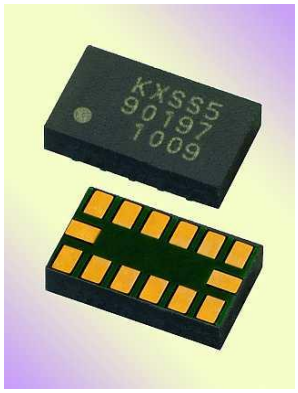
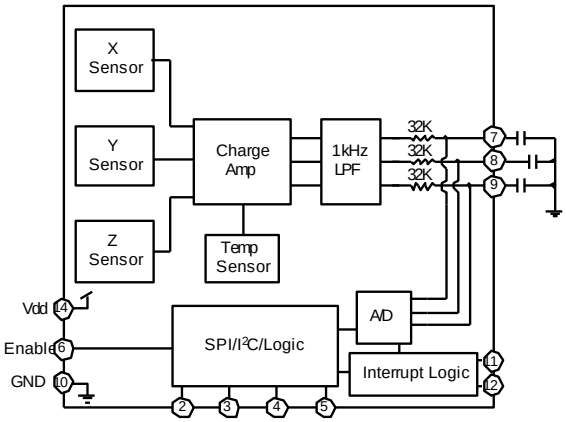
Kionix®	**± 3g Tri-axis Accelerometer Specifications**	**PART NUMBER:** **KXSS5-4457** **Rev. 3** **Feb-2013**

Product Description



The KXSS5-4457 is a Tri-axis, silicon micromachined accelerometer with a full-scale output range of +/-3g (29.4 m/s/s). The sense element is fabricated using Kionix's proprietary plasma micromachining process technology. Acceleration sensing is based on the principle of a differential capacitance arising from acceleration-induced motion of the sense element, which further utilizes common mode cancellation to decrease errors from process variation, temperature, and environmental stress. The sense element is hermetically sealed at the wafer level by bonding a second silicon lid wafer to the device using a glass frit. A separate ASIC device packaged with the sense element provides signal conditioning, self-test, and temperature compensation. The accelerometer is delivered in a 5 x 3 x 0.9 mm LGA plastic package operating from a 1.8 – 3.6V DC supply. The ASIC will trigger interrupt signals if an acceleration threshold is exceeded in any axis (motion interrupt), or if the total acceleration falls below a threshold (freefall interrupt). The thresholds can be set by the customer or default to factory calibrated values. Either I^2C or SPI interfaces can be used to communicate to the chip to trigger A/D conversions, set thresholds or threshold delays, or manage power consumption.

Functional Diagram



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

© 2013 Kionix – All Rights Reserved

Kionix®	**± 3g Tri-axis Accelerometer Specifications**	**PART NUMBER:** **KXSS5-4457** **Rev. 3** **Feb-2013**

Product Specifications

Table 1. Mechanical

(specifications are for operation at 1.8V and T = 25C unless stated otherwise)

Parameters	Units	Min	Typical	Max
Operating Temperature Range	ºC	-25	-	70
Zero-g Offset (analog)	V	0.852	0.9	0.948
Zero-g Offset (digital)	counts	1939	2048	2157
Zero-g Offset Variation from RT over Temp.	mg/ ºC		0.6	
Sensitivity (analog)	mV/g	233	240	247
Sensitivity (digital)	counts/g	530	546	562
Sensitivity Variation from RT over Temp.	%/ ºC		0.01 (xy) 0.03 (z)	
Offset Ratiometric Error (V_{dd} = 1.8V ± 5%)	%		0.3	
Sensitivity Ratiometric Error (V_{dd} = 1.8V ± 5%)	%		0.6 (xy) 0.3 (z)	
Self Test Output change on Activation	g		0.8 (xy) 0.6 (z)	
Non-Linearity	% of FS		0.1	
Cross Axis Sensitivity	%		2	
Noise Density (on filter pins)	µg / √Hz		175	
Freefall threshold[1]	g		0.4	
Freefall delay[1]	ms		4	
Motion threshold[1]	g		2.5	
Motion delay[1]	ms		4	

Notes:

1. Factory default settings. User can adjust thresholds and delays using I^2C or SPI interface.

© 2013 Kionix – All Rights Reserved

Table 2. Electrical

(specifications are for operation at 1.8V and T = 25C unless stated otherwise)

Parameters		Units	Min	Typical	Max
Supply Voltage (V_{dd})	Operating	V	1.7	1.8	3.6
Current Consumption	Operating	µA	400	700	1000
	Standby	µA		0.0012	
Input Low Voltage		V	-	-	0.2 * V_{dd}
Input High Voltage		V	0.8 * V_{dd}	-	-
Input Pull-down Current		µA		0	
Analog Output Resistance(R_{out})		kΩ	24	32	40
Bandwidth (-3dB)[1]		Hz	800	1000	1200
Power Up Time[2]		ms		0.8	
A/D Conversion time		µs		200	
SPI Communication Rate[3]		MHz		1	
I^2C Communication Rate		kHz		400	

Notes:

1. Internal 1 kHz low pass filter. Lower frequencies are user definable with external capacitors.
2. Power up time is determined after the enabling of the part. The typical value reported is when using the internal 1kHz low pass filter only. When a user defined low pass filter is used, the power up time is 5 times the RC time constant of the filter.
3. SPI Communication Rate can be optimized for faster communication per the SPI timing diagram below.

© 2013 Kionix – All Rights Reserved

KXSS5 SPI Timing Diagram

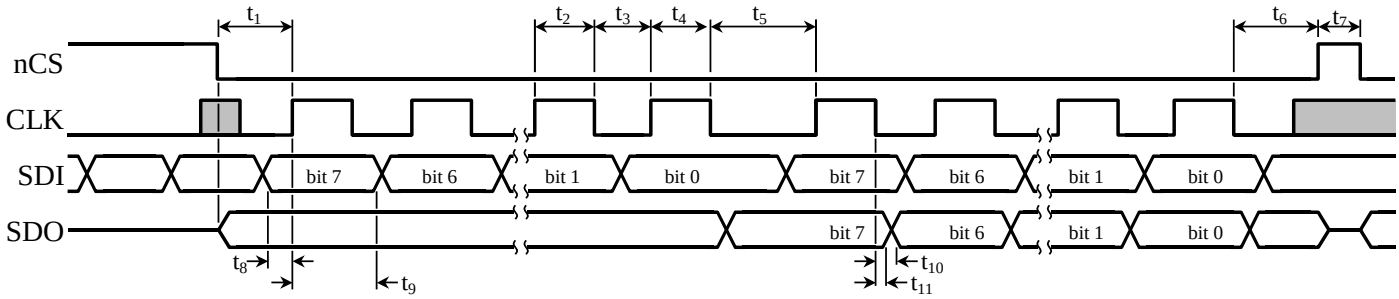


Table 3. SPI Timing

Number	Description	MIN	MAX	Units
-	Enable transition from low to high after Vdd above 1.6V	1		ms
t_1	nCS low to first CLK setup time	130	-	ns
t_2	CLK pulse width: high (Does not apply to the last bit of a byte.)	130	-	ns
t_3	CLK pulse width: low (Does not apply to the last bit of a byte.)	130	-	ns
t_4	CLK pulse width: high (Only on last bit of a byte.)	200	-	ns
t_5	CLK pulse width: low (Only on last bit of a byte.)	350	-	ns
t_6	nCS low after the final CLK falling edge	350	-	ns
t_7	nCS pulse width: high	130	-	ns
t_8	SDI valid to CLK rising edge	10	-	ns
t_9	CLK rising edge to SDI invalid	100	-	ns
t_{10}	CLK falling edge to SDO valid	-	130	ns
t_{11}	CLK falling edge to SDO invalid	0	-	ns
Notes	Recommended SPI CLK	1	-	us
	A/D conversion CLK hold (t_5)	200	-	us

© 2013 Kionix – All Rights Reserved

Table 4. Environmental

Parameters		Units	Min	Typical	Max
Supply Voltage (V_{dd})	Absolute Limits	V	-0.3	-	7.0
Operating Temperature Range		ºC	-40	-	85
Storage Temperature Range		ºC	-55	-	150
Mech. Shock (powered and unpowered)		g	-	-	5000 for 0.5ms
ESD	HBM	V	-	-	2000



Caution: ESD Sensitive and Mechanical Shock Sensitive Component, improper handling can cause permanent damage to the device.



