SDAccel Development Environment

Release Notes, Installat on, and Licensing Guide

UG1238 (v2018.2) July 10, 2018

ATTENTION! SDAccel Development Environment 2018.2 XDF users: Click here to view the 2018.2.xdf version of this guide.

Revision History

The following table shows the revision history for this document.

Section	Revision Summary
07/10/2018 Version 2018.2	
Chapter 1: Release Notes	Added sections for Known Issues and Release Notes from Previous Versions.
07/02/2018 Version 2018.2	
Document restructure	This document has been made specific to SDAccel™. For the SDSoC™ release information, refer to *SDSoC Environments Release Notes, Installation, and Licensing Guide* (UG1294).
06/06/2018 Version 2018.2	
Chapter 1: Release Notes	Added information about updates to SDx™ for 2018.2.
04/04/2018 Version 2018.1	
SDSoC - SDAccel Common Updates for 2018.1	Added information about updates to SDx™ for 2018.1.
SDSoC Development Changes for 2018.1	Added details about feature changes for 2018.1.
SDAccel Development Environment Changes for 2018.1	Added details about feature changes for 2018.1.

Table of Contents

Revision History 2

Chapter 1: Release Notes 4
- SDAccel Development Environment What's New for 2018.2 4
- Known Issues 9
- Release Notes from Previous Versions 9

Chapter 2: Introduction to the SDAccel Development Environment 10
- About the SDx Environments 10
- Hardware Requirements 10
- Software Requirements 11
- About the SDAccel Installation 11

Chapter 3: Obtaining a License on the Xilinx Licensing Site 15

Chapter 4: Installing the SDAccel Environment 17
- Preparing to Install the Tools 17
- Installing the SDAccel Environment 17

Appendix A: Additional Resources and Legal Notices 26
- Xilinx Resources 26
- Documentation Navigator and Design Hubs 26
- References 27
- Please Read: Important Legal Notices 27

XILINX

Chapter 1

Release Notes

SDAccel Development Environment What's New for 2018.2

The following SDAccel™ Development Environment updates are included in the 2018.2 release.

- Ease-of-Use enhancements to simplify **Project Management**.
- A one-stop **SDx™ IDE** Assistant from which to drive all flows and actions.
- Actionable Guidance with improved **Reporting** and visualization.
- Introduction of transaction-level hardware **Debug**.
- **runtime** enhancements that allow applications to execute faster and with less memory.
- Additional options for **Platform and RTL Kernels** are provided.

***Note*:** New updates to the **HLS Compiler** for 2018.2 are also provided below.

Supported Platforms

For a list of all supported platforms and devices, see the SDAccel Product Page.

Project Management

- Seamless and direct access to the HLS compiler is now provided from within the SDx GUI.
- The Vivado® HLS Compiler may be launched directly on any kernel, allowing detailed analysis and optimization to be performed within the HLS compiler environment. All changes are automatically saved into the SDx project when exiting the HLS Compiler.
- Improvements to the SDx project directory structure allows for easier access to report and log files.
- Ability to export and import a project to move between workspaces. This includes custom build/debug configurations
- Prior releases had one project for Vivado synthesis and a second for Vivado implementation. Now there is a single project with both synthesis and implementation for easier debug flow.

SDx IDE

- New SDx Assistant view to manage SDx software projects. The view is used to manage hardware functions, build projects, and view reports. Common actions are in a pop-up menu for ease of use.
- New tabs in Debug Perspective to display contents of OpenCL™ Command Queues, Memory Buffers, Monitors and Protocol Checkers.

Reporting

- The new Guidance feature provides actionable feedback, enabling users to more easily optimize for performance.
- Online Guidance is augmented with new documentation guides, including a Programmers Guide, Profiling and Optimization Guide and a Verification and Debug Guide.
- Profile Summary enhanced to report compute unit stalls and execution monitoring.
    - Compiler and Linker Options enhanced to control granularity of profile counter and trace data collection, monitor and report compute unit stalls, and execution of compute units
- Application Timeline View enhanced to provide more visibility into kernel operations, enabling faster identification and optimization of performance bottlenecks.
    - Reporting of compute unit port level transfers, NDRange Start/Stop Time, and event dependencies.
    - Display of function level activity within a compute unit.
- New reporting features allow dynamically trading off SDAccel execution-time versus detailed reporting.
    - Detailed, but time consuming, hardware level reports are no longer generated by default but are available through the GUI and xocc command options.

Debug

- xocc provides a new switch, `--dk chipscope` for users familiar with the FPGA design flow to compile their application and get transaction level visibility when running on hardware.

Runtime

- Applications now execute faster and using less memory with runtime management now implemented in dedicated hardware.

Platform and RTL Kernels

- Optional Reset for RTL Kernels to improve timing closure.
- The Feature ROM is enhanced to include information on the DSA Power Monitoring, Scheduler, and Debugging.

HLS Compiler

- New Schedule Viewer accessible from the Analysis Perspective to graphically display dependencies of operations and control steps.
- Overall faster processing of source code embedded directives (pragmas).
- Redesigned "dataflow" with strict-mode checks to help guide toward optimal solution.
- Performance enhancements with both higher clock rates (4% higher on average) and reduced design latency with 10% fewer clock cycles for design completion.
- Five additional `math.h` optimized functions for fixed-point data types (`pow`, `abs`, `sincos`, `acos`, `asin`).
- XFast xfOpenCV libraries are now fully supported in Vivado HLS.
- Co-simulation waveforms are enhanced to clearly display DATAFLOW transaction views.
- New DRCs added to the Vivado HLS GUI DRC tab to expedite timing closure and strengthen pragma checks.

Changes to Existing Behavior

The following table specifies changes to existing flows and scripts which are required when using the 2018.2 release.

