ISM330DLC: 3D accelerometer and 3D gyroscope with digital output for industrial applications

Introduction

This document is intended to provide usage information and application hints related to ST’s ISM330DLC iNEMO inertial module.

The ISM330DLC is a 3D digital accelerometer and 3D digital gyroscope system-in-package with a digital I^2C/SPI serial interface standard output, performing at 0.75 mA in combo High-Performance mode.

The device has a dynamic user-selectable full-scale acceleration range of ±2/±4/±8/±16 *g* and an angular rate range of ±125/±250/±500/±1000/±2000 dps.

It embeds smart features which simplify and optimize the application design and allow the usage of complex motion-sensing information also in power-constrained applications.

The ISM330DLC can be configured to generate interrupt signals by using hardware recognition of free-fall events, 6D orientation, tap and double-tap sensing, activity or inactivity, and wake-up events.

The availability of different connection modes to external sensors allows implementing additional functionalities such as a sensor hub, auxiliary SPI, etc.

The integrated smart first-in first-out (FIFO) buffer of up to 4 kbyte size allows dynamic batching of significant data (i.e. accelerometer and gyroscope data, external sensor data, timestamp and temperature data).

The ISM330DLC is available in a small plastic land grid array package (LGA-14L) and it is guaranteed to operate over an extended temperature range from -40 °C to +85 °C.

1 Pin description

Figure 1. Pin connections

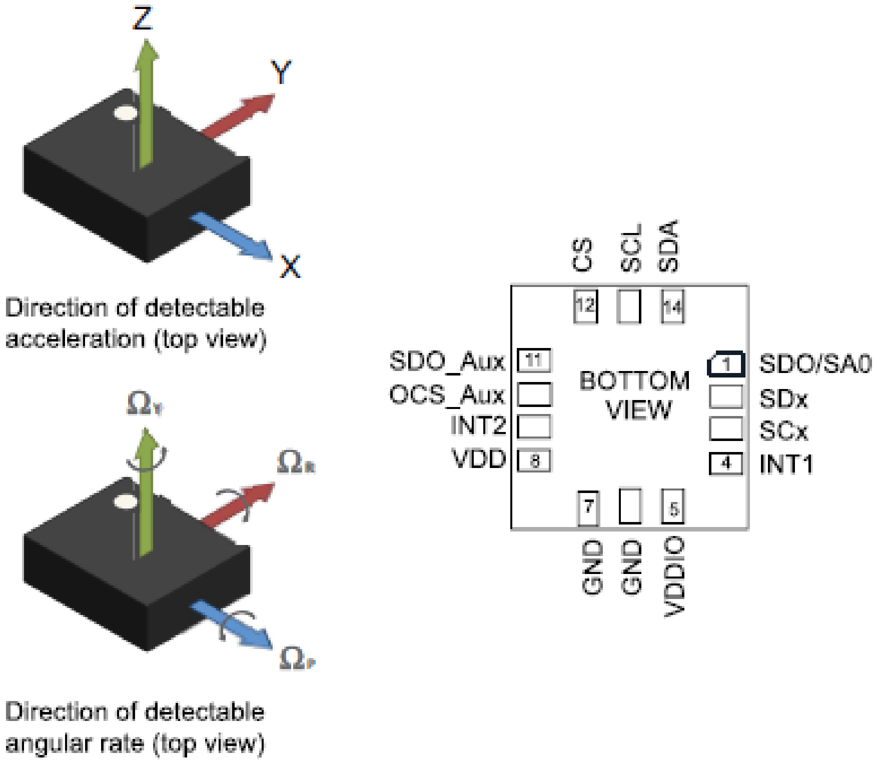


Table 1. Pin status

Pin #	Name	Mode 1 function	Mode 2 function	Mode 3/4 function	Pin status Mode 1	Pin status Mode 2	Pin status Mode 3/4
1	SDO SA0	SPI 4-wire interface serial data output (SDO) I^{2}C least significant bit of the device address (SA0)	SPI 4-wire interface serial data output (SDO) I^{2}C least significant bit of the device address (SA0)	SPI 4-wire interface serial data output (SDO) I^{2}C least significant bit of the device address (SA0)	Default: input without pull-up. Pull-up is enabled if bit SIM = 1 (SPI 3-wire) in reg 12h.	Default: input without pull-up. Pull-up is enabled if bit SIM = 1 (SPI 3-wire) in reg 12h.	Default: input without pull-up. Pull-up is enabled if bit SIM = 1 (SPI 3-wire) in reg 12h.
2	SDx	Connect to VDDIO or GND	I^{2}C serial data master (MSDA)	Auxiliary SPI 3/4-wire interface serial data input (SDI) and SPI 3-wire serial data output (SDO)	Default: input without pull-up. Pull-up is enabled if bit PULL_UP_EN = 1 in reg 1Ah.	Default: input without pull-up. Pull-up is enabled if bit PULL_UP_EN = 1 in reg 1Ah.	Default: input without pull-up. Pull-up is enabled if bit PULL_UP_EN = 1 in|reg 1Ah.
3	SCx	Connect to VDDIO or GND	I^{2}C serial clock master (MSCL)	Auxiliary SPI 3/4-wire interface serial port clock (SPC_Aux)	Default: input without pull-up. Pull-up is enabled if bit PULL_UP_EN = 1 in reg 1Ah.	Default: input without pull-up. Pull-up is enabled if bit PULL_UP_EN = 1 in reg 1Ah.	Default: input without pull-up. Pull-up is enabled if bit PULL_UP_EN = 1 in reg 1Ah.
4	INT1	Programmable interrupt 1	Programmable interrupt 1	Programmable interrupt 1	Default: output forced to ground	Default: output forced to ground	Default: output forced to ground
5	Vdd_IO	Power supply for I/O pins	Power supply for I/O pins	Power supply for I/O pins			
6	GND	0 V supply	0 V supply	0 V supply			
7	GND	0 V supply	0 V supply	0 V supply			
8	Vdd	Power supply	Power supply	Power supply			
9	INT2	Programmable interrupt 2 (INT2) / Data enabled (DEN)	Programmable interrupt 2 (INT2) / Data enabled (DEN) / I^{2}C master external synchronization signal (MDRDY)	Programmable interrupt 2 (INT2) / Data enabled (DEN)	Default: output forced to ground	Default: output forced to ground	Default: output forced to ground
10	OCS	Leave unconnected	Leave unconnected	Auxiliary SPI 3/4-wire interface enable	Default: input with pull-up. (See note below to disable pull-up)	Default: input with pull-up. (See note below to disable pull-up)	Input without pull-up
11	SDO_Aux	Connected to VDDIO or leave unconnected	Connected to VDDIO or leave unconnected	Auxiliary SPI 3/4-wire interface: leave unconnected / Auxiliary SPI 4-wire interface: serial data output (SDO_Aux)	Default: input with pull-up. (See note below to disable pull-up)	Default: input with pull-up. (See note below to disable pull-up)	Default: input without pull-up. Pull-up is enabled if bit SIM_OIS = 1 (Aux_SPI 3-wire) in reg 70h.
12	CS	I^{2}C/SPI mode selection (1:SPI idle mode / I^{2}C communication enabled; 0: SPI communication mode / I^{2}C disabled)	I^{2}C/SPI mode selection (1:SPI idle mode / I^{2}C communication enabled; 0: SPI communication mode / I^{2}C disabled)	I^{2}C/SPI mode selection (1:SPI idle mode / I^{2}C communication enabled; 0: SPI communication mode / I^{2}C disabled	Default: input with pull-up. Pull-up is disabled if bit I2C_disable = 1 in reg 13h.	Default: input with pull-up. Pull-up is disabled if bit I2C_disable = 1 in reg 13h.	Default: input with pull-up. Pull-up is disabled if bit I2C_disable = 1 in reg 13h.
13	SCL	I^{2}C serial clock (SCL) / SPI serial port clock (SPC)	I^{2}C serial clock (SCL) / SPI serial port clock (SPC)	I^{2}C serial clock (SCL) / SPI serial port clock (SPC)	Input without pull-up	Input without pull-up	Input without pull-up
14	SDA	I^{2}C serial data (SDA) / SPI serial data input (SDI) / 3-wire interface serial data output (SDO)	I^{2}C serial data (SDA) / SPI serial data input (SDI) / 3-wire interface serial data output (SDO)	I^{2}C serial data (SDA) / SPI serial data input (SDI) / 3-wire interface serial data output (SDO)	Input without pull-up	Input without pull-up	Input without pull-up

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

Note: Procedure to disable pull-up on pins 10-11

1. *From primary I^{2}C/SPI interface: write 80h in register at address 00h*
2. *From primary I^{2}C/SPI interface: write 01h in register at address 05h (disable the pull-up on pins 10 & 11)*
3. *From primary I^{2}C/SPI interface: write 00h in register at address 00h*

2 Registers

Table 2. Registers

Register name	Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
FUNC_CFG_ACCESS	01h	FUNC_CFG_EN	0	0	0	0	0	0	0
SENSOR_SYNC_TIME_FRAME	04h	0	0	0	0	TPH_3	TPH_2	TPH_1	TPH_0
SENSOR_SYNC_RES_RATIO	05h	0	0	0	0	0	0	RR_1	RR_0
FIFO_CTRL1	06h	FTH_7	FTH_6	FTH_5	FTH_4	FTH_3	FTH_2	FTH_1	FTH_0
FIFO_CTRL2	07h	FIFO_TIMER_EN	0	0	0	FIFO_TEMP_EN	FTH_10	FTH_9	FTH_8
FIFO_CTRL3	08h	0	0	DEC_FIFO_GYRO2	DEC_FIFO_GYRO1	DEC_FIFO_GYRO0	DEC_FIFO _XL2	DEC_FIFO _XL1	DEC_FIFO _XL0
FIFO_CTRL4	09h	STOP_ON_FTH	ONLY_HIGH _DATA	DEC_DS4 _FIFO2	DEC_DS4 _FIFO1	DEC_DS4 _FIFO0	DEC_DS3 _FIFO2	DEC_DS3 _FIFO1	DEC_DS3 _FIFO0
FIFO_CTRL5	0Ah	0	ODR_FIFO _3	ODR_FIFO_2	ODR_FIFO_1	ODR_FIFO_0	FIFO_MODE_2	FIFO_MODE_1	FIFO_MODE_0
DRDY_PULSE_CFG	0Bh	DRDY_PULSED	0	0	0	0	0	0	0
INT1_CTRL	0Dh	0	0	INT1_FULL_FLAG	INT1_FIFO_OVR	INT1_FTH	INT1_BOOT	INT1_DRDY _G	INT1_DRDY _XL
INT2_CTRL	0Eh	0	0	INT2_FULL_FLAG	INT2_FIFO_OVR	INT2_FTH	INT2_DRDY _TEMP	INT2_DRDY _G	INT2_DRDY _XL
WHO_AM_I	0Fh	0	1	1	0	1	0	1	0
CTRL1_XL	10h	ODR_XL3	ODR_XL2	ODR_XL1	ODR_XL0	FS_XL1	FS_XL0	LPF1_BW _SEL	BW0_XL
CTRL2_G	11h	ODR_G3	ODR_G2	ODR_G1	ODR_G0	FS_G1	FS_G0	FS_125	0
CTRL3_C	12h	BOOT	BDU	H_LACTIVE	PP_OD	SIM	IF_INC	BLE	SW_RESET
CTRL4_C	13h	DEN_XL_EN	SLEEP	INT2_on_INT1	DEN_DRDY _INT1	DRDY _MASK	I2C_disable	LPF1_SEL _G	0
CTRL5_C	14h	ROUNDING2	ROUNDING1	ROUNDING0	DEN_LH	ST1_G	ST0_G	ST1_XL	ST0_XL
CTRL6_C	15h	TRIG_EN	LVL1_EN	LVL2_EN	XL_HM_MODE	USR_OFF_W	0	FTYPE_1	FTYPE_0
CTRL7_G	16h	G_HM_MODE	HP_G_EN	HPM1_G	HPM0_G	0	ROUNDING_STATUS	0	0
CTRL8_XL	17h	LPF2_XL_EN	HPCF_XL1	HPCF_XL0	HP_REF_MODE	INPUT_COMPOSITE	HP_SLOPE_XL_EN	0	LOW_PASS_ON_6D
CTRL9_XL	18h	DEN_X	DEN_Y	DEN_Z	DEN_XL_G	0	SOFT_EN	0	0
CTRL10_C	19h	0	0	TIMER_EN	0	TILT_EN	FUNC_EN	0	0
MASTER_CONFIG	1Ah	DRDY_ON _INT1	DATA_VALID_SEL_FIFO	0	START _CONFIG	PULL_UP_EN	PASS _THROUGH_MODE	IRON_EN	MASTER _ON
WAKE_UP_SRC	1Bh	0	0	FF_IA	SLEEP _STATE_IA	WU_IA	X_WU	Y_WU	Z_WU
TAP_SRC	1Ch	0	TAP_IA	SINGLE_TAP	DOUBLE_TAP	TAP_SIGN	X_TAP	Y_TAP	Z_TAP
D6D_SRC	1Dh	DEN_DRDY	D6D_IA	ZH	ZL	YH	YL	XH	XL

Register name	Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
STATUS_REG / STATUS_SPIAux	1Eh	0	0	0	0	0	TDA / GYRO_ SETTLING	GDA / GDA	XLDA / XLDA
OUT_TEMP_L	20h	Temp7	Temp6	Temp5	Temp4	Temp3	Temp2	Temp1	Temp0
OUT_TEMP_H	21h	Temp15	Temp14	Temp13	Temp12	Temp11	Temp10	Temp9	Temp8
OUTX_L_G	22h	D7	D6	D5	D4	D3	D2	D1	D0
OUTX_H_G	23h	D15	D14	D13	D12	D11	D10	D9	D8
OUTY_L_G	24h	D7	D6	D5	D4	D3	D2	D1	D0
OUTY_H_G	25h	D15	D14	D13	D12	D11	D10	D9	D8
OUTZ_L_G	26h	D7	D6	D5	D4	D3	D2	D1	D0
OUTZ_H_G	27h	D15	D14	D13	D12	D11	D10	D9	D8
OUTX_L_XL	28h	D7	D6	D5	D4	D3	D2	D1	D0
OUTX_H_XL	29h	D15	D14	D13	D12	D11	D10	D9	D8
OUTY_L_XL	2Ah	D7	D6	D5	D4	D3	D2	D1	D0
OUTY_H_XL	2Bh	D15	D14	D13	D12	D11	D10	D9	D8
OUTZ_L_XL	2Ch	D7	D6	D5	D4	D3	D2	D1	D0
OUTZ_H_XL	2Dh	D15	D14	D13	D12	D11	D10	D9	D8
SENSORHUB1_REG	2Eh	SHub1_7	SHub1_6	SHub1_5	SHub1_4	SHub1_3	SHub1_2	SHub1_1	SHub1_0
SENSORHUB2_REG	2Fh	SHub2_7	SHub2_6	SHub2_5	SHub2_4	SHub2_3	SHub2_2	SHub2_1	SHub2_0
SENSORHUB3_REG	30h	SHub3_7	SHub3_6	SHub3_5	SHub3_4	SHub3_3	SHub3_2	SHub3_1	SHub3_0
SENSORHUB4_REG	31h	SHub4_7	SHub4_6	SHub4_5	SHub4_4	SHub4_3	SHub4_2	SHub4_1	SHub4_0
SENSORHUB5_REG	32h	SHub5_7	SHub5_6	SHub5_5	SHub5_4	SHub5_3	SHub5_2	SHub5_1	SHub5_0
SENSORHUB6_REG	33h	SHub6_7	SHub6_6	SHub6_5	SHub6_4	SHub6_3	SHub6_2	SHub6_1	SHub6_0
SENSORHUB7_REG	34h	SHub7_7	SHub7_6	SHub7_5	SHub7_4	SHub7_3	SHub7_2	SHub7_1	SHub7_0
SENSORHUB8_REG	35h	SHub8_7	SHub8_6	SHub8_5	SHub8_4	SHub8_3	SHub8_2	SHub8_1	SHub8_0
SENSORHUB9_REG	36h	SHub9_7	SHub9_6	SHub9_5	SHub9_4	SHub9_3	SHub9_2	SHub9_1	SHub9_0
SENSORHUB10_REG	37h	SHub10_7	SHub10_6	SHub10_5	SHub10_4	SHub10_3	SHub10_2	SHub10_1	SHub10_0
SENSORHUB11_REG	38h	SHub11_7	SHub11_6	SHub11_5	SHub11_4	SHub11_3	SHub11_2	SHub11_1	SHub11_0
SENSORHUB12_REG	39h	SHub12_7	SHub12_6	SHub12_5	SHub12_4	SHub12_3	SHub12_2	SHub12_1	SHub12_0
FIFO_STATUS1	3Ah	DIFF_FIFO_7	DIFF_FIFO_6	DIFF_FIFO_5	DIFF_FIFO_4	DIFF_FIFO_3	DIFF_FIFO_2	DIFF_FIFO_1	DIFF_FIFO_0
FIFO_STATUS2	3Bh	WaterM	OVER_RUN	FIFO_FULL _SMART	FIFO _EMPTY	0	DIFF_FIFO_10	DIFF_FIFO_9	DIFF_FIFO_8
FIFO_STATUS3	3Ch	FIFO_ PATTERN_7	FIFO_ PATTERN_6	FIFO_ PATTERN_5	FIFO_ PATTERN_4	FIFO_ PATTERN_3	FIFO_ PATTERN_2	FIFO_ PATTERN_1	FIFO_ PATTERN_0
FIFO_STATUS4	3Dh	0	0	0	0	0	0	FIFO_ PATTERN_9	FIFO_ PATTERN_8
FIFO_DATA_OUT_L	3Eh	DATA_OUT _FIFO_L_7	DATA_OUT _FIFO_L_6	DATA_OUT _FIFO_L_5	DATA_OUT _FIFO_L_4	DATA_OUT _FIFO_L_3	DATA_OUT _FIFO_L_2	DATA_OUT _FIFO_L_1	DATA_OUT _FIFO_L_0

Register name	Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
FIFO_DATA_OUT_H	3Fh	DATA_OUT _FIFO_H_7	DATA_OUT _FIFO_H_6	DATA_OUT _FIFO_H_5	DATA_OUT _FIFO_H_4	DATA_OUT _FIFO_H_3	DATA_OUT _FIFO_H_2	DATA_OUT _FIFO_H_1	DATA_OUT _FIFO_H_0
TIMESTAMP0_REG	40h	TIMESTAMP0_7	TIMESTAMP0_6	TIMESTAMP0_5	TIMESTAMP0_4	TIMESTAMP0_3	TIMESTAMP0_2	TIMESTAMP0_1	TIMESTAMP0_0
TIMESTAMP1_REG	41h	TIMESTAMP1_7	TIMESTAMP1_6	TIMESTAMP1_5	TIMESTAMP1_4	TIMESTAMP1_3	TIMESTAMP1_2	TIMESTAMP1_1	TIMESTAMP1_0
TIMESTAMP2_REG	42h	TIMESTAMP2_7	TIMESTAMP2_6	TIMESTAMP2_5	TIMESTAMP2_4	TIMESTAMP2_3	TIMESTAMP2_2	TIMESTAMP2_1	TIMESTAMP2_0
SENSORHUB13_REG	4Dh	SHub13_7	SHub13_6	SHub13_5	SHub13_4	SHub13_3	SHub13_2	SHub13_1	SHub13_0
SENSORHUB14_REG	4Eh	SHub14_7	SHub14_6	SHub14_5	SHub14_4	SHub14_3	SHub14_2	SHub14_1	SHub14_0
SENSORHUB15_REG	4Fh	SHub15_7	SHub15_6	SHub15_5	SHub15_4	SHub15_3	SHub15_2	SHub15_1	SHub15_0
SENSORHUB16_REG	50h	SHub16_7	SHub16_6	SHub16_5	SHub16_4	SHub16_3	SHub16_2	SHub16_1	SHub16_0
SENSORHUB17_REG	51h	SHub17_7	SHub17_6	SHub17_5	SHub17_4	SHub17_3	SHub17_2	SHub17_1	SHub17_0
SENSORHUB18_REG	52h	SHub18_7	SHub18_6	SHub18_5	SHub18_4	SHub18_3	SHub18_2	SHub18_1	SHub18_0
FUNC_SRC1	53h	0	0	TILT_IA	0	0	HI_FAIL	SI_END_OP	SENSORHUB_ END_OP
FUNC_SRC2	54h	0	SLAVE3_NACK	SLAVE2_NACK	SLAVE1_NACK	SLAVE0_NACK	0	0	0
TAP_CFG	58h	INTERRUPTS_ ENABLE	INACT_EN1	INACT_EN0	SLOPE_FDS	TAP_X_EN	TAP_Y_EN	TAP_Z_EN	LIR
TAP_THS_6D	59h	D4D_EN	SIXD_THS1	SIXD_THS0	TAP_THS4	TAP_THS3	TAP_THS2	TAP_THS1	TAP_THS0
INT_DUR2	5Ah	DUR3	DUR2	DUR1	DUR0	QUIET1	QUIET0	SHOCK1	SHOCK0
WAKE_UP_THS	5Bh	SINGLE_ DOUBLE_TAP	0	WK_THS5	WK_THS4	WK_THS3	WK_THS2	WK_THS1	WK_THS0
WAKE_UP_DUR	5Ch	FF_DUR5	WAKE_DUR1	WAKE_DUR0	TIMER_HR	SLEEP _DUR3	SLEEP _DUR2	SLEEP _DUR1	SLEEP _DUR0
FREE_FALL	5Dh	FF_DUR4	FF_DUR3	FF_DUR2	FF_DUR1	FF_DUR0	FF_THS2	FF_THS1	FF_THS0
MD1_CFG	5Eh	INT1_INACT _STATE	INT1_SINGLE_TAP	INT1_WU	INT1_FF	INT1_ DOUBLE_TAP	INT1_6D	INT1_TILT	INT1 _TIMER
MD2_CFG	5Fh	INT2_INACT _STATE	INT2_SINGLE_TAP	INT2_WU	INT2_FF	INT2_ DOUBLE_TAP	INT2_6D	INT2_TILT	INT2 _IRON
MASTER_CMD_CODE	60h	MASTER_ CMD_CODE7	MASTER_ CMD_CODE6	MASTER_ CMD_CODE5	MASTER_ CMD_CODE4	MASTER_ CMD_CODE3	MASTER_ CMD_CODE2	MASTER_ CMD_CODE1	MASTER_ CMD_CODE0
SENS_SYNC_SPI_ ERROR_CODE	61h	ERROR_CODE7	ERROR_CODE6	ERROR_CODE5	ERROR_CODE4	ERROR_CODE3	ERROR_CODE2	ERROR_CODE1	ERROR_CODE0
OUT_MAG_RAW_X_L	66h	D7	D6	D5	D4	D3	D2	D1	D0
OUT_MAG_RAW_X_H	67h	D15	D14	D13	D12	D11	D10	D9	D8
OUT_MAG_RAW_Y_L	68h	D7	D6	D5	D4	D3	D2	D1	D0
OUT_MAG_RAW_Y_H	69h	D15	D14	D13	D12	D11	D10	D9	D8
OUT_MAG_RAW_Z_L	6Ah	D7	D6	D5	D4	D3	D2	D1	D0
OUT_MAG_RAW_Z_H	6Bh	D15	D14	D13	D12	D11	D10	D9	D8

Register name	Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
INT_OIS	6Fh	INT2_DRDY_OIS	LVL2_OIS	-	-	-	-	-	-
CTRL1_OIS	70h	BLE_OIS	LVL1_OIS	SIM_OIS	MODE4_EN	FS1_G_OIS	FS0_G_OIS	FS_125_OIS	OIS_EN_SPI2
CTRL2_OIS	71h	0	0	HPM1_OIS	HPM0_OIS	0	FTYPE_1_OIS	FTYPE_0_OIS	HP_EN_OIS
CTRL3_OIS	72h	DEN_LH_OIS	FS1_XL_OIS	FS0_XL_OIS	FILTER_XL_ CONF_OIS_1	FILTER_XL_ CONF_OIS_0	ST1_OIS	ST0_OIS	ST_OIS_ CLAMPDIS
X_OFS_USR	73h	X_OFS_USR_7	X_OFS_USR_6	X_OFS_USR_5	X_OFS_USR_4	X_OFS_USR_3	X_OFS_USR_2	X_OFS_USR_1	X_OFS_USR_0
Y_OFS_USR	74h	Y_OFS_USR_7	Y_OFS_USR_6	Y_OFS_USR_5	Y_OFS_USR_4	Y_OFS_USR_3	Y_OFS_USR_2	Y_OFS_USR_1	Y_OFS_USR_0
Z_OFS_USR	75h	Z_OFS_USR_7	Z_OFS_USR_6	Z_OFS_USR_5	Z_OFS_USR_4	Z_OFS_USR_3	Z_OFS_USR_2	Z_OFS_USR_1	Z_OFS_USR_0

2.1 Embedded functions registers

The list of the registers for embedded functions available in the device is given in Table 3. Embedded functions registers.

The embedded functions registers are accessible when FUNC_CFG_EN is set to 1 in FUNC_CFG_ACCESS (01h).

Note: *All modifications to the content of the embedded functions registers have to be performed with both the accelerometer and the gyroscope sensor in Power-Down mode.*

Table 3. Embedded functions registers

Register name	Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SLV0_ADD	02h	Slave0_add6	Slave0_add5	Slave0_add4	Slave0_add3	Slave0_add2	Slave0_add1	Slave0_add0	rw_0
SLV0_SUBADD	03h	Slave0_reg7	Slave0_reg6	Slave0_reg5	Slave0_reg4	Slave0_reg3	Slave0_reg2	Slave0_reg1	Slave0_reg0
SLAVE0_CONFIG	04h	Slave0_rate1	Slave0_rate0	Aux_sens_on1	Aux_sens_on0	Src_mode	Slave0 _numop2	Slave0 _numop1	Slave0 _numop0
SLV1_ADD	05h	Slave1_add6	Slave1_add5	Slave1_add4	Slave1_add3	Slave1_add2	Slave1_add1	Slave1_add0	r_1
SLV1_SUBADD	06h	Slave1_reg7	Slave1_reg6	Slave1_reg5	Slave1_reg4	Slave1_reg3	Slave1_reg2	Slave1_reg1	Slave1_reg0
SLAVE1_CONFIG	07h	Slave1_rate1	Slave1_rate0	write_once	0	0	Slave1 _numop2	Slave1 _numop1	Slave1 _numop0
SLV2_ADD	08h	Slave2_add6	Slave2_add5	Slave2_add4	Slave2_add3	Slave2_add2	Slave2_add1	Slave2_add0	r_2
SLV2_SUBADD	09h	Slave2_reg7	Slave2_reg6	Slave2_reg5	Slave2_reg4	Slave2_reg3	Slave2_reg2	Slave2_reg1	Slave2_reg0
SLAVE2_CONFIG	0Ah	Slave2_rate1	Slave2_rate0	0	0	0	Slave2 _numop2	Slave2 _numop1	Slave2 _numop0
SLV3_ADD	0Bh	Slave3_add6	Slave3_add5	Slave3_add4	Slave3_add3	Slave3_add2	Slave3_add1	Slave3_add0	r_3
SLV3_SUBADD	0Ch	Slave3_reg7	Slave3_reg6	Slave3_reg5	Slave3_reg4	Slave3_reg3	Slave3_reg2	Slave3_reg1	Slave3_reg0
SLAVE3_CONFIG	0Dh	Slave3_rate1	Slave3_rate0	0	0	0	Slave3 _numop2	Slave3 _numop1	Slave3 _numop0
DATAWRITE_SRC _MODE_SUB_SLV0	0Eh	Slave_dataw7	Slave_dataw6	Slave_dataw5	Slave_dataw4	Slave_dataw3	Slave_dataw2	Slave_dataw1	Slave_dataw0
MAG_SI_XX	24h	MAG_SI _XX_7	MAG_SI _XX_6	MAG_SI _XX_5	MAG_SI _XX_4	MAG_SI _XX_3	MAG_SI _XX_2	MAG_SI _XX_1	MAG_SI _XX_0
MAG_SI_XY	25h	MAG_SI _XY_7	MAG_SI _XY_6	MAG_SI _XY_5	MAG_SI _XY_4	MAG_SI _XY_3	MAG_SI _XY_2	MAG_SI _XY_1	MAG_SI _XY_0
MAG_SI_XZ	26h	MAG_SI _XZ_7	MAG_SI _XZ_6	MAG_SI _XZ_5	MAG_SI _XZ_4	MAG_SI _XZ_3	MAG_SI _XZ_2	MAG_SI _XZ_1	MAG_SI _XZ_0
MAG_SI_YX	27h	MAG_SI _YX_7	MAG_SI _YX_6	MAG_SI _YX_5	MAG_SI _YX_4	MAG_SI _YX_3	MAG_SI _YX_2	MAG_SI _YX_1	MAG_SI _YX_0
MAG_SI_YY	28h	MAG_SI _YY_7	MAG_SI _YY_6	MAG_SI _YY_5	MAG_SI _YY_4	MAG_SI _YY_3	MAG_SI _YY_2	MAG_SI _YY_1	MAG_SI _YY_0
MAG_SI_YZ	29h	MAG_SI _YZ_7	MAG_SI _YZ_6	MAG_SI _YZ_5	MAG_SI _YZ_4	MAG_SI _YZ_3	MAG_SI _YZ_2	MAG_SI _YZ_1	MAG_SI _YZ_0
MAG_SI_ZX	2Ah	MAG_SI _ZX_7	MAG_SI _ZX_6	MAG_SI _ZX_5	MAG_SI _ZX_4	MAG_SI _ZX_3	MAG_SI _ZX_2	MAG_SI _ZX_1	MAG_SI _ZX_0
MAG_SI_ZY	2Bh	MAG_SI _ZY_7	MAG_SI _ZY_6	MAG_SI _ZY_5	MAG_SI _ZY_4	MAG_SI _ZY_3	MAG_SI _ZY_2	MAG_SI _ZY_1	MAG_SI _ZY_0
MAG_SI_ZZ	2Ch	MAG_SI _ZZ_7	MAG_SI _ZZ_6	MAG_SI _ZZ_5	MAG_SI _ZZ_4	MAG_SI _ZZ_3	MAG_SI _ZZ_2	MAG_SI _ZZ_1	MAG_SI _ZZ_0
MAG_OFFX_L	2Dh	MAG_OFFX _L_7	MAG_OFFX _L_6	MAG_OFFX _L_5	MAG_OFFX _L_4	MAG_OFFX _L_3	MAG_OFFX _L_2	MAG_OFFX _L_1	MAG_OFFX _L_0

Register name	Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
MAG_OFFX_H	2Eh	MAG_OFFX _H_7	MAG_OFFX _H_6	MAG_OFFX _H_5	MAG_OFFX _H_4	MAG_OFFX _H_3	MAG_OFFX _H_2	MAG_OFFX _H_1	MAG_OFFX _H_0
MAG_OFFY_L	2Fh	MAG_OFFY _L_7	MAG_OFFY _L_6	MAG_OFFY _L_5	MAG_OFFY _L_4	MAG_OFFY _L_3	MAG_OFFY _L_2	MAG_OFFY _L_1	MAG_OFFY _L_0
MAG_OFFY_H	30h	MAG_OFFY _H_7	MAG_OFFY _H_6	MAG_OFFY _H_5	MAG_OFFY _H_4	MAG_OFFY _H_3	MAG_OFFY _H_2	MAG_OFFY _H_1	MAG_OFFY _H_0
MAG_OFFZ_L	31h	MAG_OFFZ _L_7	MAG_OFFZ _L_6	MAG_OFFZ _L_5	MAG_OFFZ _L_4	MAG_OFFZ _L_3	MAG_OFFZ _L_2	MAG_OFFZ _L_1	MAG_OFFZ _L_0
MAG_OFFZ_H	32h	MAG_OFFZ _H_7	MAG_OFFZ _H_6	MAG_OFFZ _H_5	MAG_OFFZ _H_4	MAG_OFFZ _H_3	MAG_OFFZ _H_2	MAG_OFFZ _H_1	MAG_OFFZ _H_0

3 Operating modes

The ISM330DLC provides three possible operating configurations:

- only accelerometer active and gyroscope in Power-Down;
- only gyroscope active and accelerometer in Power-Down;
- both accelerometer and gyroscope active with independent ODR.

The device offers a wide VDD voltage range from 1.71 V to 3.6 V and a VDDIO range from 1.62 V to 3.6 V. In order to avoid potential conflicts, during the power-on sequence it is recommended to set the lines connected to the the device IO pins to high-impedance state on the host side. Furthermore, to guarantee proper power-off of the device it is recommended to maintain the duration of the VDD line to GND for at least 100 μs.

After the power supply is applied, the ISM330DLC performs a 15 ms boot procedure to load the trimming parameters. After the boot is completed, both the accelerometer and the gyroscope are automatically configured in Power-Down mode.

The accelerometer and the gyroscope can be independently configured in four different power modes: Power-Down, Low-Power, Normal and High-Performance mode. They are allowed to have different data rates without any limit. The gyroscope sensor can also be set in Sleep mode to reduce its power consumption.

When both the accelerometer and gyroscope are on, the accelerometer is synchronized with the gyroscope, and the data rates of the two sensors are integer multiples of each other.

Referring to the ISM330DLC datasheet, the output data rate (ODR_XL) bits of the CTRL1_XL register and the High-Performance disable (XL_HM_MODE) bit of the CTRL6_C register are used to select the power mode and the output data rate of the accelerometer (Table 4. Accelerometer ODR and power mode selection).

Table 4. Accelerometer ODR and power mode selection

ODR_XL [3:0]	ODR [Hz] when XL_HM_MODE = 1	ODR [Hz] when XL_HM_MODE = 0
0000	Power Down	Power Down
1011	1.6 Hz (Low Power only)	12.5 Hz (High Performance)
0001	12.5 Hz (Low Power)	12.5 Hz (High Performance)
0010	26 Hz (Low Power)	26 Hz (High Performance)
0011	52 Hz (Low Power)	52 Hz (High Performance)
0100	104 Hz (Normal mode)	104 Hz (High Performance)
0101	208 Hz (Normal mode)	208 Hz (High Performance)
0110	416 Hz (High Performance)	416 Hz (High Performance)
0111	833 Hz (High Performance)	833 Hz (High Performance)
1000	1.66 kHz (High Performance)	1.66 kHz (High Performance)
1001	3.33 kHz (High Performance)	3.33 kHz (High Performance)
1010	6.66 kHz (High Performance)	6.66 kHz (High Performance)

The output data rate (ODR_G) bits of the CTRL2_G register and the High-Performance disable (G_HM_MODE) bit of the CTRL7_G register are used to select the power mode and output data rate of the gyroscope sensor (Table 5. Gyroscope ODR and power mode selection).

Table 5. **Gyroscope ODR and power mode selection**

ODR_G [3:0]	ODR [Hz] when G_HM_MODE = 1	ODR [Hz] when G_HM_MODE = 0
0000	Power Down	Power Down
0001	12.5 Hz (Low Power)	12.5 Hz (High Performance)
0010	26 Hz (Low Power)	26 Hz (High Performance)
0011	52 Hz (Low Power)	52 Hz (High Performance)
0100	104 Hz (Normal mode)	104 Hz (High Performance)
0101	208 Hz (Normal mode)	208 Hz (High Performance)
0110	416 Hz (High Performance)	416 Hz (High Performance)
0111	833 Hz (High Performance)	833 Hz (High Performance)
1000	1.66 kHz (High Performance)	1.66 kHz (High Performance)
1001	3.33 kHz (High Performance)	3.33 kHz (High Performance)
1010	6.66 kHz (High Performance)	6.66 kHz (High Performance)

Table 6. Power consumption shows the typical values of power consumption for the different operating modes.

Table 6. **Power consumption**

ODR [Hz]	Accelerometer only (at Vdd = 1.8 V)	Gyroscope only (at Vdd = 1.8 V)	Combo [Acc + Gyro] (at Vdd = 1.8 V)
Power Down	-	-	10 µA
1.6 Hz (Low Power)	4.5 µA	-	-
12.5 Hz (Low Power)	9 µA	255 µA	280 µA
26 Hz (Low Power)	14 µA	270 µA	300 µA
52 Hz (Low Power)	25 µA	300 µA	350 µA
104 Hz (Normal mode)	44 µA	350 µA	440 µA
208 Hz (Normal mode)	85 µA	460 µA	550 µA
12.5 Hz (High Perf.)	180 µA	620 µA	750 µA
26 Hz (High Perf.)	180 µA	620 µA	750 µA
52 Hz (High Perf.)	180 µA	620 µA	750 µA
104 Hz (High Perf.)	180 µA	620 µA	750 µA
208 Hz (High Perf.)	180 µA	620 µA	750 µA
416 Hz (High Perf.)	180 µA	620 µA	750 µA
833 Hz (High Perf.)	180 µA	620 µA	750 µA
1.66 kHz (High Perf.)	190 µA	620 µA	750 µA
3.33 kHz (High Perf.)	190 µA	620 µA	750 µA
6.66 kHz (High Perf.)	190 µA	620 µA	750 µA

3.1 Power-Down mode

When the accelerometer/gyroscope is in Power-Down mode, almost all internal blocks of the device are switched off to minimize power consumption. The digital interfaces (I^2C and SPI) are still active to allow communication with the device. The content of the configuration registers is preserved and the output data registers are not updated, keeping the last data sampled in memory before going into Power-Down mode.

3.2 High-Performance mode

In High-Performance mode, all accelerometer/gyroscope circuitry is always on and data are generated at the data rate selected through the ODR_XL/ODR_G bits.

Data interrupt generation is active.

3.3 Normal mode

While High-Performance mode guarantees the best performance in terms of noise, Normal mode further reduces the current consumption. The accelerometer/gyroscope data reading chain is automatically turned on and off to save power. In the gyroscope device, only the driving circuitry is always on.

Data interrupt generation is active.

3.4 Low-Power mode

Low-Power mode differs from Normal mode in the available output data rates. In Low-Power mode low-speed ODRs are enabled. Four low-speed ODRs can be chosen for the accelerometer through the ODR_XL bits: 1.6 Hz, 12.5 Hz, 26 Hz and 52 Hz. Three low-speed ODRs can be chosen for the gyroscope thorough the ODR_G bits: 12.5 Hz, 26 Hz and 52 Hz.

Data interrupt generation is active.

3.5 Gyroscope Sleep mode

While the gyroscope is in Sleep mode the circuitry that drives the oscillation of the gyroscope mass is kept active. Compared to gyroscope Power-Down, turn-on time from Sleep mode to Low-Power/Normal/High-Performance mode is drastically reduced.

If the gyroscope is not configured in Power-Down mode, it enters in Sleep mode when the Sleep mode enable (SLEEP) bit of the CTRL4_C register is set to 1, regardless of the selected gyroscope ODR.

3.6 Connection modes

The ISM330DLC offers four different connection modes, described in detail in this document:

- **Mode 1:** it is the connection mode enabled by default; I^2C slave interface or SPI (3- / 4-wire) serial interface is available. When the device is configured in connection Mode1, the SCx/SDx pins cannot be left floating. It's recommended to connect both of them to VDDIO in order to optimize the power consumption during the device start-up sequence.
- **Mode 2:** it is the sensor hub mode; I^2C slave interface or SPI (3- / 4-wire) serial interface and I^2C interface master for external sensor connections are available. This connection mode is described in Section 6 Mode 2 - Sensor hub mode.
- **Mode 3:** in addition to the primary I^2C slave interface or SPI (3- / 4-wire) serial interface, an auxiliary SPI (3- / 4-wire) serial interface for external device connections (i.e. camera module) is available for the gyroscope only. This connection mode is described in Section 7 Mode 3 and Mode 4 - Auxiliary SPI modes.
- **Mode 4:** in addition to the primary I^2C slave interface or SPI (3- / 4-wire) serial interface, an auxiliary SPI (3- / 4-wire) serial interface for external device connections is available for both the gyroscope and accelerometer. This connection mode is described in Section 7 Mode 3 and Mode 4 - Auxiliary SPI modes.

3.7 Accelerometer bandwidth

The accelerometer sampling chain is represented by a cascade of four main blocks: an analog anti-aliasing low-pass filter, an ADC converter, a digital low-pass filter and the composite group of digital filters.

Figure 2. Accelerometer filtering chain (Mode 1/2/3) shows the accelerometer sampling chain when the device is configured in Mode 1, Mode 2 or Mode 3 configuration while Figure 3. Accelerometer filtering chain (Mode 4) shows it for Mode 4 configuration.

The analog signal coming from the mechanical parts is filtered by an analog low-pass anti-aliasing filter before being converted by the ADC. The anti-aliasing filter is enabled in High-Performance mode only and its bandwidth depends on the selected accelerometer ODR as shown in the following table.

