Intel® FPGA SDK for OpenCL™

Intel® Stratix® 10 GX FPGA Development Kit Reference Platform Porting Guide

Updated for Intel® Quartus® Prime Design Suite: **19.1**

Online Version
Send Feedback

UG-20102

ID: 683809
Version: 2022.03.28

Contents

683809 | 2022.03.28

Send Feedback

1. Intel® FPGA SDK for OpenCL™ Intel® Stratix® 10 GX FPGA Development Kit Reference Platform Porting Guide

The *Intel® Stratix® 10 GX FPGA Development Kit Reference Platform Porting Guide* describes procedures and design considerations for modifying the Intel Stratix 10 GX FPGA Development Kit Reference Platform (s10_ref) into your own Custom Platform for use with the Intel FPGA Software Development Kit (SDK) for OpenCL™ (1) (2).

1.1. Introduction to Intel Reference Platform

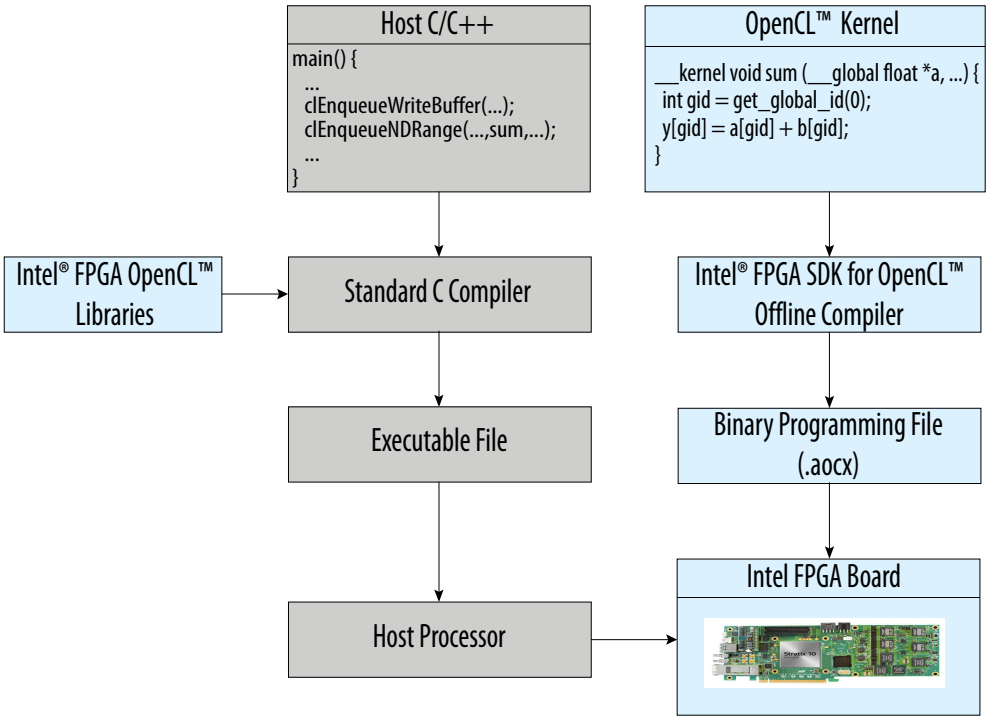
The Intel FPGA SDK for OpenCL provides you an environment to target FPGAs while abstracting FPGA details.

(1) OpenCL and the OpenCL logo are trademarks of Apple Inc. used by permission of the Khronos Group™.

(2) The Intel FPGA SDK for OpenCL is based on a published Khronos specification, and has passed the Khronos Conformance Testing Process. Current conformance status can be found at www.khronos.org/conformance.

