Triaxial Accelerometer

for Non-Safety Automotive Applications

SMA130

Robert Bosch GmbH, Reutlingen, Germany

Part No.:	0 273 141 234
Document No.:	1 279 929 756

This Technical Product Description is for customer information purposes only.

Bosch does not guarantee any parameters or information given within this document. Parameters may be subject to change at any time without notice. Please refer to the Technical Customer Documentation for specified values.

Content

1 Introduction 6

2 Technical Description 8

2.1 Working Principle of the Sensing Element (MEMS) 8

2.2 Block Diagram 8

2.3 Signal Path 9

2.4 Power Management 10

2.5 Power Modes 10

2.6 Soft Reset 14

2.7 Sensor Data 15

2.7.1 Accelerometer 15

2.7.2 Temperature Sensor 16

2.8 Fast Offset Compensation 17

2.9 Interrupt Controller 17

2.9.1 General Features 18

2.9.2 Mapping to Interrupt Pins 19

2.9.3 Electrical Behavior of Interrupt Pins 19

2.9.4 New Data Interrupt 20

2.9.5 Slope / Any-Motion Detection 20

2.9.6 Tap Detection 22

2.9.7 Orientation Recognition 24

2.9.8 Flat Detection 28

2.9.9 Low-g Interrupt 29

2.9.10 High-g Interrupt 29

2.9.11 No-Motion / Slow-Motion Detection 30

3 Application 33

3.1 Sensing Axes Orientation 33

3.2 Pin-Out 34

3.3 Dimensions and Weight 35

3.4 Marking 36

3.5 Footprint 36

3.6 SPI Connection Diagram 37

3.7 TWI Connection Diagram 38
4 Specified Parameters 39
4.1 Absolute Maximum Ratings 39
4.2 Operating Conditions 40
4.3 Accelerometer Output Signal 41
5 Communication 42
5.1 Serial Peripheral Interface (SPI) 43
5.2 Two-Wire Interface (TWI) 46
5.3 Access Restrictions (SPI and TWI) 49
5.4 Self-Test 50
6 Register Description 51
6.1 General Remarks 51
6.2 Register Map 52
6.2.1 Register 0x00 (BGW_CHIPID) 53
6.2.2 Register 0x02 (ACCD_X_LSB) 53
6.2.3 Register 0x03 (ACCD_X_MSB) 54
6.2.4 Register 0x04 (ACCD_Y_LSB) 54
6.2.5 Register 0x05 (ACCD_Y_MSB) 55
6.2.6 Register 0x06 (ACCD_Z_LSB) 55
6.2.7 Register 0x07 (ACCD_Z_MSB) 56
6.2.8 Register 0x08 (ACCD_TEMP) 56
6.2.9 Register 0x09 (INT_STATUS_0) 57
6.2.10 Register 0x0A (INT_STATUS_1) 58
6.2.11 Register 0x0B (INT_STATUS_2) 59
6.2.12 Register 0x0C (INT_STATUS_3) 60
6.2.13 Register 0x0F (PMU_RANGE) 62
6.2.14 Register 0x10 (PMU_BW) 62
6.2.15 Register 0x11 (PMU_LPW) 63
6.2.16 Register 0x12 (PMU_LOW_POWER) 64
6.2.17 Register 0x13 (ACCD_HBW) 65
6.2.18 Register 0x14 (BGW_SOFTRESET) 66
6.2.19 Register 0x16 (INT_EN_0) 67

6.2.20 Register 0x17 (INT_EN_1) 68
6.2.21 Register 0x18 (INT_EN_2) 69
6.2.22 Register 0x19 (INT_MAP_0) 70
6.2.23 Register 0x1A (INT_MAP_1) 71
6.2.24 Register 0x1B (INT_MAP_2) 72
6.2.25 Register 0x1E (INT_SRC) 73
6.2.26 Register 0x20 (INT_OUT_CTRL) 74
6.2.27 Register 0x21 (INT_RST_LATCH) 75
6.2.28 Register 0x22 (INT_0) 76
6.2.29 Register 0x23 (INT_1) 76
6.2.30 Register 0x24 (INT_2) 77
6.2.31 Register 0x25 (INT_3) 78
6.2.32 Register 0x26 (INT_4) 78
6.2.33 Register 0x27 (INT_5) 79
6.2.34 Register 0x28 (INT_6) 80
6.2.35 Register 0x29 (INT_7) 80
6.2.36 Register 0x2A (INT_8) 81
6.2.37 Register 0x2B (INT_9) 82
6.2.38 Register 0x2C (INT_A) 83
6.2.39 Register 0x2D (INT_B) 84
6.2.40 Register 0x2E (INT_C) 84
6.2.41 Register 0x2F (INT_D) 85
6.2.42 Register 0x32 (PMU_SELF_TEST) 86
6.2.43 Register 0x34 (BGW_WDT) 86
6.2.44 Register 0x36 (OFC_CTRL) 87
6.2.45 Register 0x38 (OFC_OFFSET_X) 88
6.2.46 Register 0x39 (OFC_OFFSET_Y) 88
6.2.47 Register 0x3A (OFC_OFFSET_Z) 89

7 Handling and Storage 90

7.1 Moisture Sensitivity Level (MSL) 90
7.2 Mounting Recommendations 90
7.3 Soldering Guidelines 91

7.3.1 Reflow Soldering Recommendation for Sensors in LGA Package 91
7.3.2 Classification Reflow Profiles 92
7.4 Tape on Reel 93
7.4.1 Tape on Reel Specification 93
7.4.2 Orientation within the Reel 95
7.5 Further Important Mounting and Assembly Recommendations 95
8 Test Specifications 95
8.1 Environmental Safety 95
8.2 Qualification 96
9 Legal Disclaimer 97

1 Introduction

The SMA130 is a triaxial, low-g accelerometer for non-safety related applications, e.g. for motion control in the passenger compartment. Within one package, the SMA130 offers the detection of acceleration in three perpendicular axes. The digital standard serial peripheral interface (SPI) of the SMA130 allows for bi-directional data transmission.

Basic Description

Sensor	Bosch Part Nr.	Type	Range	Resolution
SMA130	0 273 141 234	Accelerometer	±2, ±4, ±8, ±16 g	14 bit

Key Features

Triaxial accelerometer	Versatile, leading edge triaxial 14 bit accelerometer for reduced PCB space and simplified signal routing
Small package	LGA, 12 pins, footprint 2.0 x 2.0 mm², height 0.95 mm
Common voltage supplies	VDD voltage range: 1.62 ... 3.6 V
Digital interface	SPI, TWI (compatible with I^2C), 2 interrupt pins
Consumer electronics suite	MSL1, RoHS compliant, halogen-free
Operating temperature	-40 ... +85 °C
Programmable functionality	Acceleration range selectable Low-pass filter bandwidths selectable
Ultra-low power	Low current consumption, several power saving modes
On-chip interrupt controller	Motion-triggered interrupt signal generation for - high-g detection - low-g detection - no-motion / slow-motion detection - slope / any-motion detection - orientation recognition - flat detection - tap detection - new data detection
On-chip temperature sensor	Factory trimmed, 8 bit

Bosch points out that the system/product does not implement any ASIL-classified requirements (in the sense of ISO 26262). Therefore it has not been approved by Bosch for applications in which Bosch delivered system/product has an ASIL-related (above QM) role. This implies the following limitations:

- The SMA130 must not be used if it influences safety goals with ratings higher than ASIL QM. Safety goals are defined in the overall system (i.e., on item level).
- Bosch cannot provide any quantitative failure analysis (e.g. FTA or FMEDA) for the SMA130.
- The SMA130 does not provide a CRC to check communication errors within a SPI/I2C frame.
- The SMA130 does not provide error flags to detect malfunctions of the ASIC.

2 Technical Description

2.1 Working Principle of the Sensing Element (MEMS)

The accelerometer SMA130 consists of an evaluation circuitry (ASIC) and a micro-mechanical sensing element (MEMS) within a standard LGA package.

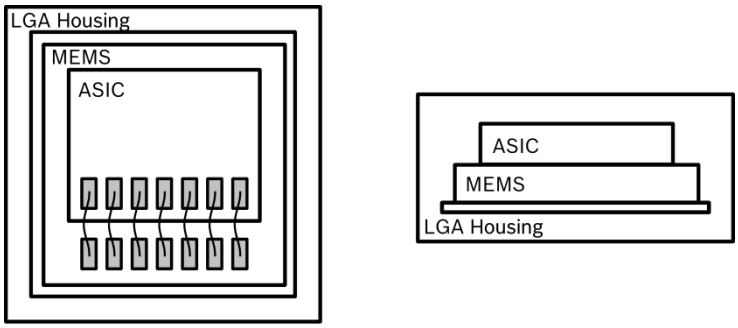


Figure 2-1: Schematics of the SMA130 mechanical design (left: top view; right: side view). The SMA130 consists of a readout ASIC stacked on top of its sensitive MEMS element.

2.2 Block Diagram

Figure 2-2 shows the basic building blocks of the SMA130. An acceleration signal along the sensitive axis of the MEMS element causes a change of the capacitances of the MEMS element. This change is converted into a digital serial bit stream which is further processed and which can be accessed via SPI.

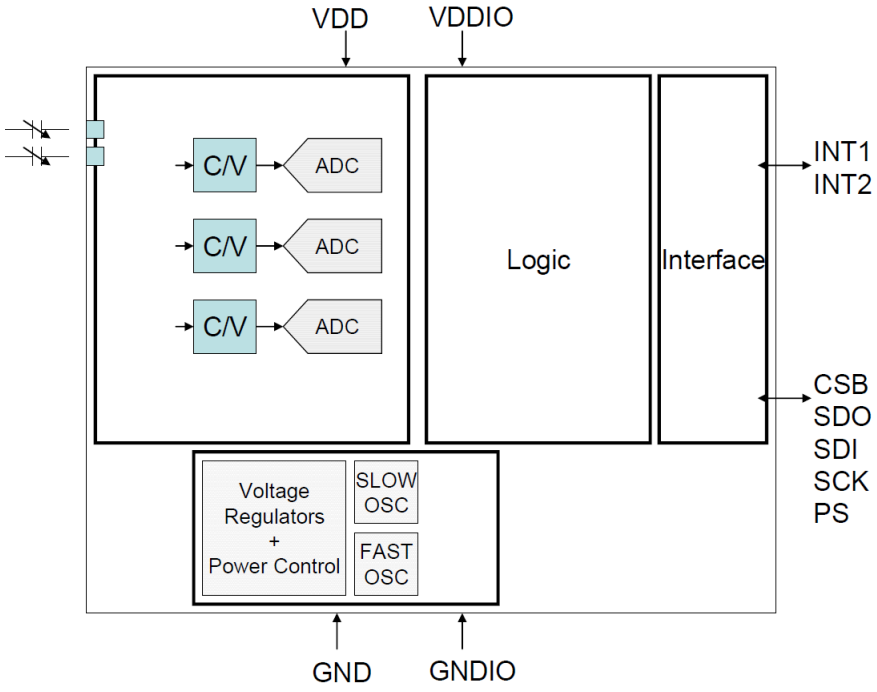


Figure 2-2: Simplified block diagram of the SMA130.

2.3 Signal Path

The SMA130 offers temperature and acceleration data for all three spatial dimensions. For the latter, the differential capacitance change (C) of the corresponding sensing element is detected. This signal corresponds to the voltage (V) entering the hybrid algorithmic analog-digital-converter (ADC), translating the formerly analog signal into a digital serial bit stream at a rate of 400 kHz. Then, the detected signal is translated into a data word of 14 bit and enters the digital signal processor (DSP).

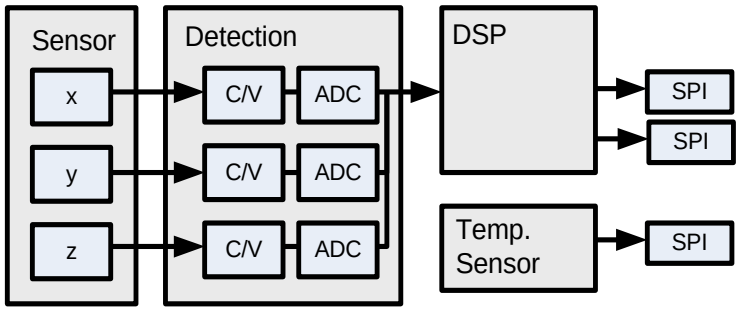


Figure 2-3: Simplified signal path of the SMA130.

Within the DSP (see Figure 2-4), the data is corrected for the analog-digital conversion, gained and offset corrected. A low-pass filter engine provides an adjustable data bandwidth. Here, the sampling rate is directly connected with the selected bandwidth.

The low-pass engine can be bypassed so that unfiltered data is accessible.

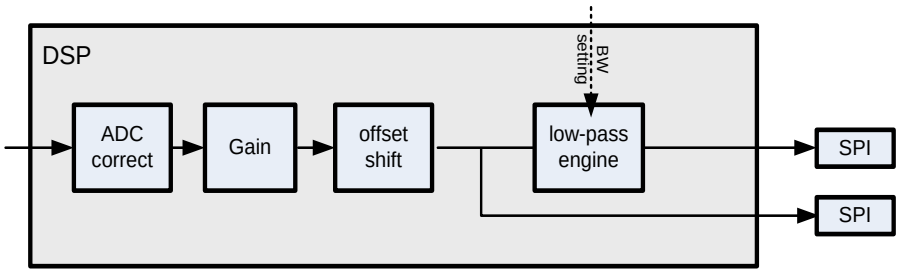


Figure 2-4: Simplified DSP element.

2.4 Power Management

The SMA130 has two distinct power supply pins:

- VDD is the main power supply for the internal blocks.
- VDDIO is a separate power supply pin mainly used for the supply of the interface.

There are no limitations regarding the voltage levels of both pins relative to each other as long as each of them is within the specified operating range. Furthermore, the device can be completely switched off (VDD = 0 V) while keeping the VDDIO supply on (VDDIO > 0 V) or vice versa.

In the case that the VDDIO supply is switched off, all interface pins (CSB, SDI, SCK, PS) must be kept close to GNDIO potential.

The SMA130 provides a **power-on reset (POR)** generator. It resets the logic part and the register values after powering on VDD and VDDIO. After POR, all settings are reset to the default values. All application specific settings which are not equal to the default settings have to be reset to their designated values after POR.

There are no constraints on the switching sequence of VDD and VDDIO. In the case that the TWI interface be used, a direct electrical connection between VDDIO and the PS pin is needed in order to ensure reliable protocol selection. For SPI mode, the PS pin must be directly connected to GNDIO.

2.5 Power Modes

The SMA130 offers six different power modes. Besides the normal mode, which represents the fully operational state of the SMA130, there are five power saving modes for power-critical applications: standby mode, suspend mode, deep suspend mode, low power mode 1 and low power mode 2.

The possible transitions between the different power modes are shown in Figure 2-5.

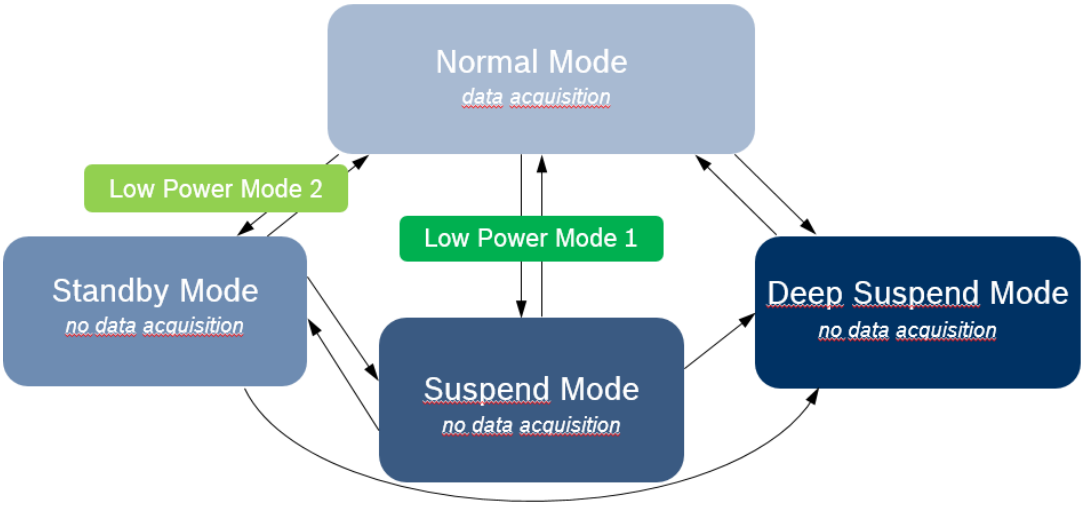


Figure 2-5: SMA130 power modes.

After power-up, the SMA130 is in **normal mode**. Here, all parts of the device are held powered up and data acquisition is performed continuously.

In **standby mode**, the analog part of the SMA130 is powered down while the digital part remains largely operational. Here, no data acquisition is performed. While in standby mode, the latest acceleration data and the content of all configuration registers are kept. Writing to and reading from registers is supported without any restrictions.

The standby mode is entered (left) by writing 1 to bit 6 (*lowpower_mode*) in register 0x12 (PMU_LOW_POWER) and writing 1 (0) to bit 7 (*suspend*) in register 0x11 (PMU_LPW). From the standby mode, it is also possible to enter the normal mode by performing a soft reset (see Section 2.6).

In **suspend mode**, the whole analog part of the SMA130 is powered down. Here, no data acquisition is performed. While in suspend mode, the latest acceleration data and the content of all configuration registers are kept. Writing to and reading from registers is supported. However, an idle time of at least 450 µs is required between two consecutive write cycles (see Section 5.3).

The suspend mode is entered (left) by writing 0 to bit 6 (*lowpower_mode*) in register 0x12 (PMU_LOW_POWER) and writing 1 (0) to bit 7 (*suspend*) in register 0x11 (PMU_LPW). From the suspend mode, it is also possible to enter the normal mode by performing a soft reset (see Section 2.6).

In **deep suspend mode**, the SMA130 reaches the lowest possible power consumption. Only the interface section is kept alive. Here, no data acquisition is performed. The content of all configuration registers is lost. All application specific settings which are not equal to the default settings must be reset to their designated values after leaving the deep suspend mode.

The TWI watchdog timer remains functional. The bit (*deep_suspend*) in register 0x11 (PMU_LPW) and the bits 2 (*i2c_wdt_en*) and 1 (*i2c_wdt_sel*) in register 0x34 (BGW_SPI3_WDT) also remain functional in deep suspend mode. The interrupt level and driver configuration bits 0 (*int1_lvl*), 1 (*int1_od*), 2 (*int2_lvl*) and 3 (*int2_od*) in register 0x20 (INT_OUT_CTRL) are still accessible.

The deep suspend mode is entered (left) by writing 1 (0) to bit 5 (*deep_suspend*) in register 0x11 (PMU_LPW) while bit 7 (*suspend*) is set to 0. From the deep suspend mode, it is also possible to enter the normal mode by performing a soft reset (see Section 2.6).

In **low power mode 1**, the SMA130 is periodically switching between a power saving sleep phase and a wake-up phase, in which data acquisition takes place. The wake-up phase corresponds to operation in normal mode with the complete circuitry powered up. The sleep phase corresponds to operation in suspend mode. Read access to registers is possible. However, unless the register access is synchronized with the wake-up phase, the restrictions of the suspend mode apply.

The low power mode 1 is entered (left) by writing 0 to bit 6 (*lowpower_mode*) in register 0x12 (PMU_LOW_POWER) and writing 1 (0) to bit 6 (*lowpower_en*) in register 0x11 (PMU_LPW).

In **low power mode 2**, the SMA130 is also periodically switching between a power saving sleep phase and a wake-up phase, in which data acquisition takes place. Low power mode 2 is similar to low power mode 1, but with a higher power consumption. The wake-up phase of low power mode 2 corresponds to operation in normal mode with the complete circuitry powered up. The sleep phase corresponds to operation in standby mode. Read access to registers is possible without restrictions.

The low power mode 2 is entered (left) by writing 1 to bit 6 (*lowpower_mode*) in register 0x12 (PMU_LOW_POWER) and writing 1 (0) to bit 6 (*lowpower_en*) in register 0x11 (PMU_LPW).

An **overview** of the register configurations of 0x11 (PMU_LPW) and 0x12 (PMU_LOW_POWER) for entering normal mode, standby mode, suspend mode, deep suspend mode, low power mode 1 and low power mode 2 is given in the following table.

Mode	**0x11 (PMU_LPW)**			**0x12 (PMU_LOW_POWER)**
	7 *suspend*	6 *lowpower_en*	5 *deep_suspend*	6 *lowpower_mode*
Normal	0	0	0	0
Standby	1	0	0	1
Suspend	1	0	0	0
Deep suspend	0	0	1	0
Low power mode 1	0	1	0	0
Low power mode 2	0	1	0	1

The **timing behavior** of low power mode 1 and 2 is event-driven. The duration of the wake-up phase depends on the number of samples which are required by the enabled interrupt engines. If an interrupt is detected, the SMA130 stays in the wake-up phase as long as the interrupt conditions endure (non-latched interrupt), until the latch time expires (temporary interrupt) or until the interrupt is reset (latched interrupt). If no interrupt is detected, the SMA130 enters the sleep phase immediately after the required number of acceleration samples have been taken and an active interface access cycle has ended. The timing diagram of low power mode 1 and 2 is shown in Figure 2-6.

