

ASM330LHH: 汽车惯性模块数字 3D 加速度计和数字 3D 陀螺仪

引言

本文档旨在提供 ST ASM330LHH 汽车惯性模块相关的使用信息和应用提示。

ASM330LHH 是系统级封装的 3D 数字加速度计和 3D 数字陀螺仪，具有数字 I²C/SPI 串口标准输出，组合工作在高性能模式下功耗只要 1.3 mA。加速度计具有智能的休眠到唤醒（活动）和返回休眠（不活动）功能，具备先进的节电能力。

该器件具有动态的用户可选择的满量程加速度范围： $\pm 2/\pm 4/\pm 8/\pm 16$ g，且角速率范围为 $\pm 125/\pm 250/\pm 500/\pm 1000/\pm 2000/\pm 4000$ dps。

经过配置，ASM330LHH 可利用硬件识别出的自由落体事件、6D 方向、活动或不活动、唤醒事件，来生成中断信号。

ASM330LHH 集成有一个 3Kbyte 的智能先进先出（FIFO）缓冲器，支持对陀螺仪数据、加速度计数据、温度数据和时间戳进行动态批处理。

ASM330LHH 采用小型塑料焊盘网格阵列封装（LGA-14L），可确保在更大的温度范围（-40 °C 至+105 °C）内正常工作。

ASM330LHH 在所有设计环节进行了优化，实现了有利于传感器辅助应用（例如航位推算和传感器融合）的更高输出稳定性、极低噪声和完全的数据同步。

1 引脚说明

图 1. 引脚连接

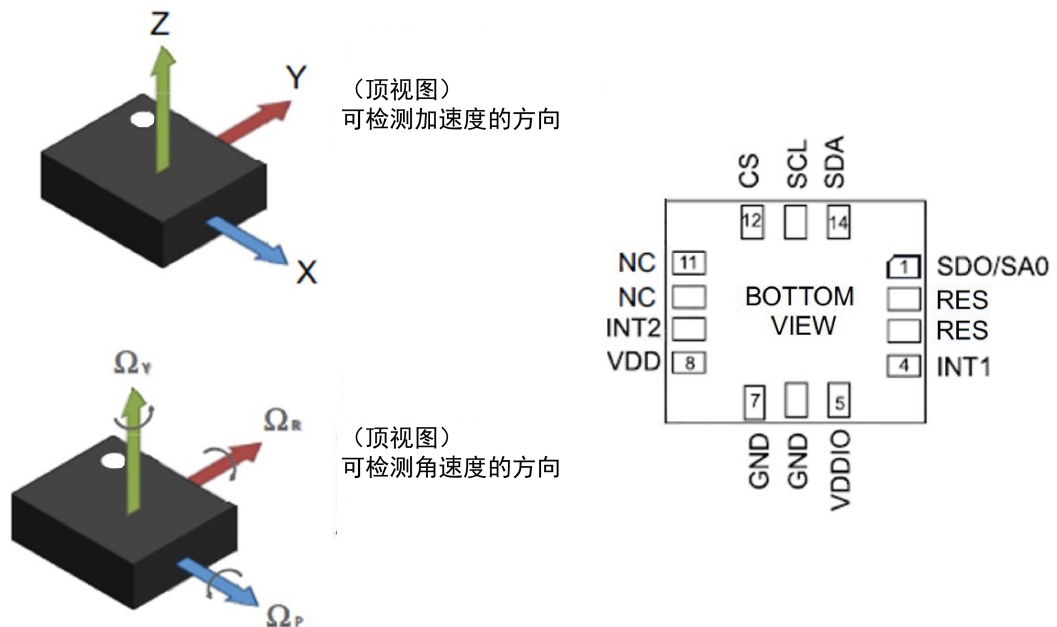


表 1. 内部引脚状态

引脚#	名称	功能	引脚状态
1	SDO	SPI 4 线接口串行数据输出 (serial data output, SDO)	默认值: 无上拉的输入 如果寄存器 02h 中的位 SDO_PU_EN = 1, 则上拉使能。
	SA0	I ² C 设备地址的最低有效位 (SA0)	
2	RES	连接到 VDDIO 或 GND	默认值: 无上拉的输入
3	RES	连接到 VDDIO 或 GND	默认值: 无上拉的输入
4	INT1	可编程中断 1	默认值: 带下拉的输入
5	VDDIO	I/O 引脚的供电	
6	GND	0 V 电源	
7	GND	0 V 电源	
8	VDD	电源	
9	INT2	可编程中断 2 (INT2) / 数据使能 (data enable, DEN)	默认值: 输出强制接地
10	NC	保持断开	默认值: 带上拉的输入
11	NC	保持断开	默认值: 带上拉的输入
12	CS	I ² C/SPI 模式选择 (1: SPI 空闲模式/ I ² C 通信使能; 0: SPI 通信模式/ I ² C 禁用)	默认值: 带上拉的输入 如果寄存器 13h 中的位 I2C_disable = 1 且寄存器 18h 中的位 DEVICE_CONF = 1, 则上拉禁用。
13	SCL	I ² C 串行时钟 (SCL) / SPI 串行时钟 (SPC)	默认值: 无上拉的输入
14	SDA	I ² C 串行数据 (SDA) / SPI 串行数据输入 (SDI) /	默认值: 无上拉的输入



引脚#	名称	功能	引脚状态
		3 线接口串行数据输出 (serial data output, SDO)	

内部上拉值范围从 30 kΩ 至 50 kΩ，取决于 VDDIO。

提示

- 正确初始化器件的步骤如下：
- 1. INT1 引脚：保持不连接或在上电期间连接外部下拉电阻。必须避免连接上拉电阻到此引脚。
 - 2. INT2 引脚：建议不要连接外部上拉电阻。
 - 3. 正确配置器件：
 - A) 对于 SPI：CTRL4_C (13h) 的 I2C_disable = 1 且 CTRL9_XL (18h) 的 DEVICE_CONF = 1。
 - A) 对于 I²C：CTRL4_C (13h) 的 I2C_disable = 0 (默认值) 且 CTRL9_XL (18h) 的 DEVICE_CONF = 1。

表 2. 寄存器

寄存器名	地址	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
PIN_CTRL	02h	0	SDO_PU_EN	1	1	1	1	1	1
FIFO_CTRL1	07h	WTM7	WTM6	WTM5	WTM4	WTM3	WTM2	WTM1	WTM0
FIFO_CTRL2	08h	STOP_ON_WTM	0	0	ODRCHG_EN	0	0	0	WTM8
FIFO_CTRL3	09h	BDR_GY_3	BDR_GY_2	BDR_GY_1	BDR_GY_0	BDR_XL_3	BDR_XL_2	BDR_XL_1	BDR_XL_0
FIFO_CTRL4	0Ah	DEC_TS_BATCH_1	DEC_TS_BATCH_0	ODR_T_BATCH_1	ODR_T_BATCH_0	0	FIFO_MODE2	FIFO_MODE1	FIFO_MODE0
COUNTER_BDR_REG1	0Bh	dataready_pulsed	RST_COUNTER_BDR	TRIG_COUNTER_BDR	0	0	0	CNT_BDR_TH_9	CNT_BDR_TH_8
COUNTER_BDR_REG2	0Ch	CNT_BDR_TH_7	CNT_BDR_TH_6	CNT_BDR_TH_5	CNT_BDR_TH_4	CNT_BDR_TH_3	CNT_BDR_TH_2	CNT_BDR_TH_1	CNT_BDR_TH_0
INT1_CTRL	0Dh	DEN_DRDY_flag	INT1_CNT_BDR	INT1_FIFO_FULL	INT1_FIFO_OVR	INT1_FIFO_TH	INT1_BOOT	INT1_DRDY_G	INT1_DRDY_XL
INT2_CTRL	0Eh	0	INT2_CNT_BDR	INT2_FIFO_FULL	INT2_FIFO_OVR	INT2_FIFO_TH	INT2_DRDY_TEMP	INT2_DRDY_G	INT2_DRDY_XL
WHO_AM_I	0Fh	0	1	1	0	1	0	1	1
CTRL1_XL	10h	ODR_XL3	ODR_XL2	ODR_XL1	ODR_XL0	FS1_XL	FS0_XL	LPF2_XL_EN	0
CTRL2_G	11h	ODR_G3	ODR_G2	ODR_G1	ODR_G0	FS1_G	FS0_G	FS_125	FS_4000
CTRL3_C	12h	BOOT	BDU	H_LACTIVE	PP_OD	SIM	IF_INC	0	SW_RESET
CTRL4_C	13h	0	SLEEP_G	INT2_on_INT1	0	DRDY_MASK	I2C_disable	LPF1_SEL_G	0
CTRL5_C	14h	0	ROUNDING1	ROUNDING0	0	ST1_G	ST0_G	ST1_XL	ST0_XL
CTRL6_C	15h	TRIG_EN	LVL1_EN	LVL2_EN	0	USR_OFF_W	FTYPE_2	FTYPE_1	FTYPE_0
CTRL7_G	16h	0	HP_G_EN	HPM1_G	HPM0_G	0	0	USR_OFF_ON_OUT	0
CTRL8_XL	17h	HPCF_XL2	HPCF_XL1	HPCF_XL0	HP_REF_MODE_XL	FASTSETTL_MODE_XL	HP_SLOPE_XL_EN	0	LOW_PASS_ON_6D
CTRL9_XL	18h	DEN_X	DEN_Y	DEN_Z	DEN_XL_G	DEN_XL_EN	DEN_LH	DEVICE_CONF	0
CTRL10_C	19h	0	0	TIMESTAMP_EN	0	0	0	0	0
ALL_INT_SRC	1Ah	TIMESTAMP_ENDCOUNT	0	SLEEP_CHANGE_IA	D6D_IA	0	0	WU_IA	FF_IA
WAKE_UP_SRC	1Bh	0	SLEEP_CHANGE_IA	FF_IA	SLEEP_STATE	WU_IA	X_WU	Y_WU	Z_WU
D6D_SRC	1Dh	DEN_DRDY	D6D_IA	ZH	ZL	YH	YL	XH	XL
STATUS_REG	1Eh	0	0	0	0	0	TDA	GDA	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





寄存器名	地址	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
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_A	28h	D7	D6	D5	D4	D3	D2	D1	D0
OUTX_H_A	29h	D15	D14	D13	D12	D11	D10	D9	D8
OUTY_L_A	2Ah	D7	D6	D5	D4	D3	D2	D1	D0
OUTY_H_A	2Bh	D15	D14	D13	D12	D11	D10	D9	D8
OUTZ_L_A	2Ch	D7	D6	D5	D4	D3	D2	D1	D0
OUTZ_H_A	2Dh	D15	D14	D13	D12	D11	D10	D9	D8
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	FIFO_WTM_IA	FIFO_OVR_IA	FIFO_FULL_IA	COUNTER_BDR_IA	FIFO_OVR_LATCHED	0	DIFF_FIFO_9	DIFF_FIFO_8
TIMESTAMP0	40h	D7	D6	D5	D4	D3	D2	D1	D0
TIMESTAMP1	41h	D15	D14	D13	D12	D11	D10	D9	D8
TIMESTAMP2	42h	D23	D22	D21	D20	D19	D18	D17	D16
TIMESTAMP3	43h	D31	D30	D29	D28	D27	D26	D25	D24
INT_CFG0	56h	0	INT_CLR_ON_READ	SLEEP_STATUS_ON_INT	SLOPE_FDS	0	0	0	LIR
INT_CFG1	58h	INTERRUPTS_ENABLE	INACT_EN1	INACT_EN0	0	0	0	0	0
THS_6D	59h	D4D_EN	SIXD_THS1	SIXD_THS0	0	0	0	0	0
WAKE_UP_THS	5Bh	0	USR_OFF_ON_WU	WK_THS5	WK_THS4	WK_THS3	WK_THS2	WK_THS1	WK_THS0
WAKE_UP_DUR	5Ch	FF_DUR5	WAKE_DUR1	WAKE_DUR0	WAKE_THS_W	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_SLEEP_CHANGE	0	INT1_WU	INT1_FF	0	INT1_6D	0	0
MD2_CFG	5Fh	INT2_SLEEP_CHANGE	0	INT2_WU	INT2_FF	0	INT2_6D	0	INT2_TIMESTAMP
INTERNAL_FREQ_FINE	63h	FREQ_FINE7	FREQ_FINE6	FREQ_FINE5	FREQ_FINE4	FREQ_FINE3	FREQ_FINE2	FREQ_FINE1	FREQ_FINE0
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
FIFO_DATA_OUT_TAG	78h	TAG_SENSOR_4	TAG_SENSOR_3	TAG_SENSOR_2	TAG_SENSOR_1	TAG_SENSOR_0	TAG_CNT_1	TAG_CNT_0	TAG_PARITY
FIFO_DATA_OUT_X_L	79h	D7	D6	D5	D4	D3	D2	D1	D0
FIFO_DATA_OUT_X_H	7Ah	D15	D14	D13	D12	D11	D10	D9	D8
FIFO_DATA_OUT_Y_L	7Bh	D7	D6	D5	D4	D3	D2	D1	D0



寄存器名	地址	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
FIFO_DATA_OUT_Y_H	7Ch	D15	D14	D13	D12	D11	D10	D9	D8
FIFO_DATA_OUT_Z_L	7Dh	D7	D6	D5	D4	D3	D2	D1	D0
FIFO_DATA_OUT_Z_H	7Eh	D15	D14	D13	D12	D11	D10	D9	D8

3 工作模式

ASM330LHH 提供了 3 种可能的操作配置：

- 只有加速度计工作，陀螺仪处于省电或睡眠模式；
- 只有陀螺仪工作，加速度计省电；
- 加速度计和陀螺仪均工作，且具有独立的 ODR。

该器件提供了较宽的 VDD 电压范围（从 2.0 V 至 3.6 V）和 1.62 V 至 3.6 V 的 VDDIO 范围。为避免潜在冲突，在上电时序期间，建议在主机侧将连接至器件 IO 引脚的管脚设置成高阻态。此外，为保证器件能正确关断，建议将 VDD 线接地的持续时间保持至少 100µs。

施加电源后，器件执行一段 10 ms（最长）的启动程序来加载修整参数。启动完成后，加速度计和陀螺仪均自动配置为省电模式。

加速度计和陀螺仪可分别配置为省电模式或高性能模式。它们可以具有不同的数据率而不受任何限制。陀螺仪传感器还能够设置为睡眠模式，以降低其功耗。

当加速度计和陀螺仪均工作时，加速度计与陀螺仪同步，两个传感器的数据率是彼此的整数倍。

参考数据手册，可以利用 CTRL1_XL 寄存器的输出数据率（ODR_XL）位来选择加速度计的输出数据率（表 3. 加速度计 ODR 选择）。

表 3. 加速度计 ODR 选择

ODR_XL [3:0]	ODR [Hz]
0000	省电
0001	12.5 Hz（高性能）
0010	26 Hz（高性能）
0011	52 Hz（高性能）
0100	104 Hz（高性能）
0101	208 Hz（高性能）
0110	417 Hz（高性能）
0111	833 Hz（高性能）
1000	1.66 kHz（高性能）
1001	3.33 kHz（高性能）
1010	6.66 kHz（高性能）

利用 CTRL2_G 寄存器的输出数据率（ODR_G）位来选择陀螺仪传感器的输出数据率（表 4. 陀螺仪 ODR 选择）。

表 4. 陀螺仪 ODR 选择

ODR_G [3:0]	ODR [Hz]
0000	省电
0001	12.5 Hz (高性能)
0010	26 Hz (高性能)
0011	52 Hz (高性能)
0100	104 Hz (高性能)
0101	208 Hz (高性能)
0110	417 Hz (高性能)
0111	833 Hz (高性能)
1000	1.66 kHz (高性能)
1001	3.33 kHz (高性能)
1010	6.66 kHz (高性能)

表 5. Vdd = 3.0 V 时的功耗 (典型值) 显示了不同工作模式下功耗典型值。

表 5. Vdd = 3.0 V 时的功耗 (典型值)

ODR [Hz]	只有加速度计	只有陀螺仪	组合 [Acc + Gyro]
省电	-	-	3 μ A
睡眠	-	430 μ A	-
12.5 Hz (高性能)	360 μ A	970 μ A	1.3 mA
26 Hz (高性能)	360 μ A	970 μ A	1.3 mA
52 Hz (高性能)	360 μ A	970 μ A	1.3 mA
104 Hz (高性能)	360 μ A	970 μ A	1.3 mA
208 Hz (高性能)	360 μ A	970 μ A	1.3 mA
416 Hz (高性能)	360 μ A	970 μ A	1.3 mA
833 Hz (高性能)	360 μ A	970 μ A	1.3 mA
1.66 kHz (高性能)	361 μ A	970 μ A	1.3 mA
3.33 kHz (高性能)	362 μ A	970 μ A	1.3 mA
6.66 kHz (高性能)	363 μ A	970 μ A	1.3 mA

3.1 省电模式

当加速度计/陀螺仪处于省电模式时，该器件几乎所有的内部模块都会被关掉，以减小功耗。数字接口 (I²C/SPI) 仍然在工作，以便能够与器件进行通信。保留配置寄存器的内容而不更新输出数据寄存器，可保存进入省电模式前存储器中采样的最后数据。

3.2 高性能模式

高性能模式下，所有的加速器/陀螺仪电路始终开启，并通过 ODR_XL/ODR_G 位来选择生成数据的数据率。数据中断产生是激活的。



3.3 陀螺仪睡眠模式

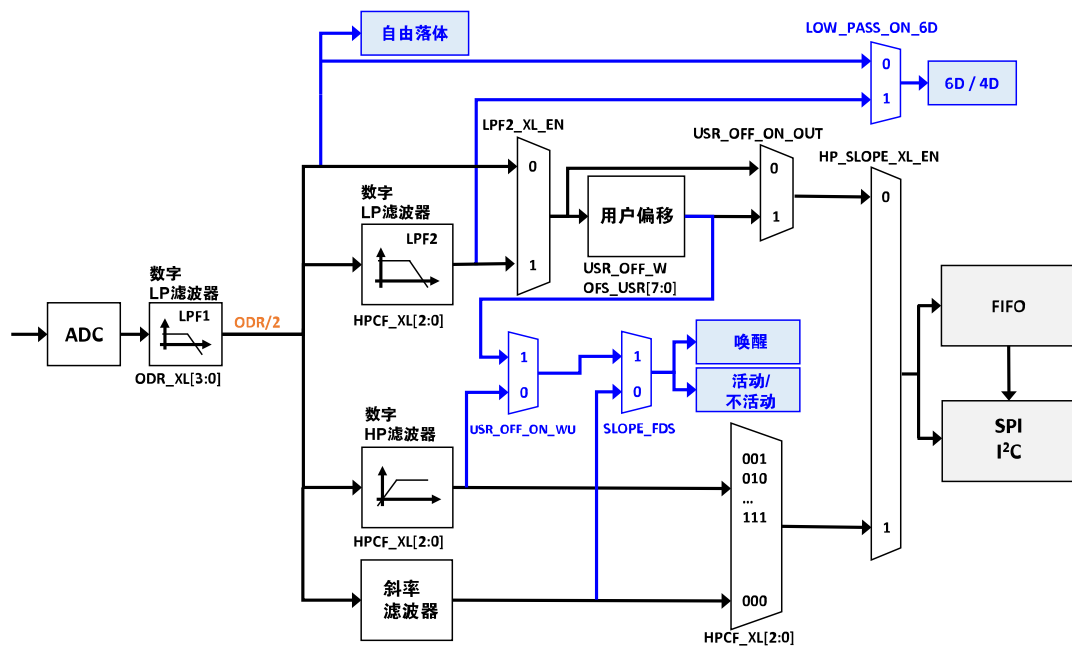
当陀螺仪处于睡眠模式时，陀螺仪模块的振荡驱动电路保持激活。相比于陀螺仪省电模式，从睡眠模式到高性能模式的打开时间大幅度减少。

如果陀螺仪未被配置为省电模式，则当 CTRL4_C 寄存器的睡眠模式使能 (SLEEP_G) 位被置为 1 时，不论所选陀螺仪 ODR 为何，它都将进入睡眠模式。

3.4 加速度计带宽

如下图所示，加速度计采样链路由 3 个级联主模块表示：一个 ADC 转换器，一个数字低通滤波器 (LPF1) 和复合数字滤波器组。

图 2. 加速度计滤波链



使用截止频率等于 $ODR/2$ 的数字低通滤波器 (LPF1) 对 ADC 转换的信号进行滤波。最后，由一个低通数字滤波器 (LPF2)、一个高通数字滤波器和一个斜率滤波器组成的组合滤波器组处理数字信号。

CTRL1_XL 寄存器的 LPF2_XL_EN 位和 CTRL8_XL 寄存器可用来配置复合滤波器组和加速度计滤波链的总带宽，如表 6. 加速度计带宽选择中所示。参考该表，在低通路径侧，如果 LPF2_XL_EN = 0，则带宽是指 LPF1 带宽；如果 LPF2_XL_EN = 1，则带宽指 LPF2 带宽。在高通路径侧，如果 HPCF_XL[2:0] = 000b，则带宽列表示斜率滤波器带宽；对于所有其他配置，则表示 HP 滤波器带宽。

表 6. 加速度计带宽选择 对于加速度计滤波链的各种配置，针对要丢弃的样本还提供了最大（最坏情况）的设置时间。

表 6. 加速度计带宽选择

HP_SLOPE_XL_EN	LPF2_XL_EN	HPCF_XL[2:0]	带宽	最大总设置时间 ⁽¹⁾ (要丢弃的采样)
0 (低通路径)	1	-	ODR / 2	See 表 8
		000	ODR / 4	See 表 8
		001	ODR / 10	10
		010	ODR / 20	19
		011	ODR / 45	38
		100	ODR / 100	75
		101	ODR / 200	150
		110	ODR / 400	296
		111	ODR / 800	595
1 (高通路径)	-	000	ODR / 4 (斜率滤波器)	See 表 8
		001	ODR / 10	14
		010	ODR / 20	19
		011	ODR / 45	38
		100	ODR / 100	75
		101	ODR / 200	150
		110	ODR / 400	296
		111	ODR / 800	595

