Peter the Great St. Petersburg Polytechnic University

Institute of Computer Science and Cybersecurity

Graduate School of Computer Technologies and Information Systems

**Lecture: Creating a System Design with Platform Designer**

Subject: Automation of discrete device design (in English)

Completed by student of group 5130901/10101 Nepomnyaschiy M.T.

(signature)

Lecturer Antonov A.P.

(signature)

Saint Petersburg

2024

**Table of contents**

[Creating a System Design with Platform Designer 4](#_Toc161095673)

[Introduction 4](#_Toc161095674)

[Part 6 – Platform Designer User Interface 4](#_Toc161095675)

[6.1. Opening Platform Designer 4](#_Toc161095676)

[6.2. Project Association 5](#_Toc161095677)

[6.3. The Platform Designer GUI (Default view) 5](#_Toc161095678)

[6.4. Views, Layouts, and Tab Management 6](#_Toc161095679)

[6.5. Basic Platform Designer System Building Flow 7](#_Toc161095680)

[6.6. Select and Configurate components from IP Catalog 8](#_Toc161095681)

[IP Catalog 8](#_Toc161095682)

[Parameter Editor/Tab 9](#_Toc161095683)

[IP Component Categories Included in IP Catalog 10](#_Toc161095684)

[System Components Tab 11](#_Toc161095685)

[6.7. Component Interfaces 12](#_Toc161095686)

[6.8. Connections Column Use 13](#_Toc161095687)

[6.9. Other Connection Methods 14](#_Toc161095688)

[6.10. Exporting Interfaces 15](#_Toc161095689)

**List of illustrations**

[Figure 1 – Launching Platform Designer 4](#_Toc161095690)

[Figure 2 – Project Association 5](#_Toc161095691)

[Figure 3 – GUI (Default Layout) 5](#_Toc161095692)

[Figure 4 – Views, Layouts, and Tab Management 6](#_Toc161095693)

[Figure 5 – System building flow 7](#_Toc161095694)

[Figure 6 – IP Catalog components 8](#_Toc161095695)

[Figure 7 – Parameter Editor/Tab 9](#_Toc161095696)

[Figure 8 – IP Component Categories 10](#_Toc161095697)

[Figure 9 – System components tab 11](#_Toc161095698)

[Figure 10 – Component Interfaces 12](#_Toc161095699)

[Figure 11 – Usage of the Connection Column 13](#_Toc161095700)

[Figure 12 – Alternative connection methods 14](#_Toc161095701)

[Figure 13 - Exporting Interfaces 15](#_Toc161095702)

# Creating a System Design with Platform Designer

The lecture provides some information about Platform Designer usage and explains the examples of its usage on practice.

# Introduction

The Platform Designer in Intel Quartus Prime comes in two versions, depending on whether you're using the Lite or Standard edition, or the Pro edition. This distinction significantly impacts the design process. In the Standard edition, a .qsys file captures all system information, requiring regeneration of RTL code for the entire system upon any parameter or connection changes, making it time-consuming for large designs. On the other hand, the Pro edition separates component information into .ip files, enabling selective regeneration and allowing incomplete components during system generation. This flexibility is particularly advantageous for ongoing development or team-based projects. The Pro edition not only maintains ease of use but also offers enhanced control and improved interactivity in system design, making it a substantial productivity boost compared to the Standard edition.

