MAX21100 USER GUIDE

Revision 1.0

July 2015

maxim integrated™

Rev. 1.0, Jul 2015

CONTENTS

TABLES

FIGURES

1 Introduction

MEMS sensors are revolutionizing the way people interact with everyday technology, making it easier and more user-friendly. Maxim can leverage its analog integration expertise to develop and manufacture new breakthrough MEMS sensors to decrease, operate at lower power, and perform more accurately than ever.

Owning the entire supply chain, Maxim brings its customers complete, reliable, and cost-effective solutions, ensuring prompt time-to-volume and time-to-market to effectively address high-volume applications in consumer and industrial market segments.

Thanks to its leadership in analog integration and its manufacturing experience in MEMS, Maxim is capable of high-volume production to meet the market's demands. Maxim’s manufacturing expertise and highest quality standards also guarantee high performance and product reliability.

Every MEMS sensor is tested and trimmed in factory so that for most consumer applications, no additional sensor calibrations are required. The end user can quickly verify the sensor’s operation without physically tilting or rotating the sensor thanks to the built-in self-test feature that accelerates the time to market for mass production.

This User Guide provides a clear picture of the guidelines for its use in consumer applications and a comprehensive description of his unique features. The final section of this guide presents the structure of the register file and the purpose of each field or every register.

2 Nomenclature

ODR	Output data rate
BW	Bandwidth
FS	Fulll scale
UI	User interface
OIS	Optical image stabilization
MSB	Most significant byte
MSb	Most significant bit
LSB	Least significant byte
LSb	Least significant bit
HPF	Highpass filter
LPF	Lowpass filter
dps	Degrees per seconds
RFU	Reserved for future use

3 MAX21100 Description

The MAX21100 is a monolithic 3-axis gyroscope plus 3-axis accelerometer Inertial Measurement Unit (IMU) with integrated 9-axis sensor fusion using proprietary Motion Merging Engine (MME) for handset and tablet applications, game controllers, motion remote controls, and other consumer devices.
The MAX21100 is the industry's most accurate 6+3 DoF inertial measurement unit available in a 3mm x 3mm x 0.83mm package and capable of working with a supply voltage as low as 1.71V. The MAX21100 can interface an external magnetometer through a dedicated I^2C master. The internal Motion Merging Engine can be flexibly configured to get the most suitable accuracy power trade-off. The MAX21100 is available in a 16-lead plastic land grid array (LGA) package and can operate within a temperature range of -40°C to +85°C.

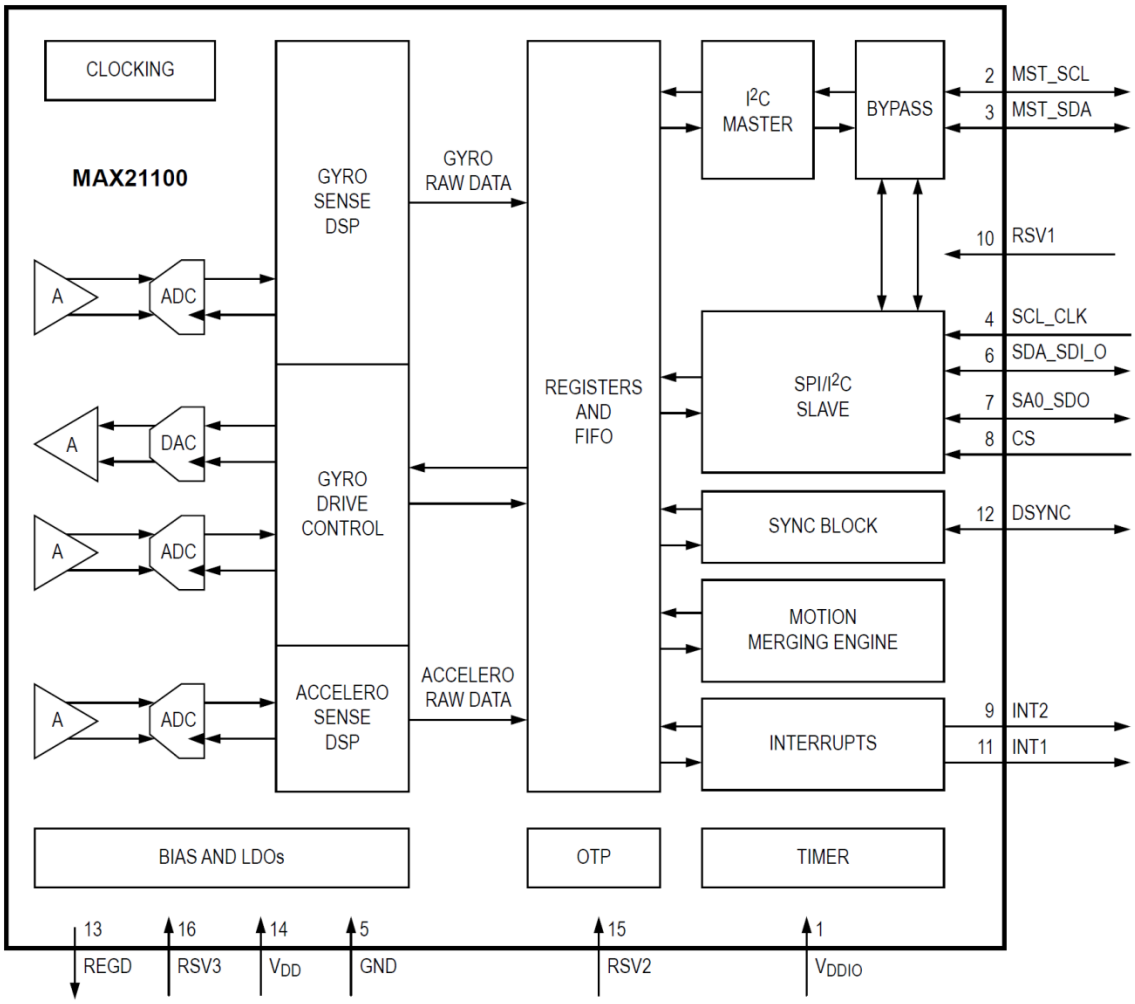

Figure 1: Block Diagram

4 Pin Description

Table 1: Pin Description

Pin	Name	Function
1	V_{DDIO}	Interface and Interrupt Pad Supply Voltage.
2	MST_SCL	I^2C Master Serial Clock. User-selectable 6kΩ internal pullup.
3	MST_SDA	I^2C Master Serial Data. User selectable 6kΩ internal pullup.
4	SCL_CLK	SPI and I^2C Slave Clock. When in I^2C mode, the IO has selectable antispike filter and delay to ensure correct hold time.
5	GND	Power-Supply Ground.
6	SDA_SDI_O	SPI In/Out Pin and I^2C Slave Serial Data. When in I^2C mode, the IO has selectable antispike filter and delay to ensure correct hold time.
7	SA0_SDO	SPI Serial Data Out and I^2C Slave Address LSB.
8	CS	SPI Chip Select/Serial Interface Selection.
9	INT2	Second Interrupt Line.
10	RSV1	Reserved. Must be connected to GND.
11	INT1	First Interrupt Line.
12	DSYNC	Data Synchronization Pin. Dynamically changes the MAX21100 power mode. Synchronizes data with external clock (e.g., GPS/camera) with various options. Synchronizes data with an external event.
13	REGD	Internal Regulator Output (2.2V max). Connect a 100nF capacitor to this pin to ensure proper device operation.
14	V_{DD}	Analog Power Supply. Bypass to GND with a 0.1µF capacitor and one 10µF.
15	RSV2	Reserved. Must be tied to V_{DD} in the application.
16	RSV3	Reserved. Leave unconnected.

5 I²C Interface

The MAX21100 can operate as an I²C slave device when communicating to the system processor, which thus acts as the master. The maximum bus speed is 3.4MHz (I²C HS); this reduces the amount of time the system processor is kept busy in supporting the exchange of data.

To connect a MAX21100 device to an I²C master, the SDA_SDI_O pin of the MAX21100 device must be connected to the SDA pin of the I²C master and the SCL_CLK pin of the MAX21100 device must be connected to the SCL pin of the I²C master. Both SDA and SCL lines must be connected to a pull-up resistor. The SA0_SDO pin must be connected to V_{DD} or GND to configure the MAX21100 I²C slave address (see *Table 3: I2C Device Addresses*).

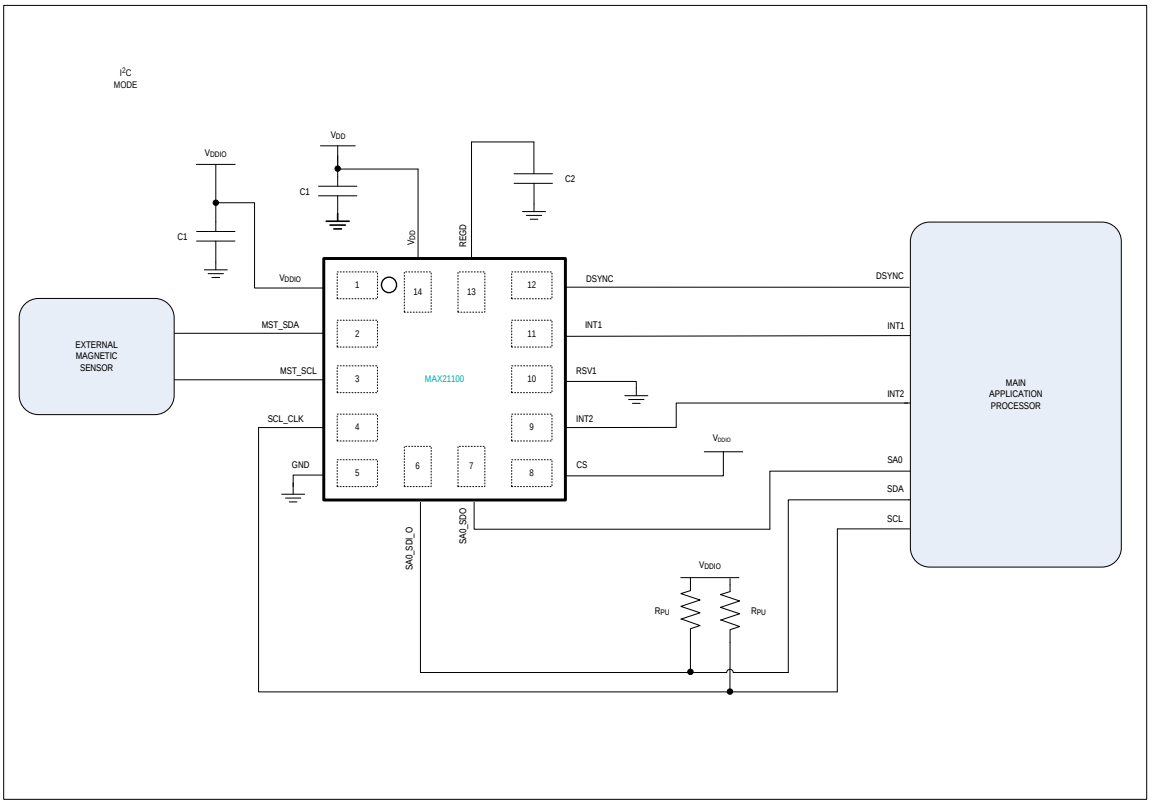

Figure 2: I²C Interface Connection to an Application Processor

Table 2: I²C External Component Properties

Component	Label	Specification	Quantity
V_{DD}/V_{DDIO} Bypass Capacitor	C1	Ceramic, X7R, 100nF ±10%, 4V	2
REGD Capacitor	C2	Ceramic, X7R, 100nF ±10%, 2V	1
Pullup Resistor	R_{PU}	1.1kΩ/10kΩ (min/max)	2

1.1 I²C Protocol

To start an I²C request, the master sends a START condition (S), followed by the MAX21100's I²C address. Then, the master sends the address of the register to be programmed. Finally, the master terminates the communication by issuing a STOP condition (P) to relinquish the control of the bus, or a repeated START condition (Sr) to keep controlling it.

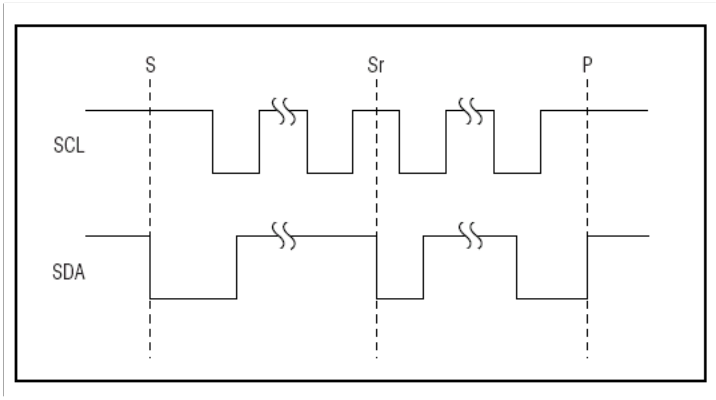

Figure 3: START (S), STOP (P), and Repeated START (Sr) Conditions

1.2 Slave Address

The slave address is used to identify the MAX21100 device in I²C communications. The address is defined as the seven most significant bits followed by the read/write bit. Set the read/write bit to 1 to request a read operation, or 0 to request a write operation (see the table below).

Table 3: I²C Device Addresses

I²C Base Address	SA0_SDO pin	R/W bit	Resulting Address
0x2C (6bit)	0	0	0xB0
0x2C	0	1	0xB1
0x2C	1	0	0xB2
0x2C	1	1	0xB3

1.3 Acknowledge

The acknowledge bit is sent after each byte. This bit allows the receiver to notify the transmitter that the byte has been received correctly and another byte can be sent. The master generates all clock pulses, including the ninth clock pulse of the acknowledge bit (8 bits of data + acknowledge bit).
To allow the receiver to send the acknowledge, the transmitter releases the SDA line during the acknowledge pulse, so that the receiver can pull the SDA line low during the ninth clock pulse to signal an acknowledge (ACK), or release the SDA line (high) to signal a not acknowledge (NACK).
The NACK is sent if the device is busy or a system fault occurs. It is also used by the master to signal the end of the transfer during a read operation.

1.4 Register Address

The I²C register address for the MAX21100 is composed by 6 address bits and 1 bit whose meaning can be configured as:

- **Autoincrement:** If 0, in case of burst operation the initial register address is autoincremented after every data byte; if 1, the operation is executed always on the same register.
- **Even parity:** This bit represents the even parity computed on the 6 bits of the register address.
- **Odd parity:** This bit represents the odd parity computed on the 6 bits of the register address.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
N/A	MS/parity	Address of the register you want to read or write					

1.5 I²C Operations

5.1.1 Write One Register

To write one register, the following steps must be executed:

1. The master sends a START condition.
2. The master sends the slave address (7 bits) plus a write bit (low).
3. If the slave address matches, the MAX21100 asserts an ACK on the data line.
4. The master sends the address (8 bits) of the register to be written.
5. The MAX21100 asserts an ACK on the data line ***only if the address is valid (NACK if not)***.
6. The master sends 8 bits of data.
7. The MAX21100 asserts an ACK on the data line.
8. The master generates a STOP condition.

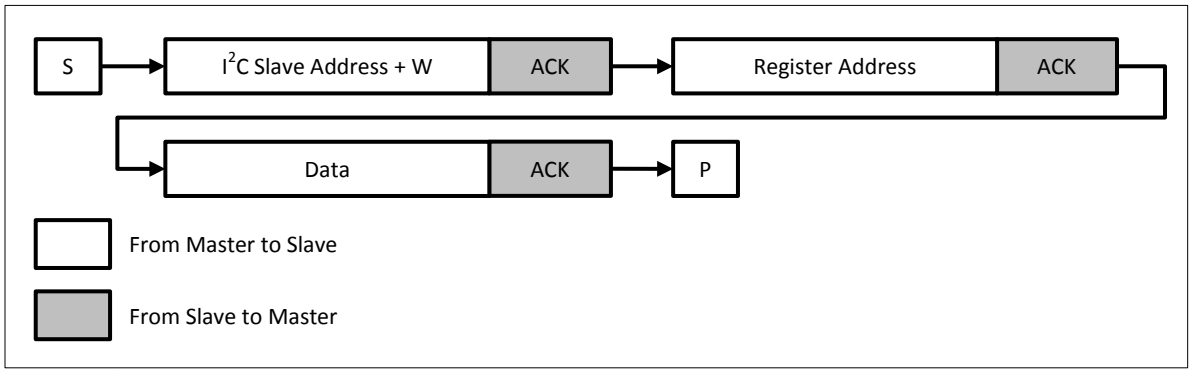

Figure 4: I²C Write One Byte

5.1.2 Write Multiple Registers

To execute a write of *n* consecutive registers, the following steps must be executed:

1. The master sends a START condition.
2. The master sends the slave address (7 bits) plus a write bit (low).
3. If the slave address matches, the MAX21100 asserts an ACK on the data line.
4. The master sends the register address of the first register to be written, setting the MS/parity bit to 0 if it is configured as autoincrement bit (see the *Register Address* section).
5. The MAX21100 asserts an ACK on the data line ***only if the address is valid (NACK if not)***.
6. The master sends 8 bits of data.
7. The MAX21100 asserts an ACK on the data line.
8. Repeat 6 and 7 *n* times.
9. The master generates a STOP condition.

