Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

Freescale Semiconductor
Data Sheet: Technical Data
An Energy-Efficient Solution by Freescale

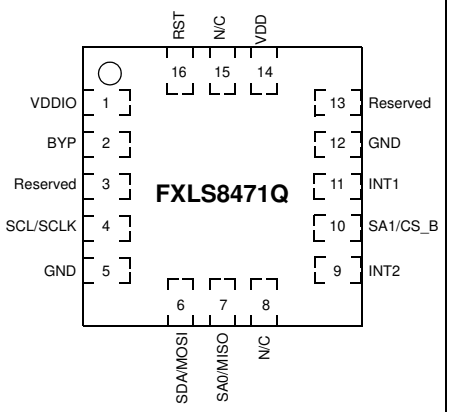
Document Number: FXLS8471Q
Rev. 1.5, 06/2015

FXLS8471Q, 3-Axis, Linear Accelerometer

FXLS8471Q is a small, low-power, 3-axis, linear accelerometer in a 3 mm x 3 mm x 1 mm QFN package. FXLS8471Q has dynamically selectable acceleration full-scale ranges of ±2 *g*/±4 *g*/±8 *g* and 14 bits of resolution. Output data rates (ODR) are programmable from 1.563 Hz to 800 Hz. I^2C and SPI serial digital interfaces are provided along with several user programmable event detection functions that can be used to reduce the overall system power consumption by off-loading the host processor. FXLS8471Q is guaranteed to operate over the extended temperature range of -40 °C to +85 °C.

FXLS8471Q

16 LEAD QFN
3 mm x 3 mm x 1 mm



Pin Connections

Features

- 1.95 V to 3.6 V VDD supply voltage, 1.62 V to 3.6 V VDDIO voltage
- ±2 *g*/±4 *g*/±8 *g* dynamically selectable acceleration full-scale ranges
- Output Data Rates (ODR) from 1.563 Hz to 800 Hz
- Low noise: typically 99 µg/Hz in low-noise mode @ 200-Hz bandwidth
- 14-bit ADC resolution: 0.244 mg/LSB in ±2 *g,* full-scale range
- Embedded programmable acceleration event functions:
  - Freefall and Motion Detection
  - Transient Detection
  - Vector-Magnitude Change Detection
  - Pulse and Tap Detection (Single and Double)
  - Orientation Detection (Portrait/Landscape)
- Programmable automatic ODR change using Auto-Wake and return to Sleep functions to save power.
- 192-byte FIFO buffer, capable of storing up to 32 samples of X/Y/Z data
- Supports 4-wire SPI interface at up to 1 MHz; I^2C Normal (100 kHz) and Fast Modes (400 kHz)
- Integrated accelerometer self-test function

Target Markets

- Industrial applications: vibration analysis, machine health monitoring, and platform stabilization
- Smartphones, tablets, digital cameras, and personal navigation devices
- Medical applications: patient monitoring, fall detection, and rehabilitation

Applications

- Shock and vibration monitoring (mechatronic compensation, shipping, and warranty usage logging)
- User interface (menu scrolling by orientation change, tap detection for button replacement)
- Orientation detection (portrait/landscape: up/down, left/right, back/front position identification)
- Gaming and real-time activity analysis (pedometry, freefall and drop detection for hard disk drives and other devices)
- Power management for mobile devices using inertial event detection

Freescale reserves the right to change the detail specifications as may be required to permit improvements in the design of its products.

ORDERING INFORMATION			
Part Number	**Temperature Range**	**Package Description**	**Shipping**
FXLS8471QR1	-40°C to +85°C	QFN-16	Tape and Reel (1 k)

Related Documentation

The FXLS8471Q device features and operations are described in a variety of reference manuals, user guides, and application notes. To find the most-current versions of these documents:

1. Go to the Freescale homepage at:

   http://www.freescale.com/

2. In the Keyword search box at the top of the page, enter the device number FXLS8471Q.

In the Refine Your Result pane on the left, click on the Documentation link.

Contents

1 Block Diagram

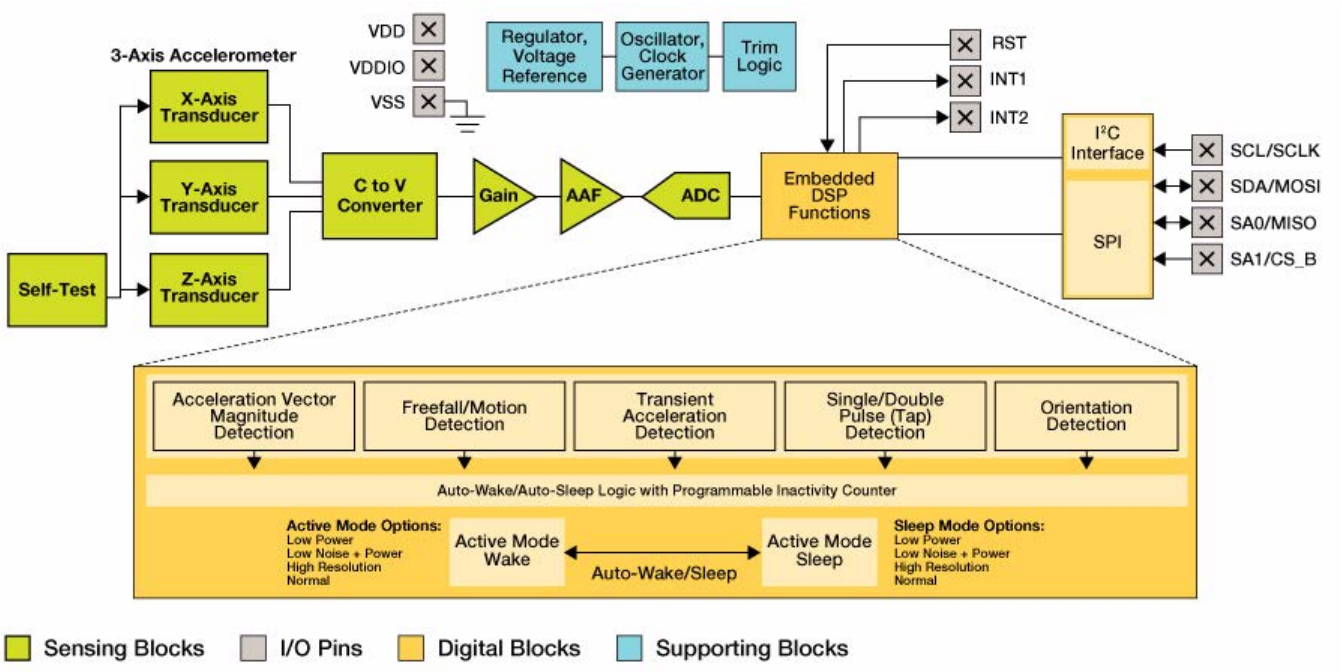
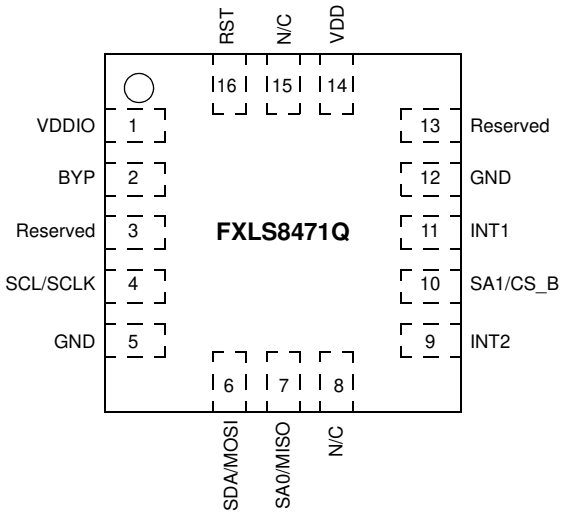


