
1 How to Build a Sample

Ensure that the AMD APP SDK Samples package has been installed before proceeding.

1.1 On Windows

Building With Visual Studio Solution Files –

The samples installation contains a Microsoft Visual Studio 2008 solution file (`OpenCLSamples.sln`) and a Microsoft Visual Studio 2010 solution file (`OpenCLSamplesVS10.sln`). These are located at `$(AMDAPPSDKSAMPLESROOT)\samples\opencl\`. The solution file contains the entire sample project. To build a sample:

1. Open the `OpenCLSamples.sln` file with Microsoft Visual Studio 2008 Professional Edition or open the or `OpenCLSamplesVS10.sln` with Microsoft Visual Studio 2010 Professional Edition.
2. Select *Build > Build Solution* to build all solutions.
3. Select the project file in the Solutions Explorer.
4. Right-click on the project file, and select *Build* to build a particular sample.

Building With VS Compiler Using `make` Files –

1. Open the Visual Studio command prompt. (Ensure `make.exe` is in the PATH. This executable can be from cygwin or MinGW. Open the 32-bit Visual Studio command prompt for 32-bit builds; open the 64-bit Visual Studio command prompt for 64-bit builds.)
2. Go to the `InstallSampleDir\samples` directory to build all samples. To build samples individually, go to the `InstallSampleDir\samples\opencl\cl\app` directory.
3. For 64-bit builds, use `make`; for 32-bit builds, use `make bitness=32`.

Building With Intel Compiler Using `make` Files –

1. Open the Intel Visual Studio command prompt. (Ensure `make.exe` is in the path. This executable can be from cygwin or MinGW. Open the 32-bit Intel Visual Studio command prompt for 32-bit builds; open the 64-bit Intel Visual Studio command prompt for 64-bit builds.)
2. Go to the `InstallSampleDir\samples` directory to build all samples. To build samples individually, go to the `InstallSampleDir\samples\opencl\cl\app` directory.
3. At the command line, enter `make clean`.
4. Build the sample using the command `make intel=1 sample_Name`.

5. For 64-bit systems use `make intel=1` to build 64-bit targets, and `make intel=1 bitness=32` to build 32-bit targets. (If your OS is 64-bit, it builds 64-bit targets by default, but to build 32-bit targets, it is necessary to explicitly specify `bitness=32`.)

Building With MinGW + GCC Using `make` Files –

1. Open the command window (Start > Run, then type `cmd.exe`).
2. Add MinGW bin and Msys bin to PATH:
`PATH=C:\MinGW\bin;C:\MinGW\msys\1.0\bin;%PATH%`
3. Go to the `InstallSampleDir\samples` directory.
4. `make bitness=32`.

Building With MinGW-x64 + GCC Using `make` Files –

1. Open the command window (Start > Run, then type `cmd.exe`).
2. Add MinGW64 bins to PATH: `PATH=C:\MinGW64\bin;C:\MinGW\msys\1.0\bin;%PATH%`
3. Go to the `InstallSampleDir\samples` directory.
4. `make bitness=64`.

Note: To use Msys under MinGW64-x64:

- a. Open the `fstab` file (available at `msys/1.0/etc/`)
- b. Modify according to your MinGW64-w64 path. For example, modify
`C:\MinGW\ /mingw` to `C:\MinGW64\ /mingw`

1.2 On Linux

Building With GCC –

This samples installation contains makefiles to build samples. To build a sample:

1. Ensure the library path is set to find `libOpenCL.so`. If it is not set, follow the instructions in the *AMD APP SDK Installation Notes*, section 2.2, step 4.
2. Go to the `samples` directory if you want to build all the samples, or go to a particular sample directory, and type `make`. If no options are specified, `make` builds for the native platform in debug configuration. To select release configuration, add the option `release = 1`. To force a 32-bit build on a 64-bit platform, add the option `bitness=32` .
3. Executables are generated in the `bin` directory at the samples installation root, as well as inside the `bin` directory of the samples: `AMDAPPSDKSAMPLESROOT/samples/opencv/bin`.

Building With the Intel Compiler –

1. Use the option `intel=1`; all other options are same as above.

2 How to Run the Application

This section describes how to run the application that was just built.

2.1 On Windows

There are three ways to run the application: by using Microsoft Visual Studio 2008 Professional Edition, or using Microsoft Visual Studio 2010 Edition, or by using the command line.

