ICM-42607x and ICM-42670x

APEX Motion

Functions: Description and Usage

InvenSense, a TDK Group Company
1745 Technology Drive, San Jose, CA 95110 U.S.A
+1(408) 988–7339
invensense.tdk.com

TABLE OF CONTENTS

1 APEX MOTION OVERVIEW

APEX (**A**dvanced **P**edometer and **E**vent Detection – Ne**X**t Generation) Motion is a digital block inside the ICM-42607x AND ICM-42670x accelerometer family of Inertial Measurement Units. APEX processes accelerometer data, extracts measurements, and asserts motion-specific interrupts. APEX features use accelerometer data, not gyroscope data.

Each APEX feature has its own set of configuration bits, and the features can run concurrently. Most APEX features assert an interrupt after a motion event (Significant Motion Detection for example asserts an interrupt but does not provide any measurements), while other features will extract data and provide it to the user (Pedometer counts steps and can also assert interrupts).

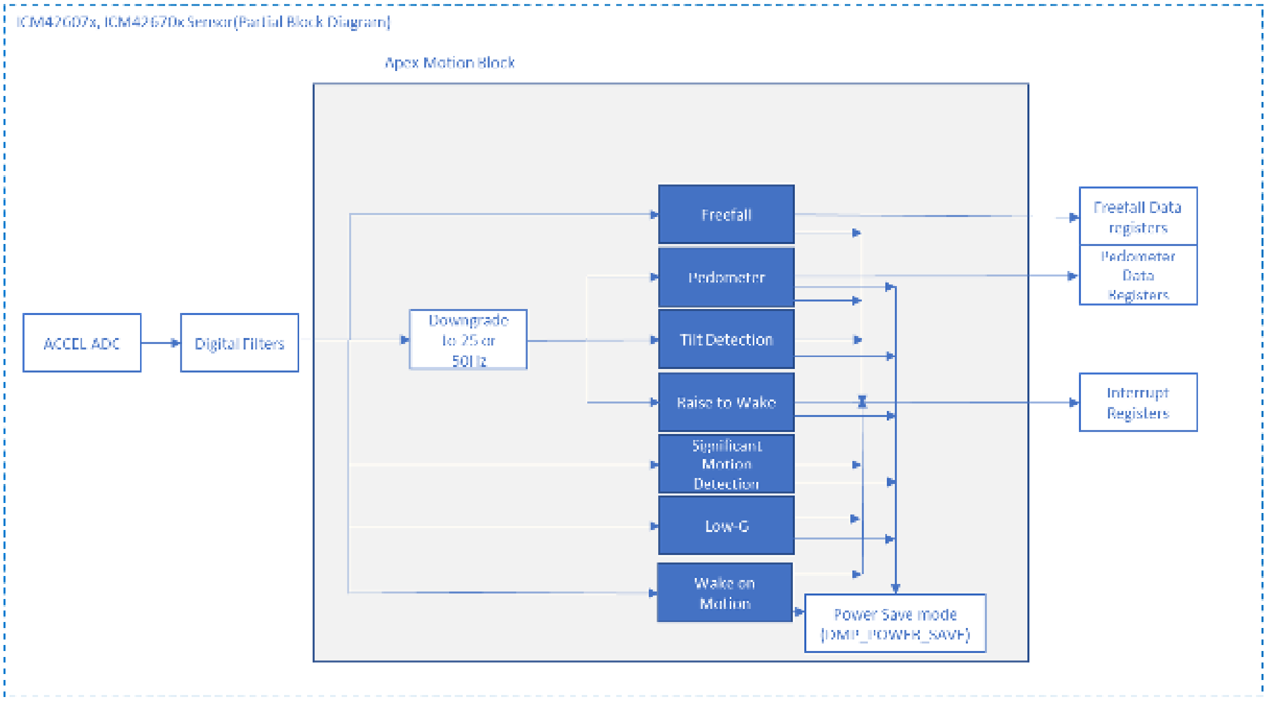


Figure 1. APEX simplified block diagram.

The APEX features in ICM-42607x and ICM-42670x accelerometer consists of the following features:

- **Tilt**: triggers an interrupt when the Tilt angle exceeds 35 degrees for more than a programmable time.
- **Low-G**: triggers an interrupt when absolute value of accelerometer combined axis falls below a programmable threshold and stays below the threshold for a programmable time.
- **Freefall**: triggers an interrupt when device freefall is detected and outputs freefall duration.
- **Pedometer**: tracks Step count and triggers a Step Detect Interrupt.
- **Wake on Motion**: detects a motion through accel samples exceeding a programmable threshold with respect to a reference value. This motion event can be used to enable chip operation from sleep mode.
- **Significant Motion Detector**: Detects significant motion based on accelerometer data.

2 REGISTER CONFIGURATION REQUIREMENTS AND OPTIONS

APEX behavior is impacted (directly or indirectly) by several ICM-426xx register settings. Accelerometer configuration, digital filtering options, as well as software/hardware interrupt routing will change the conditions under which an interrupt is asserted.

2.1 APEX ODR SUPPORT

The APEX algorithms are designed to work together with the Accel, for a variety of ODR settings. However, there is a minimum ODR required for each algorithm. Table 1 shows the relationship between the available Accel ODRs and the operation of the APEX algorithms. Note that in order to allow more flexible operation, APEX algorithms ODR is set separately from Accel ODR through the dedicated DMP_ODR field.

Table 1 shows how DMP_ODR should be configured in relation to the Accel_ODR and the performance expectation.

Accel ODR	DMP ODR	LowG	FreeFall	Tilt	Pedometer	SMD
25 Hz	0	Low Power		Low Power	Low Power	
50 Hz	2	Normal		High Performance	Normal	Normal
100 Hz	3	Normal		High Performance (50 Hz)	Normal (50 Hz)	Normal (50 Hz)
400 Hz	1	High Performance	High Performance	High Performance (50 Hz)		Normal (50 Hz)

Table 1. APEX ODR support

Wake On Motion(WoM): HW algorithm is supported at all ODR.