This product conforms to Directive 2002/95/EC of the European Parliament and of the Council of the European Union (RoHS). Specifically, this product does not contain lead, mercury, cadmium, hexavalent chromium, polybrominated biphenyls (PBB), or polybrominated diphenyl ethers (PBDE) above the maximum concentration values (MCV) by weight in any of its homogenous materials. Homogenous materials are "of uniform composition throughout."

HF

This product is halogen-free per IEC 61249-2-21. Specifically, the materials used in this product contain a maximum total halogen content of 1500 ppm with less than 900-ppm bromine and less than 900-ppm chlorine.

Soldering

Soldering recommendations are available upon request or from www.kionix.com.

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

© 2013 Kionix – All Rights Reserved

Application Schematic

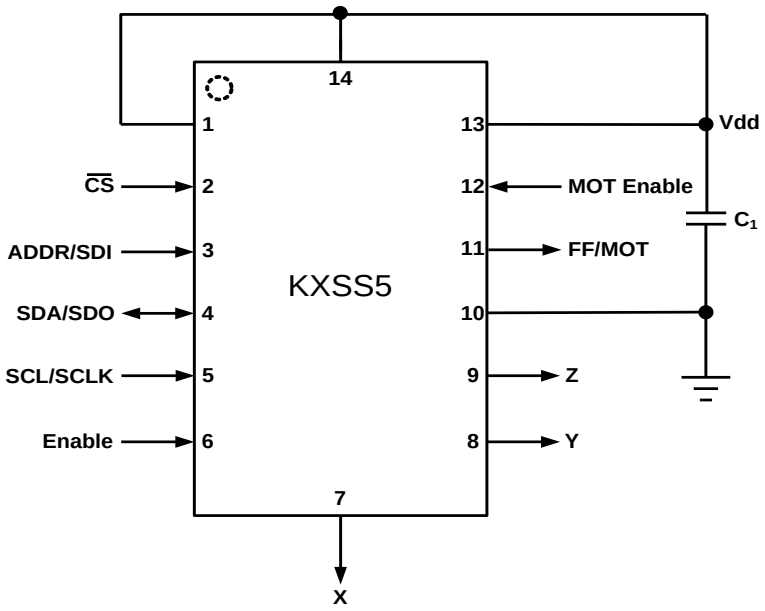


Table 5. KXSS5 Pin Descriptions

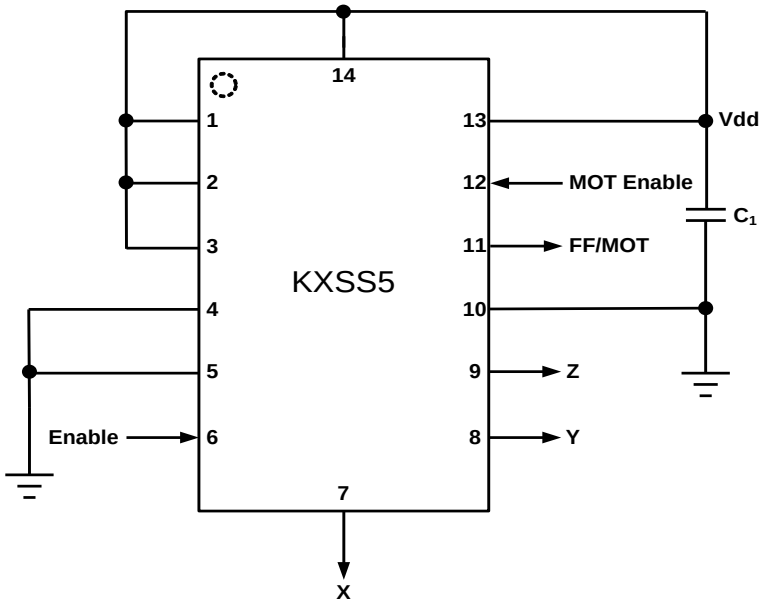
Pin	Name	Description
1	Vdd	The power supply input.
2	nCS	SPI Enable[1] I²C/SPI mode selection (1 = I²C mode, 0 = SPI mode)
3	ADDR/SDI	I²C programmable address bit/SPI Serial Data Input[1]
4	SDA/SD0	I²C Serial Data/SPI Serial Data Output[1]
5	SCL/SCLK	I²C Serial Clock/SPI Serial Clock[1]
6	Enable	**High** - Normal operation **Transition from low to high** – Default values loaded into registers from eeprom, unlatched operation[2] **Low** - Device is in standby, power down mode, I²C/SPI mode will not function
7	X Output	The output of the x-channel. Optionally, a capacitor placed between this pin and ground will form a lowpass filter in addition to the internal 1kHz internal filter.
8	Y Output	The output of y-channel. Optionally, a capacitor placed between this pin and ground will form a lowpass filter in addition to the internal 1kHz internal filter.
9	Z Output	The output of z-channel. Optionally, a capacitor placed between this pin and ground will form a lowpass filter in addition to the internal 1kHz internal filter.
10	GND	Ground
11	FF/MOT (output)	**Low:** no interrupts **High**: (all channels below Freefall threshold) OR (at least one channel above Motion threshold AND (MOT Enable=High))
12	MOT Enable (input)	**Low** – disable Motion interrupt **High** – enable Motion interrupt to "OR" with freefall interrupt onto the FF/MOT pin
13	Vdd	The power supply input.
14	Vdd	The power supply input. Decouple this pin to ground with a 0.1uF ceramic capacitor.

© 2013 Kionix – All Rights Reserved

Application Design Notes

[1] When used without digital communications, make the following connections:

nCS = Vdd (puts the part into I^2C mode, disables pullups on SDA/SDO pad)
SCL/SCLK = GND
SDA/SDO = GND
ADDR/SDI = GND or Vdd



In this mode, the interrupts operate in unlatched mode with the factory default settings for free-fall and motion thresholds and delays.

[2] Enable cannot transition from low to high until a minimum of 1 ms after Vdd reaches 1.6V.

Application Design Equations

The bandwidth is determined by the filter capacitors connected from pins 7, 8 and 9 to ground. The response is single pole. Given a desired bandwidth, f_{BW}, the filter capacitors are determined by:

$$C_2 = C_3 = C_4 = \frac{4.97x10^{-6}}{f_{BW}}$$

© 2013 Kionix – All Rights Reserved

KXSS5 Interrupt Features

As shown in the application schematic, the KXSS5 features a free-fall interrupt (FF) with an optional high-g motion interrupt (MOT) on the same output pin (FF/MOT). Each interrupt features independent, user-definable thresholds, debounce times, and latch/unlatch capabilities that are customized through the KXSS5's embedded 8-bit registers or default to factory calibrated values.

Free-fall Detection Interrupt - The free-fall interrupt goes high when a free-fall event is detected. A free-fall event occurs when the acceleration on all three accelerometer axes simultaneously falls below the low acceleration threshold for a certain amount of time. The low acceleration threshold and debounce time is set by the user (or default to factory calibrated values) during power up through the embedded 8-bit registers. Also, the free-fall interrupt can be user-defined as latched or unlatched.

High-g Motion Interrupt - The optional high-g motion interrupt goes high when a high-g event is detected. A high-g event occurs when the acceleration on any axis exceeds the high acceleration threshold for a certain amount of time. The high acceleration threshold and debounce time is set by the user (or default to factory calibrated values) during power up through the embedded 8-bit registers. The MOT Enable pin enables the Motion interrupt to logically "OR" with the free-fall interrupt onto the FF/MOT pin. Also, the high-g motion interrupt can be user-defined as latched or unlatched.

Test Specifications

! ***Special Characteristics**:*

These characteristics have been identified as being critical to the customer. Every part is tested to verify its conformance to specification prior to shipment.

