SAMPLE

OpenCL Video Filter

1 Introduction

The purpose of this sample is to demonstrate how to build and execute a Video Transcoding use-case with OpenCL video filtering using AMD Media Framework (AMF). The sample decodes H.264 elementary content, applies Sobel filtering on the decoded raw video, and re-encodes the decoded video to generate compressed H.264 Elementary stream.

2 Using the sample

2.1 Location

`$<installDirectory>\samples\amf\openCLVideoFilter\`

2.2 Contents

Package Contents

Folder:
`$<installDirectory>\samples\amf\openCLVideoFilter\src\`

File name	Description
`openCLVideoFilter.cpp`	Source file for OpenCL video filter
`main.cpp`	Source file for OpenCL Video Filter application
`SobelFilterLuma.cl`	OpenCL Video Filter Kernel

Folder:
`$<installDirectory>\samples\amf\openCLVideoFilter\inc\`

File name	Description
`openCLVideoFilter.h`	Header file for OpenCL video filter

Folder:
`$<installDirectory>\samples\amf\openCLVideoFilter\build\windows\`

File name	Description
`OpenCLVideoFilterVs10.sln`	Microsoft Visual Studio 10 solution file
`OpenCLVideoFilterVs10.vcxproj`	Microsoft Visual Studio 10 project file
`OpenCLVideoFilterVs10.vcxproj.filters`	Microsoft Visual Studio 10 project filter file
`OpenCLVideoFilterVs12.sln`	Microsoft Visual Studio 12 project solution file
`OpenCLVideoFilterVs12.vcxproj`	Microsoft Visual Studio 12 project file
`OpenCLVideoFilterVs12.vcxproj.filters`	Microsoft Visual Studio 12 project filter file

Folder:
$<*installDirectory*>\samples\amf\openCLVideoFilter\docs\

File name	Description
MediaSDK_AMF_openCLVideoFilter.pdf	Sample documentation

2.3 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:
   - `$<installDirectory>\samples\amf\openCLVideoFilter\build\windows\OpenCLVideoFilterVs12.sln`
   - `$<installDirectory>\samples\amf\openCLVideoFilter\build\windows\OpenCLVideoFilterVs10.sln`
3. Build the sample:
   - Open the `OpenCLVideoFilterVs10.sln` solution file with Microsoft Visual Studio 2010 Professional Edition or the `OpenCLVideoFilterVs12.sln` solution file with Microsoft Visual Studio 2012 Professional Edition.
   - To build all the solutions, select `Build > Build Solution`.
   - The executable `openCLVideoFilter.exe` is created in the following folders for 32-bit builds and 64-bit builds:
     `$<installDirectory>\samples\amf\openCLVideoFilter\bin\x86\`
     `$<installDirectory>\samples\amf\openCLVideoFilter\bin\x86_64\`
   - Depending on the build (i.e. 32-bit or 64-bit), the custom build step copies the appropriate `.dlls` file from the `$<installDirectory>\dll\amf\` folder into the relevant `\bin\` directory.

3 How to Run

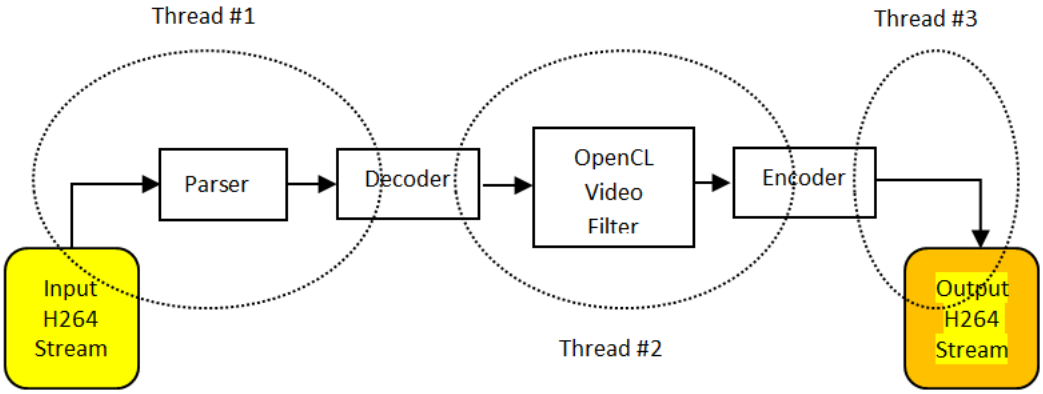
The sample can be executed on an AMD platform that includes the VCE and UVD hardware blocks.