1. 最终值的 99% 时的设置时间，考虑所有输出数据率和所有操作模式切换

通过将 HP_SLOPE_XL_EN 位置为 0，可选择复合滤波器模块的低通路径。如果 LPF2_XL_EN 位置为 0，则无需应用额外的滤波器；如果 LPF2_XL_EN 位置为 1，那么除 LPF1 外还需应用 LPF2 滤波器，并可通过配置 CTRL8_XL 寄存器的 HPCF_XL [2:0] 字段来设置加速度计链的总带宽。

通过将 CTRL8_XL 寄存器的 LOW_PASS_ON_6D 位置为 1，LPF2 低通滤波器还可用于 6D/4D 功能。

通过将 HP_SLOPE_XL_EN 位置为 1，可以选择复合滤波器模块的高通路径：HPCF_XL [2:0] 字段除了能用于使能 LPF1 滤波器之外，还可以使能斜率滤波器 (HPCF_XL[2:0] = 000b 时) 或数字高通滤波器 (其他 HPCF_XL [2:0] 配置)。HPCF_XL [2:0] 字段也可用来选择 HP 滤波器的截止频率。

高通滤波器参考模式功能可用于加速度计传感器：启用此功能后，当前的 X、Y、Z 加速度计采样将在内部存储，并从所有后续输出值中减去。为了使能参考模式，必须将 CTRL8_XL 寄存器的 HP_REF_MODE_XL 位和 HP_SLOPE_XL_EN 位置为 1，并且 HPCF_XL [2:0] 字段的值必须不等于 000b。当启用 Reference 模式功能时，LPF2 滤波器和 HP 滤波器都不可用。启用 Reference 模式之后的第一个加速度计输出数据必须被丢弃。

CTRL8_XL 寄存器的 FASTSETTL_MODE_XL 位使能加速度计 LPF2 或 HPF 快速建立模式：选择的滤波器在写入此位后设置第二个采样。仅在器件退出省电模式时应用此功能。



3.4.1 加速度计斜率滤波器

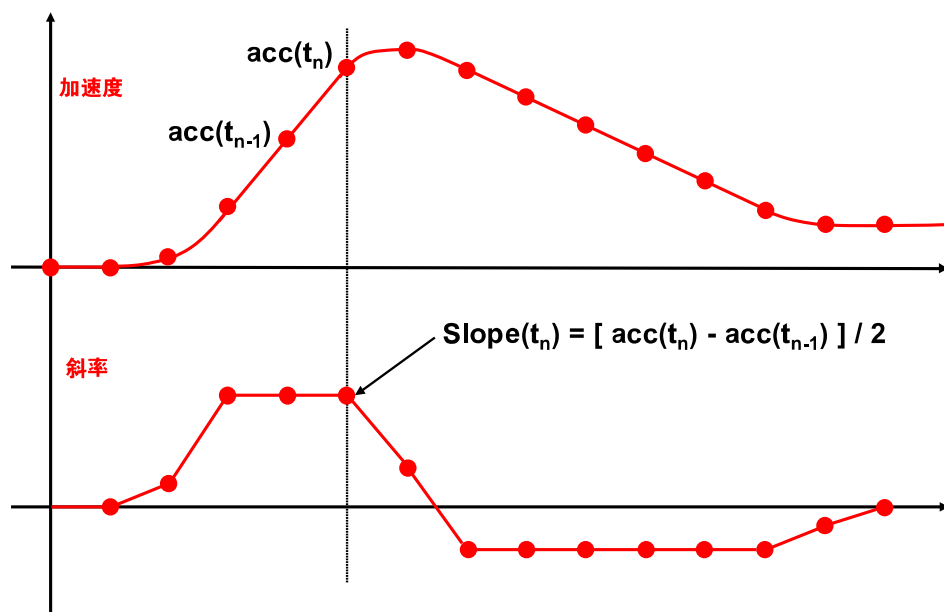
如图 2. 加速度计滤波链中所示，器件嵌入了一个数字斜率滤波器，该滤波器还可用于唤醒检测和活动/不活动嵌入功能。

该斜率滤波器输出数据利用以下公式进行计算：

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

下图举例说明了斜率数据信号的示例。

图 3. 加速度计斜率滤波器



3.5 加速度计开启/关断时间

加速度计读取链路包含了低通滤波，能够提高信噪比性能并降低混叠效应。因此，切换加速度计的功耗模式时，或改变加速度计的 ODR 时，必须考虑滤波器的设置时间。

加速度计链设置时间取决于为以下配置选择的输出数据率：

- LPF2 和 HP 滤波器禁用；
- LPF2 或 HP 滤波器使能且带宽选择了 ODR/4。

对于这两种可能的配置，下面的表 7. 加速度计开启/关闭时间（LPF2 和 HP 禁用）和 显示了切换加速度计功耗模式或加速度计 ODR 所需的最大总开启/关闭时间。表 8. 要丢弃的加速度计样本

表 7. 加速度计开启/关闭时间（LPF2 和 HP 禁用）

起始模式	目标模式	最大开启/关闭时间 ⁽¹⁾
省电	高性能	See 表 8
高性能	高性能 @ ODR < 6.66 kHz	丢弃 3 个样本
高性能	高性能 @ ODR = 6.66 kHz	丢弃 3 个样本
高性能	省电	1 μ s

1. @ 最终值的 99% 时的设置时间

表 8. 要丢弃的加速度计样本

目标模式 加速度计 ODR [Hz]	要丢弃的采样数 (LPF2 和 HP 滤波器禁用)	要丢弃的采样数 (LPF2 或 HP 滤波器使能且带宽为 ODR/4)
12.5 (高性能)	2	3
26 (高性能)	2	3
52 (高性能)	2	3
104 (高性能)	2	3
208 (高性能)	2	3
417 (高性能)	2	3
833 (高性能)	2	3
1667 (高性能)	3	3
3333 (高性能)	5	5
6667 (高性能)	11	11

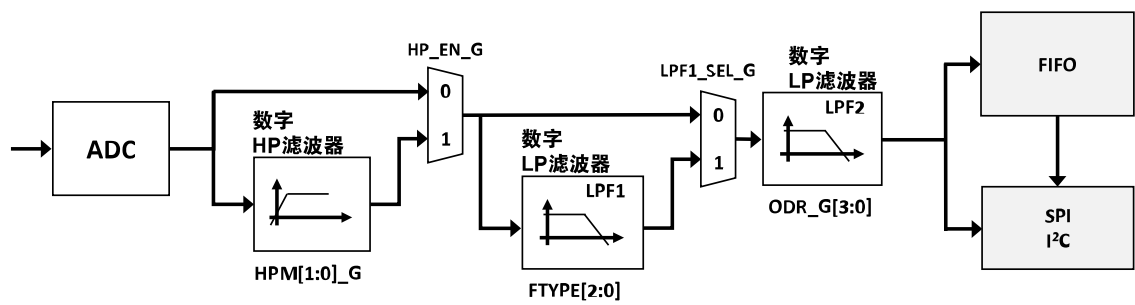
表 6. 加速度计带宽选择中已经列出了 LPF2 或 HP 数字滤波器使能且带宽不是 ODR/4 时的总设置时间。



3.6 陀螺仪带宽

陀螺仪滤波链配置如图 4. 陀螺仪数字链中所示。它是三个滤波器的级联：可选数字高通滤波器（HPF）、可选数字低通滤波器（LPF1）和数字低通滤波器（LPF2）。

图 4. 陀螺仪数字链



数字 HP 滤波器可通过将 CTRL7_G 寄存器的 HP_EN_G 位置为 1 来使能，其截止频率可通过 CTRL7_G 寄存器的字段 HPM_G[1:0]来选择，如下表所示。

表 9. 陀螺仪数字 HP 滤波器截止频率选择

HPM_G[1:0]	高通滤波器截止频率 [Hz]	总设置时间最大值 [s] ⁽¹⁾
00	0.016	45
01	0.065	11
10	0.260	3
11	1.040	0.7

1. 最终值的 99%时的设置时间

数字 LPF1 滤波器可以通过将 CTRL4_C 寄存器的 LPF1_SEL_G 位置为 1 来使能，其带宽可以通过 CTRL6_C 寄存器的字段 FTYPE_[2:0]来选择。

数字 LPF2 滤波器不能由用户配置，其截止频率取决于所选的陀螺仪 ODR。当陀螺仪 ODR 等于 6.66kHz 时，LPF2 滤波器被烧开。

下表汇总了在 CTRL4_C 寄存器的 LPF1_SEL_G 位和 CTRL6_C 寄存器的 FTYPE_[2:0]字段的配置下，不同陀螺仪 ODR 值的总陀螺仪带宽。

表 10. 陀螺仪总带宽选择

陀螺仪 ODR [Hz]	LPF1_SEL_G	FTYPE[2:0]	带宽[Hz] (@ 20 Hz 时的相位延迟)
12.5	0	-	4.3 (-35° @ 1.3 Hz)
	1	0xx	4.3 (-35° @ 1.3 Hz)
	1	100	4.3 (-35° @ 1.3 Hz)
	1	101	4.3 (-35° @ 1.3 Hz)
	1	110	4.3 (-35° @ 1.3 Hz)
	1	111	4.3 (-35° @ 1.3 Hz)
26	0	-	8.3 (-35° @ 2.5 Hz)
	1	0xx	8.3 (-35° @ 2.5 Hz)
	1	100	8.3 (-35° @ 2.5 Hz)
	1	101	8.3 (-35° @ 2.5 Hz)
	1	110	8.3 (-35° @ 2.5 Hz)
	1	111	8.3 (-35° @ 2.5 Hz)
52	0	-	16.7 (-35° @ 5 Hz)
	1	0xx	16.7 (-36° @ 5 Hz)
	1	100	16.7 (-39° @ 5 Hz)
	1	101	16.9 (-43° @ 5 Hz)
	1	110	13.4 (-44° @ 5 Hz)
	1	111	9.8 (-49° @ 5 Hz)
104	0	-	33 (-35° @ 10 Hz)
	1	0xx	33 (-38° @ 10 Hz)
	1	100	34 (-43° @ 10 Hz)
	1	101	31 (-51° @ 10 Hz)
	1	110	19 (-54° @ 10 Hz)
	1	111	11.6 (-64° @ 10 Hz)
208	0	-	67 (-35°)



陀螺仪 ODR [Hz]	LPF1_SEL_G	FTYPE[2:0]	带宽[Hz] (@ 20 Hz 时的相位延迟)
208	1	0xx	67 (-41°)
	1	100	62 (-51°)
	1	101	43 (-68°)
	1	110	23 (-74°)
	1	111	12.2 (-93°)
417	0	-	133 (-18°)
	1	000	133 (-23°)
	1	001	128 (-25°)
	1	010	112 (-28°)
	1	011	134 (-21°)
	1	100	86 (-34°)
	1	101	48 (-51°)
	1	110	24.6 (-57°)
	1	111	12.4 (-76°)
833	0	-	267 (-9°)
	1	000	222 (-14°)
	1	001	186 (-16°)
	1	010	140 (-20°)
	1	011	260 (-12°)
	1	100	96 (-25°)
	1	101	49 (-43°)
	1	110	25 (-48°)
	1	111	12.6 (-68°)
1667	0	-	539 (-5°)
	1	000	274 (-10°)
	1	001	212 (-12°)
	1	010	150 (-15°)
	1	011	390 (-8°)
	1	100	99 (-21°)
	1	101	50 (-38°)
	1	110	25 (-44°)
	1	111	12.6 (-63°)
3333	0	-	1137 (-3°)
	1	000	292 (-8°)
	1	001	220 (-10°)
	1	010	153 (-13°)
	1	011	451 (-6°)
	1	1xx	不可用
6667	0	-	>3333 (-2°)
	1	000	297 (-7°)
	1	001	223 (-9°)
	1	010	154 (-12°)
	1	011	470 (-5°)



陀螺仪 ODR [Hz]	LPF1_SEL_G	FTYPE[2:0]	带宽[Hz]（@ 20 Hz 时的相位延迟）
6667	1	1xx	不可用



3.7 陀螺仪开启/关闭时间

切换其模式或改变陀螺仪 ODR 时，还必须考虑陀螺仪传感器的开启/关断时间。

用于切换陀螺仪功耗模式或陀螺仪 ODR 的最大总开启/关闭时间（HP 滤波器禁用）显示于表 11. 陀螺仪开启/关闭时间（HP 禁用）。

表 11. 陀螺仪开启/关闭时间（HP 禁用）

起始模式	目标模式	最大开启/关闭时间 ⁽¹⁾
省电	睡眠	70 ms
省电	高性能	70 ms + 见表 12 或 表 13
睡眠	高性能	见表 12 或 表 13
高性能	高性能（ODR 改变）	丢弃 2 个样本
高性能	省电	1 μ s, 如果 XL 和 Gyro 在省电模式中 300 μ s, 如果 XL 不在省电模式中

1. @ 最终值的 99% 时的设置时间

表 12. 要丢弃的陀螺仪样本（LPF1 禁用）

陀螺仪 ODR [Hz]	要丢弃的样本数 ⁽¹⁾
12.5 Hz	2
26 Hz	3
52 Hz	3
104 Hz	3
208 Hz	3
417 Hz	3
833 Hz	3
1.66 kHz	4
3.33 kHz	5
6.66 kHz	6

1. 最终值的 99% 时的建立时间

表 13. 陀螺仪链设置时间（LPF1 使能）

FTYPE[2:0]	每种 ODR 的最长设置时间[ms] ⁽¹⁾
000	3.5
001	4.8
010	6.9
011	2.1
100	11
101	22
110	30
111	60

1. 最终值的 99% 时的设置时间

当模式变为高性能模式且 HP 滤波器使能或 HP 滤波器开启时，必须将 HP 滤波器设置时间添加到表 11. 陀螺仪开启/关闭时间（HP 禁用）。HP 滤波器设置时间如表 9. 陀螺仪数字 HP 滤波器截止频率选择所示，它与 ODR 无关。



4 读取输出数据

4.1 启动序列

当器件上电时，器件会自动从嵌入的内存中加载校准系数到内部寄存器中。当启动程序完成时，即，约 10 ms 后，加速度计和陀螺仪自动进入省电模式。

要启用加速度计并通过主 I²C/SPI 接口采集加速度数据，需要通过 CTRL1_XL 寄存器选择某一种工作模式。

以下通用线序可用来配置加速度计：

1. 写 INT1_CTRL = 01h // INT1 上，Acc 数据准备就绪中断
2. 写 CTRL1_XL = 60h // Acc = 417 Hz（高性能模式）

要启用陀螺仪并通过主 I²C / SPI 接口采集角速度数据，需要通过 CTRL2_G 来选择某一种工作模式。

以下通用序列可用来配置陀螺仪：

1. 写 INT1_CTRL = 02h // INT1 上，Gyro 数据准备就绪中断
2. 写 CTRL2_G = 60h // Gyro = 417 Hz（高性能模式）

4.2 使用状态寄存器

该器件具有一个 STATUS_REG 寄存器，应当对该寄存器进行轮询以检查一组新数据何时可用。当一组新数据在加速度计输出上可用时，XLDA 位被置为 1；当一组新数据在陀螺仪输出上可用时，GDA 位被置为 1。

对于加速度计（陀螺仪也是类似的），应当按照如下步骤对输出寄存器进行读取：

1. 读取 STATUS_REG
2. 如果 XLDA = 0，则进入 1
3. 读取 OUTX_L_A
4. 读取 OUTX_H_A
5. 读取 OUTY_L_A
6. 读取 OUTY_H_A
7. 读取 OUTZ_L_A
8. 读取 OUTZ_H_A
9. 数据处理
10. 跳到步骤 1



4.3 使用数据准备就绪信号

该器件可配置为具有一个硬件信号，以确定新的一组测量数据何时可以读取。

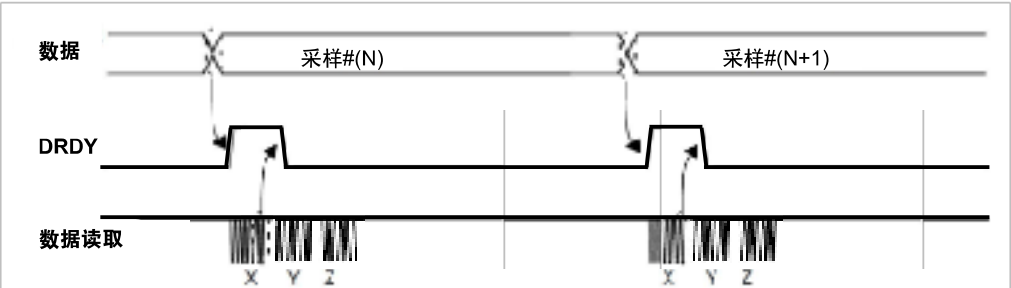
对于加速度计传感器，数据准备就绪信号由 **STATUS_REG** 寄存器的 **XLDA** 位表示。通过将 **INT1_CTRL** 寄存器的 **INT1_DRDY_XL** 位置为 1，可将该信号驱动至 **INT1** 引脚，通过将 **INT2_CTRL** 寄存器的 **INT2_DRDY_XL** 位置为 1，将其驱动至 **INT2** 引脚。

对于陀螺仪传感器，数据准备就绪信号由 **STATUS_REG** 寄存器的 **GDA** 位表示。通过将 **INT1_CTRL** 寄存器的 **INT1_DRDY_G** 位置为 1，可将该信号驱动至 **INT1** 引脚，通过将 **INT2_CTRL** 寄存器的 **INT2_DRDY_XL** 位置为 1，将其驱动至 **INT2** 引脚。

当一组新数据生成并可读取时，数据准备就绪信号升高为 1。数据就绪信号可以是锁存的或脉冲的：如果 **COUNTER_BDR_REG1** 寄存器的 **dataready_pulsed** 位被置为 0（默认值），则数据就绪信号被锁存，并且当其中某一个的较高部分（对于加速度计，为 29h、2Bh、2Dh；对于陀螺仪，为 23h、25h、27h）被读取时，中断复位。如果 **COUNTER_BDR_REG1** 寄存器的 **dataready_pulsed** 位置为 1，则数据就绪信号是脉冲的，并且在中断引脚上观察到的脉冲持续时间为 75 μ s。脉冲模式不适用于总是锁存的 **XLDA** 和 **GDA** 位。



图 5. 数据准备就绪信号



4.3.1 DRDY 屏蔽功能

如果将 CTRL4_C 寄存器的 DRDY_MASK 位置为 1，则加速度计和陀螺仪数据准备就绪信号会被屏蔽，直至完成传感器滤波器稳定稳定。

当 FIFO 处于激活状态且 DRDY_MASK 位置为 1 时，存储在 FIFO 中的加速计/陀螺仪无效采样等于 7FFFh、7FFEh 或 7FFDh。这样，存储在 FIFO 缓冲器中的无效采样被加上了一个标签，因此在数据后处理过程中，可以容易地识别出它们并将其丢弃。

注：DRDY_MASK 位仅作用于所有加速度计 ODR 的加速度计 LPF1 数字滤波器设置时间和陀螺仪 ODR ≤ 833 Hz 时的陀螺仪 LPF2 数字滤波器设置时间。

4.4 使用块数据更新 (block data update, BDU) 功能

如果读取加速度计/陀螺仪数据特别慢，并且不能（或者不需要）与 STATUS_REG 寄存器中的 XLDA/GDA 位或驱动到 INT1/INT2 引脚的 DRDY 信号同步，那么强烈建议将 CTRL3_C 寄存器中的 BDU（块数据更新）位置为 1。

此功能可以避免读取不同样本相关的值（输出数据的最高有效部分和最低有效部分）。特别地，当 BDU 被激活时，每个数据寄存器中始终包含器件产生的最新输出数据，但是，如果发起了对给定寄存器的读取（即 OUTX_H_A(G) 和 OUTX_L_A(G)，OUTY_H_A(G) 和 OUTY_L_A(G)，OUTZ_H_A(G) 和 OUTZ_L_A(G)），这对读数不能更新，直至数据的 MSB 和 LSB 部分均被读取。

请注意：BDU 只能确保 LSB 部分和 MSB 部分同一时刻被采样。例如，如果读取速度非常慢，则 X 和 Y 可在 T1 读取，Z 在 T2 采样。

BDU 功能还作用于 FIFO_STATUS1 和 FIFO_STATUS2 寄存器。当 BDU 位置为 1 时，必须首先读取 FIFO_STATUS1，然后读取 FIFO_STATUS2。

4.5 认识输出数据

测得的加速度数据被发送到 OUTX_H_A、OUTX_L_A、OUTY_H_A、OUTY_L_A、OUTZ_H_A 和 OUTZ_L_A 寄存器。这些寄存器分别包含作用于 X、Y 和 Z 轴的加速度信号的最高有效部分和最低有效部分。

测得的角速率数据被发送到 OUTX_H_G、OUTX_L_G、OUTY_H_G、OUTY_L_G、OUTZ_H_G 和 OUTZ_L_G 寄存器。这些寄存器分别容纳角速率信号在 X、Y 和 Z 轴上的最高有效部分和最低有效部分。

X、Y、Z 轴的完整输出数据由 OUTX_H_A(G) & OUTX_L_A(G)，OUTY_H_A(G) & OUTY_L_A(G)，OUTZ_H_A(G) & OUTZ_L_A(G) 合并提供，表示为 2 的补码。

加速度数据和角速率数据均表示为 16 bit 的数字。

4.5.1 输出数据示例

表 14. 输出数据寄存器内容 vs. 加速度 (FS_XL = ±2 g) 提供的几个基本示例中，会在器件受给定加速度影响的情况下读取数据寄存器中的数据。

表 15. 输出数据寄存器内容 vs. 角速度 (FS_G = ±250 dps) 提供了陀螺仪数据的一些基本示例，当器件施加了给定的角速率时，在数据寄存器中读取这些数据。

下表中所列值是在理想器件校准的假设下给出的（即，无偏移，无增益误差，...）。

表 14. 输出数据寄存器内容 vs. 加速度 (FS_XL = ±2 g)

加速度值	寄存器地址	
	OUTX_H_A (29h)	OUTX_L_A (28h)
0 g	00h	00h
350 mg	16h	69h
1 g	40h	09h
-350 mg	E9h	97h
-1 g	BFh	F7h

表 15. 输出数据寄存器内容 vs. 角速度 (FS_G = ± 250 dps)

角速率值	寄存器地址	
	OUTX_H_G (23h)	OUTX_L_G (22h)
0 dps	00h	00h
100 dps	2Ch	A4h
200 dps	59h	49h
-100 dps	D3h	5Ch
-200 dps	A6h	B7h

