

LSM303AH：
超紧凑、高性能eCompass模块

引言

本文档旨在提供ST LSM303AH eCompass六轴惯性传感器模块相关的使用信息和应用提示。

LSM303AH是系统级封装的3D数字磁力计和3D数字加速度计，具有数字I²C和3线SPI接口标准输出。由于磁力计和加速度计均具有超低噪声性能，始终具有低功耗特性，并结合了高传感精度，因此能够为客户提供最佳运动体验。器件具有超低功耗工作模式，可实现高级节能、智能睡眠唤醒以及恢复睡眠功能。

该器件的磁场动态范围高达±50高斯，其用户可选择的满量程加速度范围为±2g/±4g/±8g/±16g。

可以对LSM303AH进行配置，使其产生用于磁场检测的中断信号，并由较高的应用层自动提供硬磁补偿。

经过配置，LSM303AH可利用硬件识别出的自由落体事件、6D方向、单击和双击感应、活动或不活动、唤醒事件，来产生中断信号。

LSM303AH可兼容主要操作系统的要求，提供真实和虚拟传感器。它在硬件中进行了设计，可实现大幅运动检测、倾斜检测和计步功能。

LSM303AH内置256级先进先出（FIFO）缓冲器提供给用户存储数据，可以减少主控的干预。

LSM303AH采用纤薄的小型塑料焊盘栅格阵列封装(LGA)，可确保在更大的温度范围（-40 °C至+85 °C）内正常工作。

SMD封装的超小尺寸和重量使其成为手持便携式应用的理想选择，如智能手机、物联网（IoT）连接设备，穿戴，以及需要减小封装尺寸和重量的其他应用。

目录

1	引脚说明	7
1.1	INT_MAG/DRDY引脚配置	9
2	寄存器	10
2.1	高级配置寄存器	13
3	磁力计操作模式	14
3.1	空闲模式	14
3.2	连续模式	15
3.3	单次模式	15
4	磁力计功耗模式	16
5	磁力计低通滤波器	17
6	读取磁力计输出数据	18
6.1	启动配置	18
6.2	使用状态寄存器	18
6.3	使用数据准备就绪信号	19
6.4	使用块数据更新 (block data update, BDU) 功能	19
6.5	认识输出数据	19
6.5.1	输出数据示例	20
6.5.2	大小端序选择	20
7	磁力计重启和软件复位	21
8	磁力计偏移消除	22
9	磁力计硬磁补偿	23
10	磁力计中断产生	24
10.1	中断配置示例	25
10.2	溢出中断	25

11	磁力计自检	26
12	加速度计操作模式	28
12.1	掉电	29
12.2	高分辨率/高频模式	29
12.3	低功耗模式	29
12.4	加速度计带宽	30
12.4.1	加速度计斜率滤波器	31
13	读取加速度计输出数据	32
13.1	启动配置	32
13.2	使用状态寄存器	32
13.3	使用数据准备就绪信号	33
13.4	使用块数据更新 (block data update, BDU) 功能	33
13.5	认识输出数据	34
13.5.1	输出数据示例	34
14	加速度计中断生成和嵌入功能	35
14.1	中断引脚配置	36
14.2	事件状态	37
14.3	自由落体中断	37
14.4	唤醒中断	39
14.5	6D/4D方向检测	41
14.5.1	6D方向检测	41
14.5.2	4D方向检测	43
14.6	单击和双击识别	43
14.6.1	单击	44
14.6.2	双击	45
14.6.3	单击和双击识别配置	46
14.6.4	单击示例	48
14.6.5	双击示例	48
14.7	活动/不活动识别	49
14.8	启动状态	51
14.9	嵌入功能	51
14.9.1	计步功能: 步伐侦测和步数计算	51

14.9.2	大幅运动检测	54
14.9.3	倾斜度检测	56
85	加速度计先进先出 (FIFO) 缓冲器	57
15.1	FIFO说明	57
15.2	FIFO寄存器	59
15.2.1	FIFO_CTRL_A寄存器 (25h)	59
15.2.2	FIFO_THS_A寄存器 (2Eh)	59
15.2.3	FIFO_SRC_A (2Fh)	60
15.2.4	FIFO_SAMPLES_A (30h)	61
15.3	FIFO中断	61
15.3.1	FIFO阈值	61
15.3.2	FIFO全满	61
15.4	Module-to-FIFO	62
15.5	FIFO模式	62
15.5.1	Bypass模式	62
15.5.2	FIFO模式	62
15.5.3	连续模式	64
15.5.4	连续-FIFO模式	65
15.5.5	Bypass-to-Continuous模式	66
15.6	从FIFO中取数据	67
16	加速度计自检	68
17	温度传感器	70
17.1	温度数据计算示例	70
18	版本历史	71

表格索引

表 1.	引脚说明	8
表 2.	INT_MAG/DRDY引脚配置	9
表 3.	寄存器	10
表 4.	高级配置寄存器	13
表 5.	工作模式	14
表 6.	输出数据速率配置	15
表 7.	单模下的最大ODR	15
表 8.	电流消耗	16
表 9.	工作模式和开启时间	16
表 10.	工作模式的RMS噪声	17
表 11.	加速度计ODR和功耗模式选择	28
表 12.	功耗	29
表 13.	加速度计LPF1截止频率	30
表 14.	CTRL4_A寄存器	36
表 15.	CTRL5_A寄存器	36
表 16.	自由落体阈值LSB值	38
表 17.	6D_SRC_A寄存器	41
表 18.	4D/6D功能阈值	41
表 19.	6D定位中的6D_SRC_A寄存器	43
表 20.	TAP_SRC_A寄存器	47
表 21.	SMD阈值	55
表 22.	FIFO缓冲区完整表示（存储了第256个样本集合）	57
表 23.	FIFO缓冲区完整表示（存储了第257个样本集合、丢弃了第1个样本）	58
表 24.	FIFO_CTRL_A寄存器	59
表 25.	FIFO_THS_A寄存器	59
表 26.	FIFO_SRC_A寄存器	60
表 27.	FIFO_SRC_A特性（假定FTH[7:0] = 15）	60
表 28.	FIFO_SAMPLES_A寄存器	61
表 29.	示例：ODR阈值功能	67
表 30.	输出数据寄存器内容 vs. 温度	70
表 31.	文档版本历史	71
表 32.	中文文档版本历史	71

图片目录

图1.	引脚连接	7
图2.	磁力计滤波链	17
图3.	中断功能	24
图4.	磁力计自检步骤	27
图5.	加速度计采样链	30
图6.	加速度计斜率滤波器	31
图7.	数据准备就绪信号	33
图8.	自由落体中断	37
图9.	唤醒中断	40
图10.	6D识别方向	42
图11.	单击事件识别	44
图12.	双击事件识别 (LIR 位 = 0)	45
图13.	单击和双击事件识别 (LIR 位 = 0)	47
图14.	活动/不活动识别	50
图15.	最小阈值	53
图16.	FIFO模式特性	63
图17.	带中断触发的Continuous模式	64
图18.	Continuous-to-FIFO模式：中断锁存和非锁存	65
图19.	旁路-连续模式	66
图20.	加速度计自检步骤	69

1 引脚说明

图1. 引脚连接

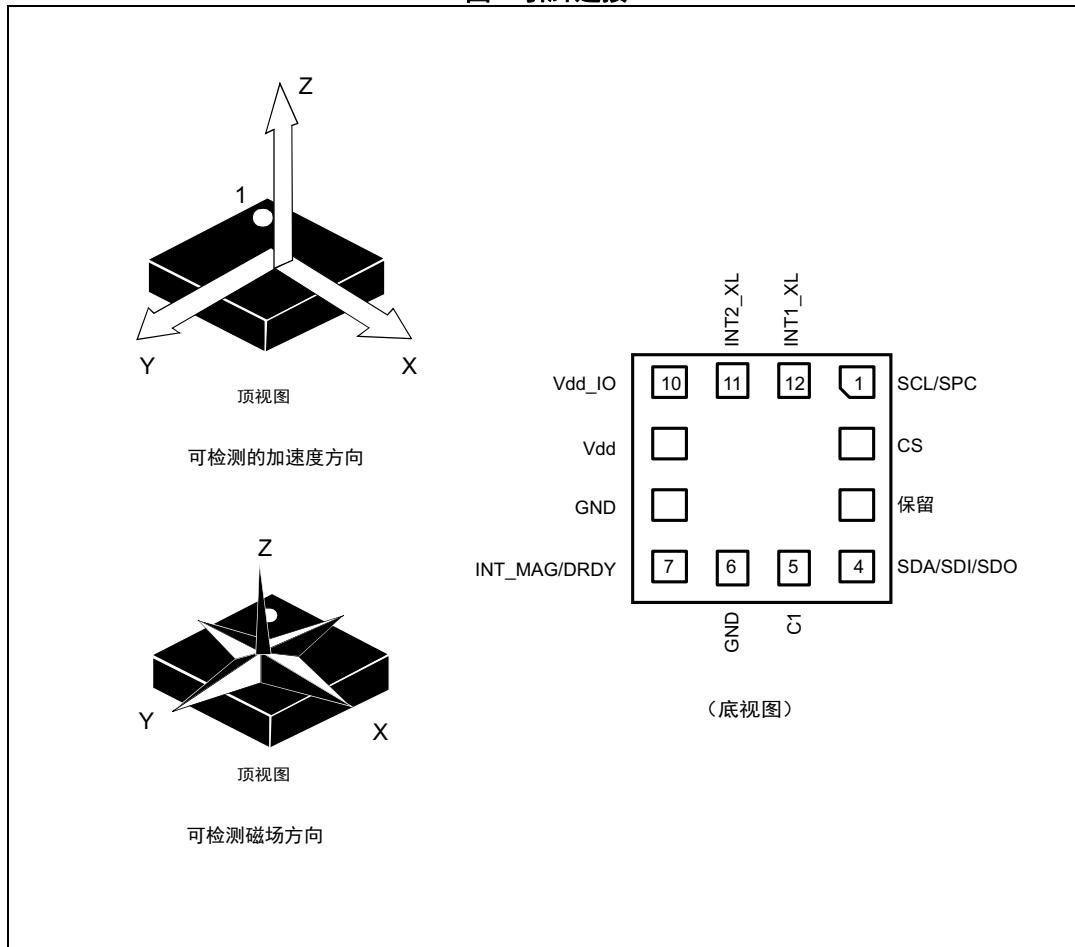


表1. 引脚说明

引脚#	名称	功能	引脚状态
1	SCL SPC	I ² C串行时钟 (SCL) SPI串口时钟 (serial port clock, SPC)	默认值: 无上拉的输入
2	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: 无上拉的输入
3	Reserved	与GND连接	
4	SDA SDI SDO	I ² C串行数据 (serial data, SDA) SPI串行数据输入 (serial data input, SDI) 3线接口串行数据输出 (serial data output, SDO)	默认值: (SDA) 无上拉的输入
5	C1	电容连接 (C1 = 220 nF)	外部电容, 器件所施加的电压
6	GND	0 V电源	
7	INT_MAG/ DRDY	磁力计中断/数据准备就绪信号	默认: 输出高阻抗
8	GND	0 V电源	
9	VDD	电源	
10	Vdd_IO	I/O引脚的供电	
11	INT2_XL	磁力计中断引脚配2	默认值: 推挽输出强制接地
12	INT1_XL	磁力计中断引脚配1	默认值: 推挽输出强制接地

1.1 INT_MAG/DRDY引脚配置

INT_MAG/DRDY引脚可配置为具有一个HW信号，以确定新的一组测量数据何时可以读取或中断事件何时发生。

通过将CFG_REG_C_M寄存器中的INT_MAG位置为1，可以将STATUS_REG寄存器中的Zyxda数据就绪信号驱动到INT/DRDY引脚（更多详情请参见[第 6.3节：使用数据准备就绪信号](#)）。

通过将CFG_REG_C_M寄存器中的INT_MAG_PIN位置为1，可以将INT_SOURCE_REG_M寄存器中的INT信号驱动到INT_MAG/DRDY引脚（更多详情请参见[第 10节：磁力计中断产生](#)）。

注：CFG_REG_C_M中的INT_MAG和INT_MAG_PIN位都会将INT_MAG/DRDY引脚配置为数字输出，但只有一个信号（INT或DRDY）可被传送到INT_MAG/DRDY引脚上。如果两个位均置位，则只有INT信号会按照下表所示进行传送。

下表总结了CFG_REG_C_M寄存器的INT_MAG和INT_MAG_PIN位的不同配置下INT_MAG/DRDY引脚状态和功能。

表2. INT_MAG/DRDY引脚配置

INT_MAG	INT_MAG_PIN	INT_MAG/DRDY 引脚状态	INT_MAG/DRDY 引脚功能
0	0	高阻抗输出	无
1	0	推挽输出	DRDY (Zyxda) 信号传送
0	1	推挽输出	INT信号传送
1	1	推挽输出	INT信号传送



2

寄存器

表3. 寄存器

寄存器名	地址	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Module_8bit_A	0Ch	Module_7	Module_6	Module_5	Module_4	Module_3	Module_2	Module_1	Module_0
WHO_AM_I_A	0Fh	0	1	0	0	0	0	1	1
CTRL1_A	20h	ODR3	ODR2	ODR1	ODR0	FS1	FS0	HF_ODR	BDU
CTRL2_A	21h	BOOT	SOFT_RESET	0	FUNC_CFG_EN	FDS_SLOPE	IF_ADD_INC	I2C_DISABLE	SPI_ENABLE
CTRL3_A	22h	ST2	ST1	TAP_X_EN	TAP_Y_EN	TAP_Z_EN	LIR	H_LACTIVE	PP_OD
CTRL4_A	23h	0	INT1_S_TAP	INT1_WU	INT1_FF	INT1_TAP	INT1_6D	INT1_FTH	INT1_DRDY
CTRL5_A	24h	DRDY_PULSED	INT2_BOOT	INT2_ON_INT1	INT2_TILT	INT2_SIG_MOT	INT2_STEP	INT2_FTH	INT2_DRDY
FIFO_CTRL_A	25h	FMODE2	FMODE1	FMODE0	INT_STEP_COUNT_OV	MODULE_TO_FIFO	0	0	IF_CS_PU_DIS
OUT_T_A	26h	TEMP7	TEMP6	TEMP5	TEMP4	TEMP3	TEMP2	TEMP1	TEMP0
STATUS_A	27h	FIFO_THS	WU_IA	SLEEP_STATE	DOUBLE_TAP	SINGLE_TAP	6D_IA	FF_IA	DRDY
OUTX_L_A	28h	X_L7	X_L6	X_L5	X_L4	X_L3	X_L2	0	0
OUTX_H_A	29h	X_H7	X_H6	X_H5	X_H4	X_H3	X_H2	X_H1	X_H0
OUTY_L_A	2Ah	Y_L7	Y_L6	Y_L5	Y_L4	Y_L3	Y_L2	0	0
OUTY_H_A	2Bh	Y_H7	Y_H6	Y_H5	Y_H4	Y_H3	Y_H2	Y_H1	Y_H0
OUTZ_L_A	2Ch	Z_L7	Z_L6	Z_L5	Z_L4	Z_L3	Z_L2	0	0
OUTZ_H_A	2Dh	Z_H7	Z_H6	Z_H5	Z_H4	Z_H3	Z_H2	Z_H1	Z_H0
FIFO_THS_A	2Eh	FTH7	FTH6	FTH5	FTH4	FTH3	FTH2	FTH1	FTH0
FIFO_SRC_A	2Fh	FTH	FIFO_OVR	DIFF8	0	0	0	0	0
FIFO_SAMPLES_A	30h	DIFF7	DIFF6	DIFF5	DIFF4	DIFF3	DIFF2	DIFF1	DIFF0
TAP_6D_THS_A	31h	4D_EN	6D_THS1	6D_THS0	TAP_THS4	TAP_THS3	TAP_THS2	TAP_THS1	TAP_THS0

表3. 寄存器

寄存器名	地址	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
INT_DUR_A	32h	LAT3	LAT2	LAT1	LAT0	QUIET1	QUIET0	SHOCK1	SHOCK0
WAKE_UP_THS_A	33h	SINGLE_DOUBLE_TAP	SLEEP_ON	WU_THS5	WU_THS4	WU_THS3	WU_THS2	WU_THS1	WU_THS0
WAKE_UP_DUR_A	34h	FF_DUR5	WU_DUR1	WU_DUR0	INT1_FSS7	SLEEP_DUR3	SLEEP_DUR2	SLEEP_DUR1	SLEEP_DUR0
FREE_FALL_A	35h	FF_DUR4	FF_DUR3	FF_DUR2	FF_DUR1	FF_DUR0	FF_THS2	FF_THS1	FF_THS0
STATUS_DUP_A	36h	OVR	WU_IA	SLEEP_STATE	DOUBLE_TAP	SINGLE_TAP	6D_IA	FF_IA	DRDY
WAKE_UP_SRC_A	37h	0	0	FF_IA	SLEEP_STATE_IA	WU_IA	X_WU	Y_WU	Z_WU
TAP_SRC_A	38h	0	TAP_IA	SINGLE_TAP	DOUBLE_TAP	TAP_SIGN	X_TAP	Y_TAP	Z_TAP
6D_SRC_A	39h	0	6D_IA	ZH	ZL	YH	YL	XH	XL
STEP_COUNTER_MINTHS_A	3Ah	RST_NSTEP	PEDO4g	SC_MTHS5	SC_MTHS4	SC_MTHS3	SC_MTHS2	SC_MTHS1	SC_MTHS0
STEP_COUNTER_L_A	3Bh	nSTEP_L7	nSTEP_L6	nSTEP_L5	nSTEP_L4	nSTEP_L3	nSTEP_L2	nSTEP_L1	nSTEP_L0
STEP_COUNTER_H_A	3Ch	nSTEP_H7	nSTEP_H6	nSTEP_H5	nSTEP_H4	nSTEP_H3	nSTEP_H2	nSTEP_H1	nSTEP_H0
FUNC_CK_GATE_A	3Dh	TIILT_INT	FS_SRC1	FS_SRC0	SIG_MOT_DETECT	RST_SIG_MOT	RST_PEDO	STEP_DETECT	CK_GATE_FUNC
FUNC_SRC_A	3Eh	0	0	0	0	0	RST_TIILT	MODULE_READY	0
FUNC_CTRL_A	3Fh	0	0	MODULE_开启	TIILT_ON	0	0	SIG_MOT_开启	STEP_CNT_ON
OFFSET_X_REG_L_M	45h	Offset_X_7	Offset_X_6	Offset_X_5	Offset_X_4	Offset_X_3	Offset_X_2	Offset_X_1	Offset_X_0
OFFSET_X_REG_H_M	46h	Offset_X_15	Offset_X_14	Offset_X_13	Offset_X_12	Offset_X_11	Offset_X_10	Offset_X_9	Offset_X_8

