Evaluating the ADXL380 Low Noise, Low Power, Wide Bandwidth, 3-Axis MEMS Accelerometer

FEATURES

- Plug-and-play rapid evaluation system of the ADXL380
- PC-based evaluation software with user friendly GUI
- Self-contained data capture with battery and SD memory card
- On-board Bluetooth® low energy (BLE) module

EVALUATION BOARD KIT CONTENTS

- EVAL-ADXL380-BB-1Z, EVAL-ADXL380-BB-2Z, EVAL-ADXL380-EB-1Z, or EVAL-ADXL380-EB-2Z evaluation board
- EVAL-ADIS-MCBZ controller board, USB cable, and flat ribbon connector cable

DOCUMENTS NEEDED

- ADXL380 data sheet
- EVAL-ADIS-MCBZ user guide

SOFTWARE NEEDED

- **EVAL-ADIS-MCBZ Software**

GENERAL DESCRIPTION

This user guide provides an overview of using the EVAL-ADXL380-BB-1Z (breakout board for the SPI variant of the ADXL380), EVAL-ADXL380-BB-2Z (breakboard for the I^2C variant of the ADXL380), EVAL-ADXL380-EB-1Z (evaluation board for the SPI variant of the ADXL380), and EVAL-ADXL380-EB-2Z (evaluation board for the I^2C variant of the ADXL380) as a PC-based solution for evaluating the ADXL380 microelectromechanical systems (MEMS) accelerometer. These evaluation boards use the motion capture board (MCB) from the EVAL-ADIS-MCBZ platform for an easy communication and data collection using the graphical user interface (GUI). Go to the EVAL-ADIS-CBZ web page to download the software and its user guide, which details the installation process and the software features. This user guide gives an overview of using the ADXL380 evaluation boards available, detailing their functionality with provided examples where relevant.

Figure 1 shows the required hardware components for the ADXL380 evaluation system. The EVAL-ADIS-MCBZ is the controller board that provides a means of communication with the EVAL-ADXL380-BB-1Z, EVAL-ADXL380-BB-2Z, EVAL-ADXL380-EB-1Z, or EVAL-ADXL380-EB-2Z evaluation board from the PC.

For full details on the ADXL380, see the ADXL380 data sheet, which should be consulted in conjunction with this user guide when using the EVAL-ADXL380-BB-1Z, EVAL-ADXL380-BB-2Z, EVAL-ADXL380-EB-1Z, or EVAL-ADXL380-EB-2Z evaluation board.

REQUIRED HARDWARE

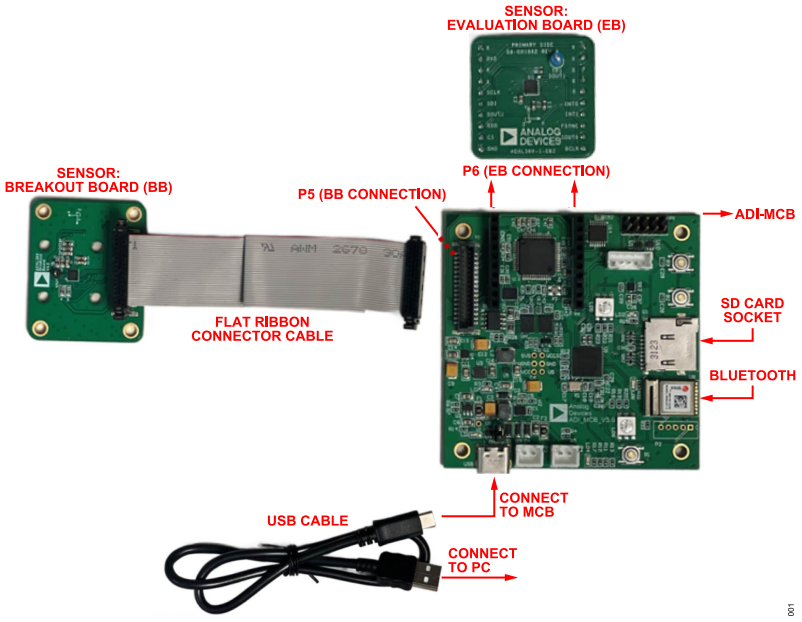

Figure 1. Required Hardware

TABLE OF CONTENTS

Features..... 1
Evaluation Board Kit Contents.....1
Documents Needed.....1
Software Needed.....1
General Description.....1
Required Hardware..... 1
Setting Up the Evaluation System.....3
Hardware Configuration..... 3
Software Installation.....3
Graphical User Interface.....4
Getting Started..... 4
Register Map.....5
Record Data..... 10
Evaluation System Features..... 10
Evaluation Board Schematics.....11
Notes..... 13

REVISION HISTORY

8/2024—Revision 0: Initial Version

SETTING UP THE EVALUATION SYSTEM

The Hardware Configuration and Software Installation sections describe the hardware and software installation process for the EVAL-ADXL380-BB-1Z, EVAL-ADXL380-BB-2Z, EVAL-ADXL380-EB-1Z, or EVAL-ADXL380-EB-2Z.