2.2 DMP ODR

- APEX algorithms require accel to run at least at the same ODR as DMP_ODR.
- DMP ODR should not be changed on the fly. The following sequence should be followed for changing the DMP ODR:
  1. Disable Pedometer, Tilt, SMD, and Freefall if they are enabled
  2. Change DMP ODR
  3. Set DMPInit_En for one cycle
  4. Poll DMPInit_En till it is reset, to know when Init is completed.
  5. Enable APEX features of interest

2.3 HOW TO SWITCH BANK TO MREG1 FOR READ AND WRITE REGISTERS

To read the registers

1. Set the following register if MSB 8 bits of address is not 0.

Field Name	Register Name	Register Address	Bank	Value (hex)	Configuration
blk_sel_r	BLK_SEL_R	0x7C	Bank 0	MSB 8 bit of address	Setting Block of register access

Set the following register for

Field Name	Register Name	Register Address	Bank	Value (hex)	Configuration
maddr_r	MADDR_R	0x7D	Bank 0	LSB 8 Bit of Address	Register Address

2. Sleep for 10 µs
3. Read M_R 0x7E register for bytes to be read
4. Sleep for 10 µs
5. Set BLK_SEL_R to 0 is MSB 8 bit of address is not 0.

Field Name	Register Name	Register Address	Bank	Value (hex)	Configuration
blk_sel_r	BLK_SEL_R	0x7C	Bank 0	MSB 8 bit of address	Setting Block of register access

To write the registers

1. Set the following register if MSB 8 bits of address is not 0.

Field Name	Register Name	Register Address	Bank	Value (hex)	Configuration
blk_sel_w	BLK_SEL_W	0x79	Bank 0	MSB 8 bit of address	Setting Block of register access

2. Set the following register for

Field Name	Register Name	Register Address	Bank	Value (hex)	Configuration
maddr_w	MADDR_W	0x7A	Bank 0	LSB 8 Bit of Address	Register Address

3. Write data to M_W 0x7B register for bytes to be written
4. Sleep for 10 µs
5. Set BLK_SEL_W to 0 is MSB 8 bit of address is not 0.

Field Name	Register Name	Register Address	Bank	Value (hex)	Configuration
blk_sel_w	BLK_SEL_W	0x79	Bank 0	MSB 8 bit of address	Setting Block of register access

3 PEDOMETER

Pedometer measures three quantities: it counts steps, measures cadence (step-rate), and classifies motion activity as running/walking/"unknown" movement. Pedometer can assert an interrupt on any new step event or if the step counter overflows its 16-bit register space.

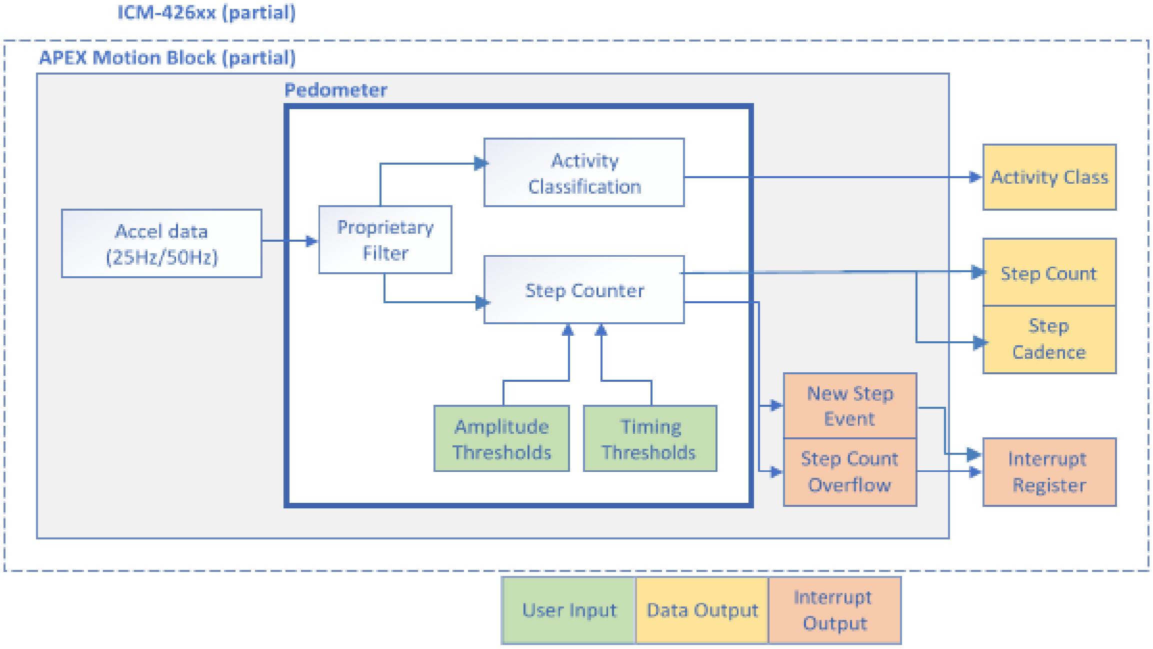


Figure 2. Pedometer

Accelerometer data is downsampled, passed through a proprietary filter, and is then compared against user-programmable timing/signal thresholds that identify new step events. Pedometer has three user-readable output fields and two interrupt output bits.

The Pedometer has a default configuration to process accelerometer data and count steps. If more (or less) step sensitivity is desired, the amplitude/timing registers can be adjusted. Pedometer processes downsampled accelerometer data at 25 Hz or 50 Hz (per the DMP_ODR field).