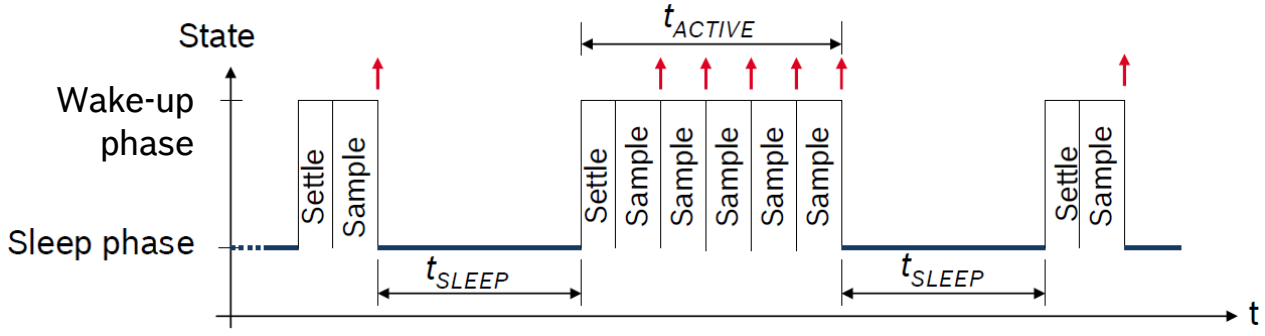


Figure 2-6: Timing diagram for low power mode 1 and 2.

The **sleep time** for low power mode 1 and 2 is set by bits <4:1> (*sleep_dur*) in register 0x11 (PMU_LPW) according to the following table.

sleep_dur	Sleep Phase Duration (t_{sleep})	*sleep_dur*	Sleep Phase Duration (t_{sleep})
0000	0.5 ms	1000	4 ms
0001	0.5 ms	1001	6 ms
0010	0.5 ms	1010	10 ms
0011	0.5 ms	1011	25 ms
0100	0.5 ms	1100	50 ms
0101	0.5 ms	1101	100 ms
0110	1 ms	1110	500 ms
0111	2 ms	1111	1 s

The **current consumption** of the SMA130 in low power mode 1 (I_{DDlp1}) and low power mode 2 (I_{DDlp2}) can be estimated with the following equations:

$$I_{DDlp1} \approx \frac{t_{sleep} \cdot I_{DDsum} + t_{active} \cdot I_{DD}}{t_{sleep} + t_{active}}$$

$$I_{DDlp2} \approx \frac{t_{sleep} \cdot I_{DDsbm} + t_{active} \cdot I_{DD}}{t_{sleep} + t_{active}}$$

For the estimation of the duration of the wake-up phase t_{active}, the corresponding wake-up time $t_{w,up1}$ or $t_{w,up2}$ and the filter update time t_{ut} have to be considered.

If the bandwidth is ≥ 31.25 Hz, t_{active} is given by:

$$t_{active} = t_{ut} + t_{w,up1/2} - 0.9\,\text{ms}$$

Else, t_{active} is given by:

$$t_{active} = 4 \cdot t_{ut} + t_{w,up1/2} - 0.9\,\text{ms}$$

The filter update times t_{ut} are given by the following table.

bw	Bandwidth	Update Time (t_{ut})
00xxx	*)	-
01000	7.81 Hz	64 ms
01001	15.63 Hz	32 ms
01010	31.25 Hz	16 ms
01011	62.5 Hz	8 ms
01100	125 Hz	4 ms
01101	250 Hz	2 ms
01110	500 Hz	1 ms
01111	unfiltered	0.5 ms
1xxxx	*)	-

*) The *bw* settings 00xxx and 1xxxx are both reserved. *bw* = 00xxx results in a bandwidth of 7.81 Hz, *bw* = 1xxxx results in an unfiltered signal. It is recommended to actively set an appropriate, application specific bandwidth and to use the *bw* range from 01000 to 01111.

During the wake-up phase, all analog modules are held powered up, while during the sleep phase, most analog modules are powered down. Consequently, a wake-up time of at least $t_{w,up1}$ or $t_{w,up2}$ is needed to settle the analog modules so that reliable acceleration data are generated.

2.6 Soft Reset

A soft reset causes all user configuration settings to be overwritten with their default values and the sensor to enter normal mode.

A soft reset is initiated by writing the value 0xB6 to register 0x14 (BGW_SOFTRESET).

2.7 Sensor Data

2.7.1 Accelerometer

The data representation of the SMA130 follows two's complement representation.

For each axis, the 14 bits of acceleration data are split into a MSB upper part (one byte containing bits <13:6> of acceleration data) and a LSB lower part (one byte containing bits <5:0> of acceleration data, one *undefined* bit with random data which is to be ignored and a *new_data* flag). Registers 0x02 (ACCD_X_LSB) and 0x03 (ACCD_X_MSB) contain the acceleration data for the x-channel, registers 0x04 (ACCD_Y_LSB) and 0x05 (ACCD_Y_MSB) for the y-channel and 0x06 (ACCD_Z_LSB) and 0x07 (ACCD_Z_MSB) for the z-channel. It is recommended to always start reading out the acceleration data registers with the LSB part first.

In order to ensure data integrity, a **shadowing procedure** can be enabled. In this case, the content of the MSB register is locked by reading the corresponding LSB register until the MSB register is read as well. This means that the MSB register always has to be read in order to remove the data lock. Shadowing can be disabled (enabled) by writing 1 (0) to bit 6 (*shadow_dis*) in the register 0x13 (ACCD_HBW). For disabled shadowing, the content of both the MSB and the LSB register is updated by new values immediately. Unused bits of the LSB registers may have any value and should be ignored.

New data can be identified by bit 0 (*new_data* flag) of each LSB register. It is set if the data registers have been updated and reset if either the corresponding MSB or LSB part is read.

Two different streams of acceleration data are available, **unfiltered and filtered** data. The unfiltered data is sampled with 2 kHz. The sampling rate (output data rate ODR) of the filtered data depends on the selected filter bandwidth (BW) and is always twice the selected bandwidth (BW = ODR/2). Which kind of data is stored in the acceleration data registers depends on bit 7 (*data_high_bw*) in register 0x13 (ACCD_HBW). If bit 7 is 0 (1), filtered (unfiltered) data is stored in the registers. Both data streams are offset-compensated.

The **bandwidth** of filtered acceleration data is determined by setting bits <4:0> (*bw*) in register 0x10 (PMU_BW) as shown in the following table.

bw	Bandwidth	Update Time
00xxx	*)	-
01000	7.81 Hz	64 ms
01001	15.63 Hz	32 ms
01010	31.25 Hz	16 ms
01011	62.5 Hz	8 ms
01100	125 Hz	4 ms
01101	250 Hz	2 ms
01110	500 Hz	1 ms
01111	unfiltered	0.5 ms
1xxxx	*)	-

*) The *bw* settings 00xxx and 1xxxx are both reserved. *bw* = 00xxx results in a bandwidth of 7.81 Hz, *bw* = 1xxxx results in an unfiltered signal. It is recommended to actively set an appropriate, application specific bandwidth and to use the *bw* range from 01000 to 01111.

The acceleration measurement **range** can be selected via bits <3:0> (*range*) in register 0x0F (PMU_RANGE) according to the table below.

range	Acceleration Measurement Range	Resolution
0011	±2 g	4096 LSB/g
0101	±4 g	2048 LSB/g
1000	±8 g	1024 LSB/g
1100	±16 g	512 LSB/g
others	reserved	-

2.7.2 Temperature Sensor

The temperature sensor data of the SMA130 are given in two's complement representation with a data width of 8 bits, which covers a temperature range of 128 K. Temperature data can be read from register 0x08 (ACCD_TEMP). The slope is typically 0.5 K/LSB.

2.8 Fast Offset Compensation

The fast offset compensation is a one-shot process by which the compensation value is set in such a way that when added to the raw acceleration, the resulting acceleration value of each selected axis approaches the target value of 0 g. For fast offset compensation, the g-range of the SMA130 has to be set to 2 g.

During the compensation process, an average of 16 consecutive acceleration values is computed and the difference between the target value and the computed value is written to bits <7:0> (*offset_x/y/z*) in registers 0x38 (OFC_OFFSET_X), 0x39 (OFC_OFFSET_Y), 0x3A (OFC_OFFSET_Z). These public registers are updated with the contents of the internal registers and can be read and over-written by the user.

The fast offset compensation is triggered for each axis individually by setting bits <6:5> (*cal_trigger*) in register 0x36 (OFC_CTRL) as shown in the following table.

cal_trigger	Selected Axis
00	none
01	x
10	y
11	z

Bits *cal_trigger* are write-only. Once triggered, the status of the fast offset compensation process is reflected in the status bit 4 (*cal_rdy*) in register 0x36 (OFC_CTRL). *cal_rdy* is 0 while the correction is in progress. Otherwise, it is 1. *cal_rdy* is 0 when *cal_trigger* is not 00.

The content of the fast offset compensation registers is reset to zero by writing 1 to bit 7 (*offset_reset*) in register 0x36 (OFC_CTRL).

The fast offset compensation should not be used during operation in any of the low power modes. In the low power modes, the availability of necessary data for a proper function of the fast offset compensation is not fulfilled.

2.9 Interrupt Controller

The SMA130 features eight programmable interrupt engines. Each interrupt can be independently enabled and configured. If the trigger condition of an enabled interrupt is fulfilled, the corresponding status bit is set to 1 and the selected interrupt pin is activated. The SMA130 provides two interrupt pins, INT1 and INT2. Interrupts can be freely mapped to any of these pins. The state of a specific interrupt pin is derived from a logic “or” combination of all interrupts mapped to it.

The interrupt status registers are updated when a new data word is written into the acceleration data registers. If an interrupt is disabled, all active status bits associated with it are immediately reset.

2.9.1 General Features

An interrupt is cleared depending on the selected interrupt mode, which is common to all interrupts. There are three different interrupt modes: non-latched, latched, and temporary. The behavior of the different interrupt modes is shown in Figure 2-7.

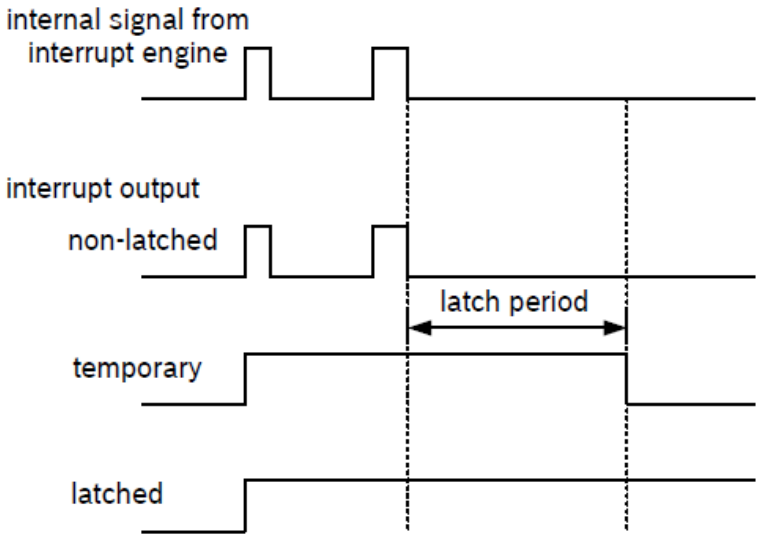


Figure 2-7: Interrupt modes.

The mode is selected by the bits <3:0> (*latch_int*) in register 0x21 (INT_RST_LATCH) according to the following table.

latch_int	Interrupt Mode	*latch_int*	Interrupt Mode
0000	non-latched	1000	non-latched
0001	temporary, 250 ms	1001	temporary, 250 µs
0010	temporary, 500 ms	1010	temporary, 500 µs
0011	temporary, 1 s	1011	temporary, 1 ms
0100	temporary, 2 s	1100	temporary, 12.5 ms
0101	temporary, 4 s	1101	temporary, 25 ms
0110	temporary, 8 s	1110	temporary, 50 ms
0111	latched	1111	latched

An interrupt is generated if its activation condition is met. It cannot be cleared as long as the activation condition is fulfilled.

In the **non-latched** mode, the interrupt status bit and the selected pin (i.e., the contribution to the “or” condition for INT1 and/or INT2) are cleared as soon as the activation condition is no more valid. Exceptions to this behavior are the new data, orientation and flat interrupts, which are automatically reset after a fixed time.

In the **latched mode**, an asserted interrupt status and the selected pin are cleared by writing 1 to bit 7 (*reset_int*) in register 0x21 (INT_RST_LATCH). If the activation condition is still valid when the bit is cleared, the interrupt status is asserted again with the next change of the acceleration registers.

In the **temporary mode**, an asserted interrupt status and the selected pin are cleared after a defined period of time (selectable between 250 µs and 8 s).

Several interrupt engines can use either unfiltered or filtered acceleration data as their input. For these interrupts, the source can be selected with the bits <5:0> in register 0x1E (INT_SRC): *int_src_data*, *int_src_tap*, *int_src_slo_no_mot*, *int_src_slope*, *int_src_high* and *int_src_low*. Setting the respective bits to 0 (1) selects filtered (unfiltered) data as input. The orientation recognition and flat detection interrupts always use filtered input data.

It is strongly recommended to set interrupt parameters prior to enabling the interrupt. Changing parameters of an already enabled interrupt may cause unwanted interrupt generation and generation of a false interrupt history. A safe way to change parameters of an enabled interrupt is to keep the following sequence:

- disable the desired interrupt
- change parameters
- wait for at least 10 ms
- re-enable the desired interrupt

2.9.2 Mapping to Interrupt Pins

Registers 0x19 (INT_MAP_0) to 0x1B (INT_MAP_2) are dedicated to the mapping of interrupts to the interrupt pins INT1 or INT2. Setting the respective bit (*int1_"inttype"*) in register 0x19 (INT_MAP_0) to 1 (0) maps (unmaps) *"inttype"* to pin INT1. Correspondingly, setting the respective bit (*int2_"inttype"*) in register 0x1B (INT_MAP_2) to 1 (0) maps (unmaps) *"inttype"* to pin INT2.

Note: *"inttype"* is to be replaced with the precise notation, given in the register map in Section 6.2. For example, setting bit 7 (*int1_flat*) in register 0x19 (INT_MAP_0) to 1 maps *int1_flat* to pin INT1.

2.9.3 Electrical Behavior of Interrupt Pins

Both interrupt pins can be configured to show the desired electrical behavior. The "active" level of each interrupt pin is determined by bits 0 (*int1_lvl*) and 2 (*int2_lvl*) in register 0x20 (INT_OUT_CTRL). If *int1/2_lvl* is 1 (0), then pin INT1/INT2 is active 1 (0).

The characteristics of the output driver of the interrupt pins may be configured with bits 1 (*int1_od*) and 3 (*int2_od*) in register 0x20 (INT_OUT_CTRL). By setting *int1_od* or *int2_od* to 1, the output driver shows open-drive characteristics. By setting the bits to 0, the output driver shows push-pull characteristics. When open-drive characteristics are selected in the design, external pull-up or pull-down resistors should be applied according to the *int1/2_lvl* configuration.

2.9.4 New Data Interrupt

The new data interrupt serves for synchronous reading of acceleration data. It is generated after storing a new value of z-axis acceleration data in the data register. The interrupt is cleared automatically when the next data acquisition cycle starts. The interrupt status is 0 for at least 50 µs.

The interrupt mode of the new data interrupt is fixed to non-latched.

The new data interrupt is enabled (disabled) by writing 1 (0) to bit 4 (*data_en*) in register 0x17 (INT_EN_1). The interrupt status is stored in bit 7 (*data_int*) in register 0x0A (INT_STATUS_1).

Due to the filter settling time, the first interrupt after wake-up from suspend or standby mode will take longer than the update time.

2.9.5 Slope / Any-Motion Detection

The slope / any-motion detection interrupt uses the slope between successive acceleration signals to detect changes in motion. An interrupt is generated when the slope (i.e., the absolute value of the acceleration difference) exceeds a configurable, preset threshold. It is cleared as soon as the slope falls below the threshold. The principle of the slope / any-motion interrupt is shown in Figure 2-8.

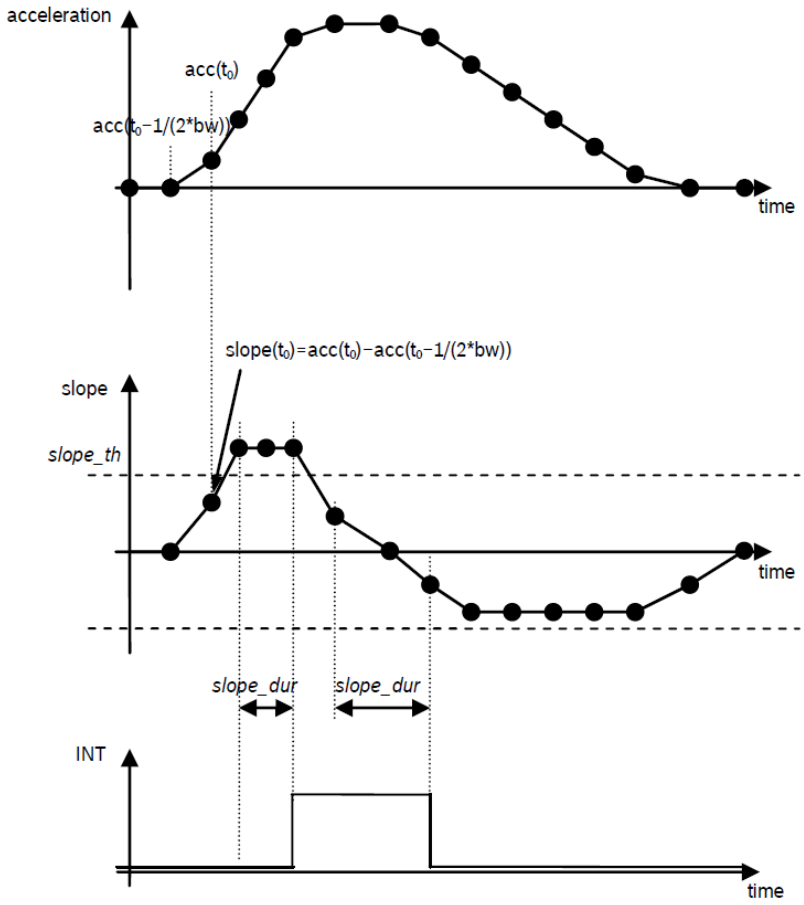


Figure 2-8: Principle of the slope / any-motion detection.

The threshold is defined through register 0x28 (INT_6), bits <7:0> (*slope_th*). The scaling of 1 LSB of *slope_th* for the different g-ranges is given in the following table.

g-Range	Scaling of 1 LSB of *slope_th*
2 g	3.91 mg
4 g	7.81 mg
8 g	15.6 mg
16 g	31.3 mg

The time difference between the successive acceleration signals depends on the selected bandwidth and equates to

$$\Delta t = \frac{1}{2 \cdot bandwidth} .$$

In order to suppress false triggers, the interrupt is only generated (cleared) if a certain number *N* of consecutive slope data points is larger (smaller) than the slope threshold given by *slope_th*. This number is set by the bits <1:0> (*slope_dur*) in register 0x27 (INT_5) according to the following equation.

$$N = slope\_dur + 1$$

2.9.5.1 Enabling (Disabling) for Each Axis

Slope / any-motion detection can be enabled (disabled) for each axis separately by writing 1 (0) to bits 0 (*slope_en_x*), 1 (*slope_en_y*) or 2 (*slope_en_z*) in register 0x16 (INT_EN_0). The interrupt is generated if the slope of any of the enabled axes exceeds the threshold *slope_th* for *N* consecutive times. As soon as the slopes of all enabled axes fall or stay below this threshold for *N* consecutive times, the interrupt is cleared (unless the interrupt signal is latched).

2.9.5.2 Axis and Sign Information

The interrupt status is stored in bit 2 (*slope_int*) in register 0x09 (INT_STATUS_0). The slope / any-motion interrupt provides additional information about the detected slope. The axis which triggered the interrupt is given by bit 0 (*slope_first_x*), 1 (*slope_first_y*) or 2 (*slope_first_z*) in register 0x0B (INT_STATUS_2) which is set to 1. The sign of the triggering slope is kept in bit (*slope_sign*) in register 0x0B until the interrupt is retriggered. If *slope_sign* is 0 (1), the sign is positive (negative).

2.9.6 Tap Detection

The tap interrupt is generated if a pre-defined slope of the acceleration signal of at least one axis is exceeded. Two different tap events are distinguished: A **single tap** is a single event within a certain time, followed by a certain quiet time. A **double tap** consists of a first event followed by a second event within a defined time frame.

The single tap interrupt is enabled (disabled) by writing 1 (0) to bit 5 (*s_tap_en*) in register 0x16 (INT_EN_0). The double tap interrupt is enabled (disabled) by writing 1 (0) to bit 4 (*d_tap_en*) in register 0x16. When the temporary latched mode is enabled, it is not recommended to simultaneously enable the single tap and the double tap interrupt.

The status of the single tap interrupt is stored in bit 5 (*s_tap_int*) in register 0x09 (INT_STATUS_0), the status of the double tap interrupt is stored in bit 4 (*d_tap_int*) in register 0x09.

The slope threshold for detecting a tap event is set by bits <4:0> (*tap_th*) in register 0x2B (INT_9).The scaling of 1 LSB of *tap_th* for the different g-ranges is given in the following table.

g-Range	Scaling of 1 LSB of *tap_th*
2 g	62.5 mg
4 g	125 mg
8 g	250 mg
16 g	500 mg

An overview of the different timing parameters of the tap sensing is given in Figure 2-9.

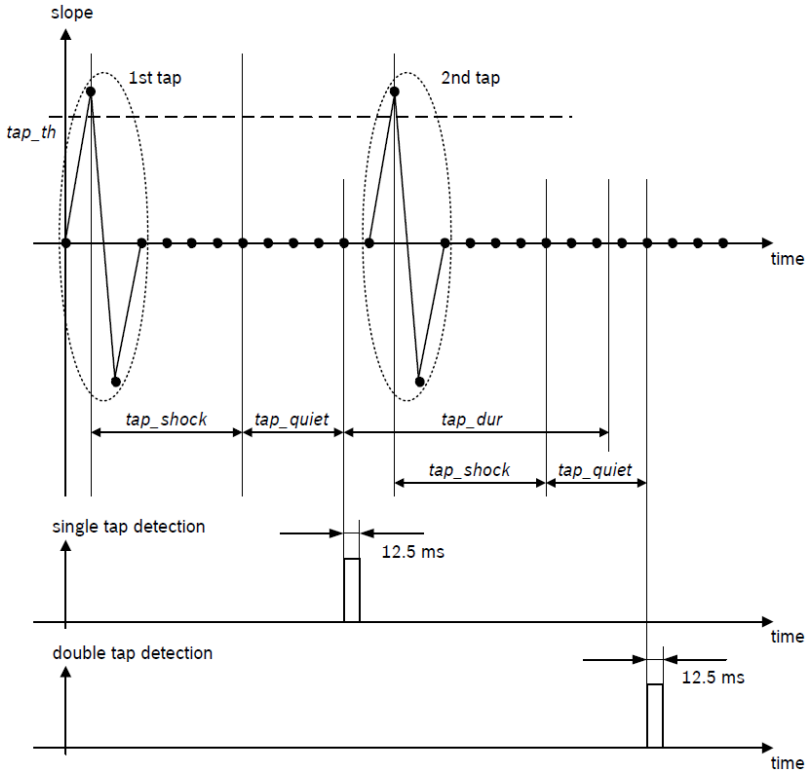


Figure 2-9: Tap detection timing.