4.6 加速度计偏移寄存器

器件提供了加速度计偏移寄存器 (X_OFS_USR、Y_OFS_USR、Z_OFS_USR)，可用于零 g 偏移校正，或者用来将偏移量应用于加速度计输出数据。

可通过 CTRL7_G 寄存器 USR_OFF_ON_OUT 位的置位来使能加速度计偏移功能块。在偏移寄存器中设置的偏移值从测得的相应轴加速度值内部减去；内部处理的数据随后被发送到加速计输出寄存器和 FIFO（如果启用）。这些寄存器值以二进制补码的形式表示为 8 位数据，并且必须在 $[-127, 127]$ 的范围内。

应用于偏移寄存器值的权重 $[g/LSB]$ 独立于加速度计所选的量程，并可利用 CTRL6_C 寄存器的 USR_OFF_W 位进行配置：

- $2^{-10}g/LSB$ ，如果 USR_OFF_W 位置为 0；
- $2^{-6}g/LSB$ ，如果 USR_OFF_W 位置为 1。

4.7 环行功能

环行功能可用来自动寻址器件寄存器，以进行多字节读取。基本上，伴随多字节读取操作，所读取的寄存器地址会自动从第一个寄存器转到最后一个寄存器，然后返回第一个寄存器。

4.7.1 FIFO 输出寄存器环行

在执行 FIFO 输出寄存器的多字节读取操作时，环行功能自动使能：在读取 FIFO_DATA_OUT_Z_H (7Eh) 后，将要读取的下一个寄存器的地址将自动回到 FIFO_DATA_OUT_TAG (78h)，从而允许用户通过一次多字节读取读取许多数据。

4.7.2 传感器输出寄存器环行

可以对其他输出寄存器应用环形功能。

以下输出寄存器组也可以使能环行功能：

- 加速度计输出寄存器，从 OUTX_L_A (28h) 到 OUTZ_H_A (2Dh)；
- 陀螺仪输出寄存器，从 OUTX_L_G (22h) 到 OUTZ_H_G (27h)；
- 陀螺仪和加速度计输出寄存器，从 OUTX_L_G (22h) 到 OUTZ_H_A (2Dh)；

输出寄存器环行模式可利用 CTRL5_C 寄存器的 ROUNDING[1:0] 位进行配置，如下表所示。

表 16. 输出寄存器环行模式

ROUNDING[1:0]	环行模式
00	无环行
01	只有加速度计
10	只有陀螺仪
11	陀螺仪 + 加速度计



4.8 DEN（数据使能）

通过使能 CTRL6_C 寄存器中的 TRIG_EN、LVL1_EN 和 LVL2_EN 位，器件可允许外部触发电平识别。有四种不同模式可供选择（参见表 17. DEN 配置）：

- 边沿感应触发模式；
- 电平感应触发模式；
- 电平感应锁存模式；
- 电平感应 FIFO 使能模式。

数据使能（DEN）输入信号必须在 INT2 引脚上被驱动，当其中一种模式使能时，INT2 被配置为输入引脚。DEN 功能仅在陀螺仪数据上被默认激活。要将此功能扩展到加速度计数据，CTRL4_C 寄存器中的 DEN_XL_EN 位必须置为 1。

DEN 有效电平默认为低电平。通过将 CTRL5_C 寄存器中的 DEN_LH 位置为 1，可以将其更改为高电平有效。

表 17. DEN 配置

TRIG_EN	LVL1_EN	LVL2_EN	功能	触发类型 (Trigger Type)	动作
0	0	0	数据使能关闭	-	-
1	0	0	边沿感应触发模式	边沿	数据生成
0	1	0	电平感应触发模式	级别	数据加戳
0	1	1	电平感应锁存模式	边沿	数据加戳
1	1	0	电平感应 FIFO 使能模式	级别	FIFO 中的数据生成和加戳

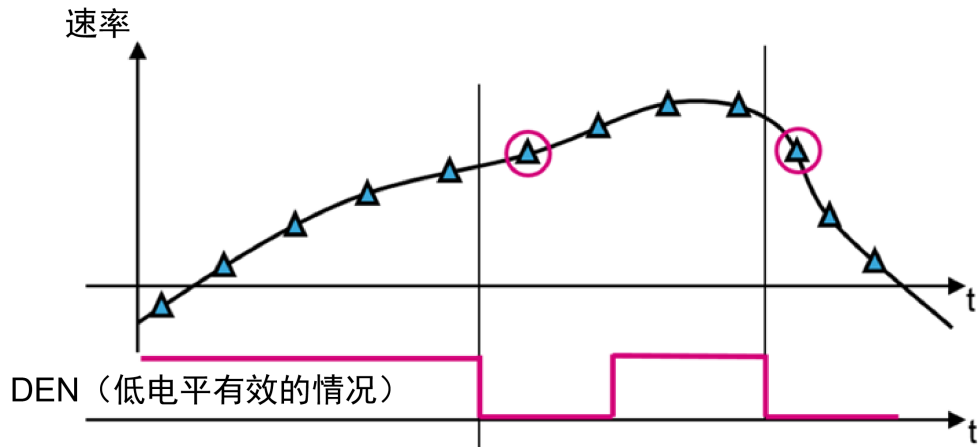
4.8.1 边沿感应触发模式

可以通过将 CTRL6_C 中的 TRIG_EN 位置为 1，并将 CTRL6_C 寄存器中的 LVL1_EN、LVL2_EN 位置为 0，来启用边沿感应触发模式。

一旦使能了边沿感应触发模式，FIFO 缓冲器和输出寄存器就会被填充上 DEN 输入信号每个上升沿（如果 DEN_LH 位等于 1）或下降沿（如果 DEN_LH 位等于 0）之后所获取的第一个采样。

图 6 用红色圆圈表示在下降沿之后（DEN 低电平有效）采集的采样。

图 6. 边沿感应触发模式，DEN 低电平有效



边沿感应触发模式启用后，仅对陀螺仪输出寄存器起作用。GDA 仅与降低采样数据相关，而加速度计输出寄存器和 XLDA 则根据 ODR_XL 进行更新。如果 DEN_XL_EN 位置为 1，则加速度计传感器也会被降低采样。这种情况下，陀螺仪和加速度计必须设置为 ODR 相同的组合模式。通过将陀螺仪设置为省电模式，可以使用加速度计独立模式。

请注意，在更新数据寄存器前会对 DEN 电平进行内部读取：如果在读取后发生电平变化，那么将在下一个 ODR 中确认 DEN。

对于 FIFO 中的边沿感应触发器，有三种可能的配置，描述如下：

- 只有陀螺仪处于触发模式，不能保存在 FIFO 中：这种情况下，FIFO 只与加速度计相关，并且正常工作。
- 只有陀螺仪处于触发模式并保存在 FIFO 中：在这种配置中，FIFO 存在以下限制：
 - 陀螺仪批处理数据率（FIFO_CTRL3 寄存器的 BDR_GY_[3:0] 位）和陀螺仪输出数据率（CTRL2_G 寄存器的 ODR_G[3:0] 位）必须置为相同值；
 - 禁用配置变化传感器（CFG-Change）（FIFO_CTRL2 寄存器的 ODRCHG_EN 位必须置为 0）；
 - 禁用 FIFO 中的时间戳抽取（FIFO_CTRL4 寄存器的 DEC_TS_BATCH_[1:0] 位必须置为 00b）。
- 陀螺仪和加速度计处于触发模式并保存在 FIFO 中：在这种配置中，FIFO 存在以下限制：
 - 陀螺仪批处理数据率（FIFO_CTRL3 寄存器的 BDR_GY_[3:0] 位）和陀螺仪输出数据率（CTRL2_G 寄存器的 ODR_G[3:0] 位）必须置为相同值；
 - 加速度计批处理数据率（FIFO_CTRL3 寄存器的 BDR_XL_[3:0] 位）和加速度计输出数据率（CTRL1_XL 寄存器的 ODR_XL[3:0] 位）必须置为相同值；
 - 陀螺仪和加速度计必须设置为相同输出数据率，否则必须将陀螺仪配置为省电模式；
 - 禁用配置变化传感器（CFG-Change）（FIFO_CTRL2 寄存器的 ODRCHG_EN 位必须置为 0）；
 - 禁用 FIFO 中的时间戳抽取（FIFO_CTRL4 寄存器的 DEC_TS_BATCH_[1:0] 位必须置为 00b）。

边沿感应触发模式可支持相机帧与陀螺仪采样同步，用于电子图像稳定（Electrical Image Stabilization, EIS）应用。来自相机模块的同步信号必须连接到 INT2 引脚。

在下面所示的例子中，FIFO 被配置为将陀螺仪数据和加速度计数据都存储在 FIFO 缓冲器中；当 DEN 信号切换时，数据在下降沿写入 FIFO 中。

- 将 44h 写入 FIFO_CTRL3 // 使能 FIFO 中的加速度计和陀螺仪 @ 104 Hz
- 将 06h 写入 FIFO_CTRL4 // 将 FIFO 设置为 Continue 模式



- | | |
|----------------------|---|
| 3. 将 80h 写入 CTRL6_C | // 使能边沿感应触发 |
| 4. 将 E8h 写入 CTRL9_XL | // INT2 引脚切换为输入模式 (DEN 信号) |
| | // 将 DEN 功能扩展到加速计传感器 |
| | // 选择 DEN 有效电平 (低电平有效) |
| 5. 将 40h 写入 CTRL1_XL | // 开启加速度计: ODR_XL = 104 Hz, FS_XL = $\pm 2g$ |
| 6. 将 4Ch 写入 CTRL2_G | // 开启陀螺仪: ODR_G = 104 Hz, FS_G = ± 2000 dps |

4.8.2

电平感应触发模式

可以通过将 CTRL6_C 寄存器中的 LVL1_EN 位置为 1，并将 CTRL6_C 寄存器中的 TRIG_EN、LVL2_EN 位置为 0，来使能电平感应触发模式。

一旦使能电平感应触发模式，那么如果 DEN 电平有效，则所选数据（在输出寄存器和 FIFO 中）的 LSB 位将被替换为 1；如果 DEN 电平未被激活，则所选数据的 LSB 位会被替换为 0。所选数据可以是加速度计或陀螺仪传感器的 X、Y、Z 轴（详情请参见第 4.8.5 节用于 DEN 冲压的 LSB 选择）。

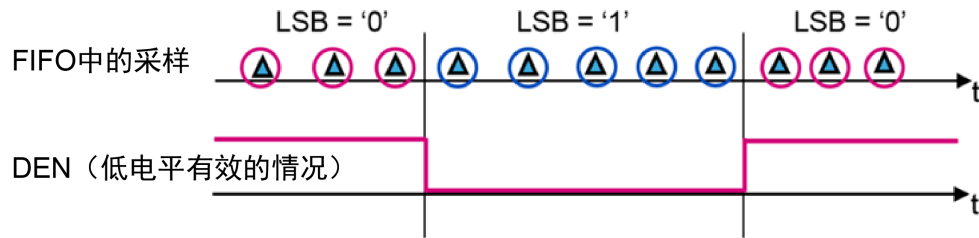
所有数据都可以根据 FIFO 设置存储在 FIFO 中。

请注意，在更新数据寄存器前会对 DEN 电平进行内部读取：如果在读取后发生电平变化，那么将在下一个 ODR 中确认 DEN。

如果通过 CTRL9_XL 寄存器 DEN_XL_EN 位的置位使能了加速度计传感器的 DEN 功能，则加速度计和陀螺仪传感器必须配置为相同 ODR，否则必须将陀螺仪设置为省电模式。

图 7. 电平感应触发模式，DEN 低电平有效 以红色圆圈显示 LSB = 0（DEN 未激活）时存储在 FIFO 中的样本，蓝色圆圈表示 LSB = 1（DEN 激活）时存储在 FIFO 中的样本。

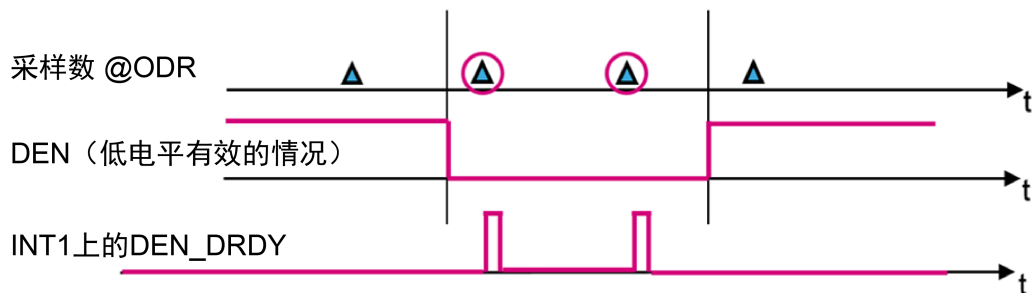
图 7. 电平感应触发模式，DEN 低电平有效



当电平感应触发模式使能时，DEN 信号也可用于过滤 INT1 引脚上的数据就绪信号。只有在 DEN 引脚处于激活状态时，INT1 才会显示数据就绪信息。为此，INT1_CTRL 寄存器的 DEN_DRDY_flag 位必须置为 1。中断信号可以根据 COUNTER_BDR_REG1 寄存器的 dataready_pulsed 位进行锁存或脉冲。

图 8. 电平感应触发模式，DEN 低电平有效，DEN_DRDY 在 INT1 上 显示了当 DEN 电平低（有效状态）时 INT1 上的数据就绪示例。

图 8. 电平感应触发模式，DEN 低电平有效，DEN_DRDY 在 INT1 上



4.8.3

电平感应锁存模式

可以通过将 CTRL6_C 寄存器中的 LVL1_EN 和 LVL2_EN 位置为 1，并将 CTRL6_C 寄存器中的 TRIG_EN 位置为 0，来使能电平感应锁存模式。

当使能电平感应锁存模式时，所选数据（在输出寄存器和 FIFO 中）的 LSB 位通常设置为 0，并且仅在 DEN 引脚上一个脉冲之后的第一个采样时变为 1。

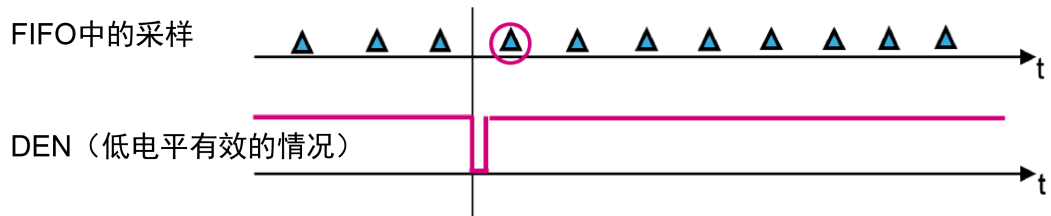
请注意，在更新数据寄存器前会对 DEN 电平进行内部读取：如果在读取后发生电平变化，那么将在下一个 ODR 中确认 DEN。

如果通过 CTRL9_XL 寄存器 DEN_XL_EN 位的置位使能了加速度计传感器的 DEN 功能，则加速度计和陀螺仪传感器必须配置为相同 ODR，否则必须将陀螺仪设置为省电模式。

数据可以通过 CTRL9_XL 中的 DEN_X、DEN_Y、DEN_Z 和 DEN_XL_G 位来选择（详情请参见第 4.8.5 节 用于 DEN 冲压的 LSB 选择）。

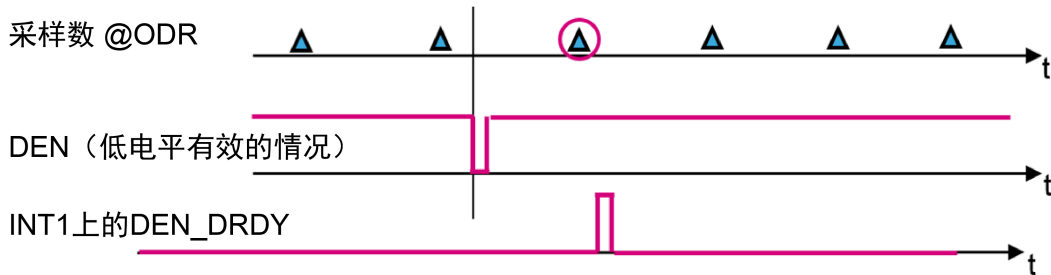
图 9 显示了 DEN 低电平有效时电平感应锁存模式的示例。在 DEN 引脚上的脉冲之后，带红色圆圈的采样在 LSB 位上的值为 1。所有其他样本的 LSB 位都为 0。

图 9. 电平感应锁存模式，DEN 低电平有效



当使能电平感应锁存模式，并且 INT1_CTRL 寄存器的 DEN_DRDY_flag 位设置为 1 时，INT1 引脚上会产生一个脉冲，对应于 DEN 脉冲出现后产生的第一个采样可用（见图 10. 电平感应锁存模式，DEN 低电平有效，DEN_DRDY 在 INT1 上）。

图 10. 电平感应锁存模式，DEN 低电平有效，DEN_DRDY 在 INT1 上



4.8.4 电平感应 FIFO 使能模式

可通过将 CTRL6_C 寄存器中的 TRIG_EN 和 LVL1_EN 位设置为 1，并将 CTRL6_C 寄存器中的 LVL2_EN 位设置为 0，来启用电平感应 FIFO 使能模式。

一旦启用了电平感应 FIFO 使能模式，只有当 DEN 引脚为有效状态时，才会将数据存储到 FIFO 中。

此模式下，所选数据的 LSB 位（在输出寄存器和 FIFO 中），对于奇数 DEN 事件，会替代为 0，对于偶数 DEN 事件，会替代为 1。该功能可将当前 DEN 激活窗口期间存储在 FIFO 中的数据与下一个 DEN 激活窗口期间存储在 FIFO 中的数据区分开来。

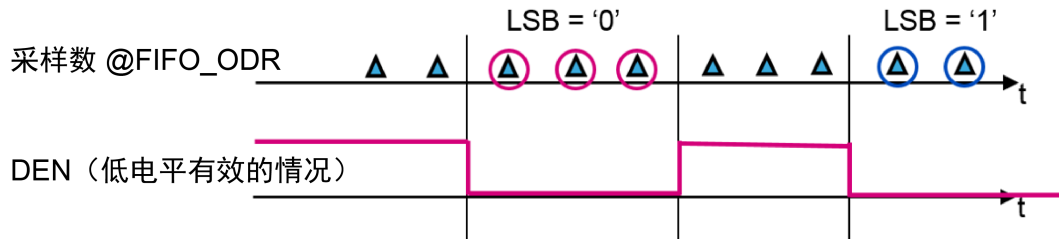
请注意，在更新数据寄存器前会对 DEN 电平进行内部读取：如果在读取后发生电平变化，那么将在下一个 ODR 中确认 DEN。

如果通过 CTRL9_XL 寄存器 DEN_XL_EN 位的置位使能了加速度计传感器的 DEN 功能，则加速度计和陀螺仪传感器必须配置为相同 ODR，否则必须将陀螺仪设置为省电模式。

所选数据可以是加速度计或陀螺仪传感器的 X、Y、Z 轴。数据可以通过 CTRL9_XL 寄存器中的 DEN_X、DEN_Y、DEN_Z 和 DEN_XL_G 位来选择（详情请参见第 4.8.5 节 用于 DEN 冲压的 LSB 选择）。

图 11 中显示了电平感应 FIFO 使能模式的示例，红圈表示存储在 FIFO 中、LSB 位为 0 的采样，而蓝圈表示 LSB 位为 1 的采样。

图 11. 电平感应 FIFO 使能模式，DEN 低电平有效



在使用电平感应 FIFO 使能模式时，在 FIFO 配置中必须考虑一些限制：

- 陀螺仪批处理数据率（FIFO_CTRL3 寄存器的 BDR_GY_[3:0] 位）和陀螺仪输出数据率（CTRL2_G 寄存器的 ODR_G[3:0] 位）必须置为相同值；
- 如果 CTRL9_XL 寄存器的 DEN_XL_EN 位置为 1，则加速度计批处理数据率（FIFO_CTRL3 寄存器的 BDR_XL_[3:0] 位）和加速度计输出数据率（CTRL1_XL 寄存器的 ODR_XL[3:0] 位）必须置为相同值；
- 禁用配置变化传感器（CFG-Change）（FIFO_CTRL2 寄存器的 ODRCHG_EN 位必须置为 0）；
- 禁用 FIFO 中的时间戳抽取（FIFO_CTRL4 寄存器的 DEC_TS_BATCH_[1:0] 位必须置为 00b）。

4.8.5 用于 DEN 冲压的 LSB 选择

当使用电平感应模式（触发或锁存）时，可以选择哪个 LSB 必须包含与 DEN 引脚特性有关的信息。根据 CTRL9_XL 寄存器的 DEN_X、DEN_Y、DEN_Z 和 DEN_XL_G 位，可以在加速度计或陀螺仪轴上标记此信息。将 DEN_X、DEN_Y、DEN_Z 位置为 1 时，DEN 信息会被标记在由 DEN_XL_G 位选择的传感器对应轴的 LSB 中。通过将 DEN_XL_G 设置为 0，可将 DEN 信息标记在选定的陀螺仪轴中，而将 DEN_XL_G 设置为 1，可将 DEN 信息标记在所选的加速度计轴中。

默认情况下，这些位被配置为包含所有陀螺仪轴上的信息。

5 中断生成

中断产生仅以加速度计数据为基础，因此要产生中断，加速度计传感器必须设置为活动工作模式（不能处于省电模式）；陀螺仪传感器可配置为省电模式，因为它与中断产生无关。

可对中断发生器进行配置，来检测：

- 自由落体；
- 唤醒；
- 6D/4D 方向检测；
- 活动/不活动和运动/静止识别。

所有这些中断信号，以及 FIFO 中断信号，可被独立地驱动至 INT1 和 INT2 中断引脚，或通过读取特定源寄存器位分别对其进行检测。

必须使用 CTRL3_C 寄存器的 H_LACTIVE 位来选择中断引脚极性。如果该位置为 0（默认值），则中断引脚为高电平激活，当检测到相关中断条件时，这些引脚从低电平变为高电平。否则，如果 H_LACTIVE 位置为 1（低电平激活），则中断引脚正常为高电平，当达到中断条件时，从高电平变为低电平。