# Part 6 – Platform Designer User Interface

## Opening Platform Designer

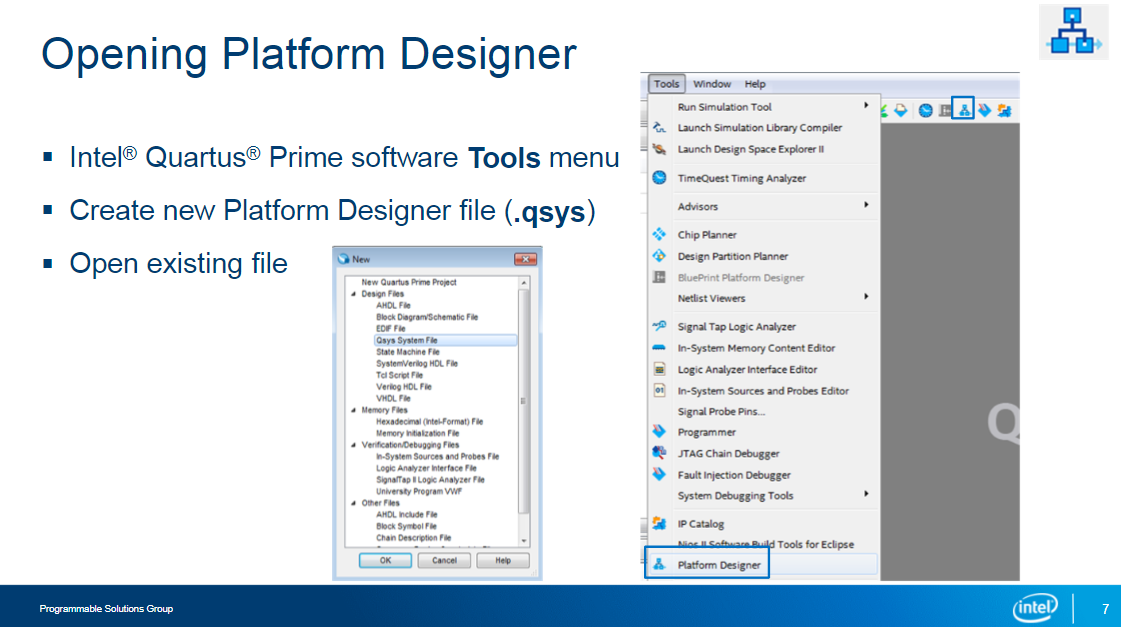


Figure 1 – Launching Platform Designer

To start creating a system design in Platform Designer, launch it from the Intel Quartus Prime software Tools menu. There’s also a toolbar button you can use as a shortcut. You can also launch Platform Designer by selecting New from the File menu. From the New file dialog box, select Qsys System File as the design file type to create. If you have an existing Platform Designer system you want to modify, from the File menu, select Open and choose the .qsys file for your system.

## Project Association

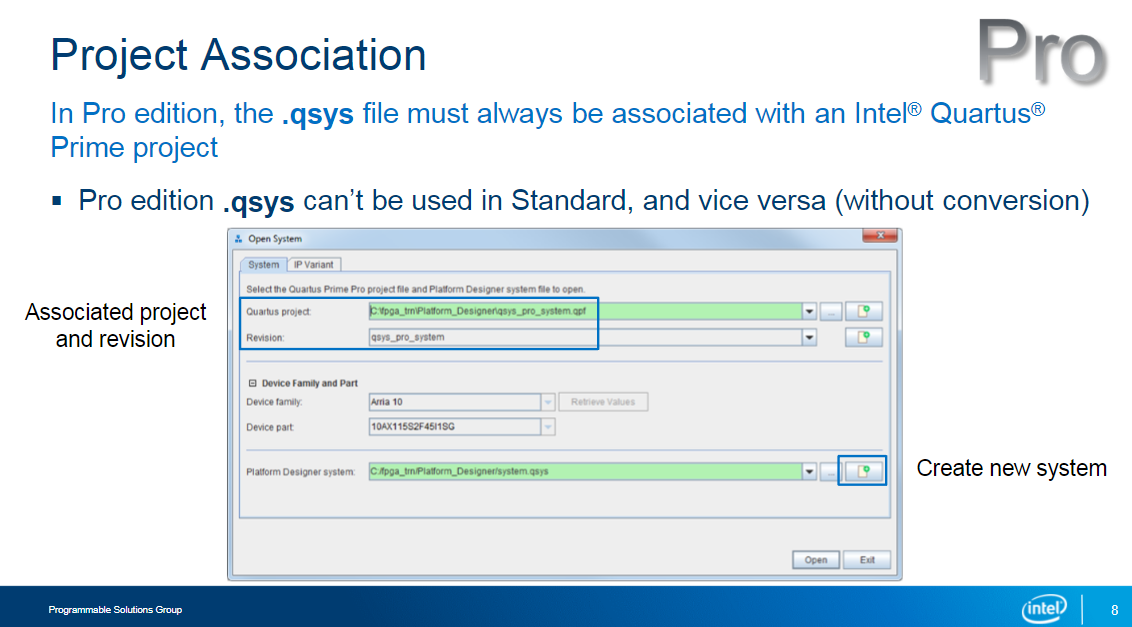


Figure 2 – Project Association

Using the Platform Designer tool is similar in the Standard and Pro editions, but in the Pro edition, your system design needs to be associated with an Intel Quartus Prime project. Unlike the Standard edition, which allows adding a design to any project easily, in the Pro edition, you start Platform Designer within a project or create a new one. Note that a .qsys file made in the Pro edition can't be used in the Standard edition, but the reverse is possible. If you switch from Pro to Standard, make a backup of the .qsys file before opening it in the Standard edition.

