AN 988: Using the Board-Aware Flow

in the Intel® Quartus® Prime Pro Edition Software

Updated for Intel® Quartus® Prime Design Suite: **22.4**

Answers to Top FAQs:

Contents

757339 | 2023.01.09

Send Feedback

1. Using the Board-Aware Flow in the Intel® Quartus® Prime Pro Edition Software

This application note demonstrates using the Intel® Quartus® Prime software board-aware flow. You can use the board-aware flow to accelerate the process of appropriately configuring, connecting, and validating IP for a target board.

What is the Intel Quartus Prime Software Board-Aware Flow?

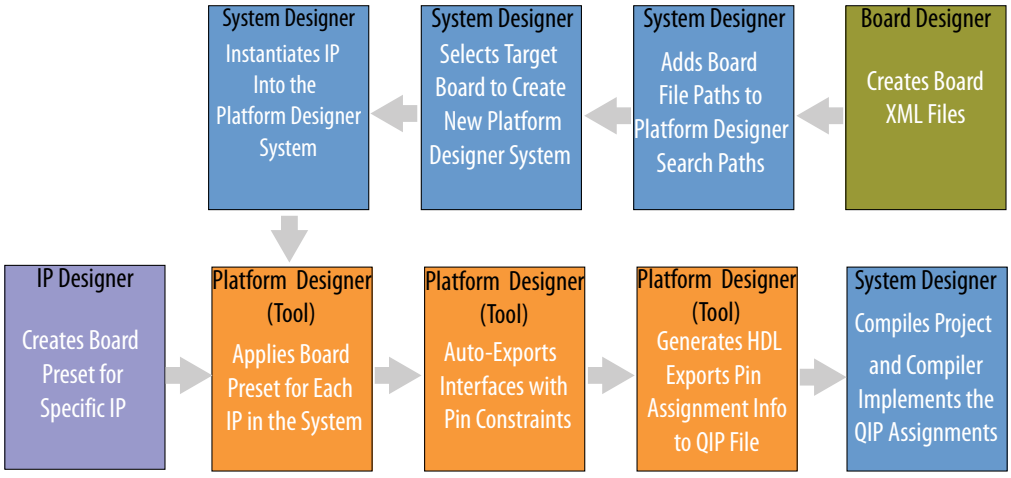
The board-aware flow simplifies the application of appropriate parameters and pin assignments for the instantiated IP in your project, thereby reducing the chance of configuration errors. You can also save your preferred and verified board and IP configurations for reuse in other projects that target the same IP or board.

In the board-aware flow, you can optionally start your project from a pre-verified design example and target a specific Intel FPGA development board, rather than just a specific device. You can then create IP presets targeting the specific board. The Intel Quartus Prime Platform Designer system integration tool is also board-aware, allowing you to automatically set pin assignments and export appropriate system interfaces for the target board.

The complete process of configuring, connecting, and validating IP for a target board is typically implemented by multiple developers performing specialized tasks. For example:

- An IP designer that creates board presets for specific IP.
- A system designer that combines individual IP components into a system.
- A board designer that creates the XML board files defining the target board.

Figure 1. Board-Aware Flow Typical Tasks and Roles



The board-aware flow helps to ensure the proper hand-off, consistency, and reuse of configuration options across multiple projects, developers, and boards.

This application note guides you through the following steps in the board-aware flow:

- Step 1: Specify a Target Board for the Project
- Step 2: Create a New Board File
- Step 3: Create IP Presets for the Board
- Step 4: Create a Design Using Board and Preset Files
- Step 5: Compile and Verify the Design
- Step 6: Configure the Board with Software

1.1. Document Prerequisites

Use of this application note requires that you already have the following:

- Installation of Intel Quartus Prime Pro Edition version 22.4, with Intel Agilex™ device support.
- Download and extract supporting design example files, as Downloading and Extracting Supporting Files on page 4 describes.
- Basic familiarity with the Intel Quartus Prime Pro Edition FPGA implementation flow and use of the Platform Designer tool.
- Connection to the internet for optional download of online Intel Quartus Prime FPGA design examples.

Related Information

- Intel Quartus Prime Pro Edition User Guide: Getting Started
- Intel Quartus Prime Pro Edition User Guide: Platform Designer

1.2. Downloading and Extracting Supporting Files

1. Download and extract the Board-Aware Design Example Supporting Files to a directory on your computer. Do not use spaces in the directory path name.
2. View the extracted tutorial design files and directory structure. `board_aware_example_agilex.zip` contains the following files:

Table 1. Supporting Design Example Files

File	Description
`pin_pio.tcl`	Tcl pin constraints file that contains appropriate pin assignments for the system. You can optionally load this file rather than manual entry.
`pio_led.out.sdc`	Synopsys Design Constraints file that contains appropriate timing constraints for the completed design.
`issp.ip`	Represents the In-System Sources and Probes Intel FPGA IP in the design for use in debugging the design.
`resetrelease.ip`	Represents the Reset Release Intel FPGA IP in the design. This IP outputs `nINIT_DONE` after finishing device initialization. User mode initialization can begin as soon as the `nINIT_DONE` signal asserts.

