H3LIS331DL: High-*g* (±400 *g*) full scale, low-power 3-axis digital accelerometer

Introduction

This document provides application information for the H3LIS331DL, a high-*g*, low-power 3-axis digital output linear MEMS accelerometer housed in an LGA package.

The H3LIS331DL is a high-*g* full scale, high-performance, low-power 3-axis linear accelerometer with standard digital I²C/SPI serial interface output.

The H3LIS331DL has a dynamically user-selectable full scales of ±100*g*/±200*g*/±400*g* and is capable of measuring acceleration with selectable output data rates from 0.5 Hz to 1 kHz.

The device features low-power operational modes with smart sleep-to-wake-up functions and interrupt signals in response to inertial events whose thresholds and timing are programmable on the fly.

The H3LIS331DL is available in a small, thin plastic land grid array (LGA) package and is guaranteed to operate over a wide temperature range of -40 °C to +85 °C.

1 Pin description

Figure 1. Pin connections

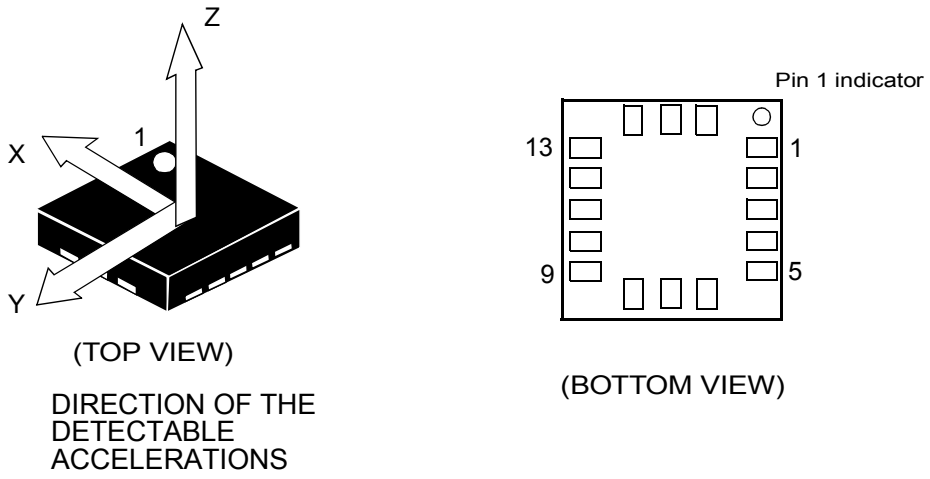


Table 1. Internal pin status

Pin #	Name	Function	Pin status
1	Vdd_IO	Power supply for I/O pins	
2	NC	Not connected	
3	NC	Not connected	
4	SCL SPC	I²C serial clock (SCL) SPI serial port clock (SPC)	Input with pull-up
5	GND	0 V supply	
6	SDA SDI SDO	I²C serial data (SDA) SPI serial data input (SDI) 3-wire interface serial data output (SDO)	Input with pull-up
7	SDO SA0	SPI serial data output (SDO) I²C less significant bit of the device address (SA0)	Input with pull-up
8	CS	SPI enable I²C/SPI mode selection (1: I²C mode; 0: SPI enabled)	Input with pull-up
9	INT 2	Inertial interrupt 2	Output forced to GND
10	Reserved	Connect to GND	
11	INT 1	Inertial interrupt 1	Output forced to GND
12	GND	0 V supply	
13	GND	0 V supply	
14	Vdd	Power supply	
15	Reserved	Connect to Vdd	
16	GND	0 V supply	

2 Registers

Table 2. **Registers**

Register name	Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
WHO_AM_I	0Fh	0	0	1	1	0	0	1	0
CTRL_REG1	20h	PM2	PM1	PM0	DR1	DR0	Zen	Yen	Xen
CTRL_REG2	21h	BOOT	HPM1	HPM0	FDS	HPen2	HPen1	HPCF1	HPCF0
CTRL_REG3	22h	IHL	PP_OD	LIR2	I2_CFG1	I2_CFG0	LIR1	I1_CFG1	I1_CFG0
CTRL_REG4	23h	BDU	BLE	FS1	FS0	0	0	0	SIM
CTRL_REG5	24h	-	-	-	-	-	-	TurnOn1	TurnOn0
HP_FILTER_RESET	25h	-	-	-	-	-	-	-	-
REFERENCE	26h	REF7	REF6	REF5	REF4	REF3	REF2	REF1	REF0
STATUS_REG	27h	ZYXOR	ZOR	YOR	XOR	ZYXDA	ZDA	YDA	XDA
OUTX_L	28h	XD7	XD6	XD5	XD4	XD3	XD2	XD1	XD0
OUTX_H	29h	XD15	XD14	XD13	XD12	XD11	XD10	XD9	XD8
OUTY_L	2Ah	YD7	YD6	YD5	YD4	YD3	YD2	YD1	YD0
OUTY_H	2Bh	YD15	YD14	YD13	YD12	YD11	YD10	YD9	YD8
OUTZ_L	2Ch	ZD7	ZD6	ZD5	ZD4	ZD3	ZD2	ZD1	ZD0
OUTZ_H	2Dh	ZD15	ZD14	ZD13	ZD12	ZD11	ZD10	ZD9	ZD8
INT1_CFG	30h	AOI	0	ZHIE	ZLIE	YHIE	YLIE	XHIE	XLIE
INT1_SRC	31h	-	IA	ZH	ZL	YH	YL	XH	XL
INT1_THS	32h	0	THS6	THS5	THS4	THS3	THS2	THS1	THS0
INT1_DURATION	33h	0	D6	D5	D4	D3	D2	D1	D0
INT2_CFG	34h	AOI	0	ZHIE	ZLIE	YHIE	YLIE	XHIE	XLIE
INT2_SRC	35h	-	IA	ZH	ZL	YH	YL	XH	XL
INT2_THS	36h	0	THS6	THS5	THS4	THS3	THS2	THS1	THS0
INT2_DURATION	37h	0	D6	D5	D4	D3	D2	D1	D0