3.1 PEDOMETER FIELDS

The amplitude/energy/timing thresholds are knobs that tune the pedometer sensitivity higher or lower. The default power-on configuration is appropriate for watch/phone/pocket/bag sensor locations, and **InvenSense recommends using the default Pedometer configuration**.

Input amplitude fields:

- *PED_AMP_TH_SEL*: the acceleration peak amplitude threshold above which the signal is considered to correspond to a valid step.
- *LOW_ENERGY_AMP_TH_SEL*: threshold to select a valid step depending on the energy of acceleration signal. This parameter helps improve the step detection during slow walk use case.
- *PEDO_HI_EN_TH_SEL*: the threshold on the signal's energy contained in the band of frequency associated to walk/run motion. This threshold is used to distinguish between acceleration generated by step motion activities from other acceleration sources. This threshold helps reduce the false detection.

- *SENSITIVITY_MODE*: default is 0. If set to "1", provides greater sensitivity for a "slow walk" use-case, at the expense of higher false-positives during car/train travel (when the accelerometer shows motion but the user is not taking steps). Setting to a 1 is not recommended if motion-rejection is a key performance metric.

Input timing fields:

- *PED_STEP_CNT_TH_SEL*: the minimum number of steps needed to be initially detected before starting to increment the step count in real time. This minimum count of steps serves to reduce false positives (peaks that don't correspond to real steps.)
- *PED_STEP_DET_TH_SEL*: the minimum number of steps needed to be initially detected before starting to report step events.
- *PED_SB_TIMER_TH_SEL*: the maximum allowed time between 2 steps. If this amount of time elapses without detecting a new step, the step count buffer is reset to 0. This parameter serves to reduce false positives.

Output data fields:

- *STEP_CNT*: the accumulated number of steps (step count) during walk and/or run activities.
- *STEP_CADENCE*: averaged number of samples between consecutive steps, scaled by 4x. To obtain user's steps per minute (SPM), perform the following calculation: *SPM = ODR/(STEP_CADENCE/4)*60*. Example calculation: if STEP_CADENCE=100, and ODR=50Hz, 50/(100/4)*60 = 120 steps/minute by the user.
- *ACTIVITY_CLASS*: the detected user activity. 0: unknown, 1: walk, 2: run.

Field Name	Register Name	Register Address	Bank	Configuration
step_cnt	APEX_DATA0	0x31 (7:0)	Bank 0	LSB Step Count
step_cnt	APEX_DATA1	0x32 (7:0)	Bank 0	MSB Step Count
step_cadence	APEX_DATA2	0x33 (7:0)	Bank 0	Step Cadence
activity_class	APEX_DATA3	0x34(1:0)	Bank 0	0: unknown activity 1: walk 2: run

Output interrupt fields:

- *STEP_DET_INT*: interrupt asserted when a new step event is detected
- *STEP_CNT_OVF_INT*: interrupt asserted when the accumulated step count STEP_CNT overflows its 16-bit value, and is reset to 0.

Field Name	Register Name	Register Address	Bank	Value	Configuration
step_det_int	INT_STATUS3	0x3C (5)	Bank 0	0/1	0 : No event 1 : Step detection event
step_cnt_ovf_int	INT_STATUS3	0x3C (4)	Bank 0	0/1	0 : No event 1 : DMP Step Counter Overflow event

3.2 PEDOMETER INITIALIZATION PROCEDURE

Field Name	Register Name	Register Address	Bank	Value	Description
accel_odr	ACCEL_CONFIG0	0x21 (3:0)	Bank 0	10	ODR value configured 50 Hz
accel_mode	PWR_MGMT0	0x1F(1:0)	Bank 0	2	Low power mode
accel_lp_clk_sel		0x1F(7)		0	Accel LP mode uses WU oscillator clock
dmp_odr	APEX_CONFIG1	0x26h(1:0)	Bank 0	2	DMP ODR 50 Hz
Wait 1 milisecond					
low_energy_amp_th_sel	APEX_CONFIG2	0x44(7:4)	MREG1	10	Low energy amplitude threshold
pedo_amp_th_sel	APEX_CONFIG3	0x45(7:4)	MREG1	8	((30 + pedo_amp_th_sel [3:0]*4 mg) * 2^25/1000)
pedo_step_cnt_th_sel		0x45(3:0)		5	Step Count Threshold
pedo_hi_en_th_sel	APEX_CONFIG4	0x46(1:0)	MREG1	1	High Energy Threshold
pedo_sb_timer_th_sel		0x46(4:2)		4	(50 + pedo_sb_timer_th_sel*25)
pedo_step_det_th_sel		0x46(7:5)		2	Step detect threshold select
sensitivity_mode	APEX_CONFIG9	0x48(0)	MREG1	0	Sensitivity mode to 0
dmp_mem_reset_en	APEX_CONFIG0	0x25(0)	Bank 0	1	Clear DMP SRAM for APEX operation
Wait 1 milisecond					
dmp_init_en	APEX_CONFIG0	0x25(2)	Bank 0	1	Enable algorithm execution
Wait 50 miliseconds					
step_det_int1_en	INT_SOURCE6	0x2F(5)	MREG1	1	Step Detect interrupt INT1
step_cnt_ovf_int1_en		0x2F(4)		1	Step Overflow interrupt INT1
step_det_int2_en	INT_SOURCE7	0x30(5)	MREG1	1	Step Detect interrupt INT2
step_cnt_ovf_int2_en		0x30(4)		1	Step Overflow interrupt INT2
ped_enable	APEX_CONFIG1	0x26(3)	Bank 0	1	Enable Pedometer Algorithm

4 TILT DETECTION

Tilt Detection measures the angle between two accelerometer vectors: the current sample and the sample at the time of the previous tilt interrupt. Tilt Detection asserts an interrupt if it calculates a change of >35°. There is a single input parameter—Tilt Wait Time—which controls how long a valid "tilt" event must be held before an interrupt is initiated.

