Optical Sensors

Application Note

VCNL4020 Software for Android Platform

This document is supplied with Vishays Android software solution for the VCNL4020. This complete software package comes with all the necessary parts to simplify the integration of the VCNL4020 into an Android platform running Android version 4.0.1 or later. For demonstration purposes, the BeagleBoard-xM running Android v4.0.1 was used as a reference platform. If the same hardware is to be used, further information regarding where to purchase such a BeagleBoard-xM can be found here.

The provided zip file "**VCNL4020_Android-SW v 1.0.zip**", depicted in Figure 2, contains the documentation, source, and patch files required to allow an Android-based platform to communicate with the VCNL4020.This communication follows the standard Android standard sensors framework, as depicted in Figure 1.
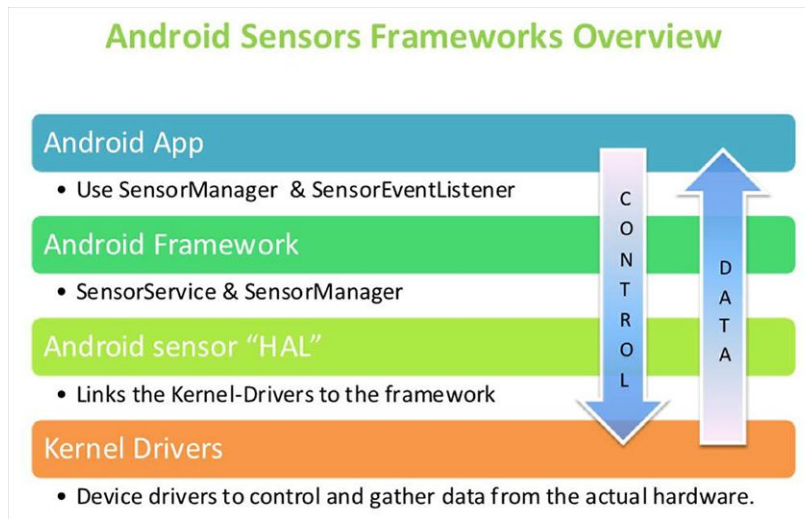

Fig. 1

The supplied Android VCNL4020 driver consists of two parts: the kernel module responsible for the data control and acquisition from the VCNL4020; and the subsequent HAL (Hardware Abstraction Layer) module to link the kernel to the main Android framework, allowing the information supplied by the sensor to be used by Android applications. To demonstrate this, a demo Android app has been included (**VCNL40x0-DemoApp.apk**), along with its associated source code, which can serve as a base upon which user-specific Android applications can be built.

APPLICATION NOTE

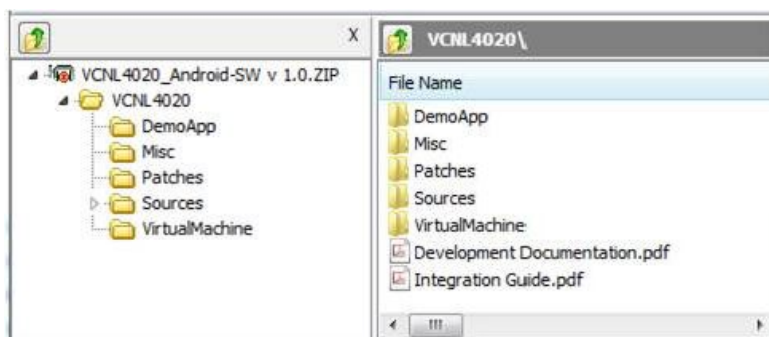# VCNL4020 Software for Android Platform

Fig. 2

DOCUMENTATION

The provided documentation files are **Development Documentation.pdf** and **Integration Guide.pdf**.

The Development Documentation file describes how the sources are to be compiled in order to be used with the reference platform. When successfully compiled, the kernel and HAL modules are generated. In order to further simplify the build process, a build script is supplied, which is also documented in this file.

The Integration Guide file describes how the VCNL4020 is to be integrated with the BeagleBoard xM. A description of both the kernel and the user space Android HAL module required for the integration is supplied. Furthermore, the document supplies information regarding the source files used to build these modules, which are also included in the zip file. The modules patches and the corresponding sources are supplied both in the form of the integration patch for all device repositories and specifically for the integration into the HAL of the reference repository. Information on how to apply these patches is also given in the guide.