Table 1: **SDAccel Migration Summary**

Area	2017.4 Behavior	Changes for 2018.2
Kernel Names	Kernels are named using long-form names, for example: `--nk Big_long_kernel_type:1:Big_long_kernel_name_1` Kernels with long names may have experienced the error: ERROR: The m_name entry length (79), exceeds the allocated space (64)	Kernels with long names may have experienced the error: ERROR: The m_name entry length (79), exceeds the allocated space (64) Kernels can be named with short-form names, for example: `--nk Big_long_kernel_type:1:short_name` This change will require command options which specify a kernel name or kernel port name (e.g., the `--nk` and `-sp` options) to be updated to use the new short-form name.
Compile	A synthesized DCP may be imported using the `--reuse_synth` option.	The `--reuse_synth` option is no longer supported. A post-implementation DCP may be imported and packaged into an xclbin using the `--reuse_impl` option.
Profile	Profiling data is generated per DDR.	Profiling data is generated on a per port basis.

Table 1: **SDAccel Migration Summary** *(cont'd)*

Area	2017.4 Behavior	Changes for 2018.2
Reporting	The `--report/-r` option may be used to generate reports.	The `--report/-r` option is no longer supported. New option `--report_level <arg>` may be used to generate reports. The `<arg>` value allows reports with varying degrees of detail to be generated: `<arg>`(default is R0): • `R0` : Minimal reports and no intermediate DCP files. • `R1` : Includes R0 reports plus: ◦ Identifies design characteristics to review for each kernel `(report_failfast)`. ◦ Identifies design characteristics to review for full design post-optimization `(report_failfast)`. ◦ Saves the post-optimization DCP. • `R2` : Includes R1 reports plus: ◦ Full Vivado default reporting including a DCP file after each implementation step. ◦ Identifies design characteristics to review for each SLR after placement `(report_failfast)`. • `Estimate`: ◦ Generates an estimate report. This option is useful for software emulation target: the estimate report is not generated by default.
HLS	When pipelining tasks, if no `-II` option is specified, an optimization target of `-II=1` is assumed.	Mode `relax_ii_for_timing` is now the default mode. If an `-II` value is not specified, the `-II` target of 1 will be automatically relaxed to provide improved timing. If an `-II` target is required, it should be explicitly specified, and may result in frequency scaling.
HLS	When specifying a dependence with option `WAR false`: • The dependence is removed in both the `WAR` and `RAW` directions • The dependence is removed for both loop-carry and non-loop-carry cases	When specifying a dependence `WAR false`: • Only the `WAR` direction is removed • Only loop-carry dependencies are removed

Table 1: **SDAccel Migration Summary** *(cont'd)*

Area	2017.4 Behavior	Changes for 2018.2
HLS	When specifying a dependence with option `WAR true`: • The dependence is removed in both the `WAR` and `RAW` directions • A `-distance 0` is assumed if no `-distance` option is specified	When specifying a dependence `WAR false`: • Only the `WAR` direction is removed • If the `-distance` option is not specified, no distance is assumed
HLS	A dependence specified on any loop in a loop nest is applied to all loops in the nest.	A dependence specified on any loop in a loop nest is only applied to that loop.

HLS	Version 4.6 of `gcc` is used and can return both:
	`T& complex<T>.real()`
	`T complex<T>.real()`
	This allowed the use of both coding styles:
	`xn_input[i].real() = constant_input;`
	`xn_input[i].imag(0.0);`
		Version 6.2 of `gcc` is used and can only return
		`T complex<T>.real()`
		This requires use of the following coding style:
		`xn_input[i].real(constant_input);`
		`xn_input[i].imag(0.0);`

HLS	Version 4.6 of `gcc` is used and static const variables may be used: `class A { static const double d = 1.0; }`	Version 6.2 of `gcc` is used and supports only static constexpr. Existing code must be updated to initialization the static const class member out of the class scope: `class A{ static const double d; }; const double A::d = 1.0;`
HLS	Arbitrary precision C types `intN/uintN` may be used with C++.	Arbitrary precision C types are no longer supported for C++. • For C, use arbitrary precision types `intN/uintN` defined in `ap_cint.h` • For C++, use arbitrary precision types `ap_int<N>/ap_uint<N>` defined in `ap_int.h`

Known Issues

Known Issues for the SDx™ Environment are available in AR#71223.

Release Notes from Previous Versions

Release notes for previous versions of the SDAccel™ Environment are available at the following link:

- What's New in SDAccel for 2017.4

XILINX

Chapter 2

Introduction to the SDAccel Development Environment

About the SDx Environments

The 2018.2 SDx™ Environment software release consists of the SDSoC™ Development Environment for Zynq® UltraScale+™ MPSoC and Zynq®-7000 SoC families, and the SDAccel™ Development Environment for Data Center and PCIe®-based accelerator systems. These environments share a common installer, but are licensed individually. All SDx Environments include the Vivado® Design Suite for programming the target devices and for developing custom hardware platforms.

Hardware Requirements

The SDAccel™ environment requires the following hardware:

- Acceleration Card. Use one of the following:
  - Xilinx® Kintex UltraScale FPGA KCU1500 Reconfigurable Acceleration card based on XCKU115-FLVB2104-2-E FPGA.
  - Xilinx Virtex UltraScale+ FPGA VCU1525 Reconfigurable Acceleration card based on XCVU9P-L2FSGD2104E FPGA.
- Host computer: Desktop computer for hosting the acceleration card. The host computer must provide the following.
  - Motherboard with a PCIe® Gen3 X8 slot
  - 64 GB RAM
  - 100GB free disk space
- Programming computer: Laptop or desktop computer running the supplied Vivado® Design Suite 2018.2 for programming the FPGA.

- Xilinx Platform Cable USB 2, part number HW-USB-II-G for connecting the programming computer to the acceleration card. See the *Platform Cable USB II Data Sheet*, (DS593).
- Additional platforms are available from partners. For more information, visit the SDAccel Developer Zone: https://www.xilinx.com/products/design-tools/software-zone/sdaccel.html.

Software Requirements

The SDAccel™ Development Environment runs on Linux operating systems. The supported operating systems are listed below.