表3. 寄存器

寄存器名	地址	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
OFFSET_Y_REG_L_M	47h	Offset_Y_7	Offset_Y_6	Offset_Y_5	Offset_Y_4	Offset_Y_3	Offset_Y_2	Offset_Y_1	Offset_Y_0
OFFSET_Y_REG_H_M	48h	Offset_Y_15	Offset_Y_14	Offset_Y_13	Offset_Y_12	Offset_Y_11	Offset_Y_10	Offset_Y_9	Offset_Y_8
OFFSET_Z_REG_L_M	49h	Offset_Z_7	Offset_Z_6	Offset_Z_5	Offset_Z_4	Offset_Z_3	Offset_Z_2	Offset_Z_1	Offset_Z_0
OFFSET_Z_REG_H_M	4Ah	Offset_Z_15	Offset_Z_14	Offset_Z_13	Offset_Z_12	Offset_Z_11	Offset_Z_10	Offset_Z_9	Offset_Z_8
WHO_AM_I_M	4Fh	0	1	0	0	0	0	0	0
CFG_REG_A_M	60h	COMP_TEMP_EN ⁽¹⁾	REBOOT	SOFT_RST	LP	ODR1	ODR0	MD1	MD0
CFG_REG_B_M	61h	0	0	0	OFF_CANC_ONE_SHOT	INT_on_DataOFF	Set_FREQ	OFF_CANC	LPF
CFG_REG_C_M	62h	0	INT_MAG_PIN	I2C_DIS	BDU	BLE	0	Self_test	INT_MAG
INT_CTRL_REG_M	63h	XIEN	YIEN	ZIEN	0	0	IEA	IEL	IEN
INT_SOURCE_REG_M	64h	P_TH_S_X	P_TH_S_Y	P_TH_S_Z	N_TH_S_X	N_TH_S_Y	N_TH_S_Z	MROI	INT
INT_THS_L_REG_M	65h	TH7	TH6	TH5	TH4	TH3	TH2	TH1	TH0
INT_THS_H_REG_M	66h	TH15	TH14	TH13	TH12	TH11	TH10	TH9	TH8
STATUS_REG_M	67h	Zyxor	zor	yor	xor	Zyxda	zda	yda	xda
OUTX_L_REG_M	68h	D7	D6	D5	D4	D3	D2	D1	D0
OUTX_H_REG_M	69h	D15	D14	D13	D12	D11	D10	D9	D8
OUTY_L_REG_M	6Ah	D7	D6	D5	D4	D3	D2	D1	D0
OUTY_H_REG_M	6Bh	D15	D14	D13	D12	D11	D10	D9	D8
OUTZ_L_REG_M	6Ch	D7	D6	D5	D4	D3	D2	D1	D0
OUTZ_H_REG_M	6Dh	D15	D14	D13	D12	D11	D10	D9	D8

1. 为了正确操作磁力计传感器, 用户必须将CFG_REG_A_M寄存器中的COMP_TEMP_EN位设置为1。



2.1 高级配置寄存器

高级配置寄存器用于配置特定的器件功能，如计步器和传感器集线器。要切换到高级配置寄存器页面，必须在CTRL2_A (21h) 中将FUNC_CFG_EN位置为“1”。要返回标准寄存器页面，必须在寄存器CTRL2_A (3Fh)中将FUNC_CFG_EN位置为“0”。

注：所有对高级配置寄存器内容的修改必须在加速度计传感器处于掉电模式时进行。

表4. 高级配置寄存器

寄存器名	地址	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PEDO_DEB_REG_A	2Bh	DEB_TIME4	DEB_TIME3	DEB_TIME2	DEB_TIME1	DEB_TIME0	DEB_STEP2	DEB_STEP1	DEB_STEP0
SM_THS_A	34h	SM_THS_7	SM_THS_6	SM_THS_5	SM_THS_4	SM_THS_3	SM_THS_2	SM_THS_1	SM_THS_0
STEP_COUNT_DELTA_A	3Ah	STEP_COUNT_D7	STEP_COUNT_D6	STEP_COUNT_D5	STEP_COUNT_D4	STEP_COUNT_D3	STEP_COUNT_D2	STEP_COUNT_D1	STEP_COUNT_D0
CTRL2_A	3Fh	BOOT ⁽¹⁾	SW_RESET ⁽¹⁾	0 ⁽¹⁾	FUNC_CFG_EN	FDS_SLOPE ⁽¹⁾	IF_ADD_INC ⁽¹⁾	I2C_DISABLE ⁽¹⁾	SIM ⁽¹⁾

1. 只读位

3 磁力计操作模式

LSM303AH磁力计具有三种工作模式：

- 空闲模式；
- 连续模式；
- 单次模式。

上电后，磁力计执行一段20 ms的启动程序来加载trimming参数。启动完成后，磁力计会自动配置为空闲模式。

为了避免潜在冲突，在上电时序期间，建议将连接到器件IO引脚的线路设置为主机侧的高阻抗状态。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100 μ s。

磁力计的工作模式可以通过CFG_REG_A的MD [1:0]位来设置，如下表所示。

表5. 工作模式

MD1	MD0	模式
0	0	连续模式
0	1	单模
1	0	空闲模式
1	1	空闲模式

在所有三种工作模式中，磁场动态范围的典型值为50高斯，这在磁场与某一感应轴完全对齐时适用。在横轴方向存在杂散场的情况下，磁场动态范围会降低到25高斯（在最坏的情况下）。

3.1 空闲模式

当磁力计处于空闲模式时，几乎所有的器件内部模块都会关闭，以最大限度地降低功耗。数字接口（I²C和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，可保持进入空闲模式前存储器中采样的最后数据。

3.2 连续模式

可通过将CFG_REG_A_M寄存器中的MD [1:0]位写入00，来使能连续模式。

在连续模式下，器件连续执行测量并将结果放入输出数据寄存器。通过配置CFG_REG_A_M中的LP位可以选择高分辨率模式或低功耗模式（请参考[第4节：磁力计功耗模式](#)）。

在连续模式下，可以使用CFG_REG_A_M寄存器的ODR [1:0]位来选择输出数据速率，如下表所示。

表6. 输出数据率配置

ODR1	ODR0	ODR (Hz)
0	0	10 (默认)
0	1	20
1	0	50
1	1	100

3.3 单次模式

LSM303AH磁力计提供了高分辨率和低功耗模式下的单模（请参考[第4节：磁力计功耗模式](#)）。

单模配置允许根据需要执行单次采集；通过在CFG_REG_A_M寄存器中将MD [1:0]位写入01，可触发采集。执行测量后，STATUS_REG_M寄存器的Zyxda、zda、yda、xda位被声明，输出寄存器中的数据可用，磁力计可通过将MD [1:0]位置为11来将自动配置为空闲模式。

单模与编程的ODR无关：它取决于微控制器/应用处理器写入MD [1:0]位的频率。单模下可达到的最大ODR频率如下表所示，严格取决于所选功耗模式（请参考[第4节：磁力计功耗模式](#)）。

表7. 单模下的最大ODR

功耗模式 (CFG_REG_A_M寄存器的LP位)	最大ODR [Hz]
高分辨率 (LP = 0)	100
低功耗 (LP = 1)	150

单模下，生成新数据所需的典型时间对应于[表9：工作模式和开启时间](#)中所示的开启时间。

4 磁力计功耗模式

LSM303AH磁力计提供了两种功耗模式，器件配置为连续或单模工作：

- 高分辨率模式；
- 低功耗模式。

通过配置CFG_REG_A_M寄存器的LP位，可以选择功耗模式：如果LP位被声明，则器件工作于低功耗模式，否则工作在高分辨率模式（默认配置）。

在低功耗模式和高分辨率模式下，磁力计电路周期性地开启/关闭，其占空比是所选ODR的函数，并且数据中断生成激活。

区分这两种模式的区别在于用来生成每个输出样本的样本数量不同，在低功耗模式下比在高分辨率模式中使用的样本数量少了四倍，从而确保了较低的功耗。

下表总结了偏移消除禁用/使能下两种功耗模式的电流消耗（传感器偏移消除功能可以使用CFG_REG_B_M寄存器的OFF_CANC位进行配置，请参考[第8节：磁力计偏移消除](#)）。

表8. 电流消耗

ODR [Hz]	电流消耗 (LP = 0并且 OFF_CANC = 0) [uA]	电流消耗 (LP = 1并且 OFF_CANC = 0) [uA]	电流消耗 (LP = 0并且 OFF_CANC = 1) [uA]	电流消耗 (LP = 1并且 OFF_CANC = 1) [uA]
10	100	25	120	50
20	200	50	235	100
50	475	125	575	235
100	950	250	1130	460

下表总结了两种不同功耗模式下器件的开启时间，其中偏移消除功能启用或禁用（请参阅[第8节：磁力计偏移消除](#)）。

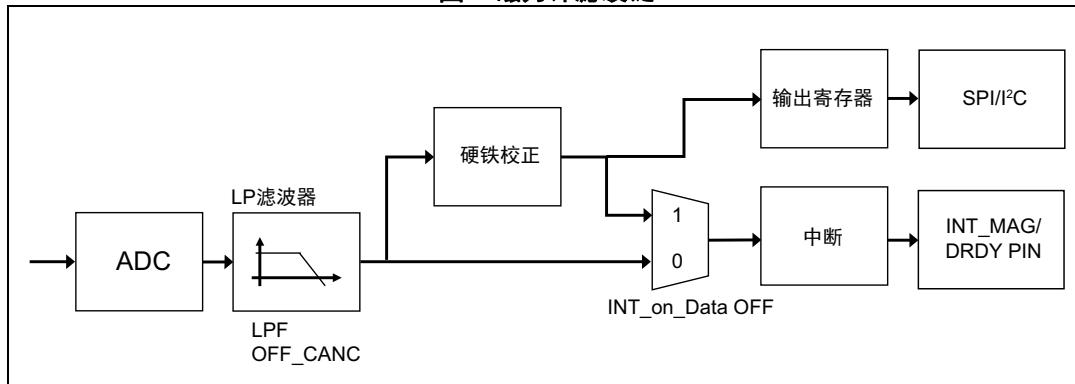
表9. 工作模式和开启时间

工作模式	开启时间	
	CFG_REG_A_M[LP]	CFG_REG_A_M[OFF_CANC = 0]
0（高分辨率）		9.4 ms
1（低功耗）		6.4 ms

5 磁力计低通滤波器

LSM303AH磁力计嵌入了数字低通滤波器以降低噪声。滤波器可以通过设置CFG_REG_B_M中的LPF位来使能。

图2. 磁力计滤波链



下表总结了不同器件配置下的带宽和RMS噪声值。

当低通滤波器使能时，带宽减小，同时噪声性能会得到改善，并且功耗不增加。

注：
偏移消除功能（请参考第8节：磁力计偏移消除）用作两级移动平均滤波器，因此具有与所用LPF滤波器相同的带宽和噪声性能。

表10. 工作模式的RMS噪声

LPF或OFF_CANC	LP = 0		LP = 1	
	BW [Hz]	噪声RMS [mg]	BW [Hz]	噪声RMS [mg]
0 (禁用)	ODR/2	4.5	ODR/2	9
1 (启用)	ODR/4	3	ODR/4	6

6 读取磁力计输出数据

6.1 启动配置

当器件上电时，磁力计会自动从嵌入的内存中加载校准系数到内部寄存器中。启动程序完成后（花费大约20 ms），磁力计会自动进入空闲模式。

要启用磁力计并采集磁力数据，需要通过CFG_REG_A_M寄存器选择某一种工作模式。可使用下列通用序列对磁力计进行配置：

- | | |
|------------------------|--|
| 1. 写入CFG_REG_A_M = 80h | // 温度补偿已启用
// ODR = 10 Hz
// 连续和高分辨率模式 |
| 2. 写入CFG_REG_A_M = 01h | // 将INT_MAG/DRDY引脚配置为数字输出，并发送数据就绪信号 |

在CFG_REG_A_M中写入81h而不是80h，这会使磁力计工作于单模而不是连续模式。

6.2 使用状态寄存器

该器件具有一个STATUS_REG_M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（Zyxda = 1）。

应当按照如下步骤进行读取：

1. 读取STATUS_REG_M
2. 如果Zyxda = 0，则进入1
3. 读取OUTX_L_REG_M
4. 读取OUTX_H_REG_M
5. 读取OUTY_L_REG_M
6. 读取OUTY_H_REG_M
7. 读取OUTZ_L_REG_M
8. 读取OUTZ_H_REG_M
9. 数据处理
10. 进入1

如果磁力计配置为单模而不是连续模式，那么执行一次后，程序将在步骤1停滞，因为器件执行单次测量，将Zyxda位设置为高电平，并返回空闲模式。请注意，MD位会返回空闲模式值。可以通过将MD位置为01来触发另一次读取。

6.3 使用数据准备就绪信号

磁力计可配置为具有一个HW信号，以确定新的一组测量数据何时可以读取。

数据就绪信号 (DRDY) 由STATUS_REG_M寄存器的Zyxda位表示。通过将STATUS_REG_M寄存器的INT_MAG位置为1，可将此信号驱动至INT_MAG/DRDY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。当其中一个通道的较高部分被读取 (OUTX_H_REG_M, OUTY_H_REG_M和OUTZ_H_REG_M寄存器) 时，信号会复位。

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

如果读取磁力计数据特别慢，并且不能（或者不需要）与STATUS_REG_M寄存器中的Zyxda事件位或驱动到INT/DRDY引脚的数据准备就绪信号同步，那么强烈建议将CFG_REG_C_M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采样相关的值（输出数据的最高有效部分和最低有效部分）。特别是在BDU被激活的情况下，与每条通道相关联的数据寄存器始终会包含由器件生成的最新输出数据，但如果发起了对给定寄存器对（即OUTX_H_REG_M和OUTX_L_REG_M、OUTY_H_REG_M和OUTY_L_REG_M、OUTZ_H_REG_M和OUTZ_L_REG_M）的读取，读取数据的MSB和LSB部分之前，都会禁止刷新该寄存器对。

注： 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。

6.5 认识输出数据

测得的磁力数据被发送到OUTX_H_REG_M、OUTX_L_REG_M、OUTY_H_REG_M、OUTY_L_REG_M、OUTZ_H_REG_M和OUTZ_L_REG_M寄存器。这些寄存器分别容纳磁力信号在X、Y和Z轴上的最高有效部分和最低有效部分。

X、Y、Z通道的完整输出数据是由OUTX_H_REG_M & OUTX_L_REG_M、OUTY_H_REG_M&OUTY_L_REG_M、OUTZ_H_REG_M&OUTZ_L_REG_M共同提供的，表示为2的补码。

磁力数据表示为16比特数字，称为LSB。它必须乘以适当的灵敏度参数， $M_{So} = 1.5 \text{ mG} / \text{LSB}$ ，以获得单位为mG的相应值。

6.5.1 输出数据示例

以下是一个简单的示例，说明如何使用LSB数据并将其转换成mG。

从传感器获取原始数据：

OUTX_L_REG_M: 21h

OUTX_H_REG_M: 00h

OUTY_L_REG_M: 1Dh

OUTY_H_REG_M: FFh

OUTZ_L_REG_M: CBh

OUTZ_H_REG_M: FEh

将寄存器串联：

OUTX_H_REG_M & OUTX_L_REG_M: 0021h

OUTY_H_REG_M & OUTY_L_REG_M: FF1Dh

OUTZ_H_REG_M & OUTZ_L_REG_M: FECBh

计算带符号的十进制值（二进制补码格式）：

X: +33

Y: -227

Z: -309

应用灵敏度：

X: +33 * 1.5 = +49.5 mG

Y: -227 * 1.5 = -340.5 mG

Z: -309 * 1.5 = -463.5 mG

6.5.2 大小端序选择

LSM303AH磁力计允许交换数据寄存器低位部分和高位部分的内容（即交换OUTX_H_REG_M与OUTX_L_REG_M的内容），以便符合小端和大端数据表示法的要求。

“小端序”意思是数字的低位字节存储在存储器的最低地址中，而高位字节存储在最高地址中。这种模式对应于CFG_REG_C_M寄存器的BLE位置为0（默认配置）。

相反，“大端序”意思是数字的高位字节存储在存储器的最低地址中，而低位字节存储在最高地址中。这种模式对应于CFG_REG_C_M寄存器的BLE位置为1。

7 磁力计重启和软件复位

器件上电后，LSM303AH磁力计执行一段20 ms的启动程序来加载trimming参数。启动完成后，磁力计会自动配置为空闲模式。

启动时间内，寄存器不可访问。

上电后，可通过将CFG_REG_A_M寄存器的REBOOT位置为1，来重载修调参数。

不需要切换器件电源线；重启完成后，器件在一次测量后都将进入空闲模式（不管所选工作模式是什么）。

如果需要复位至控制寄存器的默认值，可通过将CFG_REG_A_M寄存器的SOFT_RST位置为1来实现。软件复位过程会花费5 μ s；复位状态由CFG_REG_A_M寄存器SOFT_RST位的状态为信号：当复位完成时，该位自动置为低电平。

为了避免冲突，重启和软件复位不能同时执行（不要同时将CFG_REG_A_M寄存器的REBOOT位和SOFT_RST位同时置为1）。

必须按照以下示例中所示的顺序执行流程：

1. 将CFG_REG_A_M寄存器的SOFT_RST位置1；
2. 等待5 μ s（或等待至CFG_REG_A_M寄存器的SOFT_RST位返回0）；
3. 将CFG_REG_A_M寄存器的REBOOT位置1；
4. 等待20 ms。

8 磁力计偏移消除

LSM303AH磁力计基于AMR技术：需要一个置位脉冲来设置初始工作条件。

偏移消除是在磁力传感器中执行置位和复位脉冲的结果，可以启用此功能来移除固有传感器偏移。

偏移消除技术定义如下：

$$H_{out} = \frac{H_n + H_{n-1}}{2}$$

这里 H_n 和 H_{n-1} 是两个连续的磁场测量值，一个在置位脉冲之后，另一个在复位脉冲之后。

考虑磁力偏移（ H_{off} ），两个磁场测量值为：

- 置位： $H_n = H + H_{off}$
- 复位： $H_{n-1} = H - H_{off}$

根据偏移消除技术来消除偏移：

$$H_{out} = \frac{H_n + H_{n-1}}{2} = \frac{2H + H_{off} - H_{off}}{2} = H$$

如果器件工作于连续模式，则通过将CFG_REG_B_M中的OFF_CANC位置为1，可以启用偏移取消。这种情况下，置位/复位脉冲连续执行。对一个测量施加置位脉冲，对下一个测量施加复位脉冲。如果禁用偏移取消（OFF_CANC = 0）且选择了连续模式，则可以通过设置CFG_REG_B_M中的Set_FREQ位来配置置位脉冲频率。如果将Set_FREQ置为0，则置位脉冲每隔63个ODR释放一次，否则，如果将Set_FREQ置为1，则置位脉冲仅在从空闲模式上电时才会释放（即使偏移取消被禁用，也仍会执行一组磁传感器）。