It allows you to target Intel FPGA devices either on reference platforms provided by Intel or Intel board partners, or on your own custom platforms. A typical setup for using the SDK is illustrated in the following image:

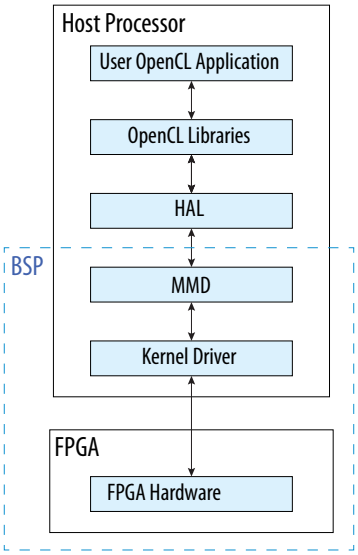
Figure 1. Setup for Using Intel FPGA SDK for OpenCL



The setup consists of a host application running on the host processor and offloading kernel tasks to the FPGA. The OpenCL kernel is converted to a hardware circuit by the SDK compiler. Leveraging this capability for your FPGA platform requires an Intel FPGA SDK for OpenCL-compatible Board Support Package (BSP). The BSP describes the reference platform to the SDK.

The following illustration depicts segments of the Intel FPGA SDK for OpenCL solution:

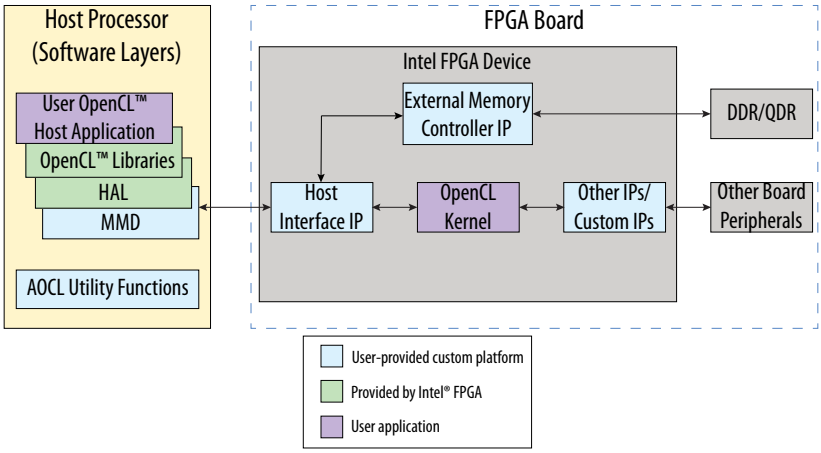
Figure 2. Layers of Intel FPGA SDK for OpenCL



Your host application communicates with the BSP layers through the Hardware Abstraction Layer (HAL). A typical Intel BSP consists of software layers and a hardware project created using the Intel Quartus® Prime Pro Edition software. The hardware project consists of FPGA board peripheral IPs and custom IPs.

The following illustration depicts the hardware project components, and how these components communicate with software layers:

Figure 3. Intel FPGA SDK for OpenCL Complete Solution



In Intel FPGA SDK for OpenCL Complete Solution, left side depicts the host application running on host processor while right side depicts the FPGA hardware acceleration board. If you are developing a custom platform to run your software applications, then you must create a custom BSP for your platform. In that case, everything that appears in blue in the image must be included in your custom BSP as follows:

- On the FPGA side, your custom BSP must include all hardware necessary to communicate with the host and the memory, that is, DDR and/or QDR memory interfaces, the DMA host interface (which can be PCIe), and any streaming interfaces to be implemented as channels. The Intel FPGA SDK for OpenCL compiles your OpenCL kernel into a data flow circuit, connects to the BSP hardware components, and generates an FPGA image for this combined circuit, which is used to configure the FPGA.
- On the host side, your custom BSP must provide the Memory Mapped Device layer (MMD) (in the form of a library) to facilitate communication between OpenCL libraries and your hardware. When you compile your host application, the host application links with both the Intel FPGA SDK for OpenCL and MMD libraries to form a host executable.