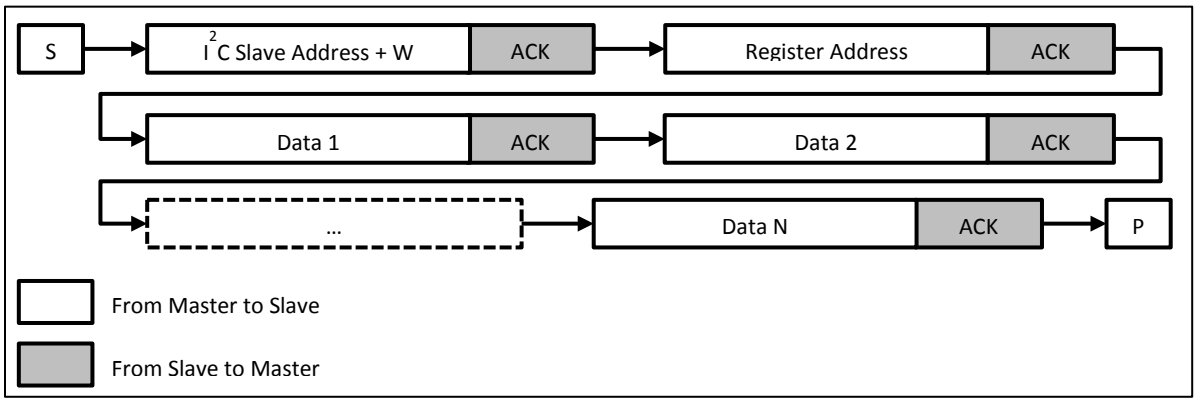

Figure 5: I²C Write a Burst of Data

5.1.3 Read One Register

To read one register, the following steps must be executed:

1. The master sends a START condition.
2. The master sends the slave address (7 bits) plus a write bit (low).
3. If the slave address matches, the MAX21100 asserts an ACK on the data line.
4. The master sends the address (8 bits) of the register to be read.
5. The MAX21100 asserts an ACK on the data line ***only if the address is valid (NACK if not)***.
6. The master sends a restart condition.
7. The master sends the slave address (7 bits) plus a read bit (high).
8. If the slave address matches, the MAX21100 asserts an ACK on the data line.
9. The MAX21100 sends the content of the requested register (8 bits).
10. The master asserts a NACK on the data line.
11. The master generates a STOP condition.

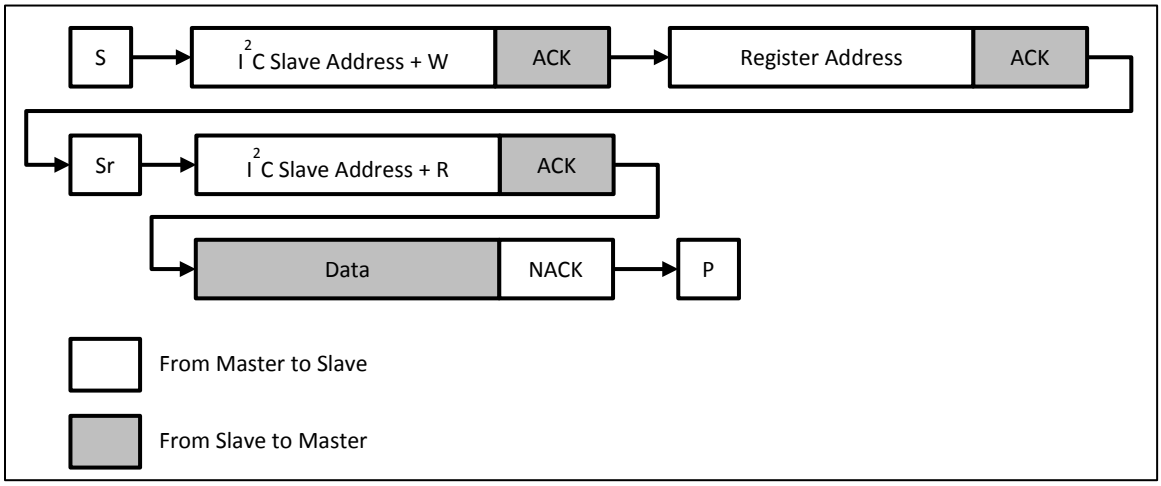

Figure 6: I²C Read One Byte

5.1.4 Read Multiple Registers

To execute a read of *n* consecutive registers, the following steps must be executed:

1. The master sends a START condition.
2. The master sends the slave address (7 bits) plus a write bit (low).
3. If the slave address matches, the MAX21100 asserts an ACK on the data line.
4. The master sends the register address of the first register to be read, setting the MS/Parity bit to 0 if it is configured as autoincrement bit (see the *Register Address* section).
5. The MAX21100 asserts an ACK on the data line ***only if the address is valid (NACK if not)***.
6. The master sends a restart condition.
7. The master sends the slave address (7 bits) plus a read bit (high).
8. If the slave address matches, the MAX21100 asserts an ACK on the data line.
9. The MAX21100 sends the content of the requested register (8 bits).
10. The master asserts a ACK on the data line.
11. Repeat 9 and 10 *n-1* times.
12. The MAX21100 sends the content of the requested register (8 bits);
13. The master asserts a NACK on the data line;
14. The master generates a STOP condition.

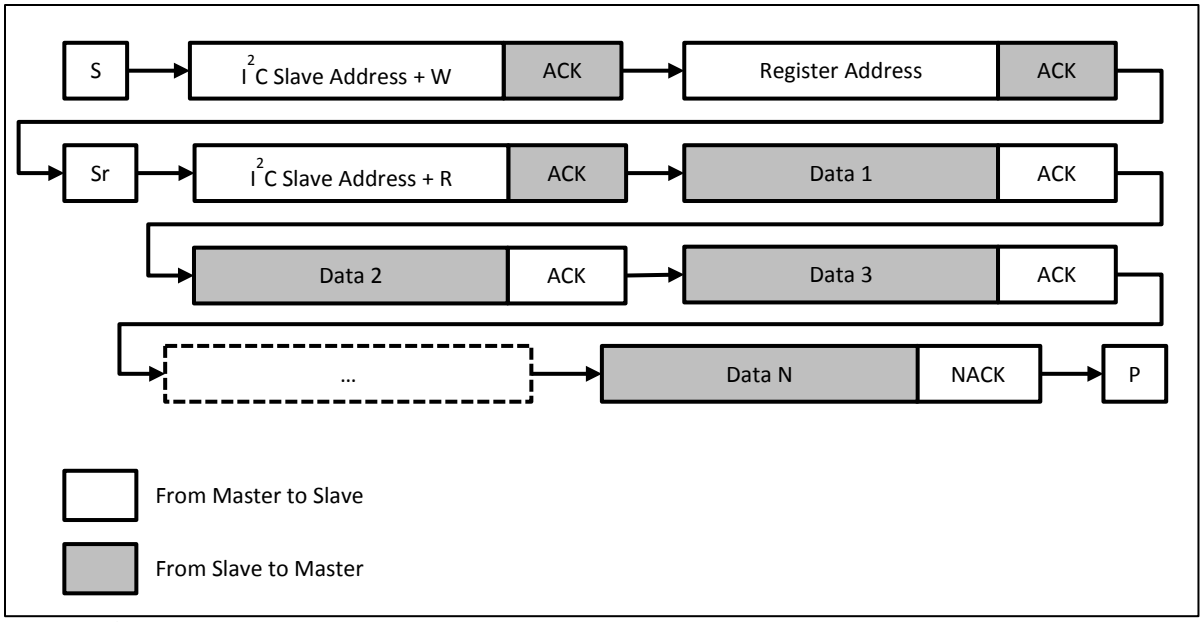

Figure 7: I²C Read a Burst of Data

5.2 Configuration Registers

Field	Description
slv_pu_dis	Enables/disables the internal pullups of the slave pads.
strong_slave	Increases the I²C slave pads speed.
i2c_setting	Selects the master interface type.
i2c_off	Turns on/off the I²C interface.
if_parity	Sets the meaning of the bit 6 of the address during the communication from the master to the slave.
parity_error	Indicates if parity error occurred..
parity_rst	Triggers a reset of the parity.

6 SPI Interface

The MAX21100 SPI interface can operate up to 10MHz, in both 3-wires (half duplex) and 4-wires mode (full duplex). It is recommended to set the I2C_OFF bit at address 0x16 if the MAX21100 is used together with other SPI devices to avoid the possibility to switch inadvertently into I^2C mode when traffic is detected with the CS unasserted.

To connect a MAX21100 device to an SPI master, the CS pin of the MAX21100 device must be connected to the CSn pin of the SPI master, the SDA_SDI_O pin of the MAX21100 device must be connected to the MOSI pin of the SPI master, the SA0_SDO pin of the MAX21100 device must be connected to the MISO pin of the SPI master and the SCL_CLK pin of the MAX21100 device must be connected to the SCLK pin of the SPI master.

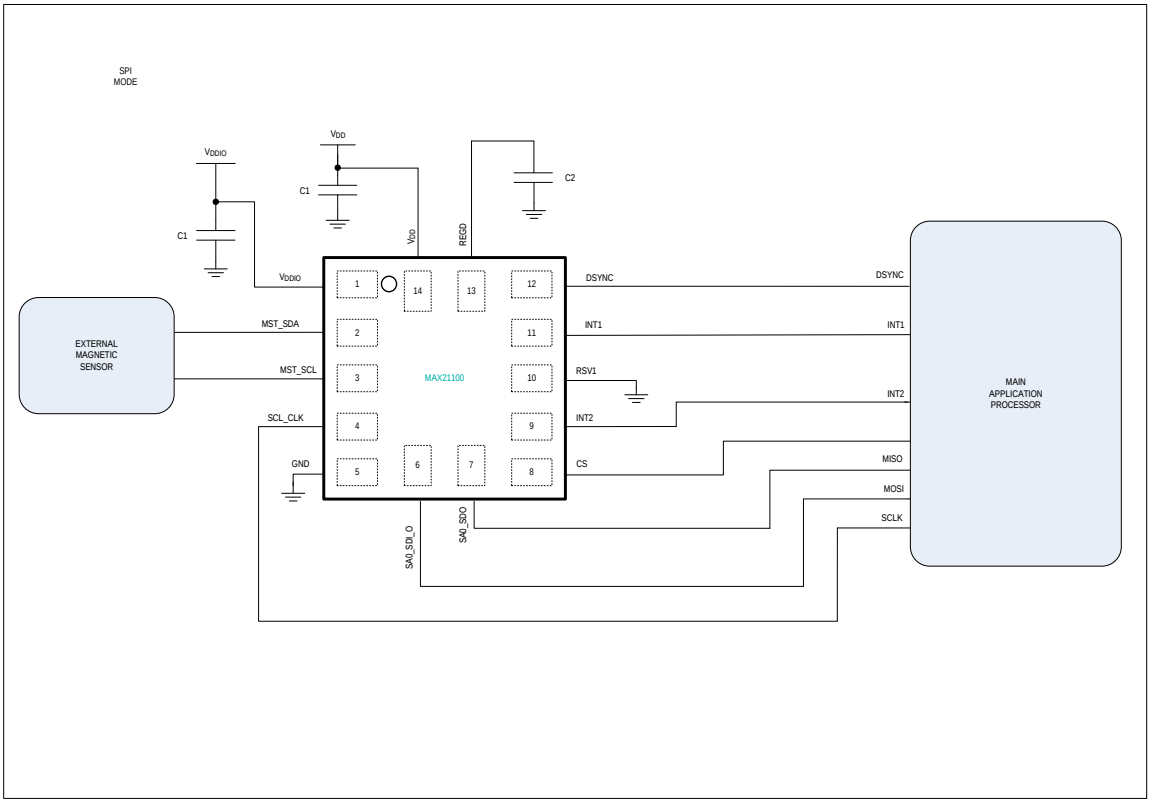

Figure 8: SPI Interface Connection to an Application Processor

Table 4: SPI External Component Properties

Component	Label	Specification	Quantity
V_{DD}/V_{DDIO} Bypass Capacitor	C1	Ceramic, X7R, 100nF ±10%, 4V	2
REGD Capacitor	C2	Ceramic, X7R, 100nF ±10%, 2V	1

6.1 SPI Protocol

CSn is the serial port enable pin and is controlled by the SPI master. It goes low at the start of the transmission and returns high at the end.
SCLK is the serial port clock and is controlled by the SPI master. It is kept high when CSn is high (no transmission). MISO and MOSI are, respectively, the serial port master data input and output. These lines are driven at the falling edge of SCLK and are sampled at the rising edge of SCLK.
Both the read register and write register commands are completed in 16 clock pulses, or in multiples of 8 in case of multiple read/write bytes.

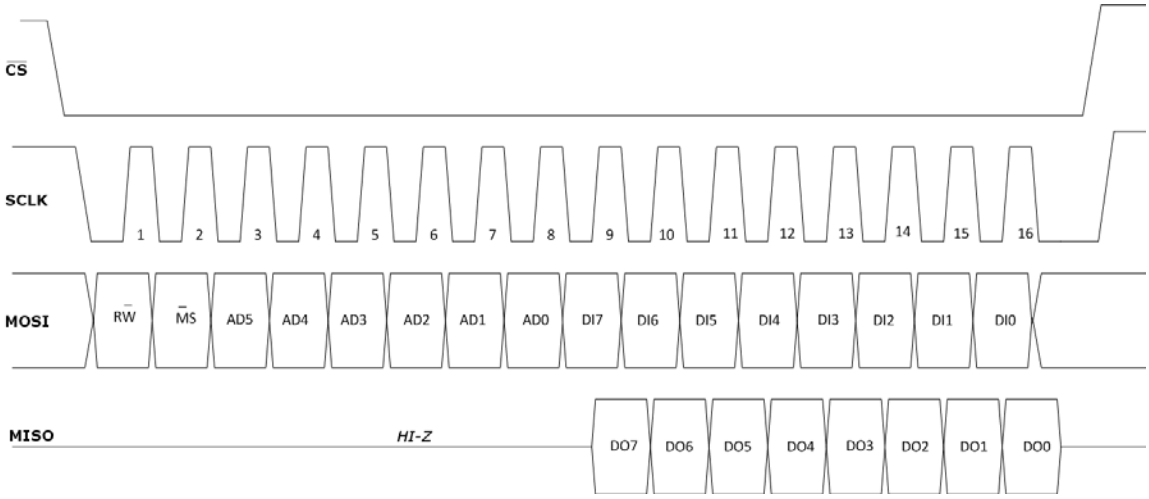

Figure 9: SPI Protocol

6.2 Register Address

The SPI register address for the MAX21100 is composed by 6 bits of address, 1 bit to select the direction of the operation (‘1’ to perform a read operation, ‘0’ to perform a write operation) and 1 bit whose meaning can be configured as:

- **Autoincrement:** If 0, in case of burst operation the initial register address is autoincremented after every data byte; if 1, the operation is executed always on the same register.
- **Even parity:** This bit represents the even parity computed on the 6 bits of the register address.
- **Odd parity:** This bit represents the odd parity computed on the 6 bits of the register address.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
R/W	MS/parity	Address of the register you want to read or write					

Here are shown some examples:

Address	Count of 1 bit	8 bits including parity	
		Even	Odd
000000 (0x00)	0	*x*0000000	x1000000
100000 (0x20)	1	x1100000	x0100000
100011 (0x23)	3	x1100011	x0100011
111111 (0x3F)	6	x0111111	x1111111

Bit 7, indicated with *x*, can be 0 or 1, depending on if you want to write or read the correspondent register. For further explanation:

- To **write** register **0x00**, using **odd parity,** send byte **01000000** (0x40) from the MCU.
- To **read** register **0x20**, using **even parity,** send byte **11100000** (0xE0) from the MCU.
- To **read** register **0x23**, using **odd parity,** send byte **10100011** (0xA3) from the MCU.
- To **read** register **0x3F**, using **even parity,** send byte **10111111** (0xBF) from the MCU.

If the parity mode is enabled, when the device receives the above bytes from the MCU, it tries to calculate the parity bit. The result of this computation is then reported to the user through a register field of the register map.

6.3 Configuration Registers

Field	Description
slv_pu_dis	Enables/disables the internal pullups of the slave pads.
strong_slave	Increases the I^2C slave pads speed.
i2c_setting	Selects the master interface type.
spi_3_wire	Sets the interface type of the SPI.
if_parity	Sets the meaning of the bit 6 of the address during the communication from the master to the slave.
parity_error	Indicates if parity error occurred.
parity_rst	Triggers a reset of the parity.

7 Interrupts

The MAX21100 is equipped with an interrupt module to control a set of interrupt flags and two interrupt lines (INT1 and INT2).
This module allows to:

1. Configure the behavior of the interrupt lines (INT1 and INT2).
2. Map each interrupt flag to one of both the interrupt lines.
3. Create a conditional interrupt (rate interrupt) based on four thresholds.

7.1 Interrupt Flags

The interrupt module provides several interrupt flags:

- **DATA_READY:** It is triggered when a new data on the active channels is available.
  It can be configured to work in one of the following modalities:
  - **ALL**: It is cleared after all the active channels are read. Data are not updated until the clear operation is accomplished.
  - **ANY**: It is cleared when at least a byte of one of the active channels is read. Data are updated independently from the clear operation.
  - **STATUS**: It is cleared when status register is read. Data are not updated until the clear operation is accomplished.
- **FIFO_EMPTY:** It is triggered when no new data on the FIFO queue is available.
- **FIFO_OVERRUN:** It is triggered when older unread FIFO data are lost.
- **FIFO_THRESHOLD:** It is triggered when the number of available data in FIFO exceeds a configured threshold (see the *FIFO* section).
- **INT_AND:** See the *Rate Interrupts* section.
- **INT_OR:** See the *Rate Interrupts* section.
- **OTP_DOWNLOADING:** It is triggered when the OTP is being downloaded.

7.2 Interrupt Lines

An interrupt line is a dedicated pin where a notification to an external application processor can be provided. The MAX21100 is equipped with two interrupt lines that can be configured independently.
For each interrupt line, it is possible to:

- Enable/disable them.
- Set the active level to low.
- Set the output type to push-pull or open-drain configuration.

To map an interrupt flag to an interrupt line, it is necessary to enable it through the mask registers: by setting its corresponding bit to 1 on the bit-mask, the interrupt flag is mapped to the related interrupt line. The output of the two INT1 and INT2 interrupt lines is then computed by applying the OR operator to all the enabled interrupt flags contained in latched interrupt status register. Note that the enable is not applied to the register, so it contains also the value of the interrupt flags that are not enabled.

An interrupt line can be configured to keep the status until the master requests to clear it by reading or writing the flag (latched). The lines can be also configured to autoclear its status after a period of time where the duration to clear the interrupt is programmable (timeout). The only exception is the **DATA_READY** flag that is always unlatched regardless of the selected clearing mode.