- Linux Support
  - Red Hat Enterprise Workstation/Server 7.3-7.4 (64-bit)
  - CentOS 7.2
  - CentOS 7.3-7.4 (64-bit)
  - Ubuntu Linux 16.04.3 LTS (64-bit)
    - Linux kernel 4.4.0 is supported
    - Ubuntu LTS enablement (also called HWE or Hardware Enablement) is *not* supported

About the SDAccel Installation

The SDAccel™ Environment runs on the Linux operating systems only with no support for Windows.

CentOS/RHEL 7.3 and 7.4 Package List

You can install the EPEL repository using the instructions at https://fedoraproject.org/wiki/EPEL. In addition, the following packages should be installed with the `yum` install command.

- ocl-icd
- ocl-icd-devel
- opencl-headers
- kernel-headers-$(uname -r)
- kernel-devel
- gcc-c++

- gcc
- gdb
- libstdc++-static
- make
- opencv
- libjpeg-turbo-devel
- libpng12-devel
- libtiff-devel
- compat-libtiff3
- python
- git
- dmidecode
- pciutils
- strace
- perl
- boost-devel
- boost-filesystem
- gnuplot
- cmake
- lm_sensors
- unzip
- redhat-lsb
- libuuid
- libuuid-devel
- mokutil
- wget
- openssl
- libuuid-devel

Ubuntu 16.04 Package List

The following packages should be installed with `apt-get install` command.

- ocl-icd-libopencl1
- opencl-headers
- ocl-icd-opencl-dev
- linux-headers
- linux-libc-dev
- g++
- gcc
- gdb
- make
- libopencv-core
- opencv
- libjpeg-dev
- libpng-dev
- libtiff5-dev
- python
- git
- dmidecode
- pciutils
- strace
- perl
- libboost-dev
- libboost-filesystem-dev
- gnuplot
- cmake
- lm-sensors
- lsb
- unzip
- linux-headers-$(uname -r)

- python3-sphinx-rtd-theme
- sphinx-common
- python3-sphinx
- libuuid1
- uuid-dev
- mokutil
- wget
- openssl

Recommended Libraries

Xilinx recommends that you install the following libraries on your operating system.

- Independent JPEG Group's JPEG runtime library (version 6.2).

```
sudo apt-get install libjpeg62 libjpeg62-dev
```

Xilinx recommends the following packages should be installed on CentOS 7.x

- PNG reference library.

```
sudo yum install libpng12
```

- The Linux Standards Base (LSB) library. The redhat-lsb package provides utilities needed for LSB Compliant Applications.

```
sudo yum install redhat-lsb
```

- The `libtiff3` package, an older version of `libtiff` library for manipulating TIFF (Tagged Image File Format) image format files.

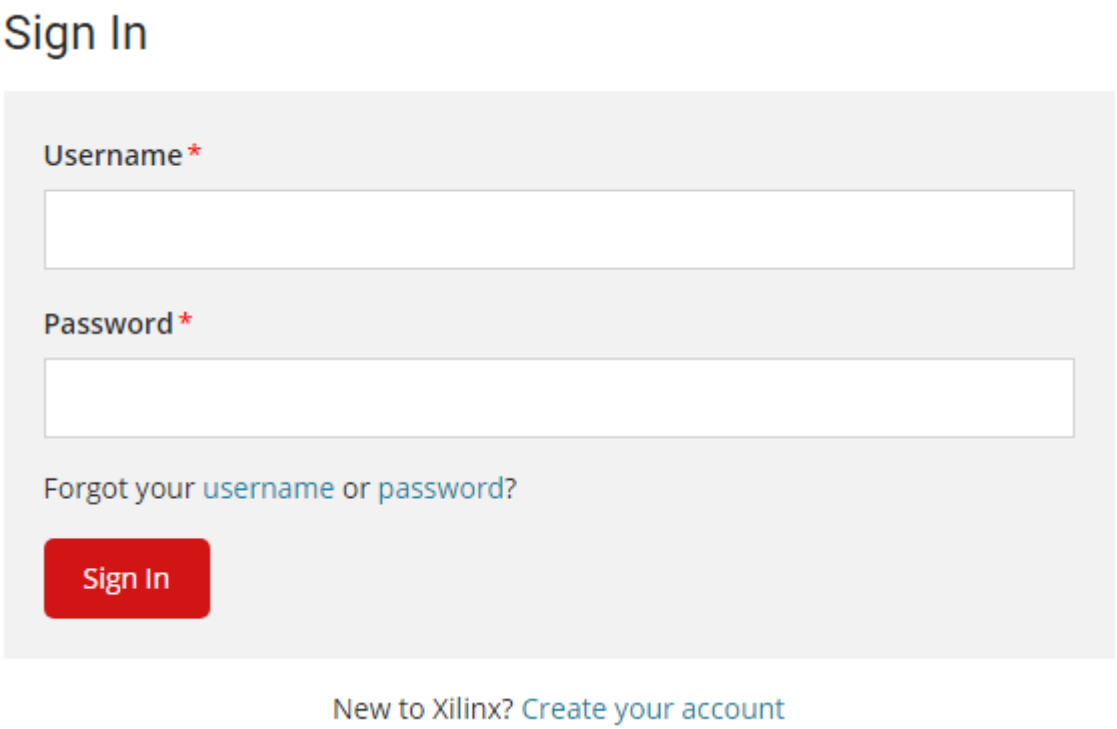
```
sudo yum install redhat-lsb
```

Obtaining a License on the Xilinx Licensing Site

This section describes the steps to obtain a license for the SDx™ development environment.