Intel provides reference BSPs for Intel FPGA development kits. Most of these reference BSPs are included in the installed directory for Intel FPGA SDK for OpenCL:

Table 1. Reference BSPs for Intel FPGA development kits

Reference BSP	Install Path
Intel Arria® 10 GX FPGA Reference Platform	Installed with the Intel FPGA SDK for OpenCL in *INTELFPGAOCLSDKROOT*/board/a10_ref.
Intel Stratix 10 GX FPGA Reference Platform	Installed with the Intel FPGA SDK for OpenCL in *INTELFPGAOCLSDKROOT*/board/s10_ref.
Intel Arria 10 SoC FPGA Reference Platform	Installed with the Intel FPGA SDK for OpenCL in *INTELFPGAOCLSDKROOT*/board/a10soc.

Attention: Cyclone® V SoC and Stratix V GX reference platforms are also available within the Intel FPGA SDK for OpenCL installation of version 18.0 and earlier.

You can use one of the above BSPs as a reference to get started with custom BSP development. You can also use reference platforms from one of the following Intel FPGA's preferred board partners and download BSP from their website if it matches your hardware requirements:

- Terasic Inc.
- REFLEX CES

Related Information

Building Custom Platforms for Intel® FPGA SDK for OpenCL™: BSP Basics

1.2. Intel Stratix 10 GX FPGA Development Kit Reference Platform: Prerequisites

The *Intel Stratix 10 GX FPGA Development Kit Reference Platform Porting Guide* assumes that you are an experienced FPGA designer familiar with Intel's FPGA design tools and concepts.

Prerequisites for the `s10_ref` Reference Platform:

- An Intel Stratix 10-based accelerator card with working PCI Express* (PCIe*) and memory interfaces

  Attention: The native Stratix 10 GX FPGA Development Kit does not automatically work with the SDK. Before using the Stratix 10 GX FPGA Development Kit with the SDK, you must setup the board by following the steps provided in the bring-up guide (included in the *INTELFPGAOCLSDKROOT*`/board/s10_ref/bringup` directory) or contact your field applications engineer or regional support center representative to configure the development kit for you.
- Intel Quartus Prime Pro Edition software
- Designing with Logic Lock regions

General prerequisites:

- FPGA architecture, including clocking, global routing, and I/Os
- High-speed design
- Timing analysis
- Platform Designer design and Avalon® interfaces
- Tcl scripting
- PCIe
- DDR4 external memory

This document also assumes that you are familiar with the following Intel FPGA SDK for OpenCL-specific tools and documentation:

- Custom Platform Toolkit and the *Intel FPGA SDK for OpenCL Custom Platform Toolkit User Guide*
- Intel Arria 10 Reference Platform (a10_ref) and the *Intel FPGA SDK for OpenCL Intel Arria 10 GX FPGA Development Kit Reference Platform Porting Guide*

  The whole software stack in the `s10_ref` is derived from the `a10_ref` Reference Platform.