The parameters *tap_shock*, defined by bit 6 in register 0x2A (INT_8), and *tap_quiet*, defined by bit 7 in register 0x2A, apply to both single tap and double tap detection. *tap_dur*, defined by bits <2:0> in register 0x2A, applies to double tap detection only. Within the duration of *tap_shock*, any slope exceeding *tap_th* after the first event is ignored. Contrary to this, within the duration of *tap_quiet*, no slope exceeding *tap_th* must occur, otherwise, the first event will be cancelled.

2.9.6.1 Single Tap Detection

A single tap interrupt is generated after the sum of the durations of *tap_shock* and *tap_quiet* if the corresponding slope conditions are fulfilled. The interrupt is automatically cleared after a delay of 12.5 ms.

If the single-tap interrupt is not in use, it is not recommended to map it to any of the INT pins.

2.9.6.2 Double Tap Detection

A double tap interrupt is generated if an event fulfilling the conditions for a single tap within the duration *tap_dur* after the completion of the first tap event. The interrupt is automatically cleared after a delay of 12.5 ms.

2.9.6.3 Timing Selection

The parameters *tap_shock* and *tap_quiet* can be set to two distinct values each. By writing 0 (1) to *tap_shock*, the duration of *tap_shock* is set to 50 ms (75 ms). By writing 0 (1) to *tap_quiet*, the duration of *tap_quiet* is set to 30 ms (20 ms).

The duration of *tap_dur* can be selected according to the following table.

tap_dur	Duration of *tap_dur*
000	50 ms
001	100 ms
010	150 ms
011	200 ms
100	250 ms
101	375 ms
110	500 ms
111	700 ms

2.9.6.4 Axis and Sign Information

The sign of the slope of the first tap which triggered the interrupt is stored in bit 7 (*tap_sign*) in register 0x0B (INT_STATUS_2). 0 indicates a positive sign, 1 indicates a negative sign. The value of the bit persists after clearing the interrupt.

The axis which triggered the interrupt is indicated by bits 4 (*tap_first_x*), 5 (*tap_first_y*) and 6 (*tap_first_z*) in register 0x0B. The bit which corresponds to the triggering axis is set to 1 while the other bits remain 0. These bits are cleared when the interrupt status is cleared.

2.9.6.5 Tap Detection in Low Power Mode

In low power mode 1 or 2, a limited number of samples is processed after wake-up to decide whether an interrupt condition is fulfilled. The number of samples is selected by bits <7:6> (*tap_samp*) in register 0x2B (INT_9) according to the following table.

tap_samp	Number of Samples
00	2
01	4
10	8
11	16

2.9.7 Orientation Recognition

The orientation recognition interrupt gives information on an orientation change of the SMA130 with respect to the gravitational field vector “g”. The measured acceleration vector components with respect to the gravitational field are defined in Figure 2-10.

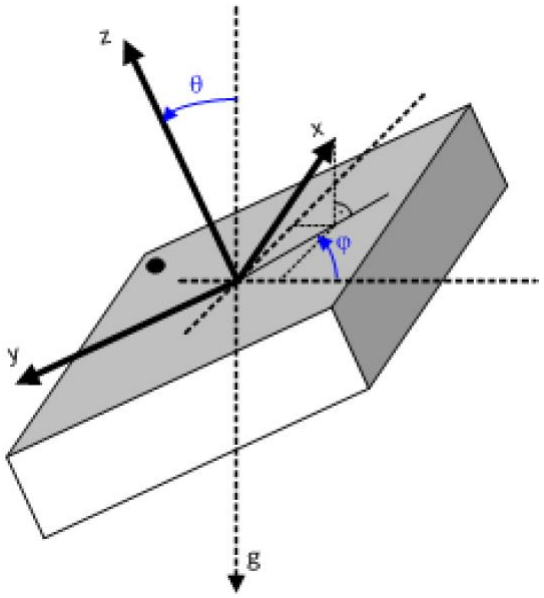


Figure 2-10: Definition of the vector components.

The calculation of the magnitudes of the different acceleration vectors is given by the following equations.

$$acc_x = 1\,\mathrm{g} \cdot \sin\theta \cdot \cos\varphi$$

$$acc_y = -1\,\mathrm{g} \cdot \sin\theta \cdot \sin\varphi$$

$$acc_z = 1\,\mathrm{g} \cdot \cos\theta$$

$$\frac{acc_y}{acc_x} = -\tan\varphi$$

Depending on the magnitudes of the acceleration vectors, the orientation of the SMA130 is determined and stored in bits <6:4> (*orient*) in register 0x0C (INT_STATUS_3). These bits may not be reset in the sleep phase of low power mode 1 or 2.

There are three orientation calculation modes with different thresholds for switching between different orientations: symmetrical, high-asymmetrical and low-asymmetrical mode. The mode is selected by setting the bits <1:0> (*orient_mode*) in register 0x2C (INT_A) as given in the following table.

orient_mode	Orientation Mode
00	symmetrical
01	high-asymmetrical
10	low-asymmetrical
11	symmetrical

For each orientation mode, the bits orient have a different meaning, as shown in the following tables for the symmetrical, high-asymmetrical and low-asymmetrical mode.

Symmetrical Mode

orient	Name	Angle	Condition
x00	portrait upright	315° < φ < 45°	$\lvert acc_y \rvert < \lvert acc_x \rvert - hyst$ and $acc_x - hyst \geq 0$
x01	portrait upside down	135° < φ < 225°	$\lvert acc_y \rvert < \lvert acc_x \rvert - hyst$ and $acc_x + hyst < 0$
x10	landscape left	45° < φ < 135°	$\lvert acc_y \rvert \geq \lvert acc_x \rvert + hyst$ and $acc_y < 0$
x11	landscape right	225° < φ < 315°	$\lvert acc_y \rvert \geq \lvert acc_x \rvert + hyst$ and $acc_y \geq 0$

High-Asymmetrical Mode

orient	Name	Angle	Condition
x00	portrait upright	297° < φ < 63°	$\lvert acc_y \rvert < 2 \cdot \lvert acc_x \rvert - hyst$ and $acc_x - hyst \geq 0$
x01	portrait upside down	117° < φ < 243°	$\lvert acc_y \rvert < 2 \cdot \lvert acc_x \rvert - hyst$ and $acc_x + hyst < 0$
x10	landscape left	63° < φ < 117°	$\lvert acc_y \rvert \geq 2 \cdot \lvert acc_x \rvert + hyst$ and $acc_y < 0$
x11	landscape right	243° < φ < 297°	$\lvert acc_y \rvert \geq 2 \cdot \lvert acc_x \rvert + hyst$ and $acc_y \geq 0$

Low-Asymmetrical Mode

orient	Name	Angle	Condition
x00	portrait upright	333° < φ < 27°	$\lvert acc_y \rvert < 0.5 \cdot \lvert acc_x \rvert - hyst$ and $acc_x - hyst \geq 0$
x01	portrait upside down	153° < φ < 207°	$\lvert acc_y \rvert < 0.5 \cdot \lvert acc_x \rvert - hyst$ and $acc_x + hyst < 0$
x10	landscape left	27° < φ < 153°	$\lvert acc_y \rvert \geq 0.5 \cdot \lvert acc_x \rvert + hyst$ and $acc_y < 0$
x11	landscape right	207° < φ < 333°	$\lvert acc_y \rvert \geq 0.5 \cdot \lvert acc_x \rvert + hyst$ and $acc_y \geq 0$

In these tables, the parameter *hyst* stands for a hysteresis which can be selected by setting the bits <6:4> (*orient_hyst*) in register 0x2C (INT_A). 1 LSB of *orient_hyst* always corresponds to 62.5 mg, irrespective of the g-range setting. Please note that by using a hysteresis ≠ 0, the actual switching angles become different from the angles given in the tables above since there is an overlap between the different orientations.

The most significant bit of *orient* (which is indicated by an "x" in the tables above) contains information about the direction of the z-axis. It is set to 0 (1) if $acc_z \geq 0$ ($acc_z < 0$).

The typical switching conditions between the four different orientations for the symmetrical mode without hysteresis are shown in Figure 2-11.

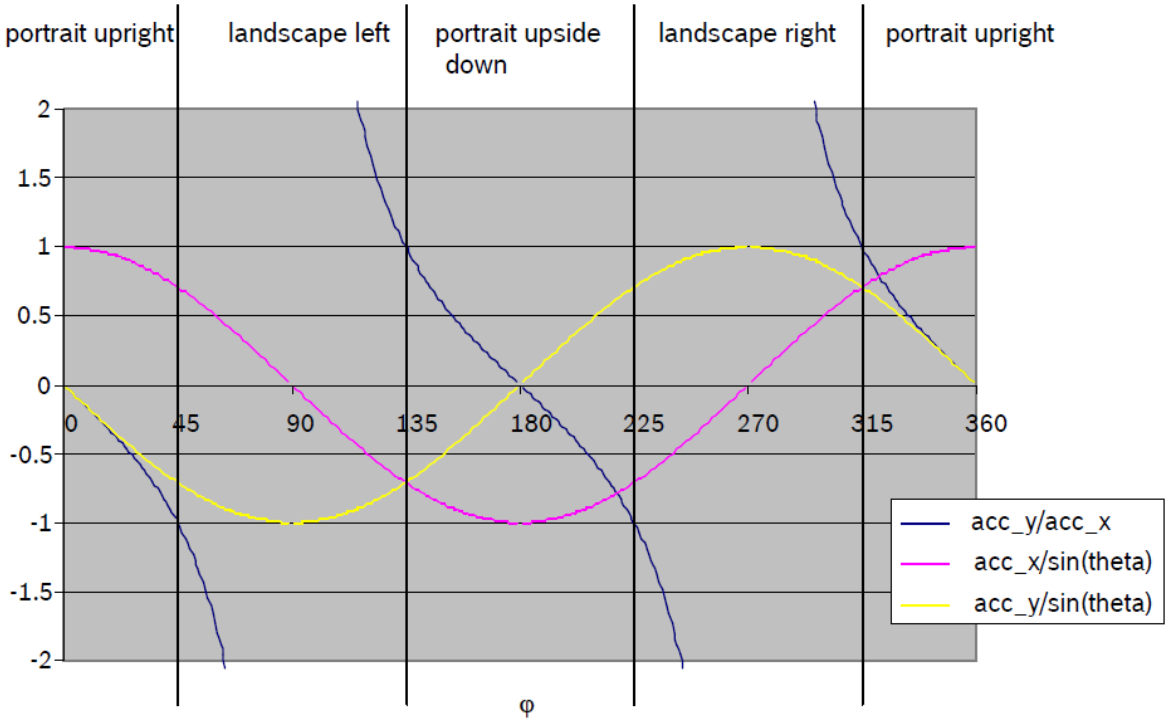


Figure 2-11: Typical orientation switching conditions without hysteresis.

The orientation interrupt is enabled (disabled) by writing 1 (0) to bit 6 (orient_en) in register 0x16 (INT_EN_0). The interrupt is generated if the value of *orient* has changed. It is automatically cleared after one stable period of the *orient* value. The interrupt status is stored in bit 6 (*orient_int*) in register 0x09 (INT_STATUS_0). *orient* always reflects the current orientation of the SMA130, irrespective of which interrupt mode has been selected. Bit 6 (*orient* <2>) reflects the device orientation with respect to the z-axis. Bits 5 and 4 (*orient* <1:0>) reflect the device orientation in the x-y-plane.

2.9.7.1 Orientation Blocking

The change of *orient* and – as a consequence – the generation of the interrupt can be blocked according to conditions selected by setting the value of the bits <3:2> (*orient_blocking*) in register 0x2C (INT_A) as shown in the following table.

orient_blocking	Conditions
00	no blocking
01	theta blocking or acceleration in any axis > 1.5 g
10	theta blocking or acceleration slope in any axis > 0.2 g or acceleration in any axis > 1.5 g
11	theta blocking or acceleration slope in any axis > 0.4 g or acceleration in any axis > 1.5 g and value of *orient* is not stable for at least 100 ms

The theta blocking is defined by the following inequality:

$$|\tan\theta| < \frac{\sqrt{blocking\_theta}}{8}$$

The parameter *blocking_theta* represents the contents of the bits <5:0> (*orient_theta*) in register 0x2D (INT_B). It is possible to define a blocking angle between 0° and 44.8°. The internal blocking algorithm saturates the acceleration values before further processing. As a consequence, the blocking angles are strictly valid only for a device at rest and can differ if the device is moving.

Example:

To get a maximum blocking angle of 19°, the parameter *blocking_theta* is determined in the following way:

$$blocking\_theta = (8 \cdot \tan(19°))^2 = 7.588$$

This means that *orient_theta* has to be chosen as 001000.

In order to avoid the unwanted generation of the orientation interrupt in a nearly flat position (z ≈ 0, sign change due to small movements or noise), a hysteresis of 0.2 g is implemented for the z-axis, i.e., after a sign change, the interrupt is only generated for |z| > 0.2 g.

2.9.7.2 Up-Down Interrupt Suppression Flag

Per default, an orientation interrupt is triggered when any of the *orient* bits change their state. The SMA130 can be configured to trigger orientation interrupts only when the device position changes in the x-y-plane while orientation changes with respect to the z-axis are ignored. A change of the orientation of the z-axis and hence a state change of bit 6 (*orient* <2>) is ignored (considered) when bit 6 (*orient_ud_en*) in register 0x2D (INT_B) is set to 0 (1).

2.9.8 Flat Detection

The flat detection provides information about the orientation of the z-axis of the SMA130 relative to the g-vector. I.e., it recognizes whether the device is in a flat position or not.

The flat angle Θ is adjustable from 0° to 44.8° via the bits <5:0> (*flat_theta*) in register 0x2E (INT_C). The flat angle can be set according to the following equation:

$$\Theta = \arctan\left(\frac{1}{8}\sqrt{flat\_theta}\right)$$

A hysteresis of the flat detection can be enabled by the bits <2:0> (*flat_hy*) in register 0x2F (INT_D). In this case, the flat position is set if the angle drops below the following threshold:

$$\Theta_{hyst,ll} = \arctan\left(\frac{1}{8}\sqrt{flat\_theta \cdot \left(1 - \frac{flat\_hy}{1024}\right) - \frac{flat\_hy}{16}}\right)$$

The flat position is reset if the angle exceeds the following threshold:

$$\Theta_{hyst,ul} = \arctan\left(\frac{1}{8}\sqrt{flat\_hy \cdot \left(1 + \frac{flat\_hy}{1024}\right) + \frac{flat\_hy}{16}}\right)$$

The flat interrupt is enabled (disabled) by writing 1 (0) to bit 7 (*flat_en*) in register 0x16 (INT_EN_0). The flat value is stored in bit 7 (*flat*) in register 0x0C (INT_STATUS_3) if the interrupt is enabled. It is 1 if the SMA130 is in the flat position; otherwise, it is 0. The flat interrupt is generated if the *flat* value has changed and the new value is stable for at least the time given by bits <5:4> (*flat_hold_time*) in register 0x2F (INT_D). The timing of *flat_hold_time* is given in the following table.

flat_hold_time	Time
00	0 ms
01	512 ms
10	1024 ms
11	2048 ms

A flat interrupt may also be generated if the flat interrupt is enabled. The actual status of the interrupt is stored in bit 7 (*flat_int*) in register 0x09 (INT_STATUS_0). The flat orientation of the SMA130 can always be determined from reading the *flat* bit after interrupt generation. If the unlatched interrupt mode is used, the value of *flat_int* and also the interrupt are automatically cleared after one sample period. If the temporary or latched interrupt mode is used, the *flat_int* value is kept fixed until the latch time expires or the interrupt is reset.

2.9.9 Low-g Interrupt

The low-g interrupt is based on the comparison of acceleration data against a low-g threshold, which is most useful for free-fall detection.

The low-g interrupt is enabled (disabled) by writing 1 (0) to bit 3 (*low_en*) in register 0x17 (INT_EN_1). There are two modes available, the single mode and the sum mode. In single mode, the acceleration of each axis is compared with the threshold. In sum mode, the sum of absolute acceleration values of all axes $|acc_x| + |acc_y| + |acc_z|$ is compared with the threshold. The mode is selected by the contents of bit 2 (*low_mode*) in register 0x24 (INT_2): 0 selects the single mode, 1 selects the sum mode.

The low-g threshold is set via bits <7:0> (*low_th*) in register 0x23 (INT_1). 1 LSB of *low_th* always corresponds to an acceleration of 7.81 mg, irrespective of the g-range setting.

A hysteresis can be selected by setting the bits <1:0> (*low_hy*) in register 0x24 (INT_2). 1 LSB of *low_hy* always corresponds to an acceleration difference of 125 mg, irrespective of the g-range setting.

The low-g interrupt is generated if the absolute values of the acceleration of all axes (“and” relation, in the case of single mode) or their sum (in the case of sum mode) are lower than the threshold for at least the time defined by the bits <7:0> (*low_dur*) in register 0x22 (INT_0). The relation between the content of *low_dur* and the actual delay of the interrupt generation is given by the following equation.

$$\text{delay [ms]} = (low\_dur + 1) \cdot 2 \text{ ms}$$

Therefore, possible delay times range from 2 ms to 512 ms.

The interrupt is reset if the absolute value of the acceleration of at least one axis (“or” relation, in the case of single mode) or the sum of absolute values (in the case of sum mode) is higher than the threshold plus the hysteresis for at least one data acquisition. The interrupt status is stored in bit 0 (*low_int*) in register 0x09 (INT_STATUS_0).

2.9.10 High-g Interrupt

The high-g interrupt is based on the comparison of acceleration data against a high-g threshold for the detection of wake-up, shock or other high-acceleration events.

The high-g interrupt is enabled (disabled) for each axis by writing 1 (0) to the respective bits 0 (*high_en_x*), 1 (*high_en_y*) or 2 (*high_en_z*) in register 0x17 (INT_EN_1). The high-g threshold is set via bits <7:0> (*high_th*) in register 0x26 (INT_4). The scaling of 1 LSB of *high_th* depends on the selected g-range and is summarized in the following table.

g-Range	Scaling of 1 LSB of *high_th*
2 g	7.81 mg
4 g	15.63 mg
8 g	31.25 mg
16 g	62.5 mg

A hysteresis can be selected by setting bits <7:6> (*high_hy*) in register 0x24 (INT_2). Analogously to *high_th*, the scaling of 1 LSB of *high_hy* depends on the selected g-range and is summarized in the following table.

g-Range	Scaling of 1 LSB of *high_hy*
2 g	125 mg
4 g	250 mg
8 g	500 mg
16 g	1000 mg

The high-g interrupt is generated if the absolute value of the acceleration of at least one of the selected axes (“or” relation) is higher than the threshold for at least the time defined by the bits <7:0> (*high_dur*) in register 0x25 (INT_3). The relation between the content of *high_dur* and the actual delay of the interrupt generation is given by the following equation.

$$\text{delay [ms]} = (\textit{high\_dur} + 1) \cdot 2 \text{ ms}$$

Thus, possible delay times range from 2 ms to 512 ms.

The interrupt is reset if the absolute value of the acceleration of all selected axes (“and” relation) is lower than the threshold minus the hysteresis for at least the time defined by *high_dur*. The interrupt status is stored in bit 1 (*high_int*) in register 0x09 (INT_STATUS_0). The high-g interrupt will be cleared immediately once the acceleration is lower than the threshold defined in *high_th*.

2.9.10.1 Axis and Sign Information

The axis which triggered the high-g interrupt is indicated by bits 0 (*high_first_x*), 1 (*high_first_y*) or 2 (*high_first_z*) in register 0x0C (INT_STATUS_3). The bit which corresponds to the triggering axis is set to 1 while the other bits are 0. These bits are cleared when the interrupt status is cleared. The sign of the triggering acceleration is stored in bit 3 (*high_sign*) in register 0x0C. If *high_sign* is 0 (1), the sign is positive (negative).

2.9.11 No-Motion / Slow-Motion Detection

The no-motion / slow-motion interrupt can be configured in two modes.

In **slow-motion** mode, an interrupt is triggered when the measured slope of at least one enabled axis exceeds the programmable slope threshold for a programmable number of samples. In order

to suppress false triggers, the interrupt is only generated (cleared) if a certain number *N* of consecutive slope data points is larger (smaller) than the slope threshold defined by bits <3:2> (*slo_no_mot_dur* <1:0>) in register 0x27 (INT_5). *N* is given by the following equation.

$$N = slo\_no\_mot\_dur\ \langle 1{:}0\rangle + 1$$

In **no-motion mode**, an interrupt is generated if the slope of all selected axes remains smaller than a programmable threshold for a programmable delay time. The timing diagram for the no-motion interrupt is shown in Figure 2-12.

