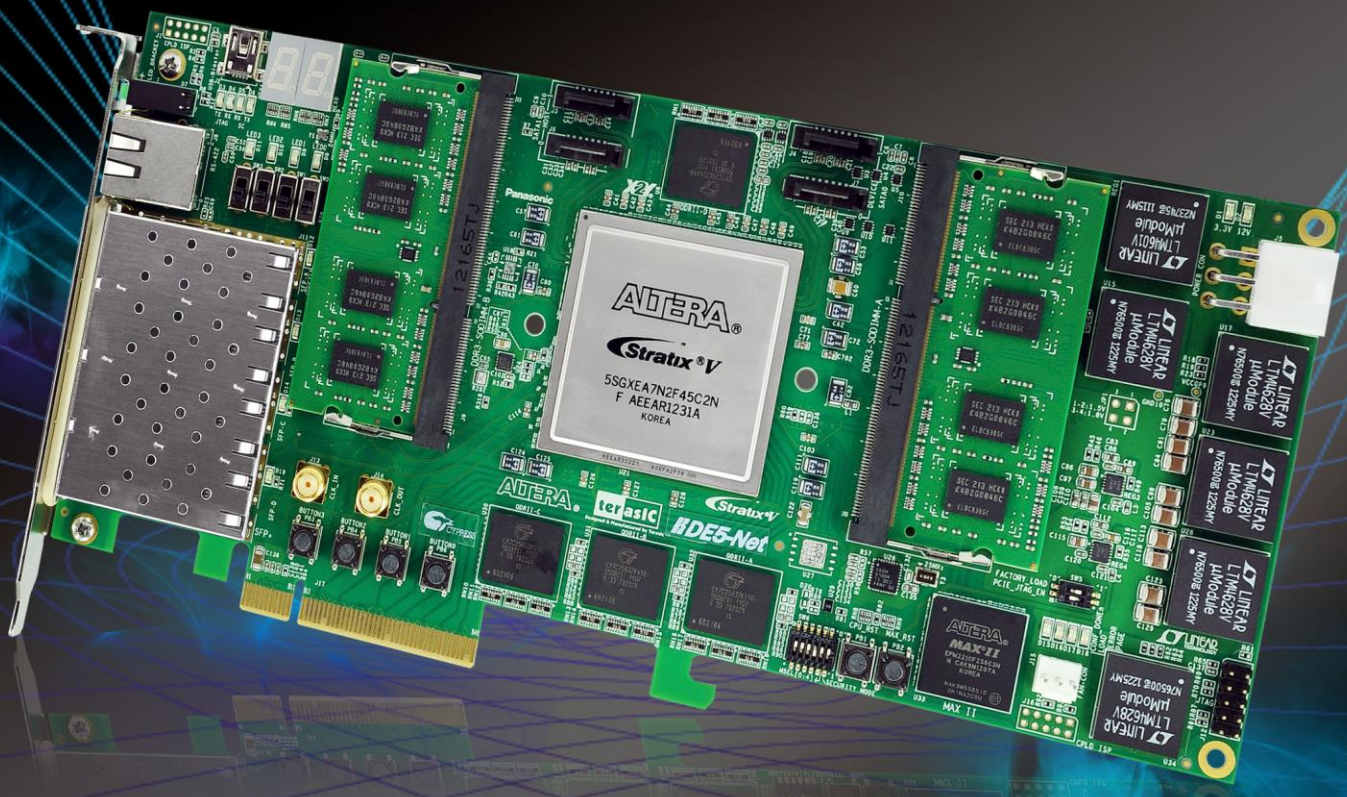
DE5-Net

OpenCL

OpenCL



Stratix V | Flash | PCI EXPRESS | DDR3 | QDRII+ | SFP+ | SERIAL ATA

terasIC www.terasic.com

ALTERA

CONTENTS

Chapter 1

DE5-Net OpenCL

DE5-NET, an unparalleled and powerful platform for high-speed computation, is now officially also an Altera certified board for Altera's Preferred Board Partner Program for OpenCL. It supports both 64-bit Windows and Linux. This document will introduce you how to setup OpenCL development environment for DE5-NET board, and how to compile and execute the example projects for DE5-Net. Note that OpenCL coding instruction is not in the scope of this document, but the user can refer to Intel® FPGA SDK for OpenCL™ Standard Edition Programming Guide for more details.

https://www.altera.com/en_US/pdfs/literature/hb/opencl-sdk/ug-aoclstd-programming-guide.pdf

1.1 System Requirement

The following items are required to set up OpenCL for DE5-NET board:

- Terasic DE5-NET Board with two 2GB DDR3-SODIMM installed
- A Host PC with
  - USB Host Port
  - One PCI Express x8/x16 slot with 12V power pin
  - 32GB memory is recommended, 24GB is minimal
  - 2x3 pin 12V Power for DE5-Net(optional)
- An USB Cable(type A to mini-B)
- 64-bit Windows7 or Windows10 or Linux Installed
- Intel Quartus Prime Standard Edition 18.0.0.614 Installed, licensed is required
- Intel FPGA SDK for OpenCL Standard Edition 18.0.0.614 Installed, license is not required
- DE5-Net OpenCL BSP Installed

- Visual Studio 2012 C/C++ installed for Windows7 or Windows10
- GNU development tools for Linux

Note, Intel FPGA SDK for OpenCL only supports 64-bit OS and x86 architecture.

1.2 OpenCL Architecture

An OpenCL project is composed of both OpenCL Kernel and Host Program as shown in **Figure 1-1**. OpenCL kernel is compiled with Intel FPGA OpenCL compiler provided by the Intel FPGA OpenCL SDK. The Host Program is compile by Visual Studio C/C++ in Windows or GCC on Linux.

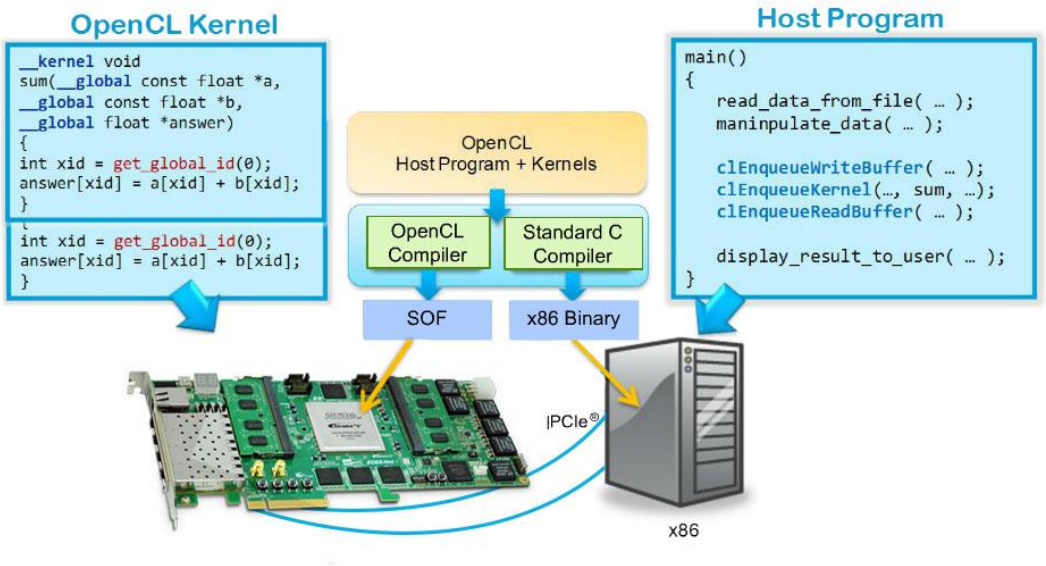


Figure 1-1 Intel FPGA OpenCL Architecture

Chapter 2

OpenCL for Windows

This chapter describes how to set up DE5-NET OpenCL development environment on 64-bit Windows, and how to compile and test the OpenCL examples for DE5-Net. For more details about Intel® FPGA SDK for OpenCL™ Standard Edition Getting Started Guide, please refer to:

https://www.altera.com/en_US/pdfs/literature/hb/opencl-sdk/ug-aoclstd-getting-started.pdf

2.1 Software Installation

This section describes where to get the required software for OpenCL.

■ Intel Quartus Prime and Intel FPGA SDK for OpenCL

Intel Quartus Prime Standard Edition 18.0.0.614 and Intel FPGA SDK for OpenCL 18.0.0.614 can be download from the web site:

http://fpgasoftware.intel.com/opencl/18.0/?edition=standard

For Intel Quartus Prime installation, please make sure that the Stratix V device is included.

