Revision History

Ver.	Date	Updates	Descriptions
0.1	2014/05/13		first Release
0.2	2014/06/18		Add user interface and operation steps.
1.0	2014/12/19		V1.0 document updates

General Introduction

This APP is helpful for GMA driver to calibrate the accelerometer.
Please refer to the following.

Application of GlobalMems Gsensor calibration.

Package Name	GLevel
Application Name	com.globalmems.sensor.level.app
Create Activity	Level.java
Min SDK Version	11

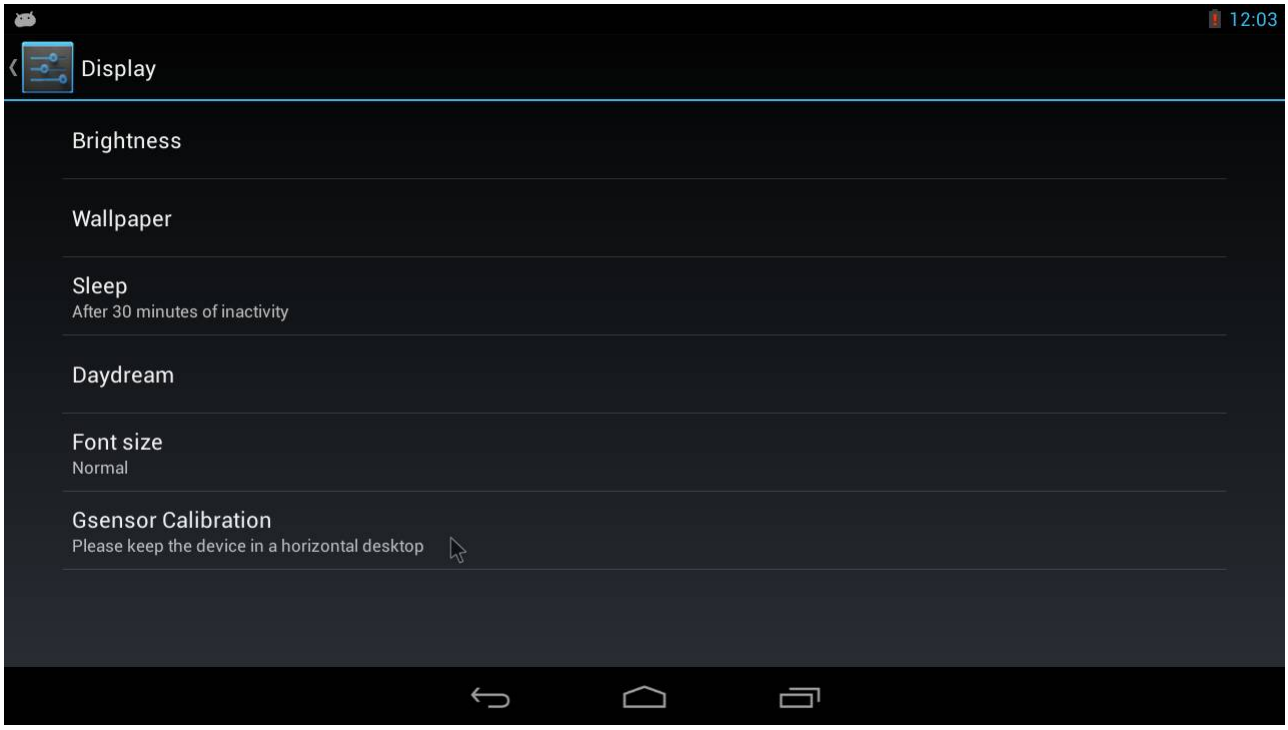
Bubble Level Features

1. Install & Run GLevel_L.apk(LUNCHER) or GLevel_D.apk(DEFAULT),
   GLevel_L1.apk & GLevel_D1.apk (Calibration options use up)
   GLevel_L2.apk & GLevel_D2.apk (Calibration options use down)
   GLevel_L9.apk & GLevel_D9.apk (Calibration options use autoZ)