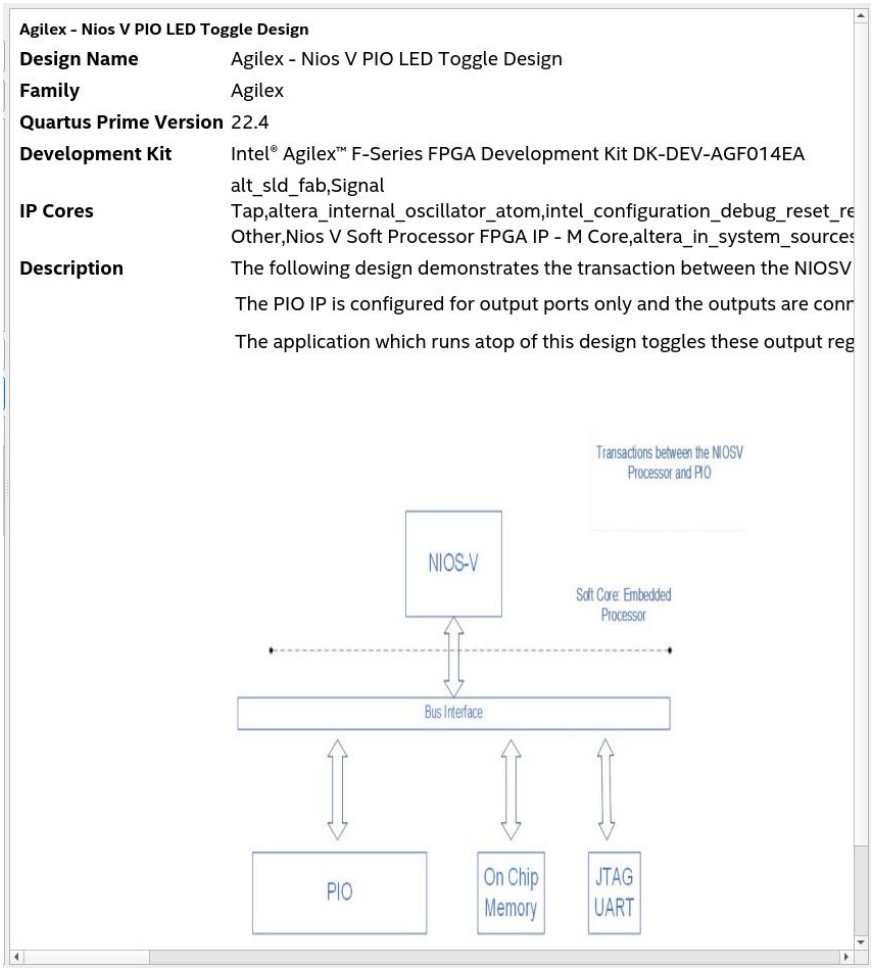
1.3. Accessing Intel FPGA Design Examples

You can optionally base your design project on a pre-verified Intel FPGA design example that targets a specific Intel FPGA board or development kit, or you can start with an empty project. Access available design examples using any of the following methods:

- **Pre-installed design examples**—you can immediately access the design examples that install along with the Intel Quartus Prime software installation at: `<quartus>\acds\quartus\common\board_designs`.
- **Online design examples**—you can access design examples hosted online, which include designs from the Intel FPGA Design Store.
- **Downloaded design examples**—you can access previously downloaded design examples, or any design example that you store in a local drive, under downloaded design examples.

The steps in this application note use the pre-installed Agilex - Nios V PIO LED Toggle Design to illustrate the board-aware flow.

Figure 2. Agilex - Nios V PIO LED Toggle Design Description



1.3.1. Accessing Pre-Installed Design Examples

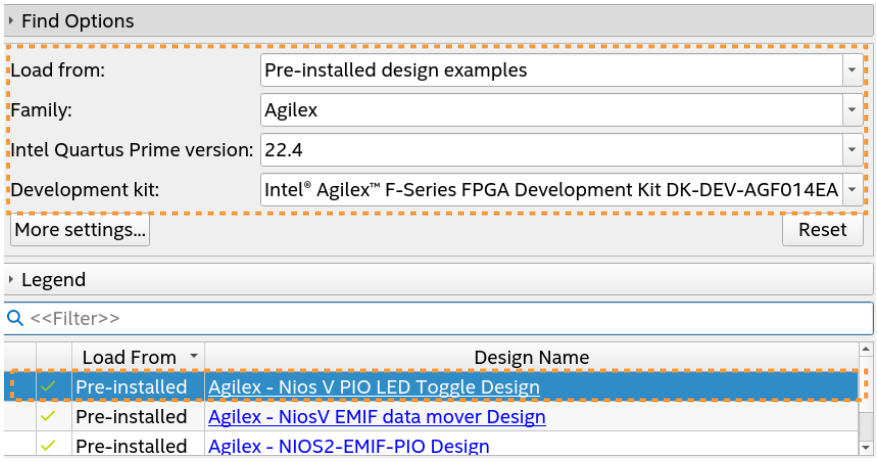
The Intel Quartus Prime software installation includes design examples for your immediate use. You can create a new project based on one of the available Intel FPGA design examples.

You can access the pre-installed design examples from the Intel Quartus Prime software **Home** page, from the **New Project Wizard**, or with the **File ➤ Open Example Project** menu command.

To create a new project based on a pre-installed design example, follow these steps:

1. Start the Intel Quartus Prime Pro Edition software.
2. Click **File ➤ Open Example Project**. The **Design Example** page of the **New Project Wizard** opens.
3. Under **What is the working directory for this project?**, specify the directory location to store your project files.
4. Under **Find Options**, specify the following settings to filter the list of design examples for the target device and board:
   a. In **Load from**, select **Pre-Installed design examples**.
   b. In **Family**, select **Agilex**.
   c. In **Intel Quartus Prime version**, select **22.4**.
   d. In **Development kit**, select **Intel Agilex F-Series FPGA Development Kit DK-DEV-AGF014EA**.

Figure 3. Location of Pre-Installed Design Examples



5. Under **Design name**, select the **Agilex - Nios V PIO LED Toggle Design** design.
6. Click **Next**, and then click **Finish**. The Agilex - Nios V PIO LED Toggle Design extracts to the working directory and opens in the Intel Quartus Prime software.