3 Start-up sequence

Once the device is powered up, it automatically downloads the calibration coefficients from the embedded Flash memory to the internal registers. When the boot procedure is complete (that is, after about 5 milliseconds), the device automatically enters power-down mode.

To turn on the device and gather acceleration data, it is necessary to select one of the operating modes through the CTRL_REG1 register, and to enable at least one of the axes. The following general-purpose sequence can be used to configure the device:

1. Write CTRL_REG1.
2. Write CTRL_REG2.
3. Write CTRL_REG3.
4. Write CTRL_REG4.
5. Write REFERENCE.
6. Write INT1_THS.
7. Write INT1_DUR.
8. Write INT2_THS.
9. Write INT2_DUR.
10. Read HP_FILTER_RESET (if filter is enabled).
11. Write INT1_CFG.
12. Write INT2_CFG.
13. Write CTRL_REG5.

Register values can be changed at any time, and with the device in any operating mode. Modifications take effect immediately.

Note that in case of changes in the full scale or ODR, the output of the device requires 1 ms + 1/ODR to settle (see Table 12). If the HP filter cutoff frequency is changed, the filter can be reset by reading the HP_FILTER_RESET register.

3.1 Reading acceleration data

3.1.1 Using the status register

The device features a STATUS_REG register that should be polled to check when a new set of data is available. The read procedure is the following:

1. Read STATUS_REG.
2. If STATUS_REG[3] = ZYXDA = 0, then go to 1.
3. If STATUS_REG[7] = ZYXOR = 1, then some data have been overwritten.
4. Read OUTX_L.
5. Read OUTX_H.
6. Read OUTY_L.
7. Read OUTY_H.
8. Read OUTZ_L.
9. Read OUTZ_H.
10. Data processing.
11. Go to 1.

The check performed at step 3 determines whether the read rate is adequate compared to the data production rate. In cases where one or more acceleration samples have been overwritten by new data due to an excessively slow read rate, the ZYXOR bit of the STATUS_REG register is set to 1.

The overrun bits are automatically cleared when all the data present inside the device have been read and new data have not been produced in the meantime.

3.1.2 Using the data-ready signal

The device may be configured to have one hardware signal to determine when a new set of measurement data is available for reading. This signal is represented by the XYZDA bit of the STATUS_REG register. The signal can be driven to the INT1 or INT2 pins and its polarity set to active-low or active-high through the CTRL_REG3 register. The interrupt is reset when the higher byte of the data of all the enabled channels has been read.

3.1.3 Using the block data update feature

If reading the acceleration data is particularly slow and cannot be (or does not need to be) synchronized with either the XYZDA bit in the STATUS_REG or with the data-ready signal routed to the INT1/INT2 pin, it is strongly recommended to set the BDU (block data update) bit in CTRL_REG4 to 1.

This feature prevents the need to read the values (most significant and least significant bytes of the acceleration data) related to different samples. In particular, when the BDU is activated, the data registers related to each channel always contain the most recent acceleration data produced by the device. But if the read of a given pair (that is, OUTX_H and OUTX_L, OUTY_H and OUTY_L, OUTZ_H and OUTZ_L) is initiated, the refresh for that pair is blocked until both the MSB and LSB of the data are read.

Note: BDU only guarantees that OUTX(Y, Z)_L and OUTX(X,Z)_H have been sampled at the same moment. If the speed of the read operation is too low, it is possible, for example, to read X and Y sampled at T1 and Z sampled at T2.

3.2 Output data rate selection and read timing

The output data rate is user selectable through the DRx bits of the CTRL_REG1 (20h) register. At power-on-reset, the DRx are reset to 0, thus providing a default output data rate of 50 Hz.

The analog signal coming from the mechanical sensor is filtered by a low-pass filter before being converted by the internal ADC. The frequency at -3 dB of the low-pass filter determines the effective system resolution. The cutoff frequency depends on the DR[1:0] bits in the CTRL_REG1 (20h) register (Table 3).

Table 3. Output data rate

DR1, DR0	Output data rate	Analog filter cutoff frequency (-3 dB)
00	50 Hz	37 Hz
01	100 Hz	74 Hz
10	400 Hz	292 Hz
11	1000 Hz	780 Hz

Note: The output data rate precision is related to the internal oscillator; an error of ±10% should be taken into account.

A typical read period is defined, which is 616 µs shorter than the output data rate period, in order to prevent the loss of any data produced. During this time period, the read of the data must be performed and the data-ready signal can be used as a trigger to begin the read sequence. At the end of the completed sequence, the data-ready signal goes down and the rising edge that follows signals that new data are available. If this minimum read frequency is not observed, some data loss is possible, and the data-ready signal is no longer considered a trigger signal. The status register can be used to infer the occurrence of an overrun.