7.3 Rate Interrupts

The rate interrupt allows an interrupt event to be generated when the gyroscope/accelerometer data falls inside a range of values defined by a set of thresholds. By setting the absolute value of the threshold (|TH|) for each axis, four ranges are defined:

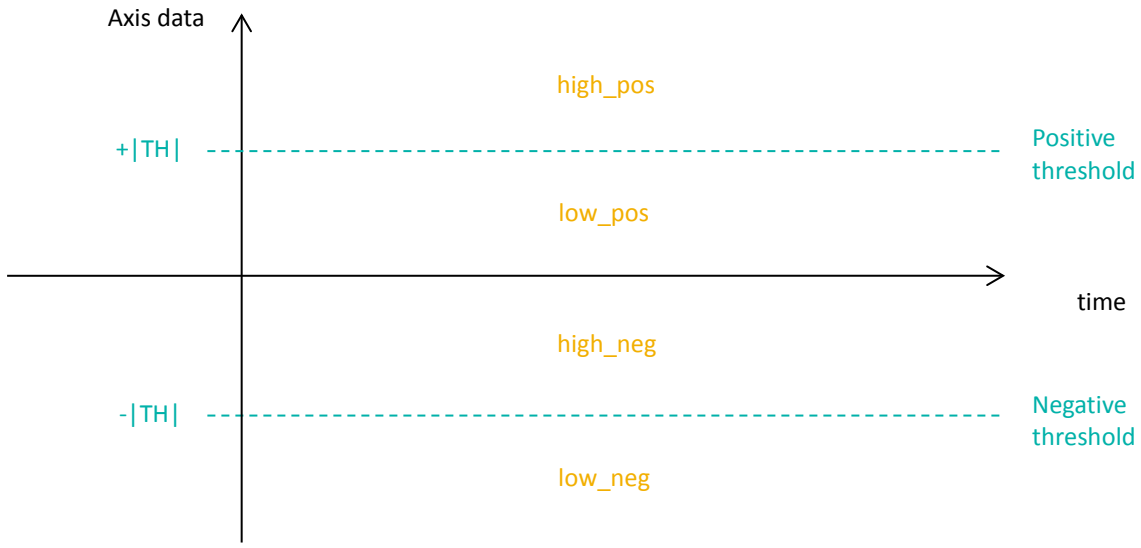

Figure 10: Rate Interrupt ranges

For each range, it is possible to select if it contributes to the generation of the rate interrupt for each axis. Then, the rate interrupts are computed as follow:

- **INT_AND:** Active if the defined conditions are satisfied for all the axes at the same time.
- **INT_OR:** Active if the defined conditions are satisfied for at least one of the axes.

The threshold absolute value can be set by writing to the *INT_REF_X*, *INT_REF_Y*, and *INT_REF_Z* registers. Those registers represent the most significant byte of the real threshold; so, to compute the actual absolute value of the threshold, the register value must be multiplied by 256. If the application requires a better resolution, it is possible to specify a 16-bit threshold by setting the *int_single_ref* register field; in this case, all the axes share the same threshold that is defined as the combination of *INT_REF_X* and *INT_REF_Y* where the first one is the most significant byte and the last one is the least significant byte of the threshold.
Once the thresholds are defined, the *INT_MSK_X*, *INT_MSK_Y*, and *INT_MSK_Z* permits to select which range contributes to the generation of the rate interrupts for each *axis* as follow:

- ***int_axis_high_pos_en*:** If enabled, the condition is satisfied if the data of the axis falls in the high_pos range.
- ***int_axis_high_neg_en*:** If enabled, the condition is satisfied if the data of the axis falls in the high_neg range.
- ***int_axis_low_pos_en*:** If enabled, the condition is satisfied if the data of the axis falls in the low_pos range.

- **int_*axis*_low_neg_en:** If enabled, the condition is satisfied if the data of the axis falls in the low_neg range.

Then, the desired axes must be enabled by setting to 1 the corresponding bit of the *int_mask_xyz_and* for the **INT_AND** and *int_mask_xyz_or* for the **INT_OR**.

Through the *INT_MSK_X*, *INT_MSK_Y*, and *INT_MSK_Z* registers, it is also possible to read the current value of the conditions, regardless of the content of the *int_mask_xyz_or* and *int_mask_xyz_and* register fields. Each of these values can be configured as latched by setting the *int_freeze* register field. For the rate interrupts, it is possible also to define a debounce value by defining the number of samples the axis data has to satisfy the condition before asserting the corresponding interrupt. Those values can be set in the *INT_DEB_X*, *INT_DEB_Y*, and *INT_DEB_Z* registers. If the required value is greater than 255, it is possible to define a 16-bit debounce value by setting *int_single_deb* register field; in this case, all the conditions share the same debounce value that is defined as the combination of *INT_DEB_X* and *INT_DEB_Y*, where the first one is the most significant byte and the last one is the least significant byte of the debounce value.

7.4 Configuration Registers

Function	Register/Field	Description
Pads	*int1_pd_en, int1_pu_en*	Enable/disable the internal pullup/down of the INT1 pad.
	int2_pd_en, int2_pu_en	Enable/disable the internal pullup/down of the INT2 pad.
	strong_int_aux	Increases the interrupt pads speed.
Functional	*dr_rst_mode*	Sets the DATA_READY interrupt reset and data-out update mode.
	INT_CFG_2	INT1/INT2 generic configuration register
	INT_TMO	Interrupts latch mode and duration (timeout) configuration
	INT_STS_UL, INT_STS	Interrupt status (unlatched and latched)
	INT_MSK	Interrupt enables
	msk_acc_int_d_rdy, msk_gyr_int_d_rdy	Enable the gyroscope/accelerometer contribution to DATA_READY interrupt assertion
Rate Interrupt	*INT_REF_X, INT_REF_Y, INT_REF_Z*	Rate interrupt thresholds for each axis
	INT_DEB_X, INT_DEB_Y, INT_DEB_Z	Rate interrupt debounce value for each axis
	INT_MSK_X, INT_MSK_Y, INT_MSK_Z	Rate interrupt thresholds ranges enables and status registers for each axis
	int_freeze	Rate interrupt latch configuration
	int_mask_xyz_and, int_mask_xyz_or	Axis enables for **INT_AND** and **INT_OR**, respectively
	sns_intp_fsc	Gyroscope full-scale used by the rate interrupt
	sns_intp_hpf	Gyroscope HPF used by the rate interrupt
	int_single_ref	Enable a single, 16-bit rate interrupt threshold for all the axes
	int_single_deb	Enable a single, 16-bit rate interrupt debounce value for all the axes
	int_src_cfg	Select the rate interrupt source

8 Reading Data from MAX21100

The MAX21100 sensor output data can be read by the host processor in two different mechanisms: synchronous (polling) and asynchronous (interrupt) reading methods.

8.1 Synchronous Reading

In synchronous reading, the host actively samples the status of the *gyro_dr* and *acc_dr* flags to check the status of the data output. When one (or both) of them is set, a new data is ready to be read, so the host can read the new data related to the asserted flag from the output registers.

8.2 Asynchronous Reading

In asynchronous reading, the host does not need to continuously monitor the status of the data output, because the data availability is notified through the interrupt line.

Only after the **DATA_READY** interrupt is asserted, the host can read the *gyro_dr* and *acc_dr* flags to verify which data are available and then read the corresponding output registers.

It is possible to configure which sensor (gyroscope or accelerometer) contributes to the **DATA_READY** interrupt assertion. If both the gyroscope and accelerometer are mapped to the **DATA_READY** signal, the interrupt is asserted when a new data of at least one sensor is available.

9 FIFO

The MAX21100 embeds a 128-byte data FIFO. This allows a consistent power saving for the system since the host processor does not need to continuously extract the data from the sensor, but it can wake up only when needed and burst the data out from the FIFO.
The data order in the FIFO depends on the endian setting:

- **Big Endian:** *GYRO_X_H, GYRO_X_L, GYRO_Y_H, GYRO_Y_L, GYRO_Z_H, GYRO_Z_L, ACC_X_H, ACC_X_L, ACC_Y_H, ACC_Y_L, ACC_Z_H, ACC_Z_L*
- **Little Endian:** *GYRO_X_L, GYRO_X_H, GYRO_Y_L, GYRO_Y_H, GYRO_Z_L, GYRO_Z_H ACC_X_L, ACC_X_H, ACC_Y_L, ACC_Y_H, ACC_Z_L, ACC_Z_H*

9.1 FIFO Modes Description

The FIFO buffer can work according to four main different modes: off, normal, interrupt, and snapshot. When configured in snapshot mode, it offers the ideal mechanism to capture the data following a rate interrupt event.

Both normal and Interrupt modes can be optionally configured to operate in overrun mode, depending on whether, in case of buffer underrun, newer or older data are lost.

Various FIFO status flags can be enabled to generate interrupt events on INT1/INT2 pins.

9.1.1 FIFO Off Mode

In this mode, the FIFO is turned off; data are stored only in the data registers. No data are available from the FIFO if read. When the FIFO is turned off, there are two options to use the device: synchronous or asynchronous reading.

9.1.2 Normal Mode

The behavior of the FIFO in normal mode varies depending on the **Overrun** setting.

The following paragraphs show a description of the behavior for both settings of the overrun. For the subsequent sections, the following descriptions are useful for the user's understanding of the FIFO:

- **Write Pointer (WP):** The write pointer is the address number where the next data is written in the FIFO. Whenever there is a new data is written, the write pointer is incremented by 1.
- **Read Pointer (RP):** The read pointer is the address number from where the first data in the FIFO is read. Whenever there is a new data is read, the read pointer is incremented by 1.
- **Level:** Level tells the difference between the write and read pointers, which also provides the total number of data available in the FIFO.

9.1.2.1 Stop On Full

1. The FIFO is turned on.
2. The FIFO is filled with the data at the selected output data rate (ODR).
3. When the FIFO is full, an interrupt can be generated.

4. When the FIFO is full, all the new incoming data is discharged. Reading only a subset of the data already stored into the FIFO keeps locked the possibility for new data to be written.
5. Only if all the data are read, the FIFO restarts saving data. If the communication speed is high, data loss can be prevented.
6. In order to prevent a FIFO full condition, the required condition is to complete the reading of the data set before the next **DATA_READY** occurs.
7. If this condition is not guaranteed, data can be lost.

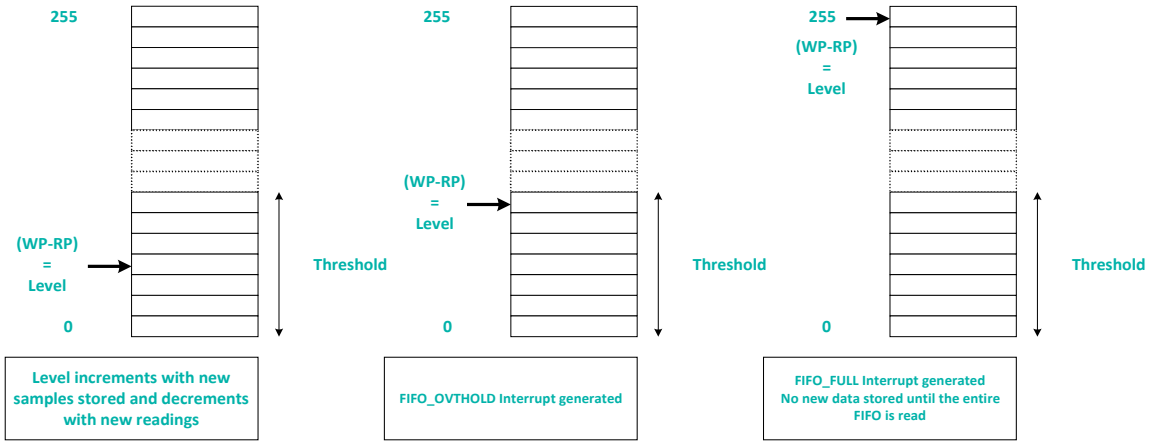

Figure 11: FIFO Normal Mode, Stop on Full

9.1.2.2 Overwrite

1. The FIFO is turned on.
2. The FIFO is filled with the data at the selected ODR.
3. When the FIFO is full, an interrupt can be generated.
4. When the FIFO is full, the oldest data will be overwritten with the new ones.
5. If communication speed is high, data integrity can be preserved.
6. To prevent a data lost condition, the requirement is to complete the reading of the data set before the next **DATA_READY** occurs.
7. If this condition is not guaranteed, data can be overwritten.
8. When an overrun condition occurs, the reading pointer is forced to the writing pointer -1 to ensure only older data are discarded and newer data have a chance to be read.

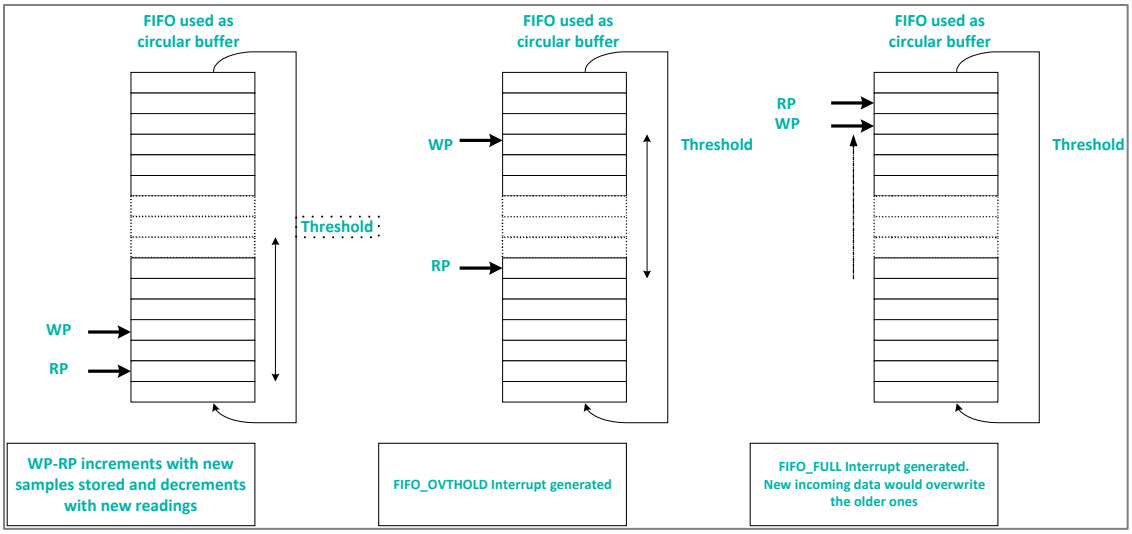

Figure 12: FIFO Normal Mode, Overwrite

9.1.3 Interrupt Mode

The behavior of the FIFO in interrupt mode varies depending on the **Overrun** setting. The following paragraphs show a description of the behavior for both settings of the overrun.

9.1.3.1 Stop on Full

1. FIFO is initially disabled. Data is stored only in the data registers.
2. When a rate interrupt (either **INT_OR** or **INT_AND**) is generated, the FIFO is turned on automatically. It stores the data at the selected ODR.
3. When the FIFO is full, all the new incoming data is discharged. Reading only a subset of the data already stored into the FIFO keeps locked the possibility for new data to be written.
4. The FIFO restarts saving data only if all the data are read.
5. If communication speed is high, data loss can be prevented.
6. To prevent a FIFO full condition, the required condition is to complete the reading of the data set before the next **DATA_READY** occurs.
7. If this condition is not guaranteed, data can be lost.

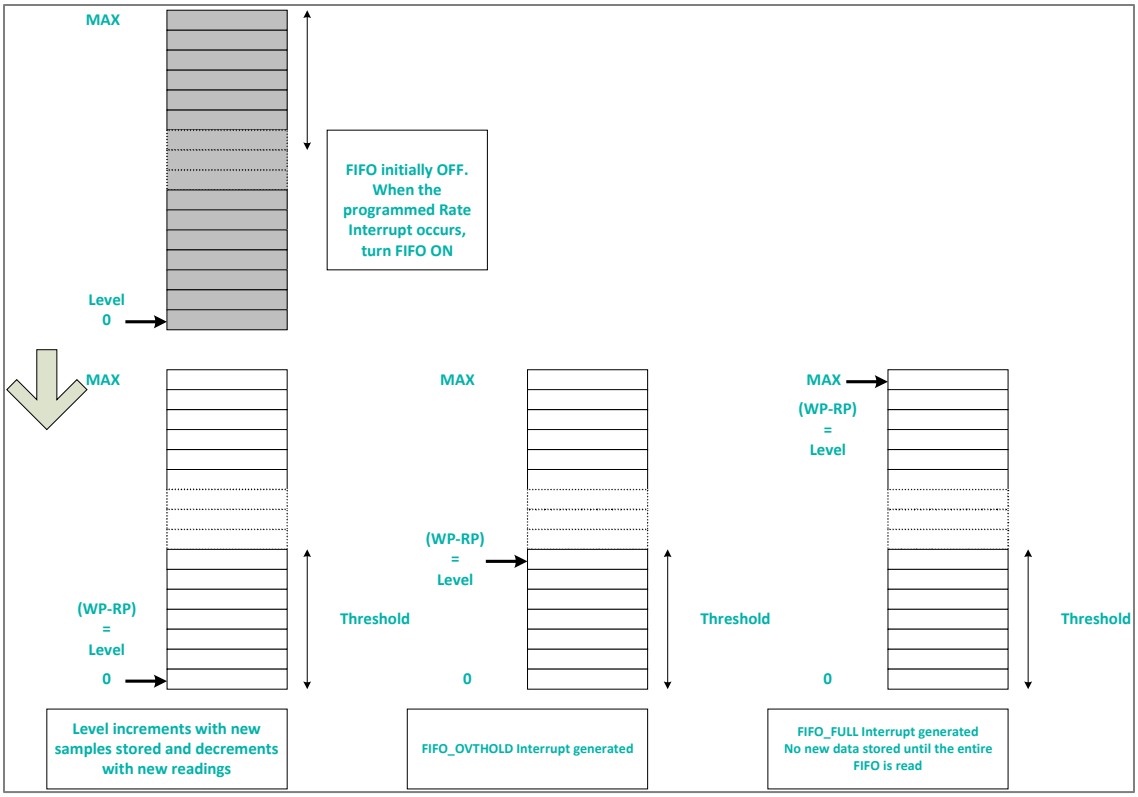

Figure 13: FIFO Interrupt Mode, Stop on Full

9.1.3.2 Overwrite

1. The FIFO is initially disabled. Data is stored only in the data registers.
2. When a rate interrupt (either **INT_OR** or **INT_AND**) is generated, the FIFO is turned on automatically. It stores the data at the selected ODR.
3. When the FIFO is full, an interrupt can be generated.
4. When the FIFO is full, the oldest data is overwritten with the new ones.
5. If communication speed is high, data integrity can be preserved.
6. In order to prevent a data lost condition, the required condition is to complete the reading of the data set before the next **DATA_READY** occurs.
7. If this condition is not guaranteed, data can be overwritten.
8. When an overrun condition occurs the reading pointer is forced to the writing pointer -1 to ensure only older data are discarded and newer data have a chance to be read.