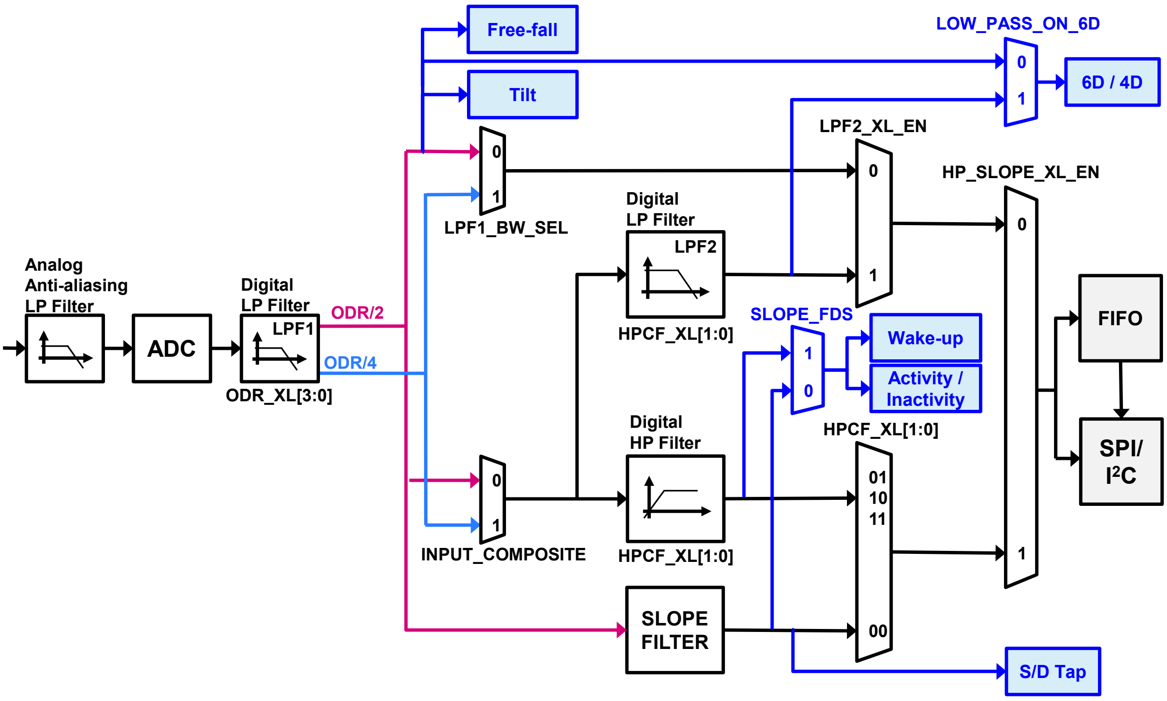
Table 7. **Accelerometer analog filter bandwidth**

Accelerometer ODR [Hz]	Analog filter BW [Hz]
≥ 1666	1500
< 1666	400

The analog filter bandwidth can be set to 400 Hz also for accelerometer ODR ≥ 1666 Hz by setting the BW0_XL bit to 1 in the CTRL1_XL register.

The digital LPF1 filter provides two outputs having different cutoff frequencies from each other; the desired LPF1 output can be selected through the LPF1_BW_SEL bit in the CTRL1_XL register and the INPUT_COMPOSITE bit in the CTRL8_XL register.

Figure 2. **Accelerometer filtering chain (Mode 1/2/3)**



Referring to Figure 2. Accelerometer filtering chain (Mode 1/2/3) and Figure 3. Accelerometer filtering chain (Mode 4), the cutoff frequency of the “ODR/2” output of the LPF1 filter is equal to ODR/2 in High-Performance mode and it is equal to 740 Hz in Low Power / Normal modes. The cutoff frequency of the “ODR/4” output is always equal to ODR/4, regardless of the selected power mode.

Finally, the composite group of filters composed of a low-pass digital filter (LPF2), a high-pass digital filter and a slope filter processes the digital signal.

When the ISM330DLC is configured in Mode 1/2/3, the CTRL8_XL register can be used to configure the composite filter group and the overall bandwidth of the accelerometer filtering chain, as shown in Table 8. Accelerometer bandwidth selection in Mode 1/2/3. Referring to this table, on the low-pass path side, the Bandwidth column refers to the LPF1 bandwidth if LPF2_XL_EN = 0; it refers to the LPF2 bandwidth if LPF2_XL_EN = 1. On the high-pass path side, the Bandwidth column refers to the Slope filter bandwidth if HPCF_XL[1:0] = 00b; it refers to the HP filter bandwidth if HPCF_XL[1:0] = 01b / 10b / 11b.

Table 8. Accelerometer bandwidth selection in Mode 1/2/3 also provides the maximum (worst case) settling time in terms of samples to be discarded for the various configurations of the accelerometer filtering chain. Further details are described in Section 3.9 Accelerometer and gyroscope turn-on/off time.

Table 8. Accelerometer bandwidth selection in Mode 1/2/3

HP_SLOPE_XL_EN	LPF2_XL_EN	LPF1_BW_SEL	HPCF_XL[1:0]	INPUT_ COMPOSITE	Bandwidth	Max. overall settling time [1] (samples to be discarded)
0 (Low-Pass path)	0	0	-	-	ODR/2	14
		1	-	-	ODR/4	14
	1	-	00	1 (low noise) 0 (low latency)	ODR/50	40
			01		ODR/100	80
			10		ODR/9	15
			11		ODR/400	320
1 (High-Pass path)	-	-	00	0	ODR/4	14
			01		ODR/100	80
			10		ODR/9	15
			11		ODR/400	320

1. *Settling time @ 99% of the final value*

Setting the HP_SLOPE_XL_EN bit to 0, the low-pass path of the composite filter block is selected. If the LPF2_XL_EN bit is set to 0, no additional filter is applied; if the LPF2_XL_EN bit is set to 1, the LPF2 filter is applied in addition to LPF1 and the overall bandwidth of the accelerometer chain can be set by configuring the HPCF_XL[1:0] field of the CTRL8_XL register.

The LPF2 low-pass filter can also be used in the 6D/4D functionality by setting the LOW_PASS_ON_6D bit of the CTRL8_XL register to 1.

Setting the HP_SLOPE_XL_EN bit to 1, the high-pass path of the composite filter block is selected: the HPCF_XL[1:0] field is used in order to enable, in addition to the LPF1 filter, either the Slope filter usage (when HPCF_XL[1:0] = 00b) or the digital high-pass filter (other HPCF_XL[1:0] configurations). The HPCF_XL[1:0] field is also used to select the cutoff frequencies of the HP filter.

The reference mode feature is available for the accelerometer sensor: when this feature is enabled, the current X, Y, Z accelerometer sample is internally stored and subtracted from all subsequent output values. In order to enable the reference mode, both the HP_REF_MODE bit and the HP_SLOPE_XL_EN bit of the CTRL8_XL register have to be set to 1, and the value of the HPCF_XL[1:0] field has to be different than 00b. When the reference mode feature is enabled, both the LPF2 filter and the HP filter are not available. The first accelerometer output data after enabling the reference mode has to be discarded.

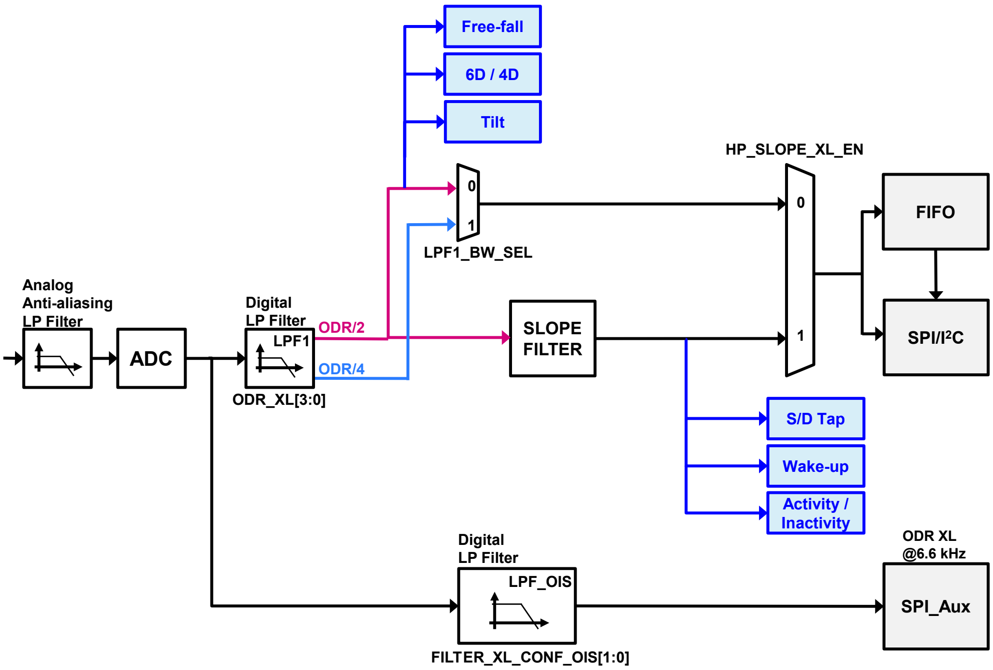


Figure 3. **Accelerometer filtering chain (Mode 4)**

When the ISM330DLC is configured in Mode 4, the accelerometer filtering chain becomes the one shown in Figure 3. Accelerometer filtering chain (Mode 4). In this configuration, two different data chains are available:

- The GP (General Purpose) chain, where the accelerometer data are provided to the primary I^{2}C/SPI with an ODR selectable from 1.6 Hz up to 6.66 kHz.
- The OIS chain, for control loop stability, where the accelerometer data are provided to the auxiliary SPI with an ODR fixed at 6.66 kHz.

Note: When the ISM330DLC is configured in Mode 4, only the Slope filter is available on the accelerometer GP chain side, whereas LPF2 and HP filters are not available: it is recommended to avoid using reference mode, LPF2 and HP filters in Mode 1/2/3 when Mode 4 is intended to be used.

The FILTER_XL_CONF_OIS_[1:0] bits of the CTRL3_OIS register can be used to select the overall accelerometer OIS chain bandwidth: its value also depends on the value of accelerometer ODR on the GP side (defined through the ODR_XL[3:0] bits in CTRL1_XL register), as described in the following table.

Table 9. **OIS chain (XL ODR = 6.66 kHz) - Accelerometer bandwidth selection in Mode 4**

FILTER_XL _CONF_OIS[1:0]	ODR_XL = 0 Hz (Power Down) ODR_XL ≥ 1600 Hz		ODR_XL ≤ 800 Hz		Max. overall settling time (samples to be discarded) [1]
	BW	Phase delay @ 20 Hz	BW	Phase delay @ 20 Hz	
00	140 Hz	9.39°	128 Hz	11.5°	40
01	68.2 Hz	17.6°	66.5 Hz	19.7°	80
10	636 Hz	2.96°	329 Hz	5.08°	15
11	295 Hz	5.12°	222 Hz	7.23°	25

1. *Settling time @ 99% of the final value*

A detailed description of Mode 4 connection mode is provided in Section 7 Mode 3 and Mode 4 - Auxiliary SPI modes.

3.7.1 Accelerometer slope filter

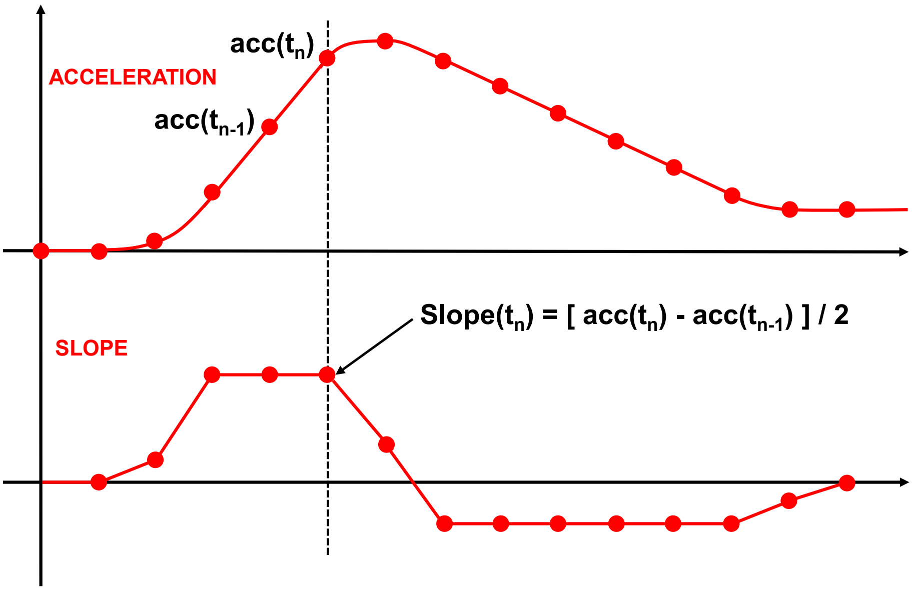
As shown in Figure 2. Accelerometer filtering chain (Mode 1/2/3), the ISM330DLC device embeds a digital Slope filter, which can also be used for some embedded features such as single/double-tap recognition, wake-up detection and activity/inactivity.

The slope filter output data is calculated using the following formula:

$$\mathrm{slope}(t_n) = [\mathrm{acc}(t_n) - \mathrm{acc}(t_{n-1})]/2$$

An example of a slope data signal is illustrated in Figure 4. Accelerometer slope filter.

Figure 4. Accelerometer slope filter

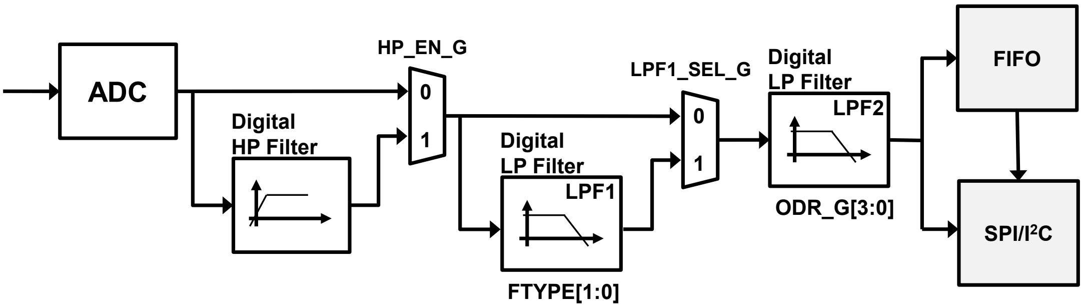


3.8 Gyroscope bandwidth

In the ISM330DLC device, the gyroscope filtering chain depends on the connection mode in use.

When Mode 1 or Mode 2 is selected, the gyroscope filtering chain configuration is the one shown in Figure 5. Gyroscope digital chain - Mode 1 and Mode 2). It is a cascade of three filters: a selectable digital high-pass filter (HPF), a selectable digital low-pass filter (LPF1) and a digital low-pass filter (LPF2).

Figure 5. Gyroscope digital chain - Mode 1 and Mode 2



The digital HP filter can be enabled by setting the HP_EN_G bit of the CTRL7_G register to 1. The digital HP filter cutoff frequency can be selected through the field HPM_G[1:0] of the CTRL7_G register, according to the table below.

Note: The embedded HP filter is available in High-Performance mode only. If the gyroscope is configured in Low-Power / Normal mode, the high-pass filter is bypassed regardless of the configuration of the HP_G_EN bit of CTRL7_G register.

Table 10. Gyroscope digital HP filter cutoff selection

HPM_G[1:0]	High-pass filter cutoff frequency [Hz]
00	0.016
01	0.065
10	0.260
11	1.040

The digital LPF1 filter can be enabled by setting the LPF1_SEL_G bit of CTRL4_C register to 1 and its bandwidth can be selected through the field FTYPE_[1:0] of the CTRL6_C register.

Note: The digital LPF1 filter is available in High-Performance mode only. If the gyroscope is configured in Low-Power / Normal mode, the LPF1 filter is bypassed regardless of the configuration of the LPF1_SEL_G bit of CTRL4_C register.

The digital LPF2 filter cannot be configured by the user (regardless of the selected power mode) and its cutoff frequency depends on the selected gyroscope ODR. When the gyroscope ODR is equal to 6.66 kHz, the LPF2 filter is bypassed.

The overall gyroscope bandwidth for different configurations of the LPF1_SEL_G bit of the CTRL4_C register and FTYPE_[1:0] of the CTRL6_C register is summarized in the following table.

Table 11. **Gyroscope overall bandwidth selection in Mode 1/2**

Gyroscope ODR [Hz]	LPF1_SEL_G	FTYPE[1:0]	Cutoff [Hz] (Phase delay @ 20 Hz)
12.5	0	-	4
	1	00	4
	1	01	4
	1	10	4
	1	11	4
26	0	-	8
	1	00	8
	1	01	8
	1	10	8
	1	11	8
52	0	-	17
	1	00	17 (144°)
	1	01	17 (146°)
	1	10	17 (149°)
	1	11	17 (142°)
104	0	-	33
	1	00	33 (75°)
	1	01	33 (77°)
	1	10	33 (79°)
	1	11	33 (73°)
208	0	-	67
	1	00	67 (40°)
	1	01	67 (42°)
	1	10	67 (45°)
	1	11	67 (39°)
416	0	-	137
	1	00	138 (23°)
	1	01	131 (25°)
	1	10	121 (28°)
	1	11	138 (21°)
833	0	-	312
	1	00	245 (14°)
	1	01	195 (17°)
	1	10	155 (19°)
	1	11	293 (13°)
1666	0	-	988
	1	00	315 (10°)
	1	01	224 (12°)
	1	10	168 (15°)
	1	11	505 (8°)

Gyroscope ODR [Hz]	LPF1_SEL_G	FTYPE[1:0]	Cutoff [Hz] (Phase delay @ 20 Hz)
3333	0	-	1161
	1	00	343 (8°)
	1	01	234 (10°)
	1	10	172 (12°)
	1	11	925 (6°)
6666	0	-	1250
	1	00	351 (7°)
	1	01	237 (9°)
	1	10	173 (11°)
	1	11	937 (5°)

If Mode 3 or Mode 4 is enabled, the gyroscope digital chain becomes the one shown in Figure 6. Gyroscope digital chain - Mode 3 and Mode 4. In this configuration, two different data chains are available:

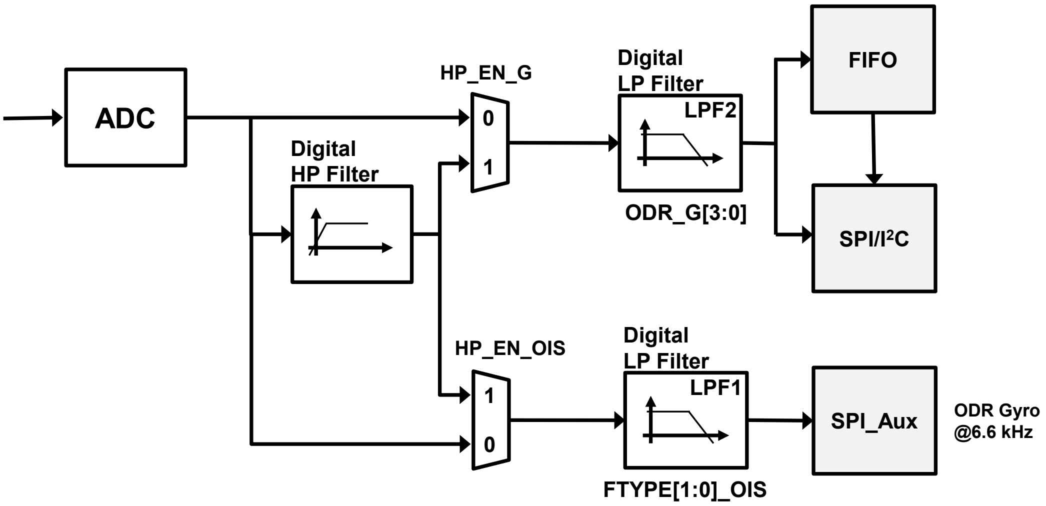
- The GP (General Purpose) chain, where the gyroscope data are provided to the primary I^{2}C /SPI with an ODR selectable from 12.5 Hz up to 6.66 kHz.
- The OIS chain, for control loop stability, where the gyroscope data are provided to the auxiliary SPI with an ODR fixed at 6.66 kHz.

In Mode 3/4, the LPF2 filter is dedicated to the GP chain only; the total bandwidth on the GP side depends on the gyroscope ODR value, as shown in Table 12. GP chain - Gyroscope overall bandwidth selection in Mode 3/4.

Table 12. **GP chain - Gyroscope overall bandwidth selection in Mode 3/4**

Gyroscope ODR [Hz]	Cutoff [Hz]
12.5	4
26	8
52	17
104	33
208	67
416	137
833	312
1666	988
3333	1161
6666	1250

Figure 6. Gyroscope digital chain - Mode 3 and Mode 4



The digital HP filter is shared between the GP and OIS chains but it can be applied to only one chain at a time:

- If bit HP_EN_G of the CTRL7_G register is set to 1, the HP filter is applied to the GP chain only, regardless of the value of the HP_EN_OIS bit of the CTRL2_OIS register;
- If bit HP_EN_G is set to 0 and bit HP_EN_OIS is set to 1, the HP filter is applied to the OIS chain.

The digital HPF cutoff frequency on the OIS chain can be selected through the field HPM_[1:0]_OIS of CTRL2_OIS, according to the table below.

Table 13. OIS chain - Gyroscope digital HP filter cutoff selection in Mode 3/4

HPM_[1:0]_OIS	High-pass filter cutoff frequency [Hz]
00	0.016
01	0.065
10	0.260
11	1.040

When the auxiliary SPI is enabled, the LPF1 digital low-pass filter is available on the OIS chain only. In this case, the OIS chain overall bandwidth can be selected through the FTYPE_[1:0]_OIS field of the CTRL2_OIS register, as shown in Table 14. OIS chain (Gyro ODR = 6.66 kHz) - Gyroscope overall bandwidth selection (Mode 3/4).

Table 14. OIS chain (Gyro ODR = 6.66 kHz) - Gyroscope overall bandwidth selection (Mode 3/4)

FTYPE_[1:0]_OIS	Bandwidth [Hz] (Phase delay @ 20 Hz)
00	351 Hz (7°)
01	237 Hz (9°)
10	173 Hz (11°)
11	937 Hz (5°)

Note: The digital LPF1 filter is not available on the gyroscope GP chain when Mode 3/4 is enabled. The recommendation is to avoid using the LPF1 filter when Mode 3/4 is intended to be used.

A detailed description of Mode 3 connection mode is provided in Section 7 Mode 3 and Mode 4 - Auxiliary SPI modes.

3.9 Accelerometer and gyroscope turn-on/off time

The accelerometer reading chain contains low-pass filtering to improve signal-to-noise performance and to reduce aliasing effects. For this reason, it is necessary to take into account the settling time of the filters when the accelerometer power mode is switched or when the accelerometer ODR is changed.

When the ISM330DLC is configured in Mode 1/2/3, the maximum overall turn-on/off time (with LPF2 and HP filters disabled) in order to switch accelerometer power modes or accelerometer ODR is shown in Table 15. Accelerometer turn-on/off time in Mode 1/2/3 (LPF2 and HP disabled)

Note: The accelerometer ODR timing is not impacted by power mode changes (a new configuration is effective after the completion of the current period).

Table 15. Accelerometer turn-on/off time in Mode 1/2/3 (LPF2 and HP disabled)

Starting mode	Target mode	Max turn-on/off time [1]
Power-Down	Low-Power / Normal	See Table 16.
Power-Down	High-Performance	See Table 16.
Low-Power / Normal	High-Performance	See Table 16. + discard 1 additional sample
Low-Power / Normal	Low-Power / Normal (ODR Change)	See Table 16.
High-Performance	Low-Power / Normal	See Table 16. + discard 1 additional sample
High-Performance @ ODR ≤ 833 Hz	High-Performance @ ODR ≤ 833 Hz	See Table 16. + discard 1 additional sample
High-Performance @ ODR ≤ 833 Hz	High-Performance @ ODR > 833 Hz	See Table 16. + discard 1 additional sample
High-Performance @ ODR > 833 Hz	High-Performance @ ODR ≤ 833 Hz	See Table 16. + discard 1 additional sample
High-Performance @ ODR > 833 Hz	High-Performance @ ODR > 833 Hz	Discard 5 samples
Low-Power / Normal / High-Performance	Power-Down	1 µs

1. *Settling time @ 99% of the final value*

Table 16. Accelerometer samples to be discarded in Mode 1/2/3

Target mode
Accelerometer ODR [Hz]
	Number of sample to be discarded with
	(LPF1_BW_SEL = 0 and
	LPF2_XL_EN = 0 and
	HP_SLOPE_XL_EN = 0)
		Number of sample to be discarded with
		(LPF1_BW_SEL = 1 and
		LPF2_XL_EN = 0 and
		HP_SLOPE_XL_EN = 0)
		OR
		(HPCF_XL = 00 and
		HP_SLOPE_XL_EN = 1)

1.6 (Low-Power)	0 (first sample correct)	1
12.5 (Low-Power)	0 (first sample correct)	1
26 (Low-Power)	0 (first sample correct)	1
52 (Low-Power)	0 (first sample correct)	1
104 (Normal)	0 (first sample correct)	1
208 (Normal)	0 (first sample correct)	1
12.5 (High-Performance)	0 (first sample correct)	1

Target mode
Accelerometer ODR [Hz]
	Number of sample to be discarded with
	(LPF1_BW_SEL = 0 and
	LPF2_XL_EN = 0 and
	HP_SLOPE_XL_EN = 0)
		Number of sample to be discarded with
		(LPF1_BW_SEL = 1 and
		LPF2_XL_EN = 0 and
		HP_SLOPE_XL_EN = 0)
		OR
		(HPCF_XL = 00 and
		HP_SLOPE_XL_EN = 1)

26 (High-Performance)	0 (first sample correct)	1
52 (High-Performance)	1	1
104 (High-Performance)	1	2
208 (High-Performance)	1	2
416 (High-Performance)	1	2
833 (High-Performance)	1	2
1666 (High-Performance)	2	2
3333 (High-Performance)	3	4
6666 (High-Performance)	13	13

When the ISM330DLC is configured in Mode 4, the maximum overall settling time in order to switch accelerometer power modes or accelerometer ODR on the GP chain is still the one shown in Table 15. Accelerometer turn-on/off time in Mode 1/2/3 (LPF2 and HP disabled); the maximum settling time for the OIS chain is shown in the right column of Table 9. OIS chain (XL ODR = 6.66 kHz) - Accelerometer bandwidth selection in Mode 4.

Turn-on/off time has to be considered also for the gyroscope sensor when switching its modes or when the gyroscope ODR is changed.

When the ISM330DLC is configured in Mode 1/2, the maximum overall turn-on/off time (with HP filter disabled) in order to switch gyroscope power modes or gyroscope ODR is shown in Table 17. Gyroscope turn-on/off time in Mode 1/2 (HP disabled).

Note: The gyroscope ODR timing is not impacted by power mode changes (a new configuration is effective after the completion of the current period).

Table 17. Gyroscope turn-on/off time in Mode 1/2 (HP disabled)

Starting mode	Target mode	Max turn-on/off time [(1)]
Power-Down	Sleep	70 ms
Power-Down	Low-Power / Normal	70 ms + discard 1 sample
Power-Down	High-Performance	70 ms + see Table 18. or Table 19.
Sleep	Low-Power / Normal	Discard 1 sample
Sleep	High-Performance	See Table 18. or Table 19.
Low-Power / Normal	High-Performance	Discard 2 samples
Low-Power / Normal	Low-Power / Normal (ODR change)	Discard 1 sample
High-Performance	Low-Power / Normal	Discard 1 sample
High-Performance	High-Performance (ODR change)	Discard 2 samples
Low-Power / Normal / High-Performance	Power-Down	1 µs if both XL and Gyro in PD 300 µs if XL not in PD

1. *Settling time @ 99% of the final value*

Table 18. Gyroscope samples to be discarded in Mode 1/2 (LPF1 disabled)

Gyroscope ODR [Hz]	Number of samples to be discarded
12.5 Hz	2
26 Hz	3
52 Hz	3
104 Hz	3
208 Hz	3
416 Hz	3
833 Hz	3
1.66 kHz	135
3.33 kHz	270
6.66 kHz	540

Table 19. Gyroscope samples to be discarded in Mode 1/2 (LPF1 enabled) for all ODRs

Gyroscope ODR [Hz]	FTYPE[1:0]	Number of samples to be discarded
12.5 Hz	00	2
	01	2
	10	2
	11	2
26 Hz	00	3
	01	3
	10	3
	11	3
52 Hz	00	3
	01	3
	10	3
	11	3
104 Hz	00	4
	01	4
	10	4
	11	4
208 Hz	00	4
	01	4
	10	5
	11	4
416 Hz	00	5
	01	6
	10	6
	11	5

Gyroscope ODR [Hz]	FTYPE[1:0]	Number of samples to be discarded
833 Hz	00	7
	01	8
	10	9
	11	6
1.66 kHz	00	135
	01	135
	10	135
	11	135
3.33 kHz	00	270
	01	270
	10	270
	11	270
6.66 kHz	00	540
	01	540
	10	540
	11	540

When the ISM330DLC is configured in Mode 3/4, the maximum overall turn-on time in order to switch gyroscope power modes or gyroscope ODR on the GP chain is still the one shown in Table 17. Gyroscope turn-on/off time in Mode 1/2 (HP disabled) and Table 18. Gyroscope samples to be discarded in Mode 1/2 (LPF1 disabled) (HP and LPF1 disabled case); the maximum turn-on time for the OIS chain is given in Table 20. OIS chain (Gyro ODR = 6.66 kHz) - Gyroscope turn-on/off time in Mode 3/4.

Table 20. **OIS chain (Gyro ODR = 6.66 kHz) - Gyroscope turn-on/off time in Mode 3/4**

GP chain gyroscope current mode	OIS chain gyroscope starting mode	OIS chain gyroscope target mode	Max. gyroscope turn-on time [1]
Power Down	Power-Down	Mode 3 / 4	70 ms + see Table 21.
Gyro ODR > 0 Low-Power / Normal / High-Performance	Power-Down	Mode 3 / 4	see Table 21.
-	Mode 3	Mode 4	First sample correct

1. *Settling time @ 99% of the final value*

Table 21. **Gyroscope samples to be discarded in Mode 3/4**

FTYPE[1:0]	Number of samples to be discarded
00	27
01	36
10	48
11	19

Note: The GYRO_SETTLING bit in the STATUS_REG/STATUS_SPIAux register is equal to 1 when the gyroscope OIS chain is in settling phase. The data read during this settling data are not valid: the recommendation is to check the status of this bit to understand when valid data are available in order to minimize the turn-on time.

3.9.1 GP chain settling time on OIS chain enable/disable

The accelerometer and gyroscope GP chains are affected by the OIS chains enable/disable.

The table below clarifies how many accelerometer/gyroscope samples have to be discarded on the GP side when the OIS interface is switched on/off.

Table 22. **GP chains settling time on OIS enable/disable**

GP starting mode	Target mode	GP chain sensor	Settling time on Mode 3/4 enable[1]	Settling time on Mode 3/4 disable[1]
Gyroscope High-Performance	Enable and then disable Mode 3/4, without ODR change on GP side	Gyroscope	First sample correct	First sample correct
Accelerometer High-Performance	Enable and then disable Mode 4, without ODR change on GP side	Accelerometer	First sample correct	First sample correct
Gyroscope Low-Power / Normal	Enable and then disable Mode 3/4, without ODR change on GP side	Gyroscope	First sample correct (Gyroscope switches to High-Performance)	First sample correct (Gyroscope switches to Low-Power / Normal)
Accelerometer Low-Power / Normal	Enable and then disable Mode 4, without ODR change on GP side	Accelerometer	See Table 15.: Low Power / Normal mode to High-Performance case (Accelerometer switches to High-Performance)	See Table 15.: High Performance to Low-Power / Normal mode case (Accelerometer comes back to Low-Power / Normal)

1. *Settling time @ 99% of the final value*

Note: When Mode 3 is enabled, only the gyroscope OIS chain is turned-on: since the accelerometer OIS chain is not considered, the accelerometer GP chain is not impacted by Mode 3 enable.

The gyroscope GP and the accelerometer GP settling time are independent of each other. An ODR or power mode change of one sensor does not affect the settling of the other one.

Each enable/disable of an OIS chain event can be detected from the GP side by reading the OIS_EN_SPI2 bit in the CTRL1_OIS register. This register is read-only when accessed from the GP side.

4 Mode 1 - Reading output data

4.1 Startup sequence

Once the device is powered up, it automatically downloads the calibration coefficients from the embedded flash to the internal registers. When the boot procedure is completed, i.e. after approximately 15 milliseconds, the accelerometer and gyroscope automatically enter Power-Down mode.

To turn on the accelerometer and gather acceleration data through the primary I^2C / SPI interface, it is necessary to select one of the operating modes through the CTRL1_XL register.

The following general-purpose sequence can be used to configure the accelerometer:

```
1.   Write CTRL1_XL = 60h          // Acc = 416 Hz (High-Performance mode)
2.   Write INT1_CTRL = 01h         // Acc data-ready interrupt on INT1
```

To turn on the gyroscope and gather angular rate data through the primary I^2C / SPI interface, it is necessary to select one of the operating modes through the CTRL2_G.

The following general-purpose sequence can be used to configure the gyroscope:

```
1.   Write CTRL2_G = 60h           // Gyro = 416 Hz (High-Performance mode)
2.   Write INT1_CTRL = 02h         // Gyro data-ready interrupt on INT1
```

4.2 Using the status register

The device is provided with a STATUS_REG register which should be polled to check when a new set of data is available. The XLDA bit is set to 1 when a new set of data is available at accelerometer output; the GDA bit is set to 1 when a new set of data is available at gyroscope output.

For the accelerometer (the gyroscope is similar), the read of the output registers should be performed as follows:

1. Read STATUS
2. If XLDA = 0, then go to 1
3. Read OUTX_L_XL
4. Read OUTX_H_XL
5. Read OUTY_L_XL
6. Read OUTY_H_XL
7. Read OUTZ_L_XL
8. Read OUTZ_H_XL
9. Data processing
10. Go to 1

4.3 Using the data-ready signal

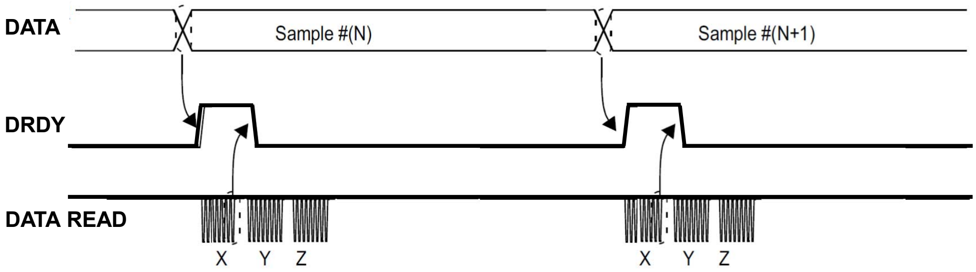
The device can be configured to have one HW signal to determine when a new set of measurement data is available to be read.

For the accelerometer sensor, the data-ready signal is represented by the XLDA bit of the STATUS_REG register. The signal can be driven to the INT1 pin by setting the INT1_DRDY_XL bit of the INT1_CTRL register to 1 and to the INT2 pin by setting the INT2_DRDY_XL bit of the INT2_CTRL register to 1.

For the gyroscope sensor, the data-ready signal is represented by the GDA bit of the STATUS_REG register. The signal can be driven to the INT1 pin by setting the INT1_DRDY_G bit of the INT1_CTRL register to 1 and to the INT2 pin by setting the INT2_DRDY_G bit of the INT2_CTRL register to 1.

The data-ready signal rises to 1 when a new set of data has been generated and it is available to be read. The data-ready signal can be either latched or pulsed: if the DRDY_PULSED bit of the DRDY_PULSE_CFG register is set to 0 (default value), then the data-ready signal is latched and the interrupt is reset when the higher part of one of the enabled channels is read (29h, 2Bh, 2Dh for the accelerometer; 23h, 25h, 27h for the gyroscope). If the DRDY_PULSED bit of the DRDY_PULSE_CFG register is set to 1, then the data-ready is pulsed and the duration of the pulse observed on the interrupt pins is 75 µs. Pulsed mode is not applied to the XLDA and GDA bits which are always latched.

Figure 7. Data-ready signal



4.3.1 DRDY mask functionality

Setting the DRDY_MASK bit of the CTRL4_C register to 1, the accelerometer and gyroscope data-ready signals are masked until the settling of the sensor filters is completed.

When FIFO is active and the DRDY_MASK bit is set to 1, accelerometer/gyroscope invalid samples stored in FIFO can be equal to 7FFFh, 7FFEh or 7FFDh. In this way, a tag is applied to the invalid samples stored in the FIFO buffer so that they can be easily identified and discarded during data post-processing.

Note: The DRDY_MASK bit acts only on the accelerometer LPF1 and the gyroscope LPF2 digital filters settling time.

4.4 Using the block data update (BDU) feature

If reading the accelerometer/gyroscope data is particularly slow and cannot be synchronized (or it is not required) with either the XLDA/GDA bits in the STATUS_REG register or with the DRDY signal driven to the INT1/INT2 pins, it is strongly recommended to set the BDU (Block Data Update) bit to 1 in the CTRL3_C register.

This feature avoids reading values (most significant and least significant parts of output data) related to different samples. In particular, when the BDU is activated, the data registers related to each channel always contain the most recent output data produced by the device, but, in case the read of a given pair (i.e. OUTX_H_XL(G) and OUTX_L_XL(G), OUTY_H_XL(G) and OUTY_L_XL(G), OUTZ_H_XL(G) and OUTZ_L_XL(G)) is initiated, the refresh for that pair is blocked until both MSB and LSB parts of the data are read.

Note: BDU only guarantees that the LSB part and MSB part have been sampled at the same moment. For example, if the reading speed is too slow, X and Y can be read at T1 and Z sampled at T2.

4.5 Understanding output data

The measured acceleration data are sent to the OUTX_H_XL, OUTX_L_XL, OUTY_H_XL, OUTY_L_XL, OUTZ_H_XL, and OUTZ_L_XL registers. These registers contain, respectively, the most significant part and the least significant part of the acceleration signals acting on the X, Y, and Z axes.

The measured angular rate data are sent to the OUTX_H_G, OUTX_L_G, OUTY_H_G, OUTY_L_G, OUTZ_H_G, and OUTZ_L_G registers. These registers contain, respectively, the most significant part and the least significant part of the angular rate signals acting on the X, Y, and Z axes.

The complete output data for the X, Y, Z channels is given by the concatenation OUTX_H_XL(G) & OUTX_L_XL(G), OUTY_H_XL(G) & OUTY_L_XL(G) , OUTZ_H_XL(G) & OUTZ_L_XL(G) and it is expressed as a two’s complement number.

Both acceleration data and angular rate data are represented as 16-bit numbers.

4.5.1 Big-little endian selection

The ISM330DLC allows swapping the content of the lower and the upper part of the output data registers (i.e. OUTX_H_XL(G) with OUTX_L_XL(G), and OUT_TEMP_H with OUT_TEMP_L) in order 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. This mode corresponds to the BLE bit of the CTRL3_C register set to 0 (default configuration).

On the contrary, “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. This mode corresponds to the BLE bit of the CTRL3_C register set to 1.

4.5.2 Examples of output data

Table 23. Output data registers content vs. acceleration (FS_XL = ±2 *g*) provides a few basic examples of the accelerometer data that is read in the data registers when the device is subject to a given acceleration.

Table 24. Output data registers content vs. angular rate (FS_G = ±250 dps) provides a few basic examples of the gyroscope data that is read in the data registers when the device is subject to a given angular rate.

The values listed in the following tables are given under the hypothesis of perfect device calibration (i.e. no offset, no gain error,....) and practically show the effect of the BLE bit.

Table 23. Output data registers content vs. acceleration (FS_XL = ±2 *g*)

Acceleration values	BLE = 0		BLE = 1	
	Register address			
	OUTX_H_XL (29h)	OUTX_L_XL (28h)	OUTX_H_XL (29h)	OUTX_L_XL (28h)
0 *g*	00h	00h	00h	00h
350 m*g*	16h	69h	69h	16h
1 *g*	40h	09h	09h	40h
-350 m*g*	E9h	97h	97h	E9h
-1 *g*	BFh	F7h	F7h	BFh

Table 24. Output data registers content vs. angular rate (FS_G = ±250 dps)

Angular rate values	BLE = 0		BLE = 1	
	Register address			
	OUTX_H_G (23h)	OUTX_L_G (22h)	OUTX_H_G (23h)	OUTX_L_G (22h)
0 dps	00h	00h	00h	00h
100 dps	2Ch	A4h	A4h	2Ch
200 dps	59h	49h	49h	59h
-100 dps	D3h	5Ch	5Ch	D3h
-200 dps	A6h	B7h	B7h	A6h

4.6 Accelerometer offset registers

The ISM330DLC provides the accelerometer offset registers (X_OFS_USR, Y_OFS_USR, Z_OFS_USR) which can be used for zero-g offset correction or, in general, to apply an offset to the accelerometer output data.

The offset value set in the offset registers is internally subtracted from the measured acceleration value for the Z-axis, and it is internally added to the measured acceleration value for the X and Y axes; internally processed data are then sent to the accelerometer output register and to the FIFO (if enabled). These registers values are expressed as an 8-bit word in two’s complement and must be in the range [-127, 127].

The weight [*g*/LSB] to be applied to the offset register values is independent of the accelerometer selected full-scale and can be configured using the USR_OFF_W bit of the CTRL6_C register:

- 2^{-10}*g*/LSB if the USR_OFF_W bit is set to 0;
- 2^{-6}*g*/LSB if the USR_OFF_W bit is set to 1.