CTR3_C 的 PP_OD 位允许将中断引脚性质从推挽更改为开漏。如果 PP_OD 位置为 0，则中断引脚处于推挽配置（对于高电平和低电平均为低阻抗输出）。当 PP_OD 位置为 1 时，只有中断活动状态是低阻抗输出。

5.1 中断引脚配置

该器件具有两个引脚，可激活引脚来生成数据准备就绪或中断信号。这些引脚的功能，对于 INT1 引脚是通过 MD1_CFG 和 INT1_CTRL 寄存器来进行选择，对于 INT2 引脚是通过 MD2_CFG 和 INT2_CTRL 寄存器来进行选择。

以下概述给出了这些中断控制寄存器的简要描述；这些位的默认值等于 0，对应于“禁用”。要使能引脚上特定中断信号的线路，须将有关位置为 1。

表 18. INT1_CTRL 寄存器

b7	b6	b5	b4	b3	b2	b1	b0
DEN_ DRDY_ 标志	INT1_ CNT_ BDR	INT1_ FIFO_ FULL	INT1_ FIFO_ OVR	INT1_ FIFO_TH	INT1_ BOOT	INT1_ DRDY_G	INT1_ DRDY_ XL

- DEN_DRDY_flag: INT1 上的 DEN_DRDY 标记中断
- INT1_CNT_BDR: INT1 上 FIFO COUNTER_BDR_IA 中断
- INT1_FIFO_FULL: INT1 上 FIFO 全满标志中断
- INT1_FIFO_OVR: INT1 上 FIFO 溢出标志中断
- INT1_FIFO_TH: INT1 上 FIFO 阈值中断
- INT1_BOOT: INT1 上启动中断
- INT1_DRDY_G: INT1 上陀螺仪数据准备就绪
- INT1_DRDY_XL: INT1 上加速度计数据准备就绪

表 19. MD1_CFG 寄存器

b7	b6	b5	b4	b3	b2	b1	b0
INT1_ SLEEP_ CHANGE	0	INT1_ WU	INT1_ FF	0	INT1_ 6D	0	0

- INT1_SLEEP_CHANGE: INT1 上的活动/不活动识别事件中断
- INT1_WU: INT1 上唤醒中断
- INT1_FF: INT1 上自由落体中断
- INT1_6D: INT1 上 6D 检测中断

表 20. INT2_CTRL 寄存器

b7	b6	b5	b4	b3	b2	b1	b0
0	INT2_ CNT_ BDR	INT2_ FIFO_ FULL	INT2_ FIFO_ OVR	INT2_ FIFO_ TH	INT2_ DRDY_ TEMP	INT2_ DRDY_ G	INT2_ DRDY_ XL

- INT2_CNT_BDR: INT2 上 FIFO COUNTER_BDR_IA 中断
- INT2_FIFO_FULL: INT2 上 FIFO 全满标志中断
- INT2_FIFO_OVR: INT2 上 FIFO 溢出标志中断
- INT2_FIFO_TH: INT2 上 FIFO 阈值中断
- INT2_DRDY_TEMP: INT2 上温度数据准备就绪
- INT2_DRDY_G: INT2 上陀螺仪数据准备就绪
- INT2_DRDY_XL: INT2 上加速度计数据准备就绪

表 21. MD2_CFG 寄存器

b7	b6	b5	b4	b3	b2	b1	b0
INT2_ SLEEP_ CHANGE	0	INT2_ WU	INT2_ FF	0	INT2_ 6D	0	INT2_ TIME STAMP

- INT2_SLEEP_CHANGE: INT2 上的活动/不活动识别事件中断
- INT2_WU: INT2 上唤醒中断
- INT2_FF: INT2 上自由落体中断
- INT2_6D: INT2 上 6D 检测中断
- INT2_TIMESTAMP: INT2 上的时间戳溢出报警中断

如果多个中断信号发送到同一个引脚上 (INTx)，则该引脚的逻辑电平为所选中断信号组合的“或”。要识别出是什么事件产生了中断条件，则须读取相关源寄存器：

- WAKE_UP_SRC 和 D6D_SRC (基本中断功能)
- STATUS_REG (用于数据准备就绪信号)
- FIFO_STATUS2 (用户 FIFO)。

ALL_INT_SRC 寄存器在一个寄存器中汇集了基本中断功能事件状态 (6D/4D、自由落体、唤醒、活动/不活动和时间戳)：可以读取该寄存器，以便为后续特定的源寄存器读取寻址。

CTRL4_C 寄存器的 INT2_on_INT1 引脚能够将所有已使能的中断信号进行逻辑“与”后驱动到 INT1 引脚上 (通过将该位置为 1)。当该位置为 0 时，中断信号分配到 INT1 和 INT2 引脚上。

必须通过设置 INT_CFG1 寄存器中的 INTERRUPTS_ENABLE 位使能基本中断。

INT_CFG0 寄存器的 LIR 位使能基本中断功能的锁存中断：当此位置为 1 且中断标记发送至 INT1 引脚和/或 INT2 引脚时，中断保持激活状态，直至 ALL_INT_SRC 寄存器或相应的源寄存器被读取，它在下一个 ODR 周期复位。仅当功能路由至 INT1 或 INT2 引脚时才对功能使能锁存中断：如果使能了锁存模式，但是中断信号未驱动至中断引脚，那么锁存功能不起作用。



注：如果使能了锁存模式（ $LIR = 1$ ），不建议连续轮询 `ALL_INT_SRC` 或专用源寄存器，因为通过读取它们，嵌入功能将实现内部复位；本例中建议同步（有中断事件）读取源寄存器。

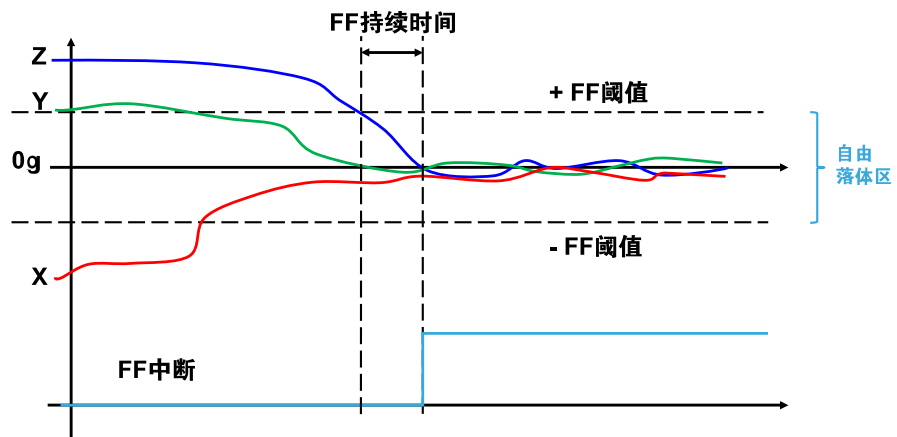
在使能锁存模式（ $LIR=1$ ）后，可以在读取 `ALL_INT_SRC`（或相关源寄存器）时强制路由至 `INT1` 或 `INT2` 引脚的中断信号及其相应中断状态位立即复位。为了执行立即复位，`INT_CFG0` 寄存器的 `INT_CLR_ON_READ` 位必须置为 1。当 `INT_CLR_ON_READ` 位等于 0 时，在下一个 `ODR` 周期发生复位。



5.2 自由落体中断

自由落体检测涉及特定的寄存器配置，可以识别器件何时处于自由落体：沿各轴所测得的加速度均为 **0**。真实情境下，一个“自由落体”定义为大约零-**g** 水平，其中所有加速度均足够小，可以产生中断。自由落体事件检测相关的可配置的阈值和持续时间参数：阈值参数定义了自由落体的幅度；持续时间参数定义了可识别的自由落体中断事件的最小持续时间（图 12. 自由落体中断）。

图 12. 自由落体中断



通过将 INT_CFG1 寄存器中的 INTERRUPTS_ENABLE 位置为 1，可启用该自由落体中断信号，将 MD1_CFG 寄存器的 INT1_FF 位或 MD2_CFG 寄存器的 INT2_FF 位置为 1，可将该中断信号驱动至两个中断引脚上；还可通过读取 WAKE_UP_SRC 寄存器的 FF_IA 位对其进行检查。

如果锁存模式禁用（INT_CFG0 的 LIR 位置为 0），则当检测不到自由落体条件时，中断信号会自动复位。如果锁存模式启用且自由落体中断信号被驱动至中断引脚，那么当发生自由落体事件且声明了中断引脚时，必须通过读取 WAKE_UP_SRC 或 ALL_INT_SRC 寄存器来将其复位。如果启用了锁存模式，但是中断信号未驱动至中断引脚，那么锁存功能不起作用。

FREE_FALL 寄存器用来配置阈值参数；无符号阈值与 FF_THS[2:0] 的值相关，如表 22. 自由落体阈值 LSB 值所示。此表中给出的值对于每个加速度计满量程值均有效。

表 22. 自由落体阈值 LSB 值

FREE_FALL - FF_THS[2:0]	阈值 LSB 值 [mg]
000	156
001	219
010	250
011	312
100	344
101	406
110	469
111	500

持续时间在 N/ODR_XL 中测得，其中 N 为 FREE_FALL / WAKE_UP_DUR 寄存器 FF_DUR[5:0] 字段的内容，ODR_XL 为加速度计数据率。

下面给出了自由落体事件识别的基本 SW 程序。

1. 将 60h 写入 CTRL1_XL // 启动加速度计
// ODR_XL = 417 Hz, FS_XL = ±2 g
2. 将 41h 写入 INT_CFG0 // 启用锁存模式并在读取时复位
3. 将 80h 写入 INT_CFG1 // 启用中断功能
4. 将 00h 写入 WAKE_UP_DUR // 设置事件持续时间（FF_DUR5 位）
5. 将 33h 写入 FREE_FALL // 设置 FF 阈值（FF_THS[2:0] = 011b）
// 设置六个采样事件持续时间（FF_DUR[5:0] = 000110b）
6. 将 10h 写入 MD1_CFG // FF 中断驱动至 INT1 引脚

示例代码中利用设置为 312mg 的阈值，用于自由落体识别，该事件由硬件通过 INT1 引脚进行通知。FREE_FALL / WAKE_UP_DUR 寄存器的 FF_DUR[5:0] 字段像这样配置：忽略短于 $6/ODR_{XL} = 6/412 \text{ Hz} \approx 15 \text{ ms}$ 的事件，以避免错误检测。

5.3 唤醒中断

唤醒功能可利用斜率滤波器（更多详细信息参见第 3.4.1 节 加速度计斜率滤波器）或高通数字滤波器来实现，如图 2. 加速度计滤波链所示。所用滤波器可通过 INT_CFG0 寄存器的 SLOPE_FDS 位来选择：如果该位被置为 0（默认值），则使用斜率滤波器；如果被置为 1，则使用 HPF 数字滤波器。此外，可以将唤醒功能配置为在可编程位置执行绝对唤醒。这可以通过将 INT_CFG0 寄存器的 SLOPE_FDS 位或 WAKE_UP_THS 寄存器的 USR_OFF_ON_WU 位置为 1 来实现。使用此配置时，唤醒功能的输入数据来自于低通滤波器路径，而可编程位置则作为偏移量减去。可编程位置可通过 X_OFS_USR、Y_OFS_USR 和 Z_OFS_USR 寄存器进行配置（更多详细信息见第 4.6 节 加速度计偏移寄存器）。

如果一定数量的连续滤波数据超出了所配置阈值，则产生唤醒中断信号（图 13. 唤醒中断（利用斜率滤波器））。该无符号阈值由 WAKE_UP_THS 寄存器的 WK_THS[5:0] 位来定义；这些 6 位的 1 LSB 值取决于所选加速度计满量程和 WAKE_UP_DUR 寄存器的 WAKE_THS_W 位的值：

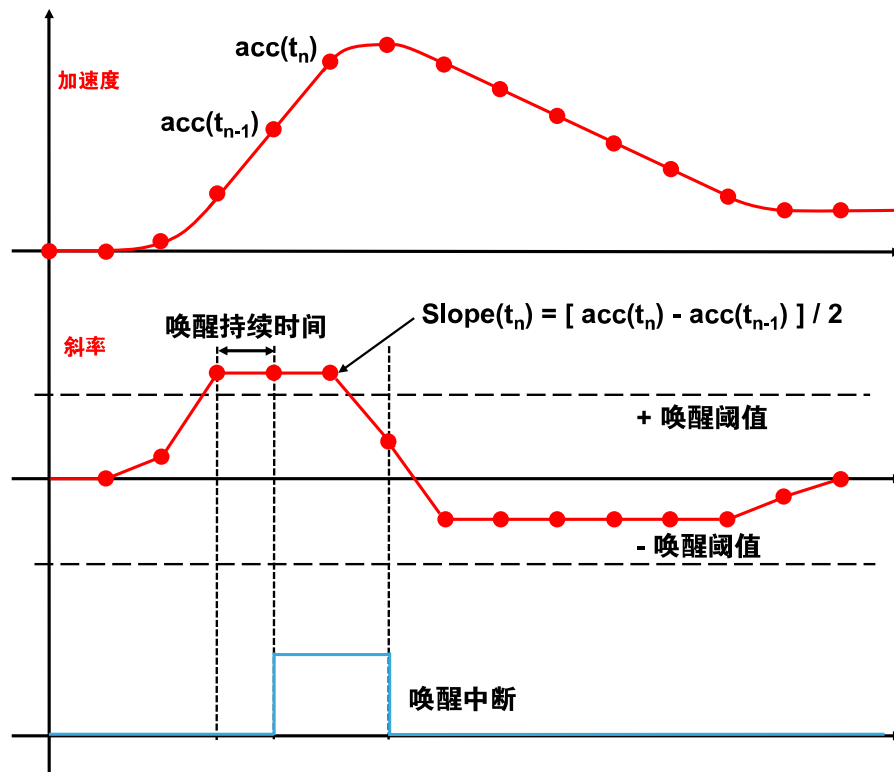
- 如果 WAKE_THS_W = 0，则 $1 \text{ LSB} = \text{FS_XL} / 2^6$ ；
- 如果 WAKE_THS_W = 1，则 $1 \text{ LSB} = \text{FS_XL} / 2^8$ 。

阈值可应用于正负数据：对于唤醒中断生成，滤波数据的绝对值必须大于阈值。

持续时间参数定义了所识别的唤醒事件的最小持续时间；其值由 WAKE_UP_DUR 寄存器的 WAKE_DUR[1:0] 位来设置：1 LSB 对应于 $1/\text{ODR_XL}$ 时间，这里 ODR_XL 为加速度计输出数据率。要避免因输入信号寄生尖峰而产生不期望的唤醒中断，适当定义持续时间参数是非常重要的。

通过将 INT_CFG1 寄存器中的 INTERRUPTS_ENABLE 位置为 1，可启用该中断信号，将 MD1_CFG 寄存器的 INT1_WU 位或 MD2_CFG 寄存器的 INT2_WU 位置为 1，可将该中断信号驱动至两个中断引脚上；还可通过读取 WAKE_UP_SRC 或 ALL_INT_SRC 寄存器的 WU_IA 位对其进行检查。WAKE_UP_SRC 寄存器的 X_WU、Y_WU、Z_WU 位指示哪个轴触发了唤醒事件。

图 13. 唤醒中断（利用斜率滤波器）





如果锁存模式禁用（INT_CFG0 的 LIR 位置为 0），则当滤波数据低于阈值时，中断信号会自动复位。如果锁存模式使能且唤醒中断信号被驱动至中断引脚，那么当发生唤醒事件且声明了中断引脚时，必须通过读取 WAKE_UP_SRC 寄存器或 ALL_INT_SRC 寄存器来将其复位。X_WU、Y_WU 和 Z_WU 位在执行读取前维持生成中断的状态，并在下一个 ODR 周期释放。如果除了 WU_IA 位还必须评估 WU_X、WU_Y 和 WU_Z 位，建议直接读取 WAKE_UP_SRC 寄存器（此特定情况下不使用 ALL_INT_SRC 寄存器）。如果使能了锁存模式，但是中断信号未驱动至中断引脚，那么锁存功能不起作用。

以下给出使用高通数字滤波器进行唤醒事件识别的基本软件程序。

1. 将 60h 写入 CTRL1_XL // 启动加速度计
// ODR_XL = 417 Hz, FS_XL = $\pm 2 g$
2. 将 51h 写入 INT_CFG0 // 使能锁存模式（在读取时复位）和数字高通滤波器
3. 将 80h 写入 INT_CFG1 // 使能中断功能
4. 将 00h 写入 WAKE_UP_DUR // 无持续时间和唤醒阈值权重选择（1 LSB = $FS_XL / 2^6$ ）
5. 将 02h 写入 WAKE_UP_THS // 设置唤醒阈值
6. 将 20h 写入 MD1_CFG // 唤醒中断驱动至 INT1 引脚

由于持续时间被置为 0，因此每个 X、Y、Z 滤波数据超出所配置阈值时，会生成唤醒中断信号。本例中，WAKE_UP_THS 寄存器的 WK_THS 字段被置为 000010b，因此活动/不活动阈值为 $62.5 mg (= 2 * FS_XL / 2^6)$ 。

如果唤醒功能利用斜率/高通数字滤波器实现，有必要考虑此功能使能后滤波器的设置时间。例如，当使用斜率滤波器时（不过对于使用高通数字滤波器要做类似考虑），唤醒功能基于阈值与两次(x,y,z)采样（当前和前次）加速度差的一半相比较（参考第 3.4.1 节 加速度计斜率滤波器）。

在第一个采样处，斜率滤波器输出计算为当前采样[例如，(x,y,z) = (0,0,1g)]与前次采样(x,y,z)=(0,0,0)（因为前次采样不存在）之差的一半。因此，在 z 轴上，斜率滤波器的第一个输出值为 $(1g - 0)/2 = 500 mg$ ，发生伪中断事件的情况下，该值可能会大于阈值。中断信号保持为高电平，并持续 1 个 ODR，然后变为低电平。

要避免产生这种伪中断，可以有多种解决方案。下面三个备选解决方案（对于斜率滤波器的情形）：

- a. 忽略第一个产生的唤醒信号；
- b. 将中断信号驱动到 INT1/2 引脚前，增加一段高于 1 个 ODR 的等待时间；
- c. 初始时设置一个较高的 ODR（833 Hz），这样会在较短时间内产生最初的 2 个采样，然后按下述步骤设置所需 ODR（例如 12.5 Hz）并驱动中断信号至引脚：

1. 将 00h 写入 WAKE_UP_DUR // 无持续时间和唤醒阈值权重选择（1 LSB = $FS_XL / 2^6$ ）
2. 将 02h 写入 WAKE_UP_THS // 设置唤醒阈值
3. 将 51h 写入 INT_CFG0 // 使能中断并应用斜率滤波器；锁存模式禁用
4. 将 80h 写入 INT_CFG1 // 使能中断功能
5. 将 70h 写入 CTRL1_XL // 启动加速度计
// ODR_XL = 833 Hz, FS_XL = $\pm 2 g$
6. 等待 4 ms // 插入（减少）等待时间
7. 将 10h 写入 CTRL1_XL // ODR_XL = 12.5 Hz
8. 将 20h 写入 MD1_CFG // 唤醒中断驱动至 INT1 引脚

5.4 6D/4D 定向检测

器件能够检测空间中器件的方向，可以很容易地实现移动设备的节能程序和自动图像旋转。

5.4.1 6D 定向检测

可以检测器件在空间中的六个方向；当器件从一个方向转向另一个方向时，中断信号产生。只要保持其位置，中断就不会重新产生。

对于两个连续采样，当只有一个轴超出所选阈值，其他两轴上测得的加速度值低于阈值时，会产生 6D 中断：

D6D_SRC 寄存器的 ZH, ZL, YH, YL, XH, XL 位表示出哪个轴触发了 6D 事件。

更具体地说：

表 23. D6D_SRC 寄存器

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

- 当器件从一个方向转向另一个方向时，D6D_IA 被置为高电平。
- 当垂直于 Z(Y, X)轴的面几乎是平面，Z(Y, X) 轴上测得的加速度为正且绝对值大于阈值时，ZH (YH, XH) 被置为高电平。
- 当垂直于(Y,X)轴的面几乎是平面，Z(Y,X) 轴上测得的加速度为负且绝对值大于阈值时，ZL (YL, XL)被置为高电平。

THS_6D 寄存器的 SIXD_THS[1:0]位用来选择阈值，该阈值用于检测器件方向变化。下表中给出的阈值对于每个加速度计满量程值均有效。

表 24. 4D/6D 功能阈值

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

通过将 CTRL8_XL 寄存器的 LOW_PASS_ON_6D 位置为 1，此低通滤波器 LPF2 还可用于 6D 功能。

通过将 INT_CFG1 寄存器中的 INTERRUPTS_ENABLE 位置为 1，可使能该中断信号，将 MD1_CFG 寄存器的 INT1_6D 位或 MD2_CFG 寄存器的 INT2_6D 位置为 1，可将该中断信号驱动至两个中断引脚上；还可通过读取 D6D_SRC 寄存器的 D6D_IA 位对其进行检查。

如果锁存模式禁用（INT_CFG0 的 LIR 位置为 0），则中断信号仅激活 1/ODR_XL[s]，然后自动取消（ODR_XL 为加速度计输出数据率）。如果锁存模式使能，并且 6D 中断信号被驱动至中断引脚，那么当方向发生了改变且中断引脚被声明时，对 D6D_SRC 或 ALL_INT_SRC 寄存器的读取会清除请求，器件将识别另一个不同的方向。XL、XH、YL、YH、ZL 和 ZH 位不受 LIR 配置的影响：在读取 D6D_SRC 寄存器时，它们对应于器件的当前状态。如果使能了锁存模式，但是中断信号未驱动至中断引脚，那么锁存功能不起作用。