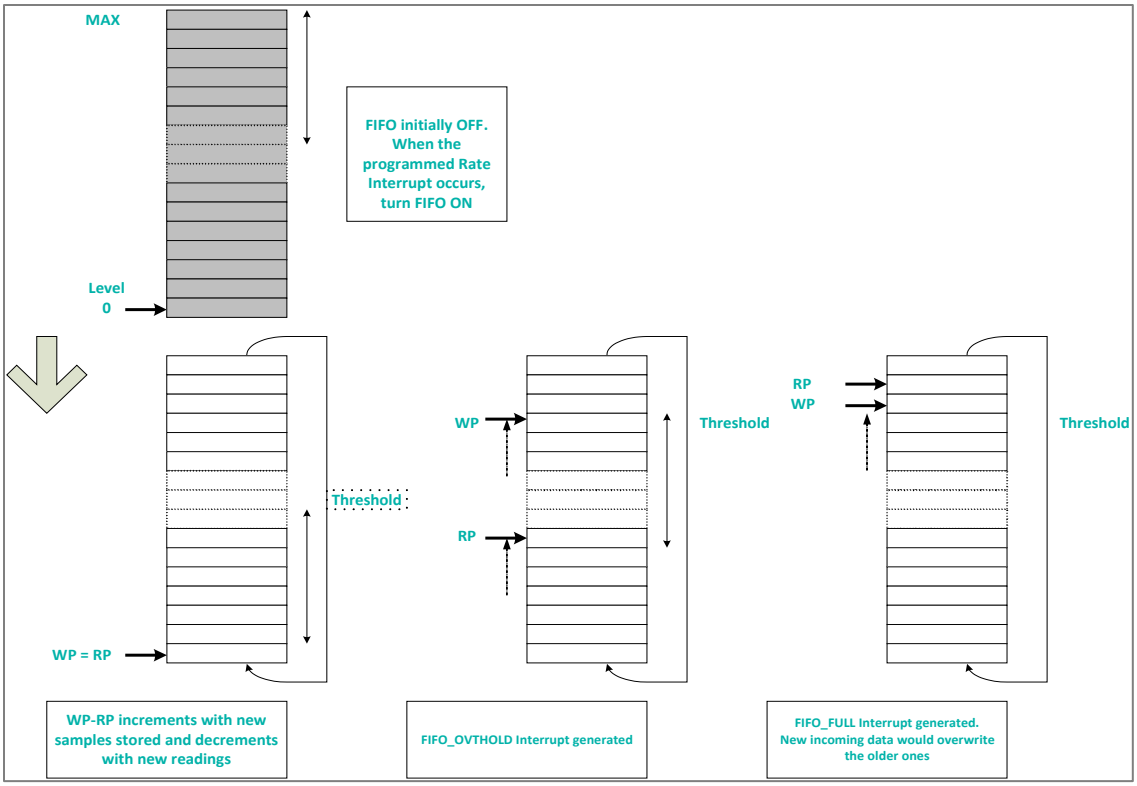

Figure 14: FIFO Interrupt Mode, Overwrite

9.1.4 Snapshot Mode

1. The FIFO is initially in normal mode with overrun enabled.
2. When a rate interrupt (either **INT_OR** or **INT_AND**) is generated, the FIFO switches automatically to not-overrun mode. It stores the data at the selected ODR until the FIFO becomes full.
3. When FIFO is full, an interrupt can be generated.
4. When FIFO is full, all the new incoming data will be discharged. Reading only a subset of the data already stored into the FIFO keeps locked the possibility for new data to be written.
5. The FIFO restarts saving data only if all the data are read.
6. If communication speed is high, data loss can be prevented.
7. To prevent a FIFO full condition, the required condition is to complete the reading of the data set before the next **DATA_READY** occurs.
8. If this condition is not guaranteed, data can be lost.

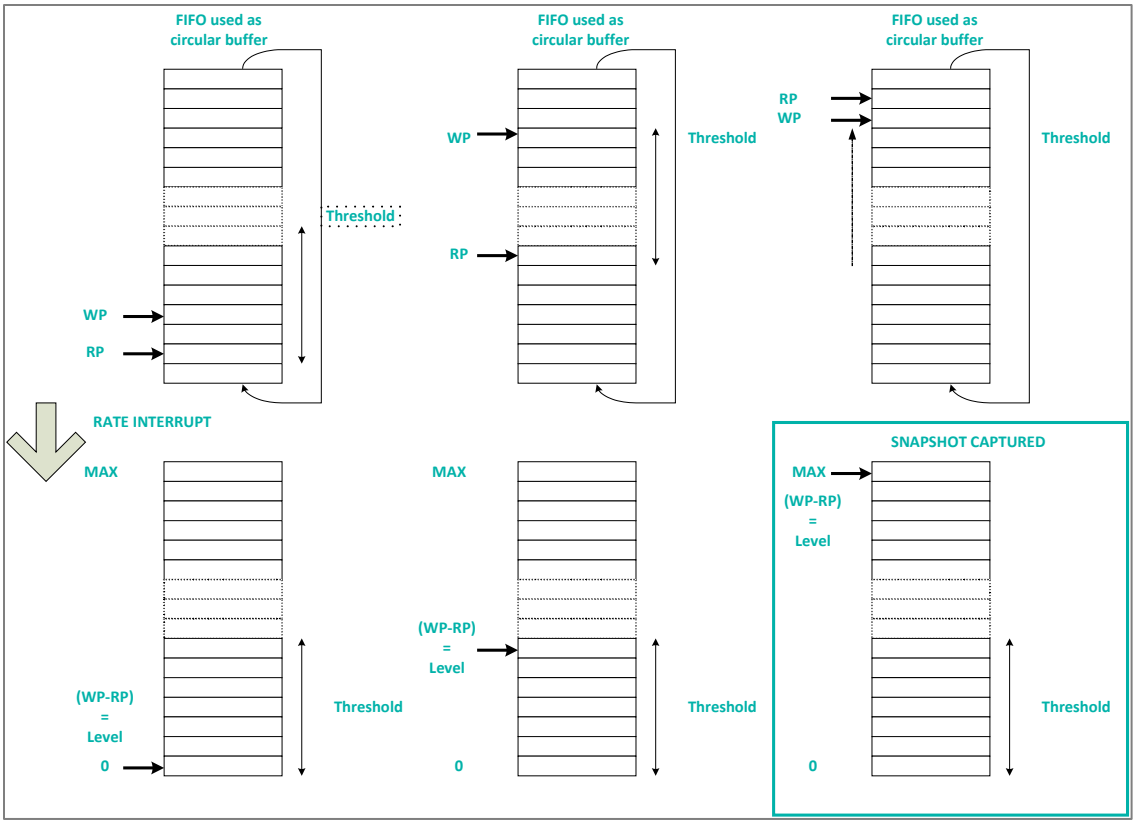

Figure 15: FIFO Snapshot Mode

9.2 Example of FIFO Read/Write Pointers Evolution

Figure 6 assumes:

- A reading frequency roughly twice the writing frequency (ODR)
- A FIFO threshold = 126
- FIFO is in normal mode

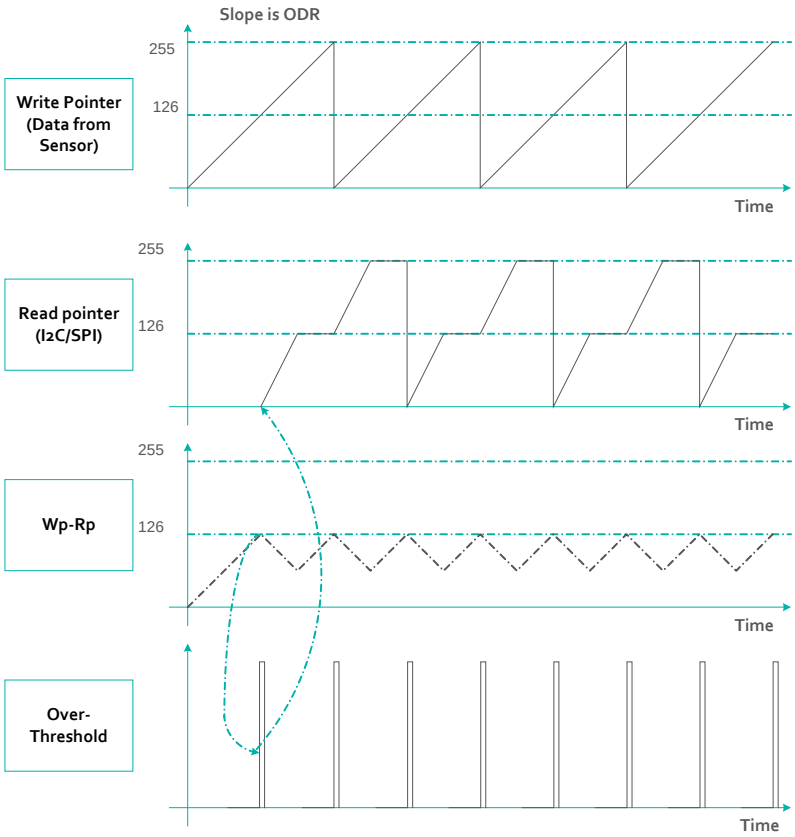

Figure 16: FIFO Read/Write Pointer Evolution

9.3 Configuration Registers

Register/Field	Description
fifo_store_gyr, *fifo_store_acc,* *fifo_store_mag,* *fifo_store_quat*	Enable the data storage of gyroscope/accelerometer/magnetometer/quaternion data in FIFO
FIFO_THS	FIFO threshold
FIFO_CFG	FIFO configuration
sts_ul_fifo_empty, *sts_ul_fifo_overrun,* *sts_ul_fifo_ths,* *sts_i1_fifo_empty,* *sts_i1_fifo_overrun,* *sts_i1_fifo_ths*	FIFO status interrupt
msk_i1_fifo_empty, *msk_i1_fifo_overrun,* *msk_i1_fifo_ths*	FIFO interrupt enables
FIFO_COUNT	Number of FIFO words available on FIFO
FIFO_STATUS	FIFO status register
FIFO_DATA	FIFO data register

10 Programming Example

This section includes examples for executing various operations on the device.
Three functions are defined to abstract the SPI/I^2C communication:

- **Write (<Register Address>, <Value>):**
  Writes the <Value> to the register at the <Register Address>.
- **Read (<Register Address>):**
  Returns the value of the register at the *<Register Address>*.
- **ReadBurstNoInc (<Register Address>, <count>):**
  Returns an array of *count* elements with the result of a burst read with no autoincrement on <Register Address> register.
- **ReadBurstInc (<Start Address>, <count>):**
  Returns an array of *count* elements with the result of a burst read with autoincrement, starting from <Start Address> register address.

10.1 Simple Read-Out Sequence, No FIFO, No Interrupts

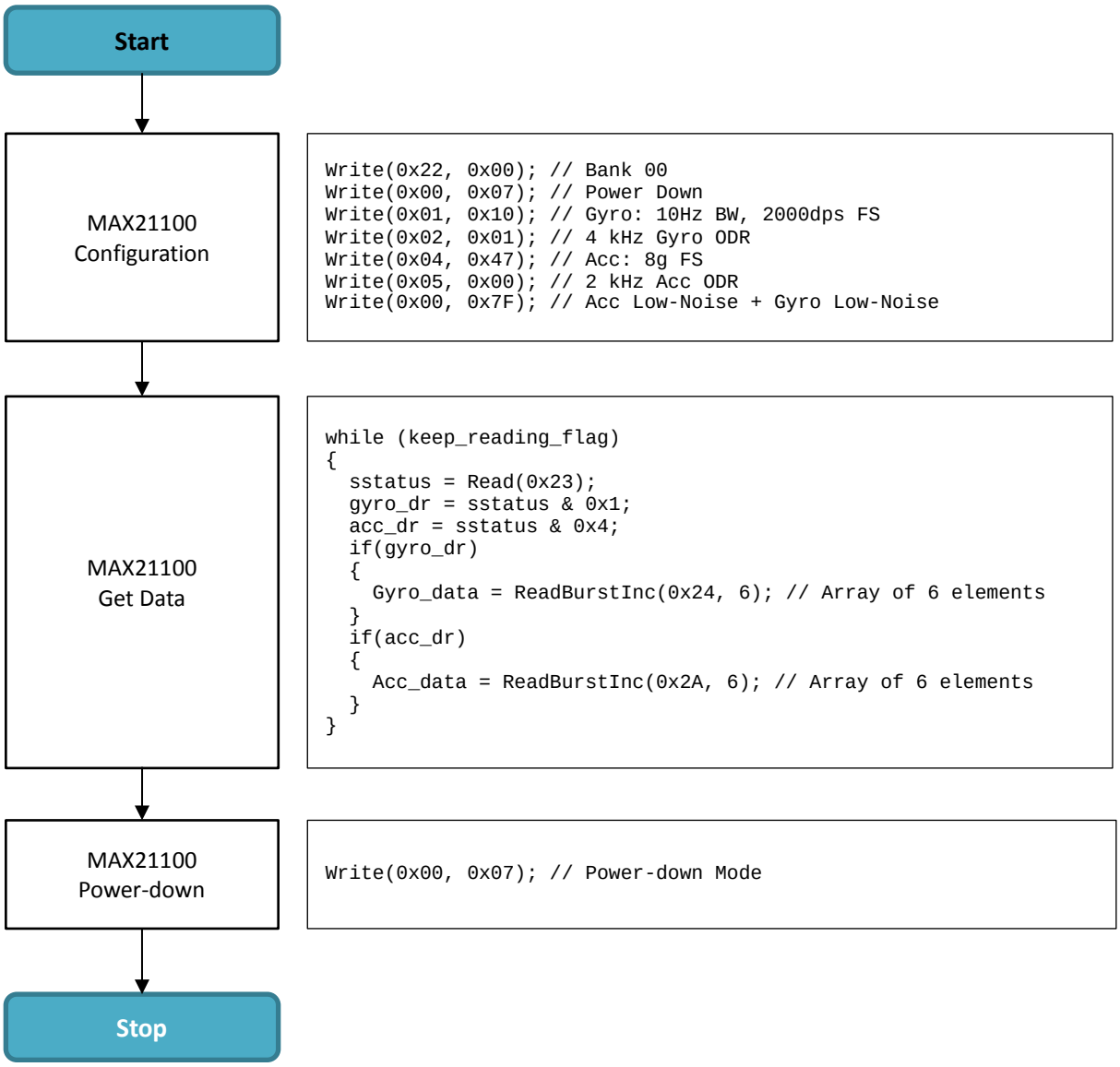

Figure 17: Simple Read-Out Sequence, No FIFO, No Interrupts

10.2 Simple Read-Out Sequence, FIFO Normal Mode, No Interrupts

Figure 18: Simple Read-Out Sequence, FIFO Normal, No Interrupts

10.3 Simple Readout Sequence, Normal Mode, Data Ready Interrupts

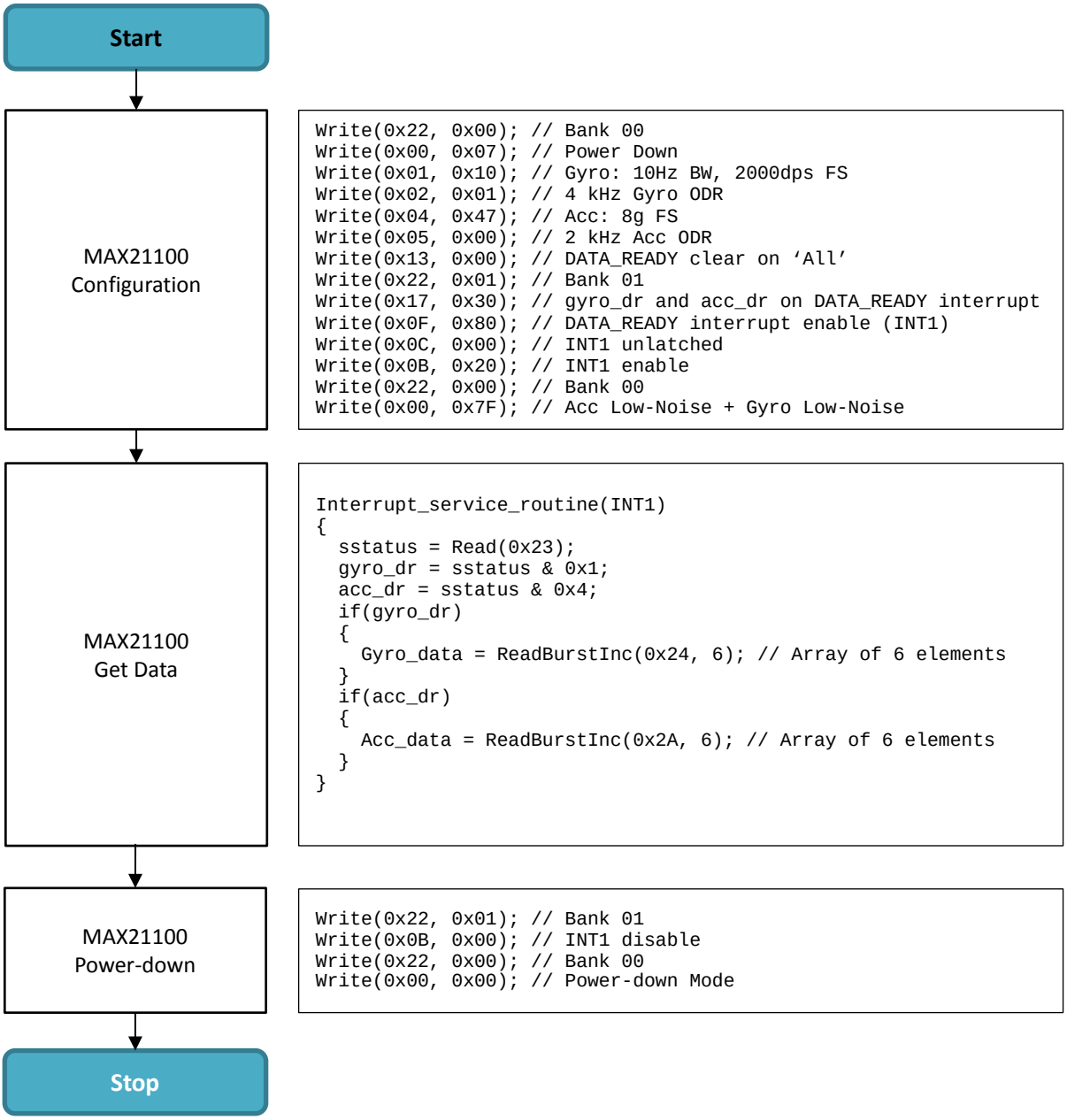

Figure 19: Simple Read-Out Sequence, Normal Mode, with Data Ready Interrupts and No FIFO

11 Motion Merging Engine

The MAX21100 IMU includes a Motion Merging Engine (MME) that can be used to fuse together the embedded 3-axis gyroscope and the 3-axis accelerometer, as well as for an external magnetometer that can be optionally connected to the embedded I^2C master. Using the MME reduces the host µC processing overhead and increases the accuracy in the angle estimation, thanks to the internal calibrated and low-jitter timing reference.