Figure 1. Block diagram

2 Pin Description



Top View
16 Lead QFN-COL
3 mm x 3 mm x 1 mm

Figure 2. Pinout diagram

Table 1. Pin Description

Pin	Name	Function
1	VDDIO	Interface power supply
2	BYP	Internal regulator output bypass capacitor connection
3	Reserved	Test reserved, connect to GND
4	SCL/SCLK	I^2C Serial Clock/SPI Clock
5	GND	Ground
6	SDA/MOSI	I^2C Serial Data/SPI Master Out, Slave In
7	SA0/MISO(1)	I^2C address selection bit 0/SPI Master In, Slave Out
8	N/C	Internally not connected
9	INT2	Interrupt 2
10	SA1/CS_B	I^2C address selection bit 1(2)/SPI Chip Select (active low)
11	INT1	Interrupt 1
12	GND	Ground
13	Reserved	Test reserved, connect to GND
14	VDD	Power supply
15	N/C	Internally not connected
16	RST	Reset input, active high. Connect to GND if unused

1. The SA0 pin is also used to select the desired serial interface mode during POR and also after a hard/soft reset event. Please see Section 6.2.3, "I^2C/SPI auto detection" for more information
2. See Table 8 for I^2C address options selectable using the SA0 and SA1 pins.

Device power is supplied through the VDD pin. Power-supply, decoupling capacitors (100 nF ceramic plus 4.7 µF or larger bulk) should be placed as close as possible to pin 14 of the device. The digital interface supply voltage (VDDIO) must also be decoupled with a 100 nF ceramic capacitor placed as close as possible to pin 1 of the device.

The digital control signals SCL, SDA, SA0, SA1, and RST are not tolerant of voltages more than VDDIO + 0.3 V. If VDDIO is removed, these pins will clamp any logic signals through their internal ESD protection diodes.

The function and timing of the two interrupt pins (INT1 and INT2) are user-programmable through the I²C/SPI interface. The SDA and SCL I²C connections are open drain and therefore require a pullup resistor as shown in the application diagram in Figure 3. The INT1 and INT2 pins may also be configured for open-drain operation. If they are configured for open drain, external pullup resistors are required.

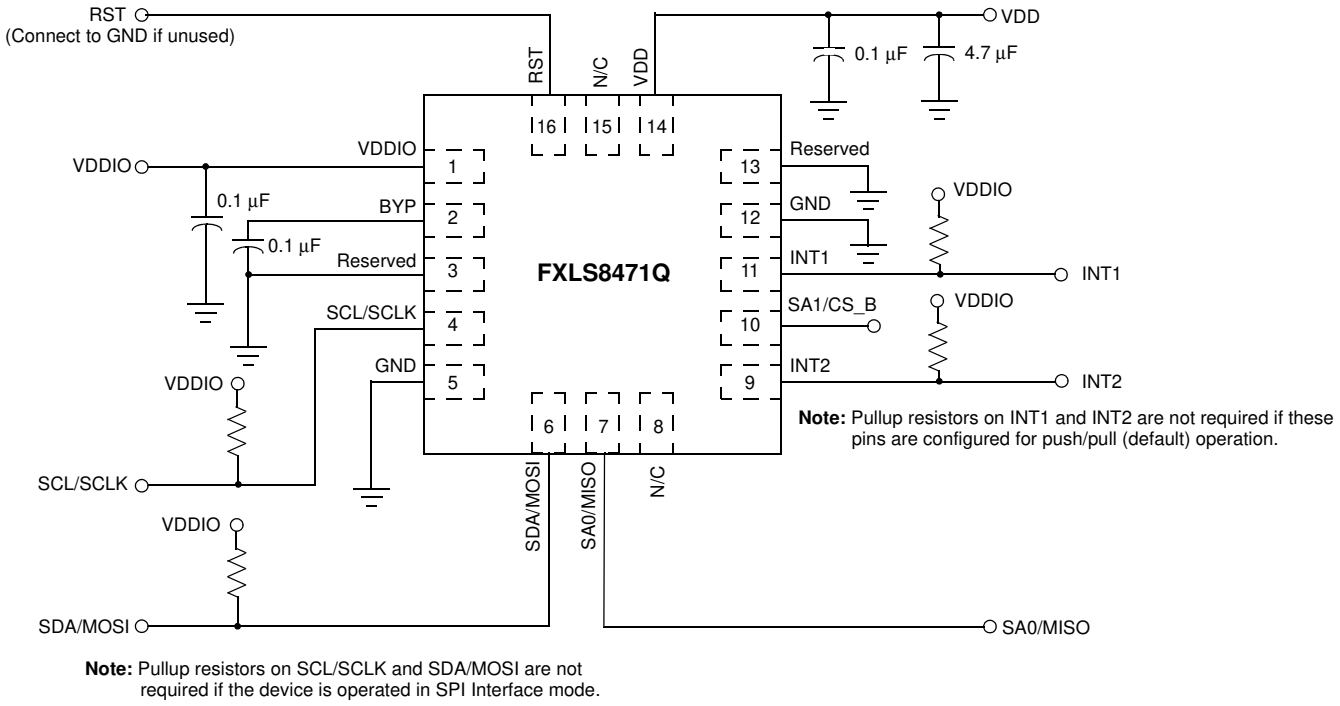


Figure 3. Electrical connection

2.1 Soldering information

The QFN package is compliant with the RoHS standards. Please refer to Freescale application note AN4077 for more information.