1.3.2. Accessing Downloaded Design Examples

You can create a new project based on a design example that you have previously downloaded. To test this method, you can download a design example `.par` file from the Intel FPGA Design Store, or from another online repository that stores design examples in `.par` format, into your working directory. Platform Designer also classifies designs that you create yourself and store in a local drive as downloaded examples.

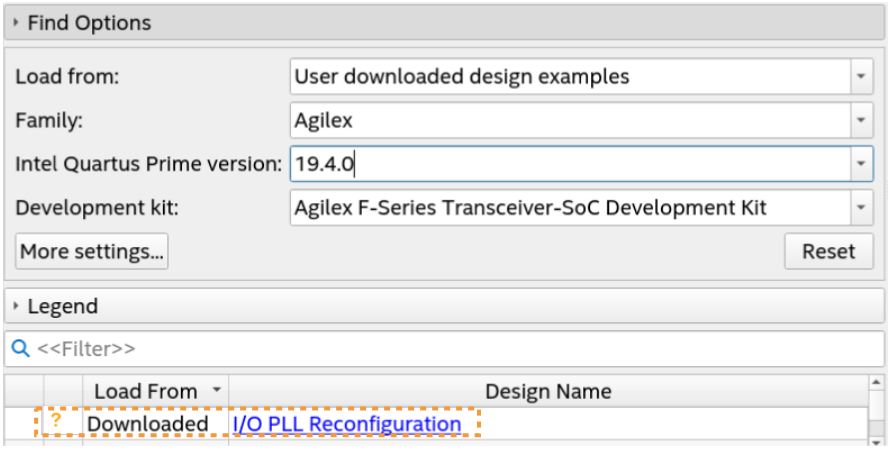
To create a new project based on a downloaded design example, follow these steps:

1. Start the Intel Quartus Prime Pro Edition software.
2. Click **File ➤ Open Example Project**. The **Design Example** page of the **New Project Wizard** opens.
3. Under **What is the working directory for this project?**, specify the directory location to store your project files.
4. Click the **Design Store** button. The Design Store web page displays an unfiltered list of available design examples.

Figure 4. Design Store Listing Available Design Examples

5. From the left navigation menu, click on **Agilex FPGAs and SOC FPGAs** to filter the list of design examples.
6. Click the **I/O PLL Reconfiguration** design example.
7. Download the `.par` file for the I/O PLL Reconfiguration design example to your working directory. Accept the license agreement when prompted.
8. On the **Design Example** page, click the **More Settings** button.
9. Click the **Design Examples Search Locations** tab.
10. In the **Design examples search directories** box, specify the working directory to store your downloaded design example `.par` file from step 3.
11. Under **Find Options**, specify the following settings:
    a. In **Load from**, select **Downloaded design examples**.
    b. In **Family**, select **Agilex**.
    c. In **Intel Quartus Prime version**, select **19.4**.
    d. In **Development kit**, select **Agilex F-series Transceiver-SoC Development Kit**.
12. In the design example list, select the **I/O PLL Reconfiguration** design. The ? symbol indicates that the design is not yet validated for the current Intel Quartus Prime software version.

Figure 5. Downloaded Agilex I/O PLL Reconfiguration Design



13. Click **Next**, and then click **Finish**. The I/O PLL Reconfiguration design extracts to the working directory and opens in the Intel Quartus Prime software.

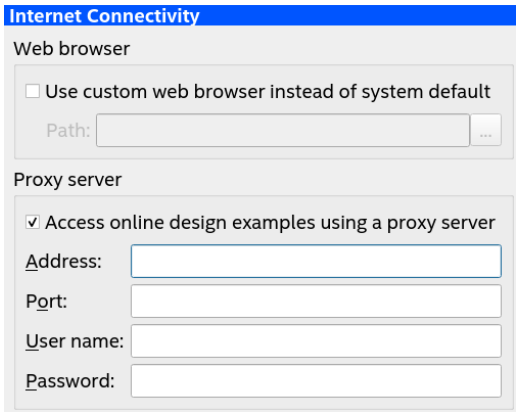
1.3.3. Accessing Online Design Examples

You can create a new project based on a design example that you access from an online repository. To use this method, you may need to specify a proxy server for access and the download path.

To create a new project based on an online design example, follow these steps:

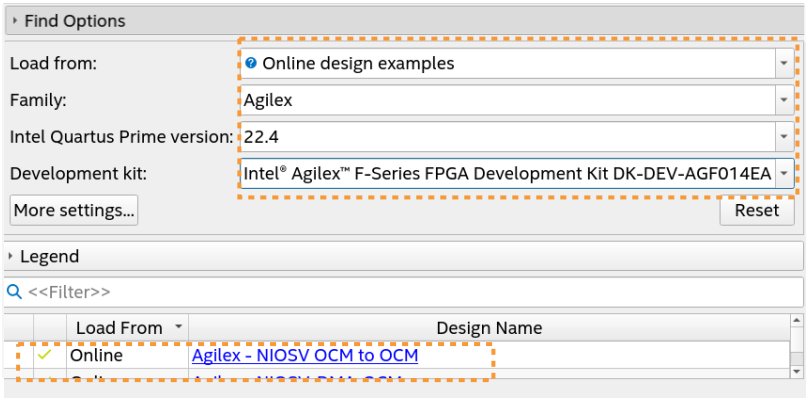
1. Start the Intel Quartus Prime Pro Edition software.
2. Click **File ➤ Open Example Project**. The **Design Example** page of the **New Project Wizard** opens.
3. Under **What is the working directory for this project?**, specify the directory location to store your project files.
4. Click the **More Settings** button. The **Options** dialog box opens with the **Internet Connectivity** tab open by default.