如果磁力计工作于单模，那么要使能偏移取消，CFG_REG_B_M中的OFF_CANC和OFF_CANC_ONE_SHOT位必须置为1。使能这些位，脉冲极性在单次读取和下一次读取之间会发生反转。连续模式下偏移消除可由器件自动管理，如果此功能在单模下启用，则用户必须使用以下公式来手动消除偏移：

$$H_{out} = \frac{H_n + H_{n-1}}{2}$$

使用单次读取的偏移消除仅在读取时间接近时才有效，因此要确保偏移在两次连续读取之间不会漂移。

9 磁力计硬磁补偿

当磁性物体放置在磁力计附近时，会出现硬磁干扰，并在传感器输出中表现为固定的偏移。硬磁校准包含了对来硬磁干扰的磁力数据的补偿。

该操作定义如下：

$$H_{out} = H_{read} - H_{HI}$$

其中：

- H_{read} 为传感器读取的未经补偿的磁场数据；
- H_{HI} 是硬磁干扰的磁场；
- H_{out} 是补偿的磁数据。

硬磁的大小的计算应由外部处理器执行。在完成硬磁干扰的计算之后，可以补偿测得的磁数据。

该器件可以将硬铁数据存储在地址45h到4Ah的六个专用寄存器内。

每个寄存器都包含8位，因此，硬磁数据可以表示为16位二进制补码。OFFSET_X_REG_H_M、OFFSET_Y_REG_H_M和OFFSET_Z_REG_H_M寄存器应分别包含沿X、Y和Z轴估算的硬磁场的MSB。OFFSET_X_REG_L_M, OFFSET_Y_REG_L_M和OFFSET_Z_REG_L_M寄存器应分别包含沿X、Y和Z轴估算的硬磁场的LSB。硬磁数据与磁输出数据具有相同的格式和灵敏度。存储在专用寄存器中的硬磁值将自动从输出数据中减去。

10 磁力计中断产生

在LSM303AH中，磁力计中断信号的产生基于磁力计输出数据与可编程阈值之间的比较。

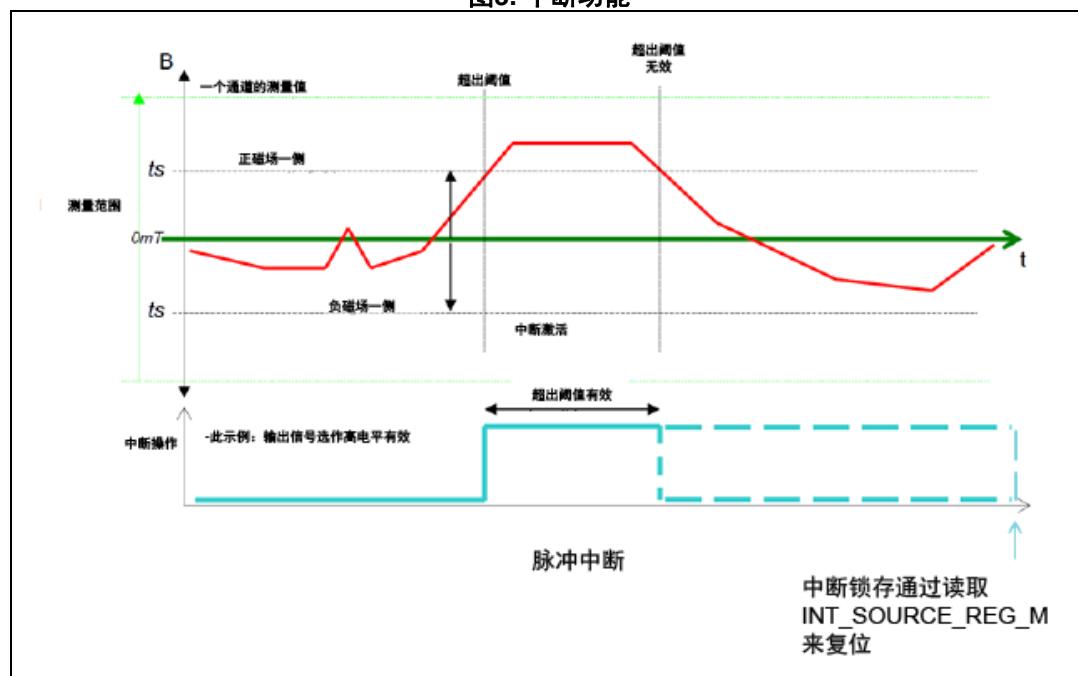
要使能中断功能，必须将INT_CTRL_REG_M中的IEN位置为1。可以在每个轴上选择性地启用中断功能。为此，需要正确设置INT_CTRL_REG_M中的XIEN、YEEN和ZIEN位。

阈值可以通过设置INT_THS_L_REG_M和INT_THS_H_REG_M寄存器进行编程。

阈值以绝对值表示为15位无符号数。阈值与磁数据具有相同的灵敏度。

当磁数据超过正或负阈值时，会产生一个中断信号，并且INT_SOURCE_REG_M寄存器中的INT位会变高。关于哪个轴触发了唤醒事件的信息也可在INT_SOURCE_REG_M寄存器中找到；特别是当磁数据超过正阈值时，P_TH_S_[X, Y, Z]位被置为1，而如果数据超过负阈值，则N_TH_S_[X, Y, Z]位被置为1。如果磁数据处于正阈值和负阈值之间，则不会产生中断信号。

图3. 中断功能



中断功能具有两种不同的方式：

- 典型：传感器读取的磁数据和可编程阈值之比；
- 高级：硬铁校正后的磁数据和可编程阈值之比。

这些方式可通过设置CFG_REG_B_M中的INT_on_DataOFF位进行配置。

如果INT_on_DataOFF置为0，则选择典型方式，否则，如果置为1，则选择高级方式。

硬件中断信号可以是脉冲的或锁存的：

- 脉冲中断信号：当磁数据超过两个阈值之一时，变为有效电平；当磁数据在两个阈值（正或负）之间时，变为低电平。这类中断可通过将INT_CTRL_REG_M寄存器中的IEL位置为0来选择。
- 锁存中断信号：当数据超过两个阈值中的某一个时，它变为有效电平，但它只在读取源寄存器时才会复位，而不是在磁数据返回两个阈值之间时复位。这类中断可通过将INT_CTRL_REG_M寄存器中的IEL位置为1来选择。

中断信号极性可以使用INT_CTRL_REG_M中的IEA位进行设置。

如果IEA置为1，则中断信号为高电平有效，如果置为0，则中断信号为低电平有效。

10.1 中断配置示例

下面给出了阈值事件识别的基本SW程序。

1. 将80h写入CFG_REG_A_M // 温度补偿已启用
// ODR = 10 Hz
2. 将40h写入CFG_REG_C_M // 将INT_MAG/DRDY引脚配置为数字输出，并发送中断信号
3. 将80h写入INT_THS_L_REG_M // 将阈值设置为等于128（以LSB表示）
4. 将E7h写入INT_CTRL_REG_M // 启用三个轴上的锁存高电平有效中断

示例代码中使用设置为192 mG（128 LSB * 1.5 mG / LSB）的阈值，并且事件由硬件通过INT_MAG/DRDY引脚进行通知。

10.2 溢出中断

在内部ADC电平发生测量范围溢出时，INT_SOURCE_REG_M中的MROI位会向用户发出警报。只有中断发生器有效（IEN位=1），该功能才有效。MROI特性始终为锁存的：一旦发生内部测量范围溢出，MROI位将通过读取INT_SOURCE_REG_M而复位。

11 磁力计自检

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。当磁力计自检使能时，电流会进入器件内部的线圈中。该电流会产生一个磁场，这将引起磁力计输出信号的变化。如果输出信号在幅度限制内变化，则传感器正常工作，接口芯片的参数在定义范围内。

当CFG_REG_C_M寄存器的Self_test位被禁用时，磁力计自检功能关闭；将Self_test位置为1将启用自检。

当磁力计自检功能激活时，传感器输出电平由作用在传感器上的磁场和施加电流所产生的信号的代数和给出。

该过程包括：

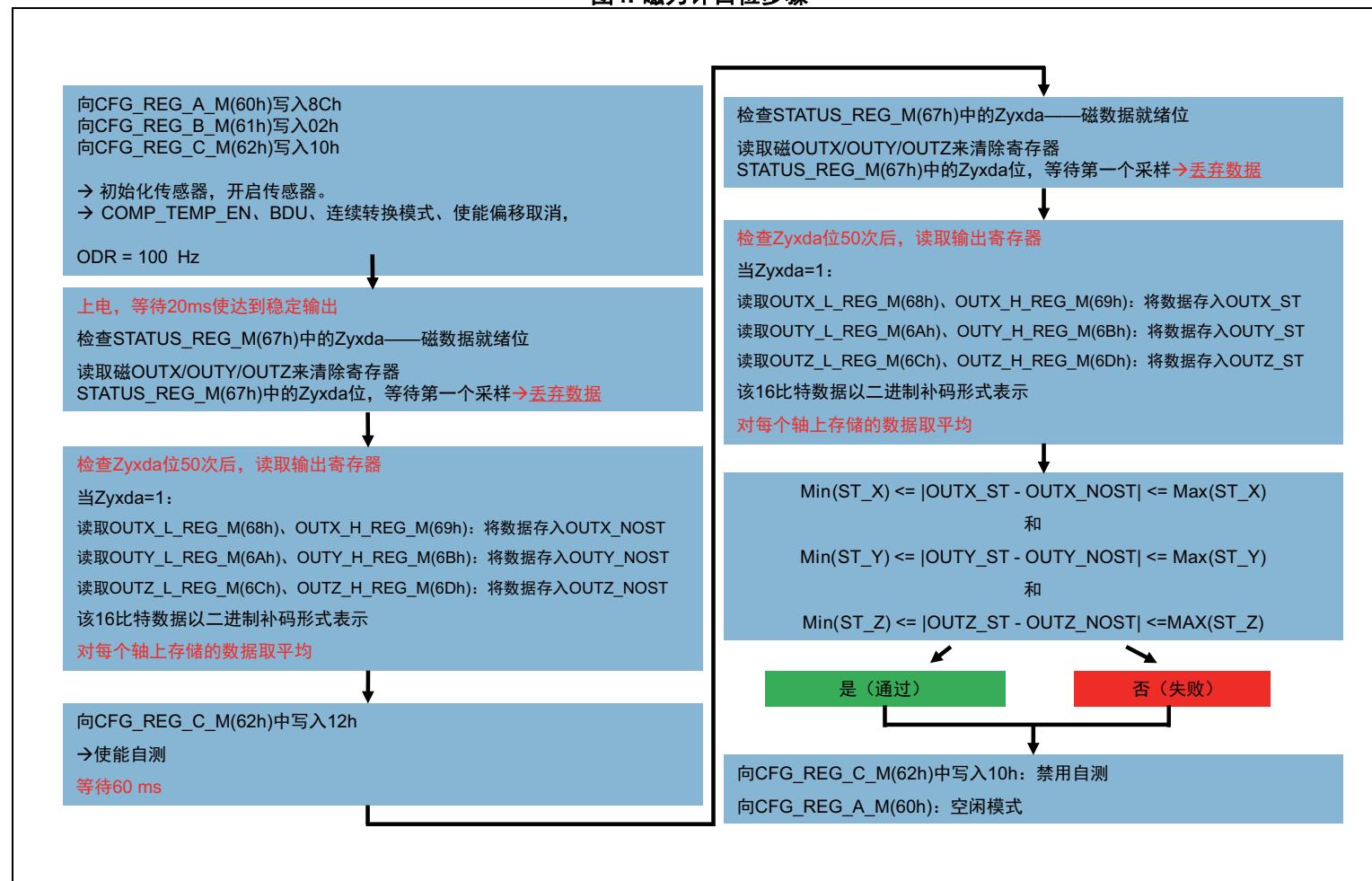
1. 启用磁力计；
2. 启动自检之前，对50个采样取平均；
3. 启动自检之后，对50个采样取平均；
4. 计算每个轴的模差，并验证它是否落在给定范围内：数据表中给出了最小值和最大值。

完整的磁力计自检过程如图 4中所示。

注：

自检过程中保持器件静止。

图4. 磁力计自检步骤



12 加速度计操作模式

LSM303AH磁力计提供两种功耗模式：高分辨率（HR）/高频模式（HF）和低功耗（LP）模式。

施加电源后，加速度计执行一段20 ms的启动程序来加载修整参数。启动完成后，器件会自动配置为掉电模式。

参考LSM303AH数据手册，可以利用输出数据率（ODR）和CTRL1_A寄存器的高频（HF_ODR）位，来选择功耗模式和加速度计传感器的输出数据率（[表 11：加速度计ODR和功耗模式选择](#)）。

表11. 加速度计ODR和功耗模式选择

ODR [3:0]	HF_ODR	模式	ODR选择[Hz]	分辨率 (位号)
0000	0	省电	省电	-
1000	0	LP	1	10
1001	0	LP	12.5	10
1010	0	LP	25	10
1011	0	LP	50	10
1100	0	LP	100	10
1101	0	LP	200	10
1110	0	LP	400	10
1111	0	LP	800	10
0001	0	HR	12.5	14
0010	0	HR	25	14
0011	0	HR	50	14
0100	0	HR	100	14
0101	0	HR	200	14
0110	0	HR	400	14
0111	0	HR	800	14
0101	1	HF	1600	12
0110	1	HF	3200	12
0111	1	HF	6400	12

输出数据具有不同的分辨率，并且是左对齐的。例如，在10位分辨率的情况下，输出数据是OUT_H_A&OUT_L_A级联的10个最高有效位，原始值必须右移6位。

表 12列出了不同工作模式下的LSM303AH加速度计功耗值。

表12. 功耗

ODR [Hz]	HR/HF (μA)	LP (μA)
1	162	4.5
12.5	162	6
25	162	8
50	162	10
100	162	16
200	162	28
400	162	52
800	162	100
1600	162	-
3200	162	-
6400	162	-

12.1 省电

加速度计处于掉电模式时，器件的全部内部块几乎都会关闭，以最大限度地降低功耗。数字接口（I²C和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，可保持进入掉电模式前存储器中采样的最后数据。

12.2 高分辨率/高频模式

HR/HF模式下，所有的加速器电路始终接通，并通过ODR位来选择生成数据的数据率。数据中断产生是活动的。

HR模式采用14位分辨率，而HF模式则采用12位分辨率（参见表 11：加速度计ODR和功耗模式选择）。

12.3 低功耗模式

低功耗模式下，加速度计电路周期性地接通/断开，其占空比是所选ODR的函数。此模式与HR/HF模式的可用输出数据速率不同。低功耗模式下，具有与HR模式加1 Hz的情形相同的数据速率（从12.5 Hz到800 Hz，但功耗较低）。

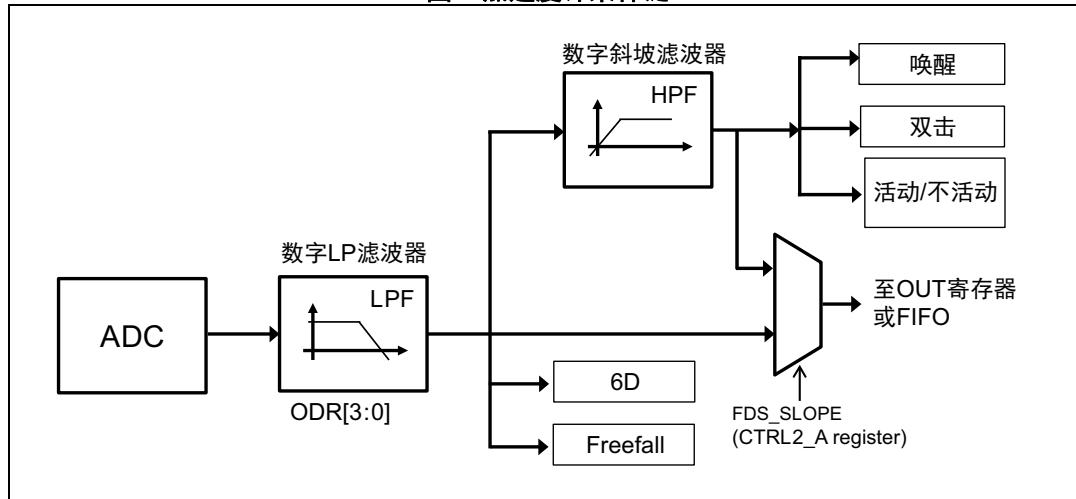
数据中断产生是活动的。

LP模式采用10位分辨率（参见表 11：加速度计ODR和功耗模式选择）。

12.4 加速度计带宽

加速度计采样链（图 5）由几个级联模块表示：ADC转换器、数字低通滤波器和数字斜率滤波器。

图5. 加速度计采样链



数字信号由低通数字滤波器（LPF）进行滤波，该滤波器截止频率取决于所选加速度计ODR，如表 13中所示。

表13. 加速度计LPF1截止频率

模式	ODR选择[Hz]	LPF截止[Hz]
LP	1	3200
LP	12.5	3200
LP	25	3200
LP	50	3200
LP	100	3200
LP	200	3200
LP	400	3200
LP	800	3200
HR	12.5	5.5
HR	25	11
HR	50	22
HR	100	44
HR	200	88
HR	400	177
HR	800	355
HF	1600	710
HF	3200	1420
HF	6400	2840

发送到OUT寄存器的信号（LPF或HPF）选择由CTRL2_A寄存器的FDS_SLOPE位决定。当它为逻辑‘1’时，选择HPF信号，否则为LPF信号。

发送到数字功能（唤醒，双击，活动/不活动和6D方向）的信号始终是HPF信号。ADC采样频率和数字LPF截止频率可确保实现抗混叠滤波。抗混叠滤波仅在HR/HF模式下可用。当加速度计处于LP模式时，电路会周期性地接通/关闭（降低功耗），具有固定的导通时间，并且周期是所选ODR的函数。因此，LPF截止频率固定为3.2 kHz，所以用户必须注意选择合适的ODR值和应用采样频率，以避免混叠（基于所用系统的噪声特性）。

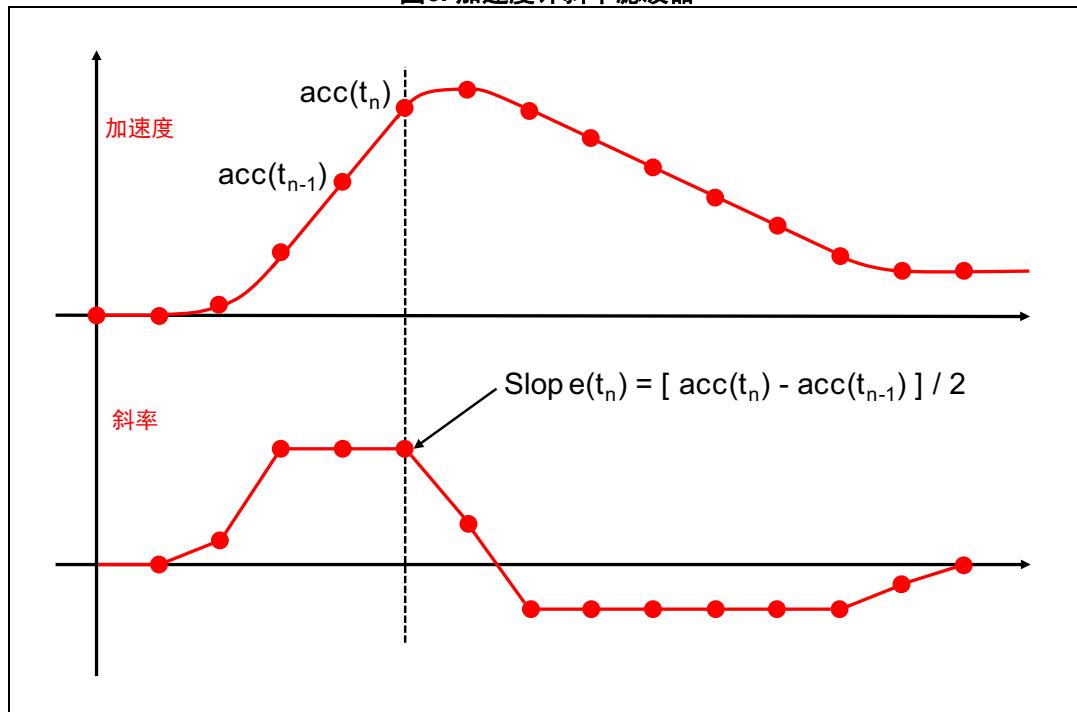
12.4.1 加速度计斜率滤波器

如图6中所示，LSM303AH加速度计嵌入了一个数字斜率滤波器，用于唤醒和单击/双击功能。该斜率滤波器输出数据利用以下公式进行计算：

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

图6中举例说明了斜率数据信号的示例。

图6. 加速度计斜率滤波器



斜率滤波器带宽为 $\sim \text{ODR}/4$ ，通过将CTRL2_A (21h) 的FDS_SLOPE位置为“1”，可在输出寄存器和FIFO中获取其数据。