2.2 Orientation

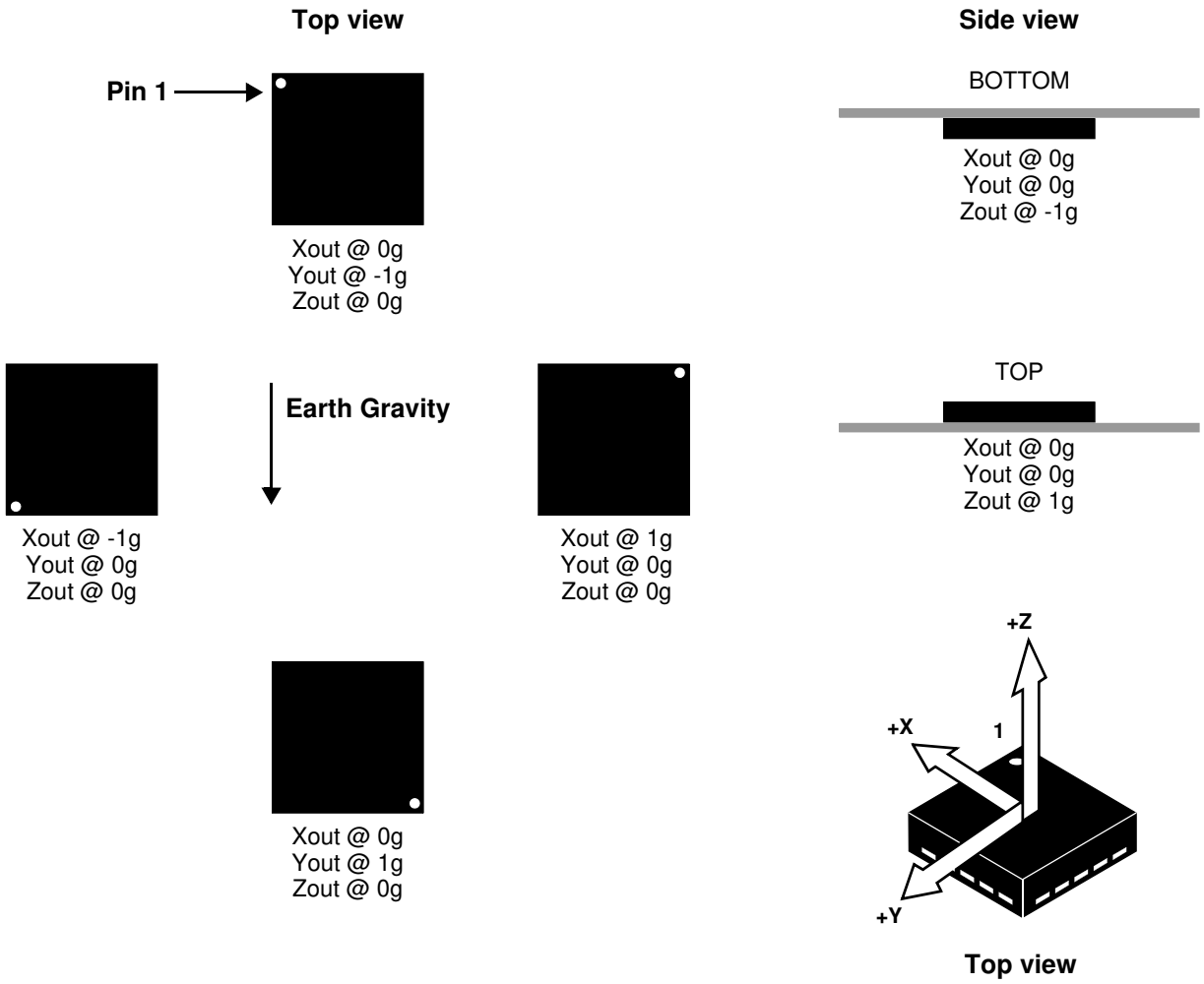


Figure 4. Component axes orientation and response to gravity stimulus

3 Example FXLS8471Q Driver Code

3.1 Introduction

It is very straightforward to configure the FXLS8471Q and start receiving data from the three accelerometerchannels. Unfortunately, since every hardware platform will be different, it is not possible to provide completely portable software drivers. This section therefore provides real FXLS8471Q driver code for a Kinetis uC board running under the MQX operating system. The I^2C functions s_i2c_read_regs and s_i2c_write_regs are not provided here and should be replaced with the corresponding low level I^2C driver code on the development platform.

3.2 FXLS8471Q Addresses

This section lists the I^2C address of the FXLS8471Q. The I^2C address depends on the logic level of FXLS8471Q pins SA0 and SA1 so the I^2C address may be 0x1C, 0x1D, 0x1E or 0x1F.

Example 1.

```
// FXLS8471Q I2C address
#define FXLS8471Q_SLAVE_ADDR      0x1E    // with pins SA0=0, SA1=0
```

Some of the key FXLS8471Q internal register addresses are listed below.

Example 2.

```
// FXLS8471Q internal register addresses
#define FXLS8471Q_STATUS          0x00
#define FXLS8471Q_WHOAMI          0x0D
#define FXLS8471Q_XYZ_DATA_CFG    0x0E
#define FXLS8471Q_CTRL_REG1       0x2A
#define FXLS8471Q_WHOAMI_VAL      0x6A
```

The reference driver here does a block read of the FXLS8471Q status byte plus three 16-bit accelerometer channels.

Example 3.

```
// number of bytes to be read from FXLS8471Q
#define FXLS8471Q_READ_LEN 7// status plus 3 accelerometer channels
```

3.3 Sensor data structure

The high and low bytes of the three accelerometer are placed into a structure of type SRAWDATA containing three signed short integers.

Example 4.

```
typedef struct
{
   int16_t  x;
   int16_t  y;
   int16_t  z;
} SRAWDATA;
```

3.4 FXLS8471Q Configuration function

This function configures the FXLS8471Q for a 200-Hz ODR. The code is self-explanatory and can be easily customized for different settings.

Example 5.

```
// function configures FXLS8471Q accelerometer sensor
static _mqx_int s_FXLS8471Q_start(MQX_FILE_PTR aFP)
{
   uint8_t databyte;

   // read and check the FXLS8471Q WHOAMI register
   if (s_i2c_read_regs(aFP, FXLS8471Q_SLAVE_ADDR, FXLS8471Q_WHOAMI, &databyte,
      (uint8_t) 1) != 1)
   {
      return (I2C_ERROR);
   }
   if (databyte != FXLS8471Q_WHOAMI_VAL)
   {
      return (I2C_ERROR);
   }

   // write 0000 0000 = 0x00 to accelerometer control register 1 to place FXLS8471Q into
   // standby
   // [7-1] = 0000 000
   // [0]: active=0
   databyte = 0x00;
   if (s_i2c_write_regs(aFP, FXLS8471Q_SLAVE_ADDR, FXLS8471Q_CTRL_REG1, &databyte,
      (uint8_t) 1) != 1)
   {
      return (I2C_ERROR);
   }

   // write 0000 0001= 0x01 to XYZ_DATA_CFG register
   // [7]: reserved
   // [6]: reserved
   // [5]: reserved
   // [4]: hpf_out=0
   // [3]: reserved
   // [2]: reserved
   // [1-0]: fs=01 for accelerometer range of +/-4g with 0.488mg/LSB
   databyte = 0x01;
   if (s_i2c_write_regs(aFP, FXLS8471Q_SLAVE_ADDR, FXLS8471Q_XYZ_DATA_CFG,
      &databyte, (uint8_t) 1) != 1)
   {
      return (I2C_ERROR);
   }

   // write 0001 0101b = 0x15 to accelerometer control register 1
   // [7-6]: aslp_rate=00
   // [5-3]: dr=010 for 200Hz data rate
   // [2]: lnoise=1 for low noise mode
   // [1]: f_read=0 for normal 16 bit reads
   // [0]: active=1 to take the part out of standby and enable sampling
   databyte = 0x15;
   if (s_i2c_write_regs(aFP, FXLS8471Q_SLAVE_ADDR, FXLS8471Q_CTRL_REG1, &databyte,
      (uint8_t) 1) != 1)
```

```
  {
    return (I2C_ERROR);
  }

  // normal return
  return (I2C_OK);
}
```

3.5 FXLS8471Q Data Read function

This function performs a block read of the status and acceleration data and places the bytes read into the structures of type SRAWDATA as signed short integers.

Example 6.

```
// read status and the three channels of accelerometer data from
// FXLS8471Q (7 bytes)
int16_t ReadAccel(SRAWDATA *pAccelData)
{
  MQX_FILE_PTR fp;                          // I2C file pointer
  uint8_t Buffer[FXLS8471Q_READ_LEN];       // read buffer

  // read FXLS8471Q_READ_LEN=7 bytes (status byte and the three channels of data)
  if (s_i2c_read_regs(fp, FXLS8471Q_SLAVE_ADDR, FXLS8471Q_STATUS, Buffer,
    FXLS8471Q_READ_LEN) == FXLS8471Q_READ_LEN)
  {
    // copy the 14 bit accelerometer byte data into 16 bit words
    pAccelData->x = ((Buffer[1] << 8) | Buffer[2])>> 2;
    pAccelData->y = ((Buffer[3] << 8) | Buffer[4])>> 2;
    pAccelData->z = ((Buffer[5] << 8) | Buffer[6])>> 2;
  }
  else
  {
    // return with error
    return (I2C_ERROR);
  }

  // normal return
  return (I2C_OK);
}
```

4 Terminology

4.1 Sensitivity

The sensitivity is represented in LSB/g. In 2-*g* mode the sensitivity is 4096 LSB/g. In 4-*g* mode the sensitivity is 2048 LSB/ *g* and in 8-*g* mode the sensitivity is 1024 LSB/g.

4.2 Zero-g Offset

Zero-g Offset describes the deviation of an actual output signal from the ideal output signal if the sensor is stationary. A sensor stationary on a horizontal surface will measure 0 *g* in X-axis and 0 *g* in Y-axis whereas the Z-axis will measure 1 *g*. A deviation from ideal value in this case is called Zero-*g* offset. Offset is to some extent a result of stress on the MEMS sensor and therefore the offset can slightly change after mounting the sensor onto a printed circuit board or exposing it to extensive mechanical stress.

