register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Slave3_ add6 | Slave3_ add5 | Slave3_ add4 | Slave3_ add3 | Slave3_ add2 | Slave3_ add1 | Slave3_ add0 | r_3 |

**Table 394. SLV3_ADD register description**

| | |
|---|---|
| Slave3_add[6:0] | I²C slave address of Sensor 4 that can be read by the sensor hub. |
| r_3 | Read operation on Sensor 4 enable. Default value: 0<br>(0: read operation disabled; 1: read operation enabled) |

chanhvinh.com



## 15.30 SLV3_SUBADD (1Fh)

Address of register on the fourth external sensor (Sensor 4) register (r/w)

**Table 395. SLV3_SUBADD register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Slave3_ reg7 | Slave3_ reg6 | Slave3_ reg5 | Slave3_ reg4 | Slave3_ reg3 | Slave3_ reg2 | Slave3_ reg1 | Slave3_ reg0 |

**Table 396. SLV3_SUBADD register description**

| | |
|---|---|
| Slave3_reg[7:0] | Address of register on Sensor 4 that has to be read according to the r_3 bit value in SLV3_ADD (1Eh). |

## 15.31 SLAVE3_CONFIG (20h)

Fourth external sensor (Sensor 4) configuration register (r/w)

**Table 397. SLAVE3_CONFIG register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0(1) | 0(1) | 0(1) | 0(1) | BATCH_EXT _SENS_3_EN | Slave3_ numop2 | Slave3_ numop1 | Slave3_ numop0 |

1. *This bit must be set to '0' for the correct operation of the device.*

**Table 398. SLAVE3_CONFIG register description**

| | |
|---|---|
| BATCH_EXT_SENS_3_EN | Enable FIFO data batching of fourth slave. Default value: 0 |
| Slave3_numop[2:0] | Number of read operations on Sensor 4. Default value: 000 |

## 15.32 DATAWRITE_SLV0 (21h)

Data to be written in the slave device register (r/w)

**Table 399. DATAWRITE_SLV0 register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Slave0_ dataw7 | Slave0_ dataw6 | Slave0_ dataw5 | Slave0_ dataw4 | Slave0_ dataw3 | Slave0_ dataw2 | Slave0_ dataw1 | Slave0_ dataw0 |

**Table 400. DATAWRITE_SLV0 register description**

| | |
|---|---|
| Slave0_dataw[7:0] | Data to be written into the slave 0 device according to the rw_0 bit in register SLV0_ADD (15h).<br>Default value: 00000000 |

chanhvinh.com



## 15.33 STATUS_MASTER (22h)

Sensor hub source register (r)

**Table 401. STATUS_MASTER register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| WR_ONCE_DONE | SLAVE3_NACK | SLAVE2_NACK | SLAVE1_NACK | SLAVE0_NACK | 0 | 0 | SENS_HUB_ENDOP |

**Table 402. STATUS_MASTER register description**

| | |
|---|---|
| WR_ONCE_DONE | When the bit WRITE_ONCE in MASTER_CONFIG (14h) is configured as 1, this bit is set to 1 when the write operation on slave 0 has been performed and completed. Default value: 0 |
| SLAVE3_NACK | This bit is set to 1 if Not acknowledge occurs on slave 3 communication. Default value: 0 |
| SLAVE2_NACK | This bit is set to 1 if Not acknowledge occurs on slave 2 communication. Default value: 0 |
| SLAVE1_NACK | This bit is set to 1 if Not acknowledge occurs on slave 1 communication. Default value: 0 |
| SLAVE0_NACK | This bit is set to 1 if Not acknowledge occurs on slave 0 communication. Default value: 0 |
| SENS_HUB_ENDOP | Sensor hub communication status. Default value: 0<br>(0: sensor hub communication not concluded;<br>1: sensor hub communication concluded) |

chanhvinh.com



# 16 Soldering information

The LGA package is compliant with the ECOPACK, RoHS and "Green" standard.
It is qualified for soldering heat resistance according to JEDEC J-STD-020.
Land pattern and soldering recommendations are available at www.st.com/mems.

chanhvinh.com

chanhvinh.com

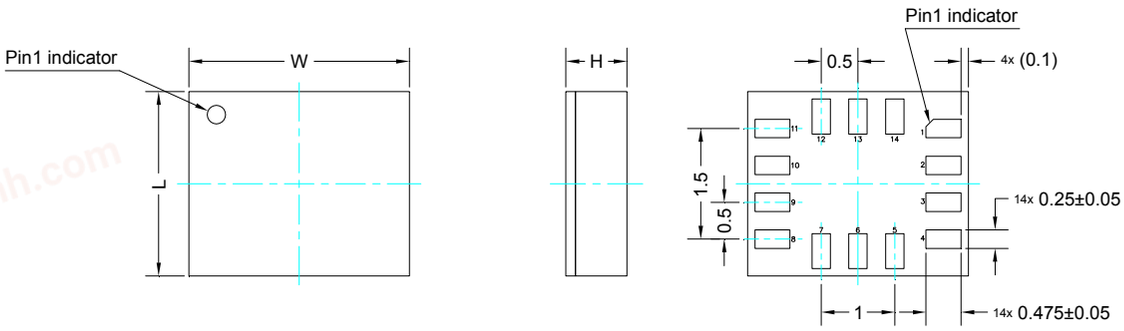


# 17 Package information

In order to meet environmental requirements, ST offers these devices in different grades of ECOPACK packages, depending on their level of environmental compliance. ECOPACK specifications, grade definitions and product status are available at: www.st.com. ECOPACK is an ST trademark.

## 17.1 LGA-14L package information

**Figure 23. LGA-14L 2.5 x 3.0 x 0.86 mm package outline and mechanical data**




TOP VIEW BOTTOM VIEW

Dimensions are in millimeter unless otherwise specified
General tolerance is +/-0.1mm unless otherwise specified

**OUTER DIMENSIONS**

| ITEM | DIMENSION [mm] | TOLERANCE [mm] |
|---|---|---|
| Length [L] | 2.50 | ±0.1 |
| Width [W] | 3.00 | ±0.1 |
| Height [H] | 0.86 | MAX |

