Embedded Orientation Detection Using the MMA8451, 2, 3Q

by: **Kimberly Tuck**
Applications Engineer

1.0 Introduction

Many hand-held devices integrate consumer grade accelerometers into mobile handsets. By incorporating more intelligence from hand gestures, the accelerometer has had a significant impact on user interaction with handsets. Minimizing the need for buttons has changed the design of the handset interface by including motion inputs such as flicks, taps, shakes and varied orientations, all of which are interpreted by handsets. Further, handsets can provide entertainment in terms of motion-based games. Most motion-based hand gestures and games are derived from analyses of static acceleration. These analyses, in turn, are based on gravity to determine the change in tilt angles.

Accelerometers provide a new way for navigating, scrolling, and viewing information. With these sensors, even user activity levels can be monitored, e.g. while carrying the device and by counting steps.

This application note targets the ***portrait/landscape orientation detection feature*** which has become standard in many hand-held electronic devices. Additionally, this application note aims to explain uses as well as highlight some of the challenges of designing an embedded algorithm into the sensor. Included in content, the embedded settings of the MMA8451, 2, 3Q are explained and detailed for implementation.

Note: Although embedded algorithms typically lack flexibility, the algorithm of the MMA8451, 2, 3Q was designed to offer a variety of settings for the user.

1.1 Key Words

Accelerometer, Static Acceleration, Tilt Angles, Portrait/Landscape Orientation, Embedded Algorithm MMA8450Q, Z-Angle Lockout, XYZ Output Data, Low Current Consumption, Motion and Tap Detection, Design Flexibility, Hysteresis, 3-axis Accelerometer, Offset Considerations, Sample Rate, Debounce

TABLE OF CONTENTS

1.2 Summary

A. The key advantage of having the orientation detection as an embedded algorithm is that it permits the user to set up an interrupt service routine to get an update when the orientation has changed.

B. The status register is only read when a change has occurred.

C. Less processing is required on the microcontroller than having to poll the XYZ registers and calculate out the corresponding orientations.

D. There is a choice of up to 10 different trip angles from Portrait to Landscape with hysteresis from 0° to 24°.

E. There are 4 front/back trip angles and there are 8 different settings for the Z-Angle lockout from 14° to 42°.

F. The orientation detection can be used at 8 different sample rates and can automatically switch from a higher rate to a lower rate with the ability to adjust the debounce counter to make the transitions smooth without long delays.

G. The debounce counter is changeable in either the active or standby mode to allow for adjustments after the part transitions from wake to sleep mode.

2.0 MMA8451, 2, 3Q Consumer 3-axis Accelerometer 3 x 3 x 1 mm

The MMA8451, 2, 3Q has a selectable dynamic range of ±2g, ±4g, ±8g. The device has 8 different output data rates, selectable high pass filter cut-off frequencies, and high pass filtered data. The available resolution of the data and the embedded features is dependant on the specific device.

Note: The MMA8450Q has a different memory map and has a slightly different pin-out configuration.

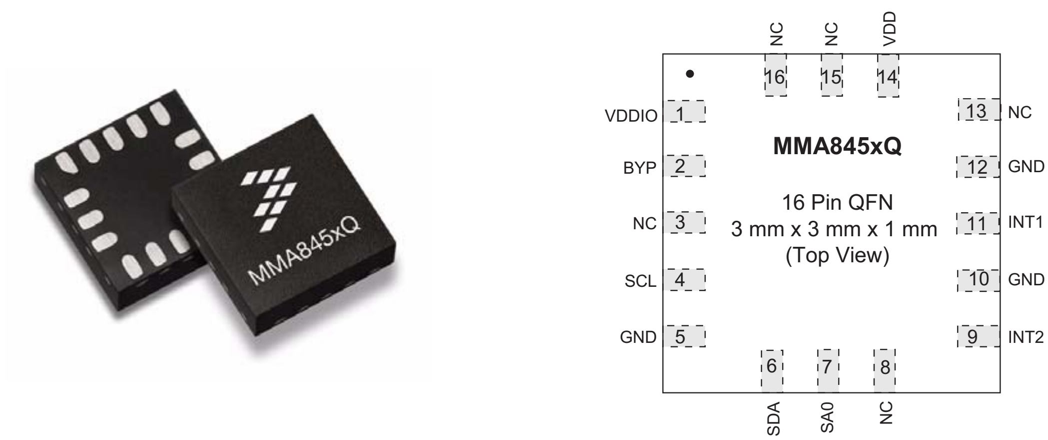

Figure 1. MMA8451, 2, 3Q Consumer 3-axis Accelerometer 3 x 3 x 1 mm

2.1 Output Data, Sample Rates and Dynamic Ranges of all Three Products

2.1.1 MMA8451Q

1. **14-bit data**
   2g (4096 counts/g = 0.25 mg/LSB) **4g** (2048 counts/g = 0.5 mg/LSB) **8g** (1024 counts/g = 1 mg/LSB)
2. **8-bit data**
   2g(64 counts/g = 15.6 mg/LSB) **4g** (32 counts/g = 31.25 mg/LSB) **8g** (16 counts/g = 62.5 mg/LSB)
3. **Embedded 32 sample FIFO (MMA8451Q)**