Large acceleration without a change in sensor orientation (relative to gravity) should not trigger a tilt event. For example, a rotation of the sensor about the observer's z-axis (direction of gravity) should not result in a tilt interrupt. A sharp turn while driving a car or a user turning around rapidly while walking should not trigger a tilt event if the rotation is about the axis of gravity.

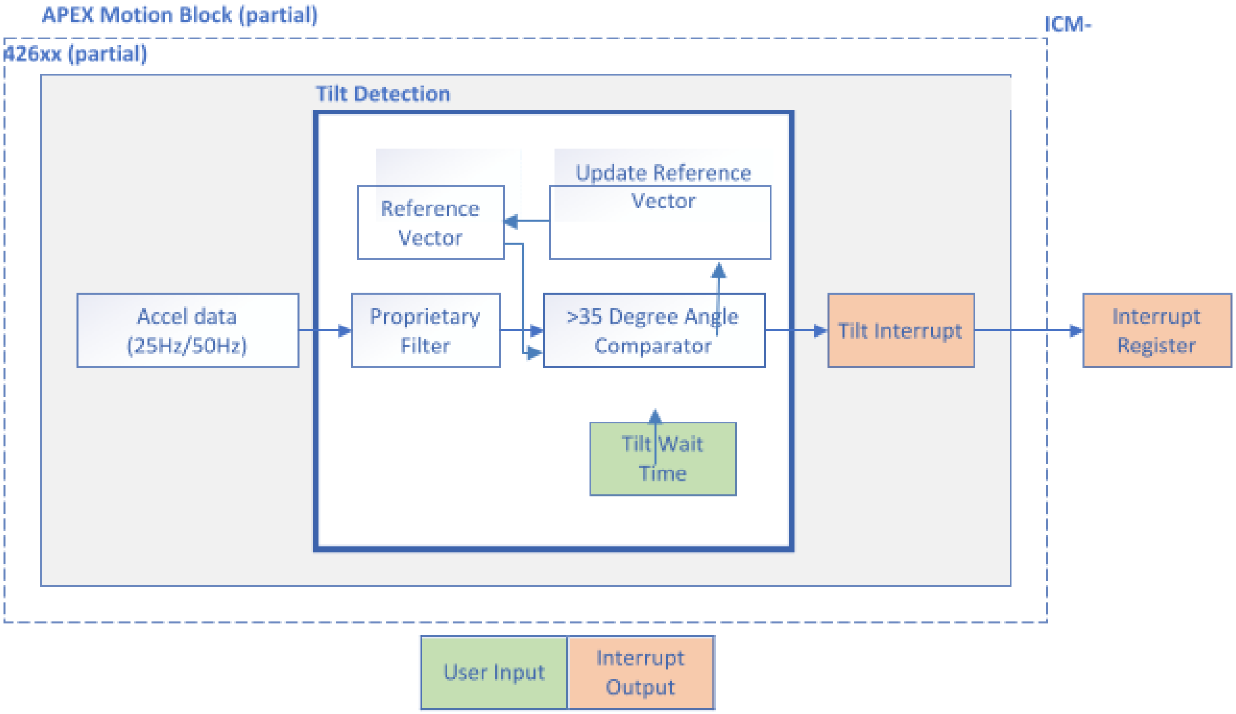


Figure 3. Tilt detection

4.1 TILT DETECTION FIELDS

Input field:

- *TILT_WAIT_TIME_SEL*: defines the polling period when the sensor checks for a tilt event. Longer polling periods result in fewer, and more delayed, interrupts.

  Note: the default time is 4 seconds.

Field Name	Register Name	Register Address	Bank	Value	Configuration
tilt_wait_time_sel	APEX_CONFIG5	0x47 (7:6)	MREG1	00 01 10 11	00 :0 seconds 01 : 2 seconds 10 : 4 seconds 11 : 6 seconds

- Tilt ODR configuration

Accel ODR	Tilt Function	Performance
below 25 Hz	tilt disabled	
25 Hz	tilt enabled	low power, reduced detection accuracy
50 Hz	tilt enabled	high performance

Table 2. Tilt ODR Table

Output interrupt field:

1. Read tilt_det bit in interrupt register

   Register INT_STATUS3[3] (Bank0@0x3C)

Field Name	Register Name	Register Address	Bank	Value	Configuration
tilt_det_int	INT_STATUS3	0x3C (3)	Bank 0	0/1	0 : No event 1 : Tilt detection event

4.2 TILT DETECTION INITIALIZATION PROCEDURE

Field Name	Register Name	Register Address	Bank	Value	Description
accel_odr	ACCEL_CONFIG0	0x21 (3:0)	Bank 0	5-11	ODR value configured
accel_mode	PWR_MGMT0	0x1F(1:0)	Bank 0	2	Low power mode
accel_lp_clk_sel		0x1F(7)		0	Accel LP mode uses WU oscillator clock
dmp_odr	APEX_CONFIG1	0x26h(1:0)	Bank 0	0 or 2	DMP ODR 25 Hz or 50 Hz
Wait 1 milisecond					
dmp_mem_reset_en	APEX_CONFIG0	0x25h(1:0)	Bank 0	1	Clear DMP SRAM for APEX operation
Wait 1 milisecond					
tilt_wait_time_sel	APEX_CONFIG5	0x47 (7:6)	MREG1	0-3	Selection of tilt wait time
Wait 1 milisecond					
dmp_init_en	APEX_CONFIG0	0x25(2)	Bank 0	1	Enable algorithm execution
Wait 50 miliseconds					
tilt_enable	APEX_CONFIG1	0x26(4)	Bank 0	1	Enable Tilt Algorithm

5 LOW-G

The DMP low-G detector triggers when absolute value of any of the accelerometer axis falls below a predefined low_peak_thres (typically around 375 mg) and stays below the threshold for a predefined low_time_thres (typically 20 ms). A hysteresis low_peak_thres_hyst (typically about 125 mg) can be added to the threshold after the initial threshold is met.

