1 Introduction

This sample demonstrates how to build a simple transcode pipeline and also demonstrates how to use AMD's Unified Video Decoder (UVD) for video decoding and the Video Coding Engine (VCE) for video encoding. The sample accepts an H264 elementary stream as input and produces an H264 elementary stream as output.

2 Using the sample

2.1 Location

`$<installDirectory>\samples\mediaFoundation\simpleTranscode\`

2.2 Contents

Package contents

Folder: `$<installDirectory>\samples\mediaFoundation\simpleTranscode\src\`

File name	Description
`AsyncState.cpp`	Contains CAsyncState class and functions
`CustomByteStreamHandler.cpp`	Contains CustomByteStreamHandle class and functions
`CustomMediaSink.cpp`	Contains CustomMediaSink class and functions
`CustomMediaSource.cpp`	Contains CustomMediaSource class and functions
`CustomSinkActivate.cpp`	Contains CustomSinkActivate class and functions
`CustomSource.cpp`	Contains CustomSource class and functions
`CustomStream.cpp`	Contains CustomStream class and functions
`CustomStreamSink.cpp`	Contains CustomStreamSink class and functions
`SimpleTranscodeConfig.cpp`	Contains functions for configuration parsing
`SimpleTranscodeMain.cpp`	Transcode main
`SimpleTranscodeSession.cpp`	Contains transcode session class and functions for building transcode pipeline
`SourceOperation.cpp`	Contains SourceOperation class and functions
`VideoInput.cpp`	Contains VideoInput class and functions

Folder:
`$<installDirectory>\samples\mediaFoundation\simpleTranscode\inc\`

File name	Description
`AsyncState.h`	Header file for CAsyncState class
`Common.h`	Header file which contains enumerated SourceStates
`CustomByteStreamHandler.h`	Header file for CustomByteStreamHandle class

Simple Transcode

CustomMediaSink.h	Header file for CustomMediaSink class
CustomMediasinkActivate.h	Header file for CustomMediaSource class
CustomMediaSinkGUIDs.h	Header file which contains GUID of custom media sink
CustomMediaSource.h	Header file for CustomMediaSource class
CustomSinkActivate.h	Header file for CustomSinkActivate class
CustomSource.h	Header file for CustomSource class
CustomStream.h	Header file for CustomStream class
CustomStreamSink.h	Header file for CustomStreamSink class
LinkList.h	Header file for linked list implementation
SimpleTranscodeApi.h	Header file for transcode API
SimpleTranscodeConfig.h	Header file which contains transcode configuration function definition
SimpleTranscodeSession.h	Header file for transcode session class
SourceOperation.h	Header file for SourceOperation class
VideoInput.h	Header file for VideoInput class

Folder:
$<*installDirectory*>\samples\mediaFoundation\simpleTranscode\config\

File name	Description
exampleConfig.cfg	Configuration file for encoder

Folder:
$<*installDirectory*>\samples\mediaFoundation\simpleTranscode\docs\

File name	Description
MediaSDK_MFT_simpleTranscode.pdf	Sample documentation

Folder:
$<*installDirectory*>\samples\mediaFoundation\simpleTranscode\build\windows\

File name	Description
simpleTranscodeVs10.sln	Visual Studio 10 solution file
simpleTranscodeVs10.vcxproj	Visual Studio 10 project file
simpleTranscodeVs10.vcxproj.filters	Visual Studio 10 project filter file
simpleTranscodeVs12.sln	Visual Studio 12 project solution file
simpleTranscodeVs12.vcxproj	Visual Studio 12 project file
simpleTranscodeVs12.vcxproj.filters	Visual Studio 12 project filter file

Simple Transcode

2.3   Parameters   Encoder-specific configuration parameters

Parameter name	Default value	Supported range	Remarks
encWidth	1920		Input H264 Elementary Stream Width (supports only input-stream resolution, not resizing)
encHeight	1080		Input H264 Elementary Stream Height (supports only input-stream resolution, not resizing)
encGOPSize	20		Max number of frames in a GOP (0=auto)
encMeanBitrate	3000000		Bitrate of encoded video (bits per second)
encMaxBitrate	4000000		Maximum bitrate of encoded video (used only for VBR) in bits per second
encBufferSize	2000000		VBR buffer size
encNumBFrames	1	0 - 3	Specifies the number of B frames to be inserted
encCompressionStandard	77	For supported values, see http://msdn.microsoft.com/en-us/library/windows/desktop/dd318776(v=vs.85).aspx	Compression standard
encRateControlMethod	1	`eAVEncCommonRateControlMode_CBR = 0,` `eAVEncCommonRateControlMode_PeakConstrainedVBR = 1,` `eAVEncCommonRateControlMode_UnconstrainedVBR = 2,` `eAVEncCommonRateControlMode_Quality = 3`	For more details, see http://msdn.microsoft.com/en-us/library/windows/desktop/dd388772(v=vs.85).aspx
encLowLatencyMode	0	`1 - True` `0 - False`	Specifies whether the output stream should be structured so that the encoded stream has a low decoding latency.
encQualityVsSpeed	60	`0 - Low quality faster encoding` `100 - Higher quality, slower encoding`	
encCommonQuality	50	`0 to 100` `0 - low quality` `100 - highest quality`	This parameter is used only when `encRateControlMethod` is set to `eAVEncCommonRateControlMode_Quality`. in this mode the encoder selects the bit rate to match the quality settings.