参考图 14. 6D 识别方向中所示的六种可能情形，表 25. 6D 定位下的 D6D_SRC 寄存器中显示了每个位置对应的 D6D_SRC 寄存器内容所示。

图 14. 6D 识别方向

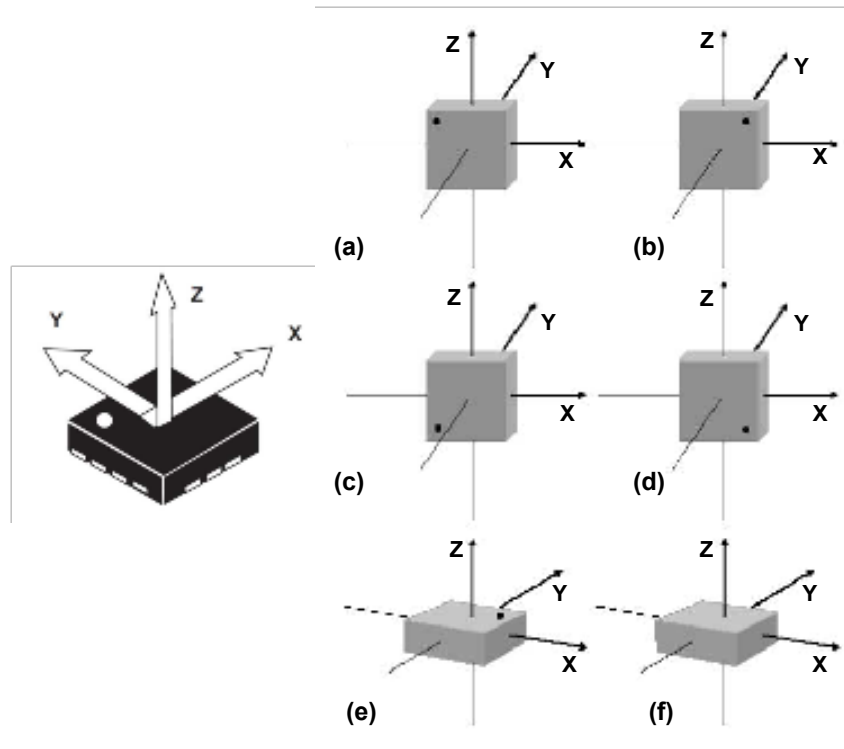


表 25. 6D 定位下的 D6D_SRC 寄存器

用例	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

6D 方向检测的基本软件例程如下。

1. 将 60h 写入 CTRL1_XL // 启动加速度计
// ODR_XL = 417 Hz, FS_XL = $\pm 2 g$
2. 将 41h 写入 INT_CFG0 // 使能锁存模式并在读取时复位
3. 将 80h 写入 INT_CFG1 // 使能中断功能
4. 将 40h 写入 THS_6D // 设置 6D 阈值 (SIXD_THS[1:0] = 10b = 60 degrees)
5. 将 01h 写入 CTRL8_XL // 将 LPF2 滤波器用于 6D 功能
6. 将 04h 写入 MD1_CFG // 6D 中断驱动至 INT1 引脚

5.4.2 4D 方向检测

4D 方向功能是 6D 功能的子集，它被专门定义来进行移动设备中的纵向和横向计算。它可通过将 THS_6D 寄存器的 D4D_EN 位置为 1 来使能。这种配置下，Z 轴位置检测被禁用，因此位置识别减少为表 25. 6D 定位下的 D6D_SRC 寄存器的(a)、(b)、(c)和(d)的情形。

5.5 活动/不活动和运动/静止识别

活动/不活动和运动/静止嵌入功能的工作原理与唤醒功能类似。如果在一段可编程的时间内未检测到运动状态，则生成不活动/静止状态事件；否则，在加速度计数据超过可配置的阈值时，生成活动/运动状态事件。

当活动/不活动识别功能激活时，器件能够自动将加速度计采样率降低至 12.5 Hz，当检测到唤醒中断事件时自动增加加速度计 ODR 和带宽。这个功能可以扩展到陀螺仪，有三种可能的选择：

- 陀螺仪配置不变；
- 陀螺仪进入睡眠模式；
- 陀螺仪进入省电模式。

利用此功能，根据用户所选的加速事件，系统可以高效地从低功耗模式（仅加速度计模式）转换成全性能模式，反之亦然，因此可以保证节能和灵活性。

通过将 INTERRUPTS_ENABLE 位置为 1，并配置 INT_CFG1 寄存器的 INACT_EN[1:0]位，可使能活动/不活动识别功能。如果 INT_CFG1 寄存器的 INACT_EN[1:0]位等于 00b，则运动/静止嵌入功能使能。下表总结了不活动事件的可能配置。

表 26. 不活动事件配置

INACT_EN[1:0]	加速度计	陀螺仪
00	不活动事件禁用	不活动事件禁用
01	XL ODR = 12.5 Hz	陀螺仪配置不变
10	XL ODR = 12.5 Hz	Gyro 处于睡眠模式
11	XL ODR = 12.5 Hz	陀螺仪处于省电模式



可以利用斜率滤波器（更多详细信息见第 3.4.1 节 加速度计斜率滤波器）或高通数字滤波器来实现活动/不活动和运动/静止识别功能，如图 2. 加速度计滤波链中所示。所用滤波器可通过 INT_CFG1 寄存器的 SLOPE_FDS 位来选择：如果该位被置为 0（默认值），则使用斜率滤波器；如果被置为 1，则使用高通数字滤波器。

此功能可完全由用户编程，利用专门的寄存器组对所期望的滤波数据幅度和时序进行编程（图 15. 活动/不活动识别（利用斜率滤波器））。

该无符号阈值由 WAKE_UP_THS 寄存器的 WK_THS[5:0] 位来定义：这些 6 位的 1 LSB 值取决于所选加速度计满量程和 WAKE_UP_DUR 寄存器的 WAKE_THS_W 位的值：

- 如果 WAKE_THS_W = 0，则 $1 \text{ LSB} = \text{FS_XL} / 2^6$ ；
- 如果 WAKE_THS_W = 1，则 $1 \text{ LSB} = \text{FS_XL} / 2^8$ 。

该阈值可适用于正负滤波数据。

当一定数量的连续 X、Y、Z 滤波数据小于所配置阈值时，忽略 CTRL1_XL 寄存器的 ODR_XL [3:0] 位（不活动），加速度计被内部地设置为 12.5 Hz，尽管 CTRL1_XL 内容保持不变。陀螺仪的特性随着 INT_CFG1 寄存器 INACT_EN[1:0] 位的配置而变化。待识别的不活动状态的持续时间由 WAKE_UP_DUR 寄存器的 SLEEP_DUR[3:0] 位来定义：1 LSB 对应于 $512/\text{ODR_XL}$ 的时间，这里 ODR_XL 为加速度计输出数据率。如果 SLEEP_DUR[3:0] 位被置为 0000b，则待识别不活动状态的持续时间等于 $16 / \text{ODR_XL}$ 的时间。

当检测到不活动状态时，中断被置为高电平并持续 $1/\text{ODR_XL}[\text{s}]$ 的时间周期，然后自动取消。

当一个轴的滤波数据大于可配置时间的阈值时，会立即恢复 CTRL1_XL 寄存器设置（活动），并且陀螺仪恢复到先前的状态。待识别活动状态的持续时间由 WAKE_UP_DUR 寄存器的 WAKE_DUR[1:0] 位来定义。1 LSB 对应于 $1/\text{ODR_XL}$ 的时间，这里 ODR_XL 为加速度计输出数据率。

当检测到活动状态时，中断被置为高电平并持续 $1/\text{ODR_XL}[\text{s}]$ 的时间周期，然后自动取消。

当使能活动/不活动检测功能时，通过将 MD1_CFG 寄存器的 INT1_SLEEP_CHANGE 位或 MD2_CFG 寄存器的 INT2_SLEEP_CHANGE 位置为 1，可将其状态驱动至两个中断引脚；还可通过读取 WAKE_UP_SRC 或 ALL_INT_SRC 寄存器的 SLEEP_CHANGE_IA 位来对其进行检查。

SLEEP_CHANGE_IA 位默认为脉冲模式。可通过将 INT_CFG0 寄存器的 LIR 位置为 1 和将 MD1_CFG 寄存器的 INT1_SLEEP_CHANGE 位或 MD2_CFG 寄存器的 INT2_SLEEP_CHANGE 位置为 1 来选择锁存模式。

WAKE_UP_SRC 寄存器的 SLEEP_STATE 位不受 LIR 配置的影响：在读取 WAKE_UP_SRC 寄存器时，它对应于器件的当前状态。

通过将 INT_CFG0 寄存器的 SLEEP_STATUS_ON_INT 位置为 1，将路由至 INT1 或 INT2 引脚的信号配置为活动/不活动状态（WAKE_UP_SRC 寄存器的 SLEEP_STATE 位）而不是睡眠-变化信号：它在不活动状态时为高电平，在活动状态时为低电平。此配置不支持锁存模式。



活动/不活动检测的基本软件例程如下所示：

1. 将 50h 写入 CTRL1_XL // 启动加速度计
// ODR_XL = 208 Hz, FS_XL = $\pm 2\text{ g}$
2. 将 40h 写入 CTRL2_G // 开启陀螺仪
// ODR_G = 104 Hz, FS_G = $\pm 250\text{ dps}$
3. 将 02h 写入 WAKE_UP_DUR // 设置不活动检测的持续时间
// 选择活动/不活动阈值分辨率和持续时间
4. 将 02h 写入 WAKE_UP_THS // 设置活动/不活动阈值
5. 将 00h 写入 INT_CFG0 // 选择睡眠-变化通知
// 选择斜率滤波器
6. 将 E0h 写入 INT_CFG1 // 使能中断
// 不活动配置：加速度计至 12.5 Hz
// 陀螺仪至省电模式
7. 将 80h 写入 MD1_CFG // 活动/不活动中断驱动至 INT1 引脚

本例中，WAKE_UP_THS 寄存器的 WK_THS 字段被置为 000010b，因此活动/不活动阈值为 62.5 mg ($= 2 * \text{FS_XL} / 2^6$ ，因为 WAKE_UP_DUR 寄存器的 WAKE_THS_W 位被置为 0)。

进行不活动检测前，X、Y、Z 斜率数据必须小于所配置阈值并持续一段时间，该时间由 WAKE_UP_DUR 寄存器的 SLEEP_DUR 字段定义：该字段被置为 0010b，对应 4.92 s ($= 2 * 512 / \text{ODR_XL}$)。经过这段时间之后，加速度计 ODR 内部设置为 12.5 Hz，陀螺仪内部设置为省电模式。

只要一个采样的（至少）一个轴的斜率数据大于阈值，就会检测到活动状态并立即恢复 CTRL1_XL 寄存器设置和开启陀螺仪，因为 WAKE_UP_DUR 寄存器的 WAKE_DUR[1:0]位被配置为 00b。

5.5.1 静止/运动检测

静止/运动检测是“活动/不活动”功能的特殊情况，其中，当检测到睡眠条件（相当于静止条件）时，ODR/电源模式不改变。通过将 INT_CFG1 寄存器的 INACT_EN[1:0]位置为 00b 激活静止/运动检测。



5.5.2

启动状态

器件上电后，执行一段 10 ms（最长）的启动程序来加载修整参数。启动完成后，加速度计和陀螺仪均自动配置为省电模式。启动时间内，寄存器不可访问。

上电后，可通过将 CTRL3_C 寄存器的 BOOT 位置为 1，来重载配置参数。

不需要切换设备电源线，器件控制寄存器内容不被修改，因此启动后器件工作模式不变。如果需要复位至控制寄存器的默认值，可通过将 CTRL3_C 寄存器的 SW_RESET 位置为 1 来实现。当此位置为 1 时，以下寄存器复位至其默认值：

- PIN_CTRL (02h);
- FIFO_CTRL1 (07h) 至 FIFO_CTRL4 (0Ah);
- COUNTER_BDR_REG1 (0Bh) 和 COUNTER_BDR_REG2 (0Ch);
- INT1_CTRL (0Dh) 和 INT2_CTRL (0Eh);
- CTRL1_XL (10h) 至 CTRL10_C (19h);
- FIFO_STATUS1 (3Ah) 和 FIFO_STATUS2 (3Bh);
- INT_CFG0 (56h) 至 MD2_CFG (5Fh);
- X_OFS_USR (73h), Y_OFS_USR (74h) 和 Z_OFS_USR (75h)。

SW_RESET 过程会花费 50 μ s；复位状态由 CTRL3_C 寄存器 SW_RESET 位的状态为信号：当复位完成时，该位自动置为低电平。

启动状态信号可通过将 INT1_CTRL 寄存器的 INT1_BOOT 位置为 1，来把启动状态信号驱动至 INT1 中断引脚：当正在运行启动时此信号被置为高电平，启动过程结束时此信号被重新置为低电平。

重启流程如下：

1. 将加速度计和陀螺仪设置为省电模式；
2. 将 INT1_CTRL 寄存器的 INT1_BOOT 位置为 1[可选]；
3. 将 CTRL3_C 寄存器的 BOOT 位置 1；
4. 监控重启状态，有三种可能：
 - a. 等待 10 ms；
 - b. 监控 INT1 引脚，直至其重新变为 0（本例中必须执行步骤 2）；
 - c. 轮询 CTRL3_C 寄存器的 BOOT 位，直至其重新变为 0。

复位流程如下：

1. 将加速度计和陀螺仪设置为省电模式；
2. 将 CTRL3_C 寄存器的 SW_RESET 位设置为 1；
3. 监控软件复位状态，有两种可能：
 - a. 等待 50 μ s
 - b. 轮询 CTRL3_C 寄存器的 SW_RESET 位，直至其重新变为 0。

为了避免冲突，重启和 sw 复位不能同时执行（不要同时将 CTRL3_C 寄存器的 BOOT 位和 SW_RESET 位同时置为 1）。上述流程必须顺序执行。

6 时间戳

器件能够提供与传感器数据对应的时间戳信息。

为了使能该功能，必须将 CTRL10_C 寄存器的 TIMESTAMP_EN 位置 1。时间步数由 TIMESTAMP3 & TIMESTAMP2 & TIMESTAMP1 & TIMESTAMP0 寄存器输出，表示为一个 32 位无符号数字。标称时间戳精度为 25 μ s。可通过 INTERNAL_FREQ_FINE 寄存器的 FREQ_FINE[7:0] 位获取实际时间戳精度值，其中包含实际 ODR（和时间戳速率）相对于标称值的百分比差值。

$$t_{actual}[\mu s] = \frac{1}{40000 \cdot (1 + 0.0015 \cdot FREQ_FINE)}$$

同样地，可以使用以下公式获取实际输出数据率：

$$ODR_{actual}[Hz] = \frac{6667 + 0.0015 \cdot FREQ_FINE \cdot 6667}{ODR_{coeff}}$$

其中的 ODR_{coeff} 值如下表所示。

表 27. ODR_{coeff} 值

选择的 ODR [Hz]	ODR_{coeff}
12.5	512
26	256
52	128
104	64
208	32
417	16
833	8
1667	4
3333	2
6667	1

如果加速度计和陀螺仪均处于省电模式，则时间戳计数器不工作且时间戳值冻结为最后的值。

当达到相当于约 30 小时的最大值 4294967295 LSB（等于 FFFFFFFFh）时，计数器自动复位至 00000000h 并继续计数。可通过向 TIMESTAMP2 寄存器中写入值 AAh，来随时将定时器计数复位至零。

在时间戳溢出条件发生前 6.4 ms，ALL_INT_SRC 的 TIMESTAMP_ENDCOUNT 位变为高电平。在读取 ALL_INT_SRC 寄存器时，此标记复位。还可以通过将 MD2_CFG 寄存器的 INT2_TIMESTAMP 位置为 1 将此信号路由至 INT2 引脚（75 μ s 持续时间脉冲）。

时间戳可以在 FIFO 中进行批处理（详细信息见第 7 节 先进先出缓冲区）。



7 先进先出缓冲区

为了限制主处理器干预并简化事件识别的后处理数据，ASM330LHH 嵌入了一个 3 kB 的先进先出缓冲器（FIFO）。

FIFO 可配置为存储以下数据：

- 陀螺仪传感器数据；
- 加速度计传感器数据；
- 时间戳数据；
- 温度传感器数据；

基于 FIFO 字将数据保存在 FIFO 中。一个 FIFO 字包含：

- 标签，1 字节
- 数据，6 字节

可通过六个专用寄存器（从地址 79h 至 7Eh）从 FIFO 检索数据：FIFO_DATA_OUT_X_L、FIFO_DATA_OUT_X_H、FIFO_DATA_OUT_Y_L、FIFO_DATA_OUT_Y_H、FIFO_DATA_OUT_Z_L 和 FIFO_DATA_OUT_Z_H。

FIFO_DATA_OUT_TAG 寄存器的 FIFO_TAG 字段可用于识别 FIFO 中字的含义，使得 FIFO 流的重建成为一项简单的任务。在使用专用 FIFO 配置的情况下选择传感器批处理数据率时，应用具有最大的灵活性。

利用 FIFO_CTRL4 寄存器的 FIFO_MODE[2:0]位，有六种不同的 FIFO 操作模式可供选择：

- Bypass 模式；
- FIFO 模式；
- Continue 模式；
- Continue-FIFO 模式；
- Bypass-Continue 模式；
- Bypass-FIFO 模式。

要监控 FIFO 状态（满，溢出，存储的采样数，等等...），可以使用两个专用寄存器：FIFO_STATUS1 和 FIFO_STATUS2。

可编程 FIFO 阈值可以利用 WTM[8:0]位在 FIFO_CTRL1 和 FIFO_CTRL2 中进行设置。

通过 INT1_CTRL 寄存器的 INT1_FIFO_FULL、INT1_FIFO_FTH 和 INT1_FIFO_OVR 位，以及 INT2_CTRL 寄存器的 INT2_FIFO_FULL、INT2_FIFO_FTH 和 INT2_FIFO_OVR 位，可以使能 FIFO 满、FIFO 阈值和 FIFO 溢出事件，在两个中断引脚（INT1 和 INT2）上产生专门的中断。



7.1 FIFO 说明和批处理传感器

FIFO 分成 512 个字，每个字 7 字节。1 个 FIFO 字包含 1 个带 TAG 信息的字节和 6 个数据字节：整个 FIFO 缓冲区大小等于 3584 字节，可包含 3072 字节的数据。TAG 字节包含表明数据保存在 FIFO 数据字段中的信息和其他有用信息。

FIFO 运行时间可配置：可使能元信息标记，以便在批处理传感器配置更改后通知用户。

批处理传感器可分为两个不同类别：

1. 主要传感器，即物理传感器：
 - a. 加速度计传感器；
 - b. 陀螺仪传感器；
2. 辅助传感器，其中包含设备状态信息：
 - a. 时间戳传感器；
 - b. 配置变化传感器（CFG-Change）；
 - c. 温度传感器。

可通过六个专用寄存器从 FIFO 检索数据：FIFO_DATA_OUT_X_L、FIFO_DATA_OUT_X_H、FIFO_DATA_OUT_Y_L、FIFO_DATA_OUT_Y_H、FIFO_DATA_OUT_Z_L 和 FIFO_DATA_OUT_Z_H。

通过内部数据就绪信号触发对 FIFO 的每次写操作（加速度计与陀螺仪之间的最快速传感器）。

7.2 FIFO 寄存器

FIFO 缓冲器由以下几项管理：

- 六个控制寄存器：FIFO_CTRL1、FIFO_CTRL2、FIFO_CTRL3、FIFO_CTRL4、COUNTER_BDR_REG1 和 COUNTER_BDR_REG2；
- 两个状态寄存器：FIFO_STATUS1 和 FIFO_STATUS2；
- 七个输出寄存器（标签 + 数据）：FIFO_DATA_OUT_TAG、FIFO_DATA_OUT_X_L、FIFO_DATA_OUT_X_H、FIFO_DATA_OUT_Y_L、FIFO_DATA_OUT_Y_H、FIFO_DATA_OUT_Z_L 和 FIFO_DATA_OUT_Z_H；
- 一些额外的位，将 FIFO 事件路由至两条中断线路：INT1_CTRL 寄存器的 INT1_CNT_BDR、INT1_FIFO_FULL、INT1_FIFO_OVR 和 INT1_FIFO_TH 位，以及 INT2_CTRL 寄存器的 INT2_CNT_BDR、INT2_FIFO_FULL、INT2_FIFO_OVR 和 INT2_FIFO_TH 位。

7.2.1 FIFO_CTRL1

FIFO_CTRL1 寄存器包含 9 位 FIFO 深度阈值的低位部分。对于完整的深度阈值配置，还要考虑 FIFO_CTRL2 寄存器的 WTM8 位。FIFO 阈值的 1 LSB 值称为 FIFO 字（7 字节）。

当 FIFO 中存储的字节数大于或等于深度阈值时，FIFO 深度标记（FIFO_STATUS2 寄存器的 FIFO_WTM_IA 位）升高。

要将 FIFO 深度限制为深度级别，FIFO_CTRL2 寄存器中 STOP_ON_WTM 位必须设置为 1。

表 28. FIFO_CTRL1 寄存器

位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
WTM7	WTM6	WTM5	WTM4	WTM3	WTM2	WTM1	WTM0

7.2.2 FIFO_CTRL2

表 29. FIFO_CTRL2 寄存器

位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
STOP_ON_WTM	0	0	ODRCHG_EN	0	0	0	WTM8

FIFO_CTRL2 寄存器包含 9 位 FIFO 深度阈值的高位部分（WTM8 位）。对于完整的深度阈值配置，还要考虑 FIFO_CTRL1 寄存器的 WTM[7:0] 位。寄存器包含了位 STOP_ON_WTM，可将 FIFO 深度限制在深度级别。

此外，FIFO_CTRL2 寄存器包含 ODRCHG_EN 位，可将此位置为 1，以使能 CFG-Change 辅助传感器在 FIFO 中的批处理（描述见下文）。



7.2.3 FIFO_CTRL3

表 30. FIFO_CTRL3 寄存器

位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
BDR_GY_3	BDR_GY_2	BDR_GY_1	BDR_GY_0	BDR_XL_3	BDR_XL_2	BDR_XL_1	BDR_XL_0

FIFO_CTRL3 寄存器包含用于选择 FIFO 中加速度计和陀螺仪传感器数据写入频率的字段。选择的批处理数据率必须小于等于通过 CTRL1_XL 和 CTRL2_G 寄存器的 ODR_XL 和 ODR_G 字段配置的输出数据率。

下面的表格显示了所有可选择的批处理数据率。

表 31. 加速度计批处理数据率

BDR_XL[3:0]	批处理数据率[Hz]
0000	FIFO 中不批处理
0001	12.5
0010	26
0011	52
0100	104
0101	208
0110	417
0111	833
1000	1667
1001	3333
1010	6667