■ Visual Studio 2012

If developers don't have Visual studio C/C++ 2012, they can use the trial version of Visual Studio 2012 Express. The software can be downloaded from the web site:

http://www.microsoft.com/en-us/download/details.aspx?id=34673

■ DE5-NET OpenCL BSP (Board Support Package)

After Intel Quartus Prime and OpenCL SDK are installed, please create a "terasic" folder under the folder "C:\intelFPGA\18.0\hld\board" as shown in **Figure 2-1,** where assumed Quartus II is installed on the folder "C:\intelFPGA\18.0\hld\board".

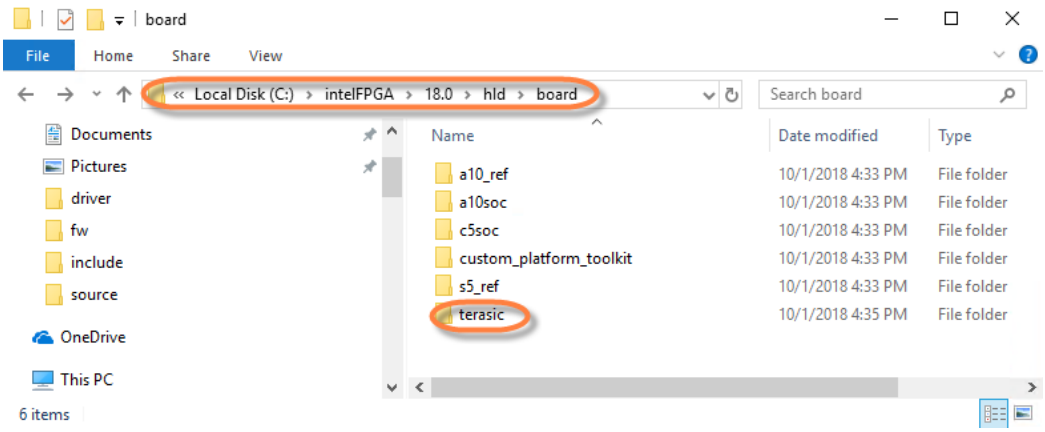


Figure 2-1 Create "Terasic" Folder under OpenCL SDK installed folder

Then, download the DE5NET_openCL_BSP.zip DE5-NET BSP from the web:

http://cd-de5-net.terasic.com

Decompress DE5NET_OpenCL_BSP_18.0.zip and copy content into the "terasic" folder, as shown in **Figure 2-2**.

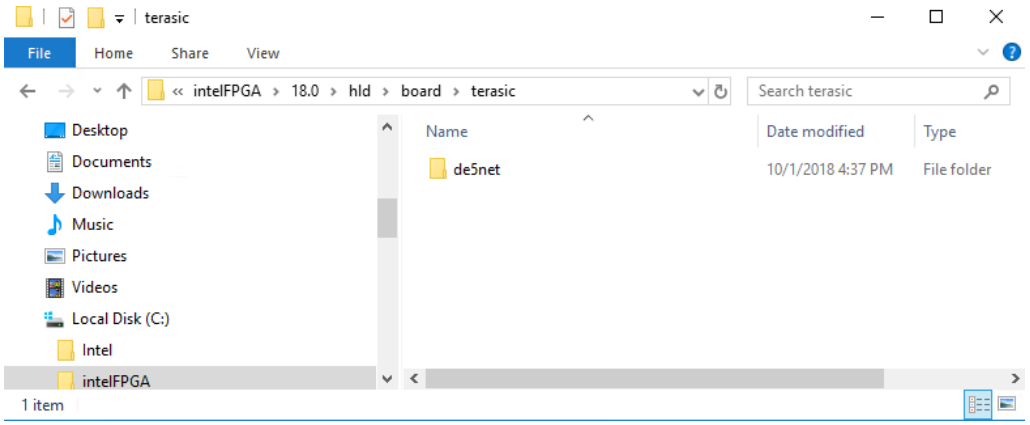


Figure 2-2 DE5-Net OpenCL BSP Content

2.2 Configure

For Intel FPGA SDK for OpenCL to be able to find the kit location of DE5-NET correctly, developers need to create an environment variable **AOCL_BOARD_PACKAGE_ROOT**, and set its value as:

"%INTELFPGAOCLSDKROOT%\board\terasic\de5net"

Also, append "%AOCL_BOARD_PACKAGE_ROOT%\windows64\bin" into the **PATH** environment variable so OpenCL SDK can find the binary file provided by DE5-NET BSP.

Here are the procedures to create the required **AOCL_BOARD_PACKAGE_ROOT** environment variable on Windows 10:

1. Open the Start Menu and open the **Windows System**, then right click **This PC** and open **more**. Select **Properties**.
2. Select **Advanced system settings.**
3. In the **Advanced** tab, select **Environment Variables.**
4. Select **New**.
5. In the popup dialog edit **New User Variable,** type the name in the **Variable name** edit box and type the value in the **Variable name** edit box.
   First edit the environment variable name **ALTERAOCLSDKROOT** to **INTELFPGAOCLSDKROOT** shown in **Figure 2-3**
   Then create environment variable **AOCL_BOARD_PACKAGE_ROOT**, and set its value as: "“%INTELFPGAOCLSDKROOT%\board\terasic\de5net" as show in **Figure 2-4**.
6. Before you get started to execute "aocl install", you will need to disable Digital signature and will then be able to install.
7. In Command Prompt window, type "aocl install" to install PCI Express driver. Note that users need to have administrator privileges to install the driver.

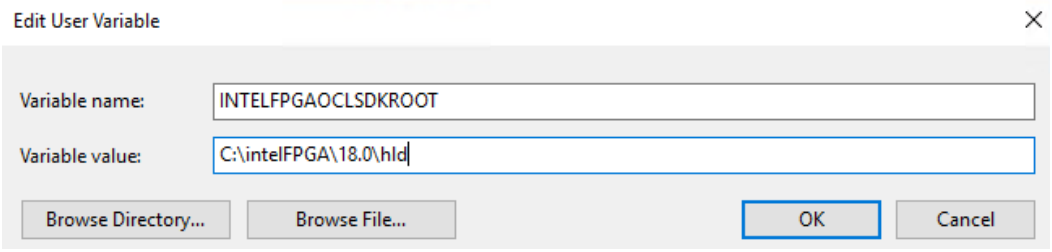


Figure 2-3 Edit INTELFPGAOCLSDKROOT Environment Variable

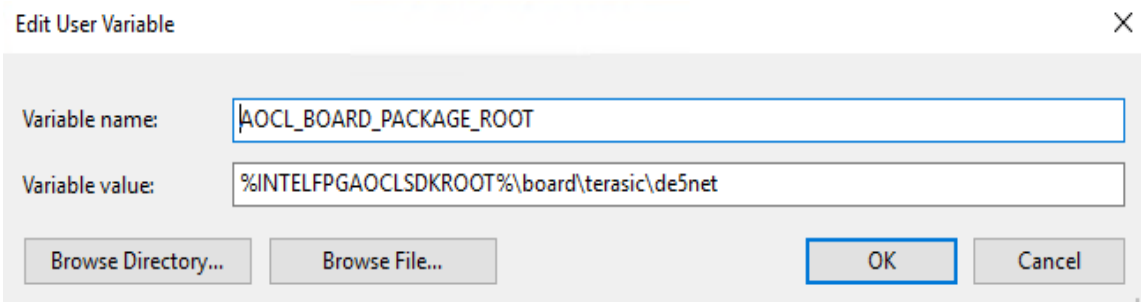


Figure 2-4 Setup AOCL_BOARD_PACKAGE_ROOT Environment Variable

Here are the procedures to add "%AOCL_BOARD_PACKAGE_ROOT%\windows64\bin" into the **PATH** environment variable on Windows 10:

1. Open the Start Menu and open the **Windows System**, then right click **This PC** and open **more**. Select **Properties**.
2. Select **Advanced system settings.**
3. In the **Advanced** tab, select **Environment Variables.**
4. In the Environment Variables window (as shown below), highlight the **Path** variable in the User Variable section and click the **Edit** button as shown in **Figure 2-5**.
5. In the Edit dialog, Append "%AOCL_BOARD_PACKAGE_ROOT%\windows64\bin", "%INTELFPGAOCLSDKROOT%\host\windows64\bin"and"%INTELFPGAOCLSDKROOT%\bin" into the Variable value edit box. Note, each different directory should be separated with a semicolon as shown in **Figure 2-6**.