Figure 6. Intel Quartus Prime Software Internet Connectivity Settings



5. If your internet connection requires a proxy server (using VPN), turn on the **Access online design examples using a proxy server** option, and then specify your proxy **Address**, **Port**, **User name**, and **Password**. If your internet connection does not require a proxy server, skip this step.
6. On the **Design Example Search Locations** tab, specify the **Download path** for download of the design example `.par` file.
7. Click **OK**.
8. Under **Find Options**, specify the following settings:
   a. In **Load from**, select **Online design examples**.
   b. In **Family**, select **Agilex**.
   c. In **Intel Quartus Prime version**, select **22.4**.
   d. In **Development kit**, select **Intel Agilex F-Series FPGA Development Kit DK-DEV-AGF014EA**.
9. In the design example list, select the **Agilex -NIOSV OCM to OCM** design.

Figure 7. Online Agilex -NIOSV OCM to OCM Design



10. Click **Next**, and then click **Finish**. The Agilex -NIOSV OCM to OCM design extracts to the working directory and opens in the Intel Quartus Prime software.

1.4. Using the Board-Aware Flow in Platform Designer

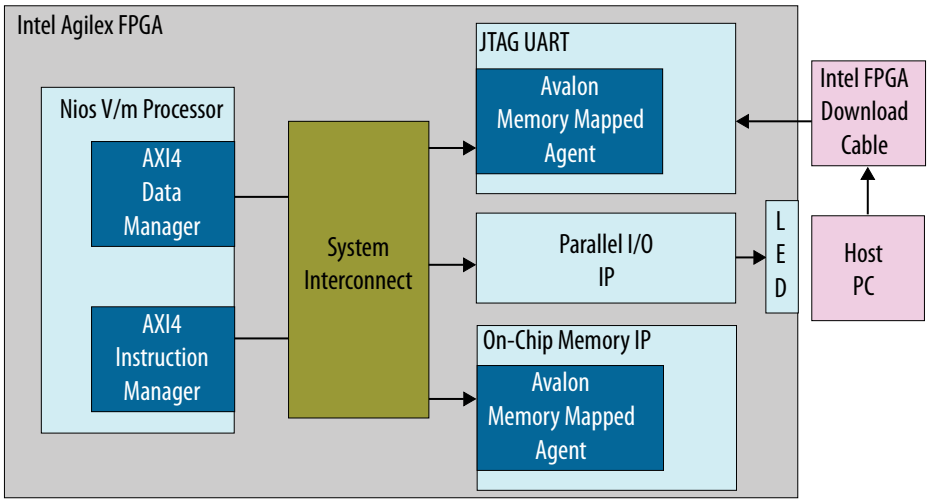
The board-aware flow is fully integrated within the Platform Designer GUI to simplify and accelerate the process of appropriately configuring IP and systems for a target board.

Platform Designer allows you to create a system that targets a specific development board, rather than only targeting a specific FPGA device. When you target a specific development board, Platform Designer is *aware* of the target board (board-aware) which simplifies the IP parameterization, pin assignments, and export of interfaces for the system.

The board-aware flow uses IP presets together with a board definition file that specifies the details of a target board. You can use (and reuse) the board definition and IP presets to automatically include the appropriate IP pin assignments, parameters, and exported interfaces for the target development board during system generation.

The following example board-aware flow example creates an LED system based on the PIO Intel FPGA IP design example:

Figure 8. PIO IP-Based LED Design Block Diagram



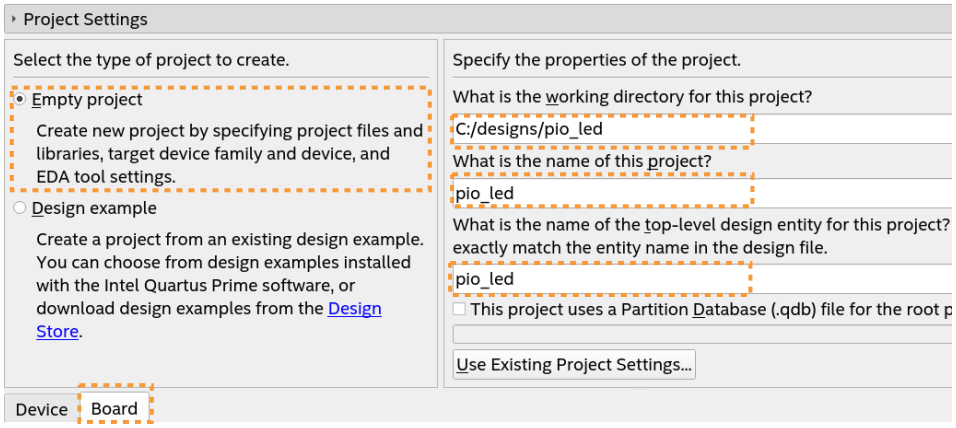
- Step 1: Specify a Target Board for the Project
- Step 2: Create a New Board File
- Step 3: Create IP Presets for the Board
- Step 4: Create a Design Using Board and Preset Files
- Step 5: Compile and Verify the Design
- Step 6: Configure the Board with Software

1.4.1. Step 1: Specify a Target Board for the Project

To create an Intel Quartus Prime project that targets a specific board containing the target FPGA device, follow these steps:

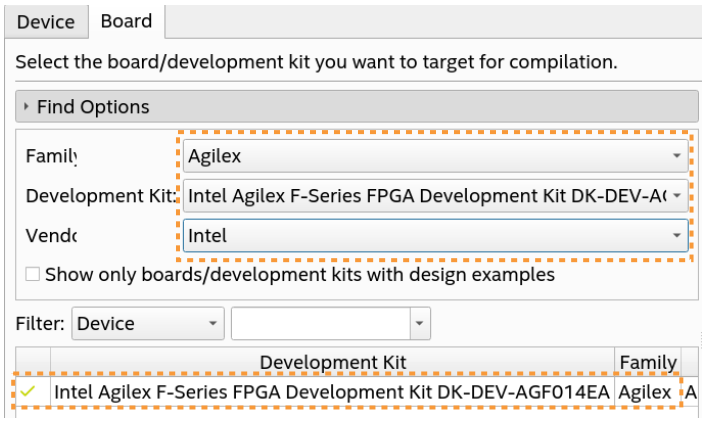
1. Start the Intel Quartus Prime Pro Edition software.
2. Click **File ➤ New Project Wizard**. If the **Introduction** page appears, click **Next**.
3. Under **Project Settings**, enter `pio_led` as the **working directory**, the **name of the project**, and the **top-level design entity**.