11.1 Sensor Fusion Algorithm

The Sensor Fusion algorithm running in the MME is a Maxim proprietary algorithm based on the well-known complementary filter, evolved to make it as good, in most practical cases, as a Kalman Filter, typically being 1 order of magnitude more complex. Sometimes, the Maxim Adaptive Complementary filter has been proved to exceed the performances of a Kalman filter.

11.2 Rationale for the Motion Merging Engine

The underlying assumption for the Motion Merging Engine (MME) is that the sensor calibration (including bias compensation value calculation) typically occurs once every tenth of a second. At the opposite the evaluation of the angular information (quaternion) has be performed continuously.

As such, it makes sense that the recurring operation (fusing together accelerometer, gyroscope and magnetometer) is performed in an application specific hardware block to save processing load to the system Host, whereas the actual Bias estimation can still performed in SW once in a while, with negligible impact on the power consumption of the whole system.

This appears to be of particular relevance for embedded applications, in which the system host processing capabilities can be limited and every saved µA counts.

11.3 Motion Merging Engine Architecture

The functional connections between the Motion Merging Engine and the other blocks in the architecture can be shown as follows:

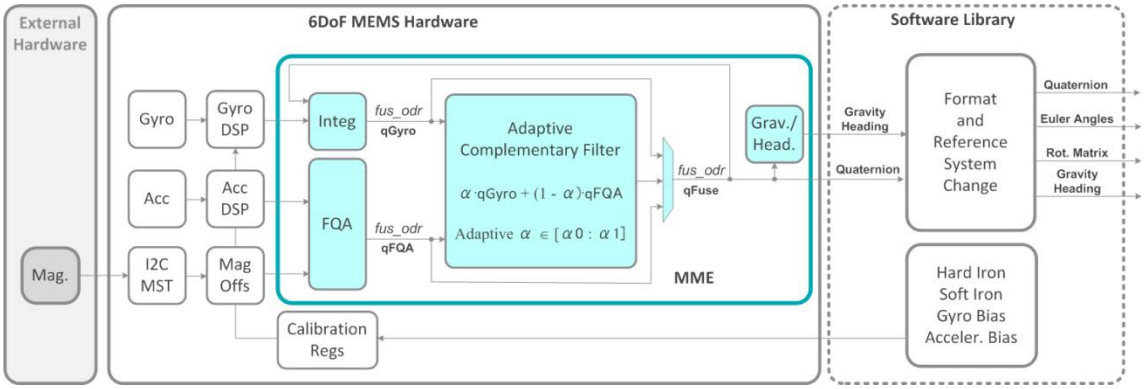

Figure 20: Motion Merging Engine Architecture

The system host is responsible for:

1. Detecting the conditions when a new calibration is needed and subsequently triggering the execution of bias calibrations algorithms.
2. Calculating the bias compensation values of the gyroscope and the accelerometer using various calibrations algorithms (outside the IMUs).
3. Writing back the bias compensation values into the Maxim IMUs.
4. Implementing a SW interpolation of the bias correction to prevent the occurrence of a step correction when the bias correction change can determine an observable discontinuity on the output bias-compensated raw data signal.
5. Applying the appropriate bias compensation configuration to select the desired behavior (e.g., whether the bias correction should be applied only data going to the MME or to the raw data as well).
6. Configuring the sensors and turning them ON with the required power mode.
7. Configuring the MME operating mode (G, G+A, A+M, G+A+M).
8. Configuring the MME parameters (ODR, alphas, etc.).
9. Enable the MME.

The Maxim IMUs would be accordingly responsible for:

1. Applying the bias compensation values to the data going to the data registers and to the FIFO, if configured to do so.
2. Applying the bias compensation values to the data going to the MME, if configured to do so.
3. Applying the bias compensation values to the data going to the MCS, if configured to do so.
4. Calculating the quaternion output or, in alternative, the equivalent (gravity; heading) pair.

11.4 Benefits

Using the MME is beneficial is terms of:

Table 5: Motion Merging Engine Benefits

Benefit	Description
Low Power	The calculation of the quaternion carrying the information about the angular orientation is performed on a small and specifically designed power optimized DSP. The power advantage of the embedded DSP becomes even more evident when the gyroscope output data rate increases, as necessary to increase the quaternion calculation accuracy.
Higher Timing Accuracy	The transformation of the rate (rotational speed) into an angle requires integration over time. For this operation to be done in SW without loss of accuracy, a number of conditions have to be met, including the accurate measurement of the ODR made in SW and the need to not miss any data.
Higher Integration Accuracy	It can be proved that to achieve higher accuracy in the quaternion calculation, the gyroscope ODR has to be preferably increased up to 1kHz for optimum performances. At such an ODR, the power consumption necessary to perform the calculation increases dramatically (tenths on mA), whilst in the MME it keeps in the tenths on µA range.
Embedded Low Power PDR Precursor	Having the capability to calculate the angular information inside the sensor means that the application processor and the sensor hub can be turned off completely for an extended amount of time, as the IMUs would keep calculating the quaternion and would save it in the FIFO whenever appropriate (i.e., in correspondence of a new step).

11.5 Operating Modes

Depending on a number of factors, the MME can be configured for a given application in a different mode, as summarized in Table 6:

Table 6: Motion Merging Engine Operating Modes

Mode	Description	Benefits	Downsides
Gyro Only	The MME is used to integrate the gyro signal only	Inherent immunity to magnetic disturbances and linear accelerations.	Can require additional processing in SW to compensate the gyro errors as well as integration errors over x, y, and z.
Gyro + Accelerometer	The MME uses the accelerometer to compensate for the gyro bias over x and y. This corresponds to the Rotation Vector sensor in Android.	Inherent immunity to magnetic disturbances.	Can require additional processing in SW to compensate the gyro errors as well as integration errors z. Some sensitivity to linear accelerations left.
Gyro + Accelerometer + Magnetometer	The MME uses the accelerometer to compensate for the gyro bias over x and y. The MME uses the magnetometer to compensate for the gyro bias over z. This corresponds to the Geomagnetic Rotation Vector sensor in Android.	Highest accuracy	Does not account for soft iron effects. Can be subject to magnetic disturbances, if present.
Accelerometer + Magnetometer	The MME generates at ultra-low power levels the quaternions using only the accelerometer and the magnetometer (so-called Virtual Gyroscope)	Lowest power	Noisy, lag noticeable, artifacts may be present. Does not account for soft iron effects. May be subject to magnetic disturbances, if present. Suitable only for movements with limited bandwidth in an environment without magnetic disturbances.

11.6 Configuration Registers

Register/Field	Description
FUS_CFG0, FUS_CFG1	MME configuration registers

12 Register Map

The following sections illustrate the register map of the device. When not specified differently, register/field values are expressed in binary format.

12.1 Register Overview

12.1.1 Common

Name	Address	Access	Default	Description
WHO_AM_I	0x20	R	10110010	Device ID
REVISION_ID	0x21	R	00000000	Revision ID register
BANK_SELECT	0x22	R/W	00000000	Register bank selection
SYSTEM_STATUS	0x23	R	00000000	System Status register
GYRO_X_H	0x24	R	00000000	Bits [15:8] of the gyroscope X measurement
GYRO_X_L	0x25	R	00000000	Bits [7:0] of the gyroscope X measurement
GYRO_Y_H	0x26	R	00000000	Bits [15:8] of the gyroscope Y measurement
GYRO_Y_L	0x27	R	00000000	Bits [7:0] of the gyroscope Y measurement
GYRO_Z_H	0x28	R	00000000	Bits [15:8] of the gyroscope Z measurement
GYRO_Z_L	0x29	R	00000000	Bits [7:0] of the gyroscope Z measurement
ACC_X_H	0x2A	R	00000000	Bits [15:8] of the accelerometer X measurement
ACC_X_L	0x2B	R	00000000	Bits [7:0] of the accelerometer X measurement
ACC_Y_H	0x2C	R	00000000	Bits [15:8] of the accelerometer Y measurement
ACC_Y_L	0x2D	R	00000000	Bits [7:0] of the accelerometer Y measurement
ACC_Z_H	0x2E	R	00000000	Bits [15:8] of the accelerometer Z measurement
ACC_Z_L	0x2F	R	00000000	Bits [7:0] of the accelerometer Z measurement
MAG_X_H	0x30	R	00000000	Bits [15:8] of the magnetometer X measurement
MAG_X_L	0x31	R	00000000	Bits [7:0] of the magnetometer X measurement
MAG_Y_H	0x32	R	00000000	Bits [15:8] of the magnetometer Y measurement

MAG_Y_L	0x33	R	00000000	Bits [7:0] of the magnetometer Y measurement
MAG_Z_H	0x34	R	00000000	Bits [15:8] of the magnetometer Z measurement
MAG_Z_L	0x35	R	00000000	Bits [7:0] of the magnetometer Z measurement
TEMP_H	0x36	R	00000000	Bits [15:8] of the temperature measurement
TEMP_L	0x37	R	00000000	Bits [7:0] of the temperature measurement
FIFO_COUNT	0x3C	R	00000000	Available number of FIFO samples for data set
FIFO_STATUS	0x3D	R	00000000	FIFO status flags
FIFO_DATA	0x3E	R/W	00000000	FIFO data, to be read in burst mode
RST_REG	0x3F	R/W	00000000	Reset register

12.1.2Bank 00

Name	Address	Access	Default	Description
POWER_CFG	0x00	R/W	00000111	Power mode configuration
GYRO_CFG1	0x01	R/W	00101000	Gyroscope configuration register 1
GYRO_CFG2	0x02	R/W	00000100	Gyroscope configuration register 2
GYRO_CFG3	0x03	R/W	00000000	Gyroscope configuration register 3
PWR_ACC_CFG	0x04	R/W	11000111	Accelerometer power configuration.
ACC_CFG_1	0x05	R/W	00000010	Accelerometer configuration register 1
ACC_CFG_2	0x06	R/W	00000000	Accelerometer configuration register 2
MAG_SLV_CFG	0x07	R/W	00000110	Magnetometer slave configuration
MAG_SLV_ADD	0x08	R/W	00000000	Magnetometer slave address
MAG_SLV_REG	0x09	R/W	00000000	Magnetometer slave register
MAG_MAP_REG	0x0A	R/W	00000000	Magnetometer mapping register
I2C_MST_ADD	0x0B	R/W	00000000	I^2C master register address
I2C_MST_RW	0x0C	R/W	00000000	I^2C master register data.
MAG_OFS_X_MSB	0x0D	R/W	00000000	Magnetometer offset X, MSB
MAG_OFS_X_LSB	0x0E	R/W	00000000	Magnetometer offset X, LSB
MAG_OFS_Y_MSB	0x0F	R/W	00000000	Magnetometer offset Y, MSB
MAG_OFS_Y_LSB	0x10	R/W	00000000	Magnetometer offset Y, LSB
MAG_OFS_Z_MSB	0x11	R/W	00000000	Magnetometer offset Z, MSB
MAG_OFS_Z_LSB	0x12	R/W	00000000	Magnetometer offset Z, LSB
DR_CFG	0x13	R/W	00000001	Data ready configuration
IO_CFG	0x14	R/W	00000000	Input/output configuration
I2C_PAD	0x15	R/W	00000100	PADs configuration
I2C_CFG	0x16	R/W	00000000	Serial interfaces configuration
FIFO_THS	0x17	R/W	00000000	FIFO threshold configuration
FIFO_CFG	0x18	R/W	00000000	FIFO mode configuration
DSYNC_CFG	0x1A	R/W	00000000	DSYNC configuration
DSYNC_CNT	0x1B	R/W	00000000	DSYNC counter
ITF_OTP	0x1C	R/W	00000000	Serial interfaces and OTP control

12.1.3Bank 01

Name	Address	Access	Default	Description
INT_REF_X	0x00	R/W	00000000	Interrupt reference for X axis
INT_REF_Y	0x01	R/W	00000000	Interrupt reference for Y axis
INT_REF_Z	0x02	R/W	00000000	Interrupt reference for Z axis
INT_DEB_X	0x03	R/W	00000000	Interrupt debounce, X axis
INT_DEB_Y	0x04	R/W	00000000	Interrupt debounce, Y axis
INT_DEB_Z	0x05	R/W	00000000	Interrupt debounce, Z axis
INT_MSK_X	0x06	R/W	00000000	Interrupt mask, X axis zones
INT_MSK_Y	0x07	R/W	00000000	Interrupt mask, Y axis zones
INT_MSK_Z	0x08	R/W	00000000	Interrupt mask, Z axis zones
INT_MASK_AO	0x09	R/W	00000000	Interrupt masks, and/or
INT_CFG_1	0x0A	R/W	00000000	Interrupt Configuration register 1
INT_CFG_2	0x0B	R/W	00100100	Interrupt Configuration register 2
INT_TMO	0x0C	R/W	00000000	Interrupt timeout
INT_STS_UL	0x0D	R	00000000	Interrupt sources, unlatched
INT_STS	0x0E	R	00000000	Interrupt Status register
INT_MSK	0x0F	R/W	10000010	Interrupt Mask register
INT_SRC_SEL	0x17	R/W	00111100	Interrupt source selection
SERIAL_5	0x1A	R	00000000	Unique serial number, byte 5
SERIAL_4	0x1B	R	00000000	Unique serial number, byte 4
SERIAL_3	0x1C	R	00000000	Unique serial number, byte 3
SERIAL_2	0x1D	R	00000000	Unique serial number, byte 2
SERIAL_1	0x1E	R	00000000	Unique serial number, byte 1
SERIAL_0	0x1F	R	00000000	Unique serial number, byte 0

12.1.4Bank 02

Name	Address	Access	Default	Description
QUAT0_H	0x00	R	00000000	Bits [15:8] of the quaternion 0 measurement.
QUAT0_L	0x01	R	00000000	Bits [7:0] of the quaternion 0 measurement.
QUAT1_H	0x02	R	00000000	Bits [15:8] of the quaternion 1 measurement.
QUAT1_L	0x03	R	00000000	Bits [7:0] of the quaternion 1 measurement.
QUAT2_H	0x04	R	00000000	Bits [15:8] of the quaternion 2 measurement.
QUAT2_L	0x05	R	00000000	Bits [7:0] of the quaternion 2 measurement.
QUAT3_H	0x06	R	00000000	Bits [15:8] of the quaternion 3 measurement.
QUAT3_L	0x07	R	00000000	Bits [7:0] of the quaternion 3 measurement.
BIAS_GYRO_X_H	0x13	R/W	00000000	Gyroscope bias compensation, X axis, MSB
BIAS_GYRO_X_L	0x14	R/W	00000000	Gyroscope bias compensation, X axis, LSB
BIAS_GYRO_Y_H	0x15	R/W	00000000	Gyroscope bias compensation, Y axis, MSB
BIAS_GYRO_Y_L	0x16	R/W	00000000	Gyroscope bias compensation, Y axis, LSB
BIAS_GYRO_Z_H	0x17	R/W	00000000	Gyroscope bias compensation, Z axis, MSB
BIAS_GYRO_Z_L	0x18	R/W	00000000	Gyroscope bias compensation, Z axis, LSB
BIAS_ACC_X	0x19	R/W	00000000	Accelerometer bias compensation, X axis
BIAS_ACC_Y	0x1A	R/W	00000000	Accelerometer bias compensation, Y axis
BIAS_ACC_Z	0x1B	R/W	00000000	Accelerometer bias compensation, Z axis
FUS_CFG0	0x1C	R/W	00000000	MME configuration register 1
FUS_CFG1	0x1D	R/W	01011000	MME configuration register 2
GYR_ODR_TRIM	0x1F	R	01110000	Gyroscope ODR correction factor register

12.2 Registers Description

12.2.1 Common

WHO_AM_I

Address	Common - 0x20 (Hex) - 32 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	10110010							

Description

Device ID.
Identifier of the product family.