13 读取加速度计输出数据

13.1 启动配置

当器件上电时，磁力计会自动从嵌入的内存中加载校准系数到内部寄存器中。启动程序完成后，也就是大约20毫秒后，加速度计会自动进入掉电。

要启用加速度计并采集加速度数据，需要通过CTRL1_A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. 写入CTRL1_A = 60h // Acc = 400 Hz (高分辨率模式)
2. 写入CTRL4_A = 01h // INT1_XL上，Acc数据准备就绪中断

13.2 使用状态寄存器

该器件具有一个STATUS_A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。当加速度计输出中有一组新的数据可用时，DRDY位被置为1。

对于加速度计，应当按照如下步骤进行读取：

1. 读取STATUS_A
2. 如果DRDY = 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. 进入1

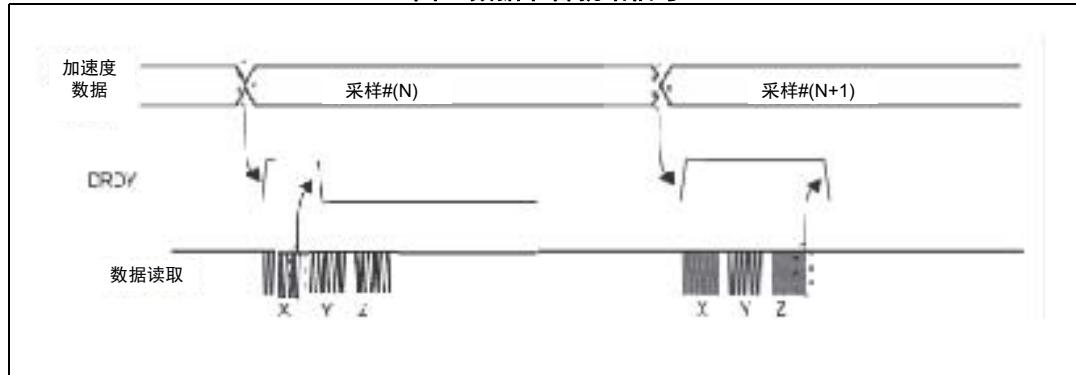
13.3 使用数据准备就绪信号

该器件可配置为具有一个HW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_A寄存器的DRDY位表示。通过将CTRL4_A寄存器的INT1_DRDY位置为1，可将该信号驱动至INT1_XL引脚，通过将CTRL5_A寄存器的INT2_DRDY位置为1，将其驱动至INT2_XL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。在DRDY锁存模式（CTRL5_A寄存器中的DRDY_PULSED位=0）下，这是默认条件，当其中一个通道的较高部分已被读取（29h、2Bh、2Dh）时，信号将复位。在DRDY脉冲模式（DRDY_PULSED = 1）下，脉冲持续时间约为75 μ s。

图7. 数据准备就绪信号



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

如果读取加速度计数据特别慢，并且不能（或者不需要）与STATUS_A寄存器中的DRDY事件位或驱动到INT1_XL/INT2_XL引脚的DRDY信号同步，那么强烈建议将CTRL1_A（20h）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采样相关的值（输出数据的最高有效部分和最低有效部分）。特别是在BDU被激活的情况下，与每条通道相关联的数据寄存器始终会包含由器件生成的最新输出数据，但如果发起了对给定寄存器对（即OUTX_H_A和OUTX_L_A、OUTY_H_A和OUTY_L_A、OUTZ_H_A和OUTZ_L_A）的读取，读取数据的MSB和LSB部分之前，都会禁止刷新该寄存器对。

注：
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。

13.5 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。这些寄存器分别容纳加速度信号在X、Y和Z轴上的最高有效部分和最低有效部分。

X、Y、Z通道的完整输出数据是由OUTX_H_A & OUTX_L_A、OUTY_H_A & OUTY_L_A、OUTZ_H_A & OUTZ_L_A共同提供的，表示为2的补码。

加速度数据表示为16位数字，称为LSB，但根据所选择的工作模式（LP/HR/HF）不同，其分辨率也不同。请参见表 11：加速度计ODR和功耗模式选择。

计算LSB后，必须乘以适当的灵敏度参数，得到按mg计的相应值。

13.5.1 输出数据示例

以下是一个简单的示例，说明如何使用LSB数据并将其转换成mg。

这些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，……）。

从传感器获取原始数据（HR模式，ODR 200 Hz）：

OUTX_L_A: 5Ch

OUTX_H_A: FDh

OUTY_L_A: 74h

OUTY_H_A: 00h

OUTZ_L_A: F8h

OUTZ_H_A: 42h

将寄存器串联：

OUTX_H_A & OUTX_L_A: FD5Ch

OUTY_H_A & OUTY_L_A: 0074h

OUTZ_H_A & OUTZ_L_A: 42F8h

应用灵敏度（例如，满刻度为 $\pm 2\text{ g}$ 时，使用0.061）：

X: $-676 * 0.061 = -41\text{ mg}$

Y: $+116 * 0.061 = +7\text{ mg}$

Z: $+17144 * 0.061 = +1046\text{ mg}$

14 加速度计中断生成和嵌入功能

在LSM303AH加速度计中，中断生成基于加速度计数据，因此，为了生成中断，加速度计传感器必须设置为活动工作模式（不是掉电）。

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；
- 单击和双击感测；
- 活动/不活动检测。

此外，LSM303AH加速度计能够高效运行先进操作系统中特定的传感器相关功能，节能并且反应时间更快。特别地，经过专门设计，它可在硬件中实现：

- 大幅运动检测；
- 倾斜度检测；
- 计步功能；

所有这些中断信号，以及FIFO中断信号和传感器数据就绪信号，可被独立地驱动至INT1_XL和INT2_XL中断引脚，或通过读取特定源寄存器位分别对其进行检测。

注意：当MODULE_ON位置为1时，嵌入功能（计步、倾斜和大幅运动检测）不可用。

当DRDY信号被发送到中断引脚时，必须使用CTRL3_A寄存器的H_LACTIVE位来选择它们的极性。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。

当DRDY信号被发送到中断引脚时，CTRL3_A的PP_OD位还允许将这些中断引脚的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。

CTRL3_A的LIR位可支持中断信号应用锁存模式（不影响DRDY信号）。LIR位置为1时，中断引脚一旦被声明，就必须通过读取相关中断源寄存器才能将其复位。如果LIR位置为0，则当不再检测到中断条件或一定时间后（针对不同的中断类型），中断信号可自动复位。

14.1 中断引脚配置

该器件具有两个引脚，可激活这些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_XL引脚是通过CTRL4_A寄存器来进行选择，对于INT2_XL引脚是通过CTRL5_A寄存器来进行选择。

以下是这些中断控制寄存器的说明；这些比特的默认值等于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14. CTRL4_A寄存器

b7	b6	b5	b4	b3	b2	b1	b0
0	INT1_S_TAP	INT1_WU	INT1_FF	INT1_TAP	INT1_6D	INT1_FTH	INT1_DRDY

- INT1_S_TAP: 单击事件识别被发送到INT1_XL引脚。
- INT1_WU: 唤醒事件识别被发送到INT1_XL引脚。
- INT1_FF: 自由落体事件识别被发送到INT1_XL引脚。
- INT1_TAP: 点击事件识别被发送到INT1_XL引脚。
- INT1_6D: 6D事件识别被发送到INT1_XL引脚。
- INT1_FTH: FIFO阈值事件被发送到INT1_XL引脚。
- INT1_DRDY: 加速度计数据就绪信号被发送到INT1_XL引脚。

表15. CTRL5_A寄存器

b7	b6	b5	b4	b3	b2	b1	b0
DRDY_PULSED	INT2_BOOT	INT2_ON_INT1	INT2_TILT	INT2_SIG_MOT	INT2_STEP	INT2_FTH	INT2_DRDY

- DRDY_PULSED: 数据就绪中断模式选择：锁存模式/脉冲模式。
- INT2_BOOT: 启动状态发送到INT2_XL引脚。
- INT2_ON_INT1: 所有INT2信号也都发送到INT1_XL引脚。
- INT2_TILT: 倾斜事件识别发送到INT2_XL引脚。
- INT2_SIG_MOT: 大幅运动事件识别发送到INT2_XL引脚。
- INT2_STEP: 计步事件识别发送到INT2_XL引脚。
- INT2_FTH: FIFO阈值事件发送到INT2_XL引脚。
- INT2_DRDY: 加速度计数据就绪信号在INT2_XL引脚。

14.2 事件状态

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位所选中断信号组合的‘或’。为了知道哪个事件产生了中断条件，应用程序要读取正确的状态寄存器，这也清除事件。

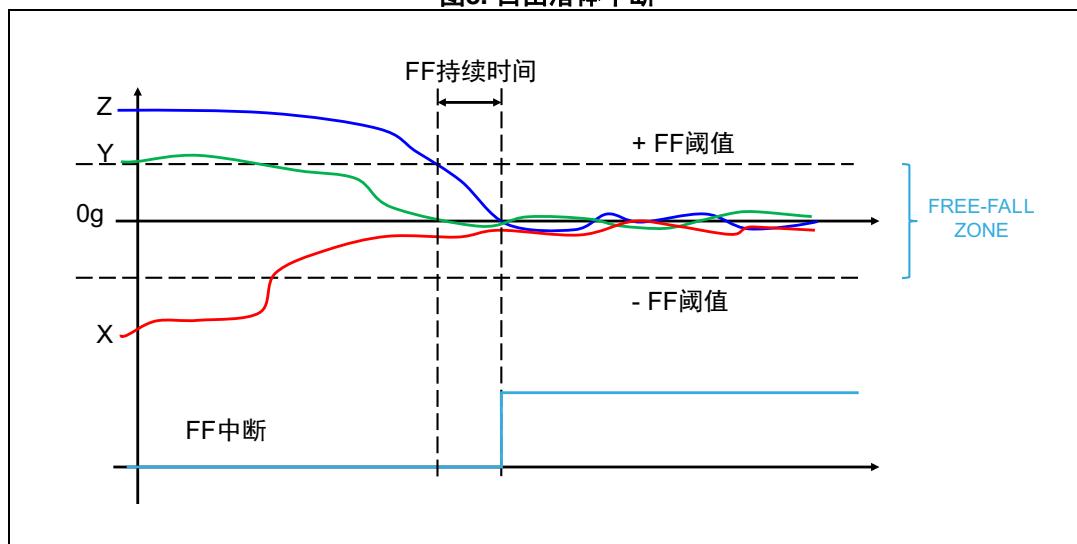
如下所示，STATUS_A寄存器在地址36h处被复制，以便对连续寄存器 (36h/37h/38h/39h) 进行多次读取。

- STATUS_A (27h)或STATUS_DUP_A (36h)
- WAKE_UP_SRC_A (37h)
- TAP_SRC_A (38h)
- 6D_SRC_A (39h)
- FUNC_CK_GATE_A (3Dh)

14.3 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‘自由落体区域’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。自由落体事件检测关联了可配置的阈值和持续时间参数：阈值参数定义了自由落体区幅度；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 8：自由落体中断](#)）。

图8. 自由落体中断



通过将CTRL4_A寄存器的INT1_FF位置为1，可将自由落体事件信号传送至INT1_XL引脚上；还可通过读取WAKE_UP_SRC_A寄存器的FF_IA位对其进行检查。

如果锁存模式禁用 (CTRL3_A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。如果锁存模式使能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_A寄存器来将其复位。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（当不再核验到自由落体条件时，WAKE_UP_SRC_A中的FF_IA位复位）。

用来配置阈值参数的寄存器命名为FREE_FALL_A；无符号阈值与FF_THS[2:0]字段值的值相关，如表 16 所示，以31.25 mg为单位表示。此表中给出的LSB值对于任何加速度计满量程值均有效。

表16. 自由落体阈值LSB值

FREE_FALL_A - FF_THS[2:0]	阈值LSB值
000	5
001	7
010	8
011	10
100	11
101	13
110	15
111	16

持续时间在N/ODR中测得，其中N为FREE_FALL_A / WAKE_UP_DUR_A寄存器FF_DUR[5:0]字段的内容，ODR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_A // 启动加速度计
// ODR = 400 Hz, FS = ±2 g
2. 将00h写入WAKE_UP_DUR_A // 设置事件持续时间 (FF_DUR5 = 0)
3. 将33h写入FREE_FALL_A // 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)
4. 将10h写入CTRL4_A // FF中断驱动至INT1_XL引脚
5. 将04h写入CTRL3_A // 锁存中断

示例代码中利用设置为~310 mg (31.25 mg * 10) 的阈值，用于自由落体识别，该事件由硬件通过INT1_XL引脚进行通知。FREE_FALL_A / WAKE_UP_DUR_A寄存器的FF_DUR[5:0]字段像这样配置：忽略短于6/ODR = 6/400 Hz = 15 ms的事件，以避免错误检测。

14.4 唤醒中断

在LSM303AH器件中，唤醒功能可利用斜率滤波器（更多详细信息参见[第 12.4.1 节：加速度计斜率滤波器](#)）来实现，如[图 6：加速度计斜率滤波器](#)中所示。如果一定数量的连续斜率滤波数据超出了所配置阈值（[图 9：唤醒中断](#)），则产生唤醒中断信号。

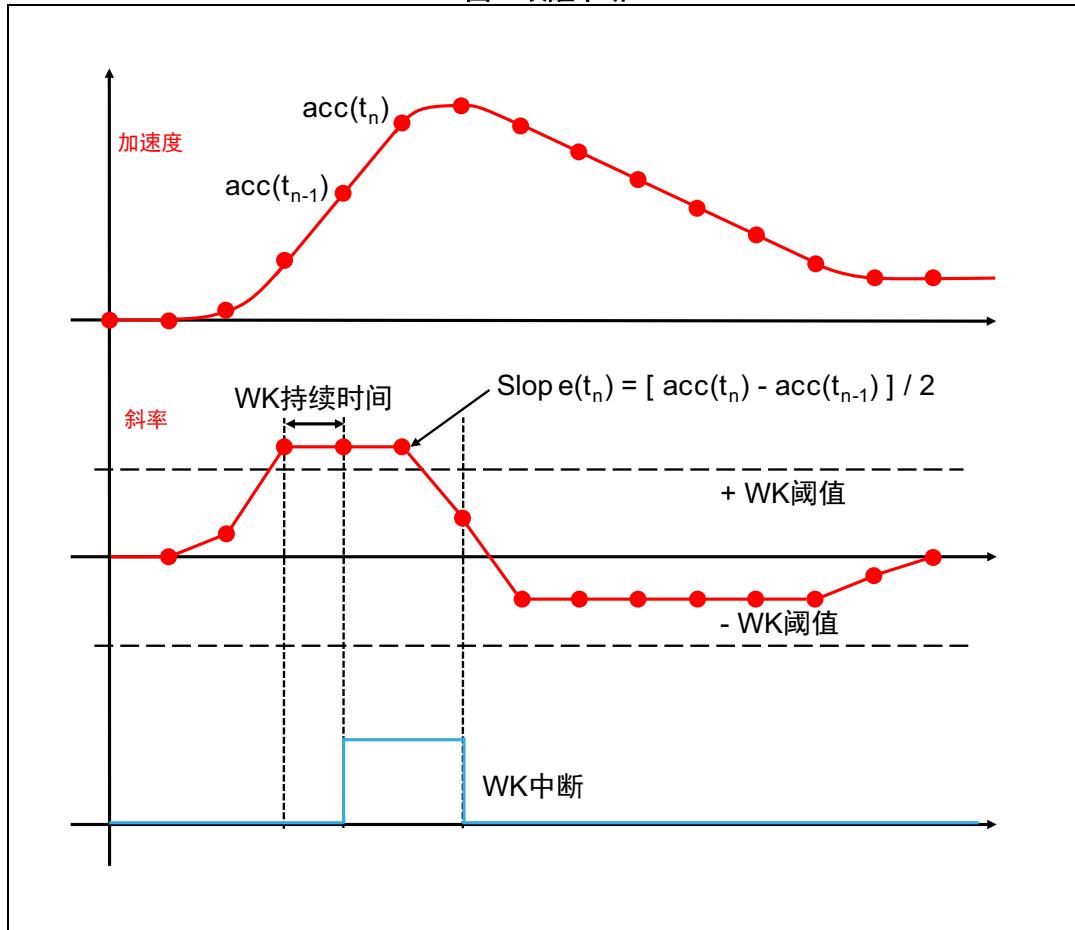
该无符号阈值由WAKE_UP_THS_A寄存器的WU_THS[5:0]位来定义；这些6位的1LSB值取决于所选加速度计满量程：1 LSB = FS/64。阈值可应用于正负数据：对于唤醒中断生成，三个轴中至少有一个必须大于阈值。

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_A寄存器的WU_DUR[1:0]位来设置：1LSB对应于1*ODR时间，这里ODR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CTRL4_A寄存器的INT1_WU位置为1，可将该中断信号驱动至INT1_XL中断引脚上；还可通过读取WAKE_UP_SRC_A寄存器的WU_IA位对其进行检查。WAKE_UP_SRC_A寄存器的X_WU、Y_WU、Z_WU位指示哪个轴触发了唤醒事件。

如果锁存模式禁用（CTRL3_A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_A寄存器来将其复位。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（当不再检验到自由落体条件时，WAKE_UP_SRC_A中的WU_IA位复位）。

图9. 唤醒中断



下面给出的示例代码实现了用于唤醒事件识别的SW程序。

1. 将60h写入CTRL1_A // 启动加速度计
// ODR = 400 Hz, FS = ±2 g
2. 将00h写入WAKE_UP_DUR_A // 无持续
3. 将02h写入WAKE_UP_THS_A // 设置唤醒阈值
4. 将20h写入CTRL4_A // 唤醒中断驱动至INT1_XL引脚