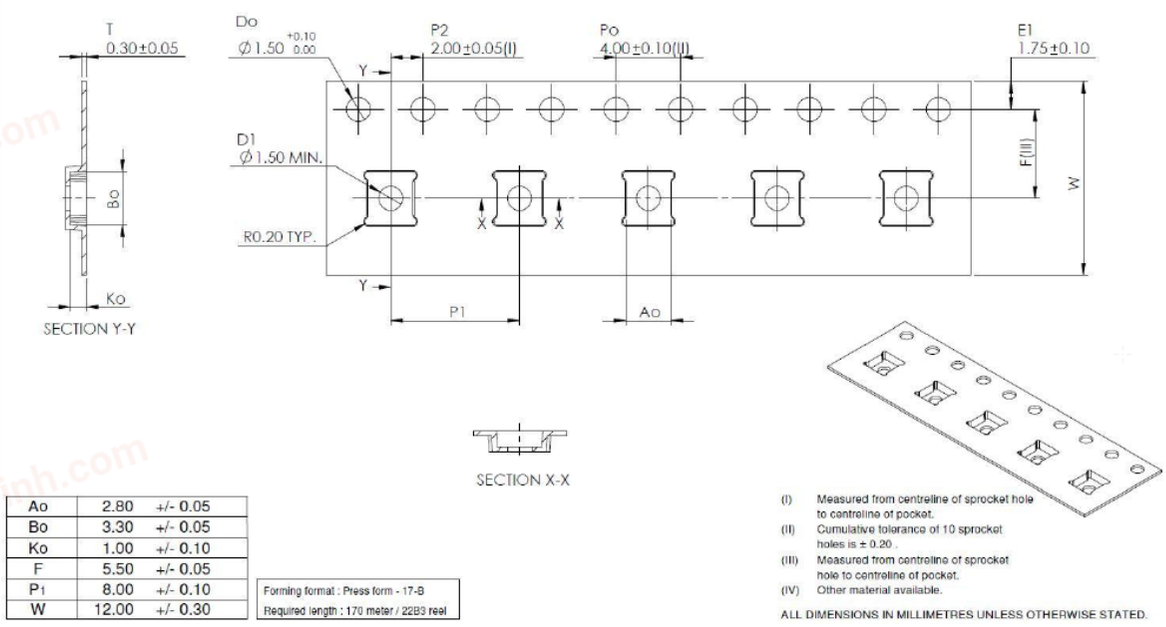
DM00249496_1

chanhvinh.com



## 17.2 LGA-14 packing information

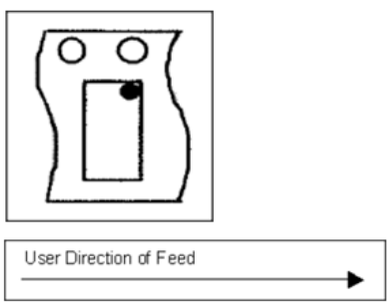
Figure 24. **Carrier tape information for LGA-14 package**




| | | |
|---|---|---|
| Ao | 2.80 | +/- 0.05 |
| Bo | 3.30 | +/- 0.05 |
| Ko | 1.00 | +/- 0.10 |
| F | 5.50 | +/- 0.05 |
| P1 | 8.00 | +/- 0.10 |
| W | 12.00 | +/- 0.30 |

Forming format : Press form - 17-B
Required length : 170 meter / 22B3 reel

Figure 25. **LGA-14 package orientation in carrier tape**



chanhvinh.com



Figure 26. **Reel information for carrier tape of LGA-14 package**

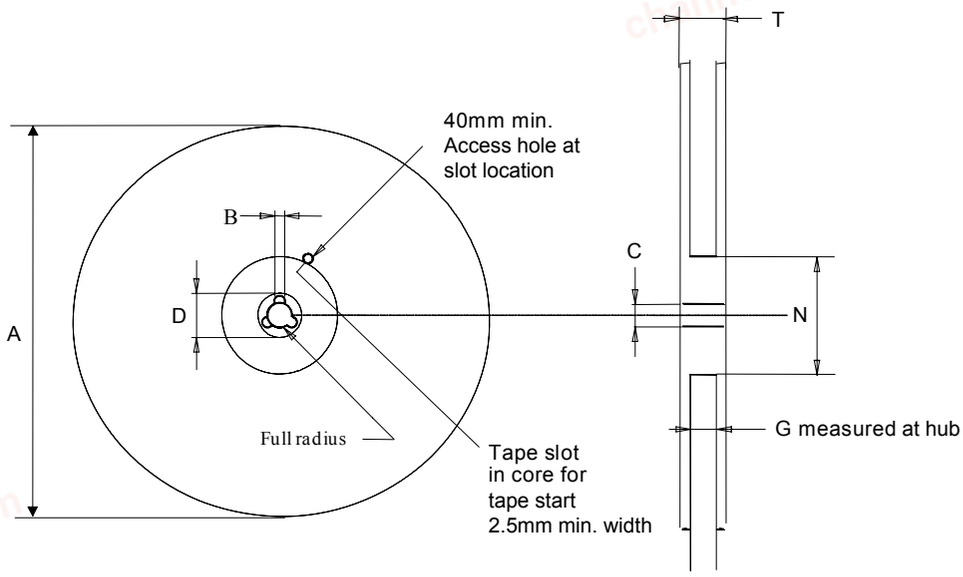



Table 403. **Reel dimensions for carrier tape of LGA-14 package**

| Reel dimensions (mm) | |
|---|---|
| A (max) | 330 |
| B (min) | 1.5 |
| C | 13 ±0.25 |
| D (min) | 20.2 |
| N (min) | 60 |
| G | 12.4 +2/-0 |
| T (max) | 18.4 |

chanhvinh.com



## Revision history

Table 404. **Document revision history**

| Date | Revision | Changes |
|---|---|---|
| 17-Mar-2021 | 1 | Initial release |

chanhvinh.com

chanhvinh.com



# Contents

1 **Overview** . . . . . 3
2 **Embedded low-power features** . . . . . 4
2.1 Tilt detection . . . . . 4
2.2 Significant Motion Detection . . . . . 4
2.3 Finite State Machine . . . . . 4
2.4 Machine Learning Core . . . . . 6
3 **Pin description** . . . . . 7
3.1 Pin connections . . . . . 8
4 **Module specifications** . . . . . 10
4.1 Mechanical characteristics . . . . . 10
4.2 Electrical characteristics . . . . . 13
4.3 Temperature sensor characteristics . . . . . 13
4.4 Communication interface characteristics . . . . . 14
4.4.1 SPI - serial peripheral interface . . . . . 14
4.4.2 I²C - inter-IC control interface . . . . . 15
4.5 Absolute maximum ratings . . . . . 16
4.6 Terminology . . . . . 17
4.6.1 Sensitivity . . . . . 17
4.6.2 Zero-g and zero-rate level . . . . . 17
5 **Digital interfaces** . . . . . 18
5.1 I²C/SPI interface . . . . . 18
5.1.1 I²C serial interface . . . . . 18
5.1.2 SPI bus interface . . . . . 21
5.2 MIPI I3CSM interface . . . . . 25
5.2.1 MIPI I3CSM slave interface . . . . . 25
5.2.2 MIPI I3CSM CCC supported commands . . . . . 25
5.3 I²C/I3C coexistence in LSM6DSO . . . . . 27
5.4 Master I²C interface . . . . . 28
6 **Functionality** . . . . . 29
6.1 Operating modes . . . . . 29