4.7 Rounding functions

The rounding function can be used to auto address the ISM330DLC registers for a circular burst-mode read. Basically, with a multiple read operation the address of the register that is being read goes automatically from the first register to the last register of the pattern and then goes back to the first one.

4.7.1 Rounding of FIFO output registers

The rounding function is automatically enabled when performing a multiple read operation of the FIFO output registers FIFO_DATA_OUT_L (3Eh) and FIFO_DATA_OUT_H (3Fh).

4.7.2 Rounding of source registers

It is possible to apply the rounding function also to the source registers of the ISM330DLC device, in order to verify with one multiple read whether new data was generated or a new interrupt event was detected.

The rounding function on the source registers can be enabled by setting the ROUNDING_STATUS bit of the CTRL7_G register to 1. When this function is enabled, with a multiple read operation the address of the register that is being read cycles automatically on WAKE_UP_SRC(1Bh), TAP_SRC(1Ch), D6D_SRC(1Dh), STATUS_REG (1Eh) and FUNC_SRC1 (53h) and goes back to WAKE_UP_SRC (1Bh).

4.7.3 Rounding of sensor output registers

The rounding function can also be enabled for the following groups of output registers:

- Gyroscope output registers, from OUTX_L_G (22h) to OUTZ_H_G (27h);
- Accelerometer output registers, from OUTX_L_XL (28h) to OUTZ_H_XL (2Dh);
- First group of sensor hub output registers, from SENSORHUB1_REG (2Eh) to SENSORHUB6_REG (33h);
- Second group of sensor hub output registers, from SENSORHUB7_REG (34h) to SENSORHUB12_REG (39h).

The output registers rounding pattern can be configured using the bits ROUNDING[2:0] of the CTRL5_C register, as indicated in Table 25. Output registers rounding pattern.

Table 25. **Output registers rounding pattern**

ROUNDING[2:0]	Rounding pattern
000	No rounding
001	Accelerometer only
010	Gyroscope only
011	Gyroscope + Accelerometer
100	1st group of Sensor Hub only
101	Accelerometer + 1st group of Sensor Hub
110	Gyroscope + Accelerometer + 1st group of Sensor Hub + 2nd group of Sensor Hub
111	Gyroscope + Accelerometer + 1st group of Sensor Hub

4.8 Edge-sensitive and level-sensitive data enable (DEN)

The ISM330DLC allows an external trigger level recognition through enabling the TRIG_EN, LVL1_EN, LVL2_EN bits in CTRL6_C.

Four different modes can be selected (Table 26. DEN configurations):

- Edge-sensitive trigger mode
- Level-sensitive trigger mode
- Level-sensitive latched mode
- Level-sensitive FIFO enable mode

The Data Enable (DEN) input signal is driven on the INT2 pin, which is configured as an input pin when one of the trigger modes is enabled.

The DEN functionality is active by default on the gyroscope data only. To extend this feature to the accelerometer data, the bit DEN_XL_EN in CTRL4_C must be set to 1.

The DEN active level is low by default. It can be changed to active-high by setting the bit DEN_LH in CTRL5_C to 1.

Table 26. **DEN configurations**

TRIG_EN	LVL1_EN	LVL2_EN	Function
1	0	0	Edge-sensitive trigger mode
0	1	0	Level-sensitive trigger mode
0	1	1	Level-sensitive latched mode
1	1	0	Level-sensitive FIFO enable mode

4.8.1 Edge-sensitive trigger mode

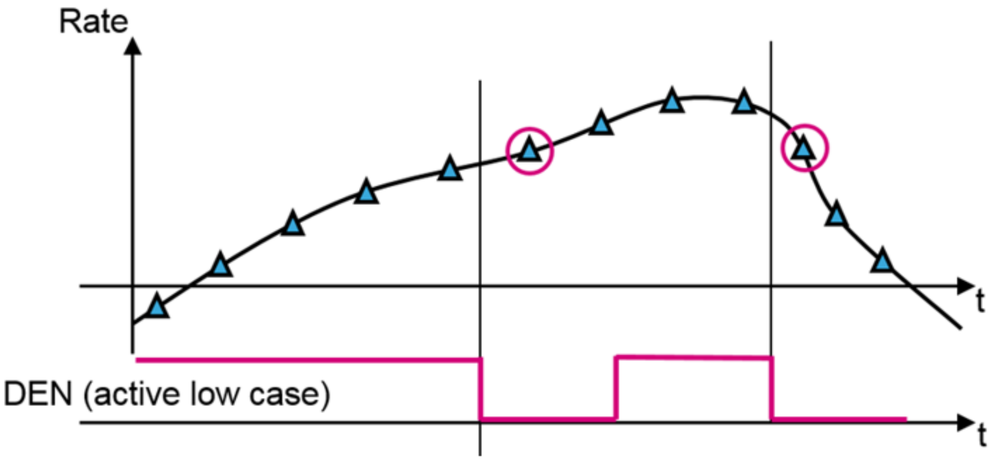
Edge-sensitive trigger mode can be enabled by setting the TRIG_EN bit in CTRL6_C to 1, and LVL1_EN, LVL2_EN bits in CTRL6_C to 0.

The edge-sensitive trigger works only when the low-pass filter LPF2 is disabled (LPF2_XL_EN = 0 in CTRL8_XL register).

Once the edge-sensitive trigger mode is enabled, the FIFO buffer and output registers are filled with the first sample acquired after every rising edge (if DEN_LH bit is equal to 1) or falling edge (if DEN_LH bit is equal to 0) of the DEN input signal.

Figure 8. Edge-sensitive trigger mode, DEN active low shows, with red circles, the samples acquired after the falling edges (DEN active low).

Figure 8. **Edge-sensitive trigger mode, DEN active low**



Edge-sensitive trigger mode, when enabled, acts only on the gyroscope output registers. The DRDY_G is related only to downsampled data, while the accelerometer output registers and DRDY_XL are updated according to ODR_XL. If the DEN_XL_EN bit is set to 1, the accelerometer sensor is downsampled too. In this case, the gyroscope and accelerometer have to be set in combo mode at the same ODR. The accelerometer standalone mode could be used by setting the gyroscope in Power-Down. In this case, DRDY_XL relates to downsampled data only.

Please note that the DEN trigger is internally latched before the update of the data registers: if a trigger occurs after this event, DEN will be acknowledged in the next ODR.

There are three possible configurations for the edge-sensitive trigger in FIFO, described below:

1. **Only gyroscope in trigger mode but not saved in FIFO:** in this case, FIFO is related only to the accelerometer and works as usual.
2. **Only gyroscope in trigger mode and saved in FIFO:** in this case the gyroscope decimation bits DEC_FIFO_GYRO [2:0] of the FIFO_CTRL3 register have to be set to 001 (gyroscope sensor in FIFO without decimation). Doing this, FIFO is driven by an external trigger. With this configuration, since also accelerometer data is written when the trigger occurs, possible repetition or loss of data for the accelerometer may occur.
3. **Gyroscope and accelerometer in trigger mode and saved in FIFO:** this configuration can be used by setting DEN_XL_EN to 1 and the gyroscope and accelerometer decimation bits DEC_FIFO_GYRO [2:0] and DEC_FIFO_XL [2:0] of the FIFO_CTRL3 register to 001 (gyroscope and accelerometers in FIFO without decimation). In this case, data of both sensors are written in FIFO when trigger occurs.

In the example shown below, the FIFO has been configured to store both the gyroscope data and the accelerometer data in the FIFO buffer; when the DEN signal toggles, the data are written to FIFO on the rising edge.

1.	Write 09h to FIFO_CTRL3	// Enable gyroscope and accelerometer in FIFO (no decimation)
2.	Write 26h to FIFO_CTRL5	// Set FIFO in Continuous mode, FIFO ODR = 104 Hz
3.	Write 80h to CTRL6_C	// Enable the edge-sensitive trigger // INT2 pin is switched to input mode (DEN signal)
4.	Write 80h to CTRL4_C	// Extend DEN functionality to accelerometer sensor
5.	Write 40h to CTRL1_XL	// Turn on the accelerometer: ODR_XL = 104 Hz, FS_XL = ±2 *g*
6.	Write 4Ch to CTRL2_G	// Turn on the gyroscope // ODR_G = 104 Hz, FS_G = ±2000 dps

4.8.2 Level-sensitive trigger mode

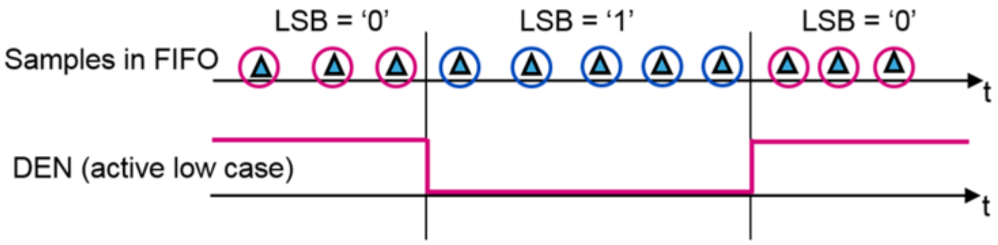
Level-sensitive trigger mode can be enabled by setting the LVL1_EN bit in CTRL6_C to 1, and the TRIG_EN, LVL2_EN bits in CTRL6_C to 0.

Once the level-sensitive trigger mode is enabled, the LSB bit of the selected data (in output registers and FIFO) is replaced by 1 if the DEN level is active, or 0 if the DEN level is not active. The selected data can be the X, Y, Z axes of the accelerometer or gyroscope sensor, defined through the DEN_X, DEN_Y, DEN_Z, DEN_XL_G bits in CTRL9_XL.

All data can be stored in the FIFO according to the FIFO settings.

Figure 9. Level-sensitive trigger mode, DEN active low shows with red circles the samples stored in the FIFO with LSB = 0 (DEN not active) and with blue circles the samples stored in the FIFO with LSB = 1 (DEN active).

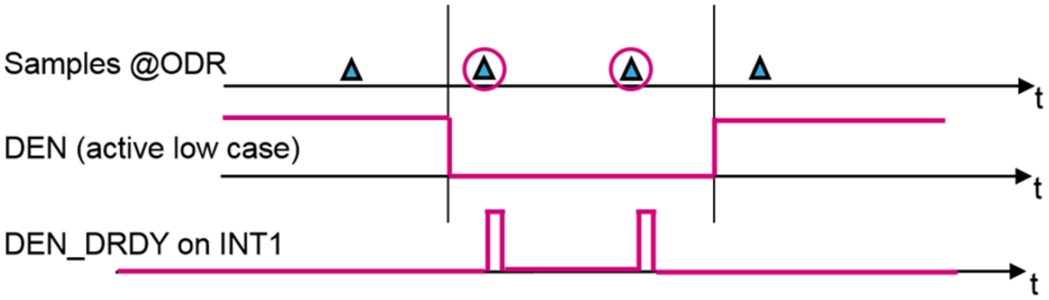
Figure 9. **Level-sensitive trigger mode, DEN active low**



When the level-sensitive trigger mode is enabled, the DEN signal can also be used to filter the data-ready signal on the INT1 pin. INT1 will show data-ready information only when the DEN pin is in the active state. To do this, the bit DEN_DRDY_INT1 of the CTRL4_C register must be set to 1. The interrupt signal can be latched or pulsed according to the DRDY_PULSED bit of the DRDY_PULSE_CFG register.

Figure 10. Level-sensitive trigger mode, DEN active low, DEN_DRDY on INT1 shows an example of data-ready on INT1 when the DEN level is low (active state).

Figure 10. **Level-sensitive trigger mode, DEN active low, DEN_DRDY on INT1**



4.8.3 Level-sensitive latched mode

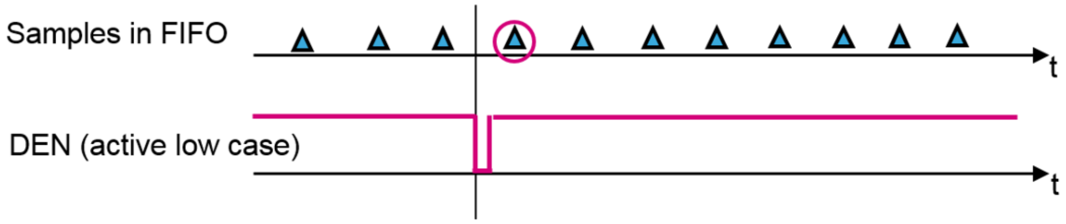
Level-sensitive latched mode can be enabled by setting the LVL1_EN and LVL2_EN bits in CTRL6_C to 1, and the TRIG_EN bit in CTRL6_C to 0.

When the level-sensitive latched mode is enabled, the LSB bit of the selected data (in output registers and FIFO) is normally set to 0 and becomes 1 only on the first sample after a pulse on the DEN pin.

Data can be selected through the DEN_X, DEN_Y, DEN_Z, DEN_XL_G bits in CTRL9_XL.

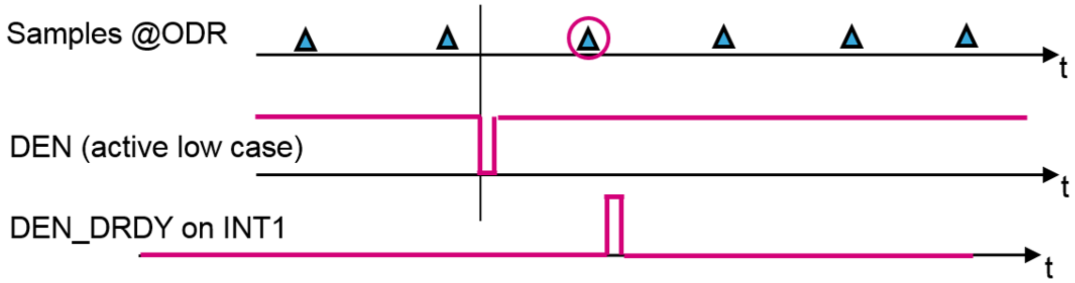
Figure 11. Level-sensitive latched mode, DEN active low shows an example of level-sensitive latched mode with DEN active low. After the pulse on the DEN pin, the sample with a red circle will have the value 1 on the LSB bit. All the other samples will have LSB bit 0.

Figure 11. **Level-sensitive latched mode, DEN active low**



When the level-sensitive latched mode is enabled and the bit DEN_DRDY_INT1 of the CTRL4_C register is set to 1, a pulse is generated on INT1 pin in corresponding to the availability of the first sample generated after the DEN pulse occurrence (see Figure 12. Level-sensitive latched mode, DEN active low, DEN_DRDY on INT1).

Figure 12. **Level-sensitive latched mode, DEN active low, DEN_DRDY on INT1**



4.8.4 Level-sensitive FIFO enabled

Level-sensitive FIFO enable mode can be enabled by setting the TRIG_EN and LVL1_EN bits in CTRL6_C to 1, and the LVL2_EN bit in CTRL6_C to 0.

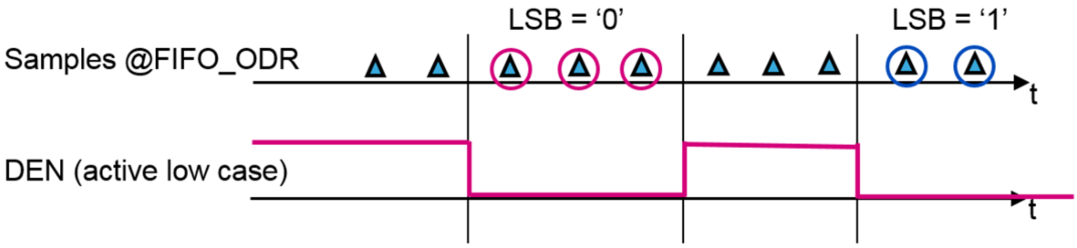
Once the level-sensitive FIFO enable mode is enabled, data is stored in the FIFO only when the DEN pin is equal to the active state.

In this mode, the LSB bit of the selected data (in output registers and FIFO) is replaced by 0 for odd DEN events and by 1 for even DEN events. This feature allows distinguishing the data stored in FIFO during the current DEN active window from the data stored in FIFO during the next DEN active window.

The selected data can be the X, Y, Z axes of the accelerometer or gyroscope sensor. Data can be selected through the DEN_X, DEN_Y, DEN_Z, DEN_XL_G bits in CTRL9_XL.

An example of level-sensitive FIFO enable mode is shown in Figure 13. Level-sensitive FIFO enable mode, DEN active low, the red circles show the samples stored in the FIFO with LSB bit 0, while the blue circles show the samples with LSB bit 1.

Figure 13. **Level-sensitive FIFO enable mode, DEN active low**



4.8.5 LSB selection for DEN stamping

When level-sensitive modes (trigger or latched) are used, it is possible to select which LSB have to contain information related to DEN pin behavior. This information can be stamped on the accelerometer or gyroscope axes in accordance with bits DEN_X, DEN_Y, DEN_Z and DEN_XL_G of the CTRL9_XL register. Setting to 1 the DEN_X, DEN_Y, DEN_Z bits, DEN information is stamped in the LSB of the corresponding axes of the sensor selected with the DEN_XL_G bit. By setting DEN_XL_G to 0, the DEN information is stamped in the selected gyroscope axes, while by setting DEN_XL_G to 1, the DEN information is stamped in the selected accelerometer axes.

By default, the bits are configured to have information on all the gyroscope axes.

4.8.6 OIS DEN mode

The level-sensitive modes can be enabled, for the gyroscope only, on the OIS chain through the bits LVL1_OIS in register CTRL1_OIS and LVL2_OIS in register INT_OIS:

- Level-sensitive trigger mode is selected when LVL1_OIS = 1 and LVL2_OIS = 0;
- Level-sensitive latched mode is selected when LVL1_OIS = 1 and LVL2_OIS = 1.

It is possible to set the DEN active level through the bit DEN_LH_OIS in register CTRL3_OIS:

- DEN is active-low when DEN_LH_OIS = 0;
- DEN is active-high when DEN_LH_OIS = 1.

Once one of the two OIS DEN modes is enabled, the LSB bit of all three axes changes as described in the previous paragraphs. In this case, there is no possibility to select one or two axes only.

5 Interrupt generation

In the ISM330DLC device the interrupt generation is based on accelerometer data only, so, for interrupt-generation purposes, the accelerometer sensor has to be set in an active operating mode (not in Power-Down); the gyroscope sensor can be configured in Power-Down mode since it's not involved in interrupt generation.

The interrupt generator can be configured to detect:

- Free-fall;
- Wake-up;
- 6D/4D orientation detection;
- Single-tap and double-tap sensing;
- Activity/Inactivity recognition;
- Relative tilt;
- Timestamp.

All these interrupt signals, together with the FIFO interrupt signals, can be independently driven to the INT1 and INT2 interrupt pins or checked by reading the dedicated source register bits.

The H_LACTIVE bit of the CTRL3_C register must be used to select the polarity of the interrupt pins. If this bit is set to 0 (default value), the interrupt pins are active high and they change from low to high level when the related interrupt condition is verified. Otherwise, if the H_LACTIVE bit is set to 1 (active low), the interrupt pins are normally at high level and they change from high to low when interrupt condition is reached.

The PP_OD bit of CTR3_C allows changing the behavior of the interrupt pins from push-pull to open drain. If the PP_OD bit is set to 0, the interrupt pins are in push-pull configuration (low-impedance output for both high and low level). When the PP_OD bit is set to 1, only the interrupt active state is a low-impedance output.

The LIR bit of TAP_CFG allows applying the latched mode to the interrupt signals. When the LIR bit is set to 1, once the interrupt pin is asserted, it must be reset by reading the related interrupt source register. If the LIR bit is set to 0, the interrupt signal is automatically reset when the interrupt condition is no longer verified or after a certain amount of time.

5.1 Interrupt pin configuration

The device is provided with two pins that can be activated to generate either data-ready or interrupt signals. The functionality of these pins is selected through the MD1_CFG and INT1_CTRL registers for the INT1 pin, and through the MD2_CFG and INT2_CTRL registers for the INT2 pin.

A brief description of these interrupt control registers is given in the following summary; the default value of their bits is equal to 0, which corresponds to 'disable'. In order to enable the routing of a specific interrupt signal on the pin, the related bit has to be set to 1.

Table 27. **INT1_CTRL register**

b7	b6	b5	b4	b3	b2	b1	b0
0	0	INT1_FULL_FLAG	INT1_FIFO_OVR	INT1_FTH	INT1_BOOT	INT1_DRDY_G	INT1_DRDY_XL

- INT1_FULL_FLAG: FIFO full flag interrupt on INT1
- INT1_FIFO_OVR: FIFO overrun flag interrupt on INT1
- INT1_FTH: FIFO threshold interrupt on INT1
- INT1_BOOT: Boot interrupt on INT1
- INT1_DRDY_G: Gyroscope data-ready on INT1
- INT1_DRDY_XL: Accelerometer data-ready on INT1

Table 28. **MD1_CFG register**

b7	b6	b5	b4	b3	b2	b1	b0
INT1_ INACT_ STATE	INT1_ SINGLE_ TAP	INT1_ WU	INT1_ FF	INT1_ DOUBLE_ TAP	INT1_ 6D	INT1_ TILT	INT1_ TIMER

- INT1_INACT_STATE: Inactivity interrupt on INT1
- INT1_SINGLE_TAP: Single-tap interrupt on INT1
- INT1_WU: Wake-up interrupt on INT1
- INT1_FF: Free-fall interrupt on INT1
- INT1_DOUBLE_TAP: Double-tap interrupt on INT1
- INT1_6D: 6D detection interrupt on INT1
- INT1_TILT: Tilt interrupt on INT1
- INT1_TIMER: Timer interrupt on INT1

Table 29. **INT2_CTRL register**

b7	b6	b5	b4	b3	b2	b1	b0
0	0	INT2_ FULL_ FLAG	INT2_ FIFO_ OVR	INT2_ FTH	INT2_ DRDY_ TEMP	INT2_ DRDY_G	INT2_ DRDY_ XL

- INT2_FULL_FLAG: FIFO full flag interrupt on INT2
- INT2_FIFO_OVR: FIFO overrun flag interrupt on INT2
- INT2_FTH: FIFO threshold interrupt on INT2
- INT2_DRDY_TEMP: Temperature data-ready on INT2
- INT2_DRDY_G: Gyroscope data-ready on INT2
- INT2_DRDY_XL: Accelerometer data-ready on INT2

Table 30. **MD2_CFG register**

b7	b6	b5	b4	b3	b2	b1	b0
INT2_ INACT_ STATE	INT2_ SINGLE_ TAP	INT2_ WU	INT2_ FF	INT2_ DOUBLE_ TAP	INT2_ 6D	INT2_ TILT	INT2_ IRON

- INT2_INACT_STATE: Inactivity interrupt on INT2
- INT2_SINGLE_TAP: Single-tap interrupt on INT2
- INT2_WU: Wake-up interrupt on INT2
- INT2_FF: Free-fall interrupt on INT2
- INT2_DOUBLE_TAP: Double-tap interrupt on INT2
- INT2_6D: 6D detection interrupt on INT2
- INT2_TILT: Tilt interrupt on INT2
- INT2_IRON: Soft-iron / hard-iron interrupt on INT2

If multiple interrupt signals are routed on the same pin (INTx), the logic level of this pin is the “OR” combination of the selected interrupt signals. In order to know which event has generated the interrupt condition, the related source registers have to be read: WAKE_UP_SRC, D6D_SRC, TAP_SRC, FUNC_SRC1 and FUNC_SRC2.

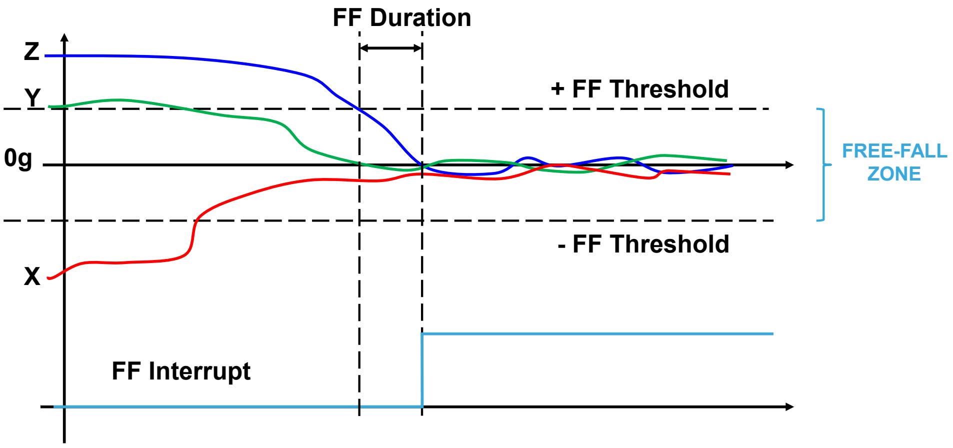
The INT2_on_INT1 pin of CTRL4_C register allows driving all the enabled interrupt signals in logic “OR” on the INT1 pin (by setting this bit to 1). When this bit is set to 0, the interrupt signals are divided between the INT1 and INT2 pins.

The basic interrupts (6D/4D, free-fall, wake-up, tap, inactivity) have to be enabled by setting the INTERRUPTS_ENABLE bit in the TAP_CFG register.

5.2 Free-fall interrupt

Free-fall detection refers to a specific register configuration that allows recognizing when the device is in free-fall: the acceleration measured along all the axes goes to zero. In a real case a “free-fall zone” is defined around the zero-*g* level where all the accelerations are small enough to generate the interrupt. Configurable threshold and duration parameters are associated to free-fall event detection: the threshold parameter defines the free-fall zone amplitude; the duration parameter defines the minimum duration of the free-fall interrupt event to be recognized (Figure 14. Free-fall interrupt).

Figure 14. **Free-fall interrupt**



The free-fall interrupt signal can be enabled by setting the INTERRUPTS_ENABLE bit in the TAP_CFG register to 1 and can be driven to the two interrupt pins by setting the INT1_FF bit of the MD1_CFG register to 1 or the INT2_FF bit of the MD2_CFG register to 1; it can also be checked by reading the FF_IA bit of the WAKE_UP_SRC register.

If latched mode is disabled (LIR bit of TAP_CFG is set to 0), the interrupt signal is automatically reset when the free-fall condition is no longer verified. If latched mode is enabled and the free-fall interrupt signal is driven to the interrupt pins, once a free-fall event has occurred and the interrupt pin is asserted, it must be reset by reading the WAKE_UP_SRC register. If latched mode is enabled but the interrupt signal is not driven to the interrupt pins, the latch feature does not take effect.

The FREE_FALL register used to configure the threshold parameter; the unsigned threshold value is related to the value of the FF_THS[2:0] field value as indicated in Table 31. Free-fall threshold LSB value. The values given in this table are valid for each accelerometer full-scale value.

Table 31. **Free-fall threshold LSB value**

FREE_FALL - FF_THS[2:0]	Threshold LSB value [mg]
000	156
001	219
010	250

FREE_FALL - FF_THS[2:0]	Threshold LSB value [mg]
011	312
100	344
101	406
110	469
111	500

Duration time is measured in N/ODR_XL, where N is the content of the FF_DUR[5:0] field of the FREE_FALL / WAKE_UP_DUR registers and ODR_XL is the accelerometer data rate.

A basic SW routine for free-fall event recognition is given below.

1.	Write 60h to CTRL1_XL	// Turn on the accelerometer // ODR_XL = 416 Hz, FS_XL = ±2 *g*
2.	Write 81h to TAP_CFG	// Enable interrupts and latch interrupt
3.	Write 00h to WAKE_UP_DUR	// Set event duration (FF_DUR5 bit)
4.	Write 33h to FREE_FALL	// Set FF threshold (FF_THS[2:0] = 011b) // Set six samples event duration (FF_DUR[5:0] = 000110b)
5.	Write 10h to MD1_CFG	// FF interrupt driven to INT1 pin

The sample code exploits a threshold set to 312 m*g* for free-fall recognition and the event is notified by hardware through the INT1 pin. The FF_DUR[5:0] field of the FREE_FALL / WAKE_UP_DUR registers is configured like this to ignore events that are shorter than 6/ODR_XL = 6/412 Hz ~= 15 msec in order to avoid false detections.

5.3 Wake-up interrupt

In the ISM330DLC device the wake-up feature can be implemented using either the slope filter (see Section 3.7.1 Accelerometer slope filter for more details) or the high-pass digital filter, as illustrated in Figure 2. Accelerometer filtering chain (Mode 1/2/3). The filter to be applied can be selected using the SLOPE_FDS bit of the TAP_CFG register: if this bit is set to 0 (default value), the slope filter is used; if it’s set to 1, the HPF digital filter is used. If Mode 4 is enabled, the wake-up feature is implemented using the slope filter, regardless of the value of the SLOPE_FDS bit.

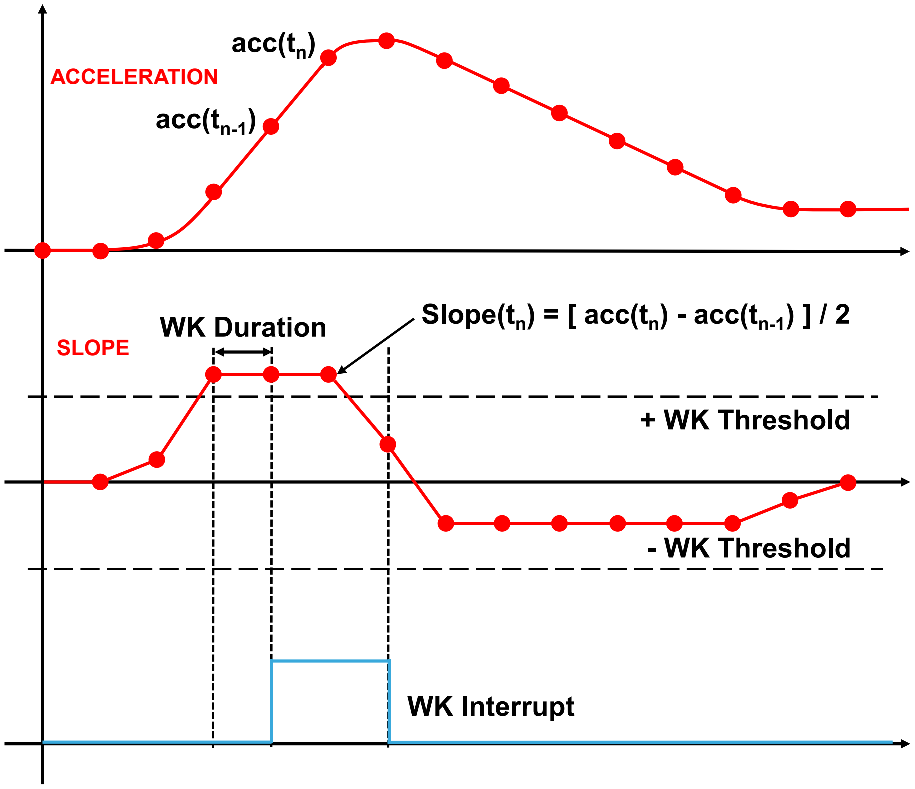
The wake-up interrupt signal is generated if a certain number of consecutive filtered data exceed the configured threshold (Figure 15. Wake-up interrupt (using the slope filter)).

The unsigned threshold value is defined using the WK_THS[5:0] bits of the WAKE_UP_THS register; the value of 1 LSB of these 6 bits depends on the selected accelerometer full scale: 1 LSB = (FS_XL)/(2^6). The threshold is applied to both positive and negative data: for wake-up interrupt generation, the absolute value of the filtered data must be bigger than the threshold.

The duration parameter defines the minimum duration of the wake-up event to be recognized; its value is set using the WAKE_DUR[1:0] bits of the WAKE_UP_DUR register: 1 LSB corresponds to 1/ODR_XL time, where ODR_XL is the accelerometer output data rate. It is important to appropriately define the duration parameter to avoid unwanted wake-up interrupts due to spurious spikes of the input signal.

This interrupt signal can be enabled by setting the INTERRUPTS_ENABLE bit in TAP_CFG register to 1 and can be driven to the two interrupt pins setting to 1 the INT1_WU bit of the MD1_CFG register or the INT2_WU bit of the MD2_CFG register; it can also be checked by reading the WU_IA bit of the WAKE_UP_SRC register. The X_WU, Y_WU, Z_WU bits of the WAKE_UP_SRC register indicate which axis has triggered the wake-up event.

Figure 15. **Wake-up interrupt (using the slope filter)**



If latch mode is disabled (LIR bit of TAP_CFG is set to 0), the interrupt signal is automatically reset when the filtered data falls below the threshold. If latch mode is enabled and the wake-up interrupt signal is driven to the interrupt pins, once a wake-up event has occurred and the interrupt pin is asserted, it must be reset by reading the WAKE_UP_SRC register. If the latch mode is enabled but the interrupt signal is not driven to the interrupt pins, the latch feature does not take effect.

A basic SW routine for wake-up event recognition using the high-pass digital filter is given below.

1.	Write 60h to CTRL1_XL	// Turn on the accelerometer // ODR_XL = 416 Hz, FS_XL = ±2 *g*
2.	Write 90h to TAP_CFG	// Enable interrupts and apply high-pass digital filter; latched // mode disabled
3.	Write 00h to WAKE_UP_DUR	// No duration
4.	Write 02h to WAKE_UP_THS	// Set wake-up threshold
5.	Write 20h to MD1_CFG	// Wake-up interrupt driven to INT1 pin

Since the duration time is set to zero, the wake-up interrupt signal is generated for each X,Y,Z filtered data exceeding the configured threshold. The WK_THS field of the WAKE_UP_THS register is set to 000010b, therefore the wake-up threshold is 62.5 m*g* (= 2 * FS_XL / 2^6).

Since the wake-up functionality is implemented using the slope/high-pass digital filter, it is necessary to consider the settling time of the filter just after this functionality is enabled. For example, when using the slope filter (but a similar consideration can be done for the high-pass digital filter usage) the wake-up functionality is based on the comparison of the threshold value with half of the difference of the acceleration of the current (x,y,z) sample and the previous one (refer to Section 3.7.1 Accelerometer slope filter).

At the very first sample, the slope filter output is calculated as half of the difference of the current sample [e.g. (x,y,z) = (0,0,1*g*)] with the previous one which is (x,y,z) = (0,0,0) since it doesn't exist. For this reason, on the z-axis the first output value of the slope filter is (1*g* - 0)/2 = 500 m*g* and it could be higher than the threshold value in which case a spurious interrupt event is generated. The interrupt signal is kept high for 1 ODR then it goes low.

In order to avoid this spurious interrupt generation, multiple solutions are possible. Hereafter are three alternative solutions (for the slope filter case):

a. Ignore the first generated wake-up signal;

b. Add a wait time higher than 1 ODR before driving the interrupt signal to the INT1/2 pin;

c. Initially set a higher ODR (833 Hz) so the first 2 samples are generated in a shorter period of time, reducing the slope filter latency time, then set the desired ODR (e.g. 12.5 Hz) and drive the interrupt signal on the pin, as indicated in the procedure below:

1.	Write 00h to WAKE_UP_DUR	// No duration
2.	Write 02h to WAKE_UP_THS	// Set wake-up threshold
3.	Write 80h to TAP_CFG	// Enable interrupts and apply slope filter; latch mode disabled
4.	Write 70h to CTRL1_XL	// Turn on the accelerometer // ODR_XL = 833 Hz, FS_XL = ±2 *g*
5.	Wait 4 ms	// Insert (reduced) wait time
6.	Write 10h to CTRL1_XL	// ODR_XL = 12.5 Hz
7.	Write 20h to MD1_CFG	// Wake-up interrupt driven to INT1 pin

5.4 6D/4D orientation detection

The ISM330DLC device provides the capability to detect the orientation of the device in space, enabling easy implementation of energy-saving procedures and automatic image rotation for mobile devices.

5.4.1 6D orientation detection

Six orientations of the device in space can be detected; the interrupt signal is asserted when the device switches from one orientation to another. The interrupt is not re-asserted as long as the position is maintained.

6D interrupt is generated when, for two consecutive samples, only one axis exceeds a selected threshold and the acceleration values measured from the other two axes are lower than the threshold: the ZH, ZL, YH, YL, XH, XL bits of the D6D_SRC register indicate which axis has triggered the 6D event.

In more detail:

Table 32. **D6D_SRC register**

b7	b6	b5	b4	b3	b2	b1	b0
DEN_ DRDY	D6D_IA	ZH	ZL	YH	YL	XH	XL

- D6D_IA is set high when the device switches from one orientation to another.
- ZH (YH, XH) is set high when the face perpendicular to the Z (Y, X) axis is almost flat and the acceleration measured on the Z (Y, X) axis is positive and in the absolute value bigger than the threshold.
- ZL (YL, XL) is set high when the face perpendicular to the Z (Y, X) axis is almost flat and the acceleration measured on the Z (Y, X) axis is negative and in the absolute value bigger than the threshold.

The SIXD_THS[1:0] bits of the TAP_THS_6D register are used to select the threshold value used to detect the change in device orientation. The threshold values given in Table 33. Threshold for 4D/6D function are valid for each accelerometer full-scale value.

Table 33. Threshold for 4D/6D function

SIXD_THS[1:0]	Threshold value [degrees]
00	80
01	70
10	60
11	50

The low-pass filter LPF2 can also be used in 6D functionality by setting the LOW_PASS_ON_6D bit of the CTRL8_XL register to 1. If Mode 4 is enabled, the LFP2 is not applied to the 6D feature, regardless of the value of the LOW_PASS_ON_6D bit.

This interrupt signal can be enabled by setting the INTERRUPTS_ENABLE bit in the TAP_CFG register to 1 and can be driven to the two interrupt pins by setting to 1 the INT1_6D bit of the MD1_CFG register or the INT2_6D bit of the MD2_CFG register; it can also be checked by reading the D6D_IA bit of the D6D_SRC register.

If latched mode is disabled (LIR bit of TAP_CFG is set to 0), the interrupt signal is active only for 1/ODR_XL[s] then it is automatically disserted (ODR_XL is the accelerometer output data rate). If latched mode is enabled and the 6D interrupt signal is driven to the interrupt pins, once an orientation change has occurred and the interrupt pin is asserted, a reading of the D6D_SRC register clears the request and the device is ready to recognize a different orientation. If latched mode is enabled but the interrupt signal is not driven to the interrupt pins, the latch feature does not take effect.

Referring to the six possible cases illustrated in Figure 16. 6D recognized orientations, the content of the D6D_SRC register for each position is shown in Table 34. D6D_SRC register in 6D positions.

Figure 16. **6D recognized orientations**

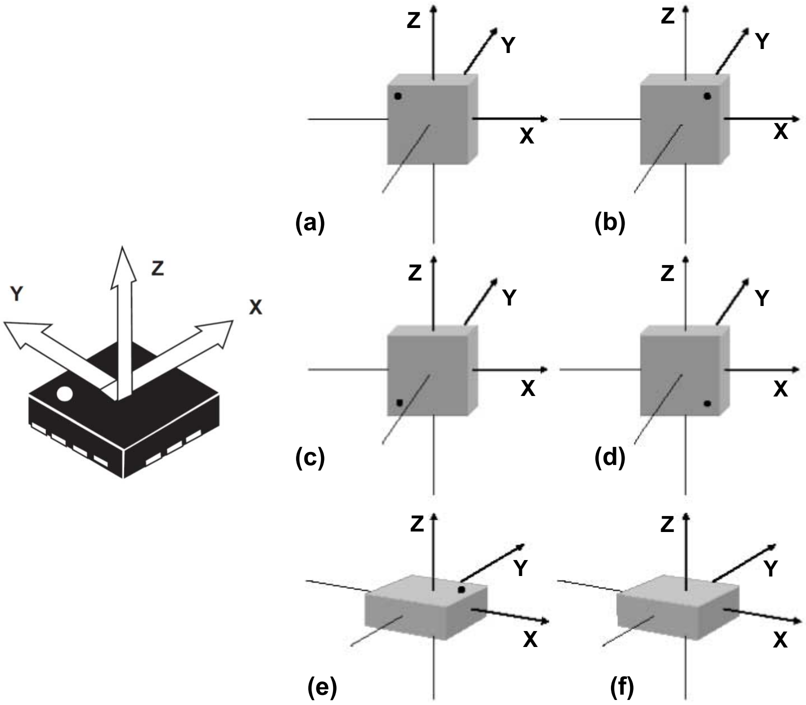


Table 34. **D6D_SRC register in 6D positions**

Case	D6D_IA	ZH	ZH	YH	YL	XH	XL
(a)	1	0	0	1	0	0	0
(b)	1	0	0	0	0	0	1
(c)	1	0	0	0	0	1	0
(d)	1	0	0	0	1	0	0
(e)	1	1	0	0	0	0	0
(f)	1	0	1	0	0	0	0

A basic SW routine for 6D orientation detection is as follows.

