# 9 Axes Motion Shield Motion Example

## Accelerometer using 9 AXES MOTION SHIELD

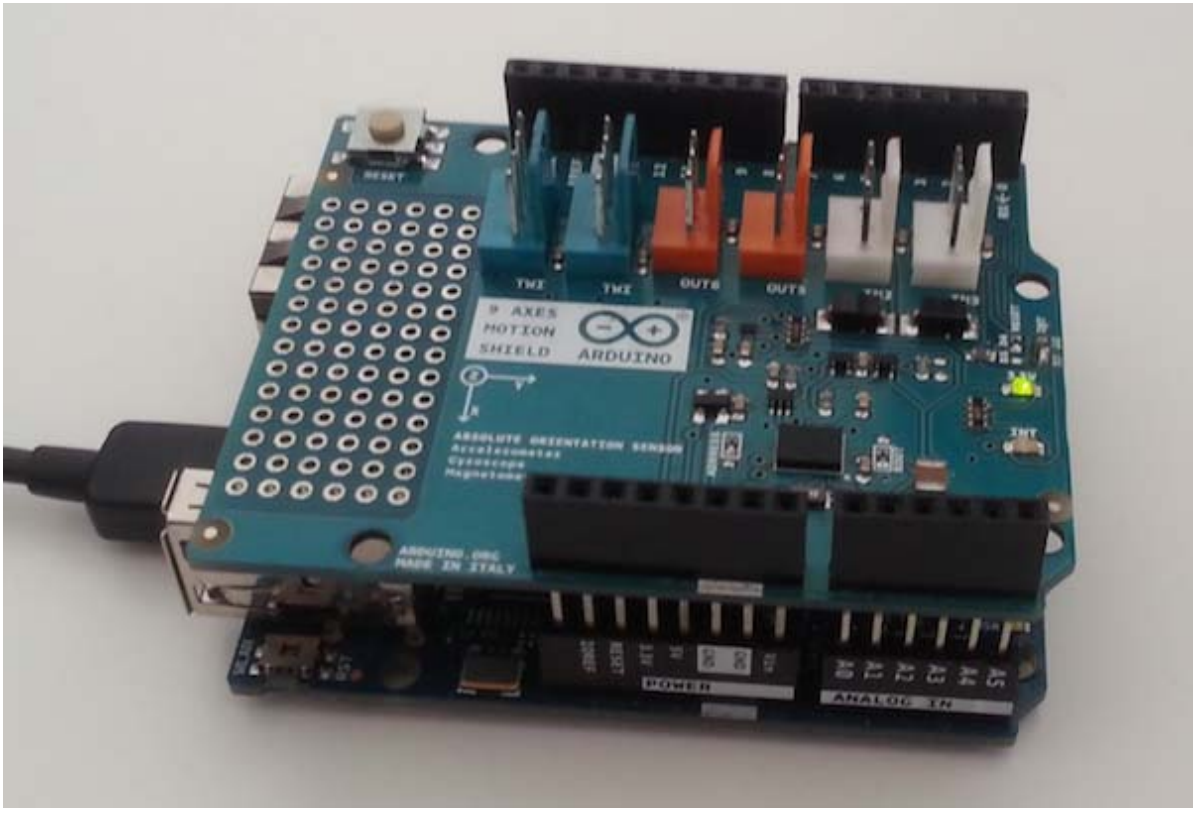
The Arduino 9 Axes Motion Shield is based on the BNO055 absolute orientation sensor from Bosch Sensortec GmbH.
This shield contains: Accelerometer, Gyroscope and Magnetometer.
The Library provided allows you to easily setup and use its sensors.
The following example is a little game made to demonstrate the Any motion and No motion Interrupt features of the shield.

## Hardware

- Arduino compatible board
- 9 Axes Motion Shield

## Circuit

Simply attach the 9 Axes Motion Shield to your Arduino Board, being careful to match all PINs.
You are ready to go!



