

Introduction

There is a growing interest in 6 Degrees-of-Freedom inertial sensor solutions for purpose of enhancing end user experience. This application note will help developers quickly implement proof-of-concept designs using Kionix's **KMX62**, a tri-axis accelerometer, tri-axis magnetometer, and temperature sensor on a single chip combo solution. Please refer to the KMX62 product specification for additional implementation guidelines. Kionix strives to ensure that our sensors will meet design expectations by default, but it is not possible to provide default settings to work in every environment. Depending on the intended application, it is very likely that some customization will be required to optimize performance. The information provided here will help the developer get the most out of the combo sensor.

Circuit Schematic

This section shows recommended wiring for the KMX62, based on proven operation of the part. Specific applications may require modifications from these recommendations. Please refer to the KMX62 product specifications for all pin descriptions.

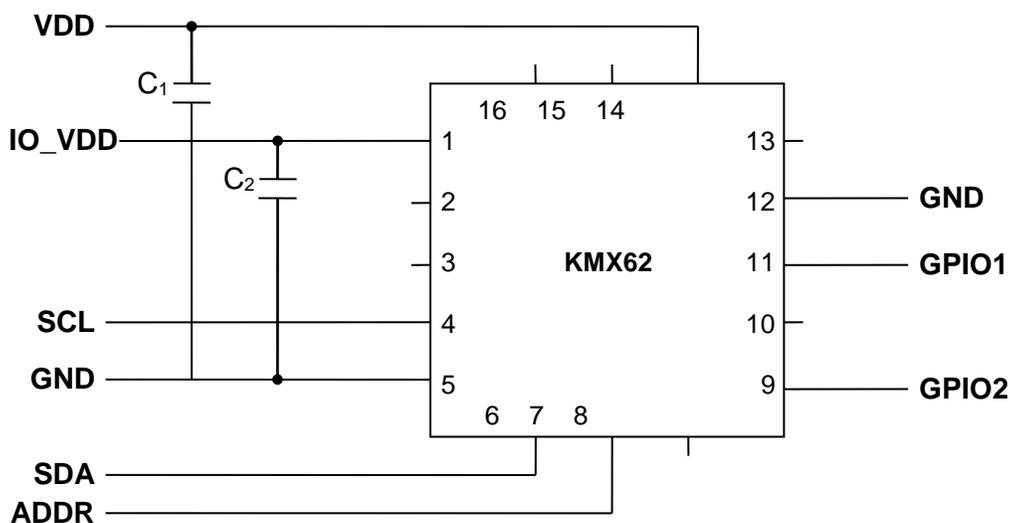


Figure 1. Application Schematic

Implementation

Here we present several basic ways to initialize the part. These can vary based on desired operation, but generally the initial operations a developer wants to do are: 1) read back acceleration and magnetic field data asynchronously, 2) read back acceleration and magnetic field data when next data is ready via interrupt, 3) use the Wake-Up function of the accelerometer and 4) use the Wake-Up function of the magnetometer to place sensor 5) Free Fall function of the sensor, 6) Buffer full function. These cursory solutions are provided as a means for configuring the part to a known operational state. Note that these conditions just provide a starting point, and the values may vary as developers refine their application requirements.

1. Asynchronous Read Back of Acceleration and Magnetic Field Data

- a) Write 0x00 to Control Register 2 (CNTL2) to disable the sensor.

Register Name	Address	Value
CNTL2	0x3A	0x00

- b) Write 0x22 to Output Data Rate (ODR) Control Register (ODCNTL) to set the ODR of the accelerometer and magnetometer to 50 Hz. (Note: This is also the default value.)

Register Name	Address	Value
ODCNTL	0x2C	0x22

- c) Write 0x0F to Control Register 2 (CNTL2) to (1) disable the temperature sensor, (2) set the G-range to $\pm 2g$, (3) enable High Resolution mode, and (4) set the accelerometer and magnetometer into operating mode.