1. Sign in to the Xilinx® licensing website: https://www.xilinx.com/getproduct. See the following figure.

Figure 1: **Xilinx Licensing Site Sign-in Screen**



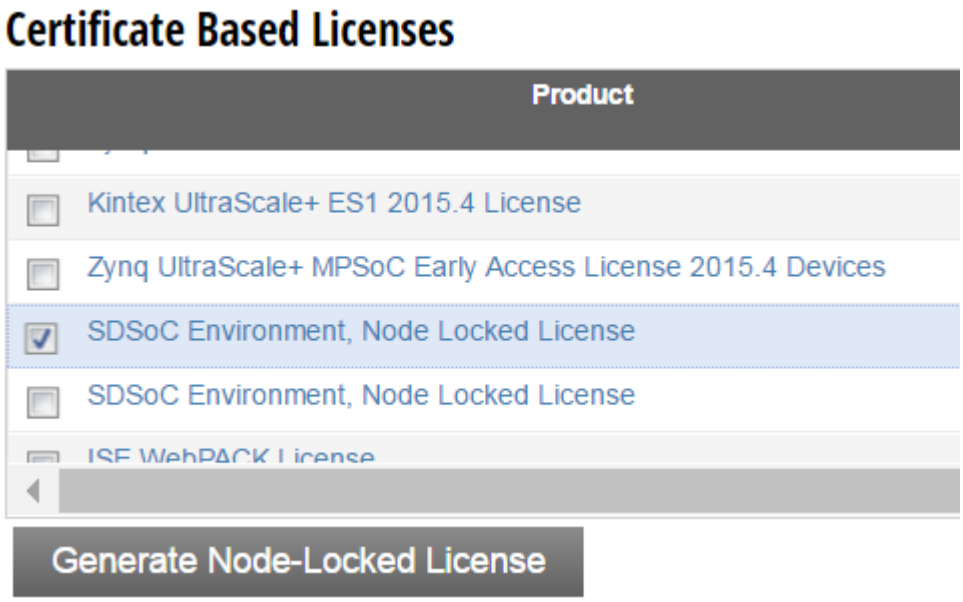
   If this is your first time generating a license for the SDAccel™ - Xilinx OpenCL™ Design Environment, contact your Xilinx representative to enable your access to the SDAccel licensing website.

2. In the account drop-down menu, select **XILINX - SDAccel Environment**.

***Note*:** This only shows up if you have purchased or redeemed an SDAccel license.

3. From the Certificate Based Licenses menu, select **SDAccel Environment, Node-Locked License**.

Figure 2: **Certificate Based Licenses Menu**



4. Click **Generate node-locked license**.
5. Enter a Host ID in the **License Generation** screen and click **Next**.
6. Verify that the Host ID for the license is correct and click **Next**.
7. Accept the licensing agreement by clicking **Accept**.

   You will receive an email from `xilinx.notification@entitlenow.com` with the license file.
8. Set the `XILINXD_LICENSE_FILE` environment variable to point to the location of the license file on your system.

Chapter 4

Installing the SDAccel Environment

This chapter explains the installation process for the SDAccel environment.

Preparing to Install the Tools

***Note*:** Before starting installation, you must complete the following steps.

1. Make sure your system meets the requirements described in the following topics:
   - Software Requirements
   - Hardware Requirements
2. Disable anti-virus software to reduce installation time.
3. Close all open programs before you begin installation.

Installing the SDAccel Environment

The SDAccel Development Environment installation is available on the Xilinx® Downloads Website.

***Note*:** There is no separate Vivado® version of the installation, only "stand alone" and SDx™ installations. Any difference in the installation happens if you use a different xsetup installer than the one from SDx or the one from the Vivado® Design Suite. Running the SDx xsetup installer would get you the same Vivado Design Suite as the 2018.2 Vivado xsetup installer, and offers the same options as the Vivado System Edition installation. The SDx xsetup installer is a super-set of the Vivado System Edition version.

Using the Web Installer

Xilinx recommends that you use the web installer.

Using the web installer you can pick and choose what you would like to install up front and that is the only data that you need download for installation. Also, in the case of a network failure, you can resume from where you last stopped, instead of starting again from the beginning.

***Note*:** The following devices are pre-selected in the individual installers:

- For the SDAccel™-specific web installer, 7 series, UltraScale™, and UltraScale+™ are pre-selected.
- For the combined SDx™ SFD (single file download) image, no devices are pre-selected.

Downloading and Installing the Full Installation File

If you downloaded the full product installation, decompress the file and run `xsetup` to launch the installation.

If you downloaded the web installer client, launch the downloaded file.You are prompted to log in and use your regular Xilinx login credentials to continue with the installation process.

Xilinx recommends the use of 7-zip or WinZip (v.15.0 or newer) to decompress the downloaded `tar.gz` file.

- The **Download and Install Now** choice allows you to select specific tools and device families on following screens, downloads only the files required to install those selections, and then installs them for you. After entering your login credentials, you can select between a traditional web-based installation or a full image download.
- The **Download Full Image** requires you to select a download destination and to choose whether you want a Windows only, Linux only, or an install that supports both operating systems. There are no further options to choose with the **Download Full Image** selection, and installation needs to be done separately by running the xsetup application from the download directory.

Batch Mode Installation Flow

The installer can run in an unattended batch process. To run unattended, a standard edition and install location must be specified or a configuration file must be present which tells the installer the install location and which of the tools, devices and options you wish to install. The installer has a mode in which it can generate a reference option file for you based on common configurations, which you can further edit to customize your installation.

Xilinx recommends that you generate this reference for each new quarterly release, so that new devices, tools, options or other changes will be accounted for in your options file.

To begin using batch mode, open a command shell and change to the directory where you have stored your extracted installer.

For Windows, open the command window with administrator privileges and run the xsetup.bat file, found in the `\bin` directory, and not `xsetup.exe` with the options below.

Generating a Configuration File

1. Run:

```
xsetup -b ConfigGen
```

   This will put you in an interactive mode where you will see the following menu. Choose the SDx™ IDE for SDAccel development environment edition.

