

SDAccel Development Environment

Release Notes

UG1202 (v2015.4) February 16, 2016



Revision History

The following table shows the revision history for this document.

Date	Version	Changes
2/16/2016	2015.4	Xilinx OpenCL runtime supports IBM Power8 Eclipse IDE enhanced with examples. Coding templates and performance optimization guide. New DSA v2.1 for Alpha Data ADM-PCIE-7V3 and ADM-PCIE-KU3 board. New DSA v2.1 for Alpha Data ADM-PCIE-7V3 with 1-DDR New DSA v2.1 for ADM-PCIE-KU3 board with 1-DDR and 2-DDR.
11/9/2015	2015.3	New DSA v2.0 for Alpha Data ADM-PCIE-7V3 Virtex-7 690T board and Alpha Data ADM-PCIE-KU3 Kintex® UltraScale® KU60 board. Beta release of Eclipse IDE with Debug, Profile and Application Trace.
6/1/2015	2015.1	Initial Xilinx 2015.1 release.

What's New

The SDAccel® Development Environment 2015.4 release introduces OpenCL runtime support for IBM Power8 servers for executing OpenCL, C, and C++ applications on FPGA device boards compiled with SDAccel. The SDAccel 2015.4 release doubles kernel to global memory bandwidth by enabling two DDR3 with DSA v2.1 for Alpha Data ADM-PCIE-KU3 board. This release provides ease-of-use enhancements with examples in the Eclipse-based SDAccel IDE and coding templates. In addition, we are releasing *SDAccel Development Environment Methodology Guide: Performance Optimization*, (UG1207) to accelerate the optimization and deployment of OpenCL™, C, and C++ kernels.

• Performance

- Support for Alpha Data ADM-PCIE-KU3 with 2 DDR3 doubles kernel to global memory bandwidth.
- Examples showing how to improve application memory throughput from host to global and kernel to global memory.
- Coding templates and examples with supported attributes and techniques for improving kernel performance.
- *SDAccel Development Environment Methodology Guide: Performance Optimization*, (UG1207) with tips and tricks for application acceleration.
- Timing check severity has been increased from Warning to Error. Existing designs will need to be updated to meet timing.

• Usability

- Streamlined Xilinx Board Installation with new command line utility: `xbinst`.
- `xocc`: Command line compiler similar to `gcc` for the creation of FPGA programming binaries.
 - Allows parallel compilation of kernels using all cores on the developer workstation.
 - Allows parallel compilation of kernels using `lsf` cluster job dispatch.
 - Link stage allows you to customize a mix of kernels in FPGA programming binary.
 - Link stage enables the insertion of RTL based kernels in a makefile based environment.
 - Xilinx command line compiler (`xocc`) also has a simple mode where compile and link stages are combined into a single command sequence as is the case of `gcc`.

- Debug support
 - Integrated debug flow using GDB for host code.
 - Ability to set breakpoints in OpenCL kernel code in CPU emulation flow.
 - printf support in all development flows: CPU emulation, hardware emulation and while executing kernel in hardware.
- Host application profiling in all development flows. Profiling report includes API calls, kernel execution, data transfer, top ten kernel execution, top ten buffer writes, and top ten buffer reads.
- CTRL-C support to terminate applications running on the Alpha Data card

• **Language support**

- OpenCL Installable Client Driver (ICD)
- OpenCL 2.0 pipes support for passing data between kernels.
- Xilinx OpenCL pipes extension supports blocking read and write for passing data between pipes.
- OpenCL 2.0 on-chip global memory for passing data between kernels
- RTL kernel packaging into xo kernel containers using Vivado®

• **Device Support Archive (DSA)**

- New DSA v2.1 for Alpha Data ADM-PCIE-7V3 Virtex-7 690T board with Xilinx DMA
 - PCIe Gen3x8
 - Kernel clock frequency: 200 MHz
- New 1DDR DSA v2.1 for Alpha Data ADM-PCIE-KU3 Kintex® UltraScale® KU60 board with Xilinx DMA
 - PCIe Gen3x8, 1 DDR
 - Kernel clock frequency: 200 MHz
- New 2DDR DSA v2.1 for Alpha Data ADM-PCIE-KU3 Kintex UltraScale KU60 board with Xilinx DMA
 - PCIe Gen3x8, 2 DDR
 - Kernel clock frequency: 200 MHz

Beta Feature

- OpenCL runtime supports Power8 architecture for executing OpenCL, C, and C++ applications on FPGA boards compiled with SDAccel.
- OpenCL runtime has been enhanced to support multiple devices with single host applications.

