A ROHM Group Company

TECHNICAL REFERENCE MANUAL

KX132-1211

Rev. 1.0

31-Jul-2019

Kionix, Inc.
36 Thornwood Drive
Ithaca, NY 14850
USA
TEL: +1-607-257-1800
http://www.kionix.com
info@kionix.com

Overview

This technical reference manual contains information for KX132-1211. Explanation for embedded registers and embedded applications are included. For pin assignment and specifications of the device, please also refer to KX132-1211 Specifications.

36 Thornwood Dr. – Ithaca, NY 14850
tel: 607-257-1080 – fax:607-257-1146
www.kionix.com - info@kionix.com

Table of Contents

36 Thornwood Dr. – Ithaca, NY 14850
tel: 607-257-1080 – fax:607-257-1146
www.kionix.com - info@kionix.com

1 Embedded Registers

1.1 Register Map

The KX132-1211 has embedded 8-bit registers that are accessible by the user. This section contains the addresses for all embedded registers and describes bit functions of each register. Table 1 below provides a listing of the accessible 8-bit registers and their addresses.

Address	Register Name	R/W
00	MAN_ID	R
01	PART_ID	R
02	XADP_L	R
03	XADP_H	R
04	YADP_L	R
05	YADP_H	R
06	ZADP_L	R
07	ZADP_H	R
08	XOUT_L	R
09	XOUT_H	R
0A	YOUT_L	R
0B	YOUT_H	R
0C	ZOUT_L	R
0D	ZOUT_H	R
0E-11	Kionix Reserved[3]	
12	COTR	R
13	WHO_AM_I	R
14	TSCP	R
15	TSPP	R
16	INS1	R
17	INS2	R
18	INS3	R
19	STATUS_REG	R
1A	INT_REL	R

Address	Register Name	R/W
1B	CNTL1[1]	R/W
1C	CNTL2[2]	R/W
1D	CNTL3[1]	R/W
1E	CNTL4[1]	R/W
1F	CNTL5[2]	R/W
20	CNTL6[2]	R/W
21	ODCNTL[1]	R/W
22	INC1[1]	R/W
23	INC2[1]	R/W
24	INC3[1]	R/W
25	INC4[1]	R/W
26	INC5[1]	R/W
27	INC6[1]	R/W
28	Kionix Reserved[3]	
29	TILT_TIMER[2]	R/W
2A	TDTRC[2]	R/W
2B	TDTC[2]	R/W
2C	TTH[2]	R/W
2D	TTL[2]	R/W
2E	FTD[2]	R/W
2F	STD[2]	R/W
30	TLT[2]	R/W
31	TWS[2]	R/W
32	FFTH[2]	R/W

Address	Register Name	R/W
33	FFC[2]	R/W
34	FFCNTL[1]	R/W
35-36	Kionix Reserved[3]	
37	TILT_ANGLE_LL[2]	R/W
38	TILT_ANGLE_HL[2]	R/W
39	HYST_SET[2]	R/W
3A	LP_CNTL1[1]	R/W
3B	LP_CNTL2[1]	R/W
3C-48	Kionix Reserved[3]	
49	WUFTH[2]	R/W
4A	BTSWUFTH[2]	R/W
4B	BTSTH[2]	R/W
4C	BTSC[2]	R/W
4D	WUFC[2]	R/W
4E-5C	Kionix Reserved[3]	
5D	SELF_TEST	W
5E	BUF_CNTL1[2]	R/W
5F	BUF_CNTL2[2]	R/W
60	BUF_STATUS_1	R
61	BUF_STATUS_2	R
62	BUF_CLEAR[2]	W
63	BUF_READ	R
64-76	ADP_CNTL(1-19)[2]	R/W
77-7F	Kionix Reserved[3]	

Table 1: Register Map

Notes

1. When changing the contents of these registers, the PC1 bit in CNTL1 must first be set to “0”.
2. On-The-Fly (OTF) register can be written to while the KX132-1211 is enabled (PC1 bit in CNTL1 register is 1) and the change will be accepted with no interruption in the operation.
3. Reserved registers should not be written.

1.2 MAN_ID (0x00)

A burst read (reading using the auto-increment) of 4 bytes starting at address 00, returns the manufacturing ID: "K" "i" "o" "n" in ASCII codes "0x4B" "0x69" "0x6F" "0x6E".

R	R	R	R	R	R	R	R
MANID7	MANID6	MANID5	MANID4	MANID3	MANID2	MANID1	MANID0
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x00		

1.3 PART_ID (0x01)

A burst read (reading using the auto-increment) of 2 bytes starting at address 01, returns Who-Am-I value ("WAI") as the first byte (LSB) and a 2nd byte (MSB) that returns silicon specific ID.

R	R	R	R	R	R	R	R
PARTID7	PARTID6	PARTID5	PARTID4	PARTID3	PARTID2	PARTID1	PARTID0
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x01		

Note: A burst read (reading using the auto-increment) of 6 bytes starting at address 00, returns the MAN_ID followed by the 2 bytes of PART_ID

1.4 ADP OUTPUT REGISTERS (0x02 – 0x07)

Output from the Advanced Data Path is routed to registers 0x02 – 0x07 (XADP_L – ZADP_H) when ADPE bit is set to 1 in CNTL5 register. Data is updated at the rate set by OADP<3:0> bits in ADP_CNTL1 register. However, if data is routed via RMS block first (ADP_RMS_OSEL bit is set to 1 in ADP_CNTL2 register), the rate is also scaled down by RMS_AVC<2:0> bits in ADP_CNTL1 register. The output data is provided in 2's complement data format and is protected while reading using auto increment mode.

XADP_L

X-axis Advanced Data Path (ADP) output least significant byte.

R	R	R	R	R	R	R	R
XHP7	XHP6	XHP5	XHP4	XHP3	XHP2	XHP1	XHP0
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x02		

XADP_H

X-axis Advanced Data Path (ADP) output most significant byte.

R	R	R	R	R	R	R	R
XHP15	XHP14	XHP13	XHP12	XHP11	XHP10	XHP9	XHP8
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x03		

YADP_L

Y-axis Advanced Data Path (ADP) output least significant byte.

R	R	R	R	R	R	R	R
YHP7	YHP6	YHP5	YHP4	YHP3	YHP2	YHP1	YHP0
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x04		

YADP_H

Y-axis Advanced Data Path (ADP) output most significant byte.

R	R	R	R	R	R	R	R
YHP15	YHP14	YHP13	YHP12	YHP11	YHP10	YHP9	YHP8
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x05		

ZADP_L

Z-axis Advanced Data Path (ADP) output least significant byte.

R	R	R	R	R	R	R	R
ZHP7	ZHP6	ZHP5	ZHP4	ZHP3	ZHP2	ZHP1	ZHP0
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x06		

ZADP_H

Z-axis Advanced Data Path (ADP) output most significant byte.

R	R	R	R	R	R	R	R
ZHP15	ZHP14	ZHP13	ZHP12	ZHP11	ZHP10	ZHP9	ZHP8
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x07		

1.5 ACCELEROMETER OUTPUT REGISTERS (0x08 – 0x0D)

When accelerometer is enabled (PC1 bit is set to 1 in CNTL1 register), the 16-bits of valid acceleration data for each axis is routed to registers 0x08-0x0D (XOUT_L – ZOUT_H). However, the user may choose to read only the most significant byte(s) of the output data thus reading an effective 8-bit resolution. The data is updated every user-defined ODR period at the rate set by OSA<3:0> bits in ODCNTL register. It is recommended to read the output registers using the auto-increment mode to ensure that content of the registers doesn't change during the data read out. The output data is available in 2's complement data format and can be converted from digital counts to acceleration (g) per Table 2 below. For example, if N = 16 bits, then the Counts range is from -32768 to 32767, and if N = 8 bits, then the Counts range is from -128 to 127.

16-bit Register Data (2's complement)	Equivalent Counts in decimal	Range = ±2g	Range = ±4g	Range = ±8g	Range = ±16g
0111 1111 1111 1111	32767	+1.99994g	+3.99988g	+7.99976g	+15.99951g
0111 1111 1111 1110	32766	+1.99988g	+3.99976g	+7.99951g	+15.99902g
…	…	…	…	…	…
0000 0000 0000 0001	1	+0.00006g	+0.00012g	+0.00024g	+0.00049g
0000 0000 0000 0000	0	0.00000g	0.00000g	0.00000g	0.00000g
1111 1111 1111 1111	-1	-0.00006g	-0.00012g	-0.00024g	-0.00049g
…	…	…	…	…	…
1000 0000 0000 0001	-32767	-1.99994g	-3.99988g	-7.99976g	-15.99951g
1000 0000 0000 0000	-32768	-2.00000g	-4.00000g	-8.00000g	-16.00000g

8-bit Register Data (2's complement)	Equivalent Counts in decimal	Range = ±2g	Range = ±4g	Range = ±8g	Range = ±16g
0111 1111	127	+1.98438g	+3.96875g	+7.93750g	+15.87500g
0111 1110	126	+1.96875g	+3.93750g	+7.87500g	+15.75000g
…	…	…	…	…	…
0000 0001	1	+0.01563g	+0.03125g	+0.06250g	+0.12500g
0000 0000	0	0.0000g	0.0000g	0.0000g	0.0000g
1111 1111	-1	-0.01563g	-0.03125g	-0.06250g	-0.12500g
…	…	…	…	…	…
1000 0001	-127	-1.98438g	-3.96875g	-7.93750g	-15.87500g
1000 0000	-128	-2.00000g	-4.00000g	-8.00000g	-16.00000g

Table 2: Acceleration (g) Calculation

XOUT_L

X-axis accelerometer output least significant byte.

R	R	R	R	R	R	R	R
XOUT7	XOUT6	XOUT5	XOUT4	XOUT3	XOUT2	XOUT1	XOUT0
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x08		

XOUT_H

X-axis accelerometer output most significant byte.

R	R	R	R	R	R	R	R
XOUT15	XOUT14	XOUT13	XOUT12	XOUT11	XOUT10	XOUT9	XOUT8
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x09		

YOUT_L

Y-axis accelerometer output least significant byte.

R	R	R	R	R	R	R	R
YOUT7	YOUT6	YOUT5	YOUT4	YOUT3	YOUT2	YOUT1	YOUT0
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x0A		

YOUT_H

Y-axis accelerometer output most significant byte.

R	R	R	R	R	R	R	R
YOUT15	YOUT14	YOUT13	YOUT12	YOUT11	YOUT10	YOUT9	YOUT8
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x0B		

ZOUT_L

Z-axis accelerometer output least significant byte.

R	R	R	R	R	R	R	R
ZOUT7	ZOUT6	ZOUT5	ZOUT4	ZOUT3	ZOUT2	ZOUT1	ZOUT0
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x0C		

ZOUT_H

Z-axis accelerometer output most significant byte.

R	R	R	R	R	R	R	R
ZOUT15	ZOUT14	ZOUT13	ZOUT12	ZOUT11	ZOUT10	ZOUT9	ZOUT8
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x0D		

1.6 COTR (0X12)

The Command Test Response (COTR) register is used to verify proper integrated circuit functionality. The value of this register will change from a default value of 0x55 to 0xAA when COTC bit in CNTL2 register is set. After reading 0xAA from this register, the byte value returns to the default value of 0x55 and COTC bit in CNTL2 register is cleared.

R	R	R	R	R	R	R	R	
DCSTR7	DCSTR6	DCSTR5	DCSTR4	DCSTR3	DCSTR2	DCSTR1	DCSTR0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	01010101
					Address: 0x12			

1.7 WHO_AM_I (0X13)

This register can be used for supplier recognition, as it can be factory written to a known byte value. WHO_AM_I is the first byte (LSB) of the new PART ID. The default value is 0x3D.

R	R	R	R	R	R	R	R	
WAI7	WAI6	WAI5	WAI4	WAI3	WAI2	WAI1	WAI0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00111101
					Address: 0x13			

1.8 TILT POSITION REGISTERS (0x14 – 0x15)

These two registers report previous and current position data that is updated at the user-defined ODR frequency determined by OTP<1:0> in CNTL3. Data is protected during register read. Table 3 describes the reported position for each bit value.

TSCP

Current Tilt Position Register.

R	R	R	R	R	R	R	R	
0	0	LE	RI	DO	UP	FD	FU	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00100000
					Address: 0x14			

TSPP

Previous Tilt Position Register.

R	R	R	R	R	R	R	R	
0	0	LE	RI	DO	UP	FD	FU	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00100000
					Address: 0x15			

Bit	Description
LE	Left State (X-)
RI	Right State (X+)
DO	Down State (Y-)
UP	Up State (Y+)
FD	Face-Down State (Z-)
FU	Face-Up State (Z+)

Table 3: Tilt Position

1.9 INTERRUPT SOURCE REGISTERS (0x16 – 0x18)

These three registers report interrupt state changes. The status is updated when a new interrupt event occurs, and the bit remains set until it is cleared as indicated in each case.

INS1

This register contains Tap™/Double-Tap™ axis specific interrupts. Data is updated at the ODR settings determined by OTDT<2:0> in CNTL3. These bits are cleared when the interrupt latch release register (INT_REL) is read.

R	R	R	R	R	R	R	R
Reserved	Reserved	TLE	TRI	TDO	TUP	TFD	TFU
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x16		

Bit	Description
TLE	X Negative (X-) Reported
TRI	X Positive (X+) Reported
TDO	Y Negative (Y-) Reported
TUP	Y Positive (Y+) Reported
TFD	Z Negative (Z-) Reported
TFU	Z Positive (Z+) Reported

Table 4: Directional-Tap™ Reporting

INS2

This register tells which function caused an interrupt.

R	R	R	R	R	R	R	R
FFS	BFI	WMI	DRDY	TDTS1	TDTS0	Reserved	TPS
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x17		

FFS *– Free fall Status (FFS) bit. This bit is cleared when the interrupt latch release register (INT_REL) is read.*

FFS = 0 – No Free fall
FFS = 1 – Free fall has activated the interrupt

BFI *– Buffer Full Interrupt (BFI) bit indicates that buffer has been filled. This bit is automatically cleared when at least one sample from the buffer is read.*

BFI = 0 – Buffer is not full
BFI = 1 – Buffer is full

WMI – *Watermark Interrupt bit indicates that user-defined buffer's sample threshold (watermark) has been exceeded when in FIFO or Stream modes. Not used in Trigger mode. This bit is automatically cleared when buffer is read, and the content is below the watermark.*

WMI = 0 – Buffer watermark has not been exceeded
WMI = 1 – Buffer watermark has been exceeded

DRDY – *Data Ready (DRDY) interrupt bit indicates that new acceleration data is available in output data registers 0x08 to 0x0D. This bit is cleared when acceleration data is read or the interrupt latch release register (INT_REL) is read.*

DRDY = 0 - new acceleration data is not available
DRDY = 1 - new acceleration data is available

TDTS<1:0> – *Tap™/Double-Tap™ Status bits. These bits are cleared when the interrupt latch release register (INT_REL) is read.*

TDTS1	TDTS0	Event
0	0	No Tap
0	1	Single-Tap
1	0	Double-Tap
1	1	undefined

Table 5: Tap™/Double-Tap™ Status Reporting Bits

TPS – *Tilt Position Status bit*

TPS = 0 – Position has not changed
TPS = 1 – Position has changed

INS3

This register reports the axis and direction of detected motion that triggered the wakeup interrupt.

R	R	R	R	R	R	R	R
WUFS	BTS	XNWU	XPWU	YNWU	YPWU	ZNWU	ZPWU
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x18		

WUFS *– Wake up interrupt. This bit is cleared when the interrupt latch release register (INT_REL) is read.*

WUFS = 1 – Motion is above wake up threshold
WUFS = 0 – Motion is below wake up threshold

BTS *– Back to sleep interrupt. This bit is cleared when the interrupt latch release register (INT_REL) is read.*

BTS = 1 – Motion is below back to sleep threshold
BTS = 0 – Motion is above back to sleep threshold

XNWU / XPWU / YNWU / YPWU / ZNWU / ZPWU

Bit	Description
XNWU	X Negative (X-) Reported
XPWU	X Positive (X+) Reported
YNWU	Y Negative (Y-) Reported
YPWU	Y Positive (Y+) Reported
ZNWU	Z Negative (Z-) Reported
ZPWU	Z Positive (Z+) Reported

Table 6: Motion Detection™ Reporting

1.10 STATUS_REG (0X19)

This register reports the status of the interrupt.

R	R	R	R	R	R	R	R
0	0	0	INT	0	0	0	WAKE
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x19		

INT *– reports the combined (OR) interrupt information according to interrupt setting.*
0 = no interrupt event
1 = interrupt event has occurred

WAKE *– reports the wake/back to sleep state*
0 = back-to-sleep state
1 = wake state

Note: Wake is the default state at power-up, shown in STATUS_REG register. For wake engine only operation, set MAN_SLEEP bit to 1 in CNTL5 register to put KX132-1211 in sleep state for the first time.

1.11 INT_REL (0X1A)

Interrupt latch release. Latched interrupt source information (INS1-INS3) is cleared and physical interrupt latched pin is changed to its inactive state when this register is read.

R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
					Address: 0x1A		

Notes:

1. WMI and BFI are not cleared by this command.
2. The latched interrupts are not cleared when INT_REL register is read using the auto increment read mode in SPI communication, unless it is the starting address.

1.12 CONTROL REGISTERS (0x1B – 0x20)

The main control functions of the accelerometer can be set via CNTL1 – CNTL6 registers.

CNTL1

Control register 1. Read/write control register that controls the main feature set. Note that to properly change the value of these registers, the PC1 bit in CNTL1 register must first be set to "0".