Figure 9. New Project Wizard



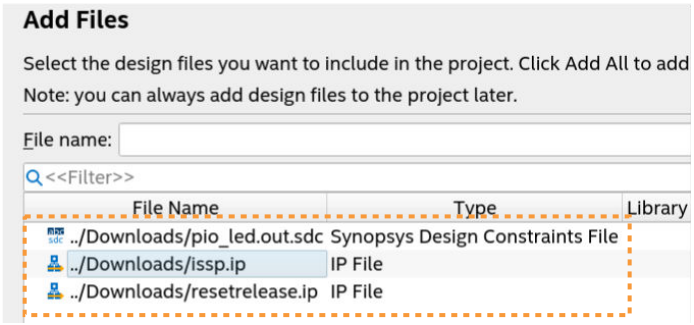
4. Click the **Board** tab below the settings you just specified. The **Board** tab allows you to target a specific FPGA device board, rather than just a specific FPGA device.
5. Under **Find Options**, specify the following filters:
   - **Family**—select **Agilex**.
   - **Development Kit**—select **Intel Agilex F-Series FPGA Development Kit DK-DEV-AGF014EA**.
   - **Vendor**—select **Intel**.

Figure 10. Board Tab Settings



6. Click the **Intel Agilex F-Series FPGA Development Kit** in the list. The board details appear in the right pane.
7. Click **Next**. The **Add Files** page appears.
8. Next to **File name**, click browse (**...**) to add the `pio_led.out.sdc`, `issp.ip`, and `resetrelease.ip` files that support this application note, as Download Supporting Files describes.

Figure 11. Adding IP and Constraint Files to Project



9. Click **Next**. the **EDA** page appears. Retain the default settings and click **Next** again to view the **Summary** page. The **Summary** page displays the details of your project based on the board and other options you specified.
10. When you are done viewing the **Summary**, click **Finish**. The PIO_LED project opens in the Intel Quartus Prime software.

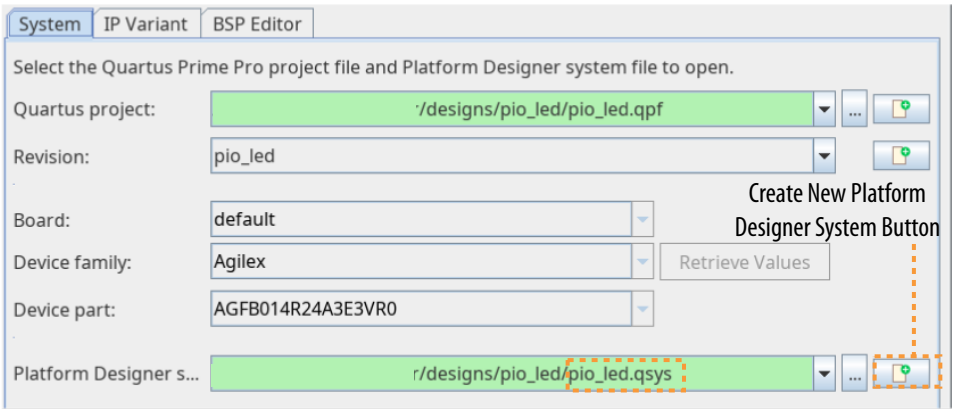
1.4.2. Step 2: Create a New Board File

You can define a board file that specifies the details of the target board. You can reuse this file with other projects that target the same board.

To create a new board file for the target board, follow these steps:

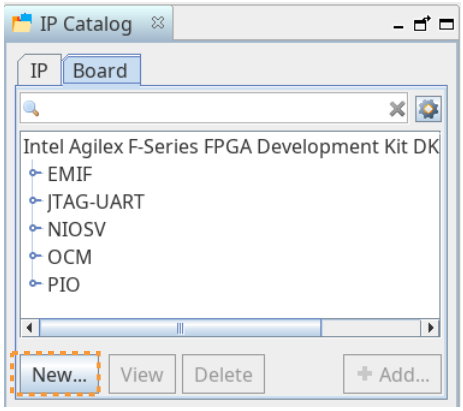
1. Click **Tools ➤ Platform Designer**. The **Open System** dialog box displays the project's name, **Revision**, **Board**, **Device family**, and **Device part**. Click **Retrieve Values** if this field requires re-synchronization.
2. For the **Platform Designer system** setting, click the **Create new platform designer system** button, specify the `pio_led.qsys` **File Name**, and click **Create**.

Figure 12. Open System Dialog Box



3. In IP Catalog, click the **Board** tab. The Board Catalog lists any existing board files. By default the preset path is: `<quartus>\ip\altera\board_preset_files`.