A sum mode allows the threshold to be checked against the sum of the absolute value of all three axis of the accelerometer data.

Note: low-G can't be enabled alone. It is enabled when running freefall algorithm. Thus if the user wants to run low-G only, then enabling freefall but unmasking low-G interrupt only in register INT_SOURCE7 is recommended. By doing this, only low-G interrupts are signaled to the host; FF interrupts are ignored.

5.1 LOW-G FIELDS

Input field:

- *LOWG_PEAK_TH_SEL:* this parameter defines the threshold for accel values below which algorithm considers it enters low-G.

Typical value is 17 (i.e. 563mGee).

Bitfield	Threshold (mGee)	Bitfield	Threshold (mGee)	Bitfield	Threshold (mGee)	Bitfield	Threshold (mGee)
0	31	8	281	16	531	24	781
1	63	9	313	17	563(typical)	25	813
2	94	10	344	18	594	26	844
3	125	11	375	19	625	27	875
4	156	12	406	20	656	28	906
5	188	13	438	21	688	29	938
6	219	14	469	22	719	30	969
7	250	15	500	23	750	31	1000

Table 3. Low G Peak Threshold

- *LOWG_TIME_TH_SEL:* this parameter defines the number of samples device should stay in low-G before triggering interrupt.

It is coded as: number_of_samples = lowg_time_th_sel + 1

Typical value is 4 (i.e. 5 samples).

- *LOWG_PEAK_TH _HYST_SEL:* this threshold is added to the low-G peak threshold after the initial threshold is met.

Typical value is 4 (i.e. 156mGee).

Bitfield is coded as follow:

Bitfield	Threshold(mGee)
0	31
1	63
2	94
3	125
4	156(typical)
5	188
6	219
7	250

Table 4. Low G Peak Hysterisis

Field Name	Register Name	Register Address	Bank	Configuration
lowg_peak_th_sel	APEX_CONFIG10	0x49 (7:3)	MREG1	Peak threshold from Table 3
lowg_time_th_sel	APEX_CONFIG10	0x49 (2:0)	MREG1	Parameter defines the number of samples device should stay in low-G before triggering interrupt
lowg_peak_th_hyst_sel	APEX_CONFIG5	0x47 (5:3)	MREG1	Hysterisis look up from Table 4

Output interrupt field:

1. Read Low-G bit in interrupt register

   Register INT_STATUS3[3] (Bank0@0x3C)

Field Name	Register Name	Register Address	Bank	Value	Configuration
lowg_det_int	INT_STATUS3	0x3C (1)	Bank 0	0/1	0 : No event 1 : Low-G detection event

5.2 LOW-G INITIALIZATION PROCEDURE

Field Name	Register Name	Register Address	Bank	Value	Description
accel_odr	ACCEL_CONFIG0	0x21 (3:0)	Bank 0	5-11	ODR value configured
accel_ui_avg_ind	ACCEL_CONFIG1	0x24(6:4)	Bank 0	0	2x Averaging for power saving.
Accel_mode	PWR_MGMT0	0x1F(1:0)	Bank 0	2	Low power mode
accel_lp_clk_sel		0x1F(7)		0	Accel LP mode uses WU oscillator clock
dmp_odr	APEX_CONFIG1	0x26h(1:0)	Bank 0	0 or 2	DMP ODR 25 Hz or 50 Hz
Wait 1 milisecond					
lowg_peak_th_sel	APEX_CONFIG10	0x49 (7:3)	MREG1	17	Low G threshold
lowg_time_th_sel		0x49 (2:0)		4	Low G threshold time
lowg_peak_th_hyst_sel	APEX_CONFIG5	0x47 (5:3)	MREG1	4	Threshold is added to the low-G peak threshold after the initial threshold is met.
Dmp_mem_reset_en	APEX_CONFIG0	0x25h(1:0)	Bank 0	1	Clear DMP SRAM for APEX operation
Wait 1 milisecond					
dmp_init_en	APEX_CONFIG0	0x25(2)	Bank 0	1	Enable algorithm execution
Wait 50 miliseconds					
int_lowg_int1_en	INT_SOURCE6	0x2F(6)	MREG1	1	Low-G Interrupt source INT1
int_lowg_int2_en	INT_SOURCE7	0x30(6)	MREG1	1	Low-G interrupt source INT2

6 FREEFALL

The APEX freefall detects device freefall.

It uses low-G and a high-G detector to detect freefall start and freefall end. It triggers the event and outputs the related freefall duration when freefall ends. The duration is given in number of samples and it can be converted to freefall distance in meters by applying the following formula:

ff_distance(m) = 0.5*9.81*(ff_duration*dmp_odr_s)^2)

Note: dmp_odr_s in the duration of DMP_ODR expressed in seconds.

It is possible to set the minimum duration and maximum duration to trigger an event, outside this time interval the freefalls are ignored. A debounce duration is used as well to ignore device bounces after a freefall.

6.1 FREEFALL FIELDS

Input fields:

- *LOWG_PEAK_TH_SEL:* See low g input field for details
- *LOWG_TIME_TH_SEL:* See low g input field for details
- *LOWG_PEAK_TH_HYST_SEL:* See low g input field for details
- *HIGHG_PEAK_TH_SEL*

This parameter defines the threshold for accel values above which algo considers it enters high-g.

Typical value is 9 (i.e. 2500 mGee).