Back to Register Overview

REVISION_ID

Address	Common - 0x21 (Hex) - 33 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Revision ID register.
Identifier of the silicon revision.

Back to Register Overview

BANK_SELECT

Address	Common - 0x22 (Hex) - 34 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	***trim_bank_en***	RFU	RFU		***bank_sel***			
Access	R	R	R		R/W			
Default	0	—	—		0000			

Description

Register bank selection.

Fields

trim_bank_en: Enables test banks read/write operations.

0: Disabled
1: Enabled

bank_sel: Allows addressing 3 pages of registers, other than the common bank. Pages are 32 bytes sized.

0000: Bank 0
0001: Bank 1
0010: Bank 2

Back to Register Overview

SYSTEM_STATUS

Address	Common - 0x23 (Hex) - 35 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	*quat_err*	*quat_dr*	*magn_err*	*magn_dr*	*acc_err*	*acc_dr*	*gyro_err*	*gyro_dr*
Access	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0

Description

System Status Register.

Fields

quat_err: Quaternion output error: It goes high when a new quaternion data is generated before or during data reading.

quat_dr: Quaternion output available: It goes high when a new set of quaternion data is available.

magn_err: Magnetometer output error: It goes high when a new accelerometer data is generated before or during data reading.

magn_dr: Magnetometer output available: It goes high when a new set of accelerometer data is available.

acc_err: Accelerometer output error: It goes high when a new accelerometer data is generated before or during data reading.

acc_dr: Accelerometer output available: It goes high when a new set of accelerometer data is available.

gyro_err: Gyroscope output error: It goes high when a new gyroscope data is generated before or during data reading.

gyro_dr: Gyroscope output available: It goes high when a new set of gyroscope data is available.

Back to Register Overview

GYRO_X_H

Address	Common - 0x24 (Hex) - 36 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [15:8] of the gyroscope X measurement.
Most significant byte of the X axis gyroscope data.

Back to Register Overview

GYRO_X_L

Address	Common - 0x25 (Hex) - 37 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [7:0] of the gyroscope X measurement.
Least significant byte of the X axis gyroscope data.

Back to Register Overview

GYRO_Y_H

Address	Common - 0x26 (Hex) - 38 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [15:8] of the gyroscope Y measurement.
Most significant byte of the Y axis gyroscope data.

Back to Register Overview

GYRO_Y_L

Address	Common - 0x27 (Hex) - 39 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [7:0] of the gyroscope Y measurement.
Least significant byte of the Y axis gyroscope data.

Back to Register Overview

GYRO_Z_H

Address	Common - 0x28 (Hex) - 40 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [15:8] of the gyroscope Z measurement.
Most significant byte of the Z axis gyroscope data.

Back to Register Overview

GYRO_Z_L

Address	Common - 0x29 (Hex) - 41 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [7:0] of the gyroscope Z measurement.
Least significant byte of the Z axis gyroscope data.

Back to Register Overview

ACC_X_H

Address	Common - 0x2A (Hex) - 42 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [15:8] of the accelerometer X measurement.
Most significant byte of the X axis accelerometer data.

Back to Register Overview

ACC_X_L

Address	Common - 0x2B (Hex) - 43 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [7:0] of the accelerometer X measurement.
Least significant byte of the X axis accelerometer data.

Back to Register Overview

ACC_Y_H

Address	Common - 0x2C (Hex) - 44 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [15:8] of the accelerometer Y measurement.
Most significant byte of the Y axis accelerometer data.

Back to Register Overview

ACC_Y_L

Address	Common - 0x2D (Hex) - 45 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [7:0] of the accelerometer Y measurement.
Least significant byte of the Y axis accelerometer data.

Back to Register Overview

ACC_Z_H

Address	Common - 0x2E (Hex) - 46 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [15:8] of the accelerometer Z measurement.
Most significant byte of the Z axis accelerometer data.

Back to Register Overview

ACC_Z_L

Address	Common - 0x2F (Hex) - 47 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [7:0] of the accelerometer Z measurement.
Least significant byte of the Z axis accelerometer data.

Back to Register Overview

MAG_X_H

Address	Common - 0x30 (Hex) - 48 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [15:8] of the magnetometer X measurement.
Most significant byte of the X axis magnetometer data.

Back to Register Overview

MAG_X_L

Address	Common - 0x31 (Hex) - 49 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [7:0] of the magnetometer X measurement.
Least significant byte of the X axis magnetometer data.

Back to Register Overview

MAG_Y_H

Address	Common - 0x32 (Hex) - 50 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [15:8] of the magnetometer Y measurement.
Most significant byte of the Y axis magnetometer data.

Back to Register Overview

MAG_Y_L

Address	Common - 0x33 (Hex) - 51 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [7:0] of the magnetometer Y measurement.
Least significant byte of the Y axis magnetometer data.

Back to Register Overview

MAG_Z_H

Address	Common - 0x34 (Hex) - 52 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [15:8] of the magnetometer Z measurement.
Most significant byte of the Z axis magnetometer data.

Back to Register Overview

MAG_Z_L

Address	Common - 0x35 (Hex) - 53 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [7:0] of the magnetometer Z measurement.
Least significant byte of the Z axis magnetometer data.

Back to Register Overview

TEMP_H

Address	Common - 0x36 (Hex) - 54 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [15:8] of the temperature measurement.
Most significant byte of the temperature data.

Back to Register Overview

TEMP_L

Address	Common - 0x37 (Hex) - 55 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [7:0] of the temperature measurement.
Least significant byte of the temperature data.

Back to Register Overview

FIFO_COUNT

Address	Common - 0x3C (Hex) - 60 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Available number of FIFO samples for data set.
Contains the number of FIFO words available on the FIFO.

Back to Register Overview

FIFO_STATUS

Address	Common - 0x3D (Hex) - 61 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU			*fifo_data_lost*	*fifo_read_empty*	*fifo_ovthold*	*fifo_full*	*fifo_empty*
Access	R			R	R	R	R	R
Default	—			0	0	0	0	0

Description

FIFO Status Flags

Fields

fifo_data_lost: Indicates if at least one data has been lost when the FIFO was full.

fifo_read_empty: Indicates if a read occurred when the FIFO was empty.

fifo_ovthold: Indicates if the number of data in FIFO exceeds the threshold.

fifo_full: Indicates if the FIFO is full.

fifo_empty: Indicates if the FIFO is empty.

Back to Register Overview

FIFO_DATA

Address	Common - 0x3E (Hex) - 62 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R/W							
Default	00000000							

Description

FIFO data to be read in burst mode.
FIFO data register. Burst reading autoincrement the FIFO read pointer.

Back to Register Overview

RST_REG

Address	Common - 0x3F (Hex) - 63 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU					***parity_rst***	***hpf_acc_rst***	***hpf_gyro_rst***
Access	R					R/W	R/W	R/W
Default	—					0	0	0

Description

Reset register

Fields

parity_rst: Triggers a reset of the parity.

0: Idle
1: Trigger

hpf_acc_rst: Sets the accelerometer HP filter and restore the output to baseline.

0: Idle
1: Trigger

hpf_gyro_rst: Sets the gyroscope HP filter and restore the output to baseline.

0: Idle
1: Trigger

Back to Register Overview

12.2.2Bank 00

POWER_CFG

Address	Bank 00 - 0x00 (Hex) - 0 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	*pwr_aux*	*pwr_mode*				*sns_en_z*	*sns_en_y*	*sns_en_x*
Access	R/W	R/W				R/W	R/W	R/W
Default	0	0000				1	1	1

Description

Power mode configuration

Fields

pwr_aux: When 1, pwr_mode becomes a transition between power modes controllable by AUX pin.

0: Normal
1: DSYNC

pwr_mode: Sets the power mode.

POWER_CFG[pwr_aux]	Value	Mnemonic
0	0000	Power-down
0	0001	Gyro sleep
0	0010	Gyro low power
0	0011	Gyro low noise
0	1000	Acc low power
0	1100	Acc low noise
0	1101	Acc low noise + gyro sleep
0	1110	Acc low noise + gyro low power
0	1111	Acc low noise + gyro low noise
0	01XX	Don't care
0	1001	Don't care
0	101X	Don't care
1	0000	Power-down/acc low power
1	0001	Power-down/acc low noise
1	0010	Power-down/gyro low power
1	0011	Power-down/gyro low noise
1	0100	Power-down/acc low noise + gyro low noise

1	0101	Power-down/acc low noise + gyro low ower
1	0110	Gyro sleep/gyro low noise
1	0111	Gyro sleep/acc low noise + gyro low noise
1	1000	Gyro low power/acc low noise + gyro low power

sns_en_z: Enables gyroscope Z axis.

0: Disable
1: Enable

sns_en_y: Enables gyroscope Y axis.

0: Disable
1: Enable

sns_en_x: Enables gyroscope X axis.

0: Disable
1: Enable

Back to Register Overview

GYRO_CFG1

Address	Bank 00 - 0x01 (Hex) - 1 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	*self_test*		*sns_lpf_bnd*				*sns_dout_fsc*	
Access	R/W		R/W				R/W	
Default	00		1010				00	

Description

Gyroscope configuration register 1

Fields

self_test: Sets the gyroscope self-test mode.

00: Disabled
01: Positive sign
10: Negative sign
11: Negative sign

sns_lpf_bnd: Sets the gyroscope lowpass filter cutoff frequency.

GYRO_CFG2[sns_gyr_ois_lpf]	Value	Mnemonic
0	0000	2Hz
0	0001	4Hz
0	0010	6Hz
0	0011	8Hz
0	0100	10Hz
0	0101	14Hz
0	0110	22Hz
0	0111	32Hz
0	1000	50Hz
0	1001	75Hz
0	1010	100Hz
0	1011	150Hz
0	1100	200Hz
0	1101	250Hz
0	1110	300Hz
0	1111	400Hz
1	0XXX	1kHz
1	1XXX	2kHz

sns_dout_fsc: Sets the gyroscope full scale.

00: 2000 dps
01: 1000 dps
10: 500 dps
11: 250 dps

Back to Register Overview

GYRO_CFG2

Address	Bank 00 - 0x02 (Hex) - 2 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU		***sns_gyr_ois_lpf***	***sns_gyr_hpf_en***	***sns_odr***			
Access	R		R/W	R/W	R/W			
Default	—		0	0	0100			

Description

Gyroscope configuration register 2

Fields

sns_gyr_ois_lpf: Enables the gyroscope low-pass filter for OIS (low-phase delay).

0: Off
1: On (full scales are halved)

sns_gyr_hpf_en: Enables the gyroscope highpass filter (this option is not available in low-power mode).

0: Disabled
1: Enabled

sns_odr: Sets the gyroscope output data rate (ODR).

If ***pwr_mode*** is gyro low power or acc low noise + gyro low power:

Value	Mnemonic
0000	250Hz
0001	250Hz
0010	250Hz
0011	250Hz
0100	250Hz
0101	250Hz
0110	125Hz
0111	62.5Hz
1XXX	31.25Hz

Otherwise:

Value	Mnemonic
0000	8kHz
0001	4kHz
0010	2kHz
0011	1kHz
0100	500Hz
0101	250Hz
0110	125Hz
0111	62.5Hz
1000	31.25Hz
1001	15.625Hz
1010	7.8125Hz
1011	3.90625Hz
11XX	3.90625Hz

Back to Register Overview

GYRO_CFG3

Address	Bank 00 - 0x03 (Hex) - 3 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU			*wait_data_mode*		*sns_hpf_co*		
Access	R			R/W		R/W		
Default	—			00		000		

Description

Gyroscope configuration register 3

Fields

wait_data_mode: To start the data generation, two triggers are defined (Trig1/Trig2):
Trig1: from driveON to senseON
Trig2: from senseON to NormalMode

00: None/None
01: Freq lock/freq lock
10: Freq lock/amp lock
11: Amp lock/amp lock

sns_hpf_co: Sets the gyroscope highpass filter cutoff frequency.

000: 0.08Hz
001: 0.24Hz
010: 0.8Hz
011: 2Hz
100: 5Hz
101: 10Hz
110: 20Hz
111: 50Hz

Back to Register Overview

PWR_ACC_CFG

Address	Bank 00 - 0x04 (Hex) - 4 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	*sns_acc_fsc*		*acc_self_test*			*acc_en_z*	*acc_en_y*	*acc_en_x*
Access	R/W		R/W			R/W	R/W	R/W
Default	11		000			1	1	1

Description

Accelerometer power configuration

Fields

sns_acc_fsc: Sets the accelerometer full scale.

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

acc_self_test: Sets the accelerometer self-test mode.

000: No test
001: Positive force/Y axis self test
010: Positive force/X axis self test
011: Positive force/Z axis self test
101: Negative force/Y axis self test
110: Negative force/X axis self test
111: Negative force/Z axis self test

acc_en_z: Enables accelerometer Z axis.

0: Disabled
1: Enabled

acc_en_y: Enables accelerometer Y axis.

0: Disabled
1: Enabled

acc_en_x: Enables accelerometer X axis.

0: Disabled
1: Enabled

Back to Register Overview

ACC_CFG_1

Address	Bank 00 - 0x05 (Hex) - 5 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	*sns_acc_hpf_co*		*sns_acc_lpf_co*		*sns_acc_odr*			
Access	R/W		R/W		R/W			
Default	00		00		0010			

Description

Accelerometer configuration register 1.

Fields

sns_acc_hpf_co: Set the accelerometer highpass filter cutoff frequency.

00: ODR/400
01: ODR/200
10: ODR/100
11: ODR/50

sns_acc_lpf_co: Set the accelerometer lowpass filter cutoff frequency.

00: ODR/48
01: ODR/22
10: ODR/9
11: ODR/3

sns_acc_odr: Sets the accelerometer output data rate (ODR).

If ***pwr_mode*** is acc low power:

Value	Mnemonic
0000	250Hz
0001	250Hz
0010	250Hz
0011	250Hz
0100	125Hz
0101	62.5Hz
0110	31.25Hz
0111	15.625Hz
1000	7.8125Hz
1001	3.90625Hz
1010	1.953125Hz
1011	0.9765625Hz
11XX	0.9765625Hz

Otherwise:

Value	Mnemonic
0000	2000Hz
0001	1000Hz
0010	500Hz
0011	250Hz
0100	125Hz
0101	62.5Hz
011X	31.25Hz
1XXX	31.25Hz

Back to Register Overview

ACC_CFG_2

Address	Bank 00 - 0x06 (Hex) - 6 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU				***mag_odr***			***sns_acc_hpf_en***
Access	R				R/W			R/W
Default	—				000			0

Description

Accelerometer configuration register 2

Fields

mag_odr: Sets the magnetometer ODR. It is computed as:
sns_acc_odr / 2^[0:7]

sns_acc_hpf_en: Enables the accelerometer highpass filter.

0: Disabled
1: Enabled

Back to Register Overview

MAG_SLV_CFG

Address	Bank 00 - 0x07 (Hex) - 7 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	*mag_en*	*mag_swap*	*mag_safe*	*mag_grp*	*mag_i2c_std_mode*	*mag_i2c_len*		
Access	R/W	R/W	R/W	R/W	R/W	R/W		
Default	0	0	0	0	0	110		

Description

Magnetometer slave configuration

Fields

mag_en: Enables the magnetometer.

0: Disabled
1: Enabled

mag_swap: Selects the endian of the magnetometer data.

0: MSB First
1: MSB Last

mag_safe: Enables the safe mode.
Since the external magnetometer data read operation is performed asynchronously, setting the safe mode guarantees the read data are correct if the external magnetometer data rate is set to half the transfer rate.

0: Off
1: On

mag_grp: Data read configuration.

Value	Mnemonic	Description
0	mag_i2c_len	Reads mag_i2c_len bytes from external magnetometer.
1	mag_i2c_len + 1	Reads mag_i2c_len+1 bytes from external magnetometer and discard the first one.

mag_i2c_std_mode: Sets the I^2C transfer rate.

0: 400kHz (fast mode)
1: 100kHz (standard mode)

mag_i2c_len: Sets the number of bytes to be read from the magnetometer.

Back to Register Overview

MAG_SLV_ADD

Address	Bank 00 - 0x08 (Hex) - 8 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU	***mag_i2c_slv_add***						
Access	R	R/W						
Default	—	0000000						

Description

Magnetometer slave address

Fields

mag_i2c_slv_add: Magnetometer I^2C slave address.

Back to Register Overview

MAG_SLV_REG

Address	Bank 00 - 0x09 (Hex) - 9 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R/W							
Default	00000000							

Description

Magnetometer slave register

Magnetometer I^2C register address.

Back to Register Overview

MAG_MAP_REG

Address	Bank 00 - 0x0A (Hex) - 10 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU		*mag_chmap*			*mag_invsgn_z*	*mag_invsgn_y*	*mag_invsgn_x*
Access	R		R/W			R/W	R/W	R/W
Default	—		000			0	0	0

Description

Magnetometer mapping register

Fields

mag_chmap: These three bits are used to cross the three axes in the interface between mechanic and analog front end.

Multiplexing of input signals (from A, B, C to Xi, Yi, Zi).

000: [Xi , Yi , Zi] <= [A , B , C]
001: [Xi , Yi , Zi] <= [A , C , B]
010: [Xi , Yi , Zi] <= [B , A , C]
011: [Xi , Yi , Zi] <= [C , A , B]
100: [Xi , Yi , Zi] <= [B , C , A]
101: [Xi , Yi , Zi] <= [C , B , A]
110: [Xi , Yi , Zi] <= [A , B , C]
111: [Xi , Yi , Zi] <= [A , B , C]

mag_invsgn_z: Invert the sign of the Z axis data produced by the external magnetometer sensor.

0: No
1: Yes

mag_invsgn_y: Invert the sign of the Y axis data produced by the external magnetometer sensor.

0: No
1: Yes

mag_invsgn_x: Invert the sign of the X axis data produced by the external magnetometer sensor.

0: No
1: Yes

Back to Register Overview

I2C_MST_ADD

Address	Bank 00 - 0x0B (Hex) - 11 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R/W							
Default	00000000							

Description

I^2C master register address.