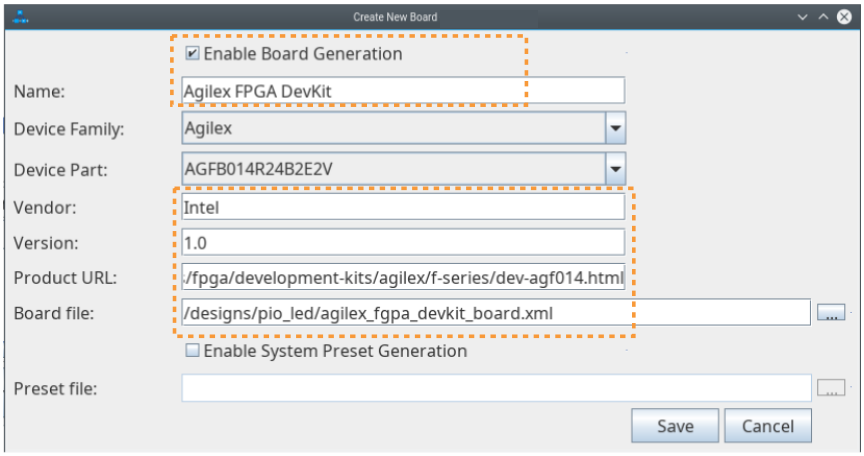
Figure 13. Board Catalog



4. Click **New**. The **Create New Board** dialog box appears.
5. Specify the following settings for the new board definition:

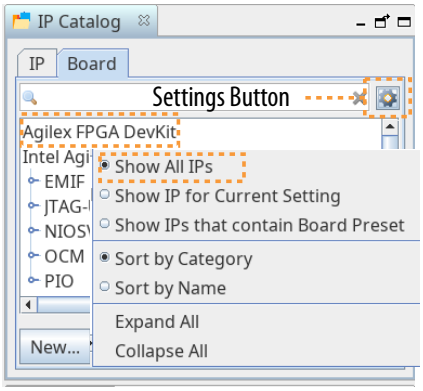
- **Enable Board Generation**—retain enabled setting.
- **Name**—enter `Agilex FPGA DevKit`.
- **Device Family**—retain the **Agilex** setting.
- **Device Part**—retain the **AGFB014R24B2E2V** setting.
- **Vendor**—enter `Intel` as the board vendor.
- **Version**—enter `1.0` as the board file version.
- **Product URL**—Add Agilex F-Series FPGA Development Kit page URL.
- **Enable System Preset Generation**—leave disabled.
- **Board file**—click browse (**...**) and enter the name of the new board file as `agilex_fpga_devkit`. Click **OK**. The `_board.xml` extension automatically appends to the file name that you specify.

Figure 14. Create New Board Dialog Box



6. In the **Create New Board** dialog box, click **Save**. The `agilex_fpga_devkit_board.xml` file saves to the project directory.
7. To view the new board in the Board Catalog, click the Settings button on the right side of the search field, and select **Show All IPs**. Because there are no IPs associated with the Agilex FPGA DevKit board file, the Board Catalog displays only the board name. This Agilex FPGA DevKit board definition is now available for any projects targeting the same board.

Figure 15. Agilex FPGA DevKit Board Visible in Board Catalog

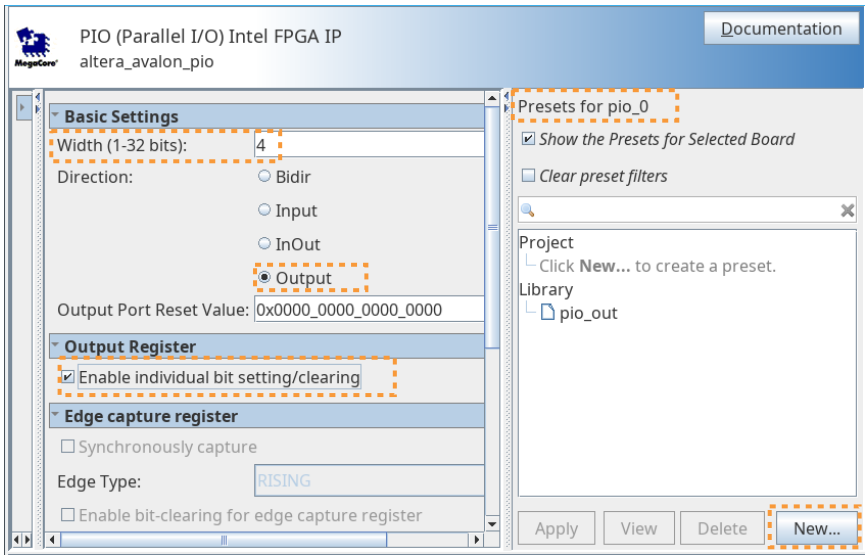


1.4.3. Step 3: Create IP Presets for the Board

To create preset parameter settings that are appropriate for the target board, follow these steps:

1. In Platform Designer IP Catalog, click the **IP** tab and type `pio` to search for the PIO (Parallel I/O) Intel FPGA IP.
2. Double-click the **PIO (Parallel I/O) Intel FPGA IP** name in IP Catalog. The IP parameter editor appears.
3. Specify the following parameter values in the PIO (Parallel I/O) Intel FPGA IP parameter editor:
   - **Width (1-32 bits)**—enter 4.
   - **Direction**—select **Output**.
   - **Output Register**—turn on **Enable individual bit setting/clearing**.

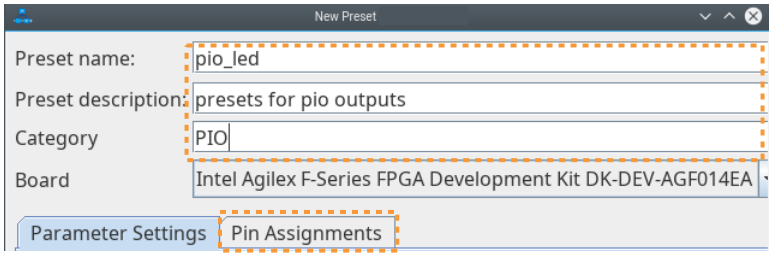
Figure 16. PIO (Parallel I/O) Intel FPGA IP Parameter Editor



4. In the **Presets** pane, click the **New** button. The **New Preset** dialog box appears (alternatively, click **View ➤ Presets**).
5. Specify the following options to identify the new preset:
   - **Preset name**—enter `pio_led`.
   - **Preset description**—enter `presets for pio outputs`.
   - **Category**—enter `PIO`.