- IDE with debug, profiling and application trace view.
 - Eclipse based IDE with support for integrated GDB for debug.
 - Host code debug in all three flows (CPU emulation, Hardware emulation and in hardware)
 - Kernel debug in CPU emulation flow
 - Profiling reports for application optimization in all three flows with increased information and accuracy from CPU emulation to execution on hardware.
 - Application timeline trace provides a holistic view of memory transfers between the host and the device as well as between kernel compute units and device global memory.
 - Enables you to quickly pinpoint data transfer bottlenecks and discover inefficient memory access patterns.
 - Enables you to analyze the impact of concurrent operation of multiple compute units on system performance.
- Half precision floating-point data type support.
- Device Support Archive (DSA) creation through Vivado® IP Integrator.
- Kernels defined from RTL sources in the SDAccel script based mode. This capability is not available in the GUI based flow.

Known Issues

• SDAccel Installation Known Issues

- Setup script does not add all High Level Synthesis (HLS) simulation libraries.
 - Solution: Manually add missing libraries.

```
<SDACCEL_INSTALL_DIR>/Vivado_HLS/HEAD/lnx64/tools/opencv
<SDACCEL_INSTALL_DIR>/Vivado_HLS/HEAD/lnx64/tools/fpo_v6_1
<SDACCEL_INSTALL_DIR>/Vivado_HLS/HEAD/lnx64/tools/fpo_v7_0
<SDACCEL_INSTALL_DIR>/Vivado_HLS/HEAD/lnx64/tools/fft_v9_0
<SDACCEL_INSTALL_DIR>/Vivado_HLS/HEAD/lnx64/tools/fir_v7_0
<SDACCEL_INSTALL_DIR>/Vivado_HLS/HEAD/lnx64/tools/dds_v6_0
```
- Setup script does not add path to g++.
 - Solution: Manually add g++ (<SDACCEL_INSTALL_DIR>/lnx64/tools/gcc/bin/g++) in path before executing SDAccel.
- xocc makefile based flow requires additional setup:
 - `setenv XILINX_OPENCL $SDACCEL_OPENCL`
 - `setenv LD_LIBRARY_PATH ${XILINX_OPENCL}/runtime/lib/x86_64:${XILINX_OPENCL}/lib/lnx64.o:${LD_LIBRARY_PATH}`

- **Device Support Archive (DSA) Known Issues**

- Maximum number of compute units is limited to 10.
- In certain cases, the Alpha Data ADM-PCIE-7V3 and ADM-PCIE-KU3 FPGA card may not link up in the PCIe Gen3x8 configuration. This is due to a known errata regarding Avago Technologies ExpressLane™ PEX 8747 (rev CA) PLX technology Gen 3 PCIe switch. The errata of PEX 8747 (rev CA) links up with Xilinx PCIe endpoint as Gen1 x8 instead of Gen3 x8. An eeprom upgrade is required for the PLX switch which customers should be able to obtain directly from Avago Technologies. A Confidential Disclosure Agreement (CDA) may need to be signed for obtaining the patch.

- **OpenCL Compiler Known Issues**

- A `struct` type argument of kernel function cannot contain vector or array type member field.

- **Emulation Flow Known Issues**

- Emulation fails with symbol lookup error:
`./shared0: undefined symbol: _Z13native_divideff for CPP applications`
Solution: Wrap the entire top function with `extern "C" {...}`

- **printf Known Issues**

- `printf` function calls within a kernel have a limit of 254 items which can be recorded per work item. Exceeding this capacity will overflow the buffer used for `printf` data and result in undefined system behavior. Vectors in `printf` statements are treated as one item per dimension.
- `printf` function calls within a kernel only provides values for the X dimension of the workgroup. `printf` calls for work items in the Y and Z dimensions do not output anything.

- **Debugger Known Issues**

- `printf` did not execute in case of segmentation fault.
Solution: Use GDB debugging to step through code and see values.
- `print i` in for loop causes no symbol "i" in current context.
Solution: `printf` support is not available for multiple kernels. To debug applications using `printf`, use single kernels.