4.3 Self-Test

Self-Test can be used to verify the transducer functionality without applying an external mechanical stimulus. When Self-Test is activated, an electrostatic actuation force is applied to the sensor, simulating a small acceleration. In this case, the sensor outputs will exhibit a change in their DC levels which are related to the selected full scale through the device sensitivity. When Self-Test is activated, the device output level is given by the algebraic sum of the signals produced by the acceleration acting on the sensor and by the electrostatic self-test force.

5 Device Characteristics

5.1 Mechanical characteristics

Table 2. Mechanical characteristics @ VDD = 2.5 V, VDDIO = 1.8 V T = 25°C unless otherwise noted.

Parameter	Test Conditions	Symbol	Min	Typ	Max	Unit
Measurement range(1)	±2 *g* mode	FS_{ACC}		±2		*g*
	±4 *g* mode			±4		
	±8 *g* mode			±8		
Sensitivity	±2 *g* mode	SEN_{ACC}		4096		LSB/*g*
				0.244		mg/LSB
	±4 *g* mode			2048		LSB/*g*
				0.488		mg/LSB
	±8 *g* mode			1024		LSB/*g*
				0.976		mg/LSB
Sensitivity change with temperature(1)	±2 *g*, ±4 *g*, ±8 *g* modes	TCS_{ACC}		±0.01		%/°C
Sensitivity accuracy	@ 25°C	$SEN\text{-}TOL_{ACC}$		±2.5		$\%SEN_{ACC}$
Zero-*g* level offset accuracy(2)	±2 *g*, ±4 *g*, ±8 *g* modes	OFF_{ACC}		±20		mg
Zero-*g* level offset accuracy post-board mount(3)	±2 *g*, ±4 *g*, ±8 *g* modes	$OFF\text{-}PBM_{ACC}$		±30		mg
Zero-*g* level change versus temperature	-40°C to 85°C(1)	TCO_{ACC}		±0.2		mg/°C
Nonlinearity (deviation from straight line)(4)(5)	Over ±1 *g* range normal mode	NL_{ACC}		±0.5		$\%FS_{ACC}$
Self-Test output change(6) X Y Z	Set to ±2 *g* mode	$STOC_{ACC}$	 +192 +270 +1275			LSB
Output noise density(4)(7)	ODR = 400 Hz, normal mode	$ND_{ACC\text{-}NM}$		126		µg/√Hz
	ODR = 400 Hz, low-noise mode(1)	$ND_{ACC\text{-}LNM}$		99		µg/√Hz
Operating temperature range		Top	-40		+85	°C

1. Dynamic range is limited to ±4 *g* when in the low-noise mode.
2. Before board mount.
3. Post-board mount offset specifications are based on a 2-layer PCB design.
4. Evaluation only.
5. After post-board mount corrections for sensitivity, cross axis and offset. Refer to AN4399 for more information.
6. Self-test is only exercised along one direction for each sensitive axis.
7. Measured using earth's gravitational field (1 *g*) with the device oriented horizontally (+Z axis up) and stationary.

5.2 Electrical characteristics

Table 3. Electrical characteristics @ VDD = 2.5 V, VDDIO = 1.8 V T = 25°C unless otherwise noted.

Parameter	Test Conditions	Symbol	Min	Typ	Max	Unit
Supply voltage		VDD	1.95	2.5	3.6	V
Interface supply voltage		VDDIO	1.62	1.8	3.6	V
Low-power acceleration mode	ODR = 12.5 Hz	$Idd_{ACC-LPM}$		8		µA
	ODR = 100 Hz			35		
	ODR = 400 Hz			130		
Normal acceleration mode	ODR = 50 Hz	Idd_{ACC-NM}		35		µA
	ODR = 200 Hz			130		
	ODR = 800 Hz			240		
Current during boot sequence, 0.9 mS max duration using recommended regulator bypass capacitor	VDD = 2.5 V	Idd_{BOOT}			3	mA
Value of capacitor on BYP pin	-40°C to 85°C	C_{BYP}	75	100	470	nF
Standby mode current @ 25°C	Standby mode	Idd_{STBY}		2		µA
Standby mode current over-temperature range	Standby mode	Idd_{STBY}			10	µA
Digital high-level input voltage RST pin		VIH_{RST}	1.04			V
Digital low-level input voltage RST pin		VIL_{RST}			0.68	V
Digital high-level input voltage SCL, SDA, SA0, SA1		VIH	0.75*VDDIO			V
Digital low-level input voltage SCL, SDA, SA0, SA1		VIL			0.3*VDDIO	V
High-level output voltage INT1, INT2	I_O = 500 µA	VOH	0.9*VDDIO			V
Low-level output voltage INT1, INT2	I_O = 500 µA	VOL			0.1*VDDIO	V
Low-level output voltage SDA	I_O = 500 µA	VOL_{SDA}			0.1*VDDIO	V
SCL, SDA pin leakage	25°C			1.0		nA
	-40°C to 85°C			4.0		
SCL, SDA pin capacitance				3		pf
VDD rise time			0.001		1000	ms
Boot time(1)		T_{BOOT}			1000	µs
Turn-on time 1(2)		$T_{POR \rightarrow ACT}$		2/ODR + 2		ms
Turn-on time 2(3)		$T_{STBY \rightarrow ACT}$		2/ODR + 1		ms
Operating temperature range		T_{OP}	-40		+85	°C

1. Time from VDDIO on and VDD > VDD min until I^2C/SPI interface ready for operation.
2. Time to obtain valid data from power-down mode to Active mode.
3. Time to obtain valid data from Standby mode to Active mode.

5.3 Absolute maximum ratings

Stresses above those listed as “absolute maximum ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device under these conditions is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

Table 5. Maximum ratings

Rating	Symbol	Value	Unit
Maximum acceleration (all axes, 100 µs)	g_{max}	5,000	*g*
Supply voltage, IO voltage	VDD_{max}/ $VDDIO_{max}$	-0.3 to +3.6	V
Input voltage on any control pin (SA0/MISO, SA1/CS_B, SCL/SCLK, SDA/MOSI, RST)	VIN_{max}	-0.3 to VDDIO + 0.3	V
Drop-Test height	D_{drop}	1.8	m
Storage temperature range	T_{STG}	-40 to +125	°C

Table 6. ESD and latchup protection characteristics

Rating	Symbol	Value	Unit
Human Body Model	HBM	±2000	V
Machine Model	MM	±200	V
Charge Device Model	CDM	±500	V
Latchup current at T = 85°C	I_{LU}	±100	mA

This device is sensitive to mechanical shock. Improper handling can cause permanent damage to the part or cause the part to otherwise fail.

This device is sensitive to ESD, improper handling can cause permanent damage to the part.

6 Digital Interfaces

6.1 I²C interface characteristics

Table 7. I²C slave timing values(1)

Parameter	Symbol	I²C Fast Mode		Unit
		Min	Max	
SCL Clock Frequency	f_{SCL}	0	400	kHz
Bus Free Time between STOP and START condition	t_{BUF}	1.3		µs
(Repeated) START Hold Time	$t_{HD;STA}$	0.6		µs
(Repeated) START Setup Time	$t_{SU;STA}$	0.6		µs
STOP Condition Setup Time	$t_{SU;STO}$	0.6		µs
SDA Data Hold Time	$t_{HD;DAT}$	0.05	0.9(2)	µs
SDA Valid Time(3)	$t_{VD;DAT}$		0.9(2)	µs
SDA Valid Acknowledge Time(4)	$t_{VD;ACK}$		0.9(2)	µs
SDA Setup Time	$t_{SU;DAT}$	100		ns
SCL Clock Low Time	t_{LOW}	1.3		µs
SCL Clock High Time	t_{HIGH}	0.6		µs
SDA and SCL Rise Time	t_r	20 + 0.1 C_b(5)	300	ns
SDA and SCL Fall Time	t_f	20 + 0.1 C_b(5)	300	ns
Pulse width of spikes on SDA and SCL that must be suppressed by internal input filter	t_{SP}	0	50	ns

1. All values referred to VIH (min) and VIL (max) levels.
2. This device does not stretch the LOW period (t_{LOW}) of the SCL signal.
3. $t_{VD;DAT}$ = time for Data signal from SCL LOW to SDA output.
4. $t_{VD;ACK}$ = time for Acknowledgement signal from SCL LOW to SDA output (HIGH or LOW, depending on which one is worse).
5. C_b = total capacitance of one bus line in pF.

