Wearable feature set
Application Note

Application Note – Wearable feature set

Document revision	1.0
Document release date	August 2019
Document number	BST-MAS-AN032-00
Document valid for	BMA423, BMA425, BMA456
Notes	Data and descriptions in this document are subject to change without notice. Product photos and pictures are for illustration purposes only and may differ from the real product appearance **Confidential and under NDA**

Index of Contents

1. Wearable feature set

1.1. Feature interrupt engine

Global configuration

The Register FEATURES_IN provides description of configuration parameters of the feature interrupt engine. To reconfigure the features, user must perform a burst read of the full content of the Register FEATURES_IN, followed by a modification of the content, and finally a burst write of the modified content to the Register FEATURES_IN. The content of the successive bytes read or written in burst mode correspond to the single bytes 0x00 to 0x3F described in FEATURES_IN.

Prior to the reconfiguration of the feature/s, successful initialization of sensor must be ensured (see description in chapter 1.)

Read the status of the feature/s interrupt from the register listed below.

Feature interrupt status registers

Feature	Output status
Single-tap	INT_STATUS_0. single_tap_out
Step detector	INT_STATUS_0.step_counter_out
Activity recognition	INT_STATUS_0.activity_type_out
Wrist ware wakeup	INT_STATUS_0.wrist_wear_wakeup_out
Double-tap	INT_STATUS_0.double_tap_out
Any-motion	INT_STATUS_0.any_motion_out
No-motion	INT_STATUS_0.no_motion_out
Error Interrupt	INT_STATUS_0.error_int_out

The interrupt status register also contains a bit for error interrupt, which indicates that the sensor have been stopped due to a fatal error. In such condition, the sensor needs to be re-initialized to ensure correct functioning of the device.

For step detector and counter, in addition to an interrupt for each step detection, the number of steps that user took since the start of feature engine is stored in the registers: STEP_COUNTER_0, STEP_COUNTER_1, STEP_COUNTER_2 and STEP_COUNTER_3.

Accelerometer operation mode

When the accelerometer is configured to be in performance mode (ACC_CONF.acc_perf_mode is 0b1), consecutive samples are measured at maximum output data rate (ODR) with samples equi-spaced in time. The processing of the feature engine carried-out correctly ensuring the functional performance. When the operational mode of accelerometer is set to Performance mode, user ODR and bandwidth settings has no influence on processing of feature engine.

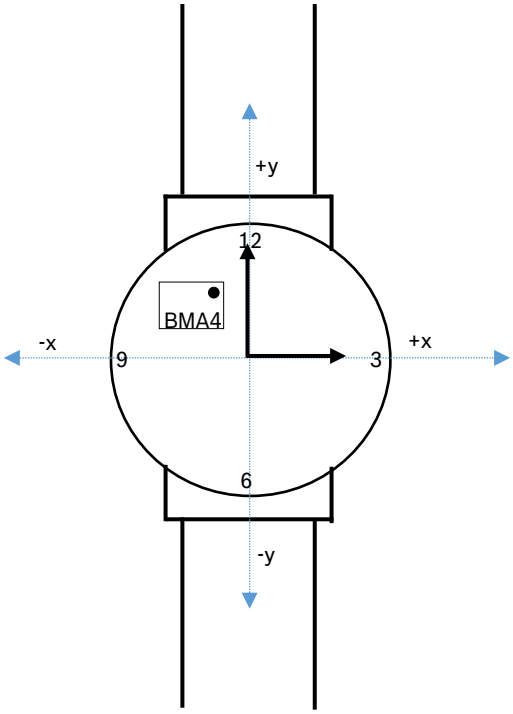
When the configuration of the accelerometer is set to duty-cycling mode, i.e. performance mode is disabled (ACC_CONF.acc_perf_mode is 0b0), consecutive samples may not be measured at maximum ODR and are not equi-spaced in time. For feature engine to work correctly, user ODR and/or bandwidth settings must comply with the following requirements:

1. The user ODR shall be >= 50Hz when single-/double-tap is disabled
2. The user ODR shall be >= 200Hz when single-/double-tap is enabled

In case of non-fulfillment of the user ODR requirement, the corresponding error flag is set in the register INTERNAL_STATUS.

Axes remapping for interrupt features

The interrupt feature engine assumes that the sensor co-ordinate system is in alignment with the device co-ordinate system. The assumed default device co-ordinate system is as depicted in the figure below.



If the sensor placement in the user system is different from the one depicted in the figure above, to ensure the correct functioning of the features in the interrupt engine, appropriate axes remapping needs to be applied.

The axes remapping configuration register allows the remapping of each sensing axis to any other axis. Along with the remapping of the sensing axis, it also allows the inversion of the axis. An error interrupt is triggered when more than one axis are remapped to same axis, E.g., x = x, y = x, z = z.

Note:
The axes remapping is applied only on the acceleration signal input to the sensor features. The DATA_0 to DATA_13 registers and FIFO are unmodified by the settings of axes remapping. If required, appropriate remapping should be applied on the driver level.

Configuration settings:

1. FEATURES_IN.general_settings.axes_remapping.map_x_axis – describes which axis shall be mapped to x axis.
2. FEATURES_IN.general_settings.axes_remapping.map_x_axis_sign – describes whether the mapped axis shall be inverted or not to be inverted.
3. FEATURES_IN.general_settings.axes_remapping.map_y_axis – describes which axis shall be mapped to y axis.
4. FEATURES_IN.general_settings.axes_remapping.map_y_axis_sign – describes whether the mapped axis shall be inverted or not to be inverted.
5. FEATURES_IN.general_settings.axes_remapping.map_z_axis –describes which axis shall be mapped to z axis.
6. FEATURES_IN.general_settings.axes_remapping.map_z_axis_sign – describes whether the mapped axis shall be inverted or not to be inverted.

Step Detector and Counter

Step detector feature performs the detection of step taken by the user while walking, running, stair walking, etc. using the tri-axial acceleration signal. Realization of the Step detection functionality is in compliance to the requirements of Android 4.4 and above as described under https://source.android.com/devices/sensors/sensor-types.html#step_detector to report the step detect with low latency. Precise optimization of the parameters defining the functional behaviour of the step detector algorithm for the device position on the user body as "Wrist" assures the best step detection performance.

Step counter extends the functionality of the step detector by implementation of additional step validation approach and hence yielding the higher accuracy at the cost of slight increase in reporting delay. Step counter implementation comply with the requirements of Android 4.4 and higher stated under https://source.android.com/devices/sensors/sensor-types.html#step_counter. The approach of step validation rejects or accepts the detected step as legitimate step for counting by analyzing the regularity of occurrence and dynamicity of the scenario.

Step detection and counting features are facilitated to work in parallel with full independence. However, the feature interrupt engine reports the status of step detection and watermark interrupt of step counter on the multiplexed interrupt line. Hence, the functionalities of step detection and watermark interrupt for step counter are mutually exclusive.

Step Counter:
Step counter feature can be activated by setting the FEATURES_IN.step_counter.settings_26.en_counter. The step counter reports the number of steps taken by the user since the first activation of the feature without any reset. The step counts is reported as 32-bit unsigned integer number stored in registers STEP_COUNTER_0, STEP_COUNTER_1, STEP_COUNTER_2 and STEP_COUNTER_3. The sum of step detector events may differ from the step count value due to additional step validation by post-processing.

The step count value is reset to zero by enabling the FEATURES_IN.step_counter.settings_26.reset_counter flag. On resetting the step count value, the flag is automatically cleared and counting is restarted.

The watermark interrupt for step counter is useful when the host needs to receive an interrupt every time the user took a certain fixed number of steps. The step counter watermark interrupt is enabled by setting FEATURES_IN.step_counter.settings_26.en_detector to 0 and FEATURES_IN.step_counter.settings_26.watermark_level to greater than 0. E.g., when FEATURES_IN.step_counter.settings_26.watermark_level is set to 10 (holding an implicit factor of 20x), for every 200 steps are elapsed an interrupt will be raised on INT_STATUS_0.step_counter_out.

Step Detector:
Step Detector feature reports the detected step of the user while walking, running, etc., with low latency. If FEATURES_IN.step_counter.settings_26.en_detector is set, an interrupt is triggered on INT_STATUS_0.step_counter_out for every step detected.

The table below provides the default configuration values for a wearable optimized step detector and counter.

Parameter Name	Wrist Configuration
PARAM_1	301
PARAM_2	31700
PARAM_3	315
PARAM_4	31451
PARAM_5	4
PARAM_6	31551
PARAM_7	27853
PARAM_8	1219
PARAM_9	2437
PARAM_10	1219
PARAM_11	-6420
PARAM_12	17932
PARAM_13	1
PARAM_14	39
PARAM_15	25
PARAM_16	150
PARAM_17	160
PARAM_18	1
PARAM_19	12
PARAM_20	15600
PARAM_21	256
PARAM_22	1
PARAM_23	3
PARAM_24	1
PARAM_25	14

The feature is by default initialized with the configuration optimized for wrist to use. For modifying the configuration, the steps provided below must to be followed:

1. Disable step counter, step detector, and activity detection
2. Modify the 25 parameters of step counter
3. Enable step counter, step detector, and activity detection

After re-enabling the features, the new configuration parameters will be applied for step detection and counting.

Customized Step Counter Sensitivity Configuration (overwrites Step Counter Presets)

The Step detector and counter performance can be optimized to any specific use-case scenarios by re-determining and manually re-configuring the parameters in the register map with the support of the corresponding field application engineer. The default parameters are set to wrist configuration during the device initialization.

Configuration settings:

1. FEATURES_IN.step_counter.settings_26.watermark_level - A step-counter watermark interrupt is triggered for every time user takes 20 times watermark_level number of steps. The range is 20 to 20460 steps, with resolution of 20 steps. When set to 0, the Step counter watermark functionality is disabled.
2. FEATURES_IN.step_counter.settings_26.reset_counter – Flag to reset the number of steps counted. The reset is applicable only if step counter feature is enabled at-least once and steps count is greater than zero.
3. FEATURES_IN.step_counter.settings_26.en_counter – enable the Step Counter feature.
4. FEATURES_IN.step_counter.settings_26.en_detector – enable the Step Detector feature.
5. FEATURES_IN.step_counter.settings_1.param_1 to param_25 – there are 25 parameters, which configure the use case (wrist or phone).

Activity Classification

The feature can classify the locomotion state of the user. Still, walking and running are the locomotion states that the feature can distinguish. Every change of the activity is reported by an interrupt on INT_STATUS_0.activity_type_out. The actual activity/locomotion state is updated in the register ACTIVITY_TYPE.activity_type_out. The activity classification feature is enabled by setting FEATURES_IN.step_counter.settings_26.en_activity.

Configuration settings:

FEATURES_IN.step_counter.settings_26.en_activity – enables activity classification.

Wrist Wear Wakeup

Wrist wear wakeup feature is designed to detect any natural movement of the user arm to see the dial of watch when wearing a classical wristwatch. Feature is intended to be used as wakeup gesture (i.e. for triggering screen-on or screen-off) in wrist wearable devices.

The feature classifies if the arm is in one of the following two positions:

- Focus position: In this position, the user arm in front of the body and the user is able to look comfortably at the watch dial.
- Non-focus position: In this position, the user is not able to look at the watch dial.

The positive use-case scenarios for the wrist wear wakeup gesture are described in the table below.

