Intel® Media Software Development Kit for Linux* Servers
Sample Guide

Overview

The **Intel® Media Software Development Kit for Linux* Servers (Intel® Media SDK) Samples** demonstrate how to incorporate the Intel Media SDK into an application. Included are:

- Command-line (console) Decoding from an elementary stream to raw (uncompressed) frames
- Command-line Encoding from raw frames to an elementary stream
- Command-line Video Processing to and from raw frames
- Command-line Transcoding from and to an elementary stream
- Command-line Video Conferencing sample (encoding with Video Conferencing features control)
- Command-line Full Transcoding Sample which shows complete media files processing, including container splitting and muxing, audio and video transcoding
- Splitters and Muxers Sample Library which shows how to use Intel Media SDK Splitter and Muxer API using the example FFmpeg* implementation wrapper
- Command-line Decoding with Video Processing

Each sample includes:

- A readme file. Read this material first and follow the instructions therein. The readme file is located in the first folder listed in boldface type in the Sample Location table.
- Source and header files for the samples.

Software Requirements

See `<install-folder>/mediasdk_release_notes.pdf` for Intel® Media SDK general requirements. To build Samples you additionally need the following components to be installed and properly configured on the system:

For **Ubuntu* 12.04**:

```
$ sudo apt-get install gcc g++ make cmake perl xserver-xorg-dev
```

For **SUSE* Linux Enterprise Server (SLES) 11 SP3**:

```
$ sudo zypper install gcc46 gcc46-c++ make cmake perl xorg-x11-devel
$ ln -s gcc-4.6 gcc
$ ln -s g++-4.6 g++
```

Samples can be built with GCC/G++ compiler version 4.6 and CMake* version 2.6.2 or higher.

For Splitters and Muxers Sample and Full Transcoding Sample you will also need several additional dynamic libraries which are the part of FFmpeg* codec libraries, particularly:

libavutil, version 52.38.100

libavcodec, version 55.18.102

libavformat, version 55.12.100

You can install them from the package manager or build from sources. Please, check the official compilation guide at https://trac.ffmpeg.org/wiki/CompilationGuide for build instructions.

Sample Locations

Material for each sample application is located in the following folder:

Sample	Location
Console Decoding	**<install-folder>/samples/sample_decode**
Console Encoding	**<install-folder>/samples/sample_encode**
Console Video Processing	**<install-folder>/samples/sample_vpp**
Console Transcoding	**<install-folder>/samples/sample_multi_transcode**
Full Transcoding	**<install_folder>/samples/sample_full_transcode**
Splitters and Muxers	**<install_folder>/samples/sample_spl_mux**
Video Conferencing	**<install_folder>/samples/sample_videoconf**
Console Decoding with VPP	**<install_folder>/samples/sample_decvpp**

Build Instructions

To build samples the following environment variable should be setup:

```
$ export MFX_HOME=/mediasdk/installation/folder
```

Go to the samples directory and execute build.pl script without arguments to see the help:

```
$ cd $MFX_HOME/samples
$ ./build.pl
Copyright (c) 2012 Intel Corporation. All rights reserved.
This script performs Intel(R) Media SDK Samples projects creation and build.

Usage: perl build.pl --cmake=ARCH,GENERATOR,CONFIG [--clean] [--build]

Possible variants:
        ARCH = intel64
        GENERATOR = make
        CONFIG = debug | release

Environment variables:
        MFX_HOME=/path/to/mediasdk/package # required
        MFX_VERSION="0.0.000.0000"         # optional

Optional flags:
        --clean - clean build directory before projects generation / build
        --build - try to build projects before generation (requires
cmake>=2.8.0)

Examples:
        perl build.pl --cmake=intel64,make,debug                 [ only
generate projects    ]
        perl build.pl --cmake=intel64,make,debug --build         [ generate
and then build   ]
        perl build.pl --cmake=intel64,make,debug --build --clean [ generate,
clean and build ]
```

Script invokes specified CMake* projects generator and optionally builds them (option available for cmake>=2.8.0). At the moment only make files generator for UNIX-like systems is supported. Project files will be placed in the folder named by the requested configuration; for example:

```
$ ls -1 $MFX_HOME/samples/__cmake
intel64.make.release
intel64.make.debug
```

To build generated project files use generator-specific approaches. For example, to build samples from make files invoke:

```
$ cd $MFX_HOME/samples/__cmake/intel64.make.release && make
# or
$ make -C $MFX_HOME/samples/__cmake/intel64.make.release
```

With CMake older than 2.8.0 all samples can be built at once with the following command:

```
$ ./build.pl --cmake=intel64,make,release --clean --build
```

Binaries will appear in the following folder:

```
$ ls -1 __cmake/intel64.make.release/__bin/release/
sample_decode_drm
sample_decode_x11
sample_decvpp_drm
sample_decvpp_x11
sample_encode_drm
sample_encode_x11
sample_multi_transcode_drm
sample_multi_transcode_x11
sample_videoconf_drm
sample_videoconf_x11
sample_vpp_drm
sample_vpp_x11
```

Samples with Hardware Acceleration support are buildable in a few variants depending on the availability of LibVA backends. For example:

- sample_decode_drm – sample variant with HW acceleration support to be run on the system without Graphic Server installation (i.e. LibVA DRM backend is used).

- sample_decode_x11 – sample variant with HW acceleration support to be run under X Server (i.e. LibVA X11 backend is used).

Running the Software

DRM backend specific notes

- For application to work thru DRM application should be authorized to access graphics card. VA-API DRM backend supports 2 authentication models:
    - The first model can be applied on the system with no installation of Graphic Server. In this case you need root privileges to run:

```
$ sudo LD_LIBRARY_PATH=$MEDIASDK_INSTALL_FOLDER/bin/x64 \
    sample_decode_drm h264 -i input.264 -o output.yuv -d3d -hw
```

    - The second model assumes that X server is installed and running. In this case DRM authentication will actually go thru LibVA X11 backend and, thus, thru X server which already has access to the graphic card. The only thing user should be sure in is that he is logged on to the X server (or has access) and DISPLAY environment variable is set properly. For example:

```
$ export DISPLAY=:0.0
$ sudo LD_LIBRARY_PATH=$MEDIASDK_INSTALL_FOLDER/bin/x64 \
    sample_decode_drm h264 -i input.264 -o output.yuv -d3d -hw
```

- It can be noted that DRM-itself authentication can still be tried out even with running X server, but you need to remove DISPLAY environment variable and use root privileges:

```
$ export -n DISPLAY
$ sudo LD_LIBRARY_PATH=$MEDIASDK_INSTALL_FOLDER/bin/x64 \
    sample_decode_drm h264 -i input.264 -o output.yuv -d3d -hw
```

X11 backend specific notes

- To use this backend user should be sure that he is logged into X server or is allowed to make connections to the X server.

- If user is allowed to use X and logged into machine remotely (thru SSH) he needs DISPLAY environment variable properly set. For example:

```
$ export DISPLAY=:0.0
$ sample_decode_x11 h264 -i input.264 -o output.yuv -d3d -hw
```

Legal Information

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.

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