由于持续时间被置为0，因此每个X、Y、Z斜率数据超出所配置阈值时，会生成唤醒中断信号。WAKE_UP_THS_A寄存器的WU_THS字段被置为000010b，因此唤醒阈值为62.5 mg (= 2 * FS / 64)。

14.5 6D/4D方向检测

LSM303AH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。

14.5.1 6D方向检测

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，中断信号被声明。只要保持其位置，中断就不会重新声明。

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：6D_SRC_A寄存器的ZH、ZL、YH、YL、XH、XL位可表示出哪个轴触发了6D事件。

更多细节：

表17. 6D_SRC_A寄存器

b7	b6	b5	b4	b3	b2	b1	b0
0	6D_IA	ZH	ZL	YH	YL	XH	XL

- 当器件从一个方向转向另一个方向时，6D_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模块中的加速度大于阈值时，ZH (YH、XH) 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负且模块中的加速度大于阈值时，ZL (YL、XL) 被置为高电平。

TAP_6D_THS_A寄存器的6D_THS[1:0]位用来选择阈值，该阈值用于检测器件方向变化。

表 18: 4D/6D功能阈值中给出的阈值对于每种加速度计满量程值均有效。

表18. 4D/6D功能阈值

6D_THS[1:0]	阈值[degrees]
00	80
01	70
10	60
11	50

通过将CTRL4_A寄存器的INT1_6D位置为1，可将该中断信号驱动至INT1_XL中断引脚上；还可通过读取6D_SRC_A寄存器的6D_IA位对其进行检查。

如果锁存模式禁用 (CTRL3_A的LIR位置为0)，则中断信号仅激活1/ODR[s]，然后自动失效 (ODR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6D_SRC_A寄存器的读取会清除请求，器件将识别另一个不同的方向。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 10: 6D 识别方向](#)中所示的六种可能情形, [表 19: 用于6D定位的6D_SRC_A寄存器](#)中显示了每个位置对应的D6D_SRC_A寄存器内容所示。

图10. 6D识别方向

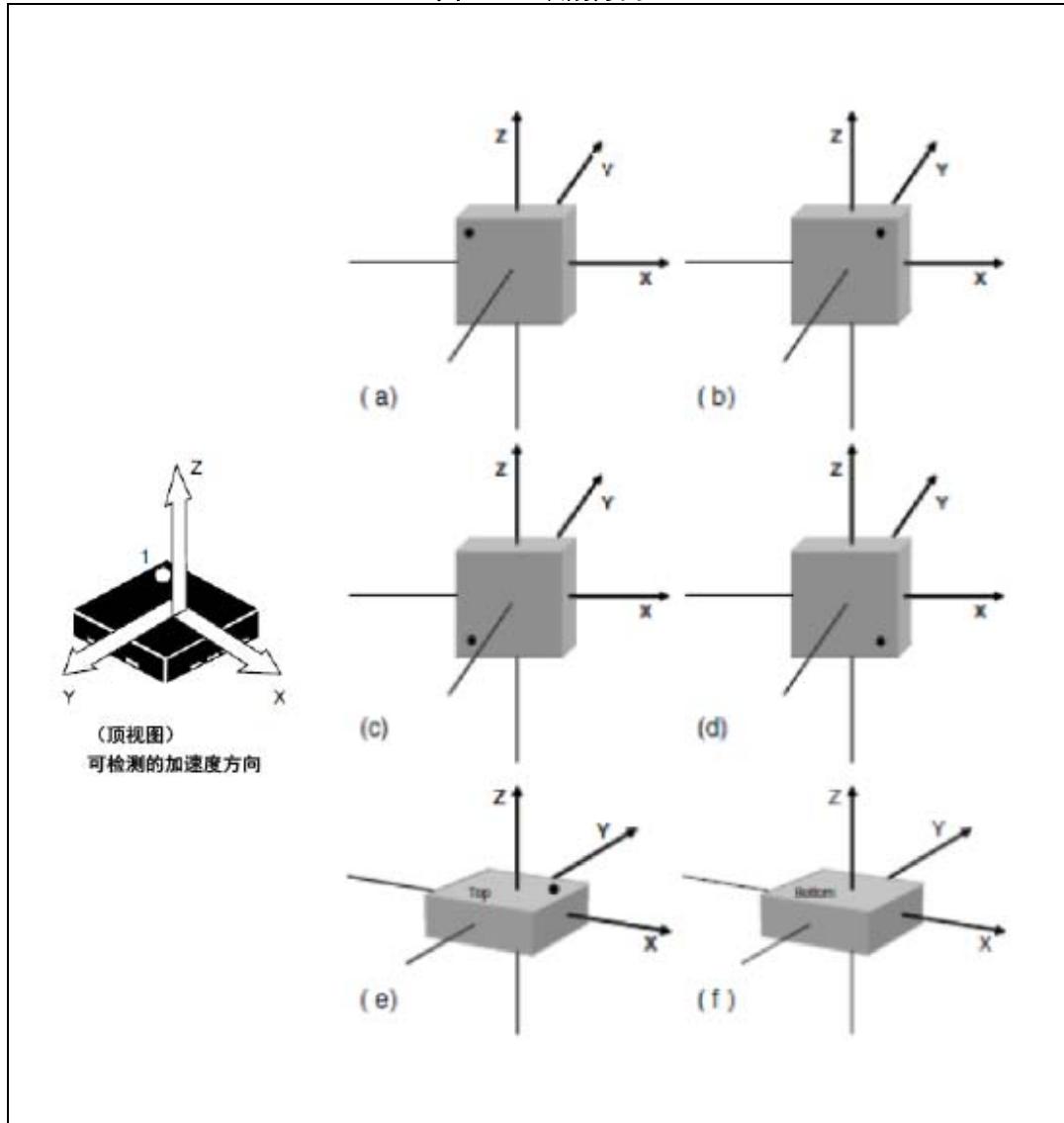


表19. 用于6D定位的6D_SRC_A寄存器

情形	6D_IA	ZH	ZL	YH	YL	XH	XL
(a)	1	0	0	0	1	0	0
(b)	1	0	0	0	0	0	1
(c)	1	0	0	0	0	1	0
(d)	1	0	0	1	0	0	0
(e)	1	1	0	0	0	0	0
(f)	1	0	1	0	0	0	0

下面的示例实现了用于6D方向检测的SW程序：

1. 将60h写入CTRL1_A // 启动加速度计
// ODR = 400 Hz, FS = ±2 g
2. 将40h写入TAP_6D_THS_A // 设置6D阈值 (6D_THS[1:0] = 10b = 60 degrees)
3. 将04h写入CTRL4_A // 6D中断驱动至INT1_XL引脚

14.5.2 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TAP_6D_THS_A寄存器的4D_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 19: 用于6D定位的6D_SRC_A寄存器的(a)、(b)、(c) 和 (d) 的情形。

14.6 单击和双击识别

LSM303AH具有单击和双击识别功能，能够在极少加载软件的情况下帮助创建人机界面。该器件可配置为，当在任一方向点击时，在专用引脚上输出一个中断信号。

如果传感器施加单个输入激励，那么它会在中断引脚INT1_XL上产生中断请求。其高级功能可支持在识别出一个双输入激励（它在两个事件之间具有可编程时间）时产生中断请求，从而使能类似于鼠标按钮的功能。

LSM303AH器件中，单击和双击识别功能利用两个连续加速度采样之间的斜率来检测点击事件；斜率数据利用以下公式计算：

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

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斜率数据幅度和时序进行编程。

单击和双击识别仅在ODR ≥ 400 Hz时有意义。

14.6.1 单击

如果器件配置为单击事件检测，那么当所选通道的斜率数据超出了所编程阈值时，会产生一个中断，并在Shock时间窗口内返回低电平。

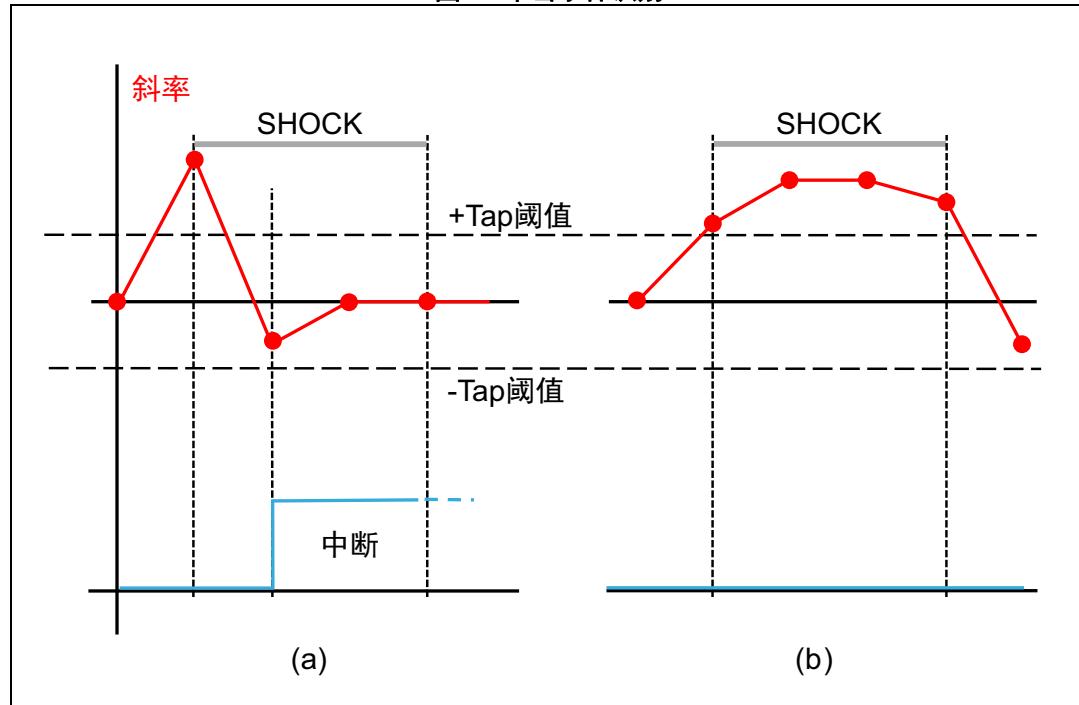
在单击情况下，如果CTRL3_A寄存器的LIR位被置为0，则中断在Quiet窗口持续时间内保持高电平。

为了在单击中断信号上使能锁存功能，CTRL3_A的LIR位必须置为1：中断保持高电平，直至TAP_SRC_A寄存器被读取。

要实现仅使能单击识别，则WAKE_UP_THS_A的SINGLE_DOUBLE_TAP位必须置为0。

图 11：单击事件识别的情况 (a) 中识别出了单击事件，而在情况 (b) 中，由于在经过 Shock时间窗口之后斜率数据低于阈值，因此未识别出点击。

图11. 单击事件识别



14.6.2 双击

如果器件配置为双击事件检测，那么在第一次点击后、识别出第二次点击时，生成中断。只有当事件满足Shock、Latency和Quiet时间窗口所定义的规则时，才进行第二次点击识别。

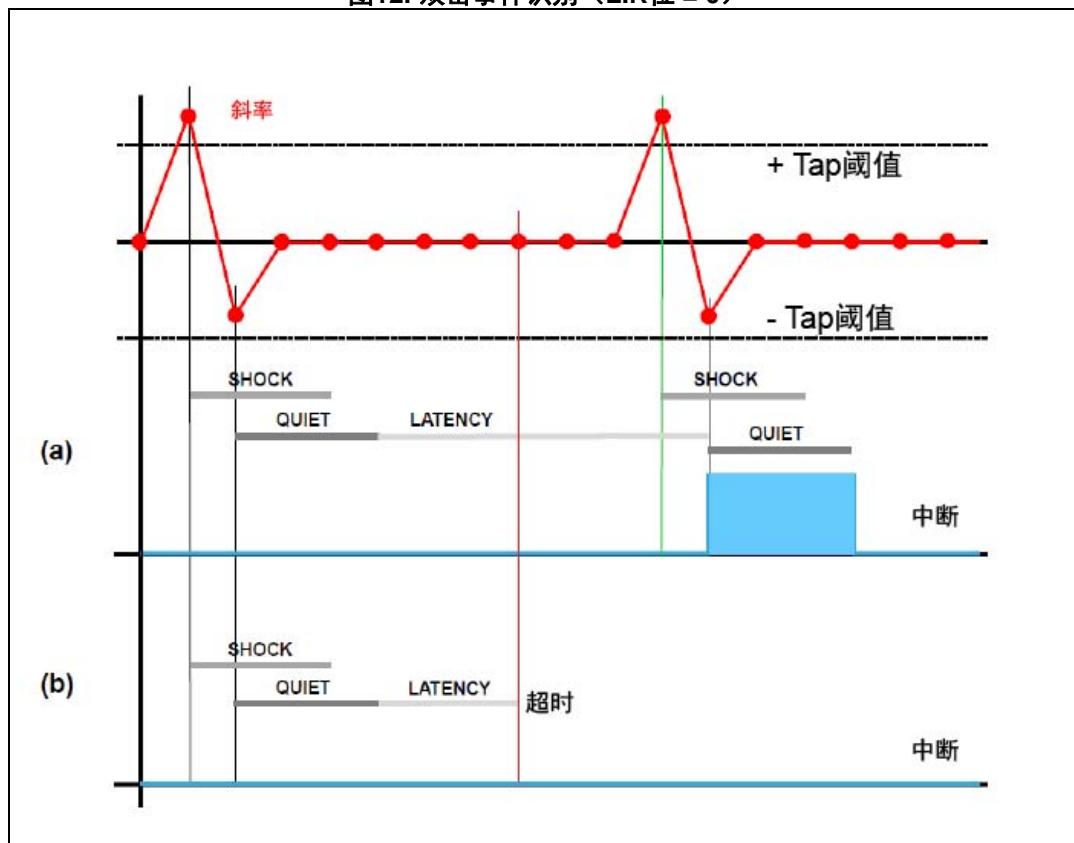
特别地，识别出第一次点击后，第二次点击检测过程会延迟Quiet时间所定义的时间间隔。这意味着，识别出第一次点击后，只有在Quiet窗口之后、Latency窗口结束前，斜率数据超过阈值时，才开始第二次点击识别过程。[图 12：双击事件识别 \(LIR 位 = 0\)](#) 的情况 (a) 中，正确识别出了双击事件，而在情况 (b) 中，由于在经过Latency窗口间隔之后斜率数据超出了阈值，因此未产生中断。

一旦第二次点击检测过程开始，则会按照与第一次相同的规则来识别第二次点击：在Shock窗口结束之前，斜率数据必须返回到低于阈值之下。

要避免因输入信号伪突变而产生不期望的点击，适当定义Quiet窗口是非常重要的。

在双击情况下，如果CTRL3_A寄存器的LIR位被置为0，则中断在Quiet窗口持续时间内保持高电平。如果LIR位被置为1，则中断保持高电平直至TAP_SRC_A寄存器被读取。

图12. 双击事件识别 (LIR 位 = 0)



14.6.3 单击和双击识别配置

可对LSM303AH器件进行配置，使其在任一方向发生点击（一次或两次）时均输出中断信号：CTRL3_A寄存器的TAP_X_EN、TAP_Y_EN和TAP_Z_EN位必须置为1，分别使能X、Y、Z方向上的点击识别。

点击识别功能的可配置参数为点击阈值和shock、quiet和latency时间窗。有效的ODR为400 Hz、800 Hz和1600 Hz。

TAP_6D_THS_A寄存器的TAP_THS[4:0]位用来选择用于检测点击事件的无符号阈值。这5个比特的1 LSB值取决于所选加速度计满量程：1 LSB = FS/32。无符号阈值可应用于正负斜率数据上。

Shock时间窗口定义了超阈值事件的最大持续时间：在Shock窗口结束前，加速度必须返回到低于阈值之下，否则不能检测到该点击事件。INT_DUR_A寄存器的SHOCK[1:0]位用来设置Shock时间窗口值：这几个位的默认值为00b，对应于4/ODR的时间，这里ODR为加速度计输出数据率。如果SHOCK[1:0]位被置为其他不同的值，那么1 LSB对应于8/ODR的时间。

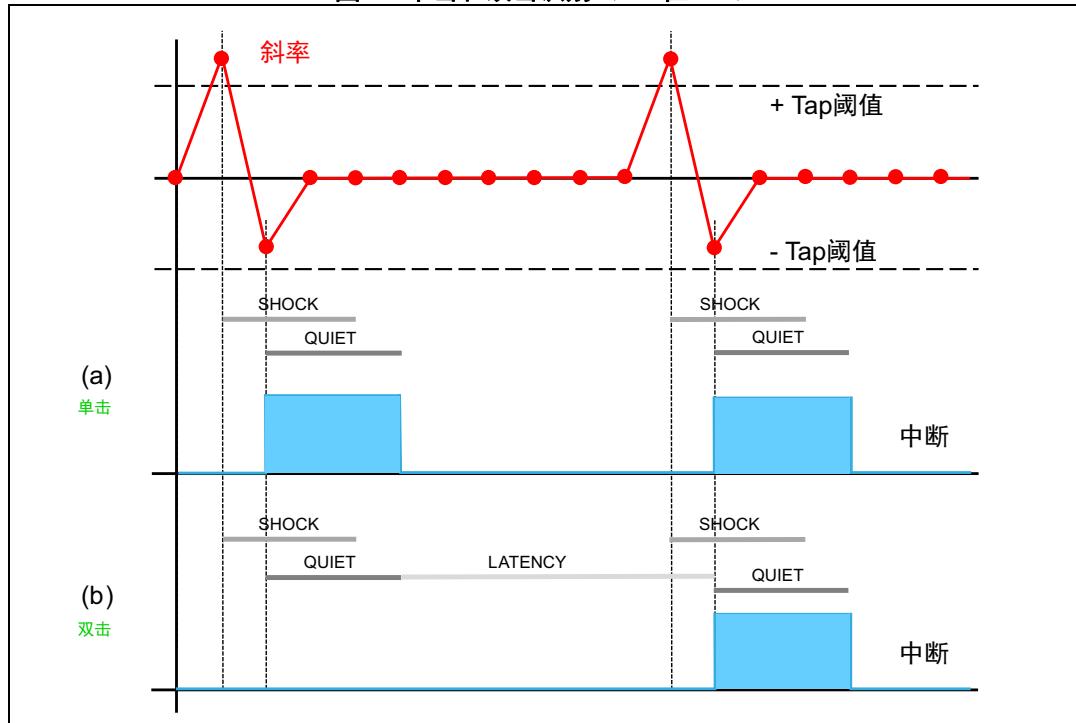
双击情况下，Quiet时间窗口定义了第一次点击识别后的时间，期间不能发生超阈值。当锁存模式禁用（CTRL3_A的LIR位置为0）时，Quiet时间还定义了中断脉冲的长度（单击和双击情况下均如此）。INT_DUR_A寄存器的QUIET[1:0]位用来设置Quiet时间窗口值：这几个位的默认值为00b，对应于2/ODR的时间，这里ODR为加速度计输出数据率。如果QUIET[1:0]位被置为其他不同的值，那么1 LSB对应于4/ODR的时间。