## The Platform Designer GUI (Default view)

Изображение выглядит как текст, снимок экрана, линия, число

Автоматически созданное описание

Figure 3 – GUI (Default Layout)

Here's an example of the Platform Designer UI displaying the main view and visible tabs. We'll delve deeper into these aspects later. The UI consists of tabs for constructing your system and analyzing results. You can enable or disable tabs as per your information requirements. While some windows overlap, viewing information in various ways and formats often enhances your comprehension of the system and its connections.

## Views, Layouts, and Tab Management

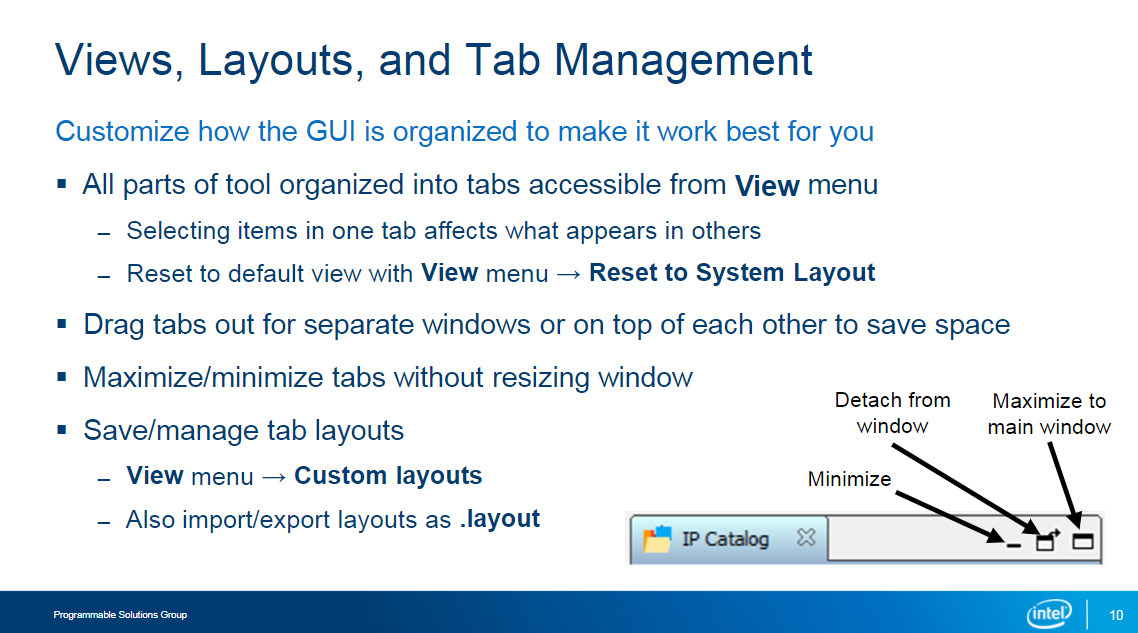


Figure 4 – Views, Layouts, and Tab Management

1. Enabling and Disabling Views and Tabs:

* Access the View menu to enable or disable various views and tabs.
* Click once to toggle, and a second time to restore the previous state.

\*All tabs are interconnected, so actions in one tab affect others.

1. Reset and Reposition Tabs:

* Restore all tabs to default view with "Reset to System Layout" from the View menu.
* Click and drag tabs to reposition, allowing customization of the interface.

1. Tab Interactions and Customization:

* Tabs can be linked or detached for efficient use across multiple monitors.
* Drag tabs to stack or position them in the main interface for optimal display.
* Minimize or maximize tabs using icons in the upper right corner.

1. Preserving and Exporting Layouts:

* Platform Designer maintains the current tab layout even after system closure.
* Save the current layout as a custom layout in the View menu.
* Export the layout as a .layout file for reuse in different system designs.
* Create multiple .layout files for diverse setups to enhance workflow optimization.

## Basic Platform Designer System Building Flow



Figure 5 – System building flow

**System building flow:**

1. Component Selection and Configuration:

* Begin by choosing components from the IP Catalog for your system.
* Utilize each component's parameter editor to configure optional settings graphically.

1. Interface Connection in Platform Designer UI:

* Use the Platform Designer UI to interconnect interfaces of selected components.
* Remember that a component interface adheres to a standard, comprising one or more signals.
* Ensure that only interfaces of the same or compatible types are connected.

1. Validation of Connections:

* Review the established connections to verify their correctness.
* All necessary connections between components must be accurate to implement the intended data flow in your design.