Environment	Scenario	Device initial position	User movement
Outdoor / Indoor / train / bus	Walking	Arm swinging / hand in pocket	Lifts and brings the arm in-front of the body to be able to comfortably look at the watch dial
Outdoor / Indoor	Walking / Running / Jogging	Arm swinging	Lift and bring the arm in-front of the body to be able to comfortably look at the watch dial
Outdoor / Indoor / train / bus	Sitting / Standing	Arm is down on side of the body / hand in pocket	Lifts and brings the arm in-front of the body to be able to comfortably look at the watch dial
		Arm is in-front of the body	Rolls the wrist towards the user to look at the watch dial
Indoor / train	Working with computer	Arm on table or arm rest	Rolls the wrist towards the user to look at the watch dial

Intended positive use-case scenarios of wrist wear wakeup feature.

Reliable functioning of Wrist wear wakeup is dependent on the sensor placement in the user system. Implementation of the feature assumes that the sensor co-ordinate frame is in alignment with the end-device/system co-ordinate frame. The depiction of assumed default device/system co-ordinate frame and required relative placement of the sensor is shown in the figure below “Axes remapping” section under Global configuration. Refer to the section on axes remapping to apply the correct remapping, if required.

Configuration settings

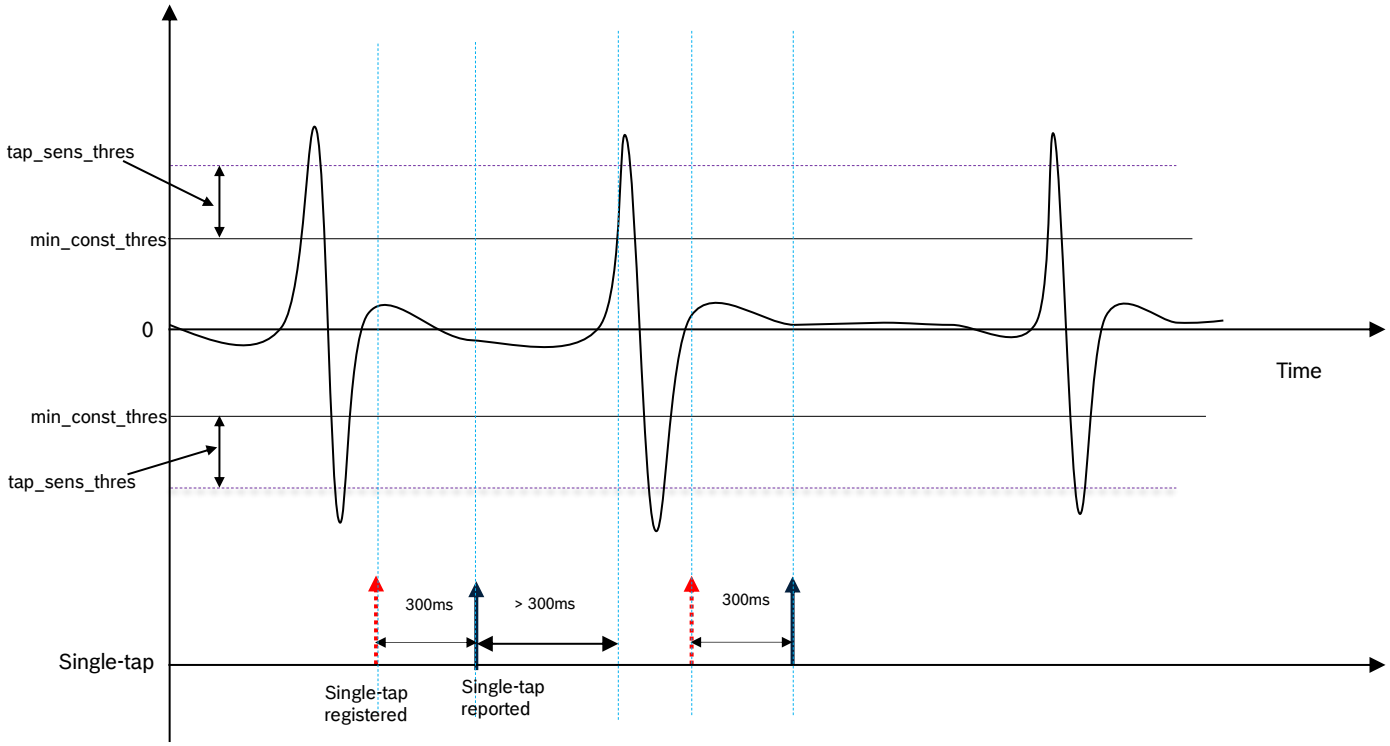
FEATURES_IN.wrist_ware_wakeup.settings.enable – enables/disables the Wrist ware wakeup feature.

Tap detector

Tap detector realizes the detection of the user interaction of tapping on the surface of the device. The detected taps on the devices depending on spread over time are classified as single-tap or double-tap gesture events. Detection of tap is performed using acceleration signal input from one of the sensing axis of the sensor. The default axis used is set to z-axis. If the direction of the tap is different from z-axis, appropriate axes remapping needs to be applied. Refer to the section on axis remapping

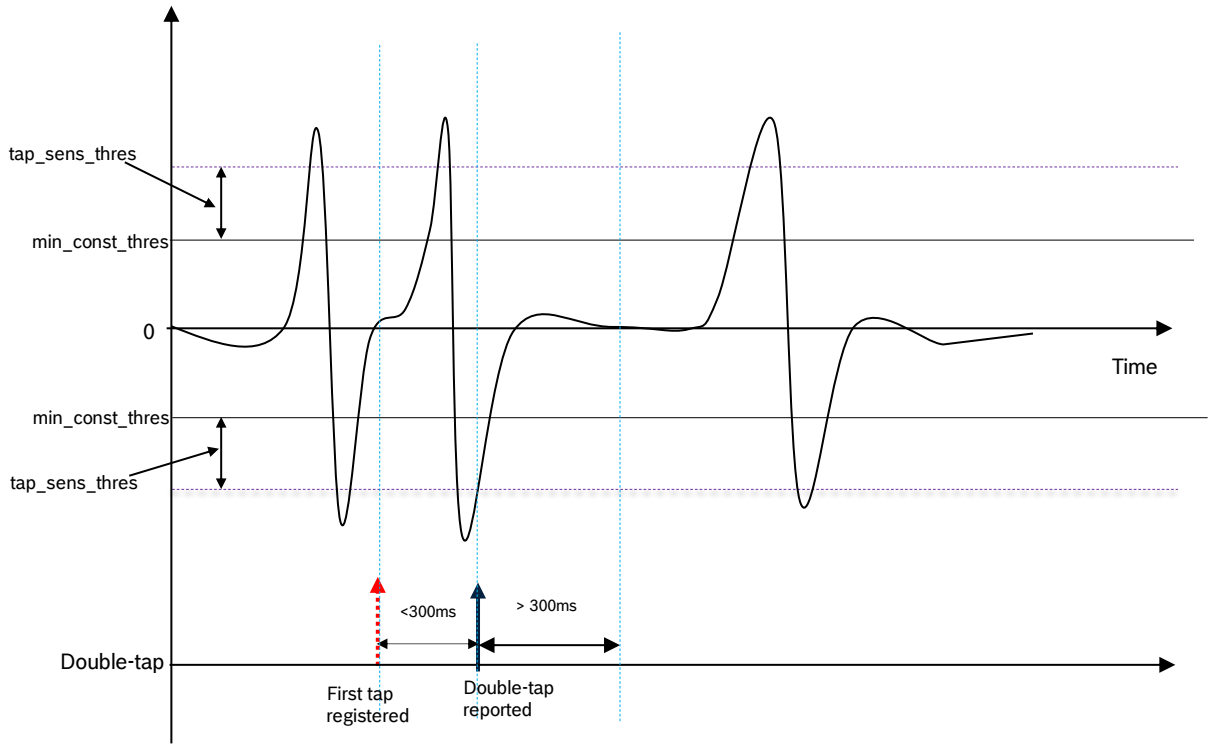
Single-tap and double-tap gesture detection is implemented as mutually exclusive events i.e. double-tap on the device does not report a single-tap event. Double-tap and single-tap events are mapped on to dedicated interrupt lines, as shown in Register (0x1C) INT_STATUS_0.

Timing diagram for single-tap:



When a single-tap gesture is performed, the single-tap event is reported with a delay of 300ms. The delay in reporting is the confirmation time to ensure that there is no tap in the vicinity. If a second tap is detected within 300ms from the first tap, the probable single tap event is nullified and double-tap events is output. On reporting a gesture event by the feature, gesture detection is inherently suspended for 300ms.

Timing diagram for double-tap:



How to perform the gesture ideally:

1. Tap on the surface of the device using one of the finger, slightly harder than just touching.
2. Tap again immediately, if “Double tap” gesture was intended.
3. Wait for 300ms before performing the next gesture.

Configuration settings- Double tap

1. FEATURES_IN.double-tap.settings.enable – Enable detection of double-tap gesture.
2. FEATURES_IN.double-tap.settings.sensitivity – configures detection sensitivity, the range goes from 0 (high sensitive) to 7 (low sensitive).

Configuration settings- Single tap

1. FEATURES_IN.single-tap.settings.enable – enable single tap.
2. FEATURES_IN.single-tap.settings.sensitivity – configures detection sensitivity, the range goes from 0 (high sensitive) to 7 (low sensitive).

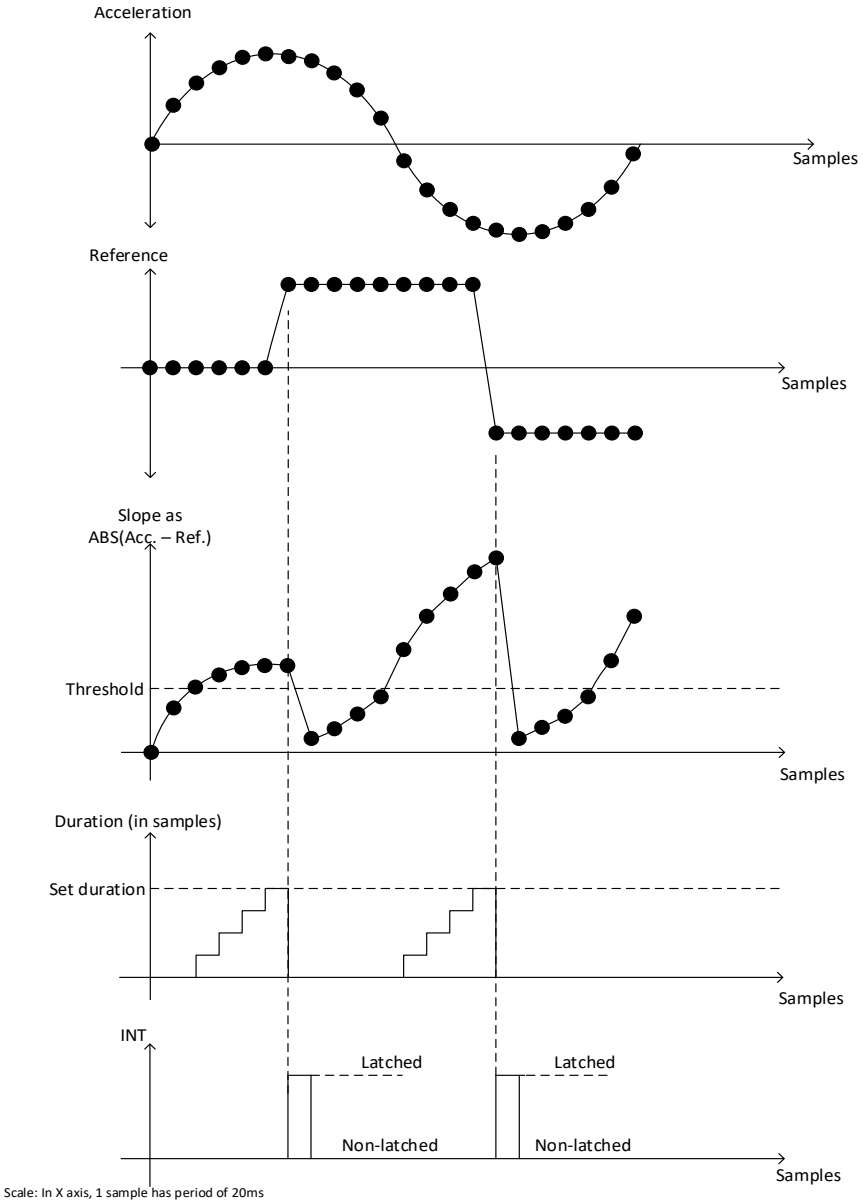
Any Motion / No motion detection

Any-motion detection:
The any-motion detection uses the slope between two acceleration signal samples to detect changes in motion. The feature is enabled by setting at least one of the following flags: FEATURES_IN.any_motion.settings_2.x_en, FEATURES_IN.any_motion.settings_2.y_en and FEATURES_IN.any_motion.settings_2.z_en, respectively for each axis.