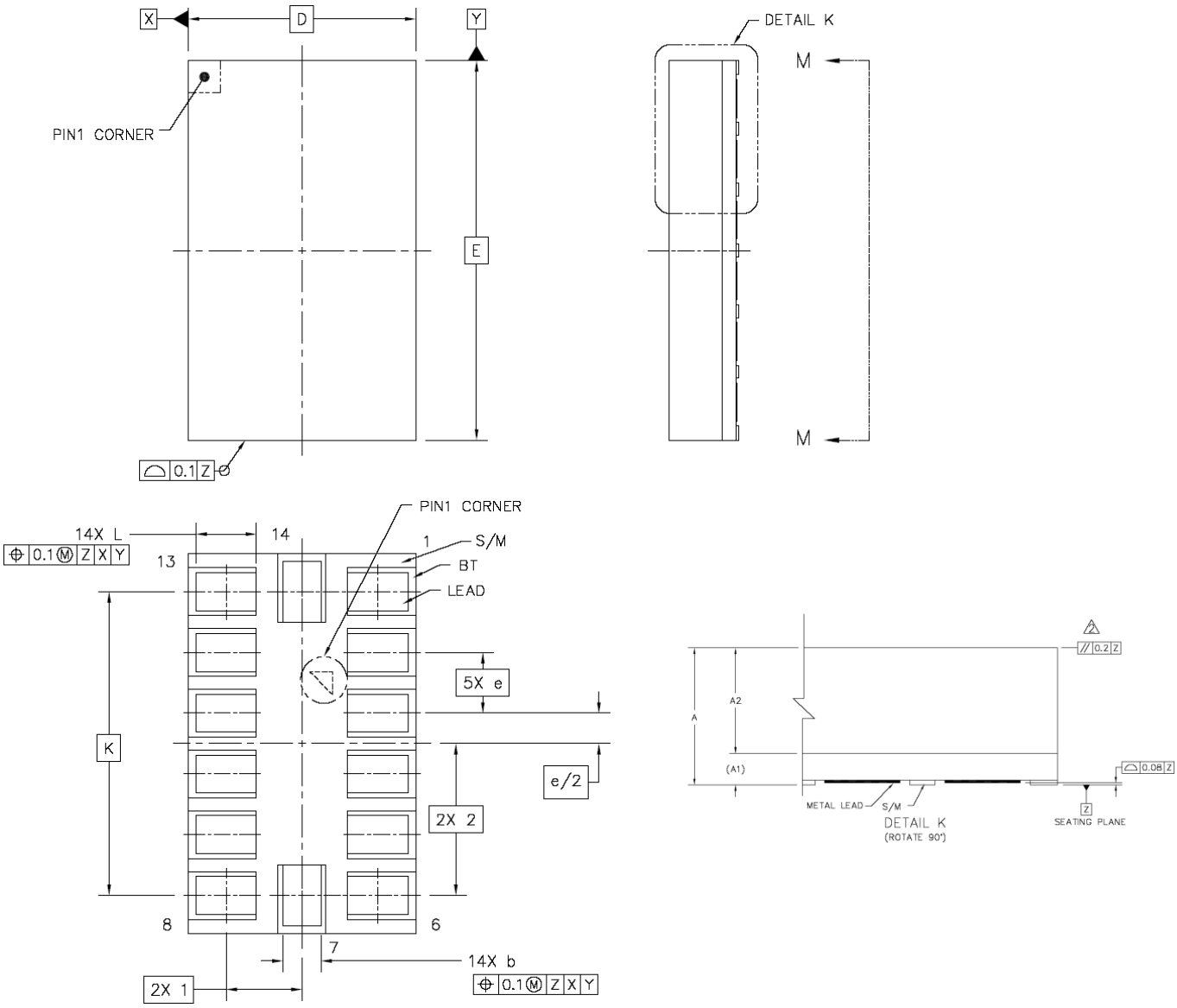
Table 6. Test Specifications

Parameter	Specification	Test Conditions
Zero-g Offset @ RT	0.9 +/- 0.048 V	25C, Vdd = 1.8 V
Sensitivity @ RT	240 +/- 7.2 mV/g	25C, Vdd = 1.8 V
Current Consumption -- Operating	400 <= Idd <= 1000 uA	25C, Vdd = 1.8 V

© 2013 Kionix – All Rights Reserved

Package Dimensions and Orientation

3 x 5 x 0.9 mm LGA

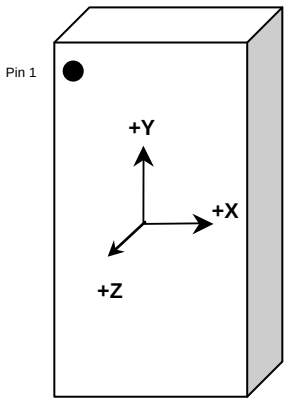


© 2013 Kionix – All Rights Reserved

Dimension	mm			inch		
	Min	Nom	Max	Min	Nom	Max
A	---	0.91	1.0	---	0.036	0.039
A1		0.21 REF			0.008 REF	
A2	0.66	0.7	0.74	0.026	0.028	0.029
b	0.45	0.5	0.55	0.018	0.020	0.022
D		3 BSC			0.118 BSC	
E		5 BSC			0.197 BSC	
K		4 BSC			0.157 BSC	
e		0.8 BSC			0.031 BSC	
L	0.75	0.8	0.85	0.029	0.031	0.033

All dimensions and tolerances conform to ASME Y14.5M-1994

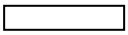
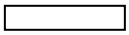
Orientation

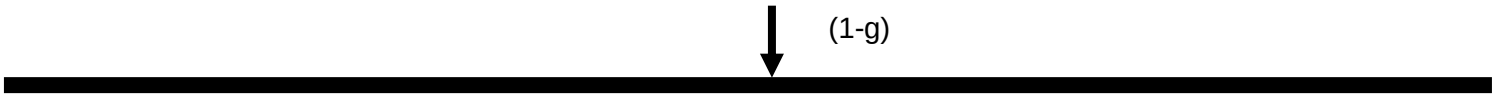


When device is accelerated in +X, +Y or +Z direction, the corresponding output will increase.

© 2013 Kionix – All Rights Reserved

Static X/Y/Z Output Response versus Orientation to Earth's surface (1-g):

Position	1	2	3	4	5	6
Diagram					Top  Bottom	Bottom  Top
X	0.9 V	1.14 V	0.9 V	0.66 V	0.9 V	0.9 V
Y	1.14 V	0.9 V	0.66 V	0.9 V	0.9 V	0.9 V
Z	0.9 V	0.9 V	0.9 V	0.9 V	1.14 V	0.66 V
X-Polarity	**0**	+	**0**	-	**0**	**0**
Y-Polarity	+	**0**	-	**0**	**0**	**0**
Z-Polarity	**0**	**0**	**0**	**0**	+	-



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

© 2013 Kionix – All Rights Reserved

KXSS5 Digital Interfaces

The Kionix KXSS5 digital accelerometer has the ability to communicate on both I^2C and SPI digital serial interface busses. This flexibility allows for easy system integration by eliminating analog-to-digital converter requirements and by providing direct communication with system micro-controllers. In doing so, all of the digital communication pins have shared responsibilities.

The serial interface terms and descriptions as indicated in Table 7 below will be observed throughout this document.

Term	Description
Transmitter	The device that transmits data to the bus.
Receiver	The device that receives data from the bus.
Master	The device that initiates a transfer, generates clock signals and terminates a transfer.
Slave	The device addressed by the Master.

Table 7. Serial Interface Terminologies

I^2C Serial Interface

The KXSS5 has the ability to communicate on an I^2C bus. I^2C is primarily used for synchronous serial communication between a Master device and one or more Slave devices. The Master, typically a micro controller, provides the serial clock signal and addresses Slave devices on the bus. The KXSS5 always operates as a Slave device during standard Master-Slave I^2C operation as shown in Figure 1 on the following page.

I^2C is a two-wire serial interface that contains a Serial Clock (SCL) line and a Serial Data (SDA) line. SCL is a serial clock that is provided by the Master, but can be held low by any Slave device, putting the Master into a wait condition. SDA is a bi-directional line used to transmit and receive data to and from the interface. Data is transmitted MSB (Most Significant Bit) first in 8-bit per byte format, and the number of bytes transmitted per transfer is unlimited. The I^2C bus is considered free when both lines are high.