Related Information

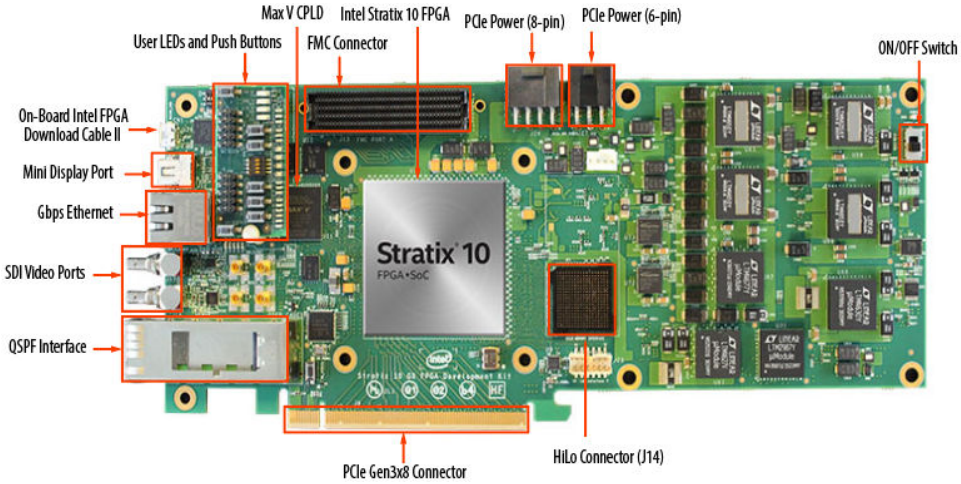
- Intel FPGA SDK for OpenCL Custom Platform Toolkit User Guide
- Intel FPGA SDK for OpenCL Intel Arria 10 GX FPGA Development Kit Reference Platform Porting Guide
- Intel FPGA SDK for OpenCL Intel Arria 10 SoC Development Kit Reference Platform Porting Guide
- Intel FPGA SDK for OpenCL Intel Cyclone V SoC Development Kit Reference Platform Porting Guide
- Intel FPGA SDK for OpenCL Intel Stratix V Network Reference Platform Porting Guide

1.3. Features of the Intel Stratix 10 GX FPGA Development Kit Reference Platform

Prior to designing an Intel FPGA SDK for OpenCL Custom Platform, decide on design considerations that allow you to fully utilize the available hardware on your computing card.

The Intel Stratix 10 GX FPGA Development Kit Reference Platform targets a subset of the hardware features available in the Intel Stratix 10 GX FPGA Development Kit.

Figure 4. Hardware Features of the Intel Stratix 10 GX FPGA Development Kit



Features of the s10_ref Reference Platform:

- OpenCL Host

  The s10_ref Reference Platform uses a PCIe-based host that connects to the Intel Stratix 10 PCIe Gen3x8 hard IP core.

- OpenCL Global Memory

  The hardware provides one 2-gigabyte (GB) DDR4 SDRAM daughtercard that is mounted on the HiLo connector and instantiated in the design using Intel Stratix 10 External Memory Interface IP (J14 in Hardware Features of the Intel Stratix 10 GX FPGA Development Kit).

- FPGA Programming

  Via external cable and the Intel Stratix 10 GX FPGA Development Kit's on-board Intel FPGA Download Cable II interface.

- Guaranteed Timing

  The s10_ref Reference Platform relies on the Intel Quartus Prime Pro Edition compilation flow to provide guaranteed timing closure. The timing-clean s10_ref Reference Platform is preserved in the form of a precompiled post-fit netlist (that is, the `base.qdb` Intel Quartus Prime Database Export File). The Intel FPGA SDK for OpenCL Offline Compiler imports this preserved post-fit netlist into each OpenCL kernel compilation.

1.3.1. Intel Stratix 10 GX FPGA Development Kit Reference Platform Board Variants

The Intel Stratix 10 GX FPGA Development Kit Reference Platform has one board variant (that is, s10gx) that targets the Intel Stratix 10 GX FPGA Development Kit containing the Intel Stratix 10 DDR4-1866 SDRAM.

To compile your OpenCL kernel for a specific board variant, include the `-board=<board_name>` option in your `aoc` command (for example, `aoc -board=s10gx myKernel.cl`).

Related Information

Compiling a Kernel for a Specific FPGA Board and Custom Platform (-board=<board_name>) and (-board-package=<board_package_path>)

1.4. Contents of the Intel Stratix 10 GX FPGA Development Kit Reference Platform

Familiarize yourself with directories and files within the Intel Stratix 10 GX FPGA Development Kit Reference Platform (*INTELFPGAOCLSDKROOT*`/board/s10_ref` because they are referenced throughout this document.