It generates an interrupt when the absolute value of the slope exceeds the configurable FEATURES_IN.any_motion.settings_1.threshold for consecutive FEATURES_IN.any_motion.settings_2.duration samples.

The slope is computed between the current acceleration sample and the reference sample. The reference sample is updated when the any-motion interrupt is triggered. The interrupt is reset as soon as the slope is less than the FEATURES_IN.any_motion.setings_1.threshold.

Any-motion (slope) interrupt detection



Configuration settings:

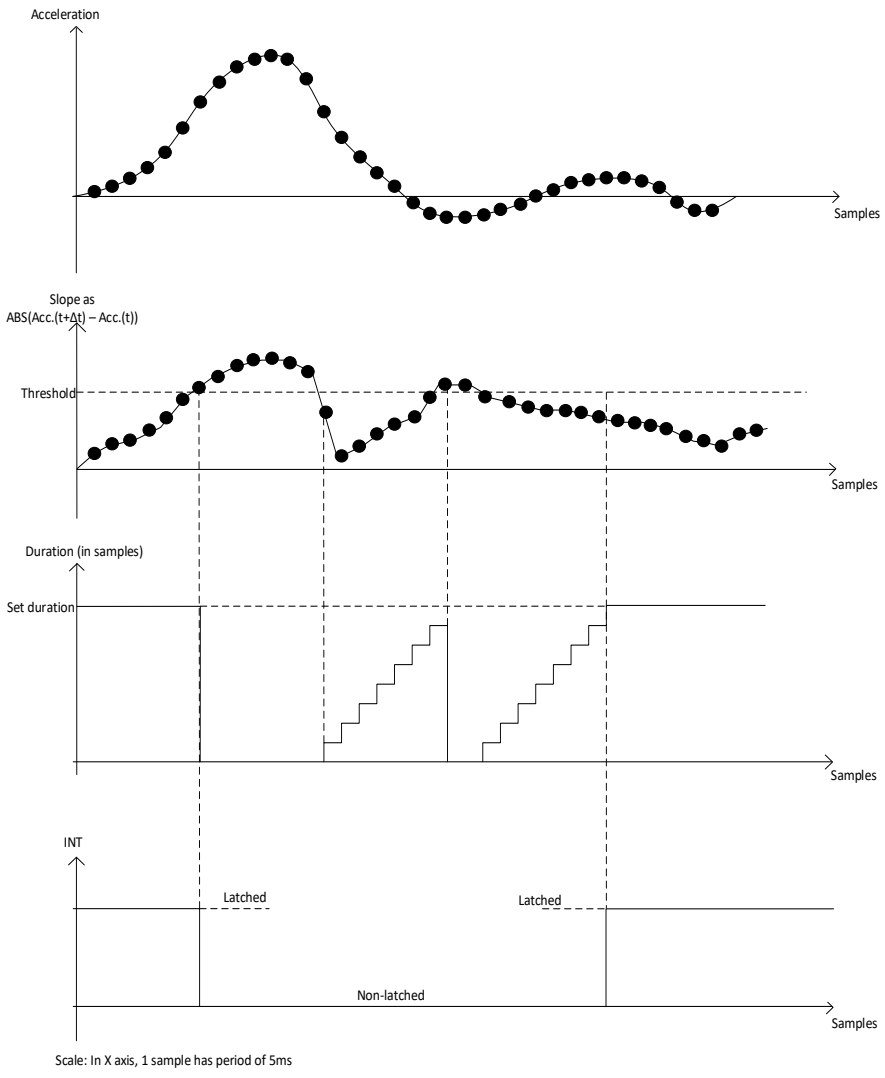
1. FEATURES_IN.any_motion.settings_1.threshold – the slope threshold.
2. FEATURES_IN.any_motion.settings_2.duration – the number of consecutive data points for which the threshold condition must be respected, for interrupt assertion.
3. FEATURES_IN.any_motion.settings_2.x_en – indicates if this feature is enabled for x axis
4. FEATURES_IN.any_motion.settings_2.y_en – indicates if this feature is enabled for y axis
5. FEATURES_IN.any_motion.settings_2.z_en –indicates if this feature is enabled for z axis

No Motion Detection:

The no-motion detection uses the slope between two consecutive acceleration signal samples to detect static state of the device. The interrupt is enabled by setting at least one of the following flags: FEATURES_IN.no_motion.settings_2.x_en, FEATURES_IN.no_motion.settings_2.y_en and FEATURES_IN.no_motion.settings_2.z_en, respectively for each axis.

No-motion interrupt is triggered when the slope on all selected axis remains smaller than a configurable duration configured by FEATURES_IN.no_motion.settings_1.threshold . The signals and timings relevant to the no-motion interrupt functionality are depicted in the figure below.

Signal timings No-motion interrupt



Register FEATURES_IN.no_motion.settings_2.duration defines the number of consecutive slope data points of the selected axis which must exceed the threshold for an interrupt to be asserted.

Configuration settings:

1. FEATURES_IN.no_motion.settings_1.threshold – the slope threshold.
2. FEATURES_IN.no_motion.settings_2.duration – the number of consecutive data points for which the threshold condition must be respected, for interrupt assertion.
3. FEATURES_IN.no_motion.settings_2.x_en – indicates if this feature is enabled for x axis
4. FEATURES_IN.no_motion.settings_2.y_en – indicates if this feature is enabled for y axis
5. FEATURES_IN.no_motion.settings_2.z_en –indicates if this feature is enabled for z axis

1.2. General Interrupt Pin configuration

Electrical Interrupt Pin Behavior

Both interrupt pins INT1 and INT2 can be configured to show the desired electrical behavior.
Interrupt pins can be enabled in INT1_IO_CTRL.output_en respectively INT2_IO_CTRL.output_en.
The characteristic of the output driver of the interrupt pins may be configured with bits INT1_IO_CTRL.od and INT2_IO_CTRL.od. By setting these bits to 0b1, the output driver shows open-drive characteristic, by setting the configuration bits to 0b0, the output driver shows push-pull characteristic.
The electrical behavior of the Interrupt pins, whenever an interrupt is triggered, can be configured as either “active-high” or “active-low” via INT1_IO_CTRL.lvl respectively INT2_IO_CTRL.lvl.

Both interrupt pins can be configured as input pins via INT1_IO_CTRL.input_en respectively INT2_IO_CTRL.input_en. This is necessary when FIFO tag feature is used (see Datasheet). If both input_en and output_en are enabled on the same pin, the input (e.g. marking FIFO) is driven by the interrupt output.
BMA supports edge and level triggered interrupt outputs, this can be configured through INT1_IO_CTRL.edge_ctrl respectively INT2_IO_CTRL.edge_ctrl.

BMA supports non-latched and latched interrupts modes for data-ready, FIFO full and FIFO watermark. The mode is selected by INT_LATCH.int_latch. The feature interrupts described in chapter 1.1 support only latched mode described below.
In latched mode an asserted interrupt status in INT_STATUS_0 or INT_STATUS_1 and the selected pin are cleared if the corresponding status register is read. If more than one interrupt pin is used in latched mode, all interrupts in INT_STATUS_0 should be mapped to one pin and all interrupts in INT_STATUS_1 should be mapped to the other pin. If just one interrupt pin is used all interrupts may be mapped to this pin. If the activation condition still holds when it is cleared, the interrupt status is asserted again when the interrupt condition holds again.
In the non-latched mode (only for data-ready, FIFO full and FIFO watermark) the interrupt status bit and the selected pin are reset as soon as the activation condition is not valid anymore.

Interrupt Pin Mapping

In order, for the Host to react to the features output, they can be mapped to the external pin INT1 or pin INT2, by setting the corresponding bits from the registers INT1_MAP, respectively INT2_MAP.
To disconnect the features outputs to the external pins, the same corresponding bits must be reset, from the registers, INT1_MAP, respectively INT2_MAP.

Once a feature triggered the output pin, the Host can read out the corresponding bit from the register, INT_STATUS_0 (Feature Interrupts) or INT_STATUS_1 (FIFO and data ready).

2. Register Description

2.1. General Remarks

Registers can be read and written in all power configurations with the exception of FEATURES_IN and FIFO_DATA which need PWR_CONF.adv_power_save set to 0b0.

2.2. Register Map

read/write	read only	write only	reserved

			ID:								
Register Address	**Register Name**	**Default Value**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0x7E	CMD	0x00	cmd							
0x7D	PWR_CTRL	0x00	reserved					acc_en	reserved	aux_en
0x7C	PWR_CONF	0x03	reserved						fifo_self_wakeup	adv_power_save
0x7B	-	-	reserved							
...	-	-	reserved							
0x74	-	-	reserved							
0x73	OFFSET_2	0x00	off_acc_z							
0x72	OFFSET_1	0x00	off_acc_y							
0x71	OFFSET_0	0x00	off_acc_x							
0x70	NV_CONF	0x00	reserved				acc_off_en	i2c_wdt_en	i2c_wdt_sel	spi_en
0x6F	-	-	reserved							
0x6E	-	-	reserved							
0x6D	ACC_SELF_TEST	0x00	reserved				acc_self_test_amp	acc_self_test_sign	reserved	acc_self_test_en
0x6C	-	-	reserved							
0x6B	IF_CONF	0x00	reserved			if_mode	reserved			spi3
0x6A	NVM_CONF	0x00	reserved						nvm_prog_en	reserved
0x69	-	-	reserved							
...	-	-	reserved							
0x60	-	-	reserved							
0x5F	INTERNAL_ERROR	0x00	reserved					int_err_2	int_err_1	reserved
0x5E	FEATURES_IN	0x00	features_in							
0x5D	-	-	reserved							
...	-	-	reserved							
0x5A	-	-	reserved							