Appendix A: Multiple Devices Support

SDAccel runtime supports applications targeting multiple FPGA boards in CPU emulation, hardware emulation and running on the actual hardware. The host system must meet the requirements below to be able to run applications on multiple boards:

- All FPGA boards must be the same type
- All FPGA boards must be programmed from the same device support archive (DSA)

Configuration File

A configuration file describing the board configuration on the host system is required to run CPU and hardware emulation targeting multiple boards. You need to manually create the configuration file that matches the actual hardware configuration on the host system. An example configuration file for two Alpha Data ADM-PCIE-7V3 cards is shown below. More examples can be found in the SDAccel installation directory, `<SDACCEL_INSTALL_DIR>/data/emulation`.

```
{
  "MajorVersion": "1",
  "MinorVersion": "0",
  "Platform":
  {
    "Boards": [
      {
        "Devices" : [
          {
            "Name": "xilinx:adm-pcie-7v3:1ddr:2.1",
            "DdrBanks": [{"Size": "8G"}],
            "NumDevices": "2"
          }
        ]
      }
    ]
  }
}
```

Details of attributes in the configuration file are described in the table below. Note that the configuration file is case sensitive.

Attributes	Descriptions
MajorVersion	Major version for the configuration file format. Set to 1 for SDAccel 2015.4.
MinorVersion	Minor version for the configuration file format. Set to 0 for SDAccel 2015.4.
Platform	Denote section of platform description
Boards	Denote section of boards description
Devices	Denote section of devices description
Name	Device name. The following devices are included in SDAccel 2015.4 installation. xilinx:adm-pcie-7v3:1ddr:2.1 xilinx:adm-pcie-ku3:1ddr:2.1 xilinx:adm-pcie-ku3:2ddr:2.1
DdrBanks	DDR bank description. Each DDR bank is listed as Size:value pair. Below is an example of two DDR banks with 8GBytes each: "DdrBanks" : [{"Size":"8G"}, {"Size":"8G"}],
NumDevices	Number of the same devices described in the current "devices" section.

SDAccel runtime searches for configuration files in the order specified below and uses the first one it finds:

1. The `config.json` file from the directory where the host executable is executed.
2. The configuration file specified by the `XCL_EMULATION_CONFIGFILE` environment variable.
3. The `config.json` file in `$HOME/.Xilinx/sdaccel/`
4. If the runtime does not find any configuration file in the paths above, it uses a default `config.json` file in `<SDACCEL_INSTALL_DIR>/data/emulation.`

Host Application Targeting Multiple Devices

Applications targeting multiple devices must first find all devices in the current JSON configuration or installed on the host system by calling `clGetDeviceIDs`:

```
err = clGetDeviceIDs(platform_id, CL_DEVICE_TYPE_ACCELERATOR, 16, devices,
&num_devices);
```

Only a single context needs to be created for all devices returned by `clGetDeviceIDs`:

```
context = clCreateContext(0, num_devices, devices, NULL, NULL, &err);
```

To enqueue kernels on all devices, separate command queue needs to be created for each device.

```
commands[0] = clCreateCommandQueue(context, devices[0], 0, &err);
commands[1] = clCreateCommandQueue(context, devices[1], 0, &err);
```

The host code will also need to create a separate program object for each device:

```
program[0] = clCreateProgramWithBinary(context, 1, &target_devices[0], &n[0], const
unsigned char **) &kernelbinary[0], &status, &err);
program[1] = clCreateProgramWithBinary(context, 1, &target_devices[1], &n[1], (const
unsigned char **) &kernelbinary[1], &status, &err);
```


After context, command queue, and program are set up in the host code, you can call other APIs targeting different devices based on different program and command queue.

Availability

To learn more about the SDAccel development environment, visit www.xilinx.com/sdaccel where you will find [QuickTake video tutorials](#), documentation and links to the SDAccel Development Environment-qualified Alliance members. To access the capabilities of the SDAccel Development Environment, please contact your [local sales representative](#).

Please Read: Important Legal Notices

The information disclosed to you hereunder (the "Materials") is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available "AS IS" and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of Xilinx's limited warranty, please refer to Xilinx's Terms of Sale which can be viewed at <http://www.xilinx.com/legal.htm#tos>; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in such critical applications, please refer to Xilinx's Terms of Sale which can be viewed at <http://www.xilinx.com/legal.htm#tos>. © Copyright 2012 - 2016 Xilinx, Inc. Xilinx, the Xilinx logo, Artix, ISE, Kintex, Spartan, Virtex, Vivado, Zynq, and other designated brands included herein are trademarks of Xilinx in the United States and other countries. All other trademarks are the property of their respective owners.