chanhvinh.com



6.2 Accelerometer power modes . . . 29
6.2.1 Accelerometer ultra-low-power mode . . . 29
6.3 Gyroscope power modes . . . 29
6.4 Block diagram of filters . . . 30
6.4.1 Block diagrams of the accelerometer filters . . . 30
6.4.2 Block diagrams of the gyroscope filters . . . 32
6.5 FIFO . . . 33
6.5.1 Bypass mode . . . 33
6.5.2 FIFO mode . . . 34
6.5.3 Continuous mode . . . 34
6.5.4 Continuous-to-FIFO mode . . . 34
6.5.5 Bypass-to-Continuous mode . . . 34
6.5.6 Bypass-to-FIFO mode . . . 35
6.5.7 FIFO reading procedure . . . 35
**7 Application hints . . . 36**
7.1 LSM6DSO32X electrical connections in Mode 1 . . . 36
7.2 LSM6DSO32X electrical connections in Mode 2 . . . 37
**8 Register mapping . . . 40**
**9 Register description . . . 43**
9.1 FUNC_CFG_ACCESS (01h) . . . 43
9.2 PIN_CTRL (02h) . . . 43
9.3 FIFO_CTRL1 (07h) . . . 44
9.4 FIFO_CTRL2 (08h) . . . 44
9.5 FIFO_CTRL3 (09h) . . . 45
9.6 FIFO_CTRL4 (0Ah) . . . 46
9.7 COUNTER_BDR_REG1 (0Bh) . . . 47
9.8 COUNTER_BDR_REG2 (0Ch) . . . 47
9.9 INT1_CTRL (0Dh) . . . 48
9.10 INT2_CTRL (0Eh) . . . 49
9.11 WHO_AM_I (0Fh) . . . 49
9.12 CTRL1_XL (10h) . . . 50

chanhvinh.com



9.13 CTRL2_G (11h) . . . 51
9.14 CTRL3_C (12h) . . . 52
9.15 CTRL4_C (13h) . . . 53
9.16 CTRL5_C (14h) . . . 54
9.17 CTRL6_C (15h) . . . 55
9.18 CTRL7_G (16h) . . . 56
9.19 CTRL8_XL (17h) . . . 57
9.20 CTRL9_XL (18h) . . . 59
9.21 CTRL10_C (19h) . . . 60
9.22 ALL_INT_SRC (1Ah) . . . 60
9.23 WAKE_UP_SRC (1Bh) . . . 61
9.24 TAP_SRC (1Ch) . . . 62
9.25 D6D_SRC (1Dh) . . . 63
9.26 STATUS_REG (1Eh) . . . 64
9.27 OUT_TEMP_L (20h), OUT_TEMP_H (21h) . . . 65
9.28 OUTX_L_G (22h) and OUTX_H_G (23h) . . . 65
9.29 OUTY_L_G (24h) and OUTY_H_G (25h) . . . 66
9.30 OUTZ_L_G (26h) and OUTZ_H_G (27h) . . . 66
9.31 OUTX_L_A (28h) and OUTX_H_A (29h) . . . 67
9.32 OUTY_L_A (2Ah) and OUTY_H_A (2Bh) . . . 67
9.33 OUTZ_L_A (2Ch) and OUTZ_H_A (2Dh) . . . 68
9.34 EMB_FUNC_STATUS_MAINPAGE (35h) . . . 68
9.35 FSM_STATUS_A_MAINPAGE (36h) . . . 69
9.36 FSM_STATUS_B_MAINPAGE (37h) . . . 69
9.37 MLC_STATUS_MAINPAGE (38h) . . . 70
9.38 STATUS_MASTER_MAINPAGE (39h) . . . 71
9.39 FIFO_STATUS1 (3Ah) . . . 71
9.40 FIFO_STATUS2 (3Bh) . . . 72
9.41 TIMESTAMP0 (40h), TIMESTAMP1 (41h), TIMESTAMP2 (42h), and TIMESTAMP3 (43h) 73
9.42 TAP_CFG0 (56h) . . . 74
9.43 TAP_CFG1 (57h) . . . 75

chanhvinh.com



9.44 TAP_CFG2 (58h) . . . 75
9.45 TAP_THS_6D (59h) . . . 76
9.46 INT_DUR2 (5Ah) . . . 77
9.47 WAKE_UP_THS (5Bh) . . . 77
9.48 WAKE_UP_DUR (5Ch) . . . 78
9.49 FREE_FALL (5Dh) . . . 79
9.50 MD1_CFG (5Eh) . . . 80
9.51 MD2_CFG (5Fh) . . . 81
9.52 I3C_BUS_AVB (62h) . . . 82
9.53 INTERNAL_FREQ_FINE (63h) . . . 82
9.54 X_OFS_USR (73h) . . . 82
9.55 Y_OFS_USR (74h) . . . 83
9.56 Z_OFS_USR (75h) . . . 83
9.57 FIFO_DATA_OUT_TAG (78h) . . . 84
9.58 FIFO_DATA_OUT_X_L (79h) and FIFO_DATA_OUT_X_H (7Ah) . . . 85
9.59 FIFO_DATA_OUT_Y_L (7Bh) and FIFO_DATA_OUT_Y_H (7Ch) . . . 85
9.60 FIFO_DATA_OUT_Z_L (7Dh) and FIFO_DATA_OUT_Z_H (7Eh) . . . 85

**10 Embedded functions register mapping . . . 86**

**11 Embedded functions register description . . . 88**

11.1 PAGE_SEL (02h) . . . 88
11.2 EMB_FUNC_EN_A (04h) . . . 88
11.3 EMB_FUNC_EN_B (05h) . . . 89
11.4 PAGE_ADDRESS (08h) . . . 89
11.5 PAGE_VALUE (09h) . . . 90
11.6 EMB_FUNC_INT1 (0Ah) . . . 90
11.7 FSM_INT1_A (0Bh) . . . 91
11.8 FSM_INT1_B (0Ch) . . . 92
11.9 MLC_INT1 (0Dh) . . . 93
11.10 EMB_FUNC_INT2 (0Eh) . . . 94
11.11 FSM_INT2_A (0Fh) . . . 95
11.12 FSM_INT2_B (10h) . . . 96