Register Name	Address	Value
CNTL2	0x3A	0x0F

- d) When reading data asynchronously, there is potential to read the same value back if reading faster than 1/ODR cycle. It is recommended to wait 1/ODR cycle between readings.
- e) Acceleration and Magnetic field data can now be read asynchronously from the ACCEL_XOUT_L, ACCEL_XOUT_H, ACCEL_YOUT_L, ACCEL_YOUT_H, ACCEL_ZOUT_L, ACCEL_ZOUT_H, MAG_XOUT_L, MAG_XOUT_H, MAG_YOUT_L, MAG_YOUT_H, MAG_ZOUT_L, and MAG_ZOUT_H registers in 2's complement format.

2. Synchronous Hardware Interrupt Read Back of Acceleration and Magnetic Field Data

- a) Write 0x00 to Control Register 2 (CNTL2) to disable the sensor.

Register Name	Address	Value
CNTL2	0x3A	0x00

- b) Write 0x18 to Interrupt Control Register 1 (INC1) to enable reporting availability of new accelerometer and magnetometer data via physical interrupt pin GPIO1

Register Name	Address	Value
INC1	0x2A	0x18

- c) Write 0x04 to Interrupt Control Register 3 (INC3) to configure the physical interrupt pin to push/pull, active high, and latched operation.

Register Name	Address	Value
INC3	0x2C	0x04

- d) Write 0x22 to Output Data Rate (ODR) Control Register (ODCNTL) to set ODR of the accelerometer and magnetometer to 50 Hz. (Note: This is also the default value.)

Register Name	Address	Value
ODCNTL	0x38	0x22

- e) Write 0x0F to Control Register 2 (CNTL2) to (1) disable the temperature sensor, (2) set the G-range to $\pm 2g$, (3) enable High Resolution mode, and (4) set the accelerometer and magnetometer into operating mode.

Register Name	Address	Value
CNTL2	0x3A	0x0F

- f) Acceleration and Magnetic field data can now be read synchronously from the ACCEL_XOUT_L, ACCEL_XOUT_H, ACCEL_YOUT_L, ACCEL_YOUT_H, ACCEL_ZOUT_L, ACCEL_ZOUT_H, MAG_XOUT_L, MAG_XOUT_H, MAG_YOUT_L, MAG_YOUT_H, MAG_ZOUT_L, and MAG_ZOUT_H registers in 2's complement format.

3. Accelerometer Motion Detection Function

- a) Write 0x00 to Control Register 2 (CNTL2) to disable the sensor.

Register Name	Address	Value
CNTL2	0x3A	0x00

- b) Write 0x02 to Interrupt Control Register 1 (INC1) to enable reporting of the new accelerometer motion detection via physical interrupt pin GPIO1.

Register Name	Address	Value
INC1	0x2A	0x02

- c) Write 0x04 to Interrupt Control Register 3 (INC3) to configure the physical interrupt pin to push/pull, active high, and latched operation.

Register Name	Address	Value
INC3	0x2C	0x04

- d) Write 0x3F to Interrupt Control Register 4 (INC4) to define the direction of detected motion for all positive and negative directions.

Register Name	Address	Value
INC4	0x2D	0x3F

- e) Write 0x08 to Accelerometer Motion Control Register 1 (AMI_CNTL1) to set the threshold level to 0.5g. The following formula is used:

$$AMITH \text{ (counts)} = \text{Desired Threshold (g)} \times 16 \text{ (counts/g)}$$

$$AMITH \text{ (counts)} = 0.5 \text{ g} \times 16 \text{ count/g} = 8 \text{ counts}$$

Register Name	Address	Value
AMI_CNTL1	0x2F	0x08

- f) Write 0x86 to Accelerometer Motion Control Register 3 (AMI_CNTL3) to enable the accelerometer motion interrupt engine, set the interrupt to latched mode, and set the Output Data Rate (OAMI<2:0>) of the accelerometer motion detection to 50 Hz.

Register Name	Address	Value
AMI_CNTL3	0x31	0x86

- g) Write 0x05 to Accelerometer Motion Control Register 2 (AMI_CNTL2) to set the time motion must be present before a wake-up interrupt is set to 0.1 second. The following formula is used:

$$AMICT \text{ (counts)} = \text{Desired Delay Time (sec)} \times \text{Wake-up Function ODR (Hz)}$$

$$AMICT \text{ (counts)} = 0.1 \text{ sec} \times 50 \text{ Hz} = 5 \text{ counts}$$

Since the desired delay time is 0.1 second and the Wake-up Function ODR is set to 50 Hz, then the motion detection timer is 5 counts (0x05).