HARDWARE CONFIGURATION

To configure the hardware, follow these steps:

1. Connect the EVAL-ADXL380-BB-1Z or EVAL-ADXL380-BB-2Z (breakout board) or the EVAL-ADXL380-EB-1Z or EVAL-ADXL380-EB-2Z (evaluation board) to the EVAL-ADIS-MCBZ.. The breakout board connector or the evaluation board connector are located on the top left of the EVAL-ADIS-MCBZ. The evaluation board connector has two 10-pin, single-row connectors. The breakout board connects to the EVAL-ADIS-MCBZ by using a flat-ribbon connector cable and the 32-pin Harwin M55-7003242R connector.
2. Connect the EVAL-ADIS-MCBZ to the PC by using the provided USB cable. Figure 2 shows the hardware setup with the breakout board.

Figure 2. Hardware Setup

SOFTWARE INSTALLATION

Go to the EVAL-ADIS-MCB product page and download the EVAL-ADIS_MCB software package into a dedicated folder on the PC. Extract the files and then double-click on the **EVAL-ADIS-MCB.exe** file (see Figure 3) to download the EVAL-ADIS-MCB software. For more details about the GUI elements, see the EVAL-ADIS-MCB user guide on the EVAL-ADIS-MCB product page.

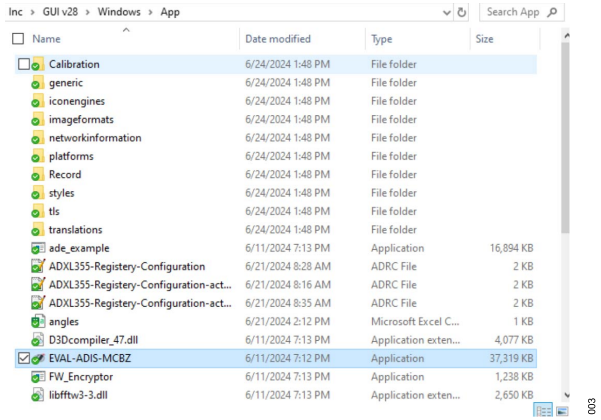

Figure 3. Software App

GRAPHICAL USER INTERFACE

GETTING STARTED

The demonstration platform, using the EVAL-ADIS-MCBZ hardware and software setup, provides an intuitive interface to easily communicate with and collect data from the ADXL380 in different operating modes. After connecting the EVAL-ADXL380-EB-1Z, EVAL-ADXL380-EB-2Z, EVAL-ADXL380-BB-1Z, or EVAL-ADXL380-BB-2Z to the EVAL-ADIS-MCBZ, and when the board is recognized by the software, take the following steps to communicate with the sensor:

- Got to the **Basic Configuration** tab, located in the software on the top right side of the GUI.
- Go to the **Operating Mode** dropdown menu to change the operating mode from standby to the desired measurement mode to start data collection.
- Adjust the acceleration range (**Accel Range (*g*)**) and filter details dropdown menus to the desired values.
- Use the tab at the bottom of the plot to change the unit, adjust the range, and enable the desired axis of acceleration and temperature to display in real time.

Figure 4 shows some features of the **Basic Configuration** tab.

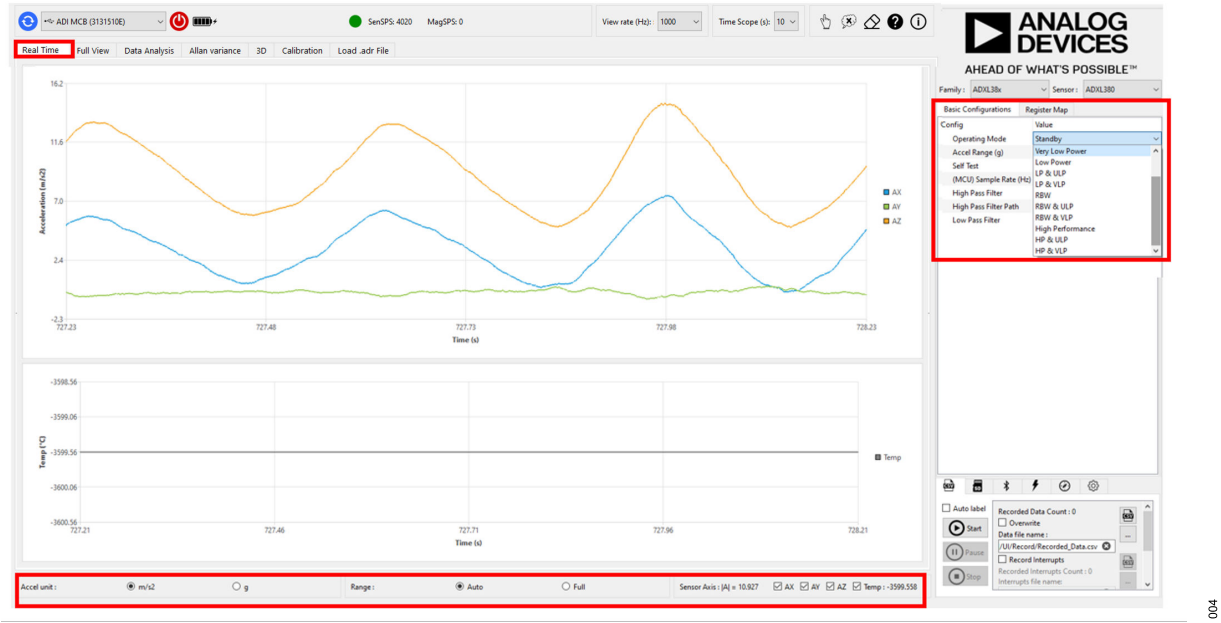

Figure 4. Real-Time Data Capturing (Basic Configuration)