1.	Write 60h to CTRL1_XL	// Turn on the accelerometer // ODR_XL = 416 Hz, FS_XL = ±2 *g*
2.	Write 80h to TAP_CFG	// Enable interrupts; latched mode disabled
3.	Write 40h to TAP_THS_6D	// Set 6D threshold (SIXD_THS[1:0] = 10b = 60 degrees)
4.	Write 01h to CTRL8_XL	// Enable LPF2 filter to 6D functionality
5.	Write 04h to MD1_CFG	// 6D interrupt driven to INT1 pin

5.4.2 4D orientation detection

The 4D direction function is a subset of the 6D function especially defined to be implemented in mobile devices for portrait and landscape computation. It can be enabled by setting the D4D_EN bit of the TAP_THS_6D register to 1. In this configuration, the Z-axis position detection is disabled, therefore reducing position recognition to cases (a), (b), (c), and (d) of Table 34. D6D_SRC register in 6D positions.

5.5 Single-tap and double-tap recognition

The single-tap and double-tap recognition functions featured in the ISM330DLC help to create a man-machine interface with little software loading. The device can be configured to output an interrupt signal on a dedicated pin when tapped in any direction.

If the sensor is exposed to a single input stimulus, it generates an interrupt request on the inertial interrupt pin INT1 and/or INT2. A more advanced feature allows the generation of an interrupt request when a double input stimulus with programmable time between the two events is recognized, enabling a mouse button-like function.

In the ISM330DLC device the single-tap and double-tap recognition functions use the slope between two consecutive acceleration samples to detect the tap events; the slope data is calculated using the following formula:

$$\text{slope}(t_n) = [\ \text{acc}(t_n) - \text{acc}(t_{n-1})\ ] / 2$$

This function can be fully programmed by the user in terms of expected amplitude and timing of the slope data by means of a dedicated set of registers.

Single and double-tap recognition work independently of the selected output data rate. Recommended accelerometer ODRs for these functions are 416 Hz and 833 Hz.

In order to enable the single-tap and double-tap recognition functions it is necessary to set the INTERRUPTS_ENABLE bit in TAP_CFG register to 1.

5.5.1 Single tap

If the device is configured for single-tap event detection, an interrupt is generated when the slope data of the selected channel exceeds the programmed threshold, and returns below it within the Shock time window.

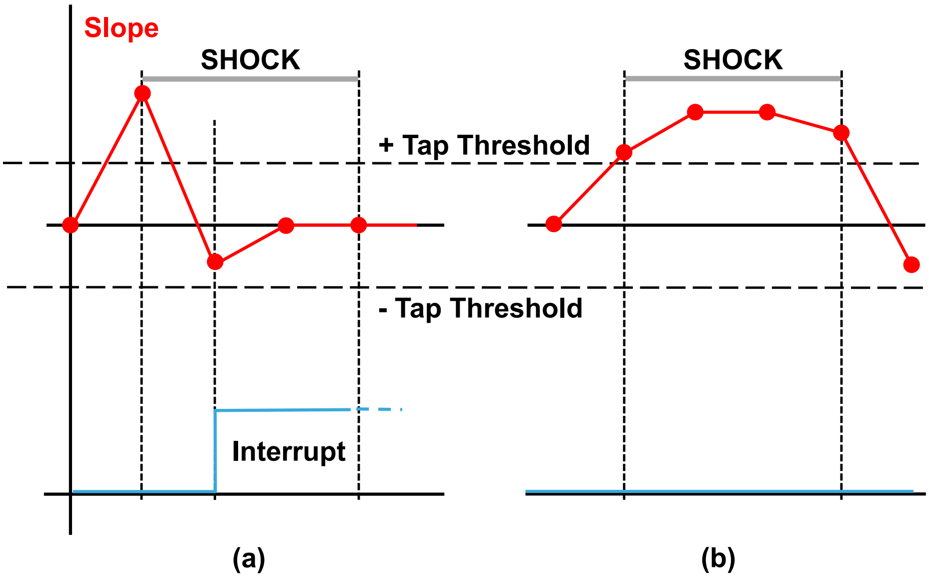
In the single-tap case, if the LIR bit of the TAP_CFG register is set to 0, the interrupt is kept active for the duration of the Quiet window.

In order to enable the latch feature on the single-tap interrupt signal, both the LIR bit and the INT1_DOUBLE_TAP (or INT2_DOUBLE_TAP) bit of MD1_CFG (MD2_CFG) have to be set to 1: the interrupt is kept active until the TAP_SRC register is read.

The SINGLE_DOUBLE_TAP bit of WAKE_UP_THS has to be set to 0 in order to enable single-tap recognition only.

In case (a) of Figure 17. Single-tap event recognition the single-tap event has been recognized, while in case (b) the tap has not been recognized because the slope data falls below the threshold after the Shock time window has expired.

Figure 17. Single-tap event recognition



5.5.2 Double tap

If the device is configured for double-tap event detection, an interrupt is generated when, after a first tap, a second tap is recognized. The recognition of the second tap occurs only if the event satisfies the rules defined by the Shock, the Latency and the Duration time windows.

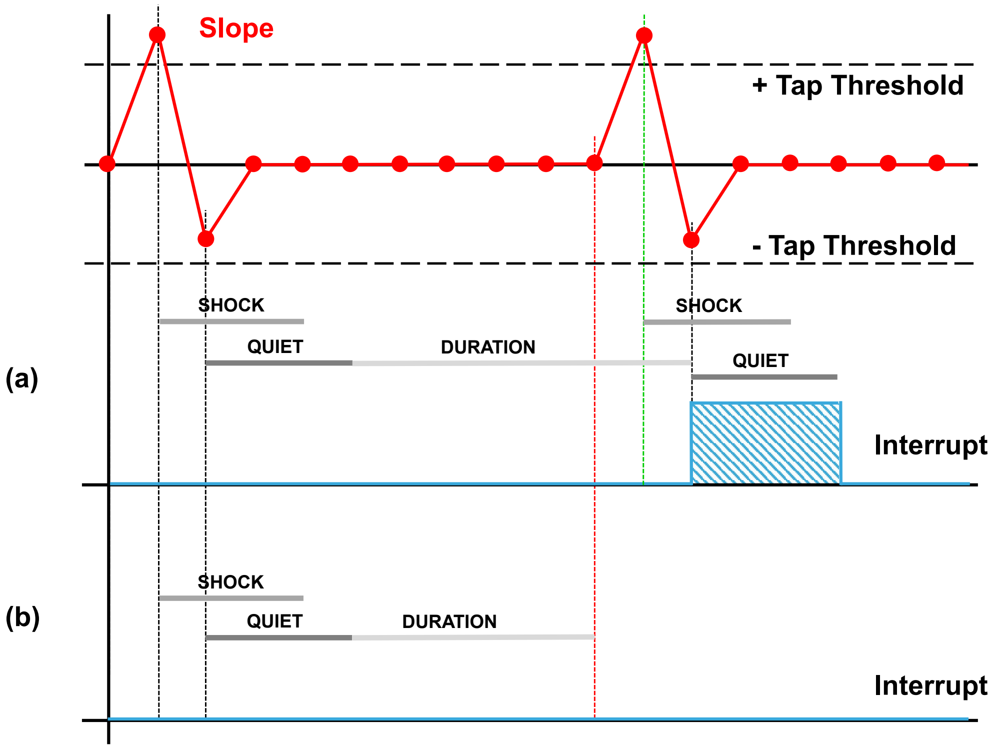
In particular, after the first tap has been recognized, the second tap detection procedure is delayed for an interval defined by the Quiet time. This means that after the first tap has been recognized, the second tap detection procedure starts only if the slope data exceeds the threshold after the Quiet window but before the Duration window has expired. In case (a) of Figure 18. Double-tap event recognition (LIR bit = 0), a double-tap event has been correctly recognized, while in case (b) the interrupt has not been generated because the slope data exceeds the threshold after the window interval has expired.

Once the second tap detection procedure is initiated, the second tap is recognized with the same rule as the first: the slope data must return below the threshold before the Shock window has expired.

It is important to appropriately define the Quiet window to avoid unwanted taps due to spurious bouncing of the input signal.

In the double-tap case, if the LIR bit of the TAP_CFG register is set to 0, the interrupt is kept active for the duration of the Quiet window. If the LIR bit is set to 1, the interrupt is kept active until the TAP_SRC register is read.

Figure 18. **Double-tap event recognition (LIR bit = 0)**



5.5.3 Single-tap and double-tap recognition configuration

The ISM330DLC device can be configured to output an interrupt signal when tapped (once or twice) in any direction: the TAP_X_EN, TAP_Y_EN and TAP_Z_EN bits of the TAP_CFG register must be set to 1 to enable the tap recognition on the X, Y, Z directions, respectively. In addition, the INTERRUPTS_ENABLE bit of the TAP_CFG register has to be set to 1.

Configurable parameters for tap recognition functionality are the tap threshold and the Shock, Quiet and Duration time windows.

The TAP_THS[4:0] bits of the TAP_THS_6D register are used to select the unsigned threshold value used to detect the tap event. The value of 1 LSB of these 5 bits depends on the selected accelerometer full scale: 1 LSB = (FS_XL)/(2^5). The unsigned threshold is applied to both positive and negative slope data.

Note: Tap threshold (in mg) set through the TAP_THS[4:0] bits of the TAP_THS_6D register must be higher than the wake-up threshold (in mg) set through the WK_THS[5:0] bits of the WAKE_UP_THS register.

The Shock time window defines the maximum duration of the overcoming threshold event: the acceleration must return below the threshold before the Shock window has expired, otherwise the tap event is not detected. The SHOCK[1:0] bits of the INT_DUR2 register are used to set the Shock time window value: the default value of these bits is 00b and corresponds to 4/ODR_XL time, where ODR_XL is the accelerometer output data rate. If the SHOCK[1:0] bits are set to a different value, 1 LSB corresponds to 8/ODR_XL time.

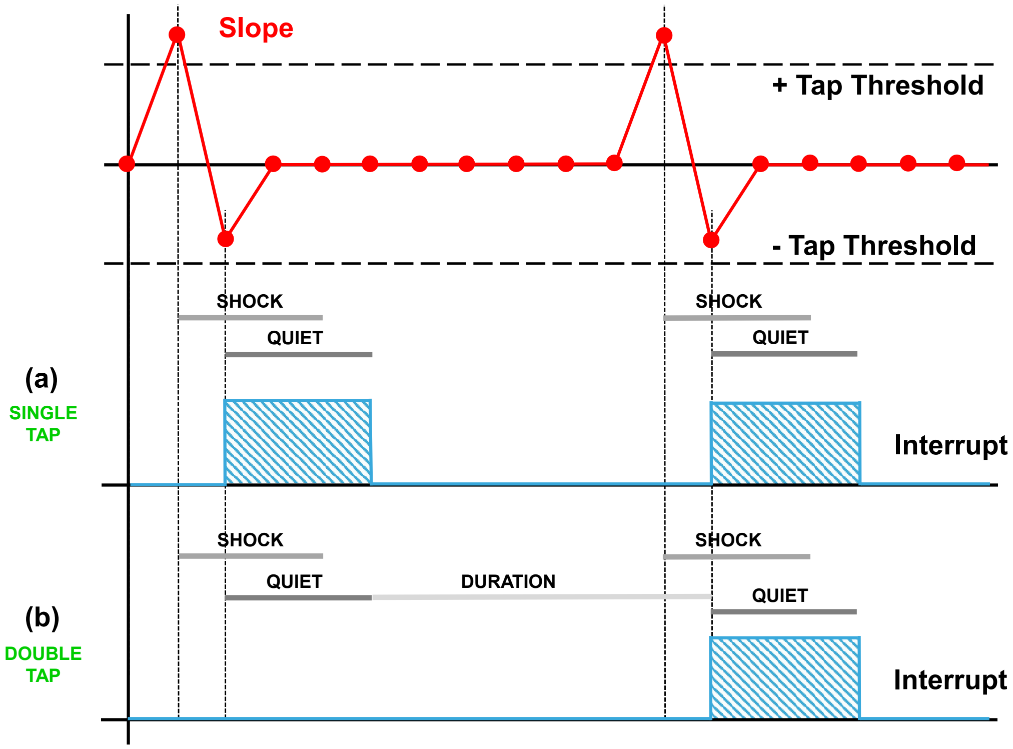
In the double-tap case, the Quiet time window defines the time after the first tap recognition in which there must not be any overcoming threshold event. When latched mode is disabled (LIR bit of TAP_CFG is set to 0), the Quiet time also defines the length of the interrupt pulse (in both single and double-tap case). The QUIET[1:0] bits of the INT_DUR2 register are used to set the Quiet time window value: the default value of these bits is 00b and corresponds to 2/ODR_XL time, where ODR_XL is the accelerometer output data rate. If the QUIET[1:0] bits are set to a different value, 1 LSB corresponds to 4/ODR_XL time.

In the double-tap case, the Duration time window defines the maximum time between two consecutive detected taps. The Duration time period starts just after the completion of the Quiet time of the first tap. The DUR[3:0] bits of the INT_DUR2 register are used to set the Duration time window value: the default value of these bits is 0000b and corresponds to 16/ODR_XL time, where ODR_XL is the accelerometer output data rate. If the DUR[3:0] bits are set to a different value, 1 LSB corresponds to 32/ODR_XL time.

Figure 19. Single and double-tap recognition (LIR bit = 0) illustrates a single-tap event (a) and a double-tap event (b). These interrupt signals can be driven to the two interrupt pins by setting to 1 the INT1_SINGLE_TAP bit of the MD1_CFG register or the INT2_SINGLE_TAP bit of the MD2_CFG register for the single-tap case, and setting to 1 the INT1_DOUBLE_TAP bit of the MD1_CFG register or the INT2_DOUBLE_TAP bit of the MD2_CFG register for the double-tap case.

No single/double-tap interrupt is generated if the accelerometer is in Inactivity status (see Section 5.6 Activity/ Inactivity recognition for more details).

Figure 19. **Single and double-tap recognition (LIR bit = 0)**



Tap interrupt signals can also be checked by reading the TAP_SRC (1Ch) register, described in Table 35. TAP_SRC register.

Table 35. **TAP_SRC register**

b7	b6	b5	b4	b3	b2	b1	b0
0	TAP_IA	SINGLE _TAP	DOUBLE _TAP	TAP_ SIGN	X_TAP	Y_TAP	Z_TAP

- TAP_IA is set high when a single-tap or double-tap event has been detected.
- SINGLE_TAP is set high when a single tap has been detected.
- DOUBLE_TAP is set high when a double tap has been detected.
- TAP_SIGN indicates the acceleration sign when the tap event is detected. It is set low in case of positive sign and it is set high in case of negative sign.
- X_TAP (Y_TAP, Z_TAP) is set high when the tap event has been detected on the X (Y, Z) axis.

Single and double-tap recognition works independently. Setting the SINGLE_DOUBLE_TAP bit of the WAKE_UP_THS register to 0, only the single-tap recognition is enabled: double-tap recognition is disabled and cannot be detected. When the SINGLE_DOUBLE_TAP is set to 1, both single and double-tap recognition are enabled.

If latched mode is enabled and the interrupt signal is driven to the interrupt pins, the value assigned to SINGLE_DOUBLE_TAP also affects the behavior of the interrupt signal: when it is set to 0, the latched mode is applied to the single-tap interrupt signal; when it is set to 1, the latched mode is applied to the double-tap interrupt signal only. The latched interrupt signal is kept active until the TAP_SRC register is read. If latched mode is enabled but the interrupt signal is not driven to the interrupt pins, the latch feature does not take effect.

5.5.4 Single-tap example

A basic SW routine for single-tap detection is given below.

```
1.  Write 60h to CTRL1_XL        // Turn on the accelerometer
                                 // ODR_XL = 416 Hz, FS_XL = ±2 g
2.  Write 8Eh to TAP_CFG         // Enable interrupts and tap detection on X, Y, Z-axis
3.  Write 89h to TAP_THS_6D      // Set tap threshold
4.  Write 06h to INT_DUR2        // Set Quiet and Shock time windows
5.  Write 00h to WAKE_UP_THS     // Only single-tap enabled (SINGLE_DOUBLE_TAP = 0)
6.  Write 40h to MD1_CFG         // Single-tap interrupt driven to INT1 pin
```

In this example the TAP_THS field of the TAP_THS_6D register is set to 01001b, therefore the tap threshold is 562.5 m*g* (= 9 * FS_XL / 2^5).

The SHOCK field of the INT_DUR2 register is set to 10b: an interrupt is generated when the slope data exceeds the programmed threshold, and returns below it within 38.5 ms (= 2 * 8 / ODR_XL) corresponding to the Shock time window.

The QUIET field of the INT_DUR2 register is set to 01b: since latched mode is disabled, the interrupt is kept high for the duration of the Quiet window, therefore 9.6 ms (= 1 * 4 / ODR_XL).

5.5.5 Double-tap example

A basic SW routine for double-tap detection is given below.

```
1.  Write 60h to CTRL1_XL        // Turn on the accelerometer
                                 // ODR_XL = 416 Hz, FS_XL = ±2 g
2.  Write 8Eh to TAP_CFG         // Enable interrupts and tap detection on X, Y, Z-axis
3.  Write 8Ch to TAP_THS_6D      // Set tap threshold
4.  Write 7Fh to INT_DUR2        // Set Duration, Quiet and Shock time windows
5.  Write 80h to WAKE_UP_THS     // Single & double-tap enabled (SINGLE_DOUBLE_TAP = 1)
6.  Write 08h to MD1_CFG         // Double-tap interrupt driven to INT1 pin
```

In this example the TAP_THS field of the TAP_THS_6D register is set to 01100b, therefore the tap threshold is 750 m*g* (= 12 * FS_XL / 2^5).

For interrupt generation, during the first and the second tap the slope data must return below the threshold before the Shock window has expired. The SHOCK field of the INT_DUR2 register is set to 11b, therefore the Shock time is 57.7 ms (= 3 * 8 / ODR_XL).

For interrupt generation, after the first tap recognition there must not be any slope data overthreshold during the Quiet time window. Furthermore, since latched mode is disabled, the interrupt is kept high for the duration of the Quiet window. The QUIET field of the INT_DUR2 register is set to 11b, therefore the Quiet time is 28.8 ms (= 3 * 4 / ODR_XL).

For the maximum time between two consecutive detected taps, the DUR field of the INT_DUR2 register is set to 0111b, therefore the Duration time is 538.5 ms (= 7 * 32 / ODR_XL).

5.6 Activity/Inactivity recognition

The Activity/Inactivity recognition function allows reducing system power consumption and developing new smart applications.

When the Activity/Inactivity recognition function is activated, the ISM330DLC device is able to automatically decrease the accelerometer sampling rate to 12.5 Hz, increasing the accelerometer ODR and bandwidth as soon as the wake-up interrupt event has been detected. In the ISM330DLC this feature can be extended to the gyroscope, with three possible options:

- Gyroscope configurations do not change;
- Gyroscope enters in Sleep mode;
- Gyroscope enters in Power-Down mode.

With this feature the system may be efficiently switched from low-power consumption to full performance and vice-versa depending on user-selectable acceleration events, thus ensuring power saving and flexibility.

The maximum allowed accelerometer ODR (configurable through the ODR_XL [3:0] bits of the CTRL1_XL register) for using the Activity/Inactivity feature is 3.3 kHz.

The Activity/Inactivity recognition function is enabled by setting the INTERRUPTS_ENABLE bit to 1 and configuring the INACT_EN bits of the TAP_CFG register. Possible configurations of the inactivity event are summarized in Table 36. Inactivity event configuration.

Table 36. **Inactivity event configuration**

INACT_EN[1:0]	Accelerometer	Gyroscope
00	Inactivity event disabled	Inactivity event disabled
01	XL ODR = 12.5 Hz (Low-Power mode)	Gyro configuration unchanged
10	XL ODR = 12.5 Hz (Low-Power mode)	Gyro in Sleep mode
11	XL ODR = 12.5 Hz (Low-Power mode)	Gyro in Power-Down mode

In the ISM330DLC device the Activity/Inactivity recognition function can be implemented using either the slope filter (see Section 3.7.1 Accelerometer slope filter for more details) or the high-pass digital filter, as illustrated in Figure 2. Accelerometer filtering chain (Mode 1/2/3). The filter to be applied can be selected using the SLOPE_FDS bit of the TAP_CFG register: if this bit is set to 0 (default value), the slope filter is used; if it is set to 1, the high-pass digital filter is used. If Mode 4 is enabled, the Activity/Inactivity recognition feature is implemented using the slope filter, regardless of the value of SLOPE_FDS bit.

This function can be fully programmed by the user in terms of expected amplitude and timing of the filtered data by means of a dedicated set of registers (Figure 20. Activity/Inactivity recognition (using the slope filter)).

The unsigned threshold value is defined using the WK_THS[5:0] bits of the WAKE_UP_THS register; the value of 1 LSB of these 6 bits depends on the selected accelerometer full scale: 1 LSB = (FS_XL)/(2^6). The threshold is applied to both positive and negative filtered data.

When a certain number of consecutive X,Y,Z filtered data is smaller than the configured threshold, the ODR_XL [3:0] bits of the CTRL1_XL register are bypassed (Inactivity) and the accelerometer is internally set to 12.5 Hz although the content of CTRL1_XL is left untouched. The gyroscope behavior varies according to the configuration of the INACT_EN bits of the TAP_CFG register. The duration of the Inactivity status to be recognized is defined by the SLEEP_DUR[3:0] bits of the WAKE_UP_DUR register: 1 LSB corresponds to 512/ODR_XL time, where ODR_XL is the accelerometer output data rate.

When the Inactivity status is detected, the interrupt is set high for 1/ODR_XL[s] period then it is automatically deasserted.

When a single sample of X,Y,Z filtered data on one axis becomes bigger than the threshold, the CTRL1_XL register settings are immediately restored (Activity) and the gyroscope is restored to the previous state.

When the Activity status is detected, the interrupt is set high for 1/ODR_XL[s] period then it is automatically deasserted.

Once the Activity/Inactivity detection function is enabled, the status can be driven to the two interrupt pins by setting to 1 the INT1_INACT_STATE bit of the MD1_CFG register or the INT2_INACT_STATE bit of the MD2_CFG register; it can also be checked by reading the SLEEP_STATE_IA bit of the WAKE_UP_SRC register.

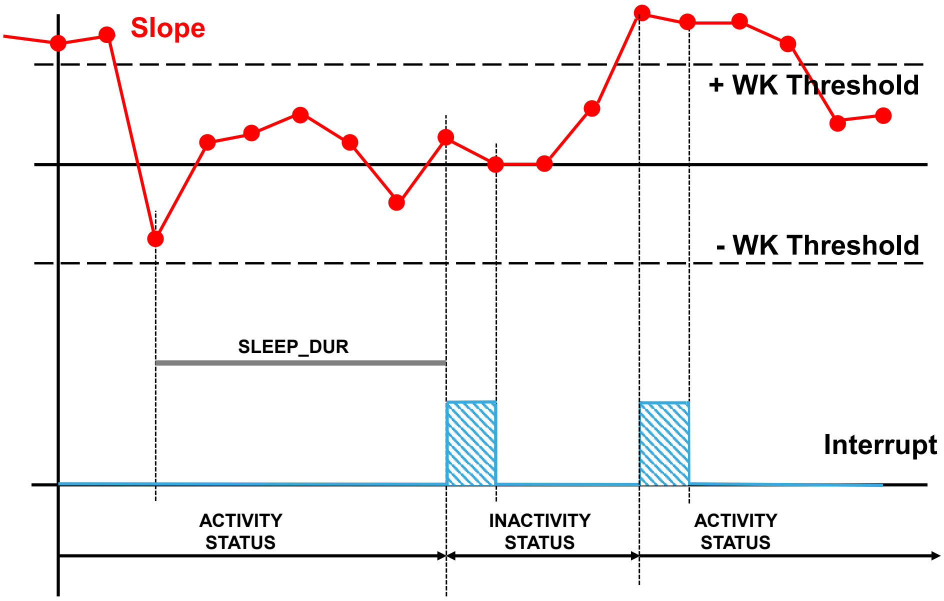


Figure 20. **Activity/Inactivity recognition (using the slope filter)**

A basic SW routine for Activity/Inactivity detection is as follows:

1.	Write 50h to CTRL1_XL	// Turn on the accelerometer // ODR_XL = 208 Hz, FS_XL = ±2 *g*
2.	Write 40h to CTRL2_G	// Turn on the gyroscope // ODR_G = 104 Hz, FS_G = ±250 dps
3.	Write 02h to WAKE_UP_DUR	// Set duration for Inactivity detection
4.	Write 02h to WAKE_UP_THS	// Set Activity/Inactivity threshold

5.	Write E0h to TAP_CFG	// Enable interrupts
		// Inactivity configuration: acc to 12.5 LP, gyro to Power-Down
		// Enable slope filter

6.	Write 80h to MD1_CFG	// Activity/Inactivity interrupt driven to INT1 pin

In this example the WK_THS field of the WAKE_UP_THS register is set to 000010b, therefore the Activity/Inactivity threshold is 62.5 m*g* (= 2 * FS_XL / 2^6).

Before Inactivity detection, the X,Y,Z slope data must be smaller than the configured threshold for a period of time defined by the SLEEP_DUR field of the WAKE_UP_DUR register: this field is set to 0010b, corresponding to 4.92 s (= 2 * 512 / ODR_XL). After this period of time has elapsed, the accelerometer ODR is internally set to 12.5 Hz and the gyroscope is internally set to Power-Down mode.

The Activity status is detected and the CTRL1_XL register settings immediately restored and the gyroscope is turned on if the slope data of (at least) one axis are bigger than the threshold.

5.7 Boot status

After the device is powered up, the ISM330DLC performs a 15 ms boot procedure to load the trimming parameters. After the boot is completed, both the accelerometer and the gyroscope are automatically configured in Power-Down mode. During the boot time the registers are not accessible.

After power up, the trimming parameters can be re-loaded by setting the BOOT bit of the CTRL3_C register to 1.

No toggle of the device power lines is required and the content of the device control registers is not modified, so the device operating mode doesn't change after boot. If the reset to the default value of the control registers is required, it can be performed by setting the SW_RESET bit of the CTRL3_C register to 1. The SW_RESET procedure can take 50 µs; the status of reset is signaled by the status of the SW_RESET bit of the CTRL3_C register: once the reset is completed, this bit is automatically set low.

The boot status signal is driven to the INT1 interrupt pin by setting the INT1_BOOT bit of the INT1_CTRL register to 1: this signal is set high while the boot is running and it is set low again at the end of the boot procedure.

The reboot flow is as follows:

1. Set the gyroscope in Power-Down mode;
2. Set the accelerometer in High-Performance mode;
3. Set to 1 the BOOT bit of the CTRL3_C register;
4. Wait 15 ms.

Reset flow is as follows:

1. Set the gyroscope in Power-Down mode;
2. Set the accelerometer in High-Performance mode;
3. Set to 1 the SW_RESET bit of the CTRL3_C register;
4. Wait 50 µs (or wait until the SW_RESET bit of the CTRL3_C register returns to 0).

In order to avoid conflicts, the reboot and the sw reset must not be executed at the same time (do not set to 1 at the same time both the BOOT bit and SW_RESET bit of CTRL3_C register). The above flows must be performed serially.

5.8 Relative tilt

The tilt function allows detecting when an activity change occurs (e.g. when phone is in a front pocket and the user goes from sitting to standing or from standing to sitting): in the ISM330DLC device it has been implemented in hardware using only the accelerometer.

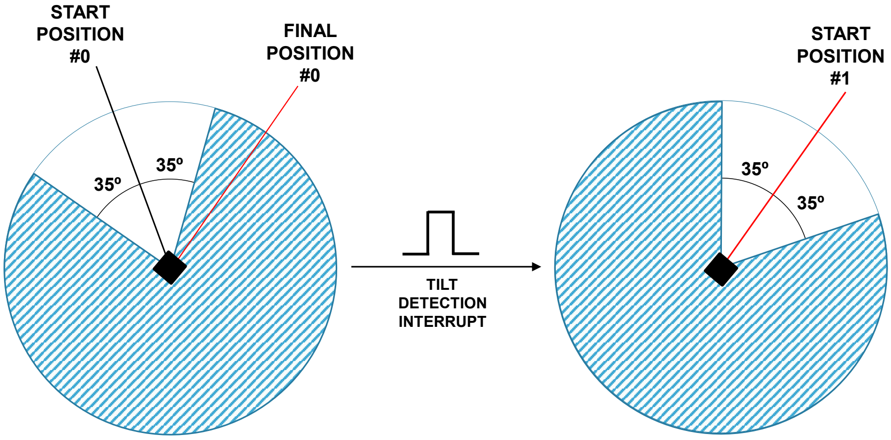
In order to enable the tilt detector it is necessary to set to 1 both the FUNC_EN and the TILT_EN bits of the CTRL10_C register.

If the device is configured for tilt event detection, an interrupt is generated when the device is tilted by an angle greater than 35 degrees from the start position. The start position is defined as the position of the device when the tilt detection is enabled or the position of the device when the last tilt interrupt was generated.

After this function is enabled, for the generation of the first tilt interrupt the device should be continuously tilted by an angle greater than 35 degrees from the start position for a period of time of 2 seconds. After the first tilt interrupt is generated, the tilt interrupt signal is set high as soon as the device is tilted by an angle greater than 35 degrees from the position of the device corresponding to the last interrupt detection (no need to wait 2 seconds).

In the example shown in Figure 21. Tilt example tilt detection is enabled when the device orientation corresponds to “start position #0”: the first interrupt is generated if the device is rotated by an angle greater than 35 degrees from the start position and remains in the blue zone for a period of time of at least 2 seconds. After the first tilt detection interrupt is generated, the new start position (#1) corresponds to the position of the device when the previous interrupt was generated (final position #0), and the next interrupt signal will be generated as soon as the device is tilted by an angle greater than 35 degrees, entering the blue zone surrounding the start position #1.

Figure 21. Tilt example



This interrupt signal can be driven to the two interrupt pins by setting to 1 the INT1_TILT bit of the MD1_CFG register or the INT2_TILT bit of the MD2_CFG register; it can also be checked by reading the TILT_IA bit of the FUNC_SRC1 register.

If latched mode is disabled (LIR bit of TAP_CFG is set to 0), the interrupt signal generated by the tilt function is pulsed: the duration of the pulse observed on the interrupt pins is about 150 µs; the duration of the pulse observed on the TILT_IA bit of FUNC_SRC1 register is 1/26 Hz.

If latched mode is enabled (LIR bit of TAP_CFG is set to 1) and the interrupt signal is driven to the interrupt pins, once a tilt is detected, a reading of the FUNC_SRC1 register clears the request on both the pins and the TILT_IA bit of FUNC_SRC1 register, and the device is ready to recognize the next tilt event. If latched mode is enabled but the interrupt signal is not driven to the interrupt pins, the interrupt signal observed on the TILT_IA bit of the FUNC_SRC1 register is pulsed, with a fixed duration of 1/26 Hz.

The tilt function works at 26 Hz, so the accelerometer ODR must be set at a value of 26 Hz or higher.

Hereafter a basic SW routine which shows how to enable the tilt detection function:

```
1.    Write 20h to CTRL1_XL       // Turn on the accelerometer
                                  // ODR_XL = 26 Hz, FS_XL = ±2 g
2.    Write 0Ch to CTRL10_C       // Enable embedded functions
                                  // Enable tilt detection
3.    Write 02h to MD1_CFG        // Tilt detector interrupt driven to INT1 pin
```

5.9 Timestamp

Together with sensor data the ISM330DLC device can provide timestamp information.

If both the accelerometer and the gyroscope are in Power-Down mode, the timestamp counter does not work.

To enable this functionality the TIMER_EN bit of the CTRL10_C register has to be set to 1: the time step count is given by the concatenation of the TIMESTAMP_REG2 & TIMESTAMP_REG1 & TIMESTAMP_REG0 registers and is represented as a 24-bit unsigned number.

The timestamp resolution can be configured using the TIMER_HR bit of the WAKE_UP_DUR register: when this bit is set to 0, 1 LSB of the time step count corresponds to 6.4 ms (low-resolution mode); when this bit is set to 1, 1 LSB of the time step count corresponds to 25 μs (high-resolution mode).

When the maximum value 16777215 LSB (corresponding to FFFFFFh) is reached, the counter is automatically reset to 000000h and continues to count. The timer count can be reset to zero at any time by writing the reset value AAh in the TIMESTAMP_REG2 register.

An interrupt is generated around 1.638 seconds before timer saturation in both high-resolution mode (when the timer step count reaches the value FF0000h) and low-resolution mode (when the timer step count reaches the value FFFF00h). This interrupt signal can be driven to the INT1 pin by setting the INT1_TIMER bit of the MD1_CFG register to 1. Once the interrupt pin is asserted, it must be reset to 0 by writing AAh in the TIMESTAMP_REG2 register (the timer step count will also be reset).

The timestamp count can be stored in FIFO as a fourth data set (see Section 8.8 Timestamp data in FIFO for details).

The timestamp resolution has to be set before enabling the timestamp functionality; a basic SW routine is as follows:

```
                                            // Turn on the accelerometer
1.    Write 50h to CTRL1_XL
                                            // ODR_XL = 208 Hz, FS_XL = ±2 g
2.    Write 10h to WAKE_UP_DUR              // Timestamp resolution = 25 µs
3.    Write 20h to CTRL10_C                 // Enable timestamp count
4.    Write 01h to MD1_CFG                  // End counter interrupt driven to INT1 pin
```

When switching from a low timestamp resolution to a high resolution, the timer count must be reset as indicated in the example below:

```
                                             // Turn on the accelerometer
1.       Write 50h to CTRL1_XL
                                             // ODR_XL = 208 Hz, FS_XL = ±2 g
2.       Write 00h to WAKE_UP_DUR            // Timestamp resolution = 6.4 ms
3.       Write 20h to CTRL10_C               // Enable timestamp count
         ...
N        Write 10h to WAKE_UP_DUR            // Timestamp resolution = 25 µs
N+1      Write AAh to TIMESTAMP_REG2         // Reset timer counter
```

6 Mode 2 - Sensor hub mode

The hardware flexibility of the ISM330DLC allows connecting the pins with different mode connections to external sensors to expand functionalities such as adding a sensor hub. When sensor hub mode (Mode 2) is enabled, both the primary I^{2}C/SPI (3- and 4-wire) slave interface and the I^{2}C master interface for the connection of external sensors are available. Mode 2 connection mode is described in detail in the following paragraphs.

6.1 Sensor hub mode description

In sensor hub mode (Mode 2) up to 4 external sensors can be connected to the I^{2}C master interface of the ISM330DLC device. The sensor hub trigger signal can be synchronized with the accelerometer data-ready signal (up to 104 Hz); alternatively, an external signal connected to the INT2 pin can be used as the sensor hub trigger. In this second case, the maximum ODR supported for external sensors depends on the number of read / write operations that can be executed between two consecutive trigger signals.

On the sensor hub trigger signal, all the write and read I²C operations configured through the registers SLVx_ADD, SLVx_SUBADD, SLAVEx_CONFIG and DATAWRITE_SRC_MODE_SUB_SLV0 are performed sequentially from external sensor 0 to external sensor 3 (depending on the external sensors enabled through the Aux_sens_on[1:0] field in the SLAVE0_CONFIG register).

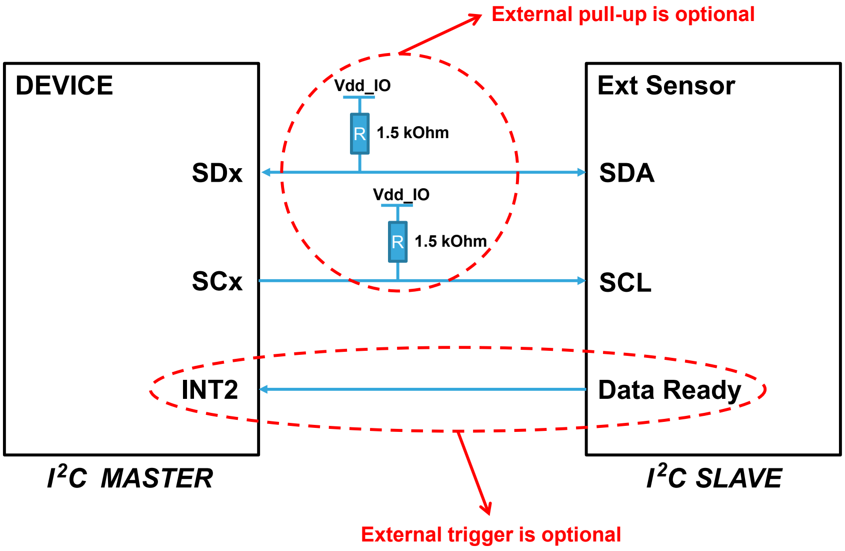
External sensor data can also be stored in FIFO with a configurable decimation factor (see Section 8 First-in first-out (FIFO) buffer for details).

If both the accelerometer and the gyroscope are in Power-Down mode, the sensor hub does not work.

All external sensors have to be connected in parallel to the SDx/SCx pins of the device, as illustrated in Figure 22. External sensor connections in Mode 2 for a single external sensor.

External pull-up resistors and the external trigger signal connection are optional and depend on the configuration of the registers.

Figure 22. External sensor connections in Mode 2



6.2 Sensor hub mode registers

In order to enable the embedded functionalities of the ISM330DLC , the FUNC_EN bit of the CTRL10_C register has to be set to 1; after enabling the embedded functionalities, the MASTER_CONFIG register has to be used for the configuration of the I^{2}C master interface.

A set of registers SLVx_ADD, SLVx_SUBADD, SLAVEx_CONFIG is dedicated to the configuration of the 4 slave interfaces associated to the 4 connectable external sensors. An additional register, DATAWRITE_SRC_MODE_SUB_SLV0, is associated to slave #0 only: it can be used to implement the writing and the source mode conditioned reading of the registers of the external sensor associated to slave #0.

Finally, 18 registers (from SENSORHUB1_REG to SENSORHUB18_REG) are available to store the data read from the external sensors.

6.2.1 CTRL10_C (19h)

Table 37. **CTRL10_C register**

b7	b6	b5	b4	b3	b2	b1	b0
X	0	X	X	X	FUNC_EN	X	X

- FUNC_EN must be set to 1 in order to enable the embedded functionalities of the ISM330DLC (tilt, ironing).

6.2.2 MASTER_CONFIG (1Ah)

This register is used to configure the I^{2}C master interface.

Table 38. **MASTER_CONFIG register**

b7	b6	b5	b4	b3	b2	b1	b0
DRDY_ON_INT1	X	0	START_CONFIG	PULL_UP_EN	PASS_THROUGH_MODE	X	MASTER_ON

- DRDY_ON_INT1 bit has to be set to 1 to drive the I^{2}C master Data-Ready signal on the INT1 pin (corresponding to the behavior of the SENSORHUB_END_OP bit of the FUNC_SRC1 register). Please refer to Section 6.2.3 FUNC_SRC1 (53h) for more details about the SENSORHUB_END_OP bit. If the DRDY_PULSED bit of the DRDY_PULSE_CFG register is set to 1, the I^{2}C master data-ready signal is pulsed with a duration of 150 µs.

The START_CONFIG bit selects the sensor hub trigger signal.

- When this bit is set to 0, the accelerometer sensor has to be active (not in Power-Down mode) and the sensor hub trigger signal is the accelerometer data-ready signal, with a frequency corresponding to the accelerometer ODR up to 104 Hz.
- When this bit is set to 1, at least one sensor between the accelerometer and the gyroscope has to be active and the sensor hub trigger signal is the INT2 pin; in fact, when both the MASTER_ON bit and START_CONFIG bit are set to 1, the INT2 pin is configured as an input signal. In this case, the INT2 pin has to be connected to the data-ready pin of the external sensor (Figure 22. External sensor connections in Mode 2) in order to trigger the reading/writing operations on the external sensor registers. The sensor hub interrupt from INT2 is 'high-level triggered' (not programmable).

Note: In case of external trigger signal usage (START_CONFIG=1), if the INT2 pin is connected to the Data-Ready pin of the external sensor (Figure 22. External sensor connections in Mode 2) and the latter is in Power-Down mode, then no data-ready signal can be generated by the external sensor. For this reason, the initial configuration of the external sensor's register has to be performed using the internal trigger signal

(START_CONFIG=0). After the external sensor is activated and the data-ready signal is available, the external trigger signal can be used by switching the START_CONFIG bit to 1.

- PULL_UP_EN bit enables/disables the internal pull-up on the auxiliary I^{2}C line. When this bit is set to 0, the internal pull-up is disabled and the external pull-up resistors on the SDx/SCx pins are required, as shown in Figure 22. External sensor connections in Mode 2. When this bit is set to 1, the internal pull-up is enabled and the external pull-up resistors on the SDx/SCx pins are not required.
- PASS_THROUGH_MODE bit is used to enable/disable the I^{2}C interface pass-through. When this bit is set to 1, the main I^{2}C line (e.g. connected to an external microcontroller) is short-circuited with the auxiliary one in order to implement a direct access to the external sensor registers. See Section 6.3 Sensor hub pass-through feature for details.
- MASTER_ON bit has to be set to 1 to enable the auxiliary I^{2}C master of the ISM330DLC device (sensor hub mode).

6.2.3 FUNC_SRC1 (53h)

Table 39. **FUNC_SRC1 register**

b7	b6	b5	b4	b3	b2	b1	b0
X	X	X	X	X	X	X	SENSOR HUB_ END_OP

- SENSORHUB_END_OP bit reports the status of the I²C master: during the idle state of the I²C master, this bit is equal to 1; it goes to 0 during I²C master read/write operations.