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
PC1	RES	DRDYE	GSEL1	GSEL0	TDTE	Reserved	TPE	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x1B			

PC1 *– controls the operating mode.*

PC1 = 0 – stand-by mode
PC1 = 1 – High-Performance or Low Power mode

RES *– The RES bit determines the performance mode of the KX132-1211. The noise varies with ODR, RES and different LP_CNTL1 settings possibly reducing the effective resolution.*

RES = 0 – Low Power mode (higher noise, lower current, 16-bit output data)
RES = 1 – High-Performance mode (lower noise, higher current, 16-bit output data)

DRDYE *– Data Ready Engine enable bit.*

DRDYE = 0 – Data Ready Engine is disabled
DRDYE = 1 – Data Ready Engine is enabled

GSEL<1:0> *– G-range Select (GSEL) bits select the acceleration range of the accelerometer outputs per Table 7. This range is also called a full-scale range of the accelerometer.*

GSEL1	GSEL0	Range
0	0	±2g
0	1	±4g
1	0	±8g
1	1	±16g

Table 7: Selected Acceleration Range

TDTE *– Tap/Double-Tap™ Engine (TDTE) enable bit.*

TDTE = 0 – Tap/Double-Tap™ Engine is disabled
TDTE = 1 – Tap/Double-Tap™ Engine is enabled

Reserved *– The value of reserved bit should not be changed*

TPE *– Tilt Position Engine (TPE) enable bit.*
TPE = 0 – Tilt Position Engine is disabled
TPE = 1 – Tilt Position Engine is enabled

CNTL2

Control register 2. Read/write control register that primarily controls tilt position state enabling. This register has also settings to verify proper power up. This register is On-The-Fly (OTF) register and can be written to while the KX132-1211 is enabled (PC1 bit in CNTL1 register is set to "1") and the change will be accepted with no interruption in the operation. The exception is the SRST bit 7. To change the value of the SRST bit, the PC1 bit in CNTL1 register must first be set to 0.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
SRST	COTC	LEM	RIM	DOM	UPM	FDM	FUM	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00111111
					Address: 0x1C			

SRST *– The Software Reset bit initiates software reset, which performs the RAM reboot routine. This bit will remain 1 until the RAM reboot routine is finished. Please refer to Technical Note TN027 Power-On Procedure for more information on software reset.*

SRST = 0 – no action
SRST = 1 – start POR / RAM reboot routine

Note 1: For I^2C Communication: Setting SRST = 1 will NOT result in an ACK, since the part immediately enters the RAM reboot routine. NACK may be used to confirm this command.

Note 2: To change the value of the SRST bit, the PC1 bit in CNTL1 register must first be set to 0.

COTC *– The Command Test Control bit is used to verify proper ASIC functionality.*

COTC = 0 – no action
COTC = 1 – sets COTR register to 0xAA. When COTR register is then read, sets COTC bit to 0 and sets COTR register to 0x55.

LEM, RIM, DOM, UPM, FDM, FUM *– these bits control the tilt axis mask. Per Table 8, if a direction's bit is set to one (1), tilt in that direction will generate an interrupt. If it is set to zero (0), tilt in that direction will not generate an interrupt.*

Bit	Description
LEM	*Left state enable* (X-)
RIM	*Right state enable* (X+)
DOM	*Down state enable* (Y-)
UPM	*Up state enable* (Y+)
FDM	*Face-Down state enable* (Z-)
FUM	*Face-Up state enable* (Z+)

Table 8: Tilt Direction™ Axis Mask

CNTL3

Control register 3. Read/write control register that provides control of the Output Data Rate (ODR) for Tilt, Tap, and Wake-up engines. Note that to properly change the value of these registers, the PC1 bit in CNTL1 register must first be set to "0".

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
OTP1	OTP0	OTDT2	OTDT1	OTDT0	OWUF2	OWUF1	OWUF0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	10101000
					Address: 0x1D			

***OTP*<1:0>** – *ODR Tilt Position (OTP) sets the output data rate for the Tilt Position function per* Table 9*. The default Tilt Position ODR is 12.5Hz.*

OTP1	OTP0	Output Data Rate (Hz)
0	0	1.563
0	1	6.25
1	0	12.5
1	1	50

Table 9: Tilt Position Function Output Data Rate

***OTDT*<2:0>** – *ODR Tap/Double-Tap™ (OTDT) sets the output data rate for the Directional-Tap™ function per Table 10. The default Directional-Tap™ ODR is 400Hz.*

OTDT2	OTDT1	OTDT0	Output Data Rate (Hz)
0	0	0	12.5
0	0	1	25
0	1	0	50
0	1	1	100
1	0	0	200
1	0	1	400
1	1	0	800
1	1	1	1600

Table 10: Directional-Tap™ Function Output Data Rate

***OWUF*<2:0>** – *ODR Wake-Up Function (OWUF) sets the output data rate (per Table 11) at which the wake up (motion detection) performs its function. The default Motion Wake-Up ODR is 0.781Hz.*

Note1: ODR Wake-Up Function setting (OWUF<2:0>) needs to be less than or equal to accelerometer ODR setting (OSA<3:0>) to avoid irregular resulting acceleration ODRs.

Note 2: If Advanced Data Path data is routed to the Wake-Up engine (ADPE = 1, ADP_WB_ISEL = 1), OADP<3:0> also sets the ODR for the Wake-Up engines. In this case, the ODR set by OWUF<2:0> is ignored.

OWUF2	OWUF1	OWUF0	Output Data Rate (Hz)
0	0	0	0.781
0	0	1	1.563
0	1	0	3.125
0	1	1	6.25
1	0	0	12.5
1	0	1	25
1	1	0	50
1	1	1	100

Table 11: Motion Wake Up Function Output Data Rate

CNTL4

Control register 4. Read/write control register that provides more feature set control. Note that to properly change the value of these registers, the PC1 bit in CNTL1 register must first be set to "0".

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
C_MODE	TH_MODE	WUFE	BTSE	PR_MODE	OBTS2	OBTS1	OBTS0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	01000000
					Address: 0x1E			

C_MODE *– defines debounce counter clear mode*
C_MODE = 0 – debounce counter is reset if accelerometer data is below threshold
C_MODE = 1 – debounce counter is decremented if accelerometer data is below threshold

TH_MODE *– defines wake / back-to-sleep threshold mode*
TH_MODE = 0 – absolute threshold
TH_MODE = 1 – relative threshold

WUFE *– Wake-Up Function Engine enable bit*
WUFE = 0 – Wake-Up Function Engine is disabled
WUFE = 1 – Wake-Up Function Engine is enabled

BTSE *– Back-to-Sleep Engine enable bit*
BTSE = 0 – Back-to-Sleep Engine is disabled
BTSE = 1 – Back-to-Sleep Engine is enabled

PR_MODE *– defines Pulse Reject mode*
PR_MODE = 0 – standard operation
PR_MODE = 1 – reject pulse-like motion only in case motion detection in positive and negative directions is enabled (XPWUE and XNWUE bits are set to 1 in INC2 register).

OBTS<2:0> *– sets the output data rate (per Table 12) at which the back-to-sleep (motion detection) performs its function during wake state. The default Back-to-Sleep ODR is 0.781Hz*
Note 1: ODR Back-to-Sleep Function setting (OBTS<2:0>) needs to be less than or equal to accelerometer ODR setting (OSA<3:0>) to avoid irregular resulting acceleration ODRs.
Note 2: If Advanced Data Path data is routed to the Back-to-Sleep engine (ADPE = 1, ADP_WB_ISEL = 1), OADP<3:0> also sets the ODR for the Back-to-Sleep engines. In this case, the ODR set by OBTS<2:0> is ignored.

OBTS2	OBTS1	OBTS0	Output Data Rate
0	0	0	0.781Hz
0	0	1	1.563Hz
0	1	0	3.125Hz
0	1	1	6.250Hz
1	0	0	12.5Hz
1	0	1	25Hz
1	1	0	50Hz
1	1	1	100Hz

Table 12: Motion Back-to-Sleep Function Output Data Rate

CNTL5

Control register 5. Read/write control register that provides more feature set control. This register is On-The-Fly (OTF) register and can be written to while the KX132-1211 is enabled (PC1 bit in CNTL1 register is set to "1") and the change will be accepted with no interruption in the operation.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reserved	Reserved	Reserved	ADPE	Reserved	Reserved	MAN_WAKE	MAN_SLEEP	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x1F			

ADPE − *Advanced Data Path (ADP) enable*
ADPE = 0 – Advanced Data Path is disabled.
ADPE = 1 – Advanced Data Path is enabled. Outputs are available in XADP, YADP, ZADP registers.

MAN_WAKE − *manual wake-sleep engine overwrite*
MAN_WAKE = 0 – default
MAN_WAKE = 1 – forces wake state (bit is self-cleared)

MAN_SLEEP – *manual wake-sleep engine overwrite*
MAN_SLEEP = 0 – default
MAN_SLEEP = 1 – forces sleep state (bit is self-cleared)

Reserved – *these bits are reserved, and their values should not be changed.*

Notes:

1. *Once a wake interrupt has occurred, no additional wake interrupt events are registered until the part is put back to sleep using the BTS interrupt or manually using MAN_SLEEP bit.*

2. *Wake is the default state at power-up, shown in STATUS_REG register. For wake engine only operation, set MAN_SLEEP bit to 1 in CNTL5 register to put KX132-1211 in sleep state for the first time.*

CNTL6

Control register 6. Read/write control register that provides more feature set control. This register is On-The-Fly (OTF) register and can be written to while the KX132-1211 is enabled (PC1 bit in CNTL1 register is set to "1") and the change will be accepted with no interruption in the operation.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
I2C_ALE	Reserved	Reserved	Reserved	Reserved	Reserved	I2C_ALC1	I2C_ALC0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x20			

I2C_ALE *– enables the I²C auto release function. When this function is enabled, the KX132-1211 would release the SDA line if the 9th clock pulse is not detected from the host after the delay defined by I2C_ALC<1:0> bits.*

I2C_ALE = 0 – I²C auto release function is disabled
I2C_ALE = 1 – I²C auto release function is enabled

I2C_ALC<1:0> *– I²C auto release function counter select*

00 = 0.5sec
01 = 1.0sec
10 = 2.0sec
11 = 4.0sec

Reserved *– these bits are reserved, and their values should not be changed.*

1.13 ODCNTL (0X21)

Output data control register that configures the acceleration outputs. Note that to properly change the value of these registers, the PC1 bit in CNTL1 register must first be set to "0".

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
IIR_BYPASS	LPRO	FSTUP	Reserved	OSA3	OSA2	OSA1	OSA0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000110
					Address: 0x21			

IIR_BYPASS *– IIR Filter Bypass mode enable bit*

IIR_BYPASS = 0 – IIR filter is not bypassed, i.e. filtering is applied (default)
IIR_BYPASS = 1 – IIR filter is bypassed.

Notes for IIR_BYPASS = 1 setting:

1. *Not recommended at OSA<3:0> = 1111 (ODR = 25600Hz)*
2. *Not recommended in Low Power Mode with AVC<2:0> = 000 setting (no averaging)*
3. *This setting may reduce the resolution of the output data.*

LPRO *– Low-Pass filter Roll-Off control*

LPRO = 0 – IIR filter corner frequency set to ODR/9 (default)
LPRO = 1 – IIR filter corner frequency set to ODR/2

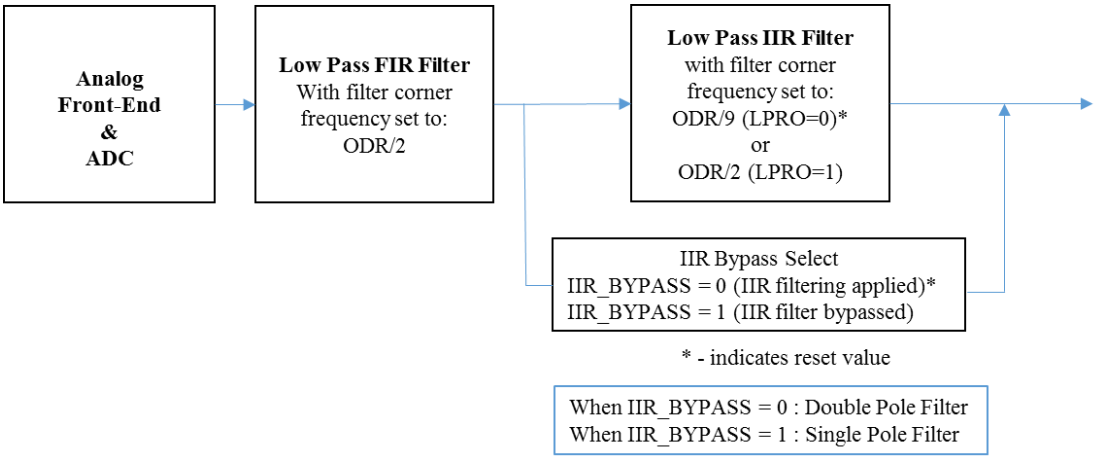

Figure 1: Low-Pass Filter Design and Control Circuitry

FSTUP *– Fast Start Up Enable bit. The setting of this bit controls the start up time only when accelerometer operates in High-Performance mode with ODR ≤ 200Hz. If fast start up is disabled (FSTUP=0), the start up time in High-Performance mode would vary with ODR. If fast start up is enabled (FSTUP=1), the start up time in High Performance mode would be fixed. See KX132-1211 Product specifications for details.*

FSTUP = 0 – Fast Start is disabled
FSTUP = 1 – Fast Start is enabled

Reserved *– this bit is reserved, and its value should not be changed.*

***OSA*<3:0>** *– Output Data Rate (ODR) settings for accelerometer sensor. The default ODR is 50Hz.*

OSA3	OSA2	OSA1	OSA0	Output Data Rate (Hz)
0	0	0	0	0.781*
0	0	0	1	1.563*
0	0	1	0	3.125*
0	0	1	1	6.25*
0	1	0	0	12.5*
0	1	0	1	25*
0	1	1	0	50*
0	1	1	1	100*
1	0	0	0	200*
1	0	0	1	400*
1	0	1	0	800**
1	0	1	1	1600**
1	1	0	0	3200**
1	1	0	1	6400**
1	1	1	0	12800**
1	1	1	1	25600**

Table 13: Accelerometer Output Data Rates (ODR)

* Available in Low Power and High-Performance modes
** Available in High-Performance mode only. Accelerometer will default to High-Performance mode regardless of the RES bit setting in CNTL1 register.

1.14 INTERRUPT CONTROL REGISTERS (0X22 – 0X27)

INC1

Interrupt Control 1. This register controls the settings for the physical interrupt pin INT1. Note that to properly change the value of this register, the PC1 bit in CNTL1 register must first be set to "0".

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
PW11	PW10	IEN1	IEA1	IEL1	Reserved	STPOL	SPI3E	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00010000
					Address: 0x22			

PW1<1:0> *– Pulse INT1 pin width configuration*

00 = 50µsec (10µsec if accelerometer ODR (OSA<3:0>) > 1600Hz)
*01 = 1 * OSA period*
*10 = 2 * OSA periods*
11 = Real time mode
When PW1 > 0, Interrupt source auto-clearing (ACLR1=1) should be set to keep consistency between the internal status and the physical interrupt.

IEN1 *– enables/disables the physical interrupt pin*

IEN1 = 0 – physical interrupt pin is disabled
IEN1 = 1 – physical interrupt pin is enabled

IEA1 *– Interrupt active level control for interrupt pin*

IEA1 = 0 – active LOW
IEA1 = 1 – active HIGH

IEL1 *– Interrupt latch control for physical interrupt pin*

IEL1 = 0 – latched until cleared by reading INT_REL
IEL1 = 1 – pulsed. The pulse width is configurable by PW1.

Reserved *– this bit is reserved, and its value should not be changed.*

STPOL *– sets the polarity of Self-Test.*

STPOL = 0 – Negative
STPOL = 1 – Positive

SPI3E *– sets the 3-wire SPI interface*

SPI3E = 0 – disabled
SPI3E = 1 – enabled

INC2

Interrupt Control 2. This register defines interrupt behavior for the Wake-Up Function (WUF) and Back-to-Sleep (BTS) engines. Note that to properly change the value of this register, the PC1 bit in CNTL1 register must first be set to "0".

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reserved	AOI	XNWUE	XPWUE	YNWUE	YPWUE	ZNWUE	ZPWUE	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00111111
					Address: 0x23			

Reserved *– this bit is reserved, and its value should not be changed.*

AOI *– AND-OR configuration on motion detection*
AOI = 0 – OR combination between selected directions
AOI = 1 – AND combination between selected axes

Ex. If all directions are enabled,
Active state in OR configuration = (XN || XP || YN || YP || ZN || ZP)
Active state in AND configuration = (XN || XP) && (YN || YP) && (ZN || ZP)

XNWUE *– X Negative (X-) mask for WUF and BTS*
XNWUE = 0 - WUF/BTS ignores (X-) axis
XNWUE = 1 - enable the (X-) interrupt source

XPWUE *– X Positive (X+) mask for WUF and BTS*
XPWUE = 0 - WUF/BTS ignores (X+) axis
XPWUE = 1 - enable the (X+) interrupt source

YNWUE *– Y Negative (Y-) mask for WUF and BTS*
YNWUE = 0 - WUF/BTS ignores (Y-) axis
YNWUE = 1 - enable the (Y-) interrupt source

YPWUE *– Y Positive (Y+) mask for WUF and BTS*
YPWUE = 0 - WUF/BTS ignores (Y+) axis
YPWUE = 1 - enable the (Y+) interrupt source

ZNWUE *– Z negative (Z-) mask for WUF and BTS*
ZNWUE = 0 - WUF/BTS ignores (Z-) axis
ZNWUE = 1 - enable the (Z-) interrupt source

ZPWUE *– Z positive (Z+) mask for WUF and BTS*
ZPWUE = 0 - WUF/BTS ignores (Z+) axis
ZPWUE = 1 - enable the (Z+) interrupt source

INC3

Interrupt Control 3. This register controls which axis and direction of tap/double tap can cause an interrupt. If a direction's bit is set to "1", a single or double tap in that direction will generate an interrupt. If it is set to "0", a single or double tap in that direction will not generate an interrupt. Note that to properly change the value of this register, the PC1 bit in CNTL1 register must first be set to "0".