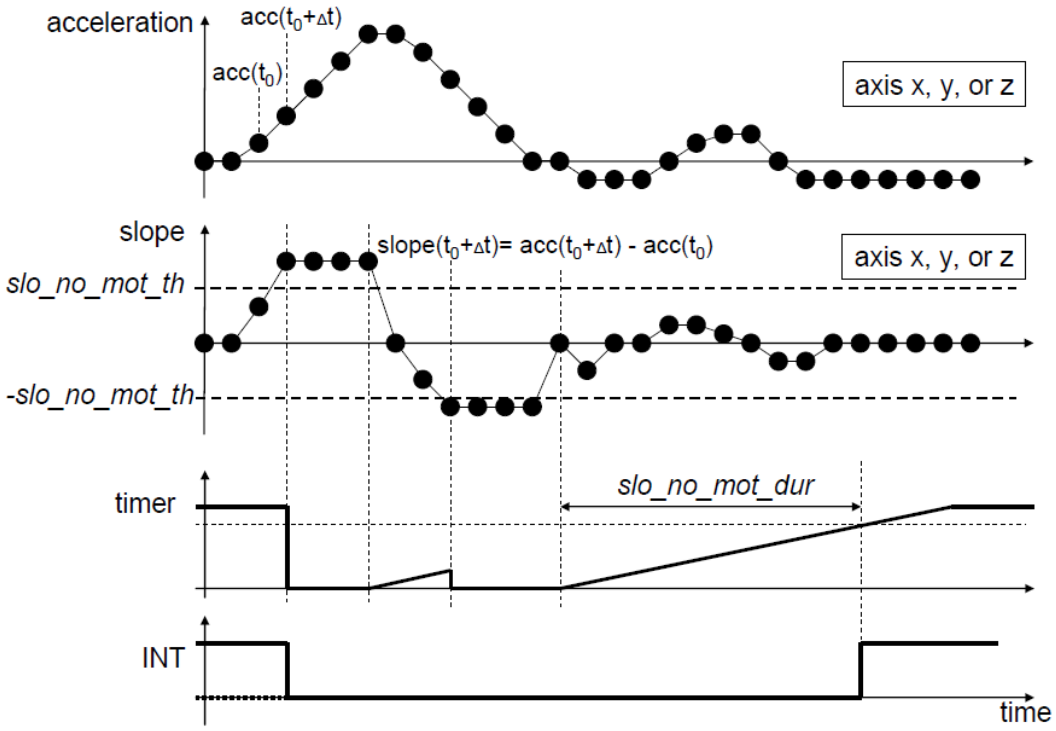


Figure 2-12: Timing of the no-motion interrupt.

The scaling of the threshold value is identical to the one of the slow-motion interrupt. However, in no-motion mode, bits <7:2> (*slo_no_mot_dur* <5:0>) in register 0x27 define the delay time before the no-motion interrupt is triggered. The table below lists the delay times which can be set via *slo_no_mot_dur*. The timer tick period is 1 s. Thus, using short delay times can result in considerable timing uncertainties.

slo_no_mot_dur **(DEC)**	**Delay Time**	***slo_no_mot_dur*** **(DEC)**	**Delay Time**	***slow_no_mot_dur*** **(DEC)**	**Delay Time**
0	1 s	16	40 s	32	88 s
1	2 s	17	48 s	33	96 s
2	3 s	18	56 s	34	104 s
...	...	19	64 s	...	...
14	15 s	20	72 s	62	328 s
15	16 s	21	80 s	63	336 s

Note: *slo_no_mot_dur* values 22 to 31 are not specified.

The delay times can be calculated with the help of the following equations:

slo_no_mot_dur <5:4> = 00: delay time = *slo_no_mot_dur* <3:0> + 1
slo_no_mot_dur <5:4> = 01: delay time = *slo_no_mot_dur* <3:0> · 4 + 20
slo_no_mot_dur <5> = 1: delay time = *slo_no_mot_dur* <4:0> · 8 + 88

The no-motion / slow-motion mode selection takes place via bit 3 (*slo_no_mot_sel*) in register 0x18 (INT_EN_2). If *slo_no_mot_sel* is set to 1 (0), the no-motion (slow-motion) mode is selected. In both modes, the slopes of the axes are monitored which have been enabled via bits 0 (*slo_no_mot_en_x*), 1 (*slo_no_mot_en_y*) and 2 (*slo_no_mot_en_z*) in register 0x18. The measured slope values are continuously compared against the threshold value defined by bits <7:0> (*slo_no_mot_th*) in register 0x29 (INT_7). The scaling of 1 LSB of *slo_no_mot_th* for the different g-ranges and the corresponding maximum value is given by the table below.

g-Range	Scaling of 1 LSB of *slo_no_mot_th*	Maximum Value
2 g	3.91 mg	996 mg
4 g	7.81 mg	1.99 g
8 g	15.6 mg	3.98 g
16 g	31.3 mg	7.97 g

The time difference between the successive acceleration samples depends on the selected filter bandwidth and is given by 1/(2·bandwidth). The interrupt status is stored in bit 3 (*slo_no_mot_int*) in register 0x09 (INT_STATUS_0).

3 Application

Proper function of the sensor in the overall system must be validated by the customer.

3.1 Sensing Axes Orientation

If the sensor is accelerated in the sensing directions indicated in Figure 3-1, the corresponding channels of the device will deliver a positive acceleration signal (dynamic acceleration). If the sensor is at rest and the force of gravity is acting along the indicated sensing directions, the output of the corresponding acceleration channels will be negative (static acceleration).

Example:

If the sensor is at rest or at uniform motion in a gravity field according to Figure 3-1, the output signals are

- ±0 g for the x-channel,
- ±0 g for the y-channel and
- +1 g for the z-channel.

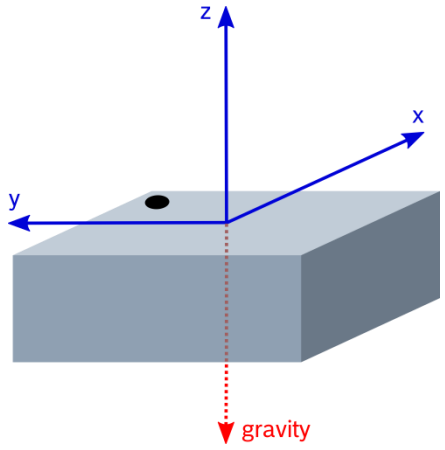
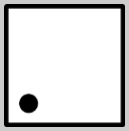
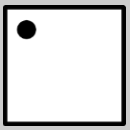
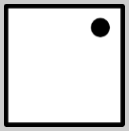
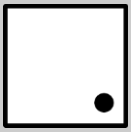


Figure 3-1: Orientation of the sensing axes.

The table below lists all corresponding output signals on x, y and z while the sensor is at rest or at uniform motion in a gravity field under assumption of a ±2 g range setting and a top down gravity vector as shown above.

Sensor orientation (gravity vector ↓)						
Output signal x	0 g 0 LSB	1 g 4096 LSB	0 g 0 LSB	-1 g - 4096 LSB	0 g 0 LSB	0 g 0 LSB
Output signal y	-1 g - 4096 LSB	0 g 0 LSB	1 g 4096 LSB	0 g 0 LSB	0 g 0 LSB	0 g 0 LSB
Output signal z	0 g 0 LSB	0 g 0 LSB	0 g 0 LSB	0 g 0 LSB	1 g 4096 LSB	-1 g - 4096 LSB

3.2 Pin-Out

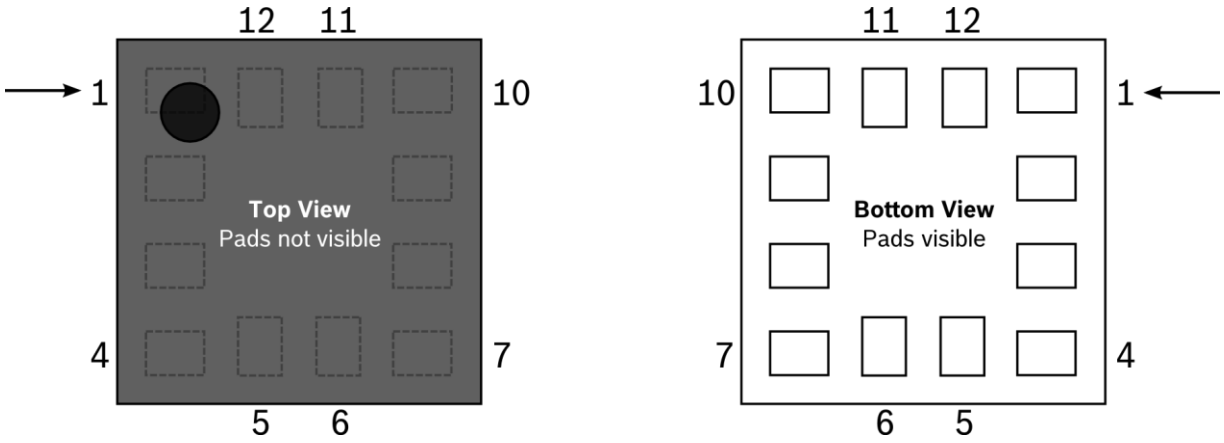


Figure 3-2: Pin-out top view (left) and bottom view (right). The arrow marks pin 1.

Pin	Name	I/O Type	Description	Connect to - SPI -	Connect to - TWI -
1	SDO	Digital out	SPI: serial data out TWI: address select	SDO	GND for default address
2	SDx	Digital I/O	SPI: SDI (serial data in) TWI: SDA (serial data I/O)	SDI	SDA
3	VDDIO	Supply	Digital I/O supply voltage	VDDIO	VDDIO
4	NC	-	-	GND	GND
5	INT1	Digital out	Interrupt pin 1	INT1 / DNC	INT1 / DNC
6	INT2	Digital out	Interrupt pin 2	INT2 / DNC	INT2 / DNC
7	VDD	Supply	Power supply analog & digital domain	VDD	VDD
8	GNDIO	Ground	Ground for I/O	GND	GND
9	GND	Ground	Ground for analog & digital domain	GND	GND
10	CSB	Digital in	SPI: chip select TWI: DNC	CSB	DNC (float)
11	PS	Digital in	Protocol select	GND	VDDIO
12	SCx	Digital in	Serial clock	SCK	SCL

DNC: Do not connect

INTx: If not needed, DNC

3.3 Dimensions and Weight

Dimensions [mm]: width: 2.0; length: 2.0; height 0.95
Weight [mg]: 8.65

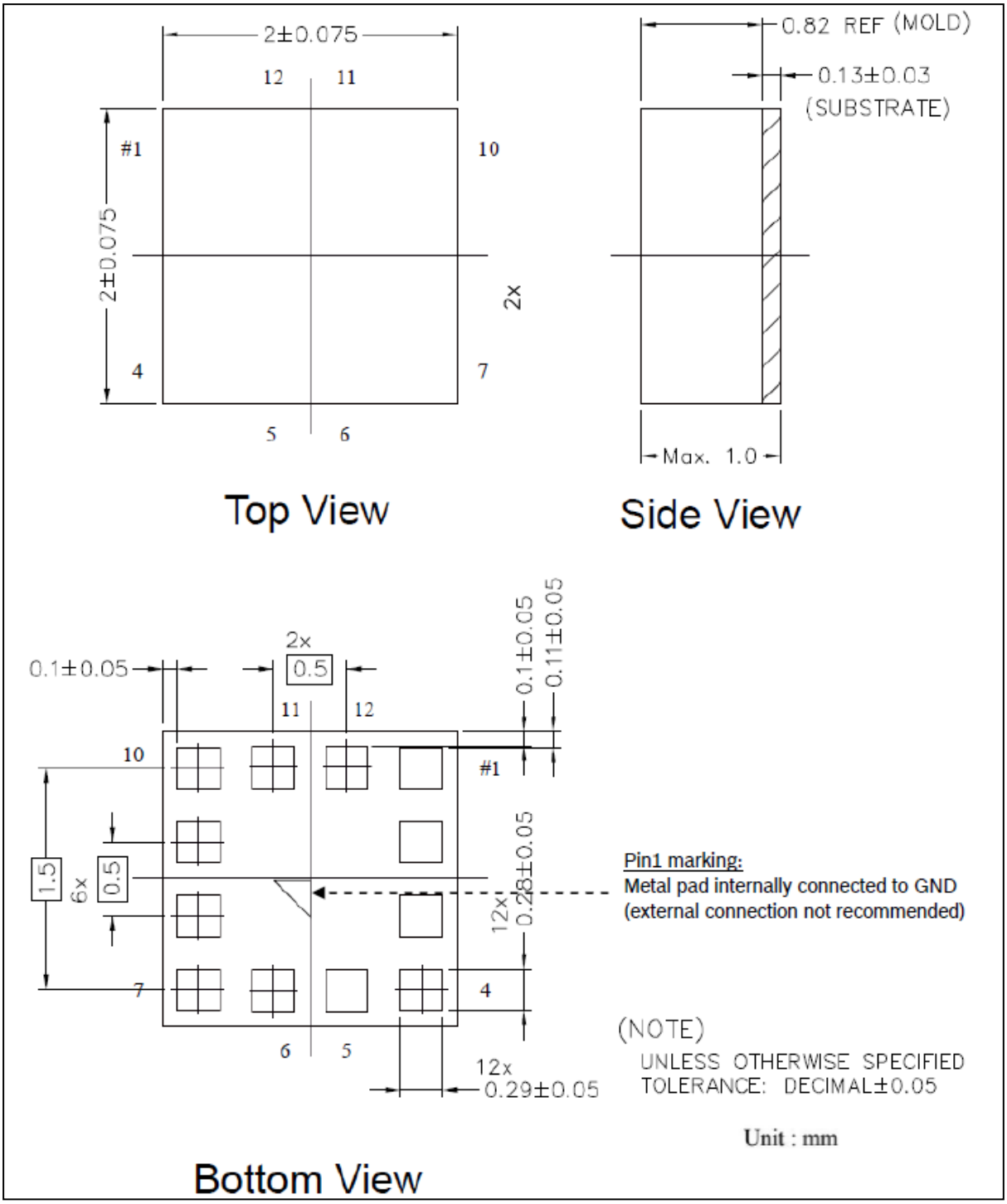


Figure 3-3: SMA130 package outline drawing.

3.4 Marking

Type ID	6	1 character to identify product type for the SMA130, this character is fixed as “6”
Lot ID	CCC	3 alphanumeric digits, variable to generate trace-code
Subcon ID	A	1 alphanumeric digit to identify subcon for the SMA130, this digit is fixed as “A”
Pin 1 identifier	●	--

3.5 Footprint

For the design of the landing patterns, the dimensioning as shown in Figure 3-4 is recommended.

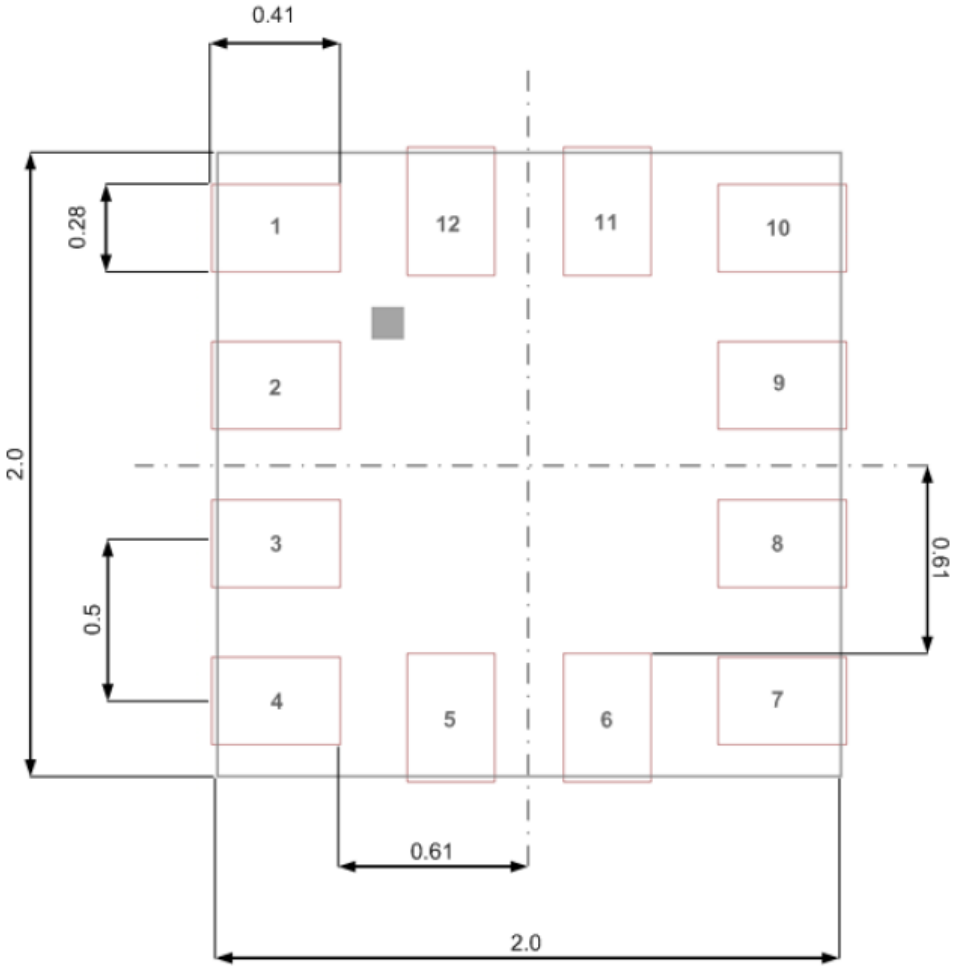


Figure 3-4: SMA130 footprint. All dimensions are given in mm.

The same tolerances as given for the outline dimensions (Section 3.3, Figure 3-3) should be assumed.

A wiring no-go area in the top layer of the PCB below the sensor is strongly recommended (e.g. no vias, wires or other metal structures).

3.6 SPI Connection Diagram

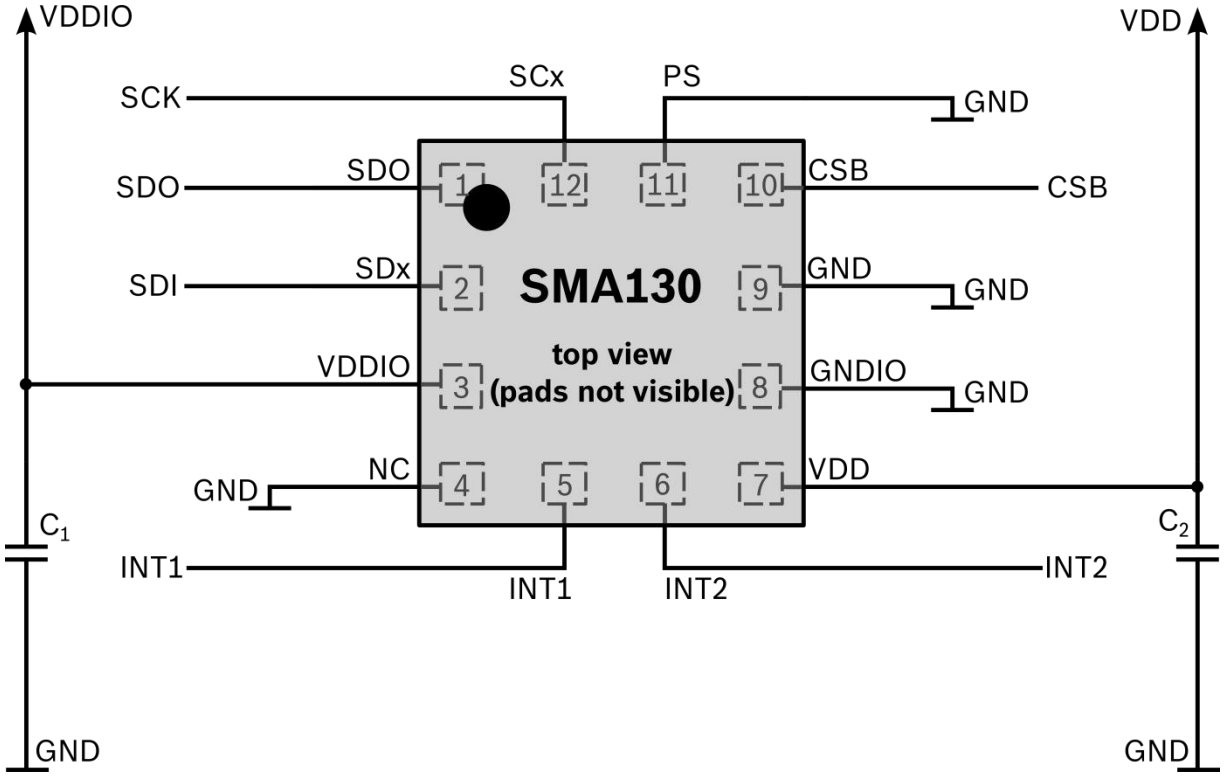


Figure 3-5: SPI connection diagram.