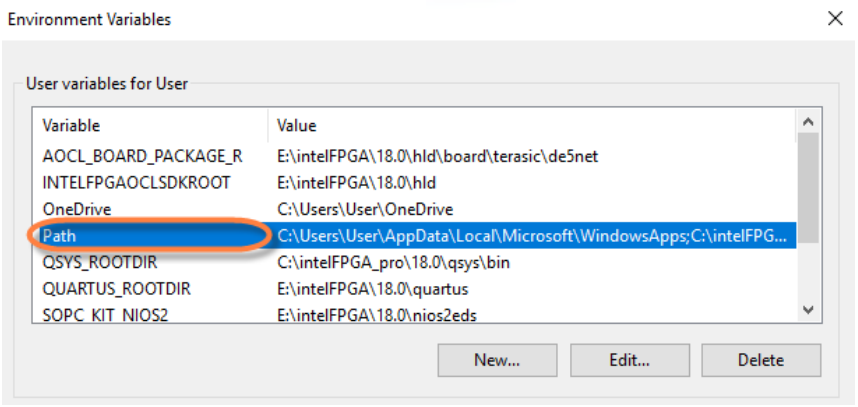


Figure 2-5 Select "Path" and click "Edit" bottom

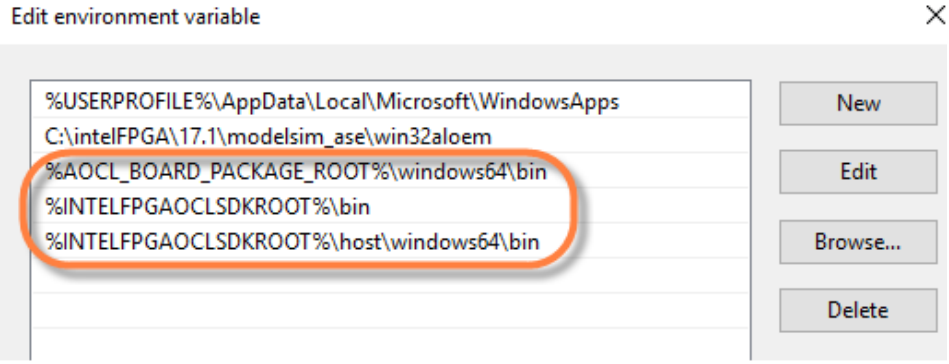


Figure 2-6 Append "%AOCL_BOARD_PACKAGE_ROOT%\windows64\bin"

2.3 Board Setup

Before testing OpenCL on DE5-NET, please following below procedures to set up DE5-NET board on your PC as shown in **Figure 2-7**.

1. Make sure your PC is powered off.
2. Insert DE5-NET board into PCI Express x8 or x16 slot.
3. Connect PC's 12V PCI Express 6-pin power source to the DE5-NET
4. Connect PC's USB port to DE5-NET mini USB port using an USB cable.

Note, the usb cable can be removed later if OpenCL code had been programming to the startup configuration flash of DE5-NET by "aocl flash" command.

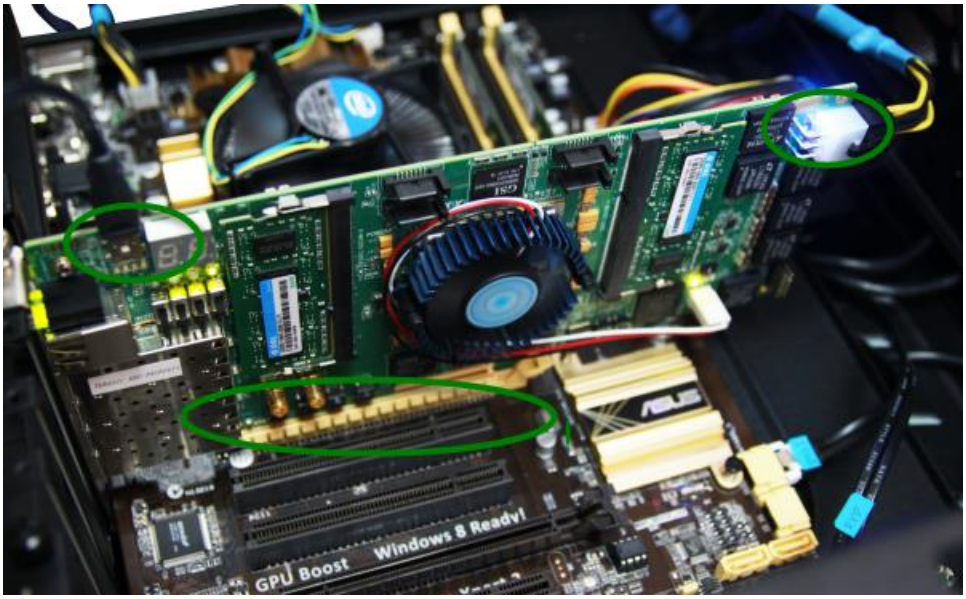


Figure 2-7 Setup DE5-NET board on PC

2.4 OpenCL Environment Verify and Flash CvP

This section will show how to make sure the OpenCL environment is setup correctly. Firstly, please open **Command Prompt** windows by click Windows **Start** button, clicking **All Programs**, clicking **Accessories**, and then click **Command Prompt**.

■ Target Board

In Command Prompt window, type "aoc -list-boards" command, and make sure "de5net_a7" is listed in **Board list** as shown in **Figure 2-8**.

Figure 2-8 'de5net_a7' is listed in Board list

■ Test "aocl flash" Command

In Command Prompt window, type "cd C:\intelFPGA\18.0\hld\board\terasic\de5net\tests\hello_world\bin" to go to hello world OpenCL project folder, then type "aocl flash acl0 hello_world.aocx" to write **hello_world.aocx** OpenCL image onto the startup configuration flash of DE5-NET. Before flash programming, the programmer will ask users which startup configuration image area will be used as shown in **Figure 2-9**. This is because DE5-NET provides two startup configuration image areas, called as Factory Image and User Image. We recommend users to key in '1' to select User Image area.

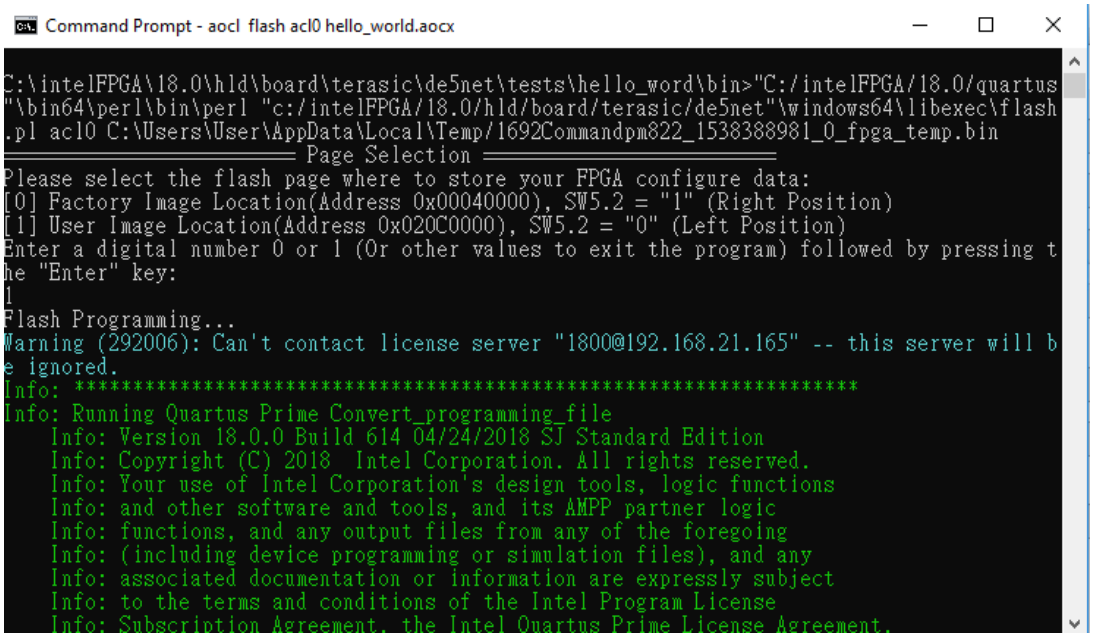


Figure 2-9 Select Flash Page

After users select desired flash area, it will take about 20 minutes for flash programming. **Figure 2-10** is the screen shot when flash programming is done successfully.