表 32. 陀螺仪批处理数据率

BDR_GY[3:0]	批处理数据率[Hz]
0000	FIFO 中不批处理
0001	12.5
0010	26
0011	52
0100	104
0101	208
0110	417
0111	833
1000	1667
1001	3333
1010	6667
1011	6.5

7.2.4 FIFO_CTRL4

FIFO_CTRL4 寄存器包含用于选择 FIFO 中时间戳批处理抽取因子和温度传感器批处理数据率的字段。



时间戳写入速率被配置为加速度计和陀螺仪批处理数据率中的最大速率除以 DEC_TS_BATCH_[1:0] 字段指定的抽取因子。下表显示了可编程抽取因子。

表 33. 时间戳批处理数据率

DEC_TS_BATCH[1:0]	时间戳批处理数据率[Hz]
00	FIFO 中不批处理
01	$\max(\text{BDR_GY}[\text{Hz}], \text{BDR_XL}[\text{Hz}])$
10	$\max(\text{BDR_GY}[\text{Hz}], \text{BDR_XL}[\text{Hz}]) / 8$
11	$\max(\text{BDR_GY}[\text{Hz}], \text{BDR_XL}[\text{Hz}]) / 32$

温度批处理数据率可通过 ODR_T_BATCH_[1:0] 字段配置，如下表所示。

表 34. 温度传感器批处理数据率

ODR_T_BATCH[1:0]	温度批处理数据率[Hz]
00	FIFO 中不批处理
01	1.6
10	12.5
11	52

FIFO_CTRL4 寄存器还包含 FIFO 操作模式位。FIFO 操作模式如第 7.6 节 FIFO 模式中所示。

表 35. FIFO_CTRL4 寄存器

位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
DEC_TS_BATCH_1	DEC_TS_BATCH_0	ODR_T_BATCH_1	ODR_T_BATCH_0	0	FIFO_MODE2	FIFO_MODE1	FIFO_MODE0

7.2.5 COUNTER_BDR_REG1

由于 FIFO 可能包含温度数据、时间戳数据和元信息（即 CFG-Change 传感器），FIFO 提供了一种在 FIFO 中保存的加速度计或陀螺仪实际采样数的基础上同步 FIFO 读取的方式：BDR 计数器。

BDR 计数器可通过 COUNTER_BDR_REG1 和 COUNTER_BDR_REG2 寄存器进行配置。

表 36. COUNTER_BDR_REG1 寄存器

位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
-	RST_ COUNTER _BDR	TRIG_ COUNTER _BDR	0	0	0	CNT_BDR _TH_9	CNT_BDR _TH_8

可以将 RST_COUNTER_BDR 置位以便复位 BDR 计数器：它将自动复位至零。

TRIG_COUNTER_BDR 选择 BDR 计数器的触发信号：如果配置为 0，则选择加速度计传感器，否则选择陀螺仪传感器。

用户可以在 FIFO_STATUS2 寄存器中选择生成 COUNTER_BDR_IA 事件的阈值。在内部 BDR 计数器达到阈值后，将 COUNTER_BDR_IA 位置为 1。阈值可通过 CNT_BDR_TH_[9:0] 位进行配置。字段上部包含在寄存器 COUNTER_BDR_REG1 中。CNT_BDR_TH 阈值的 1 LSB 值称为一个加速度计/陀螺仪采样（X、Y 和 Z 数据）。

7.2.6 COUNTER_BDR_REG2

COUNTER_BDR_REG2 寄存器包含 BDR 计数器阈值的下部。

表 37. COUNTER_BDR_REG2 寄存器

位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
CNT_BDR _TH_7	CNT_BDR _TH_6	CNT_BDR _TH_5	CNT_BDR _TH_4	CNT_BDR _TH_3	CNT_BDR _TH_2	CNT_BDR _TH_1	CNT_BDR _TH_0

7.2.7 FIFO_STATUS1

FIFO_STATUS1 寄存器，连同 FIFO_STATUS2 寄存器一起，提供 FIFO 中存储的采样数相关信息。DIFF_FIFO 的 1 LSB 值称为一个 FIFO 字（7 字节）。

表 38. FIFO_STATUS1 寄存器

位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
DIFF_ FIFO_7	DIFF_ FIFO_6	DIFF_ FIFO_5	DIFF_ FIFO_4	DIFF_ FIFO_3	DIFF_ FIFO_2	DIFF_ FIFO_1	DIFF_ FIFO_0

7.2.8

FIFO_STATUS2

FIFO_STATUS2 寄存器，连同 FIFO_STATUS1 寄存器一起，可提供 FIFO 中存储的采样数相关信息和 FIFO 缓冲器当前状态信息（深度，溢出，满，BDR 计数器）。

表 39. FIFO_STATUS2 寄存器

位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
FIFO_WTM_IA	FIFO_OVR_IA	FIFO_FULL_IA	COUNTER_BDR_IA	FIFO_OVR_LATCHED	0	DIFF_FIFO_9	DIFF_FIFO_8

- FIFO_WTM_IA 表示深度状态。当 FIFO 中已存储的 FIFO 字（每个 7 字节）数量大于等于深度阈值水平时，此位变为高电平。通过将 INT1_CTRL 寄存器的 INT1_FIFO_TH 位或 INT2_CTRL 寄存器的 INT2_FIFO_TH 位置为 1，可将深度状态信号驱动至两个中断引脚上。
- 当 FIFO 完全填满，至少一个采样已经被覆盖掉以存储新数据时，FIFO_OVR_IA 变为高电平。通过将 INT1_CTRL 寄存器的 INT1_FIFO_OVR 位或 INT2_CTRL 寄存器的 INT2_FIFO_OVR 位置为 1，可将此信号驱动至两个中断引脚上。
- 当要存储在 FIFO 中的下一组数据会使 FIFO 全满（即，DIFF_FIFO_9 = 1）或生成 FIFO 溢出时，FIFO_FULL_IA 被置为高电平。通过将 INT1_CTRL 寄存器的 INT1_FIFO_FULL 位或 INT2_CTRL 寄存器的 INT2_FIFO_FULL 位置为 1，可将此信号驱动至两个中断引脚上。
- COUNTER_BDR_IA 表示 BDR 计数器状态。当加速度计或陀螺仪批处理采样数达到通过 COUNTER_BDR_REG1 和 COUNTER_BDR_REG2 寄存器的 CNT_BDR_TH[9:0]位配置的 BDR 计数器阈值时，此位被置为高电平。在读取 FIFO_STATUS2 寄存器时 COUNTER_BDR_IA 位自动复位。通过将 INT1_CTRL 寄存器的 INT1_CNT_BDR 位或 INT2_CTRL 寄存器的 INT2_CNT_BDR 位置为 1，可将 BDR 计数器状态驱动至两个中断引脚上。
- 当 FIFO 完全填满，至少一个采样已经被覆盖掉以存储新数据时，FIFO_OVR_LATCHED（如同 FIFO_OVR_IA 一样）变为高电平。两个标记之间的差异在于，FIFO_OVR_LATCHED 在读取 FIFO_STATUS2 寄存器时复位，而 FIFO_OVR_IA 在读取至少一个 FIFO 字时复位。这样就可以在从 FIFO 读取数据时检测 FIFO 溢出条件。
- DIFF_FIFO_[9:8]包含 FIFO 中存储的未读字数的高位部分。其低位部分由 FIFO_STATUS1 中的 DIFF_FIFO_[7:0]位表示。DIFF_FIFO_[9:0]字段的值对应 FIFO 中的 7 字节字的数量。

寄存器内容会与 FIFO 写操作和读操作同步更新。

注：BDU 功能还作用于 FIFO_STATUS1 和 FIFO_STATUS2 寄存器。当 BDU 位置为 1 时，必须首先读取 FIFO_STATUS1，然后读取 FIFO_STATUS2。

7.2.9

FIFO_DATA_OUT_TAG

通过读取 FIFO_DATA_OUT_TAG 寄存器，可以知道当前读取的数据属于哪个传感器并检查数据是否一致。

表 40. FIFO_DATA_OUT_TAG 寄存器

位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
TAG_SENSOR_4	TAG_SENSOR_3	TAG_SENSOR_2	TAG_SENSOR_1	TAG_SENSOR_0	TAG_CNT_1	TAG_CNT_0	TAG_PARITY

- TAG_SENSOR_[4:0] 字段标识 6 个数据字节中保存的传感器（表 41）；
- TAG_CNT_[1:0] 字段标识 FIFO 时隙（如后面的章节所述）；
- TAG_PARITY 位标识 FIFO_DATA_OUT_TAG 寄存器的内容是否损坏。

下表包含了 TAG_SENSOR_[4:0] 字段的所有可能值和相关传感器类型。

表 41. TAG_SENSOR 字段和相关传感器

TAG_SENSOR_[4:0]	传感器名称	传感器类别	说明
0x01	陀螺仪	主路	陀螺仪数据
0x02	加速度计	主路	加速度计数据
0x03	温度	辅助	温度数据
0x04	时间戳	辅助	时间戳数据
0x05	CFG_Change	辅助	元信息数据

TAG_PARITY 位可用于检查 FIFO_DATA_OUT_TAG 寄存器的内容。为此，用户可以执行以下程序：

1. 读取 FIFO_DATA_OUT_TAG 寄存器；
2. 对等于 1 的位进行计数；
3. 如果等于 1 的位数为偶数，则 FIFO_DATA_OUT_TAG 内容可靠，否则不可靠。

7.2.10

FIFO_DATA_OUT

可通过六个专用寄存器（从地址 79h 至地址 7Eh）从 FIFO 检索数据：FIFO_DATA_OUT_X_L、FIFO_DATA_OUT_X_H、FIFO_DATA_OUT_Y_L、FIFO_DATA_OUT_Y_H、FIFO_DATA_OUT_Z_L 和 FIFO_DATA_OUT_Z_H。

FIFO 输出寄存器的内容取决于传感器类别和类型，如下一节所述。

7.3

FIFO 批处理传感器

如前文所述，批处理传感器可分为两个不同类别：

1. 主要传感器；
2. 辅助传感器。

本部分将提供每个类别的所有详细信息。

7.4 主要传感器

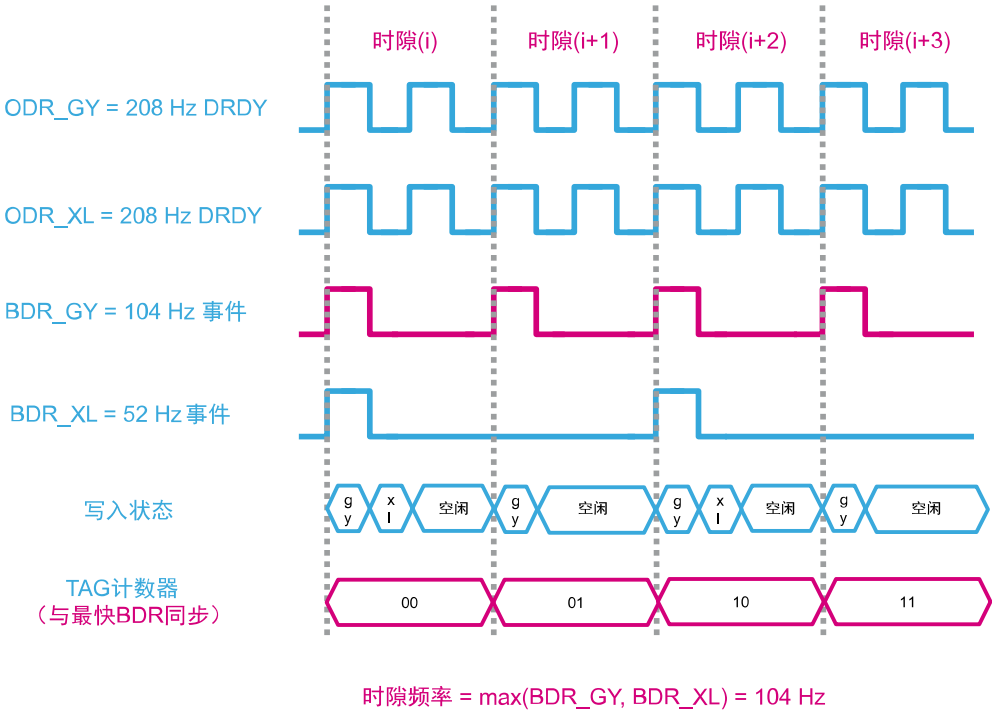
主要传感器是 ASM330LHH 设备物理传感器：加速度计和陀螺仪。批处理数据率可通过 FIFO_CTRL3 寄存器的 BDR_XL[3:0]和 BDR_GY[3:0]字段进行配置。批处理数据率必须小于等于通过 CTRL1_XL 和 CTRL2_G 寄存器的 ODR_XL[3:0]和 ODR_G[3:0]字段配置的相对传感器输出数据率。

主要传感器定义了 FIFO 时基。这意味着其他传感器中的每一个都与主要传感器定义的时基时隙相关。最快主要传感器的批处理事件也会使 TAG 计数器值（FIFO_DATA_OUT_TAG 寄存器的 TAG_CNT 字段）递增。此计数器由两个比特位组成，其值连续递增（从 00b 至 11b）以标识不同时序。

图 16. 主要传感器和时序的定义显示了批处理数据率事件的示例。BDR_GY 事件和 BDR_XL 事件标识相应传感器数据写入 FIFO 的时间。TAG 计数器的递增标识不同时间隙，其频率等于 BDR_XL 和 BDR_GY 中的最大值。



图 16. 主要传感器和时序的定义





下表显示了主要传感器的 FIFO 字格式，代表了从 78h 至 7Eh 的设备地址。

表 42. FIFO 中的主要传感器输出数据格式

标记	X_L	X_H	Y_L	Y_H	Z_L	Z_H
----	-----	-----	-----	-----	-----	-----

7.5 辅助传感器

辅助传感器被视为主要传感器的服务传感器。辅助传感器包括：

- 温度传感器（必须正确地配置 FIFO_CTRL4 寄存器的 ODR_T_BATCH_[1:0] 位）；
- 时间戳传感器：它保存对应于 FIFO 时隙的时间戳（必须正确地配置 CTRL10_C 寄存器的 TIMESTAMP_EN 位和 FIFO_CTRL4 寄存器的 DEC_TS_BATCH_[1:0] 位）；
- CFG-Change 传感器：它识别器件的一些配置中的变化（FIFO_CTRL2 寄存器的 ODRCHG_EN 位必须置为 1）。

辅助传感器无法触发 FIFO 中的写操作。在发生第一个主要传感器事件时写入它们的寄存器（即使为它们配置了更高的批处理数据率）。

下表显示了 FIFO 中的温度输出数据格式。

表 43. FIFO 中的温度输出数据格式

数据	FIFO_DATA_OUT 寄存器
TEMPERATURE[7:0]	FIFO_DATA_OUT_X_L
TEMPERATURE[15:8]	FIFO_DATA_OUT_X_H
0	FIFO_DATA_OUT_Y_L
0	FIFO_DATA_OUT_Y_H
0	FIFO_DATA_OUT_Z_L
0	FIFO_DATA_OUT_Z_H

下表显示了 FIFO 中的时间戳输出数据格式。

表 44. FIFO 中的时间戳输出数据格式

数据	FIFO_DATA_OUT 寄存器
TIMESTAMP[7:0]	FIFO_DATA_OUT_X_L
TIMESTAMP[15:8]	FIFO_DATA_OUT_X_H
TIMESTAMP[23:16]	FIFO_DATA_OUT_Y_L
TIMESTAMP[31:24]	FIFO_DATA_OUT_Y_H
0	FIFO_DATA_OUT_Z_L[3:0]
0	FIFO_DATA_OUT_Z_L[7:4]
BDR_XL	FIFO_DATA_OUT_Z_H[3:0]
BDR_GY	FIFO_DATA_OUT_Z_H[7:4]

如表 44 所示，时间戳数据还包含一些元信息，如果 CFG-Change 传感器在 FIFO 中不是批处理，则可以使用这些元信息检测 BDR 变化：主要传感器的批处理数据率。

CFG-Change 识别主要传感器的输出数据率、批处理数据率或其他配置中的运行时间变更。在应用支持的运行时间更改时，在第一个新的主要传感器事件时写入该传感器，然后是时间戳传感器（当时时间戳传感器不是批处理时）。该传感器可用于将传感器数据关联到器件时间戳，无需每次保存时间戳。它还可用于在嵌入式滤波器设置时间或其他配置变更（即切换模式、输出数据率等）的情况下，通知用户丢弃数据。

下表显示了 FIFO 中的 CFG-Change 输出数据格式。

表 45. FIFO 中的 CFG-change 输出数据格式

数据	FIFO_DATA_OUT 寄存器
LPF1_SEL_G	FIFO_DATA_OUT_X_H[0]
FTYPE[2:0]	FIFO_DATA_OUT_X_H[3:1]
0	FIFO_DATA_OUT_X_H[4]
FS_125	FIFO_DATA_OUT_X_H[5]
FS[1:0]_G	FIFO_DATA_OUT_X_H[7:6]
LPF2_XL_EN	FIFO_DATA_OUT_Y_L[0]
HPCF_XL[2:0]	FIFO_DATA_OUT_Y_L[3:1]
0	FIFO_DATA_OUT_Y_L[4]
0	FIFO_DATA_OUT_Y_L[5]
FS[1:0]_XL	FIFO_DATA_OUT_Y_L[7:6]
0	FIFO_DATA_OUT_Y_H[3:0]
0	FIFO_DATA_OUT_Y_H[5]
陀螺仪启动 ⁽¹⁾	FIFO_DATA_OUT_Y_H[6]
0	FIFO_DATA_OUT_Y_H[7]
ODR_XL	FIFO_DATA_OUT_Z_L[3:0]
ODR_GY	FIFO_DATA_OUT_Z_L[7:4]
BDR_XL	FIFO_DATA_OUT_Z_H[3:0]
BDR_GY	FIFO_DATA_OUT_Z_H[7:4]

1. 在陀螺仪启动阶段（最长启动时间为 70 ms）结束时生成的内部信号。

7.6 FIFO 模式

通过 FIFO_CTRL4 寄存器的 FIFO_MODE_[2:0] 字段，ASM330LHH FIFO 缓冲器可配置为六种不同的可选工作模式。可用配置确保了高度灵活性，并扩展了可用于应用开发的功能数量。

以下段落描述了 Bypass、FIFO、Continue、Continue-FIFO、Bypass-Continue 和 Bypass-FIFO 模式。

7.6.1 Bypass 模式

使能 Bypass 模式时，FIFO 不使用，缓冲器内容被清零，并保持为空，直至选择了另一种模式。当 FIFO_MODE_[2:0] 位被置为 000b 时，选用 Bypass 模式。当准备使用其他模式时，为了停止和复位 FIFO 缓冲器，必须使用 Bypass 模式。注意，将 FIFO 缓冲器置于 Bypass 模式时，整个缓冲器内容会被清零。

7.6.2

FIFO 模式

FIFO 模式中，缓冲器继续填充直至填满。然后停止采集数据，FIFO 内容保持不变，直至选用不同模式。

请按照以下步骤配置 FIFO 模式：

1. 使能以相应批处理数据率将传感器数据保存到 FIFO 中（如果可以配置）；
2. 将 FIFO_CTRL4 寄存器中的 FIFO_MODE_[2:0] 位置为 001b 来使能 FIFO 模式。

当选用此模式时，FIFO 开始采集数据。FIFO_STATUS1 和 FIFO_STATUS2 寄存器根据所存储的采样数来更新。

当 FIFO 已满时，FIFO_STATUS2 寄存器的 DIFF_FIFO_9 位被置为 1，不再有数据存储到 FIFO 缓冲区中。通过读取所有 FIFO_DATA_OUT（78h 至 7Eh）寄存器获取 FIFO_STATUS1 和 FIFO_STATUS2 寄存器的 DIFF_FIFO_[9:0] 位所指定的次数，可重新获取数据。

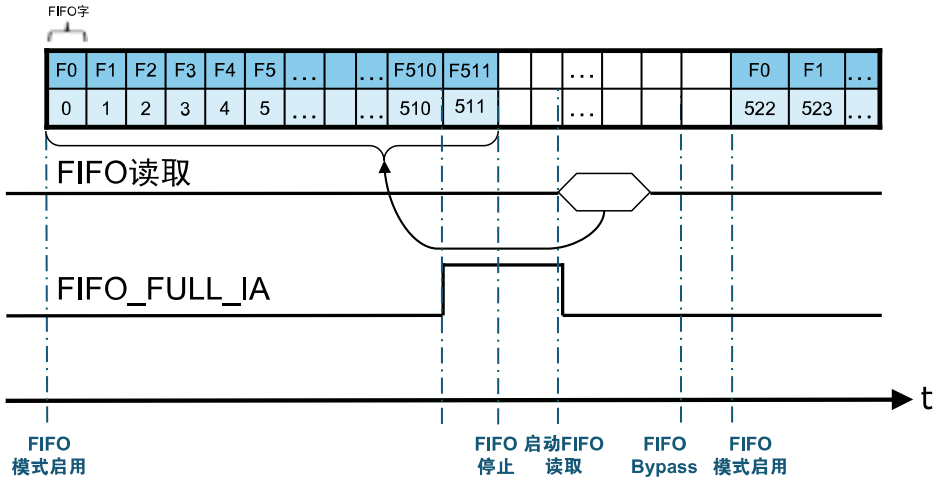
利用 FIFO_STATUS2 寄存器的 FIFO_WTM_IA 位，如果应用要求 FIFO 中采样数较低，则数据还可在达到阈值（FIFO_CTRL1 和 FIFO_CTRL2 寄存器中的 WTM[8:0]）时恢复。

如果 FIFO_CTRL2 寄存器的 STOP_ON_WTM 位置为 1，则 FIFO 空间大小被限制为 FIFO_CTRL1 和 FIFO_CTRL2 寄存器中 WTM[8:0] 位的值。这种情况下，如果下一次 FIFO 写操作时 FIFO 中的采样数达到或超过 WTM[8:0] 值，那么 FIFO_STATUS2 寄存器的 FIFO_FULL_IA 位会被置为高电平。

FIFO 模式下通信速度不是很重要，因为数据采集已停止，不存在已采集数据被覆盖的风险。重启 FIFO 模式之前，需要首先设置为 Bypass 模式，以完全清空 FIFO 内容。