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
0	TMEN	TLEM	TRIM	TDOM	TUPM	TFDM	TFUM	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00111111
					Address: 0x24			

TMEN *– enables/disables alternate tap masking scheme*
TMEN = 0 – alternate tap masking scheme disabled
TMEN = 1 – alternate tap masking scheme enabled

TLEM *– Tap Left (X-) state mask*
TLEM = 0 – Tap engine ignores (X-) axis
TLEM = 1 – Enable the (X-) interrupt source

TRIM *– Tap Right (X+) state mask*
TRIM = 0 – Tap engine ignores (X+) axis
TRIM = 1 – Enable the (X+) interrupt source

TDOM *– Tap Down (Y-) state mask*
TDOM = 0 – Tap engine ignores (Y-) axis
TDOM = 1 – Enable the (Y-) interrupt source

TUPM *– Tap Up (Y+) state mask*
TUPM = 0 – Tap engine ignores (Y+) axis
TUPM = 1 – Enable the (Y+) interrupt source

TFDM *– Tap Face Down (Z-) state mask*
TFDM = 0 – Tap engine ignores (Z-) axis
TFDM = 1 – Enable the (Z-) interrupt source

TFUM *– Tap Face Up (Z+) state mask*
TFUM = 0 – Tap engine ignores (Z+) axis
TFUM = 1 – Enable the (Z+) interrupt source

Reserved *– this bit is reserved, and its value should not be changed.*

INC4

Interrupt Control 4. This register controls routing of an interrupt reporting to physical interrupt pin INT1. Note that to properly change the value of this register, the PC1 bit in CNTL1 register must first be set to "0".

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
FFI1	BFI1	WMI1	DRDYI1	BTSI1	TDTI1	WUFI1	TPI1	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x25			

FFI1 *– Free fall interrupt reported on physical interrupt pin INT1*
FFI1 = 0 – disable
FFI1 = 1 – enable

BFI1 *– Buffer full interrupt reported on physical interrupt pin INT1*
BFI = 0 – disable
BFI = 1 – enable

WMI1 *– Watermark interrupt reported on physical interrupt pin INT1*
WMI1 = 0 – disable
WMI1 = 1 – enable
Note: WMI & BFI1 are level triggered interrupt source. If the valid condition persists, and the interrupt stays enabled, the interrupt will block any further interrupts from other sources from triggering the INT1 pin. To let other interrupt sources through, WMI/BFI1 needs to be cleared once detected.

DRDYI1 *– Data ready interrupt reported on physical interrupt pin INT1*
DRDYI1 = 0 – disable
DRDYI1 = 1 – enable

BTSI1 *– Back to sleep interrupt reported on physical interrupt pin INT1*
BTSI1 = 0 – disable
BTSI1 = 1 – enable

TDTI1 *– Tap/Double Tap interrupt reported on physical interrupt pin INT1*
TDTI1 = 0 – disable
TDTI1 = 1 – enable

WUFI1 *– Wake-Up (motion detect) interrupt reported on physical interrupt pin INT1*
WUFI1 = 0 – disable
WUFI1 = 1 – enable

TPI1 *– Tilt position interrupt reported on physical interrupt pin INT1*
TPI1 = 0 – disable
TPI1 = 1 – enable

INC5

Interrupt Control 5. This register controls the settings for the physical interrupt pin INT2. Note that to properly change the value of this register, the PC1 bit in CNTL1 register must first be set to "0".

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
PW21	PW20	IEN2	IEA2	IEL2	Reserved	ACLR2	ACLR1	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00010000
					Address: 0x26			

PW2<1:0> *– Pulse INT2 pin width configuration*

00 = 50µsec (10µsec if accelerometer ODR (OSA<3:0>) > 1600Hz)
*01 = 1 * OSA period*
*10 = 2 * OSA periods*
11 = Real time mode
When PW2 > 0, Interrupt source auto-clearing (ACLR2=1) should be set to keep consistency between the internal status and the physical interrupt.

IEN2 *– enables/disables the physical interrupt pin*

IEN2 = 0 – physical interrupt pin is disabled
IEN2 = 1 – physical interrupt pin is enabled

IEA2 *– Interrupt active level control for interrupt pin*

IEA2 = 0 – active LOW
IEA2 = 1 – active HIGH

IEL2 *– Interrupt latch control for interrupt pin*

IEL2 = 0 – latched
IEL2 = 1 – pulsed. The pulse width is configurable by PW2.

Reserved *– this bit is reserved, and its value should not be changed.*

ACLR2 *– Latched interrupt source information (INS1-INS3) is cleared and physical interrupt-2 latched pin is changed to its inactive state at pulse interrupt-2 trailing edge. Note: WMI and BFI are not auto-cleared by a pulse interrupt trailing edge.*

ACLR2 = 0 – disable
ACLR2 = 1 – enable

ACLR1 *– Latched interrupt source information (INS1-INS3) is cleared and physical interrupt-1 latched pin is changed to its inactive state at pulse interrupt-1 trailing edge. Note: WMI and BFI are not auto-cleared by a pulse interrupt trailing edge.*

ACLR1 = 0 – disable
ACLR1 = 1 – enable

INC6

Interrupt Control 6. This register controls routing of interrupt reporting to physical interrupt pin INT2. Note that to properly change the value of this register, the PC1 bit in CNTL1 register must first be set to "0".

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
FFI2	BFI2	WMI2	DRDYI2	BTSI2	TDTI2	WUFI2	TPI2	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x27			

FFI2 *– Free fall interrupt reported on physical interrupt pin INT2*
FFI2 = 0 – disable
FFI2 = 1 – enable

BFI2 *– Buffer full interrupt reported on physical interrupt pin INT2*
BF2 = 0 – disable
BF2 = 1 – enable

WMI2 *– Watermark interrupt reported on physical interrupt pin INT2*
WMI2 = 0 – disable
WMI2 = 1 – enable

Note: WMI is a level triggered interrupt source. If the valid condition persists, and the interrupt stays enabled, the interrupt will block any further interrupts from other sources from triggering the INT2 pin. To let other interrupt sources through, WMI needs to be cleared once detected.

DRDYI2 *– Data ready interrupt reported on physical interrupt pin INT2*
DRDYI2 = 0 – disable
DRDYI2 = 1 – enable

BTSI2 *– Back to sleep interrupt reported on physical interrupt pin INT2*
BTSI2 = 0 – disable
BTSI2 = 1 – enable

TDTI2 *– Tap/Double Tap interrupt reported on physical interrupt pin INT2*
TDTI2 = 0 – disable
TDTI2 = 1 – enable

WUFI2 *– Wake-Up (motion detect) interrupt reported on physical interrupt pin INT2*
WUFI2 = 0 – disable
WUFI2 = 1 – enable

TPI2 *– Tilt position interrupt reported on physical interrupt pin INT2*
TPI2 = 0 – disable
TPI2 = 1 – enable

1.15 TILT_TIMER (0X29)

Initial count register for the tilt position state timer (0 to 255 counts). Every count is calculated as 1/ODR delay period, where the ODR is user-defined per Table 9. The sensor must remain for the duration of the timer count in the new tilt position before the change is accepted. This register is On-The-Fly (OTF) register and can be written to while the KX132-1211 is enabled (PC1 bit in CNTL1 register is set to "1") and the change will be accepted with no interruption in the operation.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
TSC7	TSC6	TSC5	TSC4	TSC3	TSC2	TSC1	TSC0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x29			

1.16 TAP / DOUBLE-TAP CONTROL REGISTERS (0x2A – 0x31)

The Tap™/Double-Tap™ engine is enabled with TDTE bit in CNTL1 register and can be configured via dedicated set of control registers 0x2A – 0x31. Please refer to *Directional-Tap Detection Feature Description* section for detailed information on the Tap™/Double-Tap™ engine. These registers are On-The-Fly (OTF) registers and can be written to while the KX132-1211 is enabled (PC1 bit in CNTL1 register is set to "1") and the change will be accepted with no interruption in the operation.

TDTRC

Tap™/Double-Tap™ Report Control.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
0	0	0	0	0	0	DTRE	STRE	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000011
					Address: 0x2A			

DTRE *– enables/disables the double tap interrupt*
DTRE = 0 – do not update INS1 or DTDS if double tap occurs
DTRE = 1 – update INS1 and DTDS in INS2 with double tap events

STRE *– enables/disables single tap interrupt*
STRE = 0 – do not update INS1 or DTDS if single tap occurs.
STRE = 1 – update INS1 and DTDS in INS2 single tap events

TDTC

This register contains counter information for the detection of a double tap event. When the Directional-Tap™ ODR is 400Hz or less, every count is calculated as 1/ODR delay period. When the Directional-Tap™ ODR is 800Hz, every count is calculated as 2/ODR delay period. When the Directional-Tap™ ODR is 1600Hz, every count is calculated as 4/ODR delay period. The Directional-Tap™ ODR is user-defined per Table 10. The TDTC counts starts at the beginning of the fist tap and it represents the minimum time separation between the first tap and the second tap in a double tap event. More specifically, the second tap event must end outside of the TDTC. The Kionix recommended default value is 0.3 seconds (0x78).

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
TDTC7	TDTC6	TDTC5	TDTC4	TDTC3	TDTC2	TDTC1	TDTC0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	01111000
					Address: 0x2B			

TTH

The Tap Threshold High (TTH) register represents the 8-bit jerk high threshold to determine if a tap is detected. The value is compared against the upper 8 bits of the 4g output value (independent of the actual g-range setting of the device). Though this is an 8-bit register, the register value is internally multiplied by two to set the high threshold. This multiplication results in a range of 0 to 510 with a resolution of two counts. The Performance Index (PI) is the jerk signal that is expected to be less than this threshold, but greater than the TTL threshold during single and double tap events. Equation 1 shows how to calculate the Performance Index. The Kionix recommended default value is 203 (0xCB). See *AN092 Getting Started* for recommended settings (LINK).

$$X' = X\ (\text{current}) - X\ (\text{previous})$$
$$Y' = Y\ (\text{current}) - Y\ (\text{previous})$$
$$Z' = Z\ (\text{current}) - Z\ (\text{previous})$$

$$PI = |X'| + |Y'| + |Z'|$$

Equation 1: Performance Index

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
TTH7	TTH6	TTH5	TTH4	TTH3	TTH2	TTH1	TTH0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	11001011
					Address: 0x2C			

TTL

The Tap Threshold Low (TTL) register represents the 8-bit (0-255) jerk low threshold to determine if a tap is detected. The value is compared against the upper 8 bits of the 4g output value (independent of the actual g-range setting of the device). The Performance Index (PI) is the jerk signal that is expected to be greater than this threshold and less than the TTH threshold during single and double tap events. The Kionix recommended default value is 26 (0x1A). See *AN092 Getting Started* for recommended settings (LINK).

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
TTL7	TTL6	TTL5	TTL4	TTL3	TTL2	TTL1	TTL0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00011010
					Address: 0x2D			

FTD

This register contains counter information for the detection of any tap event. When the Directional-Tap™ ODR is 400Hz or less, every count is calculated as 1/ODR delay period. When the Directional-Tap™ ODR is 800Hz, every count is calculated as 2/ODR delay period. When the Directional-Tap™ ODR is 1600Hz, every count is calculated as 4/ODR delay period. The Directional-Tap™ ODR is user-defined per Table 10. To ensure that only tap events are detected, these time limits are used. A tap event must be above the performance index threshold for at least the low limit (FTDL0 – FTDL2) and no more than the high limit (FTDH0 – FTDH4). The Kionix recommended default value for the high limit is 0.05 seconds and for the low limit is 0.005 seconds (0xA2).

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
FTDH4	FTDH3	FTDH2	FTDH1	FTDH0	FTDL2	FTDL1	FTDL0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	10100010
					Address: 0x2E			

STD

This register contains counter information for the detection of a double tap event. When the Directional-Tap™ ODR is 400Hz or less, every count is calculated as 1/ODR delay period. When the Directional-Tap™ ODR is 800Hz, every count is calculated as 2/ODR delay period. When the Directional-Tap™ ODR is 1600Hz, every count is calculated as 4/ODR delay period. The Directional-Tap™ ODR is user-defined per Table 10. To ensure that only tap events are detected, this time limit is used. This register sets the total amount of time that the two taps in a double tap event can be above the PI threshold (TTL). The Kionix recommended default value for STD is 0.09 seconds (0x24).

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
STD7	STD6	STD5	STD4	STD3	STD2	STD1	STD0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00100100
					Address: 0x2F			

TLT

This register contains counter information for the detection of a tap event. When the Directional-Tap™ ODR is 400Hz or less, every count is calculated as 1/ODR delay period. When the Directional-Tap™ ODR is 800Hz, every count is calculated as 2/ODR delay period. When the Directional-Tap™ ODR is 1600Hz, every count is calculated as 4/ODR delay period. The Directional-Tap™ ODR is user-defined per Table 10. To ensure that only tap events are detected, this time limit is used. This register sets the total amount of time that the tap algorithm will count samples that are above the PI threshold (TTL) during a potential tap event. It is used during both single and double tap events. However, reporting of single taps on the physical interrupt pin INT1 or INT2 will occur at the end of the TWS. The Kionix recommended default value for TLT (TDT Latency Timer) is 0.1 seconds (0x28).

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
TLT7	TLT6	TLT5	TLT4	TLT3	TLT2	TLT1	TLT0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00101000
					Address: 0x30			

TWS

This register contains counter information for the detection of single and double taps. When the Directional-Tap™ ODR is 400Hz or less, every count is calculated as 1/ODR delay period. When the Directional-Tap™ ODR is 800Hz, every count is calculated as 2/ODR delay period. When the Directional-Tap™ ODR is 1600Hz, every count is calculated as 4/ODR delay period. The Directional-Tap™ ODR is user-defined per Table 10. It defines the time window for the entire tap event, single or double, to occur. Reporting of single taps on the physical interrupt pin INT1 or INT2 will occur at the end of this tap window. The Kionix recommended default value for TWS is 0.4 seconds (0xA0).

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
TWS7	TWS6	TWS5	TWS4	TWS3	TWS2	TWS1	TWS0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	10100000
					Address: 0x31			

1.17 FREE FALL CONTROL REGISTERS (0x32 – 0x34)

The Free fall engine is enabled with FFIE bit in FFCNTL register and can be configured via control registers 0x32 – 0x34. Please refer to *Free fall Detect* section for detailed information on the Free fall engine.

FFTH

Free Fall Threshold. This register contains the threshold of the Free fall detection. This value is compared to the top 8 bits of the accelerometer 8g output (independent of the actual g-range setting of the device). This register is On-The-Fly (OTF) register and can be written to while the KX132-1211 is enabled (PC1 bit in CNTL1 register is set to “1”) and the change will be accepted with no interruption in the operation.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
FFTH7	FFTH6	FFTH5	FFTH4	FFTH3	FFTH2	FFTH1	FFTH0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x32			

FFC

Free Fall Counter. This register contains the counter setting of the Free fall detection. Every count is calculated as 1/ODR delay period where ODR is set bit OFFI<2:0> in FFCNTL register. This register is On-The-Fly (OTF) register and can be written to while the KX132-1211 is enabled (PC1 bit in CNTL1 register is set to “1”) and the change will be accepted with no interruption in the operation.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
FFC7	FFC6	FFC5	FFC4	FFC3	FFC2	FFC1	FFC0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x33			

FFCNTL

Free Fall Control. This register provides the main control of the free fall engine. Note that to properly change the value of these registers, the PC1 bit in CNTL1 register must first be set to "0".

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
FFIE	ULMODE	FFDC1	FFDC0	DCRM	OFFI2	OFFI1	OFFI0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x34			

FFIE *– Free fall engine enable*

FFIE = 0 – disable
FFIE = 1 – enable

ULMODE *– Free fall interrupt latch/un-latch control*

ULMODE = 0 – latched
ULMODE = 1 – unlatched

DCRM *– Debounce methodology control*

DCRM = 0 – count up/down
DCRM = 1 – count up/reset

FFDC<1:0> *– Free fall interrupt delayed clear duration for unlatched mode*

00 = 0 sec delay
01 = 1 sec delay
10 = 2 sec delay
11 = 4 sec delay

OFFI<2:0> *– Output Data Rate at which the Free fall engine performs its function. The default Free fall ODR is 12.5Hz.*

OFFI	Output Data Rate (Hz)
000	12.5
001	25
010	50
011	100
100	200
101	400
110	800
111	1600

Table 14: Free Fall Function Output Data Rate

1.18 TILT ANGLE CONTROL REGISTERS (0X37 – 0X39)

The Tilt engine is enabled with TPE bit in CNTL1 register and can be configured via control registers 0x37 – 0x39. Please refer to *Orientation Detection Feature* section for detailed information on the Tilt engine. These registers are On-The-Fly (OTF) registers and can be written to while the KX132-1211 is enabled (PC1 bit in CNTL1 register is set to "1") and the change will be accepted with no interruption in the operation.

TILT_ANGLE_LL

Tilt Angle Low Limit: This register sets the low-level threshold for tilt angle detection. The low-level threshold value is compared against the upper 8 bits of the 4g output value (independent of the actual g-range setting of the device). The default tilt angle low level threshold is set to 22° from the horizontal. Note that the minimum suggested tilt angle is 10°. See *AN092 Getting Started* for recommended settings (LINK).

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
LL7	LL6	LL5	LL4	LL3	LL2	LL1	LL0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00001100
					Address: 0x37			

TILT_ANGLE_HL

Tilt Angle High Limit: This register sets the high-level threshold for tilt angle detection. The high-level threshold is used by an internal algorithm to eliminate dynamic g-variations caused by the device movement. Instead, only static g-variation (gravity) caused by the actual tilt changes are used. The high-level threshold value is compared against the upper 8 bits of the 4g output value (independent of the actual g-range setting of the device). The default tilt angle high level threshold is set to just above 1g plus some margin of error to account for external factors (e.g. device mounting). See *AN092 Getting Started* for recommended settings (LINK).