Register Name	Address	Value
AMI_CNTL2	0x30	0x05

- h) Write 0x01 to Control Register 2 (CNTL2) to enable the accelerometer. The accelerometer must be enabled for motion detection.

Register Name	Address	Value
CNTL2	0x3A	0x01

- i) Monitor the physical interrupt GPIO1, if the acceleration input profile satisfies the criteria previously established for the 0.5g motion detect threshold level in both positive and negative directions of the X, Y, Z axis for more than 0.1 second, then there should be positive latched interrupt present. Accelerometer Motion wake-up event will also be reflected in bit 1 (AMI) of the Interrupt Status 1 (INS1) register.
- j) Read Interrupt Release (INL) register to unlatch (clear) the output interrupt created by the motion detection function.

Register Name	Address	Value
INL	0x05	n/a

4. Magnetometer Motion Detection Function

- a) Write 0x00 to Control Register 2 (CNTL2) to disable the sensor.

Register Name	Address	Value
CNTL2	0x3A	0x00

- b) Write 0x01 to Interrupt Control Register 1 (INC1) to enable reporting of the new magnetometer motion detection via physical interrupt pin GPIO1.

Register Name	Address	Value
INC1	0x2A	0x01

- c) Write 0x04 to Interrupt Control Register 3 (INC3) to configure the physical interrupt pin to push/pull, active high, and latched operation.

Register Name	Address	Value
INC3	0x2C	0x04

- d) Write 0x3F to Interrupt Control Register 5 (INC5) to define the direction of detected motion for all positive and negative directions: x positive (x+), x negative (x-), y positive (y+), y negative (y-), z positive (z+), z negative (z-) directions.

Register Name	Address	Value
INC5	0x2E	0x3F

- e) Write 0x01 to Magnetometer Motion Control Register 1 (MMI_CNTL1) to set the threshold level to 9.35 μ T. The following formula is used:

$$MMITH \text{ (counts)} = \text{Desired Threshold } (\mu T) \times 0.107 \text{ (counts}/\mu T)$$

$$MMITH \text{ (counts)} = 9.35 \mu T \times 0.107 \text{ counts}/\mu T = 1 \text{ count}$$

Note that this threshold is differential with respect to the previous reading.

Register Name	Address	Value
MMI_CNTL1	0x32	0x01

- f) Write 0x86 to Magnetometer Motion Control Register 3 (MMI_CNTL3) to enable the magnetometer motion interrupt engine, set the interrupt to latched mode, and set the Output Data Rate (OMMI<2:0>) of the magnetometer motion detection to 50 Hz.

Register Name	Address	Value
MMI_CNTL3	0x34	0x86

- g) Write 0x05 to Magnetometer Motion Control Register 2 (MMI_CNTL2) to set the time motion must be present before a wake-up interrupt is set to 0.1 second. The following formula is used:

$$MMICT \text{ (counts)} = \text{Desired Delay Time (sec)} \times \text{Wake-up Function ODR (Hz)}$$

$$MMICT \text{ (counts)} = 0.1 \text{ sec} \times 50 \text{ Hz} = 5 \text{ counts}$$

Since the desired delay time is 0.1 second and the OMMI<2:0> is set to 50, then the motion detection timer is 5 counts (0x05).

Register Name	Address	Value
MMI_CNTL2	0x33	0x05

- h) Write 0x02 to Control Register 2 (CNTL2) to enable the magnetometer. The accelerometer must be enabled for motion detection.

Register Name	Address	Value
CNTL2	0x3A	0x02

- a) Monitor the physical interrupt GPIO1, if the magnetometer input profile satisfies the criteria previously established for the 9.35 μ T motion detect threshold level in both positive and negative directions of the X, Y, Z axis for more than 0.1 second, then there should be positive latched interrupt present. Magnetometer Motion wake-up event will also be reflected in bit 0 (MMI) of the Interrupt Status 1 (INS1) register.
- i) Read Interrupt Release (INL) register to unlatch (clear) the output interrupt created by the motion detection function.