Bitfield is coded as follow:

Bitfield	Threshold (mGee)	Bitfield	Threshold (mGee)	Bitfield	Threshold (mGee)	Bitfield	Threshold (mGee)
0	250	8	2250	16	4250	24	6250
1	500	9	2500(typical)	17	4500	25	6500
2	750	10	2750	18	4750	26	6750
3	1000	11	3000	19	5000	27	7000
4	1250	12	3250	20	5250	28	7250
5	1500	13	3500	21	5500	29	7500
6	1750	14	3750	22	5750	30	7750
7	2000	15	4000	23	6000	31	8000

Table 5. High G Peak Threshold

- *HIGHG_ TIME_TH_SEL*: this parameter defines the number of samples device should stay in high-G before triggering interrupt.

It is coded as: number_of_samples = highg_time_th_sel + 1

Typical value is 4 (i.e. 5 samples).

- *HIGHG_PEAK_TH_HYST_SEL:* this threshold is added to the high-G peak threshold after the initial threshold is met.

  Bitfield is coded as follow:

Bitfield	Threshold(mGee)
0	31
1	63
2	94
3	125
4	156(typical)
5	188
6	219
7	250

Table 6. High G Peak Hysterisis

- *FF_MIN_DURATION_SEL:* this parameter defines the minimum freefall length that this algorithm should report. Smaller freefalls are ignored. Typical value is 0 (10 cm).

  Bitfield is coded as follow:

Bitfield	Min Freefall Length (cm)	Bitfield	Min Freefall Length (cm)
0	10(typical)	8	28
1	12	9	31
2	13	10	34
3	16	11	38
4	18	12	41
5	20	13	45
6	23	14	48
7	25	15	52

Table 7. Minimum Freefall Length

- *FF_MAX_DURATION_SEL:* this parameter defines the maximum freefall length that this algo should report. Longer freefalls are ignored. Typical value is 5 (204 cm).

Bitfield is coded as follow:

Bitfield	Min Freefall Length (cm)	Bitfield	Min Freefall Length (cm)
0	102	8	281
1	120	9	310
2	139	10	339
3	159	11	371
4	181	12	403
5	204(typical)	13	438
6	228	14	473
7	254	15	510

Table 8. Maximum Freefall Length

- *FF_DEBOUNCE_DURATION_SEL:* this parameter defines the time during which low-G and high-G events are not considered after an high-G event. The goal is to avoid detecting bounces as free falls. Typical value is 7 (2000 ms).

  Bitfield is coded as follow:

Bitfield	Debounce Duration (ms)	Bitfield	Debounce Duration (ms)
0	0	8	2125
1	1250	9	2250
2	1375	10	2375
3	1500	11	2500
4	1625	12	2625
5	1750	13	2750
6	1875	14	2875
7	2000(typical)	15	3000

Table 9. Debounce Duration

- Freefall works at ODR of 400 Hz

Field Name	Register Name	Register Address	Bank	Configuration
lowg_peak_th_sel	APEX_CONFIG10	0x49 (7:3)	MREG1	Peak threshold from Table 3
lowg_time_th_sel	APEX_CONFIG10	0x49 (2:0)	MREG1	Parameter defines the number of samples device should stay in low-G before triggering interrupt
lowg_peak_th_hyst_sel	APEX_CONFIG5	0x47 (5:3)	MREG1	Hysterisis look up from Table 4
highg_peak_th_sel	APEX_CONFIG11	0x4a (7:3)	MREG1	Peak threshold from Table 5
highg_time_th_sel	APEX_CONFIG11	0x4a (2:0)	MREG1	Parameter defines the number of samples device should stay in high-g before triggering interrupt
highg_peak_th_hyst_sel	APEX_CONFIG5	0x47 (2:0)	MREG1	Hysterisis look up from Table 6
ff_debounce_duration_sel	APEX_CONFIG9	0x48(7:4)	MREG1	This parameter defines time during which low-G and high-G events are not considered after an high-G event. Table 9.
Ff_min_duration_sel	APEX_CONFIG12	0x67(3:0)	MREG1	This parameter defines the minimum freefall length Table 7
ff_max_duration_sel	APEX_CONFIG12	0x67(7:4)	MREG1	This parameter defines the maximum freefall length Table 8

Output data fields:

- *FF_DURATION*: Free Fall duration. The duration is given in number of samples and it can be converted to freefall distance by applying the following formula:

  ff_distance(m) = 0.5*9.81*(ff_duration*dmp_odr_s)^2)

Field Name	Register Name	Register Address	Bank	Configuration
ff_dur[7:0]	APEX_DATA4	0x29 (7:0)	Bank 0	LSB freefall duration
ff_dur[15:8]	APEX_DATA5	0x30 (7:0)	Bank 0	MSB freefall duration

Output interrupt fields:

- *INT_STATUS_FF*: interrupt asserted when a freefall event is detected

Field Name	Register Name	Register Address	Bank	Value	Configuration
ff_det_int	INT_STATUS3	0x3C (2)	Bank 0	0/1	0 : No event 1 : Freefall event