0x59	INIT_CTRL	0x90	init_ctrl							
0x58	INT_MAP_DATA	0x00	reserved	int2_drdy	int2_fwm	int2_ffull	reserved	int1_drdy	int1_fwm	int1_ffull
0x57	INT2_MAP	0x00	error_int_out	no_motion_out	any_motion_out	double_tap_out	wrist_wake_wakeup_out	activity_type_out	step_counter_out	single_tap_out
0x56	INT1_MAP	0x00	error_int_out	no_motion_out	any_motion_out	double_tap_out	wrist_wake_wakeup_out	activity_type_out	step_counter_out	single_tap_out
0x55	INT_LATCH	0x00	reserved							int_latch
0x54	INT2_IO_CTRL	0x00	reserved			input_en	output_en	od	lvl	edge_ctrl
0x53	INT1_IO_CTRL	0x00	reserved			input_en	output_en	od	lvl	edge_ctrl
0x52	-	-	reserved							
...	-	-	reserved							
0x50	-	-	reserved							
0x4F	AUX_WR_DATA	0x02	write_data							
0x4E	AUX_WR_ADDR	0x4C	write_addr							
0x4D	AUX_RD_ADDR	0x42	read_addr							
0x4C	AUX_IF_CONF	0x83	aux_manual_en	reserved					aux_rd_burst	
0x4B	AUX_DEV_ID	0x20	i2c_device_addr							reserved
0x4A	-	-	reserved							
0x49	FIFO_CONFIG_1	0x10	reserved	fifo_acc_en	fifo_aux_en	fifo_header_en	fifo_tag_int1_en	fifo_tag_int2_en	reserved	
0x48	FIFO_CONFIG_0	0x02	reserved						fifo_time_en	fifo_stop_on_full
0x47	FIFO_WTM_1	0x02	reserved			fifo_water_mark_12_8				
0x46	FIFO_WTM_0	0x00	fifo_water_mark_7_0							
0x45	FIFO_DOWNS	0x80	acc_fifo_filt_data	acc_fifo_downs			reserved			
0x44	AUX_CONF	0x46	aux_offset				aux_odr			
0x43	-	-	reserved							
0x42	-	-	reserved							
0x41	ACC_RANGE	0x01	reserved						acc_range	
0x40	ACC_CONF	0xA8	acc_perf_mode	acc_bwp			acc_odr			

0x3F	-	-	reserved						
...	-	-	reserved						
0x2B	-	-	reserved						
0x2A	INTERNAL_STATUS	0x00	odr_high_error	odr_50hz_error	axes_remap_error	message			
0x29	-	-	reserved						
0x28	-	-	reserved						
0x27	ACTIVITY_TYPE	0x00	reserved					activity_type_out	
0x26	FIFO_DATA	0x00	fifo_data						
0x25	FIFO_LENGTH_1	0x00	reserved	fifo_byte_counter_13_8					
0x24	FIFO_LENGTH_0	0x00	fifo_byte_counter_7_0						
0x23	-	-	reserved						
0x22	TEMPERATURE	0x00	temperature						
0x21	STEP_COUNTER_3	0x00	step_counter_out_3						
0x20	STEP_COUNTER_2	0x00	step_counter_out_2						
0x1F	STEP_COUNTER_1	0x00	step_counter_out_1						
0x1E	STEP_COUNTER_0	0x00	step_counter_out_0						
0x1D	INT_STATUS_1	0x00	acc_drdy_int	reserved	aux_drdy_int	reserved		fwm_int	ffull_int
0x1C	INT_STATUS_0	0x00	error_int_out	no_motion_out	any_motion_out	double_tap_out	wrist_wake_wakeup_out	activity_type_out	step_counter_out	single_tap_out
0x1B	EVENT	0x01	reserved						por_detected
0x1A	SENSORTIME_2	0x00	sensor_time_23_16						
0x19	SENSORTIME_1	0x00	sensor_time_15_8						
0x18	SENSORTIME_0	0x00	sensor_time_7_0						
0x17	DATA_13	0x00	acc_z_11_4						
0x16	DATA_12	0x00	acc_z_3_0				reserved		
0x15	DATA_11	0x00	acc_y_11_4						
0x14	DATA_10	0x00	acc_y_3_0				reserved		
0x13	DATA_9	0x00	acc_x_11_4						

0x12	DATA_8	0x00	acc_x_3_0								reserved			
0x11	DATA_7	0x00	aux_r_11_4											
0x10	DATA_6	0x00	aux_r_3_0								reserved			
0x0F	DATA_5	0x00	aux_z_11_4											
0x0E	DATA_4	0x00	aux_z_3_0								reserved			
0x0D	DATA_3	0x00	aux_y_11_4											
0x0C	DATA_2	0x00	aux_y_3_0								reserved			
0x0B	DATA_1	0x00	aux_x_11_4											
0x0A	DATA_0	0x00	aux_x_3_0								reserved			
0x09	-	-	reserved											
...	-	-	reserved											
0x04	-	-	reserved											
0x03	STATUS	0x10	drdy_acc		reserved		drdy_aux		cmd_rdy		reserved		aux_man_op	reserved
0x02	ERR_REG	0x00	aux_err		fifo_err		reserved		error_code				cmd_err	fatal_err
0x01	-	-	reserved											
0x00	CHIP_ID	0x13	chip_id											

FEATURES_IN

Register Address	Register Name	Default Value	7	6	5	4	3	2	1	0
0x5E: 0x45	general_settings.axes_remapping[1]	0x00	reserved							map_z_axis_sign
0x5E: 0x44	general_settings.axes_remapping[0]	0x88	map_z_axis		map_y_axis_sign		map_y_axis		map_x_axis_sign	map_x_axis
0x5E: 0x43	general_settings.config_id[1]	0x00	identification							
0x5E: 0x42	general_settings.config_id[0]	0x00	identification							
0x5E: 0x41	wrist_ware_wakeup.settings[1]	0x00	reserved							
0x5E: 0x40	wrist_ware_wakeu	0x00	reserved							enable

	p.settings[0]							
0x5E: 0x3F	double_tap.settings[1]	0x00	reserved					
0x5E: 0x3E	double_tap.settings[0]	0x06	reserved			sensitivity		enable
0x5E: 0x3D	single_tap.settings[1]	0x00	reserved					
0x5E: 0x3C	single_tap.settings[0]	0x06	reserved			sensitivity		enable
0x5E: 0x3B	step_counter.settings_26[1]	0x00	reserved	en_activity	en_counter	en_detector	reset_counter	watermark_level
0x5E: 0x3A	step_counter.settings_26[0]	0x00	watermark_level					
0x5E: 0x39	step_counter.settings_25[1]	0x00	param_25					
0x5E: 0x38	step_counter.settings_25[0]	0x0E	param_25					
0x5E: 0x37	step_counter.settings_24[1]	0x00	param_24					
0x5E: 0x36	step_counter.settings_24[0]	0x01	param_24					
0x5E: 0x35	step_counter.settings_23[1]	0x00	param_23					
0x5E: 0x34	step_counter.settings_23[0]	0x03	param_23					
0x5E: 0x33	step_counter.settings_22[1]	0x00	param_22					

0x5E: 0x32	step_counter.settings_22[0]	0x01	param_22
0x5E: 0x31	step_counter.settings_21[1]	0x01	param_21
0x5E: 0x30	step_counter.settings_21[0]	0x00	param_21
0x5E: 0x2F	step_counter.settings_20[1]	0x3C	param_20
0x5E: 0x2E	step_counter.settings_20[0]	0xF0	param_20
0x5E: 0x2D	step_counter.settings_19[1]	0x00	param_19
0x5E: 0x2C	step_counter.settings_19[0]	0x0C	param_19
0x5E: 0x2B	step_counter.settings_18[1]	0x00	param_18
0x5E: 0x2A	step_counter.settings_18[0]	0x01	param_18
0x5E: 0x29	step_counter.settings_17[1]	0x00	param_17
0x5E: 0x28	step_counter.settings_17[0]	0xA0	param_17
0x5E: 0x27	step_counter.settings_16[1]	0x00	param_16
0x5E: 0x26	step_counter.setti	0x96	param_16

	ngs_16[0]		
0x5E: 0x25	step_counter.settings_15[1]	0x00	param_15
0x5E: 0x24	step_counter.settings_15[0]	0x19	param_15
0x5E: 0x23	step_counter.settings_14[1]	0x00	param_14
0x5E: 0x22	step_counter.settings_14[0]	0x27	param_14
0x5E: 0x21	step_counter.settings_13[1]	0x00	param_13
0x5E: 0x20	step_counter.settings_13[0]	0x01	param_13
0x5E: 0x1F	step_counter.settings_12[1]	0x46	param_12
0x5E: 0x1E	step_counter.settings_12[0]	0x0C	param_12
0x5E: 0x1D	step_counter.settings_11[1]	0xE6	param_11
0x5E: 0x1C	step_counter.settings_11[0]	0xEC	param_11
0x5E: 0x1B	step_counter.settings_10[1]	0x04	param_10
0x5E: 0x1A	step_counter.settings_10[0]	0xC3	param_10

0x5E: 0x19	step_counter.settings_9[1]	0x09	param_9
0x5E: 0x18	step_counter.settings_9[0]	0x85	param_9
0x5E: 0x17	step_counter.settings_8[1]	0x04	param_8
0x5E: 0x16	step_counter.settings_8[0]	0xC3	param_8
0x5E: 0x15	step_counter.settings_7[1]	0x6C	param_7
0x5E: 0x14	step_counter.settings_7[0]	0xCD	param_7
0x5E: 0x13	step_counter.settings_6[1]	0x7B	param_6
0x5E: 0x12	step_counter.settings_6[0]	0x3F	param_6
0x5E: 0x11	step_counter.settings_5[1]	0x00	param_5
0x5E: 0x10	step_counter.settings_5[0]	0x04	param_5
0x5E: 0x0F	step_counter.settings_4[1]	0x7A	param_4
0x5E: 0x0E	step_counter.settings_4[0]	0xDB	param_4
0x5E: 0x0D	step_counter.settings_3[1]	0x01	param_3
0x5E: 0x0C	step_counter.settings_3[0]	0x3B	param_3
0x5E: 0x0B	step_counter.settings_2[1]	0x7B	param_2
0x5E: 0x0A	step_counter.settings_2[0]	0xD4	param_2

0x5E: 0x09	step_counter.settings_1[1]	0x01	param_1			
0x5E: 0x08	step_counter.settings_1[0]	0x2D	param_1			
0x5E: 0x07	no_motion.settings_2[1]	0x00	z_en	y_en	x_en	duration
0x5E: 0x06	no_motion.settings_2[0]	0x05	duration			
0x5E: 0x05	no_motion.settings_1[1]	0x00	reserved			threshold
0x5E: 0x04	no_motion.settings_1[0]	0xAA	threshold			
0x5E: 0x03	any_motion.settings_2[1]	0x00	z_en	y_en	x_en	duration
0x5E: 0x02	any_motion.settings_2[0]	0x05	duration			
0x5E: 0x01	any_motion.settings_1[1]	0x00	reserved			threshold
0x5E: 0x00	any_motion.settings_1[0]	0xAA	threshold			

Register (0x00) CHIP_ID

DESCRIPTION: Chip identification code
RESET: see datasheet of your product
DEFINITION (Go to register map):

Name	**Register (0x00) CHIP_ID**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value				
Content	chip_id			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value				
Content	chip_id			

chip_id: Chip identification code your product (please see data sheet)

Register (0x02) ERR_REG

DESCRIPTION: Reports sensor error conditions
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x02) ERR_REG**			
Bit	7	6	5	4
Read/Write	R	R	n/a	R
Reset Value	0	0	0	0
Content	aux_err	fifo_err	reserved	error_code
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	error_code		cmd_err	fatal_err

fatal_err: Fatal Error, chip is not in operational state (Boot-, power-system). This flag will be reset only by power-on-reset or softreset.
cmd_err: Command execution failed.
error_code: Error codes for persistent errors

error_code		
0x00	no_error	no error is reported
0x01	acc_err	error in Register ACC_CONF

fifo_err: Error in FIFO detected: Input data was discarded in stream mode. This flag will be reset when read.
aux_err: Error in I2C-Master detected. This flag will be reset when read.