1. HDL Code Generation and System Integration:

* Finally, generate HDL code files representing your system.
* Add the system to an Intel Quartus Prime software project for compilation.

And now we’ll look directly at each one of the steps.

## Select and Configurate components from IP Catalog

### IP Catalog

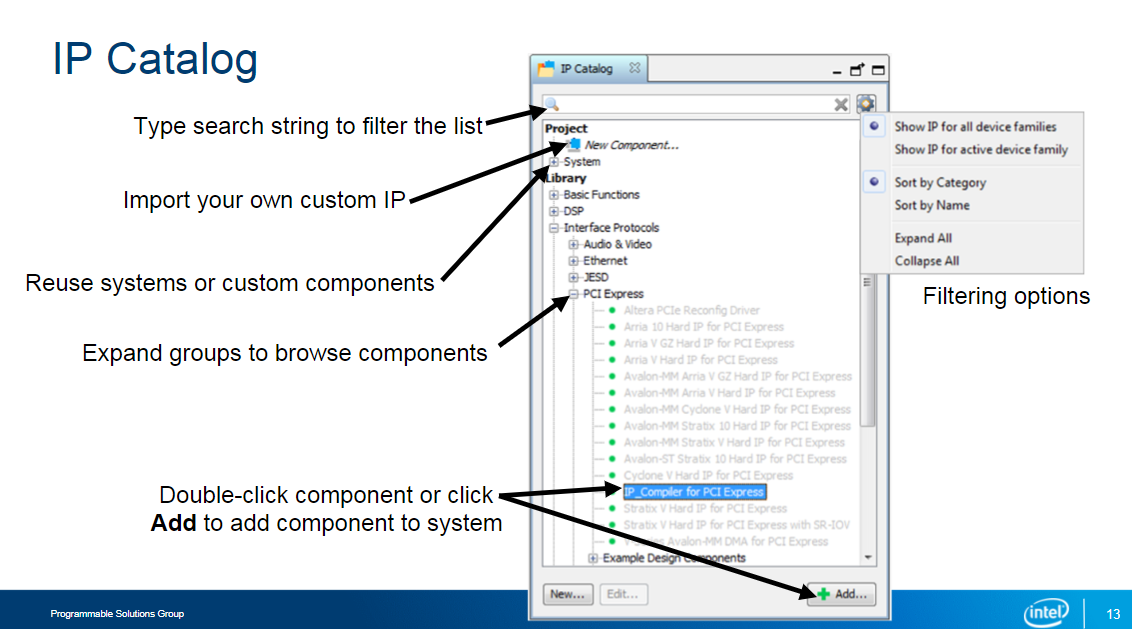


Figure 6 – IP Catalog components

Let’s start from Select and Configurate components from IP Catalog:

1. IP Catalog Overview:

* The IP Catalog houses available IP for integration into your current Platform Designer system.
* Unlike the Intel Quartus Prime software's IP Catalog, manual addition to a project is not required for Platform Designer.

1. Categorization of IP:

* Standard, ready-to-use IP is categorized under the Library section.
* Custom IP and subsystems are found in the Project section.
* Custom IP creation details are covered in a subsequent training session linked at the end.

1. Organization for Accessibility:

* The Library section in the IP Catalog is structured into categories for ease of navigation.
* Users can expand categories, subcategories, and select components for system inclusion.
* The search field facilitates quick component location by name, dynamically filtering the list as characters are typed.

1. Search and Compatibility Features:

* The search field enables users to filter the IP Catalog by name, aiding efficient component identification.
* Users can narrow down the displayed IP to those compatible with their targeted FPGA device.

1. Adding Components to the System:

* To add a component into the system, double-click its name or select it and click Add.
* The chosen component appears in the System Contents tab, and its parameter editor opens for configuration.