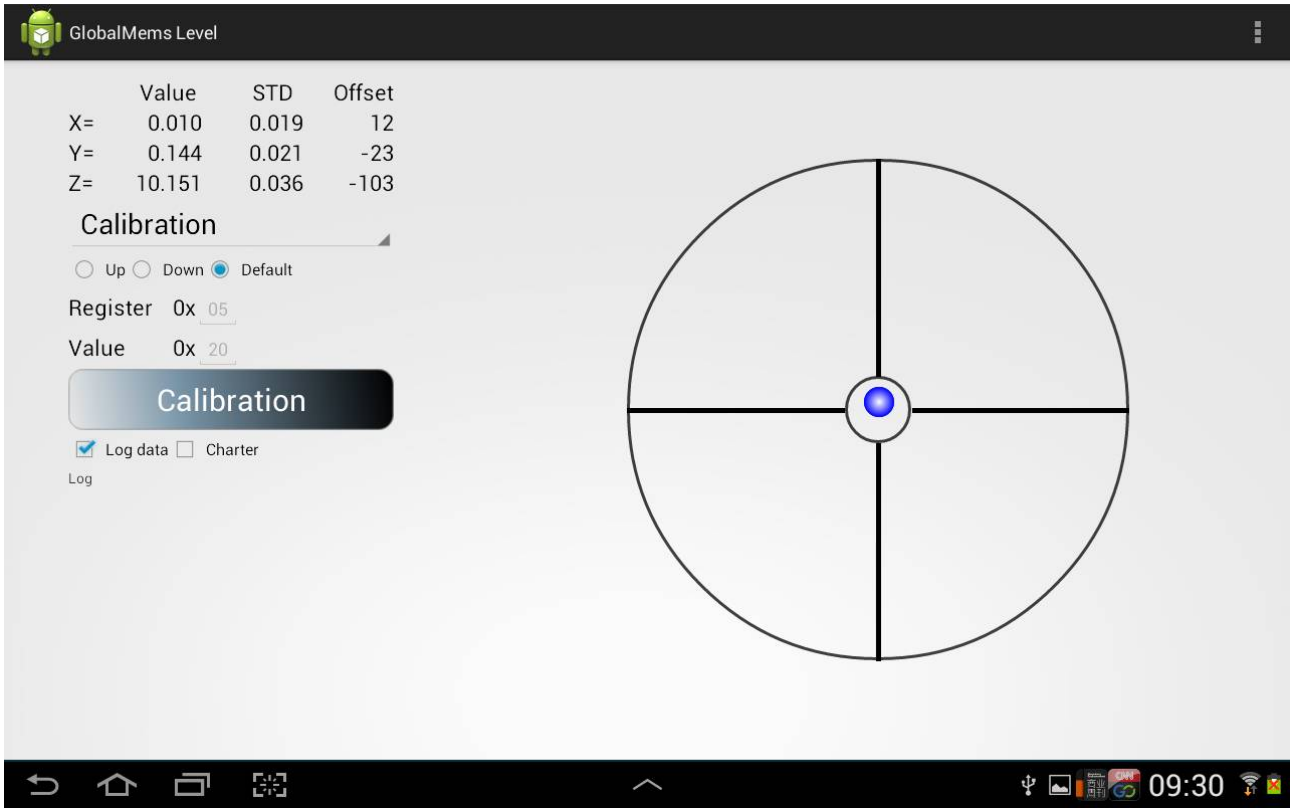
   We recommend that you should use GLevel_L9.apk or GLevel_D9.apk.

2. First boot, it will trigger BoardCastReceiver..
   check File("/data/misc/sensor/offset.txt")
   if offset (X=0 & Y=0 & Z=0)
   Broadcast starts APP.

3. Before Calibration
   Note : Please keep the device in a horizontal desktop.



4. Click Button “Calibration”
   Make sure the bubble moves to the center.
   Calibration has been successful at this time.
   X/Y/Z value of close to 0/0/9.807
   And the offset value save to "/data/misc/sensor/offset.txt ".



The APK put inside the "setting > display"

The following is placed inside the apk: “setting” > “display”

1. File “$Android\packages\apps\Settings\res\xml\display_settings.xml”

Add the following code to add an option to perform link “Level”

```
<PreferenceScreen
        android:key="accelerometerAdjust"
        android:title="@string/gsensor_settings"
        android:summary="@string/gsensor_settings_summary" >
        <intent
            android:action="android.intent.action.MAIN"
            android:targetPackage="com.globalmems.sensor.level.app"
            android:targetClass="com.globalmems.sensor.level.app.Level" />
</PreferenceScreen>
```

2. File ”$Android\packages\apps\Settings\res\values\strings.xml”

Add the following code to link @string

```
<string name="gsensor_settings">Gsensor Calibration</string>
<string name="gsensor_settings_summary">Please keep the device in a horizontal desktop</string>
```

3. Modify Level AndroidManifest.xml

```
<category android:name=”android.intent.category.LAUNCHER”/>
<category android:name=”android.intent.category.DEFAULT”/>
```