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
HL7	HL6	HL5	HL4	HL3	HL2	HL1	HL0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00101010
					Address: 0x38			

HYST_SET

Hysteresis Setting: This register sets the hysteresis that is placed in between the Screen Rotation states. The KX132-1211 ships from the factory with HYST_SET set to ±15° of hysteresis.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reserved	Reserved	HYST5	HYST4	HYST3	HYST2	HYST1	HYST0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00010100
					Address: 0x39			

Reserved *– these bits are reserved, and their values should not be changed.*

1.19 LP_CNTL1 (0X3A)

Low Power Control 1: The Averaging Filter Control setting determines both - the number of internal acceleration samples to be averaged in Low Power mode and the number of internal acceleration samples to be averaged for digital engines operation (Directional-Tap™ / Double-Tap™, Tilt, Wake-Up, Back-to-Sleep, Free fall, Advanced Data Path) both in *High Performance* and *Low Power* modes. In Low Power mode, this setting has a direct effect on power consumption and noise performance and thus can be used to optimize the performance of the accelerometer. Note that to properly change the value of this register, the PC1 bit in CNTL1 must first be set to "0".

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reserved	AVC2	AVC1	AVC0	Reserved	Reserved	Reserved	Reserved	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	01000011
					Address: 0x3A			

AVC<2:0> *– Averaging Filter Control.*

```
000 = No Averaging (requires IIR_BYPASS = 0 setting in ODCNTL register)
001 = 2 Samples Averaged
010 = 4 Samples Averaged
011 = 8 Samples Averaged
100 = 16 Samples Averaged (default)
101 = 32 Samples Averaged
110 = 64 Samples Averaged
111 = 128 Samples Averaged
```

Reserved *– these bits are reserved, and their values should not be changed.*

1.20 LP_CNTL2 (0X3B)

Low Power Control 2: The advanced low power control setting reduces the power consumption of the KX132-1211 even further in Low Power and Standby modes. Note: This setting cannot be used in Low Power mode if any of the following digital engines is enabled: Advanced Data Path, Tap™/Double-Tap™, Free fall, or Tilt. This setting can be used with Wake-up / Back-to-Sleep engines. This setting has no effect in High-performance mode. Note that to properly change the value of this register, the PC1 bit in CNTL1 must first be set to "0".

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	LPSTPSEL	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	10011010
					Address: 0x3B			

LPSTPSEL *– Digital power shut-off select*

LPSTPLEL = 0 – Digital power shut-off disabled (default)
LPSTPLEL = 1 – Digital power shut-off enabled.

Reserved *– these bits are reserved, and their values should not be changed.*

1.21 WAKE-UP & BACK-TO-SLEEP THRESHOLD AND COUNTER SETUP REGISTERS (0x49 – 0x4D)

The threshold and counter values of the Wake-up and Back-to-Sleep engines of the KX132-1211 can be configured via registers 0x49 – 0x4D. See *Motion Interrupt Feature Description* section for additional details on how to configure these registers. These registers are On-The-Fly (OTF) registers and can be written to while the KX132-1211 is enabled (PC1 bit in CNTL1 register is set to "1") and the change will be accepted with no interruption in the operation.

WUFTH, BTSWUFTH, BTSTH

Wake-up/Back-to-sleep engine thresholds. See *Equation 7* in Motion Interrupt Feature Description section for details on how to configure these registers.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Address	Register	Reset Value
WUFTH7	WUFTH6	WUFTH5	WUFTH4	WUFTH3	WUFTH2	WUFTH1	WUFTH0	0x49	**WUFTH**	10000000
0	BTSTH10	BTSTH9	BTSTH8	0	WUFTH10	WUFTH9	WUFTH8	0x4A	**BTSWUFTH**	00000000
BTSTH7	BTSTH6	BTSTH5	BTSTH4	BTSTH3	BTSTH2	BTSTH1	BTSTH0	0x4B	**BTSTH**	10000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0			

WUFTH<10:0> – *Wake-Up Function Threshold (WUFTH)* value

BTSTH<10:0> – *Back-To-Sleep function Threshold* (BTSTH) value

BTSC

Debounce counter register for the Back-to-Sleep (BTS) engine. See *Equation 5* in Motion Interrupt Feature Description section for details on how to configure this register.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
BTSC7	BTSC6	BTSC5	BTSC4	BTSC3	BTSC2	BTSC1	BTSC0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x4C			

WUFC

Debounce counter register for the Wake-up Function (WUF) engine. See *Equation 5* in Motion Interrupt Feature Description section for details on how to configure this register.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
WUFC7	WUFC6	WUFC5	WUFC4	WUFC3	WUFC2	WUFC1	WUFC0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x4D			

1.22 SELF_TEST (0X5D)

Self-Test Enable register.

W	W	W	W	W	W	W	W	
0	0	0	0	0	0	0	0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x5D			

To perform the self-test, the following procedure is required:

1. Set STPOL bit to 1 in INC1 register to set the polarity of the self-test mode to positive.
2. Set PC1 bit to 1 in CNTL1 register to enable KX132-1211.
3. Write 0xCA to this register to enable the MEMS self-test function.

Once the self-test function is enabled, electrostatic-actuation of the accelerometer, results in a DC shift of the X, Y and Z axis outputs. Calculate the self-test (ST) response using Equation 2:

$$ST[g] = \frac{\left(OUTPUT_{ST\_ON}[counts]\right) - \left(OUTPUT_{ST\_OFF}[counts]\right)}{Sensitivity\left[\frac{counts}{g}\right]}$$

Equation 2: Self-Test (ST) Response Calculation

The self test response should be compared to the product specifications to determine if the MEMS response is within the specified range (consult Mechanical Specification table of KX132-1211 product specifications).

To disable the self-test mode any of the following methods can be used:

- Power cycle KX132-1211

  Or

- Perform software reset by setting SRST bit to 1 in CNTL 2 register

  Or

1. Set PC1 bit to 0 in CNTL1 register to set KX132-1211 in Standby mode.
2. Write 0x00 to this register to disable the self-test mode.

Note, this is a write-only register. Read back value from this register will always be 0x00.

1.23 OUTPUT BUFFER REGISTERS (0x5E – 0x63)

BUF_CNTL1

Read/write control register that controls the buffer sample threshold. This register is On-The-Fly (OTF) register and can be written to while the KX132-1211 is enabled (PC1 bit in CNTL1 register is set to "1") and the change will be accepted with no interruption in the operation.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
SMP_TH7	SMP_TH6	SMP_TH5	SMP_TH4	SMP_TH3	SMP_TH2	SMP_TH1	SMP_TH0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x5E			

SMP_TH[7:0] Sample Threshold *– determines the number of samples that will trigger a watermark interrupt or will be saved prior to a trigger event. When BRES=1, the maximum number of samples is 86; when BRES=0, the maximum number of samples is 171. The minimum number of samples must be greater than or equal to 2.*

Buffer Model	Sample Function
Bypass	None
FIFO	Specifies how many buffer samples are needed to trigger a watermark interrupt.
Stream	Specifies how many buffer samples are needed to trigger a watermark interrupt.
Trigger	Specifies how many buffer samples before the trigger event are retained in the buffer.

Table 15: Sample Threshold Operation by Buffer Mode

BUF_CNTL2

Read/write control register that controls sample buffer operation. This register is On-The-Fly (OTF) register and can be written to while the KX132-1211 is enabled (PC1 bit in CNTL1 register is set to "1") and the change will be accepted with no interruption in the operation.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
BUFE	BRES	BFIE	0	0	0	BM1	BM0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x5F			

BUFE *− controls activation of the sample buffer.*

BUFE = 0 – sample buffer inactive
BUFE = 1 – sample buffer active

Note 1: Disabling the sample buffer (BUFE = 0) will clear the buffer. The buffer will also be cleared (1) following write to BUF_CLEAR register and/or (2) after setting PC1 bit in CNTL1 register to 0 (standby mode).

Note 2: Additional control of data to be buffered is available via ADP_BUF_SEL bit7 in ADP_CNTL2 register.

BRES *– determines the resolution of the acceleration data samples collected by the sample buffer.*

BRES = 0 – 8-bit samples are accumulated in the buffer
BRES = 1 – 16-bit samples are accumulated in the buffer

BFIE *– buffer full interrupt enable bit*

BFIE = 0 – buffer full interrupt is disabled
BFIE = 1 – buffer full interrupt is enabled and updated in INS2

BM1, BM0 *selects the operating mode of the sample buffer per Table 16.*

BM1	BM0	Mode	Description
0	0	FIFO	The buffer collects 171 sets of 8-bit low resolution values or 86 sets of 16-bit high resolution values and then stops collecting data, collecting new data only when the buffer is not full.
0	1	Stream	The buffer holds the last 171 sets of 8-bit low resolution values or 86 sets of 16-bit high resolution values. Once the buffer is full, the oldest data is discarded to make room for newer data.
1	0	Trigger	When a trigger event occurs, the buffer holds the last data set of SMP[9:0] samples before the trigger event and then continues to collect data until full. New data is collected only when the buffer is not full.

Table 16: Selected Buffer Mode

BUF_STATUS_1 and BUF_STATUS_2

These register reports the status of the sample buffer. Note that BUF_STATUS_1 and BUF_STATUS_2 registers may have a delay of up to 1µsec to update the sample level after a buffer read.

R	R	R	R	R	R	R	R	**BUF_STATUS_1**
SMP_LEV7	SMP_LEV6	SMP_LEV5	SMP_LEV4	SMP_LEV3	SMP_LEV2	SMP_LEV1	SMP_LEV0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x60			

R	R	R	R	R	R	R	R	**BUF_STATUS_2**
BUF_TRIG	0	0	0	0	0	SMP_LEV9	SMP_LEV8	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x61			

SMP_LEV[9:0] − *Sample Level reports the number of data bytes that have been stored in the sample buffer. When BRES=1, this count will increase by 6 for each 3-axis sample in the buffer; when BRES=0, the count will increase by 3 for each 3-axis sample. If this register reads 0, no data has been stored in the buffer.*

BUF_TRIG − *reports the status of the buffer's trigger function if this mode has been selected. A trigger event is the combined interrupt events of*
BUF_TRIG = FFS | TDTS1 | TDTS0 | WUFS | TPS | TRIG

BUF_CLEAR

Latched buffer status information and the entire sample buffer are cleared when any data is written to this register. This causes the sample level bits SMP_LEV[9:0] to be cleared in BUF_STATUS_1 and BUF_STATUS_2 registers. In addition, if the sample buffer is set to Trigger mode, the BUF_TRIG bit in BUF_STATUS_2 is cleared too. Finally, the BFI and WMI bits in INS2 will be cleared and physical interrupt latched pin will be changed to its inactive state. This register is On-The-Fly (OTF) register and can be written to while the KX132-1211 is enabled (PC1 bit in CNTL1 register is set to "1") and the change will be accepted with no interruption in the operation.

W	W	W	W	W	W	W	W	
X	X	X	X	X	X	X	X	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x62			

BUF_READ

Buffer output register: Data in the buffer can be read while continuing to fill according to the BRES and BM settings in BUF_CNTL2. To prevent any data loss, data must be read on a single byte basis or as complete datasets (6 bytes for high-resolution samples and 3 bytes for low-resolution samples) using auto-increment (burst read). In STREAM and TRIGGER (before the trigger event) modes any burst read of the buffer shall last no longer than the current 1/ODR cycle minus 30µsec (1/ODR-30µsec) for asynchronous reads and no longer than twice the current 1/ODR cycle minus 30µsec (2*(1/ODR)-30µsec) for synchronous reads. In FIFO mode, there is no restriction other than the buffer must not run out of space. Output data is in 2's Complement format.

R	R	R	R	R	R	R	R	
X	X	X	X	X	X	X	X	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x63			

1.24 ADVANCED DATA PATH CONTROL REGISTERS (0x64 – 0x76)

The advanced data path (ADP) engine of the KX132-1211 can be configured via control registers 0x64 – 0x76 shown below. The ADP engine is enabled by setting ADPE bit to 1 in CNTL5 register. The ADP output data can be routed to the output registers 0x02 – 0x07, and/or to the sample buffer. See *AN097 Getting Started with Advanced Data Path* for more information about recommended register setup. These registers are On-The-Fly (OTF) registers and can be written to while the KX132-1211 is enabled (PC1 bit in CNTL1 register is set to "1") and the change will be accepted with no interruption in the operation.

ADP_CNTL1

Advanced Data Path (ADP) Output Control register 1.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reserved	RMS_AVC2	RMS_AVC1	RMS_AVC0	OADP3	OADP2	OADP1	OADP0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x64			

Reserved *– this bit is reserved, and its value should not be changed.*

RMS_AVC<2:0> *– Number of samples used to calculate RMS output. Each sample is determined by the Advanced Data Path ODR as set by OADP<3:0> bits.*

Note 1: If ADP data is routed to the Wake-up / Back-to-Sleep engines, RMS_AVC<2:0> also sets the number of samples averaged for these engines.

Note 2: The input data to the Advanced Data Path is first averaged per AVC<2:0> setting in LP_CNTL1 (0X3A) register.

```
000 = 2 samples      100 =  32 samples
001 = 4 samples      101 =  64 samples
010 = 8 samples      110 =  128 samples
011 = 16 samples     111 =  256 samples
```

OADP<3:0> *– Output Data Rate (ODR) for Advanced Data Path. For filter-1 and filter-2 stages, the ODR set by OADP<3:0> is the effective ODR. For RMS block the effective ODR is scaled down by RMS_AVC<2:0> setting.*

Note 1: If ADP data is routed to the Wake-up / Back-to-Sleep engines, OADP<3:0> also sets the ODR for these engines.
Note 2: OADP setting needs to be ≤OSA to avoid irregular resulting acceleration ODRs.

```
0000 = 0.781Hz       1000 = 200Hz
0001 = 1.563Hz       1001 = 400Hz*
0010 = 3.125Hz       1010 = 800Hz*
0011 = 6.25Hz        1011 = 1600Hz*
0100 = 12.5Hz        1100 = 3200Hz*
0101 = 25Hz          1101 = 6400Hz*
0110 = 50Hz          1110 = 12800Hz*
0111 = 100Hz         1111 = 25600Hz*
```

** Higher ODR values will increase the power consumption of the sensor.*

ADP_CNTL2

Advanced Data Path (ADP) Control register 2*.*

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
ADP_BUF_SEL	ADP_WB_ISEL	RMS_WB_OSEL	ADP_FLT2_BYP	ADP_FLT1_BYP	0	ADP_RMS_OSEL	ADP_F2_HP	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000010
						Address: 0x65		

ADP_BUF_SEL *– Select data to be routed to the sample buffer*

ADP_BUF_SEL = 0 – Accelerometer data is routed to the sample buffer (default)
ADP_BUF_SEL = 1 – ADP data is routed to the sample buffer. Note, if ADP data sent to the sample buffer bypasses the RMS block (ADP_RMS_OSEL = 0), the data is buffered at ODR set by the OADP<3:0>. If not, the effective ODR is scaled down by RMS_AVC<2:0> setting.

ADP_WB_ISEL *– Input select for the Wake-up/Back-to-Sleep engines*

ADP_WB_ISEL = 0 – Accelerometer data is selected (ADP data is bypassed)
ADP_WB_ISEL = 1 – ADP data is selected. Note, ODR for the Wake-up / Back-to-Sleep would be set but OADP<3:0> and not by OWUF<2:0> and OBTS<2:0>.

RMS_WB_OSEL *– RMS select data for the Wake-up/Back-to-Sleep engines. Note: ADP_WB_ISEL bit should be set to 1 for this control to have an effect.*

RMS_WB_OSEL = 0 – Output data from ADP is not routed to the Wake-up/Back-to-Sleep engines
RMS_WB_OSEL = 1 – ADP RMS output data is routed to the Wake-up/Back-to-Sleep engines.

ADP_FLT2_BYP *– Advanced Data Path Filter-2 bypass control*

ADP_FLT2_BYP = 0 – Filter-2 is not bypassed
ADP_FLT2_BYP = 1 – Filter-2 is bypassed

ADP_FLT1_BYP *– Advanced Data Path Filter-1 bypass control*

ADP_FLT1_BYP = 0 – Filter-1 is not bypassed
ADP_FLT1_BYP = 1 – Filter-1 is bypassed

ADP_RMS_OSEL *– Select data out to XADP, YADP, ZADP registers and to the output buffer. Note, ADP_RMS_OSEL must be set to 1 if RMS_WB_OSEL is set to 1.*

ADP_RMS_OSEL = 0 – Filter output (RMS engine is bypassed)
ADP_RMS_OSEL = 1 – RMS output

ADP_F2_HP *– Filter-2 High-pass enable*

ADP_F2_HP = 0 – Filter-2 is set to Low-pass filter
ADP_F2_HP = 1 – Filter-2 is set to High-pass filter

ADP_CNTL3

Advanced Data Path (ADP) Control register 3.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
0	ADP_F1_1A6	ADP_F1_1A5	ADP_F1_1A4	ADP_F1_1A3	ADP_F1_1A2	ADP_F1_1A1	ADP_F1_1A0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x66			

ADP_F1_1A<6:0> *– ADP filter-1 coefficient (1/A)*

ADP_CNTL4, ADP_CNTL5, ADP_CNTL6

Advanced Data Path (ADP) Control registers 4, 5 and 6.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	**ADP_CNTL4**
ADP_F1_BA7	ADP_F1_BA6	ADP_F1_BA5	ADP_F1_BA4	ADP_F1_BA3	ADP_F1_BA2	ADP_F1_BA1	ADP_F1_BA0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x67			

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	**ADP_CNTL5**
ADP_F1_BA15	ADP_F1_BA14	ADP_F1_BA13	ADP_F1_BA12	ADP_F1_BA11	ADP_F1_BA10	ADP_F1_BA9	ADP_F1_BA8	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x68			

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	**ADP_CNTL6**
0	ADP_F1_BA22	ADP_F1_BA21	ADP_F1_BA20	ADP_F1_BA19	ADP_F1_BA18	ADP_F1_BA17	ADP_F1_BA16	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x69			

ADP_F1_BA<22:0> *– ADP filter-1 coefficient (B/A)*

.