INCLUDED FILES

Figure 3 includes an overview of the files supplied with the VCNL4020 software package.
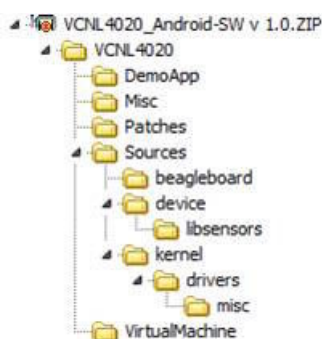

Fig. 3

The DemoApp folder contains the demonstration app along with its source code. This is to serve as a base to help Android application developers communicate with the sensor.

The Sources folder contains all the necessary sources in order to compile the required Android kernel and HAL patches. Further information regarding the files located in the sub folders can be found in the Integration Guide. The current versions of the compiled patches can be found in the Patches folder. If the source code is expanded and the source needs to be recompiled, the build script used to ease this process is located in the Misc folder.

The remaining VirtualMachine folder contains an incomplete virtual machine for our set-up build environment. The folder is incomplete as it is only needed for our own evaluation with the BeagleBoard-xM reference platform. The entire environment has not been included as the zip file size would increase to about 50 GB. However, if the build environment is required, it can be sent upon request and will be delivered on a 64 GB USB stick. Please contact sensorstechsupport@vishay.com.

APPLICATION NOTE

VCNL4020 Software for Android Platform

BEAGLEBOARD-XM EXPANSION

The development was done using the BeagleBoard-xM as a reference platform, with the VCNL4020 sensor board connected via the expansion connector. This connector provides the I^2C lines required to interface with the VCNL4020. The set-up is shown in Figure 4.
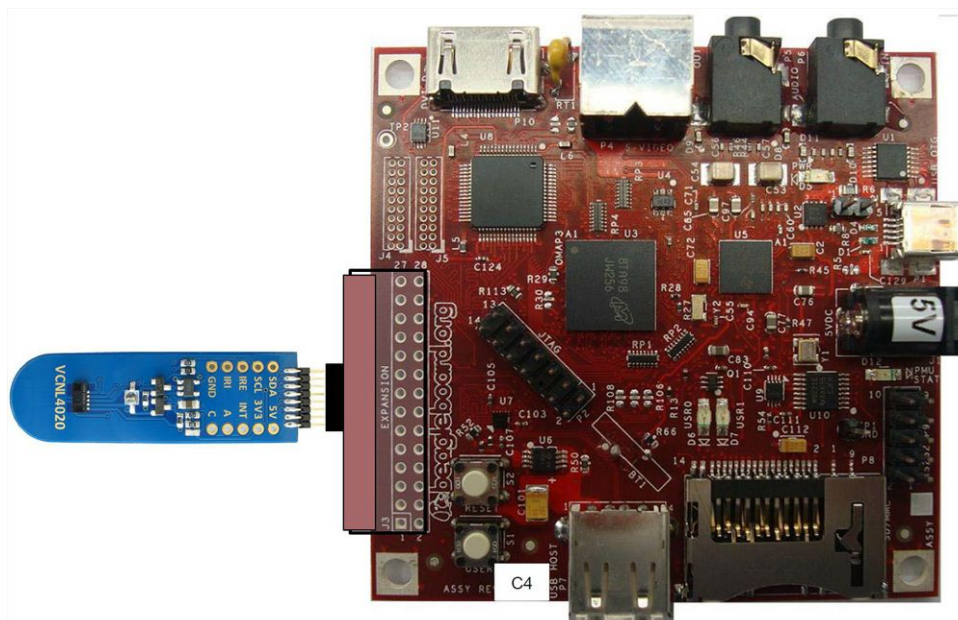

Fig. 4

This interconnection board takes care of the I^2C bus as well as the needed power supply. The 3.3 V supply voltage for the sensor is created by a small voltage regulator tied to 5 V. The necessary I^2C bus pull-up resistors are connected to 1.8 V. The connections are depicted in Figure 5.
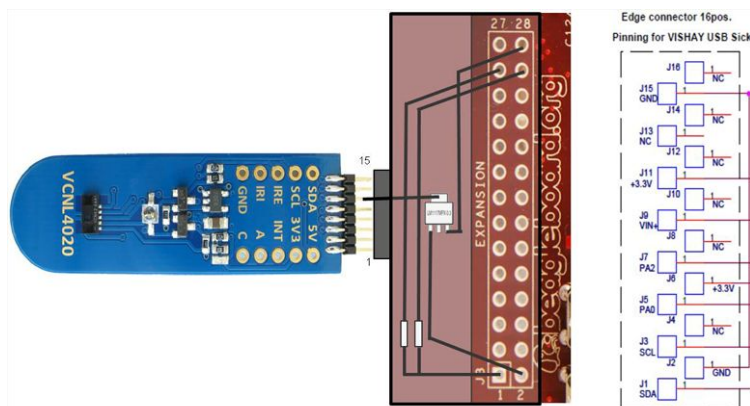