6.2 FREEFALL INITIALIZATION PROCEDURE

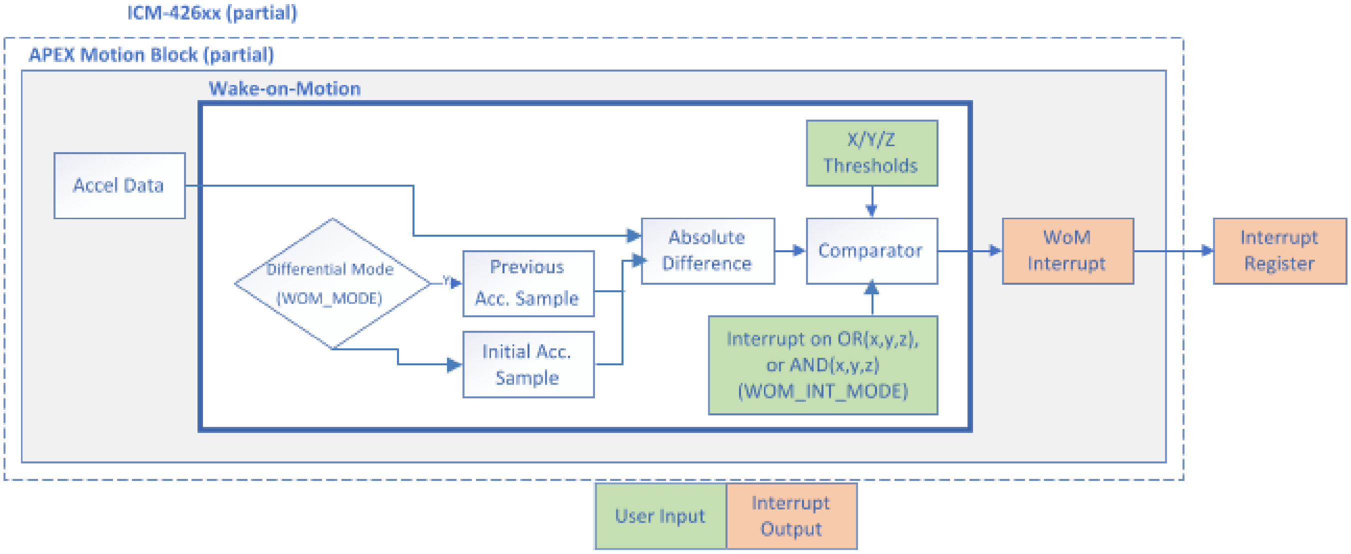
Field Name	Register Name	Register Address	Bank	Value	Description
accel_odr	ACCEL_CONFIG0	0x21 (3:0)	Bank 0	5-11	ODR value configured
accel_ui_avg	ACCEL_CONFIG1	0x24(6:4)	Bank 0	0	2x Averaging for power saving.
Accel_mode	PWR_MGMT0	0x1F(1:0)	Bank 0	2	Low power mode
accel_lp_clk_sel		0x1F(7)		0	Accel LP mode uses WU oscillator clock
dmp_odr	APEX_CONFIG1	0x26h(1:0)	Bank 0	0 or 2	DMP ODR
Wait 1 milisecond					
lowg_peak_th_sel	APEX_CONFIG10	0x49 (7:3)	MREG1	9	Low-G threshold
lowg_time_th_sel		0x49 (2:0)		4	Low-G threshold time
lowg_peak_th_hyst_sel	APEX_CONFIG5	0x47 (5:3)	MREG1	4	Threshold is added to the low-G peak threshold after the initial threshold is met.
Highg_peak_th_sel	APEX_CONFIG11	0x4a (7:3)	MREG1	9	High-G threshold
highg_time_th_sel		0x4a (2:0)		4	High-G threshold time
highg_peak_th_hyst_sel	APEX_CONFIG5	0x47 (2:0)	MREG1	4	Threshold is added to the high-G peak threshold after the initial threshold is met.
Ff_debounce_duration_sel	APEX_CONFIG9	0x48(7:4)	MREG1	7	This parameter defines time during which low-G and high-G events are not considered after an high-G event.
Ff_min_duration_sel	APEX_CONFIG12	0x67(3:0)	MREG1	0	This parameter defines the minimum freefall length
ff_max_duration_sel		0x67(7:4)		5	This parameter defines the maximum freefall length
dmp_mem_reset_en	APEX_CONFIG0	0x25h(1:0)	Bank 0	1	Clear DMP SRAM for APEX operation
Wait 1 milisecond					
dmp_init_en	APEX_CONFIG0	0x25(2)	Bank 0	1	Enable algorithm execution
Wait 50 miliseconds					
ff_int1_en	INT_SOURCE6	0x2F(7)	MREG1	1	Free fall Interrupt source INT1
ff_int2_en	INT_SOURCE7	0x30(7)	MREG1	1	Free Fall interrupt source INT2
Wait 50 miliseconds					
ff_enable	APEX_CONFIG1	0x26(5)	Bank 0	1	Free Fall Algorithm enabled

7 WAKE ON MOTION (WOM)

Wake on Motion detects when the change in accelerometer output exceeds a user-programmable threshold.

WoM can assert an interrupt if the accelerometer *1st order difference* is greater than a threshold or if the accelerometer change (relative to the first sample after WoM is enabled) is greater than a threshold.

The user can program independent acceleration thresholds for all 3 axes and configure the part to assert an interrupt when ANY axis trips its threshold, or when ALL axes trip their thresholds simultaneously.



7.1 WOM FIELDS

Input fields

- *ACCEL_WOM_X_TH*: threshold value for the X-axis Wake on Motion Interrupt. WoM thresholds range from 0g~1g in increments of 1/256 g, so WOM_X_TH=10 represents 39.1 mg (10/256 *1000 mg).
- *ACCEL_WOM_Y_TH*: threshold value for the Y-axis Wake on Motion Interrupt. Same scale as X-axis.
- *ACCEL_WOM_Z_TH*: threshold value for the Z-axis Wake on Motion Interrupt. Same scale as X-axis.
- *WOM_INT_MODE*: set WoM interrupt on the logical OR of all enabled accelerometer thresholds (when set to 0), or logical AND (when set to 1).
- *WOM_MODE*: Controls if WoM operates as differential (when set to 0) or absolute (when set to 1).
- *WOM_INT_DUR*: number of overthreshold events required to trigger interrupt (0: one overthreshold event, 1: two overthreshold events, 2: three overthreshold events, 3: four overthreshold events).