GRAPHICAL USER INTERFACE

REGISTER MAP

Besides the **Basic Configuration** tab, which provides basic and limited features to communicate with the sensor, the **Register Map** tab allows communication with all the registers of the sensor. The register map, displayed on the right side of the GUI, shows all of the available registers as detailed further in the ADXL380 data sheet. The **Register Map** tab displays a tree view of the digital sensor register map of the ADXL380. This map includes detailed information such as register address, register names, bit names, bit values, reset values, and access type (read, write, or both). The registers can be modified for specific purposes, such as activity and inactivity detection and tap detection.

Users can display register and reset values in either hexadecimal or decimal format by selecting the desired option in the bottom right corner of the **Register Map** tab or by entering the register values in either format; the GUI automatically detects these formats. Additionally, users can update the register map by using the selectable **Register Update Rate** tab.

To change the sensor register values based on the desired application, modify the register values as follows:

1. Click the **LOCK** button to halt register map updates.
2. Double-click the value of the corresponding register to edit it.
3. Click **Write Map**.

Note that changes to the registers are applied, and the current register value displays after adjusting the **Registry Update Rate** from **Never** to the desired rate.

To discard changes, click the **UNLOCK** button. In addition, save the current registry and load it for future use. The sensor register map configuration files are saved in **.adrc** format and can be loaded by dragging and dropping the corresponding file (see Figure 5).

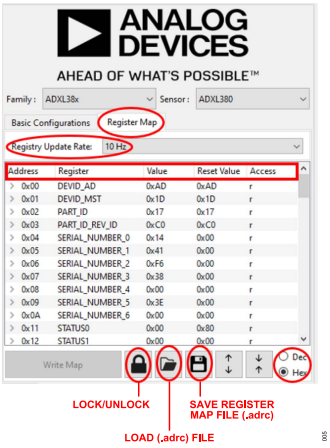

Figure 5. Register Map

In the following subsections, some examples of modifying the register map for different purposes are presented.

Start-Up Routine

The ADXL380 offers various operational modes for different applications and purposes. Select the **Operation Mode** dropdown menu in the **Basic Configuration** tab to change from standby mode to the desired mode in the **EVAL-ADIS-MCBZ Software**. The operational mode can also be selected in the **Register Map** tab.

Figure 6 shows changing the operational mode from **Standby** to **High Performance** in both the **Basic Configuration** tab by using the dropdown menu and the **Register Map** tab by writing 0x1C to Register 0x26 (OP_MODE). Refer to the Register 0x26 in the ADXL380 data sheet for more details on the different operation modes and registers.

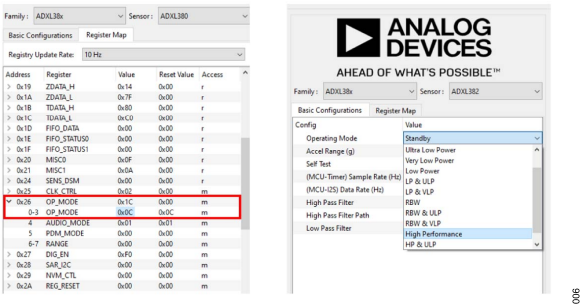

Figure 6. Operation Mode (High Performance)

FIFO Mode

The built-in FIFO can be enabled with the following example sequence:

1. Write 0x78 to Register 0x27 (DIG_EN).
   - Enables x, y, and z channels and enables FIFO mode. Data will not start accumulating in the FIFO until standby mode is exited.
2. Write 0x60 to Register 0x30 (FIFO_CFG0).
   - Enables FIFO channel ID and FIFO stream mode.
3. Write 0x0C to Register 0x31 (FIFO_CFG1).
   - Sets FIFO_SAMPLES (Register 0x30, Bit 0 and Register 0x31, Bits[7:0]) to 12. The FIFO_WATERMARK bit (Register 0x11, Bit 3) asserts when four full samples (X, Y, and Z) are stored in the FIFO.
   - This value can be changed to fit the application needs.
4. Write 0x08 to Register 0x2B (INT0_MAP0).
   - Maps the FIFO_WATERMAK bit (Register 0x11, Bit 3) to the INT0 pin.
5. Write 0x0C to Register 0x26 (OP_MODE).
   - Enables HP mode. Data now starts accumulating in the FIFO.

After entering HP mode, use the watermark interrupt to trigger reading data from the FIFO. Use a multibyte read of Register 0x1D (FIFO_DATA) to read the entire contents of the FIFO. Do not read more samples than are currently stored in the FIFO. Read

GRAPHICAL USER INTERFACE

FIFO_ENTRIES (Register 0x1E, Bits[7:0] and Register 0x1F, Bit 0) to determine the number of samples that are currently in the FIFO before each FIFO read.