Using Visual Studio

1. Open `OpenCLSamples.sln` with Microsoft Visual Studio 2008 Professional Edition, or `OpenCLSamplesVS10.sln` with Microsoft Visual Studio 2010 Professional Edition, and build it.
2. Select the desired project file in the Solutions Explorer.
3. Right-click on it, and select *Set as StartUp Project*. To run the application, press Ctrl+F5. To run the application in debug mode, simply press F5.

Using the Command Line

1. Open a command prompt.
2. Go to the `$(AMDAPPSDKSAMPLESROOT)/samples/opencv/bin.`
3. Go into the appropriate architecture directory (`x86` or `x86_64`).
4. Run the samples by typing the name of their executables. See the individual sample documents for their respective command line arguments.

2.2 On Linux

1. Ensure the path is set to include the location of `libOpenCL.so`.
2. Open a terminal window.
3. Go to the `$(AMDAPPSDKSAMPLESROOT)/samples/opencv/bin/` directory.
4. Go into the appropriate architecture directory (`x86` or `x86_64`).
5. Run the samples by typing the name of their executables. You may have to prepend the executable name with `./`. See the individual sample documents for their respective command line arguments.

2.3 Sample Code

The simplest OpenCL sample in the SDK is the *Template* sample. It is for developers that are new to OpenCL programming. See the Template sample documentation (in `$(AMDAPPSDKSAMPLESROOT)/samples/opencv/cl/app/Template/docs`), which explains the workflow for setting up a basic AMD APP application using OpenCL. The *Template* sample is included for instructional purposes.

There are two kinds of OpenCL samples in the AMD APP SDK. One is written using native OpenCL calls (in `$(AMDAPPSDKSAMPLESROOT)/samples/opencv/cl`); the other is written using the AMD C++ bindings to OpenCL (in `$(AMDAPPSDKSAMPLESROOT)/samples/opencv/cpp_cl`).

Most of the OpenCL samples make use of a utility library called the AMD APP SDK Utility Library (SDKUtil). This library provides commonly used routines, such as parsing command line options,

loading and writing bitmap images, printing formatted output, comparing results, and reading files. This is an object-oriented library that is structured as a set of classes. The SDKUtil `cpp` files and headers are available in the `$(AMDAPPSDKSAMPLESROOT)\samples\opencl\SDKUtil` folder.

3 Important Notes

- For SDK 2.6, the following values are returned when querying strings from OpenCL:

`CL_PLATFORM_VERSION`: OpenCL 1.1 AMD-APP (*build.revision*)

`CL_PLATFORM_NAME`: AMD Accelerated Parallel Processing

`CL_PLATFORM_VENDOR`: Advanced Micro Devices, Inc.

- Check the Platform Vendor string, not the Platform Name, to determine AMD hardware. For example code that shows how to check and use the `CL_PLATFORM_VENDOR` string, see the AMD APP v2 Samples.

4 Supported Devices

The following table shows the devices supported by the current version of the AMD APP SDK, as well as the computing features they support. (* = beta only.)

Product Line	Card	Double-Precision	UAV	Compute Kernel (Shader)
AMD APU Family with AMD Radeon™ HD	C-30 with HD 6250	No	Yes	Yes
	C-50 with HD 6250	No	Yes	Yes
	E-240 with HD6310	No	Yes	Yes
	E-350 with HD 6310	No	Yes	Yes
		No	Yes	Yes
	E2-3000M with HD 6380G	No	Yes	Yes
	A8-3530MX with HD6620G	No	Yes	Yes
	A8-3510MX with HD6620G	No	Yes	Yes
	A8-3500MX with HD6620G	No	Yes	Yes
	A6-3410MX with HD6520G	No	Yes	Yes
	A6-3400MX with HD6520G	No	Yes	Yes
	A4-3310MX with HD6480G	No	Yes	Yes
	A4-3300MX with HD6480G	No	Yes	Yes
		No	Yes	Yes
	A8-3850 with HD6550D	No	Yes	Yes
	A8-3800 with HD6550D	No	Yes	Yes
	A6-3650 with HD6530D	No	Yes	Yes
	A6-3600 with HD6530D	No	Yes	Yes
	A4-3400 with HD6410D	No	Yes	Yes
	E2-3200M with HD 6370D	No	Yes	Yes
AMD Radeon™ HD	6990	Yes	Yes	Yes
	6970	Yes	Yes	Yes
	6950	Yes	Yes	Yes
	6870	No	Yes	Yes
	6850	No	Yes	Yes
	6790	No	Yes	Yes
	6770	No	Yes	Yes
	6750	No	Yes	Yes
	6670	No	Yes	Yes
	6570	No	Yes	Yes
	6450	No	Yes	Yes