Figure 2. Read timing

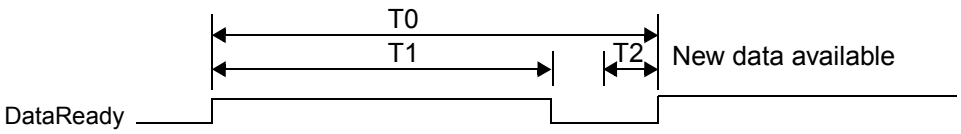


Table 4. Timing value to prevent data loss

Time	Description	Typ
T0	Data rate	1/ODR
T1	Read period	T0-T2
T2	New data generation	616 µs

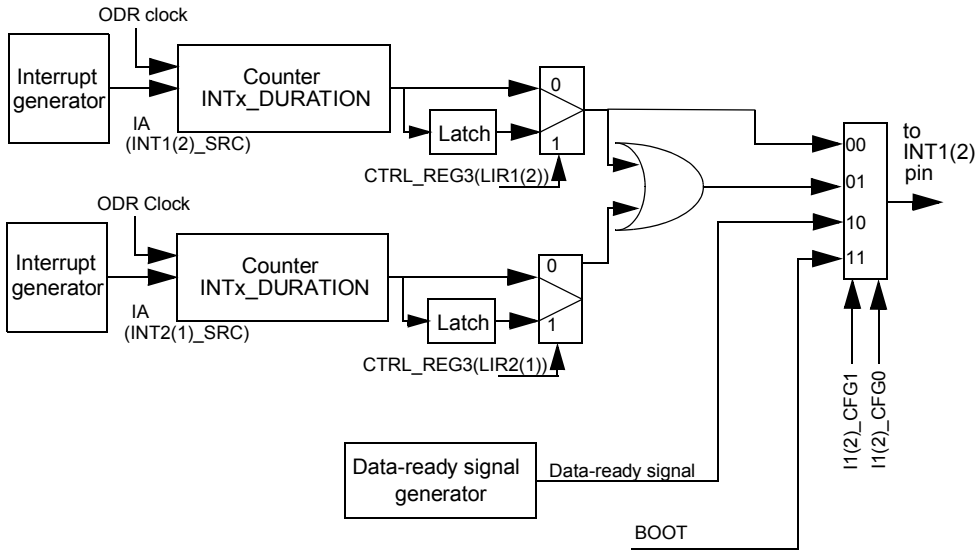
3.3 Data ready vs. interrupt signal

The device has two pins, which can be activated to generate either the data-ready signal or the interrupt signal. The functionality of the pins is selected by acting on bit I1(2)_CFGx bits of the CTRL_REG3 register, according to Table 5 and the block diagram shown in Figure 4.

Table 5. Data signal on INT 1 and INT 2 pins

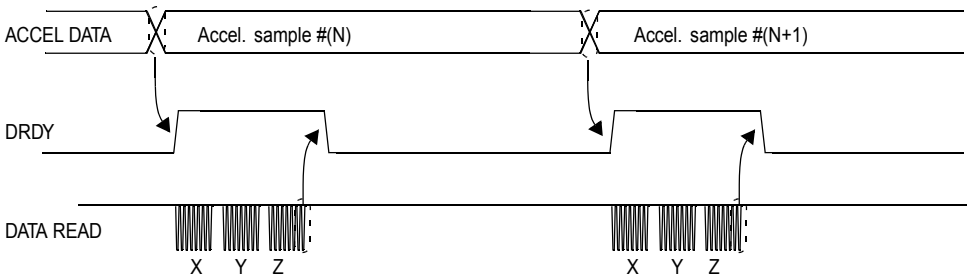
I1(2)_CFG1	I1(2)_CFG0	INT 1(2) pin
0	0	Interrupt 1 (2) source
0	1	Interrupt 1 source OR Interrupt 2 source
1	0	Data ready
1	1	Boot running

Figure 3. Block diagram of interrupt and data-ready signal generation



In particular, the data-ready signal rises to 1 when a new set of acceleration data has been generated and is available to be read. The signal is reset after all the enabled channels are read through the serial interface.

Figure 4. Data-ready signal



3.4 Understanding acceleration data

The measured acceleration data are sent to the OUTX_H, OUTX_L, OUTY_H, OUTY_L, OUTZ_H and OUTZ_L registers. These registers contain, respectively, the most significant byte and the least significant byte of the acceleration signals acting on the X, Y, and Z axes.

The complete acceleration data for the X (Y, Z) channel is given by the concatenation OUTX_H & OUTX_L (OUTY_H & OUTY_L, OUTZ_H & OUTZ_L) and is expressed as a two's complement number.

3.4.1 Data alignment

Acceleration data are represented as 16-bit numbers and are left-justified (12-bit format).

3.4.2 Big-little endian selection

The H3LIS331DL allows the swapping of the content of the lower and the upper part of the acceleration registers (that is, OUTX_H with OUTX_L), to be compliant with both little-endian and big-endian data representations.