To ensure that no samples are lost, it is important to finish reading from the FIFO and de-assert the chip select line before a new sample becomes available. Reading from the FIFO while a new sample becomes available results in that sample being lost. To avoid the sample being lost, ensure that the digital communication is fast enough (use a fast, serial peripheral interface (SPI) clock or high speed, inter-IC (I^2C)) to complete the multibyte read before a new sample becomes available.

Inter-IC Sound (I^2S) Mode

I^2S protocol for audio data is suitable for obtaining high speed, synchronous accelerometer data. Because latency is often important in audio applications, it is most common to set the system clock to 768 kHz as described in the Low Latency Mode section of the ADXL380 data sheet. This setting of the system clock requires setting up the clock divider (EXT_CLK_RATE, Register 0x25, Bits[7:4]) such that the BCLK divides down to 768 kHz.

See the example sequence that follows for a 3.072 MHz BCLK and 48 kHz FSYNC:

1. Write 0x00 to Register 0x32 (SPT_CFG0).
   - Sets the word width to 32 bits.
   - Configures the device to expect stereo FSYNC (50% duty cycle).
2. Write 0x88 to Register 0x33 (SPT_CFG1).
   - Sets time slots for x-axis and y-axis data-words.
   - Sets the output to repeat samples if ODR < FSYNC rate.
3. Write 0x5A to Register 0x34 (SPT_CFG2).
   - Sets time slots for z-axis data-word.
4. Write 0x80 to Register 0x35 (SYNC_CFG).
   - Sets BCLK and FSYNC source (external).
5. Optional: write 0x01 to Register 0x59 (S_{OUT0}) and 0x01 to Register 0x5A (MCLK).
   - Increases drive strength of the data output pins.
6. Write 0xC0 to Register 0x49 (TRIG_CFG).
   - Sets the internal decimation and filter bypasses to select the output data rate of 48 kHz.
7. Write 0x32 to Register 0x25 (CLK_CTRL).
   - Sets the external clock and sets the clock divider (EXT_CLK_RATE is divide by 4), which divides the 3.072 MHz BCLK down to the expected system clock (768 kHz).
8. Write 0x40 to Register 0x50 (FILTER).
   - Sets LPF and HPF settings and disables the EQ filter.
9. Write 0x74 to Register 0x27 (DIG_EN).
   - Enables x, y, and z channels and sets the DOUBLE_SPEED bit in the DIG_EN register (Register 0x27, Bit 2), which is required for low latency mode.
10. Write 0x1C to Register 0x26 (OP_MODE).
    - Enables HP mode and audio mode.

Time Division Multiplexing (TDM) Mode

Similar to the I^2S protocol, the ADXL380 supports TDM protocol communication. Because latency is often important in audio applications, it is most common to set the ADXL380 system clock to 768 kHz as described in the Low Latency Mode section of the ADXL380 data sheet, which requires setting up the clock divider (EXT_CLK_RATE, Register 0x25, Bits[7:4]) such that the BCLK divides down to 768 kHz.

See the example sequence that follows for a 6.144 MHz BCLK and 48 kHz FSYNC:

1. Write 0x08 to Register 0x32 (SPT_CFG0).
   - Sets the word width to 32 bits.
   - Configures the device to expect a sync pulse that is a single bit clock wide.
2. Write 0x88 to Register 0x33 (SPT_CFG1).
   - Sets time slots for x-axis and y-axis data-words.
   - Sets the output to repeat samples if ODR < FSYNC rate.
3. Write 0x1A to Register 0x34 (SPT_CFG2).
   - Sets time slots for z-axis data-word.
4. Write 0x80 to Register 0x35 (SYNC_CFG).
   - Sets BCLK and FSYNC source (external).
5. Optional: write 0x01 to Register 0x59 (S_{OUT0}) and 0x01 to Register 0x5A (MCLK).
   - Increases drive strength of the data output pins.
6. Write 0xC0 to Register 0x49 (TRIG_CFG).
   - Sets the internal decimation and filter bypasses to select the output data rate of 48 kHz.
7. Write 0x52 to Register 0x25 (CLK_CTRL).
   - Sets the external clock and sets the clock divider (EXT_CLK_RATE is divide by 8), which divides the 6.144 MHz BCLK down to the expected system clock (768 kHz).
8. Write 0x40 to Register 0x50 (FILTER).
   - Sets LPF and HPF settings and disables the EQ filter.
9. Write 0x74 to Register 0x27 (DIG_EN).
   - Enables x, y, and z channels and sets the DOUBLE_SPEED bit in the DIG_EN register (Register 0x27, Bit 2), which is required for low latency mode.
10. Write 0x1C to Register 0x26 (OP_MODE).
    - Enables HP mode and audio mode.