Interrupt output:

- *WOM_X_INT*: Wake on Motion Interrupt on X-axis
- *WOM_Y_INT*: Wake on Motion Interrupt on Y-axis
- *WOM_Z_INT*: Wake on Motion Interrupt on Z-axis

Field Name	Register Name	Register Address	Bank	Value	Configuration
wom_x_int	INT_STATUS2	0x3B(2)	Bank 0	0/1	0 : No event 1 : WOM event
wom_y_int	INT_STATUS2	0x3B(1)	Bank 0	0/1	0 : No event 1 : WOM event
wom_z_int	INT_STATUS2	0x3B(0)	Bank 0	0/1	0 : No event 1 : WOM event

7.2 WOM INITIALIZATION PROCEDURE

Field Name	Register Name	Register Address	Bank	Value	Description
accel_odr	ACCEL_CONFIG0	0x21 (3:0)	Bank 0	10	ODR value configured
accel_mode	PWR_MGMT0	0x1F(1:0)	Bank 0	2	Low power mode
accel_lp_clk_sel		0x1F(7)		0	Accel LP mode uses WU oscillator clock
Wait 1 milisecond					
accel_wom_x_th	ACCEL_WOM_X_THR	0x4B(7:0)	MREG1	98	Accel WOM X-axis threshold setting
accel_wom_y_th	ACCEL_WOM_Y_THR	0x4C(7:0)	MREG1	98	Accel WOM Y-axis threshold setting
accel_wom_z_th	ACCEL_WOM_Z_THR	0x4D(7:0)	MREG1	98	Accel WOM Z-axis threshold setting
Wait 1 milisecond					
wom_mode	WOM_CONFIG	0x27(1)	Bank 0	1	0: Initial sample is stored. 1: Compare current sample to previous sample
wom_int_mode		0x27(2)		1	0: WOM from 3 axes are Ored 1: WOM from 3 axes are ANDed
wom_int_dur		0x27(4:3)		0	0: WoM interrupt asserted at first overthreshold event 1: WoM interrupt asserted at second overthreshold event 2: WoM interrupt asserted at third overthreshold event 3: WoM interrupt asserted at fourth overthreshold event
wom_en	WOM_CONFIG	0x27(0)	Bank 0	1	1: enable wake-on-motion detection. 0: disable wake-on-motion detection.

8 SIGNIFICANT MOTION DETECTION (SMD)

Significant Motion Detection (SMD) needs Pedometer enabled to run and so pedometer must be configured.

8.1 SMD FIELDS

- *SMD_SENSITIVTY_SEL*

Field Name	Register Name	Register Address	Bank	Value	Configuration
smd_sensitivity_sel	APEX_CONFIG9	0x48 (3:1)	MREG1	0-4	SMD sensitivity configuration based on formula (316 << smd_sensitivity_sel[2-0]) – 1

Interrupt output field:

- *INT_STATUS_SMD:* Significant Motion Detection Interrupt

Field Name	Register Name	Register Address	Bank	Value	Configuration
smd_int	INT_STATUS2	0x3B (3)	Bank 0	0/1	0 : No event 1 : SMD event

8.2 SMD INITIALIZATION PROCEDURE

1. Initialize pedometer register as per Pedometer Initialization Procedure.
2. Continue below steps for SMD configuration.

Field Name	Register Name	Register Address	Bank	Value	Description
smd_sensitivity_sel	APEX_CONFIG9	0x48 (3:1)	MREG1	1	SMD Sensitivity
dmp_mem_reset_en	APEX_CONFIG0	0x25(0)	Bank 0	1	Clear DMP SRAM for APEX operation
Wait 1 milisecond					
dmp_init_en	APEX_CONFIG0	0x25(2)	Bank 0	1	Enable algorithm execution
Wait 50 miliseconds					
smd_int1_en	INT_SOURCE1	0x2C(3)	Bank 0	1	Enable Interupt for INT1.
Smd_int2_en	INT_SOURCE4	0x2E(3)	Bank 0	1	Enable Interupt. For INT2.
Ped_enable	APEX_CONFIG1	0x26(3)	Bank 0	1	Enable Pedometer Algorithm
smd_enable	APEX_CONFIG1	0x26(6)	Bank 0	1	Enable SMD Algorithm

9 REVISION HISTORY

REVISION DATE	REVISION	DESCRIPTION
01/08/2021	1.0	Initial release
07/01/2021	1.1	WoM ODR support added

10 COMPLIANCE DISCLAIMER

InvenSense believes the environmental and other compliance information given in this document to be correct but cannot guarantee accuracy or completeness. Conformity documents substantiating the specifications and component characteristics are on file.

InvenSense subcontracts manufacturing, and the information contained herein is based on data received from vendors and suppliers, which has not been validated by InvenSense.

This information furnished by InvenSense or its affiliates ("TDK InvenSense") is believed to be accurate and reliable. However, no responsibility is assumed by TDK InvenSense for its use, or for any infringements of patents or other rights of third parties that may result from its use. Specifications are subject to change without notice. TDK InvenSense reserves the right to make changes to this product, including its circuits and software, in order to improve its design and/or performance, without prior notice. TDK InvenSense makes no warranties, neither expressed nor implied, regarding the information and specifications contained in this document. TDK InvenSense assumes no responsibility for any claims or damages arising from information contained in this document, or from the use of products and services detailed therein. This includes, but is not limited to, claims or damages based on the infringement of patents, copyrights, mask work and/or other intellectual property rights.

Certain intellectual property owned by InvenSense and described in this document is patent protected. No license is granted by implication or otherwise under any patent or patent rights of InvenSense. This publication supersedes and replaces all information previously supplied. Trademarks that are registered trademarks are the property of their respective companies. TDK InvenSense sensors should not be used or sold in the development, storage, production or utilization of any conventional or mass-destructive weapons or for any other weapons or life threatening applications, as well as in any other life critical applications such as medical equipment, transportation, aerospace and nuclear instruments, undersea equipment, power plant equipment, disaster prevention and crime prevention equipment.

TDK InvenSense