Fig. 5

For technical questions, contact: sensorstechsupport@vishay.com

VCNL4020 Software for Android Platform

VCNL40x0 DEMO APP

The Android software package comes with an Android demonstration app, **VCNL40x0-DemoApp.apk**, which can be run on any Android platform running v.4.0.1 or higher. It is to be used both for testing purposes and by developers as a base for building further Android applications.

To install this .apk, simply use an Android file manager such as "ASTRO File Manager" or copy the file onto a web server in order to open the file via a link directly from the phone. The Android Debug Bridge can also be used with the command: *adb install -r <application.apk>*.

After successful installation, the icon shown in Figure 6 will be added to your home screen.

Fig. 6

Selecting the icon will launch the application and take you to the "**Proximity**" tab, as shown in Figure 7.
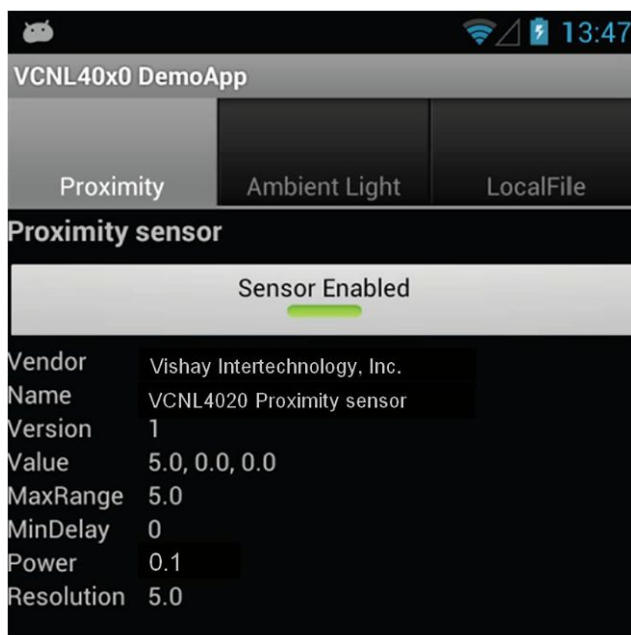

Fig. 7

The displayed information shows that the device is successfully communicating with the VCNL4020 device. The "Value" is an indication of the information sent to the app through the Android sensor manager. This value changes to 0.0 when an object is within 5 mm of the sensor, as a proximity event has been through the Android sensor API. The "Sensor Enabled" button registers the app for system updates.

VCNL4020 Software for Android Platform

Selecting the "**Ambient Light**" tab will display the ambient light menu, as shown in Figure 8.

Fig. 8

Again the sensor information is read and displayed and the "Sensor Enabled" button registers the app for system updates. The "Value" field continuously updates with the current ambient light value as read by the VCNL4020.

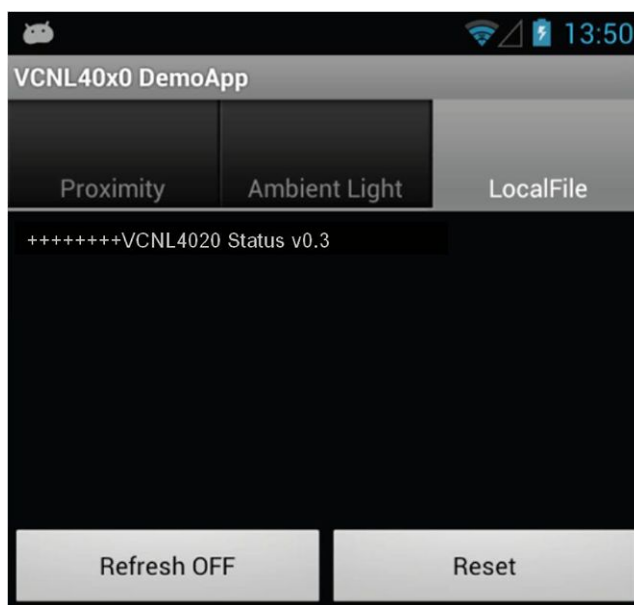The "**LocalFile**" tab brings up a blank screen unless the VCNL4020 is being used (Figure 9).