© 2013 Kionix – All Rights Reserved

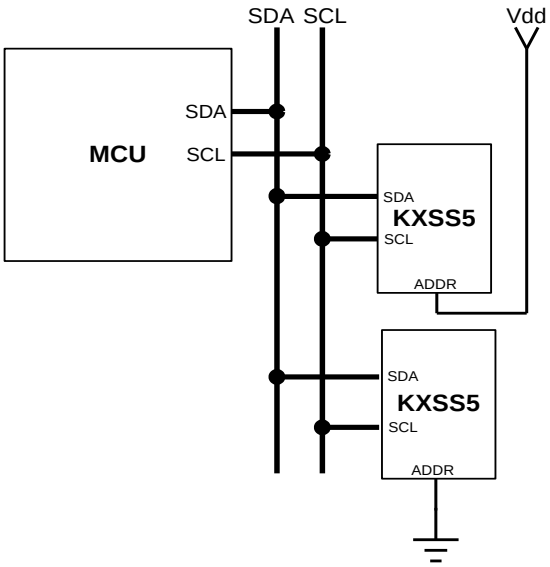


Figure 1 Multiple KXSS5 I²C Connection

I²C Operation

Transactions on the I²C bus begin after the Master transmits a start condition (S), which is defined as a high-to-low transition on the data line while the SCL line is held high. The bus is considered busy after this condition. The next byte of data transmitted after the start condition contains the Slave Address (SAD) in the seven MSBs (Most Significant Bits), and the LSB (Least Significant Bit) tells whether the Master will be receiving data '1' from the Slave or transmitting data '0' to the Slave. When a Slave Address is sent, each device on the bus compares the seven MSBs with its internally-stored address. If they match, the device considers itself addressed by the Master. The KXSS5's Slave Address is comprised of a programmable part and a fixed part, which allows for connection of multiple KXSS5's to the same I²C bus.

The Slave Address associated with the KXSS5 is **001100X**, where the programmable bit, X, is determined by the assignment of ADDR (pin 3) to GND or Vdd. Figure 1 above shows how two KXSS5's would be implemented on an I²C bus.

It is mandatory that receiving devices acknowledge (ACK) each transaction. Therefore, the transmitter must release the SDA line during this ACK pulse. The receiver then pulls the data line low so that it remains stable low during the high period of the ACK clock pulse. A receiver that has been addressed, whether it is Master or Slave, is obliged to generate an ACK after each byte of data has been received. To conclude a transaction, the Master must transmit a stop condition (P) by transitioning the SDA line from low to high while SCL is high. The I²C bus is now free.

© 2013 Kionix – All Rights Reserved

Writing to a KXSS5 8-bit Register

Upon power up, the Master must write to the KXSS5's control registers to set its operational mode. Therefore, when writing to a control register on the I^2C bus, as shown Sequence 1 on the following page, the following protocol must be observed: After a start condition, SAD+W transmission, and the KXSS5 ACK has been returned, an 8-bit Register Address (RA) command is transmitted by the Master. This command is telling the KXSS5 to which 8-bit register the Master will be writing the data. Since this is I^2C mode, the MSB of the RA command should always be zero (0). The KXSS5 acknowledges the RA and the Master transmits the data to be stored in the 8-bit register. The KXSS5 acknowledges that it has received the data and the Master transmits a stop condition (P) to end the data transfer. The data sent to the KXSS5 is now stored in the appropriate register. The KXSS5 automatically increments the received RA commands and, therefore, multiple bytes of data can be written to sequential registers after each Slave ACK as shown in Sequence 2 on the following page.

Reading from a KXSS5 8-bit Register

When reading data from a KXSS5 8-bit register on the I^2C bus, as shown in Sequence 3 on the next page, the following protocol must be observed: The Master first transmits a start condition (S) and the appropriate Slave Address (SAD) with the LSB set at '0' to write. The KXSS5 acknowledges and the Master transmits the 8-bit RA of the register it wants to read. The KXSS5 again acknowledges, and the Master transmits a repeated start condition (Sr). After the repeated start condition, the Master addresses the KXSS5 with a '1' in the LSB (SAD+R) to read from the previously selected register. The Slave then acknowledges and transmits the data from the requested register. The Master does not acknowledge (NACK) it received the transmitted data, but transmits a stop condition to end the data transfer. Note that the KXSS5 automatically increments through its sequential registers, allowing data reads from multiple registers following a single SAD+R command as shown below in Sequence 4 on the following page.

If a receiver cannot transmit or receive another complete byte of data until it has performed some other function, it can hold SCL low to force the transmitter into a wait state. Data transfer only continues when the receiver is ready for another byte and releases SCL. For instance, after the Master has requested to read acceleration data from the KXSS5, the KXSS5 can hold SCL low to force the Master into a wait state while it completes the A/D conversion. After the A/D conversion, the KXSS5 will release SCL and transmit the acceleration data to the Master. Note that the KXSS5 will hold for A/D conversions only if the CLKhld bit is set in CTRL_REGB.

Data Transfer Sequences

The following information clearly illustrates the variety of data transfers that can occur on the I^2C bus and how the Master and Slave interact during these transfers. Table 8 on the following page defines the I^2C terms used during the data transfers.

© 2013 Kionix – All Rights Reserved

Term	Definition
S	Start Condition
Sr	Repeated Start Condition
SAD	Slave Address
W	Write Bit
R	Read Bit
ACK	Acknowledge
NACK	Not Acknowledge
RA	Register Address
Data	Transmitted/Received Data
P	Stop Condition

Table 8. I^2C Terms

Sequence 1. The Master is writing one byte to the Slave.

Master	S	SAD + W		RA		DATA		P
Slave			ACK		ACK		ACK	

Sequence 2. The Master is writing multiple bytes to the Slave.

Master	S	SAD + W		RA		DATA		DATA		P
Slave			ACK		ACK		ACK		ACK	

Sequence 3. The Master is receiving one byte of data from the Slave.

Master	S	SAD + W		RA		Sr	SAD + R			NACK	P
Slave			ACK		ACK			ACK	DATA		

Sequence 4. The Master is receiving multiple bytes of data from the Slave.

Master	S	SAD + W		RA		Sr	SAD + R			ACK		NACK	P
Slave			ACK		ACK			ACK	DATA		DATA		

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

© 2013 Kionix – All Rights Reserved

SPI Interface

The KXSS5 also utilizes an integrated Serial Peripheral Interface (SPI) for digital communication. The SPI interface is primarily used for synchronous serial communication between one Master device and one or more Slave devices. The Master, typically a micro controller, provides the SPI clock signal (SCLK) and determines the state of Chip Select (nCS). The KXSS5 always operates as a Slave device during standard Master-Slave SPI operation.

SPI is a 4-wire synchronous serial interface that uses two control and two data lines. With respect to the Master, the Serial Clock output (SCLK), the Data Output (MOSI) and the Data Input (MISO) are shared among the Slave devices. The Master generates an independent Chip Select (nCS) for each Slave device that goes low at the start of transmission and goes back high at the end. The Slave Data Output (SDO) line, remains in a high-impedance (hi-z) state when the device is not selected, so it does not interfere with any active devices. This allows multiple Slave devices to share a master SPI port as shown in Figure 2 below.

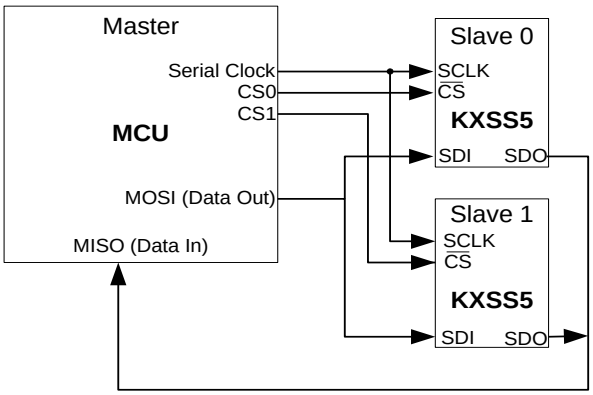


Figure 2 KXSS5 SPI Connections

Read and Write Control Registers

The control registers embedded in the KXSS5 have 8-bit addresses. Upon power up, the Master must write to the accelerometer's control registers to set its operational mode. On the falling edge of nCS, a 2-byte command is written to the appropriate control register. The first byte initiates the write to the appropriate register, and is followed by the user-defined, operational-mode byte. The MSB (Most Significant Bit) of the control register address byte will indicate "0" when writing to the register and "1" when reading from the register. This operation occurs over 16 clock cycles. All commands are sent MSB first, and the host must return nCS high for at least 130 ns before the next data request. Figure 3 below shows the timing diagram for carrying out the 8-bit control register write operation.