ADP_CNTL7, ADP_CNTL8, ADP_CNTL9

Advanced Data Path (ADP) Control registers 7, 8 and 9.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	**ADP_CNTL7**
ADP_F1_CA7	ADP_F1_CA6	ADP_F1_CA5	ADP_F1_CA4	ADP_F1_CA3	ADP_F1_CA2	ADP_F1_CA1	ADP_F1_CA0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x6A			

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	**ADP_CNTL8**
ADP_F1_CA15	ADP_F1_CA14	ADP_F1_CA13	ADP_F1_CA12	ADP_F1_CA11	ADP_F1_CA10	ADP_F1_CA9	ADP_F1_CA8	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x6B			

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	**ADP_CNTL9**
0	ADP_F1_CA22	ADP_F1_CA21	ADP_F1_CA20	ADP_F1_CA19	ADP_F1_CA18	ADP_F1_CA17	ADP_F1_CA16	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x6C			

ADP_F1_CA<22:0> *– ADP filter-1 coefficient (C/A)*

ADP_CNTL10

Advanced Data Path (ADP) Control register 10.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
0	0	0	ADP_F1_ISH4	ADP_F1_ISH3	ADP_F1_ISH2	ADP_F1_ISH1	ADP_F1_ISH0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x6D			

ADP_F1_ISH<4:0> *– ADP filter-1 input scale shift value*

ADP_CNTL11

Advanced Data Path (ADP) Control register 11.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
ADP_F1_OSH	ADP_F2_1A6	ADP_F2_1A5	ADP_F2_1A4	ADP_F2_1A3	ADP_F2_1A2	ADP_F2_1A1	ADP_F2_1A0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x6E			

ADP_F2_1A<6:0> *– ADP filter-2 coefficient (1/A)*

ADP_F1_OSH *– ADP filter-1 output scale shift value*

36 Thornwood Dr. – Ithaca, NY 14850
tel: 607-257-1080 – fax:607-257-1146
www.kionix.com - info@kionix.com

ADP_CNTL12, ADP_CNTL13

Advanced Data Path (ADP) Control registers 12 and 13.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	**ADP_CNTL12**
ADP_F2_BA7	ADP_F2_BA6	ADP_F2_BA5	ADP_F2_BA4	ADP_F2_BA3	ADP_F2_BA2	ADP_F2_BA1	ADP_F2_BA0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x6F			

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	**ADP_CNTL13**
0	ADP_F2_BA14	ADP_F2_BA13	ADP_F2_BA12	ADP_F2_BA11	ADP_F2_BA10	ADP_F2_BA9	ADP_F2_BA8	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x70			

ADP_F2_BA<14:0> *– ADP filter-2 coefficient (B/A)*

ADP_CNTL14, ADP_CNTL15, ADP_CNTL16, ADP_CNTL17

Advanced Data Path (ADP) Control register 14, 15, 16 and 17. These registers are purposely set to a value of 0.

R	R	R	R	R	R	R	R	
0	0	0	0	0	0	0	0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x71, 0x72, 0x73, 0x74			

ADP_CNTL18

Advanced Data Path (ADP) Control register 18.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
0	0	0	ADP_F2_ISH4	ADP_F2_ISH3	ADP_F2_ISH2	ADP_F2_ISH1	ADP_F2_ISH0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x75			

ADP_F2_ISH<4:0> *– ADP filter-2 input scale shift value*

ADP_CNTL19

Advanced Data Path (ADP) Control register 19.

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
0	0	0	ADP_F2_OSH4	ADP_F2_OSH3	ADP_F2_OSH2	ADP_F2_OSH1	ADP_F2_OSH0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000
					Address: 0x76			

ADP_F2_OSH<4:0> *– ADP filter-2 output scale shift value*

2 Embedded Applications

2.1 Orientation Detection Feature

The orientation detection feature of the KX132-1211 will report changes in face up, face down, ± vertical and ± horizontal orientation. This intelligent embedded algorithm considers very important factors that provide accurate orientation detection from low cost tri-axis accelerometers. Factors such as: hysteresis, device orientation angle and delay time are described below as these techniques are utilized inside the KX132-1211.

Hysteresis

A 45° tilt angle threshold seems like a good choice because it is halfway between 0° and 90°. However, a problem arises when the user holds the device near 45°. Slight vibrations, noise and inherent sensor error will cause the acceleration to go above and below the threshold rapidly and randomly, so the screen will quickly flip back and forth between the 0° and the 90° orientations. This problem is avoided in the KX132-1211 by choosing a 30° threshold angle. With a 30° threshold, the screen will not rotate from 0° to 90° until the device is tilted to 60° (30° from 90°). To rotate back to 0°, the user must tilt back to 30°, thus avoiding the screen flipping problem. This example essentially applies ± 15° of hysteresis in between the four screen rotation states. Table 17 shows the acceleration limits implemented for ϕ_T =30°.

Orientation	X Acceleration (g)	Y Acceleration (g)
0°/360°	-0.5 < a_x < 0.5	a_y > 0.866
90°	a_x > 0.866	-0.5 < a_y < 0.5
180°	-0.5 < a_x < 0.5	a_y < -0.866
270°	a_x < -0.866	-0.5 < a_y < 0.5

Table 17: Acceleration at the four orientations with ± 15° of hysteresis

The KX132-1211 allows the user to change the amount of hysteresis in between the four screen rotation states. By simply writing to the HYST_SET register, the user can adjust the amount of hysteresis up to ± 45°. The plot in Figure 2 shows the typical amount of hysteresis applied for a given digital count value of HYST_SET.

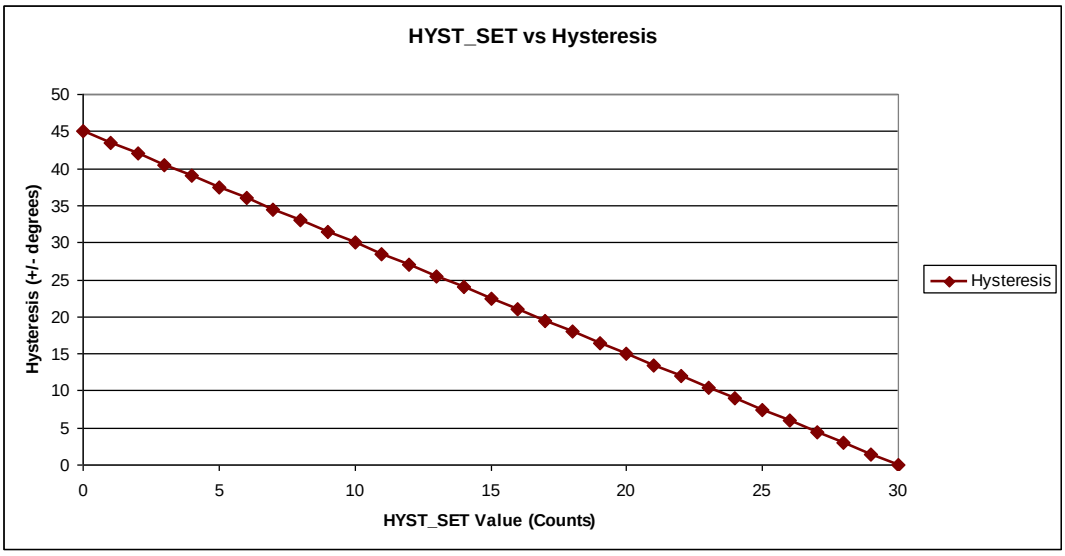

Figure 2: HYST_SET vs Hysteresis

Device Orientation Angle (Tilt Angle)

To ensure that horizontal and vertical device orientation changes are detected, even when it isn't in the ideal vertical orientation – where the angle θ in Figure 3 is 90°, the KX132-1211 considers device orientation angle in its algorithm.

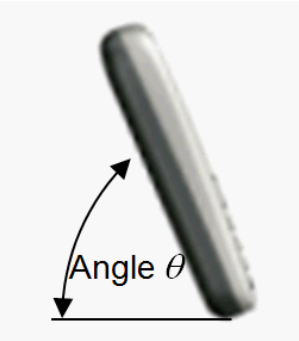

Figure 3: Device Orientation Angle

As the angle in Figure 3 is decreased, the maximum gravitational acceleration on the X-axis or Y-axis will also decrease. Therefore, when the angle becomes small enough, the user will not be able to make the screen orientation change. When the device orientation angle approaches 0° (device is flat on a desk or table), $a_x = a_y = 0g$, $a_z = +1g$, and there is no way to determine which way the screen should be oriented, the internal algorithm determines that the device is in either the face-up or face-down orientation, depending on the sign of the z-axis. The KX132-1211 will only change the screen orientation when the orientation angle is above the factory-defaulted/user-defined threshold set in the TILT_ANGLE_LL register. Equation 3 can be used to determine what value to write to the TILT_ANGLE_LL register to set the device orientation angle. The value for HL is preset at the factory but can be adjusted in special cases (e.g. to reduce the effect of transient g-variation such as when device is being moved rather than just being rotated).

$$\text{TILT\_ANGLE\_LL (counts)} = \sin\theta * (32\ \text{(counts/g)})$$

Equation 3: Tilt Angle Threshold

Tilt Timer

The 8-bit register, TILT_TIMER can be used to qualify changes in orientation. The KX132-1211 does this by incrementing a counter with a size that is specified by the value in TILT_TIMER for each set of acceleration samples to verify that a change to a new orientation state is maintained. A user defined Tilt Position output data rate (ODR) as set by OTP<1:0> bits in CNTL3 register, determines the time period for each sample. Equation 4 shows how to calculate the TILT_TIMER register value for a desired delay time.

$$\text{TILT\_TIMER (counts)} = \text{Delay Time (sec)} \times \text{Tilt Position ODR (Hz)}$$

Equation 4: Tilt Position Delay Time

2.2 Motion Interrupt Feature Description

KX132-1211 features an advanced threshold interrupt by the internal Wake-Up and Back-to-Sleep digital engines. These engines allow the KX132-1211 to trigger interrupts when accelerometer activity falls below a defined threshold window (Back-to-Sleep) or exceeds a threshold window (Wake-Up event). Note that this function only generates an interrupt and doesn't trigger any changes to the part configuration (e.g. power mode, ODR, etc.).

Enabling / Disabling

The Wake-up and Back-to-sleep detection can be enabled/disabled using WUFE and BTSE bits in CNTL4 register and the direction of motion detection can be set for any axis in INC2 register.

Debounce Counter

The Wake-Up and Back-to-Sleep digital engines have an internal debounce counter to qualify motion status detection. The debounce counter function can be set using C_MODE bit in CNTL4 register. The counter can be configured to either reset or decrement itself if accelerometer data has either fallen below or risen above the threshold for wake-up or back-to-sleep functionality respectively. Note that each Wake-Up Function Counter (WUFC) count qualifies 1 (one) user-defined Wake-Up Function ODR period as set by OWUF<2:0> bits in CNTL3 register. Similarly, each Back-to-Sleep Counter (BTSC) count qualifies 1 (one) user-defined Back-to-Sleep function ODR period as set by OBTS<2:0> bits in CNTL4 register. Equation 5 shows how to calculate the WUFC and BTSC register values for a desired Wake-Up and Back-to-Sleep delay times.

WUFC (counts) = Wake-Up Delay Time (sec) x Wake-up Function ODR (Hz)

BTSC (counts) = Back-to-Sleep Delay Time (sec) x Back-to-Sleep Function ODR (Hz)

Equation 5: Wake-Up and Back-to-Sleep counts

Pulse Reject Mode

The Wake-Up and Back-to-Sleep digital engines can be configured to ignore pulse-like motion using PR_MODE bit in CNTL4 register. This mode is only available if both positive and negative motion directions are enabled for any particular axis.

Integration with Advanced Data Path (ADP)

KX132-1211 features a user-configurable 3-stage ADP consisting of a 2nd order low-pass filter, 1st order low-pass or high-pass filter, and RMS calculation block. Each stage of the ADP is individually configured and can be bypassed using the corresponding register settings. The output from the Advanced Data Path can be routed to the Wake-Up / Back-to-Sleep engines using a series of internal MUXes. Please refer to *AN109 Introduction to Advanced Data Path* and *AN097 Getting Started with Advanced Data Path* application notes for details.

Threshold Resolution

The motion interrupt threshold values are set by WUFTH<10:0> and BTSTH<10:0> bits in WUFTH, BTSWUFTH, BTSTH registers. The values in these registers are compared to the top 11 bits of the accelerometer 8g output (regardless of GSEL<1:0> setting in CNTL1 register. This results in threshold resolution of 3.9 mg/count per Equation 6.

$$2^{11}\ counts\ /\ 8\ g = 2048\ counts\ /\ 8\ g = 256\ counts/g\ or\ 3.9\ mg/count$$

Equation 6: Wake-Up / Back-to-Sleep Resolution Calculations

Threshold Calculation

To calculate the desired wake-up threshold (WUFTH) and back-to-sleep threshold (BTSTH), use Equation 7. Please note that the wake-up engine function is independent of the user selected g-range.

WUFTH (counts) = Wake-Up Threshold (g) x 256 (counts/g)

BTSTH (counts) = Back-to-Sleep Threshold (g) x 256 (counts/g)

Equation 7: Wake-Up/Back-to-Sleep Threshold

Relative / Absolute Threshold Modes Select

The type of threshold used for motion interrupt is controlled using TH_MODE bit in CNTL4 register. The threshold can be set to either an absolute acceleration value or a relative acceleration value.

2.2.1.1 Relative Threshold Mode

When KX132-1211 operates in the Relative Threshold mode, the Motion interrupt engine reports qualified changes in the high-pass filtered acceleration (i.e. acceleration difference between the two consecutive samples) based on the Wake-Up Threshold (WUFTH) and Back-to-Sleep Threshold (BTSTH). If the high-pass filtered acceleration on any axis is greater than the user-defined Wake-Up Threshold (WUFTH), the device has transitioned from an inactive state to an active state. On the other hand, if the high-pass filtered acceleration on any axis is less than the user-defined Back-to-Sleep Threshold (BTSTH), the device has transitioned from an active state to an inactive state.

To illustrate how the algorithm works, consider the Figure 4 that shows the latched response (active LOW) of the motion detection algorithm. In this example, the debounce counter (WUFC) is set to 8 counts and the counter mode is set to *reset* (C_MODE = 0). The pointing up red arrow represents the *relative* Wake-Up Function Threshold (WUFTH) value in units of gravity (g). While the accelerometer data is collected at the output data rate set by OSA<3:0> bits in the ODCNTL register, the wake-up engine evaluates the acceleration samples at the sampling rate set by OWUF<2:0> bits in CNTL3 register. Once the algorithm starts to run, on the 1st wake-up function ODR clock (WUF ODR) cycle, the acceleration motion is equal to some *init* value (shown in units of g). On the 2nd clock cycle, the motion is still at the same *init* (g) level. The relative difference between the current (2nd clock) sample the previous (1st clock) sample, is 0g and it is below the threshold value. On the 3rd clock cycle, the motion has increased by g1 (g) and is equal to *init+g1* (g). The relative difference between the current and previous samples is still below the threshold and thus the debounce counter remains inactive. Fast forward to the 6th clock cycle, the acceleration has increased by *g2* (g) relatively to the previous sample. Note that *g2* (g) difference is higher than threshold. As a result, the debounce counter begins its count. The new *reference* sample for the algorithm now becomes the acceleration sample obtained on previous clock cycle 5. While the debounce counter continues the count, the difference between each *new* acceleration sample and the *reference* sample is calculated and is measured against the threshold. The difference has remained higher than threshold on the 7th clock cycle but below it on the 8th clock cycle. This resulted in reset of the debounce counter and

reset of the reference sample. At this point, the algorithm starts evaluating the difference between the current and the previous acceleration samples again. On the 9th clock cycle, the difference has exceeded the threshold. The sample evaluated on the 8th clock cycle becomes a new *reference* sample for the algorithm. The debounce counter starts the count on the next clock cycle (9th). During the next 8 clock cycles (9th-16th), the acceleration difference between each new sample and the *references* sample (8th clock cycle) has remained above the threshold and the debounce counter has incremented at each clock. Once the count value reached the WUFC value of 8, the wake-up interrupt was issued on the next clock cycle (17th clock cycle). In the example, since the interrupt was set to active low, the interrupt line was pulled low to indicate an interrupt. The operation of the back-to-sleep functionality is equivalent to the wake-up function.

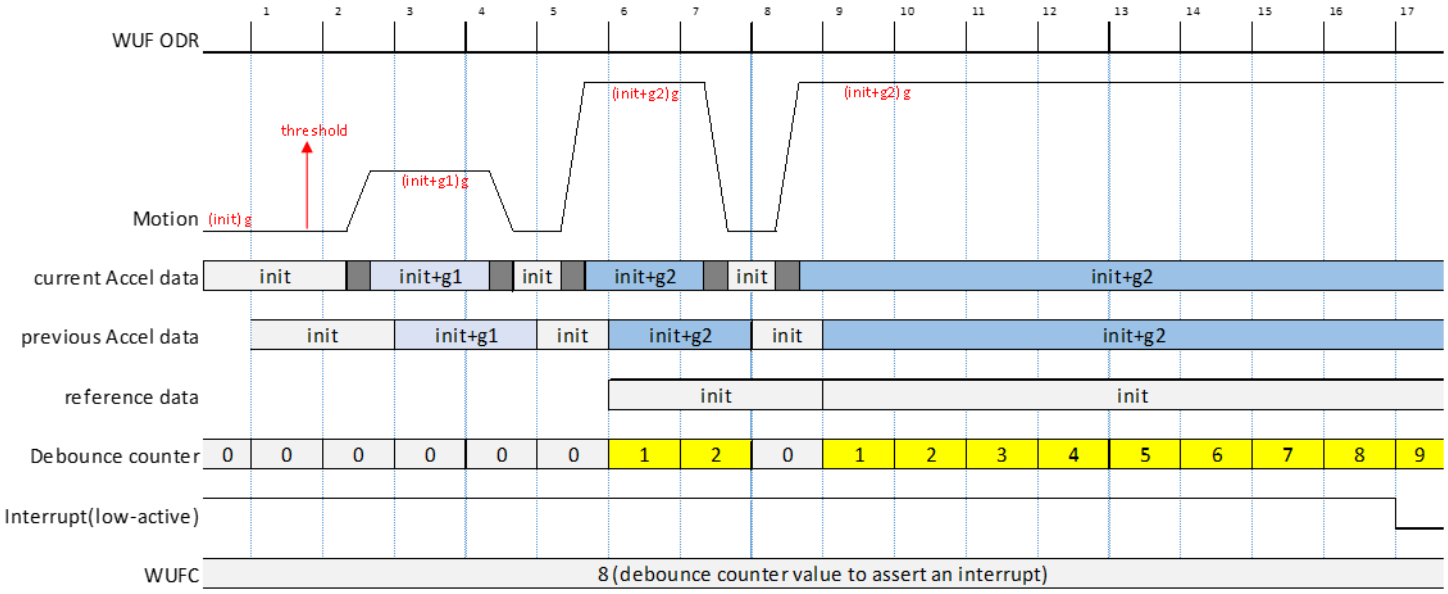

Figure 4: Latched Motion Interrupt Response in the Relative Threshold Mode

In cases when acceleration changes are gradual relatively to the wake-up, the motion may not get detected because the difference between consecutive samples is below the threshold value. In such cases, consider reducing the motion wake-up sampling rate as set by OWUF<2:0> bits in CNTL3 register or using the absolute threshold mode. Consider the motion shown in the Figure 5. Even though the difference between the initial acceleration measured on the 2nd clock (*init*) and final sample measured at the 7th clock (*initi+g2)* is higher than the threshold, the difference between consecutive intermediate samples (4th-3rd, 5th-4th, 6th-5th, 7th-6th) was below the threshold. Thus, the condition was never met to start the debounce counter.