2.1.2 MMA8452Q

1. **12-bit data**
   2g (1024 counts/g = 1 mg/LSB) **4g** (512 counts/g = 2 mg/LSB) **8g** (256 counts/g = 3.9 mg/LSB)
2. **8-bit data**
   2g (64 counts/g = 15.6 mg/LSB) **4g** (32 counts/g = 31.25 mg/LSB) **8g** (16 counts/g = 62.5 mg/LSB)

2.1.3 MMA8453Q Note: No HPF Data

1. **10-bit data**
   2g (256 counts/g = 3.9 mg/LSB) **4g** (128 counts/g = 7.8 mg/LSB) **8g** (64 counts/g = 15.6 mg/LSB)
2. **8-bit data**
   2g (64 counts/g = 15.6 mg/LSB) **4g** (32 counts/g = 31.25 mg/LSB) **8g** (16 counts/g = 62.5 mg/LSB)

2.2 Application Notes for the MMA8451, 2, 3Q

The following is a list of all the application notes available for the MMA8451, 2, 3Q:

- **AN4068, *Embedded Orientation Detection Using the MMA8451, 2, 3Q***
- **AN4069**, *Offset Calibration of the MMA8451, 2, 3Q*
- **AN4070**, *Motion and Freefall Detection Using the MMA8451, 2, 3Q*
- **AN4071**, *High Pass Filtered Data and Functions Using the MMA8451, 2,3Q*
- **AN4072**, *MMA8451, 2, 3Q Single/Double and Directional Tap Detection*
- **AN4073**, *Using the 32 Sample First In First Out (FIFO) in the MMA8451Q*
- **AN4074**, *Auto-Wake/Sleep Using the MMA8451, 2, 3Q*
- **AN4075**, *How Many Bits are Enough? The Trade-off Between High Resolution and Low Power Using Oversampling Modes*
- **AN4076**, *Data Manipulation and Basic Settings of the MMA8451, 2, 3Q*
- **AN4077**, *MMA8451, 2, 3Q Design Checklist and Board Mounting Guidelines*

3.0 Fundamentals of Tilt for Orientation Detection

The accelerometer sensor is used to add intelligence into hand-held devices. The accelerometer can detect the orientation of the device which can be used to alert the hand-held device to update the image based on the sensor orientation data. This is implemented so that images on the screen always appear upright to the user. Figure 2 shows all the different orientations. More detail of the different orientations will be discussed in the following sections.