GRAPHICAL USER INTERFACE

Pulse Density Modulation (PDM) Mode

The PDM bypasses the digital filters and offers the best performance in terms of latency. Additional filtering and decimation may be required at the system level. See the example initialization sequence that follows:

1. Write 0x39 to Register 0x36 (PDM_CFG).
   - Configures x-data in Slot A, and y-data in Slot B (S_{OUT0} pin).
   - Configures z-data in Slot A (FSYNC pin).
2. Optional: write 0x01 to Register 0x59 (SOUT0) and 0x01 to Register 0x5C (FSYNC).
   - Increases drive strength of the data output pins.
3. Write 0x02 to Register 0x25 (CLK_CTRL).
   - Sets BCLK pin as the external clock source.
   - Provide an external clock of 768 kHz on the BCLK pin. If a higher clock rate is required, choose an external clock divider such that the system clock remains at 768 kHz. For example, with a 6.144 MHz external clock, set EXT_CLK_RATE (Register 0x25, Bits[7:4]) to divide by 8.
4. Write 0x74 to Register 0x27 (DIG_EN).
   - Enables x, y, and z channels and sets the DOUBLE_SPEED bit in the DIG_EN register (Register 0x27, Bit 2), which is required for low latency mode.
5. Write 0x2C to Register 0x26 (OP_MODE).
   - Enables HP mode and enables PDM mode.

External Sync Mode

External synchronization allows the user to synchronize acceleration sampling to an external synchronization signal. This feature is only supported for the high performance signal chain (HP, RBW, or LP operational modes). In order to enable the external sync with the internal clock mode follow these steps:

1. Optional: write 0x00 to Register 0x25 (CLK_CTRL).
   - This setting configures the ADXL380 to use the internal clock. If an external clock is desired, write 0b10 to the CLK_SRC bits (Register 0x25, Bits[1:0]). See the External Synchronization section of the ADXL380 data sheet for more details.
2. Write 0x10 to register 0x35 (SYNC_CFG).
   - SYNC_MODE = 1. This setting synchronizes the output data rate of the ADXL380 with the external sync signal.
3. Write 0x70 to Register 0x27 (DIG_EN).
   - Enables x, y, and z channels.
4. Write 0x0C to Register 0x26 (OP_MODE).
   - Enables HP mode.

Heart Sound (HS) Mode

The ADXL380 has a heart sounds mode that optimizes performance and power consumption. Heart sounds mode is a single channel mode (z-axis only) that has low noise density while maintaining low power consumption (see the Specifications table in the ADXL380 data sheet). Heart sounds mode can be activated with the following example sequence:

1. Write 0x10 to Register 0x27 (DIG_EN).
   - Enables z channel only.
2. Perform any configuration writes to the HPF settings if desired.
3. Write 0x01 to Register 0x26 (OP_MODE).
   - Enables HS mode.

GRAPHICAL USER INTERFACE

Activity and Inactivity Detection Mode

The ADXL380 features built-in logic that detects activity (presence of acceleration more than a threshold) and inactivity (lack of acceleration more than a threshold). Activity and inactivity events can be used as triggers to manage the accelerometer mode of operation, trigger an interrupt to a host processor, and/or autonomously drive a motion switch.

See the example sequence that follows to configure activity and inactivity detection:

1. Write 0xA0 to Register 0x2B (INT0_MAP0) and 0x00 to Register 0x2C (INT0_MAP1) to map the activity interrupt to the INT0 pin.
2. Write 0x40 to Register 0x2D (INT1_MAP0) and 0x00 to Register 0x2E (INT1_MAP1), to map the inactivity interrupt to the INT1 pin.
3. Write 0x05 to Register 0x37 (ACT_INACT_CTL) to enable activity and inactivity detection.
4. Write 0x03 to Register 0x38 (SNSR_AXIS_EN) to enable x-axis and y-axis for activity and inactivity detection.
5. Write 0x00 to Register 0x3B (TIME_ACT_H), 0x00 to Register 0x3C (TIME_ACT_M), and 0xC8 to Register 0x3D (TIME_ACT_L) to set the activity time to 100 ms.
6. Write 0x00 to Register 0x40 (TIME_INACT_H), 0x00 to Register 0x41 (TIME_INACT_M), and 0xC8 to Register 0x42 (TIME_INACT_L) to set the activity time to 100 ms.
7. Write 0x02 to Register 0x26 (OP_MODE) to enable ultra-low power (ULP) mode.
8. Write 0x01 to Register 0x39 (THRESH_ACT_H) and 0x00 to Register 0x3A (THRESH_ACT_L) to set the activity threshold to 500 mg.
9. Write 0x00 to Register 0x3E (THRESH_INACT_H) and 0x80 to Register 0x3F (THRESH_INACT_L) to set the inactivity threshold to 250 mg.

Note that activity and inactivity detection only works in ULP or very low power (VLP) mode.

Figure 7 shows an example of activity and inactivity detection.

007

Figure 7. Activity and Inactivity Detection

GRAPHICAL USER INTERFACE

Tap Detection Mode

TheADXL380 have a built-in single, double, and triple tap detection. The tap threshold, duration, window, and latency can all be configured to fit the application needs (see the Tape Interrupt Function with Valid Single Tap, Double Tap, and Triple Tap figure in the ADXL380 data sheet).