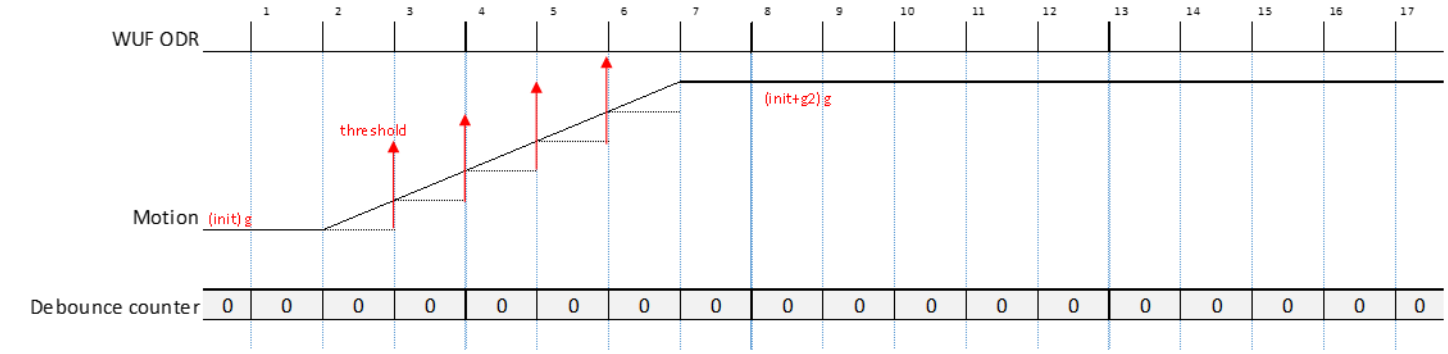

Figure 5: Motion Interrupt Response in the Relative Threshold Mode with Slow Acceleration Transition

2.2.1.2 Absolute Threshold Mode

The Absolute Threshold Mode of the KX132-1211 operates similar to the Relative Threshold Mode described in details before with the only difference being that each acceleration sample is being compared to the same user defined wake-up threshold (WUFTH) or back-to-sleep threshold (BTSTH) regardless of previous sample. Once the acceleration sample exceeds the threshold, the debounce counter begins the count. If at any point during the count, the acceleration sample falls below the threshold, the counter will get either reset or decremented based on the C_MODE bit setting. Once the debounce counter reaches the user defined count (WUFC), the interrupt is issued as shown in Figure 6.

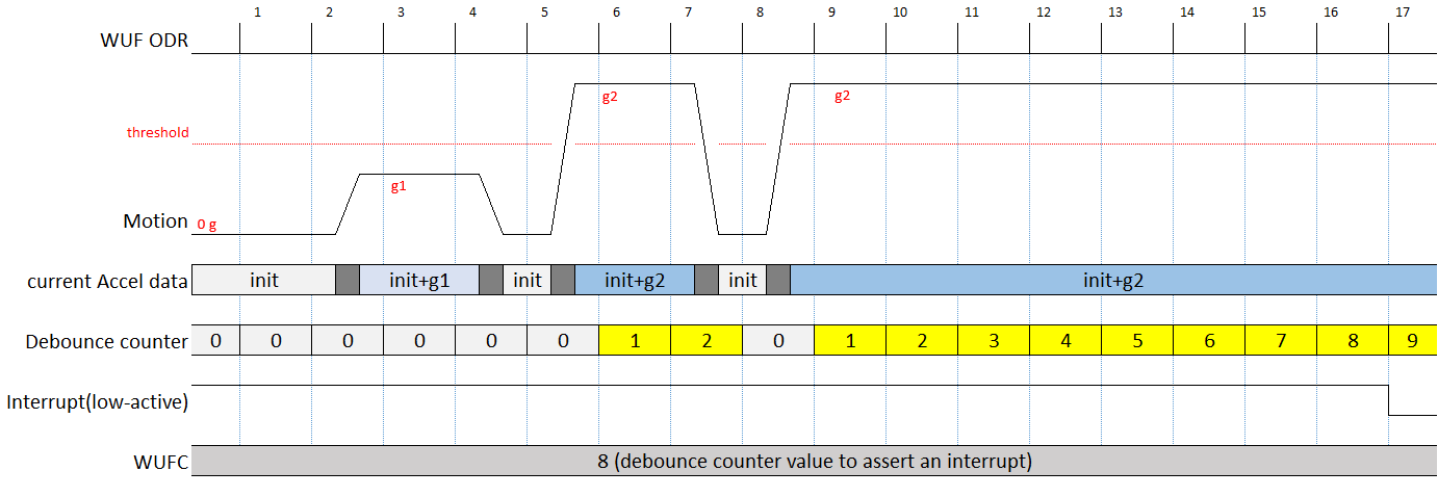

Figure 6: Latched Motion Interrupt Response in the Absolute Threshold Mode

Examples

2.2.1.3 Case 1: XPWUE=1, XNWUE=0, C_MODE=0, PR_MODE=0

Positive direction enabled, negative direction disabled, reset debounce counter, pulse reject disabled

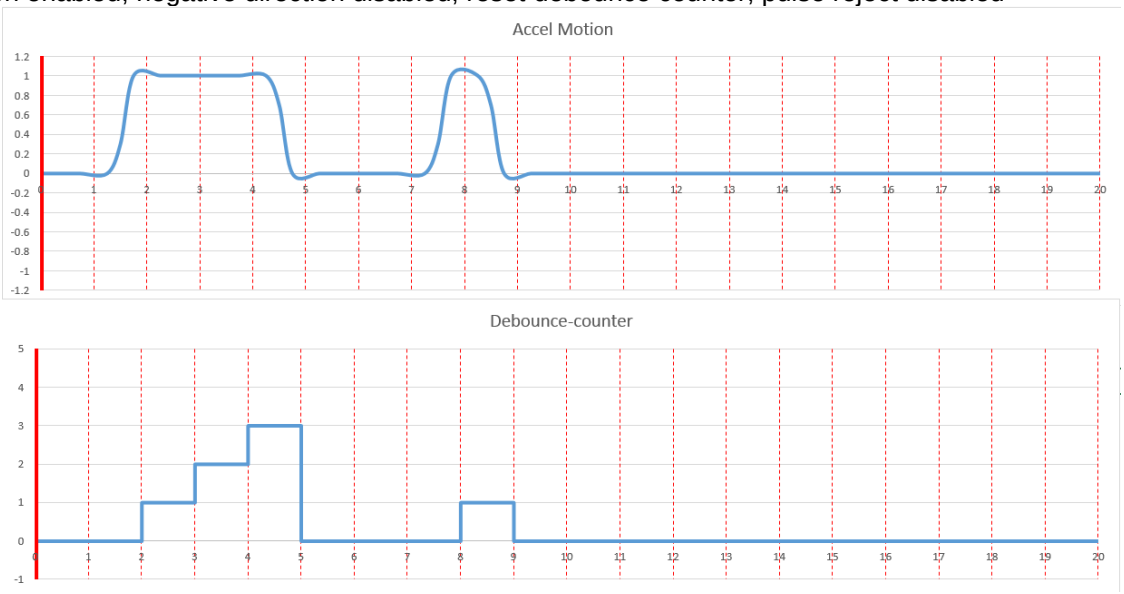

Figure 7: Case 1 Positive Motion

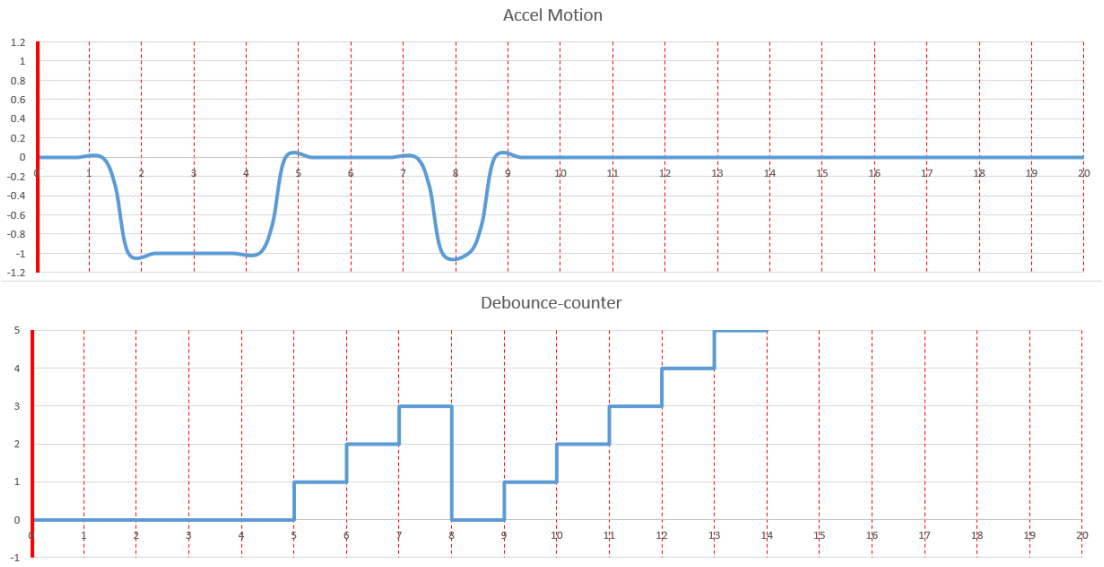

Figure 8: Case 1 Negative Motion

2.2.1.4 Case 2: XPWUE=0, XNWUE=1, C_MODE=0, PR_MODE=0

Positive direction disabled, negative direction enabled, reset debounce counter, pulse reject disabled

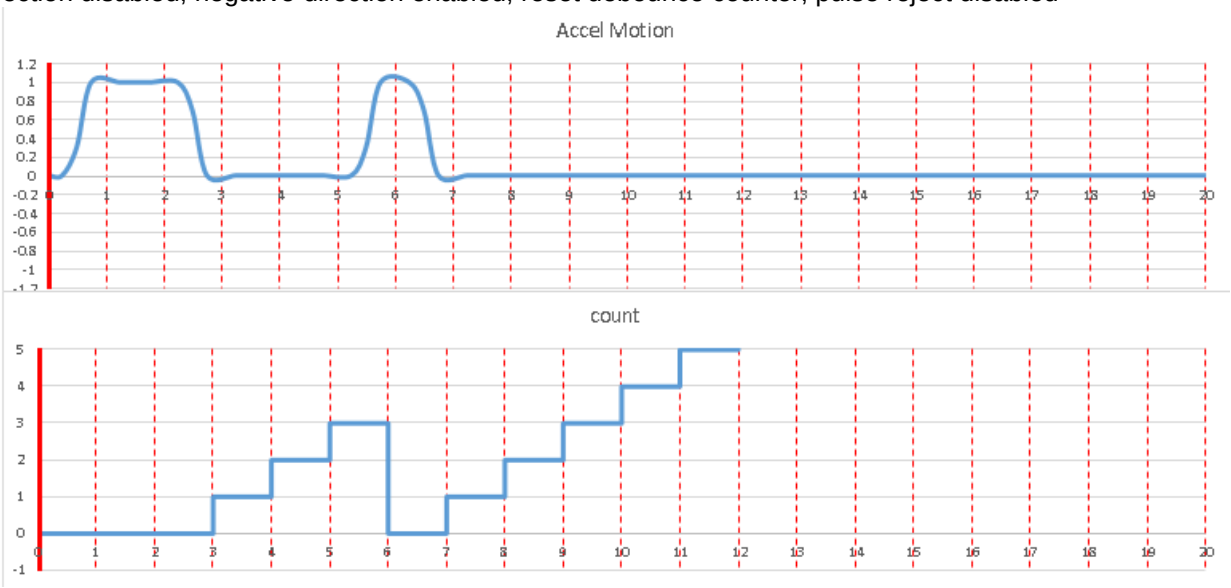

Figure 9: Case 2 Positive Motion

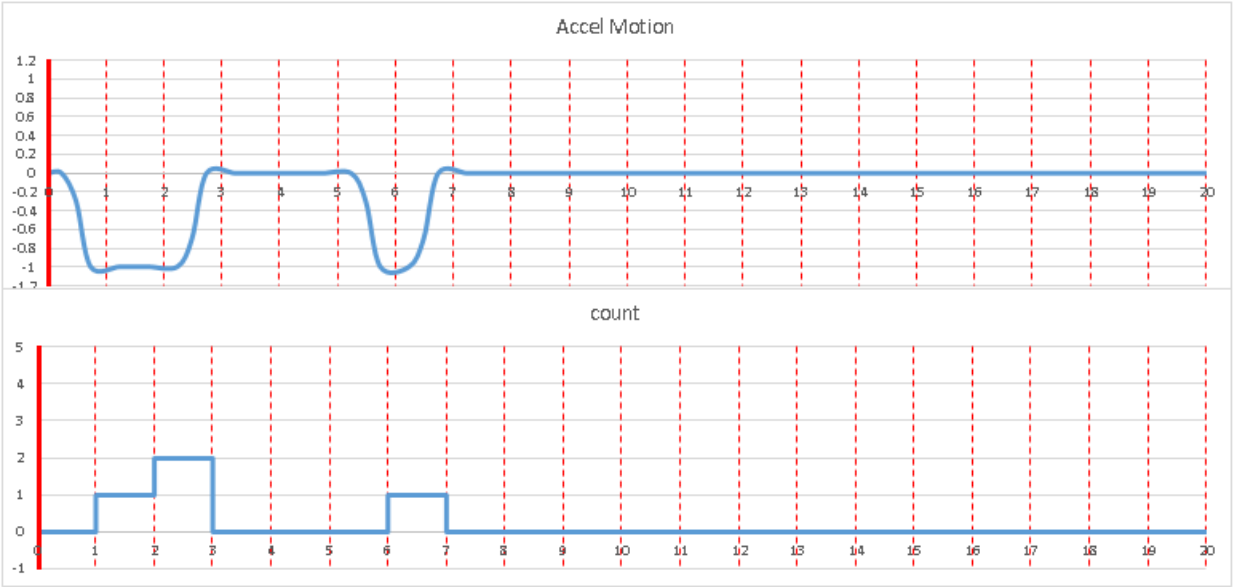

Figure 10: Case 2 Negative Motion

2.2.1.5 Case 3: XPWUE=1, XNWUE=0, C_MODE=1, PR_MODE=0

Positive direction enabled, negative direction disabled, decrement debounce counter, pulse reject disabled

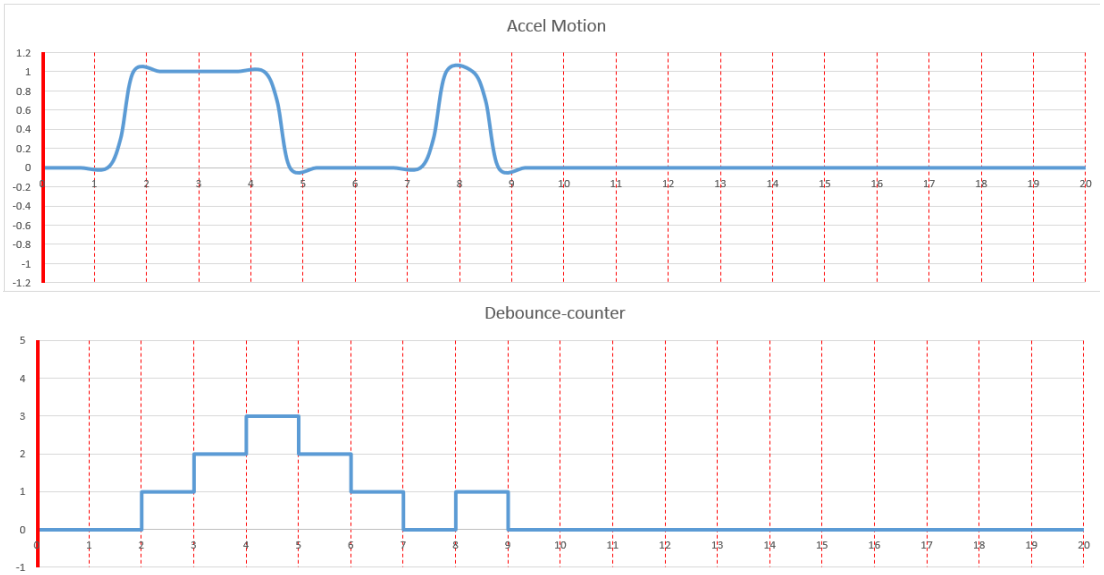

Figure 11: Case 3 Positive Motion

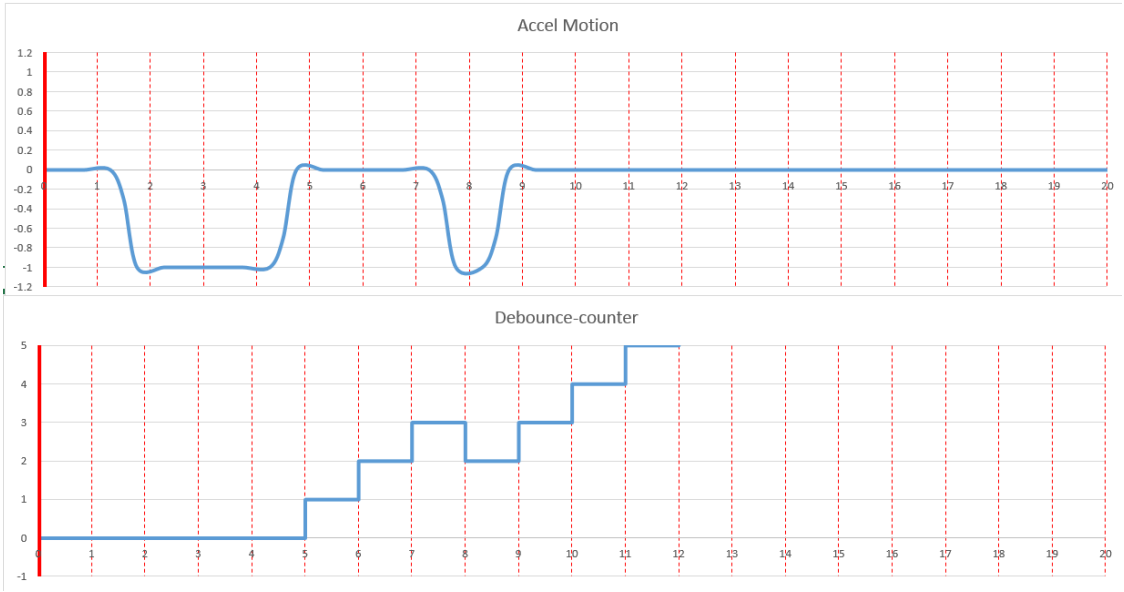

Figure 12: Case 3 Negative Motion

2.2.1.6 Case 4: XPWUE=1, XNWUE=1, C_MODE=0, PR_MODE=0

Positive direction enabled, negative direction enabled, reset debounce counter, pulse reject disabled