Common configuration parameters

Parameter name	Default value	Supported range	Remarks
useSWCodec	0	Enable=1 Disable=0	Enables/Disables the use of software Encoder and Decoder. If set to 0, hardware-based codecs are used to encode and decode the stream; otherwise software-based codecs are used.

2.4   Compile

1. Ensure that the following tools and SDKs are present:

    ☐   Microsoft Visual Studio 2010 or 2012
         If Windows Software Development Kit (SDK) is not installed, install it from
         http://msdn.microsoft.com/en-us/library/windows/desktop/hh852363.aspx.

2. Open one of the following solution files:
    ```
    $<installDirectory>\samples\mediaFoundation\simpleTranscode\build\win
    dows\simpleTranscodeVs12.sln
    $<installDirectory>\samples\mediaFoundation\simpleTranscode\build\win
    dows\simpleTranscodeVs10.sln
    ```

3. Build the sample:

    ☐   Open the `simpleTranscodeVs10.sln` solution file with Microsoft Visual Studio
        2010 Professional Edition or the `simpleTranscodeVs12.sln` solution file with
        Microsoft Visual Studio 2012 Professional Edition.

    ☐   To build all the solutions, select `Build > Build Solution`.

    ☐   Select the project file in the Solutions Explorer.
        The executable `simpleTranscode.exe` is created in the following folders for 32-bit
        builds and 64-bit builds:
        ```
        $<installDirectory>\samples\mediaFoundation\simpleTranscode\bin
        \x86\
        $<installDirectory>\samples\mediaFoundation\simpleTranscode\bin
        \x86_64\
        ```

3 Run

The sample can be executed on an AMD platform that includes the VCE and UVD hardware
blocks.

On the command prompt, change to the directory that contains the executable, and execute the
following command:
```
simpleTranscode.exe -i <input.h264> -o <output.h264> -c <configfile>
-l <0, 1, or 2 >
```

-l enables the logging. Setting "0" means no logging. "1" generates the log at the API level. "2"
generates logs at the transcoding session level.

The `$<installDirectory>\inc\ErrorCodes.h` file contains information about the error
codes. You can also print the Microsoft error codes by using the `getMicrosoftErrorCode()`

Simple Transcode

API, as shown in `SimpleTranscodeMain.cpp`. The Microsoft error codes can be understood from the `Mferror.h` file that Microsoft provides as part of its Windows kits.
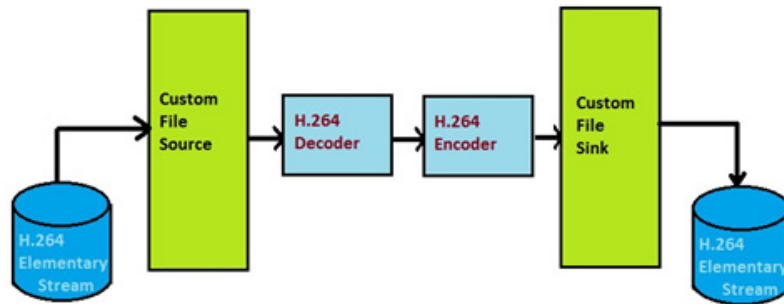
The configuration file associated with this sample is located in the following folder: $<*installDirectory*>\samples\mediaFoundation\simpleTranscode\config\

`Output.h264` contains the H264 encoded stream.

4 Implementation Details

The sample implements the following transcode pipeline:

5 Supported formats

The following file formats are supported:

- Input file format: H264 elementary stream
- Video decoder supported: H264
- Output file format: H264 elementary stream
- Video encoder supported: H264

A sample H264 elementary stream is available in $<*installDirectory*>\data\HSA_1280x720.h264.

For information about how to generate an elementary stream from container formats, see the *Media SDK FAQ* document.

6 Known limitations

The sample is currently supported on the following platforms:

- Windows 7 (DirectX 9)
- Windows 8/8.1 (DirectX 9 and DirectX 11)

Contact

Advanced Micro Devices, Inc.
One AMD Place
P.O. Box 3453
Sunnyvale, CA, 94088-3453
Phone: +1.408.749.4000

For AMD Accelerated Parallel Processing:

URL: **developer.amd.com/appsdk**
Developing: **developer.amd.com/**
Forum: **developer.amd.com/openclforum**

AMD