Master I^2C address register.

Back to Register Overview

I2C_MST_RW

Address	Bank 00 - 0x0C (Hex) - 12 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R/W							
Default	00000000							

Description

I^2C master register data.

Master I^2C data read/write register.

Back to Register Overview

MAG_OFS_X_MSB

Address	Bank 00 - 0x0D (Hex) - 13 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R/W							
Default	00000000							

Description

Magnetometer offset X, MSB.

Magnetometer MSB of the offset for compensation on X direction.

Back to Register Overview

MAG_OFS_X_LSB

Address	Bank 00 - 0x0E (Hex) - 14 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R/W							
Default	00000000							

Description

Magnetometer offset X, LSB.

Magnetometer LSB of the offset for compensation on X direction.

Back to Register Overview

MAG_OFS_Y_MSB

Address	Bank 00 - 0x0F (Hex) - 15 (Dec)							
Bit #	7	6	5	4	3	2	1	0
Fields	-							
Access	R/W							
Default	00000000							

Description

Magnetometer offset Y, MSB.
Magnetometer MSB of the offset for compensation on Y direction.

Back to Register Overview

MAG_OFS_Y_LSB

Address	Bank 00 - 0x10 (Hex) - 16 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R/W							
Default	00000000							

Description

Magnetometer offset Y, LSB.

Magnetometer LSB of the offset for compensation on Y direction.

Back to Register Overview

MAG_OFS_Z_MSB

Address	Bank 00 - 0x11 (Hex) - 17 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R/W							
Default	00000000							

Description

Magnetometer offset Z, MSB.

Magnetometer MSB of the offset for compensation on Z direction.

Back to Register Overview

MAG_OFS_Z_LSB

Address	Bank 00 - 0x12 (Hex) - 18 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R/W							
Default	00000000							

Description

Magnetometer offset Z, LSB.

Magnetometer LSB of the offset for compensation on Z direction.

Back to Register Overview

DR_CFG

Address	Bank 00 - 0x13 (Hex) - 19 (Dec)							
Bit #	7	6	5	4	3	2	1	0
Fields	***mst_pad_bypass_en***	RFU	***mst_i2c_sngl_rw***	***mst_i2c_sngl_en***	***dr_rst_mode***		***coarse_temp***	***temp_en***
Access	R/W	R	R/W	R/W	R/W		R/W	R/W
Default	0	—	0	0	00		0	1

Description

Data ready configuration

Fields

mst_pad_bypass_en: Enables the bypass of the I^2C master pads.

0: Normal
1: Bypass

mst_i2c_sngl_rw: Sets the direction of the I^2C master operation.

0: Write
1: Read

mst_i2c_sngl_en: Enables a one-shot single byte read/write operation.

0: Disable
1: Enable

dr_rst_mode: Sets the DATA_READY interrupt reset and data-out update mode.

Value	Mnemonic	Description
00	All	DATA_READY is cleared after all the active channels are read. Data are not updated until the clear operation is accomplished.
01	Any	DATA_READY is cleared when at least a byte of one of the active channels is read. Data are updated independently from the clear operation.
10	Status	DATA_READY is cleared when status register is read. Data are not updated until the clear operation is accomplished.

coarse_temp: Sets the temperature data update granularity.

Value	Mnemonic	Description
0	Fine	Temperature data is updated only when both the temperature data registers are read (TEMP_H, TEMP_L).
1	Coarse	Temperature data is updated only when the most significant byte of the temperature data is read (TEMP_H).

temp_en: Enables the temperature sensor.

0: Disabled
1: Enabled

Back to Register Overview

IO_CFG

Address	Bank 00 - 0x14 (Hex) - 20 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	*aux_pd_en*	*aux_pu_en*	*int1_pd_en*	*int1_pu_en*	*int2_pd_en*	*int2_pu_en*	*mst_pu_dis*	*slv_pu_dis*
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

Description

Input/output configuration

Fields

aux_pd_en: Connects the internal pulldown of the AUX pad.

0: Disconnected
1: Connected

aux_pu_en: Connects the internal pullup of the AUX pad.

0: Disconnected
1: Connected

int1_pd_en: Connects the internal pulldown of the INT1 pad.

0: Disconnected
1: Connected

int1_pu_en: Connects the internal pullup of the INT1 pad.

0: Disconnected
1: Connected

int2_pd_en: Connects the internal pulldown of the INT2 pad.

0: Disconnected
1: Connected

int2_pu_en: Connects the internal pullup of the INT2 pad.

0: Disconnected
1: Connected

mst_pu_dis: Disconnects the internal pullups of the I^2C master pads.

0: Connected
1: Disconnected

slv_pu_dis: Disconnect the internal pullups of the I^2C slave pads.

0: Connected
1: Disconnected

Back to Register Overview

I2C_PAD

Address	Bank 00 - 0x15 (Hex) - 21 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU					***strong_slave***	***strong_mst***	***strong_int_aux***
Access	R					R/W	R/W	R/W
Default	—					1	0	0

Description

PADs configuration

Fields

strong_slave: Increases the I^2C slave pads speed. This mode absorbs more current.

0: Normal (low power)
1: Fast

strong_mst: Increases the I^2C master pads speed. This mode absorbs more current.

0: Normal (low power)
1: Fast

strong_int_aux: Increases the aux pad speed. This mode absorbs more current.

0: Normal (low power)
1: Fast

Back to Register Overview

I2C_CFG

Address	Bank 00 - 0x16 (Hex) - 22 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	*mst_i2c_pad_settings*		*i2c_setting*			*spi_3_wire*	*endian*	*i2c_off*
Access	R/W		R/W			R/W	R/W	R/W
Default	00		000			0	0	0

Description

Serial interfaces configuration

Fields

mst_i2c_pad_settings: Sets the I^2C master pad configuration.

00: Without antispike filter
01: Standard configuration
10: Without filters and delays

i2c_setting: Sets the I^2C configuration.

000: I^2C fast mode standard configuration
001: I^2C fast mode without anti-spike filter
010: I^2C high-speed standard configuration
011: I^2C high-speed without anti-spike filter
100: I^2C mode without filters and delays
101: SPI interface recommended

spi_3_wire: Sets the SPI mode.

0: SPI Mode 4
1: SPI Mode 3

endian: Sets the data output endian.

0: Big endian
1: Little endian

i2c_off: Turns on/off the I^2C interface.

0: Active
1: Off

Back to Register Overview

FIFO_THS

Address	Bank 00 - 0x17 (Hex) - 23 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R/W							
Default	00000000							

Description

FIFO threshold configuration

Sets the FIFO threshold. When the number of sample stored in FIFO is above the threshold (fifo_ths_msb, fifo_ths_lsb), an interrupt is generated. The FIFO threshold shall be different from 0.

Back to Register Overview

FIFO_CFG

Address	Bank 00 - 0x18 (Hex) - 24 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	*fifo_mode*		*fifo_int_mode*	*fifo_overrun*	*fifo_store_quat*	*fifo_store_mag*	*fifo_store_acc*	*fifo_store_gyr*
Access	R/W		R/W	R/W	R/W	R/W	R/W	R/W
Default	00		0	0	0	0	0	0

Description

FIFO mode configuration

Fields

fifo_mode: Sets the FIFO mode.

Value	Mnemonic	Description
00	Off	FIFO is disabled and the data are exposed through the registers only.
01	Normal	FIFO starts collecting the data immediately.
10	Interrupt	FIFO starts collecting the data right after a Rate Interrupt event (either OR or AND) is generated.
11	Snapshot	FIFO starts collecting the data immediately, overwriting the oldest data in case of FIFO full. When a Rate Interrupt event (either OR or AND) is generated, data are collected until the FIFO is full, then the data collection is stopped without overwriting the oldest values. This mechanism is useful in case a post-processing of the data that generated the Rate Interrupt event is requested to better classify the event itself.

fifo_int_mode: When an interrupt mode is selected, these bits define which kind of mask must be used.

0: OR mask
1: AND mask

fifo_overrun: Sets the FIFO overrun mode. If overwrite, the FIFO is full the old data are overwritten. If stop on full, the FIFO stops collecting data when full.

0: Stop on full
1: Overwrite

fifo_store_quat: Stores the quaternion data in the FIFO.

0: No Quaternions in the FIFO
1: Quaternions in the FIFO

fifo_store_mag: Stores the magnetometer data in the FIFO.

0: No Magnetometer in the FIFO
1: Magnetometer in the FIFO

fifo_store_acc: Stores the accelerometer data in the FIFO.

0: No Accelerometer in the FIFO
1: Accelerometer in the FIFO

fifo_store_gyr: Stores the gyroscope data in the FIFO.

0: No Gyroscope in the FIFO
1: Gyroscope in the FIFO

Back to Register Overview

DSYNC_CFG

Address	Bank 00 - 0x1A (Hex) - 26 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	***ds_queue_en_r***	***ds_queue_en_f***	***ds_wakeup_act_edg***	***ds_wakeup_act_low***	***ds_gyro_map_en***	***ds_acc_map_en***	***ds_mag_map_en***	***ds_temp_map_en***
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

Description

DSYNC configuration

Fields

ds_queue_en_r: Enables the data queueing when data_sync rising edge.
0: Disabled
1: Enabled

ds_queue_en_f: Enables the data queueing when data_sync falling edge.
0: Disabled
1: Enabled

ds_wakeup_act_edg: Selects if the data_sync is active on edge or on level.
0: Active on level
1: Active on edge

ds_wakeup_act_low: Selects the active level of the data_sync to control the wake-up.
0: Active high
1: Active low

ds_gyro_map_en: Enables the mapping of the data_sync signal to the gyroscope LSB.
0: Disabled
1: Enabled

ds_acc_map_en: Enables the mapping of the data_sync signal to the accelerometer LSB.
0: Disabled
1: Enabled

ds_mag_map_en: Enables the mapping of the data_sync signal to the mag LSB.
0: Disabled
1: Enabled

ds_temp_map_en: Enables the mapping of the data_sync signal to the temperature LSB.
0: Disabled
1: Enabled

Back to Register Overview

DSYNC_CNT

Address	Bank 00 - 0x1B (Hex) - 27 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU				***data_sync_count***			
Access	R				R/W			
Default	—				0000			

Description

DSYNC counter

Fields

data_sync_count: This register specifies the number of samples to be stored into the FIFO upon detecting a DATA_SYNC active edge.

Back to Register Overview

ITF_OTP

Address	Bank 00 - 0x1C (Hex) - 28 (Dec)							
Bit #	7	6	5	4	3	2	1	0
Fields	RFU	*parity_error*	*if_parity*		*otp_ecc_stat*		*chrp_in_prgs*	*otp_dld_restart*
Access	R	R	R/W		R		R	R/W
Default	—	0	00		00		0	0

Description

Serial interfaces and OTP control

Fields

parity_error: Indicates if an error in SPI/I^2C address occurred.

if_parity: Parity setting.

00: Bit 6 is used for autoincrement mode.
01: Bit 6 represents the even parity bit.
10: Bit 6 represents the odd parity bit.

otp_ecc_stat: OTP download status.

00: Program OK
01: Program fail
10: Program fail
11: Program fail

chrp_in_prgs: Status of the stiction recovery operation.

0: Idle
1: In progress

otp_dld_restart: Triggers the OTP download.

0: Idle
1: Trigger

Back to Register Overview

12.2.3Bank 01

INT_REF_X

Address	Bank 01 - 0x00 (Hex) - 0 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R/W							
Default	00000000							

Description

Interrupt reference for X axis.

Sets the MSB of the reference for interrupt of X axis.
When int_single_ref is set, the reference will be {int_ref_x, int_ref_y}, with int_ref_x MSB byte and int_ref_y LSB byte.

Back to Register Overview

INT_REF_Y

Address	Bank 01 - 0x01 (Hex) - 1 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R/W							
Default	00000000							

Description

Interrupt reference for Y axis.

Sets the MSB of the reference for interrupt of Y axis.
When int_single_ref is set, the reference will be (int_ref_x, int_ref_y) wth int_ref_x MSB byte and int_ref_y LSB byte.

Back to Register Overview

INT_REF_Z

Address	Bank 01 - 0x02 (Hex) - 2 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R/W							
Default	00000000							

Description

Interrupt reference for Z axis

Set the MSB of the reference for interrupt of Z axis.
When int_single_ref is set, the reference will be (int_ref_x, int_ref_y) wth int_ref_x MSB byte and int_ref_y LSB byte.

Back to Register Overview

INT_DEB_X

Address	Bank 01 - 0x03 (Hex) - 3 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU				***int_deb_x***			
Access	R				R/W			
Default	—				0000			

Description

Interrupt debounce, X axis

Fields

int_deb_x: Sets the number of contiguous X axis samples that have to respect the interrupt condition before issuing the interrupt signal.
The actual value is 2 x int_deb_x + 1.

Back to Register Overview

INT_DEB_Y

Address	Bank 01 - 0x04 (Hex) - 4 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU				***int_deb_y***			
Access	R				R/W			
Default	—				0000			

Description

Interrupt debounce, Y axis

Fields

int_deb_y: Sets the number of contiguous Y axis samples that have to respect the interrupt condition before issuing the interrupt signal.
The actual value is 2 x int_deb_y + 1.

Back to Register Overview

INT_DEB_Z

Address	Bank 01 - 0x05 (Hex) - 5 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU				***int_deb_z***			
Access	R				R/W			
Default	—				0000			

Description

Interrupt debounce, Z axis

Fields

int_deb_z: Sets the number of contiguous Z axis samples that have to respect the interrupt condition before issuing the interrupt signal.
The actual value is 2 x int_deb_z + 1.

Back to Register Overview

INT_MSK_X

Address	Bank 01 - 0x06 (Hex) - 6 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	*int_mask_x*				*int_x_high_pos*	*int_x_low_pos*	*int_x_high_neg*	*int_x_low_neg*
Access	R/W				R	R	R	R
Default	0000				0	0	0	0

Description

Interrupt mask, X axis zones

Fields

int_mask_x: For each bit mask it enables (1) or disables (0) the interrupt generation for threshold event detection on X axis.
Bit 0: enable (1)/disable (0) the event int_x_low_neg.
Bit 1: enable (1)/disable (0) the event int_x_high_neg.
Bit 2: enable (1)/disable (0) the event int_x_low_pos.
Bit 3: enable (1)/disable (0) the event int_x_high_pos.

For example: writing int_mask_x = 4'b0100 only the event int_x_low_pos will generate an interrupt.

int_x_high_pos: Signal is positive, above threshold.
0: False
1: True

int_x_low_pos: Signal is positive, below threshold.
0: False
1: True

int_x_high_neg: Signal is negative, above threshold.
0: False
1: True

int_x_low_neg: Signal is negative below threshold.
0: False
1: True

Back to Register Overview

INT_MSK_Y

Address	Bank 01 - 0x07 (Hex) - 7 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	*int_mask_y*				*int_y_high_pos*	*int_y_low_pos*	*int_y_high_neg*	*int_y_low_neg*
Access	R/W				R	R	R	R
Default	0000				0	0	0	0

Description

Interrupt mask, Y axis zones

Fields

int_mask_y: For each bit mask it enables (1) or disables (0) the interrupt generation for threshold event detection on Y axis.
Bit 0: enable (1)/disable (0) the event int_y_low_neg.
Bit 1: enable (1)/disable (0) the event int_y_high_neg.
Bit 2: enable (1)/disable (0) the event int_y_low_pos.
Bit 3: enable (1)/disable (0) the event int_y_high_pos.

For example: writing int_mask_y = 4'b0100 only the event int_y_low_pos will generate an interrupt.

int_y_high_pos: Signal is positive, above threshold.
0: False
1: True

int_y_low_pos: Signal is positive, below threshold.
0: False
1: True

int_y_high_neg: Signal is negative, above threshold.
0: False
1: True

int_y_low_neg: Signal is negative below threshold.
0: False
1: True

Back to Register Overview

INT_MSK_Z

Address	Bank 01 - 0x08 (Hex) - 8 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	*int_mask_z*				*int_z_high_pos*	*int_z_low_pos*	*int_z_high_neg*	*int_z_low_neg*
Access	R/W				R	R	R	R
Default	0000				0	0	0	0

Description

Interrupt mask, Z axis zones.

Fields

int_mask_z: For each bit mask it enables (1) or disables (0) the interrupt generation for threshold event detection on Z axis.
Bit 0: enable (1)/disable (0) the event int_z_low_neg.
Bit 1: enable (1)/disable (0) the event int_z_high_neg.
Bit 2: enable (1)/disable (0) the event int_z_low_pos.
Bit 3: enable (1)/disable (0) the event int_z_high_pos.

For example: writing int_mask_z = 4'b0100 only the event int_z_low_pos will generate an interrupt.

int_z_high_pos: Signal is positive, above threshold.
0: False
1: True

int_z_low_pos: Signal is positive, below threshold.
0: False
1: True

int_z_high_neg: Signal is negative, above threshold.
0: False
1: True

int_z_low_neg: Signal is negative below threshold.
0: False
1: True

Back to Register Overview

INT_MASK_AO

Address	Bank 01 - 0x09 (Hex) - 9 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU	***int_freeze***	***int_mask_xyz_and***			***int_mask_xyz_or***		
Access	R	R/W	R/W			R/W		
Default	—	0	000			000		

Description

Interrupt masks, and/or

Fields

int_freeze: Enables interrupt on threshold freeze.
When enabled, the INT{X,Y,Z}_{HIGH,LOW}_{NEG,POS} flags hold values until interrupt will be cleared.

0: Disabled
1: Enabled

int_mask_xyz_and: Each bit activates an axis. The active axes are ANDed together to generate the AND interrupt.

int_mask_xyz_or: Each bit activates an axis. The active axes are ORed together to generate the OR interrupt.

Back to Register Overview

INT_CFG_1

Address	Bank 01 - 0x0A (Hex) - 10 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	*sns_intp_fsc*		*int1_clk_out*	*int2_clk_out*	*int_single_deb*	*int_single_ref*	*sns_intp_hpf*	RFU
Access	R/W		R/W	R/W	R/W	R/W	R/W	R
Default	00		0	0	0	0	0	—

Description

Interrupt configuration register 1

Fields

sns_intp_fsc: Sets the full-scale applied to the gyroscope data used to issue the AND/OR interrupts.