Register (0x03) STATUS

DESCRIPTION: Sensor status flags
RESET: 0x10
DEFINITION (Go to register map):

Name	**Register (0x03) STATUS**			
Bit	7	6	5	4
Read/Write	R	n/a	R	R
Reset Value	0	0	0	1
Content	drdy_acc	reserved	drdy_aux	cmd_rdy
Bit	3	2	1	0
Read/Write	n/a	R	n/a	n/a
Reset Value	0	0	0	0
Content	reserved	aux_man_op	reserved	

aux_man_op: ‘1’(‘0’) indicate a (no) manual auxiliary interface operation is ongoing.
cmd_rdy: CMD decoder status. `0´ -> Command in progress `1´ -> Command decoder is ready to accept a new command
drdy_aux: Data ready for auxiliary sensor. It gets reset when one auxiliary DATA register is read out
drdy_acc: Data ready for accelerometer. It gets reset when one accelerometer DATA register is read out

Register (0x0A) DATA_0

DESCRIPTION: AUX_X(LSB)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x0A) DATA_0**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	aux_x_3_0			
Bit	3	2	1	0
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			

Register (0x0B) DATA_1

DESCRIPTION: AUX_X(MSB)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x0B) DATA_1**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	aux_x_11_4			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	aux_x_11_4			

Register (0x0C) DATA_2

DESCRIPTION: AUX_Y(LSB)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x0C) DATA_2**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	aux_y_3_0			
Bit	3	2	1	0
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			

Register (0x0D) DATA_3

DESCRIPTION: AUX_Y(MSB)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x0D) DATA_3**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	aux_y_11_4			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	aux_y_11_4			

Register (0x0E) DATA_4

DESCRIPTION: AUX_Z(LSB)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x0E) DATA_4**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	aux_z_3_0			
Bit	3	2	1	0
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			

Register (0x0F) DATA_5

DESCRIPTION: AUX_Z(MSB)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x0F) DATA_5**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	aux_z_11_4			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	aux_z_11_4			

Register (0x10) DATA_6

DESCRIPTION: AUX_R(LSB)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x10) DATA_6**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	aux_r_3_0			
Bit	3	2	1	0
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			

Register (0x11) DATA_7

DESCRIPTION: AUX_R(MSB)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x11) DATA_7**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	aux_r_11_4			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	aux_r_11_4			

Register (0x12) DATA_8

DESCRIPTION: ACC_X(LSB)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x12) DATA_8**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	acc_x_3_0			
Bit	3	2	1	0
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			

Register (0x13) DATA_9

DESCRIPTION: ACC_X(MSB)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x13) DATA_9**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	acc_x_11_4			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	acc_x_11_4			

Register (0x14) DATA_10

DESCRIPTION: ACC_Y(LSB)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x14) DATA_10**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	acc_y_3_0			
Bit	3	2	1	0
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			

Register (0x15) DATA_11

DESCRIPTION: ACC_Y(MSB)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x15) DATA_11**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	acc_y_11_4			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	acc_y_11_4			

Register (0x16) DATA_12

DESCRIPTION: ACC_Z(LSB)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x16) DATA_12**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	acc_z_3_0			
Bit	3	2	1	0
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			

Register (0x17) DATA_13

DESCRIPTION: ACC_Z(MSB)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x17) DATA_13**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	acc_z_11_4			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	acc_z_11_4			

Register (0x18) SENSORTIME_0

DESCRIPTION: Sensor time <7:0>
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x18) SENSORTIME_0**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	sensor_time_7_0			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	sensor_time_7_0			

sensor_time_7_0: Sensor time <7:0> in units of 39.0625 us.

Register (0x19) SENSORTIME_1

DESCRIPTION: Sensor time <15:8>
RESET: 0x00
DEFINITION (Go to register map):

Name	Register (0x19) SENSORTIME_1			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	sensor_time_15_8			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	sensor_time_15_8			

sensor_time_15_8: Sensor time <15:8> in units of 10 ms.

Register (0x1A) SENSORTIME_2

DESCRIPTION: Sensor time <23:16>
RESET: 0x00
DEFINITION (Go to register map):

Name	Register (0x1A) SENSORTIME_2			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	sensor_time_23_16			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	sensor_time_23_16			

sensor_time_23_16: Sensor time <23:16> in units of 2.56 s.

Register (0x1B) EVENT

DESCRIPTION: Sensor status flags
RESET: 0x01
DEFINITION (Go to register map):

Name	**Register (0x1B) EVENT**			
Bit	7	6	5	4
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			
Bit	3	2	1	0
Read/Write	n/a	n/a	n/a	R
Reset Value	0	0	0	1
Content	reserved			por_detected

por_detected: '1' after device power up or softreset. Clear-on-read

Register (0x1C) INT_STATUS_0

DESCRIPTION: Interrupt/Feature Status. Will be cleared on read.
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x1C) INT_STATUS_0**			
Bit	7	6	5	4
Read/Write	R	R	R	n/a
Reset Value	0	0	0	0
Content	error_int_out	no_motion_out	any_motion_out	double_tap_out
Bit	3	2	1	0
Read/Write	R	R	R	n/a
Reset Value	0	0	0	0
Content	wrist_wear_wakeup_out	activity_type_out	step_counter_out	single_tap_out

single_tap_out: Single tap output
step_counter_out: Step-counter watermark or Step-detector output.
activity_type_out: Step counter activity output (Running, Walking, Still)
wrist_wear_wakeup_out: wrist wear wakeup output
double_tap_out: Double tap output
any_motion_out: Any-motion detection output
no_motion_out: No-motion detection output
error_int_out: Error interrupt output

Register (0x1D) INT_STATUS_1

DESCRIPTION: Interrupt Status. Will be cleared on read.
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x1D) INT_STATUS_1**			
Bit	7	6	5	4
Read/Write	R	n/a	R	n/a
Reset Value	0	0	0	0
Content	acc_drdy_int	reserved	aux_drdy_int	reserved
Bit	3	2	1	0
Read/Write	n/a	n/a	R	R
Reset Value	0	0	0	0
Content	reserved		fwm_int	ffull_int

ffull_int: FIFO Full Interrupt
fwm_int: FIFO Watermark Interrupt
aux_drdy_int: Auxiliary sensor data ready interrupt
acc_drdy_int: Accelerometer data ready interrupt

Register (0x1E) STEP_COUNTER_0

DESCRIPTION: Step counting value byte-0
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x1E) STEP_COUNTER_0**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	step_counter_out_0			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	step_counter_out_0			

step_counter_out_0: Step counting value byte-0 (least significant byte), unit is step.

Register (0x1F) STEP_COUNTER_1

DESCRIPTION: Step counting value byte-1
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x1F) STEP_COUNTER_1**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	step_counter_out_1			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	step_counter_out_1			

step_counter_out_1: Step counting value byte-1

Register (0x20) STEP_COUNTER_2

DESCRIPTION: Step counting value byte-2
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x20) STEP_COUNTER_2**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	step_counter_out_2			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	step_counter_out_2			

step_counter_out_2: Step counting value byte-2

Register (0x21) STEP_COUNTER_3

DESCRIPTION: Step counting value byte-3
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x21) STEP_COUNTER_3**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	step_counter_out_3			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	step_counter_out_3			

step_counter_out_3: Step counting value byte-3 (most significant byte)

Register (0x22) TEMPERATURE

DESCRIPTION: Contains the temperature value of the sensor
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x22) TEMPERATURE**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	temperature			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	temperature			

temperature: Temperature value in two's complement representation in units of 1 Kelvin: 0x00 corresponds to 23 degree Celsius.

Register (0x24) FIFO_LENGTH_0

DESCRIPTION: FIFO byte count register (LSB)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x24) FIFO_LENGTH_0**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	fifo_byte_counter_7_0			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	fifo_byte_counter_7_0			

fifo_byte_counter_7_0: Current fill level of FIFO buffer(unit: byte).

Register (0x25) FIFO_LENGTH_1

DESCRIPTION: FIFO byte count register (MSB)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x25) FIFO_LENGTH_1**			
Bit	7	6	5	4
Read/Write	n/a	n/a	R	R
Reset Value	0	0	0	0
Content	reserved		fifo_byte_counter_13_8	
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	fifo_byte_counter_13_8			

fifo_byte_counter_13_8: FIFO byte counter bits 13..8.

Register (0x26) FIFO_DATA

DESCRIPTION: FIFO data output register
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x26) FIFO_DATA**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	fifo_data			
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	fifo_data			

fifo_data: FIFO read data, for burst read.

Register (0x27) ACTIVITY_TYPE

DESCRIPTION: Step counter activity output(Running, Walking, Still)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x27) ACTIVITY_TYPE**			
Bit	7	6	5	4
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			
Bit	3	2	1	0
Read/Write	n/a	n/a	R	R
Reset Value	0	0	0	0
Content	reserved		activity_type_out	

activity_type_out: Step counter activity output (Running, Walking, Still)

activity_type_out		
0x00	still	user not moving
0x01	walking	user walking
0x02	running	user running
0x03	unknown	unknown state

Register (0x2A) INTERNAL_STATUS

DESCRIPTION: Error bits and message indicating internal status
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x2A) INTERNAL_STATUS**			
Bit	7	6	5	4
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	odr_high_error	odr_50hz_error	axes_remap_error	message
Bit	3	2	1	0
Read/Write	R	R	R	R
Reset Value	0	0	0	0
Content	message			

message: Internal Status Message

message		
0x00	not_init	ASIC is not initialized
0x01	init_ok	ASIC initialized
0x02	init_err	Initialization error
0x03	drv_err	Invalid driver
0x04	sns_stop	Sensor stopped

axes_remap_error: Axes remapped wrongly because a source axis is not assigned to more than one target axis.

odr_50hz_error: The minimum bandwidth conditions are not respected for the features which require 50 Hz data.

odr_high_error: The minimum bandwidth conditions are not respected for the single/double tap Detection.