On the command prompt, change to the directory that contains the executable, and execute the following command:
`openCLVideoFilter.exe <InputFile> <OutputFile> <MemoryType>`

InputFile: Specify the H.264 elementary stream filename or path
OutputFile: Specify the H.264 elementary stream output filename
BufferMemoryType: Specify the memory type, either `DX9` or `DX11`

4 Implementation Details

The sample implements the following OpenCL Video Filter topology:



The Data in the Video Filtering Pipeline flows through the following processing elements:

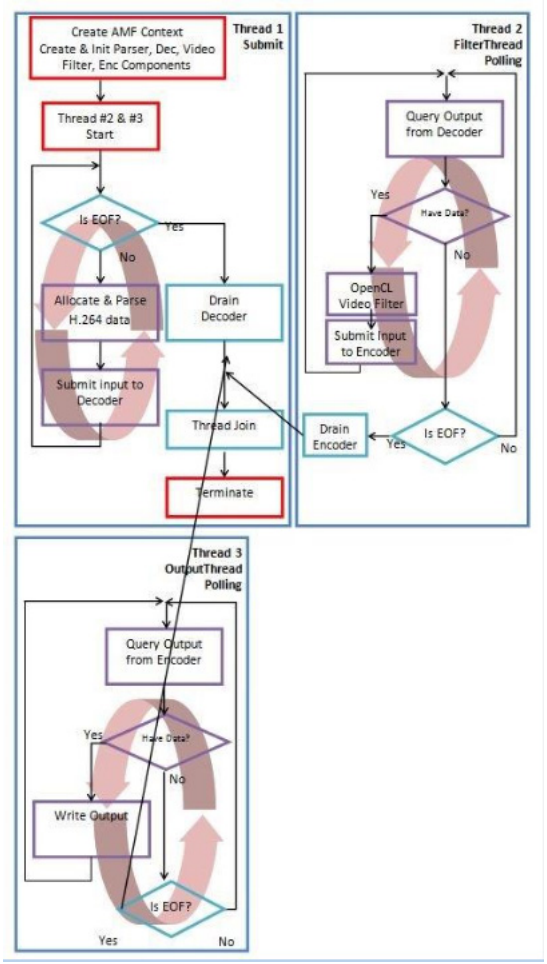
- Parser: H.264 Elementary stream data is first read by the parser which finds the SPS & PPS, finds NALU unit, and populate data structures which are fed to the H.264 Decoder.
- Decoder: HW Accelerated (UVD) H.264 Video Decoder. Decodes the input content to generate NV12 frames.
- OpenCL Video Filter: Applies Sobel filter on the video frames: http://en.wikipedia.org/wiki/Sobel_operator
- Encoder: HW Accelerated (VCE) H.264 Video Encoding. Encodes the input content to generate compressed H.264 Elementary stream.

Parameter Name	Description	Value
`frameRateOut`	Encoding Frame Rate	30
`bitRateOut`	Encoding Bitrate	1000000

The sample prints the following performance parameters:

- Latency in ms
- Average transcoding + Filtering time in ms / frame
- Average time in ms to write one transcoded frame to file

The sample is constructed using Native AMF APIs executing over three threads, as shown in the following figure.



5 Supported formats

The following file formats are supported:

- Video Transcoders supported: H.264
- Output file format: H.264 Compressed Elementary Stream

6 Known Limitations

The sample is currently supported on the following platforms:

- Windows 7 (DirectX 9)
- Windows 8.1 (DirectX 9 and DirectX 11.1)

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