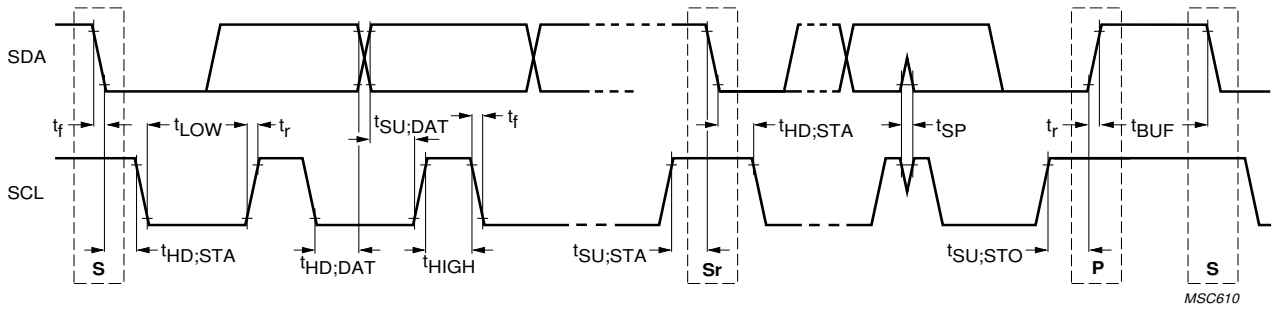


Figure 5. I²C slave timing diagram

6.1.1 General I²C operation

There are two signals associated with the I²C bus: the Serial Clock Line (SCL) and the Serial Data line (SDA). The latter is a bidirectional line used for sending and receiving the data to/from the interface. External pullup resistors connected to VDDIO are required for SDA and SCL. When the bus is free both the lines are high. The I²C interface is compliant with fast mode (400 kHz), and normal mode (100 kHz) I²C standards. Operation at frequencies higher than 400 kHz is possible, but depends on several factors including the pullup resistor values, and total bus capacitance (trace + device capacitance). See Table 8 for more information.

A transaction on the bus is started through a start condition (ST) signal, which is defined as a HIGH-to-LOW transition on the data line while the SCL line is held HIGH. After the ST signal has been transmitted by the master, the bus is considered busy. The next byte of data transmitted contains the slave address in the first seven bits, and the eighth bit, the read/write bit, indicates whether the master is receiving data from the slave or transmitting data to the slave. When an address is sent, each device in the system compares the first seven bits after the ST condition with its own address. If they match, the device considers itself addressed by the master. The 9th clock pulse, following the slave address byte (and each subsequent byte) is the acknowledge (ACK). The transmitter must release the SDA line during the ACK period. The receiver must then pull the data line low so that it remains stable low during the high period of the acknowledge clock period.

The number of bytes per transfer is unlimited. If a receiver can't receive another complete byte of data until it has performed some other function, it can hold the clock line, SCL low to force the transmitter into a wait state. Data transfer only continues when the receiver is ready for another byte and releases the data line. This delay action is called clock stretching. Not all receiver devices support clock stretching. Not all master devices recognize clock stretching. This part does not use clock stretching.

A low to high transition on the SDA line while the SCL line is high is defined as a stop condition (SP) signal. A write or burst write is always terminated by the master issuing the SP signal. A master should properly terminate a read by not acknowledging a byte at the appropriate time in the protocol. A master may also issue a repeated start signal (SR) during a transfer

The slave addresses that may be assigned to the FXLS8471Q part are 0x1C, 0x1D, 0x1E, or 0x1F. The selection is made through the logic level of the SA1 and SA0 inputs.

Table 8. I²C slave address

SA1	SA0	Slave address
0	0	0x1E
0	1	0x1D
1	0	0x1C
1	1	0x1F

6.1.2 I²C Read/Write operations

Single-byte read

The master (or MCU) transmits a start condition (ST) to the FXLS8471Q, followed by the slave address, with the R/W bit set to “0” for a write, and the FXLS8471Q sends an acknowledgement. Then the master (or MCU) transmits the address of the register to read and the FXLS8471Q sends an acknowledgement. The master (or MCU) transmits a repeated start condition (SR), followed by the slave address with the R/W bit set to “1” for a read from the previously selected register. The FXLS8471Q then acknowledges and transmits the data from the requested register. The master does not acknowledge (NAK) the transmitted data, but transmits a stop condition to end the data transfer.

Multiple-byte read

When performing a multi-byte or “burst” read, the FXLS8471Q automatically increments the register address read pointer after a read command is received. Therefore, after following the steps of a single-byte read, multiple bytes of data can be read from sequential registers after each FXLS8471Q acknowledgment (AK) is received until a no acknowledge (NAK) occurs from the master followed by a stop condition (SP) signaling an end of transmission.

Single-byte write

To start a write command, the master transmits a start condition (ST) to the FXLS8471Q, followed by the slave address with the R/W bit set to “0” for a write, and the FXLS8471Q sends an acknowledgement. Then the master (or MCU) transmits the address of the register to write to, and the FXLS8471Q sends an acknowledgement. Then the master (or MCU) transmits the 8-bit data to write to the designated register and the FXLS8471Q sends an acknowledgement that it has received the data. Since this transmission is complete, the master transmits a stop condition (SP) to end the data transfer. The data sent to the FXLS8471Q is now stored in the appropriate register.

Multiple-byte write

The FXLS8471Q automatically increments the register address write pointer after a write command is received. Therefore, after following the steps of a single-byte write, multiple bytes of data can be written to sequential registers after each FXLS8471Q acknowledgment (ACK) is received.