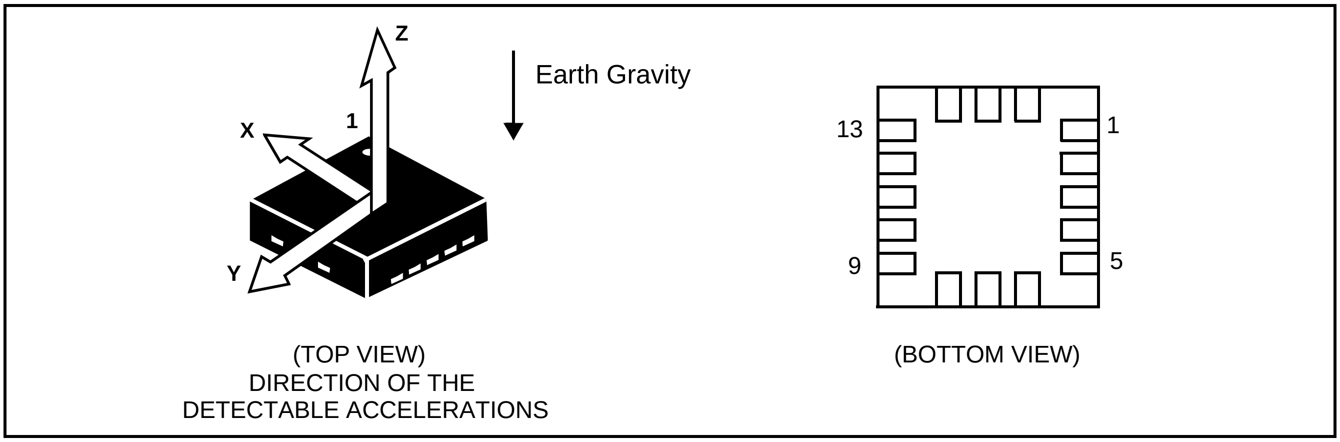

Figure 2. Direction of the Detectable Accelerations

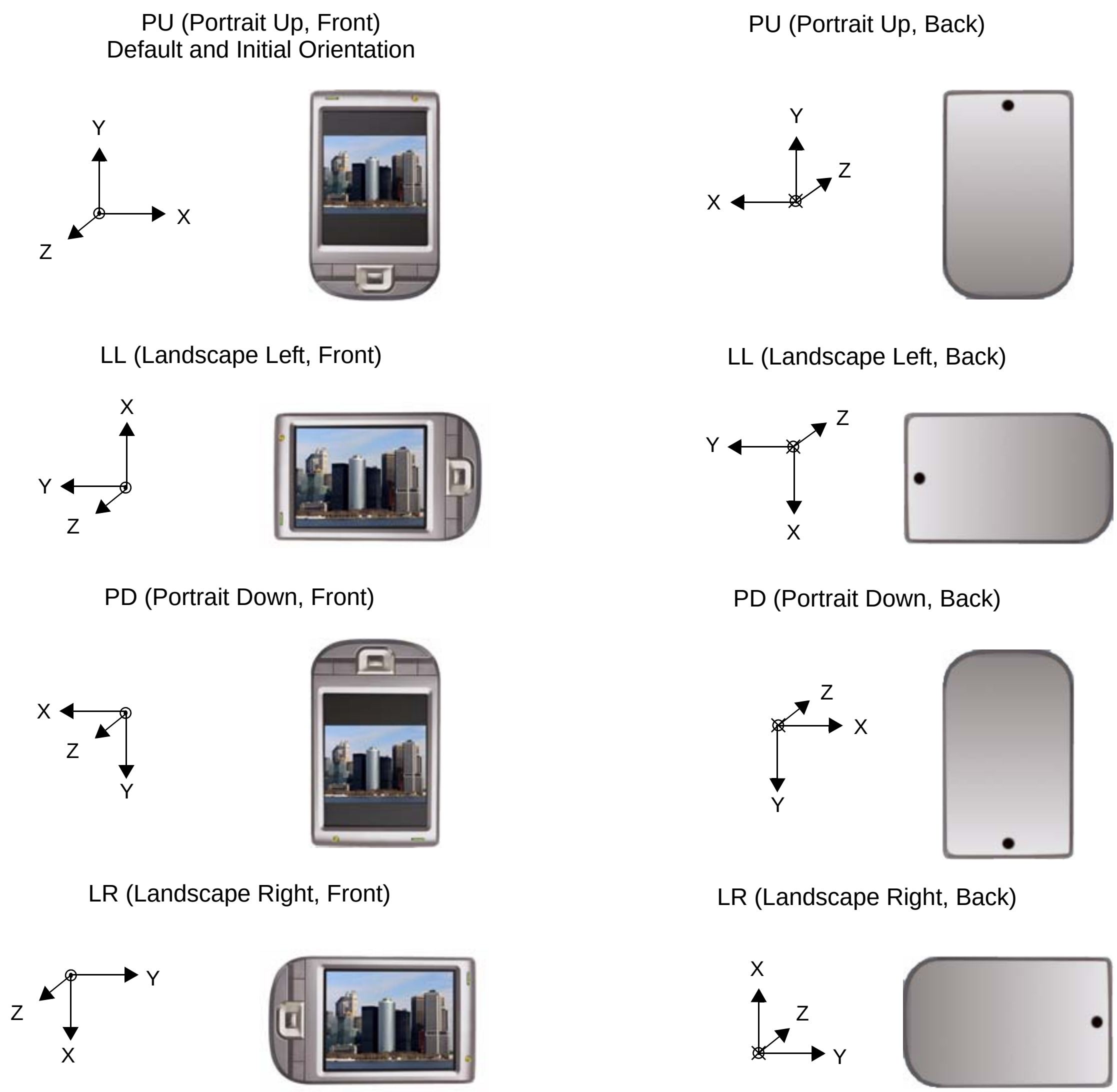

Figure 3. Landscape to Portrait Screen Orientation Change Positions

This application is based on tilt sensing. For more details on tilt sensing, please refer to Freescale application note, AN3461. Tilt is a static measurement. The force of gravity is used as an input to determine the orientation of an object calculating the degree of tilt. The accelerometer will experience acceleration in the range from -1g to +1g through 180° of tilt. Figure 4 is a graphical representation of the change in acceleration of both the X and the Y-axis.