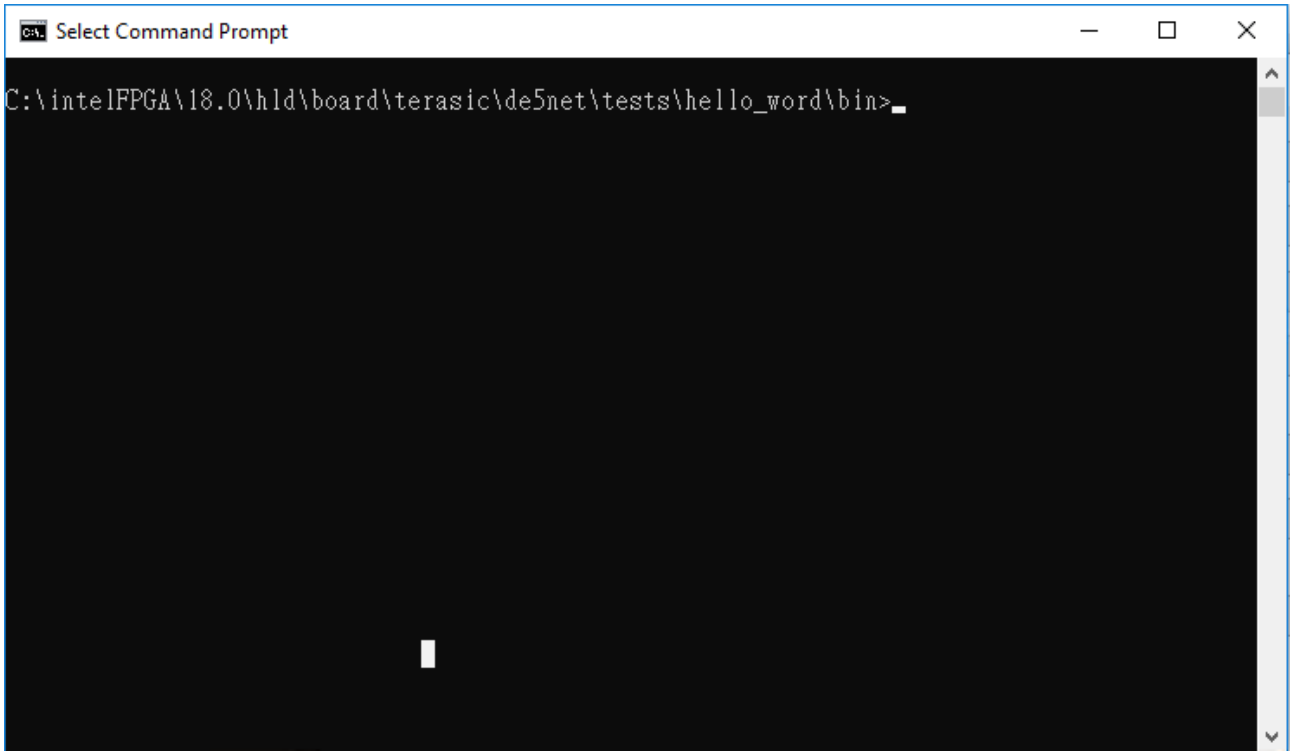


Figure 2-10 "aocl flash acl0 hello_world.aocx" successfully

To make sure a correct image is used when FPGA boots up, please make sure the dip switch SW5.2 on DE5-NET is changed to the correct location. If a User Image area is selected, the dip switch SW5.2 on the DE5-NET should be moved to **left** position as shown in **Figure 2-11.**

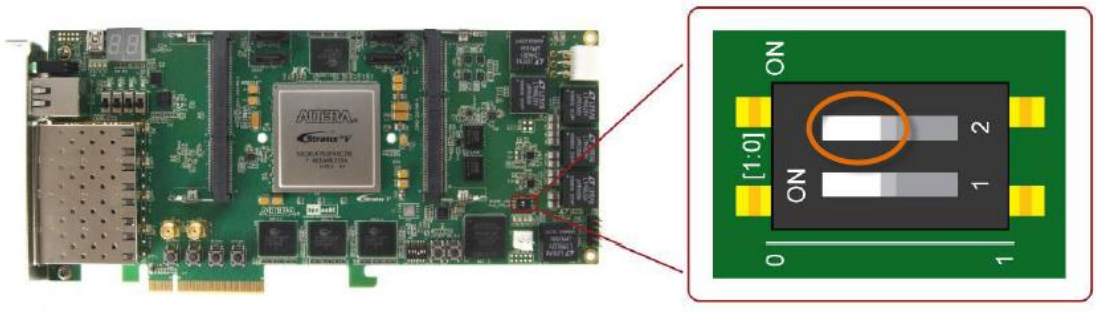


Figure 2-11 Set SW5.2 to Left Position (User Image Page)

After flash programming is done successfully and SW5.2 is set to correct position, developers can power off PC and turn it back on and check whether the **hello_world** OpenCL image, which is CvP enabled, configures the FPGA successfully. In Command Prompt window, type "cd C:\intelFPGA\18.0\hld\board\terasic\de5net\tests\hello_world\bin" to go to **hello_world\bin** project folder, then type "aocl program acl0 hello_world.aocx" to configure the FPGA with **hello_world.aocx** OpenCL image. If the programming message displays "Program succeed" as

shown in Figure 2-12, it means the **hello_world** OpenCL image is programmed into the flash correctly and CvP works well.

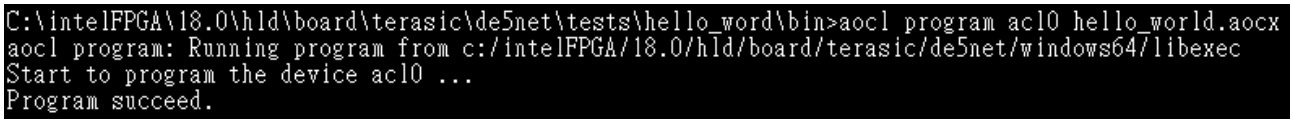


Figure 2-12 "aocl program acl0 hello_world.aocx" use CvP

2.5 Compile and Test OpenCL Project

This section will show how to compile and test OpenCL kernel and OpenCL Host Program for the vector_add project. Developers can use the same procedures to compile and test other OpenCL examples for DE5-NET.

■ Compile OpenCL Kernel

The utility **aoc** (Intel(R) FPGA SDK for OpenCL(TM) Kernel Compiler) is used to compile OpenCL kernel. In Command Prompt window, type "cd C:\intelFPGA\18.0\hld\board\terasic\de5net\tests\vector_add" to go to **vector_add** project folder, then type "aoc device/vector_add.cl -o bin/vector_add.aocx" to compile the OpenCL kernel. It will take about one hour for compiling. When the compilation process is finished, OpenCL image file bin\vector_add.aocx is generated. Figure 2-13 is the screenshot when OpenCL kernel is compiled successfully. For required parameters to compile vectorAdd.cl, please refer to the README.html that is in the same folder as the vectorAdd.cl. For detailed usage of **aoc**, please refer to the **Intel® FPGA SDK for OpenCL™ Standard Edition Programming Guide**:

https://www.altera.com/en_US/pdfs/literature/hb/opencl-sdk/ug-aoclstd-programming-guide.pdf

Figure 2-13 "aoc vector_add.cl" OpenCL kernel compile successfully

■ Compile Host Program

Visual Studio C/C++ 2012 is used to compile the Host Program. Launch Visual Studio, and select menu item "FILE→Open Project…". In the Open Project dialog, go to the folder "C:\intelFPGA\18.0\hld\board\terasic\tests\vector_add", and select "vector_add.sln" as shown Figure 2-14.

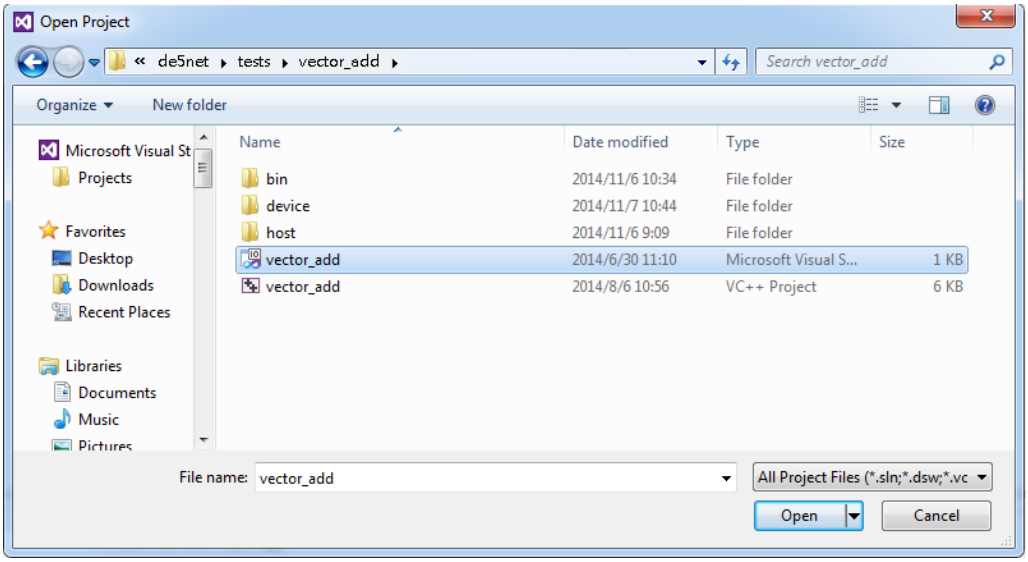


Figure 2-14 Open vector_add.sln Host Program

After vector_add Host Program project is opened successfully, in Visual Studio IDE select menu item "BUILD→Build Solution" to build host program. When build is successfully, you will see successful message as show in **Figure 2-15**. The execute file is generate in:

"C:\intelFPGA\18.0\hld\board\terasic\tests\vector_add\bin\host.exe"

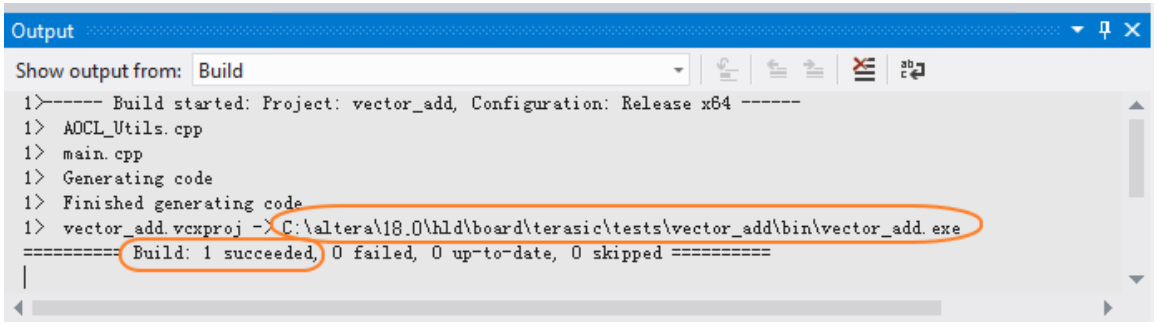


Figure 2-15 Message for vector_add Host Program build successfully

■ Test vector_add project

Firstly, use the compiled OpenCL image file vector_add.aocx to configure the FPGA. In Command Prompt window, type "cd C:\intelFPGA\18.0\hld\board\terasic\de5net\tests\vector_add\bin" to go to **vector_add\bin** project folder, then type "aocl program acl0 vector_add.aocx" to configure FPGA with the OpenCL Image vector_add.aocx. If configuration is successfully, you will see the successful message as shown in **Figure 2-16.**

Figure 2-16 "aocl program acl0 vector_add.aocx" configured successfully

In Command Prompt window, execute "host.exe". **Figure 2-17** is the screen shot when the test is successful.

Figure 2-17 "vector_add" test successfully

Chapter 3

OpenCL for Linux

This chapter describe how to setup DE5-NET OpenCL development environment on 64-bit Linux (CentOS 7.X are recommended), and how to compile and test the OpenCL examples for DE5-Net. For more details about Intel FPGA OpenCL, please refer to Intel® FPGA SDK for OpenCL™ Standard Edition Getting Started Guide:

https://www.altera.com/en_US/pdfs/literature/hb/opencl-sdk/ug-aoclstd-getting-started.pdf

3.1 Software Installation

This section describes how to download and install the required software for OpenCL.

■ Intel Quartus Prime and Intel FPGA SDK for OpenCL

Intel Quartus Prime Standard Edition 18.0.0.614 and Intel FPGA SDK for OpenCL 18.0.0.614 can be download from the web site:

http://fpgasoftware.intel.com/opencl/18.0/?edition=standard

For Intel Quartus Prime installation, please make sure that the Stratix V device is included.

Open the link and select the Linux SDK as Figure 3-1 shows.

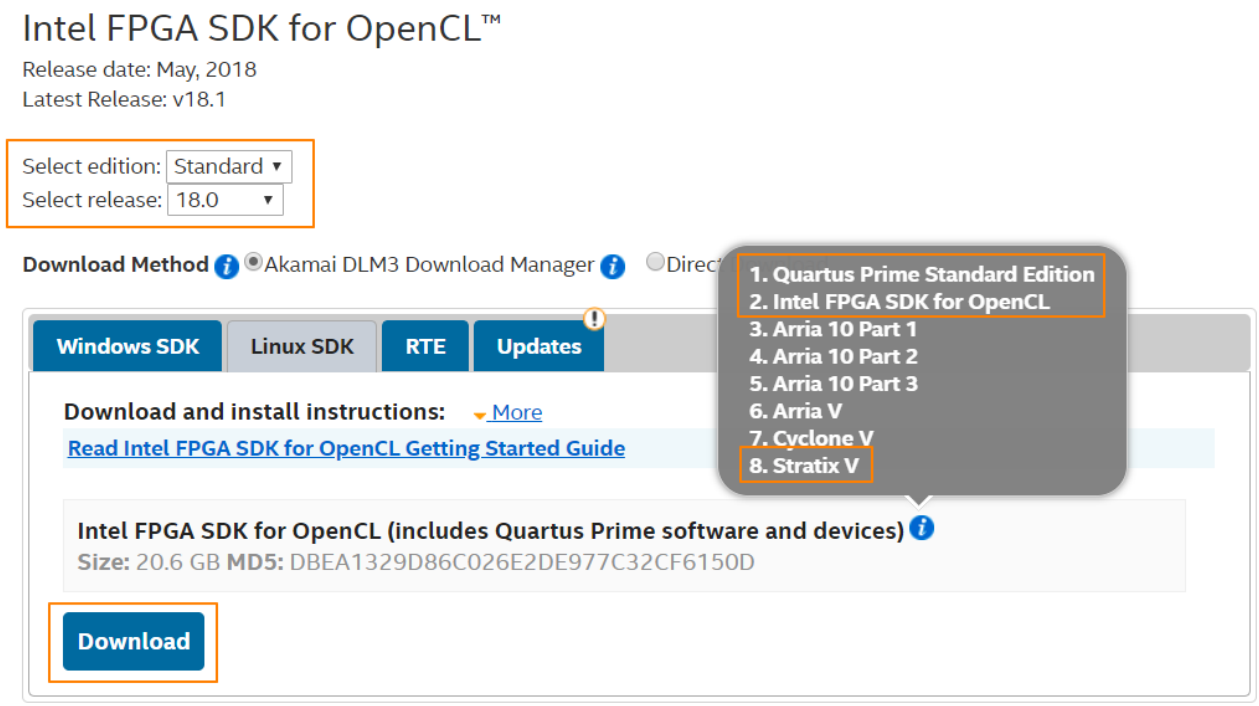


Figure 3-1 Linux SDK table

Intel Quartus Prime software uses the built-in USB-Blaster II drivers on Linux to access USB-Blaster II download cable on DE5-Net. But after installed the Intel Quartus Prime software with built-in drivers, User need to change the port permission for USB-Blaster II via issuing

"gedit /etc/udev/rules.d/51-usbblaster.rules"

to create and add the following lines to the **/etc/udev/rules.d/51-usbblaster.rules** file.

```
# USB-Blaster
ENV{ID_BUS}=="usb" ENV{ID_VENDOR_ID}=="09fb", ENV{ID_MODEL_ID}=="6001", MODE="0666"
ENV{ID_BUS}=="usb" ENV{ID_VENDOR_ID}=="09fb", ENV{ID_MODEL_ID}=="6002", MODE="0666"
ENV{ID_BUS}=="usb" ENV{ID_VENDOR_ID}=="09fb", ENV{ID_MODEL_ID}=="6003", MODE="0666"

# USB-Blaster II
ENV{ID_BUS}=="usb" ENV{ID_VENDOR_ID}=="09fb", ENV{ID_MODEL_ID}=="6010", MODE="0666"
ENV{ID_BUS}=="usb" ENV{ID_VENDOR_ID}=="09fb", ENV{ID_MODEL_ID}=="6810", MODE="0666"
```

Note: You must have system administration (root) privileges to configure the USB-Blaster

download cable drivers.

■ GNU development tools

GNU development tools such as **gcc** (include **g++**) and **make** are required to build the driver and application under Linux. User can issue 'yum install gcc gcc-c++ compat-gcc-48-c++ make' command to download and install them and their dependencies via internet.

Note: To install the SDK on Linux, you must install it in a directory that you own (that is, a directory that is not a system directory). You must also have sudo or root privileges.

■ DE5-NET OpenCL BSP (Board Support Package)

After Intel Quartus Prime and OpenCL SDK are installed, please create a "terasic" folder under the folder "/root/intelFPGA/18.0/hld/board", where assumed Intel Quartus Prime is installed on the folder "/root/intelFPGA/18.0".

Then, download the DE5NET_OpenCL_BSP_18.0.tar.gz DE5-NET Linux BSP from the web:

http://cd-de5-net.terasic.com

Decompress DE5NET_OpenCL_BSP_18.0.tar.gz and copy content into the "**terasic**" folder, as shown in **Figure 3-2**.