See the example sequence that follows to configure tap detection:

1. Write 0x01 to Register 0x2C (INT0_MAP1).
   - Maps the single tap detect interrupt to the INT0 pin.
2. Write 0x03 to Register 0x2E (INT1_MAP1).
   - Maps the double and triple tap detect interrupts to the INT1 pin. When multiple interrupt functions are mapped to the same interrupt pin, the logical OR of the two functions determines when the interrupt is asserted. A read of Register 0x12 (STATUS1) can determine which tap event (double or triple) has occurred.
3. Write 0x80 to Register 0x43 (TAP_THRESH) to set the tap detect threshold to 4 *g*.
4. Write 0x40 to Register 0x44 (TAP_DUR) to set the tap duration to 40 ms.
5. Write 0x20 to Register 0x45 (TAP_LATENT) to set the latency time to 40 ms.
   - A value of 0x00 in this register disables double and triple tap detection.
6. Write 0x7D to Register 0x46 (TAP_WINDOW) to set the tap window duration to 320 ms, which sets the duration time where additional taps can be detected (second or third).
7. Write 0x06 to Register 0x48 (TAP_CFG).
   - This enables triple tap detection and sets the z-axis as the active tap detection axis.
8. Write 0x70 to Register 0x27 (DIG_EN).
   - Enables x, y, and z channels.
9. Write 0x03 to Register 0x26 (OP_MODE).
   - Enables VLP mode.

Concurrent Mode

The successive approximation register (SAR) path modes (ULP and VLP) can be combined with any one of the Σ-Δ modes (HP, RBW, or LP) for concurrent data sampling. In concurrent mode, the HP, RBW, or, LP data is only available on the audio port (I^2S/TDM/PDM), and the ULP and VLP data is accessed via the control port (SPI or I^2C).

Implementing Free Fall Detection

The ADXL380 has built-in, free fall detection using an inactivity interrupt.

When an object is in true free fall, acceleration on all axes is 0 *g*. Therefore, free fall detection is achieved by looking for acceleration on all axes to fall to less than a certain threshold (close to 0 *g*) for a certain amount of time. The inactivity detection functionality, when used in absolute mode, does exactly this.

To use inactivity to implement free fall detection, set the value in the THRESH_INACT_x register (Register 0x3E and Register 0x3F) to the desired free fall threshold. Values between 300 m*g* and 600 m*g* are recommended; the register setting for these values varies based on the *g* range setting of the ADXL380, as follows:

THRESH_INACT_x =

Threshold Value (*g*) × *Scale Factor* (LSB per *g*)

Set the value in the TIME_INACT_x register (Register 0x40 through Register 0x42) to implement the minimum amount of time that the acceleration on all axes must be less than the free fall threshold to generate a free fall condition. Values between 100 ms and 350 ms are recommended; the register setting for this varies based on the output data rate.

TIME_INACT_x = *Time* (sec) × *Data Rate* (Hz)

When a free fall condition is detected, the inactivity status is set to 1, and, if the function is mapped to an interrupt pin (INT0 or INT1), an inactivity interrupt triggers on that pin.

See the ADXL380 data sheet for additional register information.

GRAPHICAL USER INTERFACE

RECORD DATA

In addition to capturing visual data and saving graphs in **.png** format (by using the mouse right-click option), tabulated data vs. time can also be recorded on the PC or a SD card.

Data Recording on PC

Save data in **.csv** format directly onto the PC. Access recording options through the tab at the bottom right of the GUI. Click the **CSV** icon to display the **Save** option and specify a file name. Check off the **Auto label** box to automatically generate a name based on the sensor name, date, and time. Use the **Overwrite** option to replace data in an existing file. Record interrupts (**Record Interrupts**) and magnetometer data (**Record Magnet Data**) by checking their respective boxes. The **Start**, **Pause**, and **Stop** buttons offer flexibility to initiate, pause, and terminate data recording as needed. During data recording, the **Recorded Data Count** indicator displays the number of recorded data points, providing real-time monitoring of the recording progress. Figure 8 shows the **Recording** tab on the PC.

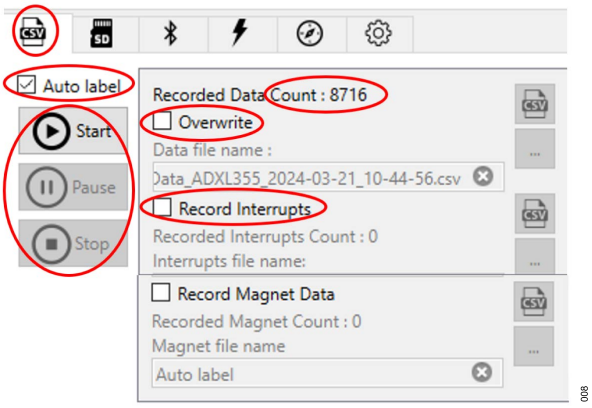

Figure 8. Data Recording on PC

SD Card

Save data on the SD card by selecting the **SD** tab located in the bottom right corner of the GUI. Upon connecting the SD card to the EVAL-ADIS-MCBZ, the **SD card detected** status displays. This status also indicates the number of files stored on the SD card and its remaining memory capacity. To commence data capture, click **Start**. To pause or stop data capture, utilize the **Pause** and **Stop** buttons, respectively. When data capturing is stopped, a window appears displaying a report detailing the recorded file, including its name and the number of samples recorded. The SD card functionality also includes options to refresh (**REFRESH**) or format (**FORMAT**) the card, which are accessible via icons situated on the right side of the **SD Card** tab. It is essential to note that SD cards must be formatted before initial use. To save captured data on the PC, click the **DOWNLOAD** icon to save the data as an **.adr** file. See Figure 9 for the icons associated with the SD card functions.