© 2013 Kionix – All Rights Reserved

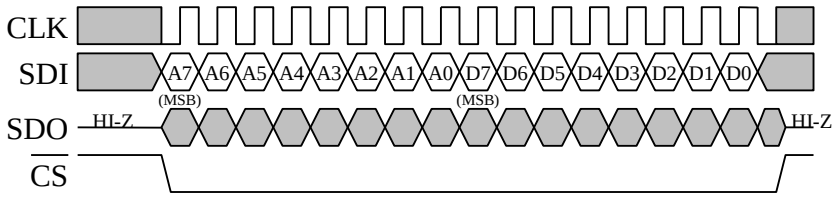


Figure 3 Timing Diagram for 8-Bit Control Register Write Operation

In order to read an 8-bit control register, an 8-bit read command must be written to the accelerometer to initiate the read. The MSB of this control register address byte will indicate "0" when writing to the register and "1" when reading from the register. Upon receiving the command, the accelerometer returns the 8-bit operational-mode data stored in the appropriate control register. This operation also occurs over 16 clock cycles. All returned data is sent MSB first, and the host must return nCS high for at least 130 ns before the next data request. Figure 4 shows the timing diagram for an 8-bit control register read operation.

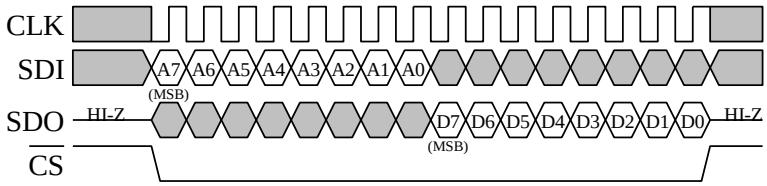


Figure 4 Timing Diagram for 8-Bit Control Register Read Operation

Accelerometer Read Back Operation

The KXSS5 has an onboard 12-bit ADC that can sample, convert and read back sensor data at any time. Transmission of an 8-bit axis-conversion command (see Table 10) begins on the falling edge of nCS. The MSB of this command indicates if you are writing to (0) or reading from (1) the register. After the eight clock cycles used to send the command, the host must hold SCLK low for at least 200µs during the A/D conversion time. Note that all returned data is sent MSB first. Once the data is received, nCS must be returned high for at least 130 ns before the next data request. Figure 5 on the following page shows the timing and diagram for the accelerometer 12-bit ADC read operation.

The Read Back Operation is a 3-byte SPI command. The first byte of SDI contains the command to convert one of the axes. The second and third bytes of SDO contain the 12 bits of the A/D result plus four bits of padding in the LSB to make a total of 16 bits. See Figure 6 below.

© 2013 Kionix – All Rights Reserved

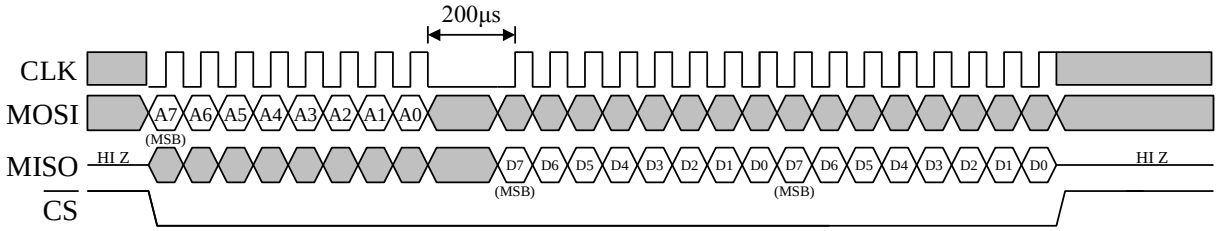


Figure 5 Timing Diagram for an A/D conversion and 12-Bit data read operation.

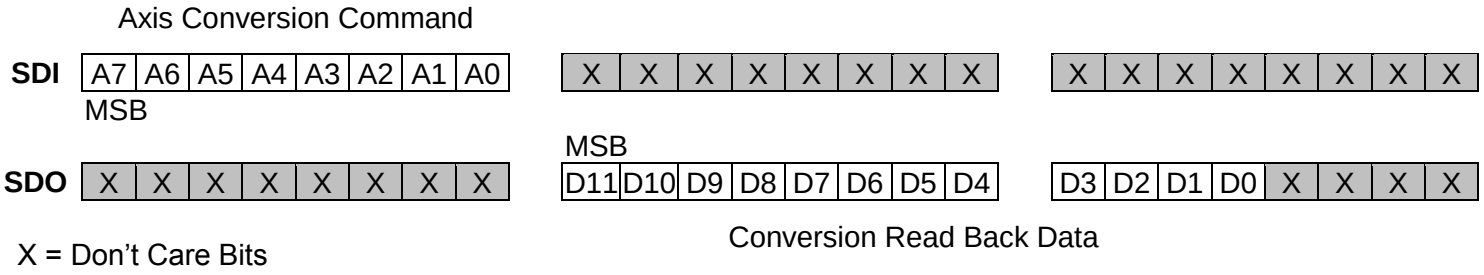


Figure 6 Register Diagram for 12-Bit ADC Read Operation

Digital Accelerometer SPI Sequence

An example of a SPI sequence for reading sensor data is as follows:

- Power up digital accelerometer
- nCS low to select
- Write operational mode commands to the 8-bit control registers
  CTRL_REGB and CTRL_REGC
- nCS high for at least 130 ns
- nCS low to select
- Send convert axis command
  There should be a minimum of 200µs between the first and second bytes in order to give the A/D conversion adequate time to complete.
- The 12-bit A/D data is read to the second and third SDO bytes.
  The KXSS5 auto-increments register transmits on SDO. Therefore, Y-axis, Z-axis, CTRL_REGA, CTRL_REGB, and CTRL_REGC will follow the two X-axis bytes automatically.
- After receiving the last byte of required data, return nCS high for at least 130 ns to reset the auto-increment.
- Repeat data read cycle

© 2013 Kionix – All Rights Reserved

- Recommend reading X-axis, Y-axis, Z-axis, and the three Control Registers for each read cycle to verify the mode selections and status

KXSS5 Embedded Registers

The KXSS5 has 14 embedded 8-bit registers that are accessible by the user. This section contains the addresses for all embedded registers and also describes bit functions of each register. Table 8 and Table 9 below provide a listing of the accessible 8-bit registers and their addresses when in I^2C mode and SPI Mode.