双击情况下，latency时间窗口定义了连续两次检测到点击之间的最大时间。latency时间周期在第一次点击的quiet时间结束后开始。INT_DUR_A寄存器的LAT[3:0]位用来设置latency时间窗口值：这几个位的默认值为0000b，对应于16*ODR_XL的时间，这里ODR_XL为加速度计输出数据率。如果LAT[3:0]位被置为其他不同的值，那么1 LSB对应于32/ODR的时间。

图 13: 单击和双击识别 (LIR位 = 0) 显示了单击事件(a) 和双击事件(b)。这些中断信号可被驱动至INT1_XL中断引脚，单击情况下通过将CTRL4_A寄存器的INT1_S_TAP位置为1来实现，双击情况下通过将CTRL4_A寄存器的INT1_TAP位置为1来实现。

如果加速度计处于不活动状态，则不产生单击/双击中断（更多详细信息见[第 14.7 节：活动/不活动识别](#)）。

图13. 单击和双击识别 (LIR位 = 0)



还可通过读取TAP_SRC_A (38h) 寄存器来检查点击中断信号，如表 20: TAP_SRC_A 寄存器所述。

表20. TAP_SRC_A寄存器

b7	b6	b5	b4	b3	b2	b1	b0
0	TAP_IA	SINGLE_TAP	DOUBLE_TAP	TAP_SIGN	X_TAP	Y_TAP	Z_TAP

- 当检测到单击或双击事件时，TAP_IA置为高电平。
- 当检测到单击时，SINGLE_TAP置为高电平。
- 当检测到双击时，DOUBLE_TAP置为高电平。
- TAP_SIGN指示检测到点击事件时的加速度符号。符号为正时它为低电平，符号为负时它为高电平。
- 当在X (Y、Z) 轴上检测到点击事件时，X_TAP (Y_TAP、Z_TAP) 置为高电平

单击和双击识别独立工作。将WAKE_UP_THS_A的SINGLE_DOUBLE_TAP位置为0，则仅使能单击识别：双击识别被禁用，不能被检测到。当SINGLE_DOUBLE_TAP置为1时，单击和双击识别均使能。

如果锁存模式使能，且中断信号被驱动至中断引脚，则指定到SINGLE_DOUBLE_TAP的值还会影晌中断信号的特性：当它被置为0时，单击中断信号可应用锁存模式；当它被置为1时，只有双击中断信号可应用锁存模式。锁存的中断信号保持为高电平，直至TAP_SRC_A寄存器被读取。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。

14.6.4 单击示例

下面的示例代码实现了用于单击检测的SW程序。

1. 将60h写入CTRL1_A // 启动加速度计
// ODR = 400 Hz, FS = ±2 g
2. 将38h写入CTRL3_A // 使能X、Y、Z轴上的点击检测
3. 将09h写入TAP_6D_THS_A // 设置点击阈值
4. 将06h写入INT_DUR_A // 设置Quiet和Shock时间窗口
5. 将00h写入WAKE_UP_THS_A // 只使能单击 (SINGLE_DOUBLE_TAP = 0)
6. 将40h写入CTRL4_A // 单击中断驱动至INT1_XL引脚

本例中, TAP_6D_THS寄存器的TAP_THS字段被置为01001b, 因此点击阈值为562.5 mg (= 9 * FS / 32)。

INT_DUR_A寄存器的SHOCK字段被置为10b: 当斜率数据超出所编程阈值时, 产生中断, 并在40 ms (= 2 * 8 / ODR) 内返回到低于该阈值, 这段时间对应于Shock时间窗口。

INT_DUR_A寄存器的QUIET字段被置为01b: 由于锁存模式禁用, 中断会保持高电平并持续Quiet窗口的时间, 因此为10 ms (= 1 * 4 / ODR)。

14.6.5 双击示例

下面给出的示例代码实现了用于单击检测的SW程序。

1. 将60h写入CTRL1_A // 启动加速度计
// ODR = 400 Hz, FS = ±2 g
2. 将38h写入TAP_CFG_A // 使能X、Y、Z轴上的点击检测
3. 将0Ch写入TAP_6D_THS_A // 设置点击阈值
4. 将7Fh写入INT_DUR_A // 设置Duration、Quiet和Shock时间窗口
5. 将80h写入WAKE_UP_THS_A // 使能单击 & 双击 (SINGLE_DOUBLE_TAP = 1)
6. 将08h写入CTRL4_A // 双击中断驱动至INT1_XL引脚

本例中, TAP_6D_THS寄存器的TAP_THS字段被置为01100b, 因此点击阈值为750mg (= 12 FS / 32)。

要实现中断生成, 在第一次和第二次点击过程中, Shock结束前, 斜率数据必须返回到低于阈值。INT_DUR_A寄存器的SHOCK字段被置为11b, 因此Shock时间为60 ms (= 3 * 8 / ODR)。

对于中断生成, 第一次点击识别后, 在Quiet时间窗口内斜率数据不能超阈值。而且, 由于锁存模式禁用, 因此中断会保持高电平, 并持续Quiet窗口的时间。INT_DUR_A寄存器的QUIET字段被置为11b, 因此Quiet时间为30 ms (= 3 * 4 / ODR)。

要使连续两次检测到的点击之间时间达到最大, INT_DUR_A寄存器的LAT字段被置为0111b, 因此Duration时间为560 ms (= 7 * 32 / ODR)。

14.7 活动/不活动识别

活动/不活动识别功能能够减少系统功耗，可支持开发新型智能应用。

当活动/不活动识别功能激活时，LSM303AH器件能够自动进入低功耗模式并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重新增加加速度计ODR和带宽。

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WAKE_UP_THS_A寄存器的SLEEP_ON位置为1，可使能活动/不活动识别。

活动/不活动识别功能利用两个连续加速度采样之间的斜率来检测活动/不活动事件；斜率数据利用以下公式进行计算：

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

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斜率数据幅度和时序进行编程（[图 14：活动/不活动识别](#)）。

该无符号阈值由WAKE_UP_THS_A寄存器的WK_THS[5:0]位来定义；这些6位的LSB值取决于所选加速度计满量程：1 LSB = FS的1 / 64。该阈值可适用于正斜率数据和负斜率数据。

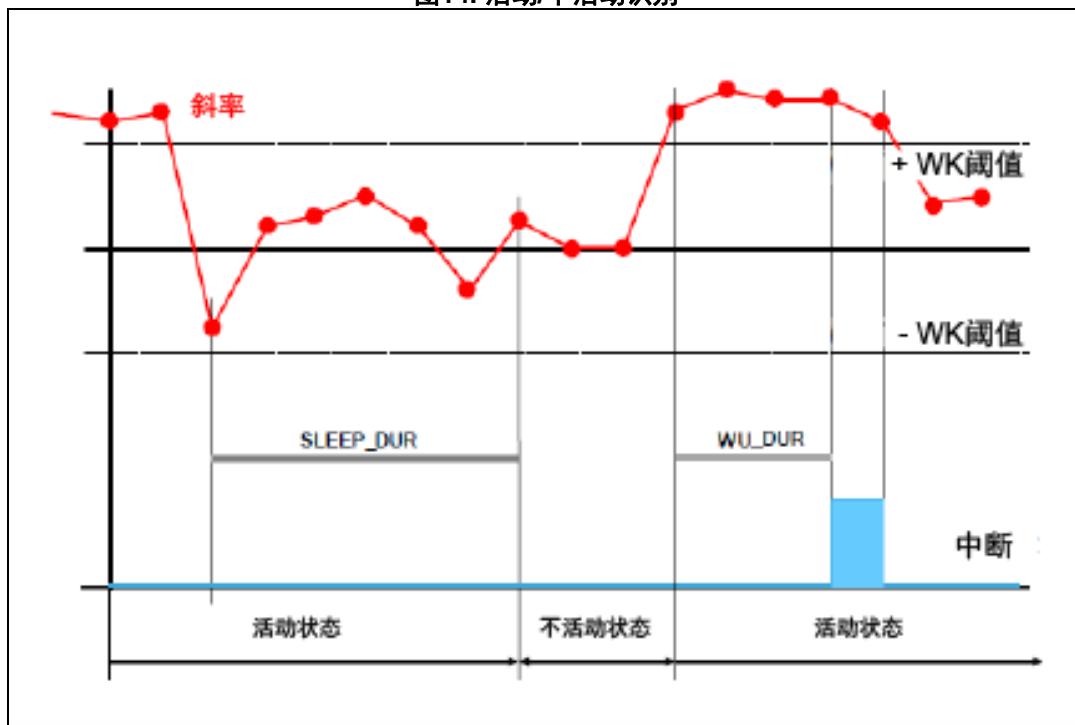
当一定数量的连续X、Y、Z斜率数据小于所配置阈值时，忽略CTRL1_A寄存器的ODR[3:0]位（不活动），加速度计被内部地设置为12.5 Hz，尽管CTRL1_A内容保持不变。待识别的不活动状态的持续时间由WAKE_UP_DUR_A寄存器的SLEEP_DUR[3:0]位来定义：1 LSB对应于512/ODR的时间，这里ODR为加速度计输出数据率。

当检测到不活动状态时，不会对应用处理器产生中断（WAKE_UP_SRC_A寄存器的SLEEP_STATE_IA位不会传送到引脚）。

当一个轴上有一个采样的斜率数据大于阈值时，立即恢复CTRL1_A寄存器设置（活动）。唤醒中断事件可以通过WAKE_UP_DUR_A寄存器WU_DUR[1:0]位的值来延迟：1 LSB对应于1/ODR的时间，这里ODR为加速度计输出数据率。为了在不活动/活动事件的同时产生中断，WU_DUR[1:0]必须置为0。

当检测到唤醒事件时，中断设置为高电平并持续1/ODR时长，然后自动置为无效（必须将WU_IA事件传送到引脚上，可通过将CTRL4_A寄存器的INT1_WU位设置为1来实现）。

图14. 活动/不活动识别



下面给出的代码是实现活动/不活动检测的基本程序。

1. 将50h写入CTRL1_A // 启动加速度计 // ODR = 200 Hz, FS = $\pm 2 g$
2. 将42h写入WAKE_UP_DUR_A // 设置不活动检测的持续时间 // 设置唤醒检测的持续时间
3. 将42h写入WAKE_UP_THS_A // 设置活动/不活动阈值 // 使能活动/不活动检测
4. 将20h写入CTRL4_A // 活动（唤醒）中断驱动至INT1_XL引脚

本例中，WAKE_UP_THS_A寄存器的WU_THS字段被置为000010b，因此活动/不活动阈值为62.5 mg ($= 2 * FS / 64$)。

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_A寄存器的SLEEP_DUR字段定义：该字段被置为0010b，对应5.12s ($= 2 * 512 / ODR$)。这段时间之后，加速度计ODR被内部地设置为12.5 Hz。

如果（至少）一个轴的斜率数据大于阈值，并且在一段时间间隔之后通知发生唤醒中断，则会检测到活动状并立即恢复CTRL1_A寄存器设置，该时间间隔由WAKE_UP_DUR_A寄存器WU_DUR字段定义：此字段被置为10b，对应10 ms ($= 2 * 1 / ODR$)。

14.8 启动状态

器件上电后, LSM303AH加速度计执行一段20ms的启动程序来加载修整参数。启动完成后, 加速度计会自动配置为掉电模式。

上电后, 可通过将CTRL2_A寄存器的BOOT位置为1, 来重载修调参数。

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CTRL32_A寄存器的SW_RESET位置为1来实现。

通过将CTRL5_A寄存器的INT2_BOOT位置为1, 可以将启动状态信号驱动到INT2_XL中断引脚: 在进行启动时该信号变为“1”, 完成后返回“0”。

要将器件恢复为掉电默认设置, 在任何操作模式下都请执行以下步骤:

1. 将SW_RESET位置为“1”
2. 等待, 直至SW_RESET位返回“0”
3. 将BOOT位置为“1”
4. 等待20 ms

14.9 嵌入功能

LSM303AH加速度计在硬件中实现主要操作系统专用的传感器相关功能; 功耗可忽略且高性能的专用IP模块可实现以下功能:

- 计步功能 (步伐侦测和步数计算)
- 大幅运动检测
- 倾斜度检测

14.9.1 计步功能: 步伐侦测和步数计算

LSM303AH加速度计的专用IP模块来专门实现计步功能: 步伐侦测和步数计算。

计步功能工作于25 Hz, 因此加速度计ODR必须设置为25 Hz或更高的值。可以检测的最大计步频率约为3 Hz。

为了启用计步功能, FUNC_CTRL_A寄存器的STEP_CNT_ON位必须置为1。当计步器启用时, Module_8bit_A (0Ch) 寄存器也可用。

步伐侦测功能会在每次识别出一步时生成一个中断。如果在一定时间段内检测到了至少一步, 即可生成中断, 而不是每次识别出一步才生成中断。该时间段通过将高级配置寄存器中的STEP_COUNT_DELTA_A寄存器的位[0:7]置为一个高于00h的值来定义 (STEP_COUNT_DELTA_A寄存器的值的1 LSB对应于1.6384秒)。

在随机行走事件中，须检测到连续7步才能生成第一个中断（去抖动功能），以避免出现错误步数检测。

可以通过高级配置寄存器中的寄存器PEDO_DEB_REG_A的DEB_STEP[2:0]位来修改去抖动步数。1LSB对应1步。去抖动功能在一段时间（去抖动时间）后重启，并可通过高级配置寄存器中的寄存器PEDO_DEB_REG_A的DEB_TIME[4:0]位来进行修改。1LSB对应于80ms，默认值 = 13 (13 *80 = 1040 ms)。

通过将CTRL5_A寄存器的INT2_STEP位置为1，可将该中断信号驱动至INT2_XL中断引脚上；还可通过读取FUNC_CK_GATE_A寄存器的STEP_DETECT位对其进行检查。

使能计步功能后，计步器显示算法检测到的步数。步数由STEP_COUNTER_H_A和STEP_COUNTER_L_A寄存器级联给出，表示为一个16位无符号数字。当加速度计被配置为掉电或计步器禁用时，步数不会复位至0；可通过将STEP_COUNTER_MINTHS_A寄存器的RST_NSTEP位置为1来将其复位至0。

通过将FIFO_CTRL_A (25h) 的INT2_STEP_COUNT_OV位（位#4）置为1，可以将计步器溢出条件（计数达到2216）驱动到INT2_XL引脚。

如果锁存模式禁用（CTRL3_A的LIR位置为0），则计步功能所产生的中断信号为脉冲的：中断引脚上观测到的脉冲持续时间约为60 μ s。

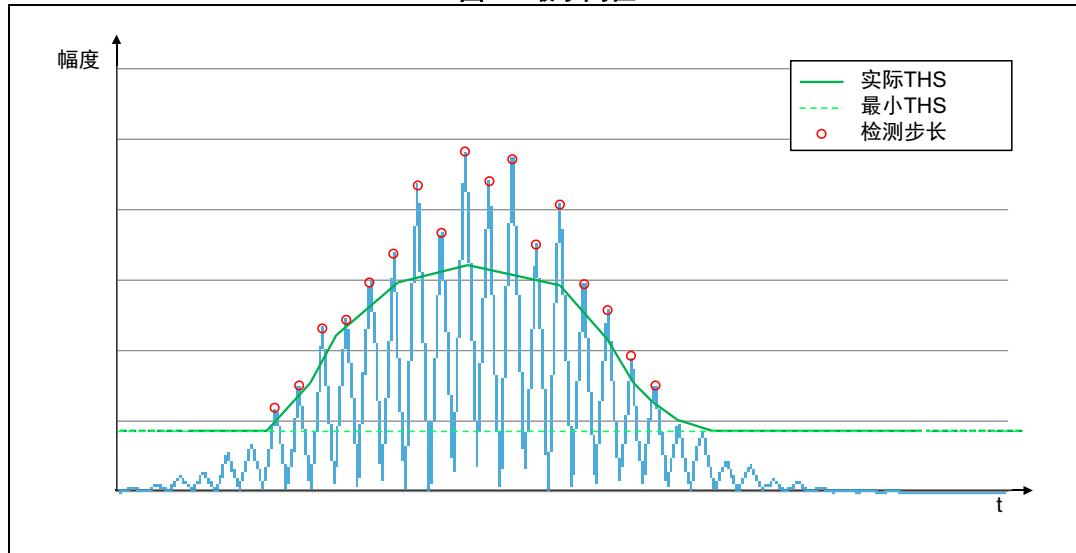
如果锁存模式使能（CTRL3_A的LIR位置为1），且中断信号被驱动至中断引脚，则当进行一步时，对FUNC_CK_GATE_A寄存器的读取会将两个引脚上的请求清除，器件准备好识别下一步。如果锁存模式使能但是该事件未驱动至中断引脚，那么FUNC_CK_GATE_A寄存器的STEP_DETECT位是脉冲的，且持续时间为1/25 Hz。

默认情况下，计步器工作的满量程为2 g，与配置的器件满量程无关，但假如CTRL1_A寄存器中的满量程设置为至少4 g的话，可以通过将STEP_COUNTER_MINTHS_A寄存器的PEDO4g位置为1来将其配置为4 g。

还可以设置‘最小阈值’，它是未检测到步数时步数识别阈值渐进趋向的值，并且不能下降到比它更小的值。此配置位于STEP_COUNTER_MINTHS_A寄存器的SC_MTHS[5:0]字段。这6位的1 LSB 的值取决于所选计步器满量程

(± 2 g or ± 4 g): 1 LSB = FS / 64.

图15. 最小阈值



以下是一个基本SW程序，显示如何使能计步功能：

1. 将20h写入CTRL1_A // 启动加速度计
// ODR = 25 Hz, FS = $\pm 2 g$
2. 将01h写入FUNC_CTRL_A // 使能计步算法
3. 将04h写入CTRL5_A // 步数检测器中断驱动至INT2_XL引脚

当识别出一步时，产生该中断信号，可通过读取STEP_COUNTER_H_A / STEP_COUNTER_L_A寄存器来获取步数。

14.9.2 大幅运动检测

大幅运动检测功能可用于基于位置的应用，用来接收指示用户何时改变位置的通知。

当检测到可能是由于用户位置变化引起的‘大幅运动’时，大幅运动功能会产生一个中断。要被认定为大幅运动，那么应至少运动了5步。

在LSM303AH器件中，该功能已在硬件中使用加速度计实现，并以25Hz工作，因此加速度计ODR必须设置为25 Hz或更高的值。

要启用大幅运动检测功能，FUNC_CTRL_A寄存器的SIGN_MOT_ON位必须置为1。

通过将CTRL5_A寄存器的INT2_SIG_MOT位置为1，可将大幅运动中断信号驱动至INT2_XL中断引脚上；还可通过读取FUNC_CK_GATE_A寄存器的SIG_MOT_DETECT位对其进行检查。

如果锁存模式禁用（CTRL3_A的LIR位置为0），大幅运动检测功能所产生的中断信号为脉冲的：中断引脚上观测到的脉冲持续时间约为60 μ s；FUNC_CK_GATE_A寄存器的SIG_MOT_DETECT位上观测到的脉冲持续时间为1/25 Hz。

如果锁存模式使能（CTRL3_A的LIR位置为1），且中断信号被驱动至中断引脚，则当检测到“大幅运动”时，对FUNC_CK_GATE寄存器的读取会将INT2_XL上的请求以及FUNC_CK_GATE_A寄存器的SIG_MOT_DETECT位清除，器件准备好识别下一事件。如果锁存模式使能但是中断信号未驱动至中断引脚，那么FUNC_CK_GATE_A寄存器的SIG_MOT_DETECT位是脉冲的，且持续时间固定为1/25 Hz。

用户可配置大幅运动最小检测阈值，该阈值是产生大幅运动中断前，位置改变时用户所行走的步数。为此，可以使用高级配置寄存器的SM_THS_A寄存器（34h）：

b7	b6	b5	b4	b3	b2	b1	b0
SM_THS_7	SM_THS_6	SM_THS_5	SM_THS_4	SM_THS_3	SM_THS_2	SM_THS_1	SM_THS_0

请注意，当SMD阈值（SM_THS_A（34h）寄存器的SM_THS[7:0]位）等于或高于计步器去抖动阈值（PEDO_DEB_REG_A（2Bh）高级寄存器的DEB_STEP[2:0]位）时，该SMD阈值有效。下表显示了SMD阈值的可能情况。

表21. SMD阈值

配置	SM_THS[7:0]	DEB_STEP[2:0]
计步器去抖动未激活	√	
计步器去抖动激活, SMD阈值 \geq 计步器去抖动阈值	√	
计步器去抖动激活, SMD阈值 $<$ 计步器去抖动阈值		√

当计步器去抖动激活且SMD阈值低于默认值 (= 6) 时, 必须相应地减小PEDO_DEB_REG_A寄存器的位DEB_STEP [2:0]。

注: 过度减少计步器去抖动阈值可能导致计步器步数检测报错!