图 1 显示了 FIFO 模式使用示例：只有一个传感器的数据存储到 FIFO 中。这些条件下，FIFO 缓冲区中可存储 512 个采样。在标记为 510 后，FIFO_STATUS2 寄存器的 FIFO_FULL_IA 位升高，以通知 FIFO 缓冲区将在下一次 FIFO 写操作时达到全满。在 FIFO 已满（FIFO_DIFF_9 = 1）后，数据采集停止。

图 17. FIFO 模式（STOP_ON_WTM = 0）



7.6.3 Continue 模式

Continue 模式中，FIFO 连续填充。当缓冲器满时，FIFO 索引重新从头开始，原有数据被新数据替代。最早先的数据继续被覆盖，直至读取操作释放了 FIFO 位置。要实现空间释放速度快于新数据产生速度，主处理器读取速度很重要。要停止此配置，必须选用 Bypass 模式。

按照这些步骤进行 Continue 模式配置（如果加速度计/陀螺仪数据准备就绪被用作 FIFO 触发）：

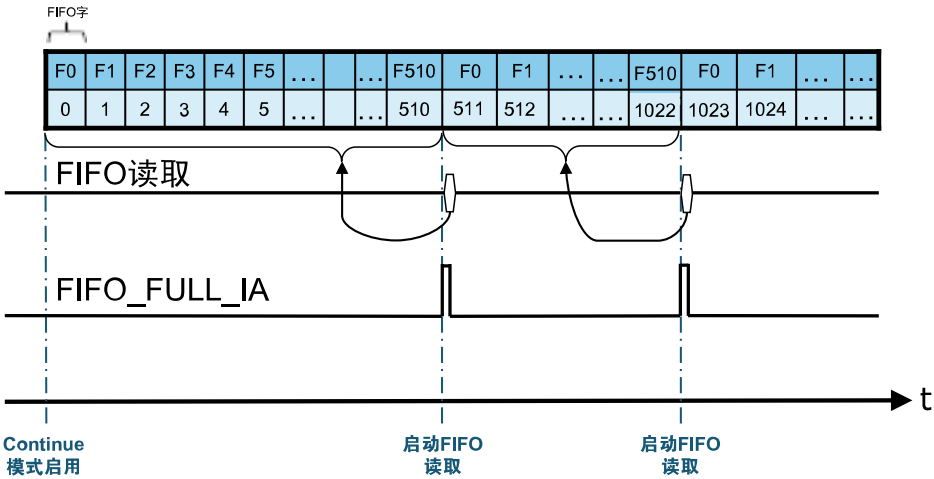
1. 使能以相应批处理数据率将传感器数据保存到 FIFO 中（如果可以配置）；
2. 将 FIFO_CTRL4 寄存器中的 FIFO_MODE_[2:0] 位置为 110b 来使能 FIFO 模式。

当选用此模式时，FIFO 连续采集数据。FIFO_STATUS1 和 FIFO_STATUS2 寄存器根据所存储的采样数来更新。当下一个 FIFO 写操作会使 FIFO 全满或生成 FIFO 溢出时，FIFO_STATUS2 寄存器的 FIFO_FULL_IA 位变为 1。FIFO_STATUS2 寄存器的 FIFO_OVR_IA 和 FIFO_OVR_LATCHED 位表示至少有一个 FIFO 字被覆盖以存储新数据。通过读取 FIFO_DATA_OUT（78h 至 7Eh）寄存器获取 FIFO_STATUS1 和 FIFO_STATUS2 寄存器的 DIFF_FIFO_[9:0] 位所指定的次数，数据可在 FIFO_FULL_IA 事件后恢复。利用 FIFO_STATUS2 寄存器的 FIFO_WTM_IA 位，数据还可在达到阈值（FIFO_CTRL1 和 FIFO_CTRL2 寄存器中的 WTM[8:0]）时恢复。如果 FIFO_CTRL2 寄存器的 STOP_ON_WTM 位置为 1，则 FIFO 空间大小被限制为 FIFO_CTRL1 和 FIFO_CTRL2 寄存器中 WTM[8:0] 位的值。这种情况下，如果下一次 FIFO 写操作时 FIFO 中的采样数达到或超过 WTM[8:0] 值，那么 FIFO_STATUS2 寄存器的 FIFO_FULL_IA 位会被置为高电平。

图 18. Continue 模式 显示了 Continue 模式使用的示例。示例中，仅一个传感器的数据存储在 FIFO 中，在发生 FIFO_FULL_IA 事件时读取 FIFO 采样且读取速度快于 $1 * ODR$ ，因此无数据丢失。这些条件下，所存储的采样数为 511。



图 18. Continue 模式





7.6.4 Continue-FIFO 模式

此模式是先前所述的 Continue 和 FIFO 模式的组合。在 Continue-FIFO 模式中，FIFO 缓冲器开始工作于 Continue 模式，当发生事件条件时切换为 FIFO 模式。

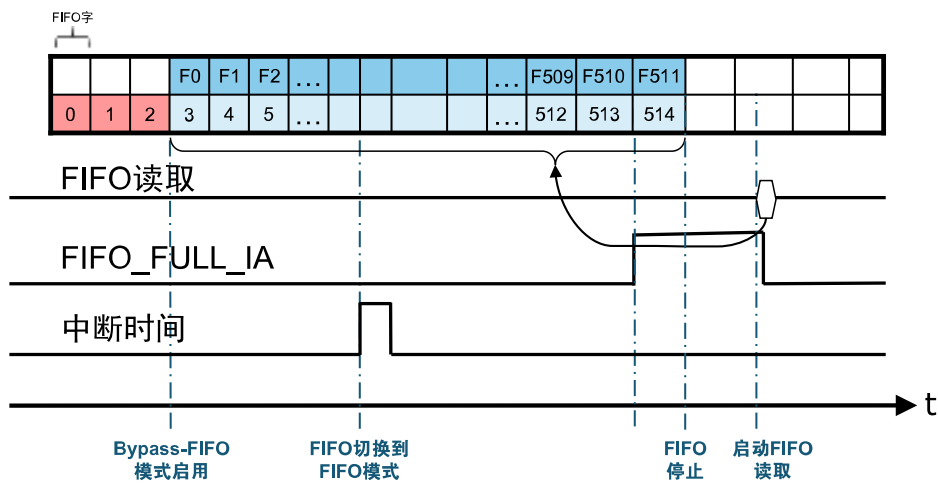
事件条件可为以下之一：

- 自由落体：必须配置事件检测，并且 MD2_CFG 寄存器的 INT2_FF 位必须被置为 1；
- 唤醒：必须配置事件检测，并且 MD2_CFG 寄存器的 INT2_WU 位必须被置为 1；
- 6D：必须配置事件检测，并且 MD2_CFG 寄存器的 INT2_6D 位必须被置为 1。

Continue-FIFO 模式对中断信号的边沿感应。在第一次中断事件时，FIFO 从 Continue 模式变为 FIFO 模式，并维持该模式，直到 Bypass 模式被置位。



图 19. Continue-FIFO 模式





按照这些步骤进行 Continue-FIFO 模式配置（如果加速度计/陀螺仪数据准备就绪被用作 FIFO 触发）：

1. 配置先前所述的其中一个事件；
2. 使能以相应批处理数据率将传感器数据保存到 FIFO 中（如果可以配置）；
3. 将 FIFO_CTRL4 寄存器中 FIFO_MODE_[2:0]位置为 011b 来使能 Continue-FIFO 模式。

在 Continue-FIFO 模式下，FIFO 缓冲区会继续填入数据。当 FIFO 将在下一次 FIFO 写操作时全满或溢出时，FIFO_FULL_IA 位变为高电平。

如果 FIFO_CTRL2 寄存器的 STOP_ON_WTM 位置为 1，则 FIFO 空间大小被限制为 FIFO_CTRL1 和 FIFO_CTRL2 寄存器中 WTM[8:0]位的值。这种情况下，如果下一次 FIFO 写操作时 FIFO 中的采样数达到或超过 WTM[8:0]值，那么 FIFO_STATUS2 寄存器的 FIFO_FULL_IA 位会被置为高电平。

发生触发事件时，可观察到两种不同的情况：

1. 如果 FIFO 缓冲器已满，则事件触发后第一次采样时即停止采集数据。FIFO 内容由该事件之前所采集的采样组成。
2. 如果 FIFO 尚未满，则继续填充直至填满，然后停止采集数据。

Continue-FIFO 可用来分析生成中断的采样历史。标准操作是在 FIFO 模式已触发、FIFO 缓冲区已满并停止时读取 FIFO 内容。



7.6.5 Bypass-Continue 模式

此模式是先前所述的 **Bypass** 和 **Continue** 模式的组合。在 **Bypass-Continue** 模式中，FIFO 缓冲器开始工作于 **Bypass** 模式，当发生事件条件时切换为 **Continue** 模式。

事件条件可为以下之一：

- 自由落体：必须配置事件检测，并且 MD2_CFG 寄存器的 INT2_FF 位必须被置为 1；
- 唤醒：必须配置事件检测，并且 MD2_CFG 寄存器的 INT2_WU 位必须被置为 1；
- 6D：必须配置事件检测，并且 MD2_CFG 寄存器的 INT2_6D 位必须被置为 1。

Bypass-Continue 模式对中断信号的边沿感应：在第一次中断事件时，FIFO 从 **Bypass** 模式切换到 **Continue** 模式，并维持该模式，直到 **Bypass** 模式被置位。

按照这些步骤进行 **Bypass-Continue** 模式配置（如果加速度计/陀螺仪数据准备就绪被用作 FIFO 触发）：

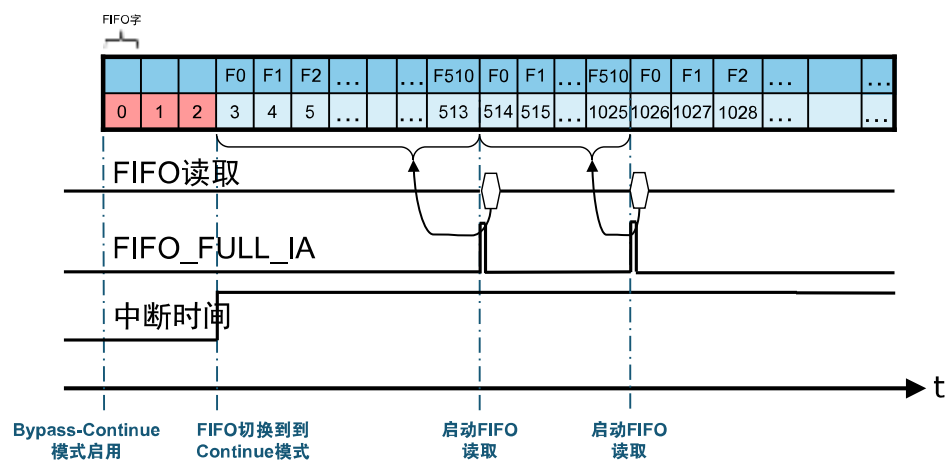
1. 配置先前所述的其中一个事件；
2. 使能以相应批处理数据率将传感器数据保存到 FIFO 中（如果可以配置）；
3. 将 FIFO_CTRL4 寄存器中 FIFO_MODE[2:0]位置为 100b 来使能 FIFO 旁路-连续模式。

当出现触发条件且缓冲器切换至 **Continue** 模式时，FIFO 缓冲器继续填充。当下一个要存储的数据集将会使 FIFO 满或溢出时，FIFO_FULL_IA 位被置为高电平。

Bypass-Continue 可用来在产生所配置中断时启动采集。



图 20. Bypass-Continue 模式





7.6.6 Bypass-FIFO 模式

此模式是先前所述的 Bypass 和 FIFO 模式的组合。在 Bypass-FIFO 模式中，FIFO 缓冲器开始工作于 Bypass 模式，当发生事件条件时切换为 FIFO 模式。

事件条件可为以下之一：

- 自由落体：必须配置事件检测，并且 MD2_CFG 寄存器的 INT2_FF 位必须被置为 1；
- 唤醒：必须配置事件检测，并且 MD2_CFG 寄存器的 INT2_WU 位必须被置为 1；
- 6D：必须配置事件检测，并且 MD2_CFG 寄存器的 INT2_6D 位必须被置为 1。

Bypass-FIFO 模式对中断信号的边沿感应。在第一次中断事件时，FIFO 从 Bypass 模式变为 FIFO 模式，并维持该模式，直到 Bypass 模式被置位。

按照这些步骤进行 Bypass-FIFO 模式配置（如果加速度计/陀螺仪数据准备就绪被用作 FIFO 触发）：

1. 配置先前所述的其中一个事件；
2. 使能以相应批处理数据率将传感器数据保存到 FIFO 中（如果可以配置）；
3. 将 FIFO_CTRL4 寄存器中 FIFO_MODE_[2:0]位置为 111b 来使能 Bypass-FIFO 模式。

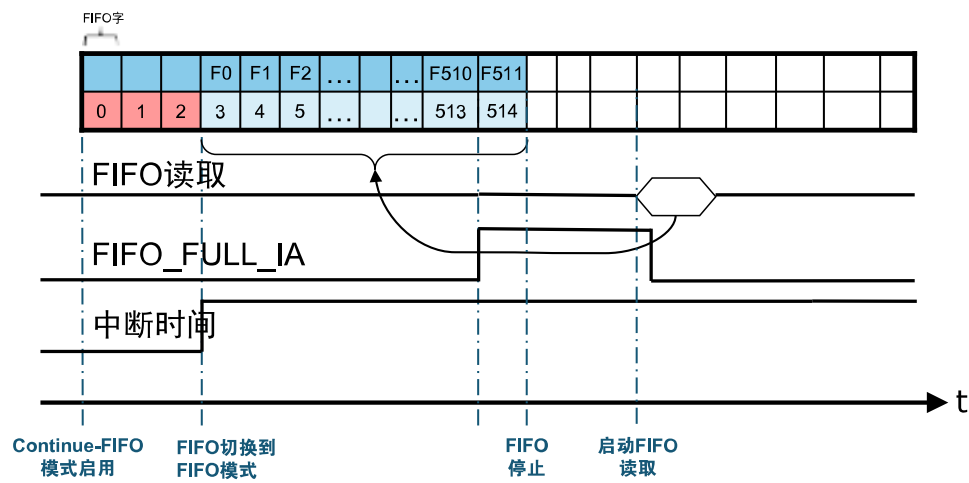
当出现触发条件且缓冲器切换至 FIFO 模式时，FIFO 缓冲器开始填充。当下一个要存储的数据集将会使 FIFO 满或溢出时，FIFO_FULL_IA 位被置为高电平，FIFO 停止采集数据。



Bypass-FIFO 用来分析生成中断的采样历史。



图 21. Bypass-FIFO 模式





7.6.7

从 FIFO 读取数据

当 FIFO 使能且其模式不是 Bypass 模式时，读取 FIFO 输出寄存器会返回原 FIFO 采样集。无论何时读取这些寄存器，其内容均会移动到 SPI/I²C 输出缓冲器。

理想地，FIFO 插槽会向上移动一格，以便释放空间给新的采样，并且 FIFO 输出寄存器载入 FIFO 缓冲器中存储的当前最旧的值。

从 FIFO 中恢复数据的建议方法如下：

1. 读取 FIFO_STATUS1 和 FIFO_STATUS2 寄存器来检查 FIFO 中存储了多少字。此信息在 DIFF_FIFO_[9:0]位中。
2. 对于 FIFO 中的每个字，读取 FIFO 字（标记和输出数据）并基于 FIFO 标记解读它。
3. 转至步骤 1。

通过从 FIFO 输出寄存器执行一定次数的读操作，直至缓冲器为空（FIFO_STATUS1 和 FIFO_STATUS2 寄存器的 DIFF_FIFO_[9:0]位等于 0），可以恢复全部 FIFO 内容。

建议避免在 FIFO 为空时读取它。

必须按 7 字节的倍数读取 FIFO 输出数据，从 FIFO_DATA_OUT_TAG 寄存器开始读取。为了通过一次多字节读取操作读取许多字，在器件中自动执行从地址 FIFO_DATA_OUT_Z_H 至 FIFO_DATA_OUT_TAG 的环行功能。



7.7 FIFO 深度阈值

FIFO 阈值是 ASM330LHH FIFO 的功能，可用于检查 FIFO 中的采样数何时达到定义的深度阈值水平。

FIFO_CTRL1 和 FIFO_CTRL2 寄存器的 WTM[8:0] 中有深度阈值水平。WTM[8:0] 字段的分辨率为 7 字节，相当于一个完整的 FIFO 字。因此，用户能够在 0 至 511 之间选择所需的值。

FIFO_STATUS2 寄存器的位 FIFO_WTM_IA 表示深度状态。如果 FIFO 中的字数达到或超过深度水平，则此位被置为高电平。通过将 FIFO_CTRL2 寄存器的 STOP_ON_WTM 位置为 1，FIFO 空间大小可由阈值水平来限制。



图 22. FIFO 阈值 (STOP_ON_WTM = 0)

WTM[8:0] = 21
STOP_ON_WTM = 0

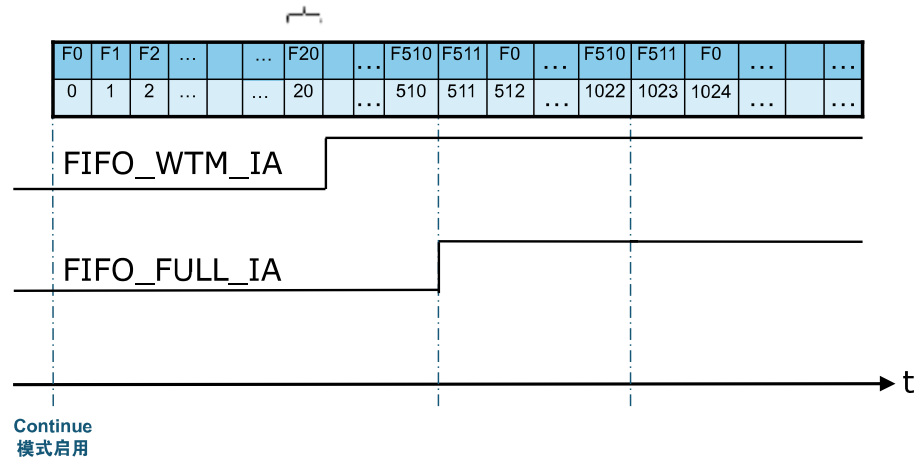


图 22. FIFO 阈值 ($STOP_ON_WTM = 0$) 显示了只存储加速度计 (或陀螺仪) 数据时, FIFO 阈值水平使用的示例。FIFO_CTRL2 寄存器中 $STOP_ON_WTM$ 位置为 0。利用 $WTM[8:0]$ 位, 阈值水平设置为 21。达到 21 后 (FIFO 中有 21 个字), FIFO_STATUS2 寄存器的 $FIFO_WTM_IA$ 位升高。因此, $STOP_ON_WTM$ 位置 0, FIFO 不会在第 21 个数据组时停止, 而是会继续存储数据, 直至 $FIFO_FULL_IA$ 标记被置为高电平。



图 23. FIFO 模式下的 FIFO 阈值 (STOP_ON_WTM = 1)

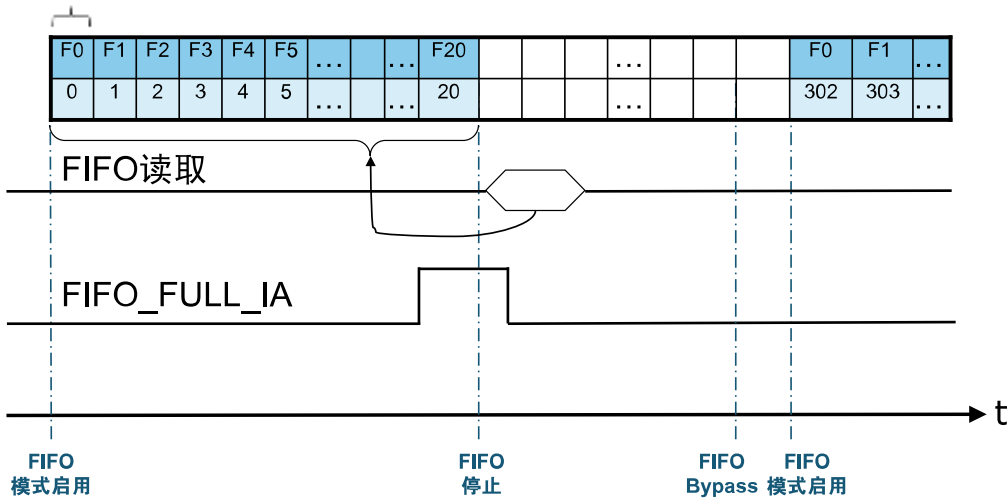


图 23. FIFO 模式下的 FIFO 阈值 ($STOP_ON_WTM = 1$) 显示了 FIFO 模式下 FIFO 阈值使用的示例，其中 FIFO_CTRL2 寄存器中 STOP_ON_WTM 位被置为 1。本例中只存储了加速度计（或陀螺仪）数据。利用 WTM[8:0] 位，阈值水平设置为 21，并定义当前 FIFO 空间大小。在 FIFO 模式下，数据保存在 FIFO 缓冲区，直至 FIFO 已满。在 FIFO 中保存下一个数据时，FIFO_STATUS2 寄存器的 FIFO_FULL_IA 位升高，将生成 FIFO 已满或溢出条件。当 FIFO 已满时，FIFO_STATUS2 寄存器的 FIFO_WTM_IA 位升高。



图 24. Continue 模式下的 FIFO 阈值 (STOP_ON_WTM = 1)