Register Name	Address	Value
INL	0x05	n/a

5. Free Fall Function

- b) Write 0x00 to Control Register 2 (CNTL2) to disable the sensor.

Register Name	Address	Value
CNTL2	0x3A	0x00

- c) Write 0x04 to Interrupt Control Register 1 (INC1) to enable reporting of the free fall event via physical interrupt pin GPIO1.

Register Name	Address	Value
INC1	0x2A	0x04

- d) Write 0x04 to Interrupt Control Register 3 (INC3) to configure the physical interrupt pin to push/pull, active high, and latched operation.

Register Name	Address	Value
INC3	0x2C	0x04

- e) Write 0x08 to the Free Fall Control Register 1 (FFI_CNTL1). This sets the free fall interrupt threshold (FFITH<7:0>) to 0.5g.

$$FFITH \text{ (counts)} = \text{Free Fall Threshold (g)} \times 16 \text{ (counts/g)}$$

$$FFITH \text{ (counts)} = 0.5 \text{ g} \times 16 \text{ counts/g} = 8 \text{ counts}$$

Register Name	Address	Value
FFI_CNTL1	0x35	0x08

- f) Write 0x80 to the Free Fall Control Register 3 (FFI_CNTL3). This enables the free fall engine, configures for latched control, count up/down debouncing, and output data rate of the free fall detection as set by bits OFFI<2:0> to 12.5Hz

Register Name	Address	Value
FFI_CNTL3	0x37	0x80

- g) Write 0x04 to the Free Fall Control Register 2 (FFI_CNTL2). This sets the amount of time *all three* accelerometer axes must simultaneously remain below the FFITH acceleration threshold before the Free fall interrupt flag is sent through the GPIO1 or the GPIO2 output pins. The delay of 0.320 sec is a good starting point.

$$FFICT \text{ (counts)} = \text{Desired Delay Time (sec)} \times \text{Free fall ODR (Hz)}$$

$$FFICT \text{ (counts)} = 0.320 \text{ sec} \times 12.5 \text{ Hz} = 4 \text{ counts}$$

Register Name	Address	Value
FFI_CNTL2	0x36	0x04

- h) Write 0x01 to Control Register 2 (CNTL2) to enable the accelerometer. The accelerometer must be enabled for free fall detection.

Register Name	Address	Value
CNTL2	0x3A	0x01

- i) Monitor the physical interrupt GPIO1 of the accelerometer, if the acceleration input profile satisfies the criteria previously established for the 0.5g free fall detect threshold level in both positive and negative directions of the X, Y, Z axis for more than 0.320 seconds, then there should be positive latched interrupt present. Free fall events will now be reflected in bit 2 (FFI) of the Interrupt Status 1 (INS1) register.
- j) Read Interrupt Release (INL) register to unlatch (clear) the output interrupt created by the motion detection function.

Register Name	Address	Value
INL	0x05	n/a

6. Buffer Full Function

- a) Write 0x00 to Control Register 2 (CNTL2) to disable the sensor.

Register Name	Address	Value
CNTL2	0x3A	0x00

- b) Write 0x40 to Interrupt Control Register 1 (INC1) to enable reporting of the buffer full event via physical interrupt pin GPIO1.

Register Name	Address	Value
INC1	0x2A	0x40

- c) Write 0x04 to Interrupt Control Register 3 (INC3) to configure the physical interrupt pin to push/pull, active high, and latched operation.

Register Name	Address	Value
INC3	0x2C	0x04

- d) Write 0x22 to the Output Data Rate (ODR) Control Register (ODCNTL). This sets the ODR for the accelerometer and magnetometer to 50Hz (default).

Register Name	Address	Value
ODCNTL	0x38	0x22

- e) Write 0x00 to the Buffer Control Register 2 (BUF_CNTL_2). This sets the buffer FIFO mode and will collect data until full.