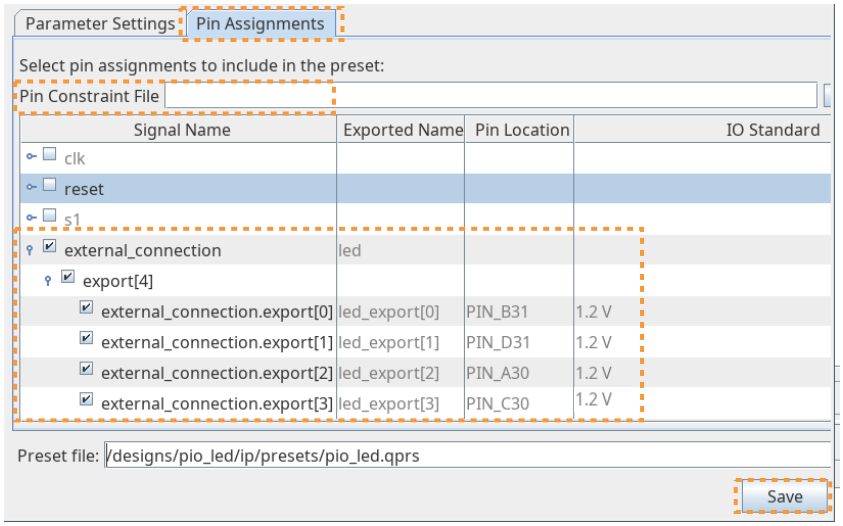
     Note: The **Board** option displays the target board from the project settings.

Figure 17. New Preset Dialog Box



6. View the **Parameter Settings** tab. The parameter values already reflect the parameter values that you set in 3 on page 15.
7. To specify pin location and I/O standard assignments for the preset, click the **Pin Assignments** tab.
8. Enable the **external_connection** interface checkbox and type `led` in the **Exported Name** cell. The `led_export[n]` prefix automatically replaces all port names of the interface.
9. For the `led_export[n]` ports, specify the following **Pin Locations** and **I/O Standard**.
   - led_export[0]—Pin location **PIN_B31**, IO standard **1.2V**
   - led_export[1]—Pin location **PIN_D31**, IO standard **1.2V**
   - led_export[2]—Pin location **PIN_A30**, IO standard **1.2V**
   - led_export[3]—Pin location **PIN_C30**, IO standard **1.2V**

Figure 18. Pin Assignments Tab of Presets Pane



Note: Alternatively, you can populate the pin assignments by specifying the provided `pio_pin.tcl` file for the **Pin Constraint File** option.

10. Click the **Save** button to save the IP preset file.
11. Click **Finish** to generate the PIO IP and add to the system.

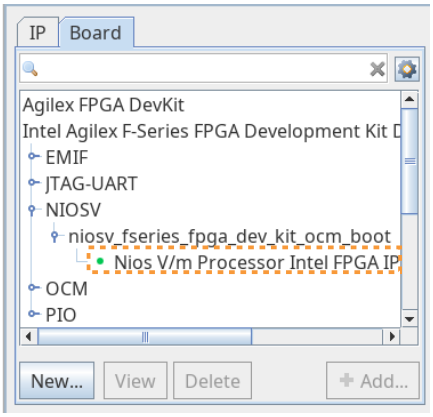
    Note: You can now reuse this preset file for projects targeting this board and IP.

1.4.4. Step 4: Create a Design Using Board and Preset Files

Once you create board and preset files, you can use these files to create a design that is appropriately configured for the target board. The following steps describe adding a new instance of the PIO (Parallel I/O) Intel FPGA that derives parameters from the pio_led preset and targets the same Intel Agilex F-Series Development Kit board.

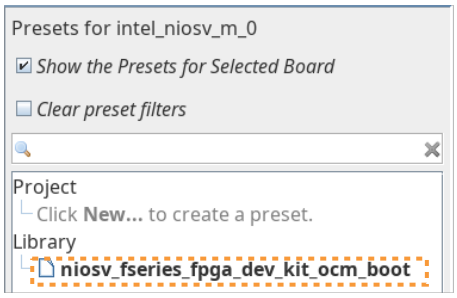
1. View the Clock Bridge Intel FPGA IP, Reset Bridge Intel FPGA IP, and PIO Intel FPGA IP components in Platform Designer's **System View**.
2. Right-click the PIO Intel FPGA IP (`pio_0`) in the **System View**, and then click **Remove**.
3. In the IP Catalog, click the **Board** tab.
4. Under the **Intel Agilex F-Series FPGA Development Kit DK-DEV-AGF014EA** board, click **NIOSV** to expand the **niosv_fseries_fpga_dev_kit_ocm_boot** IP preset name. Click the IP preset name to show **Nios V/m Processor Intel FPGA IP**.

Figure 19. IP and Presets in Board Tab



5. Double-click **Nios V/m Processor Intel FPGA IP**. The IP Parameter Editor opens.
6. In the parameter editor **Preset** tab, double-click the preset name to apply to the selected IP (or click **Apply** in the **Preset** tab). The parameter editor displays the **Preset** tab with the active **niosv_fseries_fpga_dev_kit_ocm_boot** preset in bold.

Figure 20. Active Preset in Bold



7. Click **Finish** to add the preset IP to the system.
8. Repeat steps 4 through 6 to add the following IP with presets to the system. For the PIO Intel FPGA IP, the external connection interface exports automatically per the `pio_led` preset.