2. After you make a selection, you will be prompted with the location/filename for your configuration file and the interactive mode exits. Below is a sample configuration file:

```
Edition=SDx Development Environments

# Path where Xilinx software will be installed.
Destination=/opt/Xilinx

# Choose the Products/Devices the you would like to install.
Modules=Built-in Platforms and associated devices for SDSoC:1,Zynq
UltraScale+ MPSoC:1,Virtex UltraScale+ HBM ES:0,Zynq-7000:1,Kintex
UltraScale+ ES:0,Kintex UltraScale+:1,ARM Cortex-A53:1,Spartan-7:1,ARM
Cortex-A9:1,ARM Cortex R5:1,Virtex UltraScale+ ES:0,System Generator
for DSP:0,Artix-7:1,Built-in Platforms and associated devices for
SDAccel:1,DocNav:1,Kintex-7:1,Virtex UltraScale+:1,Model
Composer:0,Zynq UltraScale+ RFSoC ES:0,Kintex UltraScale:1,Engineering
Sample Devices for Custom Platforms:0,Virtex UltraScale:1,Zynq
UltraScale+ MPSoC ES:0,MicroBlaze:1,Virtex-7:1

# Choose the post install scripts you'd like to run as part of the
finalization step. Please note that some of these scripts may require
user interaction during runtime.
InstallOptions=Acquire or Manage a License Key:0,Enable WebTalk for SDK
to send usage statistics to Xilinx:1,Enable WebTalk for Vivado to send
usage statistics to Xilinx (Always enabled for WebPACK license):1

## Shortcuts and File associations ##
# Choose whether Start menu/Application menu shortcuts will be created
or not.
CreateProgramGroupShortcuts=1

# Choose the name of the Start menu/Application menu shortcut. This
setting will be ignored if you choose NOT to create shortcuts.
ProgramGroupFolder=Xilinx Design Tools

# Choose whether shortcuts will be created for All users or just the
Current user. Shortcuts can be created for all users only if you run
the installer as administrator.
CreateShortcutsForAllUsers=0

# Choose whether shortcuts will be created on the desktop or not.
CreateDesktopShortcuts=1

# Choose whether file associations will be created or not.
CreateFileAssociation=1
```

Each option in the configuration file matches a corresponding option in the GUI. A value of `1` means that option is selected; a value of `0` means the option is unselected.

***Note*:** In this configuration file, by default there are no devices selected for installation (all devices have a value of 0). You MUST update a device to a value of 1 in order to install that device.

Running the Installer

Now that you have edited your configuration file to reflect your installation preferences, you are ready to run the installer. As part of the command line installer, you must indicate your acceptance of the Xilinx® End-User License Agreement Xilinx End-User License Agreement and Third Party End-User License Agreement Third Party End-User License Agreement, and confirm you understand the WebTalk Terms and Conditions.

WebTalk Terms and Conditions

The WebTalk Terms and Conditions, which you must agree to when running the installer, reads as follows:

By indicating `I AGREE`, I also confirm that I have read Section 13 of the terms and conditions above concerning WebTalk and have been afforded the opportunity to read the WebTalk FAQ posted at https://www.xilinx.com/webtalk. I understand that I am able to disable WebTalk later if certain criteria described in Section 13(c) apply. If they don't apply, I can disable WebTalk by uninstalling the Software or using the Software on a machine not connected to the internet. If I fail to satisfy the applicable criteria or if I fail to take the applicable steps to prevent such transmission of information, I agree to allow Xilinx to collect the information described in Section 13(a) for the purposes described in Section 13(b).

When using the command line, use the command-line switch, `-a` or `--agree`, to indicate your agreement to each of the above. If one of the above is left out of the list, or the agree switch is not specified, the installer exits with an error and does not install.

Batch Mode Installation

This is an example of the command line for a typical new installation using a configuration file.

```
xsetup --agree XilinxEULA,3rdPartyEULA,WebTalkTerms
--batch Install --config install_config.txt
```

If you wish to use one of Xilinx's default Edition configurations, you do not have to specify the `--config` option, but since the destination directory is included in the configuration file, you will be required to specify this on the command line.

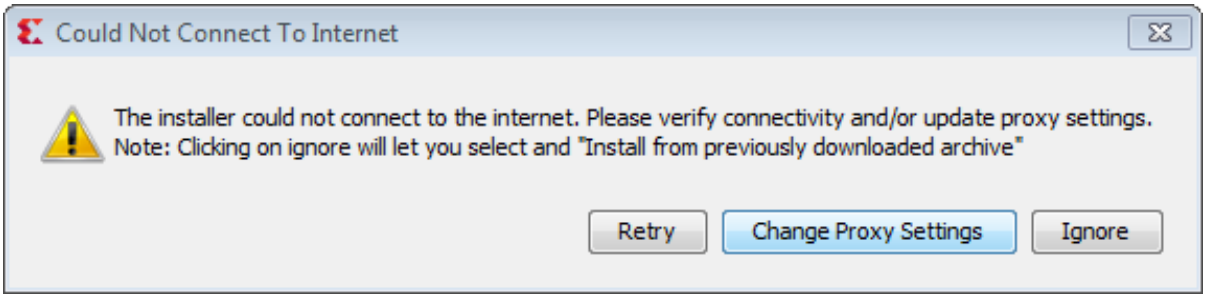
```
xsetup --agree 3rdPartyEULA,WebTalkTerms,XilinxEULA
--batch Install --edition "SDx Development Environments" --location
"C:\Xilinx"
```

The above command uses the default configuration options for the edition specified. To see the default configuration options, use the `-b ConfigGen` mode as described above. The batch mode of the SDx™ installer can also perform uninstallation and upgrades (adding additional tools and devices). For the full list of the options in the installer batch mode run `xsetup -h or xsetup --help`.

Verifying Connectivity

The installer connects to the Internet through the system proxy settings in Windows. These settings can be found under **Control Panel → Network and Internet → Internet Options**. For Linux users, the installer uses Firefox browser proxy settings (when explicitly set) to determine connectivity.