Register Name	Address	Value
BUF_CNTL_2	0x78	0x00

- f) Write 0xFF to the Buffer Control Register 3 (BUF_CNTL_3). This setting enables the buffer and configures the buffer to collect magnetometer x, y, and z data; accelerometer x, y and z data; and temperature data.

Register Name	Address	Value
BUF_CNTL_3	0x79	0xFF

- g) Write 0x43 to Control Register 2 (CNTL2) to enable the accelerometer, magnetometer and temperature sensors. These sensors must be enabled for buffer filling.

Register Name	Address	Value
CNTL2	0x3A	0x43

- h) Monitor the physical interrupt GPIO1, when the buffer is full an interrupt will be generated and there should be positive latched interrupt present. Buffer full events will also be reflected in bit 6 (BFI) of the Interrupt Status 1 (INS1) register.
- i) Once Buffer-Full Interrupt is issued on INT1 pin, data can then be read from the Buffer Read (BUF_READ) register at address 0x7E in 2's complement format. Since the data being stored in the buffer is a combination of magnetometer, accelerometer and temperature data; the data read back (1 byte each) will be in the following order:

ACCEL_XOUT_L, ACCEL_XOUT_H,
ACCEL_YOUT_L, ACCEL_YOUT_H,
ACCEL_ZOUT_L, ACCEL_ZOUT_H,
MAG_XOUT_L, MAG_XOUT_H,
MAG_YOUT_L, MAG_YOUT_H,
MAG_ZOUT_L, MAG_ZOUT_H,
TEMP_OUT_L, TEMP_OUT_H.

Register Name	Address	Value
BUF_READ	0x7E	n/a

Timing Requirements

There are several timing requirements that developers should keep in mind when working with the KMX62. For detailed information, see *TN005 Power-On Procedure* technical note ([LINK](#)).

I²C Clock - The I²C Clock can support Fast Mode up to **400 kHz** and High-Speed mode up to **3.4 MHz**.

Power Up to Communication - After the part is powered up, it takes maximum **50 ms** before it is ready for I²C communication.

Enable to Valid Outputs - After the part is enabled (ACCEL_EN or MAG_EN bit in CNTL2 is asserted), it takes from **1 ms** to **1300 ms** (varies with Output Data Rate (ODR) and mode setting RES). See KMX62 Product Specification for details.

Software Reset - After a Software Reset, the part takes maximum of **2ms** before it is ready for I²C communication.

Interrupt Configuration

Each physical interrupt (GPIO1 and GPIO2) have 6 possible configurations, based on two states for each of the three customizable variables located in Interrupt Control Register 3:

Latched/Pulsed (IEL)

- 0 – Latched mode – When an interrupt is triggered, it will remain active on the pin until cleared.
- 1 – Pulse mode – When an interrupt is triggered, it will cause a short (~50µs) pulse on the pin and clear itself.

Polarity (IEA)

- 0 – Active Low – The interrupt pin will normally be HIGH, but will transition to LOW when an interrupt is triggered.
- 1 – Active High – The interrupt pin will normally be LOW, but will transition to HIGH when an interrupt is triggered.

Enable/Disable (IEN)

- 0 – Disabled – Interrupt conditions will not be reflected on the physical interrupt pin.
- 1 – Enabled – Interrupt conditions will be reflected on the physical interrupt pin.

A Few Interrupt Tips

Read the Interrupt Release Register to Clear

In latched mode, the INL register must be read in order to clear the physical interrupt pin. This will also clear the Interrupt Source Registers and the INT bit (0x10) in the Status Register.

Microcontroller/GPIO Interrupt Handling –

GPIO configuration is based solely on the connected hardware. The KMX62 can be configured to issue interrupts depending on how the GPIO is programmed to catch them (if this is not the case, please contact your Kionix Sales Representative). Generally, when an interrupt is triggered, the developer should take the following steps:

- 1- Disable GPIO interrupt
- 2- Clear GPIO interrupt and generate desired functionality
- 3- Enable GPIO interrupt

These steps should be taken without calling any digital communication transactions if done in an interrupt context, because the operating system or kernel will not allow busy-waiting on an I/O operation during an interrupt service routine.

