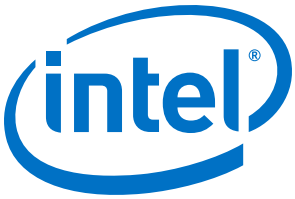


Interlaken IP Core (2nd Generation) Design Example User Guide

UG-20051
2016.10.31

Contents

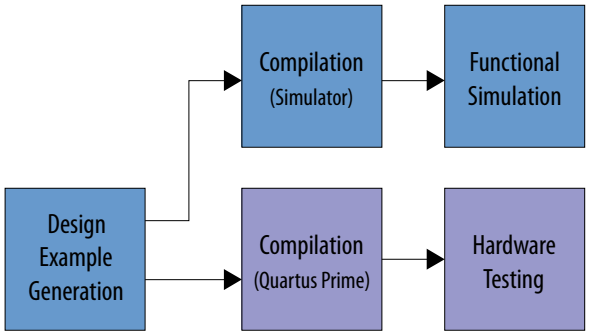
1 Quick Start Guide

The Interlaken IP core (2nd Generation) feature a simulating testbench and a hardware example design that supports compilation and hardware testing, to help you understand usage.

When you generate the example design, the parameter editor automatically creates the files necessary to simulate, compile, and test the design in hardware. You can download the compiled hardware design and run it on the Stratix® 10 Transceiver Signal Integrity Development Kit. The testbench and example design support numerous variants (parameter combinations) of the Interlaken IP core . However, they do not cover all possible parameterizations of the Interlaken IP core (2nd Generation).

In addition, for most IP core variations, Intel® provides a compilation-only example project that you can use to quickly estimate IP core area and timing.

Figure 1. Development Steps



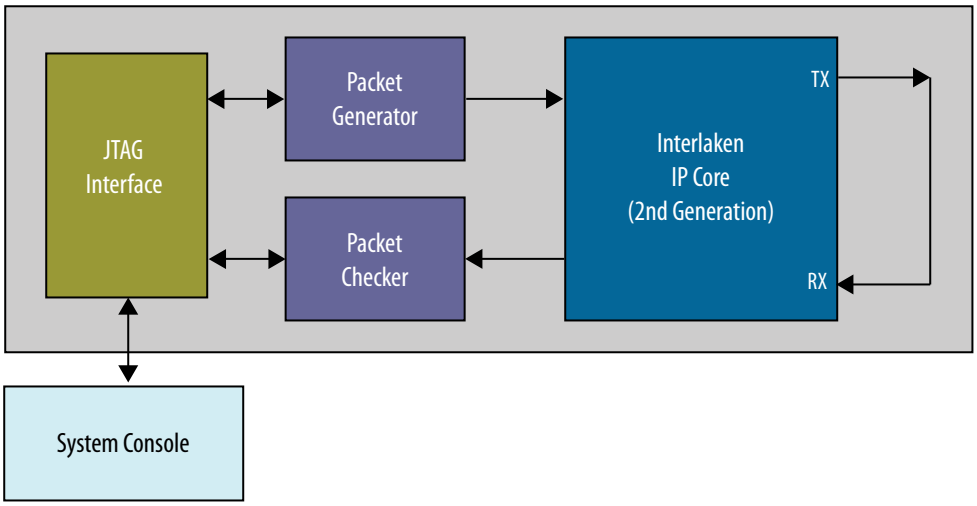
1.1 Directory Structure

Figure 2. Directory Structure of the Generated Example Design

The hardware configuration, simulation, and test files are located in ***<example_design_install_dir>*/uflex_ilk_0_example_design**.