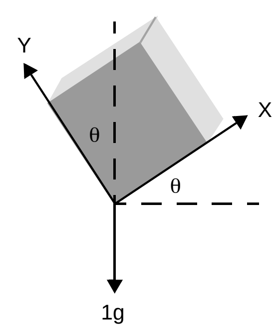

Figure 4. Reference Frame for Tilt

Note: 1g = -9.8 m/s^2

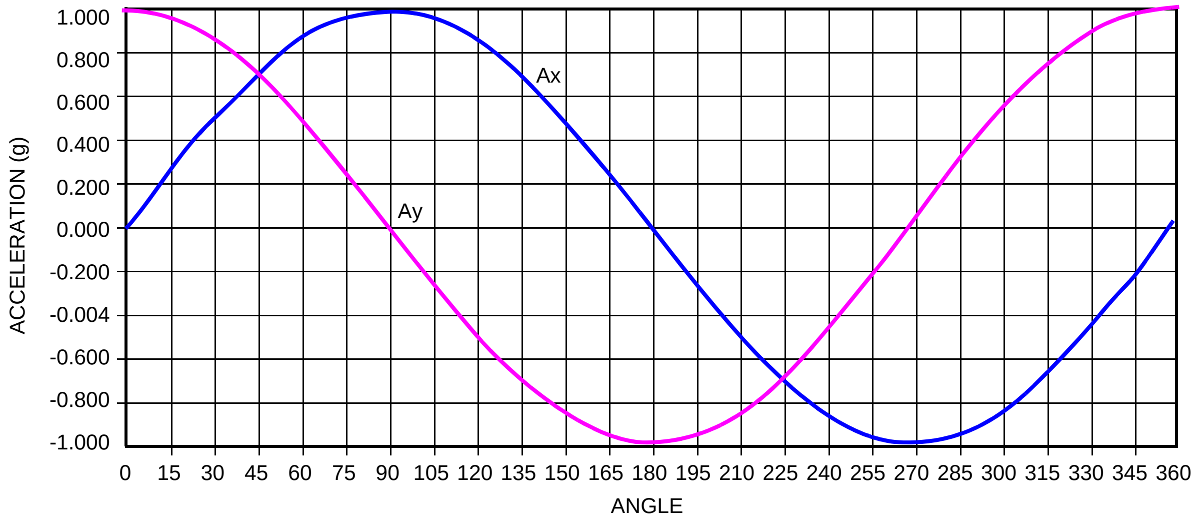

Figure 5. Sine Function of the X Output and Cosine Function of the Y Output

Figure 5 demonstrates that it is easy to detect the different orientations of the device:

- At 0° the device would be in the Portrait Up position
- At 90° the device is at the Landscape Left position
- At 180° the device is in the Portrait Down position
- At 270° the device is in the Landscape Right position

The ideal trip angle to change between states, would be at 45°, which is in the middle of the two states.

4.0 Challenges and Advanced Features for Orientation Detection

Embedded algorithms are typically somewhat restrictive. However, there are enhancements that have been made to allow more flexibility in the design of the MMA8451, 2, 3Q without the need to poll the acceleration XYZ outputs and analyze data using the processor. With a larger demand for sensors in hand-held devices, this intelligence is expected by users.

4.1 Front and Back Detection

When the device is facing up, it is considered in the front view and when the device is facing down, it is considered in the back view as shown in Figure 6.

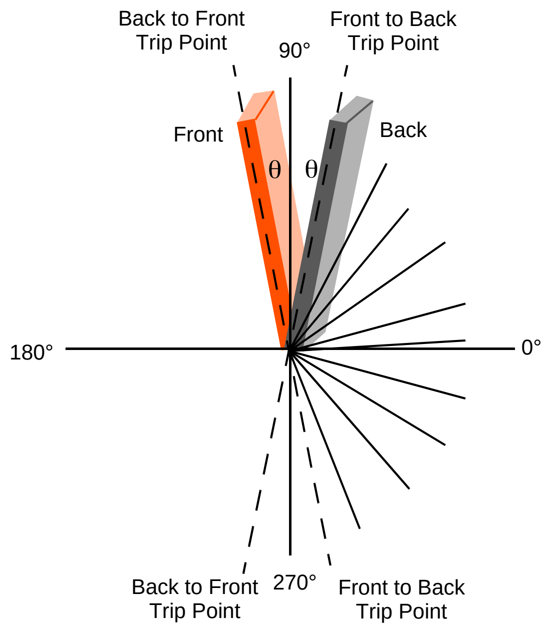

Figure 6. Front/Back Trip Point

The angle for switching between front and back is calculated by taking the inverse sine of the acceleration of the Z-axis.

$$\theta = \sin^{-1}(a_z)$$

The MMA8451, 2, 3Q allows the user to choose between four different Back to Front/Front to Back trip angles as described below. This allows for more flexibility in the usability for the application as compared to an embedded design that would typically only allow for one trip angle. From the following options, the user can select where this trip point should be.

10°: Z < 80° and Z > 280° Back	Z > 100° and Z < 260° Front
15°: Z < 75° and Z > 285° Back	Z > 100° and Z < 255° Front
20°: Z < 70° and Z > 290° Back	Z > 110° and Z < 250° Front
25°: Z < 65° and Z > 295° Back	Z > 115° and Z < 245° Front

Note: If the first position of the device is in-between the front/back trip angles then the MMA8451, 2, 3Q will not know which state it is in and will assume no state until the device transitions past the trip point.