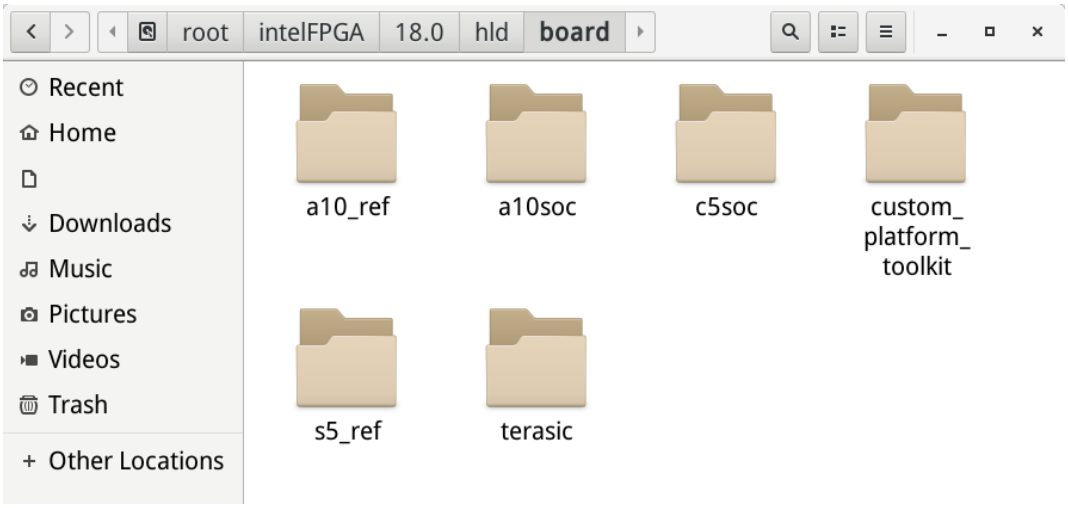


Figure 3-2 Copy DE5-Net OpenCL BSP to Terasic Folder

3.2 Configure

If you install the Intel FPGA development tools software and OpenCL SDK on a system that does not contain any .cshrc or Bash Resource file (.bashrc) in your directory, you must set the INTELFPGAOCLSDKROOT and PATH environment variables manually. And for Intel FPGA SDK for OpenCL able to find the kit location of DE5-NET correctly, the developers need to create an environment variable for the DE5-NET board **AOCL_BOARD_PACKAGE_ROOT**, and set its value as:

"$INTELFPGAOCLSDKROOT\board\terasic\de5net"

Alternatively, you can edit the "/etc/profile" **profile** file, and append the environment variables to it. To do this type "*gedit /etc/profile" command on Linux Terminal* to open the **profile** file by the **gedit** editor tool, and append the following setting to the **profile** file. Then, save the file and type "*source /etc/profile"* command in Linux Terminal to make the settings make effect.

```
export QUARTUS_ROOTDIR=/root/intelFPGA/18.0/quartus
export INTELFPGAOCLSDKROOT=/root/intelFPGA/18.0/hld
export AOCL_BOARD_PACKAGE_ROOT="$INTELFPGAOCLSDKROOT/board/terasic/de5net"
export PATH="$PATH:$QUARTUS_ROOTDIR/bin:$INTELFPGAOCLSDKROOT/bin"
export LD_LIBRARY_PATH="$AOCL_BOARD_PACKAGE_ROOT/linux64/lib:"\
"$INTELFPGAOCLSDKROOT/host/linux64/lib:$AOCL_BOARD_PACKAGE_ROOT/tests/extlibs/lib"
export LM_LICENSE_FILE=/root/intelFPGA/18.0/license.dat
```

3.3 Board Setup

Before testing OpenCL on DE5-NET, please following the below procedure to setup DE5-NET board on your PC as shown in **Figure 3-3**.

1. Make sure your PC is power off.
2. Insert DE5-NET board into PCI Express x8 or x16 slot.
3. Connect PC's 12V PCI Express 6-pin power to the DE5-NET source (if there's not, ignore this step)

4. Connector PC's USB port to DE5-NET mini USB port using an USB cable.

Note, the usb cable can be removed later if any one of OpenCL code had been programming to the startup configuration flash of DE5-NET by "aocl flash" command.

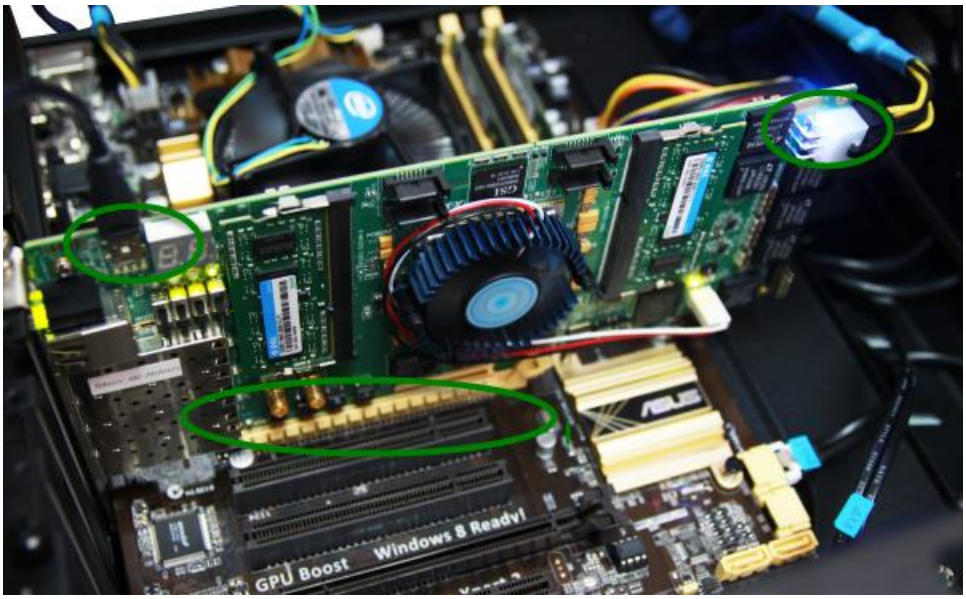


Figure 3-3 Setup DE5-NET board on PC

3.4 OpenCL Environment Verify and Flash CvP

This section will show how to make sure the OpenCL environment is setup correctly.

Firstly, please open the Linux system **terminal** window by right click the Mouse on system desktop, then clicking on **Open Terminal**.

■ Target Board

In the Linux terminal, type "*aoc -list-boards"* command, and make sure "de5net_a7" is listed in **Board list** as shown in **Figure 3-4**.

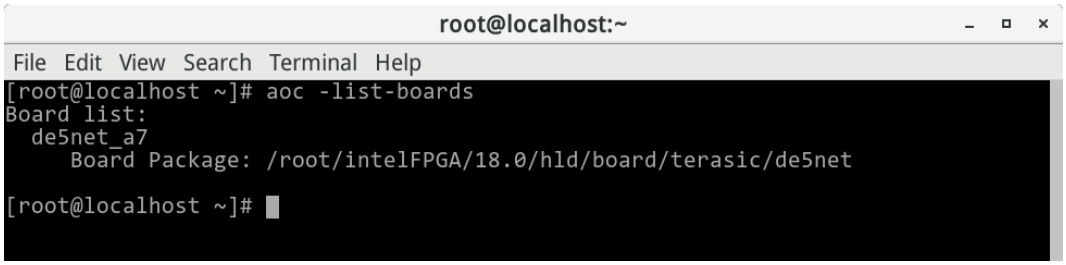


Figure 3-4 'de5net_a7' is listed in Board list

Type "*yum install kernel-devel"* to install the kernel related development package matched the current kernel as shown in **Figure 3-5**.

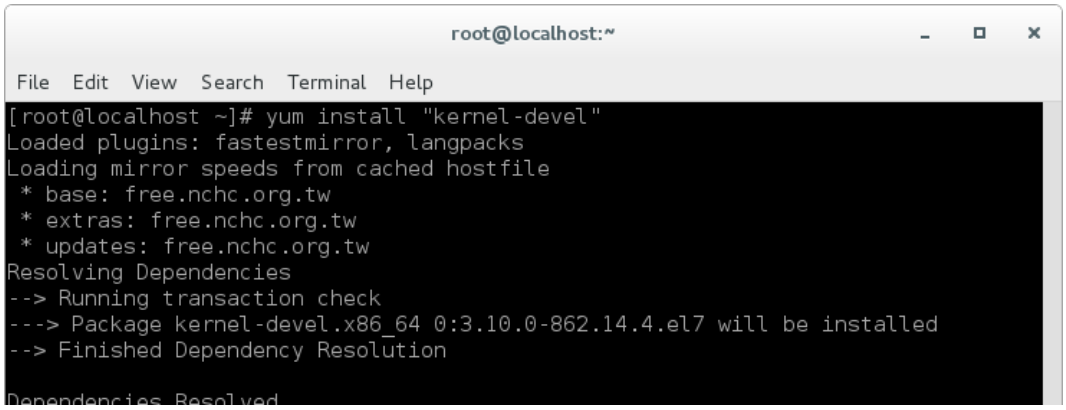


Figure 3-5 development package installation

Then type "*aocl install"* to install the PCIe driver for the DE5-NET as shown in **Figure 3-6**.