00: 2000 dps
01: 1000 dps
10: 500 dps
11: 250 dps

int1_clk_out: INT1 pad drives out the internal clock (8.8MHz).

0: Off
1: On

int2_clk_out: INT2 pad drives out the internal clock (8.8MHz).

0: Off
1: On

int_single_deb: When on, the same duration (int_deb_x) is used for all the axes.

0: Off
1: On

int_single_ref: When on, the same threshold (int_ref_x, int_ref_y) is used for all the axes.

0: Off
1: On

sns_intp_hpf: Enable the highpass filtering to the gyroscope data out used to issue the AND/OR interrupts.

0: Without highpass filtering
1: With highpass filtering

Back to Register Overview

INT_CFG_2

Address	Bank 01 - 0x0B (Hex) - 11 (Dec)							
Bit #	7	6	5	4	3	2	1	0
Fields	RFU	*merge_int2_to_int1*	*int1_enable*	*int1_active_level*	*int1_out_mode*	*int2_enable*	*int2_active_level*	*int2_out_mode*
Access	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	—	0	1	0	0	1	0	0

Description

Interrupt configuration register 2

Fields

merge_int2_to_int1: Enables to merge the INT1 and INT2. When set, the interrupt INT1 changes its output depending on both INT1 and INT2.

0: INT1 and INT2 separated
1: INT2 merged with INT1

int1_enable: Enables interrupts on INT1.

0: Disabled
1: Enabled

int1_active_level: Sets the INT1 active level.

0: INT1 active high
1: INT1 active low

int1_out_mode: Sets the INT1 output configuration.

0: Push-pull
1: Open drain

int2_enable: Enables interrupts on INT2.

0: Disabled
1: Enabled

int2_active_level: Sets the INT2 active level.

0: INT2 active high
1: INT2 active low

int2_out_mode: Sets the INT2 output configuration.

0: Push-pull
1: Open drain

Back to Register Overview

INT_TMO

Address	Bank 01 - 0x0C (Hex) - 12 (Dec)							
Bit #	7	6	5	4	3	2	1	0
Fields	*int1_latch_mode*		*int2_latch_mode*		*int_timeout*			
Access	R/W		R/W		R/W			
Default	00		00		0000			

Description

Interrupt timeout

Fields

int1_latch_mode: Set the INT1 latch mode.

00: Not latched
01: Latched - Clear on read
10: Latched - Clear on write

int2_latch_mode: Set the INT2 latch mode.

00: Not latched
01: Latched - Clear on read
10: Latched - Clear on write

int_timeout: Interrupt temporary period. This is shared between INT1 and INT2.

0000: 125µs
0001: 250µs
0010: 500µs
0011: 1ms
0100: 2ms
0101: 5ms
0110: 10ms
0111: 20ms
1000: 50ms
1001: 100ms
1010: 200ms
1011: 500ms

Back to Register Overview

INT_STS_UL

Address	Bank 01 - 0x0D (Hex) - 13 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	***sts_ul_data_ready***	***sts_ul_fifo_empty***	***sts_ul_fifo_overrun***	***sts_ul_fifo_ths***	***sts_ul_int_and***	***sts_ul_int_or***	***sts_ul_otp_downloading***	***sts_ul_data_sync***
Access	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0

Description

Interrupt sources, unlatched.

Unlatched interrupts status bits.

Fields

sts_ul_data_ready: DATA_READY status

sts_ul_fifo_empty: FIFO_EMPTY status

sts_ul_fifo_overrun: FIFO_OVERRUN status

sts_ul_fifo_ths: FIFO_THRESHOLD status

sts_ul_int_and: INT_AND status

sts_ul_int_or: INT_OR status

sts_ul_otp_downloading: OTP_DOWNLOADING status

sts_ul_data_sync: DATA_SYNC status

Back to Register Overview

INT_STS

Address	Bank 01 - 0x0E (Hex) - 14 (Dec)							
Bit #	7	6	5	4	3	2	1	0
Fields	***sts_i1_data_ready***	***sts_i1_fifo_empty***	***sts_i1_fifo_overrun***	***sts_i1_fifo_ths***	***sts_i2_int_and***	***sts_i2_int_or***	***sts_i2_otp_downloading***	***sts_i2_data_sync***
Access	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0

Description

Interrupt status register

Latched status bits.

Fields

sts_i1_data_ready: DATA_READY status on INT1
Added for user convenience, it is the same as sts_ul_data_ready in INT_STATUS_UL, so it is unlatched.

sts_i1_fifo_empty: FIFO_EMPTY status on INT1

sts_i1_fifo_overrun: FIFO_OVERRUN status on INT1

sts_i1_fifo_ths: FIFO_THRESHOLD status on INT1

sts_i2_int_and: INT_AND status on INT2

sts_i2_int_or: INT_OR status on INT2

sts_i2_otp_downloading: OTP_DOWNLOADING status on INT2

sts_i2_data_sync: DATA_SYNC status on INT2

Back to Register Overview

INT_MSK

Address	Bank 01 - 0x0F (Hex) - 15 (Dec)							
Bit #	7	6	5	4	3	2	1	0
Fields	***msk_i1_data_ready***	***msk_i1_fifo_empty***	***msk_i1_fifo_overrun***	***msk_i1_fifo_ths***	***msk_i2_int_and***	***msk_i2_int_or***	***msk_i2_otp_downloading***	***msk_i2_data_sync***
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	1	0	0	0	0	0	1	0

Description

Interrupt mask register

Enables the interrupt status (0: disabled, 1: enabled).

Fields

msk_i1_data_ready: DATA_READY status on INT1

msk_i1_fifo_empty: FIFO_EMPTY status on INT1

msk_i1_fifo_overrun: FIFO_OVERRUN status on INT1

msk_i1_fifo_ths: FIFO_THRESHOLD status on INT1

msk_i2_int_and: INT_AND status on INT2

msk_i2_int_or: INT_OR status on INT2

msk_i2_otp_downloading: OTP_DOWNLOADING status on INT2

msk_i2_data_sync: DATA_SYNC status on INT2

Back to Register Overview

INT_SRC_SEL

Address	Bank 01 - 0x17 (Hex) - 23 (Dec)							
Bit #	7	6	5	4	3	2	1	0
Fields	RFU		***msk_gyr_int_d_rdy***	***msk_acc_int_d_rdy***	***msk_mag_int_d_rdy***	***msk_qua_int_d_rdy***	***int_src_cfg***	
Access	R		R/W	R/W	R/W	R/W	R/W	
Default	—		1	1	1	1	00	

Description

Interrupt source selection

Fields

msk_gyr_int_d_rdy: If set, a new gyroscope data contributes to DATA_READY interrupt assertion.

0: None
1: New gyroscope data to data ready

msk_acc_int_d_rdy: If set, a new accelerometer data contributes to DATA_READY interrupt assertion.

0: None
1: New accelerometer data to data ready

msk_mag_int_d_rdy: If set, a new magnetometer data set contributes to data ready interrupt assertion.

0: None
1: New magnetometer data to data ready

msk_qua_int_d_rdy: When 1 a new quaternion data set contributes to data ready interrupt assertion.

0: None
1: New quaternion data to data ready

int_src_cfg: Configures the interrupt source.

00: gyr_x, gyr_y, gyr_z
10: norm(a_hpf), norm(a_hpf), norm(a_pre_hpf)
11: norm(a_hpf), norm(a_pre_hpf), norm(a_pre_hpf)

Back to Register Overview

SERIAL_5

Address	Bank 01 - 0x1A (Hex) - 26 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Unique serial number, byte 5

Serial number.

Back to Register Overview

SERIAL_4

Address	Bank 01 - 0x1B (Hex) - 27 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Unique serial number, byte 4.

Serial number.

Back to Register Overview

SERIAL_3

Address	Bank 01 - 0x1C (Hex) - 28 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Unique serial number, byte 3.

Serial number.

Back to Register Overview

SERIAL_2

Address	Bank 01 - 0x1D (Hex) - 29 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Unique serial number, byte 2

Serial number.

Back to Register Overview

SERIAL_1

Address	Bank 01 - 0x1E (Hex) - 30 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Unique serial number, byte 1

Serial number.

Back to Register Overview

SERIAL_0

Address	Bank 01 - 0x1F (Hex) - 31 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Unique serial number, byte 0.

Serial number.

Back to Register Overview

12.2.4Bank 02

QUAT0_H

Address	Bank 02 - 0x00 (Hex) - 0 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [15:8] of the quaternion 0 measurement.

Most significant byte of the X component of the quaternion.

Back to Register Overview

QUAT0_L

Address	Bank 02 - 0x01 (Hex) - 1 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [7:0] of the quaternion 0 measurement

Least significant byte of the X component of the quaternion.

Back to Register Overview

QUAT1_H

Address	Bank 02 - 0x02 (Hex) - 2 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [15:8] of the quaternion 1 measurement

Most significant byte of the Y component of the quaternion.

Back to Register Overview

QUAT1_L

Address	Bank 02 - 0x03 (Hex) - 3 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [7:0] of the quaternion 1 measurement

Least significant byte of the Y component of the quaternion.

Back to Register Overview

QUAT2_H

Address	Bank 02 - 0x04 (Hex) - 4 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [15:8] of the quaternion 2 measurement

Most significant byte of the Z component of the quaternion.

Back to Register Overview

QUAT2_L

Address	Bank 02 - 0x05 (Hex) - 5 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [7:0] of the quaternion 2 measurement

Least significant byte of the Z component of the quaternion.

Back to Register Overview

QUAT3_H

Address	Bank 02 - 0x06 (Hex) - 6 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [15:8] of the quaternion 3 measurement.

Most significant byte of the scalar component of the quaternion.

Back to Register Overview

QUAT3_L

Address	Bank 02 - 0x07 (Hex) - 7 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	—							
Access	R							
Default	00000000							

Description

Bits [7:0] of the quaternion 3 measurement.

Least significant byte of the scalar component of the quaternion.

Back to Register Overview

BIAS_GYRO_X_H

Address	Bank 02 - 0x13 (Hex) - 19 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU			***bias_comp_gyr_x_msb***				
Access	R			R/W				
Default	—			00000				

Description

Gyroscope bias compensation, X axis, MSB

Fields

bias_comp_gyr_x_msb: Most significant bits of the 13-bit gyroscope offset compensation register for the X axis.
The real value is computed as: bias_comp_gyr x 8.33mdps.
The real value range is: [-34.13; +34.13] dps.

Back to Register Overview

BIAS_GYRO__X_L

Address	Bank 02 - 0x14 (Hex) - 20 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	***bias_comp_gyr_x_lsb***							
Access	R/W							
Default	00000000							

Description

Gyroscope bias compensation, X axis, LSB

Fields

bias_comp_gyr_x_lsb: Least significant byte of the 13-bit gyroscope offset compensation register for the X axis.
The real value is computed as: bias_comp_gyr x 8.33mdps.
The real value range is: [-34.13; +34.13] dps.

Back to Register Overview

BIAS_GYRO_Y_H

Address	Bank 02 - 0x15 (Hex) - 21 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU			***bias_comp_gyr_y_msb***				
Access	R			R/W				
Default	—			00000				

Description

Gyroscope bias compensation, Y axis, MSB

Fields

bias_comp_gyr_y_msb: Most significant bits of the 13-bit gyroscope offset compensation register for the Y axis.
The real value is computed as: bias_comp_gyr x 8.33mdps.
The real value range is: [-34.13; +34.13] dps.

Back to Register Overview

BIAS_GYRO_Y_L

Address	Bank 02 - 0x16 (Hex) - 22 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	***bias_comp_gyr_y_lsb***							
Access	R/W							
Default	00000000							

Description

Gyroscope bias compensation, Y axis, LSB

Fields

bias_comp_gyr_y_lsb: Least significant byte of the 13-bit gyroscope offset compensation register for the Y axis.
The real value is computed as: bias_comp_gyr x 8.33mdps.
The real value range is: [-34.13; +34.13] dps.

Back to Register Overview

BIAS_GYRO_Z_H

Address	Bank 02 - 0x17 (Hex) - 23 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU			***bias_comp_gyr_z_msb***				
Access	R			R/W				
Default	—			00000				

Description

Gyroscope bias compensation, Z axis, MSB.

Fields

bias_comp_gyr_z_msb: Most significant bits of the 13-bit gyroscope offset compensation register for the Z axis.
The real value is computed as: bias_comp_gyr x 8.33mdps.
The real value range is: [-34.13; +34.13] dps.

Back to Register Overview

BIAS_GYRO_Z_L

Address	Bank 02 - 0x18 (Hex) - 24 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	***bias_comp_gyr_z_lsb***							
Access	R/W							
Default	00000000							

Description

Gyroscope bias compensation, Z axis, LSB

Fields

bias_comp_gyr_z_lsb: Least significant byte of the 13-bit gyroscope offset compensation register for the Z axis.
The real value is computed as: bias_comp_gyr x 8.33mdps.
The real value range is: [-34.13; +34.13] dps.

Back to Register Overview

BIAS_ACC_X

Address	Bank 02 - 0x19 (Hex) - 25 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU	***bias_comp_acc_x***						
Access	R	R/W						
Default	—	0000000						

Description

Accelerometer bias compensation, X axis

Fields

bias_comp_acc_x: Accelerometer offset compensation register for the X axis.
The real value is computed as: bias_comp_acc x 4.2666mg.
The real value range is: [-0.54; +0.54] g.

Back to Register Overview

BIAS_ACC_Y

Address	Bank 02 - 0x1A (Hex) - 26 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU	***bias_comp_acc_y***						
Access	R	R/W						
Default	—	0000000						

Description

Accelerometer bias compensation, Y axis

Fields

bias_comp_acc_y: Accelerometer offset compensation register for the Y axis.
The real value is computed as: bias_comp_acc x 4.2666mg.
The real value range is: [-0.54; +0.54] g.

Back to Register Overview

BIAS_ACC_Z

Address	Bank 02 - 0x1B (Hex) - 27 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	RFU	***bias_comp_acc_z***						
Access	R	R/W						
Default	—	0000000						

Description

Accelerometer bias compensation, Z axis

Fields

bias_comp_acc_z: Accelerometer offset compensation register for the Z axis.
The real value is computed as: bias_comp_acc x 4.2666 mg.
The real value range is: [-0.54; +0.54] g.

Back to Register Overview

FUS_CFG0

Address	Bank 02 - 0x1C (Hex) - 28 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	*fus_odr*			*fus_gr_hd_en*	*fus_auto_mode*	*fus_mode*		*fus_en*
Access	R/W			R/W	R/W	R/W		R/W
Default	000			0	0	00		0

Description

MME configuration register 1

Fields

fus_odr: Fusion output data rate (ODR), computed as a ratio of the sense ODR (sns_odr).
The sense ODR is the maximum between the accelerometer ODR and the gyroscope ODR.

Fusion ODR = ***sns_odr*** / (***fus_odr*** + 1)

fus_gr_hd_en: When = 1, gravity and heading estimation is enabled and outputted in place of quaternion.

0: Disabled
1: Enabled

fus_auto_mode: Enables the automatic managing of the fusion mode.
When enabled, the fusion mode is automatically managed according to power modes and sensor anomalies.

0: Normal
1: Auto

fus_mode: Sets the fusion mode.

00: G+A+M
01: A+M
10: G only
11: G+A+M

fus_en: Enables the fusion computation to produce the quaternion data.

0: Disabled
1: Enabled

Back to Register Overview

FUS_CFG1

Address	Bank 02 - 0x1D (Hex) - 29 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	*fus_alpha0_trim*			*fus_alpha1_trim*		*force_mag_anomaly*	*mag_anomaly_mon_en*	*mag_sw*
Access	R/W			R/W		R/W	R/W	R/W
Default	010			11		0	0	0

Description

MME configuration register 2

Fields

fus_alpha0_trim: Complementary adaptive filter minimum alpha0 factor.

000: alpha0 = 0.95
001: alpha0 = 0.96
010: alpha0 = 0.97
011: alpha0 = 0.98
100: alpha0 = 0.99
101: alpha0 = 0.994
110: alpha0 = 0.997
111: alpha0 = 0.999

fus_alpha1_trim: Complementary adaptive filter minimum alpha1 factor.

00: alpha1 = alpha0
01: alpha1 = alpha0*2/3 + 1/3
10: alpha1 = alpha0*1/3 + 2/3
11: alpha1 = 1

force_mag_anomaly: When written high, magnetometer anomaly is emulated through SW.

mag_anomaly_mon_en: When high, Mag X read data LSB is used as active high magnetometer anomaly monitor information.

mag_sw: When = 1, magnetometer triad is provide externally through SW on ofs fields after SW compensation.

Back to Register Overview

GYR_ODR_TRIM

Address	Bank 02 - 0x1F (Hex) - 31 (Dec)							
Bit	7	6	5	4	3	2	1	0
Fields	*gyr_odr_corr*				*gyr_ok*	*acc_ok*	*mag_ok*	RFU
Access	R				R	R	R	R
Default	0111				0	0	0	0

Description

Gyroscope ODR correction factor register

Fields

gyr_odr_corr: Real gyroscope output data rate correction factor.

0000: 0.901
0001: 0.915
0010: 0.928
0011: 0.942
0100: 0.956
0101: 0.971
0110: 0.985
0111: 1.000
1000: 1.015
1001: 1.030
1010: 1.046
1011: 1.061
1100: 1.077
1101: 1.093
1110: 1.110
1111: 1.126

gyr_ok: Gyroscope drive loop is locked in amplitude and gyro sense chain is active.

0: Not OK
1: OK

acc_ok: Accelerometer sense chain is active.

0: Not OK
1: OK

mag_ok: Readback information: magnetometer read on I^2C master is active and no magnetic anomaly is detected.

0: Not OK
1: OK

Back to Register Overview

REVISION HISTORY

Revision, Date	Change	Page(s)
1.0, 07/15	Initial release	—