Register Name	Type Read/Write	Address Hex	Address Binary
XOUT_H	R	0x00	0000 0000
XOUT_L	R	0x01	0000 0001
YOUT_H	R	0x02	0000 0010
YOUT_L	R	0x03	0000 0011
ZOUT_H	R	0x04	0000 0100
ZOUT_L	R	0x05	0000 0101
Reset_write	W	0x06	0000 0110
FF_INT	R/W	0x08	0000 1000
FF_DELAY	R/W	0x09	0000 1001
MOT_INT	R/W	0x0A	0000 1010
MOT_DELAY	R/W	0x0B	0000 1011
CTRL_REGC	R/W	0x0C	0000 1100
CTRL_REGB	R/W	0x0D	0000 1101
CTRL_REGA	R	0x0E	0000 1110

Table 9. I^2C Mode Register Map

© 2013 Kionix – All Rights Reserved

Register Name	Type Read/Write	Read Address Hex	Read Address Binary	Write Address Hex	Write Address Binary
XOUT_H	R	0x80	1000 0000	xxxx	xxxx xxxx
XOUT_L	R	0x81	1000 0001	xxxx	xxxx xxxx
YOUT_H	R	0x82	1000 0010	xxxx	xxxx xxxx
YOUT_L	R	0x83	1000 0011	xxxx	xxxx xxxx
ZOUT_H	R	0x84	1000 0100	xxxx	xxxx xxxx
ZOUT_L	R	0x85	1000 0101	xxxx	xxxx xxxx
Reset_write	W	xxxx	xxxx xxxx	0x06	0000 0110
FF_INT	R/W	0x88	1000 1000	0x08	0000 1000
FF_DELAY	R/W	0x89	1000 1001	0x09	0000 1001
MOT_INT	R/W	0x8A	1000 1010	0x0A	0000 1010
MOT_DELAY	R/W	0x8B	1000 1011	0x0B	0000 1011
CTRL_REGC	R/W	0x8C	1000 1100	0x0C	0000 1100
CTRL_REGB	R/W	0x8D	1000 1101	0x0D	0000 1101
CTRL_REGA	R	0x8E	1000 1110	xxxx	xxxx xxxx

Table 10. SPI Mode Register Map

Register Descriptions

XOUT_H

X-axis accelerometer output most significant byte

R	R	R	R	R	R	R	R
XOUTD11	XOUTD10	XOUTD9	XOUTD8	XOUTD7	XOUTD6	XOUTD5	XOUTD4
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0

I^2C Address: 0x00h
SPI Read Address: 0x80h

XOUT_L

X-axis accelerometer output least significant byte

R	R	R	R	R	R	R	R
XOUTD3	XOUTD2	XOUTD1	XOUTD0	X	X	X	X
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0

I^2C Address: 0x01h
SPI Read Address: 0x81h

© 2013 Kionix – All Rights Reserved

YOUT_H

Y-axis accelerometer output most significant byte

R	R	R	R	R	R	R	R
YOUTD11	YOUTD10	YOUTD9	YOUTD8	YOUTD7	YOUTD6	YOUTD5	YOUTD4
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0

I^2C Address: 0x02h

SPI Read Address: 0x82h

YOUT_L

Y-axis accelerometer output least significant byte

R	R	R	R	R	R	R	R
YOUTD3	YOUTD2	YOUTD1	YOUTD0	X	X	X	X
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0

I^2C Address: 0x03h

SPI Read Address: 0x83h

ZOUT_H

Z-axis accelerometer output most significant byte

R	R	R	R	R	R	R	R
ZOUTD11	ZOUTD10	ZOUTD9	ZOUTD8	ZOUTD7	ZOUTD6	ZOUTD5	ZOUTD4
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0

I^2C Address: 0x04h

SPI Read Address: 0x84h

ZOUT_L

Z-axis accelerometer output least significant byte

R	R	R	R	R	R	R	R
ZOUTD3	ZOUTD2	ZOUTD1	ZOUTD0	X	X	X	X
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0

I2C Address: 0x05h

SPI Read Address: 0x85h

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

© 2013 Kionix – All Rights Reserved

Reset_write

When the key (11001010) is written to this register the offset, sensitivity and temperature correction values will be loaded into RAM and used for all further measurements. This can also be accomplished by transitioning the Enable pin (6) from low to high.

W	W	W	W	W	W	W	W
1	1	0	0	1	0	1	0
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0

I^2C Address: 0x06h
SPI Write Address: 0x06h

CTRL_REGA

Read-only status register

R	R	R	R	R	R	R	R
X	X	X	X	X	X	MOTI	FFI
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0

I^2C Address: 0x0Eh
SPI Read Address: 0x8Eh SPI Write Address: 0x0Eh

FFI *reflects the status of the free-fall interrupt. When FFI = 1, the free-fall interrupt pin is high. When FFI = 0, the free-fall interrupt pin is low. The free-fall interrupt is reset by setting FFI = 0.*

MOTI *reflects the status of the motion interrupt. When MOTI = 1, the motion- interrupt pin is high. When MOTI = 0, the motion-interrupt pin is low. The motion interrupt is reset by setting MOTI = 0.*

CTRL_REGB

Read/write control register: Hardwired power up/reset default value (0x42h)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
CLKhld	ENABLE	ST	0	0	X	FFIen	X	01000010
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

I^2C Address: 0x0Dh
SPI Read Address: 0x8Dh SPI Write Address: 0x0Dh

FFIen *enables the freefall interrupt.*

FFIen = 1 - an interrupt will be generated when the KXSS5 is in a predetermined free-fall state
FFIen = 0 – a free-fall interrupt is never generated

© 2013 Kionix – All Rights Reserved

ST *activates the self-test function for the sensor elements on all three axes. A correctly functioning KXSS5 will increase all channel outputs when Self test = 1 and Enable = 1. This bit can be read or written.*

Enable *powers up the KXSS5 for operation.*
Enable = 1 – normal operation
Enable = 0 – low-power standby

CLKhld *allows the KXSS5 to hold the serial clock, SCL, low in I^2C mode to force the transmitter into a wait state during A/D conversions.*
CLKhld = 1 – SCL held low during A/D conversions
CLKhld = 0 – SCL unaffected
CLKhld should be set to 0 when Enable is set to 0 (disabled) to prevent potential holding of the CLK line.

CTRL_REGC

Read/write control register: Hardwired power up/reset default value (0x00h)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
X	X	X	FFLat	MOTLat	0	IntSpd1	IntSpd0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

I^2C Address: 0x0Ch

SPI Read Address: 0x8Ch SPI Write Address: 0x0Ch

IntSpd0 *is the first of two bits used to select the rate at which the accelerometer is sampled when debouncing a potential interrupt event. See Table 11 below.*

IntSpd1 *is the second of two bits used to select the rate at which the accelerometer is sampled when debouncing a potential interrupt event. See Table 11 below.*

IntSpd1	IntSpd0	Interrupt Frequency
0	0	250 Hz
0	1	1 kHz
1	0	4 kHz
1	1	16 kHZ

Table 11. Interrupt Frequencies

© 2013 Kionix – All Rights Reserved

MOTLat *switches the motion interrupt function between latching and non-latching as shown in Figures 7 and 8.*

MOTLat = 0 - The motion interrupt output will go high whenever the criterion for motion detection is met. The output will return low when the criterion is not met.

MOTLat = 1 - The motion interrupt output will go high whenever the criterion for motion detection is met. The interrupt output will remain high until the user toggles the MOT Enable pin (12) low.

Typical Motion Interrupt Example (nonLatching)