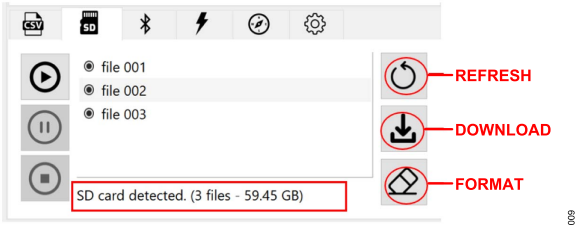

Figure 9. SD Card

EVALUATION SYSTEM FEATURES

The evaluation system for the ADXL380 is based on the EVAL-ADIS-MCBZ evaluation platform. Therefore, the ADXL380 system includes all the features of the EVAL-ADIS-MCBZ platform, such as Bluetooth communication and current measurements. For more details in the data buffering method, data analysis, Allan variance, and calibration, refer to the EVAL-ADIS-MCBZ evaluation system.

EVALUATION BOARD SCHEMATICS

Schematics of the EVAL-ADXL380-EB-1Z or EVAL-ADXL380-EB-2Z evaluation board and the EVAL-ADXL380-BB-1Z or EVAL-ADXL380-BB-2Z breakout board for both the SPI and I²C protocols are shown in Figure 10 and Figure 11.

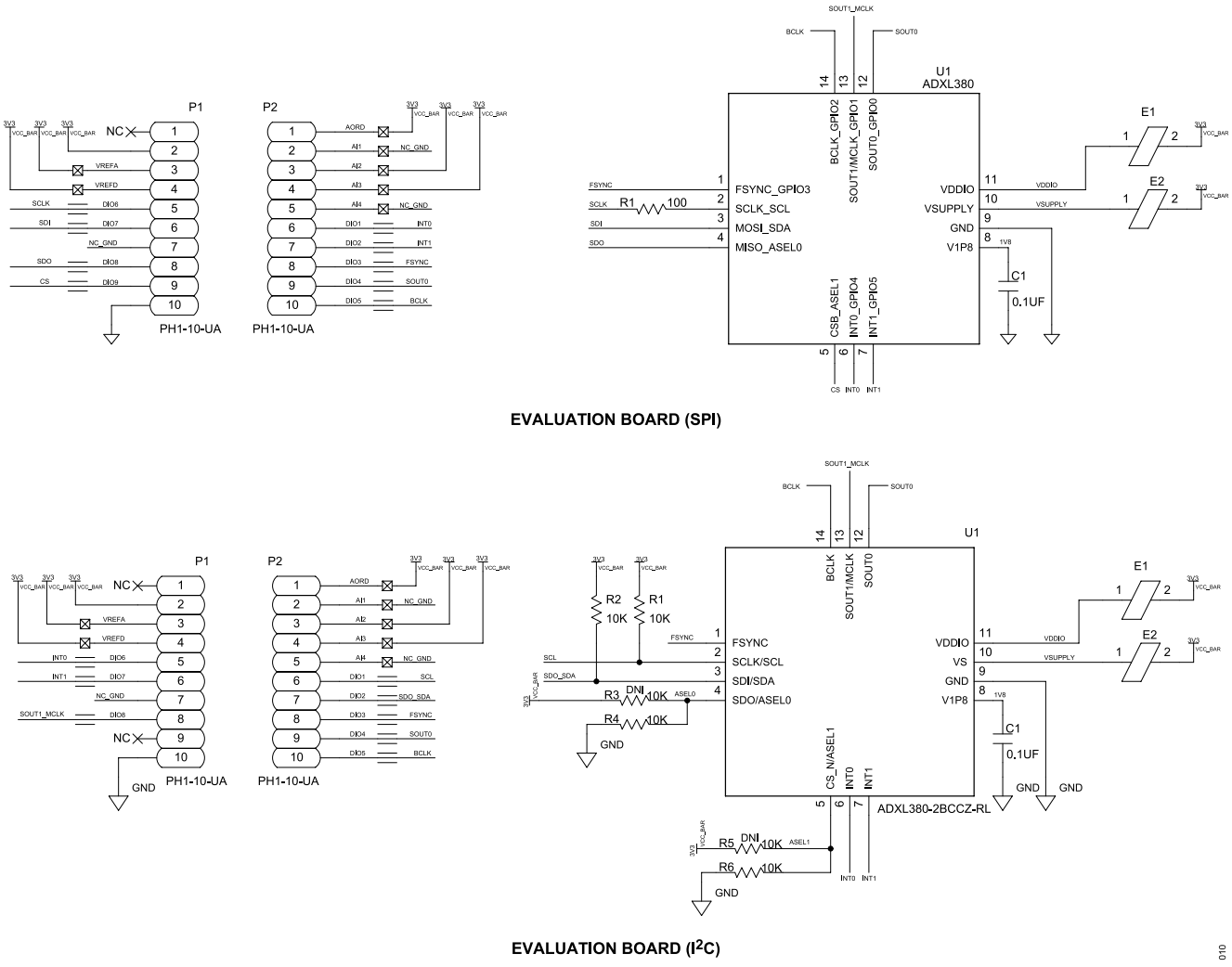