4.2 Setting the Threshold Angle and the Hysteresis

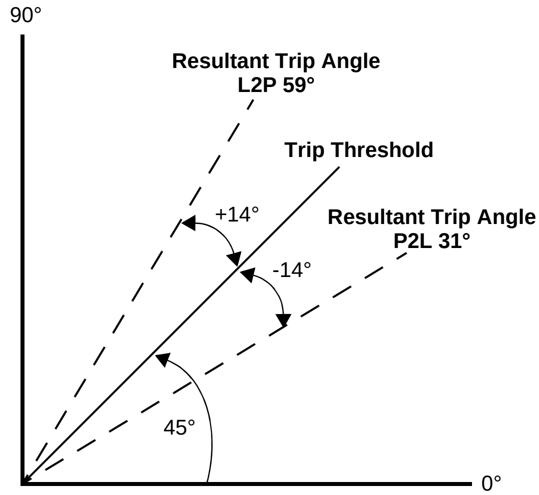

Figure 7. Setting the Trip Angle

The MMA8451, 2, 3Q allows the user to select a range of 10 different trip angles from 15° to 75° with increments of 5°. Figure 7 shows the default angle at 45°.Although the 45° trip angle seems like the ideal angle between the two states, a usability problem could occur when the device is held near the 45° angle. A very slight movement can cause the device to flicker slightly above and then below the 45° angle that could make the screen jump back and forth between portrait and landscape. This issue can be seen particularly when using higher sampling rates without any debounce filtering. This can be entirely avoided by adding in a buffer around the trip threshold. By adding in the hysteresis angle this problem can be avoided. The default hysteresis value is ±14° as shown above in Figure 7. With the hysteresis built-in the orientation will change from Portrait to Landscape at 31° and from Landscape to Portrait at 59°. With separate trip points in this manner allows for smooth transitions from one state to the next. The user must tilt the device to 59° to go from Landscape to Portrait. And then to return immediately to Landscape, the user would rotate the device back to 31° to make it trip to Landscape. The transition from Portrait to Landscape and Landscape to Portrait is shown in Figure 8.

Figure 8. Changing from Portrait to Landscape

4.3 Z-Lockout and Effects on Choosing Two Trip Angles P2L and L2P

The next consideration is the angle of Z-axis on which a user holds the mobile device while rotating it to change the image from Portrait to Landscape. The angle at which the image no longer detects the orientation change is referred to as the “Z-lockout angle”. Based on known functionality of linear accelerometers, it is not possible to rotate the device about the Z-axis and to detect change in acceleration at slow angular speeds as shown in Figure 9.

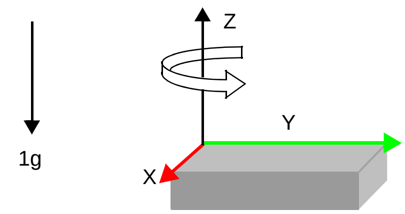

Figure 9. Image of Z-Lockout, showing no acceleration sensing at low speed

In a 3-axis accelerometer, it is required that acceleration sensed in the X and Y axes must differ by a minimum amount to sense when the tilt angle has met a pre-set condition. When the device is placed on a flat surface and the Z angle is 0 degrees with respect to a line in the direction of the earth’s gravity, there is no way to determine the orientation of the device and the screen should stay in the last position. A minimum angle called the “Z-lockout” angle is defined where the last position is held until the device is tilted beyond this limit. Ideally, the X and Y outputs would change from 0g to 1g on the X and Y axes as the device rotated if a user held the device perfectly vertical as in image #1 of Figure 10. However, this is not a very likely scenario. Most users will hold the device from 25° - 30° up to 60° - 75° from the horizontal as shown in the images below in #2, #3 and #4. Note now that in #2 - #4, the magnitude of the acceleration of X and Y is now smaller. They are now scaled by the sine of the Z-tilt angle.

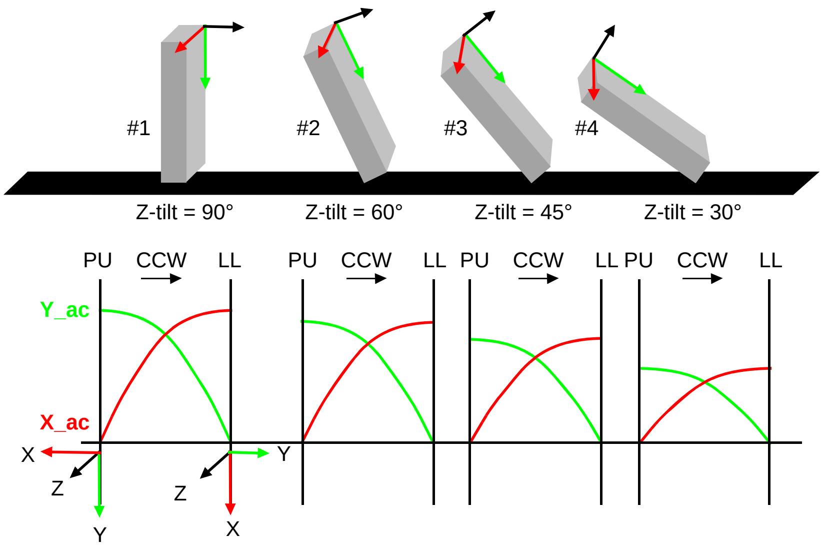

Figure 10. Examples of How the Z-tilt Angle Affects Scaling on the X and Y Axes

Since the output acceleration of X and Y will be scaled based on the Z-axis component, choosing the threshold trip angle is made more difficult. However, the algorithm designed into the MMA8451, 2, 3Q takes into account the Z-lockout range expected. The MMA8451, 2, 3Q Z-lockout angle selections range from 14° to 42° with increment changes of approximately 4°. The selection choices for the Z-lockout are the following: Angle 14°, 18°, 21°, 25°, 29°, 33°, 37°, 42°. All angles are accurate to about 2°. These values are all settable in the register configuration of the device for the Z-lockout angle. The max and min Z-lockout angles are shown in Figure 11.