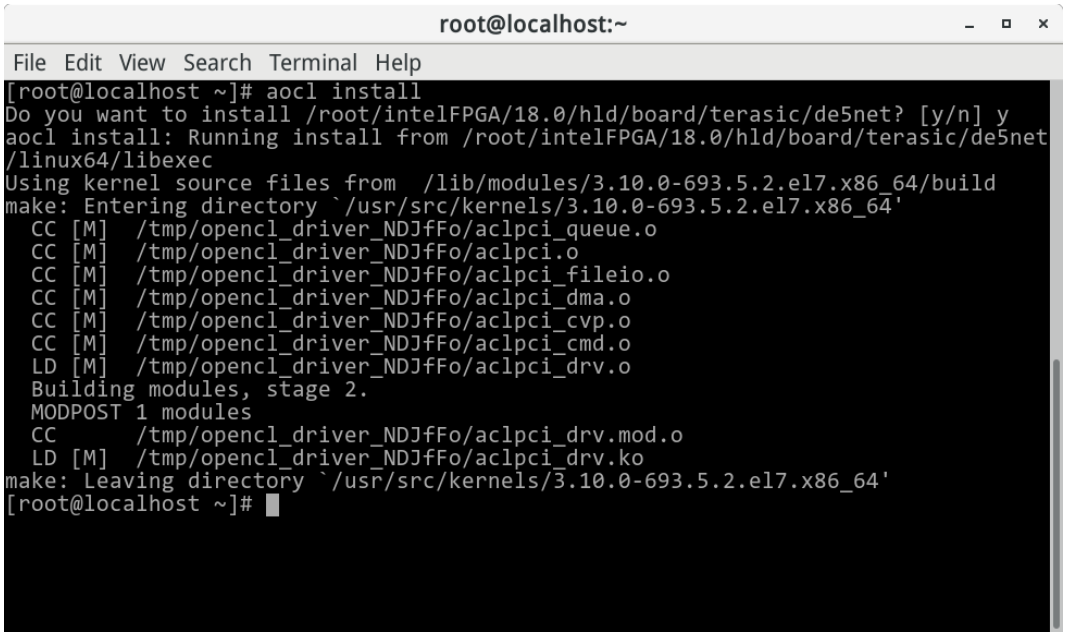


Figure 3-6 driver installation

■ Test "aocl flash" Command

In the terminal, type "*cd /root/intelFPGA/18.0/hld/board/terasic/de5net/tests/hello_world/bin*" to go to hello world OpenCL project folder, then type "*aocl flash acl0 hello_world.aocx*" to program **hello_world.aocx** OpenCL image onto the startup configuration flash of DE5-NET.

Before flash programming, the programmer will ask users which startup configuration image area will be used as shown in **Figure 3-7**. This is because DE5-NET provides two startup configuration image areas, called as Factory Image and User Image. Typing '1' to select User Image area is recommended.

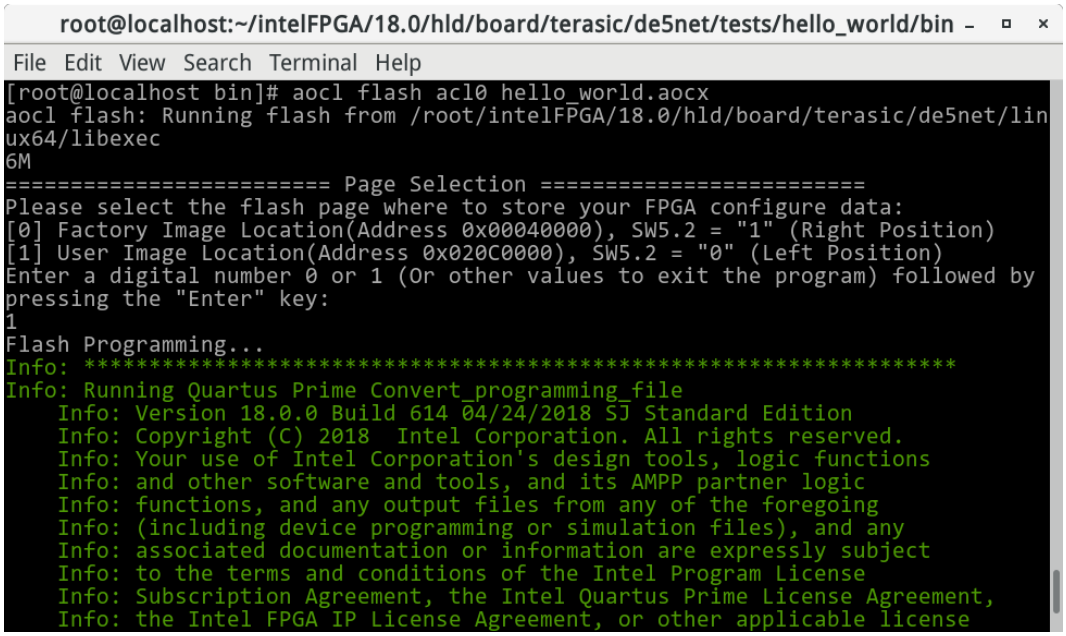


Figure 3-7 Select Flash Page

After selecting the desired flash area, it will take about 20 minutes for flash programming. **Figure 3-8** is the screen shot when flash programming is done successfully.

```
root@localhost:~/intelFPGA/18.0/hld/board/terasic/de5net/tests/hello_world/bin -
File  Edit  View  Search  Terminal  Help
Info (209005): Programming status: programming flash memory at byte address 0x04
090000
Info (209005): Programming status: programming flash memory at byte address 0x04
0A0000
Info (209005): Programming status: programming flash memory at byte address 0x04
0B0000
Info (209005): Programming status: programming flash memory at byte address 0x04
0C0000
Info (209005): Programming status: programming flash memory at byte address 0x04
0D0000
Info (209005): Programming status: programming flash memory at byte address 0x04
0E0000
Info (209005): Programming status: programming flash memory at byte address 0x04
0F0000
Info (209005): Programming status: programming flash memory at byte address 0x04
100000
Info (209011): Successfully performed operation(s)
Info (209061): Ended Programmer operation at Fri Sep 28 14:54:02 2018
Info: Quartus Prime Programmer was successful. 0 errors, 0 warnings
    Info: Peak virtual memory: 2686 megabytes
    Info: Processing ended: Fri Sep 28 14:54:02 2018
    Info: Elapsed time: 00:05:19
    Info: Total CPU time (on all processors): 00:00:25
[root@localhost bin]#
```

Figure 3-8 "aocl flash acl0 hello_world.aocx" successfully

To make sure correct image is used when FPGA boot, please make sure the dip switch SW5.2 on DE5-NTE is located at correct location. If User Image area is selected, the dip switch SW5.2 on the DE5-NET should be move to **left** position as shown in **Figure 3-9**.

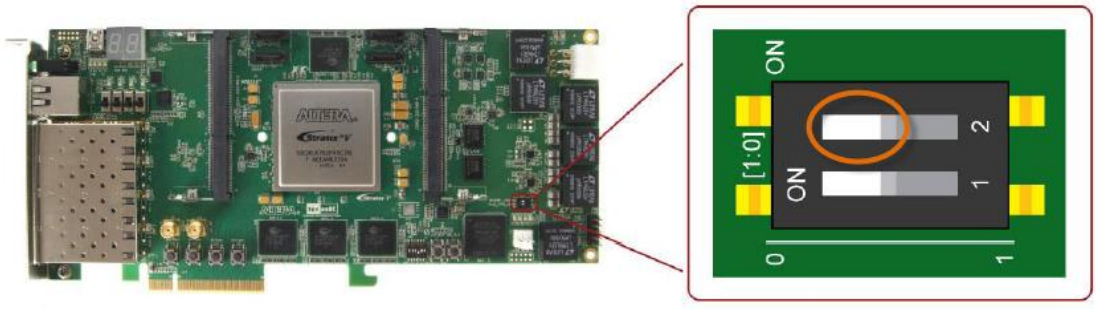


Figure 3-9 Set SW5.2 to Left Position (User Image Page)

After flash programming is done successfully and SW5.2 is set to correct position, developers can reboot the PC and check whether the **hello_world** OpenCL image, which is CvP enabled, configures the FPGA successfully when DE5-NET is power on. In the Linux terminal, type "*cd /root/intelFPGA/18.0/hld/board/terasic/de5net/tests/hello_world/bin*" to go to hello_world project folder, then type "*aocl program acl0 hello_world.aocx*" to configure the FPGA with **hello_world.aocx** OpenCL image. If the programming message displays "Program succeed" as shown in **Figure 3-10**, it means the **hello_world** OpenCL image is programmed into the flash correctly and CvP works well.