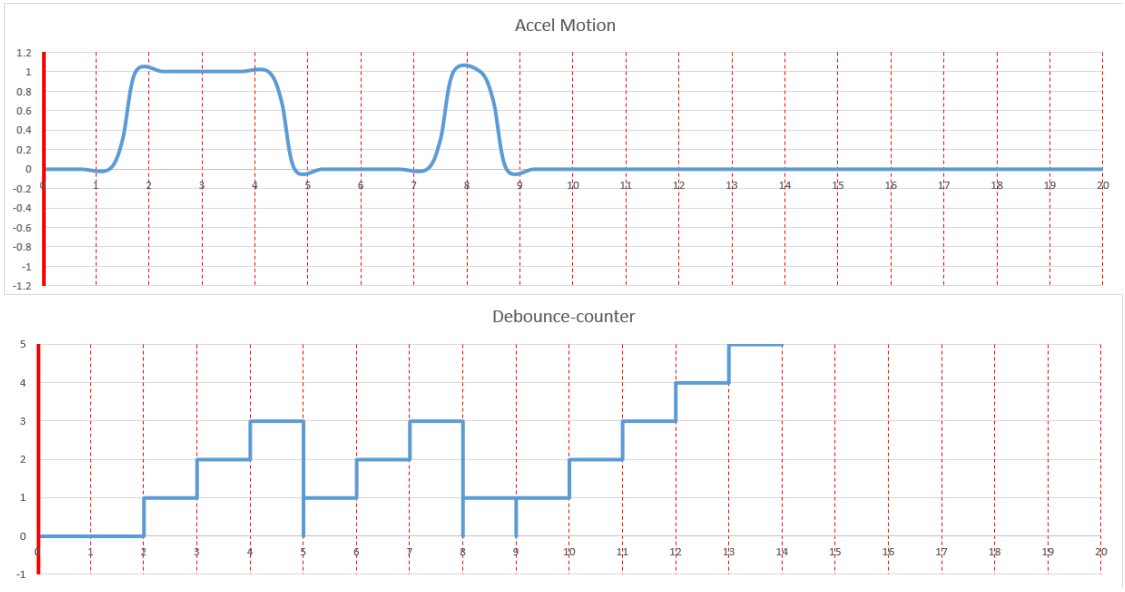

Figure 13: Case 4 Positive Motion

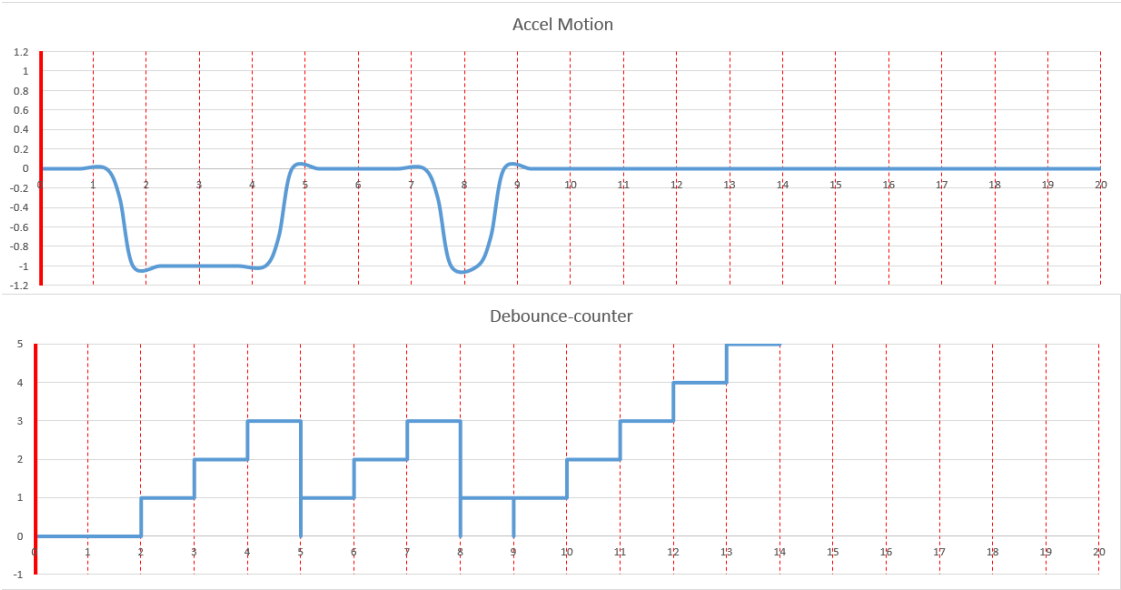

Figure 14: Case 4 Negative Motion

2.2.1.7 Case 5: XPWUE=1, XNWUE=1, C_MODE=0, PR_MODE=1

Positive direction enabled, negative direction enabled, reset debounce counter, pulse reject enabled

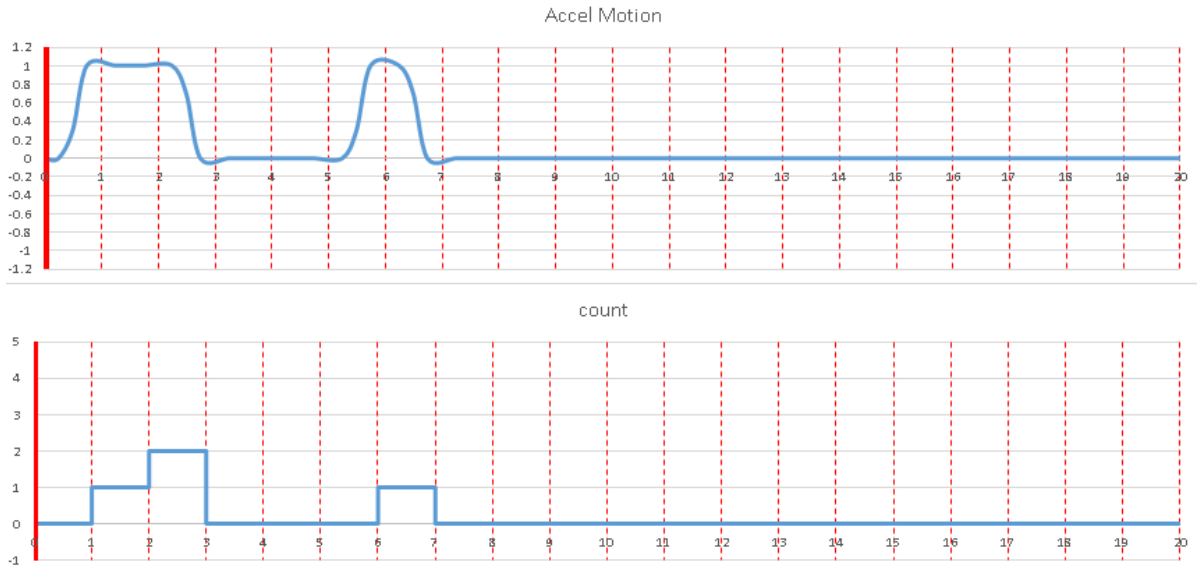

Figure 15: Case 5 Positive Motion

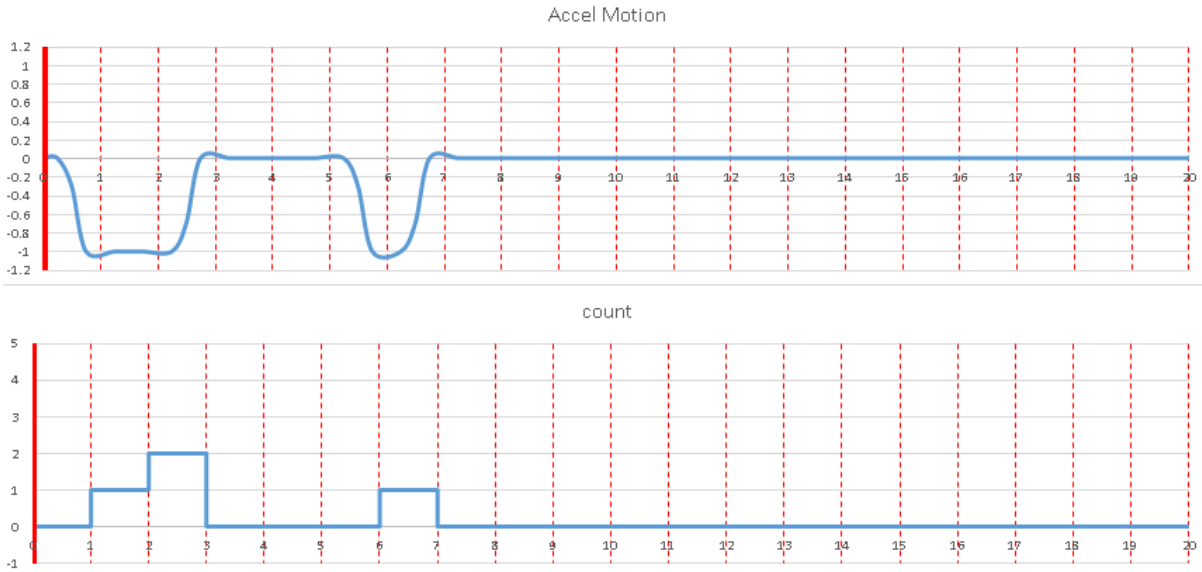

Figure 16: Case 5 Negative Motion

2.3 Directional-Tap Detection Feature Description

The Directional-Tap™ Detection feature of the KX132-1211 recognizes single and double tap inputs and reports the acceleration axis and direction that each tap occurred. Eight performance parameters, as well as a user-selectable ODR are used to configure the KX132-1211 for a desired tap detection response.

Performance Index

The Directional-Tap™ detection algorithm uses low and high thresholds to help determine when a tap event has occurred. A tap event is detected when the previously described jerk summation exceeds the low *threshold (TTL) for more than the tap detection low limit, but less than the tap detection high limit as contained* in FTD. Samples that exceed the high limit (TTH) will be ignored. Figure 17 shows an example of a single tap event meeting the performance index criteria.

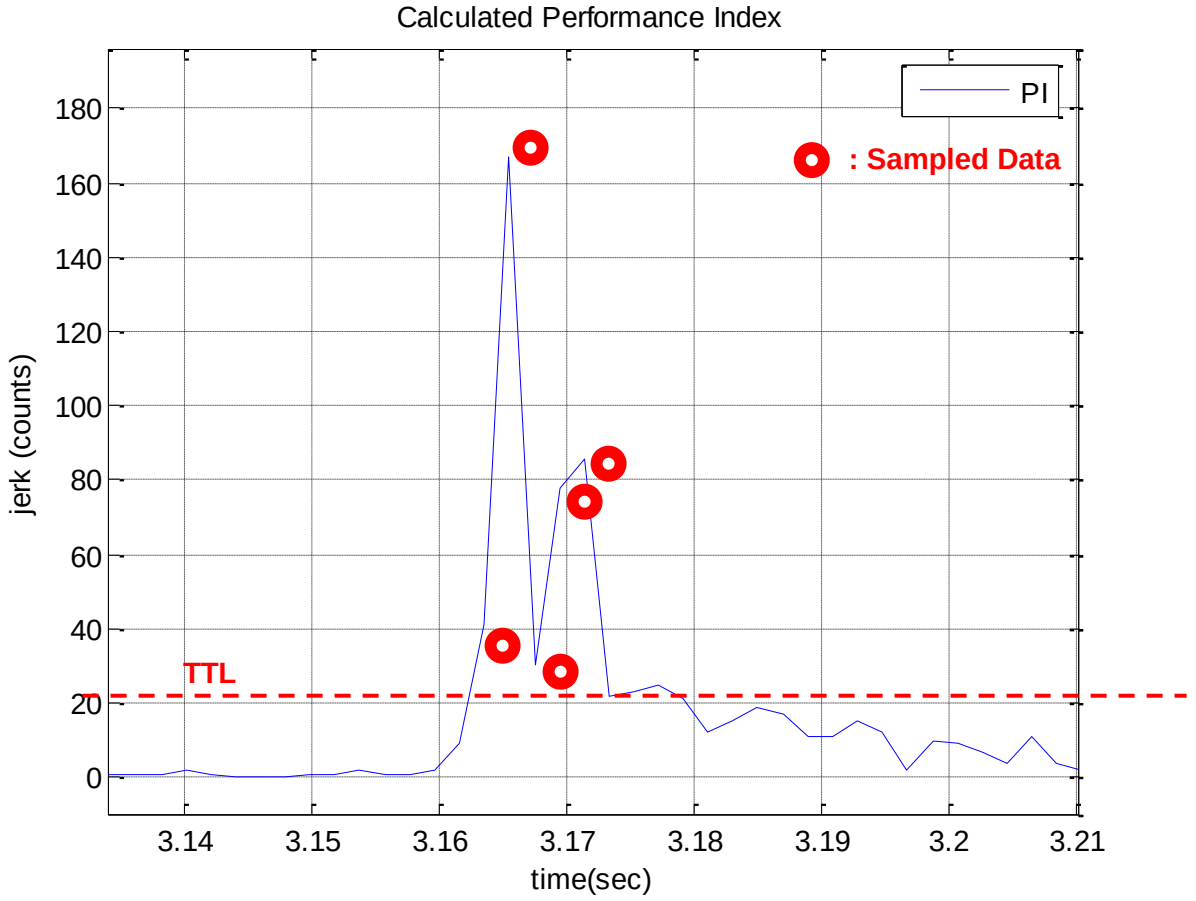

Figure 17: Jerk Summation vs Threshold

Single Tap Detection

The latency timer (TLT) sets the time period that a tap event will only be characterized as a single tap. A second tap must occur outside of the latency timer. If a second tap occurs inside the latency time, it will be ignored as it occurred too quickly. The single tap will be reported at the end of the TWS. Figure 18 shows a single tap event meeting the PI, latency and window requirements.

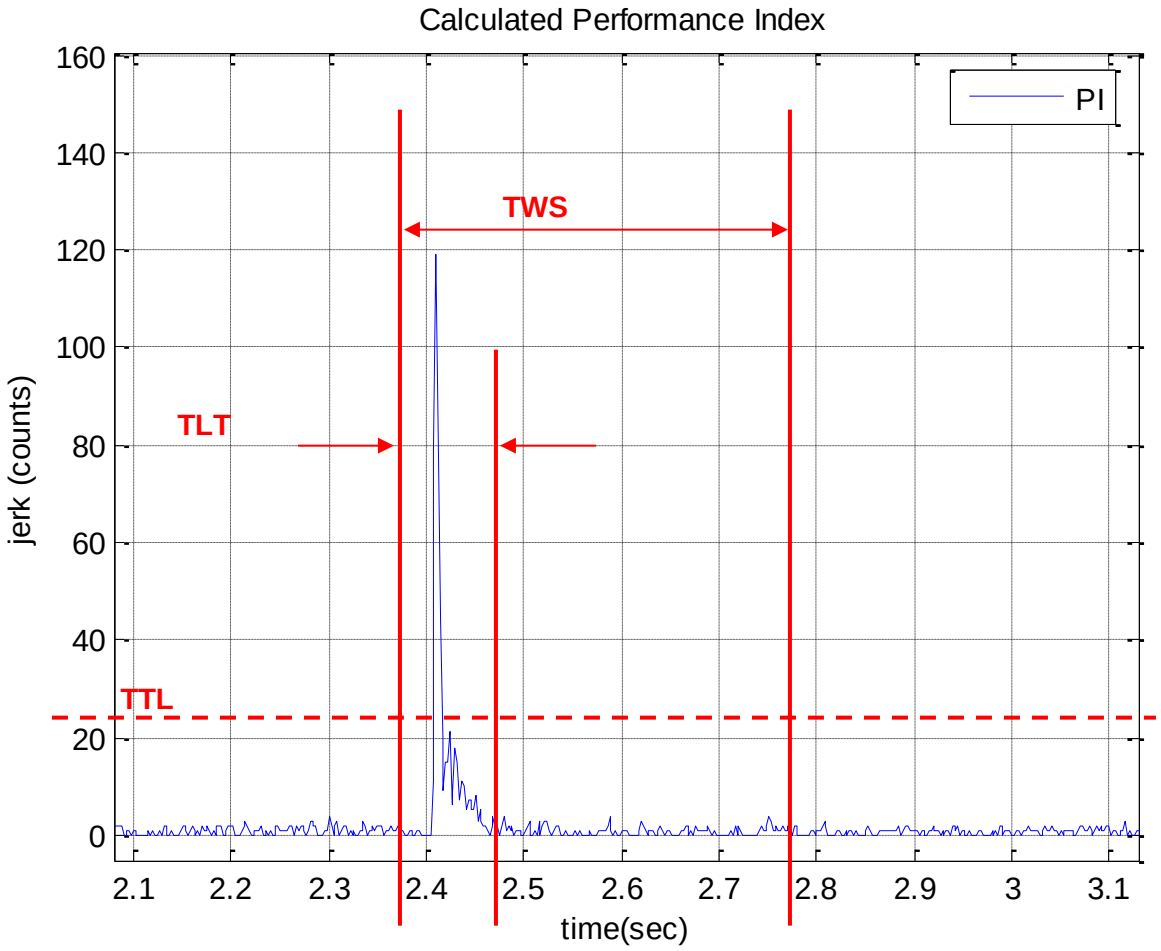

Figure 18: Single Directional-Tap™ Timing

Double-Tap Detection

An event can be characterized as a double tap if the second tap crosses the performance index (TTL) inside the TWS period and ends outside the TDTC. This means that the TDTC determines the minimum time separation that must exist between the two taps of a double tap event. Similar to the single tap, the first tap event must exceed the performance index for the time limit contained in FTD. Also, the duration when the first and second events combined exceed the performance index should not exceed STD. The double tap will be reported at the end of the second TLT. Figure 19 shows a double tap event meeting the PI, latency and window requirements.