Figure 3: **Internet Connectivity**



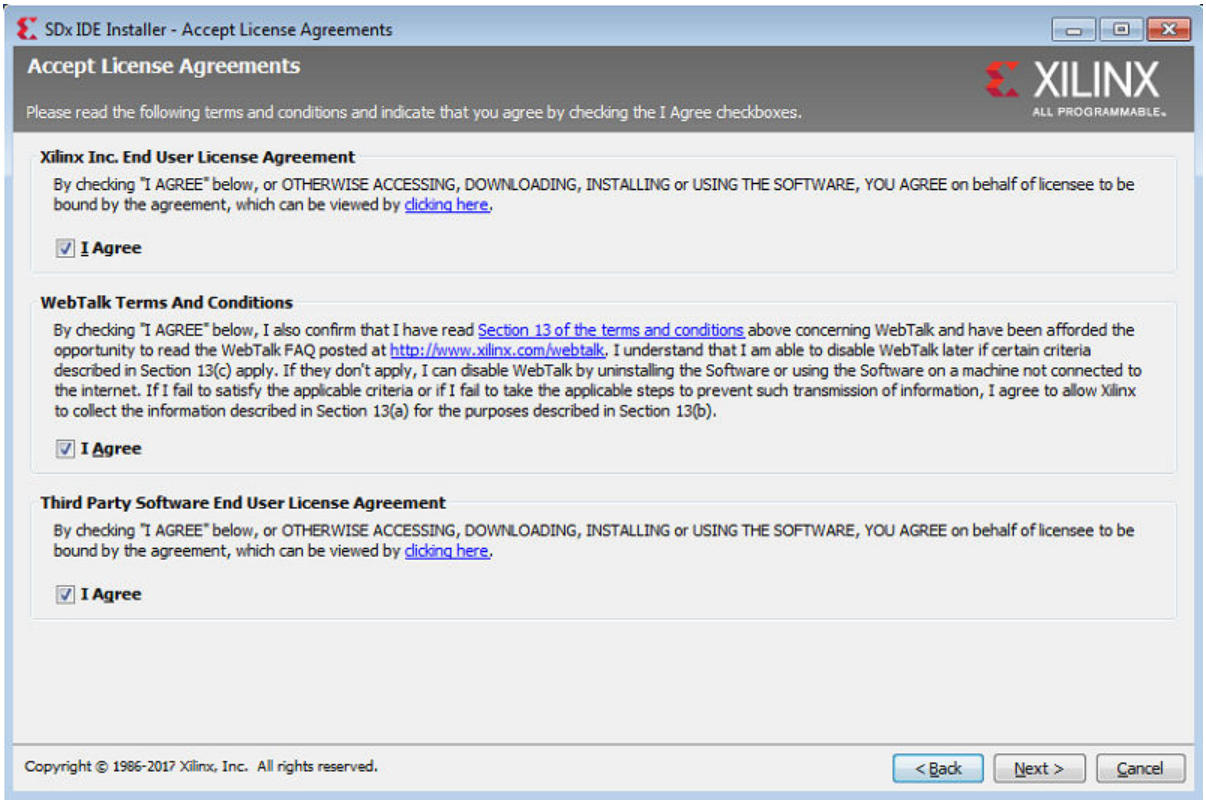
If there are connectivity issues, verify the following:

1. If you are using alternate proxy settings to the ones referred to, select the **Manual Proxy Configuration** option to specify the settings.
2. Check whether your company firewall requires a proxy authentication with a user name and password. If so, select the **Manual Proxy Configuration** option in the dialog box above.
3. For Linux users, if either the **Use System settings** or the **Auto detect settings** option is selected in the Firefox browser, you must manually set the proxy in installer.

Accepting License Agreements

Carefully read the license agreements before continuing with the installation. If you do not agree to the terms and conditions, cancel the installation and contact Xilinx.