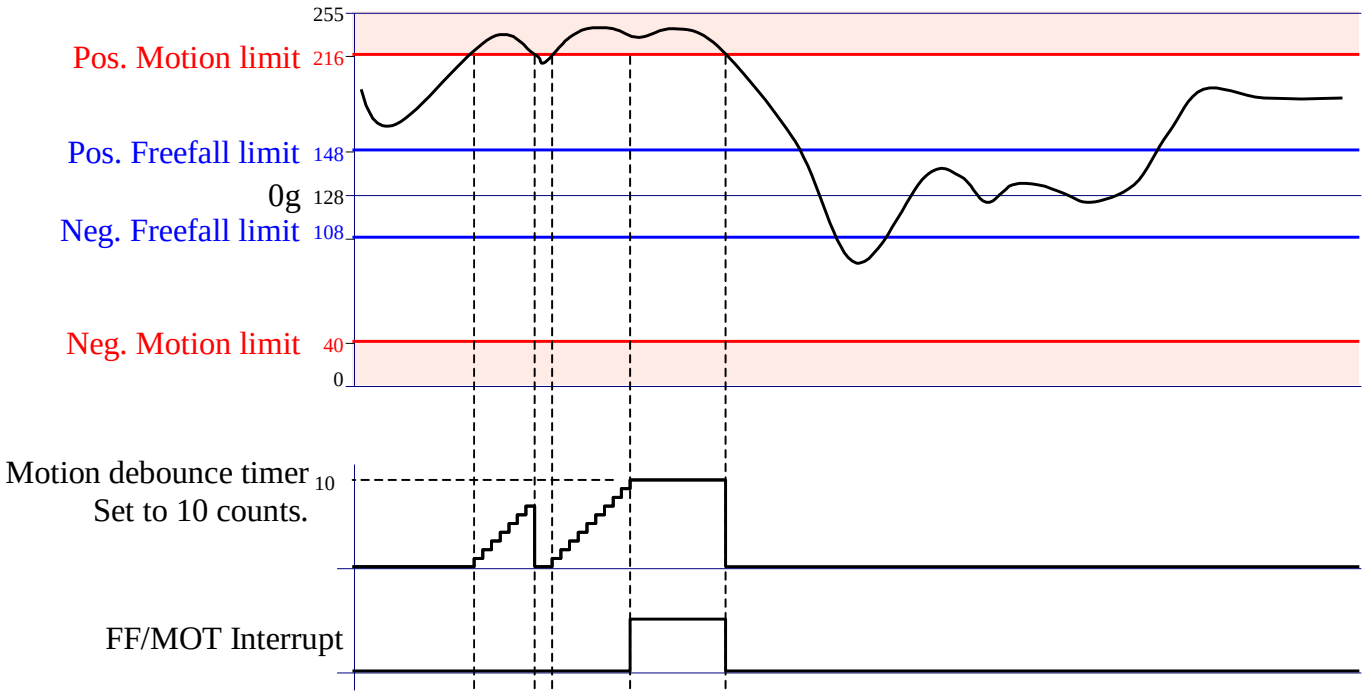


Figure 7. Typical Motion Interrupt Example (MOTLat = 0, MOTen = 1)

© 2013 Kionix – All Rights Reserved

Typical Motion Interrupt Example (Latching)

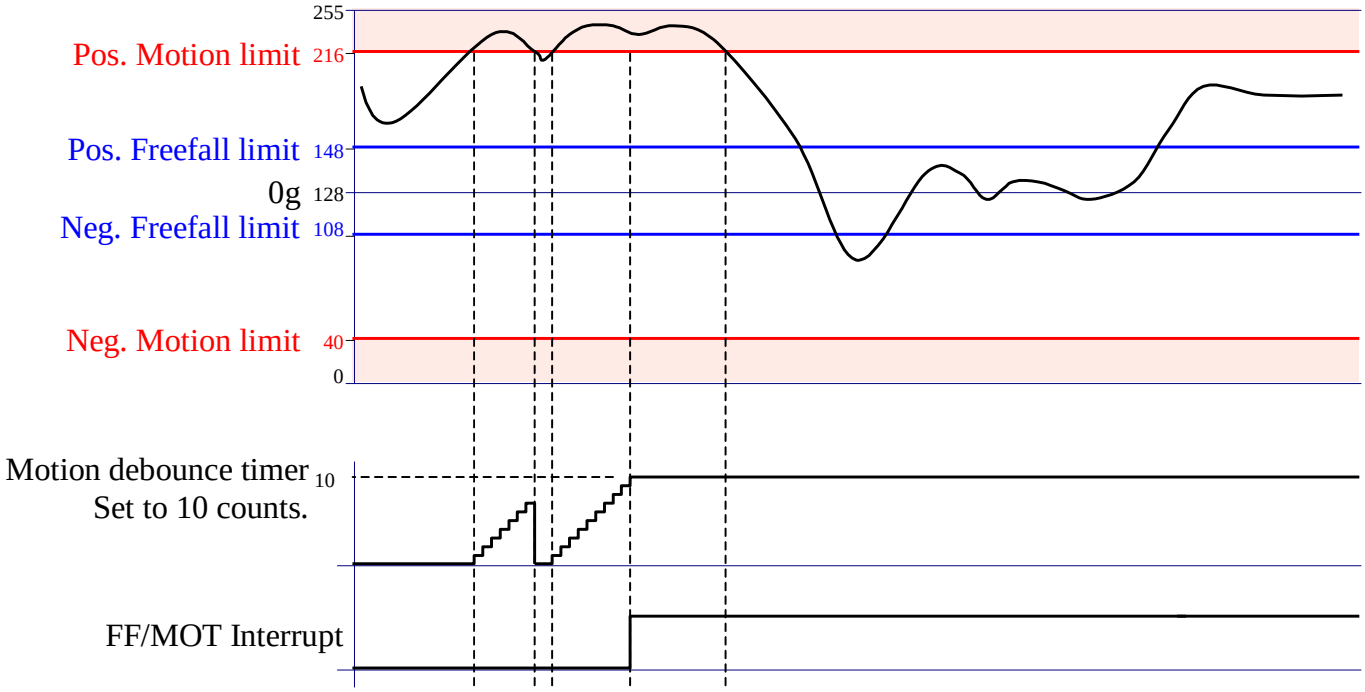


Figure 8. Typical Motion Interrupt Example (MOTLat = 1, MOTen = 1)

© 2013 Kionix – All Rights Reserved

FFLat *switches the free-fall interrupt function between latching and non-latching as shown in Figures 9 and 10.*

FFLat = 0 - The free-fall interrupt output will go high whenever the criterion for free-fall detection is met. The output will return low when the criterion is not met.

FFLat = 1 - The free-fall interrupt output will go high whenever the criterion for free-fall detection is met. The output will remain high until FFIen bit in CTRL_REGB is cycled low.

Typical Freefall Interrupt Example (nonLatching)

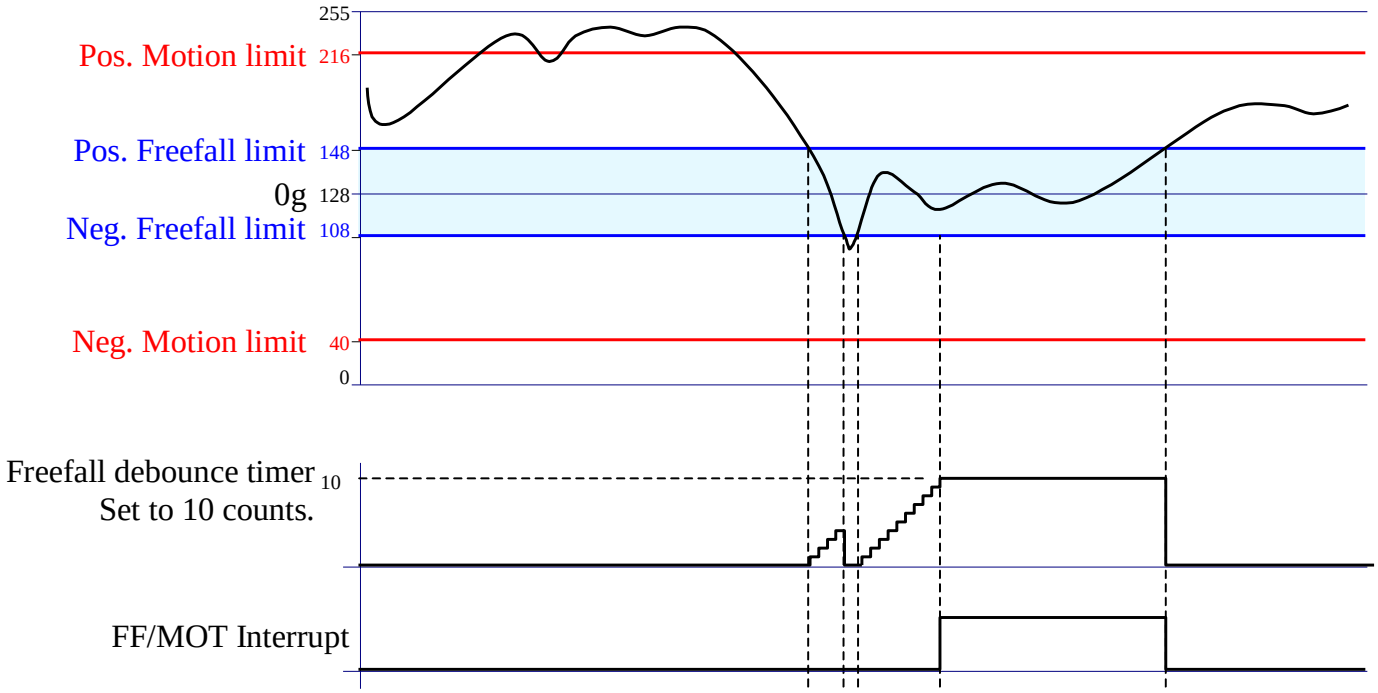


Figure 9. Typical Free-fall Interrupt Example (FFLat = 0, MOTen = 0)