Fig. 9

For technical questions, contact: sensorstechsupport@vishay.com

THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE. THE PRODUCTS DESCRIBED HEREIN AND THIS DOCUMENT
ARE SUBJECT TO SPECIFIC DISCLAIMERS, SET FORTH AT www.vishay.com/doc?91000

APPLICATION NOTE

VCNL4020 Software for Android Platform

If set up in the same way as our reference platform, and applying the supplied patches to the set up reference platform, the LocalFile menu displays what is shown in Figure 10.
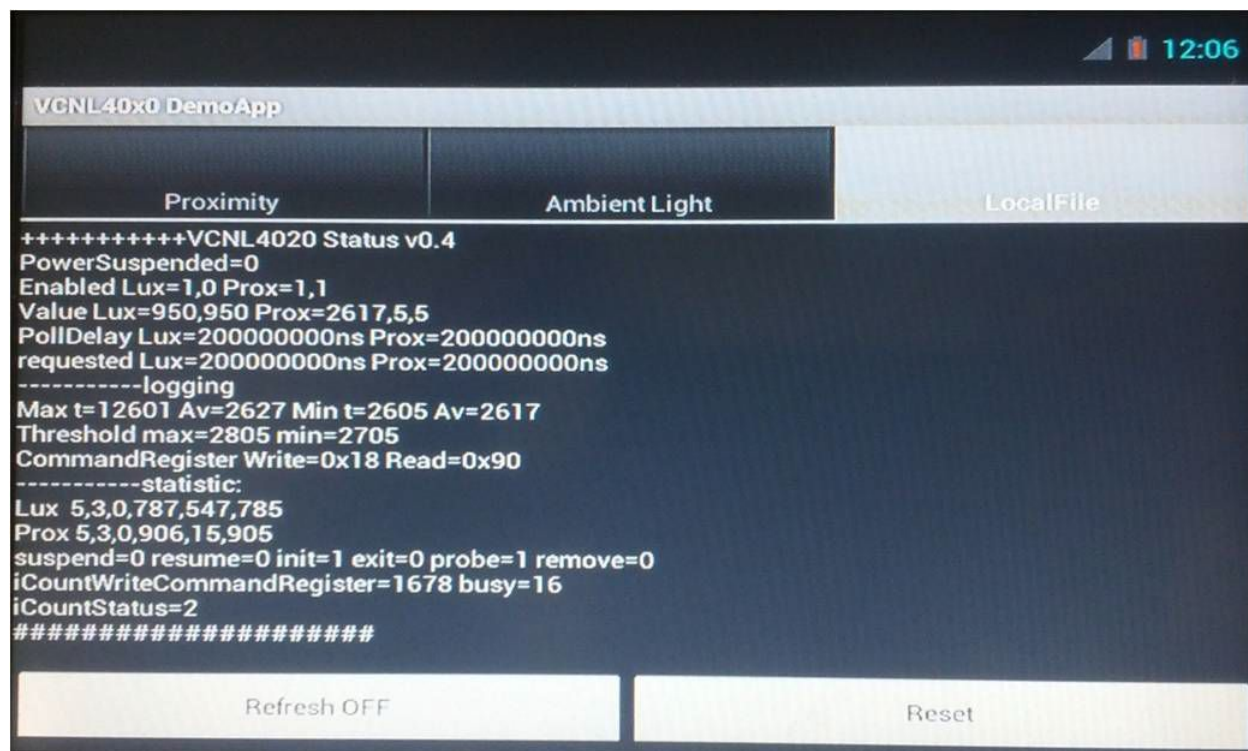

Fig. 10

Here the current status of the proximity and ambient light measurements are displayed, along with the corresponding threshold values. If the "Refresh" button on the bottom left is toggled ON, these values are continuously updated and the latest values read by the VCNL4020 sensor are shown.

The "Reset" button resets the values along with the threshold values. For example, if the environment is changed by placing glass over the sensor - as it would be in a mobile phone - the "Reset" button can be pressed after the glass has been put into place. The proximity detection in the Proximity menu will now react to the newly set threshold, taking the new offset values into consideration.

The new threshold values are taken and the new upper limit is set 200 counts above this. The new lower limit used to switch on again is taken as half the upper limit counts (i.e. 100 counts). These values are displayed as "Threshold max" and "Threshold min" in the logging data.

Further information such as the amount of measurements taken, "iCountStatus", or an average of the measured proximity values, "Av", is also displayed.

REMARK:

This Android software may also be used for the VCNL4010 without any changes and for the VCNL3020, where just the ambient Light tab is not applicable.

APPLICATION NOTE