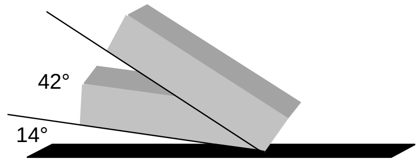

Figure 11. Z-Lockout Max and Min Angles

4.4 0g Offset Considerations

The accelerometers are trimmed according to sensitivity and offset specifications by adjusting the gain and offset trim codes. Test tolerances are placed on the device during this process. After the accelerometer has been assembled onto a PCB the offset can shift due to package stresses. This ultimately results in the accelerometer appearing to be rotated or tilted relative to the desired reference position. Often this shift is very small and for most applications it will not be noticeable. A larger offset shift of 0.1g will result in an angle shift of about 5.7°. The algorithm designed into the MMA8451, 2, 3Q has accounted for typical offset shifts seen after board mounting. For more accuracy in setting the trip angles and the lockout angle the calibration registers 0x2F, 0x30 and 0x31 can be used. Details on how to calibrate the MMA8451, 2, 3Q can be found in Freescale application note AN4069.

4.5 1g Lockout Threshold Settings

When the accelerometer is not moving, the Root Mean Square (RMS) of the acceleration vectors is equal to 1g. There are many circumstances where users may be jogging or in a train or on a bus where they may be bouncing above 1g. When the device is experiencing acceleration above a set threshold, the screen orientation should not interpret this as a change and the screen should lock in the last known valid position. The set threshold is 1.25g in the MMA8451, 2, 3Q.

4.6 Sample Rate and Debounce Counter Settings

4.6.1 Sample Rate

Some applications may require faster response times such as for transient detection or for tap detection. The portrait/landscape application does not typically require a fast response time and it could be run at 12.5 Hz or 6.25 Hz. The embedded orientation detection will still work in sleep mode (typically lower sample rates from 1.56 Hz to 50 Hz) in the MMA8451, 2, 3Q. A debounce counter can be set to filter out faster movements and can be changed in either the Standby or Active mode. This creates a delay in the reaction time of the orientation update, which may be desired. The device can be used at a high sample rate (400 Hz) to be able to detect fast transitions from the XYZ output data and at the same time the orientation detection update rate can be modified to be much slower by using the debounce counter to filter out fast transitions.

4.6.2 Debounce Counter Settings

A debounce counter is often used to improve the reliability of the screen orientation. For example, jittery hands and small vibrations can cause false accelerations, tripping the orientation to change even when nothing has really happened. These false accelerations are smoothed out using a debounce counter to ensure that the orientation has been steady in the new position long enough to warrant a change in position. The MMA8451, 2, 3Q debounce counter is an 8-bit value which is dependent on the oversampling mode and sample rate. Therefore up to 256 samples can be averaged. The debounce counter will be set for the output data rate value assumed for the active mode but may need to be readjusted if the sample rate changes significantly when the device goes into sleep mode. For this reason the debounce counter is accessible to change while the device is active.

Note: The longer the time set for the debounce counter, the longer the delay. This can significantly slow down the response time.

5.0 Details for Configuring the MMA8451, 2, 3Q for Orientation Detection

The MMA8451, 2, 3Q data sheets review in detail the register settings and information on how to configure the angle settings. The intent is not to try to repeat the data sheet but to highlight the registers of interest and explain their use. Refer to the data sheet and use the tables to calculate all the correct angle settings. Table 1 lists the registers of importance for setting up the orientation detection.

Table 1. Registers of Importance for Configuring the Orientation Detection in the in the MMA8451, 2, 3Q