Figure 4: **License Agreements**

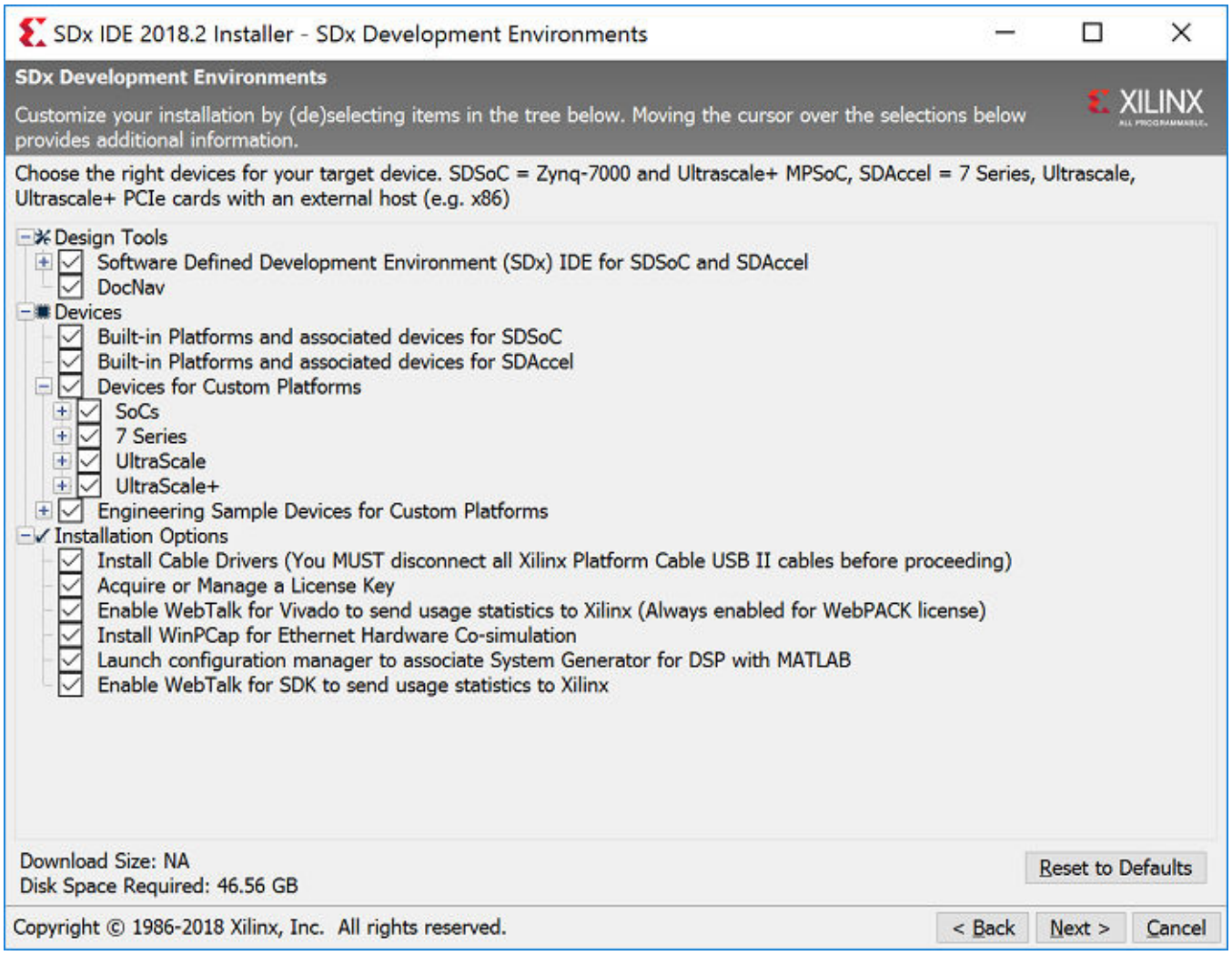


Selecting Tool and Device Options

Customize the installation by choosing the design tools, device families and installation options. Selecting only what you need helps to minimize the time taken to download and install the product. You will be able to add to this installation later by clicking **Add Design Tools or Devices** from either the operating system Start Menu or the **Vivado → Help** menu.

When you launch the installer for the product you want to use, the devices are preselected for you.

Figure 5: **Design Tools and Device Options**



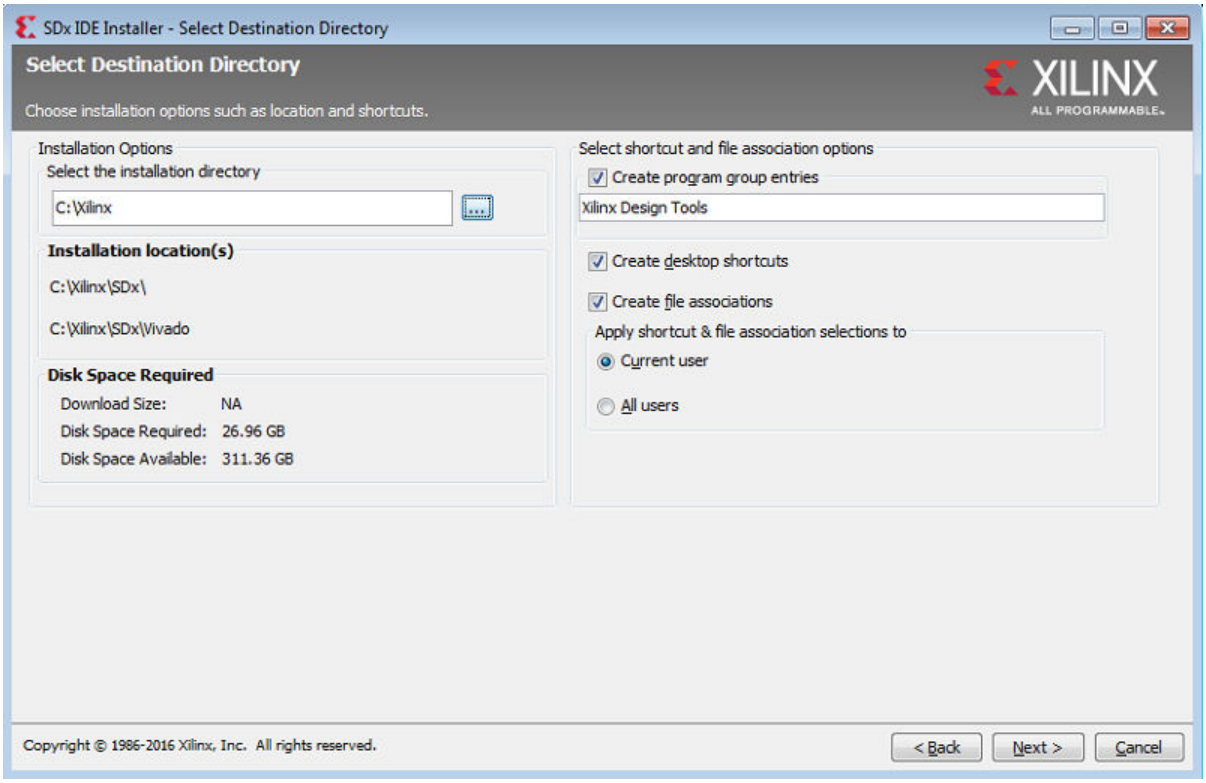
Setting Destination Directory and Installation Options

Define the installation directory for the software, as shown in the following figure.

***Note*:** The installation directory name must not contain any spaces in any part of the directory path.

You can customize the creation of the program group entries (Start Menu) and the creation of desktop shortcuts. The shortcut creation and file association options can be applied to the current user or all users.