下面是显示如何使能大幅运动检测功能的基本SW程序:

1. 将20h写入CTRL1_A // 启动加速度计
// ODR = 25 Hz, FS = $\pm 2 g$
2. 将02h写入FUNC_CTRL_A // 启用大幅运动检测算法
3. 设置CTRL2_A中的10h位 // 启用对高级配置寄存器的访问:
// 将寄存器21h的位#4写入1, 其他位保持不变
4. 在SM_THS[7:0]中设置阈值 // 在高级配置寄存器的reg 34h中写入阈值。// 1LSB = 1步。默认值 = 6。
5. 将00h写入CTRL2_A // 禁用对高级配置寄存器的访问:
// 将reg 3Fh的位#4写入0。
请注意, 所有位都是只读的, 第4位除外。
6. 将08h写入CTRL5_A // 大幅运动中断驱动至INT1_XL引脚

14.9.3 倾斜度检测

倾斜度检测功能支持检测何时发生活动改变（例如，当电话在前口袋中用户从坐到站或从站到坐时）：在LSM303AH器件中，可在硬件中实现它。

要启用倾斜度检测，FUNC_CTRL_A寄存器的TILT_ON位必须置为1。

如果器件配置为用于倾斜度检测，那么当器件距起始位置的倾斜角度大于35度时，会产生一个中断。起始位置定义为倾斜检测使能时器件的位置，或上一次倾斜中断产生时的器件位置。

使能此功能后，要产生第一次倾斜中断，器件距起始位置的倾斜角度应大于35度并持续至少2秒的时间。产生第一次倾斜中断后，当器件距离上次中断检测时的器件位置的倾斜角度大于35度时（不需要等待2秒），倾斜中断信号即被置为高电平。

通过将CTRL5_A寄存器的INT2_TILT位置为1，可将该中断信号驱动至INT2_XL引脚上；还可通过读取FUNC_CK_GATE_A寄存器的TILT_INT位对其进行检查。

如果锁存模式禁用（CTRL3_A的LIR位置为0），则倾斜度检测功能所产生的中断信号为脉冲的：中断引脚上观测到的脉冲持续时间约为60μs；FUNC_CK_GATE_A寄存器的TILT_INT位上观测到的脉冲持续时间为1/25 Hz。

如果锁存模式使能（CTRL3_A的LIR位置为1），且中断信号被驱动至中断引脚，那么当检测到倾斜时，对FUNC_CK_GATE_A寄存器的读取会将INT2_XL引脚上的请求以及FUNC_CK_GATE_A寄存器的TILT_INT位清零，器件准备识别下一次倾斜事件。如果锁存模式使能但是中断信号未驱动至中断引脚，那么FUNC_CK_GATE_A寄存器的TILT_INT位是脉冲的，且持续时间为1/25 Hz。

倾斜度检测功能工作于25 Hz，独立于器件ODR。

下面是显示如何使能倾斜度检测功能的基本SW程序：

1. 将20h写入CTRL1_A // 启动加速度计
// ODR = 25 Hz, FS = ±2 g
2. 将10h写入FUNC_CTRL_A // 使能倾斜度检测
3. 将10h写入CTRL5_A // 倾斜探测器中断驱动至INT1_XL引脚

15 加速度计先进先出 (FIFO) 缓冲区

为了限制主机处理器干预并便于对事件识别数据进行后处理, LSM303AH为三条加速度计输出通道X、Y和Z分别嵌入了先进先出 (FIFO) 缓冲区。

使用FIFO可使系统实现一致的节能效率, 仅当需要时才会唤醒FIFO, 并会从FIFO批量输出重要数据。

FIFO缓冲区可在五种不同模式下工作, 各个模式可确保在应用开发过程中实现高度灵活性: Bypass模式、FIFO模式、Continuous模式、Bypass-Continuous模式和Continuous-FIFO模式。

可使能可编程水印等级和FIFO_FULL事件在INT1_XL引脚上生成专用中断。

15.1 FIFO描述

FIFO缓冲区能够为每个通道存储多达256个14位的加速度采样, 或是存储多达768个加速度模计算输出 (参见[第 15.4 节: Module-to-FIFO](#)) ; 数据存储为14位的二进制补码左对齐表示, 这意味着它们必须被右移两位。

数据样本集合由6个字节 (XI、Xh、YI、Yh、ZI和Zh) 和组成, 它们会以选定的输出数据速率 (ODR) 释放到FIFO中。

新样本集合会放在第一个空闲的FIFO位置中, 缓冲区被占满后, 新样本集合会覆盖最早的价值。

表22. FIFO缓冲区完整表示 (存储了第256个样本集合)

输出寄存器	28h	29h	2Ah	2Bh	2Ch	2Dh
	XI	Xh	YI	Yh	ZI	Zh
FIFO 索引	FIFO样本集合					
FIFO(0)	XI (0)	Xh (0)	YI (0)	Yh (0)	ZI (0)	Zh (0)
FIFO(1)	XI (1)	Xh (1)	YI (1)	Yh (1)	ZI (1)	Zh (1)
FIFO(2)	XI (2)	Xh (2)	YI (2)	Yh (2)	ZI (2)	Zh (2)
FIFO(3)	XI (3)	Xh (3)	YI (3)	Yh (3)	ZI (3)	Zh (3)
...
FIFO(254)	XI (254)	Xh (254)	YI (254)	Yh (254)	ZI (254)	Zh (254)
FIFO(255)	XI (255)	Xh (255)	YI (255)	Yh (255)	ZI (255)	Zh (255)

表23. FIFO缓冲区完整表示 (存储了第257个样本集合、丢弃了第1个样本)

输出寄存器	28h	29h	2Ah	2Bh	2Ch	2Dh
	Xl	Xh	Yl	Yh	Zl	Zh
FIFO 索引	样本集合					
FIFO(0)	Xl (1)	Xh (1)	Yl (1)	Yh (1)	Zl (1)	Zh (1)
FIFO(1)	Xl (2)	Xh (2)	Yl (2)	Yh (2)	Zl (2)	Zh (2)
FIFO(2)	Xl (3)	Xh (3)	Yl (3)	Yh (3)	Zl (3)	Zh (3)
FIFO(3)	Xl (4)	Xh (4)	Yl (4)	Yh (4)	Zl (4)	Zh (4)
...
FIFO(255)	Xl (256)	Xh (256)	Yl (256)	Yh (256)	Zl (256)	Zh (256)

表 22: FIFO缓冲区完整表示 (存储了第256个样本集合) 表示的是存储了256个样本时 FIFO已满的状态, 而 *表 23: FIFO缓冲区完整表示 (存储了第257个样本集合、丢弃了第1个样本)* 表示下一步, 当第257个采样插入到FIFO中, 同时第1个采样被覆盖。新的最早样本集合在输出寄存器中可用。

如果FIFO已使能, 并且所处模式不是Bypass模式, LSM303AH加速度计输出寄存器 (28h到2Dh) 始终会包含最早的FIFO样本集合。

15.2 FIFO寄存器

FIFO缓冲区由四个不同的加速度计寄存器进行管理，其中两个寄存器可使能并配置FIFO特性，另外两个寄存器会提供关于缓冲区状态的信息。

一些其他寄存器用于传送引脚上的FIFO事件以中断应用处理器。这些在[第 15.3 节：FIFO中断](#)中进行讨论。

15.2.1 FIFO_CTRL_A寄存器 (25h)

FIFO_CTRL_A寄存器包含有设置FIFO的模式。默认复位时，FIFO模式为Bypass，表示关闭；只要模式设置为Bypass以外的其他模式，FIFO就会启用并开始存储采样值（或模值）。

表24. FIFO_CTRL_A寄存器

b7	b6	b5	b4	b3	b2	b1	b0
FMODE2	FMODE1	FMODE0	INT2_STEP_COUNT_OV	MODULE_TO_FIFO	0	0	IF_CS_PU_DIS

FMODE[2:0]位可选择FIFO缓冲区行为：

1. FMODE[2:0] = 000b: Bypass模式 (FIFO关闭)
2. FMODE[2:0] = 001b: FIFO模式
3. FMODE[2:0] = 011b: Continuous-FIFO模式
4. FMODE[2:0] = 100b: Bypass-Continuous模式
5. FMODE[2:0] = 110b: Continuous模式

MODULE_TO_FIFO使X/Y/Z采样 ($\sqrt{x^2+y^2+z^2}$) 的模值存入FIFO中，而非存入采样值本身。请注意，FUNC_CTRL_A寄存器的MODULE_ON位必须启用。当MODULE_ON位置为1时，嵌入功能（计步、倾斜和大幅运动检测）不可用。

15.2.2 FIFO_THS_A寄存器 (2Eh)

该寄存器可以用来设置FIFO阈值等级。

表25. FIFO_THS_A寄存器

b7	b6	b5	b4	b3	b2	b1	b0
FTH7	FTH6	FTH5	FTH4	FTH3	FTH2	FTH1	FTH0

FTH[7:0]位定义水印等级；当FIFO内容大于或等于此值时，FIFO_SRC_A寄存器中的FTH位置为“1”。

15.2.3 FIFO_SRC_A (2Fh)

该寄存器每个ODR会更新一次，会提供关于FIFO缓冲区状态的信息。

表26. FIFO_SRC_A寄存器

b7	b6	b5	b4	b3	b2	b1	b0
FTH	FIFO_OVR	DIFF8	0	0	0	0	0

- 如果FIFO内容超过水印等级，FTH位会置为高电平。此标志可发送至引脚（参见第 15.3 节：[FIFO中断](#)）。
- 在FIFO缓冲区满后，重写第一个采样时，FIFO_OVR位置为高电平。这意味着FIFO缓冲区包含256个未读采样。第一个样本集合已被读取时，FIFO_OVR位会复位。
- DIFF8位（或FIFO_FULL位）与位FIFO_SAMPLES（DIFF[7:0]）一起使用，可提供用了多少FIFO空间的信息（00000000b表示FIFO为空，10000000b表示FIFO已满）。此标志可发送至引脚（参见第 15.3 节：[FIFO中断](#)）。

寄存器内容会与FIFO写操作和读操作同步更新。

表27. FIFO_SRC_A特性（假定FTH[7:0] = 15）

FTH	DIFF8 (FIFO_FULL)	FIFO_OVR	DIFF[8:0]	未读FIFO样本	时序
0	0	0	000000000	0	t0
0	0	0	000000001	1	t0 + 1/ODR
0	0	0	000000010	2	t0 + 2/ODR
...
0	0	0	000001110	14	t0 + 14/ODR
1	0	0	000001111	15	t0 + 15/ODR
...
1	0	0	011111111	255	t0 + 255/ODR
1	1	0	100000000	256	t0 + 256/ODR
1	1	1	100000000	256	t0 + 257/ODR

15.2.4 FIFO_SAMPLES_A (30h)

该寄存器的内容与FIFO_SRC_A寄存器 (DIFF [7:0]) 的DIFF8位一起使用, 可提供用了多少FIFO空间的信息 (00000000b表示FIFO为空, 10000000b表示FIFO已满)。

表28. FIFO_SAMPLES_A寄存器

b7	b6	b5	b4	b3	b2	b1	b0
DIFF7	DIFF6	DIFF5	DIFF4	DIFF3	DIFF2	DIFF1	DIFF0

15.3 FIFO中断

有两个特定的FIFO事件可以传输到引脚, 以中断主处理器: FIFO阈值和FIFO已满。

第三个FIFO事件, FIFO_OVR, 不会被传送到引脚, 而是通过读取FIFO_SRC_A寄存器的相应位来进行轮询。

15.3.1 FIFO阈值

FIFO阈值是可用于生成特定中断的可配置功能, 可用于确定FIFO缓冲区何时包含的样本数至少为定义为阈值等级的数目。用户可以使用FIFO_THS_A寄存器中的FTH [7:0]字段在0到255之间的范围内选择所需的等级。

如果FIFO (DIFF [8:0]) 中的条目数大于或等于FTH [7:0]中编程的值, 则FIFO_SRC_A寄存器中的FTH位置为高电平。

DIFF[8:0]会以ODR频率增加一步, 每次由用户执行样本集合读取操作时, DIFF[8:0]会减小一步。

阈值标志 (FTH) 可以传送到INT1_XL和INT2_XL引脚, 为应用处理器提供专用中断, 使每次中断之间功耗更少。CTRL4_A寄存器的INT1_FTH位和CTRL5_A寄存器的INT2_FTH位专门用于此目的。

15.3.2 FIFO已满

只要FIFO已满, 就可以配置器件, 使之产生一个中断。为此, 只需将WAKE_UP_DUR_A寄存器的INT1_FSS7位置为“1”即可。为避免丢失样本, FIFO读取操作必须在1个ODR窗口内开始并完成。

15.4 Module-to-FIFO

如果模计算开启 (FUNC_CTRL_A寄存器的MODULE_ON位为“1”)，且FIFO_CTRL_A的MODULE_TO_FIFO位置为“1”，则FIFO缓冲区中不是加速度采样，而是其模计算值 (sqrt (x2+y2+z2))。

由于模大小为14位，所以FIFO缓冲区最多可能包含768个模值 (256 * 3)。当模值存储在FIFO中时，FIFO_THS_A中设置的阈值和FIFO_SAMPLES_A寄存器中可用的存储数据数量由1LSb = 3样本表示。

以下是使FIFO中存入加速度采样模值的简单过程：

1. 在FUNC_CTRL_A (3Fh) 中将MODULE_ON置为1，使能模计算
2. 将FIFO_CTRL_A (25h) 的MODULE_TO_FIFO位置为“1”，使模值保存在FIFO缓冲区中
3. 在其中一个模式下使能FIFO (参见第 15.5节: FIFO模式)

15.5 FIFO模式

通过FIFO_CTRL_A寄存器的FMODE[2:0]字段，LSM303AH FIFO缓冲器可配置为五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、Continuous、Bypass-Continuous和Continuous-FIFO模式。

15.5.1 Bypass模式

启用Bypass模式后，FIFO不可运行：缓冲区内容会被清空、输出寄存器 (0x28到0x2D) 会冻结为最后载入的值，在选择其他模式之前，FIFO缓冲区会保持空白状态。

可以通过在FIFO_CTRL_A寄存器中将FMODE [2:0]字段置为000b来激活Bypass模式。

当在不同模式下工作时，要停止和复位FIFO缓冲器，必须使用Bypass模式。请注意，将FIFO缓冲区置于Bypass模式会清除整个缓冲区的内容。

15.5.2 FIFO模式

FIFO模式中，缓冲区继续填充直至充满（存储了256个样本集合）。一旦FIFO_OVR标志变为‘1’，FIFO就停止采集数据，其内容保持不变，直至选择不同的模式。

可以通过在FIFO_CTRL_A寄存器中将FMODE [2:0]字段置为001b来激活FIFO模式。

选择该模式后，FIFO会开始进行数据采集，DIFF[8:0]也会根据存储的样本数发生变化。程序结束时，FIFO_OVR标志上升为1，然后可以恢复数据，从输出寄存器读取256个样本集合。由于在FIFO模式下数据采集已停止，并且不存在覆盖已获取数据的风险，因此通信速度并不重要。重新启动FIFO模式之前，请务必在读取程序之后退出Bypass模式。

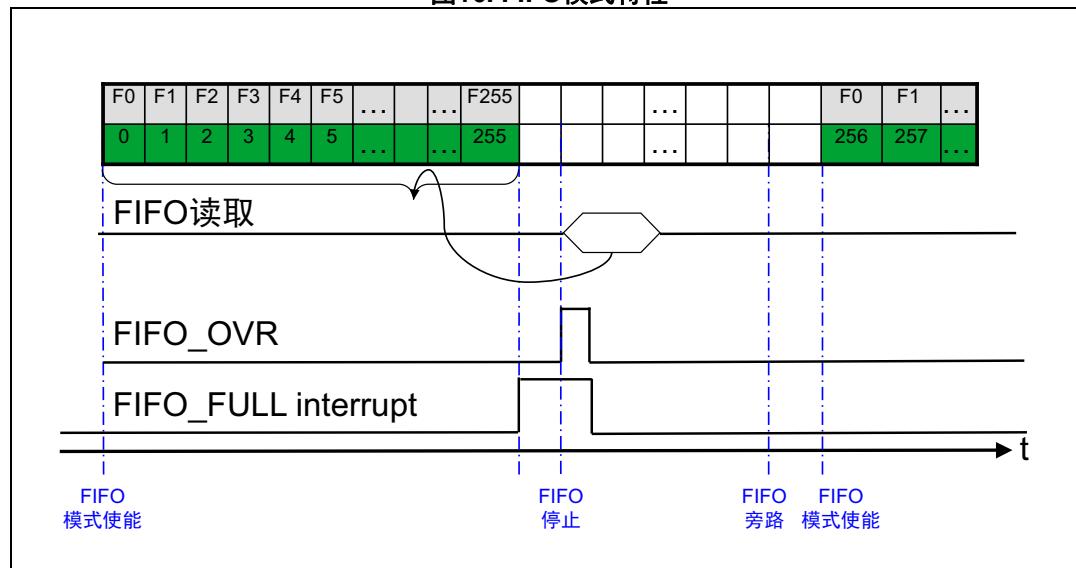
要尽快处理FIFO_FULL事件，建议将其传送到引脚，以产生一个中断，该中断随后会由一个特定的处理程序进行管理：