© 2013 Kionix – All Rights Reserved

Typical Freefall Interrupt Example (Latching)

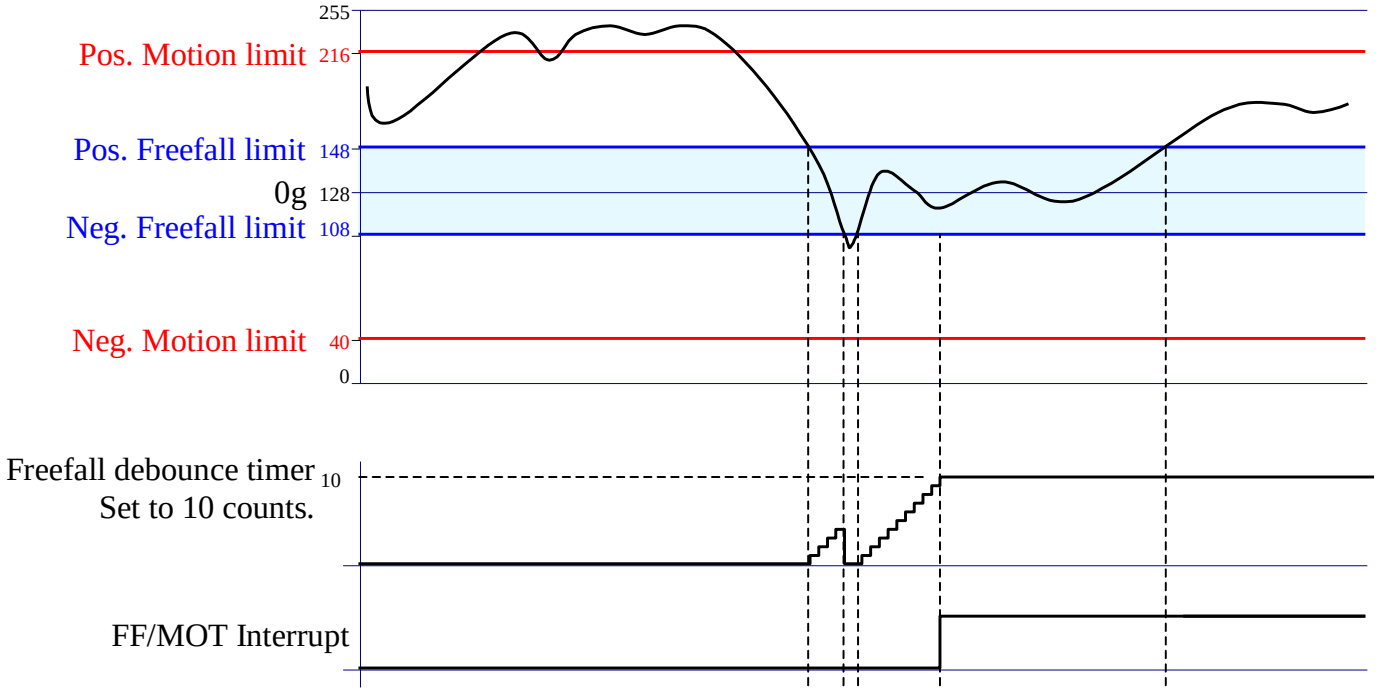


Figure 10. Typical Free-fall Interrupt Example (FFLat = 1, MOTen = 0)

FF_INT

Sets the free-fall interrupt threshold to this value

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
FFI7	FFI6	FFI5	FFI4	FFI3	FFI2	FFI1	FFI0	00001110
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

I^2C Address: 0x08h

SPI Read Address: 0x88h SPI Write Address: 0x08h

© 2013 Kionix – All Rights Reserved

FF_DELAY

Sets the free-fall delay/debounce time to this value

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
FFD7	FFD6	FFD5	FFD4	FFD3	FFD2	FFD1	FFD0	00000001
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

I^2C Address: 0x09h

SPI Read Address: 0x89h SPI Write Address: 0x09h

Free-fall Detect

The KXSS5 features a free-fall interrupt that sends a flag through pin 11 when the accelerometer senses a free-fall event. 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 KXSS5 gives the user the option to define the acceleration threshold value through the FF_INT 8-bit register where 256 counts cover the g range of the accelerometer. Equation 1 below shows how to calculate the FF_INT value needed for a desired acceleration threshold based on the Sensitivity.

$$FF_INT(counts) = \frac{Threshold(g) * Sensitivity(counts/g)}{16}$$

Equation 1. FF_INT Calculation

Through the FF_DELAY 8-bit register, the user can set the amount of time all three accelerometer axes must simultaneously remain below the FF_INT acceleration threshold before the free-fall interrupt flag is sent through pin 11. This delay/debounce time is defined by the available 0 to 255 counts, which represent accelerometer samples taken at the rate defined by IntSpd0 and IntSpd1. Equation 2 below shows how to calculate FF_DELAY for a desired debounce time (Delay) based on the Interrupt Sampling Rate (IntSpd0 and IntSpd1).

$$FF_DELAY(counts) = Delay(\text{sec}) * Interrupt\ Sampling\ Rate(Hz)$$

Equation 2. FF_DELAY Calculation

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.

© 2013 Kionix – All Rights Reserved

MOT_INT

Sets the motion activated interrupt acceleration threshold

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
MOTI7	MOTI6	MOTI5	MOTI4	MOTI3	MOTI2	MOTI1	MOTI0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	01010101

I^2C Address: 0x0Ah

SPI Read Address: 0x8Ah SPI Write Address: 0x0Ah

MOT_DELAY

Sets the motion activated delay/debounce time to this value

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
MOTD7	MOTD6	MOTD5	MOTD4	MOTD3	MOTD2	MOTD1	MOTD0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000001

I^2C Address: 0x0Bh

SPI Read Address: 0x8Bh SPI Write Address: 0x0Bh

Motion Detect

The KXSS5 also features a high-g motion interrupt that sends a flag through pin 11 when the accelerometer senses a high-g acceleration. A high-g acceleration is evident when any of the three accelerometer axes sense acceleration above a certain threshold for a set amount of time. The KXSS5 gives the user the option to define the acceleration threshold value through the MOT_INT 8-bit register where 256 counts cover the g range of the accelerometer. Equation 3 shows how to calculate the MOT_INT value needed for a desired acceleration threshold based on the Sensitivity.

$$MOT_INT(counts) = \frac{Threshold(g) * Sensitivity(counts/g)}{16}$$

Equation 3. MOT_INT Calculation

Through the MOT_DELAY 8-bit register, the user can set the amount of time that any of the three accelerometer axes has to sense acceleration above a certain threshold before the motion interrupt flag is sent through pin 11. This delay/debounce time is defined by the available 0 to 255 counts, which represent accelerometer samples taken at the rate defined by IntSpd0 and IntSpd1. Equation 4 below shows how to calculate MOT_DELAY for a desired debounce time (Delay) based on the Interrupt Sampling Rate (IntSpd0 and IntSpd1).

© 2013 Kionix – All Rights Reserved

$$MOT_DELAY(counts) = Delay(\text{sec}) * Interrupt\ Sampling\ Rate(Hz)$$

Equation 4. MOT_DELAY Calculation

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

© 2013 Kionix – All Rights Reserved

Kionix®	**± 3g Tri-axis Accelerometer Specifications**	**PART NUMBER:** **KXSS5-4457** **Rev. 3** **Feb-2013**

Revision History

REVISION	DESCRIPTION	DATE
1	Initial release	31-Jul-2009
2	Updated to new format and revision numbering	17-Dec-2009
3	Update max VDD from 5.25V to 3.6V	25-Feb-2013

"Kionix" is a registered trademark of Kionix, Inc. Products described herein are protected by patents issued or pending. No license is granted by implication or otherwise under any patent or other rights of Kionix. The information contained herein is believed to be accurate and reliable but is not guaranteed. Kionix does not assume responsibility for its use or distribution. Kionix also reserves the right to change product specifications or discontinue this product at any time without prior notice. This publication supersedes and replaces all information previously supplied.

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

© 2013 Kionix – All Rights Reserved