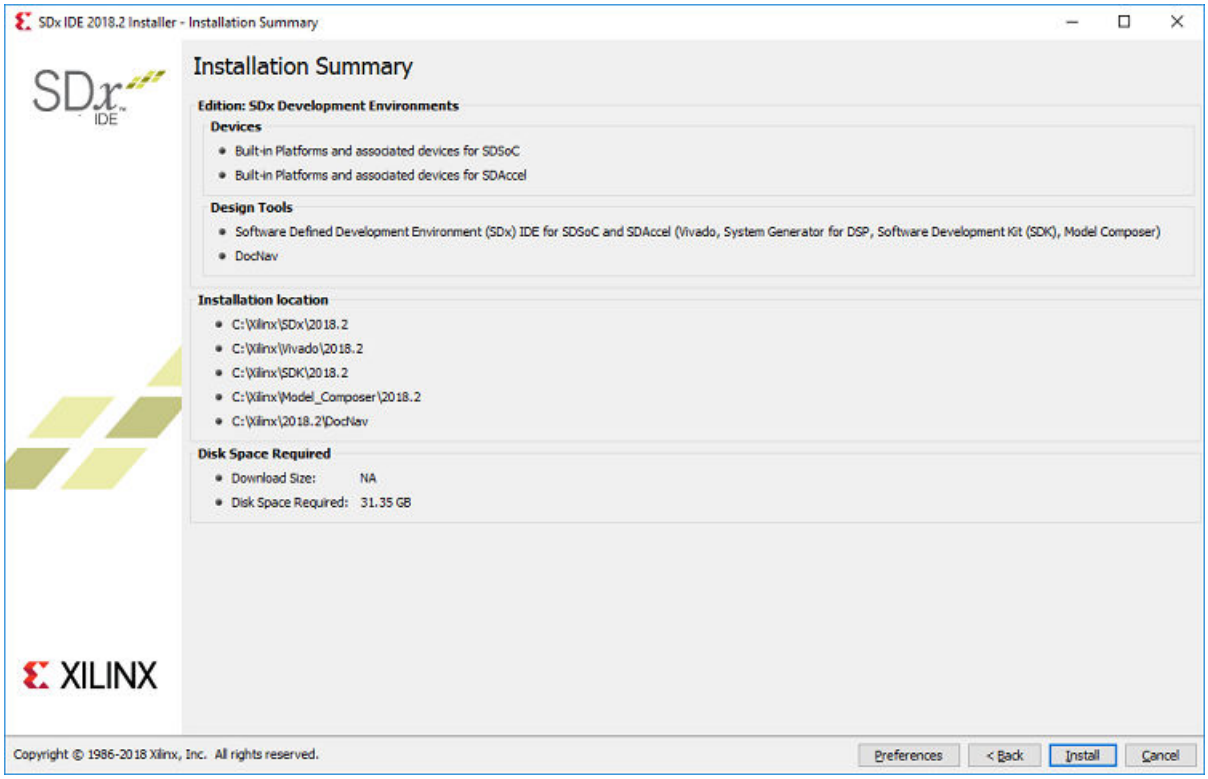
Figure 6: **Destination Directory and Installation Options**



Reviewing the Installation Details

Review the installation details shown in the **Installation Summary** screen.

Figure 7: **Installation Summary**



When you click **Install**, the installation process takes several minutes to complete.

Setting Up the Environment to Run SDx

1. To set up the environment to run SDx, source the file below so that the `sdx` command is in the `PATH`.

   For Linux:

   ```
   C Shell: source <SDX_INSTALL_DIR>/settings64.csh
   Bash: source <SDX_INSTALL_DIR>/settings64.sh
   ```

   For Windows:

   ```
   C:> <SDX_INSTALL_DIR>\SDx\2018.2\settings64.bat
   ```

Appendix A

Additional Resources and Legal Notices

Xilinx Resources

For support resources such as Answers, Documentation, Downloads, and Forums, see Xilinx Support.

Documentation Navigator and Design Hubs

Xilinx® Documentation Navigator provides access to Xilinx documents, videos, and support resources, which you can filter and search to find information. The Xilinx Documentation Navigator (DocNav) is installed with SDSoC™ and SDAccel™. To open it:

- On Windows, select **Start** → **All Programs** → **Xilinx Design Tools** → **DocNav**.
- At the Linux command prompt, enter `docnav`.

Xilinx Design Hubs provide links to documentation organized by design tasks and other topics, which you can use to learn key concepts and address frequently asked questions. To access the Design Hubs:

- In the Xilinx Documentation Navigator, click the **Design Hubs View** tab.
- On the Xilinx website, see the Design Hubs page.

***Note*:** For more information on Documentation Navigator, see the Documentation Navigator page on the Xilinx website.

References

1. *SDAccel Environments Release Notes, Installation, and Licensing Guide* (UG1238)
2. *SDAccel Environment User Guide* (UG1023)
3. *SDAccel Environment Profiling and Optimization Guide* (UG1207)
4. *SDAccel Environment Getting Started Tutorial* (UG1021)
5. SDAccel Development Environment web page
6. Vivado® Design Suite Documentation
7. *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994)
8. *Vivado Design Suite: Creating and Packaging Custom IP* (UG1118)
9. *Vivado Design Suite User Guide: Partial Reconfiguration* (UG909)
10. *Vivado Design Suite User Guide: High-Level Synthesis* (UG902)
11. *UltraFast Design Methodology Guide for the Vivado Design Suite* (UG949)
12. *Vivado Design Suite Properties Reference Guide* (UG912)
13. Khronos Group web page: Documentation for the OpenCL standard
14. Xilinx Virtex UltraScale+ FPGA VCU1525 Acceleration Development Kit
15. Xilinx Kintex UltraScale FPGA KCU1500 Acceleration Development Kit

Please Read: Important Legal Notices

Xilinx's limited warranty, please refer to Xilinx's Terms of Sale which can be viewed at https://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 https://www.xilinx.com/legal.htm#tos.

AUTOMOTIVE APPLICATIONS DISCLAIMER

AUTOMOTIVE PRODUCTS (IDENTIFIED AS "XA" IN THE PART NUMBER) ARE NOT WARRANTED FOR USE IN THE DEPLOYMENT OF AIRBAGS OR FOR USE IN APPLICATIONS THAT AFFECT CONTROL OF A VEHICLE ("SAFETY APPLICATION") UNLESS THERE IS A SAFETY CONCEPT OR REDUNDANCY FEATURE CONSISTENT WITH THE ISO 26262 AUTOMOTIVE SAFETY STANDARD ("SAFETY DESIGN"). CUSTOMER SHALL, PRIOR TO USING OR DISTRIBUTING ANY SYSTEMS THAT INCORPORATE PRODUCTS, THOROUGHLY TEST SUCH SYSTEMS FOR SAFETY PURPOSES. USE OF PRODUCTS IN A SAFETY APPLICATION WITHOUT A SAFETY DESIGN IS FULLY AT THE RISK OF CUSTOMER, SUBJECT ONLY TO APPLICABLE LAWS AND REGULATIONS GOVERNING LIMITATIONS ON PRODUCT LIABILITY.

Copyright