C_1, C_2: 100 nF

3.7 TWI Connection Diagram

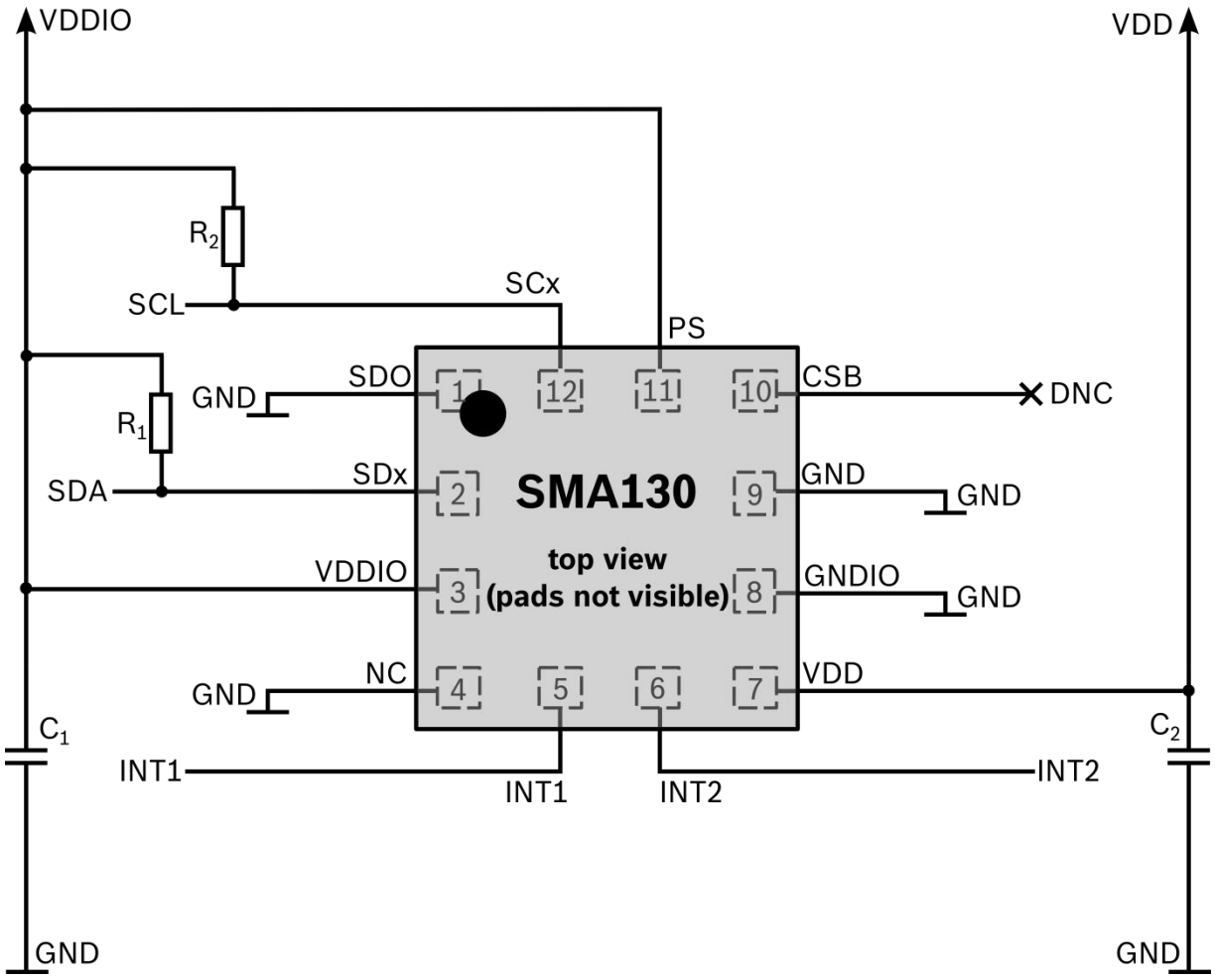


Figure 3-6: TWI connection diagram.

C_1, C_2: 100 nF
R_1, R_2: pull-up resistors

4 Specified Parameters

The data in this chapter, unless otherwise noted, apply for the valid operation conditions given in Section 4.2. All following figures include voltage, temperature and lifetime effects if not noted otherwise. All figures except for sensitivity are only valid without an external stimulus being applied. All operation conditions are only valid if no failure flags indicate any malfunction. All figures except for the noise itself exclude noise effects.

Proper function of the sensor in the overall system must be validated by the customer.

In any case, the electrical stability (power supply and EMC) of each system design including the SMA130 must be evaluated in advance to guarantee proper functionality during operation.

In any case, the mechanical stability of each system design including the SMA130 must be evaluated in advance to guarantee proper functionality during operation.

4.1 Absolute Maximum Ratings

Any values beyond the given ratings may seriously damage the device. The sensor must be discarded when exceeding these limits.

ABSOLUTE MAXIMUM RATINGS				
Parameter	**Condition**	**Min**	**Max**	**Units**
Voltage at supply pin	VDD Pin	-0.3	4.25	V
Voltage at supply pin	VDDIO Pin	-0.3	4.25	V
Voltage at any logic pin	Non-supply pin	-0.3	VDDIO + 0.3	V
Mechanical shock	Free fall onto hard surfaces		1.8	m
Mechanical shock	Duration ≤ 1 ms		2000	g
ESD	HBM, at any pin		2	kV
ESD	CDM		500	V
ESD	MM		200	V

4.2 Operating Conditions

OPERATING CONDITIONS						
Parameter	**Sym-bol**	**Condition**	**Min**	**Typical**	**Max***	**Units**
Supply voltage internal do-mains	VDD		1.62	2.4	3.6	V
Supply voltage I/O domain	VDDIO		1.2	2.4	3.6	V
Voltage input low level	V_{IL}				0.3 VDDIO	-
Voltage input high level	V_{IH}		0.7 VDDIO			-
Voltage output low level	V_{OL}	I_{OL} = 3 mA			0.2 VDDIO	-
Voltage output high level	V_{OH}	I_{OH} = 3 mA, SPI	0.8 VDDIO			-
Total supply current in nor-mal mode	I_{DD}	T = 25 °C, ODR_{max}, VDD = VDDIO = 2.4 V		130		µA
Total supply current in standby mode	I_{DDsbm}	T = 25 °C, VDD = VDDIO = 2.4 V		62		µA
Total supply current in sus-pend mode	I_{DDsum}	T = 25 °C, VDD = VDDIO = 2.4 V		1.4		µA
Total supply current in deep suspend mode	I_{DDdsum}	T = 25 °C, VDD = VDDIO = 2.4 V		1.0		µA
Total supply current in low power mode 1	I_{DDlp1}	T = 25 °C, unfiltered, VDD = VDDIO = 2.4 V, sleep duration = 25 ms		6.9		µA
Total supply current in low power mode 2	I_{DDlp2}	T = 25 °C, unfiltered, VDD = VDDIO = 2.4 V, sleep duration = 25 ms		66		µA
Start-up time	$t_{s,up}$	POR, ODR_{max}		1.2		ms
Wake-up time 1	$t_{w,up1}$	from low power mode 1, sus-pend mode or deep suspend mode, ODR_{max}		1.4		ms
Wake-up time 2	$t_{w,up2}$	from low power mode 2 or standby mode, ODR_{max}		1.0		ms
Operating tem-perature	T		-40		85	°C
Lifetime		According to AEC-Q100 grade 3 requirements				

* For specified maximum values, please refer to the Technical Customer Documentation.

4.3 Accelerometer Output Signal

ACCELEROMETER OUTPUT SIGNAL (all data for range 2 g, BW unfiltered)						
Parameter	**Sym-bol**	**Condition / Comment**	**Min**	**Typi-cal**	**Max***	**Units**
Measurement range	g_{FS}	selectable		±2 ±4 ±8 ±16		g
Sensitivity	S_{2g} S_{4g} S_{8g} S_{16g}	g_{FS2g}, T = 25 °C g_{FS4g}, T = 25 °C g_{FS8g}, T = 25 °C g_{FS16g}, T = 25 °C		4096 2048 1024 512		LSB/g
Sensitivity error		including temp., axis and lifetime effects			±5.0	%
Sensitivity temperature drift	TCS	nominal VDD supply, over full temp. range, no lifetime		0.014		%/K
Zero-g offset		including temp., axis and lifetime effects			±150	mg
Zero-g offset		T = 25 °C over lifetime		45		mg
Zero-g offset temperature drift	TCO	nominal VDD supply, over full temp. range, no lifetime		±0.7		mg/K
Bandwidth	BW	selectable 2nd order filter		8 16 31 63 125 250 500		Hz
Output data rate	ODR_{max}	unfiltered		2000		Hz
Nonlinearity	NL	best fit straight line, no life-time, T = 25 °C, 16 g range		±2.5		% FS
Noise rms		T = 25 °C, including axis and lifetime effects		3.9		mg
Cross axis sensitivity		T = 25 °C, no lifetime		1		%
Alignment error		MEMS element relative to package outline			±2	°
Temperature sensor slope				0.5		K/LSB
Temperature sensor offset		T = 25 °C		±5		K

* For specified maximum values, please refer to the Technical Customer Documentation.

5 Communication

The SMA130 supports two serial digital interface protocols for communication as a slave with a host device, SPI and I^2C compatible TWI. The active interface is selected by the state of the protocol select (PS) pin: 0 (1) selects SPI (TWI).

Both interfaces share the same pins. The mapping for each interface is given in the table below.

Pin	Name	Use with SPI	Use with TWI	Description
1	SDO	SDO	Address	SPI: serial data output TWI: used to set LSB of TWI address
2	SDx	SDI	SDA	SPI: serial data input TWI: serial data
10	CSB	CSB	Unused	SPI: chip select (enable) TWI: do not connect
12	SCx	SCK	SCL	Serial clock

The electrical specifications of the interface pins are shown in the table below.

Parameter	Symbol	Condition	Min	Typical	Max	Units
Pull-up resistance, CSB pin	R_{up}	Internal pull-up resistance to VDDIO	75	100	125	kΩ
Input capacitance	C_{in}			5	10	pF
TWI bus load capacitance (max. drive capability)	C_{TWI_load}				400	pF

5.1 Serial Peripheral Interface (SPI)

The timing specification for SPI of the SMA130 is given in the table below.

Parameter	Symbol	Condition	Min	Max	Units
Clock frequency	f_{SPI}	max. load on SDI or SDO = 25 pF, VDDIO ≥ 1.62 V		10	MHz
Clock frequency	f_{SPI}	max. load on SDI or SDO = 25 pF, VDDIO < 1.62 V		7.5	MHz
SCK low pulse	t_{SCKL}		20		ns
SCK high pulse	t_{SCKH}		20		ns
SDI setup time	t_{SDI_setup}		20		ns
SDI hold time	t_{SDI_hold}		20		ns
SDO output delay	t_{SDO_OD}	load = 25 pF, VDDIO ≥ 1.62 V		30	ns
SDO output delay	t_{SDO_OD}	load = 25 pF, VDDIO < 1.62 V		50	ns
SDO output delay	t_{SDO_OD}	load = 250 pF, VDDIO > 2.4 V		40	ns
CSB setup time	t_{CSB_setup}		20		ns
CSB hold time	t_{CSB_hold}		40		ns
Idle time between write accesses, normal mode, standby mode, low power mode 2	$t_{IDLE\_wacc\_nm}$		2		µs
Idle time between write accesses, suspend mode, low power mode 1	$t_{IDLE\_wacc\_sum}$		450		µs

Figure 5-1 shows the definition of the SPI timings.

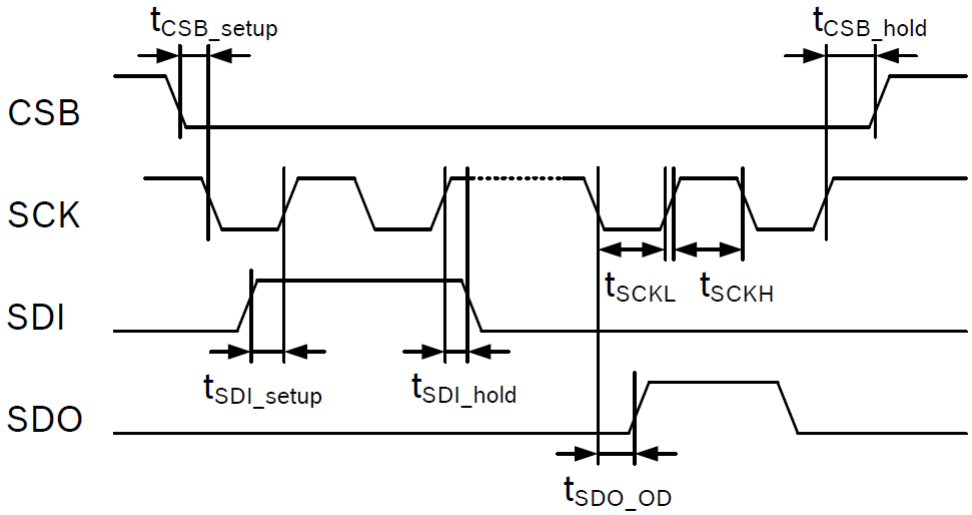


Figure 5-1: SPI timing diagram.

The SPI interface of the SMA130 is compatible with two modes, 00 and 11. The automatic selection between [CPOL = 0 and CPHA = 0] and [CPOL = 1 and CPHA = 1] is controlled based on the value of SCK after a falling edge of CSB. For single byte read as well as write operations, 16-bit protocols are used. The SMA130 also supports multiple-byte read operations.

For the standard SPI configuration, CSB (chip select low active), SCK (serial clock), SDI (serial data input) and SDO (serial data output) pins are used. The communication starts when CSB is pulled low by the SPI master and stops when CSB is pulled high. SCK is also controlled by the SPI master. SDI and SDO are driven at the falling edge of SCK and should be captured at the rising edge of SCK.

The basic write operation waveform for SPI configuration is depicted in Figure 5-2. During the full write cycle, SDO remains in high-impedance state.

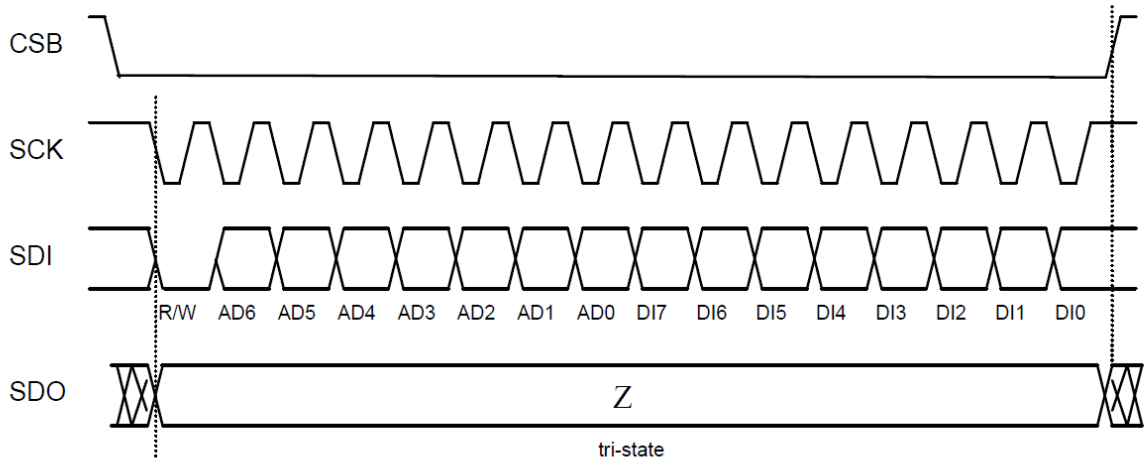


Figure 5-2: Basic SPI write sequence (mode 11).

The basic read operation waveform for SPI configuration is depicted in Figure 5-3.

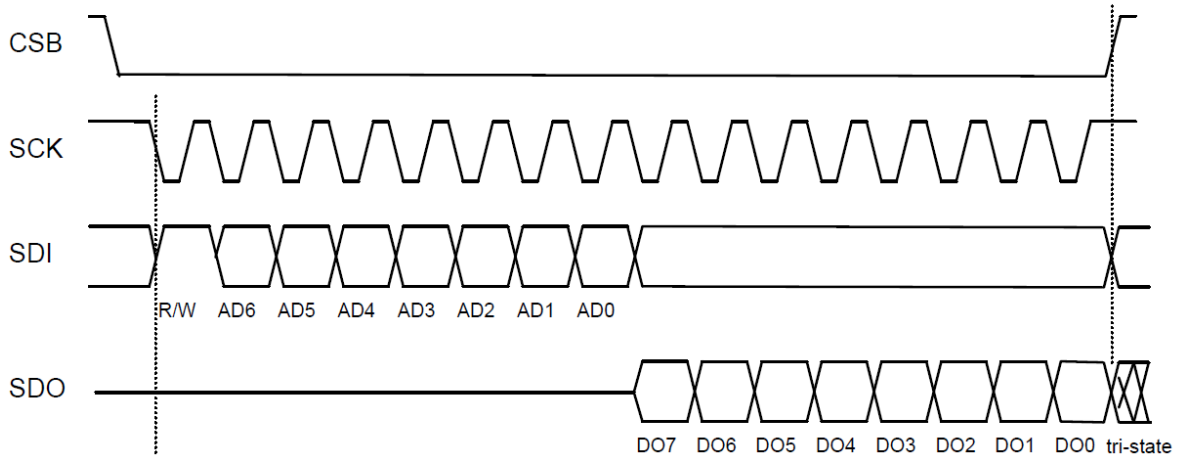


Figure 5-3: Basic SPI read sequence (mode 11).

The data bits are used as follows:

Bit 15: Read/write bit. When 0, the data SDI is written into the chip. When 1, the data SDO from the chip is read.

Bits <14:8>: Address AD(6:0).

Bits <7:0>: When in write mode, these are the data SDI which will be written into the address. When in read mode, these are the data SDO which are read from the address.

Multiple read operations are possible by keeping CSB low and continuing the data transfer. Only the first register address has to be written. Addresses are automatically incremented after each read access as long as CSB stays active low.

The principle of a multiple read operation is shown in Figure 5-4.

	Control byte								Data byte								Data byte								Data byte								
Start	RW	Register adress (02h)							Data register - adress 02h								Data register - adress 03h								Data register - adress 04h								Stop
CSB = 0	1	0	0	0	0	0	1	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	CSB = 1

Figure 5-4: SPI multiple read.

5.2 Two-Wire Interface (TWI)

The TWI bus uses SCL (= SCx pin, serial clock) and SDA (= SDx pin, serial data input and output) signal lines. Both lines are connected to VDDIO externally via pull-up resistors so that they are pulled high when the bus is free.

With some exceptions, the TWI interface of the SMA130 is compatible to the I²C specification UM10204 Rev. 03 (19 June 2007), available at http://www.nxp.com:

- The SMA130 supports the I²C standard and fast mode, but only the 7-bit address mode.
- For VDDIO = 1.2 ... 1.8 V the granted voltage output levels are slightly relaxed compared to the specification.
- The internal data hold time (t_{HDDAT}) of 300 ns is not met under all operation conditions. The device achieves a minimum value of 120 ns across process corners and temperature.
- The minimum data fall time (t_F) of ≥ 20 ns cannot be met.
- Only single byte write is supported.
- Detection of a stop condition is not supported. All data transfer protocols are fully operational by means of detecting the start condition only.
- The device does not support the high-impedance mode while VDDIO is tied to GND.
- The device does not perform clock stretching, i.e., clock frequencies may not exceed the one specified in the parameter section, and wait times between subsequent write accesses (as specified in Section 5.3) have to be ensured by the bus master.

The default TWI address of the SMA130 is 0x18 (0011000). It is used if the SDO pin is pulled to GND. The alternative address 0x19 (0011001) is selected by pulling the SDO pin to VDDIO.

The TWI timing specification for the SMA130 is given in the table below.

Parameter	Symbol	Min	Max	Units
Clock frequency	f_{SCL}		400	kHz
SCL low period	t_{LOW}	1.3		µs
SCL high period	t_{HIGH}	0.6		µs
SDA setup time	t_{SUDAT}	0.1		µs
SDA hold time	t_{HDDAT}	0.0		µs
Setup time for a repeated start condition	t_{SUSTA}	0.6		µs
Hold time for a start condition	t_{HDSTA}	0.6		µs
Setup time for a stop condition	t_{SUSTO}	0.6		µs
Time before a new transmission can start	t_{BUF}	1.3		µs
Idle time between write accesses, normal mode, standby mode, low power mode 2	$t_{IDLE\_wacc\_nm}$	2		µs
Idle time between write accesses, suspend mode, low power mode 1	$t_{IDLE\_wacc\_sum}$	450		µs

Figure 5-5 shows the definition of the TWI timing given in the table above.

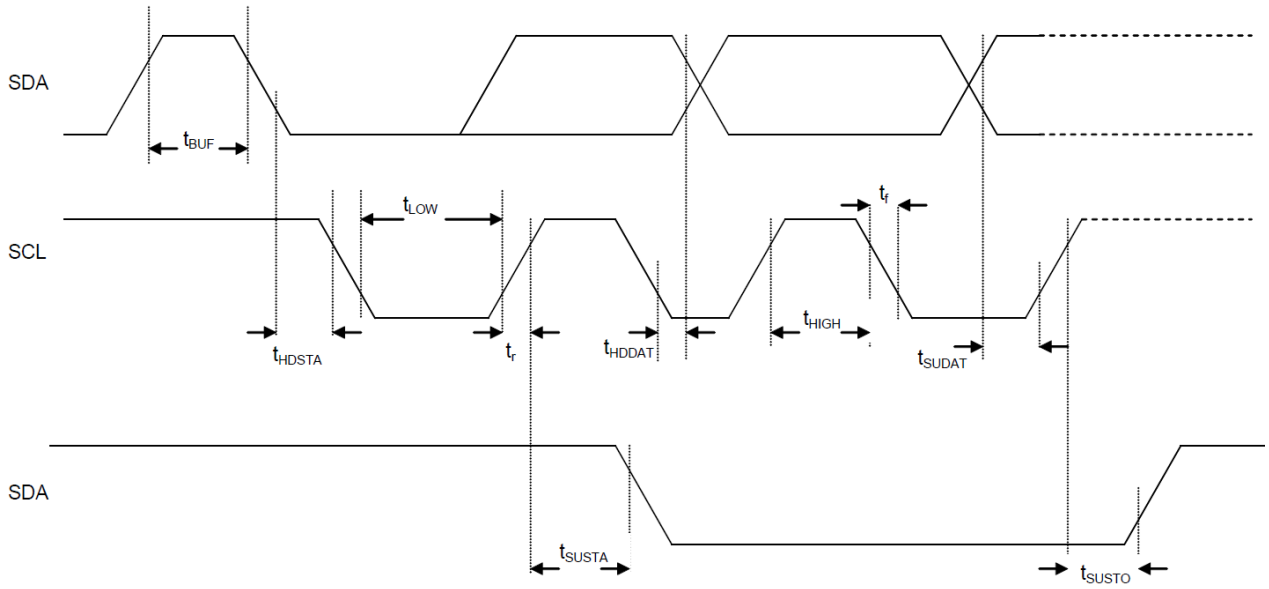


Figure 5-5: SMA130 TWI timing specification.

The TWI protocol works as follows:

START: Data transmission on the bus begins with a high to low transition on the SDA line while SCL is held high (start condition (S) indicated by the TWI bus master). Once the start signal is transferred by the master, the bus is considered busy.

STOP: Each data transfer should be terminated by a stop signal (P) generated by the master. The stop condition is a low to high transition on the SDA line while SCL is held high.

