



Intel® INDE Visual Coding Framework Android* Samples Guide

Version Beta-1 2016

LEGAL DISCLAIMER

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS OTHERWISE AGREED IN WRITING BY INTEL, THE INTEL PRODUCTS ARE NOT DESIGNED NOR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE INTEL PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or by visiting [Intel's Web Site](#).

MPEG is an international standard for video compression/decompression promoted by ISO. Implementations of MPEG CODECs, or MPEG enabled platforms may require licenses from various entities, including Intel Corporation.

Intel and the Intel logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

*Other names and brands may be claimed as the property of others.

Copyright © 2007-2015, Intel Corporation. All Rights reserved.

Optimization Notice

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel.

Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804

Contents

1	Overview	5
2	Sample code Inventory	5
3	Requirements.....	5
4	Build and Execute Instructions.....	6

1 Overview

The Intel® INDE Visual Coding Framework (VCF) Software Development Kit (SDK) is a software development library that enables programmatic access to the VCF cross-platform runtime.

This document provides an overview of the available samples for Android* OS/platforms. Please refer to corresponding documents for sample guides for other OS/platforms.

For general information about VCF, please refer to the following web page:

<https://software.intel.com/en-us/visual-coding-framework>

All VCF sample code is hosted on GitHub repository: <https://github.com/INDExOS/visual-coding-framework>

The VCF API currently supports language bindings for C and C++.

The sample code requires input such as media content and VCF GraphML input files. These files are located in the “**content**” folder.

2 Sample code Inventory

Sample Name	Description
simple_transcode	<p>This sample showcases basic usage of the C++ VCF API. The sample has a simple Java class that calls a native method to run a simple transcode workload, by loading and running a video only transcoding graph. (File->Decode->Encode->FileOut)</p> <p>The sample utilizes the following files from the content folder:</p> <ul style="list-style-type: none">- tos1152x480_1000.264- android_vcf_transcode.graphml
VCFPlayer	<p>This sample showcases VCF video decode and rendering. Note that video rendering requires the application to create a SurfaceView and provide a reference of this surface to the VCF RenderNode. In addition to this, this sample also showcases how to subscribe to VCF events.</p> <p>The sample utilizes the following files from the content folder:</p> <ul style="list-style-type: none">- tos1152x480_1000.264- android_vcf_decode_render.graphml

3 Requirements

Software

- Microsoft Windows* 7 or 8
- Android* Studio 1.3 or up - <https://developer.android.com/sdk/index.html>
- Android* NDK - <https://developer.android.com/ndk/downloads/index.html>
 - Download and Install Android NDK r10d or better
 - Add Path to ndk-build to Windows Environment Variables.

Hardware

- Host Development Environment:
 - 3rd generation Intel® Core™ processor and Intel® micro architecture (code name Ivy Bridge) or better.
- Target Device:
 - Android* KitKat* (4.4) or Android* Lollipop* (5.x) on Intel® Atom™ with Intel® Processor Graphics and Intel® Quick Sync Video.

Environment

- All samples depend on the "INTELVCFSDKROOT" environment variable to be set. This variable is set automatically when installing VCF. Users may also set this variable manually via Control Panel/System/Advanced System Settings/Environment Variables.
The variable must point to the install location of the VCF SDK. E.g.
"C:\Intel\INDE\Visual_Coding_Framework_<version>\sdk\"
- A reboot of the system is recommended in order for the Environment Variables to take effect.

4 Build and Execute Instructions

Complete the following steps to build Android samples:

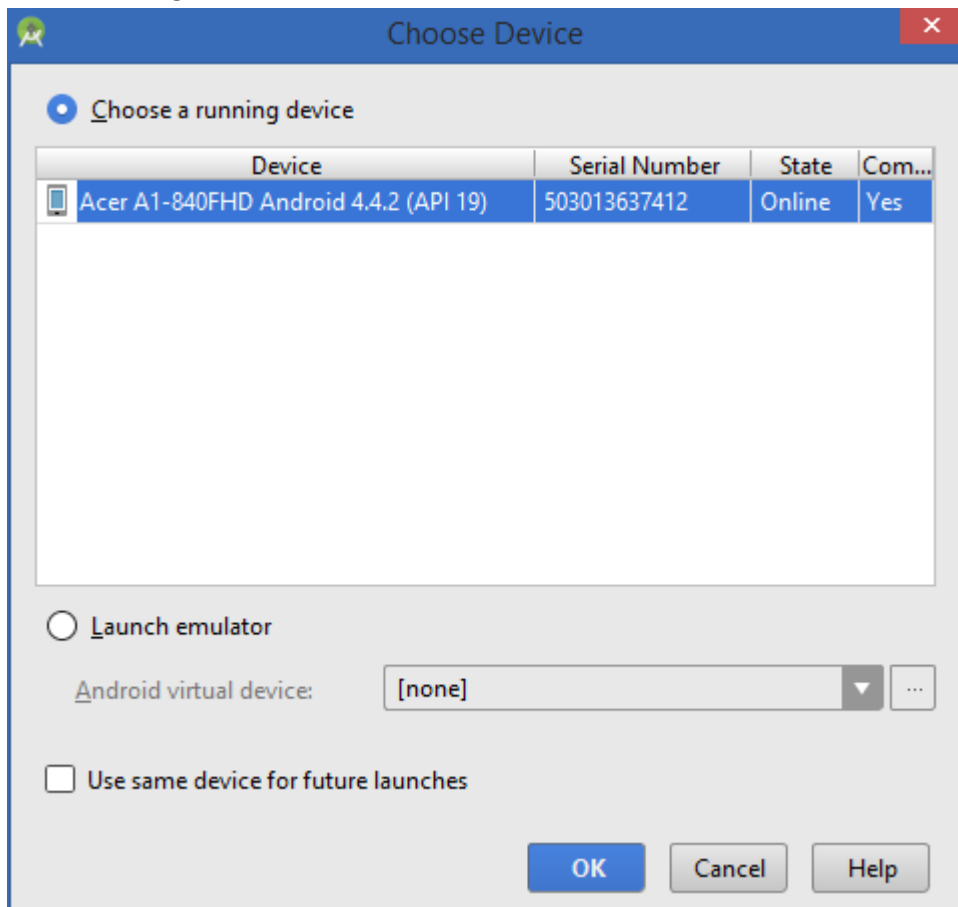
1. Connect the Target Device to the Development machine using USB.
2. Copy `tos1152x480_1000.264`, `android_vcf_transcode.graphml`, and `android_vcf_decode_render.graphml` to the target device "`/sdcard/Movies/Content`" directory.
3. Start Android Studio and Open the Sample project
4. In Android Studio, go to Terminal tab at the bottom (Alt+F12)
5. Navigate to the "jni" directory and build native lib as shown

```
Terminal
+ Microsoft Windows [Version 6.3.9600]
X (c) 2013 Microsoft Corporation. All rights reserved.

C:\Work\VCFPlayer>cd app\src\main\jni

C:\Work\VCFPlayer\app\src\main\jni>ndk-build
```

6. Run the app by pressing the Play button
7. Select the Target Device



8. When the App is running on the Target Device, observe the Logcat in Android Studio

Sample Logcat output:

```
sample.vcf.intel.com.vcfplayer V/VCFPlayerActivity: Play
sample.vcf.intel.com.vcfplayer V/vcfplayer: native_play
sample.vcf.intel.com.vcfplayer V/vcfplayer: native_decode
```

```
sample.vcf.intel.com.vcfplayer V/VCFCALLBACK: StatusProgress : 24
sample.vcf.intel.com.vcfplayer V/VCFCALLBACK: StatusProgress : 112
sample.vcf.intel.com.vcfplayer V/VCFCALLBACK: StatusProgress : 200
sample.vcf.intel.com.vcfplayer V/VCFCALLBACK: StatusProgress : 288
sample.vcf.intel.com.vcfplayer V/VCFCALLBACK: StatusProgress : 376
sample.vcf.intel.com.vcfplayer V/VCFCALLBACK: StatusProgress : 464
sample.vcf.intel.com.vcfplayer V/VCFCALLBACK: StatusProgress : 555
sample.vcf.intel.com.vcfplayer V/VCFCALLBACK: StatusProgress : 648
sample.vcf.intel.com.vcfplayer V/VCFCALLBACK: StatusProgress : 792
sample.vcf.intel.com.vcfplayer V/vcfplayer: native_stop
sample.vcf.intel.com.vcfplayer V/VCFCALLBACK: Benchmark Data :
status:0,execTimeS:36.4102,initTimeS:0.128023,numPackets:796,execFps:2
1.862,latencyAvgMs:63.5234,latencyMaxMs:136.855,latencyMinMs:44.9892,
sample.vcf.intel.com.vcfplayer V/VCFCALLBACK: Execution stopped
sample.vcf.intel.com.vcfplayer V/VCFCALLBACK: StatusProgress : 796
sample.vcf.intel.com.vcfplayer V/VCFPlayerActivity: Stop
```