Register (0x40) ACC_CONF

DESCRIPTION: Sets the output data rate, the bandwidth, and the performance mode of the acceleration sensor
RESET: 0xA8
DEFINITION (Go to register map):

Name	**Register (0x40) ACC_CONF**			
Bit	7	6	5	4
Read/Write	RW	RW	RW	RW
Reset Value	1	0	1	0
Content	acc_perf_mode	acc_bwp		
Bit	3	2	1	0
Read/Write	RW	RW	RW	RW
Reset Value	1	0	0	0
Content	acc_odr			

acc_odr: ODR in Hz. The output data rate is independent of the power mode setting for the sensor, but not all settings are supported in all power modes.

acc_odr		
0x00	reserved	Reserved
0x01	odr_0p78	25/32
0x02	odr_1p5	25/16
0x03	odr_3p1	25/8
0x04	odr_6p25	25/4
0x05	odr_12p5	25/2
0x06	odr_25	25
0x07	odr_50	50
0x08	odr_100	100
0x09	odr_200	200
0x0a	odr_400	400
0x0b	odr_800	800
0x0c	odr_1k6	1600
0x0d	odr_3k2	Reserved
0x0e	odr_6k4	Reserved
0x0f	odr_12k8	Reserved

acc_bwp: Bandwidth parameter, determines filter configuration (acc_perf_mode=1) and averaging for undersampling mode (acc_perf_mode=0)

acc_bwp		
0x00	osr4_avg1	acc_perf_mode = 1 -> OSR4 mode; acc_perf_mode = 0 -> no averaging
0x01	osr2_avg2	acc_perf_mode = 1 -> OSR2 mode; acc_perf_mode = 0 -> average 2 samples
0x02	norm_avg4	acc_perf_mode = 1 -> normal mode; acc_perf_mode = 0 -> average 4 samples
0x03	cic_avg8	acc_perf_mode = 1 -> Reserved; acc_perf_mode = 0 -> average 8 samples

0x04	res_avg16	acc_perf_mode = 1 -> Reserved; acc_perf_mode = 0 -> average 16 samples
0x05	res_avg32	acc_perf_mode = 1 -> Reserved; acc_perf_mode = 0 -> average 32 samples
0x06	res_avg64	acc_perf_mode = 1 -> Reserved; acc_perf_mode = 0 -> average 64 samples
0x07	res_avg128	acc_perf_mode = 1 -> Reserved; acc_perf_mode = 0 -> average 128 samples

acc_perf_mode: Select accelerometer filter performance mode:

acc_perf_mode		
0x00	cic_avg	averaging mode.
0x01	cont	continuous filter function.

Register (0x41) ACC_RANGE

DESCRIPTION: Selection of the Accelerometer g-range
RESET: 0x01
DEFINITION (Go to register map):

Name	**Register (0x41) ACC_RANGE**			
Bit	7	6	5	4
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			
Bit	3	2	1	0
Read/Write	n/a	n/a	RW	RW
Reset Value	0	0	0	1
Content	reserved		acc_range	

acc_range: Accelerometer g-range

acc_range		
0x00	range_2g	+/-2g
0x01	range_4g	+/-4g
0x02	range_8g	+/-8g
0x03	range_16g	+/-16g

Register (0x44) AUX_CONF

DESCRIPTION: Sets the output data rate of the Auxiliary interface
RESET: 0x46
DEFINITION (Go to register map):

Name	**Register (0x44) AUX_CONF**			
Bit	7	6	5	4
Read/Write	RW	RW	RW	RW
Reset Value	0	1	0	0
Content	aux_offset			
Bit	3	2	1	0
Read/Write	RW	RW	RW	RW
Reset Value	0	1	1	0
Content	aux_odr			

aux_odr: Select the poll rate for the sensor attached to the Auxiliary interface.

aux_odr		
0x00	reserved	Reserved
0x01	odr_0p78	25/32
0x02	odr_1p5	25/16
0x03	odr_3p1	25/8
0x04	odr_6p25	25/4
0x05	odr_12p5	25/2
0x06	odr_25	25
0x07	odr_50	50
0x08	odr_100	100
0x09	odr_200	200
0x0a	odr_400	400
0x0b	odr_800	800
0x0c	odr_1k6	Reserved
0x0d	odr_3k2	Reserved
0x0e	odr_6k4	Reserved
0x0f	odr_12k8	Reserved

aux_offset: trigger-readout offset in units of 2.5 ms. If set to zero, the offset is maximum, i.e. after readout a trigger is issued immediately.

Register (0x45) FIFO_DOWNS

DESCRIPTION: Configure Accelerometer downsampling rates for FIFO
RESET: 0x80
DEFINITION (Go to register map):

Name	**Register (0x45) FIFO_DOWNS**			
Bit	7	6	5	4
Read/Write	RW	RW	RW	RW
Reset Value	1	0	0	0
Content	acc_fifo_filt_data	acc_fifo_downs		
Bit	3	2	1	0
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			

acc_fifo_downs: Downsampling for accelerometer data (by 2**acc_fifo_downs)

acc_fifo_filt_data: selects filtered or unfiltered Accelerometer data for FIFO

acc_fifo_filt_data		
0x00	unfiltered	Unfiltered data
0x01	filtered	Filtered data

Register (0x46) FIFO_WTM_0

DESCRIPTION: FIFO Watermark level LSB, unit is byte.
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x46) FIFO_WTM_0**			
Bit	7	6	5	4
Read/Write	RW	RW	RW	RW
Reset Value	0	0	0	0
Content	fifo_water_mark_7_0			
Bit	3	2	1	0
Read/Write	RW	RW	RW	RW
Reset Value	0	0	0	0
Content	fifo_water_mark_7_0			

Register (0x47) FIFO_WTM_1

DESCRIPTION: FIFO Watermark level MSB
RESET: 0x02
DEFINITION (Go to register map):

Name	**Register (0x47) FIFO_WTM_1**			
Bit	7	6	5	4
Read/Write	n/a	n/a	n/a	RW
Reset Value	0	0	0	0
Content	reserved			fifo_water_mark_12_8
Bit	3	2	1	0
Read/Write	RW	RW	RW	RW
Reset Value	0	0	1	0
Content	fifo_water_mark_12_8			

Register (0x48) FIFO_CONFIG_0

DESCRIPTION: FIFO frame content configuration
RESET: 0x02
DEFINITION (Go to register map):

Name	**Register (0x48) FIFO_CONFIG_0**			
Bit	7	6	5	4
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			
Bit	3	2	1	0
Read/Write	n/a	n/a	RW	RW
Reset Value	0	0	1	0
Content	reserved		fifo_time_en	fifo_stop_on_full

fifo_stop_on_full: Stop writing samples into FIFO when FIFO is full.

fifo_stop_on_full		
0x00	disable	do not stop writing to FIFO when full
0x01	enable	Stop writing into FIFO when full.

fifo_time_en: Return sensortime frame after the last valid data frame.

fifo_time_en		
0x00	disable	do not return sensortime frame
0x01	enable	return sensortime frame

Register (0x49) FIFO_CONFIG_1

DESCRIPTION: FIFO frame content configuration
RESET: 0x10
DEFINITION (Go to register map):

Name	**Register (0x49) FIFO_CONFIG_1**			
Bit	7	6	5	4
Read/Write	n/a	RW	RW	RW
Reset Value	0	0	0	1
Content	reserved	fifo_acc_en	fifo_aux_en	fifo_header_en
Bit	3	2	1	0
Read/Write	RW	RW	n/a	n/a
Reset Value	0	0	0	0
Content	fifo_tag_int1_en	fifo_tag_int2_en	reserved	

fifo_tag_int2_en: FIFO interrupt 2 tag enable

fifo_tag_int2_en		
0x00	disable	disable tag
0x01	enable	enable tag

fifo_tag_int1_en: FIFO interrupt 1 tag enable

fifo_tag_int1_en		
0x00	disable	disable tag
0x01	enable	enable tag

fifo_header_en: FIFO frame header enable

fifo_header_en		
0x00	disable	no header is stored (output data rate of all enabled sensors need to be identical)
0x01	enable	header is stored

fifo_aux_en: Store Auxiliary data in FIFO (all 3 axes)

fifo_aux_en		
0x00	disable	no Auxiliary data is stored
0x01	enable	Auxiliary data is stored

fifo_acc_en: Store Accelerometer data in FIFO (all 3 axes)

fifo_acc_en		
0x00	disable	no Accelerometer data is stored
0x01	enable	Accelerometer data is stored

Register (0x4B) AUX_DEV_ID

DESCRIPTION: Auxiliary interface slave device address
RESET: 0x20
DEFINITION (Go to register map):

Name	**Register (0x4B) AUX_DEV_ID**			
Bit	7	6	5	4
Read/Write	RW	RW	RW	RW
Reset Value	0	0	1	0
Content	i2c_device_addr			
Bit	3	2	1	0
Read/Write	RW	RW	RW	n/a
Reset Value	0	0	0	0
Content	i2c_device_addr			reserved

i2c_device_addr: I2C device address of Auxiliary slave

Register (0x4C) AUX_IF_CONF

DESCRIPTION: Auxiliary interface configuration
RESET: 0x83
DEFINITION (Go to register map):

Name	**Register (0x4C) AUX_IF_CONF**			
Bit	7	6	5	4
Read/Write	RW	n/a	n/a	n/a
Reset Value	1	0	0	0
Content	aux_manual_en	reserved		
Bit	3	2	1	0
Read/Write	n/a	n/a	RW	RW
Reset Value	0	0	1	1
Content	reserved		aux_rd_burst	

aux_rd_burst: Burst data length (1,2,6,8 byte)

aux_rd_burst		
0x00	BL1	Burst length 1
0x01	BL2	Burst length 2
0x02	BL6	Burst length 6
0x03	BL8	Burst length 8

aux_manual_en: Enable auxiliary interface manual mode.

aux_manual_en		
0x00	disable	Data mode
0x01	enable	Setup mode

Register (0x4D) AUX_RD_ADDR

DESCRIPTION: Auxiliary interface read register address
RESET: 0x42
DEFINITION (Go to register map):

Name	**Register (0x4D) AUX_RD_ADDR**			
Bit	7	6	5	4
Read/Write	RW	RW	RW	RW
Reset Value	0	1	0	0
Content	read_addr			
Bit	3	2	1	0
Read/Write	RW	RW	RW	RW
Reset Value	0	0	1	0
Content	read_addr			

read_addr: Address to read

Register (0x4E) AUX_WR_ADDR

DESCRIPTION: Auxiliary interface write register address
RESET: 0x4C
DEFINITION (Go to register map):

Name	**Register (0x4E) AUX_WR_ADDR**			
Bit	7	6	5	4
Read/Write	RW	RW	RW	RW
Reset Value	0	1	0	0
Content	write_addr			
Bit	3	2	1	0
Read/Write	RW	RW	RW	RW
Reset Value	1	1	0	0
Content	write_addr			

write_addr: Address to write

Register (0x4F) AUX_WR_DATA

DESCRIPTION: Auxiliary interface write data
RESET: 0x02
DEFINITION (Go to register map):

Name	**Register (0x4F) AUX_WR_DATA**			
Bit	7	6	5	4
Read/Write	RW	RW	RW	RW
Reset Value	0	0	0	0
Content	write_data			
Bit	3	2	1	0
Read/Write	RW	RW	RW	RW
Reset Value	0	0	1	0
Content	write_data			

write_data: Data to write

Register (0x53) INT1_IO_CTRL

DESCRIPTION: Configure the electrical behavior of the interrupt pins
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x53) INT1_IO_CTRL**			
Bit	7	6	5	4
Read/Write	n/a	n/a	n/a	RW
Reset Value	0	0	0	0
Content	reserved			input_en
Bit	3	2	1	0
Read/Write	RW	RW	RW	RW
Reset Value	0	0	0	0
Content	output_en	od	lvl	edge_ctrl

edge_ctrl: Configure trigger condition of INT1 pin (input)

edge_ctrl		
0x00	level_tr	Level
0x01	edge_tr	Edge

lvl: Configure output level of INT1 pin

lvl		
0x00	active_low	active low
0x01	active_high	active high

od: Configure output behavior of INT1 pin to open drain.