R#	Name	Definition	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0C	INT_SOURCE	Interrupt Status R	SRC_ASLP	SRC_FIFO	SRC_TRANS	SRC_LNDPRT	SRC_PULSE	SRC_FF_MT	—	SRC_DRDY
10	PL_STATUS	PL Status R	NEWLP	LO	—	—	—	LAPO[1]	LAPO[0]	BAFRO
11	PL_CFG	PL Configuration R/W	DBCNTM	PL_EN	—	—	—	—	—	—
12	PL_COUNT	PL Debounce R/W	DBNCE[7]	DBNCE[6]	DBNCE[5]	DBNCE[4]	DDBNCE[3]	DBNCE[2]	DBNCE[1]	DBNCE[0]
13	PL_BF_ZCOMP	PL Back/Front and Z Compensation R/W	BKFR[1]	BKFR[0]	—	—	—	ZLOCK[2]	ZLOCK[1]	ZLOCK[0]
14	P_L_THS_REG	Landscape to Portrait Threshold Setting R/W	P_L_THS[4]	P_L_THS[3]	P_L_THS[2]	P_L_THS[1]	P_L_THS[0]	HYS[2]	HYS[1]	HYS[0]
2D	CTRL_REG4	Control Reg4 R/W (Interrupt Enable Map)	INT_EN_ASLP	INT_EN_FIFO	INT_EN_TRANS	INT_EN_LNDPRT	INT_EN_PULSE	INT_EN_FF_MT	—	INT_EN_DRDY
2E	CTRL_REG5	Control Reg5 R/W (Interrupt Configuration)	INT_CFG_ASLP	INT_CFG_FIFO	INT_CFG_TRANS	INT_CFG_LNDPRT	INT_CFG_PULSE	INT_CFG_FF_MT	—	INT_CFG_DRDY

5.1 Example Steps for Implementing the Embedded Orientation Detection

Step 1: Put the part into Standby Mode

CTRL_REG1_Data = IIC_RegRead(0x2A); //read contents of register
CTRL_REG1_Data& = 0xFE; //Set last bit to 0.
IIC_RegWrite(0x2A, CTRL_REG1_Data); //write the updated value in CTRL_REG1

Step 2: Set the data rate to 50 Hz (for example, but can choose any sample rate).

CTRL_REG1_Data = IIC_RegRead(0x2A); //Note: Can combine this step with above
CTRL_REG1_Data& = 0xC7; //Clear the sample rate bits
CTRL_REG1_Data| = 0x20; //Set the sample rate bits to 50 Hz
IC_RegWrite(0x2A, CTRL_REG1_Data); //Write updated value into the register.

Step 3: Set the PL_EN bit in Register 0x11 PL_CFG. This will enable the orientation detection.

PLCFG_Data = IIC_RegRead (0x11);
PLCFG_Data| = 0x40;
IIC_RegWrite(0x11, PLCFG_Data);

Step 4: Set the Back/Front Angle trip points in register 0x13 following the table in the data sheet.

PLBFZCOMP_Data = IIC_RegRead(0x13);
PLBFZCOMP_Data& = 0x3F; //Clear bit 7 and 6
Select one of the following to set the B/F angle value:
    PLBFZCOMP_Data| = 0x00; //This does nothing additional and keeps bits [7:6] = 00
    PLBFZCOMP_Data| = 0x40; //Sets bits[7:6] = 01
    PLBFZCOMP_Data| = 0x80; //Sets bits[7:6] = 02PLBFZCOMP_Data| = 0xC0; //Sets bits[7:6] = 03
IIC_RegWrite(0x13,PLBFZCOMP_Data); //Write in the updated Back/Front Angle

Step 5: Set the Z-Lockout angle trip point in register 0x13 following the table in the data sheet.

PLBFZCOMP_Data = IIC_RegRead(0x1C); //Read out contents of the register
PLBFZCOMP_Data& = 0xF8; //Clear the last three bits of the register
Select one of the following to set the Z-lockout value
    PLBFZCOMP_Data| = 0x00; //This does nothing additional but the Z-lockout selection will remain at 14°
    PLBFZCOMP_Data| = 0x01; //Set the Z-lockout angle to 18°
    PLBFZCOMP_Data| = 0x02; //Set the Z-lockout angle to 21°
    PLBFZCOMP_Data| = 0x03; //Set the Z-lockout angle to 25°
    PLBFZCOMP_Data| = 0x04; //Set the Z-lockout angle to 29°
    PLBFZCOMP_Data| = 0x05; //Set the Z-lockout angle to 33°
    PLBFZCOMP_Data| = 0x06; //Set the Z-lockout angle to 37°
    PLBFZCOMP_Data| = 0x07; //Set the Z-lockout angle to 42°
IIC_RegWrite(0x13, PLBFZCOMP_Data); //Write in the updated Z-lockout angle

Step 6: Set the Trip Threshold Angle

Select the angle desired in the table, and,
Enter in the values given in the table for the corresponding angle.
Refer to Figure 7 for the reference frame of the trip angles.
    PL_PL_THS_Data = IIC_RegRead(0x14);
    PL_PL_THS_Data& = 0x07; //Clear the Threshold values
Choose one of the following options
    PL_PL_THS_Data| = (0x07)<<3; //Set Threshold to 15°
    PL_PL_THS_Data| = (0x09)<<3; //Set Threshold to 20°
    PL_PL_THS_Data| = (0x0C)<<3; //Set Threshold to 30°
    PL_PL_THS_Data| = (0x0D)<<3; //Set Threshold to 35°
    PL_PL_THS_Data| = (0x0F)<<3; //Set Threshold to 40°
    PL_PL_THS_Data| = (0x10)<<3; //Set Threshold to 45°
    PL_PL_THS_Data| = (0x13)<<3; //Set Threshold to 55°
    PL_PL_THS_Data| = (0x14)<<3; //Set Threshold to 60°
    PL_PL_THS_Data| = (0x17)<<3; //Set Threshold to 70°
    PL_PL_THS_Data| = (0x19)<<3; //Set Threshold to 75°
    IIC_RegWrite(0x14, PL_PL_THS_Data);

Step 7: Set the Hysteresis Angle

Select the hysteresis value based on the desired final trip points (threshold + hysteresis)

Enter in the values given in the table for that corresponding angle.