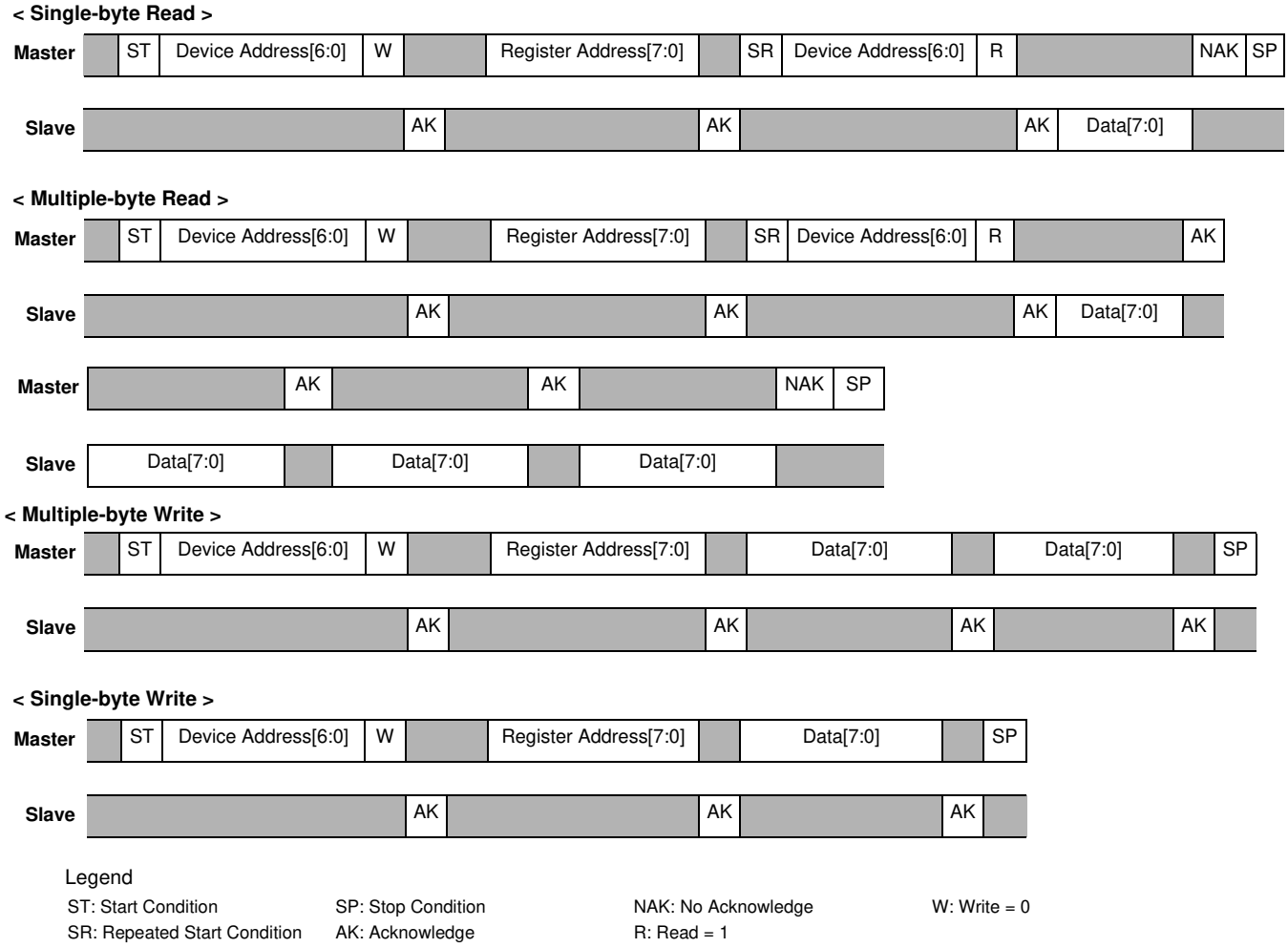


Figure 6. I²C timing diagram

6.2 SPI Interface characteristics

SPI interface is a classical master/slave serial port. The FXLS8471Q is always considered as the slave and thus is never initiating the communication.

Table 9 and Figure 7 describe the timing requirements for the SPI system.

Table 9. SPI timing

Function	Symbol	Min	Max	Unit
Operating Frequency	O*f*	—	1	MHz
SCLK Period	tSCLK	1000	—	ns
SCLK High time	tCLKH	500	—	ns
SCLK Low time	tCLKL	500	—	ns
CS_B lead time	tSCS	65	—	ns
CS_B lag time	tHCS	65	—	ns
MOSI data setup time	tSET	25	—	ns
MOSI data hold time	tHOLD	75	—	ns
MISO data valid (after SCLK low edge)	tDDLY	—	500	ns
Width CS High	tWCS	100	—	ns

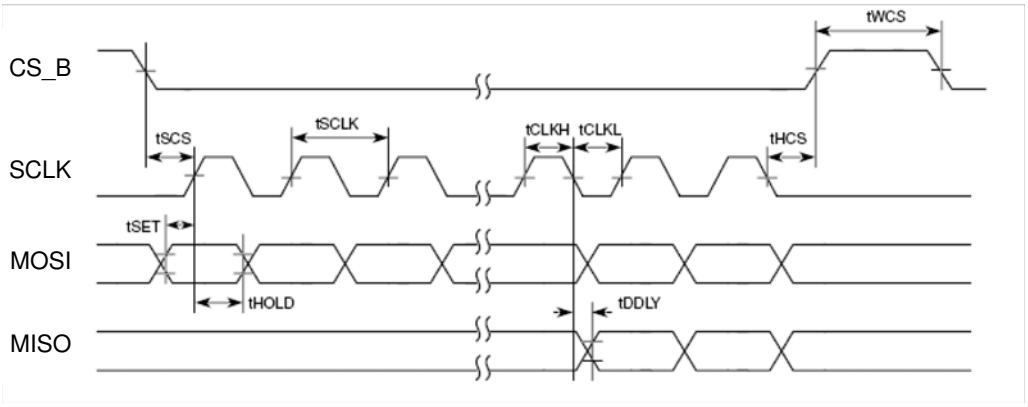


Figure 7. SPI Timing Diagram

6.2.1 General SPI operation

The CS_B pin is driven low at the start of a SPI transaction, held low for the duration of the transaction, and driven high after the transaction is complete. During a transaction the master toggles the SPI clock (SCLK) and transmits data on the MOSI pin.

A write operation is initiated by transmitting a 1 for the R/W bit. Then the 8-bit register address, ADDR[7:0] is encoded in the first and second serialized bytes. Data to be written starts in the third serialized byte. The order of the bits is as follows:

Byte 0: R/W,ADDR[6],ADDR[5],ADDR[4],ADDR[3],ADDR[2],ADDR[1],ADDR[0],

Byte 1: ADDR[7],X,X,X,X,X,X,X,

Byte 2: DATA[7],DATA[6],DATA[5],DATA[4],DATA[3],DATA[2],DATA[1],DATA[0].

Multiple bytes of DATA may be transmitted. The X indicates a bit that is ignored by the part. The register address is auto-incremented so that the next clock edges will latch the data for the next register. When desired, the rising edge on CS_B stops the SPI communication.

The FXLS8471Q SPI configuration is as follows:

- Polarity: rising/falling
- Phase: sample/setup
- Order: MSB first

Data is sampled during the rising edge of SCLK and set up during the falling edge of SCLK.

6.2.2 SPI READ/WRITE operations

A READ operation is initiated by transmitting a 0 for the R/W bit. Then the 8-bit register address, ADDR[7:0] is encoded in the first and second serialized bytes. Subsequent bits are ignored by the part. The read data is deserialized from the MISO pin.

Similarly a WRITE operation is initiated by transmitting a 1 for the R/W bit. After the first and second serialized bytes multiple-data bytes can be transmitted into consecutive registers, starting from the indicated register address in ADDR[7:0].

An SPI transaction is started by asserting the CS_B pin (high-to-low transition), and ended by deasserting the CS_B pin (low-to-high transition).

R/W bit followed by ADDR [6:0]	ADDR[7] followed by 7 “don’t care” bits	Data0*	Data1	—	Datan

* Data bytes must be transmitted to the slave (FXLS8471Q) using the MOSI pin by the master when R/W = 1. Data bytes will be transmitted by the slave (FXLS8471Q) to the master using the MISO pin when R/W = 0. The first 2 bytes are always transmitted by the master using the MOSI pin. That is, a transaction is always initiated by master. 1

Figure 8. SPI single-burst READ/WRITE transaction diagram

The registers embedded inside FXLS8471Q are accessed through either an I^2C, or a SPI serial interface. To enable either interface the VDDIO line must be connected to the interface supply voltage. If VDD is not present and VDDIO is present FXLS8471Q is in shutdown mode and communications on the interface are ignored. If VDDIO is held high, VDD can be powered off and the communications pins will be in a high impedance state. This will allow communications to continue on the bus with other devices.

Table 10. Serial interface pin descriptions

Pin Name	Pin Description
VDDIO	Digital interface power
SA1/CS_B	I^2C second least significant bit of device address/SPI chip select
SCL/SCLK	I^2C/SPI serial clock
SDA/MOSI	I^2C serial data/SPI master serial data out slave serial data in
SA0/MISO	I^2C least significant bit of the device address/SPI master serial data in slave out

6.2.3 I^2C/SPI auto detection

Table 11. I^2C/SPI auto detection

SA0	Slave address
GND	I^2C
VDDIO	I^2C
Floating	SPI

FXLS8471Q employs an interface mode, auto-detection circuit that will select either I^2C or SPI interface mode based on the state of the SA0 pin during power up or when exiting reset. Once set for I^2C or SPI operation, the device will remain in I^2C or SPI mode until the device is reset or powered down and the auto-detection process is repeated. Please note that when SPI interface mode is desired, care must be taken to ensure that no other slave device drives the common SA0/MISO pin during the 1 ms period after a hard or soft reset or powerup event.