"Little endian" means that the low-order byte of the number is stored in memory at the lowest address, and the high-order byte at the highest address (the little end comes first). This mode corresponds to bit BLE in the CTRL_REG4 reset to 0 (default configuration).

Conversely, "big endian" means that the high-order byte of the number is stored in memory at the lowest address, and the low-order byte at the highest address.

3.4.3 Example of acceleration data

The following table provides a few basic examples of the data that are read from the data registers when the device is subjected to a given acceleration. The values listed in the table are based on the assumption that the device is perfectly calibrated (that is, no offset, no gain error, and so forth) and show the effect of the BLE bit.

Table 6. Content of output data registers vs. acceleration (FS = ±100 *g*)

Acceleration values	BLE = 0		BLE = 1	
	Register address			
	28h	29h	28h	29h
0 *g*	00h	00h	00h	00h
17.150 *g*	E0h	15h	15h	E0h
50.176 *g*	00h	40h	40h	00h
-17.150 *g*	20h	EAh	EAh	20h
-50.176 *g*	00h	C0h	C0h	00h

4 Operating modes

The H3LIS331DL can operate in the following four modes, which can be selected through the configuration of CTR_REG1 and CTRL_REG5:

- Normal mode
- Power-down mode
- Low-power mode
- Sleep-to-wake mode

With reference to the datasheet of the device, the power mode (PM) and data rate (DR) bits of the CTRL_REG1 register are used to select the basic operating modes (power-down, normal mode and low-power). The turn-on bits of the CTRL_REG5 register are used to enable sleep-to-wake, which is an advanced mode involving the interrupt configuration also.

Note: The PMx bits are disabled if the TurnOnx bits of the CTRL_REG5 are not configured as zeros.

Table 7. Power mode and low-power output data rate configurations

PM2	PM1	PM0	Power mode selection	Output data rate [Hz] ODR_{LP}
0	0	0	Power-down	--
0	0	1	Normal mode	ODR
0	1	0	Low-power	0.5
0	1	1	Low-power	1
1	0	0	Low-power	2
1	0	1	Low-power	5
1	1	0	Low-power	10

Table 8. CTRL_REG1 - data rate

DR1	DR0	Data rate generation [Hz] ODR
0	0	50
0	1	100
1	0	400
1	1	1000

Table 9. CTRL_REG5 - sleep-to-wake configuration

TurnOn1	TurnOn0	Sleep to wake status
0	0	Sleep-to-wake function disabled
1	1	Sleep-to-wake function enabled

Table 10 and Table 11 show the typical power consumption values (@ Vdd = 2.5 V, T = 25 °C) for the different operating modes.

Note: Higher data rates correspond to lower device resolution.

Table 10. Power consumption - normal mode (µA)

ODR	50 Hz	100 Hz	400 Hz	1000 Hz
Power consumption	260	270	300	370

Table 11. **Power consumption - low-power mode (µA)**

ODR_{LP}	0.5 Hz	1 Hz	2 Hz	5 Hz	10 Hz
DR[1:0] = 00	7	14	22	45	90
DR[1:0] = 01	7	10	17	38	75
DR[1:0] = 10	7	10	17	38	75
DR[1:0] = 11	5	8	15	30	60

4.1 Normal mode

In normal mode, data are generated at the data rate (ODR) selected through the DR bits and for the axes enabled through the Zen, Yen, and Xen bits of the CTRL_REG1 register. Data generated for a disabled axis is 00h.

Data interrupt generation is active and configured through the INT1_CFG and INT2_CFG registers.

4.2 Power-down mode

When the device is in power-down mode, almost all internal blocks of the device are switched off to minimize power consumption. Digital interfaces (I²C and SPI) are still active to allow communication with the device. The content of the configuration register is preserved and the output data registers are not updated, thus keeping in memory the last data sampled before going in power-down mode.

Typical turn-on time to return to normal mode is 1 ms + 1/ODR.

Table 12. **Turn-on time**

Data rate generation (Hz)	Turn-on time - typ. (ms)
50	21
100	11
400	3.5
1000	2

4.3 Low-power mode

When the device is in low-power mode, data are produced at the ODR_{LP} selected by the PM bits of CTRL_REG1.

The turn-on time follows the same rules as for power-down mode (Table 12).

4.4 Sleep-to-wake mode

The sleep-to-wake function, in conjunction with low-power mode, allows further reduction of system power consumption and the development of new smart applications. The H3LIS331DL can be set in a low-power operating mode, characterized by lower date rate refreshes. In this way, the device, even if “sleeping”, continues sensing acceleration and generating interrupt requests.

When the sleep-to-wake function is activated, the H3LIS331DL is able to automatically wake up as soon as the interrupt event has been detected, increasing the output data rate and bandwidth. With this feature, the system can be efficiently switched from low-power mode to full performance, depending on user-selectable positioning and acceleration events, thus ensuring power saving and flexibility.