Note: Care must be taken. Review the final resulting angles. Make sure there isn't a resulting trip value greater than 90 or less than 0.

The following are the options for setting the hysteresis.

PL_PL_THS_Data = IIC_RegRead(0x14);
PL_PL_THS_Data& = 0xF8; //Clear the Hysteresis values
PL_PL_THS_Data| = 0x01; //Set Hysteresis to ±4°
PL_PL_THS_Data| = 0x02; //Set Threshold to ±7°
PL_PL_THS_Data| = 0x03; //Set Threshold to ±11°
PL_PL_THS_Data| = 0x04; //Set Threshold to ±14°
PL_PL_THS_Data| = 0x05; //Set Threshold to ±17°
PL_PL_THS_Data| = 0x06; //Set Threshold to ±21°
PL_PL_THS_Data| = 0x07; //Set Threshold to ±24°
IIC_RegWrite(0x14, PL_PL_THS_Data);

Step 8: Register 0x2D, Control Register 4 configures all embedded features for interrupt detection.

To set this device up to run an interrupt service routine:

Program the Orientation Detection bit in Control Register 4.
Set bit 4 to enable the orientation detection "INT_EN_LNDPRT".

CTRL_REG4_Data = IIC_RegRead(0x2D); //Read out the contents of the register
CTRL_REG4_Data| = 0x10; //Set bit 4
IIC_RegWrite(0x2D, CTRL_REG4_Data); //Set the bit and write into CTRL_REG4

Step 9: Register 0x2E is Control Register 5 which gives the option of routing the interrupt to either INT1 or INT2

Depending on which interrupt pin is enabled and configured to the processor:

Set bit 4 "INT_CFG_LNDPRT" to configure INT1, or,
Leave the bit clear to configure INT2.

CTRL_REG5_Data = IIC_RegRead(0x2E);

In the next two lines choose to clear bit 4 to route to INT2 or set bit 4 to route to INT1

CTRL_REG5_Data& = 0xEF; //Clear bit 4 to choose the interrupt to route to INT2
CTRL_REG5_Data| = 0x10; //Set bit 4 to choose the interrupt to route to INT1

I**IC_RegWrite(0x2E, CTRL_REG5_Data);** //Write in the interrupt routing selection

Step 10: Set the debounce counter in register 0x12

This value will scale depending on the application-specific required ODR.

If the device is set to go to sleep, reset the debounce counter before the device goes to sleep. This setting helps avoid long delays since the debounce will always scale with the current sample rate. The debounce can be set between 50 ms - 100 ms to avoid long delays.

IIC_RegWrite(0x12, 0x05); //This sets the debounce counter to 100 ms at 50 Hz

Step 11: Put the device in Active Mode

CTRL_REG1_Data = IIC_RegRead(0x2A); //Read out the contents of the register
CTRL_REG1_Data| = 0x01; //Change the value in the register to Active Mode.
IIC_RegWrite(0x2A, CTRL_REG1_Data); //Write in the updated value to put the device in Active Mode

Step 12: Write a Service Routine to Service the Interrupt

Register 0x0C gives the status of any of the interrupts that are enabled in the entire device.

- An interrupt service routine must be set to handle enabling and then clearing of the interrupts. Register 0x0C will be read to determine which interrupt caused the event.
- When bit 4 is set in Register 0x0C "SRC_LNDPRT" this is the indication that a new orientation has been detected.
- Then reading register 0x10(PL_Status), the interrupt is cleared in Register 0x15 and the new PL_Status can be updated.

```
Interrupt void isr_KBI (void)
{
     //clear the interrupt flag
     CLEAR_KBI_INTERRUPT;
     //Determine the source of the interrupt by first reading the system
     interrupt register
     Int_SourceSystem=IIC_RegRead(0x0C);
     // Set up Case statement here to service all of the possible interrupts
     if ((Int_SourceSystem &0x10)==0x10)
     {
          //Perform an Action since Orientation Flag has been set
          //Read the PL State from the Status Register, clear the interrupt
     Int_SourcePL=IIC_RegRead(0x10); // PL Status Register
          //Update Image on Display Screen based on the data stored
     }
}
```

How to Reach Us:

Home Page:
www.freescale.com

Web Support:
http://www.freescale.com/support

USA/Europe or Locations Not Listed:
Freescale Semiconductor, Inc.
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
1-800-521-6274 or +1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 10 5879 8000
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
1-800-441-2447 or +1-303-675-2140
Fax: +1-303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

freescale™ semiconductor

AN4068
Rev. 1
09/2010