chanhvinh.com



11.13 MLC_INT2 (11h) . . . . . 97
11.14 EMB_FUNC_STATUS (12h) . . . . . 97
11.15 FSM_STATUS_A (13h) . . . . . 98
11.16 FSM_STATUS_B (14h) . . . . . 98
11.17 MLC_STATUS (15h) . . . . . 99
11.18 PAGE_RW (17h) . . . . . 99
11.19 EMB_FUNC_FIFO_CFG (44h) . . . . . 100
11.20 FSM_ENABLE_A (46h) . . . . . 100
11.21 FSM_ENABLE_B (47h) . . . . . 100
11.22 FSM_LONG_COUNTER_L (48h) and FSM_LONG_COUNTER_H (49h) . . . . . 101
11.23 FSM_LONG_COUNTER_CLEAR (4Ah) . . . . . 101
11.24 FSM_OUTS1 (4Ch) . . . . . 102
11.25 FSM_OUTS2 (4Dh) . . . . . 102
11.26 FSM_OUTS3 (4Eh) . . . . . 103
11.27 FSM_OUTS4 (4Fh) . . . . . 103
11.28 FSM_OUTS5 (50h) . . . . . 104
11.29 FSM_OUTS6 (51h) . . . . . 104
11.30 FSM_OUTS7 (52h) . . . . . 105
11.31 FSM_OUTS8 (53h) . . . . . 105
11.32 FSM_OUTS9 (54h) . . . . . 106
11.33 FSM_OUTS10 (55h) . . . . . 106
11.34 FSM_OUTS11 (56h) . . . . . 107
11.35 FSM_OUTS12 (57h) . . . . . 107
11.36 FSM_OUTS13 (58h) . . . . . 108
11.37 FSM_OUTS14 (59h) . . . . . 108
11.38 FSM_OUTS15 (5Ah) . . . . . 109
11.39 FSM_OUTS16 (5Bh) . . . . . 109
11.40 EMB_FUNC_ODR_CFG_B (5Fh) . . . . . 110
11.41 EMB_FUNC_ODR_CFG_C (60h) . . . . . 110
11.42 STEP_COUNTER_L (62h) and STEP_COUNTER_H (63h) . . . . . 111
11.43 EMB_FUNC_SRC (64h) . . . . . 111

chanhvinh.com



11.44 EMB_FUNC_INIT_A (66h) . . . 112
11.45 EMB_FUNC_INIT_B (67h) . . . 112
11.46 MLC0_SRC (70h) . . . 112
11.47 MLC1_SRC (71h) . . . 113
11.48 MLC2_SRC (72h) . . . 113
11.49 MLC3_SRC (73h) . . . 113
11.50 MLC4_SRC (74h) . . . 114
11.51 MLC5_SRC (75h) . . . 114
11.52 MLC6_SRC (76h) . . . 114
11.53 MLC7_SRC (77h) . . . 114

**12 Embedded advanced features pages . . . 115**

**13 Embedded advanced features register description . . . 117**

13.1 Page 0 - Embedded advanced features registers . . . 117
13.1.1 MAG_SENSITIVITY_L (BAh) and MAG_SENSITIVITY_H (BBh) . . . 117
13.1.2 MAG_OFFX_L (C0h) and MAG_OFFX_H (C1h) . . . 118
13.1.3 MAG_OFFY_L (C2h) and MAG_OFFY_H (C3h) . . . 118
13.1.4 MAG_OFFZ_L (C4h) and MAG_OFFZ_H (C5h) . . . 119
13.1.5 MAG_SI_XX_L (C6h) and MAG_SI_XX_H (C7h) . . . 119
13.1.6 MAG_SI_XY_L (C8h) and MAG_SI_XY_H (C9h) . . . 120
13.1.7 MAG_SI_XZ_L (CAh) and MAG_SI_XZ_H (CBh) . . . 120
13.1.8 MAG_SI_YY_L (CCh) and MAG_SI_YY_H (CDh) . . . 121
13.1.9 MAG_SI_YZ_L (CEh) and MAG_SI_YZ_H (CFh) . . . 121
13.1.10 MAG_SI_ZZ_L (D0h) and MAG_SI_ZZ_H (D1h) . . . 122
13.1.11 MAG_CFG_A (D4h) . . . 123
13.1.12 MAG_CFG_B (D5h) . . . 123
13.2 Page 1 - Embedded advanced features registers . . . 124
13.2.1 FSM_LC_TIMEOUT_L (7Ah) and FSM_LC_TIMEOUT_H (7Bh) . . . 124
13.2.2 FSM_PROGRAMS (7Ch) . . . 124
13.2.3 FSM_START_ADD_L (7Eh) and FSM_START_ADD_H (7Fh) . . . 125
13.2.4 PEDO_CMD_REG (83h) . . . 125
13.2.5 PEDO_DEB_STEPS_CONF (84h) . . . 126
13.2.6 PEDO_SC_DELTAT_L (D0h) & PEDO_SC_DELTAT_H (D1h) . . . 126

chanhvinh.com



13.2.7 MLC_MAG_SENSITIVITY_L (E8h) and MLC_MAG_SENSITIVITY_H (E9h) . . . . . 126
**14 Sensor hub register mapping . . . . . 127**
**15 Sensor hub register description. . . . . 128**
15.1 SENSOR_HUB_1 (02h) . . . . . 128
15.2 SENSOR_HUB_2 (03h) . . . . . 128
15.3 SENSOR_HUB_3 (04h) . . . . . 128
15.4 SENSOR_HUB_4 (05h) . . . . . 129
15.5 SENSOR_HUB_5 (06h) . . . . . 129
15.6 SENSOR_HUB_6 (07h) . . . . . 129
15.7 SENSOR_HUB_7 (08h) . . . . . 130
15.8 SENSOR_HUB_8 (09h) . . . . . 130
15.9 SENSOR_HUB_9 (0Ah) . . . . . 130
15.10 SENSOR_HUB_10 (0Bh) . . . . . 131
15.11 SENSOR_HUB_11 (0Ch) . . . . . 131
15.12 SENSOR_HUB_12 (0Dh) . . . . . 131
15.13 SENSOR_HUB_13 (0Eh) . . . . . 132
15.14 SENSOR_HUB_14 (0Fh) . . . . . 132
15.15 SENSOR_HUB_15 (10h) . . . . . 132
15.16 SENSOR_HUB_16 (11h) . . . . . 133
15.17 SENSOR_HUB_17 (12h) . . . . . 133
15.18 SENSOR_HUB_18 (13h) . . . . . 133
15.19 MASTER_CONFIG (14h) . . . . . 134
15.20 SLV0_ADD (15h) . . . . . 134
15.21 SLV0_SUBADD (16h) . . . . . 135
15.22 SLAVE0_CONFIG (17h) . . . . . 135
15.23 SLV1_ADD (18h) . . . . . 135
15.24 SLV1_SUBADD (19h) . . . . . 136
15.25 SLAVE1_CONFIG (1Ah) . . . . . 136
15.26 SLV2_ADD (1Bh) . . . . . 136
15.27 SLV2_SUBADD (1Ch) . . . . . 137
15.28 SLAVE2_CONFIG (1Dh) . . . . . 137