  When a sensor hub routine is completed, this bit automatically goes to 1 and the external sensor data are available to be read from the SENSORHUBx_REG registers (depending on the configuration of the SLVx_ADD, SLVx_SUBADD, SLAVEx_CONFIG registers).

  Note: The SENSORHUB_END_OP bit is cleared by reading the FUNC_SRC1 register if the LIR bit in TAP_CFG register is set to 1, otherwise it is cleared only during an I²C master read or write operation.

  Information about the status of the I²C master can be driven to the INT1 interrupt pin by setting the DRDY_ON_INT1 bit of the MASTER_CONFIG register to 1: if the LIR bit of the TAP_CFG register is set to 0, a pulsed interrupt signal (with typical pulse duration of about 150 µs) is generated at the rising edge of the SENSORHUB_END_OP signal. If latched mode is enabled (LIR bit is set to 1) and the interrupt signal is driven to the interrupt pin INT1, this interrupt signal is cleared by reading the FUNC_SRC1 register.

6.2.4 FUNC_SRC2 (54h)

Table 40. **FUNC_SRC2 register**

b7	b6	b5	b4	b3	b2	b1	b0
0	SLAVE3_ _NACK	SLAVE2 _NACK	SLAVE1 _NACK	SLAVE0 _NACK	X	0	X

- SLAVEx_NACK bits are set to 1 if a “not acknowledge” event happens during the communication with the corresponding slave x.

6.2.5 SLV0_ADD (02h), SLV0_SUBADD (03h), SLAVE0_CONFIG (04h)

The embedded function registers (accessible when the FUNC_CFG_EN bit is set to 1 in the FUNC_CFG_ACCESS register) used to configure the I^{2}C slave interface associated to the first external sensor are described hereafter.

Table 41. **SLV0_ADD register**

b7	b6	b5	b4	b3	b2	b1	b0
Slave0 _add6	Slave0 _add5	Slave0 _add4	Slave0 _add3	Slave0 _add2	Slave0 _add1	Slave0 _add0	rw_0

- Slave0_add[6:0] bits are used to indicate the I^{2}C slave address of the first external sensor.
- rw_0 bit configures the read/write operation to be performed on the first external sensor (0: write operation; 1: read operation). The read/write operation is executed when the next sensor hub trigger event occurs. When the rw_0 bit is set to 0 (write operation selected), the content of the SENSORHUBx_REG registers is not updated.

Table 42. **SLV0_SUBADD register**

b7	b6	b5	b4	b3	b2	b1	b0
Slave0 _reg7	Slave0 _reg6	Slave0 _reg5	Slave0 _reg4	Slave0 _reg3	Slave0 _reg2	Slave0 _reg1	Slave0 _reg0

- Slave0_reg[7:0] bits are used to indicate the address of the register of the first external sensor to be written (if the rw_0 bit of the SLV0_ADD register is set to 0) or the address of the first register to be read (if the rw_0 bit is set to 1).

Table 43. **SLAVE0_CONFIG register**

b7	b6	b5	b4	b3	b2	b1	b0
Slave0 _rate1	Slave0 _rate0	Aux_sens _on1	Aux_sens _on0	Src _mode	Slave0 _numop2	Slave0 _numop1	Slave0 _numop0

- Slave0_rate[1:0] bits are used to define the decimation factor applied to read operations on the first external sensor starting from the sensor hub trigger:
  - 00: no decimation
  - 01: update every 2 sensor hub trigger events
  - 10: update every 4 sensor hub trigger events
  - 11: update every 8 sensor hub trigger events
- Aux_sens_on[1:0] bits have to be used to indicate the number of external sensors to be managed by the sensor hub:
  - 00: one external sensor
  - 01: two external sensors
  - 10: three external sensors
  - 11: four external sensors
- Src_mode bit enables/disables source mode conditioned reading. When this bit is set to 1, source mode conditioned reading is enabled; before proceeding with the reading of the register address indicated in the SLV0_SUBADD register, the content of the register at the address specified in

DATAWRITE_SRC_MODE_SUB_SLV0 is checked: if the content is non-zero the operation continues, else the reading operation is interrupted. Source mode conditioned reading is available on slave 0 only.

- Slave0_numop[2:0] bits are dedicated to define the number of consecutive read operations to be performed on the first external sensor starting from the register address indicated in the SLV0_SUBADD register.

6.2.6 SLV1_ADD (05h), SLV1_SUBADD (06h), SLAVE1_CONFIG (07h)

The embedded function registers (accessible when the FUNC_CFG_EN bit is set to 1 in the FUNC_CFG_ACCESS register) used to configure the I²C slave interface associated to the second external sensor are described hereafter.

Table 44. **SLV1_ADD register**

b7	b6	b5	b4	b3	b2	b1	b0
Slave1 _add6	Slave1 _add5	Slave1 _add4	Slave1 _add3	Slave1 _add2	Slave1 _add1	Slave1 _add0	r_1

- Slave1_add[6:0] bits are used to indicate the I^{2}C slave address of the second external sensor.
- r_1 bit enables/disables the read operation to be performed on the second external sensor (0: read operation disabled; 1: read operation enabled). The read operation is executed when the next sensor hub trigger event occurs.

Table 45. **SLV1_SUBADD register**

b7	b6	b5	b4	b3	b2	b1	b0
Slave1 _reg7	Slave1 _reg6	Slave1 _reg5	Slave1 _reg4	Slave1 _reg3	Slave1 _reg2	Slave1 _reg1	Slave1 _reg0

- Slave1_reg[7:0] bits are used to indicate the address of the register of the second external sensor to be read when the r_1 bit of SLV1_ADD register is set to 1.

Table 46. **SLAVE1_CONFIG register**

b7	b6	b5	b4	b3	b2	b1	b0
Slave1 _rate1	Slave1 _rate0	write _once	0	0	Slave1 _numop2	Slave1 _numop1	Slave1 _numop0

- Slave1_rate[1:0] bits are used to define the decimation factor applied to read operations on the second external sensor starting from the sensor hub trigger:
  - 00: no decimation
  - 01: update every 2 sensor hub trigger events
  - 10: update every 4 sensor hub trigger events
  - 11: update every 8 sensor hub trigger events
- write_once bit is used to limit the write operations on slave 0 to only one occurrence (avoiding to repeat the same write operation multiple times). If this bit is not asserted, a write operation is triggered at each ODR.

Note: In order to enable the write_once feature, the field Aux_sens_on in the SLAVE0_CONFIG register must be different than 00b (even if only slave 0 is used).

- Slave1_numop[2:0] bits are dedicated to define the number of consecutive read operations to be performed on the second external sensor starting from the register address indicated in the SLV1_SUBADD register.

6.2.7 SLV2_ADD (08h), SLV2_SUBADD (09h), SLAVE2_CONFIG (0Ah)

The embedded function registers (accessible when the FUNC_CFG_EN bit is set to 1 in the FUNC_CFG_ACCESS register) used to configure the I^{2}C slave interface associated to the third external sensor are described hereafter.

Table 47. **SLV2_ADD register**

b7	b6	b5	b4	b3	b2	b1	b0
Slave2 _add6	Slave2 _add5	Slave2 _add4	Slave2 _add3	Slave2 _add2	Slave2 _add1	Slave2 _add0	r_2

- Slave2_add[6:0] bits are used to indicate the I^{2}C slave address of the third external sensor.
- r_2 bit enables/disables the read operation to be performed on the third external sensor (0: read operation disabled; 1: read operation enabled). The read operation is executed when the next sensor hub trigger event occurs.

Table 48. **SLV2_SUBADD register**

b7	b6	b5	b4	b3	b2	b1	b0
Slave2 _reg7	Slave2 _reg6	Slave2 _reg5	Slave2 _reg4	Slave2 _reg3	Slave2 _reg2	Slave2 _reg1	Slave2 _reg0

- Slave2_reg[7:0] bits are used to indicate the address of the register of the third external sensor to be read when the r_2 bit of the SLV2_ADD register is set to 1.

Table 49. **SLAVE2_CONFIG register**

b7	b6	b5	b4	b3	b2	b1	b0
Slave2 _rate1	Slave2 _rate0	0	0	0	Slave2 _numop2	Slave2 _numop1	Slave2 _numop0

- Slave2_rate[1:0] bits are used to define the decimation factor applied to read operations on the third external sensor starting from the sensor hub trigger:
  - 00: no decimation
  - 01: update every 2 sensor hub trigger events
  - 10: update every 4 sensor hub trigger events
  - 11: update every 8 sensor hub trigger events
- Slave2_numop[2:0] bits are dedicated to define the number of consecutive read operations to be performed on the third external sensor starting from the register address indicated in the SLV2_SUBADD register.

6.2.8 SLV3_ADD (0Bh), SLV3_SUBADD (0Ch), SLAVE3_CONFIG (0Dh)

The embedded function registers (accessible when the FUNC_CFG_EN bit is set to 1 in the FUNC_CFG_ACCESS register) used to configure the I^2C slave interface associated to the fourth external sensor are described hereafter.

Table 50. **SLV3_ADD register**

b7	b6	b5	b4	b3	b2	b1	b0
Slave3 _add6	Slave3 _add5	Slave3 _add4	Slave3 _add3	Slave3 _add2	Slave3 _add1	Slave3 _add0	r_3

- Slave3_add[6:0] bits are used to indicate the I^2C slave address of the fourth external sensor.
- r_3 bit enables/disables the read operation to be performed on the fourth external sensor (0: read operation disabled; 1: read operation enabled). The read operation is executed when the next sensor hub trigger event occurs.

Table 51. **SLV3_SUBADD register**

b7	b6	b5	b4	b3	b2	b1	b0
Slave3 _reg7	Slave3 _reg6	Slave3 _reg5	Slave3 _reg4	Slave3 _reg3	Slave3 _reg2	Slave3 _reg1	Slave3 _reg0

- Slave3_reg[7:0] bits are used to indicate the address of the register of the fourth external sensor to be read when the r_3 bit of the SLV3_ADD register is set to 1.

Table 52. **SLAVE3_CONFIG register**

b7	b6	b5	b4	b3	b2	b1	b0
Slave3 _rate1	Slave3 _rate0	0	0	0	Slave3 _numop2	Slave3 _numop1	Slave3 _numop0

- Slave3_rate[1:0] bits are used to define the decimation factor applied to the read operations on the fourth external sensor starting from the sensor hub trigger:
  - 00: no decimation
  - 01: update every 2 sensor hub trigger events
  - 10: update every 4 sensor hub trigger events
  - 11: update every 8 sensor hub trigger events
- Slave3_numop[2:0] bits are dedicated to define the number of consecutive read operations to be performed on the fourth external sensor starting from the register address indicated in the SLV3_SUBADD register.

6.2.9 DATAWRITE_SRC_MODE_SUB_SLV0 (0Eh)

Table 53. DATAWRITE_SRC_MODE_SUB_SLV0 register

b7	b6	b5	b4	b3	b2	b1	b0
Slave _dataw7	Slave _dataw6	Slave _dataw5	Slave _dataw4	Slave _dataw3	Slave _dataw2	Slave _dataw1	Slave _dataw0

- Slave_dataw[7:0] bits are dedicated, when the rw_0 bit of SLV0_ADD register is set to 0 (write operation), to indicate the data to be written to the first external sensor at the address specified in the SLV0_SUBADD register. During read operations (rw_0 = 1), this register is used if the source mode conditioned reading is enabled (Src_mode bit = 1 in the SLAVE0_CONFIG register) and it indicates the address of the external sensor register to be checked before proceeding with the read operation.

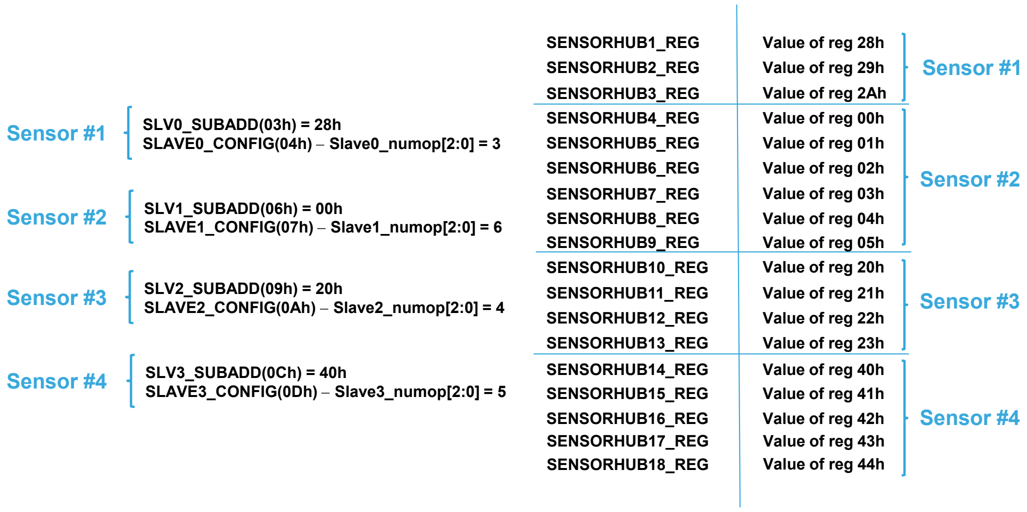
6.2.10 SENSORHUBx_REG registers

Once the auxiliary I^{2}C master is enabled, for each of the external sensors it reads a number of registers equal to the value of the Slavex_numop (x = 0, 1, 2, 3) field, starting from the register address specified in the SLVx_SUBADD (x = 0, 1, 2, 3) register. The number of external sensors to be managed is specified in the Aux_sens_on bits of the SLAVE0_CONFIG register.

Read data are consecutively stored (in the same order they are read) in the ISM330DLC registers starting from the SENSORHUB1_REG register, as in the example in Figure 23. SENSORHUBx_REG allocation example; 18 registers, from SENSORHUB1_REG to SENSORHUB18_REG, are available to store the data read from the external sensors.

The values of the registers from SENSORHUB1_REG to SENSORHUB6_REG can be saved in the FIFO buffer as a third data set; the values of the registers from SENSORHUB7_REG to SENSORHUB12_REG can be saved in the FIFO buffer as a fourth data set (see Section 8 First-in first-out (FIFO) buffer for details).

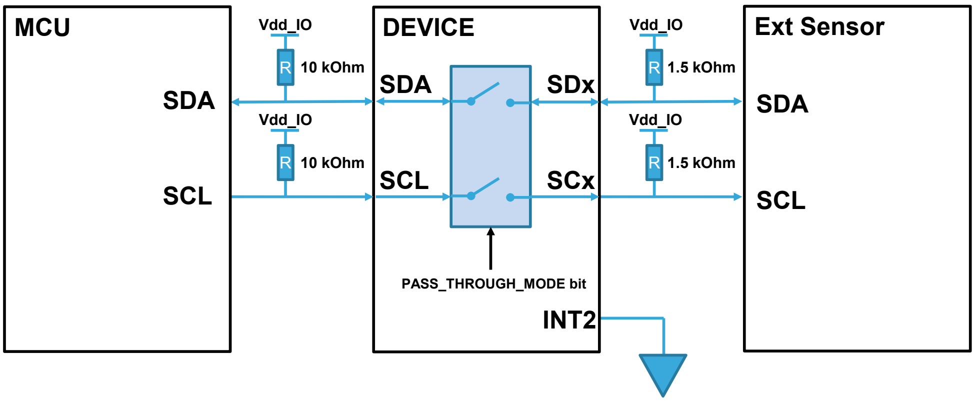
Figure 23. SENSORHUBx_REG allocation example


Sensor #1
SLV0_SUBADD(03h) = 28h
SLAVE0_CONFIG(04h) – Slave0_numop[2:0] = 3
Sensor #2
SLV1_SUBADD(06h) = 00h
SLAVE1_CONFIG(07h) – Slave1_numop[2:0] = 6
Sensor #3
SLV2_SUBADD(09h) = 20h
SLAVE2_CONFIG(0Ah) – Slave2_numop[2:0] = 4
Sensor #4
SLV3_SUBADD(0Ch) = 40h
SLAVE3_CONFIG(0Dh) – Slave3_numop[2:0] = 5
SENSORHUB1_REG Value of reg 28h
SENSORHUB2_REG Value of reg 29h
SENSORHUB3_REG Value of reg 2Ah
Sensor #1
SENSORHUB4_REG Value of reg 00h
SENSORHUB5_REG Value of reg 01h
SENSORHUB6_REG Value of reg 02h
SENSORHUB7_REG Value of reg 03h
SENSORHUB8_REG Value of reg 04h
SENSORHUB9_REG Value of reg 05h
Sensor #2
SENSORHUB10_REG Value of reg 20h
SENSORHUB11_REG Value of reg 21h
SENSORHUB12_REG Value of reg 22h
SENSORHUB13_REG Value of reg 23h
Sensor #3
SENSORHUB14_REG Value of reg 40h
SENSORHUB15_REG Value of reg 41h
SENSORHUB16_REG Value of reg 42h
SENSORHUB17_REG Value of reg 43h
SENSORHUB18_REG Value of reg 44h
Sensor #4

6.3 Sensor hub pass-through feature

The PASS_THROUGH_MODE bit of the MASTER_CONFIG register is used to enable/disable the I^{2}C interface pass-through: when it is set to 1, the main I^{2}C line (e.g. connected to an external microcontroller) is short-circuited with the auxiliary one in order to implement a direct access to the external sensor registers. It is recommended to use this feature when configuring the external sensors.

Figure 24. **Pass-through feature**



Some limitations must be considered when using the sensor hub and the pass-through feature. Three different scenarios are possible:

1. The sensor hub is used with the START_CONFIG bit of the MASTER_CONFIG register set to 0 (internal trigger) and the pass-through feature is not used: there is no limitation on INT2 pin usage.
2. The sensor hub is used with the START_CONFIG bit of the MASTER_CONFIG register set to 0 (internal trigger) and the pass-through feature is used: the INT2 pin must be connected to GND; it is not possible to switch to external trigger configuration (by setting the START_CONFIG bit to 1) and the INT2 pin cannot be used for the digital interrupts. Specific procedures have to be applied to enable/disable the pass-through feature: they are described in Section 6.3.1 Pass-through feature enable and in Section 6.3.2 Pass-through feature disable.
3. The sensor hub is used with the START_CONFIG bit of the MASTER_CONFIG register set to 1 (external trigger): the pass-through feature cannot be used; the INT2 pin has to be connected to the data-ready pin of the external sensor (trigger signal) and the procedure below has to be executed to avoid conflicts with the INT2 line:
   a. Set either the TRIG_EN or LVL1_EN or LVL2_EN bit of the CTRL6_C register to 1 (to configure the INT2 pin as input pin);
   b. Configure the external sensors (do not use the pass-through);
   c. Configure the sensor hub SLAVEx registers;
   d. Set the START_CONFIG bit of the MASTER_CONFIG register to 1;
   e. Set the MASTER_ON bit of the MASTER_CONFIG register to 1;
   f. Reset to 0 the bit in the CTRL6_C register asserted in step a.

Examples of external sensor configurations without using the pass-through are given in Section 6.4 Sensor hub mode example and Section 6.5.4 Ironing example.

6.3.1 Pass-through feature enable

When the embedded sensor hub functionality is disabled, the pass-through feature can be enabled at any time by setting the PASS_THROUGH_MODE bit of the MASTER_CONFIG register to 1.

When the embedded sensor hub functionality is enabled, a specific procedure has to be followed to enable the pass-through feature in order to prevent I^{2}C bus arbitration loss:

1. Set the START_CONFIG bit of the MASTER_CONFIG register to 1 in order to disable the sensor hub trigger (external trigger is enabled, but no trigger can be received on the INT2 pin since it's connected to GND);
2. Wait at least 5 ms (running I^{2}C operations will be completed);
3. Set the MASTER_ON bit of the MASTER_CONFIG register to 0 in order to disable the embedded sensor hub;
4. Set the START_CONFIG bit of the MASTER_CONFIG register to 0 in order to restore the sensor hub trigger;
5. Set the PULL_UP_EN bit of the MASTER_CONFIG register to 0 in order to disable the I^{2}C master pull-up;
6. Set the PASS_THROUGH_MODE bit of the MASTER_CONFIG register to 1 in order to enable the pass-through feature.

6.3.2 Pass-through feature disable

The procedure below has to be used in order to disable the pass-through:

1. Wait for the external microcontroller connected to the main I^{2}C line to complete all running I^{2}C operations. The pass-through must not be disabled in the middle of an I^{2}C transaction;
2. Set the PASS_THROUGH_MODE bit of the MASTER_CONFIG register to 0.

At this point, the internal I^{2}C master pull-up can be restored by setting the PULL_UP_EN bit of the MASTER_CONFIG register to 1, and the auxiliary I^{2}C master can be enabled by setting the MASTER_ON bit of the MASTER_CONFIG register to 1.

6.4 Sensor hub mode example

The configuration of the external sensors should be performed using the pass-through feature: this feature can be enabled by setting the PASS_THROUGH_MODE bit of the MASTER_CONFIG register to 1 and implements a direct access to the external sensor registers, allowing quick configuration.

The code provided below gives a basic routine to configure the ISM330DLC in sensor hub mode. Furthermore, this sequence configures the LIS2MDL external magnetometer sensor (refer to the datasheet for additional details) in continuous mode and reads the magnetometer output registers, saving their values in the SENSORHUB1_REG to SENSORHUB6_REG registers. The pass-through feature is not used in this example.

1.	Write 80h to FUNC_CFG_ACCESS	// Enable access to the embedded functions registers
2.	Write 3Ch to SLV0_ADD	// LIS2MDL slave address = 0011110b // Enable write operation (rw_0=0)
3.	Write 60h to SLV0_SUBADD	// 60h is the LIS2MDL register to be written

4.	Write 8Ch to DATAWRITE_SRC_MODE_SUB_SLV0	// 8Ch is the value to be written in register 60h of
		// LIS2MDL to configure it in continuous mode,
		// ODR = 100 Hz, temperature compensation enabled

5.	Write 10h to SLAVE0_CONFIG	// Set Aux_sens_on bits different from 00b
6.	Write 20h to SLAVE1_CONFIG	// Enable write_once bit
7.	Write 00h to FUNC_CFG_ACCESS	// Disable access to the embedded functions registers
8.	Write 04h to CTRL10_C	// Enable embedded functions

9.	Write 09h to MASTER_CONFIG	// Enable internal pull-up on SDx/SCx lines
		// Sensor hub trigger signal is XL Data Ready
		// Enable auxiliary I^{2}C master

10.	Write 80h to CTRL1_XL	// Turn on the accelerometer (for trigger signal)
11.	Read FUNC_SRC1	// Wait for the sensor hub communication to be concluded
12.	If SENSORHUB_END_OP = 0, go to 9	
13.	Write 00h to CTRL10_C	// Disable embedded functions
14.	Write 00h to MASTER_CONFIG	// Disable auxiliary I^{2}C master
15.	Write 00h to CTRL1_XL	// Turn off the accelerometer
16.	Write 80h into FUNC_CFG_ACCESS	// Enable access to the embedded functions registers
17.	Write 3Dh to SLV0_ADD	// LIS2MDL slave address = 0011110b // Enable read operation (rw_0=1)
18.	Write 68h to SLV0_SUBADD	// 68h is the first LIS2MDL output register to be read

19.	Write 06h to SLAVE0_CONFIG	// No decimation
		// 1 external sensor connected
		// Number of registers to read = 6

20.	Write 00h to FUNC_CFG_ACCESS	// Disable access to the embedded functions registers
21.	Write 04h to CTRL10_C	// Enable embedded functions

22.	Write 09h to MASTER_CONFIG	// Enable internal pull-up on SDx/SCx lines
		// Sensor hub trigger signal is XL Data-Ready
		// Enable auxiliary I^{2}C master

23.	Write 80h to CTRL1_XL	// Turn on the accelerometer (for trigger signal)

6.5 Magnetometer hard-iron / soft-iron correction

The ISM330DLC device supports the data acquisition of an external magnetometer with soft-iron and hard-iron correction features. For this purpose, it is required to set the MASTER_ON bit of the MASTER_CONFIG register to 1 to enable the sensor hub mode, to associate the external magnetometer to slave 0 registers (SLV0_ADD, SLV0_SUBADD and SLAVE0_CONFIG) and to set the Slave0_numop field of SLAVE0_CONFIG to 6.

The FUNC_EN bit of CTRL10_C register has to be set to 1 in order to enable the embedded ironing functionalities. Then, distortion correction algorithms can be enabled as described in Table 54. Ironing configuration: the IRON_EN bit of MASTER_CONFIG and the SOFT_EN bit of CTRL9_XL are used to enable hard-iron correction only or both hard-iron and soft-iron corrections. In the latter case, both calibrated (hard-iron & soft-iron) and uncalibrated (soft-iron only) magnetometer data are available.

Table 54. **Ironing configuration**

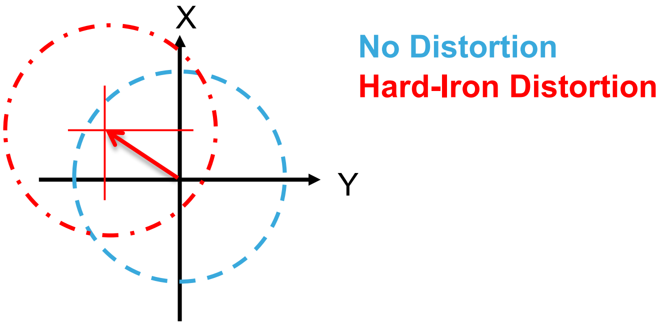
CTRL9_XL SOFT_EN bit	MASTER_CONFIG IRON_EN bit	Ironing configuration
0	0	No correction applied
0	1	Hard-iron only
1	1	Hard-iron + soft-iron corrections

6.5.1 Hard-iron correction

Hard-iron distortion is normally generated by ferromagnetic material with permanent magnetic fields that are part of the object (e.g. a tablet) in use; these materials could be permanent magnets or magnetized iron or steel. They are time invariant and deform the local geomagnetic field with different offset on different directions.

Generally, if the user performs many 3D rotations of the object in an ideal environment (no hard-iron/soft-iron distortion) and plots the collected magnetic sensor raw data, the result will be a perfect sphere with no offset. The hard-iron distortion effect is to offset the sphere along the X, Y and Z axes; in the X-Y plane, the hard-iron distortion is identified by an offset of the origin of the ideal circle from (0, 0), as shown in Figure 25. Hard-iron effect (X-Y 2D scatter plot).

Figure 25. **Hard-iron effect (X-Y 2D scatter plot)**



In the ISM330DLC device, the 3x1 hard-iron vector containing the X, Y, Z magnetic offset values calculated by the user have to be indicated in dedicated registers: the MAG_OFFX_L and MAG_OFFX_H registers are dedicated to the X-axis offset, the MAG_OFFY_L and MAG_OFFY_H registers are dedicated to the Y-axis offset, the MAG_OFFZ_L and MAG_OFFZ_H registers are dedicated to the Z-axis offset. These registers values are expressed as a 16-bit word in two's complement; the sensitivity [LSB/Gauss] to be applied to calculate the hard-iron register values corresponds to that of the external magnetometer.

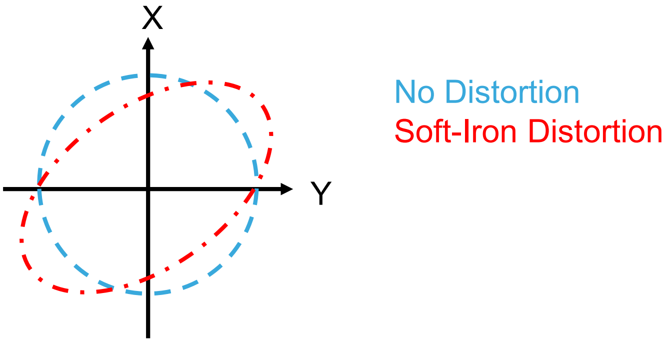
The hard-iron registers are accessible when the FUNC_CFG_EN bit of the FUNC_CFG_ACCESS register is set to 1. In order to enable the hard-iron correction algorithm, it is necessary to set to 1 both the FUNC_EN bit of the CTRL10_C register and the IRON_EN bit of the MASTER_CONFIG register (Table 54. Ironing configuration).

6.5.2 Soft-iron correction

Soft-iron distortion is generated by magnetically soft materials or current carrying PCB traces. While the hard-iron distortion is constant regardless of the orientation, the soft-iron distortion changes with the orientation of the object in the Earth’s field. Basically, the local geomagnetic field is deformed by different gain on different directions.

The effect of the soft-iron distortion is to make the ideal full round sphere become a tilted ellipsoid; in the X-Y plane, the soft-iron distortion is identified by a tilted ellipse with the origin in (0, 0), as shown in Figure 25. Hard-iron effect (X-Y 2D scatter plot).

Figure 26. **Soft-iron effect (X-Y 2D scatter plot)**



In the ISM330DLC device, the 3x3 soft-iron transformation matrix calculated by the user has to be indicated in 9 dedicated registers: MAG_SI_XX, MAG_SI_XY, MAG_SI_XZ, MAG_SI_YX, MAG_SI_YY, MAG_SI_YZ, MAG_SI_ZX, MAG_SI_ZY, MAG_SI_ZZ. These register values are expressed as an 8-bit word in sign-magnitude format; for these registers 1 LSB corresponds to 1/8, so the matrix parameters calculated by the user must be multiplied by 8 before writing them in the soft-iron registers.

The soft-iron registers are accessible when the FUNC_CFG_EN bit of the FUNC_CFG_ACCESS register is set to 1. In order to enable the soft-iron correction algorithm it is necessary to set to 1 the FUNC_EN bit of the CTRL10_C register, the IRON_EN bit of the MASTER_CONFIG register and the SOFT_EN bit of the CTRL9_XL register (Table 54. Ironing configuration).

6.5.3 Getting compensated magnetometer data

The status of magnetometer data acquisition and hard-iron/soft-iron correction can be checked using the FUNC_SRC1 register:

- SENSORHUB_END_OP bit is set high when the sensor hub routine has completed. The acquired magnetometer raw data are available in registers from address 66h (OUT_MAG_RAW_X_L) to 6Bh (OUT_MAG_RAW_Z_H).
- SI_END_OP bit is set high when the execution of the enabled hard-iron and soft-iron algorithms has completed. If the soft-iron correction is enabled, the magnetometer uncalibrated data (with soft-iron only applied) are available in registers from address 4Dh (SENSORHUB13_REG) to 52h (SENSORHUB18_REG). The magnetometer calibrated data, with both hard-iron (if enabled) and soft-iron (if enabled) correction, are available in registers from address 2Eh (SENSORHUB1_REG) to 33h (SENSORHUB6_REG).

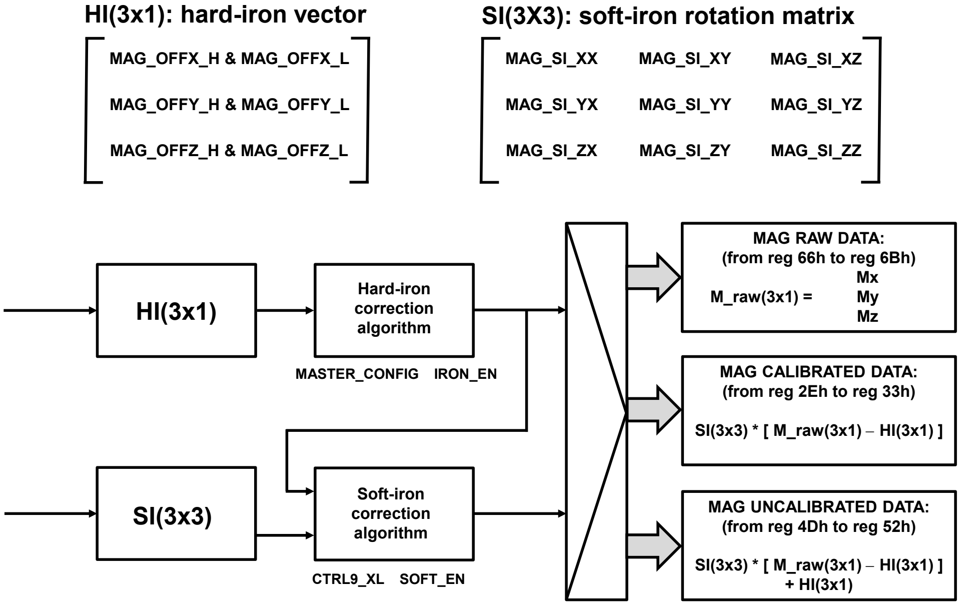
If latched mode is disabled (LIR bit of TAP_CFG is set to 0), the SENSORHUB_END_OP and SI_END_OP bits are active only for 1/100 Hz, then they are automatically deasserted. If latched mode is enabled, these two bits are cleared by reading the FUNC_SRC1 register.

The SENSORHUB_END_OP signal can be driven to the INT1 interrupt pin by setting the DRDY_ON_INT1 bit of the MASTER_CONFIG register to 1. The SI_END_OP signal can be driven to the INT2 interrupt pin by setting the INT2_IRON bit of the MD2_CFG register to 1.

A schematic representation of hard-iron and soft-iron correction feature is illustrated in Figure 27. Hard-iron / soft-iron correction block scheme below.

If the soft-iron correction is enabled and the soft-iron registers still have the default zero value, then the magnetometer calibrated data and the magnetometer uncalibrated data will also be equal to zero. As a consequence, when the soft-iron correction is enabled, the soft-iron transformation matrix must be at least initialized to the identity matrix multiplied by 8, setting the value of the MAG_SI_XX, MAG_SI_YY and MAG_SI_ZZ registers to 08h.

Figure 27. Hard-iron / soft-iron correction block scheme


HI(3x1): hard-iron vector
MAG_OFFX_H & MAG_OFFX_L
MAG_OFFY_H & MAG_OFFY_L
MAG_OFFZ_H & MAG_OFFZ_L
SI(3X3): soft-iron rotation matrix
MAG_SI_XX
MAG_SI_XY
MAG_SI_XZ
MAG_SI_YX
MAG_SI_YY
MAG_SI_YZ
MAG_SI_ZX
MAG_SI_ZY
MAG_SI_ZZ
HI(3x1)
Hard-iron correction algorithm
MASTER_CONFIG IRON_EN
SI(3x3)
Soft-iron correction algorithm
CTRL9_XL SOFT_EN
MAG RAW DATA: (from reg 66h to reg 6Bh)
M_raw(3x1) = Mx My Mz
MAG CALIBRATED DATA: (from reg 2Eh to reg 33h)
SI(3x3) * [M_raw(3x1) – HI(3x1)]
MAG UNCALIBRATED DATA: (from reg 4Dh to reg 52h)
SI(3x3) * [M_raw(3x1) – HI(3x1)] + HI(3x1)

6.5.4 Ironing example

The following example demonstrates how to define the values to be assigned to hard-iron and soft-iron correction registers starting from the calculated hard-iron vector and soft-iron rotation matrix. This example refers to the usage of the LIS2MDL magnetometer sensor.

- Hard-iron (X,Y,Z) offset values vector (gauss):

$$HI(3x1) = \begin{bmatrix} -0.335605 \\ 0.126487 \\ -0.114722 \end{bmatrix}$$

These three offset values must be divided by the LIS2MDL sensitivity value (0.0015 gauss / LSB) in order the get the LSB values to be written in the hard-iron correction registers (Table 55. Hard-iron register values).

Table 55. Hard-iron register values

	Offset values [LSB]	Register values

X	-224 (FF20h)	MAG_OFFX_H = FFh
		MAG_OFFX_L = 20h

Y	84 (0054h)	MAG_OFFY_H = 00h
		MAG_OFFY_L = 54h

Z	-76 (FFB4h)	MAG_OFFZ_H = FFh
		MAG_OFFZ_L = B4h

- Soft-iron rotation matrix:

$$SI(3x3) = \begin{bmatrix} 1.229006 & 0.173917 & 0.052327 \\ 0.173917 & 1.033307 & -0.130089 \\ 0.052327 & -0.130089 & 1.243645 \end{bmatrix}$$

These soft-iron matrix elements must be multiplied by 8 in order to get the LSB values to be written in the soft-iron correction registers (Table 56. Soft-iron register values). LSB values are expressed in sign-magnitude format.

Table 56. **Soft-iron register values**

	Soft-iron matrix elements	Register values
XX	+1.229006	MAG_SI_XX = 0Ah
XY	+0.173917	MAG_SI_XY = 01h
XZ	+0.052327	MAG_SI_XZ = 00h
YX	+0.173917	MAG_SI_YX = 01h
YY	+1.033307	MAG_SI_YY = 08h
YZ	-0.130089	MAG_SI_YZ = 81h
ZX	+0.052327	MAG_SI_ZX = 00h
ZY	-0.130089	MAG_SI_ZY = 81h
ZZ	+1.243645	MAG_SI_ZZ = 0Ah

The code provided below gives a basic routine to configure the LIS2MDL external magnetometer sensor (refer to the datasheet for additional details) in continuous mode, initialize the hard-iron and soft-iron correction registers and read the magnetometer output registers. In this case, the pass-through feature is not used for the magnetometer configuration.

```
1.  Write 80h to FUNC_CFG_ACCESS                 // Enable access to the embedded functions registers
                                                 // LIS2MDL slave address = 0011110b
2.  Write 3Ch to SLV0_ADD
                                                 // Enable write operation (rw_0=0)
3.  Write 60h to SLV0_SUBADD                     // 60h is the LIS2MDL register to be written
                                                 // 8Ch is the value to be written in register 60h of
4.  Write 8Ch to DATAWRITE_SRC_MODE_SUB_SLV0     // LIS2MDL to configure it in continuous mode,
                                                 // ODR = 100 Hz, temperature compensation enabled
5.  Write 10h to SLAVE0_CONFIG                   // Set Aux_sens_on bits different from 00b
6.  Write 20h to SLAVE1_CONFIG                   // Enable write_once bit
7.  Write 00h to FUNC_CFG_ACCESS                 // Disable access to the embedded functions registers
8.  Write 04h to CTRL10_C                        // Enable embedded functions
                                                 // Enable internal pull-up on SDx/SCx lines
9.  Write 09h to MASTER_CONFIG                   // Sensor hub trigger signal is XL Data-Ready
                                                 // Enable auxiliary I²C master
10. Write 80h to CTRL1_XL                        // Turn on the accelerometer (for trigger signal)
11. Read FUNC_SRC1                               // Wait for the sensor hub communication to be concluded
12. If SENSORHUB_END_OP = 0, go to 9
13. Write 00h to CTRL10_C                        // Disable embedded functions
14. Write 00h to MASTER_CONFIG                   // Disable auxiliary I²C master
```

15.	Write 00h to CTRL1_XL	// Turn off the accelerometer
16.	Write 80h to FUNC_CFG_ACCESS	// Enable access to the embedded functions registers
17.	Write 3Dh to SLV0_ADD	// LIS2MDL slave address = 0011110b // Enable read operation (rw_0=1)
18.	Write 68h to SLV0_SUBADD	// 68h is the first LIS2MDL output register to be read

19.	Write 06h to SLAVE0_CONFIG	// No decimation
		// 1 external sensor connected
		// Number of registers to read = 6

20.	Write FFh to MAG_OFFX_H	// X offset value initialization
21.	Write 20h to MAG_OFFX_L	// X offset value initialization
22.	Write 00h to MAG_OFFY_H	// Y offset value initialization
23.	Write 54h to MAG_OFFY_L	// Y offset value initialization
24.	Write FFh to MAG_OFFZ_H	// Z offset value initialization
25.	Write B4h to MAG_OFFZ_L	// Z offset value initialization
26.	Write 0Ah to MAG_SI_XX	// XX soft-iron element
27.	Write 01h to MAG_SI_XY	// XY soft-iron element
28.	Write 00h to MAG_SI_XZ	// XZ soft-iron element
29.	Write 01h to MAG_SI_YX	// YX soft-iron element
30.	Write 08h to MAG_SI_YY	// YY soft-iron element
31.	Write 81h to MAG_SI_YZ	// YZ soft-iron element
32.	Write 00h to MAG_SI_ZX	// ZX soft-iron element
33.	Write 81h to MAG_SI_ZY	// ZY soft-iron element
34.	Write 0Ah to MAG_SI_ZZ	// ZZ soft-iron element
35.	Write 00h to FUNC_CFG_ACCESS	// Disable access to the embedded functions registers
36.	Write 04h to CTRL10_C	// Enable embedded functions

37.	Write 0Bh to MASTER_CONFIG	// Enable internal pull-up on SDx/SCx lines
		// Sensor hub trigger signal is XL Data-Ready
		// Enable hard-iron correction
		// Enable auxiliary I^{2}C master

38.	Write 04h to CTRL9_XL	// Enable soft-iron correction
39.	Write 80h to CTRL1_XL	// Turn on the accelerometer (for trigger signal)

The acquired magnetometer raw data are available in registers from address 66h (OUT_MAG_RAW_X_L) to 6Bh (OUT_MAG_RAW_Z_L).

The magnetometer uncalibrated data (with soft-iron only applied) are available in registers from address 4Dh (SENSORHUB13_REG) to 52h (SENSORHUB18_REG).

The magnetometer calibrated data, with both hard-iron and soft-iron correction, are available in registers from address 2Eh (SENSORHUB1_REG) to 33h (SENSORHUB6_REG).