Product Line	Card	Double-Precision	Global Buffer	Compute Kernel (Shader)
<i>ATI Radeon™ HD</i>	5970	Yes	Yes	Yes
	5870	Yes	Yes	Yes
	5850	Yes	Yes	Yes
	5830	Yes	Yes	Yes
	5770	No	Yes	Yes
	5750	No	Yes	Yes
	5670	No	Yes	Yes
	5570	No	Yes	Yes
	5470	No	Yes	Yes
	4890 *	Yes	Yes	Yes
	4870 X2 *	Yes	Yes	Yes
	4870 *	Yes	Yes	Yes
	4850 X2 *	Yes	Yes	Yes
	4850 *	Yes	Yes	Yes
	4830 *	Yes	Yes	Yes
	4770 *	No	Yes	Yes
	4670 *	No	Yes	Yes
	4650 *	No	Yes	Yes
	4550 *	No	Yes	Yes
	4350 *	No	Yes	Yes
<i>ATI FirePro™</i>	V8800	Yes	Yes	Yes
	V8750 *	Yes	Yes	Yes
	V8700 *	Yes	Yes	Yes
	V7800	Yes	Yes	Yes
	V7750 *	No	Yes	Yes
	V5800	Yes	Yes	Yes
	V5700 *	No	Yes	Yes
	V4800	Yes	Yes	Yes
	V3800	Yes	Yes	Yes
	V3750 *	No	Yes	Yes
<i>AMD FireStream™</i>	9270 *	Yes	Yes	Yes
	9250 *	Yes	Yes	Yes
<i>ATI Mobility Radeon™ HD</i>	5870	No	Yes	Yes
	5850	No	Yes	Yes
	5830	No	Yes	Yes
	5770	No	Yes	Yes
	5750	No	Yes	Yes
	5730	No	Yes	Yes
	5650	No	Yes	Yes
	5470	No	Yes	Yes
	5450	No	Yes	Yes
	5430	No	Yes	Yes
	4870 *	Yes	Yes	Yes
	4860 *	Yes	Yes	Yes
	4850X2 *	Yes	Yes	Yes
	4850 *	Yes	Yes	Yes
	4830 *	Yes	Yes	Yes
	4670 *	No	Yes	Yes
	4650 *	No	Yes	Yes
4500 Series *	No	Yes	Yes	
4300 Series *	No	Yes	Yes	
<i>ATI Mobility FirePro™</i>	M7820	Yes	Yes	Yes
	M7740 *	No	Yes	Yes
	M5800	Yes	Yes	Yes
<i>ATI Radeon™ Embedded</i>	<i>E4690 Discrete GPU *</i>	No	Yes	Yes

We are constantly qualifying additional devices. For an up-to-date list of supported devices, please visit <http://developer.amd.com/stream>.

Contact

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

For Stream Computing:
URL: www.amd.com/stream
Questions: streamcomputing@amd.com
Developing: developer.amd.com/support
Forum: www.amd.com/streamdevforum



The contents of this document are provided in connection with Advanced Micro Devices, Inc. ("AMD") products. AMD makes no representations or warranties with respect to the accuracy or completeness of the contents of this publication and reserves the right to make changes to specifications and product descriptions at any time without notice. The information contained herein may be of a preliminary or advance nature and is subject to change without notice. No license, whether express, implied, arising by estoppel or otherwise, to any intellectual property rights is granted by this publication. Except as set forth in AMD's Standard Terms and Conditions of Sale, AMD assumes no liability whatsoever, and disclaims any express or implied warranty, relating to its products including, but not limited to, the implied warranty of merchantability, fitness for a particular purpose, or infringement of any intellectual property right.

AMD's products are not designed, intended, authorized or warranted for use as components in systems intended for surgical implant into the body, or in other applications intended to support or sustain life, or in any other application in which the failure of AMD's product could create a situation where personal injury, death, or severe property or environmental damage may occur. AMD reserves the right to discontinue or make changes to its products at any time without notice.

Copyright and Trademarks

© 2011 Advanced Micro Devices, Inc. All rights reserved. AMD, the AMD Arrow logo, ATI, the ATI logo, Radeon, FireStream, and combinations thereof are trademarks of Advanced Micro Devices, Inc. Other names are for informational purposes only and may be trademarks of their respective owners.