Interrupt Polling - If physical interrupts are not used, a polling mechanism can be devised, which checks the INT bit in INS1 (Interrupt Source Register 1) to be set. In addition, this mechanism can also look at the remaining bits in INS1 to determine which function caused the interrupt and what steps should be taken before clearing the interrupt source information by reading the INL register.

Troubleshooting

All Interrupt Issues

- Make sure the KMX62 is configured to issue interrupt signals in the way that your GPIO is programmed to handle them.
- An oscilloscope on the physical interrupt pin can be a valuable tool to confirm physical interrupt operation.
- Check the interrupt routing on the physical interrupt pin (INC1 for GPIO1 and INC2 for GPIO2)

AMI (Active Motion Interrupt) Not Working

- Make sure that the AMI engine is enabled (AMI_EN bit in Active Motion Control Register 3).
- Try altering the threshold requirements to achieve desired operation. If the part is waking up too easily, try increasing the threshold. If the interrupt is not firing at all, the threshold may be set too high.
- Try shortening the timer requirements, and make sure the acceleration on an unmasked axis is above the threshold until the expiration of the Active Motion Interrupt Counter (AMICT in AMI_CNTL2).
- Double check the Accelerometer Axis Mask bits in Interrupt Control 3 (INC4)

MMI (Magnetometer Motion Interrupt) Not Working

- Make sure that the MMI engine is enabled (MMI_EN bit in Magnetometer Motion Control Register 3).
- Try altering the threshold requirements to achieve desired operation. If the part is waking up too easily, try increasing the threshold. If the interrupt is not firing at all, the threshold may be set too high.
- Try shortening the timer requirements, and make sure the acceleration on an unmasked axis is above the threshold until the expiration of the Magnetometer Motion Interrupt Counter (AMICT in MMI_CNTL2).
- Double check the Magnetometer Axis Mask bits in Interrupt Control 4 (INC4)

BFI (Buffer Full Interrupt) Not Working

- Make sure that the BFI engine is enabled (BFI_EN bit in Buffer Control Register 3).
- Make sure the expected data is enabled to be buffered (BUF_(AX, AY, AZ, MX, MY, MZ, TEMP) in Buffer Control Register 3)
- Make sure the appropriate sensor is enabled.
- Read the sample level from the Buffer Status 1 (BUF_STATUS_1) register to see if it is filling. Note: It is not recommended to read from the buffer status register while the buffer is filling as it may influence data storage into the buffer.

Placement and Orientation

Placement

- It is important to note that the placement of the KMX62 within the target device can especially have a significant effect on the magnetometer portion of the sensor. Please refer to [AN043 Preliminary Review for Magnetometer Integration](#) and [AN042 Magnetometer placement in Mobile Devices](#) for further guidance on magnetometer integration.

The Kionix Advantage

Kionix technology provides 6 Degrees-of-Freedom inertial sensor system on a single, silicon chip, which is designed to strike a balance between current consumption and noise performance with excellent bias stability over temperature. A magnetometer accelerometer can be used to enable a variety of simultaneous features including, but not limited to:

- Hard Disk Drive protection
- Machine Health/Vibration analysis
- Tilt screen navigation
- Sports modeling
- Theft, man-down, accident alarm
- Image stability, screen orientation & scrolling
- Computer pointer
- Navigation, mapping
- Game playing
- Automatic sleep mode

Theory of Operation

Kionix MEMS linear tri-axis accelerometers function on the principle of differential capacitance. Acceleration causes displacement of a silicon structure resulting in a change in capacitance. A signal-conditioning CMOS technology ASIC detects and transforms changes in capacitance into an analog output voltage, which is proportional to acceleration. These outputs can then be sent to a micro-controller for integration into various applications.

Magnetic sensing is based on the principle of magnetic impedance. The magnetic sensor detects very small magnetic fields by passing an electric pulse through a special electron spin aligned amorphous wire. Due to the high Curie temperature of the wire, the sensor's thermal performance shows excellent stability.

For product summaries, specifications, and schematics, please refer to the Kionix MEMS 6-axis combo parts and 9-axis solutions product catalog at <http://www.kionix.com/parametric/6-Axis Combo Parts And 9-Axis Solutions>