Table 2. Highlights of the Intel Stratix 10 GX FPGA Development Kit Reference Platform Directory

Windows File or Folder	Linux File or Directory	Description
`board_env.xml`	`board_env.xml`	eXtensible Markup Language (XML) file that describes the Reference Platform to the Intel FPGA SDK for OpenCL.
`bringup`	`bringup`	Contains initialization binaries and Intel Stratix 10 Development Kit Initialization guide (`S10_DevKit_Initialization`).
`hardware`	`hardware`	Contains the Intel Quartus Prime project templates for the s10gx board variant. See Contents of the s10gx Directory for a list of files in this directory.
`windows64`	`linux64`	Contains the MMD library, kernel mode driver, and executable files of the SDK utilities (that is, `install`, `uninstall`, `flash`, `program`, `diagnose`) for your 64-bit operating system.
`source`	`source`	For Windows, the `source` folder contains source codes for the MMD library and SDK utilities. The MMD library and the SDK utilities are in the `windows64` folder. For Linux, the `source` directory contains source codes for the MMD library and SDK utilities. The MMD library and the SDK utilities are in the `linux64` directory.
`scripts`	`scripts`	Contains the `find_jtag_cable.tcl` script that is useful in identifying the cable and index number required for FPGA programming.

Table 3. Contents of the s10gx Directory

The following table lists the files in the *INTELFPGAOCLSDKROOT*/board/s10_ref/hardware/s10gx directory, where *INTELFPGAOCLSDKROOT* points to the location of the SDK installation.

File	Description
mem.qsys	Platform Designer system that, together with the .ip files in the ip/mem/ sub-directory, implements the mem component. It can be recognized as the memory subsystem instantiated in the board.qsys. It contains EMIF hard IP, AVMM S10 CCBs (Clock Crossing Bridge), ACL Uniphy Status and ACL SW Reset (Calibration) components.
board.qsys	Platform Designer system that implements the board interfaces (that is, the static region) of the OpenCL hardware system.
device.tcl	Tcl file that is included in all revisions and contains all device-specific information (for example, device family, ordering part number (OPN), voltage settings, pin assignments and so on). It is sourced in other QSF files, such as opencl_bsp_ip.qsf and flat.qsf.
opencl_bsp_ip.qsf	Intel Quartus Prime Settings File that collects all the required .ip files in a unique location. During flat and base revision compilations, the board.qsys and mem.qsys related IP files are added to the opencl_bsp_ip.qsf file.
flat.qsf	Intel Quartus Prime Settings File for the flat project revision. This file includes all common settings, such as VID and global signal settings, that are used in other revisions of the project (that is, base and top). The base.qsf and top.qsf files include, by reference, all settings in the flat.qsf file. The Intel Quartus Prime software compiles the flat revision with minimal constraints. The flat revision compilation does not generate a base.qar file that you can use for future import compilations and does not implement the guaranteed timing flow. It is used to make edits and check functionality of the design.
base.qsf	Intel Quartus Prime Settings File for the base project revision. This file includes, by reference, all the settings in the flat.qsf file. The Intel Quartus Prime Pro Edition software compiles this base project revision from source code unlike the top compile that uses the base.qar output.
top.qsf	Intel Quartus Prime Settings File for the SDK-user compilation flow (import compilation flow).
top.v	Top-level Verilog Design File for the OpenCL hardware system.
top.sdc	Synopsys Design Constraints File that contains board-specific timing constraints.
top_post.sdc	Platform Designer and Intel FPGA SDK for OpenCL IP-specific timing constraints.
ip/mem/<*file_name*>	Directory containing the .ip files that the Intel Quartus Prime Pro Edition software needs to parameterize the mem component. Along with mem.qsys, files in this directory are required for flat and base revision compiles. These are added to the flow by pre_flow_pr.tcl.
ip/board/<*file_name*>	Directory containing the .ip files that the Intel Quartus Prime Pro Edition software needs to parameterize the board instance. Along with board.qsys, files in this directory are required for flat and base revision compiles. These are added to the flow by pre_flow_pr.tcl

continued...