6.2.4 Power supply sequencing and I^2C/SPI mode auto-detection

FXLS8471Q does not have any specific power supply sequencing requirements between VDD and VDDIO voltage supplies to ensure normal operation. To ensure correct operation of the I^2C/SPI auto-detection function, VDDIO should be applied before or at the same time as VDD. If this order cannot be maintained, the user should either toggle the RST line or power cycle the VDD rail in order to force the auto-detect function to restart and correctly identify the desired interface. FXLS8471Q will indicate completion of the reset sequence by toggling the INT1 pin from logic high to low to high over a 500 ns period. If the INT1 pin was already low prior to the reset event, it will only go high.

7 Modes of Operation

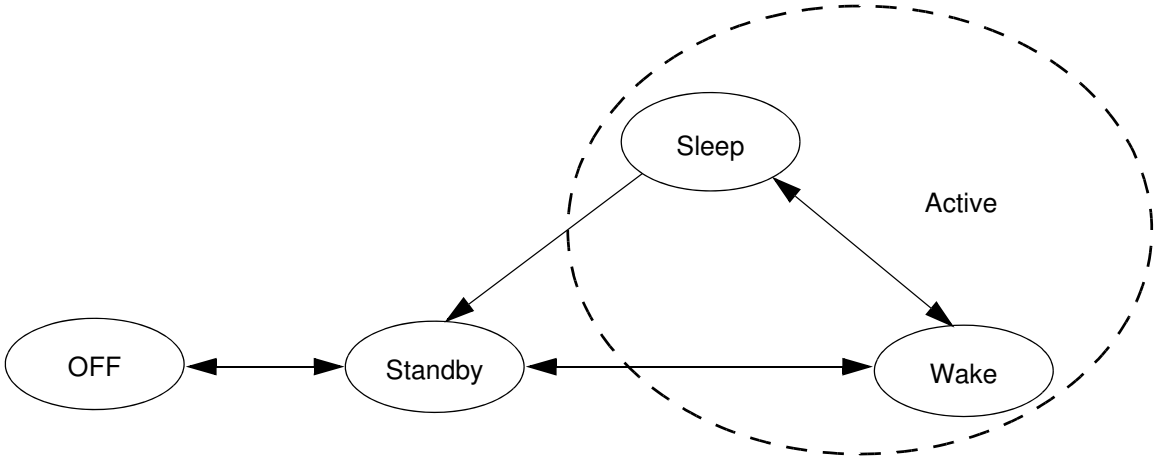


Figure 9. FXLS8471Q power mode transition diagram

Table 12. Mode of operation description

Mode	I^2C/SPI Bus state	VDD	VDDIO	Function description
OFF	Powered down	<1.8 V	VDDIO can be > VDD	The device is powered off. All analog and digital blocks are shutdown. I^2C bus inhibited.
Standby	I^2C/SPI communication with FXLS8471Q is possible	ON	VDDIO = High VDD = High Active bit is cleared	Only digital blocks are enabled. Analog subsystem is disabled. Internal clocks disabled.
Active (Wake/Sleep)	I^2C/SPI communication with FXLS8471Q is possible	ON	VDDIO = High VDD = High Active bit is set	All blocks are enabled (digital and analog).

All register contents are preserved when transitioning from Active to Standby mode, but some registers are reset when transitioning from Standby to Active. These registers are noted in Table 13, “Register Address Map,” on page 25. The Sleep and Wake modes are active modes. For more information on how to use the Sleep and Wake modes and configuring the device to transition between them, please refer to Section 8, “Embedded Functionality” or Freescale application note AN4074.

8 Embedded Functionality

FXLS8471Q is a low-power, digital output 3-axis acceleration sensor with both I^2C and SPI interface options. Extensive embedded functionality is provided to detect inertial events at low power, with the ability to notify the host processor via either of the two programmable interrupt pins. The embedded functionality includes:

- 8-bit or 14-bit accelerometer data with an option for high-pass filtered output data
- Four different oversampling options for the output data. The oversampling settings allow the end user to optimize the resolution versus power consumption trade-off in a given application.
- A low-noise accelerometer mode that functions independently of the oversampling modes for even higher resolution
- Low-power auto-wake/sleep function for conserving power in portable battery powered applications
- Accelerometer pulse detection circuit which can be used to detect directional single and double taps
- Accelerometer directional motion and freefall event detection with programmable threshold and debounce time
- Acceleration transient detection with programmable threshold and debounce time. Transient detection can employ either a high-pass filter or use the difference between reference and current sample values.
- Orientation detection with programmable hysteresis for smooth transitions between portrait and landscape orientations
- Accelerometer vector-magnitude change event detection with programmable reference, threshold, and debounce time values

Many different configurations of the above functions are possible to suit the needs of the end application. Separate application notes are available to further explain the different configuration settings and potential use cases.

8.1 Factory calibration

FXLS8471Q is factory calibrated for sensitivity and offset on each axis. The trim values are stored in Non-Volatile Memory (NVM). On startup, the trim parameters are read from NVM and applied to the internal compensation circuitry. After mounting the device to the PCB, the user may further adjust the accelerometer offsets through the OFF_X/Y/Z registers. For more information on accelerometer calibration, refer to Freescale application note AN4069.

8.2 8-bit or 14-bit data

The measured acceleration data is stored in the OUT_X_MSB, OUT_X_LSB, OUT_Y_MSB, OUT_Y_LSB, OUT_Z_MSB, and OUT_Z_LSB registers as 2's complement 14-bit numbers. The most significant 8-bits of each axis are stored in the OUT_X, Y, Z_MSB registers, so applications needing only 8-bit results simply read these three registers and ignore the OUT_X,Y, Z_LSB registers. To do this, the *f_read* mode bit in CTRL_REG1 must be set.

When the full-scale range is set to 2 *g*, the measurement range is -2 *g* to +1.999 *g*, and each count corresponds to 0.244 mg at ±14-bits resolution. When the full-scale is set to 8 *g*, the measurement range is -8 *g* to +7.996 *g*, and each count corresponds to 0.976 mg. The resolution is reduced by a factor of 64 if only the 8-bit results are used (CTRL_REG1[*f_read*] = 1). For further information on the different data formats and modes, please refer to Freescale application note AN4076.

8.3 Low-power modes versus high-resolution modes

FXLS8471Q can be optimized for lower power or higher resolution of the accelerometer output data. High resolution is achieved by setting the lnoise bit in register 0x2A. This improves the resolution (by lowering the noise), but be aware that the dynamic range becomes fixed at ±4 *g* when this bit is set. This will affect all internal embedded functions (scaling of thresholds, etc.) and reduce noise. Another method for improving the resolution of the data is through oversampling. One of the oversampling schemes of the output data can be activated when CTRL_REG2[*mods*] = 2'b10 which will improve the resolution of the output data without affecting the internal embedded functions or fixing the dynamic range.

There is a trade-off between low power and high resolution. Low power can be achieved when the oversampling rate is reduced. When CTRL_REG2[*mods*] = 2'b10, the lowest power is achieved, at the expense of higher noise. In general, the lower the selected ODR and OSR, the lower the power consumption. For more information on how to configure the device in low-power or high-resolution modes and understand the benefits and trade-offs, please refer to Freescale application note AN4075.

8.4 Auto-Wake/Sleep mode

FXLS8471Q can be configured to transition between sample rates (with their respective current consumptions) based on the status of the embedded interrupt event generators in the device. The advantage of using the Auto-Wake/Sleep is that the system can automatically transition to a higher sample rate (higher current consumption) when needed but spends the majority of the time in the Sleep mode (lower current) when the device does not require higher sampling rates. Auto-Wake refers to the device being triggered by one of the interrupt event functions to transition to a higher sample rate. This may also interrupt the processor to transition from a sleep mode to a higher power mode.