### Parameter Editor/Tab

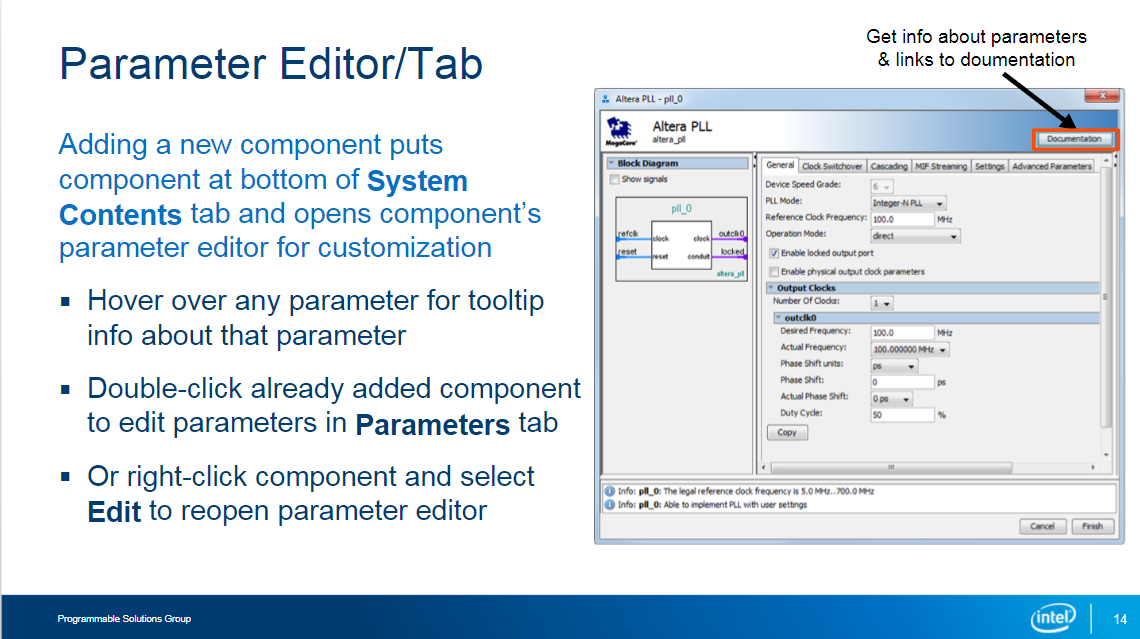


Figure 7 – Parameter Editor/Tab

1. Adding Components:

* Double-click or use "Add" to bring a component into the System Contents tab.
* Opens a window to tweak settings for your system.

1. Individual Setup:

* Configure each component separately in the parameter editor.

1. Parameter Editor Features:

* Hover over settings for info.
* Click "Documentation" for detailed explanations.

1. Compatibility Check:

* Editor prevents incompatible choices.
* Check the messages window for any issues.

1. Finalizing Configuration:

* Click "Finish" to complete setup.
* Added component now visible in the system.

1. Modification Options:

* Double-click a component to change settings.
* Right-click and choose "Edit" for the original editor window.

### IP Component Categories Included in IP Catalog

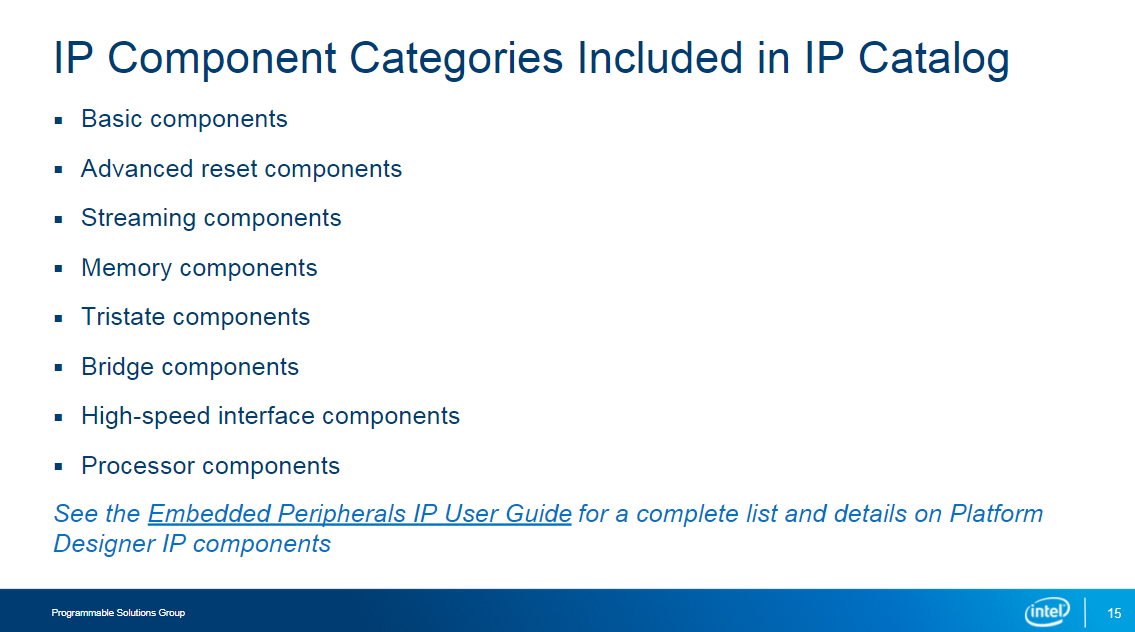


Figure 8 – IP Component Categories

1. Basic Components:

* Manage clock and reset signals with Clock Source and Reset Source components.