Figure 10. EVAL-ADXL380-EB-1Z or EVAL-ADXL380-EB-2Z Evaluation Board Schematic

EVALUATION BOARD SCHEMATICS

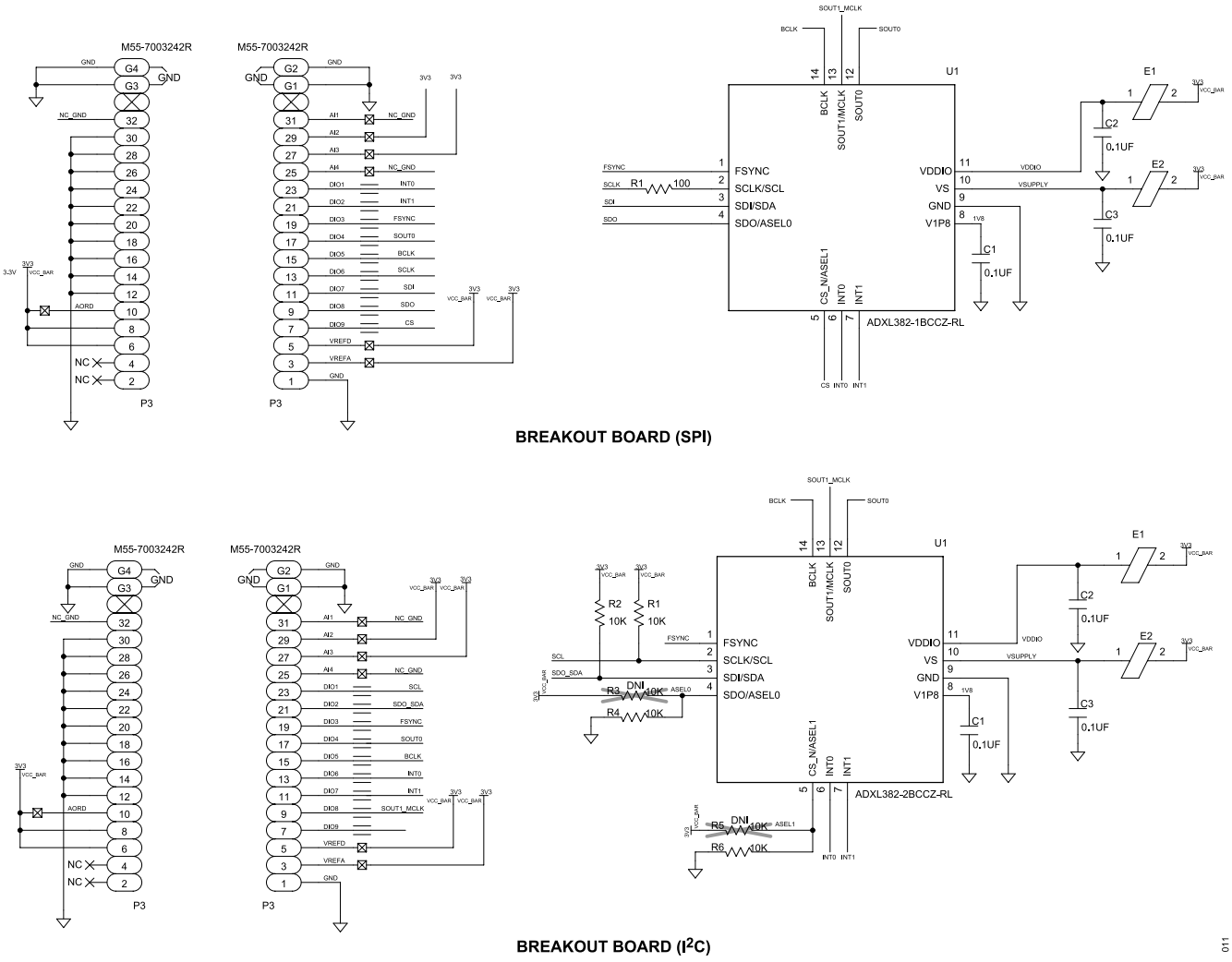

Figure 11. EVAL-ADXL380-BB-1Z or EVAL-ADXL380-BB-2Z Breakout Board Schematic

NOTES

ESD Caution
ESD (electrostatic discharge) sensitive device. Charged devices and circuit boards can discharge without detection. Although this product features patented or proprietary protection circuitry, damage may occur on devices subjected to high energy ESD. Therefore, proper ESD precautions should be taken to avoid performance degradation or loss of functionality.

Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

Analog Devices Inc.:

EVAL-ADXL380-EB-2Z EVAL-ADXL380-BB-2Z EVAL-ADXL380-EB-1Z EVAL-ADXL380-BB-1Z