Table 2. IP Components with Presets to Add to System

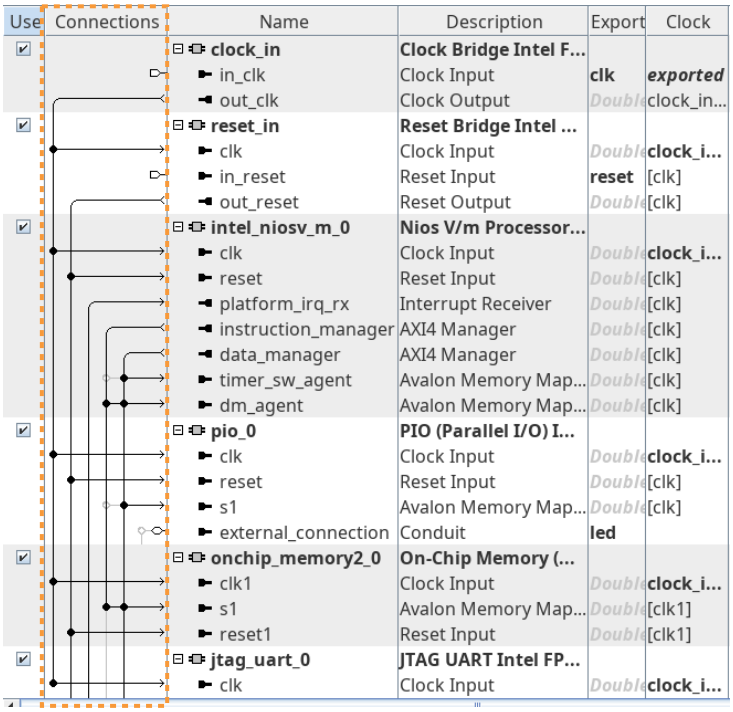
Component Category	IP Name	IP Preset Name
PIO	PIO (Parallel I/O) Intel FPGA IP	pio_led
OCM	On-Chip Memory (RAM or ROM) Intel FPGA IP	ocm_fseries_fpga_dev_kit
JTAG-UART	JTAG UART Intel FPGA IP	juart_fseries_fpga_dev_kit

9. Make connections between the following component ports in the **System View** tab by clicking inside an open connection circle. When you make a connection, Platform Designer changes the connection line to black, and fills the connection circle. Clicking a filled-in circle removes the connection:

Table 3. System Connections

Source Component/Signal	Target Component/Signal
clock_in.out_clk	intel_niosv_m_0.clk pio_0.clk onchip_memory2_0.clk1 jtag_uart_0.clk
reset_in.out_reset	intel_niosv_m_0.reset pio_0.reset onchip_memory2_0.reset1 jtag_uart_0.reset
jtag_uart_0.irq	intel_niosv_m_0.platform_irq_rx
intel_niosv_m_0.instruction_manager	intel_niosv_m_0.dm_agent onchip_memory2_0.s1
intel_niosv_m_0.data_manager	intel_niosv_m_0.timer_sw_agent intel_niosv_m_0.dm_agent pio_0.s1 onchip_memory2_0.s1 jtag_uart_0.avalon_jtag_slave

Figure 21. Completed System Connections



10. To clear the remaining system warnings, assign the missing base addresses by clicking **System ➤ Assign Base Addresses**.

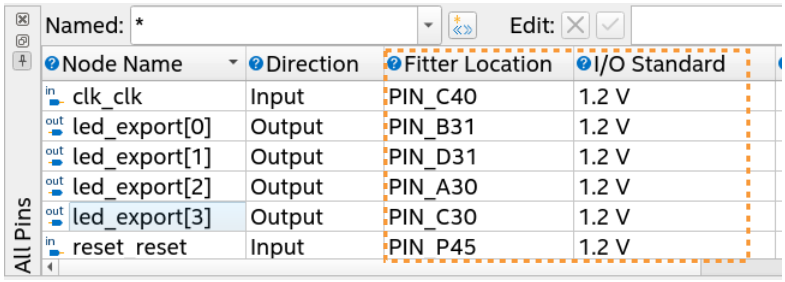
11. Click **File ➤ Save** to save the system.
12. Click the **Generate ➤ Generate HDL** and generate the HDL for the system using default settings in the **Generation** dialog box.
13. When HDL generation completes, close Platform Designer.

1.4.5. Step 5: Compile and Verify the Design

Follow these steps to compile the top-level design that includes the **pio_led** Platform Designer system.

1. After Platform Designer HDL generation is complete, click **Processing ➤ Start Compilation**. The Compiler runs for approximately 15 minutes, depending on your system, and generates the SOF programming file following successful compilation.
2. When full compilation is complete, click **Assignments ➤ Pin Planner** to verify the following appropriate pin assignments are implemented during compilation:

Figure 22. Verifying Location Assignments in Pin Planner



Node Name	Direction	Fitter Location	I/O Standard
clk_clk	Input	PIN_C40	1.2 V
led_export[0]	Output	PIN_B31	1.2 V
led_export[1]	Output	PIN_D31	1.2 V
led_export[2]	Output	PIN_A30	1.2 V
led_export[3]	Output	PIN_C30	1.2 V
reset_reset	Input	PIN_P45	1.2 V

1.4.6. Step 6: Configure the Board with Software

In the next phase of the design, you program software to the board and generate the ELF file, the Board Support Package, and Application Project File. This stage requires the `pio.c` file.

You can then program the board and observe the output from the JTAG terminal. Refer to the Agilex - Nios V PIO LED Toggle Design example design for a complete design containing board support package, application project file, and ELF file to observe output from JTAG terminal. For more information about BSP and Application file generation, refer to the *Nios V Embedded Processor Design Handbook*.

Related Information

Nios V Embedded Processor Design Handbook

1.5. Document Revision History of AN 988: Using the Board-Aware Flow in the Intel Quartus Prime Pro Edition Software

Document Version	Intel Quartus Prime Version	Changes
2023.01.09	22.4	Minor revisions throughout for style and clarity.
2022.12.12	22.4	Initial release of the document.