ACK: Each byte of data transferred must be acknowledged. It is indicated by an acknowledge bit sent by the receiver. The transmitter must release the SDA line (no pull down) during the acknowledge pulse while the receiver must then pull the SDA line low so that it remains stable low during the high period of the acknowledge clock cycle.

In the following diagrams these abbreviations are used:

S	Start
P	Stop
ACKS	Acknowledge by slave
ACKM	Acknowledge by master
NACKM	Not acknowledge by master
RW	Read / Write

A start (S) immediately followed by a stop (P) (without SCK toggling from VDDIO to GND) is not supported and not recognized by the SMA130.

TWI write access can be used to write a data byte in one sequence.

The sequence begins with a start condition generated by the master, followed by 7 bits of slave address and a write bit (RW = 0). The slave sends an acknowledge bit (ACK = 0) and releases the bus. Then the master sends the one byte register address. The slave again acknowledges the transmission and waits for the 8 bits of data which shall be written to the specified register address. After the slave acknowledges the data byte, the master generates a stop signal and terminates the writing protocol. Figure 5-6 shows an example of a TWI write access.

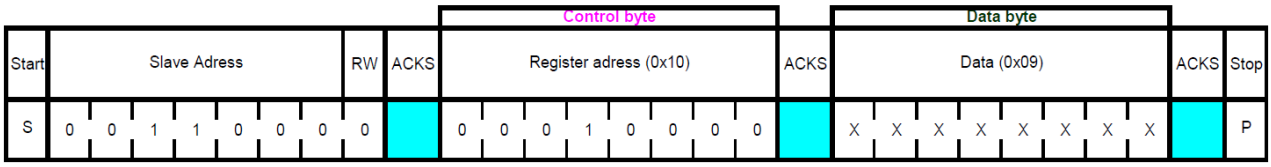


Figure 5-6: TWI write access.

TWI read access can be used to read one or multiple data bytes in one sequence.

A read sequence consists of a one-byte TWI write phase followed by the TWI read phase. Both parts of the transmission must be separated by a repeated start condition (Sr). The TWI write phase addresses the slave and sends the register address to be read. After the slave acknowledges the transmission, the master again generates a start condition and sends the slave address together with a read bit (RW = 1). Then the master releases the bus and waits for the data bytes to be read out from the slave. After each data byte the master has to generate an acknowledge bit (ACK = 0) to enable further data transfer. A NACKM (ACK = 1) from the master stops the data being transferred from the slave. The slave releases the bus so that the master can generate a stop condition and terminate the transmission.

The register address is automatically incremented. Hence, more than one byte can be sequentially read out. Once a new data read transmission starts, the start address will be set to the register address specified in the latest TWI write command. By default the start address is set as 0x00. In this way, repetitive multi-byte reads from the same starting address are possible.

In order to prevent the TWI slave from locking up the TWI bus, a watchdog timer (WDT) is implemented. The WDT observes internal TWI signals and resets the TWI interface if the bus is locked up. Activity and timer period of the WDT can be configured via bits 2 (*i2c_wdt_en*) and 1 (*i2c_wdt_sel*) in register 0x34 (BGW_ WDT).

- Writing 1 (0) to *i2c_wdt_en* activates (de-activates) the WDT.
- Writing 0 (1) to *i2c_wdt_sel* selects a timer period of 1 ms (50 ms).

Figure 5-7 shows an example of a TWI multiple read access.

Figure 5-7: TWI multiple read access.

5.3 Access Restrictions (SPI and TWI)

In order to allow for the correct internal synchronization of data written to the SMA130, certain access restrictions apply for consecutive write accesses or a write/read sequence through the SPI and TWI interface. The required waiting period depends on whether the device is operating in normal mode (also standby mode or low power mode 2) or suspend mode (also low power mode 1).

As illustrated in Figure 5-8, an interface idle time of at least 2 µs is required following a write operation when the device operates in normal mode (also standby mode or low power mode 2). In suspend mode (also low power mode 1), an interface idle time of at least 450 µs is required.

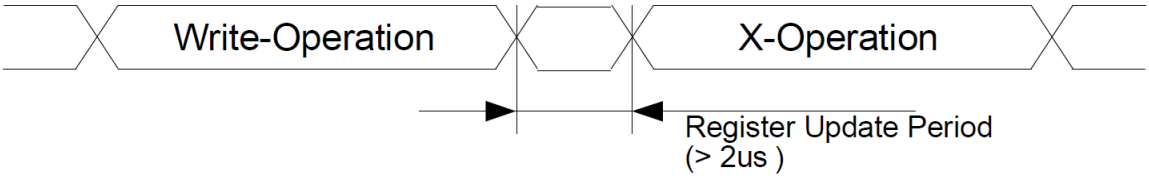


Figure 5-8: Post-write access timing constraints.

5.4 Self-Test

The self-test feature allows for checking the sensor functionality by applying electrostatic forces to the sensor core instead of external accelerations. By physically deflecting the seismic mass, the entire signal path of the sensor is tested. Activation of the self-test results in a static offset in the acceleration data. Any external acceleration or gravitational force which is applied to the sensor during a self-test will be observed in the sensor output as a superposition of the acceleration and the self-test signal.

Before enabling the self-test, the acceleration measurement range should be set to **4 g**.

The self-test is activated for **each axis separately** by setting bits <1:0> (*self_test_axis*) of register 0x32 (PMU_SELF_TEST) to 01 for the x-axis, 10 for the y-axis or 11 for the z-axis. For *self_test_axis* = 00, the self-test is disabled. The **direction of the deflection** is controlled via bit 2 (*self_test_sign*). The deflection is negative (positive) when setting *self_test_sign* to 0 (1).

After enabling the self-test, a **waiting time of 50 ms** is mandatory for each axis before the acceleration data are interpreted.

For a proper interpretation of the self-test signals, it is recommended to perform the self-test for both the positive and the negative direction and to then calculate the difference of the resulting acceleration values. The minimum difference for each axis is shown in the table below. The actually measured signal differences can be significantly larger.

	x-axis	y-axis	z-axis
minimum difference signal	800 mg	800 mg	400 mg

After performing a self-test, a reset of the device is recommended. If the reset cannot be performed, the following sequence must be kept to prevent unwanted interrupt generation:

1. Disable interrupts.
2. Change parameters of interrupts.
3. Wait for at least 50 ms.
4. Enable desired interrupts.

6 Register Description

6.1 General Remarks

The entire communication with the device is performed by reading from and writing to registers. Registers have a width of 8 bits. They are mapped to a common space of 64 addresses from 0x00 up to 0x3F. Within this range some registers are either completely or partially marked as ‘reserved’. Any reserved bit is ignored when it is written and no specific value is guaranteed when the bit is read. It is recommended not to use registers which are completely marked as ‘reserved’ at all. Furthermore it is recommended to mask out (logical and with zero) reserved bits of registers which are partially marked as reserved.

Registers with addresses from 0x00 up to 0x0E are read-only. Any attempt to write to those registers will be ignored. There are bits within some registers that trigger internal sequences. These bits are configured for write-only access and read as 0. An example for such a write-only access is the entire register 0x14 (BGW_SOFTRESET).

6.2 Register Map

Figure 6-1 shows the register map of the SMA130.

Register	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Default
0x3A	offset_z <7:0>								0x00
0x39	offset_y <7:0>								0x00
0x38	offset_x <7:0>								0x00
0x36	offset_reset	cal_trigger <1:0>		cal_rdy					0x10
0x34						i2c_wdt_en	i2c_wdt_sel		0x00
0x32						self_test_sign	self_test_axis <1:0>		0x00
0x2F			flat_hold_time <1:0>			flat_hy <2:0>			0x11
0x2E			flat_theta <5:0>						0x08
0x2D		orient_ud_en	orient_theta <5:0>						0x48
0x2C		orient_hyst <2:0>			orient_blocking <1:0>		orient_mode <1:0>		0x18
0x2B	tap_samp <1:0>			tap_th <4:0>					0x0A
0x2A	tap_quiet	tap_shock				tap_dur <2:0>			0x04
0x29	slo_no_mot_th <7:0>								0x14
0x28	slope_th <7:0>								0x14
0x27	slo_no_mot_dur <5:0>						slope_dur <1:0>		0x00
0x26	high_th <7:0>								0xC0
0x25	high_dur <7:0>								0x0F
0x24	high_hy <1:0>					low_mode	low_hy <1:0>		0x81
0x23	low_th <7:0>								0x30
0x22	low_dur <7:0>								0x09
0x21	reset_int				latch_int <3:0>				0x00
0x20					int2_od	int2_lvl	int1_od	int1_lvl	0x05
0x1E			int_src_data	int_src_tap	int_src_slo_no_mot	int_src_slope	int_src_high	int_src_low	0x00
0x1B	int2_flat	int2_orient	int2_s_tap	int2_d_tap	int2_slo_no_mot	int2_slope	int2_high	int2_low	0x00
0x1A	int2_data							int1_data	0x00
0x19	int1_flat	int1_orient	int1_s_tap	int1_d_tap	int1_slo_no_mot	int1_slope	int1_high	int1_low	0x00
0x18					slo_no_mot_sel	slo_no_mot_en_z	slo_no_mot_en_y	slo_no_mot_en_x	0x00
0x17				data_en	low_en	high_en_z	high_en_y	high_en_x	0x00
0x16	flat_en	orient_en	s_tap_en	d_tap_en		slope_en_z	slope_en_y	slope_en_x	0x00
0x14	softreset								0x00
0x13	data_high_bw	shadow_dis							0x00
0x12		lowpower_mode							0x00
0x11	suspend	lowpower_en	deep_suspend	sleep_dur <3:0>					0x00
0x10				bw <4:0>					0x0F
0x0F					range <3:0>				0x03
0x0C	flat	orient <2:0>			high_sign	high_first_z	high_first_y	high_first_x	0x00
0x0B	tap_sign	tap_first_z	tap_first_y	tap_first_x	slope_sign	slope_first_z	slope_first_y	slope_first_x	0x00
0x0A	data_int								0x00
0x09	flat_int	orient_int	s_tap_int	d_tap_int	slo_no_mot_int	slope_int	high_int	low_int	0x00
0x08	temp <7:0>								0x00
0x07	acc_z_msb <13:6>								0x00
0x06	acc_z_lsb <5:0>							new_data_z	0x00
0x05	acc_y_msb <13:6>								0x00
0x04	acc_y_lsb <5:0>							new_data_y	0x00
0x03	acc_x_msb <13:6>								0x00
0x02	acc_x_lsb <5:0>							new_data_x	0x00
0x00	chip_id <7:0>								0xFB

	w / r
	write only
	ready only
	common w / r registers *)
	user w / r registers **)
	reserved

*) Application specific settings not equal to default. Must be reset to designated values after POR, soft reset and wake-up from deep suspend.
**) Initial default content = 0x00. Freely programmable by user. Remains unchanged after POR, soft reset and wake-up from deep suspend.

Figure 6-1: SMA130 register map.

6.2.1 Register 0x00 (BGW_CHIPID)

This register contains the chip identification code.

Name	0x00 (BGW_CHIPID)			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*chip_id* <7:4>			

Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*chip_id* <3:0>			

chip_id <7:0>: Fixed value 11111011

6.2.2 Register 0x02 (ACCD_X_LSB)

This register contains the least significant bits of the x-channel acceleration readout value. When reading out x-channel acceleration values, data consistency is guaranteed if the ACCD_X_LSB is read out before the ACCD_X_MSB and *shadow_dis* = 0. In this case, after the ACCD_X_LSB has been read, the value in the ACCD_X_MSB register is locked until the ACCD_X_MSB has been read. Acceleration data may be read from register ACCD_X_LSB at any time except during power-up and in deep suspend mode.

Name	0x02 (ACCD_X_LSB)			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*acc_x_lsb* <5:2>			

Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*acc_x_lsb* <1:0>		*undefined*	*new_data_x*

acc_x_lsb <5:0>: Least significant 6 bits of the acceleration x-channel read-back value (two's complement format)

undefined: Random data, to be ignored

new_data_x: 0: acceleration value has not been updated since it has been read out last
1: acceleration value has been updated since it has been read out last

6.2.3 Register 0x03 (ACCD_X_MSB)

This register contains the most significant bits of the x-channel acceleration readout value. When reading out x-channel acceleration values, data consistency is guaranteed if the ACCD_X_LSB is read out before the ACCD_X_MSB and *shadow_dis* = 0. In this case, after the ACCD_X_LSB has been read, the value in the ACCD_X_MSB register is locked until the ACCD_X_MSB has been read. Acceleration data may be read from register ACCD_X_MSB at any time except during power-up and in deep suspend mode.

Name	**0x03 (ACCD_X_MSB)**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*acc_x_msb* <13:10>			

Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*acc_x_msb* <9:6>			

acc_x_msb <11:4>: Most significant 8 bits of the acceleration x-channel read-back value (two's complement format)

6.2.4 Register 0x04 (ACCD_Y_LSB)

This register contains the least significant bits of the y-channel acceleration readout value. When reading out y-channel acceleration values, data consistency is guaranteed if the ACCD_Y_LSB is read out before the ACCD_Y_MSB and *shadow_dis* = 0. In this case, after the ACCD_Y_LSB has been read, the value in the ACCD_Y_MSB register is locked until the ACCD_Y_MSB has been read. Acceleration data may be read from register ACCD_Y_LSB at any time except during power-up and in deep suspend mode.

Name	**0x04 (ACCD_Y_LSB)**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*acc_y_lsb* <5:2>			

Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*acc_y_lsb* <1:0>		*undefined*	*new_data_y*

acc_y_lsb <5:0>: Least significant 6 bits of the acceleration y-channel read-back value (two's complement format)

undefined: Random data, to be ignored

new_data_y: 0: acceleration value has not been updated since it has been read out last
1: acceleration value has been updated since it has been read out last

6.2.5 Register 0x05 (ACCD_Y_MSB)

This register contains the most significant bits of the y-channel acceleration readout value. When reading out y-channel acceleration values, data consistency is guaranteed if the ACCD_Y_LSB is read out before the ACCD_Y_MSB and *shadow_dis* = 0. In this case, after the ACCD_Y_LSB has been read, the value in the ACCD_Y_MSB register is locked until the ACCD_Y_MSB has been read. Acceleration data may be read from register ACCD_Y_MSB at any time except during power-up and in deep suspend mode.

Name	**0x05 (ACCD_Y_MSB)**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*acc_y_msb* <13:10>			

Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*acc_y_msb* <9:6>			

acc_y_msb <13:6>: Most significant 8 bits of the acceleration y-channel read-back value (two's complement format)

6.2.6 Register 0x06 (ACCD_Z_LSB)

This register contains the least significant bits of the z-channel acceleration readout value. When reading out z-channel acceleration values, data consistency is guaranteed if the ACCD_Z_LSB is read out before the ACCD_Z_MSB and *shadow_dis* = 0. In this case, after the ACCD_Z_LSB has been read, the value in the ACCD_Z_MSB register is locked until the ACCD_Z_MSB has been read. Acceleration data may be read from register ACCD_Z_LSB at any time except during power-up and in deep suspend mode.

Name	**0x06 (ACCD_Z_LSB)**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*acc_z_lsb* <5:2>			

Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*acc_z_lsb* <1:0>		*undefined*	*new_data_z*

acc_z_lsb <5:0>: Least significant 6 bits of the acceleration z-channel read-back value (two's complement format)

undefined: Random data, to be ignored

new_data_z: 0: acceleration value has not been updated since it has been read out last
1: acceleration value has been updated since it has been read out last

6.2.7 Register 0x07 (ACCD_Z_MSB)

This register contains the most significant bits of the z-channel acceleration readout value. When reading out z-channel acceleration values, data consistency is guaranteed if the ACCD_Z_LSB is read out before the ACCD_Z_MSB and *shadow_dis* = 0. In this case, after the ACCD_Z_LSB has been read, the value in the ACCD_Z_MSB register is locked until the ACCD_Z_MSB has been read. Acceleration data may be read from register ACCD_Z_MSB at any time except during power-up and in deep suspend mode.

Name	**0x07 (ACCD_Z_MSB)**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*acc_z_msb* <13:10>			

Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*acc_z_msb* <9:6>			

acc_z_msb <13:6>: Most significant 8 bits of the acceleration z-channel read-back value (two's complement format)

6.2.8 Register 0x08 (ACCD_TEMP)

This register contains the current chip temperature as a 8 bit data word in two's complement format. A readout value of *temp* <7:0> = 0x00 corresponds to a temperature of 23 °C.

Name	**0x08 (ACC_TEMP)**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*temp* <7:4>			

Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*temp* <3:0>			

temp <7:0>: Temperature value (two's complement format)

6.2.9 Register 0x09 (INT_STATUS_0)

This register contains the interrupt status flags *flat_int*, *orient_int*, *s_tap_int*, *d_tap_int*, *slo_no_mot_int*, *slope_int*, *high_int* and *low_int*. Each flag is associated with a specific interrupt function. It is set when the associated interrupt triggers.

The setting of the bits <3:0> (*latch_int*) in register 0x21 (INT_RST_LATCH) controls if the interrupt signal and hence the respective interrupt flag will be permanently latched, temporarily latched or non-latched.

The interrupt function associated with a specific status flag has to be enabled separately.

Name	**0x09 (INT_STATUS_0)**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*flat_int*	*orient_int*	*s_tap_int*	*d_tap_int*

Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*slo_no_mot_int*	*slope_int*	*high_int*	*low_int*

flat_int: Flat interrupt status
0: inactive
1: active

orient_int: Orientation interrupt status
0: inactive
1: active

s_tap_int: Single tap interrupt status
0: inactive
1: active

d_tap_int: Double tap interrupt status
0: inactive
1: active

slo_no_mot_int: Slow / no-motion interrupt status
0: inactive
1: active

slope_int: Slope interrupt status
0: inactive
1: active

high_int: High-g interrupt status
0: inactive
1: active

low_int: Low-g interrupt status
0: inactive
1: active

6.2.10 Register 0x0A (INT_STATUS_1)

This register contains the interrupt status flag *data_int* of the new data interrupt.

The new data interrupt allows for synchronous reading of acceleration data. It is generated after a new value of z-axis acceleration data has been stored in the data register.

The interrupt is cleared automatically when the next data acquisition cycle starts. The interrupt status is 0 for a minimum of 50 µs. It is fixed to the non-latched mode.

The interrupt function associated with the status flag has to be enabled via setting bit 4 (*data_en*) in register 0x17 (INT_EN_1) to 1.

Name	**0x0A (INT_STATUS_1)**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*data_int*	*reserved*		

Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*reserved*			

data_int: Data ready interrupt status
0: inactive
1: active

reserved: Random data, to be ignored

6.2.11 Register 0x0B (INT_STATUS_2)

This register contains the interrupt status flags *tap_sign*, *tap_first_z*, *tap_first_y*, *tap_first_x*, *slope_sign*, *slope_first_z*, *slope_first_y* and *slope_first_x*. Each flag is associated with a specific interrupt function. It is set when the associated interrupt triggers.

The setting of the bits <3:0> (*latch_int*) in register 0x21 (INT_RST_LATCH) controls if the interrupt signal and hence the respective interrupt flag will be permanently latched, temporarily latched or non-latched.

The interrupt function associated with a specific status flag has to be enabled separately.

Name	**0x0B (INT_STATUS_2)**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*tap_sign*	*tap_first_z*	*tap_first_y*	*tap_first_x*

Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*slope_sign*	*slope_first_z*	*slope_first_y*	*slope_first_x*

tap_sign: Sign of the single tap / double tap triggering signal was
0: positive
1: negative

tap_first_z: Single tap / double tap interrupt was
0: not triggered by z-axis
1: triggered by z-axis

tap_first_y: Single tap / double tap interrupt was
0: not triggered by y-axis
1: triggered by y-axis

tap_first_x: Single tap / double tap interrupt was
0: not triggered by x-axis
1: triggered by x-axis

slope_sign: Slope sign of the slope triggering signal was
0: positive
1: negative

slope_first_z: Slope interrupt was
0: not triggered by z-axis
1: triggered by z-axis

slope_first_y: Slope interrupt was
0: not triggered by y-axis
1: triggered by y-axis

slope_first_x: Slope interrupt was
0: not triggered by x-axis
1: triggered by x-axis

6.2.12 Register 0x0C (INT_STATUS_3)

This register contains the interrupt status flags *flat*, *orient*, *high_sign*, *high_first_z*, *high_first_y* and *high_first_x*. Each flag is associated with a specific interrupt function. It is set when the associated interrupt triggers.

With the exception of *orient*, the setting of the bits <3:0> (*latch_int*) in register 0x21 (INT_RST_LATCH) controls if the interrupt signal and hence the respective interrupt flag will be permanently latched, temporarily latched or non-latched.

The interrupt function associated with a specific status flag has to be enabled separately.

Name	**0x0C (INT_STATUS_3)**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*flat*	*orient* <2:0>		

Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	n/a	n/a	n/a	n/a
Content	*high_sign*	*high_first_z*	*high_first_y*	*high_first_x*

flat: Position of the SMA130 is
0: non-flat
1: flat
Only valid if bit 7 (*flat_en*) in register 0x16 (INT_EN_0) is 1

orient <2>: Orientation of the z-axis is
0: upward-looking
1: downward-looking
The flag always reflects the current orientation status, independent of the setting of *latch_int*. The flag is not updated as long as an orientation blocking condition is active.

orient <1:0>: Orientation of the x-y-plane is
00: portrait upright
01: portrait upside down
10: landscape left
11: landscape right
The flag always reflects the current orientation status, independent of the setting of *latch_int*. The flag is not updated as long as an orientation blocking condition is active.

high_sign: Sign of the acceleration signal that triggered the high-g interrupt was
0: positive
1: negative

high_first_z: High-g interrupt was
0: not triggered by z-axis
1: triggered by z-axis

high_first_y: High-g interrupt was
0: not triggered by y-axis
1: triggered by y-axis

high_first_x: High-g interrupt was
0: not triggered by x-axis
1: triggered by x-axis

6.2.13 Register 0x0F (PMU_RANGE)

This register allows for the selection of the accelerometer g-range.