1. Advanced Reset Components

* Handle multiple reset signals and gain more control over system reset.

1. Streaming Components:

* Connect and manipulate data flows between Avalon-ST interfaces.

1. Memory Components:

* On-chip RAM, ROM, FIFOs, and controllers for off-chip memory (flash, SRAM, SDRAM).
* On-chip DMA components, including regular and scatter-gather.

1. Tristate Components:

* Connect to off-chip devices using bi-directional data busses.
* Share FPGA pins with common signaling for these devices.

1. Bridge Components:

* Adjust Platform Designer interconnect topology with pipeline stages, asynchronous clock crossing logic, and logic for 32-bit to 64-bit systems.
* Bridge to external interfaces like JTAG and SPI.

1. High-speed Interface Components:

* Implement protocol layers for high-speed interfaces such as Ethernet, PCIe\*, and Interlaken.

1. Processor Components:

* Add processors like Nios II soft processor and Arm Cortex processors to your system.
* Refer to the Embedded Peripherals IP User Guide for a more comprehensive list of available IP components.

### System Components Tab

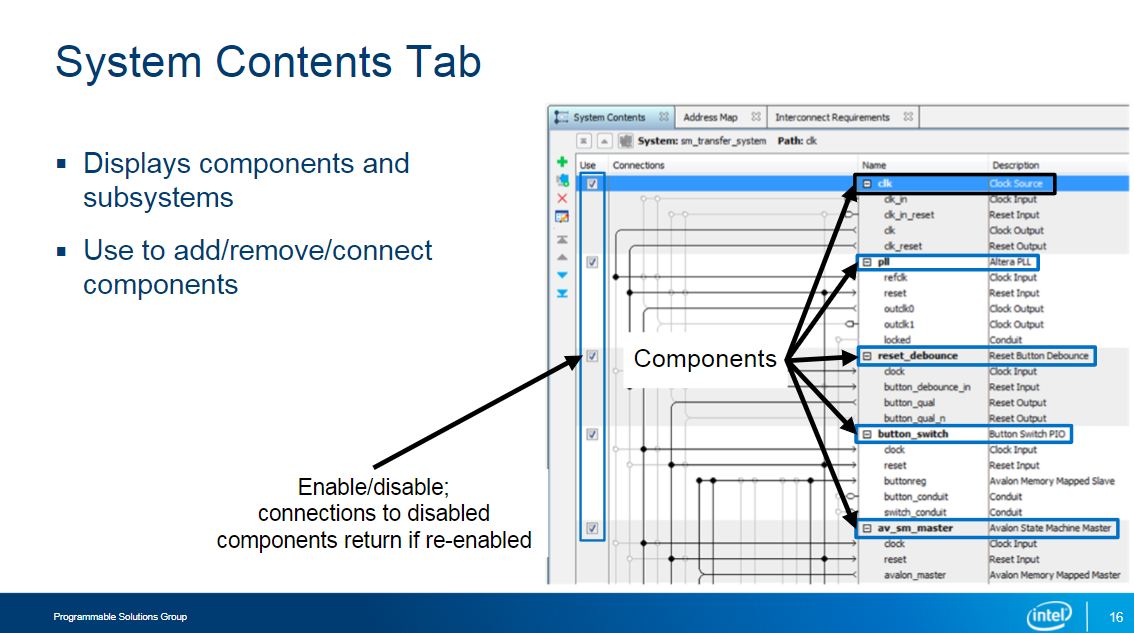


Figure 9 – System components tab

**System Contents Tab Overview:**

Displays all added components and subsystems in the current system.

* Function: Primary interface for system management and configuration.
* Possible actions:
  + Modify a component's relationships within the system.
  + Manage connections to other components and external links.
  + Define a component's location in the memory-mapped master interface.

**Example System:**

* Components like clock source, PLL, custom button debounce, parallel I/O, and custom state machine are shown.
* Use Column Feature:
  + Enables individual component enable/disable without removing them from the system.

**Naming Components:**

* Components get default names upon addition.
* Right-click to rename for clarity and meaning.
* Example names: clk, pll, reset\_debounce, button\_switch, av\_sm\_master.

**Unique Naming:**

* Unique names like clk, pll help differentiate multiple instances of the same component.
* Aids organization within the system.