chanhvinh.com



15.29 SLV3_ADD (1Eh) . . . . . 137
15.30 SLV3_SUBADD (1Fh) . . . . . 138
15.31 SLAVE3_CONFIG (20h) . . . . . 138
15.32 DATAWRITE_SLV0 (21h) . . . . . 138
15.33 STATUS_MASTER (22h) . . . . . 139
**16 Soldering information . . . . . 140**
**17 Package information . . . . . 141**
17.1 LGA-14L package information . . . . . 141
17.2 LGA-14 packing information . . . . . 142
**Revision history . . . . . 144**

chanhvinh.com

chanhvinh.com



# List of tables

Table 1. Pin description . . . 9
Table 2. Mechanical characteristics . . . 10
Table 3. Electrical characteristics . . . 13
Table 4. Temperature sensor characteristics . . . 13
Table 5. SPI slave timing values (in mode 3) . . . 14
Table 6. I²C slave timing values . . . 15
Table 7. Absolute maximum ratings . . . 16
Table 8. Serial interface pin description . . . 18
Table 9. I²C terminology . . . 18
Table 10. SAD+Read/Write patterns . . . 19
Table 11. Transfer when master is writing one byte to slave . . . 19
Table 12. Transfer when master is writing multiple bytes to slave . . . 19
Table 13. Transfer when master is receiving (reading) one byte of data from slave . . . 19
Table 14. Transfer when master is receiving (reading) multiple bytes of data from slave . . . 19
Table 15. MIPI I3C$^{SM}$ CCC commands . . . 25
Table 16. Master I²C pin details . . . 28
Table 17. Gyroscope LPF2 bandwidth selection . . . 32
Table 18. Internal pin status . . . 38
Table 19. Registers address map . . . 40
Table 20. FUNC_CFG_ACCESS register . . . 43
Table 21. FUNC_CFG_ACCESS register description . . . 43
Table 22. PIN_CTRL register . . . 43
Table 23. PIN_CTRL register description . . . 43
Table 24. FIFO_CTRL1 register . . . 44
Table 25. FIFO_CTRL1 register description . . . 44
Table 26. FIFO_CTRL2 register . . . 44
Table 27. FIFO_CTRL2 register description . . . 44
Table 28. FIFO_CTRL3 register . . . 45
Table 29. FIFO_CTRL3 register description . . . 45
Table 30. FIFO_CTRL4 register . . . 46
Table 31. FIFO_CTRL4 register description . . . 46
Table 32. COUNTER_BDR_REG1 register . . . 47
Table 33. COUNTER_BDR_REG1 register description . . . 47
Table 34. COUNTER_BDR_REG2 register . . . 47
Table 35. COUNTER_BDR_REG2 register description . . . 47
Table 36. INT1_CTRL register . . . 48
Table 37. INT1_CTRL register description . . . 48
Table 38. INT2_CTRL register . . . 49
Table 39. INT2_CTRL register description . . . 49
Table 40. WhoAmI register . . . 49
Table 41. CTRL1_XL register . . . 50
Table 42. CTRL1_XL register description . . . 50
Table 43. Accelerometer ODR register setting . . . 50
Table 44. Accelerometer full-scale selection . . . 50
Table 45. CTRL2_G register . . . 51
Table 46. CTRL2_G register description . . . 51
Table 47. Gyroscope ODR configuration setting . . . 51
Table 48. CTRL3_C register . . . 52
Table 49. CTRL3_C register description . . . 52
Table 50. CTRL4_C register . . . 53
Table 51. CTRL4_C register description . . . 53
Table 52. CTRL5_C register . . . 54

chanhvinh.com



Table 53. CTRL5_C register description . . . 54
Table 54. Angular rate sensor self-test mode selection . . . 54
Table 55. Linear acceleration sensor self-test mode selection . . . 54
Table 56. CTRL6_C register . . . 55
Table 57. CTRL6_C register description . . . 55
Table 58. Trigger mode selection . . . 55
Table 59. Gyroscope LPF1 bandwidth selection . . . 55
Table 60. CTRL7_G register . . . 56
Table 61. CTRL7_G register description . . . 56
Table 62. CTRL8_XL register . . . 57
Table 63. CTRL8_XL register description . . . 57
Table 64. Accelerometer bandwidth configurations . . . 57
Table 65. CTRL9_XL register . . . 59
Table 66. CTRL9_XL register description . . . 59
Table 67. CTRL10_C register . . . 60
Table 68. CTRL10_C register description . . . 60
Table 69. ALL_INT_SRC register . . . 60
Table 70. ALL_INT_SRC register description . . . 60
Table 71. WAKE_UP_SRC register . . . 61
Table 72. WAKE_UP_SRC register description . . . 61
Table 73. TAP_SRC register . . . 62
Table 74. TAP_SRC register description . . . 62
Table 75. D6D_SRC register . . . 63
Table 76. D6D_SRC register description . . . 63
Table 77. STATUS_REG register . . . 64
Table 78. STATUS_REG register description . . . 64
Table 79. OUT_TEMP_L register . . . 65
Table 80. OUT_TEMP_H register . . . 65
Table 81. OUT_TEMP register description . . . 65
Table 82. OUTX_L_G register . . . 65
Table 83. OUTX_H_G register . . . 65
Table 84. OUTX_H_G register description . . . 65
Table 85. OUTY_L_G register . . . 66
Table 86. OUTY_H_G register . . . 66
Table 87. OUTY_H_G register description . . . 66
Table 88. OUTZ_L_G register . . . 66
Table 89. OUTZ_H_G register . . . 66
Table 90. OUTZ_H_G register description . . . 66
Table 91. OUTX_L_A register . . . 67
Table 92. OUTX_H_A register . . . 67
Table 93. OUTX_H_A register description . . . 67
Table 94. OUTY_L_A register . . . 67
Table 95. OUTY_H_A register . . . 67
Table 96. OUTY_H_A register description . . . 67
Table 97. OUTZ_L_A register . . . 68
Table 98. OUTZ_H_A register . . . 68
Table 99. OUTZ_H_A register description . . . 68
Table 100. EMB_FUNC_STATUS_MAINPAGE register . . . 68
Table 101. EMB_FUNC_STATUS_MAINPAGE register description . . . 68
Table 102. FSM_STATUS_A_MAINPAGE register . . . 69
Table 103. FSM_STATUS_A_MAINPAGE register description . . . 69
Table 104. FSM_STATUS_B_MAINPAGE register . . . 69
Table 105. FSM_STATUS_B_MAINPAGE register description . . . 69
Table 106. MLC_STATUS _MAINPAGE register . . . 70