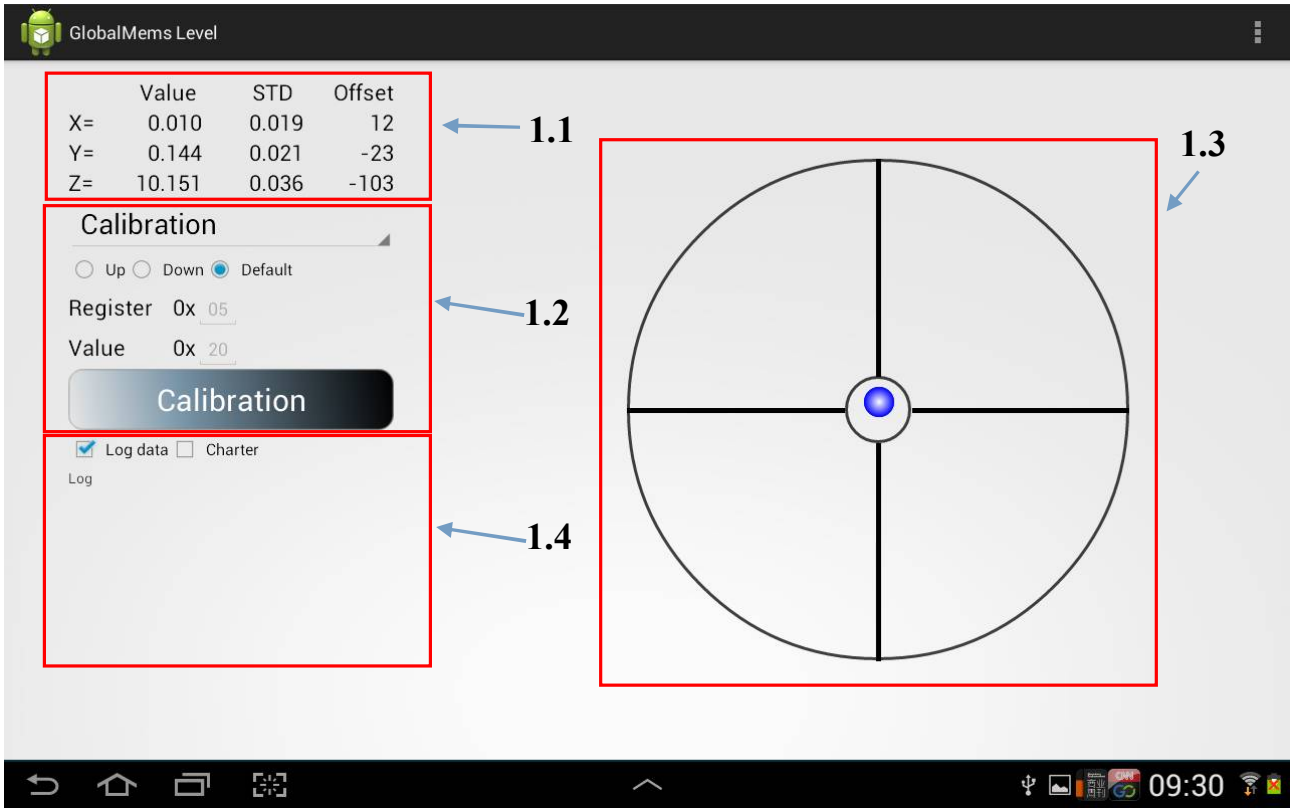
LAUNCHER : The apk on behalf of the show on the desktop icon.

DEFAULT : The apk on behalf of the show in the setting -> display.

4. Rebuilding “GLevel.apk”,and Install

Usage:

1.User Interface:



1.1 Sensor value display:

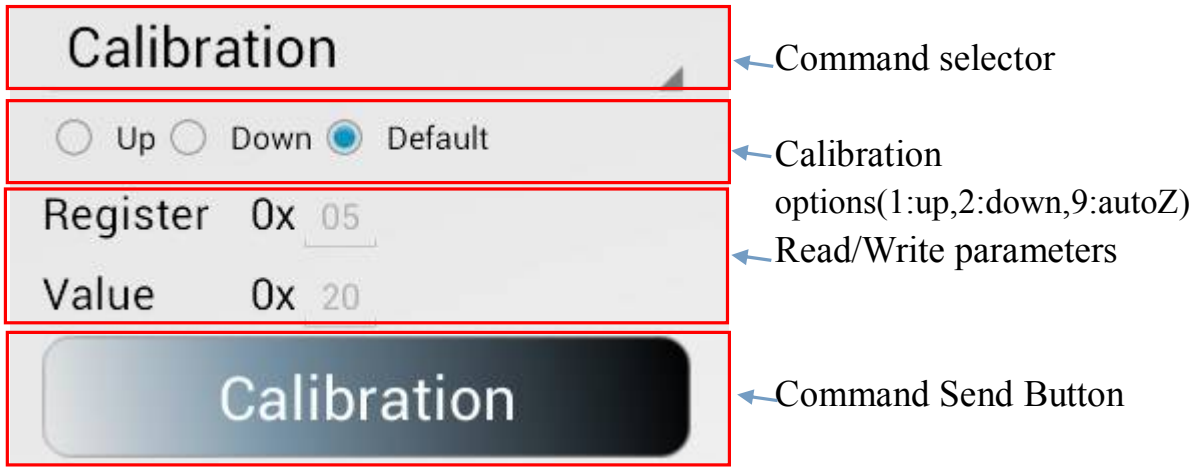
	Value	STD	Offset
X=	0.010	0.019	12
Y=	0.144	0.021	-23
Z=	10.151	0.036	-103