7 Mode 3 and Mode 4 - Auxiliary SPI modes

The Auxiliary SPI modes (Mode 3 and Mode 4) allow accessing the ISM330DLC from multiple external devices: when one of these modes is enabled, both an I^{2}C/SPI (3/4-wire) slave interface and an Auxiliary SPI (3/4-wire) slave interface are available for connecting external devices.

When Mode 3 is enabled, the gyroscope OIS chain is activated; when Mode 4 is enabled, both the accelerometer OIS chain and the gyroscope OIS chain are activated.

They can be used, for example, in applications requiring closed control loop. The device, through the dedicated auxiliary SPI interface and a configurable signal processing path (with low latency and low noise) can provide data for the control loop, while, at the same time, a second fully independent path can output data for other application purposes.

7.1 Auxiliary SPI mode description

In order to enable Mode 3, the OIS_EN_SPI2 bit in CTRL1_OIS register must be set to 1. When Mode 3 is enabled, the gyroscope output values are available through the Auxiliary SPI interface selected (3/4-wire) with full scale selected through the FS[1:0]_G_OIS and FS_125_OIS bits of CTRL1_OIS register and ODR at 6.66 kHz.

If both the OIS_EN_SPI2 bit and the MODE4_EN bit in CTRL1_OIS register are set to 1, Mode 4 is enabled and also the accelerometer output values are available at 6.66 kHz ODR through the Auxiliary SPI interface in addition to the gyroscope values. If the accelerometer GP chain is in power-down, the accelerometer full scale on the OIS chain can be selected using the FS[1:0]_XL_OIS bits of the CTRL3_OIS register. If the accelerometer GP chain is not in power-down, the accelerometer full scale on the OIS chain corresponds to the one applied on the GP side using the FS_XL[1:0] bits of CTRL1_XL, regardless of the value of the FS[1:0]_XL_OIS bits.

When Mode 3 / Mode 4 is enabled, both the GP filtering chain and the OIS filtering chain are modified, as described in Section 3.7 Accelerometer bandwidth for the accelerometer sensor and in Section 3.7.1 Accelerometer slope filter for the gyroscope. All details about turn-on/off time are provided in Section 3.9 Accelerometer and gyroscope turn-on/off time.

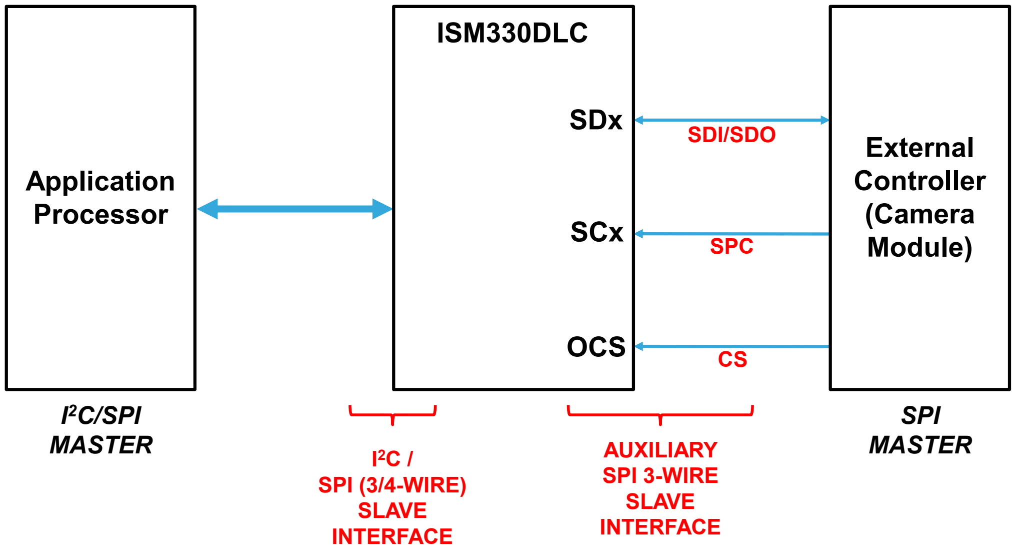
The function of the ISM330DLC pins after Mode 3 / Mode 4 is enabled is indicated in Table 57. Mode 3/4 pin description.

Table 57. **Mode 3/4 pin description**

Interface	Pin	Mode 3 function
I^{2}C slave interface	SDO/SA0	I^{2}C least significant bit of the device address (SA0) / SPI 4-wire interface serial data output (SDO)
	SCL	I^{2}C serial clock (SCL) / SPI serial port clock (SPC)
	SDA	I^{2}C serial data(SDA) / SPI serial data input (SDI), 3-wire interface serial data output (SDO)
	CS	I^{2}C / SPI mode selection (1: I^{2}C; 0: SPI)
Auxiliary SPI 3/4-wire slave interface	SDx	Auxiliary SPI 3/4-wire interface serial data input (SDI) and SPI 3-wire serial data output (SDO)
	SCx	Auxiliary SPI 3/4-wire interface serial port clock (SPC)
	OCS_Aux	Auxiliary SPI 3/4-wire enable
	SDO_Aux	Auxiliary SPI 4-wire serial data output (SDO_Aux)

The external devices have to be connected to the ISM330DLC as illustrated in Figure 28. External controller connection in Mode 3/4 (SPI 3-wire), if using the SPI 3-wire interface (SIM_OIS bit in CTRL1_OIS = 1). The setup has to be changed accordingly when using the SPI 4-wire interface (connect SDO_Aux pin too).

Figure 28. External controller connection in Mode 3/4 (SPI 3-wire)


Application Processor
ISM330DLC
SDx
SCx
OCS
SDI/SDO
SPC
CS
External Controller (Camera Module)
I²C/SPI MASTER
I²C / SPI (3/4-WIRE) SLAVE INTERFACE
AUXILIARY SPI 3-WIRE SLAVE INTERFACE
SPI MASTER

Note: When the Auxiliary SPI interface is enabled, the following rules must be observed:

- *The SLEEP bit of the CTRL4_C register has to be set to 0 (default value);*
- *The ST_XL[1:0] bits and the ST_G[1:0] bits of the CTRL5_C register have to be set to 0 (default value).*

The accelerometer/gyroscope data stored in FIFO can be accessed through the primary I^{2}C/SPI interface only.

7.2 Auxiliary SPI mode registers

The primary I^{2}C/SPI (3/4-wire) interface is always available and the gyroscope output values can be read in registers 22h to 27h with full scale and ODR selectable through the CTRL2_G register. Similarly, the accelerometer output values can be read through the primary interface in registers 28h to 2Dh with full scale and ODR selectable through the CTRL1_XL register.

The value of the bits of the INT_OIS, CTRL1_OIS, CTRL2_OIS, CTRL3_OIS registers can be modified through the Auxiliary SPI interface only (these registers are 'read-only' when accessed through the primary interface): these are the only registers that can be written through the Auxiliary SPI interface; all the other read/write registers can be written through the primary interface only.

When the OIS_EN_SPI2 bit of the CTRL1_OIS register is set to 1, Mode 3 is enabled and the gyroscope output values can be read in registers 22h to 27h through the Auxiliary SPI interface. When a new gyroscope data is available on the OIS chain, the GDA bit of STATUS_SPIAux register is set to 1; it is reset when one of the high parts of the output data registers is read. The GYRO_SETTLING bit in the STATUS_SPIAux register is equal to 1 when the gyro OIS chain is in settling phase (max 80 ms). The data read during this settling phase are not valid: the recommendation is to check the status of this bit to understand when valid data are available.

When both the OIS_EN_SPI2 bit and the MODE4_EN bit of the CTRL1_OIS register are set to 1, Mode 4 is enabled. In addition to the gyroscope output values (in registers 22h to 27h), the accelerometer output values can also be read in registers 28h to 2Dh through the Auxiliary SPI interface. When new accelerometer data is available on the OIS chain, the XLDA bit of STATUS_SPIAux register is set to 1; it is reset when one of the high parts of the output data registers is read.

Basically, the accelerometer/gyroscope output data registers (22h to 2Dh) and the STATUS_REG register (1Eh) contain different data when they are read from the primary I^{2}C/SPI interface and from the Auxiliary SPI interface. All the other registers contain the same value.

All the registers of the ISM330DLC can be read at the same time from both the external master devices.

7.2.1 INT_OIS (6Fh)

Table 58. **INT_OIS register**

b7	b6	b5	b4	b3	b2	b1	b0
INT2 _DRDY _OIS	LVL2 _OIS	-	-	-	-	-	-

- INT2_DRDY_OIS bit can be used to drive the DRDY signal of the OIS chain to the INT2 pin. The DRDY signal of the OIS chain is pulsed; latched mode is not available.
- LVL2_OIS enables, in combination with the LVL1_OIS bit of the CTRL1_OIS register, level-sensitive trigger/ latched mode on the OIS chain; refer to Section 7.2.2 CTRL1_OIS (70h) for details.

7.2.2 CTRL1_OIS (70h)

Table 59. **CTRL1_OIS register**

b7	b6	b5	b4	b3	b2	b1	b0
BLE _OIS	LVL1 _OIS	SIM _OIS	MODE4 _EN	FS1_G _OIS	FS0_G _OIS	FS_125 _OIS	OIS_EN _SPI2

- BLE_OIS bit can be used to define big/little endian selection: it allows swapping the content of the lower and the upper part of the accelerometer/gyroscope output data registers on the OIS chain, similarly to the BLE bit of the CTRL3_C register from the primary interface (refer to Section 4.5.1 Big-little endian selection for details).
- LVL1_OIS can be used, in combination with the LVL2_OIS bit of the INT_OIS register, to enable level-sensitive trigger mode on OIS (Table 59. CTRL1_OIS register).
- SIM_OIS bit has to be set to 1 in order to enable the 3-wire Auxiliary SPI interface, otherwise 4-wire Auxiliary SPI interface is used.
- MODE4_EN bit enables the accelerometer OIS chain (Mode 4); OIS_EN_SPI2 bit must also to be set to 1 to properly enable Mode 4.
- FS[1:0]_G_OIS bits can be used to select the gyroscope OIS full-scale (when FS_125_OIS bit is set to 0), similarly to the FS_G[1:0] bits of the CTRL2_G register.
- FS_125_OIS bit enables ±125 dps full-scale on the gyroscope OIS chain. If it is equal to 0, full-scale is selected through the FS[1:0]_G_OIS bits.
- OIS_EN_SPI2 bit must to be set to 1 in order to enable OIS chain data processing for gyroscope and accelerometer data in Mode 3 and Mode 4.

DEN mode on OIS side can be enabled using the LVL1_OIS bit of the CTRL1_OIS register and the LVL2_OIS bit of register INT_OIS.

DEN mode on the OIS path is active on the gyroscope sensor only. Further details about level-sensitive trigger/ latched mode are provided in Section 4.8 Edge-sensitive and level-sensitive data enable (DEN).

Table 60. **DEN mode selection**

LVL1_OIS, LVL2_OIS	DEN mode
00b	DEN mode on OIS path disabled
10b	Level-sensitive trigger mode is selected
11b	Level-sensitive latched mode is selected

7.2.3 CTRL2_OIS (71h)

Table 61. **CTRL2_OIS register**

b7	b6	b5	b4	b3	b2	b1	b0
0	0	HPM1 _OIS	HPM0 _OIS	0	FTYPE _1_OIS	FTYPE _0_OIS	HP_EN _OIS

- HPM[1:0]_OIS bits can be used to select the digital HP filter cutoff on the OIS side. The table below shows the available configurations.

Table 62. **Gyroscope OIS chain HPF cutoff selection**

HPM[1:0]_OIS	Cutoff [Hz]
00	0.016
01	0.065
10	0.260
11	1.04

- FTYPE_[1:0]_OIS bits can be used to select the digital LPF1 bandwidth. The table below shows the cutoff and phase delay values obtained with all configurations.

Table 63. **Gyroscope OIS chain LPF1 bandwidth selection**

FTYPE_[1:0]_OIS	ODR = 6.66 kHz	
	BW	Phase delay @ 20 Hz
00	351 Hz	7°
01	237 Hz	9°
10	173 Hz	11°
11	937 Hz	5°

Note: When Mode 3/4 is enabled, the LPF1 filter is not available on the gyroscope GP chain. It is recommended to avoid using the LPF1 filter in Mode 1 and Mode 2 when Mode 3 or Mode 4 is intended to be used.

- HP_EN_OIS bit can be used to enable the HP filter on the OIS chain.

Note: The HP filter is available on the OIS side only if the HP_EN_OIS bit is set to 1 and the HP_EN_G bit in CTRL7_G is set to 0.

7.2.4 CTRL3_OIS (72h)

Table 64. **CTRL3_OIS register**

b7	b6	b5	b4	b3	b2	b1	b0
DEN_LH _OIS	FS1 _XL_OIS	FS0 _XL_OIS	FILTER _XL _CONF _OIS_1	FILTER _XL _CONF _OIS_0	ST1_OIS	ST0_OIS	ST_OIS _CLAMPDIS

- DEN_LH_OIS bit can be used to select the polarity of the DEN signal on the gyroscope OIS chain. If the DEN_LH_OIS bit is set to 0, the DEN pin is active low, otherwise it is active high.
- FS[1:0]_XL_OIS bits can be used in order to select the accelerometer full-scale on the OIS chain. These two bits act only when the accelerometer GP chain is in Power-Down mode, otherwise the accelerometer full-scale corresponds to that set on the GP side through the CTRL1_XL register.
- FILTER_XL_CONF_OIS_[1:0] bits set the accelerometer OIS chain bandwidth.

Table 65. **Accelerometer OIS bandwidth selection**

FILTER_XL_CONF_OIS[1:0]	ODR_XL = 0 (Power-Down) ODR_XL ≥ 1600 Hz		ODR_XL ≤ 800 Hz	
	BW	Phase delay @ 20 Hz	BW	Phase delay @ 20 Hz
00	140 Hz	9.39°	128 Hz	11.5°
01	68.2 Hz	17.6°	66.5 Hz	19.7°
10	636 Hz	2.96°	329 Hz	5.08°
11	295 Hz	5.12°	222 Hz	7.23°

Note: If using the ISM330DLC device in Mode 4, the LPF2 and HP filters are not available on the accelerometer GP chain. It is recommended to avoid using the LPF2 and HP filters in Mode 1/2/3 when Mode 4 is intended to be used.

- ST[1:0]_OIS bits can be set in order to select the self-test on the gyroscope OIS chain (see Section 10 Self-test for further details).
- ST_OIS_CLAMPDIS bit can be used to enable/disable the OIS chain clamp in the gyroscope self-test. If the ST_OIS_CLAMPDIS bit is set to 1, once the gyroscope self-test functionality is enabled, the gyroscope output values read from the Auxiliary SPI interface show the same variation observed while reading the data from the primary interface. If the ST_OIS_CLAMPDIS bit is set to 0, when the gyroscope self-test functionality is enabled, the gyroscope output values read from the Auxiliary SPI interface are always clamped to 8000h value: for example, this feature allows the host device connected to the Auxiliary interface to detect when the self-test functionality has been enabled from the GP side. By design, the maximum gyroscope output value is one LSB lower than 8000h, so if the 8000h is read from the Auxiliary SPI it means that the self-test feature was enabled from the GP side.

7.2.5 STATUS_SPIAux (1Eh)

Table 66. **STATUS_SPIAux register**

b7	b6	b5	b4	b3	b2	b1	b0
0	0	0	0	0	GYRO_ SETTLING	GDA	XLDA

- GYRO_SETTLING bit is set to 1 during the initial settling phase of the gyroscope output. The gyroscope output data generated when this bit is equal to 1 have to be discarded.
- GDA bit is set to 1 when new gyroscope data is available in register 22h to 27h on the OIS chain. It is reset when one of the high parts of the output data registers is read.
- XLDA bit is set to 1 when new accelerometer data is available in register 28h to 2Dh on the OIS chain (Mode 3/4). It is reset when one of the high parts of the output data register is read.

7.3 Reading gyroscope data through the Auxiliary SPI

The procedure to be applied after device power-up to read the gyroscope output data through the Auxiliary SPI 3-wire interface is as follows:

1.	Wait 15 ms	// Boot time // Device in Power-Down mode after this time period
2.	Write 21h to CTRL1_OIS	// Turn on gyro through Auxiliary SPI 3-wire interface // (OIS Gyro: FS = ±250 dps / ODR = 6.6 kHz)
3.	Wait 80 ms	// Gyroscope max turn-on time
4.	Read output registers 22h to 27h	// Read gyroscope output data through Auxiliary SPI

7.4 Mode 4 - Reading gyroscope and accelerometer output data through the Auxiliary SPI

The procedure to be applied after device power-up to read the gyroscope and accelerometer output data through the Auxiliary SPI 3-wire interface is as follows:

1.	Wait 15 ms	// Boot time // Device in Power-Down mode after this time period

2.	Write 31h to CTRL1_OIS	// Turn on gyro through Auxiliary SPI 3-wire interface
		// (OIS Gyro: FS = ±250 dps / ODR = 6.66 kHz)
		// Enable Mode 4 (setting MODE4_EN)

3.	Write 00h to CTRL3_OIS	// Set XL through Auxiliary SPI 3-wire interface // (OIS XL: FS = ±2 *g* / ODR = 6.66 kHz)
4.	Wait 80 ms	// Gyroscope max turn-on time
5.	Read output registers 22h to 27h	// Read gyroscope output data through Auxiliary SPI
6.	Read output registers 28h to 2Dh	// Read accelerometer output data through Auxiliary SPI

8 First-in first-out (FIFO) buffer

In order to limit intervention by the host processor and facilitate post-processing data for event recognition, the ISM330DLC embeds a 4 kbyte first-in first-out buffer (FIFO).

The FIFO can be configured to store the following data:

- gyroscope sensor data;
- accelerometer sensor data;
- external sensor (connected to sensor hub interface) data;
- timestamp data;
- temperature sensor data.

Saving data in the FIFO buffer is based on four ‘FIFO data sets’ consisting of 6 bytes each:

- The 1st FIFO data set is reserved for gyroscope data;
- The 2nd FIFO data set is reserved for accelerometer data;
- The 3rd FIFO data set is reserved for the external sensor data stored in the registers from SENSORHUB1_REG to SENSORHUB6_REG (see Section 6.2.10 SENSORHUBx_REG registersfor details on the SENSORHUBx_REG);
- The 4th FIFO data set can be alternately associated to the external sensor data stored in the registers from SENSORHUB7_REG to SENSORHUB12_REG, to the timestamp info, or to the temperature sensor data.

All these data sets can be stored in FIFO at different ODRs, by setting the decimation factors in the FIFO_CTRL3 and FIFO_CTRL4 registers. Decimation factors are also used to select which FIFO data sets have to be stored in FIFO.

Five different FIFO operating modes can be chosen through the FIFO_MODE_[2:0] bits of the FIFO_CTRL5 register:

- Bypass mode;
- FIFO mode;
- Continuous mode;
- Continuous-to-FIFO mode;
- Bypass-to-Continuous mode.

Note: When the FIFO is used, the IF_INC and BDU bits of the CTRL3_C register must be equal to 1.

Data are retrieved from the FIFO through two dedicated registers: FIFO_DATA_OUT_L and FIFO_DATA_OUT_H. In this way, data can be read either from the FIFO (at a slower ODR) or from the device output registers (at the normal ODR).

To monitor the FIFO status (full, empty, number of samples stored, etc.), four dedicated registers are available: FIFO_STATUS1, FIFO_STATUS2, FIFO_STATUS3, FIFO_STATUS4.

Programmable FIFO thresholds can be set in FIFO_CTRL1 and FIFO_CTRL2 using the FTH_[10:0] bits.

FIFO full, FIFO threshold and FIFO overrun events can be enabled to generate dedicated interrupts on the two interrupt pins (INT1 and INT2) through the INT1_FULL_FLAG, INT1_FTH and INT1_FIFO_OVR bits of the INT1_CTRL register, and through the INT2_FULL_FLAG, INT2_FTH and INT2_FIFO_OVR bits of the INT2_CTRL register.

In order to increase the number of samples which can be stored in the FIFO, it is also possible to store (as 1st FIFO data set) only the 8 most significant bits of the accelerometer and gyroscope data by setting the bit ONLY_HIGH_DATA in the FIFO_CTRL4 register.

Writing data in the FIFO can be triggered by the accelerometer/gyroscope data-ready; it can also be triggered by the sensor hub data-ready (corresponding to the behavior of the SENSORHUB_END_OP bit of FUNC_SRC1 register): in this case the DATA_VALID_SEL_FIFO bit of the MASTER_CONFIG register must be set to 1.

8.1 FIFO registers

The FIFO buffer is managed by:

- five control registers (from FIFO_CTRL1 to FIFO_CTRL5);
- four status registers (from FIFO_STATUS1 to FIFO_STATUS4);
- two data output registers (FIFO_DATA_OUT_L and FIFO_DATA_OUT_H);
- some additional bits to enable threshold usage (STOP_ON_FTH) and route FIFO full, threshold or overrun events to the two interrupt lines (bits: INT1_FULL_FLAG, INT2_FULL_FLAG, INT1_FTH, INT2_FTH, INT1_FIFO_OVR, INT2_FIFO_OVR).

8.1.1 FIFO_CTRL1 (06h)

The FIFO_CTRL1 register contains the lower part of the 11-bit FIFO threshold level. For the complete threshold level configuration, consider also the FTH_[10:8] bits of the FIFO_CTRL2 register. The value of the FIFO threshold level is referred to data having 16-bit format.

The FIFO watermark flag (WaterM bit in FIFO_STATUS2 register) rises when the number of bytes stored in the FIFO is equal to or higher than the threshold level.

In order to limit the FIFO depth to the watermark level, the STOP_ON_FTH bit must be set to 1 in the FIFO_CTRL4 register.

Table 67. **FIFO_CTRL1 register**

b7	b6	b5	b4	b3	b2	b1	b0
FTH_7	FTH_6	FTH_5	FTH_4	FTH_3	FTH_2	FTH_1	FTH_0

8.1.2 FIFO_CTRL2 (07h)

Table 68. **FIFO_CTRL2 register**

b7	b6	b5	b4	b3	b2	b1	b0
FIFO_ TIMER _EN	0	0	0	FIFO_ TEMP_ EN	FTH_10	FTH_9	FTH_8

- FIFO_TIMER_EN enables timestamp data to be stored as the 4th FIFO data set. The content of the 6 bytes stored in the FIFO when this bit is set to 1 is described in Section 8.8 Timestamp data in FIFO.
- FIFO_TEMP_EN bit enables temperature data to be stored as the 4th FIFO data set. The content of the 6 bytes stored in the FIFO when this bit is set to 1 is described in Section 8.9 Temperature data in FIFO.
- FTH_[10:8] contains the upper part of the FIFO threshold level. For the complete threshold level configuration, consider also the FTH_[7:0] bits in the FIFO_CTRL1 register.

8.1.3 FIFO_CTRL3 (08h)

The FIFO_CTRL3 register contains the accelerometer and gyroscope FIFO decimation factors, used to choose if the data of these sensors have to be stored in the FIFO and at which rate they are stored.

When the DEC_FIFO_GYRO[2:0] bits are set to 000b, the 1st FIFO data set (reserved for gyroscope data) is not stored in the FIFO. When the DEC_FIFO_XL[2:0] bits are set to 000b, the 2nd FIFO data set (reserved for accelerometer data) is not stored in the FIFO.

Table 69. **FIFO_CTRL3 register**

b7	b6	b5	b4	b3	b2	b1	b0
0	0	DEC_ FIFO_ GYRO2	DEC_ FIFO_ GYRO1	DEC_ FIFO_ GYRO0	DEC_ FIFO_ XL2	DEC_ FIFO_ XL1	DEC_ FIFO_ XL0

Table 70. **Gyroscope FIFO decimation setting**

DEC_FIFO_GYRO [2:0]	Configuration
000	Gyroscope sensor not in FIFO
001	No decimation
010	Decimation with factor 2
011	Decimation with factor 3
100	Decimation with factor 4
101	Decimation with factor 8
110	Decimation with factor 16
111	Decimation with factor 32

Table 71. **Accelerometer FIFO decimation setting**

DEC_FIFO_XL [2:0]	Configuration
000	Accelerometer sensor not in FIFO
001	No decimation
010	Decimation with factor 2
011	Decimation with factor 3
100	Decimation with factor 4
101	Decimation with factor 8
110	Decimation with factor 16
111	Decimation with factor 32

8.1.4 FIFO_CTRL4 (09h)

The FIFO_CTRL4 register contains the decimation factors used to define at which data rate the data associated to the 3rd FIFO and the 4th FIFO data sets are stored in the FIFO.

When the DEC_DS3_FIFO[2:0] bits are set to 000b, the 3rd FIFO data set is not stored in the FIFO. When the DEC_DS4_FIFO[2:0] bits are set to 000b, the 4th FIFO data set is not stored in the FIFO.

The FIFO_CTRL4 register also contains the bit ONLY_HIGH_DATA, which allows storing in the FIFO only the upper part (Most Significant Byte) of the accelerometer and gyroscope data in order to increase the maximum number of accelerometer and gyroscope samples in the FIFO. See Section 8.7 High part of gyroscope and accelerometer data for more details about this functionality.

The FIFO_CTRL4 register contains the bit STOP_ON_FTH, which allows limiting the FIFO depth to the watermark level.

Table 72. **FIFO_CTRL4 register**

b7	b6	b5	b4	b3	b2	b1	b0
STOP_ ON_FTH	ONLY_ HIGH _DATA	DEC_ DS4 _FIFO2	DEC_ DS4 _FIFO1	DEC_ DS4 _FIFO0	DEC_ DS3 _FIFO2	DEC_ DS3 _FIFO1	DEC_ DS3 _FIFO0

Table 73. **3rd FIFO data set decimation setting**

DEC_DS3_FIFO [2:0]	Configuration
000	3rd FIFO data set not in FIFO
001	No decimation
010	Decimation with factor 2
011	Decimation with factor 3
100	Decimation with factor 4
101	Decimation with factor 8
110	Decimation with factor 16
111	Decimation with factor 32

Table 74. **4th FIFO data set decimation setting**

DEC_DS4_FIFO [2:0]	Configuration
000	4th FIFO data set not in FIFO
001	No decimation
010	Decimation with factor 2
011	Decimation with factor 3
100	Decimation with factor 4
101	Decimation with factor 8
110	Decimation with factor 16
111	Decimation with factor 32

8.1.5 FIFO_CTRL5 (0Ah)

The FIFO_CTRL5 register contains the FIFO operating mode bits (FIFO_MODE_[2:0]) and the FIFO output data rate bits (ODR_FIFO_[3:0]).

FIFO operating modes are described in Section 8.2 FIFO modes.

When the internal trigger (accelerometer/gyroscope data-ready) is used, the ODR_FIFO_[3:0] bits define the maximum data rate at which data are stored in FIFO. Data can be stored in FIFO at a lower data rate using the FIFO decimation factors. For more information about FIFO trigger and FIFO ODR configuration see Section 8.3 Setting the FIFO trigger, FIFO ODR and decimation factors.

Note: When the FIFO is used, the IF_INC bit of the CTRL3_C register must be equal to 1.

Table 75. **FIFO_CTRL5 register**

b7	b6	b5	b4	b3	b2	b1	b0
0	ODR_ FIFO_3	ODR_ FIFO_2	ODR_ FIFO_1	ODR_ FIFO_0	FIFO_ MODE_2	FIFO_ MODE_1	FIFO_ MODE_0

Table 76. **FIFO ODR selection setting**

ODR_FIFO [3:0]	Configuration
0000	FIFO disabled
0001	FIFO ODR is set to 12.5 Hz
0010	FIFO ODR is set to 26 Hz
0011	FIFO ODR is set to 52 Hz
0100	FIFO ODR is set to 104 Hz
0101	FIFO ODR is set to 208 Hz
0110	FIFO ODR is set to 416 Hz
0111	FIFO ODR is set to 833 Hz
1000	FIFO ODR is set to 1.66 kHz
1001	FIFO ODR is set to 3.33 kHz
1010	FIFO ODR is set to 6.66 kHz

Table 77. **FIFO mode selection**

FIFO_MODE [2:0]	Configuration
000	Bypass mode. FIFO disabled.
001	FIFO mode. Stops collecting data when FIFO is full.
010	Reserved
011	Continuous mode until trigger is deasserted, then FIFO mode.
100	Bypass mode until trigger is deasserted, then Continuous mode.
101	Reserved
110	Continuous mode. If the FIFO is full, the new sample overwrites the older one.
111	Reserved

8.1.6 FIFO_STATUS1 (3Ah)

The FIFO_STATUS1 register, together with the FIFO_STATUS2 register, provides information about the number of samples stored in the FIFO. Each sample is represented as 16-bit data.

Table 78. **FIFO_STATUS1 register**

b7	b6	b5	b4	b3	b2	b1	b0
DIFF_ FIFO_7	DIFF_ FIFO_6	DIFF_ FIFO_5	DIFF_ FIFO_4	DIFF_ FIFO_3	DIFF_ FIFO_2	DIFF_ FIFO_1	DIFF_ FIFO_0

8.1.7 FIFO_STATUS2 (3Bh)

The FIFO_STATUS2 register, together with the FIFO_STATUS1 register, provides information about the number of samples stored in the FIFO and about the current status (watermark, overrun, full, empty) of the FIFO buffer.

Table 79. **FIFO_STATUS2 register**

b7	b6	b5	b4	b3	b2	b1	b0
WaterM	OVER _RUN	FIFO_FULL _SMART	FIFO_ EMPTY	0	DIFF_ FIFO_10	DIFF_ FIFO_9	DIFF_ FIFO_8

- WaterM represents the watermark status. This bit is set high when the number of bytes already stored in the FIFO is equal to or higher than the watermark level (each sample is represented as 16-bit data). The watermark status can be driven to the two interrupt pins by setting to 1 the INT1_FTH bit of the INT1_CTRL register or the INT2_FTH bit of the INT2_CTRL register.
- OVER_RUN is set high when the FIFO is completely filled and at least one sample has already been overwritten to store the new data. This signal can be driven to the two interrupt pins by setting to 1 the INT1_FIFO_OVR bit of the INT1_CTRL register or the INT2_FIFO_OVR bit of the INT2_CTRL register.
- FIFO_FULL_SMART is set high when the next set of data that will be stored in FIFO will make the FIFO full. This signal can be driven to the two interrupt pins by setting to 1 the INT1_FULL_FLAG bit of the INT1_CTRL register or the INT2_FULL_FLAG bit of the INT2_CTRL register.
- FIFO_EMPTY is set high when the FIFO is empty.
- DIFF_FIFO_[10:8] contains the upper part of the number of unread words (16-bit data) stored in the FIFO. The lower part is represented by the DIFF_FIFO_[7:0] bits in FIFO_STATUS1. The value of DIFF_FIFO_[10:0] field corresponds to the number of samples in the FIFO (each sample is represented as 16-bit data). When a FIFO overrun event occurs (OVER_RUN bit is set high), the value of the DIFF_FIFO_[10:0] field is set to 0.

Register content is updated synchronously to the FIFO write and read operations, as illustrated in Table 80. FIFO_STATUS2 behavior (case with one sensor in FIFO, STOP_ON_FTH = 0).

Table 80. **FIFO_STATUS2 behavior (case with one sensor in FIFO, STOP_ON_FTH = 0)**

FIFO_OVER_RUN	FIFO_FULL	FIFO_EMPTY	DIFF_FIFO_ [10:0]	Number of FIFO samples	FIFO trigger timing
0	0	1	0	0	t0
0	0	0	3	3	t1
0	0	0	6	6	t2
...	...	...	...	...	...
0	0	0	2044	2044	t_full - 2
0	1	0	2047	2047	t_full - 1
1	1	0	0	2048 (old sample overwritten)	t_full

8.1.8 FIFO_STATUS3 (3Ch)

The FIFO_STATUS3 register, together with FIFO_STATUS4 register, specifies which axis of which sensor data will be read at the next reading. For more information on how to retrieve data from the FIFO see Section 8.5 FIFO pattern.

Table 81. **FIFO_STATUS3 register**

b7	b6	b5	b4	b3	b2	b1	b0
FIFO_ PATTERN _7	FIFO_ PATTERN _6	FIFO_ PATTERN _5	FIFO_ PATTERN _4	FIFO_ PATTERN _3	FIFO_ PATTERN _2	FIFO_ PATTERN _1	FIFO_ PATTERN _0

8.1.9 FIFO_STATUS4 (3Dh)

The FIFO_STATUS4 register, together with the FIFO_STATUS3 register, specifies which axis of which sensor data will be read at the next reading. For more information on how to retrieve data from the FIFO see Section 8.5 FIFO pattern.

Table 82. **FIFO_STATUS4 register**

b7	b6	b5	b4	b3	b2	b1	b0
0	0	0	0	0	0	FIFO_ PATTERN _9	FIFO_ PATTERN _8

8.1.10 FIFO_DATA_OUT_L (3Eh)

The FIFO_DATA_OUT_L register is the least significant byte of the FIFO output data. The most significant byte is stored in the FIFO_DATA_OUT_H register. For more information on how to retrieve data from the FIFO, see Section 8.4 Retrieving data from the FIFO.

Table 83. **FIFO_DATA_OUT_L register**

b7	b6	b5	b4	b3	b2	b1	b0
DATA_ OUT_FIFO _L_7	DATA_ OUT_FIFO _L_6	DATA_ OUT_FIFO _L_5	DATA_ OUT_FIFO _L_4	DATA_ OUT_FIFO _L_3	DATA_ OUT_FIFO _L_2	DATA_ OUT_FIFO _L_1	DATA_ OUT_FIFO _L_0

8.1.11 FIFO_DATA_OUT_H (3Fh)

The FIFO_DATA_OUT_H register is the most significant byte of the FIFO output data. The least significant byte is stored in the FIFO_DATA_OUT_L register. For more information on how to retrieve data from the FIFO, see Section 8.4 Retrieving data from the FIFO.

Table 84. **FIFO_DATA_OUT_H register**

b7	b6	b5	b4	b3	b2	b1	b0
DATA_ OUT_FIFO _H_7	DATA_ OUT_FIFO _H_6	DATA_ OUT_FIFO _H_5	DATA_ OUT_FIFO _H_4	DATA_ OUT_FIFO _H_3	DATA_ OUT_FIFO _H_2	DATA_ OUT_FIFO _H_1	DATA_ OUT_FIFO _H_0

8.2 FIFO modes

The ISM330DLC FIFO buffer can be configured to operate in five different modes selectable through the FIFO_MODE_[2:0] field of the FIFO_CTRL5 register. The available configurations ensure a high level of flexibility and extend the number of functions usable in application development.

Bypass, FIFO, Continuous, Continuous-to-FIFO and Bypass-to-Continuous modes are described in the following paragraphs.

Note: When the FIFO is used, the IF_INC bit of the CTRL3_C register must be equal to 1.

8.2.1 Bypass mode

When Bypass mode is enabled, the FIFO is not used, the buffer content is cleared, and it remains empty until another mode is selected.

Bypass mode is selected when the FIFO_MODE_[2:0] bits are set to 000b. When this mode is enabled, the FIFO_STATUS2 register contains the value 10h (FIFO empty).

Bypass mode must be used in order to stop and reset the FIFO buffer when a different mode is operating. Note that by placing the FIFO buffer into Bypass mode, the whole buffer content is cleared.

8.2.2 FIFO mode

In FIFO mode, the buffer continues filling until it becomes full. Then it stops collecting data and the FIFO content remains unchanged until a different mode is selected.

Follow these steps for FIFO mode configuration (if the accelerometer/gyroscope data-ready is used as the FIFO trigger):

1. Choose the decimation factor for each sensor through the decimation bits in the FIFO_CTRL3 and FIFO_CTRL4 registers (see Section 8.3 Setting the FIFO trigger, FIFO ODR and decimation factors for details);
2. Choose the FIFO ODR through the ODR_FIFO_[3:0] bits in the FIFO_CTRL5 register;
3. Set the FIFO_MODE_[2:0] bits in the FIFO_CTRL5 register to 001b to enable FIFO mode.

When this mode is selected, the FIFO starts collecting data. The FIFO_STATUS1 and FIFO_STATUS2 registers are updated according to the number of samples stored.

When the next stored set of data will make the FIFO full, the FIFO_FULL_SMART bit of the FIFO_STATUS2 register is set to 1 and no more data are stored in the FIFO buffer. Data can be retrieved after the FIFO_FULL_SMART event by reading the FIFO_DATA_OUT_L and FIFO_DATA_OUT_H registers for the number of times specified by the DIFF_FIFO_[10:0] bits of the FIFO_STATUS1 and FIFO_STATUS2 registers.

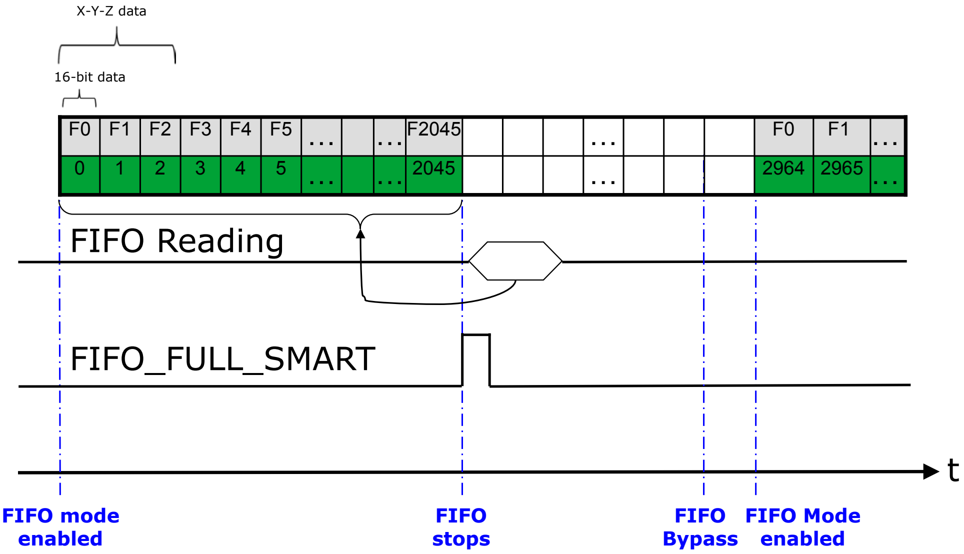
Using the WaterM bit of the FIFO_STATUS2 register, data can also be retrieved when a threshold level (FTH_[10:0] in FIFO_CTRL1 and FIFO_CTRL2 registers) is reached if the application requires a lower number of samples in the FIFO.

If the STOP_ON_FTH bit of the FIFO_CTRL4 register is set to 1, the FIFO size is limited to the value of the FTH_[10:0] bits in the FIFO_CTRL1 and FIFO_CTRL2 registers: in this case, the FIFO_FULL_SMART bit of the FIFO_STATUS2 register is set high when the number of samples in FIFO will reach or exceed the FTH_[10:0] value on the next FIFO write operation.

Communication speed is not very important in FIFO mode because the data collection is stopped and there is no risk of overwriting data already acquired. Before restarting the FIFO mode, it is necessary to set to Bypass mode first in order to completely clear the FIFO content.

Figure 29. FIFO mode (STOP_ON_FTH = 0) shows an example of FIFO mode usage. In the example X-Y-Z data (green cells indicate the sample number) from just one sensor are stored in the FIFO. In these conditions, the number of 16-bit samples that can be stored in the FIFO buffer is 2046.

Figure 29. FIFO mode (STOP_ON_FTH = 0)



8.2.3 Continuous mode

In Continuous mode, the FIFO continues filling. When the buffer is full, the FIFO index restarts from the beginning, and older data are replaced by the new data. The oldest values continue to be overwritten until a read operation frees FIFO slots. The host processor's reading speed is important in order to free slots faster than new data is made available. To stop this configuration, Bypass mode must be selected.

Follow these steps for Continuous mode configuration (if the accelerometer/gyroscope data-ready is used as the FIFO trigger):

1. Choose the decimation factor for each sensor through the decimation bits in the FIFO_CTRL3 and FIFO_CTRL4 registers (see Section 8.3 Setting the FIFO trigger, FIFO ODR and decimation factors for details);
2. Choose the FIFO ODR through the ODR_FIFO_[3:0] bits in the FIFO_CTRL5 register;
3. Set the FIFO_MODE_[2:0] bits in the FIFO_CTRL5 register to 110b to enable FIFO Continuous mode.

When this mode is selected, the FIFO collects data continuously. The FIFO_STATUS1 and FIFO_STATUS2 registers are updated according to the number of samples stored.

When the next stored set of data will make the FIFO full, the FIFO_FULL_SMART bit of the FIFO_STATUS2 register is set to 1. The OVER_RUN bit in the FIFO_STATUS2 register indicates when at least one sample has been overwritten to store the new data.

Data can be retrieved after the FIFO_FULL_SMART event by reading the FIFO_DATA_OUT_L and FIFO_DATA_OUT_H registers for the number of times specified by the DIFF_FIFO_[10:0] bits in the FIFO_STATUS1 and FIFO_STATUS2 registers.