chanhvinh.com



Table 107. MLC_STATUS_MAINPAGE register description . . . 70
Table 108. STATUS_MASTER_MAINPAGE register . . . 71
Table 109. STATUS_MASTER_MAINPAGE register description . . . 71
Table 110. FIFO_STATUS1 register . . . 71
Table 111. FIFO_STATUS1 register description . . . 71
Table 112. FIFO_STATUS2 register . . . 72
Table 113. FIFO_STATUS2 register description . . . 72
Table 114. TIMESTAMP output registers . . . 73
Table 115. TIMESTAMP output register description . . . 73
Table 116. TAP_CFG0 register . . . 74
Table 117. TAP_CFG0 register description . . . 74
Table 118. TAP_CFG1 register . . . 75
Table 119. TAP_CFG1 register description . . . 75
Table 120. TAP priority decoding . . . 75
Table 121. TAP_CFG2 register . . . 75
Table 122. TAP_CFG2 register description . . . 75
Table 123. TAP_THS_6D register . . . 76
Table 124. TAP_THS_6D register description . . . 76
Table 125. Threshold for D4D/D6D function . . . 76
Table 126. INT_DUR2 register . . . 77
Table 127. INT_DUR2 register description . . . 77
Table 128. WAKE_UP_THS register . . . 77
Table 129. WAKE_UP_THS register description . . . 77
Table 130. WAKE_UP_DUR register . . . 78
Table 131. WAKE_UP_DUR register description . . . 78
Table 132. FREE_FALL register . . . 79
Table 133. FREE_FALL register description . . . 79
Table 134. Threshold for free-fall function . . . 79
Table 135. MD1_CFG register . . . 80
Table 136. MD1_CFG register description . . . 80
Table 137. MD2_CFG register . . . 81
Table 138. MD2_CFG register description . . . 81
Table 139. I3C_BUS_AVB register . . . 82
Table 140. I3C_BUS_AVB register description . . . 82
Table 141. INTERNAL_FREQ_FINE register . . . 82
Table 142. INTERNAL_FREQ_FINE register description . . . 82
Table 143. X_OFS_USR register . . . 82
Table 144. X_OFS_USR register description . . . 82
Table 145. Y_OFS_USR register . . . 83
Table 146. Y_OFS_USR register description . . . 83
Table 147. Z_OFS_USR register . . . 83
Table 148. Z_OFS_USR register description . . . 83
Table 149. FIFO_DATA_OUT_TAG register . . . 84
Table 150. FIFO_DATA_OUT_TAG register description . . . 84
Table 151. FIFO tag . . . 84
Table 152. FIFO_DATA_OUT_X_H and FIFO_DATA_OUT_X_L registers . . . 85
Table 153. FIFO_DATA_OUT_X_H and FIFO_DATA_OUT_X_L register description . . . 85
Table 154. FIFO_DATA_OUT_Y_H and FIFO_DATA_OUT_Y_L registers . . . 85
Table 155. FIFO_DATA_OUT_Y_H and FIFO_DATA_OUT_Y_L register description . . . 85
Table 156. FIFO_DATA_OUT_Z_H and FIFO_DATA_OUT_Z_L registers . . . 85
Table 157. FIFO_DATA_OUT_Z_H and FIFO_DATA_OUT_Z_L register description . . . 85
Table 158. Register address map - embedded functions . . . 86
Table 159. PAGE_SEL register . . . 88
Table 160. PAGE_SEL register description . . . 88

chanhvinh.com



Table 161. EMB_FUNC_EN_A register . . . 88
Table 162. EMB_FUNC_EN_A register description . . . 88
Table 163. EMB_FUNC_EN_B register . . . 89
Table 164. EMB_FUNC_EN_B register description . . . 89
Table 165. PAGE_ADDRESS register . . . 89
Table 166. PAGE_ADDRESS register description . . . 89
Table 167. PAGE_VALUE register . . . 90
Table 168. PAGE_VALUE register description . . . 90
Table 169. EMB_FUNC_INT1 register . . . 90
Table 170. EMB_FUNC_INT1 register description . . . 90
Table 171. FSM_INT1_A register . . . 91
Table 172. FSM_INT1_A register description . . . 91
Table 173. FSM_INT1_B register . . . 92
Table 174. FSM_INT1_B register description . . . 92
Table 175. MLC_INT1 register . . . 93
Table 176. MLC_INT1 register description . . . 93
Table 177. EMB_FUNC_INT2 register . . . 94
Table 178. EMB_FUNC_INT2 register description . . . 94
Table 179. FSM_INT2_A register . . . 95
Table 180. FSM_INT2_A register description . . . 95
Table 181. FSM_INT2_B register . . . 96
Table 182. FSM_INT2_B register description . . . 96
Table 183. MLC_INT2 register . . . 97
Table 184. MLC_INT2 register description . . . 97
Table 185. EMB_FUNC_STATUS register . . . 97
Table 186. EMB_FUNC_STATUS register description . . . 97
Table 187. FSM_STATUS_A register . . . 98
Table 188. FSM_STATUS_A register description . . . 98
Table 189. FSM_STATUS_B register . . . 98
Table 190. FSM_STATUS_B register description . . . 98
Table 191. MLC_STATUS register . . . 99
Table 192. MLC_STATUS register description . . . 99
Table 193. PAGE_RW register . . . 99
Table 194. PAGE_RW register description . . . 99
Table 195. EMB_FUNC_FIFO_CFG register . . . 100
Table 196. EMB_FUNC_FIFO_CFG register description . . . 100
Table 197. FSM_ENABLE_A register . . . 100
Table 198. FSM_ENABLE_A register description . . . 100
Table 199. FSM_ENABLE_B register . . . 100
Table 200. FSM_ENABLE_B register description . . . 100
Table 201. FSM_LONG_COUNTER_L register . . . 101
Table 202. FSM_LONG_COUNTER_L register description . . . 101
Table 203. FSM_LONG_COUNTER_H register . . . 101
Table 204. FSM_LONG_COUNTER_H register description . . . 101
Table 205. FSM_LONG_COUNTER_CLEAR register . . . 101
Table 206. FSM_LONG_COUNTER_CLEAR register description . . . 101
Table 207. FSM_OUTS1 register . . . 102
Table 208. FSM_OUTS1 register description . . . 102
Table 209. FSM_OUTS2 register . . . 102
Table 210. FSM_OUTS2 register description . . . 102
Table 211. FSM_OUTS3 register . . . 103
Table 212. FSM_OUTS3 register description . . . 103
Table 213. FSM_OUTS4 register . . . 103
Table 214. FSM_OUTS4 register description . . . 103