Sleep mode occurs when none of the enabled interrupt event functions has detected an interrupt within the user-defined, time-out period. The device will then transition to the specified lower sample rate. It may also alert the processor to go into a lower power mode to save power during this period of inactivity. Please refer to AN4074 for more detailed information on configuring the Auto-Wake/Sleep function.

8.5 Freefall and Motion event detection

FXLS8471Q integrates a programmable threshold based acceleration detection function capable of detecting either motion or freefall events depending upon the configuration. For further details and examples on using the embedded freefall and motion detection functions, please refer to Freescale application note AN4070.

8.5.1 Freefall detection

The detection of “Freefall” involves the monitoring of the X, Y, and Z axes for the condition where the acceleration magnitude is **below** a user-specified threshold for a user-definable amount of time. Typically, the usable threshold ranges are between ±100 mg and ±500 mg.

8.5.2 Motion detection

Motion detection is often used to alert the main processor that the device is currently in use. When the acceleration exceeds a set threshold for a set amount of time, the motion interrupt is asserted. A motion can be a fast moving shake or a slow moving tilt. This will depend on the threshold and timing values configured for the event. The motion detection function can analyze static acceleration changes or faster jolts. The timing value is set by a configurable debounce counter. The debounce counter acts like a filter to indicate whether the condition exists for longer than a set amount of time (that is, 100 ms or longer). There is also directional data available in the source register to detect the direction of the motion that generated the interrupt. This is useful for applications such as directional shake or flick detection, and can also assist gesture detection algorithms by indicating that a motion gesture has started.

8.6 Transient detection

FXLS8471Q integrates an acceleration transient detection function that incorporates a high-pass filter. Acceleration data goes through the high-pass filter, eliminating the DC tilt offset and low frequency acceleration changes. The high-pass filter cutoff can be set by the user to four different frequencies which are dependent on the selected Output Data Rate (ODR). A higher cutoff frequency ensures that DC and slowly changing acceleration data will be filtered out, allowing only the higher frequencies to pass. The transient detection feature can be used in the same manner as the motion detection by bypassing the high-pass filter. There is an option in the configuration register to do this. This adds more flexibility to cover the various customer use cases.

Many applications use the accelerometer’s static acceleration readings (that is, tilt) which measure the change in acceleration due to gravity only. These functions benefit from acceleration data being filtered with a low-pass filter where high-frequency data is considered noise. However, there are many functions where the accelerometer must analyze dynamic acceleration. Functions such as tap, flick, shake and step counting are based on the analysis of the change in the dynamic acceleration. The transient detection function can be routed to either interrupt pin through bit 5 in CTRL_REG5 register (0x2E). Registers 0x1D – 0x20 are used for configuring the transient detection function. The source register contains directional data to determine the direction of the transient acceleration, either positive or negative. For further information of the embedded transient detection function along with specific application examples and recommended configuration settings, refer to Freescale application note AN4461.

8.7 Pulse detection

FXLS8471Q has embedded single/double and directional pulse detection. This function employs several timers for programming the pulse width time and the latency between pulses. The detection thresholds are independently programmable for each axis. The acceleration data input to the pulse detection circuit can be put through both high and low-pass filters, allowing for greater flexibility in discriminating between pulse and tap events. The PULSE_SRC register provides information on the axis, direction (polarity), and single/double event status for the detected pulse or tap. For more information on how to configure the device for pulse detection, please refer to Freescale application note AN4072.

8.8 Orientation detection

FXLS8471Q has an embedded orientation detection algorithm with the ability to detect all six orientations. The transition angles and hysteresis are programmable, allowing for a smooth transition between portrait and landscape orientations.

The angle at which the device no longer detects the orientation change is referred to as the “Z-lockout angle”. The device operates down to 29° from the flat position. All angles are accurate to ±2°.

For further information on the orientation detection function refer to Freescale application note, AN4068.

8.9 Acceleration Vector-Magnitude detection

FXLS8471Q incorporates an acceleration vector-magnitude change detection block that can be configured to generate an interrupt when the acceleration magnitude exceeds a preset threshold for a programmed debounce time. The function can be configured to operate in absolute or relative modes, and can also act as a wake to sleep/sleep to wake source. This function is useful for detecting acceleration transients when operated in absolute mode, or for detecting changes in orientation when operated in relative mode, refer to Freescale application note AN4692.

9 Register Map

Table 13. Register Address Map

Name	Type	Register Address	Auto-Increment Address				Default Hex Value	Comment	
			STATUS[*f_mode*] = 00, CTRL_REG1[*f_read*] = 0	STATUS[*f_mode*] > 00, CTRL_REG1[*f_read*] = 0	STATUS[*f_mode*] = 00, CTRL_REG1[*f_read*] = 1	STATUS[*f_mode*] > 00, CTRL_REG1[*f_read*] = 1			
STATUS(1)(2)	R	0x00	0x01				0x00	Real-time, data-ready status or FIFO status (DR_STATUS or F_STATUS)	
OUT_X_MSB(1)(2)	R	0x01	0x02	0x01	0x03	0x01	Data	[7:0] are 8 MSBs of 14-bit sample.	Root pointer to XYZ FIFO data.
OUT_X_LSB(1)(2)	R	0x02	0x03		0x00		Data	[7:2] are 6 LSBs of 14-bit sample	
OUT_Y_MSB(1)(2)	R	0x03	0x04		0x05	0x00	Data	[7:0] are 8 MSBs of 14-bit sample	
OUT_Y_LSB(1)(2)	R	0x04	0x05		0x00		Data	[7:2] are 6 LSBs of 14-bit sample	
OUT_Z_MSB(1)(2)	R	0x05	0x06		0x00		Data	[7:0] are 8 MSBs of 14-bit sample	
OUT_Z_LSB(1)(2)	R	0x06	0x00		0x00		Data	[7:2] are 6 LSBs of 14-bit sample	
Reserved	—	0x07-0x08	—				—	Reserved, do not modify	
F_SETUP(1)(3)	R/W	0x09	0x0A				0x00	FIFO setup	
TRIG_CFG	R/W	0x0A	0x0B				0x00	FIFO event trigger configuration register	
SYSMOD(1)(2)	R	0x0B	0x0C				Output	Current system mode	
INT_SOURCE(1)(2)	R	0x0C	0x0D				Output	Interrupt status	
WHO_AM_I(1)	R	0x0D	0x0E				0x6A	Device ID	
XYZ_DATA_CFG(1)(4)	R/W	0x0E	0x0F				0x00	Acceleration dynamic range and filter enable settings	
HP_FILTER_CUTOFF(1)(4)	R/W	0x0F	0x10				0x00	Pulse detection high-pass and low-pass filter enable bits. High-pass filter cutoff frequency selection	
PL_STATUS(1)(2)	R	0x10	0x11				0x00	Landscape/Portrait orientation status	
PL_CFG(1)(4)	R/W	0x11	0x12				0x83	Landscape/Portrait configuration.	
PL_COUNT(1)(3)	R/W	0x12	0x13				0x00	Landscape/Portrait debounce counter	
PL_BF_ZCOMP(1)(4)	R/W	0x13	0x14				0x00	Back/Front Trip angle threshold	
PL_THS_REG(1)(4)	R/W	0x14	0x15				0x1A	Portrait to Landscape Trip Threshold angle and hysteresis settings	
A_FFMT_CFG(1)(4)	R/W	0x15	0x16				0x00	Freefall/Motion function configuration	
A_FFMT_SRC(1)(2)	R	0x16	0x17				0x00	Freefall/Motion event source register	
A_FFMT_THS(1)(3)	R/W	0x17	0x18				0x00	Freefall/Motion threshold register	