1.1.1 Value: Gravity data in $m/\sec^2$

1.1.2 STD: Standard deviation

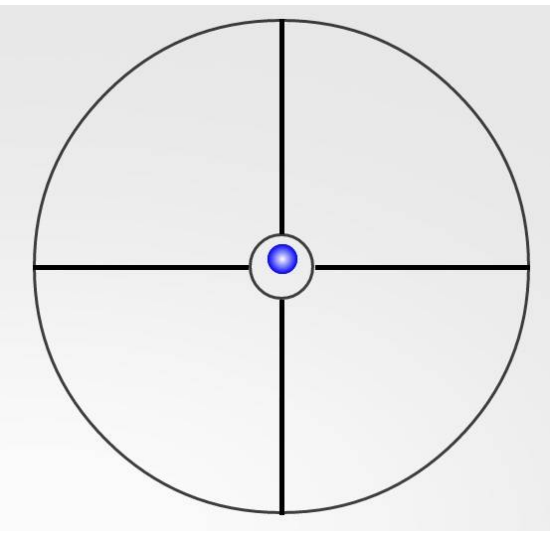
1.1.3 Offset: Sensor Calibration offset

1.2 User Commands

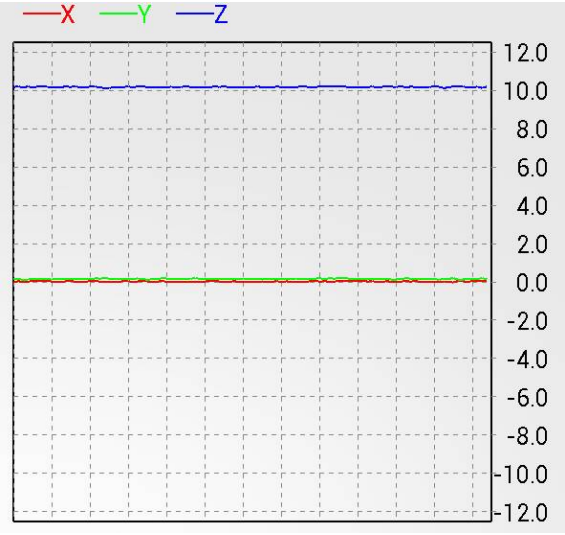


1.3 Graphic display

Check the Charter option to switch 2 graphic (bubble and charter) mode to show the sensor data.

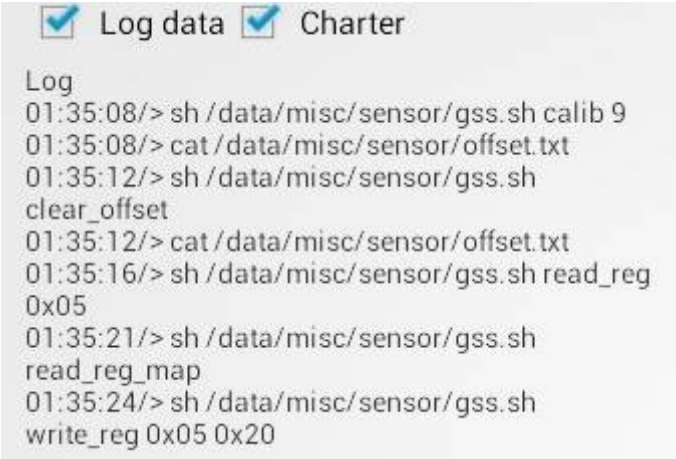


Bubble mode



Charter mode

1.4 Log data: Shows user command and return message in text. Check Log data Option to clean log data.



2.User Commands:

Click on Command selector, select a command on the menu pop out. Both the Command selector and Command Send button shows the command name.

After Select Command, click Command Send Button, app will execute user's command. There are 5 commands :

1. Calibration: Depends on purpose , user can select (1)Up, (2)Down, (3)Default options for (1) Device face up, (2) Device face down, and (3)Device face down Calibration respectively.
   Click the Command send Button, App starts to calibrate the sensor. Offset values will show on Sensor value display area and saved in the Offset.txt file.
2. Reset Offset: Clear the offset value.

3.Read Register: Read value of the register user selected.
Set the register address in Hex in the Read/Write parameter, Click Command send Button, the value of the register will show on the screen.

4.Read Register Map: Take a snapshot of the register map.

5.Write Register: Write value to the register user selected.
Set the register address and value in Hex in the Read/Write parameter, Click Command send Button, value will be write in to the register address, then it will be read back and show on the screen in red color.