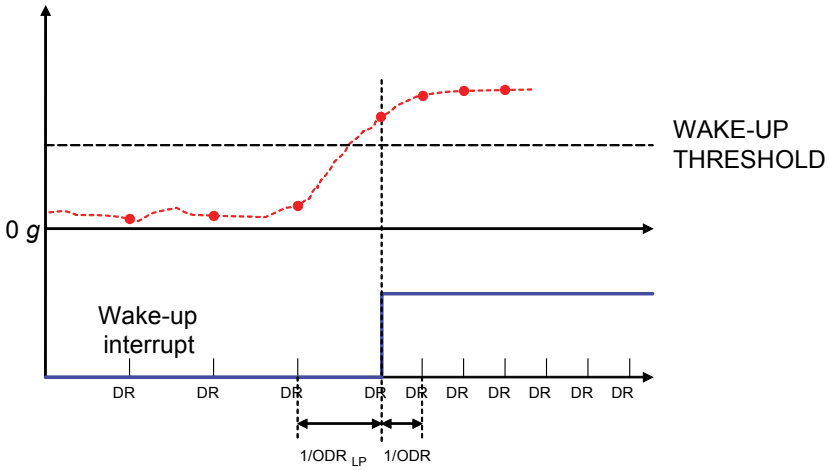
The sleep-to-wake function is activated through the TurnOnx bits of CTRL_REG5 (Table 9).

When the device is in sleep-to-wake mode, it automatically samples the acceleration data at ODR_{LP} to verify if the interrupt conditions are reached. When an interrupt event occurs, the device goes back to generate data at ODR (Figure 5). In case the interrupt conditions are not reached, the device remains in low-power mode at ODR_{LP}.

The device is ready to immediately generate valid samples as soon as it exits from sleep-to-wake mode.

Note: When an interrupt event occurs, the content of CTR_REG5 changes to 0x01 while the content of CTRL_REG1 is left untouched. The PMx bits are ignored. To return to normal mode or low-power mode, the TurnOnx bits of CTRL_REG5 must be set to zero.

Figure 5. Sleep-to-wake mode



4.4.1 Entering sleep-to-wake mode

Perform the following procedure to set up the sleep-to-wake function:

1. Configure the desired interrupt event (for example, wake-up).
2. Select the desired low power mode (ODR_{LP}) and data rate (ODR) in CTRL_REG1.
3. Enable the sleep-to-wake mode through CTRL_REG5 (TurnOn1 = TurnOn0 = 1).

Once an interrupt event occurs, the turn-on bits change to TurnOn1 = 0 and TurnOn0 = 1 and the system generates data at ODR. The user can reactivate the sleep-to-wake function by executing step 3 again.

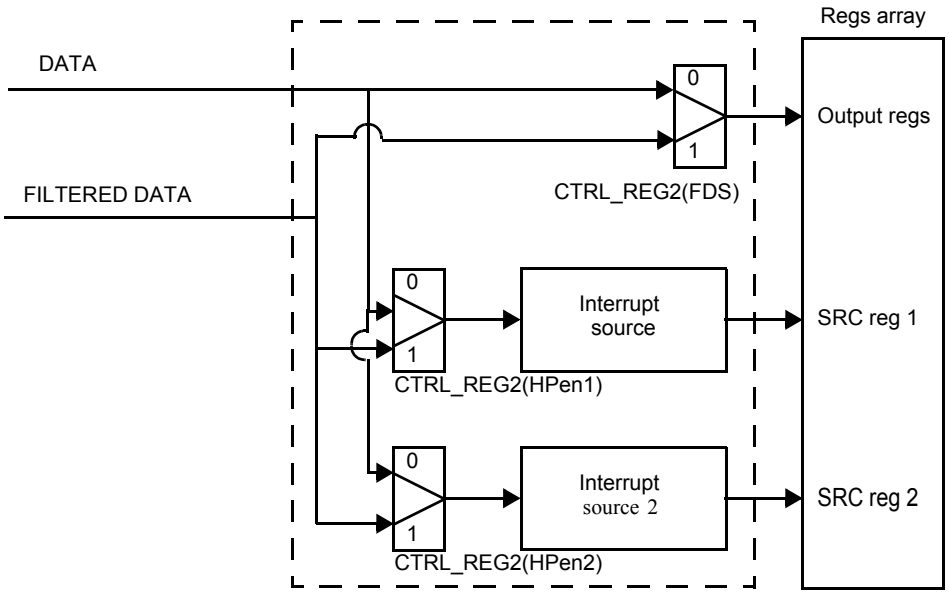
4.4.2 Exiting sleep-to-wake mode

To return to normal mode or to low-power mode, the user must disable the sleep-to-wake function by setting TurnOn1 = TurnOn0 = 0.

5 High-pass filter

The H3LIS331DL provides an embedded high-pass filtering capability to easily remove the DC component of the measured acceleration. As shown in Figure 6, it is possible to independently apply the filter to the output data and/or to the interrupt data through the FDS, HPen1, and HPen2 bits of the CTRL_REG2 register configuration. This means that it is possible, for example, to obtain filtered data while the interrupt generation works on unfiltered data.

Figure 6. Block diagram of high-pass filter connections



5.1 Filter configuration

As shown in Table 13, two operating modes are possible for the high-pass filter:

Table 13. High-pass filter mode configuration

HPM1	HPM0	High-pass filter mode
0	0	Normal mode (reset by reading HP_RESET_FILTER)
0	1	Reference mode
1	0	Normal mode (reset by reading HP_RESET_FILTER)
1	1	Not allowed

The bandwidth of the high-pass filter depends on the selected ODR and on the settings of the HPCFx bits of CTRL_REG2. The high-pass filter cut-off frequencies (f_t) are shown in Table 14.