## Component Interfaces

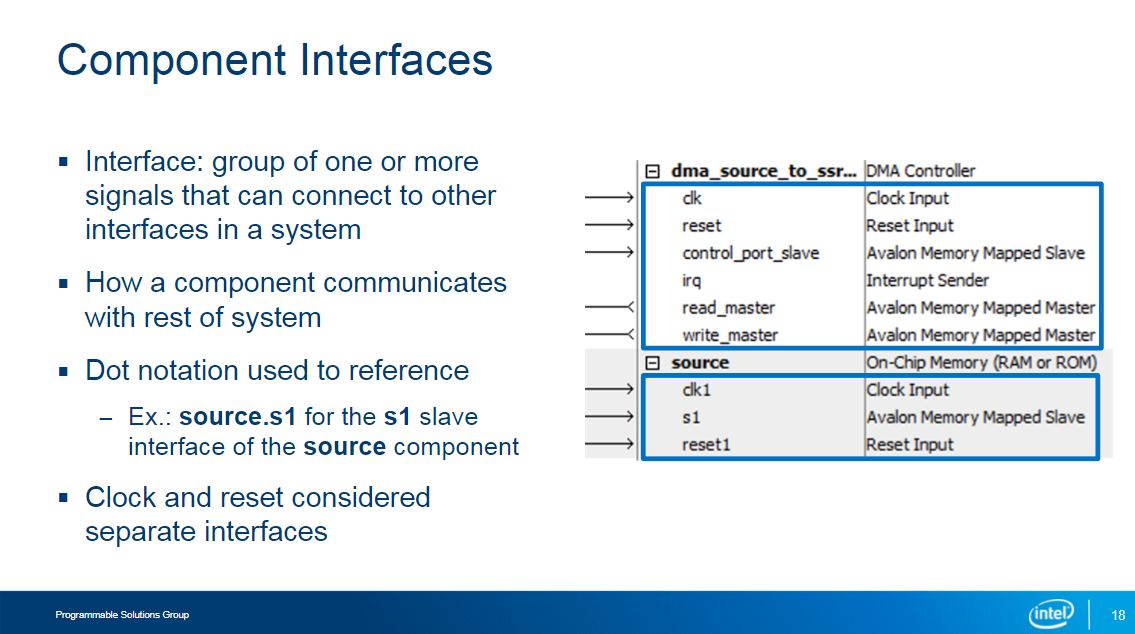


Figure 10 – Component Interfaces

Next one step in the Basic Platform Designer system building flow is Component Interfaces. It allows to connect the component interfaces to each other.

When you add a component to your system, you can see all its communication points, called interfaces, in the System Contents tab. Each interface is like a group of signals that allows the component to talk to other parts of the system. While most interfaces need to be connected, some can be left unconnected, depending on the component. The notation for interfaces uses dot notation, like `source.s1`, where `s1` is an interface of the `source` component.

Unlike component names, you can't change interface names in the Standard edition. However, the Pro edition allows this feature. Clocks and resets are important types of interfaces. Every component has a clock interface that must be connected to a clock source. For resets, they should be linked to a reset source and are synchronous to the component's clock. Keep in mind that Platform Designer doesn't support asynchronous resets. If you leave a reset unconnected, the tool will display a warning, but depending on your design, it might still work.

## Connections Column Use

Изображение выглядит как текст, снимок экрана, диаграмма, линия

Автоматически созданное описание

Figure 11 – Usage of the Connection Column

The Connections column in the System Contents tab is a useful tool for understanding how different interfaces of components are linked in your system. It visually represents connections, showing which interfaces are connected and how they connect to one another.

Possible usage:

1. Types of Connections:

* Quickly identify and make connections like masters to slaves, clock outputs to clock inputs, reset outputs to reset inputs, and more.

1. Graphical Representation:

* A black dot signifies a connection between two interfaces.
* An open gray dot indicates interfaces that are not connected but can relate to a simple click.

1. Automatic Generation of Interconnect:

* Based on parameters in the editor and connections enabled in the column, Platform Designer creates a custom-built interconnect automatically.

1. Additional Details:

* Arrows show the direction of interfaces - away for masters, clock outputs, reset outputs, sources, or senders, and towards for slaves, clock inputs, reset inputs, sinks, or receivers.

1. Highlighting Connections:

* Clicking on an interface highlights the connection lines, with black indicating existing connections and gray showing possible connections.

1. Managing Busyness:

* With many components and connections, the column can get busy.
* Hide connections by collapsing a component's interfaces - click the + sign next to a component to simplify the display.