Name	0x0F (PMU_RANGE)			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*reserved*			

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	1	1
Content	*range* <3:0>			

range <3:0>: Selection of the accelerometer g-range — Resolution [LSB/g]

0011:	±2 g	4096
0101:	±4 g	2048
1000:	±8 g	1024
1100:	±16 g	512

All other settings: reserved (do not use)

reserved: Write 0

6.2.14 Register 0x10 (PMU_BW)

This register allows for the selection of the acceleration data filter bandwidth.

Name	0x10 (PMU_BW)			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*reserved*			*bw* <4>

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	1	1	1	1
Content	*bw* <3:0>			

bw <4:0>: Selection of the data filter bandwidth

00xxx:	7.81 Hz
01000:	7.81 Hz
01001:	15.63 Hz
01010:	13.25 Hz
01011:	62.5 Hz
01100:	125 Hz
01101:	250 Hz
01110:	500 Hz
01111:	ODR_{max}
1xxxx:	ODR_{max}

reserved: Write 0

6.2.15 Register 0x11 (PMU_LPW)

This register allows for the selection of the main power modes – in combination with register 0x12 (PMU_LOW_POWER) – and the low power mode sleep period.

Name	**0x11 (PMU_LPW)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*suspend*	*lowpower_en*	*deep_suspend*	*sleep_dur* <3>

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*sleep_dur* <2:0>			*reserved*

suspend,
lowpower_en,
deep_suspend: Main power mode configuration setting

000: normal mode
001: deep suspend mode
010: low power mode 1 or low power mode 2
100: standby or suspend mode

All other settings: illegal (do not use)

Please note that only certain power mode transitions are permitted.

sleep_dur <3:0>: Sleep phase duration in low power mode 1 or low power mode 2

0000 to 0101: 0.5 ms
0110: 1 ms
0111: 2 ms
1000: 4 ms
1001: 6 ms
1010: 10 ms
1011: 25 ms
1100: 50 ms
1101: 100 ms
1110: 500 ms
1111: 1 s

reserved: Write 0

Please note that all application specific settings which are not equal to the default settings must be reset to their designated values after leaving deep suspend mode.

6.2.16 Register 0x12 (PMU_LOW_POWER)

This register – in combination with register 0x11 (PMU_LPW) – contains the configuration settings for the main power modes.

Name	**0x12 (PMU_LOW_POWER)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*reserved*	*lowpower_mode*	*reserved*	

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*reserved*			

lowpower_mode: Power mode configuration

0: low power mode 1, normal mode, suspend mode, deep suspend mode
1: low power mode 2, standby mode

All other settings: illegal (do not use)

reserved: Write 0

Overview of the different configurations of register 0x11 (PMU_LPW) and 0x12 (PMU_LOW_POWER):

Mode	**0x11 (PMU_LPW)**			**0x12 (PMU_LOW_POWER)**
	7 *suspend*	6 *lowpower_en*	5 *deep_suspend*	6 *lowpower_mode*
Normal	0	0	0	0
Standby	1	0	0	1
Suspend	1	0	0	0
Deep suspend	0	0	1	0
Low power mode 1	0	1	0	0
Low power mode 2	0	1	0	1

6.2.17 Register 0x13 (ACCD_HBW)

This register controls the acceleration data acquisition and data output format.

Name	**0x13 (ACCD_HBW)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0 (1 in 8-bit mode)	0	0
Content	*data_high_bw*	*shadow_dis*	*reserved*	

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*reserved*			

data_high_bw: Data-read from the acceleration data registers
0: filtered
1: unfiltered

shadow_dis: The shadowing mechanism for the acceleration data output registers. When shadowing is enabled, the content of the acceleration data component in the MSB register is locked when the component in the LSB is read, thereby ensuring the integrity of the acceleration data during read-out. The lock is removed when the MSB is read.
0: enable
1: disable

reserved: Write 0

6.2.18 Register 0x14 (BGW_SOFTRESET)

This register controls the user triggered soft reset of the sensor.

Name	**0x14 (BGW_SOFTRESET)**			
Bit	7	6	5	4
Read/Write	W	W	W	W
Reset Value	0	0	0	0
Content	*softreset*			

Bit	3	2	1	0
Read/Write	W	W	W	W
Reset Value	0	0	0	0
Content	*softreset*			

softreset: Writing 0xB6 to the register triggers a reset. Other values are ignored. After a delay, all user configuration settings are overwritten with their default values. This register is functional in all operation modes. Please note that all application specific settings which are not equal to the default settings (refer to the register map in Section 6.2) must be reconfigured to their designated values.

6.2.19 Register 0x16 (INT_EN_0)

This register controls which of the interrupts flat detection, orientation recognition, tap detection and slope / any-motion detection are enabled.

Name	**0x16 (INT_EN_0)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*flat_en*	*orient_en*	*s_tap_en*	*d_tap_en*

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*reserved*	*slope_en_z*	*slope_en_y*	*slope_en_x*

flat_en: Flat interrupt is
0: disabled
1: enabled

orient_en: Orientation interrupt is
0: disabled
1: enabled

s_tap_en: Single tap interrupt is
0: disabled
1: enabled

d_tap_en: Double tap interrupt is
0: disabled
1: enabled

slope_en_z: z-axis component of the slope / any-motion interrupt is
0: disabled
1: enabled

slope_en_y: y-axis component of the slope / any-motion interrupt is
0: disabled
1: enabled

slope_en_x: x-axis component of the slope / any-motion interrupt is
0: disabled
1: enabled

reserved: Write 0

6.2.20 Register 0x17 (INT_EN_1)

This register controls which of the interrupts new data, low-g and high-g are enabled.

Name	**0x17 (INT_EN_1)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*reserved*			*data_en*

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*low_en*	*high_en_z*	*high_en_y*	*high_en_x*

data_en: New data interrupt is
0: disabled
1: enabled

low_en: Low-g interrupt is
0: disabled
1: enabled

high_en_z: z-axis component of the high-g interrupt is
0: disabled
1: enabled

high_en_y: y-axis component of the high-g interrupt is
0: disabled
1: enabled

high_en_x: x-axis component of the high-g interrupt is
0: disabled
1: enabled

reserved: Write 0

6.2.21 Register 0x18 (INT_EN_2)

This register controls the settings of the no-motion / slow-motion interrupt.

Name	**0x18 (INT_EN_2)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*reserved*			

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*slo_no_mot_sel*	*slo_no_mot_en_z*	*slo_no_mot_en_y*	*slo_no_mot_en_x*

slo_no_mot_sel: Select
0: slow-motion interrupt function
1: no-motion interrupt function

slo_no_mot_en_z: z-axis component of the no-motion / slow-motion interrupt is
0: disabled
1: enabled

slo_no_mot_en_y: y-axis component of the no-motion / slow-motion interrupt is
0: disabled
1: enabled

slo_no_mot_en_x: x-axis component of the no-motion / slow-motion interrupt is
0: disabled
1: enabled

reserved: Write 0

6.2.22 Register 0x19 (INT_MAP_0)

This register controls which of the interrupts flat detection, orientation recognition, tap detection, no-motion / slow-motion, slope / any-motion, high-g and low-g are mapped to the INT1 pin.

Name	**0x19 (INT_MAP_0)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*int1_flat*	*int1_orient*	*int1_s_tap*	*int1_d_tap*

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*int1_slo_no_mot*	*int1_slope*	*int1_high*	*int1_low*

int1_flat: Mapping of flat detection interrupt to INT1 pin is
0: disabled
1: enabled

int1_orient: Mapping of orientation recognition interrupt to INT1 pin is
0: disabled
1: enabled

int1_s_tap: Mapping of single tap detection interrupt to INT1 pin is
0: disabled
1: enabled

int1_d_tap: Mapping of double tap detection interrupt to INT1 pin is
0: disabled
1: enabled

int1_slo_no_mot: Mapping of no-motion / slow motion interrupt to INT1 pin is
0: disabled
1: enabled

int1_slope: Mapping of slope / any-motion interrupt to INT1 pin is
0: disabled
1: enabled

int1_high: Mapping of high-g interrupt to INT1 pin is
0: disabled
1: enabled

int1_low: Mapping of low-g interrupt to INT1 pin is
0: disabled
1: enabled

6.2.23 Register 0x1A (INT_MAP_1)

This register controls if the new data interrupt is mapped to the INT1 and/or INT2 pin.

Name	**0x1A (INT_MAP_1)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*int2_data*	*reserved*		

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*reserved*			*int1_data*

int2_data: Mapping of new data interrupt to INT2 pin is
0: disabled
1: enabled

int1_data: Mapping of new data interrupt to INT1 pin is
0: disabled
1: enabled

reserved: Write 0

6.2.24 Register 0x1B (INT_MAP_2)

This register controls which of the interrupts flat detection, orientation recognition, tap detection, no-motion / slow-motion, slope / any-motion, high-g and low-g are mapped to the INT2 pin.

Name	**0x1B (INT_MAP_2)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*int2_flat*	*int2_orient*	*int2_s_tap*	*int2_d_tap*

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*int2_slo_no_mot*	*int2_slope*	*int2_high*	*int2_low*

int2_flat: Mapping of flat detection interrupt to INT2 pin is
0: disabled
1: enabled

int2_orient: Mapping of orientation recognition interrupt to INT2 pin is
0: disabled
1: enabled

int2_s_tap: Mapping of single tap detection interrupt to INT2 pin is
0: disabled
1: enabled

int2_d_tap: Mapping of double tap detection interrupt to INT2 pin is
0: disabled
1: enabled

int2_slo_no_mot: Mapping of no-motion / slow motion interrupt to INT2 pin is
0: disabled
1: enabled

int2_slope: Mapping of slope / any-motion interrupt to INT2 pin is
0: disabled
1: enabled

int2_high: Mapping of high-g interrupt to INT2 pin is
0: disabled
1: enabled

int2_low: Mapping of low-g interrupt to INT2 pin is
0: disabled
1: enabled

6.2.25 Register 0x1E (INT_SRC)

This register controls the data source definition for interrupts with selectable data source.

Name	**0x1E (INT_SRC)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*reserved*		*int_src_data*	*int_src_tap*

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*int_src_slo_no_mot*	*int_src_slope*	*int_src_high*	*int_src_low*

int_src_data: Data for new data interrupt are
0: filtered
1: unfiltered

int_src_tap: Data for tap detection interrupt are
0: filtered
1: unfiltered

int_src_slo_no_mot: Data for no-motion / slow-motion interrupt are
0: filtered
1: unfiltered

int_src_slope: Data for slope / any-motion interrupt are
0: filtered
1: unfiltered

int_src_high: Data for high-g interrupt are
0: filtered
1: unfiltered

int_src_low: Data for low-g interrupt are
0: filtered
1: unfiltered

reserved: Write 0

6.2.26 Register 0x20 (INT_OUT_CTRL)

This register controls the electrical behavior and configuration of the interrupt pins.

Name	**0x20 (INT_OUT_CTRL)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*reserved*			

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	1	0	1
Content	*int2_od*	*int2_lvl*	*int1_od*	*int1_lvl*

int2_od: Behavior for the INT2 pin is
0: push-pull
1: open drain

int2_lvl: Level for the INT2 pin is
0: active low
1: active high

int1_od: Behavior for the INT1 pin is
0: push-pull
1: open drain

int1_lvl: Level for the INT1 pin is
0: active low
1: active high

reserved: Write 0

6.2.27 Register 0x21 (INT_RST_LATCH)

This register contains the interrupt reset bit and the interrupt mode selection.

Name	**0x21 (INT_RST_LATCH)**			
Bit	7	6	5	4
Read/Write	W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*reset_int*	*reserved*		

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*latch_int* <3:0>			

reset_int: Latched interrupts are
0: kept active
1: cleared

latch_int <3:0>: Interrupt mode

0000: non-latched
0001: temporary, 250 ms
0010: temporary, 500 ms
0011: temporary, 1 s
0100: temporary, 2 s
0101: temporary, 4 s
0110: temporary, 8 s
0111: latched
1000: non-latched
1001: temporary, 250 µs
1010: temporary, 500 µs
1011: temporary, 1 ms
1100: temporary, 12.5 ms
1101: temporary, 25 ms
1110: temporary, 50 ms
1111: latched

reserved: Write 0

6.2.28 Register 0x22 (INT_0)

This register contains the delay time definition of the low-g interrupt.

Name	0x22 (INT_0)			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*low_dur* <7:4>			

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	1	0	0	1
Content	*low_dur* <3:0>			

low_dur <7:0>: Low-g interrupt trigger delay

delay = (*low_dur* <7:0> + 1) · 2 ms

The range is from 2 ms to 512 ms, with a default setting of 20 ms.

6.2.29 Register 0x23 (INT_1)

This register contains the threshold definition of the low-g interrupt.

Name	0x23 (INT_1)			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	1	1
Content	*low_th* <7:4>			

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*low_th* <3:0>			

low_th <7:0>: Low-g interrupt trigger threshold

Scaling of 1 LSB of *low_th* (irrespective of g-range setting):
threshold = *low_th* <7:0> · 7.81 mg

The range is from 0 g to 1.992 g, with a default setting of 375 mg.

6.2.30 Register 0x24 (INT_2)

This register contains the low-g interrupt mode selection, the low-g interrupt hysteresis setting and the high_g interrupt hysteresis setting.

Name	**0x24 (INT_2)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	1	0	0	0
Content	*high_hy* <1:0>		*reserved*	

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	1
Content	*reserved*	*low_mode*	*low_hy* <1:0>	

high_hy <1:0>: Hysteresis of the high-g interrupt

Scaling of 1 LSB of *high_hy*:
2 g-range: hysteresis = *high_hy* <1:0> · 125 mg
4 g-range: hysteresis = *high_hy* <1:0> · 250 mg
8 g-range: hysteresis = *high_hy* <1:0> · 500 mg
16 g-range: hysteresis = *high_hy* <1:0> · 1000 mg

low_mode: Low-g interrupt is in
0: single mode
1: sum mode

low_hy <1:0>: Hysteresis of the low-g interrupt

Scaling of 1 LSB of *low_hy* (irrespective of g-range setting):
hysteresis = *low_hy* <1:0> · 125 mg

reserved: Write 0

6.2.31 Register 0x25 (INT_3)

This register contains the delay time definition of the high-g interrupt.

Name	0x25 (INT_3)			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*high_dur* <7:4>			

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	1	1	1	1
Content	*high_dur* <3:0>			

high_dur <7:0>: High-g interrupt trigger delay

delay = (*high_dur* <7:0> + 1) · 2 ms

The range is from 2 ms to 512 ms, with a default setting of 32 ms.

6.2.32 Register 0x26 (INT_4)

This register contains the threshold definition of the high-g interrupt.

Name	0x26 (INT_4)			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	1	1	0	0
Content	*high_th* <7:4>			

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*high_th* <3:0>			

high_th <7:0>: High-g interrupt trigger threshold

Scaling of 1 LSB of *high_th*:
2 g-range: threshold = *high_th* <7:0> · 7.81 mg
4 g-range: threshold = *high_th* <7:0> · 15.63 mg
8 g-range: threshold = *high_th* <7:0> · 31.25 mg
16 g-range: threshold = *high_th* <7:0> · 62.5 mg

6.2.33 Register 0x27 (INT_5)

This register contains the definition of the number of samples to be evaluated for the slope / any-motion interrupt and the no-motion / slow-motion interrupt trigger delay.

Name	**0x27 (INT_5)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*slo_no_mot_dur* <5:2>			

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*slo_no_mot_dur* <1:0>		*slope_dur* <1:0>	

slow_no_mot_dur <5:0>: Function depends on whether the slow-motion or the no-motion interrupt function has been selected via *slo_no_mot_sel*.

Slow-motion interrupt function (*slo_no_mot_sel* = 0):
N = *slo_no_mot_dur* <1:0> +1 consecutive slope data points must be above the slow-motion threshold (*slo_no_mot_th*) in order to trigger the interrupt.

No-motion interrupt function (*slo_no_mot_sel* = 1):
slo_no_mot_dur <5:0> defines the time for which no slope data points must exceed the no-motion threshold (*slo_no_mot_th*) in order to trigger the interrupt.
The delay time in seconds may be calculated with the following equations.
slo_no_mot_dur <5:4> = 00
➔ delay time = *slo_no_mot_dur* <3:0> + 1
slo_no_mot_dur <5:4> = 01
➔ delay time = *slo_no_mot_dur* <3:0> · 4 + 20
slo_no_mot_dur <5> = 1
➔ delay time = *slo_no_mot_dur* <4:0> · 8 + 88

slope_dur <1:0>: The slope / any-motion interrupt triggers if *N* = *slope_dur* <1:0> + 1 consecutive slope data points are above the slope / any-motion interrupt threshold *slope_th*.

6.2.34 Register 0x28 (INT_6)

This register contains the threshold definition of the slope / any-motion interrupt.

Name	0x28 (INT_6)			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	1
Content	*slope_th* <7:4>			

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	1	0	0
Content	*slope_th* <3:0>			

slope_th <7:0>: Slope / any-motion interrupt trigger threshold

Scaling of 1 LSB of *slope_th*:
2 g-range: threshold = *slope_th* <7:0> · 3.91 mg
4 g-range: threshold = *slope_th* <7:0> · 7.81 mg
8 g-range: threshold = *slope_th* <7:0> · 15.63 mg
16 g-range: threshold = *slope_th* <7:0> · 31.25 mg

6.2.35 Register 0x29 (INT_7)

This register contains the threshold definition of the no-motion / slow-motion interrupt.

Name	0x29 (INT_7)			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	1
Content	*slo_no_mot_th* <7:4>			

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	1	0	0
Content	*slo_no_mot_th* <3:0>			

slo_no_mot_th <7:0>: No-motion / slow-motion interrupt trigger threshold

Scaling of 1 LSB of *slo_no_mot_th*:
2 g-range: threshold = *slo_no_mot_th* <7:0> · 3.91 mg
4 g-range: threshold = *slo_no_mot_th* <7:0> · 7.81 mg
8 g-range: threshold = *slo_no_mot_th* <7:0> · 15.63 mg
16 g-range: threshold = *slo_no_mot_th* <7:0> · 31.25 mg

6.2.36 Register 0x2A (INT_8)

This register contains the timing definitions for the single tap and double tap interrupt.

Name	**0x2A (INT_8)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*tap_quiet*	*tap_shock*	*reserved*	

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	1	0	0
Content	*reserved*	*tap_dur* <2:0>		

tap_quiet: Tap quiet duration is
0: 30 ms
1: 20 ms

tap_shock: Tap shock duration is
0: 50 ms
1: 75 ms

tap_dur <2:0>: Length of the time window for the second shock event for double tap detection

000: 50 ms
001: 100 ms
010: 150 ms
011: 200 ms
100: 250 ms
101: 375 ms
110: 500 ms
111: 700 ms

reserved: Write 0

6.2.37 Register 0x2B (INT_9)

This register controls the number of samples which are processed by the single tap and double tap interrupt engine after a wake-up in low power mode 1 or 2. It also defines the threshold for the single tap and double tap interrupt.

Name	**0x2B (INT_9)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*tap_samp* <1:0>		*reserved*	*tap_th* <4>

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	1	0	1	0
Content	*tap_th* <3:0>			

tap_samp <1:0>: Number of samples which are processed after a wake-up in low-power mode 1 or 2

00: 2 samples
01: 4 samples
10: 8 samples
11: 16 samples

tap_th <4:0>: Single tap and double tap interrupt threshold

Scaling of 1 LSB of *tap_th*:
2 g-range: threshold = *tap_th* <4:0> · 62.5 mg
4 g-range: threshold = *tap_th* <4:0> · 125 mg
8 g-range: threshold = *tap_th* <4:0> · 250 mg
16 g-range: threshold = *tap_th* <4:0> · 500 mg

reserved: Write 0

6.2.38 Register 0x2C (INT_A)

This register contains the definition of the hysteresis, blocking and mode for the orientation interrupt.

Name	**0x2C (INT_A)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	1
Content	*reserved*	*orient_hyst* <2:0>		

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	1	0	0	0
Content	*orient_blocking* <1:0>		*orient_mode* <1:0>	

orient_hyst <2:0>: Hysteresis of the orientation interrupt

Scaling of 1 LSB of *orient_hyst* (irrespective of g-range setting):
hysteresis = *orient_hyst* <2:0> · 62.5 mg

orient_blocking <1:0>: Blocking mode which is used for the generation of the orientation interrupt

00: no blocking
01: theta blocking or
acceleration in any axis > 1.5 g
10: theta blocking or
acceleration slope in any axis > 0.2 g or
acceleration in any axis > 1.5 g
11: theta blocking or
acceleration slope in any axis > 0.4 g or
acceleration in any axis > 1.5 g and value of *orient* is not stable for at least 100 ms

orient_mode <1:0>: Threshold for switching between the different orientations

00: symmetrical
01: high-asymmetrical
10: low-asymmetrical
11: symmetrical

reserved: Write 0

6.2.39 Register 0x2D (INT_B)

This register contains the definition of the axis orientation and the theta blocking angle for the orientation interrupt.

Name	0x2D (INT_B)			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	n/a	1	0	0
Content	*reserved*	*orient_ud_en*	*orient_theta* <5:4>	

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	1	0	0	0
Content	*orient_theta* <3:0>			

orient_ud_en: A change of *orient* <2>
1: generates an orientation interrupt
0: is ignored

orient_theta <5:0>: Defines a blocking angle between 0° and 44.8°

reserved: Write 0

6.2.40 Register 0x2E (INT_C)

This register contains the definition of the flat threshold angle for the flat interrupt.

Name	0x2E (INT_C)			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	n/a	n/a	0	0
Content	*reserved*		*flat_theta* <5:4>	

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	1	0	0	0
Content	*flat_theta* <3:0>			

flat_theta <5:0>: Defines the threshold angle for the detection of the flat position in the range from 0° to 44.8°

reserved: Write 0

6.2.41 Register 0x2F (INT_D)

This register contains the definition of the hold time and hysteresis of the flat interrupt.

Name	**0x2F (INT_D)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	1
Content	*reserved*		*flat_hold_time* <1:0>	

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	1
Content	*reserved*	*flat_hy* <2:0>		

flat_hold_time <1:0>: Delay time for which the *flat* value must remain stable in order to generate the flat interrupt

00: 0 ms
01: 512 ms
10: 1024 ms
11: 2048 ms

flat_hy <2:0>: Definition of the flat interrupt hysteresis

The flat value must change by more than twice the value of the flat interrupt hysteresis in order to detect a state change. For details, see Section 2.9.8.