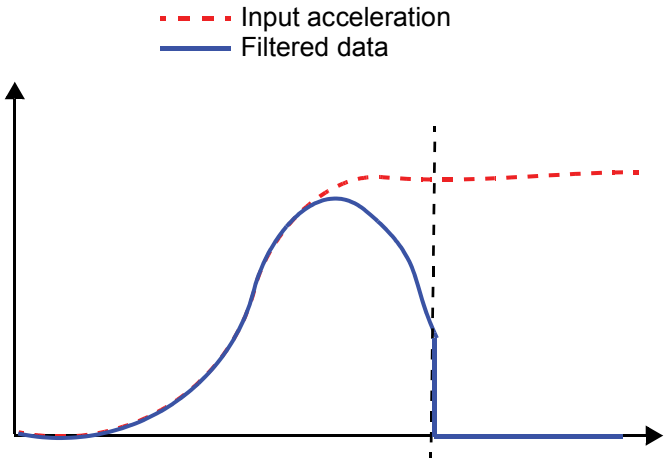
Table 14. High-pass filter cutoff frequency configuration

HPcoeff2,1	f_t [Hz] Data rate = 50 Hz	f_t [Hz] Data rate = 100 Hz	f_t [Hz] Data rate = 400 Hz	f_t [Hz] Data rate = 1000 Hz
00	1	2	8	20
01	0.5	1	4	10
10	0.25	0.5	2	5
11	0.125	0.25	1	2.5

5.1.1 Normal mode

In this configuration, the high-pass filter can be reset by reading the HP_FILTER_RESET register, instantly matching the output data to the input acceleration.

Figure 7. Reading HP_FILTER_RESET



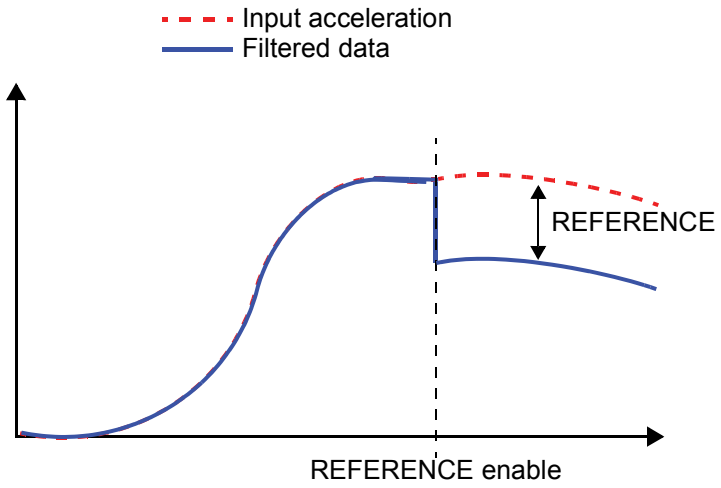
5.1.2 Reference mode

In reference mode configuration, the output data is calculated as the difference between the input acceleration and the content of the REFERENCE register. This register is in two's complement representation and the value of 1LSB of these 7-bit registers depends on the selected full scale (Table 14).

Table 15. Reference mode LSB value

Full scale	Reference mode LSB value
±100 *g*	0.8 *g*
±200 *g*	1.6 *g*
±400 *g*	3.1 *g*

Figure 8. Reference mode



6 Interrupt generation

The H3LIS331DL provides two interrupt signals and offers several possibilities for personalizing these signals. The registers involved in the interrupt generation behavior are CTRL_REG3, INT1_CFG, INT2_CFG, INT1_THS, INT2_THS, INT1_DURATION, and INT2_DURATION.

The AOI bit is used to configure the AND / OR interrupt mode, as shown in Table 16.

Table 16. Interrupt mode configuration

AOI	Interrupt mode
0	OR combination of interrupt events
1	AND combination of interrupt events

Whenever an interrupt condition is verified, the interrupt signal is generated and by reading the INT1_SRC and INT2_SRC registers, it is possible to detect which condition has occurred.

6.1 Duration

The content of the duration registers set the minimum duration of the interrupt event to be recognized. Duration steps and maximum values depend on the ODR chosen.

When in normal mode, duration time is measured in N/ODR, where N is the content of the duration register and ODR is 50, 100, 400, 1000 Hz.

Table 17. Duration LSB value in normal mode

ODR (Hz)	Duration LSB value (ms)
50	20
100	10
400	2.5
1000	1

When in low-power mode, duration time is measured in N/ODR_{LP}, where N is the content of the duration register and ODR_{LP} is 0.5, 1, 2, 5, 10 Hz.

Table 18. Duration LSB value in low-power mode

ODR (Hz)	Duration LSB value (s)
0.5	2
1	1
2	0.5
5	0.2
10	0.1

6.2 Threshold

Threshold registers define the reference accelerations used by the interrupt generation circuitry. The value of 1LSB of these 7-bit registers depends on the selected full scale (Table 19).