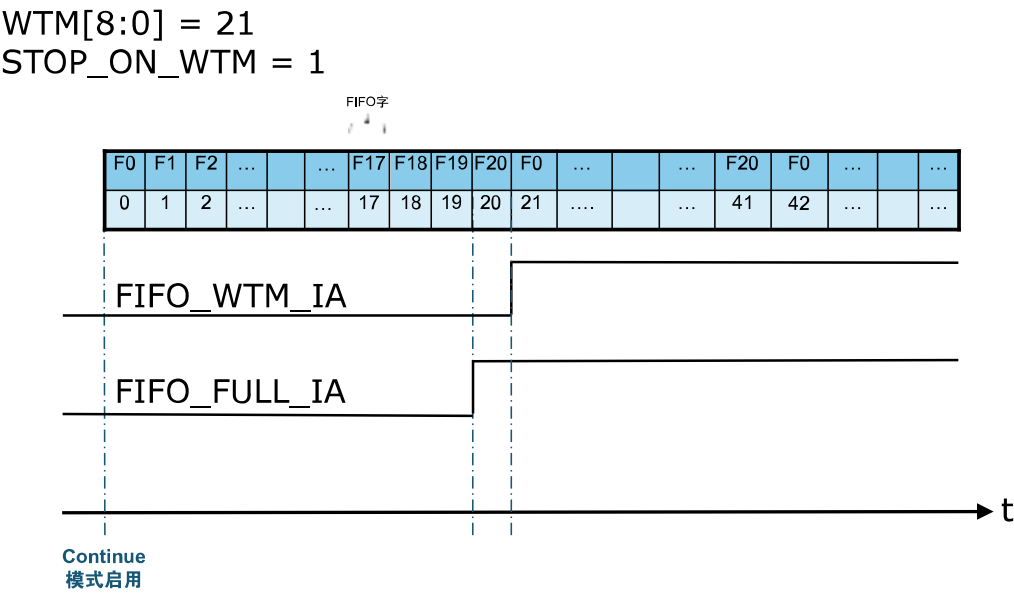


图 24. Continue 模式下的 FIFO 阈值 ($STOP_ON_WTM = 1$) 显示了 Continue 模式下 FIFO 阈值使用的示例，其中 FIFO_CTRL2 寄存器的 STOP_ON_WTM 位被置为 1。本例中只存储了加速度计（或陀螺仪）数据。利用 WTM[8:0] 位，阈值水平设置为 21。在 FIFO 中保存下一个数据时，FIFO_STATUS2 寄存器的 FIFO_FULL_IA 位升高，FIFO 已满。当 FIFO 已满时，FIFO_STATUS2 寄存器的 FIFO_WTM_IA 位升高。如果不从 FIFO 检索数据，新数据（标记为采样 21）将覆盖 FIFO 中的旧数据（标记为采样 F0）。

7.8 时间戳相关性

可通过三种不同方法重建 FIFO 流的时间戳：

1. 基本法，只使用时间戳传感器信息；
2. 存储器保存法，基于 FIFO_DATA_OUT_TAG 中的 TAG_CNT 字段
3. 混合法，基于 TAG_CNT 字段和抽取时间戳传感器的组合使用

基本法保证时间戳重建的最高精度，但会浪费 FIFO 中的大量存储空间。在每个时隙将时间戳传感器写入 FIFO。如果发生溢出条件，从 FIFO 检索数据的正确程序是丢弃新的时间戳传感器之前的每次数据读取。

存储器保存法只使用 TAG_CNT 信息，并且当 TAG_CNT 值增加时，应按如下方式更新在软件层保存的时间戳：

$$timestamp = timestamp(i-1) + \frac{1}{\max(BDR_XL, BDR_GY)}$$

存储器保存法允许用户在 FIFO 中保存尽可能多的数据。通过这种方式，将所有时间戳相关性转发到应用处理器。当可能发生溢出条件时，不建议使用此方法。

混合法是一种折衷方法，是前面两种解决方案的组合。时间戳被配置为在抽取后写入 FIFO。当 TAG_CNT 值增加时，应以存储器保存法更新在软件层保存的时间戳，而在读取时间戳传感器时，应使用来自传感器的正确值重新校准在软件层保存的时间戳。



8 温度传感器

器件具有内部温度传感器，适用于环境温度测量。

如果加速度计和陀螺仪传感器均处于省电模式，则温度传感器关闭。

温度数据率等于 52 Hz，无论加速度计和陀螺仪采用哪种配置。

对于温度传感器，数据准备就绪信号由 STATUS_REG 寄存器的 TDA 位表示。通过将 INT2_CTRL 寄存器的 INT2_DRDY_TEMP 位置为 1，可将该信号驱动至 INT2 引脚。

温度数据由 OUT_TEMP_H 和 OUT_TEMP_L 寄存器联合给出，以二进制补码的格式表示为一个 16 位的数字，其灵敏度为+256 LSB/°C。输出零值对应于 25 °C。

温度传感器数据也可以采用可配置的批处理数据率存储在 FIFO 中（详细信息见第 7 节 先进先出缓冲区）。

8.1 温度数据计算示例

下表提供了在不同环境温度值下从温度数据寄存器中读取数据的几个基本示例。本表中所列值是在理想器件校准的假设下给出的（即，无偏移，无增益误差，……）。

表 46. 输出数据寄存器内容 vs. 温度

温度值	寄存器地址	
	OUT_TEMP_H (21h)	OUT_TEMP_L (20h)
0 °C	E7h	00h
25 °C	00h	00h
50 °C	19h	00h



9 自检功能

嵌入式自检功能可支持无需移动器件而对其功能进行检查。

9.1 加速度计自检

当加速度计自检使能时，传感器上会施加一个驱动力，模拟一定的加速度。这种情况下，传感器输出会在其 DC 电平上表现出变化，该电平通过灵敏度值关联到所选量程。

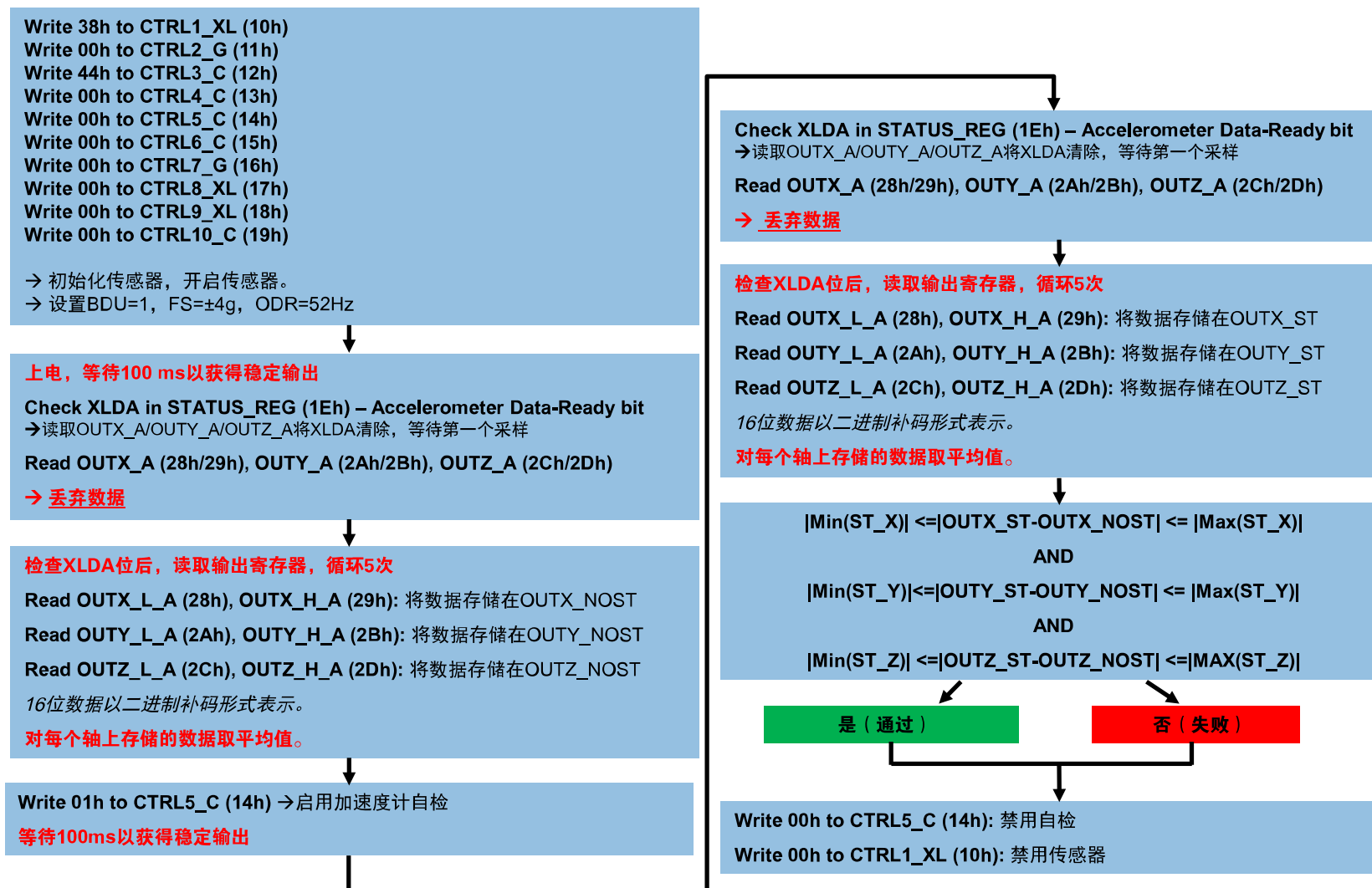
当 **CTRL5_C** 寄存器的 **ST[1:0]_XL** 位被设定为 **00b** 时，加速度计自检功能关闭；当 **ST[1:0]_XL** 位被置为 **01b**（正符号自检）或 **10b**（负符号自检）时，该功能使能。

当加速度计自检功能激活时，传感器的输出由作用在传感器上的加速度和静电测试力的代数和给出。

完整的加速度计自检过程如图 25. 加速度计自检步骤中所示。

图 25. 加速度计自检步骤

注：自检过程中使器件保持静止



根据保密协议！不可复制



9.2 陀螺仪自检

陀螺仪自检可以测试陀螺仪传感器的机械和电气部件。当此功能激活时，驱动力施加到传感器上，模拟一定的 Coriolis 力，并利用该静电测试力移动质量块。这种情况下，传感器输出会显示出一个输出变化。

当 CTRL5_C 寄存器的 ST[1:0]_G 位被设定为 00b 时，陀螺仪自检功能关闭；当 ST[1:0]_G 位被置为 01b（正符号自检）或 11b（负符号自检）时，该功能使能。

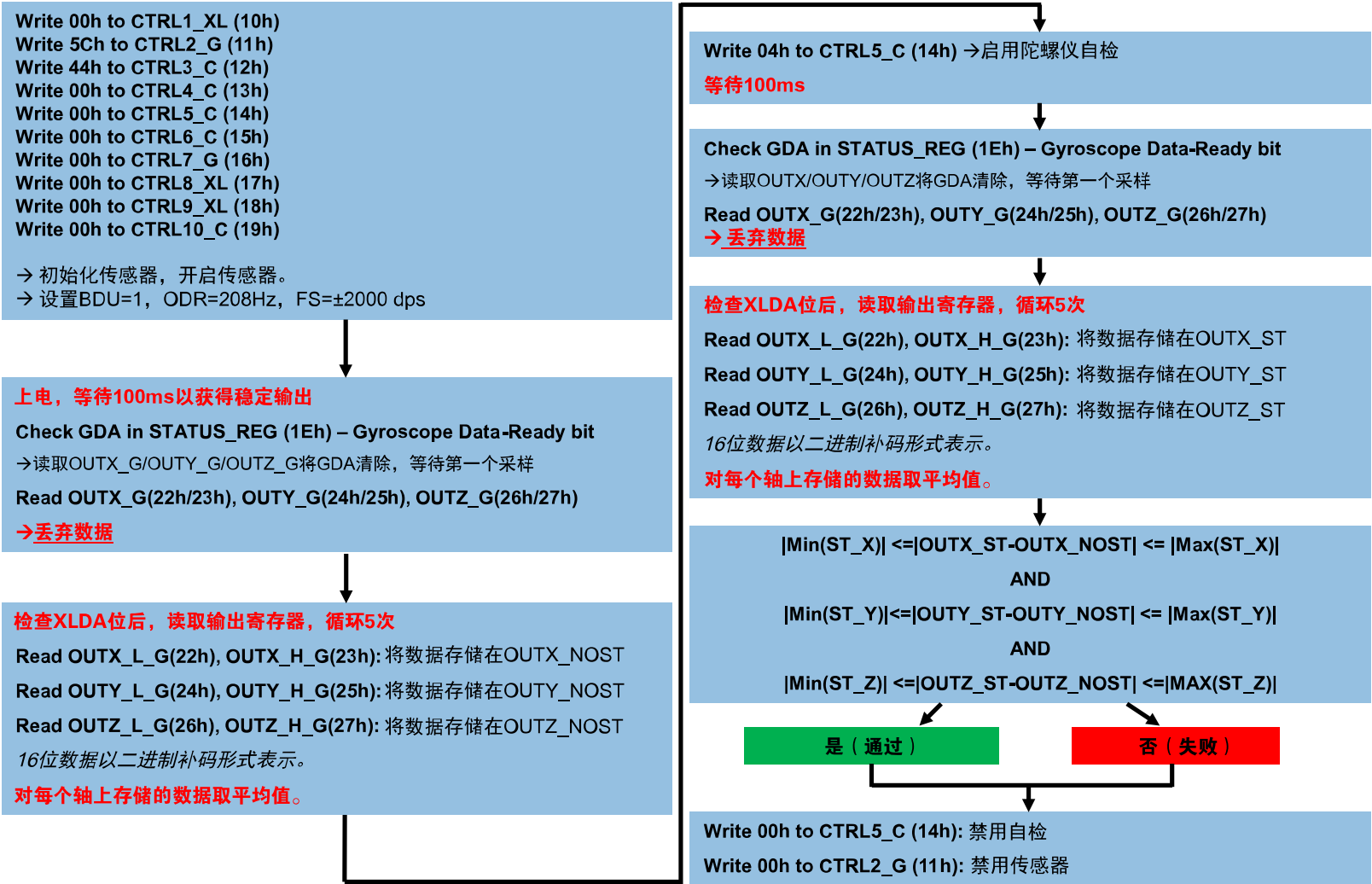
当陀螺仪自检功能激活时，传感器的输出由作用在传感器上的角速度和静电测试力的代数和给出。

完整的陀螺仪自检过程如图 26. 陀螺仪自检步骤中所示。

图 26. 陀螺仪自检步骤

注：自检过程中使器件保持静止

陀螺仪自检



根据保密协议，不可复制



版本历史

表 47. 文档版本历史

日期	版本	变更
2019 年 3 月 13 日	1	初始版本

目录

1	引脚说明.....	2
2	寄存器.....	4
3	工作模式.....	7
3.1	省电模式	8
3.2	高性能模式	8
3.3	陀螺仪睡眠模式	9
3.4	加速度计带宽	9
3.4.1	加速度计斜率滤波器.....	12
3.5	加速度计开启/关断时间	14
3.6	陀螺仪带宽	15
3.7	陀螺仪开启/关闭时间	20
4	读取输出数据	21
4.1	启动序列	21
4.2	使用状态寄存器	21
4.3	使用数据准备就绪信号.....	22
4.3.1	DRDY 屏蔽功能.....	24
4.4	使用块数据更新 (block data update, BDU) 功能	24
4.5	认识输出数据	24
4.5.1	输出数据示例	24
4.6	加速度计偏移寄存器.....	26
4.7	环行功能	26
4.7.1	FIFO 输出寄存器环行	26
4.7.2	传感器输出寄存器环行	26
4.8	DEN (数据使能)	27
4.8.1	边沿感应触发模式	28
4.8.2	电平感应触发模式	30
4.8.3	电平感应锁存模式	31
4.8.4	电平感应 FIFO 使能模式.....	32
4.8.5	用于 DEN 冲压的 LSB 选择	32

5	中断生成.....	33
5.1	中断引脚配置	33
5.2	自由落体中断	36
5.3	唤醒中断	39
5.4	6D/4D 定向检测.....	42
5.4.1	6D 定向检测	42
5.4.2	4D 方向检测	44
5.5	活动/不活动和运动/静止识别.....	44
5.5.1	静止/运动检测	47
5.5.2	启动状态.....	48
6	时间戳.....	49
7	先进先出缓冲区	50
7.1	FIFO 说明和批处理传感器	51
7.2	FIFO 寄存器.....	52
7.2.1	FIFO_CTRL1	52
7.2.2	FIFO_CTRL2	52
7.2.3	FIFO_CTRL3	53
7.2.4	FIFO_CTRL4	53
7.2.5	COUNTER_BDR_REG1.....	55
7.2.6	COUNTER_BDR_REG2.....	55
7.2.7	FIFO_STATUS1	55
7.2.8	FIFO_STATUS2	56
7.2.9	FIFO_DATA_OUT_TAG	57
7.2.10	FIFO_DATA_OUT.....	57
7.3	FIFO 批处理传感器	57
7.4	主要传感器	58
7.5	辅助传感器	61
7.6	FIFO 模式.....	62
7.6.1	Bypass 模式	62
7.6.2	FIFO 模式	63
7.6.3	Continue 模式	65



7.6.4	Continue-FIFO 模式	67
7.6.5	Bypass-Continue 模式	70
7.6.6	Bypass-FIFO 模式	72
7.6.7	从 FIFO 读取数据	75
7.7	FIFO 深度阈值	76
7.8	时间戳相关性	82
8	温度传感器	83
8.1	温度数据计算示例	83
9	自检功能	84
9.1	加速度计自检	84
9.2	陀螺仪自检	86
	版本历史	88
	目录	89
	表一览	92
	图一览	93



表一览

表 1.	内部引脚状态	2
表 2.	寄存器	4
表 3.	加速度计 ODR 选择	7
表 4.	陀螺仪 ODR 选择	8
表 5.	Vdd = 3.0 V 时的功耗 (典型值)	8
表 6.	加速度计带宽选择	11
表 7.	加速度计开启/关闭时间 (LPF2 和 HP 禁用)	14
表 8.	要丢弃的加速度计样本	14
表 9.	陀螺仪数字 HP 滤波器截止频率选择	17
表 10.	陀螺仪总带宽选择	17
表 11.	陀螺仪开启/关闭时间 (HP 禁用)	20
表 12.	要丢弃的陀螺仪样本 (LPF1 禁用)	20
表 13.	陀螺仪链设置时间 (LPF1 使能)	20
表 14.	输出数据寄存器内容 vs. 加速度 (FS_XL = $\pm 2 g$)	24
表 15.	输出数据寄存器内容 vs. 角速度 (FS_G = $\pm 250 dps$)	25
表 16.	输出寄存器环行模式	26
表 17.	DEN 配置	27
表 18.	INT1_CTRL 寄存器	33
表 19.	MD1_CFG 寄存器	33
表 20.	INT2_CTRL 寄存器	34
表 21.	MD2_CFG 寄存器	34
表 22.	自由落体阈值 LSB 值	38
表 23.	D6D_SRC 寄存器	42
表 24.	4D/6D 功能阈值	42
表 25.	6D 定位下的 D6D_SRC 寄存器	44
表 26.	不活动事件配置	44
表 27.	ODR _{coeff} 值	49
表 28.	FIFO_CTRL1 寄存器	52
表 29.	FIFO_CTRL2 寄存器	52
表 30.	FIFO_CTRL3 寄存器	53
表 31.	加速度计批处理数据率	53
表 32.	陀螺仪批处理数据率	53
表 33.	时间戳批处理数据率	54
表 34.	温度传感器批处理数据率	54
表 35.	FIFO_CTRL4 寄存器	54
表 36.	COUNTER_BDR_REG1 寄存器	55
表 37.	COUNTER_BDR_REG2 寄存器	55
表 38.	FIFO_STATUS1 寄存器	55
表 39.	FIFO_STATUS2 寄存器	56
表 40.	FIFO_DATA_OUT_TAG 寄存器	57
表 41.	TAG_SENSOR 字段和相关传感器	57
表 42.	FIFO 中的主要传感器输出数据格式	60
表 43.	FIFO 中的温度输出数据格式	61
表 44.	FIFO 中的时间戳输出数据格式	61
表 45.	FIFO 中的 CFG-change 输出数据格式	62
表 46.	输出数据寄存器内容 vs. 温度	83
表 47.	文档版本历史	88

图一览

图 1.	引脚连接	2
图 2.	加速度计滤波链	10
图 3.	加速度计斜率滤波器	13
图 4.	陀螺仪数字链	16
图 5.	数据准备就绪信号	23
图 6.	边沿感应触发模式, DEN 低电平有效	28
图 7.	电平感应触发模式, DEN 低电平有效	30
图 8.	电平感应触发模式, DEN 低电平有效, DEN_DRDY 在 INT1 上	30
图 9.	电平感应锁存模式, DEN 低电平有效	31
图 10.	电平感应锁存模式, DEN 低电平有效, DEN_DRDY 在 INT1 上	31
图 11.	电平感应 FIFO 使能模式, DEN 低电平有效	32
图 12.	自由落体中断	37
图 13.	唤醒中断 (利用斜率滤波器)	40
图 14.	6D 识别方向	43
图 15.	活动/不活动识别 (利用斜率滤波器)	46
图 16.	主要传感器和时序的定义	59
图 17.	FIFO 模式 (STOP_ON_WTM = 0)	63
图 18.	Continue 模式	66
图 19.	Continue-FIFO 模式	68
图 20.	Bypass-Continue 模式	71
图 21.	Bypass-FIFO 模式	74
图 22.	FIFO 阈值 (STOP_ON_WTM = 0)	77
图 23.	FIFO 模式下的 FIFO 阈值 (STOP_ON_WTM = 1)	79
图 24.	Continue 模式下的 FIFO 阈值 (STOP_ON_WTM = 1)	81
图 25.	加速度计自检步骤	85
图 26.	陀螺仪自检步骤	87



重要通知 - 请仔细阅读

意法半导体公司及其子公司（“ST”）保留随时对 ST 产品和/或本文档进行变更、更正、增强、修改和改进的权利，恕不另行通知。买方在订货之前应获取关于 ST 产品的最新信息。ST 产品的销售依照订单确认时的相关 ST 销售条款。

买方自行负责对 ST 产品的选择和使用，ST 概不承担与应用协助或买方产品设计相关的任何责任。

ST 不对任何知识产权进行任何明示或默示的授权或许可。

转售的 ST 产品如有不同于此处提供的信息的规定，将导致 ST 针对该产品授予的任何保证失效。

ST 和 ST 徽标是 ST 的商标。所有其他产品或服务名称均为其各自所有者的财产。

本文档中的信息取代本文档所有早期版本中提供的信息。

© 2019 STMicroelectronics - 保留所有权利