chanhvinh.com



Table 215. FSM_OUTS5 register . . . 104
Table 216. FSM_OUTS5 register description . . . 104
Table 217. FSM_OUTS6 register . . . 104
Table 218. FSM_OUTS6 register description . . . 104
Table 219. FSM_OUTS7 register . . . 105
Table 220. FSM_OUTS7 register description . . . 105
Table 221. FSM_OUTS8 register . . . 105
Table 222. FSM_OUTS8 register description . . . 105
Table 223. FSM_OUTS9 register . . . 106
Table 224. FSM_OUTS9 register description . . . 106
Table 225. FSM_OUTS10 register . . . 106
Table 226. FSM_OUTS10 register description . . . 106
Table 227. FSM_OUTS11 register . . . 107
Table 228. FSM_OUTS11 register description . . . 107
Table 229. FSM_OUTS12 register . . . 107
Table 230. FSM_OUTS12 register description . . . 107
Table 231. FSM_OUTS13 register . . . 108
Table 232. FSM_OUTS13 register description . . . 108
Table 233. FSM_OUTS14 register . . . 108
Table 234. FSM_OUTS14 register description . . . 108
Table 235. FSM_OUTS15 register . . . 109
Table 236. FSM_OUTS15 register description . . . 109
Table 237. FSM_OUTS16 register . . . 109
Table 238. FSM_OUTS16 register description . . . 109
Table 239. EMB_FUNC_ODR_CFG_B register . . . 110
Table 240. EMB_FUNC_ODR_CFG_B register description . . . 110
Table 241. EMB_FUNC_ODR_CFG_C register . . . 110
Table 242. EMB_FUNC_ODR_CFG_C register description . . . 110
Table 243. STEP_COUNTER_L register . . . 111
Table 244. STEP_COUNTER_L register description . . . 111
Table 245. STEP_COUNTER_H register . . . 111
Table 246. STEP_COUNTER_H register description . . . 111
Table 247. EMB_FUNC_SRC register . . . 111
Table 248. EMB_FUNC_SRC register description . . . 111
Table 249. EMB_FUNC_INIT_A register . . . 112
Table 250. EMB_FUNC_INIT_A register description . . . 112
Table 251. EMB_FUNC_INIT_B register . . . 112
Table 252. EMB_FUNC_INIT_B register description . . . 112
Table 253. MLC0_SRC register . . . 112
Table 254. MLC0_SRC register description . . . 112
Table 255. MLC1_SRC register . . . 113
Table 256. MLC1_SRC register description . . . 113
Table 257. MLC2_SRC register . . . 113
Table 258. MLC2_SRC register description . . . 113
Table 259. MLC3_SRC register . . . 113
Table 260. MLC3_SRC register description . . . 113
Table 261. MLC4_SRC register . . . 114
Table 262. MLC4_SRC register description . . . 114
Table 263. MLC5_SRC register . . . 114
Table 264. MLC5_SRC register description . . . 114
Table 265. MLC6_SRC register . . . 114
Table 266. MLC6_SRC register description . . . 114
Table 267. MLC7_SRC register . . . 114
Table 268. MLC7_SRC register description . . . 114

chanhvinh.com



Table 269. Register address map - embedded advanced features page 0 . . . 115
Table 270. Register address map - embedded advanced features page 1 . . . 116
Table 271. MAG_SENSITIVITY_L register . . . 117
Table 272. MAG_SENSITIVITY_L register description . . . 117
Table 273. MAG_SENSITIVITY_H register . . . 117
Table 274. MAG_SENSITIVITY_H register description . . . 117
Table 275. MAG_OFFX_L register . . . 118
Table 276. MAG_OFFX_L register description . . . 118
Table 277. MAG_OFFX_H register . . . 118
Table 278. MAG_OFFX_H register description . . . 118
Table 279. MAG_OFFY_L register . . . 118
Table 280. MAG_OFFY_L register description . . . 118
Table 281. MAG_OFFY_H register . . . 118
Table 282. MAG_OFFY_H register description . . . 118
Table 283. MAG_OFFZ_L register . . . 119
Table 284. MAG_OFFZ_L register description . . . 119
Table 285. MAG_OFFZ_H register . . . 119
Table 286. MAG_OFFZ_H register description . . . 119
Table 287. MAG_SI_XX_L register . . . 119
Table 288. MAG_SI_XX_L register description . . . 119
Table 289. MAG_SI_XX_H register . . . 119
Table 290. MAG_SI_XX_H register description . . . 119
Table 291. MAG_SI_XY_L register . . . 120
Table 292. MAG_SI_XY_L register description . . . 120
Table 293. MAG_SI_XY_H register . . . 120
Table 294. MAG_SI_XY_H register description . . . 120
Table 295. MAG_SI_XZ_L register . . . 120
Table 296. MAG_SI_XZ_L register description . . . 120
Table 297. MAG_SI_XZ_H register . . . 120
Table 298. MAG_SI_XZ_H register description . . . 120
Table 299. MAG_SI_YY_L register . . . 121
Table 300. MAG_SI_YY_L register description . . . 121
Table 301. MAG_SI_YY_H register . . . 121
Table 302. MAG_SI_YY_H register description . . . 121
Table 303. MAG_SI_YZ_L register . . . 121
Table 304. MAG_SI_YZ_L register description . . . 121
Table 305. MAG_SI_YZ_H register . . . 121
Table 306. MAG_SI_YZ_H register description . . . 121
Table 307. MAG_SI_ZZ_L register . . . 122
Table 308. MAG_SI_ZZ_L register description . . . 122
Table 309. MAG_SI_ZZ_H register . . . 122
Table 310. MAG_SI_ZZ_H register description . . . 122
Table 311. MAG_CFG_A register . . . 123
Table 312. MAG_CFG_A description . . . 123
Table 313. MAG_CFG_B register . . . 123
Table 314. MAG_CFG_B description . . . 123
Table 315. FSM_LC_TIMEOUT_L register . . . 124
Table 316. FSM_LC_TIMEOUT_L register description . . . 124
Table 317. FSM_LC_TIMEOUT_H register . . . 124
Table 318. FSM_LC_TIMEOUT_H register description . . . 124
Table 319. FSM_PROGRAMS register . . . 124
Table 320. FSM_PROGRAMS register description . . . 124
Table 321. FSM_START_ADD_L register . . . 125
Table 322. FSM_START_ADD_L register description . . . 125