od		
0x00	push_pull	push-pull
0x01	open_drain	open drain

output_en: Output enable for INT1 pin

output_en		
0x00	off	Output disabled
0x01	on	Output enabled

input_en: Input enable for INT1 pin

input_en		
0x00	off	Input disabled
0x01	on	Input enabled

Register (0x54) INT2_IO_CTRL

DESCRIPTION: Configure the electrical behavior of the interrupt pins
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x54) INT2_IO_CTRL**			
Bit	7	6	5	4
Read/Write	n/a	n/a	n/a	RW
Reset Value	0	0	0	0
Content	reserved			input_en
Bit	3	2	1	0
Read/Write	RW	RW	RW	RW
Reset Value	0	0	0	0
Content	output_en	od	lvl	edge_ctrl

edge_ctrl: Configure trigger condition of INT2 pin (input)

edge_ctrl		
0x00	level_tr	Level
0x01	edge_tr	Edge

lvl: Configure output level of INT2 pin

lvl		
0x00	active_low	active low
0x01	active_high	active high

od: Configure output behavior of INT2 pin to open drain.

od		
0x00	push_pull	push-pull
0x01	open_drain	open drain

output_en: Output enable for INT2 pin

output_en		
0x00	off	Output disabled
0x01	on	Output enabled

input_en: Input enable for INT2 pin

input_en		
0x00	off	Input disabled
0x01	on	Input enabled

Register (0x55) INT_LATCH

DESCRIPTION: Configure interrupt latch modes
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x55) INT_LATCH**			
Bit	7	6	5	4
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			
Bit	3	2	1	0
Read/Write	n/a	n/a	n/a	RW
Reset Value	0	0	0	0
Content	reserved			int_latch

int_latch: Latched/non-latched interrupt modes

int_latch		
0x00	none	non latched
0x01	permanent	latched

Register (0x56) INT1_MAP

DESCRIPTION: Map interrupt/feature sources to INT1
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x56) INT1_MAP**			
Bit	7	6	5	4
Read/Write	RW	RW	RW	n/a
Reset Value	0	0	0	0
Content	error_int_out	no_motion_out	any_motion_out	double_tap_out
Bit	3	2	1	0
Read/Write	RW	RW	RW	n/a
Reset Value	0	0	0	0
Content	wrist_ware_wakeup_out	activity_type_out	step_counter_out	single_tap_out

single_tap_out: Single tap output
step_counter_out: Step-counter watermark or Step-detector output.
activity_type_out: Step counter activity output (Running, Walking, Still)
wrist_ware_wakeup_out: Wrist ware wakeup output
double_tap_out: Double tap output
any_motion_out: Any-motion detection output
no_motion_out: No-motion detection output
error_int_out: Error interrupt output

Register (0x57) INT2_MAP

DESCRIPTION: Map interrupt/ieature sources to INT2
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x57) INT2_MAP**			
Bit	7	6	5	4
Read/Write	RW	RW	RW	n/a
Reset Value	0	0	0	0
Content	error_int_out	no_motion_out	any_motion_out	double_tap_out
Bit	3	2	1	0
Read/Write	RW	RW	RW	n/a
Reset Value	0	0	0	0
Content	wrist_ware_wakeup_out	activity_type_out	step_counter_out	single_tap_out

single_tap_out: Single tap output
step_counter_out: Step-counter watermark or Step-detector output.
activity_type_out: Step counter activity output (Running, Walking, Still)
wrist_ware_wakeup_out: Wrist ware wakeup output
double_tap_out: Double tap output
any_motion_out: Any-motion detection output

no_motion_out: No-motion detection output
error_int_out: Error interrupt output

Register (0x58) INT_MAP_DATA

DESCRIPTION: Map interrupt sources to hardware interrupts
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x58) INT_MAP_DATA**			
Bit	7	6	5	4
Read/Write	n/a	RW	RW	RW
Reset Value	0	0	0	0
Content	reserved	int2_drdy	int2_fwm	int2_ffull
Bit	3	2	1	0
Read/Write	n/a	RW	RW	RW
Reset Value	0	0	0	0
Content	reserved	int1_drdy	int1_fwm	int1_ffull

int1_ffull: FIFO Full interrupt mapped to INT1
int1_fwm: FIFO Watermark interrupt mapped to INT1
int1_drdy: Data Ready interrupt mapped to INT1
int2_ffull: FIFO Full interrupt mapped to INT2
int2_fwm: FIFO Watermark interrupt mapped to INT2
int2_drdy: Data Ready interrupt mapped to INT2

Register (0x59) INIT_CTRL

DESCRIPTION: Start initialization
RESET: 0x90
DEFINITION (Go to register map):

Name	**Register (0x59) INIT_CTRL**			
Bit	7	6	5	4
Read/Write	RW	RW	RW	RW
Reset Value	1	0	0	1
Content	init_ctrl			
Bit	3	2	1	0
Read/Write	RW	RW	RW	RW
Reset Value	0	0	0	0
Content	init_ctrl			

init_ctrl: commands to start initialization

init_ctrl		
0x00	Load configuration file	Enable the mode for accept configuration file
0x01	Start initialization	Enable sensor features after loading configuration file

Note: the commands should not been used more than once after POR or softreset, and the process of start initialization described in chapter 1 should be strictly followed.

Register (0x5E) FEATURES_IN

DESCRIPTION: Feature configuration read/write port
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x5E) FEATURES_IN**			
Bit	7	6	5	4
Read/Write	RW	RW	RW	RW
Reset Value	0	0	0	0
Content	features_in			
Bit	3	2	1	0
Read/Write	RW	RW	RW	RW
Reset Value	0	0	0	0
Content	features_in			

features_in: Feature configuration read/write data. Note that the parameters described here is in 16bit, while the address is in 8bit.

Address	**Bit**	**Name**	**Description**	**Reset**	**Access**
any_motion					
0x5E: 0x00		settings_1	Any-motion detection general configuration flags - part 1	0x00AA	
	10...0	threshold	Slope threshold value for Any-motion detection in 5.11g format. Range is 0 to 1g. Default value is 0xAA = 83mg = 0xAA * (1 / 2^11)mg.	0xAA	RW
0x5E: 0x02		settings_2	Any-motion detection general configuration flags - part 2	0x0005	
	12...0	duration	Defines the number of consecutive data points for which the threshold condition must be respected, for interrupt assertion. It is expressed in in 50 Hz samples (20 ms). Range is 0 to 163sec. Default value is 5=100ms.	0x5	RW
	13	x_en	Enables the feature on a per-axis basis	0x0	RW
	14	y_en	Enables the feature on a per-axis basis	0x0	RW
	15	z_en	Enables the feature on a per-axis basis	0x0	RW
no_motion					
0x5E: 0x04		settings_1	No-motion detection general configuration flags - part 1	0x00AA	
	10...0	threshold	Slope threshold value for No-motion detection in 5.11g format. Range is 0 to	0xAA	RW

			1g. Default value is 0xAA = 83mg = 0xAA * (1 / 2^11)mg.		
0x5E: 0x06		settings_2	No-motion detection general configuration flags - part 2	0x0005	
	12...0	duration	Defines the number of consecutive data points for which the threshold condition must be respected, for interrupt assertion. It is expressed in in 50 Hz samples (20 ms). Range is 0 to 163sec. Default value is 5=100ms.	0x5	RW
	13	x_en	Enables the feature on a per-axis basis	0x0	RW
	14	y_en	Enables the feature on a per-axis basis	0x0	RW
	15	z_en	Enables the feature on a per-axis basis	0x0	RW
step_counter					
0x5E: 0x08		settings_1	Step Counter setting	0x012D	
	15...0	param_1	Step Counter param 1	0x12D	RW
0x5E: 0x0A		settings_2	Step Counter setting	0x7BD4	
	15...0	param_2	Step Counter param 2	0x7BD4	RW
0x5E: 0x0C		settings_3	Step Counter setting	0x013B	
	15...0	param_3	Step Counter param 3	0x13B	RW
0x5E: 0x0E		settings_4	Step Counter setting	0x7ADB	
	15...0	param_4	Step Counter param 4	0x7ADB	RW
0x5E: 0x10		settings_5	Step Counter setting	0x0004	
	15...0	param_5	Step Counter param 5	0x4	RW
0x5E: 0x12		settings_6	Step Counter setting	0x7B3F	
	15...0	param_6	Step Counter param 6	0x7B3F	RW
0x5E: 0x14		settings_7	Step Counter setting	0x6CCD	
	15...0	param_7	Step Counter param 7	0x6CCD	RW
0x5E: 0x16		settings_8	Step Counter setting	0x04C3	
	15...0	param_8	Step Counter param 8	0x4C3	RW
0x5E: 0x18		settings_9	Step Counter setting	0x0985	
	15...0	param_9	Step Counter param 9	0x985	RW
0x5E: 0x1A		settings_10	Step Counter setting	0x04C3	
	15...0	param_10	Step Counter param 10	0x4C3	RW
0x5E: 0x1C		settings_11	Step Counter setting	0xE6EC	
	15...0	param_11	Step Counter param 11	0xE6EC	RW
0x5E: 0x1E		settings_12	Step Counter setting	0x460C	
	15...0	param_12	Step Counter param 12	0x460C	RW
0x5E: 0x20		settings_13	Step Counter setting	0x0001	
	15...0	param_13	Step Counter param 13	0x1	RW
0x5E: 0x22		settings_14	Step Counter setting	0x0027	
	15...0	param_14	Step Counter param 14	0x27	RW
0x5E: 0x24		settings_15	Step Counter setting	0x0019	
	15...0	param_15	Step Counter param 15	0x19	RW
0x5E: 0x26		settings_16	Step Counter setting	0x0096	
	15...0	param_16	Step Counter param 16	0x96	RW
0x5E: 0x28		settings_17	Step Counter setting	0x00A0	

	15...0	param_17	Step Counter param 17	0xA0	RW
0x5E: 0x2A		settings_18	Step Counter setting	0x0001	
	15...0	param_18	Step Counter param 18	0x1	RW
0x5E: 0x2C		settings_19	Step Counter setting	0x000C	
	15...0	param_19	Step Counter param 19	0xC	RW
0x5E: 0x2E		settings_20	Step Counter setting	0x3CF0	
	15...0	param_20	Step Counter param 20	0x3CF0	RW
0x5E: 0x30		settings_21	Step Counter setting	0x0100	
	15...0	param_21	Step Counter param 21	0x100	RW
0x5E: 0x32		settings_22	Step Counter setting	0x0001	
	15...0	param_22	Step Counter param 22	0x1	RW
0x5E: 0x34		settings_23	Step Counter setting	0x0003	
	15...0	param_23	Step Counter param 23	0x3	RW
0x5E: 0x36		settings_24	Step Counter setting	0x0001	
	15...0	param_24	Step Counter param 24	0x1	RW
0x5E: 0x38		settings_25	Step Counter setting	0x000E	
	15...0	param_25	Step Counter param 25	0xE	RW
0x5E: 0x3A		settings_26	Step Counter and Step Detector Settings	0x0000	
	9...0	watermark_level	Watermark level; the Step-counter will trigger output every time this number of steps are counted. Holds implicitly a 20x factor, so the range is 0 to 20460, with resolution of 20 steps. If 0, the output is disabled.	0x0	RW
	10	reset_counter	Flag to reset the counted steps. This is only interpreted if the step counter is enabled.	0x0	RW
	11	en_detector	Enables the Step Detector.	0x0	RW
	12	en_counter	Enables the Step Counter.	0x0	RW
	13	en_activity	Enables the activity detection(Running, Walking, Still)	0x0	RW
single_tap					
0x5E: 0x3C		settings	Single tap general configuration flags	0x0006	
	0	enable	Enables the feature	0x0	RW
	3...1	sensitivity	Configures single tap sensitivity, the range goes from 0 (high sensitive) to 7 (low sensitive).	0x3	RW
Double_tap					
0x5E: 0x3E		settings	Double tap general configuration flags	0x0006	
	0	enable	Enables the feature	0x0	RW
	3...1	sensitivity	Configures single tap sensitivity, the range goes from 0 (high sensitive) to 7 (low sensitive).	0x3	RW
wrist_ware_wakeup					
0x5E: 0x40		settings	Wrist ware wakeup configuration flags	0x0000	
	0	enable	Enables the feature	0x0	RW