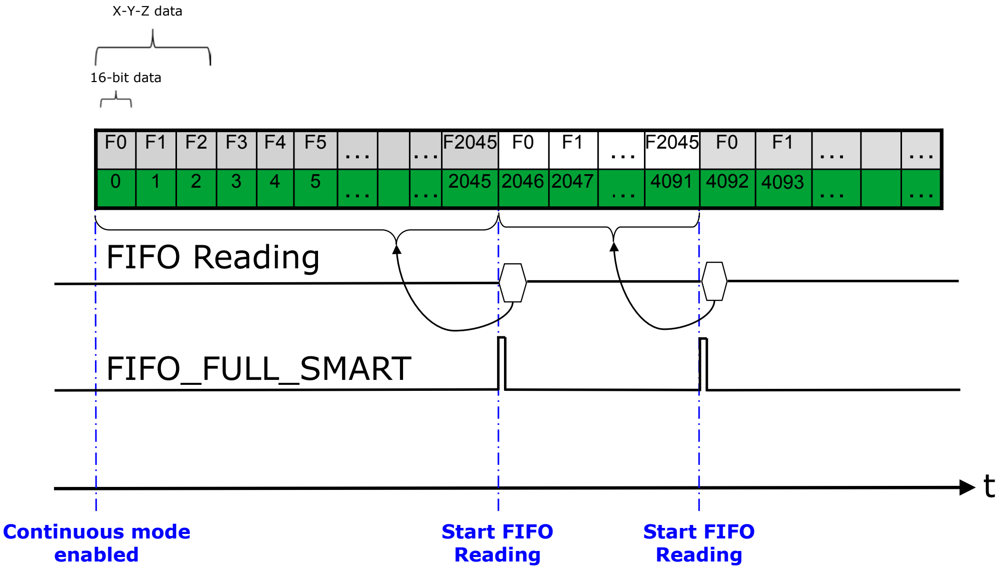
Using the WaterM bit of the FIFO_STATUS2 register, data can also be retrieved when a threshold level (FTH_[10:0] in FIFO_CTRL1 and FIFO_CTRL2 registers) is reached.

If the STOP_ON_FTH bit of FIFO_CTRL4 register is set to 1, the FIFO size is limited to the value of the FTH_[10:0] bits in the FIFO_CTRL1 and FIFO_CTRL2 registers: in this case, the FIFO_FULL_SMART bit of the FIFO_STATUS2 register is set high when the number of samples in FIFO will reach the FTH_[10:0] value on the next FIFO write operation.

It is recommended to read faster than 1*ODR at least three times the number of the enabled FIFO data set in order to free FIFO slots for the new data: this allows avoiding loss of data.

Figure 30. Continuous mode shows an example of the Continuous mode usage. In the example, X-Y-Z data (green cells indicate the sample number) from just one sensor are stored in the FIFO and the FIFO samples are read faster than 1 * ODR so that no data is lost. In these conditions, the number of 16-bit samples stored is 2046.

Figure 30. Continuous mode



8.2.4 Continuous-to-FIFO mode

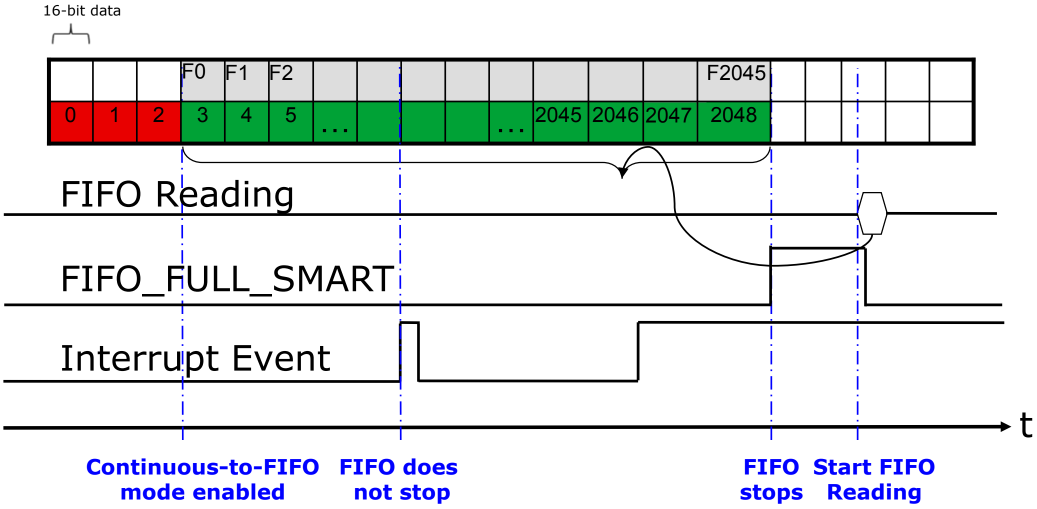
This mode is a combination of the Continuous and FIFO modes previously described. In Continuous-to-FIFO mode, the FIFO buffer starts operating in Continuous mode and switches to FIFO mode when an event condition occurs.

The event condition can be one of the following:

- Tilt: event detection has to be configured and the INT2_TILT bit of the MD2_CFG register has to be set to 1;
- Single tap: event detection has to be configured and the INT2_SINGLE_TAP bit of the MD2_CFG register has to be set to 1;
- Double tap: event detection has to be configured and the INT2_DOUBLE_TAP bit of the MD2_CFG register has to be set to 1;
- Free-fall: event detection has to be configured and the INT2_FF bit of the MD2_CFG register has to be set to 1;
- Wake-up: event detection has to be configured and the INT2_WU bit of the MD2_CFG register has to be set to 1;
- 6D: event detection has to be configured and the INT2_6D bit of the MD2_CFG register has to be set to 1.

Continuous-to-FIFO mode is sensitive to the edge of the interrupt signal: at the first interrupt event, FIFO changes from Continuous mode to FIFO mode and maintains it until Bypass mode is set.

Figure 31. **Continuous-to-FIFO mode**



Follow these steps for Continuous-to-FIFO mode configuration (if the accelerometer/gyroscope data-ready is used as the FIFO trigger):

1. Configure one of the events as previously described;
2. Choose the decimation factor for each sensor through the decimation bits in the FIFO_CTRL3 and FIFO_CTRL4 registers (see Section 8.3 Setting the FIFO trigger, FIFO ODR and decimation factors for details);
3. Choose the FIFO ODR through the ODR_FIFO_[3:0] bits in the FIFO_CTRL5 register;
4. Set the FIFO_MODE_[2:0] bits in the FIFO_CTRL5 register to 011b to enable FIFO Continuous-to-FIFO mode.

In Continuous-to-FIFO mode the FIFO buffer continues filling; when the next stored set of data will make the FIFO full, the FIFO_FULL_SMART bit is set high.

If the STOP_ON_FTH bit of the FIFO_CTRL4 register is set to 1, the FIFO size is limited to the value of the FTH_[10:0] bits in the FIFO_CTRL1 and FIFO_CTRL2 registers: in this case, the FIFO_FULL_SMART bit of the

FIFO_STATUS2 register is set high when the number of samples in FIFO will reach or exceed the FTH_[10:0] value on the next FIFO write operation.

When the trigger event occurs, two different cases can be observed:

1. If the FIFO buffer is already full (FIFO_FULL_SMART = 1), it stops collecting data at the first sample after the event trigger. The FIFO content is composed of the samples collected before the event.
2. If FIFO buffer is not full yet (initial transient), it continues filling until it becomes full (FIFO_FULL_SMART = 1) and then, if the trigger is still present, it stops collecting data.

Continuous-to-FIFO can be used in order to analyze the history of the samples which have generated an interrupt; the standard operation is to read the FIFO content when the FIFO mode is triggered and the FIFO buffer is full and stopped.

8.2.5 Bypass-to-Continuous mode

This mode is a combination of the Bypass and Continuous modes previously described. In Bypass-to-Continuous mode, the FIFO buffer starts operating in Bypass mode and switches to Continuous mode when a trigger condition occurs.

The event condition can be one of the following:

- Tilt: event detection has to be configured and the INT2_TILT bit of the MD2_CFG register has to be set to 1;
- Single tap: event detection has to be configured and the INT2_SINGLE_TAP bit of MD2_CFG register has to be set to 1;
- Double tap: event detection has to be configured and the INT2_DOUBLE_TAP bit of the MD2_CFG register has to be set to 1;
- Free-fall: event detection has to be configured and the INT2_FF bit of the MD2_CFG register has to be set to 1;
- Wake-up: event detection has to be configured and the INT2_WU bit of the MD2_CFG register has to be set to 1;
- 6D: event detection has to be configured and the INT2_6D bit of the MD2_CFG register has to be set to 1.

Bypass-to-Continuous mode is sensitive to the edge of the interrupt signal: at the first interrupt event, FIFO changes from Bypass mode to Continuous mode and maintains it until Bypass mode is set.

Follow these steps for Bypass-to-Continuous mode configuration (if the accelerometer / gyroscope data-ready is used as the FIFO trigger):

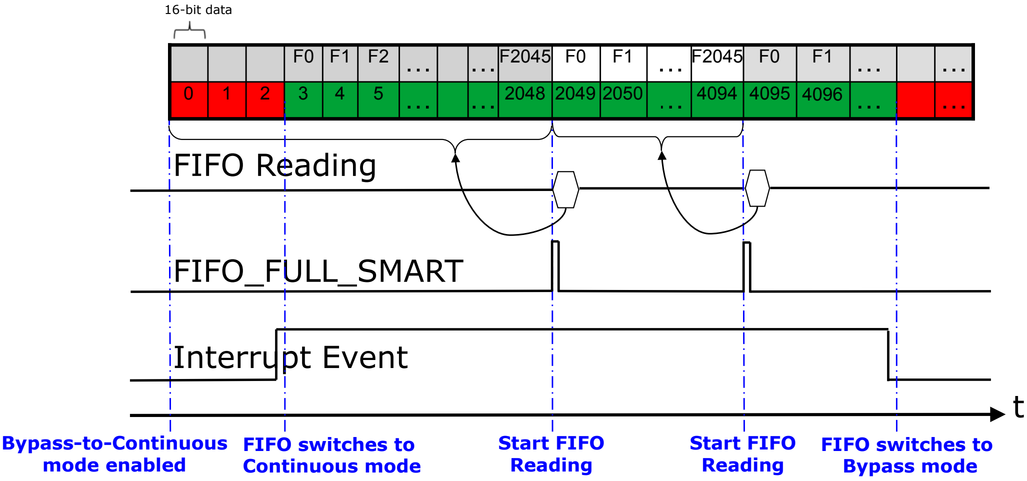
1. Configure one of the events as previously described;

2. Choose the decimation factor for each sensor through the decimation bits in the FIFO_CTRL3 and FIFO_CTRL4 registers (see Section 8.3 Setting the FIFO trigger, FIFO ODR and decimation factors for details);

3. Choose the FIFO ODR through the ODR_FIFO_[3:0] bits in the FIFO_CTRL5 register.

4. Set the FIFO_MODE_[2:0] bits in the FIFO_CTRL5 register to 100b to enable FIFO Bypass-to-Continuous mode.

Figure 32. Bypass-to-Continuous mode



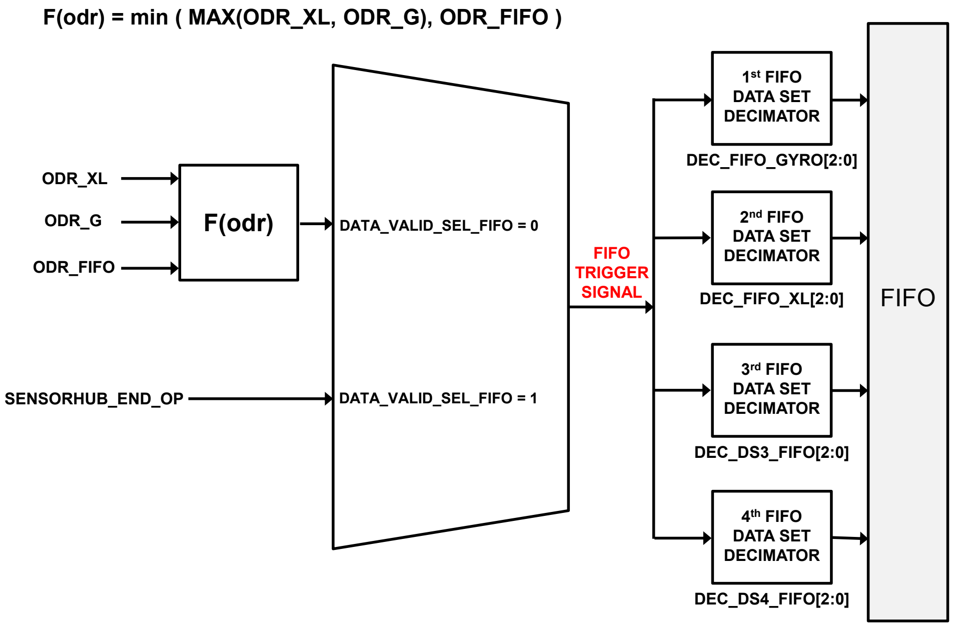
Once the trigger condition appears and the buffer switches to Continuous mode, the FIFO buffer continues filling. When the next stored set of data will make the FIFO full, the FIFO_FULL_SMART bit is set high.

Bypass-to-Continuous can be used in order to start the acquisition when the configured interrupt is generated.

8.3 Setting the FIFO trigger, FIFO ODR and decimation factors

Writing data in the FIFO can be configured to be triggered by three different sources.

Figure 33. **FIFO trigger signal selection**



As described in Figure 33. FIFO trigger signal selection, the DATA_VALID_SEL_FIFO bit of the MASTER_CONFIG register is used for this purpose:

- If the DATA_VALID_SEL_FIFO bit is set to 0, writing data in the FIFO is triggered by the accelerometer/ gyroscope data-ready. The ODR_FIFO_[3:0] bits of FIFO_CTRL5 define the maximum data rate at which data are stored in FIFO; the latter is limited to the maximum value between the accelerometer ODR (defined by the ODR_XL[3:0] bits of the CTRL1_XL register) and the gyroscope ODR (defined by the ODR_G[3:0] bits of the CTRL2_G register);
- If the DATA_VALID_SEL_FIFO bit is set to 1, writing data in the FIFO is triggered by the sensor hub (corresponding to the behavior of the SENSORHUB_END_OP bit of the FUNC_SRC1 register). The data are stored in FIFO when the sensor hub routine is complete.

Using the FIFO decimation factors, data can be stored in FIFO at a rate lower than the rate of the FIFO trigger signal. Four decimation factors can be configured, one for each FIFO data set:

- The DEC_FIFO_G[2:0] bits of the FIFO_CTRL3 register define if the gyroscope data (associated to the 1st FIFO data set) are stored in FIFO and the relative rate;
- The DEC_FIFO_XL[2:0] bits of the FIFO_CTRL3 register define if the accelerometer data (associated to the 2nd FIFO data set) are stored in FIFO and the relative rate;
- The DEC_DS3_FIFO[2:0] bits of the FIFO_CTRL4 register define if the data associated to the 3rd FIFO data set are stored in FIFO and the relative rate;
- The DEC_DS4_FIFO[2:0] bits of the FIFO_CTRL4 register define if the data associated to the 4th FIFO data set are stored in FIFO and the relative rate.

8.3.1 Procedure for ODR or FIFO configuration changes when using FIFO

Apply the following procedure when an accelerometer/gyroscope ODR or FIFO configuration change has to be performed:

1. Read all the data stored in the FIFO to empty it (see Section 8.4 Retrieving data from the FIFOfor details);
2. Set the FIFO in Bypass mode (set the FIFO_MODE bits of the FIFO_CTRL5 register to 000b);
3. Set the target ODR for the accelerometer and gyroscope through the ODR_XL bits of the CTRL1_XL register and the ODR_G bits of the CTRL2_G register respectively;
4. Set the target ODR for the FIFO through the ODR_FIFO bits of the FIFO_CTRL5 register;
5. Set the gyroscope decimation factor in the DEC_FIFO_G[2:0] bits of the FIFO_CTRL3 register and the accelerometer decimation factor in the DEC_FIFO_XL[2:0] bits of the FIFO_CTRL3 register (see Table 70. Gyroscope FIFO decimation settingand Table 71. Accelerometer FIFO decimation setting for the values to be set in the DEC_FIFO_G[2:0] bits and the DEC_FIFO_XL[2:0] bits of FIFO_CTRL3).
6. Set the desired FIFO operating mode (see Section 8.3 Setting the FIFO trigger, FIFO ODR and decimation factors for details).

8.4 Retrieving data from the FIFO

Note: When data are stored in the FIFO, the configuration must not be changed in order to be able to retrieve data correctly.

When FIFO is enabled and the mode is different from Bypass, reading the FIFO output registers (FIFO_DATA_OUT_L and FIFO_DATA_OUT_H) returns the oldest FIFO sample set. Whenever these registers are read, their content is moved to the SPI/I^{2}C output buffer. FIFO slots are ideally shifted up one level in order to release room for a new sample, and the FIFO output registers load the current oldest value stored in the FIFO buffer.

The recommended way to retrieve data from the FIFO is the following:

1. Read the FIFO_STATUS1 and FIFO_STATUS2 registers to check how many words (16-bit data) are stored in the FIFO. This information is contained in the DIFF_FIFO_[10:0] bits.
2. Read the FIFO_STATUS3 and FIFO_STATUS4 registers. The FIFO_PATTERN_[9:0] bits allows understanding which sensor and which couple of bytes are being read (see Section 8.5 FIFO pattern for more details).
3. Read the FIFO_DATA_OUT_L and FIFO_DATA_OUT_H registers to retrieve the oldest sample (16-bits format) in the FIFO. They are respectively the lower and the upper part of the oldest sample.

The entire FIFO content is retrieved by performing a certain number of read operations from the FIFO output registers until the buffer becomes empty (FIFO_EMPTY bit of FIFO_STATUS2 register is set high).

Note: Once the FIFO is empty, data must not be retrieved from the FIFO_DATA_OUT_L and FIFO_DATA_OUT_H registers.

It is recommended to read faster than 1*ODR at least three times the number of the enabled FIFO data set in order to free FIFO slots for the new data: this allows avoiding loss of data.

The rounding function (see Section 4.7 Rounding functions for details) is automatically enabled when applying a multiple read operation to the FIFO output registers FIFO_DATA_OUT_L and FIFO_DATA_OUT_H.

8.5 FIFO pattern

Data are stored in the FIFO without any tag in order to maximize the number of samples stored. To understand which couple of data and which FIFO data set is going to be read, it is necessary to check the content of the FIFO_PATTERN_[9:0] bits in the FIFO_STATUS3 and FIFO_STATUS4 registers.

Data are written to the FIFO with a specific pattern (for example GyroX, GyroY, GyroZ, AccX, AccY, AccZ). This pattern changes depending on the ODRs and decimation factors assigned to the four FIFO data sets. The FIFO_PATTERN_[9:0] bits contain a number from 0 to the index of the last sample of the pattern, then the pattern is repeated in all FIFO content.

The first sequence of data stored in the FIFO buffer contains the data of all the enabled FIFO data sets, from the first one to the fourth one. Then, data are repeated depending on the value of the decimation factor set for each FIFO data set.

The examples in the next sections explain how to use the information contained in the FIFO_PATTERN_[9:0] bits.

8.5.1 Example 1

Supposing the FIFO is storing data from the gyroscope and accelerometer at the same ODR:

- Gyroscope ODR = 104 Hz, Accelerometer ODR = 104 Hz.

If the internal trigger (accelerometer/gyroscope data-ready) is used, it’s recommended to set the ODR_FIFO_[3:0] bits of the FIFO_CTRL5 register to 0100b in order to set the FIFO trigger ODR to 104 Hz.

Both the DEC_FIFO_GYRO[2:0] and the DEC_FIFO_XL[2:0] fields of the FIFO_CTRL3 register have to be set to 001b (no decimation).

The following data pattern is repeated every 6 samples (each sample is represented as 16-bit data):

- Gx Gy Gz XLx XLy XLz (gyroscope and accelerometer data)

The FIFO_PATTERN_[9:0] bits will contain a number from 0 to 5, as shown in Table 85. Example 1: FIFO_PATTERN_[9:0] bits and next reading.

Table 85. **Example 1: FIFO_PATTERN_[9:0] bits and next reading**

Time	FIFO_PATTERN_[9:0]	Next reading from FIFO output registers
t0	0	Gx
t0	1	Gy
t0	2	Gz
t0	3	XLx
t0	4	XLy
t0	5	XLz

8.5.2 Example 2

Supposing the FIFO is storing data from the gyroscope and accelerometer at different ODRs:

- Gyroscope ODR = 208 Hz, Accelerometer ODR = 104 Hz.

If the internal trigger (accelerometer/gyroscope data-ready) is used, it’s recommended to set the ODR_FIFO_[3:0] bits of the FIFO_CTRL5 register to 0101b in order to set the FIFO trigger ODR to 208 Hz.

The DEC_FIFO_GYRO[2:0] field of the FIFO_CTRL3 register has to be set to 001b (no decimation applied to gyroscope data) and the DEC_FIFO_XL[2:0] field has to be set to 010b (decimation with factor 2 applied to accelerometer data).

Since the gyroscope ODR is twice the accelerometer ODR, the following data pattern is repeated every 9 samples (each sample is represented as 16-bit data):

- Gx Gy Gz XLx XLy XLz Gx Gy Gz

The FIFO_PATTERN_[9:0] bits will contain a number from 0 to 8, as shown in Table 86. Example 2: FIFO_PATTERN_[9:0] bits and next reading.

Table 86. **Example 2: FIFO_PATTERN_[9:0] bits and next reading**

Time	FIFO_PATTERN_[9:0]	Next reading from FIFO output registers
t0	0	Gx
t0	1	Gy
t0	2	Gz
t0	3	XLx
t0	4	XLy
t0	5	XLz
t1	6	Gx
t1	7	Gy
t1	8	Gz

8.5.3 Example 3

Supposing the FIFO is storing data from the gyroscope, accelerometer and magnetometer at different ODRs:

- Gyroscope ODR = 104 Hz, Accelerometer ODR = 208 Hz, Magnetometer ODR = 52 Hz.

If the internal trigger (accelerometer/gyroscope data-ready) is used, it's recommended to set the ODR_FIFO_[3:0] bits of the FIFO_CTRL5 register to 0101b in order to set the FIFO trigger ODR to 208 Hz.

The DEC_FIFO_GYRO[2:0] field of the FIFO_CTRL3 register has to be set to 010b (decimation with factor 2 applied to gyroscope data) and the DEC_FIFO_XL[2:0] field has to be set to 001b (no decimation applied to accelerometer data). Assuming that the magnetometer is associated to the 3rd FIFO data set, the DEC_DS3_FIFO[2:0] field of the FIFO_CTRL4 register has to be set to 100b (decimation with factor 4 applied to magnetometer data).

The following data pattern is repeated every 21 samples:

- Gx Gy Gz XLx XLy XLz Mx My Mz (gyroscope, accelerometer, mag. data - 9 samples)
- XLx XLy XLz (accelerometer data - 3 samples)
- Gx Gy Gz XLx XLy XLz (gyroscope and accelerometer data - 6 samples)
- XLx XLy XLz (accelerometer data - 3 samples)

The FIFO_PATTERN_[9:0] bits will contain a number from 0 to 20, as shown in Table 87. Example 3: FIFO_PATTERN_[9:0] bits and next reading.

Table 87. **Example 3: FIFO_PATTERN_[9:0] bits and next reading**

Time	FIFO_PATTERN_[9:0]	Next reading from FIFO output registers
t0	0	Gx
t0	1	Gy
t0	2	Gz
t0	3	XLx
t0	4	XLy
t0	5	XLz
t0	6	Mx
t0	7	My
t0	8	Mz
t1	9	XLx
t1	10	XLy
t1	11	XLz
t2	12	Gx
t2	13	Gy
t2	14	Gz
t2	15	XLx
t2	16	XLy
t2	17	XLz
t3	18	XLx
t3	19	XLy
t3	20	XLz

8.6 FIFO threshold

The FIFO threshold is a functionality of the ISM330DLC FIFO which can be used to check when the number of samples in the FIFO reaches a defined threshold level.

The bits FTH_[10:0] in the FIFO_CTRL1 and FIFO_CTRL2 registers contain the threshold level. The resolution of the FTH_[10:0] field is two bytes (1 LSB = 2 bytes, each sample is represented as 16-bit data). So, the user can select the desired level in a range between 0 and 2047.

The bit WaterM in the FIFO_STATUS2 register represents the watermark status. This bit is set high if the number of samples in the FIFO reaches or exceeds the watermark level (each sample is represented as 16-bit data).

FIFO size can be limited to the threshold level by setting the STOP_ON_FTH bit in the FIFO_CTRL4 register to 1.

Figure 34. **FIFO threshold (STOP_ON_FTH = 0)**

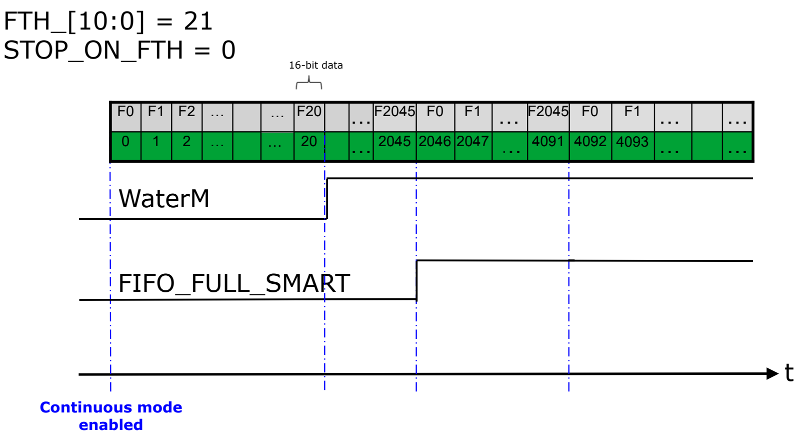


Figure 34. FIFO threshold (STOP_ON_FTH = 0) shows an example of FIFO threshold level usage when just accelerometer (or gyroscope) data are stored. The STOP_ON_FTH bit set to 0 in the FIFO_CTRL4 register. The threshold level is set to 21 through the FTH_[10:0] bits. The WaterM bit of the FIFO_STATUS2 register rises after the 21st level has been reached (21 samples in the FIFO). Since the STOP_ON_FTH bit is set to 0, the FIFO will not stop at the 21st sample, but will keep storing data until the FIFO_FULL_SMART flag is set high.

Figure 35. **FIFO threshold (STOP_ON_FTH = 1) in FIFO mode**

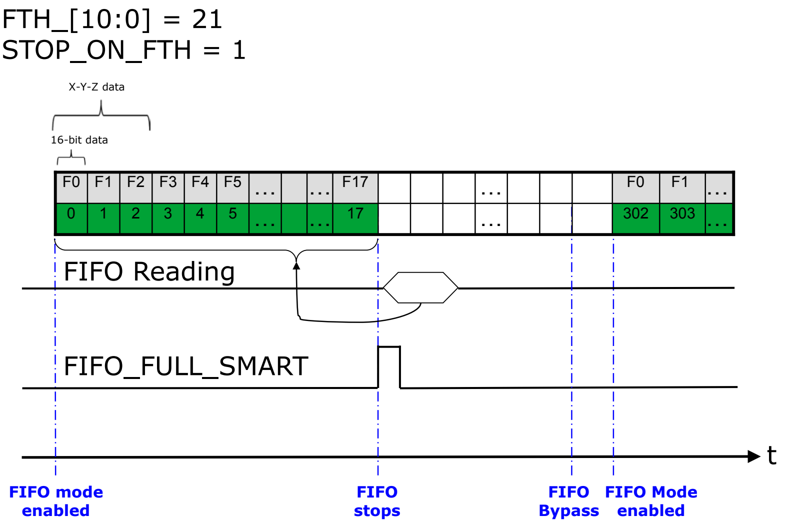


Figure 35. FIFO threshold (STOP_ON_FTH = 1) in FIFO mode shows an example of FIFO threshold level usage in FIFO mode with the STOP_ON_FTH bit set to 1 in the FIFO_CTRL4 register; just accelerometer (or gyroscope) data are stored in this example. The threshold level is set to 21 through the FTH_[10:0] bits and defines the current FIFO size. In FIFO mode, data are stored in the FIFO buffer until the FIFO_FULL_SMART signal rises; the FIFO_FULL_SMART bit of the FIFO_STATUS2 register rises when the next data stored in the FIFO will make the FIFO full, so in this example it rises after the first 18 data (16-bit each) are stored in FIFO. The WaterM bit of the FIFO_STATUS2 register cannot go to 1 since the FTH threshold level is never reached (data are no longer stored in FIFO after the FIFO is full).

Figure 36. **FIFO threshold (STOP_ON_FTH = 1) in Continuous mode**

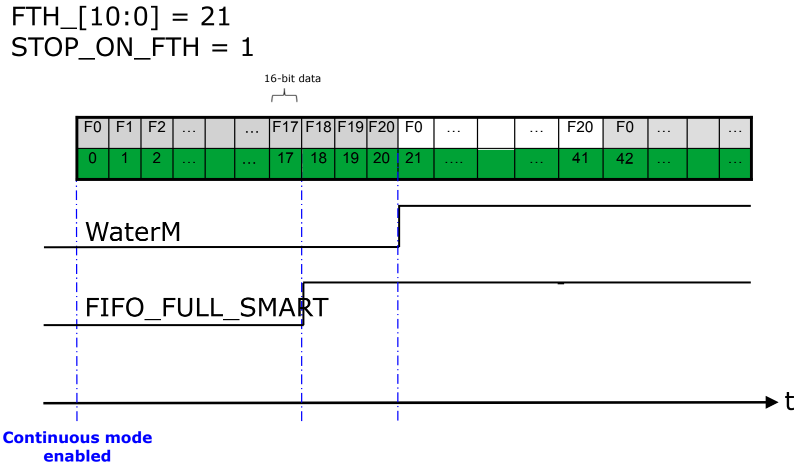


Figure 36. FIFO threshold (STOP_ON_FTH = 1) in Continuous mode shows an example of FIFO threshold level usage in Continuous mode with the STOP_ON_FTH bit set to 1 in the FIFO_CTRL4 register; just accelerometer (or gyroscope) data are stored in this example. The threshold level is set to 21 through the FTH_[10:0] bits. The FIFO_FULL_SMART bit of the FIFO_STATUS2 register rises when the next data stored in the FIFO will make the FIFO full, so in this example it rises after the first 18 data (16-bit each) are stored in FIFO. The WaterM bit of the FIFO_STATUS2 register rises after the 21st level has been reached (21 samples in the FIFO).

8.7 High part of gyroscope and accelerometer data

It is possible to increase the number of samples stored in the FIFO by storing just the high part (8 bits) of the gyroscope and accelerometer data. This feature is not valid for the other (external) sensors.

To the enable this feature, the bit ONLY_HIGH_DATA must be set to 1 in the FIFO_CTRL4 register. Gyroscope and accelerometer data will be written in the FIFO at the same ODR in the order shown in Table 88. High part of gyroscope and accelerometer data in FIFO.

Table 88. **High part of gyroscope and accelerometer data in FIFO**

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Accel_X_H	Gyro_X_H	Accel_Y_H	Gyro_Y_H	Accel_Z_H	Gyro_Z_H

When this feature is enabled, the 6 bytes containing the high part (8 bits) of the gyroscope and accelerometer data are associated to the 1st FIFO data set and the 2nd FIFO data set is not used.

The DEC_FIFO_G[2:0] field of the FIFO_CTRL3 register has to be set to a value different from 000b (1st FIFO data set stored in FIFO).

The DEC_FIFO_XL[2:0] field of the FIFO_CTRL3 register has to be set to 000b (2nd FIFO data set not in FIFO).

8.8 Timestamp data in FIFO

It is possible to store timestamp data in the FIFO. These data are stored as a 4th FIFO data set in the 6-byte data format shown in Table 89. Timestamp data in FIFO.

Table 89. Timestamp data in FIFO

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
TIMESTAMP [15:8]	TIMESTAMP [23:16]	-	TIMESTAMP [7:0]	-	-

To enable this feature, the FIFO_TIMER_EN bit must be set to 1 in the FIFO_CTRL2 register.

When this feature is enabled, the timestamp data is associated to the 4th FIFO data set: the DEC_DS4_FIFO[2:0] field of the FIFO_CTRL4 register has to be used to define the decimation factor.

Follow these steps to store timestamp data in the FIFO using the internal trigger (accelerometer/gyroscope data-ready):

1. Turn on the accelerometer;
2. Enable the timestamp (see Section 5.9 Timestamp);
3. Choose the decimation factor for the 4th FIFO data set through the DEC_DS4_FIFO[2:0] bits of the FIFO_CTRL4 register;
4. Set the FIFO_TIMER_EN bit to 1 in the FIFO_CTRL2 register;
5. Choose the FIFO ODR through the ODR_FIFO_[3:0] bits of the FIFO_CTRL5 register;
6. Configure the FIFO operating mode through the FIFO_MODE_[2:0] field of the FIFO_CTRL5 register.

8.9 Temperature data in FIFO

It is possible to store only temperature data as the 4th FIFO data set.

To enable this feature:

- Bit FIFO_TIMER_EN of the FIFO_CTRL2 register has to be set to 0;
- Bit FIFO_TEMP_EN of the FIFO_CTRL2 register has to be set to 1.

Temperature samples (16-bit) are stored in FIFO in the 6-byte data format shown in Table 90. Temperature data in FIFO.

Table 90. Temperature data in FIFO

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
-	-	TEMP [7:0]	TEMP [15:8]	-	-

Follow these steps to store 16-bit temperature data in the FIFO using the internal trigger (accelerometer/ gyroscope data-ready):

1. Turn on the accelerometer or the gyroscope;
2. Choose the decimation factor (different from 000b) for the 4th FIFO data set through the DEC_DS4_FIFO[2:0] bits in the FIFO_CTRL4 register;
3. Set to 1 the FIFO_TEMP_EN bit in the FIFO_CTRL2 register and to 0 the bit FIFO_TIMER_EN of the FIFO_CTRL2 register;
4. Choose the FIFO ODR through the ODR_FIFO_[3:0] bits of the FIFO_CTRL5 register;
5. Configure the FIFO operating mode through the FIFO_MODE_[2:0] field of the FIFO_CTRL5 register.

9 Temperature sensor

The ISM330DLC is provided with an internal temperature sensor that is suitable for ambient temperature measurement.

If both the accelerometer and the gyroscope sensors are in Power-Down mode, the temperature sensor is off.

The maximum output data rate of the temperature sensor is 52 Hz and its value depends on how the accelerometer and gyroscope sensors are configured:

- If the gyroscope is in Power-Down mode:
  - the temperature data rate is equal to 12.5 Hz if the accelerometer ODR is equal to 12.5 Hz Low-Power mode;
  - the temperature data rate is equal to 26 Hz if the accelerometer configuration is 26 Hz Low-Power mode ;
  - the temperature data rate is equal to 52 Hz for all other accelerometer configurations.
- If the gyroscope is not in Power-Down mode, the temperature data rate is equal to 52 Hz, regardless of the accelerometer and gyroscope configuration.

For the temperature sensor, the data-ready signal is represented by the TDA bit of the STATUS_REG register. The signal can be driven to the INT2 pin by setting the INT2_DRDY_TEMP bit of the INT2_CTRL register to 1.

The temperature data is given by the concatenation of the OUT_TEMP_H and OUT_TEMP_L registers and it is represented as a number of 16 bits in two's complement format with a sensitivity of 256 LSB/°C. The output zero level corresponds to 25 °C.

The ISM330DLC allows swapping, by setting the BLE bit of the CTRL3_C register to 1, the content of the lower and the upper part of the temperature output data registers (i.e. OUT_TEMP_H with OUT_TEMP_L).

Temperature sensor data can also be stored in FIFO with a configurable decimation factor (see Section 8.9 Temperature data in FIFO for details).

9.1 Example of temperature data calculation

Table 91. Output data registers content vs. temperature provides a few basic examples of the data that is read from the temperature data registers at different ambient temperature values. The values listed in this table are given under the hypothesis of perfect device calibration (i.e. no offset, no gain error,....).

Table 91. **Output data registers content vs. temperature**

Temperature values	BLE = 0		BLE = 1	
	Register address			
	OUT_TEMP_H (21h)	OUT_TEMP_L (20h)	OUT_TEMP_H (21h)	OUT_TEMP_L (20h)
0°C	E7h	00h	00h	E7
25°C	00h	00h	00h	00h
50°C	19h	00h	00h	19h

10 Self-test

The embedded self-test functions allows checking the device functionality without moving it.

10.1 Accelerometer self-test – Mode 1 / 2 / 3 / 4

When the accelerometer self-test is enabled, an actuation force is applied to the sensor, simulating a definite input acceleration. In this case, the sensor outputs exhibit a change in their DC levels which are related to the selected full scale through the sensitivity value.

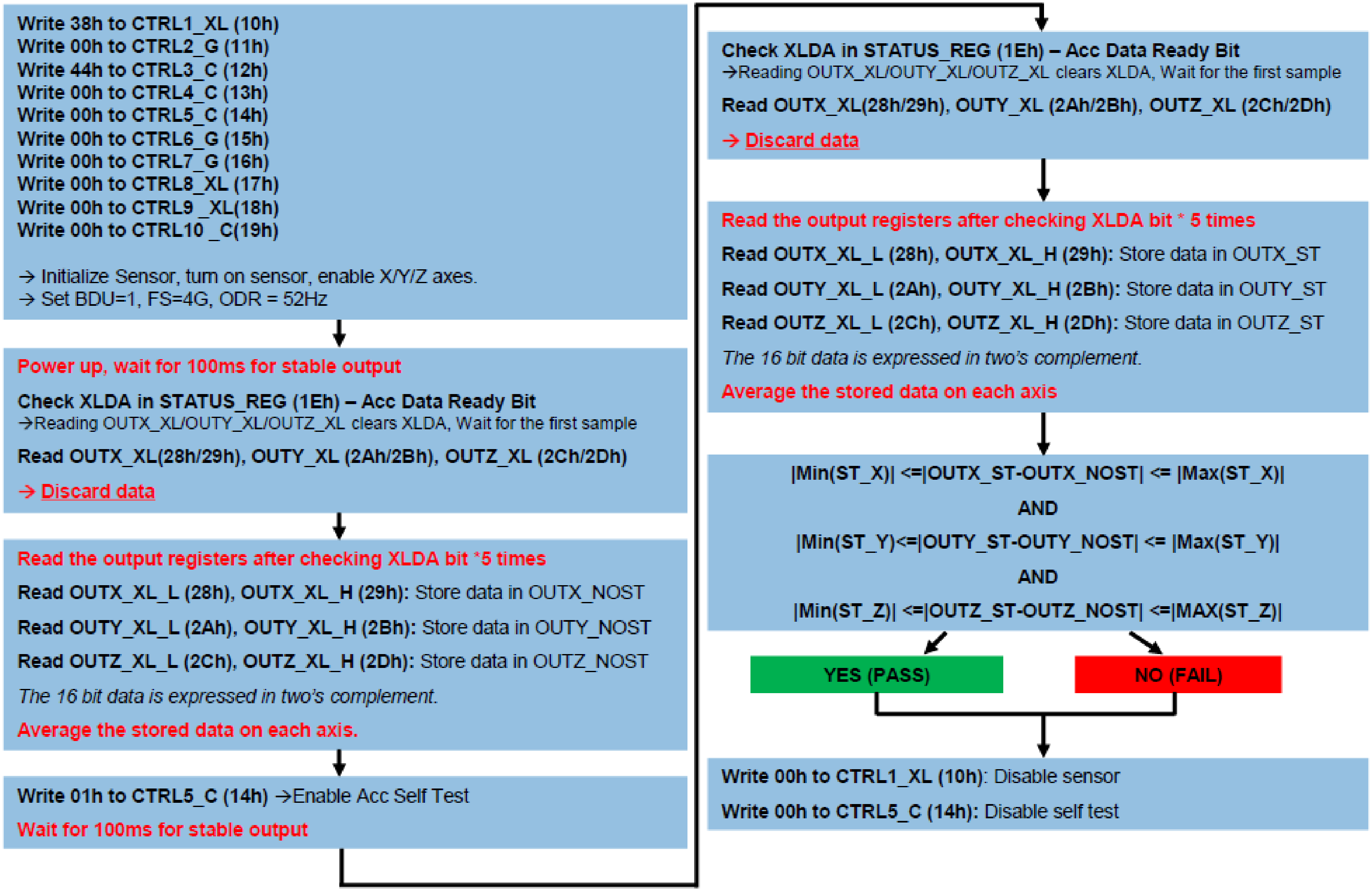
The accelerometer self-test function can be configured from the primary I^2C / SPI interface only (it's not possible to configure it through the Auxiliary SPI interface). It is off when the ST[1:0] _XL bits of the CTRL5_C register are programmed to 00b; it is enabled when the ST[1:0]_XL bits are set to 01b (positive sign self-test) or 10b (negative sign self-test).

When the accelerometer self-test is activated, the sensor output level is given by the algebraic sum of the signals produced by the acceleration acting on the sensor and by the electrostatic test-force. The accelerometer output variation is applied to both the GP chain and OIS chain and can be read from both interfaces.

The complete accelerometer self-test procedure is indicated in Figure 37. Accelerometer self-test procedure.