## Code

The code starts by doing the Shield setup and creating a Serial connection.
Interrupt setups are also done in this function.
The setup functions are commented inside the code.
The loop function updates over and over Accelerometer datas and it prints them to the Serial Monitor.
For each axe there is a plenty number of data.
Note: this code is available in ArduinoIDE or ArduinoStudio inside the "Example -> 9Axes" Menu.

```
/***************************************************************************
* Copyright (C) 2011 - 2014 Bosch Sensortec GmbH
*
* Usage:  Example code of a game to demonstrate the Any motion
*              and No motion Interrupt features
* License:
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
Neither the name of the copyright holder nor the names of the
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.
The information provided is believed to be accurate and reliable.
The copyright holder assumes no responsibility for the consequences of use
of such information nor for any infringement of patents or
other rights of third parties which may result from its use.
No license is granted by implication or otherwise under any patent or
patent rights of the copyright holder.
*/

//Contains the bridge code between the API and the Arduino Environment
#include "NAxisMotion.h"
#include <Wire.h>

//Object that for the sensor
NAxisMotion mySensor;

//Flag to indicate if an interrupt was detected
bool intDetected = false;

//At a Range of 4g, the threshold
//is set at 39.05mg or 0.3830m/s2.
//This Range is the default for NDOF Mode
int threshold = 5;

//At a filter Bandwidth of 62.5Hz,
//the duration is 8ms.
```

```
//This Bandwidth is the default for NDOF Mode
int duration = 1;

//To know which interrupt was triggered
bool anyMotion = true;

//This code is executed once
void setup() {
  //Peripheral Initialization

  //Initialize the Serial Port to view information on the Serial Monitor
  Serial.begin(115200);

  //Initialize I2C communication to the let the library communicate with the sensor.
  I2C.begin();

  //Sensor Initialization
  Serial.println("Please wait. Initialization in process.");

  //The I2C Address can be changed here inside this function in the library
  mySensor.initSensor();
  mySensor.setOperationMode(OPERATION_MODE_NDOF);

  //Can be configured to other operation modes as desired
  mySensor.setUpdateMode(MANUAL);
  //The default is AUTO. Changing to manual requires
  //calling the relevant update functions prior to calling the read functions

  //Setting to MANUAL  requires lesser reads to the sensor

  //Attach the interrupt to the Interrupt Service Routine
  //for a Rising Edge. Change the interrupt pin depending on the board
  attachInterrupt(INT_PIN, motionISR, RISING);

  //Setup the initial interrupt to trigger at No Motion
  mySensor.resetInterrupt();
  mySensor.enableSlowNoMotion(threshold, duration, NO_MOTION);
  anyMotion = false;

  //Accelerometer interrupts can be triggered from all 3 axes
  mySensor.accelInterrupts(ENABLE, ENABLE, ENABLE);

  Serial.println("This is a game to test how steady you can move an object with one hand. \n
                            Keep the device on a table and mark 2 points.");

  Serial.println("Move the Device from one place to another
                            without triggering the Any Motion Interrupt.\n\n");

  delay(1000); //Delay for the player(s) to read
  Serial.println("Move the device around and then place it at one position.\n
                            Change the threshold and duration to increase the difficulty level.");
```

```
  Serial.println("Have fun!\n\n");
}

void loop() //This code is looped forever
{
  if (intDetected)
  {
    if (anyMotion)
    {
      Serial.println("You moved!! Try again. Keep the Device at one place.\n");
      intDetected = false;

      //Reset the interrupt line
      mySensor.resetInterrupt();

      //Disable the Any motion interrupt
      mySensor.disableAnyMotion();

      //Enable the No motion interrupt
      //(can also use the Slow motion instead)
      mySensor.enableSlowNoMotion(threshold, duration, NO_MOTION);
      anyMotion = false;
    }
    else
    {
      Serial.println("Device is not moving. You may start again.\n\n\n");
      intDetected = false;

      //Reset the interrupt line
      mySensor.resetInterrupt();

      //Disable the Slow or No motion interrupt
      mySensor.disableSlowNoMotion();

      //Enable the Any motion interrupt
      mySensor.enableAnyMotion(threshold, duration);
      anyMotion = true;
    }
  }
}

//Interrupt Service Routine
//when the sensor triggers an Interrupt
void motionISR() {
  intDetected = true;
}
```