1. Interface Filtering Options:

* Further manage connections and interfaces through filtering options in the System Contents tab and other accessible tabs from the View menu.

## Other Connection Methods

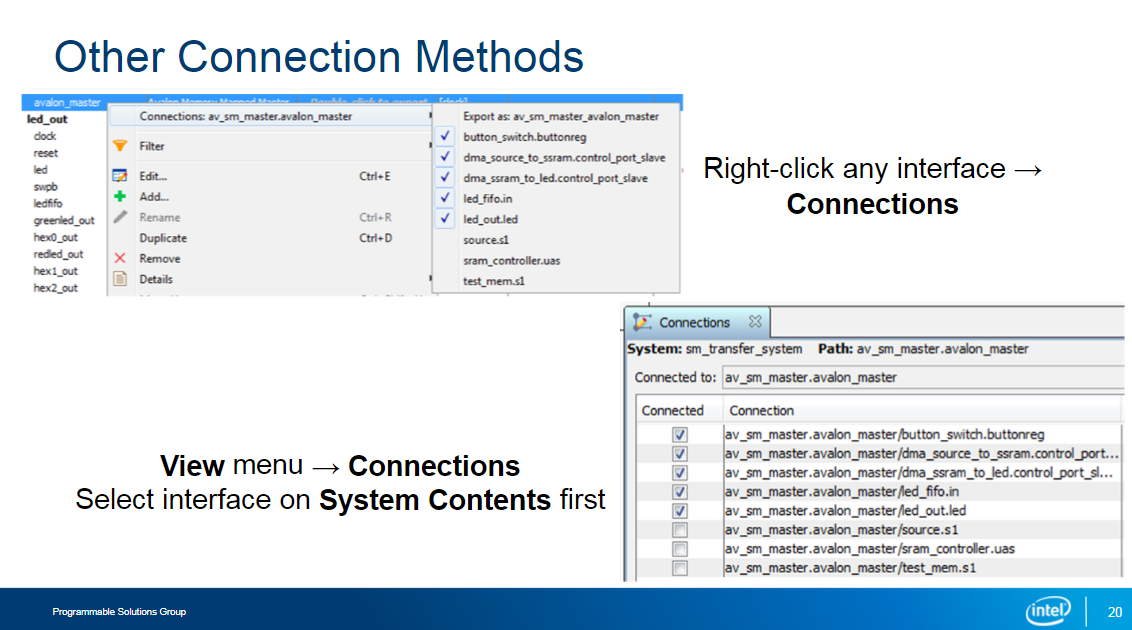


Figure 12 – Alternative connection methods

When you add a new clock, master, or streaming sink interface in the System Contents tab, a new line is created in the Connections column to connect other interfaces to it. However, in complex systems, this column can become confusing, leading to potential incorrect connections.

Due to the complexity, use alternate methods to ensure correct connections:

1. Right-Click Method:

* Right-click any interface and navigate to the Connections sub-menu.
* It lists all valid connections for the selected interface, with checkmarks indicating existing connections.
* Click an interface in the list to establish or break its connection with the selected interface.

1. Connections Tab:

* Accessible from the View menu, the Connections tab offers another method.
* Select an interface on the System Contents tab, and the Connections tab displays all potential connections for that interface.
* A slash (/) indicates a possible connection between the selected interface and other compatible interfaces in the system.
* Use the Connected checkboxes to enable or disable connections.

These alternative methods help you make and verify connections more accurately, especially in intricate system setups.

## Exporting Interfaces

Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

Автоматически созданное описание

Figure 13 - Exporting Interfaces

1. What's Exporting?

* Exporting means taking a component's communication lines (signals) to the top level of Platform Designer.
* You can connect these signals outside, either to other parts of the Intel Quartus Prime project or directly to the device's I/O pins.

1. How to Export:

* To export, go to the System Contents tab and double-click in the Export column for the interface you want to export.
* A default name shows up, but you can change it if you'd like.
* This name is essential for setting up the system design or making connections to the device's pins.

1. Indicator of Export:

* After exporting, the visual in the Connections column changes from arrows to a pin icon.
* This icon shows that the signals from this interface are either coming into or going out of the system.

1. Undoing Export:

* If you want to bring the exported signals back inside the system, just delete the name in the Export column.

1. Rules about Usage:

* Remember, an interface can't be both inside and outside the system simultaneously.
* But, you can get around this by using a bridge component, which duplicates the interface for both internal and external use.

1. Where to Learn More:

* For extra details, check out the Embedded Peripherals IP User Guide and other resources listed at the end of this training.