Table 19. Threshold LSB value

Full scale	Threshold LSB value
±100 *g*	~0.8 *g*
±200 *g*	~1.6 *g*
±400 *g*	~3.1 *g*

6.3 Interrupt modes

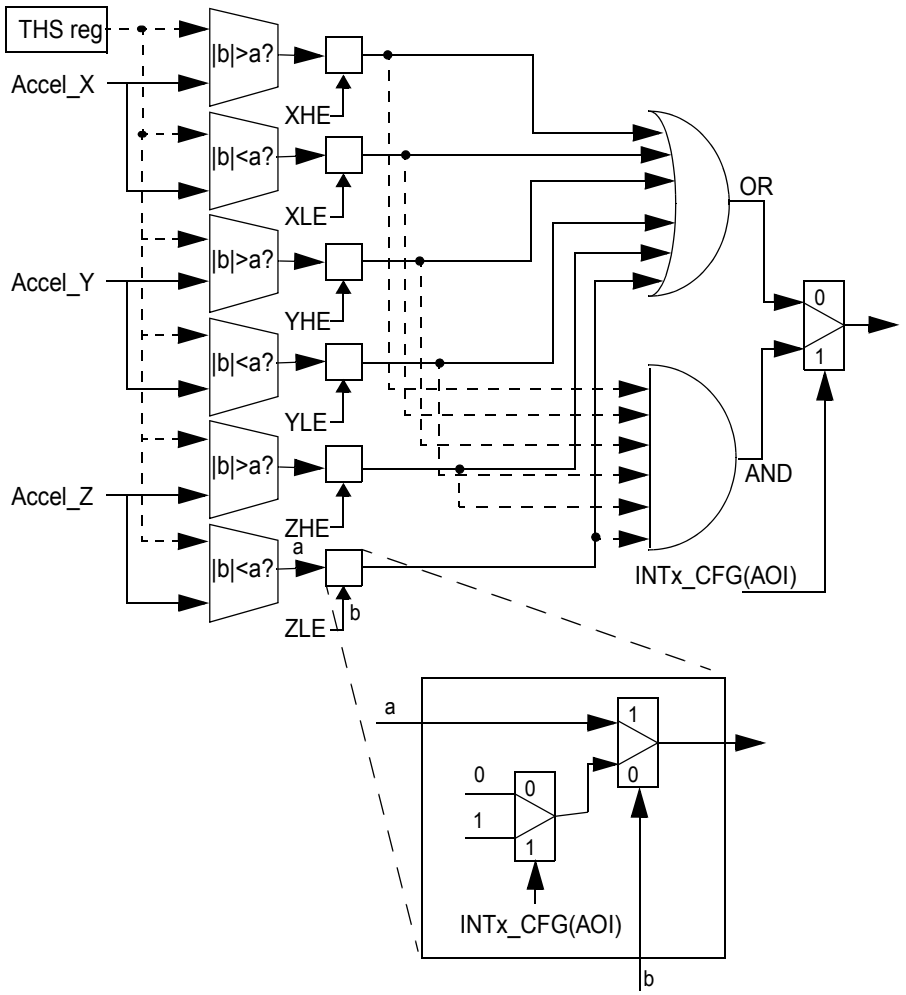
The interrupt generation block is represented in Figure 9.

The AND / OR interrupt mode is selected through the AOI bit in the INTx_CFG register. If the AOI bit is 0, signals coming from comparators are put in logical “OR”. Depending on the values written in the INT1_CFG register, every time the value of at least one of the enabled axes exceeds the threshold written in the module in INTx_THS, an interrupt is generated. The "OR" interrupt mode can be used to implement the wake-up feature; when an interrupt condition is verified, the interrupt signal is generated and by reading the INT1_SRC and INT2_SRC registers, it is possible to determine which axis has triggered the event. Otherwise, if the AOI bit is 1, signals coming from the comparators go into a “NAND” port. In this case, an interrupt signal is generated only if all the enabled axes exceed the threshold written in the INTx_THS register.

The LIRx bits of the CTRL_REG3 can be used to determine whether or not the interrupt request must be latched. If the LIRx bit is 0 (default value), the interrupt signal goes high when the interrupt condition is satisfied and immediately returns low if the interrupt condition is no longer verified. Otherwise, if the LIRx bit is 1, when an interrupt condition is satisfied, the interrupt signal remains high even if the condition returns to a noninterrupt status, until a read of the INTx_SRC register is performed.

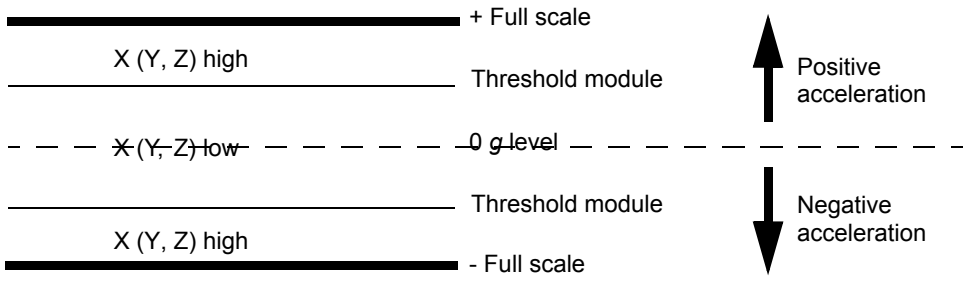
The Z HIE, ZLIE, YHIE, YLIE, XHIE and HLIE bits of the INTx_CFG register select on which axis the interrupt decision must be performed, and in which direction the threshold must be exceeded to generate the interrupt request.