000: flat detection hysteresis is disabled

reserved: Write 0

6.2.42 Register 0x32 (PMU_SELF_TEST)

This register contains the settings for the sensor self-test configuration and trigger.

Name	0x32 (PMU_SELF_TEST)			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*reserved*			

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*reserved*	*self_test_sign*	*self_test_axis* <1:0>	

self_test_sign: Sign of the self-test excitation is
1: positive
0: negative

self_test_axis <1:0>: Selects the axis to be self-tested

00: self-test disabled
01: x-axis
10: y-axis
11: z-axis

When a self-test is performed, only the acceleration data readout value of the selected axis is valid; after the self-test has been enabled, a delay of at least 50 ms is necessary for the read-out value to settle.

reserved: Write 0

6.2.43 Register 0x34 (BGW_WDT)

This register contains settings for the TWI watchdog timer.

Name	0x34 (BGW_WDT)			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*reserved*			

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*reserved*	*i2c_wdt_en*	*i2c_wdt_sel*	*reserved*

i2c_wdt_en: Watchdog timer at the SDA pin in TWI mode is
0: disabled
1: enabled

i2c_wdt_sel: Watchdog timer period is
0: 1 ms
1: 50 ms

reserved: Write 0

6.2.44 Register 0x36 (OFC_CTRL)

This register contains control signals and configuration settings for the fast offset compensation.

Name	**0x36 (OFC_CTRL)**			
Bit	7	6	5	4
Read/Write	W	W	W	R
Reset Value	0	0	0	0
Content	*offset_reset*	*cal_trigger* <1:0>		*cal_rdy*

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*reserved*			

offset_reset: The values of all offset compensation registers (0x38 to 0x3A) are
0: kept
1: reset to 0

offset_trigger <1:0>: Trigger fast offset compensation

00: disabled
01: x-axis
10: y-axis
11: z-axis

The offset compensation must not be triggered when *cal_rdy* is 0.

cal_rdy: Offset compensation is
0: in progress
1: ready to be retriggered

reserved: Write 0

6.2.45 Register 0x38 (OFC_OFFSET_X)

This register contains the offset compensation value for x-axis acceleration readout data.

Name	0x38 (OFC_OFFSET_X)			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*offset_x* <7:4>			

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*offset_x* <3:0>			

offset_x <7:0>: Offset value which is subtracted from the internal filtered and unfiltered x-axis acceleration data

The offset value is represented in two's complement notation with the following mapping:
+127 LSB -> +0.922 g
0 LSB -> 0 g
-128 LSB -> -1 g

offset_x <7:0> is generated automatically after triggering the fast offset compensation for the x-axis. However, it may also be written directly by the user.

6.2.46 Register 0x39 (OFC_OFFSET_Y)

This register contains the offset compensation value for y-axis acceleration readout data.

Name	0x39 (OFC_OFFSET_Y)			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*offset_y* <7:4>			

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*offset_y* <3:0>			

offset_y <7:0>: Offset value which is subtracted from the internal filtered and unfiltered y-axis acceleration data

The offset value is represented in two's complement notation with the following mapping:
+127 LSB -> +0.922 g
0 LSB -> 0 g
-128 LSB -> -1 g

offset_y <7:0> is generated automatically after triggering the fast offset compensation for the y-axis. However, it may also be written directly by the user.

6.2.47 Register 0x3A (OFC_OFFSET_Z)

This register contains the offset compensation value for z-axis acceleration readout data.

Name	**0x3A (OFC_OFFSET_Z)**			
Bit	7	6	5	4
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*offset_z* <7:4>			

Bit	3	2	1	0
Read/Write	R/W	R/W	R/W	R/W
Reset Value	0	0	0	0
Content	*offset_z* <3:0>			

offset_z <7:0>: Offset value which is subtracted from the internal filtered and unfiltered z-axis acceleration data

The offset value is represented in two's complement notation with the following mapping:
+127 LSB -> +0.922 g
0 LSB -> 0 g
-128 LSB -> -1 g

offset_z <7:0> is generated automatically after triggering the fast offset compensation for the z-axis. However, it may also be written directly by the user.

7 Handling and Storage

Sensors with visible damages (housing, connectors, pins, etc.) and sensors which might have exceeded the absolute maximum ratings (e.g. dropped down from a height of more than 1.8 m onto a hard surface) must not be mounted in the vehicle. These sensors must be scrapped.

7.1 Moisture Sensitivity Level (MSL)

The moisture sensitivity level (MSL) of BOSCH SMA130 corresponds to JEDEC Level 1, see also

- IPC/JEDEC J-STD-020C "Joint Industry Standard: Moisture/Reflow Sensitivity Classification for non-hermetic Solid State Surface Mount Devices"
- IPC/JEDEC J-STD-033A "Joint Industry Standard: Handling, Packing, Shipping and Use of Moisture/Reflow Sensitivity Surface Mount Devices"

The sensor IC fulfils the lead-free soldering requirements of the above-mentioned IPC/JEDEC standard, i.e., reflow soldering with a peak temperature up to 260°C.

7.2 Mounting Recommendations

MEMS sensors in general are high-precision measurement devices which consist of electronic as well as mechanical structures. BOSCH sensor devices are designed for precision, efficiency and mechanical robustness.

However, in order to achieve best possible results of your design, the following recommendations should be taken into consideration when mounting the sensor on a printed circuit board (PCB).

In order to evaluate and optimize the considered placement position of the sensor on the PCB it is recommended to use additional tools during the design in phase, e.g.:

- Regarding thermal aspects: infrared camera
- Regarding mechanical stress: warpage measurements and/or FEM-simulations
- Regarding shock robustness: drop test of the devices after soldering on the target application PCB

Recommendations in Detail

- It is recommended to keep a reasonable distance between the sensor mounting location on the PCB and the critical points described in the following examples. The exact value for a "reasonable distance" depends on many customer specific variables and must therefore be determined case by case.
- It is not recommended to place the sensor directly under or next to push-button contacts as this can result in mechanical stress.
- It is not recommended to place the sensor in direct vicinity of extremely hot spots regarding temperature (e.g. a µController or a graphic chip) as this can result in heating up the PCB and consequently also the sensor.
- It is not recommended to place the sensor in direct vicinity of a mechanical stress maximum (e.g. in the center of a diagonal crossover). Mechanical stress can lead to bending of the PCB and the sensor.

- Do not mount the sensor too closely to a PCB anchor point where the PCB is attached to a shelf (or similar) as this could also result in mechanical stress. To reduce potential mechanical stress, minimize redundant anchor points and/or loosen respective screws.
- Avoid mounting the sensor in areas where resonant amplitudes (vibrations) of the PCB are likely or to be expected.
- Please avoid partial coverage of the sensor by any kind of (epoxy) resin, as this can possibly result in mechanical stress.
- Avoid mounting (and operation) of the sensor in the vicinity of strong magnetic, strong electric and/or strong infrared radiation fields (IR).
- Avoid electrostatic charging of the sensor and of the device in which the sensor is mounted.

In case you have any questions regarding the mounting of the sensor on your PCB or the evaluation and/or optimization of the considered placement position of the sensor on your PCB, do not hesitate to contact us.

If the above mentioned recommendations cannot be realized appropriately, a specific in-line offset calibration after placement of the device onto your PCB might help to minimize potentially remaining effects.

7.3 Soldering Guidelines

Repair and manual soldering of the sensor is not permitted.

7.3.1 Reflow Soldering Recommendation for Sensors in LGA Package

Please make sure that the edges of the LGA substrate of the sensor are free of solder material. Avoid solder material forming a high meniscus covering the edge of the LGA substrate (see Figure 7-1).

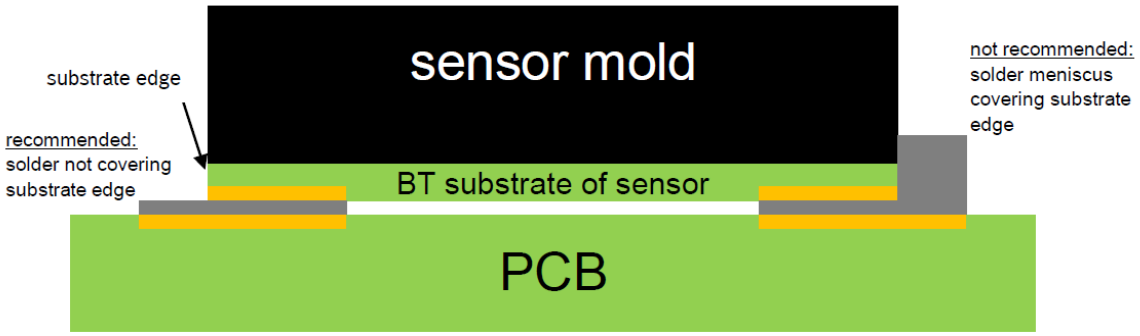


Figure 7-1: Reflow soldering recommendation.

7.3.2 Classification Reflow Profiles

Profile Feature	Pb-Free Assembly
Average ramp-up rate (Ts_{max} to Tp)	3 °C/s max.
Preheat - Temperature min (Ts_{min}) - Temperature max (Ts_{max}) - Time (ts_{min} to ts_{max})	 150 °C 200 °C 60 s – 80 s
Time maintained above: - Temperature (T_L) - Time (t_L)	 217 °C 60 s – 150 s
Peak classification temperature (Tp)	260 °C
Time within 5 °C of actual peak	20 s – 40 s
Ramp-down rate	6 °C/s max.
Time 25 °C to peak temperature	8 min max.

Note: All temperatures refer to the topside of the package, measured on the package body surface.

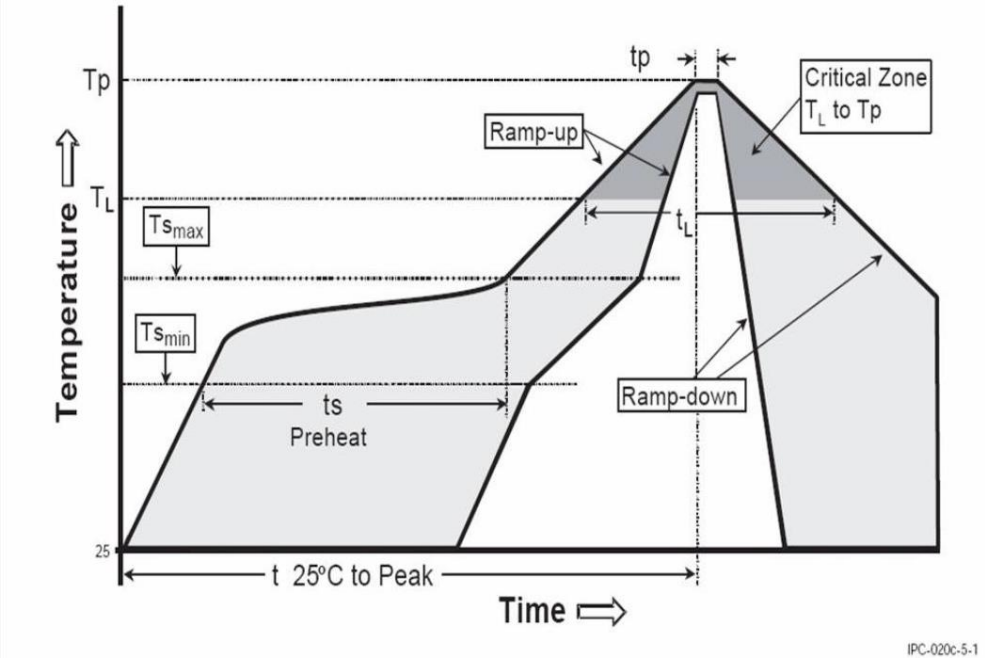
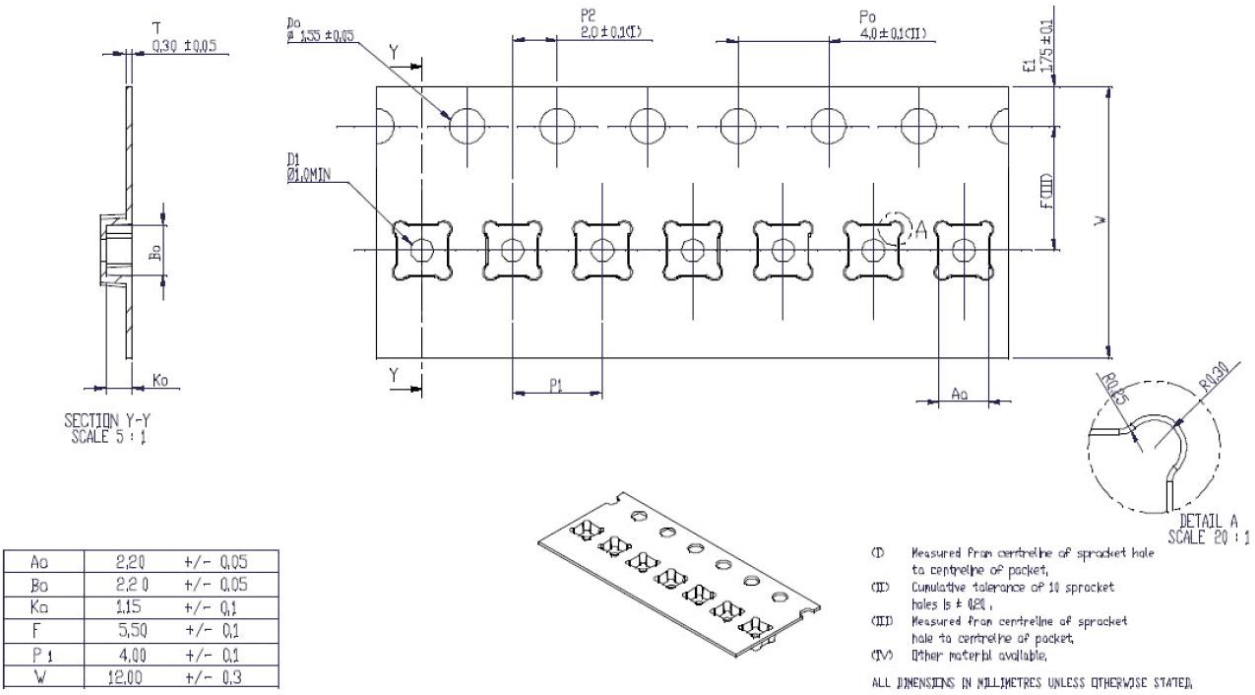


Figure 7-2: Soldering profile.

7.4 Tape on Reel

7.4.1 Tape on Reel Specification

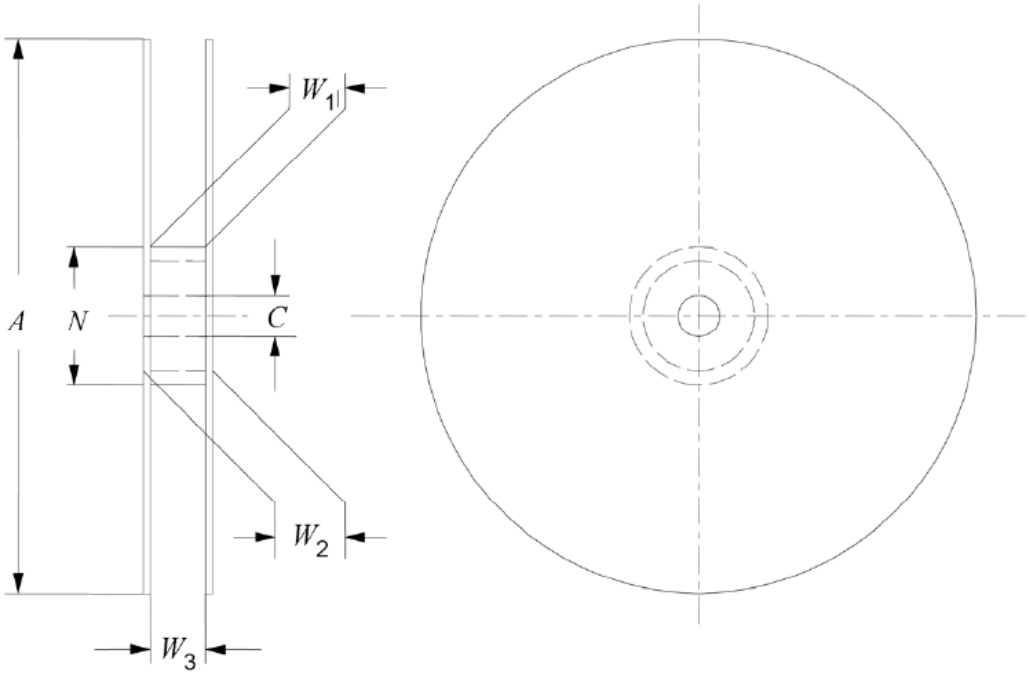
The SMA130 is shipped in a standard cardboard box.
The box dimensions for one reel are L x W x H = 35 cm x 35 cm x 6 cm.
SMA130 quantity: 10000 pieces per reel. Please handle with care.



Ao	2,20	+/- 0,05
Bo	2,20	+/- 0,05
Ko	1,15	+/- 0,1
F	5,50	+/- 0,1
P 1	4,00	+/- 0,1
W	12,00	+/- 0,3

Figure 7-3: Tape dimensions in mm.

Reel dimensions:



Parameter	Meaning	Dimensions [mm]
W (not depicted)	tape width	12
A	reel diameter	330
N	hub diameter	100
W_1	inner width of reel	12.4 +2
W_2	total width of reel	18.4
W_3, min	inner width of reel, minimum	11.9
W_3, max	inner width of reel, maximum	15.4

Details on hub hole dimension C:

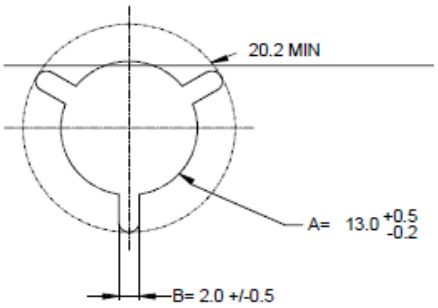


Figure 7-4: Reel dimensions in mm.

7.4.2 Orientation within the Reel

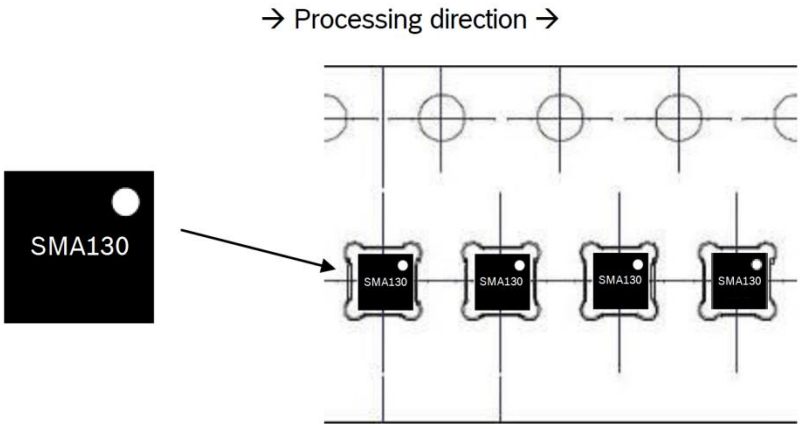


Figure 7-5: Orientation of the SMA130 devices relative to the tape.

7.5 Further Important Mounting and Assembly Recommendations

The SMA130 is designed to sense accelerations with high accuracy even at low amplitudes and contains highly sensitive structures inside the sensor element. The MEMS sensor can tolerate mechanical shocks up to several thousand g's. However, these limits might be exceeded in conditions with extreme shock loads such as e.g. hammer blow on or next to the sensor, dropping the sensor onto hard surfaces etc.

We strongly recommend to avoid any g forces beyond the limits specified in the data sheet during transport, handling and mounting of the sensors in a defined and qualified installation process.

This device has built-in protections against high electrostatic discharges or electric fields (2 kV HBM); however, anti-static precautions should be taken as for any other CMOS component.

Unless otherwise specified, proper operation can only occur when all terminal voltages are kept within the supply voltage range. Unused inputs must always be connected to a defined logic voltage level.

8 Test Specifications

8.1 Environmental Safety

The SMA130 sensor meets the requirements of the EC restriction of hazardous substances (RoHS) directive, see also:

Directive 2002/95/EC of the European Parliament and of the Council of 27 January 2003 on the restriction of the use of certain hazardous substances in electrical and electronic equipment.

Halogen content

The SMA130 is halogen-free. For more details on the analysis results, please contact your Bosch representative.

8.2 Qualification

The SMA130 passed the following qualification: AEC-Q100 grade 3.

9 Legal Disclaimer

Assessment of products returned from field

Returned products are considered good if they fulfill the specifications / test data for 0-mileage and field listed in this document.

Engineering Samples

Engineering samples are marked with (e) or (E). Samples may vary from the valid technical specifications of the series product contained in this data sheet. Therefore, they are not intended or fit for resale to third parties or for use in end products. Their sole purpose is internal client testing. The testing of an engineering sample may in no way replace the testing of a series product. Bosch assumes no liability for the use of engineering samples. The purchaser shall indemnify Bosch from all claims arising from the use of engineering samples.

Product Usage

The SMA130 is tested and qualified according to Section 8. The SMA130 only has to be used within the parameters of this product data sheet. In particular, the SMA130 is not fit for use in life-sustaining or safety sensitive systems. Safety sensitive systems are those for which a malfunction may lead to bodily harm or significant property damage. The resale and/or use of products are at the purchaser's own risk and responsibility. The examination of the SMA130 is the sole responsibility of the purchaser.

The purchaser shall indemnify Bosch from all third party claims arising from any product use not covered by the parameters of this product data sheet or not approved by Bosch and reimburse Bosch for all costs in connection with such claims.

The purchaser must monitor the market for the purchased products, particularly with regard to product safety, and inform Bosch without delay of all security relevant incidents.

Application Examples and Hints

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