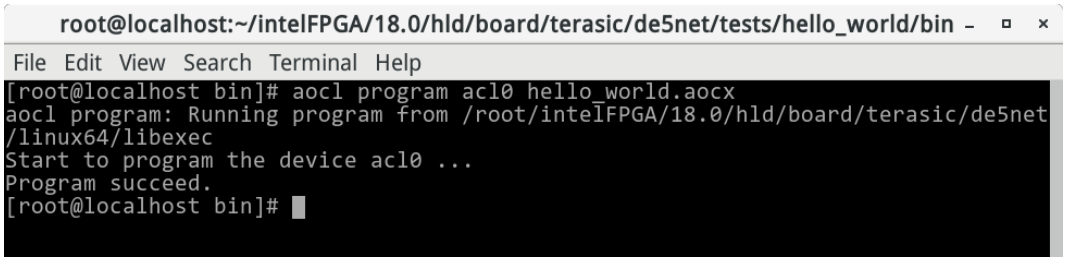


Figure 3-10 "aocl program acl0 hello_world.aocx" use CvP

3.5 Compile and Test OpenCL Project

This section will show how to compile and run the OpenCL kernel and OpenCL Host Program for the vector_add example project. Developers can use the same procedures to compile and test other OpenCL examples (included in the kit) for DE5-NET.

■ Compile OpenCL Kernel

In the terminal, type "*cd /root/intelFPGA/18.0/hld/board/terasic/de5net/tests/vector_add*" to go to **vector_add** project folder, then type "*aoc device/vector_add.cl -o bin/vector_add.aocx -report*" to compile the OpenCL kernel. It will takes about one hour for compiling. After that, the OpenCL image file *bin/vector_add.aocx* is generated. **Figure 3-11** is the screen shot when OpenCL kernel is compiled successfully. For required parameters to compile vector_add.cl, please refer to the README.html that is in the same directory.

The utility **aoc** (Intel(R) FPGA SDK for OpenCL(TM) Kernel Compiler) is used to compile OpenCL kernel. For detailed usage of **aoc**, please refer to the **Intel® FPGA SDK for OpenCL™ Standard Edition Programming Guide**:

https://www.altera.com/en_US/pdfs/literature/hb/opencl-sdk/ug-aoclstd-programming-guide.pdf

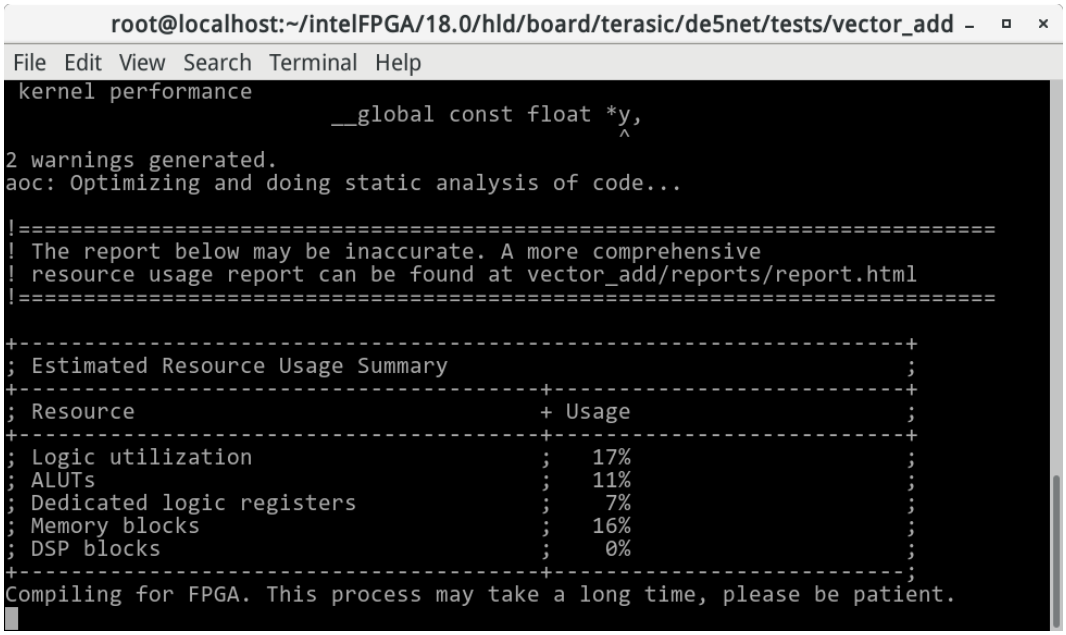


Figure 3-11 "aoc vector_add.cl" OpenCL kernel compile successfully

■ Compile Host Program

In the terminal, type "*cd /root/intelFPGA/18.0/hld/board/terasic/de5net/tests/vector_add*" and then type
"make" to compile the host program.

When build is successfully, you will see successful message as show in **Figure 3-12**. The execute file is generate in the same directory which named bin.

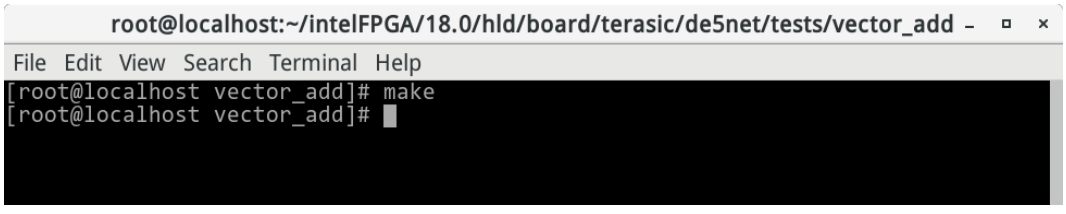


Figure 3-12 successful Message for vector_add Host Program build

■ Test vector_add project

Firstly, in the terminal, type "*cd /root/intelFPGA/18.0/hld/board/terasic/de5net/tests/vector_add/bin"* to go to the **vector_add** project folder, then type "*aocl program acl0 vector_add.aocx"* to configure FPGA with the OpenCL Image *vector_add.aocx*.

Then, launch the compiled Host Program to start vector_add execute file for test. In the terminal type "./host". **Figure 3-13** shows the execution is successful.

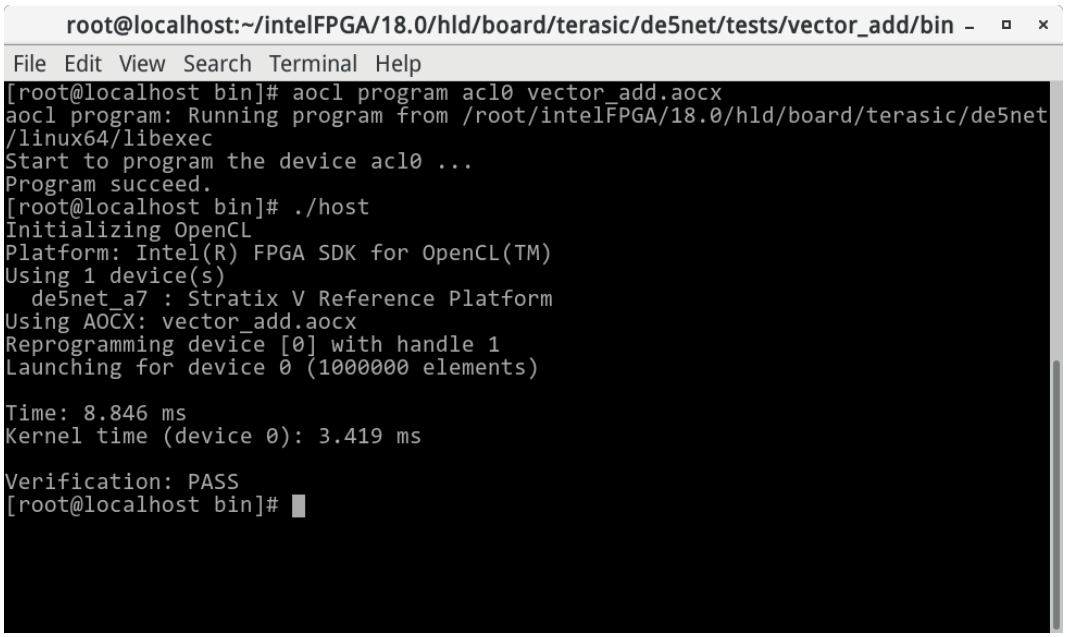


Figure 3-13 successful Message for "vector_add" test