chanhvinh.com



Table 323. FSM_START_ADD_H register . . . 125
Table 324. FSM_START_ADD_H register description . . . 125
Table 325. PEDO_CMD_REG register . . . 125
Table 326. PEDO_CMD_REG register description . . . 125
Table 327. PEDO_DEB_STEPS_CONF register . . . 126
Table 328. PEDO_DEB_STEPS_CONF register description . . . 126
Table 329. PEDO_SC_DELTAT_L register . . . 126
Table 330. PEDO_SC_DELTAT_H register . . . 126
Table 331. PEDO_SC_DELTAT_H/L register description . . . 126
Table 332. MLC_MAG_SENSITIVITY_L register . . . 126
Table 333. MLC_ MAG_SENSITIVITY_L register description . . . 126
Table 334. MLC_MAG_SENSITIVITY_H register . . . 126
Table 335. MLC_ MAG_SENSITIVITY_H register description . . . 126
Table 336. Register address map - sensor hub registers . . . 127
Table 337. SENSOR_HUB_1 register . . . 128
Table 338. SENSOR_HUB_1 register description . . . 128
Table 339. SENSOR_HUB_2 register . . . 128
Table 340. SENSOR_HUB_2 register description . . . 128
Table 341. SENSOR_HUB_3 register . . . 128
Table 342. SENSOR_HUB_3 register description . . . 128
Table 343. SENSOR_HUB_4 register . . . 129
Table 344. SENSOR_HUB_4 register description . . . 129
Table 345. SENSOR_HUB_5 register . . . 129
Table 346. SENSOR_HUB_5 register description . . . 129
Table 347. SENSOR_HUB_6 register . . . 129
Table 348. SENSOR_HUB_6 register description . . . 129
Table 349. SENSOR_HUB_7 register . . . 130
Table 350. SENSOR_HUB_7 register description . . . 130
Table 351. SENSOR_HUB_8 register . . . 130
Table 352. SENSOR_HUB_8 register description . . . 130
Table 353. SENSOR_HUB_9 register . . . 130
Table 354. SENSOR_HUB_9 register description . . . 130
Table 355. SENSOR_HUB_10 register . . . 131
Table 356. SENSOR_HUB_10 register description . . . 131
Table 357. SENSOR_HUB_11 register . . . 131
Table 358. SENSOR_HUB_11 register description . . . 131
Table 359. SENSOR_HUB_12 register . . . 131
Table 360. SENSOR_HUB_12 register description . . . 131
Table 361. SENSOR_HUB_13 register . . . 132
Table 362. SENSOR_HUB_13 register description . . . 132
Table 363. SENSOR_HUB_14 register . . . 132
Table 364. SENSOR_HUB_14 register description . . . 132
Table 365. SENSOR_HUB_15 register . . . 132
Table 366. SENSOR_HUB_15 register description . . . 132
Table 367. SENSOR_HUB_16 register . . . 133
Table 368. SENSOR_HUB_16 register description . . . 133
Table 369. SENSOR_HUB_17 register . . . 133
Table 370. SENSOR_HUB_17 register description . . . 133
Table 371. SENSOR_HUB_17 register . . . 133
Table 372. SENSOR_HUB_17 register description . . . 133
Table 373. MASTER_CONFIG register . . . 134
Table 374. MASTER_CONFIG register description . . . 134
Table 375. SLV0_ADD register . . . 134
Table 376. SLV_ADD register description . . . 134

chanhvinh.com



Table 377. SLV0_SUBADD register . . . 135
Table 378. SLV0_SUBADD register description . . . 135
Table 379. SLAVE0_CONFIG register . . . 135
Table 380. SLAVE0_CONFIG register description . . . 135
Table 381. SLV1_ADD register . . . 135
Table 382. SLV1_ADD register description . . . 135
Table 383. SLV1_SUBADD register . . . 136
Table 384. SLV1_SUBADD register description . . . 136
Table 385. SLAVE1_CONFIG register . . . 136
Table 386. SLAVE1_CONFIG register description . . . 136
Table 387. SLV2_ADD register . . . 136
Table 388. SLV2_ADD register description . . . 136
Table 389. SLV2_SUBADD register . . . 137
Table 390. SLV2_SUBADD register description . . . 137
Table 391. SLAVE2_CONFIG register . . . 137
Table 392. SLAVE2_CONFIG register description . . . 137
Table 393. SLV3_ADD register . . . 137
Table 394. SLV3_ADD register description . . . 137
Table 395. SLV3_SUBADD register . . . 138
Table 396. SLV3_SUBADD register description . . . 138
Table 397. SLAVE3_CONFIG register . . . 138
Table 398. SLAVE3_CONFIG register description . . . 138
Table 399. DATAWRITE_SLV0 register . . . 138
Table 400. DATAWRITE_SLV0 register description . . . 138
Table 401. STATUS_MASTER register . . . 139
Table 402. STATUS_MASTER register description . . . 139
Table 403. Reel dimensions for carrier tape of LGA-14 package . . . 143
Table 404. Document revision history . . . 144

chanhvinh.com
chanhvinh.com
chanhvinh.com
chanhvinh.com

chanhvinh.com



# List of figures

Figure 1. Generic state machine . . . 5
Figure 2. State machine in the LSM6DSO32X . . . 5
Figure 3. Machine Learning Core in the LSM6DSO32X . . . 6
Figure 4. Pin connections . . . 7
Figure 5. LSM6DSO32X connection modes . . . 8
Figure 6. SPI slave timing diagram (in mode 3) . . . 14
Figure 7. I²C slave timing diagram . . . 15
Figure 8. Read and write protocol (in mode 3) . . . 21
Figure 9. SPI read protocol (in mode 3) . . . 22
Figure 10. Multiple byte SPI read protocol (2-byte example) (in mode 3) . . . 22
Figure 11. SPI write protocol (in mode 3) . . . 23
Figure 12. Multiple byte SPI write protocol (2-byte example) (in mode 3) . . . 23
Figure 13. SPI read protocol in 3-wire mode (in mode 3) . . . 24
Figure 14. I²C and MIPI I3C$^{SM}$ both active (INT1 pin not connected) . . . 27
Figure 15. Only MIPI I3C$^{SM}$ active (INT1 pin connected to Vdd_IO) . . . 27
Figure 16. Block diagram of filters . . . 30
Figure 17. Accelerometer chain . . . 30
Figure 18. Accelerometer composite filter . . . 31
Figure 19. Gyroscope digital chain - Mode 1 and Mode 2 . . . 32
Figure 20. LSM6DSO32X electrical connections in Mode 1 . . . 36
Figure 21. LSM6DSO32X electrical connections in Mode 2 . . . 37
Figure 22. Accelerometer block diagram . . . 58
Figure 23. LGA-14L 2.5 x 3.0 x 0.86 mm package outline and mechanical data . . . 141
Figure 24. Carrier tape information for LGA-14 package . . . 142
Figure 25. LGA-14 package orientation in carrier tape . . . 142
Figure 26. Reel information for carrier tape of LGA-14 package . . . 143

chanhvinh.com



**IMPORTANT NOTICE – PLEASE READ CAREFULLY**

STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST’s terms and conditions of sale in place at the time of order acknowledgement.

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, please 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.

© 2021 STMicroelectronics – All rights reserved

chanhvinh.com
chanhvinh.com
chanhvinh.com
chanhvinh.com
chanhvinh.com
chanhvinh.com