Figure 9. **Interrupt generator**



The threshold module, which is used by the system to detect the interrupt events, is defined by the INTx_THS registers. The threshold value is expressed over 7 bits as an unsigned number and is symmetrical around the zero-*g* level. XH (YH, ZH) is true when the acceleration value of the X (Y, Z) channel is higher than INTx_THS. Similarly, XL, (YL, ZL) low is true when the acceleration value of the X (Y, Z) channel is lower than INTx_THS. Refer to Figure 10 for additional details.

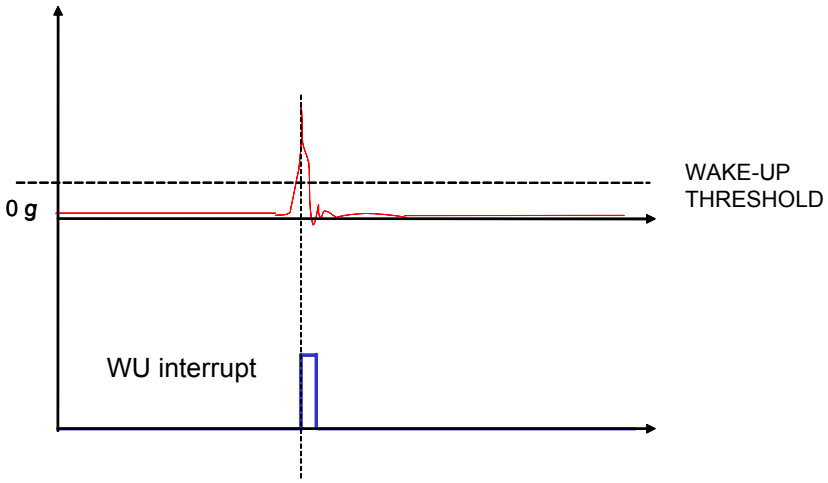
Figure 10. **X (Y, Z) high and low**



6.4 Inertial wake-up

The wake-up interrupt refers to a specific configuration of the INTx_CTRL registers that allow the interrupt generation when the acceleration on the configured axis exceeds a defined threshold (Figure 11).

Figure 11. Inertial wake-up interrupt



6.4.1 HP filter bypassed

This paragraph provides a basic algorithm that shows the practical use of the inertial wake-up feature. In particular, with the code below, the device is configured to recognize when the absolute acceleration along either the X or Y axis exceeds a preset threshold (20 *g* used in the example). The event that triggers the interrupt is latched and its occurrence is signaled through the usage of the INT1 pin.

1. Write 2Fh in CTRL_REG1 // Turn on the sensor and enable X, Y, and Z
   // ODR = 100 Hz
2. Write 00h in CTRL_REG2 // High-pass filter disabled
3. Write 04h in CTRL_REG3 // Latched interrupt active high on INT1 pin
4. Write 00h in CTRL_REG4 // FS = ±100 *g*
5. Write 00h in CTRL_REG5 // Sleep-to-wake disabled
6. Write19h in INT1_THS // Threshold = 20 *g*
7. Write 00h in INT1_DURATION // Duration = 0
8. Write 0Ah in INT1_CFG // Enable XH and YH interrupt generation
9. Poll INT1 pin; if INT1 = 0 then go to 8 // Poll INT1 pin waiting for the wake-up event
10. Read INT1_SRC // Return the event that has triggered the interrupt
11. (Wake-up event has occurred; insert your code here) // Event handling
12. Go to 8

6.4.2 Using the HP filter

The following code provides a basic routine showing the practical use of the inertial wake-up feature performed on high-pass filtered data. In particular, the device is configured to recognize when the high-frequency component of the acceleration applied along either the X, Y, or Z axis exceeds a preset threshold (20 *g* is used in the example). The event that triggers the interrupt is latched and its occurrence is signaled through the INT1 pin.

1.	Write 2Fh in CTRL_REG1	// Turn on the sensor, enable X, Y, and Z // ODR = 100 Hz
2.	Write 05h in CTRL_REG2	// High-pass filter enabled on data and interrupt1
3.	Write 04h in CTRL_REG3	// Latched interrupt active high on INT1 pin
4.	Write 00h in CTRL_REG4	// FS = ±100 *g*
5.	Write 00h in CTRL_REG5	// Sleep-to-wake disabled
6.	Write19h in INT1_THS	// Threshold = 20 *g*
7.	Write 00h in INT1_DURATION	// Duration = 0

8.	Read HP_FILTER_RESET	// Dummy read to force the HP filter to
		// actual acceleration value
		// (that is, set reference acceleration/tilt value)

9.	Write 2Ah in INT1_CFG	// Configure desired wake-up event
10.	Poll INT1 pin; if INT1 = 0 then go to 9	// Poll INT1 pin waiting for the wake-up event
11.	(Wake-up event has occurred; insert your code here)	// Event handling
12.	Read INT1_SRC	// Return the event that has triggered the // interrupt and clear interrupt
13.	(Insert your code here)	// Event handling
14.	Go to 9	

At step 8, a dummy read of the HP_FILTER_RESET register is performed to set the current/reference acceleration/tilt state against which the device performed the threshold comparison.

This read may be performed any time it is required to set the orientation/tilt of the device as a reference state without waiting for the filter to settle.

Revision history

Table 20. **Document revision history**

Date	Version	Changes
10-Nov-2022	1	Initial release

Contents

List of tables

List of figures