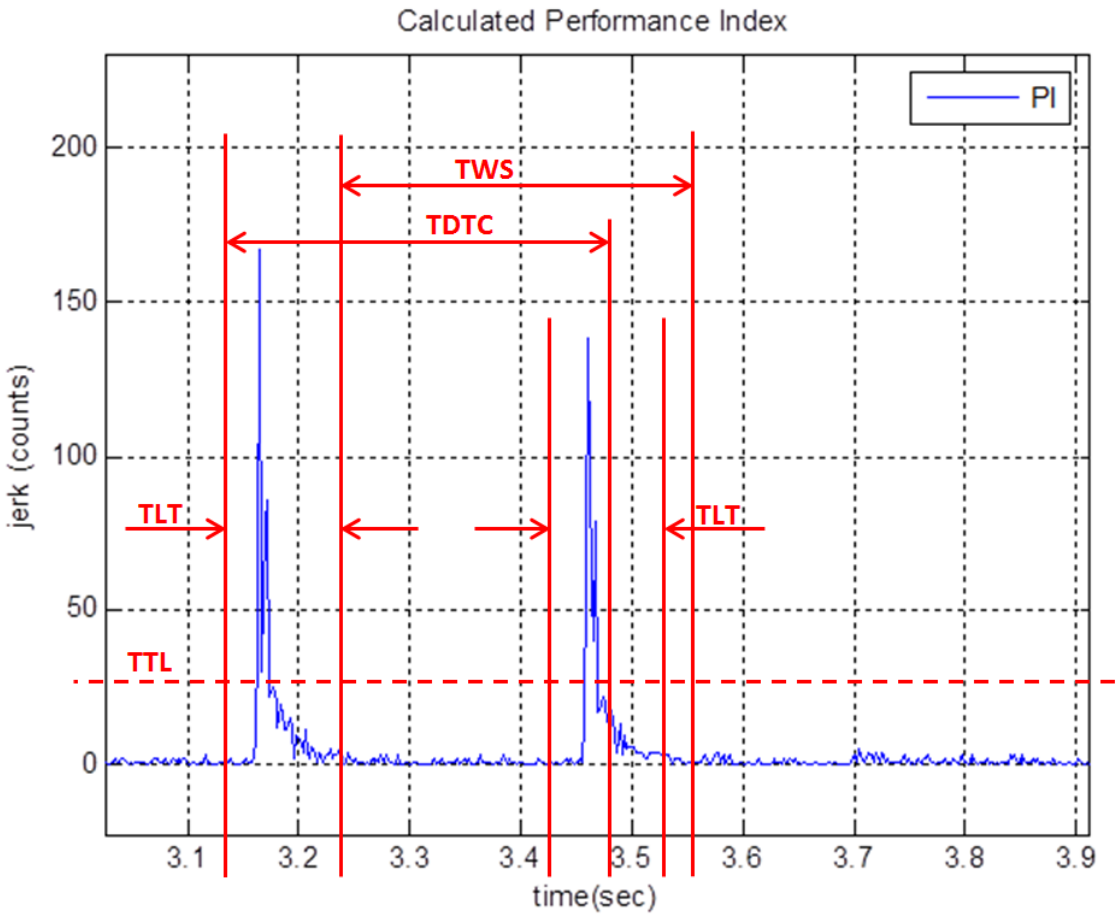

Figure 19: Double Directional-Tap™ Timing

2.4 Free fall Detect

The KX132-1211 features a Free fall interrupt that sends a flag through the INT1 or the INT2 output pins when the accelerometer senses a Free fall event. The interrupt event is also reflected on the INT (bit 4) of the STATUS_REG and FFS (bit 7) of the INS2 registers. A Free fall event is evident when all three accelerometer axes simultaneously fall below a certain acceleration threshold for a set amount of time. The KX132-1211 gives the user the option to define the acceleration threshold value through the FFTH 8-bit register where 256 counts cover the g range of the accelerometer. This value is compared to the top 8 bits of the accelerometer 8g output value (independent of the actual g-range setting of the device). Equation 8 shows how to calculate the FFTH register value for a desired Free fall threshold. The threshold of 0.5g is a good starting point.

FFTH (counts) = Free fall Threshold (g) x 16 (counts/g)

Equation 8: Free fall Threshold

Through the Free Fall Counter (FFC), the user can set the amount of time all three accelerometer axes must simultaneously remain below the FFTH acceleration threshold before the Free fall interrupt flag is sent through the INT1 or the INT2 output pins. This delay/debounce time is defined by the available 0 to 255 counts, which represent accelerometer samples taken at the Free fall ODR defined by OFFI<2:0> bits in the FFCNTL register. Every count is calculated as 1/ODR delay period. Equation 9 shows how to calculate the FFC register value for a desired Free fall delay. The delay of 0.32 sec is a good starting point.

FFC (counts) = Free fall delay (sec) x Free fall ODR (Hz)

Equation 9: Free fall Threshold

When the Free fall interrupt is enabled the part must not be in a physical state that would trigger the Free fall interrupt, or the delay will not be correct for the present Free fall.

Typical Freefall Interrupt Example (nonLatching)

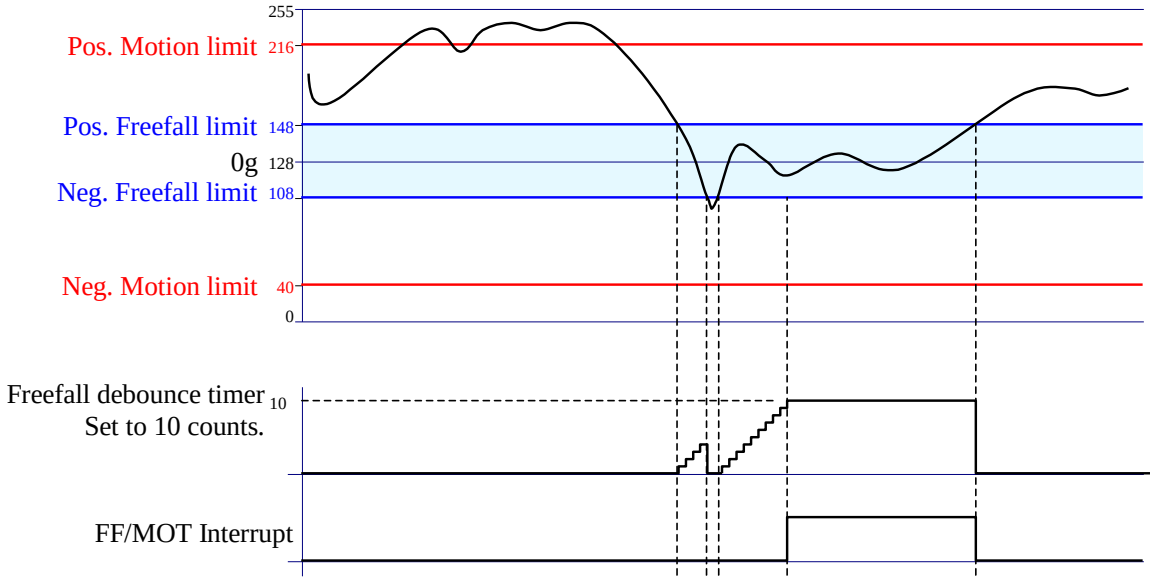

Figure 20: Typical Free fall Interrupt Example (FFCNTL ULMODE = 1)

Typical Freefall Interrupt Example (Latching)

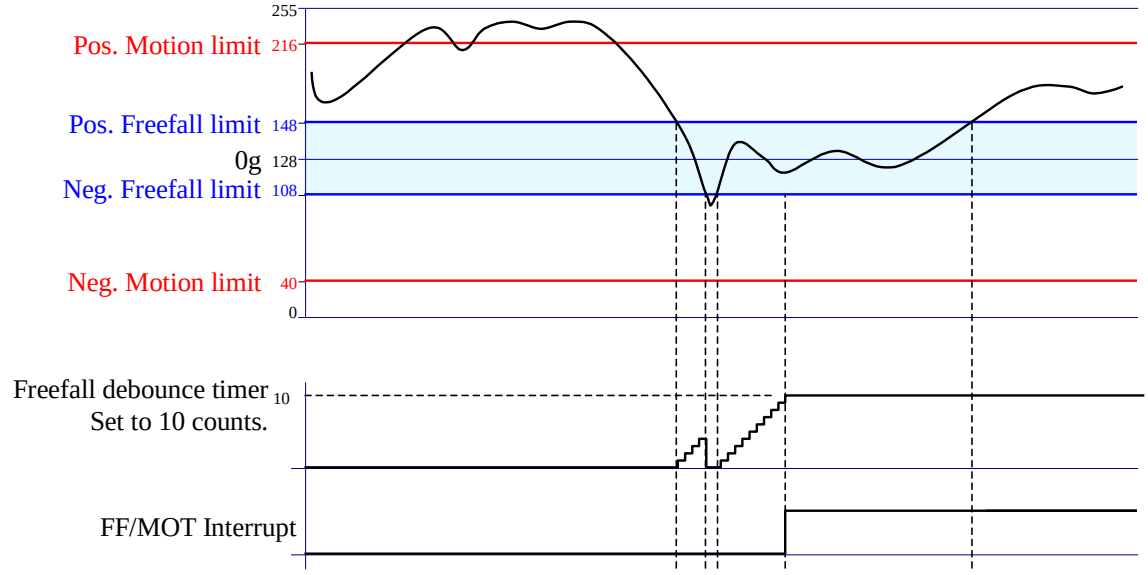

Figure 21: Typical Free fall Interrupt Example (FFCNTL ULMODE = 0)

2.5 Sample Buffer Feature Description

The sample buffer feature of the KX132-1211 accumulates and outputs acceleration data based on how it is configured. There are 3 buffer modes available, and samples can be accumulated at either low (8-bit) or high (16-bit) resolution. Acceleration data is collected at the ODR specified by OSA[3:0] in the ODCNTL register. Each buffer mode accumulates data, reports data, and interacts with status indicators in a slightly different way.

FIFO Mode

Data Accumulation

Sample collection stops when the buffer is full.

Data Reporting

Data is reported with the oldest byte of the oldest sample first (X_L or X based on resolution).

Status Indicators

A watermark interrupt occurs when the number of samples in the buffer reaches the Sample Threshold. The watermark interrupt stays active until the buffer contains less than this number of samples. This can be accomplished through clearing the buffer or explicitly reading greater than SMPX samples (calculated with Equation 10).

BUF_RES=0:
SMPX = SMP_LEV[9:0] /3 – SMP_TH [7:0]

BUF_RES=1:
SMPX = SMP_LEV[9:0] /6 – SMP_TH [7:0]

Equation 10: Samples Above Sample Threshold

Stream Mode

Data Accumulation

Sample collection continues when the buffer is full; older data is discarded to make room for newer data.

Data Reporting

Data is reported with the oldest sample first (uses FIFO read pointer).

Status Indicators

A watermark interrupt occurs when the number of samples in the buffer reaches the Sample Threshold. The watermark interrupt stays active until the buffer contains less than this number of samples. This can be accomplished through clearing the buffer or explicitly reading greater than SMPX samples (calculated with Equation 10).

Trigger Mode

Data Accumulation

When a physical interrupt is caused by one of the digital engines or when a logic high signal occurs on the TRIG pin, the trigger event is asserted and SMP_TH[7:0] samples prior to the event are retained. Sample collection continues until the buffer is full.

Data Reporting

Data is reported with the oldest sample first (uses FIFO read pointer).

Status Indicators

When a physical interrupt occurs and there are at least SMP_TH[7:0] samples in the buffer, BUF_TRIG in BUF_STATUS_2 is asserted.

Buffer Operation

The following diagrams illustrate the operation of the buffer conceptually. Actual physical implementation has been abstracted to offer a simplified explanation of how the different buffer modes operate. Figure 22 represents a high-resolution 3-axis sample within the buffer. Figure 23 – Figure 30 represent a 10-sample version of the buffer (for simplicity), with Sample Threshold set to 8.

Regardless of the selected mode, the buffer fills sequentially, one byte at a time. Figure 22 shows one 6-byte data sample.

	Index	Byte	
	0	X_L	←-- FIFO read pointer
	1	X_H	
	2	Y_L	
	3	Y_H	
	4	Z_L	
	5	Z_H	
buffer write pointer --→	6		

Figure 22: One Buffer Sample

Regardless of the selected mode, the buffer fills sequentially, one sample at a time. The buffer write pointer shows where the next sample will be written to the buffer.

	Index	Sample	
	0	Data0	← FIFO read pointer
	1	Data1	
	2	Data2	
buffer write pointer →	3		
	4		
	5		
	6		
	7		← Sample Threshold
	8		
	9		

Figure 23: Buffer Filling

The buffer continues to fill sequentially until the Sample Threshold is reached.

	Index	Sample	
	0	Data0	← FIFO read pointer
	1	Data1	
	2	Data2	
	3	Data3	
	4	Data4	
	5	Data5	
	6	Data6	
buffer write pointer →	7		← Sample Threshold
	8		
	9		

Figure 24: Buffer Approaching Sample Threshold

In FIFO and Stream modes, a watermark interrupt is issued when the number of samples in the buffer reaches the Sample Threshold. In trigger mode, this is the point where the oldest data in the buffer is discarded to make room for newer data.

	Index	Sample	
	0	Data0	← FIFO read pointer
	1	Data1	
	2	Data2	
	3	Data3	
	4	Data4	
	5	Data5	
	6	Data6	
	7	Data7	← Sample Threshold read pointer
buffer write pointer →	8		
	9		

Figure 25: Buffer at Sample Threshold

In trigger mode, data is accumulated in the buffer sequentially until the Sample Threshold is reached. Once the Sample Threshold is reached, the oldest samples are discarded when new samples are collected. Note in Figure 26 how Data0 was thrown out to make room for Data8.

	Index	Sample	
	0	Data1	← Trigger read pointer
	1	Data2	
	2	Data3	
	3	Data4	
	4	Data5	
	5	Data6	
	6	Data7	
Trigger write pointer →	7	Data8	← Sample Threshold
	8		
	9		

Figure 26: Additional Data Prior to Trigger Event

After a trigger event occurs, the buffer no longer discards the oldest samples, and instead begins accumulating samples sequentially until full. The buffer then stops collecting samples, as seen in Figure 27. This results in the buffer holding SMP_TH [7:0] samples prior to the trigger event, and SMPX samples after the trigger event.

Index	Sample	
0	Data1	← Trigger read pointer
1	Data2	
2	Data3	
3	Data4	
4	Data5	
5	Data6	
6	Data7	
7	Data8	← Sample Threshold
8	Data9	
9	Data10	

Figure 27: Additional Data after Trigger Event

In FIFO, Stream, and Trigger (after a trigger event has occurred) modes, the buffer continues filling sequentially after the Sample Threshold is reached. Sample accumulation after the buffer is full depends on the selected operation mode. FIFO and Trigger modes stop accumulating samples when the buffer is full, and Stream mode begins discarding the oldest data when new samples are accumulated.

Index	Sample	
0	Data0	← FIFO read pointer
1	Data1	
2	Data2	
3	Data3	
4	Data4	
5	Data5	
6	Data6	
7	Data7	← Sample Threshold
8	Data8	
9	Data9	

Figure 28: Buffer Full

After the buffer has been filled in Stream mode, the oldest samples are discarded when new samples are collected. Note in Figure 29 how Data0 was thrown out to make room for Data10.

Index	Sample	
0	Data1	← FIFO read pointer
1	Data2	
2	Data3	
3	Data4	
4	Data5	
5	Data6	
6	Data7	
7	Data8	← Sample Threshold
8	Data9	
9	Data10	

Figure 29: Buffer Full – Additional Sample Accumulation in Stream Mode

In FIFO, Stream, or Trigger mode, reading one sample from the buffer will remove the oldest sample and effectively shift the entire buffer contents up, as seen in Figure 30.

	Index	Sample	
	0	Data1	← FIFO read pointer
	1	Data2	
	2	Data3	
	3	Data4	
	4	Data5	
	5	Data6	
	6	Data7	
	7	Data8	← Sample Threshold
	8	Data9	
buffer write pointer →	9		

Figure 30: FIFO Read from Full Buffer

2.6 Advanced Data Path Feature

Please refer to *AN109 Introduction to Advanced Data Path* and *AN097 Getting Started with Advanced Data Path* application notes for detailed information about the ADP Engine feature.

3 Revision History

Revision	Description	Date
1.0	Production Release.	31-Jul-2019