Figure 37. **Accelerometer self-test procedure**

Note: all the read/write operations in this procedure have to be performed **through the primary I2C/SPI interface**



Write 38h to CTRL1_XL (10h)
Write 00h to CTRL2_G (11h)
Write 44h to CTRL3_C (12h)
Write 00h to CTRL4_C (13h)
Write 00h to CTRL5_C (14h)
Write 00h to CTRL6_G (15h)
Write 00h to CTRL7_G (16h)
Write 00h to CTRL8_XL (17h)
Write 00h to CTRL9 _XL(18h)
Write 00h to CTRL10 _C(19h)

→ Initialize Sensor, turn on sensor, enable X/Y/Z axes.
→ Set BDU=1, FS=4G, ODR = 52Hz

Power up, wait for 100ms for stable output

Check XLDA in STATUS_REG (1Eh) – Acc Data Ready Bit
→Reading OUTX_XL/OUTY_XL/OUTZ_XL clears XLDA, Wait for the first sample

Read OUTX_XL(28h/29h), OUTY_XL (2Ah/2Bh), OUTZ_XL (2Ch/2Dh)

→ Discard data

Read the output registers after checking XLDA bit *5 times

Read OUTX_XL_L (28h), OUTX_XL_H (29h): Store data in OUTX_NOST

Read OUTY_XL_L (2Ah), OUTY_XL_H (2Bh): Store data in OUTY_NOST

Read OUTZ_XL_L (2Ch), OUTZ_XL_H (2Dh): Store data in OUTZ_NOST

The 16 bit data is expressed in two's complement.

Average the stored data on each axis.

Write 01h to CTRL5_C (14h) →Enable Acc Self Test

Wait for 100ms for stable output

Check XLDA in STATUS_REG (1Eh) – Acc Data Ready Bit
→Reading OUTX_XL/OUTY_XL/OUTZ_XL clears XLDA, Wait for the first sample

Read OUTX_XL(28h/29h), OUTY_XL (2Ah/2Bh), OUTZ_XL (2Ch/2Dh)

→ Discard data

Read the output registers after checking XLDA bit * 5 times

Read OUTX_XL_L (28h), OUTX_XL_H (29h): Store data in OUTX_ST

Read OUTY_XL_L (2Ah), OUTY_XL_H (2Bh): Store data in OUTY_ST

Read OUTZ_XL_L (2Ch), OUTZ_XL_H (2Dh): Store data in OUTZ_ST

The 16 bit data is expressed in two's complement.

Average the stored data on each axis

|Min(ST_X)| <=|OUTX_ST-OUTX_NOST| <= |Max(ST_X)|

AND

|Min(ST_Y)<=|OUTY_ST-OUTY_NOST| <= |Max(ST_Y)|

AND

|Min(ST_Z)| <=|OUTZ_ST-OUTZ_NOST| <=|MAX(ST_Z)|

YES (PASS)

NO (FAIL)

Write 00h to CTRL1_XL (10h): Disable sensor

Write 00h to CTRL5_C (14h): Disable self test

10.2 Gyroscope self-test (GP) – Mode 1 / 2

The gyroscope self-test allows testing the mechanical and electrical parts of the gyroscope sensor: when it is activated, an actuation force is applied to the sensor, emulating a definite Coriolis force and the seismic mass is moved by means of this electrostatic test-force. In this case, the sensor output exhibits an output change.

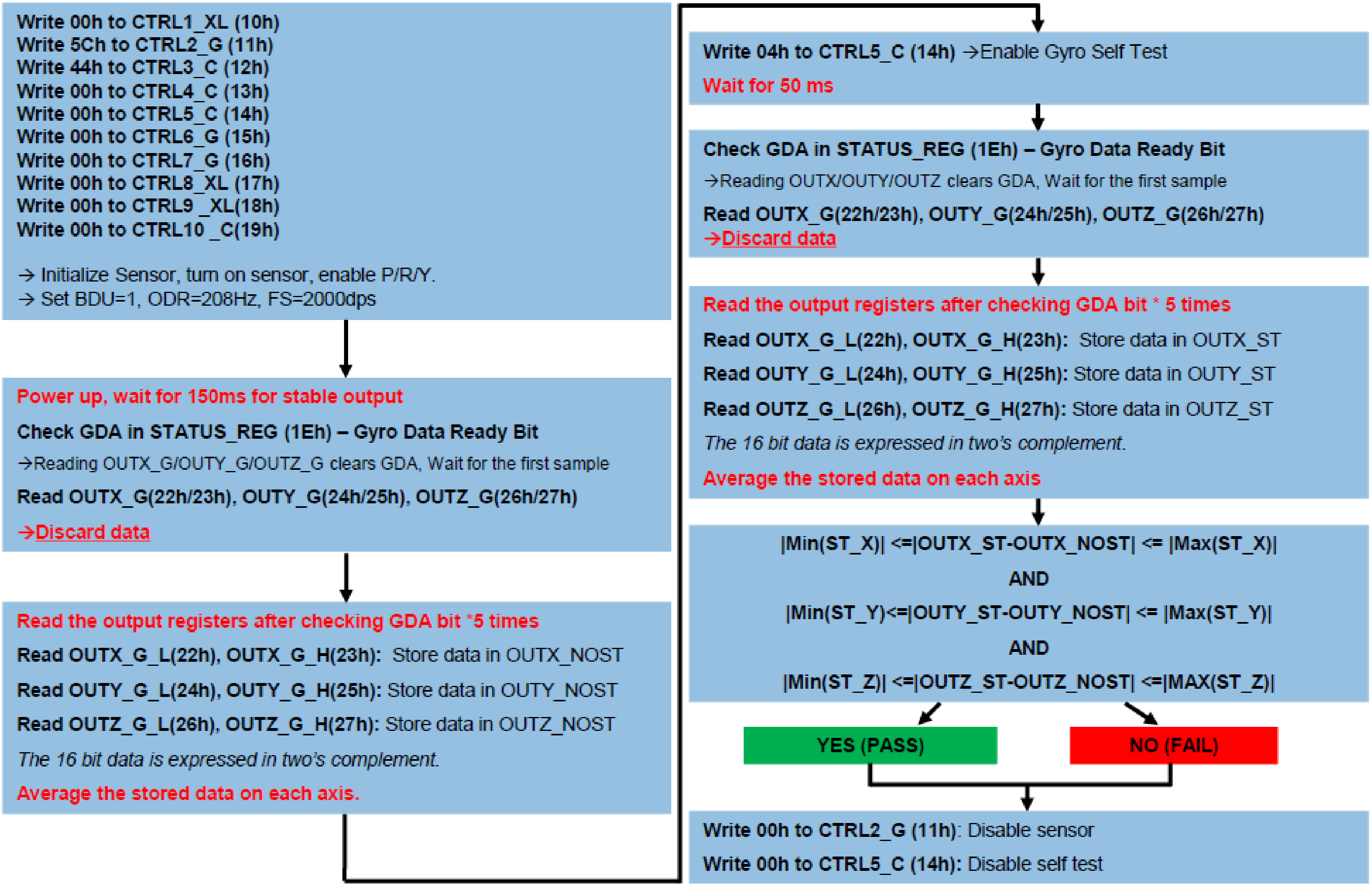
When the device is configured in Mode 1 or Mode 2, the gyroscope self-test function can be configured from the primary I^2C / SPI interface only. It is off when the ST[1:0]_G bits of the CTRL5_C register are programmed to 00b; it is enabled when the ST[1:0]_G bits are set to 01b (positive sign self-test) or 11b (negative sign self-test).

When the gyroscope self-test is active, the sensor output level is given by the algebraic sum of the signals produced by the angular rate acting on the sensor and by the electrostatic test-force.

The complete gyroscope self-test procedure in Mode 1/2 is indicated in Figure 38. Gyroscope self-test procedure in Mode 1/2.

Figure 38. Gyroscope self-test procedure in Mode 1/2

Note: all the read/write operations in this procedure have to be performed **through the primary I2C/SPI interface**



Write 00h to CTRL1_XL (10h)
Write 5Ch to CTRL2_G (11h)
Write 44h to CTRL3_C (12h)
Write 00h to CTRL4_C (13h)
Write 00h to CTRL5_C (14h)
Write 00h to CTRL6_G (15h)
Write 00h to CTRL7_G (16h)
Write 00h to CTRL8_XL (17h)
Write 00h to CTRL9 _XL(18h)
Write 00h to CTRL10 _C(19h)

→ Initialize Sensor, turn on sensor, enable P/R/Y.
→ Set BDU=1, ODR=208Hz, FS=2000dps

Power up, wait for 150ms for stable output

Check GDA in STATUS_REG (1Eh) – Gyro Data Ready Bit

→Reading OUTX_G/OUTY_G/OUTZ_G clears GDA, Wait for the first sample

Read OUTX_G(22h/23h), OUTY_G(24h/25h), OUTZ_G(26h/27h)

→Discard data

Read the output registers after checking GDA bit *5 times

Read OUTX_G_L(22h), OUTX_G_H(23h): Store data in OUTX_NOST

Read OUTY_G_L(24h), OUTY_G_H(25h): Store data in OUTY_NOST

Read OUTZ_G_L(26h), OUTZ_G_H(27h): Store data in OUTZ_NOST

The 16 bit data is expressed in two's complement.

Average the stored data on each axis.

Write 04h to CTRL5_C (14h) →Enable Gyro Self Test

Wait for 50 ms

Check GDA in STATUS_REG (1Eh) – Gyro Data Ready Bit

→Reading OUTX/OUTY/OUTZ clears GDA, Wait for the first sample

Read OUTX_G(22h/23h), OUTY_G(24h/25h), OUTZ_G(26h/27h)
→Discard data

Read the output registers after checking GDA bit * 5 times

Read OUTX_G_L(22h), OUTX_G_H(23h): Store data in OUTX_ST

Read OUTY_G_L(24h), OUTY_G_H(25h): Store data in OUTY_ST

Read OUTZ_G_L(26h), OUTZ_G_H(27h): Store data in OUTZ_ST

The 16 bit data is expressed in two's complement.

Average the stored data on each axis

|Min(ST_X)| <=|OUTX_ST-OUTX_NOST| <= |Max(ST_X)|

AND

|Min(ST_Y)<=|OUTY_ST-OUTY_NOST| <= |Max(ST_Y)|

AND

|Min(ST_Z)| <=|OUTZ_ST-OUTZ_NOST| <=|MAX(ST_Z)|

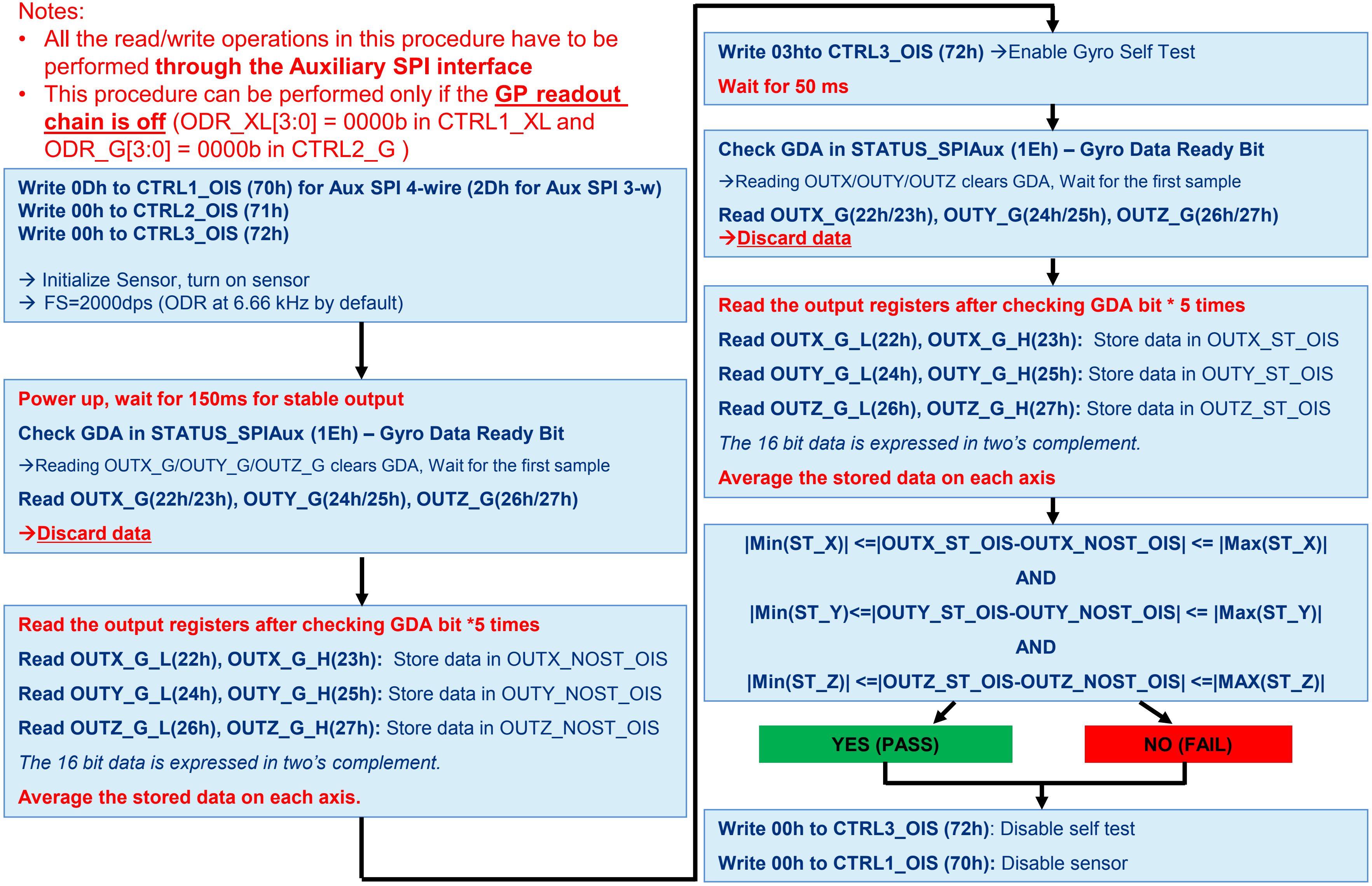
Write 00h to CTRL2_G (11h): Disable sensor
Write 00h to CTRL5_C (14h): Disable self test

10.3 Gyroscope self-test (OIS) with GP chain off – Mode 3 / 4

In case the GP chain is off, the gyroscope self-test function on the OIS chain can be enabled through the Auxiliary SPI interface by setting the ST[1:0]_OIS bits of the CTRL3_OIS register. The self-test function is off when the ST[1:0]_OIS bits are programmed to 00b; it is enabled when the ST[1:0]_OIS bits are set to 01b (positive sign self-test) or 11b (negative sign self-test).

The complete gyroscope self-test procedure on the OIS chain with the GP chain off is indicated in Figure 39. Gyroscope self-test procedure (OIS) with GP chain off. This procedure can be performed only if the GP readout chain is off (ODR_XL[3:0] = 0000b in CTRL1_XL register and ODR_G[3:0] = 0000b in CTRL2_G register).

Figure 39. **Gyroscope self-test procedure (OIS) with GP chain off**



Notes:

- All the read/write operations in this procedure have to be performed **through the Auxiliary SPI interface**
- This procedure can be performed only if the **GP readout chain is off** (ODR_XL[3:0] = 0000b in CTRL1_XL and ODR_G[3:0] = 0000b in CTRL2_G)

Write 0Dh to CTRL1_OIS (70h) for Aux SPI 4-wire (2Dh for Aux SPI 3-w)
Write 00h to CTRL2_OIS (71h)
Write 00h to CTRL3_OIS (72h)

→ Initialize Sensor, turn on sensor
→ FS=2000dps (ODR at 6.66 kHz by default)

Power up, wait for 150ms for stable output

Check GDA in STATUS_SPIAux (1Eh) – Gyro Data Ready Bit

→Reading OUTX_G/OUTY_G/OUTZ_G clears GDA, Wait for the first sample

Read OUTX_G(22h/23h), OUTY_G(24h/25h), OUTZ_G(26h/27h)

→Discard data

Read the output registers after checking GDA bit *5 times

Read OUTX_G_L(22h), OUTX_G_H(23h): Store data in OUTX_NOST_OIS

Read OUTY_G_L(24h), OUTY_G_H(25h): Store data in OUTY_NOST_OIS

Read OUTZ_G_L(26h), OUTZ_G_H(27h): Store data in OUTZ_NOST_OIS

The 16 bit data is expressed in two's complement.

Average the stored data on each axis.

Write 03hto CTRL3_OIS (72h) →Enable Gyro Self Test

Wait for 50 ms

Check GDA in STATUS_SPIAux (1Eh) – Gyro Data Ready Bit

→Reading OUTX/OUTY/OUTZ clears GDA, Wait for the first sample

Read OUTX_G(22h/23h), OUTY_G(24h/25h), OUTZ_G(26h/27h)
→Discard data

Read the output registers after checking GDA bit * 5 times

Read OUTX_G_L(22h), OUTX_G_H(23h): Store data in OUTX_ST_OIS

Read OUTY_G_L(24h), OUTY_G_H(25h): Store data in OUTY_ST_OIS

Read OUTZ_G_L(26h), OUTZ_G_H(27h): Store data in OUTZ_ST_OIS

The 16 bit data is expressed in two's complement.

Average the stored data on each axis

|Min(ST_X)| <=|OUTX_ST_OIS-OUTX_NOST_OIS| <= |Max(ST_X)|

AND

|Min(ST_Y)<=|OUTY_ST_OIS-OUTY_NOST_OIS| <= |Max(ST_Y)|

AND

|Min(ST_Z)| <=|OUTZ_ST_OIS-OUTZ_NOST_OIS| <=|MAX(ST_Z)|

YES (PASS)

NO (FAIL)

Write 00h to CTRL3_OIS (72h): Disable self test

Write 00h to CTRL1_OIS (70h): Disable sensor

10.4 Gyroscope self-test (OIS) with GP chain on – Mode 3 / 4

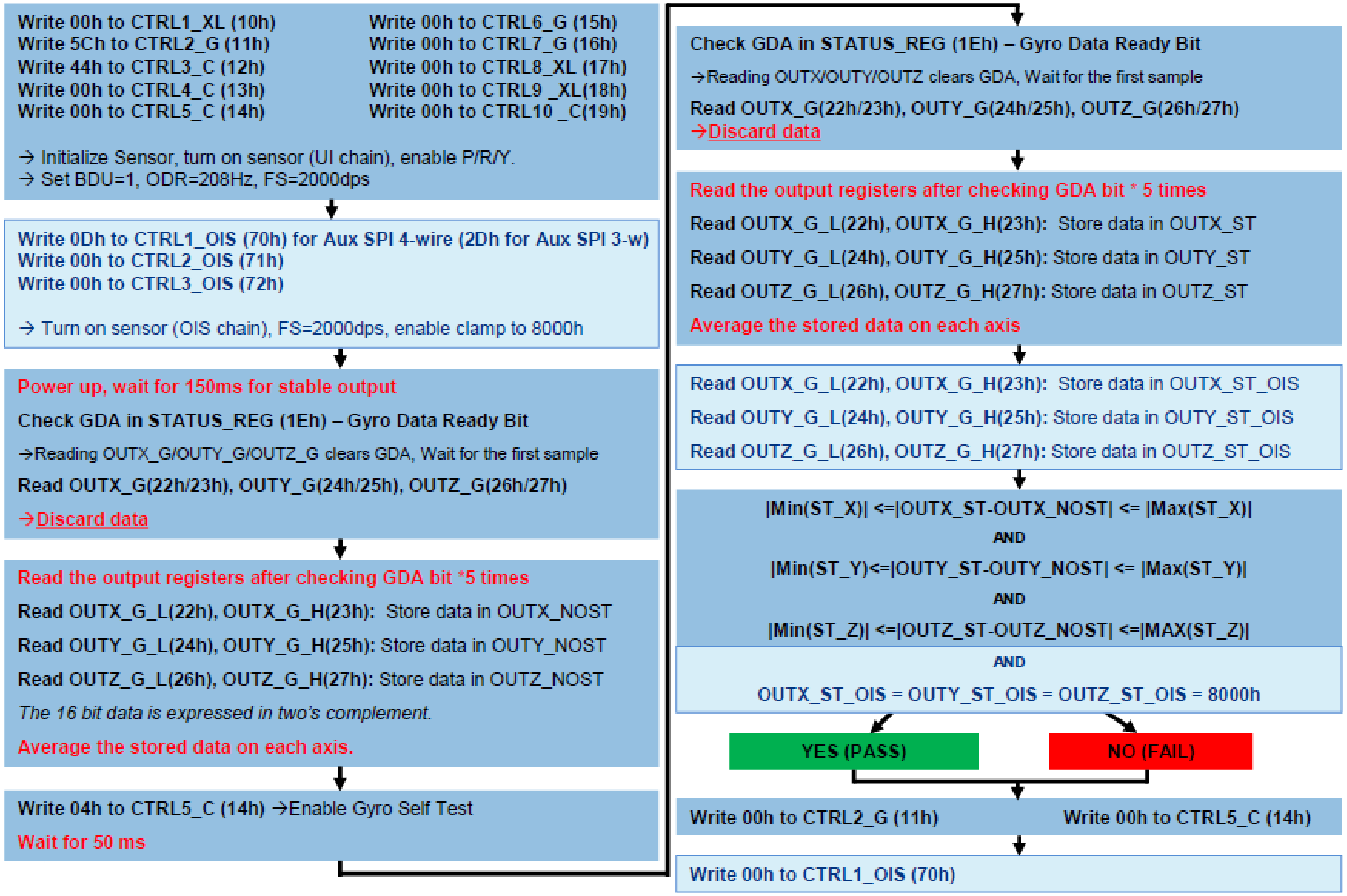
In case both the GP chain and the OIS chain are on, the gyroscope self-test function has to be enabled from one interface only: through the ST[1:0]_G bits of the CTRL5_C register if the primary is used or through the ST[1:0]_OIS bits of the CTRL3_OIS register if the Auxiliary interface is used. It cannot be enabled from both interfaces at the same time (forbidden condition).

The recommended gyroscope self-test procedure on the OIS chain with the GP chain on is indicated in Figure 40. Gyroscope self-test procedure (OIS) with GP chain on.

Figure 40. **Gyroscope self-test procedure (OIS) with GP chain on**

Notes:

- All the black colored read/write operations have to be performed through the primary I2C/SPI interface
- All the blue colored read/write operations have to be performed through the Auxiliary SPI interface



Write 00h to CTRL1_XL (10h)
Write 5Ch to CTRL2_G (11h)
Write 44h to CTRL3_C (12h)
Write 00h to CTRL4_C (13h)
Write 00h to CTRL5_C (14h)

Write 00h to CTRL6_G (15h)
Write 00h to CTRL7_G (16h)
Write 00h to CTRL8_XL (17h)
Write 00h to CTRL9 _XL(18h)
Write 00h to CTRL10 _C(19h)

→ Initialize Sensor, turn on sensor (UI chain), enable P/R/Y.
→ Set BDU=1, ODR=208Hz, FS=2000dps

Write 0Dh to CTRL1_OIS (70h) for Aux SPI 4-wire (2Dh for Aux SPI 3-w)
Write 00h to CTRL2_OIS (71h)
Write 00h to CTRL3_OIS (72h)

→ Turn on sensor (OIS chain), FS=2000dps, enable clamp to 8000h

Power up, wait for 150ms for stable output

Check GDA in STATUS_REG (1Eh) – Gyro Data Ready Bit

→Reading OUTX_G/OUTY_G/OUTZ_G clears GDA, Wait for the first sample

Read OUTX_G(22h/23h), OUTY_G(24h/25h), OUTZ_G(26h/27h)

→Discard data

Read the output registers after checking GDA bit *5 times

Read OUTX_G_L(22h), OUTX_G_H(23h): Store data in OUTX_NOST

Read OUTY_G_L(24h), OUTY_G_H(25h): Store data in OUTY_NOST

Read OUTZ_G_L(26h), OUTZ_G_H(27h): Store data in OUTZ_NOST

The 16 bit data is expressed in two's complement.

Average the stored data on each axis.

Write 04h to CTRL5_C (14h) →Enable Gyro Self Test

Wait for 50 ms

Check GDA in STATUS_REG (1Eh) – Gyro Data Ready Bit

→Reading OUTX/OUTY/OUTZ clears GDA, Wait for the first sample

Read OUTX_G(22h/23h), OUTY_G(24h/25h), OUTZ_G(26h/27h)

→Discard data

Read the output registers after checking GDA bit * 5 times

Read OUTX_G_L(22h), OUTX_G_H(23h): Store data in OUTX_ST

Read OUTY_G_L(24h), OUTY_G_H(25h): Store data in OUTY_ST

Read OUTZ_G_L(26h), OUTZ_G_H(27h): Store data in OUTZ_ST

Average the stored data on each axis

Read OUTX_G_L(22h), OUTX_G_H(23h): Store data in OUTX_ST_OIS

Read OUTY_G_L(24h), OUTY_G_H(25h): Store data in OUTY_ST_OIS

Read OUTZ_G_L(26h), OUTZ_G_H(27h): Store data in OUTZ_ST_OIS

|Min(ST_X)| <=|OUTX_ST-OUTX_NOST| <= |Max(ST_X)|

AND

|Min(ST_Y)<=|OUTY_ST-OUTY_NOST| <= |Max(ST_Y)|

AND

|Min(ST_Z)| <=|OUTZ_ST-OUTZ_NOST| <=|MAX(ST_Z)|

AND

OUTX_ST_OIS = OUTY_ST_OIS = OUTZ_ST_OIS = 8000h

YES (PASS)

NO (FAIL)

Write 00h to CTRL2_G (11h)

Write 00h to CTRL5_C (14h)

Write 00h to CTRL1_OIS (70h)

Revision history

Table 92. **Document revision history**

Date	Revision	Changes
11-Jan-2018	1	Initial release
27-Aug-2018	2	Updated Section 3.7 Accelerometer bandwidth Updated Table 22. GP chains settling time on OIS enable/disable Updated Section 5.8 Relative tilt Updated Section 6.2.3 FUNC_SRC1 (53h) Updated Table 65. Accelerometer OIS bandwidth selection Updated Section 8.2.2 FIFO mode Updated Section 8.2.3 Continuous mode Updated Figure 31. Continuous-to-FIFO mode Updated Figure 34. FIFO threshold (STOP_ON_FTH = 0) Updated Figure 35. FIFO threshold (STOP_ON_FTH = 1) in FIFO mode Updated Figure 36. FIFO threshold (STOP_ON_FTH = 1) in Continuous mode Updated Section 10.3 Gyroscope self-test (OIS) with GP chain off – Mode 3 / 4

Contents

1 Pin description 2

2 Registers 4

2.1 Embedded functions registers 8

3 Operating modes 10

3.1 Power-Down mode 12

3.2 High-Performance mode 12

3.3 Normal mode 12

3.4 Low-Power mode 12

3.5 Gyroscope Sleep mode 12

3.6 Connection modes 12

3.7 Accelerometer bandwidth 12

3.7.1 Accelerometer slope filter 15

3.8 Gyroscope bandwidth 17

3.9 Accelerometer and gyroscope turn-on/off time 20

3.9.1 GP chain settling time on OIS chain enable/disable 24

4 Mode 1 - Reading output data 26

4.1 Startup sequence 26

4.2 Using the status register 26

4.3 Using the data-ready signal 27

4.3.1 DRDY mask functionality 27

4.4 Using the block data update (BDU) feature 28

4.5 Understanding output data 28

4.5.1 Big-little endian selection 28

4.5.2 Examples of output data 29

4.6 Accelerometer offset registers 29

4.7 Rounding functions 29

4.7.1 Rounding of FIFO output registers 30

4.7.2 Rounding of source registers 30

4.7.3 Rounding of sensor output registers 30

4.8 Edge-sensitive and level-sensitive data enable (DEN) 31
4.8.1 Edge-sensitive trigger mode 31
4.8.2 Level-sensitive trigger mode 32
4.8.3 Level-sensitive latched mode 33
4.8.4 Level-sensitive FIFO enabled 34
4.8.5 LSB selection for DEN stamping 34
4.8.6 OIS DEN mode 35
5 Interrupt generation 36
5.1 Interrupt pin configuration 36
5.2 Free-fall interrupt 38
5.3 Wake-up interrupt 39
5.4 6D/4D orientation detection 41
5.4.1 6D orientation detection 41
5.4.2 4D orientation detection 43
5.5 Single-tap and double-tap recognition 44
5.5.1 Single tap 45
5.5.2 Double tap 46
5.5.3 Single-tap and double-tap recognition configuration 47
5.5.4 Single-tap example 48
5.5.5 Double-tap example 49
5.6 Activity/Inactivity recognition 49
5.7 Boot status 52
5.8 Relative tilt 52
5.9 Timestamp 54
6 Mode 2 - Sensor hub mode 55
6.1 Sensor hub mode description 55
6.2 Sensor hub mode registers 55
6.2.1 **CTRL10_C (19h)** 56
6.2.2 MASTER_CONFIG (1Ah) 56
6.2.3 FUNC_SRC1 (53h) 57
6.2.4 **FUNC_SRC2 (54h)** 57

6.2.5 SLV0_ADD (02h), SLV0_SUBADD (03h), SLAVE0_CONFIG (04h) 58
6.2.6 SLV1_ADD (05h), SLV1_SUBADD (06h), SLAVE1_CONFIG (07h) 59
6.2.7 **SLV2_ADD (08h), SLV2_SUBADD (09h), SLAVE2_CONFIG (0Ah)** 60
6.2.8 SLV3_ADD (0Bh), SLV3_SUBADD (0Ch), SLAVE3_CONFIG (0Dh). 61
6.2.9 **DATAWRITE_SRC_MODE_SUB_SLV0 (0Eh)** 62
6.2.10 **SENSORHUBx_REG registers** 62
6.3 Sensor hub pass-through feature 63
6.3.1 Pass-through feature enable 64
6.3.2 Pass-through feature disable 64
6.4 Sensor hub mode example 65
6.5 Magnetometer hard-iron / soft-iron correction 65
6.5.1 Hard-iron correction 66
6.5.2 Soft-iron correction 66
6.5.3 Getting compensated magnetometer data 67
6.5.4 **Ironing example** 68
7 Mode 3 and Mode 4 - Auxiliary SPI modes 71
7.1 Auxiliary SPI mode description 71
7.2 Auxiliary SPI mode registers 72
7.2.1 **INT_OIS (6Fh)** 72
7.2.2 CTRL1_OIS (70h) 73
7.2.3 CTRL2_OIS (71h) 73
7.2.4 CTRL3_OIS (72h) 74
7.2.5 **STATUS_SPIAux (1Eh)** 76
7.3 Reading gyroscope data through the Auxiliary SPI. 76
7.4 Mode 4 - Reading gyroscope and accelerometer output data through the Auxiliary SPI . . .76
8 First-in first-out (FIFO) buffer 77
8.1 FIFO registers 78
8.1.1 **FIFO_CTRL1 (06h)** 78
8.1.2 **FIFO_CTRL2 (07h)** 78
8.1.3 **FIFO_CTRL3 (08h)** 79
8.1.4 FIFO_CTRL4 (09h) 80
8.1.5 FIFO_CTRL5 (0Ah). 80

8.1.6 FIFO_STATUS1 (3Ah) 81
8.1.7 **FIFO_STATUS2 (3Bh)** 82
8.1.8 **FIFO_STATUS3 (3Ch)** 82
8.1.9 **FIFO_STATUS4 (3Dh)** 83
8.1.10 **FIFO_DATA_OUT_L (3Eh)** 83
8.1.11 FIFO_DATA_OUT_H (3Fh) 83
8.2 FIFO modes 83
8.2.1 Bypass mode 84
8.2.2 FIFO mode 84
8.2.3 Continuous mode 86
8.2.4 **Continuous-to-FIFO mode** 87
8.2.5 **Bypass-to-Continuous mode** 89
8.3 Setting the FIFO trigger, FIFO ODR and decimation factors 89
8.3.1 **Procedure for ODR or FIFO configuration changes when using FIFO** 91
8.4 Retrieving data from the FIFO 91
8.5 FIFO pattern 92
8.5.1 Example 1 92
8.5.2 Example 2 93
8.5.3 Example 3 94
8.6 FIFO threshold 95
8.7 High part of gyroscope and accelerometer data 96
8.8 Timestamp data in FIFO 97
8.9 Temperature data in FIFO 98
9 Temperature sensor 99
9.1 Example of temperature data calculation 99
10 Self-test 100
10.1 Accelerometer self-test – Mode 1 / 2 / 3 / 4 100
10.2 Gyroscope self-test (GP) – Mode 1 / 2 102
10.3 Gyroscope self-test (OIS) with GP chain off – Mode 3 / 4 104
10.4 Gyroscope self-test (OIS) with GP chain on – Mode 3 / 4 106
Revision history 108

List of tables

Table 1. Pin status . . . 3
Table 2. Registers . . . 4
Table 3. Embedded functions registers . . . 8
Table 4. Accelerometer ODR and power mode selection . . . 10
Table 5. Gyroscope ODR and power mode selection . . . 11
Table 6. Power consumption . . . 11
Table 7. Accelerometer analog filter bandwidth . . . 13
Table 8. Accelerometer bandwidth selection in Mode 1/2/3 . . . 14
Table 9. OIS chain (XL ODR = 6.66 kHz) - Accelerometer bandwidth selection in Mode 4 . . . 15
Table 10. Gyroscope digital HP filter cutoff selection. . . . 17
Table 11. Gyroscope overall bandwidth selection in Mode 1/2 . . . 18
Table 12. GP chain - Gyroscope overall bandwidth selection in Mode 3/4 . . . 19
Table 13. OIS chain - Gyroscope digital HP filter cutoff selection in Mode 3/4. . . . 20
Table 14. OIS chain (Gyro ODR = 6.66 kHz) - Gyroscope overall bandwidth selection (Mode 3/4) . . . 20
Table 15. Accelerometer turn-on/off time in Mode 1/2/3 (LPF2 and HP disabled) . . . 21
Table 16. Accelerometer samples to be discarded in Mode 1/2/3 . . . 21
Table 17. Gyroscope turn-on/off time in Mode 1/2 (HP disabled) . . . 22
Table 18. Gyroscope samples to be discarded in Mode 1/2 (LPF1 disabled) . . . 23
Table 19. Gyroscope samples to be discarded in Mode 1/2 (LPF1 enabled) for all ODRs . . . 23
Table 20. OIS chain (Gyro ODR = 6.66 kHz) - Gyroscope turn-on/off time in Mode 3/4 . . . 24
Table 21. Gyroscope samples to be discarded in Mode 3/4 . . . 24
Table 22. GP chains settling time on OIS enable/disable . . . 25
Table 23. Output data registers content vs. acceleration (FS_XL = ±2 *g*) . . . 29
Table 24. Output data registers content vs. angular rate (FS_G = ±250 dps) . . . 29
Table 25. Output registers rounding pattern . . . 30
Table 26. DEN configurations . . . 31
Table 27. INT1_CTRL register . . . 36
Table 28. MD1_CFG register . . . 37
Table 29. INT2_CTRL register . . . 37
Table 30. MD2_CFG register . . . 37
Table 31. Free-fall threshold LSB value. . . . 38
Table 32. D6D_SRC register . . . 41
Table 33. Threshold for 4D/6D function . . . 42
Table 34. D6D_SRC register in 6D positions . . . 43
Table 35. TAP_SRC register . . . 48
Table 36. Inactivity event configuration . . . 50
Table 37. CTRL10_C register . . . 56
Table 38. MASTER_CONFIG register. . . . 56
Table 39. FUNC_SRC1 register . . . 57
Table 40. FUNC_SRC2 register . . . 57
Table 41. SLV0_ADD register . . . 58
Table 42. SLV0_SUBADD register . . . 58
Table 43. SLAVE0_CONFIG register . . . 58
Table 44. SLV1_ADD register . . . 59
Table 45. SLV1_SUBADD register . . . 59
Table 46. SLAVE1_CONFIG register . . . 59
Table 47. SLV2_ADD register . . . 60
Table 48. SLV2_SUBADD register . . . 60
Table 49. SLAVE2_CONFIG register . . . 60
Table 50. SLV3_ADD register . . . 61
Table 51. SLV3_SUBADD register . . . 61
Table 52. SLAVE3_CONFIG register . . . 61

Table 53. DATAWRITE_SRC_MODE_SUB_SLV0 register. . . . 62
Table 54. Ironing configuration. . . . 66
Table 55. Hard-iron register values. . . . 68
Table 56. Soft-iron register values 69
Table 57. Mode 3/4 pin description. . . . 71
Table 58. INT_OIS register 73
Table 59. CTRL1_OIS register. . . . 73
Table 60. DEN mode selection. . . . 73
Table 61. CTRL2_OIS register. . . . 74
Table 62. Gyroscope OIS chain HPF cutoff selection 74
Table 63. Gyroscope OIS chain LPF1 bandwidth selection 74
Table 64. CTRL3_OIS register. . . . 75
Table 65. Accelerometer OIS bandwidth selection 75
Table 66. STATUS_SPIAux register 76
Table 67. FIFO_CTRL1 register 78
Table 68. FIFO_CTRL2 register 78
Table 69. FIFO_CTRL3 register 79
Table 70. Gyroscope FIFO decimation setting 79
Table 71. Accelerometer FIFO decimation setting 79
Table 72. FIFO_CTRL4 register 80
Table 73. 3rd FIFO data set decimation setting. . . . 80
Table 74. 4th FIFO data set decimation setting 80
Table 75. FIFO_CTRL5 register 81
Table 76. FIFO ODR selection setting. . . . 81
Table 77. FIFO mode selection 81
Table 78. FIFO_STATUS1 register 82
Table 79. FIFO_STATUS2 register 82
Table 80. FIFO_STATUS2 behavior (case with one sensor in FIFO, STOP_ON_FTH = 0) 82
Table 81. FIFO_STATUS3 register 83
Table 82. FIFO_STATUS4 register 83
Table 83. FIFO_DATA_OUT_L register 83
Table 84. FIFO_DATA_OUT_H register. . . . 83
Table 85. Example 1: FIFO_PATTERN_[9:0] bits and next reading 92
Table 86. Example 2: FIFO_PATTERN_[9:0] bits and next reading 93
Table 87. Example 3: FIFO_PATTERN_[9:0] bits and next reading 94
Table 88. High part of gyroscope and accelerometer data in FIFO 96
Table 89. Timestamp data in FIFO 97
Table 90. Temperature data in FIFO 98
Table 91. Output data registers content vs. temperature 99
Table 92. Document revision history 108

List of figures

Figure 1. Pin connections . . . 2
Figure 2. Accelerometer filtering chain (Mode 1/2/3) . . . 13
Figure 3. Accelerometer filtering chain (Mode 4) . . . 15
Figure 4. Accelerometer slope filter . . . 16
Figure 5. Gyroscope digital chain - Mode 1 and Mode 2 . . . 17
Figure 6. Gyroscope digital chain - Mode 3 and Mode 4 . . . 20
Figure 7. Data-ready signal . . . 27
Figure 8. Edge-sensitive trigger mode, DEN active low . . . 32
Figure 9. Level-sensitive trigger mode, DEN active low . . . 33
Figure 10. Level-sensitive trigger mode, DEN active low, DEN_DRDY on INT1 . . . 33
Figure 11. Level-sensitive latched mode, DEN active low . . . 34
Figure 12. Level-sensitive latched mode, DEN active low, DEN_DRDY on INT1 . . . 34
Figure 13. Level-sensitive FIFO enable mode, DEN active low . . . 34
Figure 14. Free-fall interrupt . . . 38
Figure 15. Wake-up interrupt (using the slope filter) . . . 40
Figure 16. 6D recognized orientations . . . 43
Figure 17. Single-tap event recognition . . . 45
Figure 18. Double-tap event recognition (LIR bit = 0) . . . 46
Figure 19. Single and double-tap recognition (LIR bit = 0) . . . 48
Figure 20. Activity/Inactivity recognition (using the slope filter) . . . 51
Figure 21. Tilt example . . . 53
Figure 22. External sensor connections in Mode 2 . . . 55
Figure 23. SENSORHUBx_REG allocation example . . . 62
Figure 24. Pass-through feature . . . 63
Figure 25. Hard-iron effect (X-Y 2D scatter plot) . . . 66
Figure 26. Soft-iron effect (X-Y 2D scatter plot) . . . 67
Figure 27. Hard-iron / soft-iron correction block scheme . . . 68
Figure 28. External controller connection in Mode 3/4 (SPI 3-wire) . . . 72
Figure 29. FIFO mode (STOP_ON_FTH = 0) . . . 85
Figure 30. Continuous mode . . . 86
Figure 31. Continuous-to-FIFO mode . . . 87
Figure 32. Bypass-to-Continuous mode . . . 89
Figure 33. FIFO trigger signal selection . . . 90
Figure 34. FIFO threshold (STOP_ON_FTH = 0) . . . 95
Figure 35. FIFO threshold (STOP_ON_FTH = 1) in FIFO mode . . . 95
Figure 36. FIFO threshold (STOP_ON_FTH = 1) in Continuous mode . . . 96
Figure 37. Accelerometer self-test procedure . . . 101
Figure 38. Gyroscope self-test procedure in Mode 1/2 . . . 103
Figure 39. Gyroscope self-test procedure (OIS) with GP chain off . . . 105
Figure 40. Gyroscope self-test procedure (OIS) with GP chain on . . . 107