general_settings					
0x5E: 0x42		config_id	Describes configuration identification code	0x0000	
	15...0	identification	Describes configuration identification code	0x0	R
0x5E: 0x44		axes_remapping	Describes axes remapping	0x0088	
	1...0	map_x_axis	Map the x axis to desired axis. **Value** **Name** **Description** 0x00 x axis Map to x-axis 0x01 y axis Map to y-axis 0x02 z axis Map to z-axis 0x03 reserved reserved	0x0	RW
	2	map_x_axis_sign	Map the x axis sign to the desired one. **Value** **Name** **Description** 0x00 non-inverted Clear this bit to keep the original direction of x axis 0x01 inverted Set this bit to invert the x axis	0x0	RW
	4...3	map_y_axis	Map the y axis to desired axis **Value** **Name** **Description** 0x00 x axis Map to x-axis 0x01 y axis Map to y-axis 0x02 z axis Map to z-axis 0x03 reserved reserved	0x1	RW
	5	map_y_axis_sign	Map the y axis sign to the desired one **Value** **Name** **Description** 0x00 non-inverted Clear this bit to keep the original direction of y axis 0x01 inverted Set this bit to invert the y axis	0x0	RW
	7...6	map_z_axis	Map the z axis to desired axis **Value** **Name** **Description** 0x00 x axis Map to x-axis 0x01 y axis Map to y-axis 0x02 z axis Map to z-axis 0x03 reserved reserved	0x2	RW
	8	map_z_axis_sign	Map the z axis sign to the desired one **Value** **Name** **Description** 0x00 non-inverted Clear this bit to keep the original direction of z axis 0x01 inverted Set this bit to invert the z axis	0x0	RW

Register (0x5F) INTERNAL_ERROR

DESCRIPTION: Internal error flags
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x5F) INTERNAL_ERROR**			
Bit	7	6	5	4
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			
Bit	3	2	1	0
Read/Write	n/a	R	R	n/a
Reset Value	0	0	0	0
Content	reserved	int_err_2	int_err_1	reserved

int_err_1: Internal error flag - long processing time, processing halted
int_err_2: Internal error flag - fatal error, processing halted

Register (0x6A) NVM_CONF

DESCRIPTION: NVM controller mode (Prog/Erase or Read only)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x6A) NVM_CONF**			
Bit	7	6	5	4
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			
Bit	3	2	1	0
Read/Write	n/a	n/a	RW	n/a
Reset Value	0	0	0	0
Content	reserved		nvm_prog_en	reserved

nvm_prog_en: Enable NVM programming

nvm_prog_en		
0x00	disable	disable
0x01	enable	enable

Register (0x6B) IF_CONF

DESCRIPTION: Serial interface settings
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x6B) IF_CONF**			
Bit	7	6	5	4
Read/Write	n/a	n/a	n/a	RW
Reset Value	0	0	0	0
Content	reserved			if_mode
Bit	3	2	1	0
Read/Write	n/a	n/a	n/a	RW
Reset Value	0	0	0	0
Content	reserved			spi3

spi3: Configure SPI Interface Mode for primary interface

spi3		
0x00	spi4	SPI 4-wire mode
0x01	spi3	SPI 3-wire mode

if_mode: Auxiliary interface configuration

if_mode		
0x00	p_auto_s_off	Auxiliary interface:off
0x01	p_auto_s_mag	Auxilary interface:Magnetometer

Register (0x6D) ACC_SELF_TEST

DESCRIPTION: Settings for the sensor self-test configuration and trigger
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x6D) ACC_SELF_TEST**			
Bit	7	6	5	4
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			
Bit	3	2	1	0
Read/Write	RW	RW	n/a	RW
Reset Value	0	0	0	0
Content	acc_self_test_amp	acc_self_test_sign	reserved	acc_self_test_en

acc_self_test_en: Enable accelerometer self-test

acc_self_test_en		
0x00	disabled	disabled
0x01	enabled	enabled

acc_self_test_sign: select sign of self-test excitation as

acc_self_test_sign		
0x00	negative	negative
0x01	positive	positive

acc_self_test_amp: select amplitude of the selftest deflection:

acc_self_test_amp		
0x00	low	low
0x01	high	high

Register (0x70) NV_CONF

DESCRIPTION: NVM backed configuration bits (check datasheet for details).
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x70) NV_CONF**			
Bit	7	6	5	4
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			
Bit	3	2	1	0
Read/Write	RW	RW	RW	RW
Reset Value	0	0	0	0
Content	acc_off_en	i2c_wdt_en	i2c_wdt_sel	spi_en

spi_en: disable the I2C and enable SPI for the primary interface, when it is in autoconfig mode (check datasheet for details).

spi_en		
0x00	disabled	I2C enabled
0x01	enabled	I2C disabled

i2c_wdt_sel: Select timer period for I2C Watchdog

i2c_wdt_sel		
0x00	wdt_short	I2C watchdog timeout after 1.25 ms
0x01	wdt_long	I2C watchdog timeout after 40 ms

i2c_wdt_en: I2C Watchdog at the SDA pin in I2C interface mode

i2c_wdt_en		
0x00	Disable	Disable I2C watchdog
0x01	Enable	Enable I2C watchdog

acc_off_en: Add the offset defined in the off_acc_x/y/z OFFSET register to filtered and unfiltered Accelerometer data

acc_off_en		
0x00	disabled	Disabled
0x01	enabled	Enabled

Register (0x71) OFFSET_0

DESCRIPTION: Offset compensation for Accelerometer X-axis (NVM backed)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x71) OFFSET_0**			
Bit	7	6	5	4
Read/Write	RW	RW	RW	RW
Reset Value	0	0	0	0
Content	off_acc_x			
Bit	3	2	1	0
Read/Write	RW	RW	RW	RW
Reset Value	0	0	0	0
Content	off_acc_x			

off_acc_x: Accelerometer offset compensation (X-axis).

Register (0x72) OFFSET_1

DESCRIPTION: Offset compensation for Accelerometer Y-axis (NVM backed)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x72) OFFSET_1**			
Bit	7	6	5	4
Read/Write	RW	RW	RW	RW
Reset Value	0	0	0	0
Content	off_acc_y			
Bit	3	2	1	0
Read/Write	RW	RW	RW	RW
Reset Value	0	0	0	0
Content	off_acc_y			

off_acc_y: Accelerometer offset compensation (Y-axis).

Register (0x73) OFFSET_2

DESCRIPTION: Offset compensation for Accelerometer Z-axis (NVM backed)
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x73) OFFSET_2**			
Bit	7	6	5	4
Read/Write	RW	RW	RW	RW
Reset Value	0	0	0	0
Content	off_acc_z			
Bit	3	2	1	0
Read/Write	RW	RW	RW	RW
Reset Value	0	0	0	0
Content	off_acc_z			

off_acc_z: Accelerometer offset compensation (Z-axis).

Register (0x7C) PWR_CONF

DESCRIPTION: Power mode configuration register
RESET: 0x03
DEFINITION (Go to register map):

Name	**Register (0x7C) PWR_CONF**			
Bit	7	6	5	4
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			
Bit	3	2	1	0
Read/Write	n/a	n/a	RW	RW
Reset Value	0	0	1	1
Content	reserved		fifo_self_wakeup	adv_power_save

adv_power_save		
0x00	aps_off	advanced power save disabled (fast clk always enabled).
0x01	aps_on	advanced power mode enabled (slow clk is active when no measurement is ongoing.)

fifo_self_wakeup		
0x00	fsw_off	FIFO read disabled in advanced power saving mode.
0x01	fsw_on	FIFO read enabled after interrupt in advanced power saving mode.

Register (0x7D) PWR_CTRL

DESCRIPTION: Sensor enable register
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x7D) PWR_CTRL**			
Bit	7	6	5	4
Read/Write	n/a	n/a	n/a	n/a
Reset Value	0	0	0	0
Content	reserved			
Bit	3	2	1	0
Read/Write	n/a	RW	n/a	RW
Reset Value	0	0	0	0
Content	reserved	acc_en	reserved	aux_en

aux_en		
0x00	mag_off	Disables the auxiliary sensor.
0x01	mag_on	Enables the auxiliary sensor.

acc_en		
0x00	acc_off	Disables the Accelerometer.
0x01	acc_on	Enables the Accelerometer.

Register (0x7E) CMD

DESCRIPTION: Command Register
RESET: 0x00
DEFINITION (Go to register map):

Name	**Register (0x7E) CMD**			
Bit	7	6	5	4
Read/Write	RW	RW	RW	RW
Reset Value	0	0	0	0
Content	cmd			
Bit	3	2	1	0
Read/Write	RW	RW	RW	RW
Reset Value	0	0	0	0
Content	cmd			

cmd: Available commands (Note: Register will always read as 0x00):

cmd		
0xa0	nvm_prog	Writes the NVM backed registers into NVM
0xb0	fifo_flush	Clears all data in the FIFO, does not change FIFO_CONFIG and FIFO_DOWNS registers
0xb6	softreset	Triggers a reset, all user configuration settings are overwritten with their default state

3. Legal disclaimer

3.1. Engineering samples

Engineering Samples are marked with an asterisk (*), (E) or (e). Samples may vary from the valid technical specifications of the product series contained in this data sheet. They are therefore 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 product series. Bosch Sensortec assumes no liability for the use of engineering samples. The Purchaser shall indemnify Bosch Sensortec from all claims arising from the use of engineering samples.

3.2. Product use

Bosch Sensortec products are developed for the consumer goods industry. They may only be used within the parameters of this product data sheet. They are not fit for use in life-sustaining or safety-critical systems. Safety-critical systems are those for which a malfunction is expected to lead to bodily harm, death or severe property damage. In addition, they shall not be used directly or indirectly for military purposes (including but not limited to nuclear, chemical or biological proliferation of weapons or development of missile technology), nuclear power, deep sea or space applications (including but not limited to satellite technology).

The resale and/or use of Bosch Sensortec products are at the purchaser's own risk and his own responsibility. The examination of fitness for the intended use is the sole responsibility of the purchaser.

The purchaser shall indemnify Bosch Sensortec 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 Sensortec and reimburse Bosch Sensortec for all costs in connection with such claims.

The purchaser accepts the responsibility to monitor the market for the purchased products, particularly with regard to product safety, and to inform Bosch Sensortec without delay of all safety-critical incidents.

3.3. Application examples and hints

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

4. Document history and modification

Rev. No	Chapter	Description of modification/changes	Date
1.0		Document creation	August 2019

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

contact@bosch-sensortec.com
www.bosch-sensortec.com

Document number: BST-MAS-AN032-00
Revision_1.0_082019