1. 将INT1_FSS7置为‘1’：使能FIFO_FULL中断
2. 设置FMODE[2:0] = 001b：使能FIFO模式

当生成了FIFO_FULL中断或FIFO_OVR位为高电平时（轮询模式）：

1. 从加速度计输出寄存器读取数据

图16. FIFO模式特性



如图 16: FIFO 模式特性 中所示，当使能了 FIFO 模式，缓冲区会开始采集数据，并会以所选输出数据速率填入全部 256 个位置（从 F0 到 F255）。缓冲区已满后，下一采样会进入并重写缓冲区，FIFO_OVR 位会变为高电平，数据采集会永久停止；用户可随时读取 FIFO 内容，因为在选择 Bypass 模式之前，FIFO 缓冲区的内容保持不变。读取程序可以在由 FIFO_FULL 条件 (DIFF8) 触发的中断处理程序内执行，它包括 256 个 6 字节样本集合（共 1236 字节），会从 FIFO 中存储的最早样本 (F0) 开始获取数据。第一个样本集合已被读取时，FIFO_OVR 位会复位。Bypass 模式设置会复位 FIFO 并允许用户再次使能 FIFO 模式。

15.5.3 连续模式

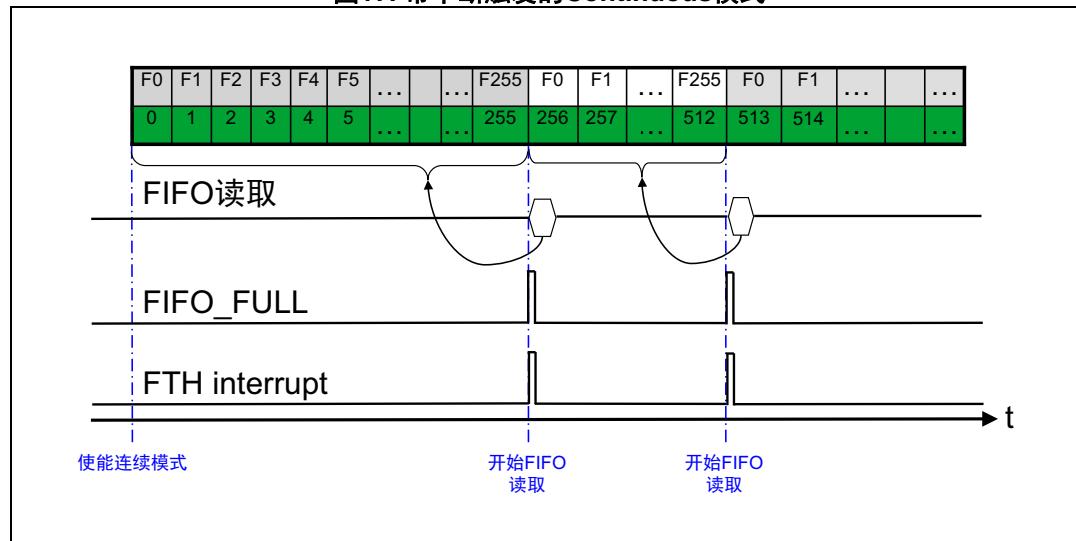
在Continuous模式下, FIFO会继续填入数据, 如果缓冲区已满, FIFO索引会从头开始使用, 较早的数据会被当前数据替代。最早先的数据继续被覆盖, 直至读取操作释放了FIFO插槽。为了使FIFO位置的释放速度快于获得新数据的速度, 主机处理器的读取速度至关重要。Bypass配置下FMODE[2:0]用于停止该模式。

按照以下步骤进行FIFO Continuous配置, 其中设置一个阈值来产生中断, 以触发应用处理器进行读取:

1. 将FTH[7:0]置为255。
2. 将INT2_FTH置为'1': 使能FIFO阈值中断
3. 将FIFO_CTRL_A寄存器 (25h) 中的FMODE[2:0]字段设为110b可激活Continuous模式。

当产生FTH中断时, 从加速度计输出寄存器读取数据。

图17. 带中断触发的Continuous模式



如图 17 所示, 当使能了Continuous模式时, FIFO缓冲区会以选定的输出数据速率持续填入数据 (从F0到F255)。当缓冲区满时, FTH中断 (以及FIFO_SRC_A (2Fh) 中的DIFF8位指示的FIFO_FULL条件, 该条件也可用于触发中断) 变为高电平, 应用处理器会立即读取所有FIFO样本 (256 * 6字节), 以免丢失数据并限制主机处理器的干预, 从而提高系统效率。关于FIFO读取速度的更多详细信息, 请参见第 15.6节: 从FIFO恢复数据。

读取命令发送到器件后, 输出寄存器内容会移动到SPI/I²C寄存器, 当前最早的FIFO值会移入输出寄存器, 以执行下一次读取操作。

15.5.4 连续-FIFO 模式

此模式是先前所述的连续和FIFO模式的组合。在Continuous-FIFO模式下，FIFO缓冲区会在Continuous模式下开始工作，并会在发生选定的中断（例如，唤醒、自由落体、点击……）后切换为FIFO模式。

可使用此模式来分析生成中断的样本历史；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。

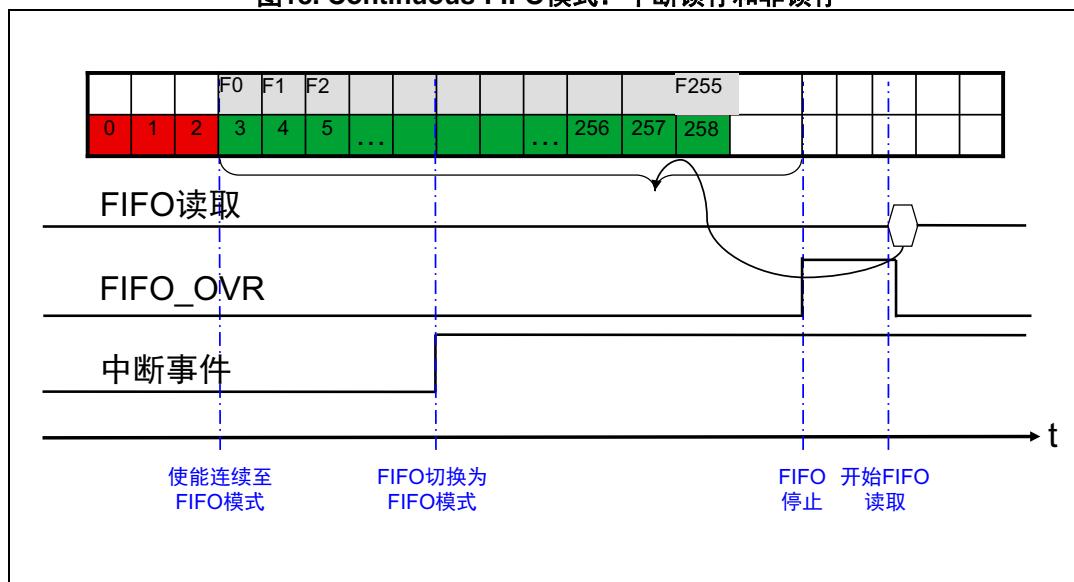
请按照以下步骤配置Continuous-FIFO模式：

1. 按照[第 14 节：加速度计中断生成和嵌入功能](#)中说明配置所需的中断发生器（确保锁定）。
2. 将FIFO_CTRL_A寄存器（25h）中的FMODE[2:0]字段设为011b可激活Continuous-to-FIFO模式。

注：当发生所请求的事件时，当且仅当事件标志被发送到INT1_XL或INT2_XL引脚时，才会触发FIFO模式更改。

在Continuous模式下，FIFO缓冲区持续填充；当请求的事件发生时，FIFO模式发生变化；然后，一旦缓冲器变满，FIFO_OVR位置为高电平，下一个采样将会覆盖早先数据，FIFO停止采集数据（见[图 18](#)）。

图18. Continuous-FIFO模式：中断锁存和非锁存



15.5.5 Bypass-to-Continuous模式

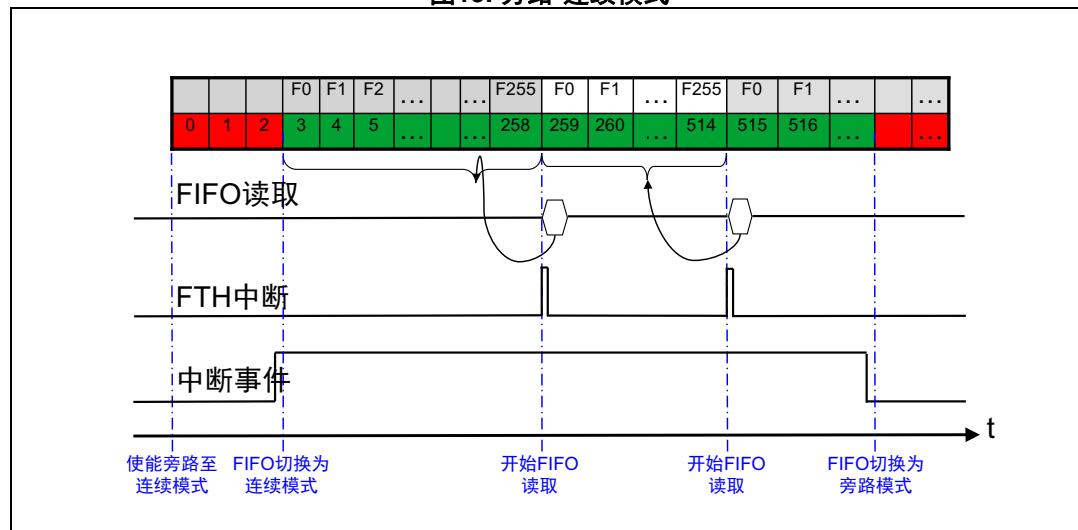
此模式是先前所述的Bypass模式和连续模式的组合。在Bypass-Continuous模式下，FIFO缓冲区会开始处于Bypass模式，在发生选定的中断（例如，唤醒、自由落体、点击……）后切换为Continuous模式。

请按照以下步骤配置Bypass-Continuous模式：

1. 按照[第 14 节：加速度计中断生成和嵌入功能](#)中说明配置所需的中断发生器（确保锁定）。
2. 将FTH[7:0]置为255。
3. 将INT2_FTH置为“1”：使能FIFO阈值中断
4. 将FIFO_CTRL_XL寄存器（25h）中的FMODE[2:0]字段设为100b可激活Bypass-to-Continuous模式。

当产生FTH中断时，从加速度计输出寄存器读取数据。

图19. 旁路-连续模式



如图 19 中所示，FIFO 初始处于 Bypass 模式，因此无采样进入 FIFO 缓冲区。一旦发生事件（例如唤醒或自由落体事件），则 FIFO 切换到 Continuous 模式并开始以所配置的数据速率存储样本。当达到所编程的阈值时，FTH 中断变为高电平，应用处理器会尽快开始读取所有 FIFO 采样（256 * 6 字节），以免丢失数据。

如果 FIFO_OVR 标志被置位，那么一旦读取第一个 FIFO 组，该标志就会变为 0，为新数据创建空间。由于 FIFO 仍处于 Continuous 模式，因此，FIFO 最终会再次达到阈值，重复此情况。

最后，要么中断事件被清除，要么 FIFO 直接进入 Bypass 模式，然后停止采集数据。

15.6 从FIFO恢复数据

当FIFO模式不是Bypass时，读取输出寄存器（28h至2Dh）会返回早先的FIFO样本集合。

读取输出寄存器时，其内容会移至SPI/I²C输出缓冲区。理想地，FIFO插槽会向上移动一格，以便释放空间接收新的采样，并且输出寄存器载入FIFO缓冲器中存储的当前最旧的值。

通过从加速度计输出寄存器执行256次读取操作，可以重新取回整个FIFO内容。无论功耗模式如何，存储在FIFO中的数据大小始终是14位。每隔一个读取操作都会返回相同的最终值，直到FIFO缓冲区中有一个新的样本集可用。

为了提高应用的灵活性，可使用每种读取字节组合从FIFO重新获取数据（例如：1536次单字节读取，256次6字节读取，1次1536字节的多字节读取等）。

建议以1536字节的多字节读取（6个输出寄存器乘以256个插槽）来读取所有FIFO插槽。

为了减少主机和从机之间的通信，可以通过将CTRL2_A寄存器的IF_ADD_INC位置为“1”，使器件的读取地址自动递增；当到达寄存器0x2D时，器件回滚到0x28。

I²C速度低于SPI，它需要大约29个时钟脉冲才能开始通信（开始、从地址，寄存器地址+写入、重新启动、寄存器地址+读取），并且每个字节读取都需要额外9个时钟脉冲（总共为83个时钟脉冲）。因此，在使用标准I²C模式的情况下（最大速率为100 kHz），单个样本集读取需要耗费830 μ s，总FIFO下载需要耗费约138.53 ms（29 + 9 * 1536时钟脉冲）。

在SPI的情况下，相反，只需要在开始启动时耗费9个时钟脉冲（r / w +寄存器地址），再加上额外的每个字节读取耗费8个时钟脉冲。使用2 MHz时钟时，单个采样集读取将需要28.5 μ s，总FIFO下载大约需要6.15 ms。

如果按照此建议，使用标准I²C（100 kHz），那么全部读取FIFO（138.53 ms）将需要耗费14/ODR，ODR为100Hz。使用SPI@2MHz（器件支持的最大速率是10MHz）时，全部读取FIFO将需要耗费1/ODR，ODR为100 Hz。

因此，为了不丢失样本，应用将在FIFO满之前读取样本，设置阈值并使用FTH中断（参见章节[第 15.3 节：FIFO中断](#)）。

表29. 示例：ODR阈值功能

ODR (Hz)	FTH_THS (I ² C @ 100 kHz)	FTH_THS (I ² C @ 400 kHz)	FTH_THS (SPI @ 2 MHz)
50	36	147	256
100	17	73	256
200	8	36	208
400	4	17	103
800	1	8	51
1600	-	4	25

16 加速度计自检

当启用加速度计自检时，致动力会施加到传感器，致使传感器的可移动部分发生挠曲，并产生加速度。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满量程。

当CTRL3_A寄存器的ST[2:1]位被设定为00b时，加速度计自检功能关闭；当ST[2:1]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，传感器输出电平由作用在传感器上的加速度和静电测试力的代数和给出。

该过程包括：

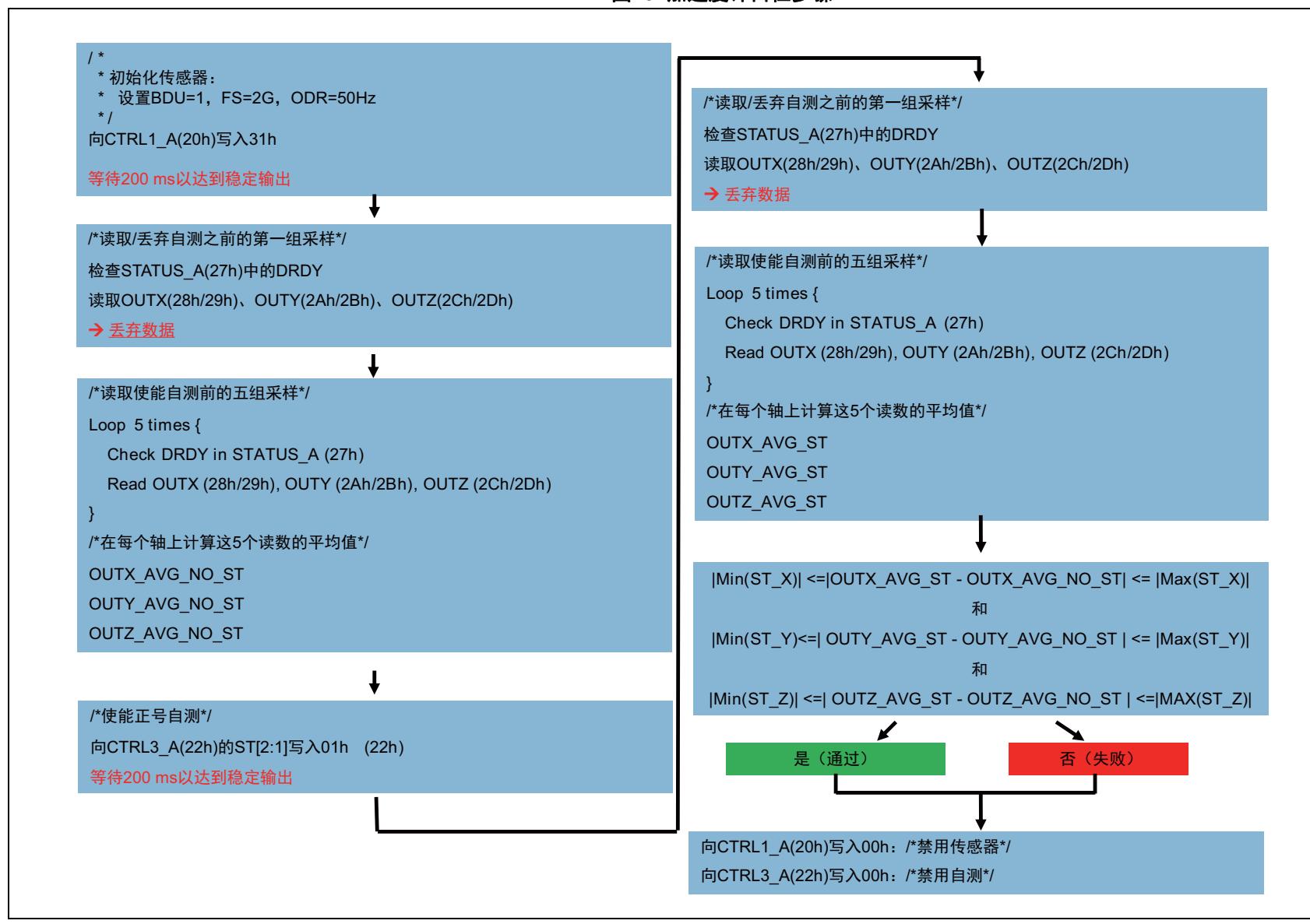
1. 使能加速度计
2. 启动自检之前，对五个采样取平均
3. 启动自检之后，对五个采样取平均
4. 计算每个轴的模差，并验证它是否落在给定的范围内。数据表中给出了最小和最大值。

完整的加速度计自检过程如图 20 中所示。

注：

自检过程中保持器件静止。

图20. 加速度计自检步骤



17 温度传感器

LSM303AH具有内部温度传感器，适用于环境温度测量。

如果加速度计传感器处于掉电模式，则温度传感器关闭并显示最后的测量值。

温度传感器的输出数据速率固定为12.5 Hz。

温度数据由OUT_T_A寄存器给出，以二进制补码的格式表示为一个8位的数字，其灵敏度为+1 LSB/°C。输出零值对应于25 °C ±15 °C。

17.1 温度数据计算示例

[表 30: 输出数据寄存器内容 vs. 温度](#)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30. 输出数据寄存器内容 vs. 温度

温度值	OUT_T_A (26h)
23 °C	FEh
24 °C	FFh
25 °C	00h
27 °C	02h

18 版本历史

表31. 文档版本历史

日期	版本	变更
2018年1月19日	1	初始版本

表32. 中文文档版本历史

日期	版本	变更
2018年9月19日	1	中文初始版本

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。本文档的中文版本为英文版本的翻译件，仅供参考之用；若中文版本与英文版本有任何冲突或不一致，则以英文版本为准。

© 2018 STMicroelectronics - 保留所有权利