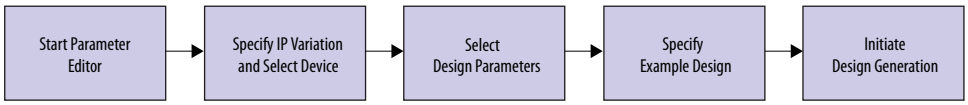
1.2 Design Components

Figure 3. Design Example Block Diagram



1.3 Generating the Design

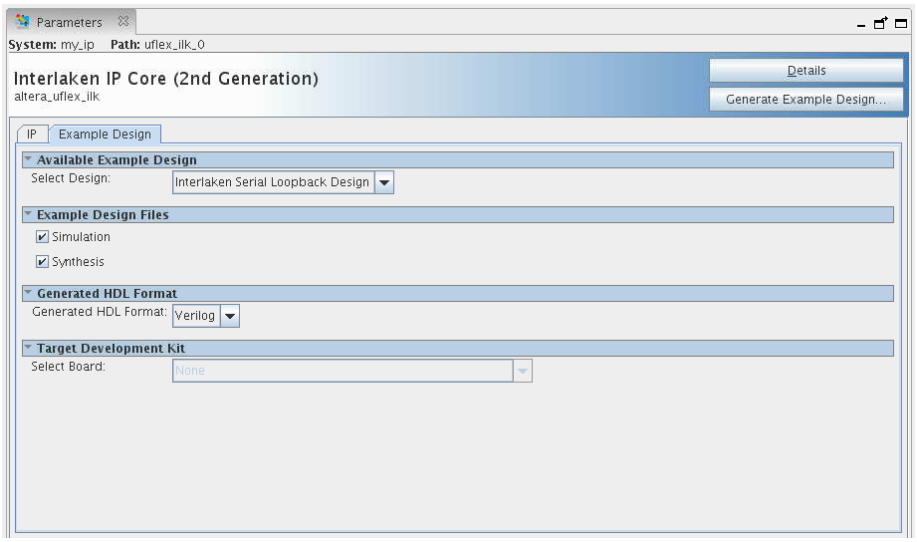
Figure 4. Procedure



Follow these steps to generate the Stratix 10 hardware example design and testbench:

1. In the IP Catalog (**Tools > IP Catalog**), select the Stratix 10 target device family.
2. In the IP Catalog, locate and double-click **Interlaken IP core (2nd Generation)**. The **New IP Variation** window appears.
3. Specify a top-level name for your custom IP variation. The parameter editor saves the IP variation settings in a file named ***<your_ip>*.ip**.
4. You must select a specific Stratix 10 device in the **Device** field, or keep the default Quartus® Prime software device selection.
5. Click **OK**. The parameter editor appears.

Figure 5. Parameter Editor

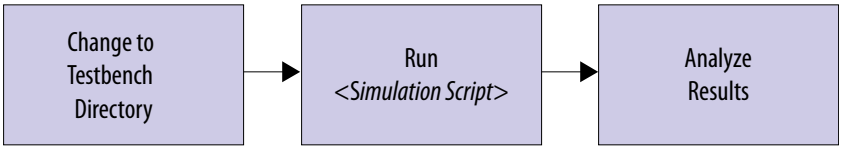


6. On the **IP** tab, specify the parameters for your IP core variation.
7. On the **Example Design** tab, select the **Simulation** option to generate the testbench, and select the **Synthesis** option to generate the hardware example design.

   Note: At least one of the **Simulation** and **Synthesis** check boxes from **Example Design Files** must be selected to allow generation of Example Design Files.
8. For **Generated HDL Format**, only **Verilog** is available.
9. For **Target Development Kit** select the **Stratix 10 Transceiver Signal Integrity Development Kit**. If you select a development kit, then the target device (selected in **step 4**) for Example Design is changed to match the device on target board.
10. Click the **Generate Example Design** button.

1.4 Simulating the Design

Figure 6. Procedure



Follow these steps to simulate the testbench:

1. Change to the simulation directory ***<example_design_install_dir>/uflex_ilk_0_example_design/ilk_uflex***.
2. Run the simulation script for the supported simulator of your choice. The script compiles and runs the testbench in the simulator. Your script should check that the SOP and EOP counts match after simulation is complete.
3. Analyze the results.

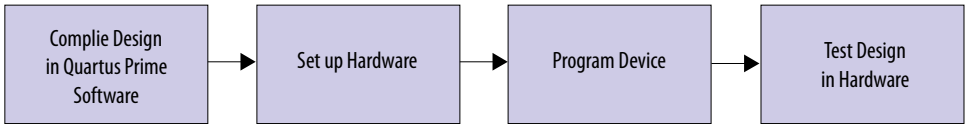
Table 1. Steps to Run Simulation

Simulator	Working Directory	Instructions
Modelsim	*<example_design_install_dir>*/uflex_ilk_0_example_design/ilk_uflex/sim/mentor	Type the following commands: `-do vlog_pro.do` A successful simulation ends with the following message, "Simulation stopped due to successful completion!"

VCS	*<example_design_install_dir>*/uflex_ilk_0_example_design/ilk_uflex/sim/synopsys/vcs	Type the following commands:
		`chmod +x *.sh`
		`./vcstest.sh`
		A successful simulation ends with the following message, "Simulation stopped due to successful completion!"

1.5 Compiling and Testing the Design

Figure 7. Procedure



To compile and run a demonstration test on the hardware example design, follow these steps:

1. Ensure hardware example design generation is complete.
2. Open the Quartus Prime project ***<user-specified location>/example_design_s10/quartus/example_design.qpf***, where *<user-specified location>* is the directory location you specified when you generated the testbench and hardware example design.
3. On the **Processing** menu, click **Start Compilation**.
4. After successful compilation, a **.sof** file will be generated in your specified directory. Follow these steps to program the hardware example design on the Stratix 10:
   a. On the **Tools** menu, click **Programmer**.
   b. In the **Programmer**, click **Hardware Setup**.
   c. Select a programming device.
   d. Select and add the **Stratix 10 Transceiver Signal Integrity Development Kit** to which your Quartus Prime session can connect.
   e. Ensure that **Mode** is set to **JTAG**.
   f. Select the Stratix 10 device and click **Add Device**. The Programmer displays a block diagram of the connections between the devices on your board.
   g. In the row with your **.sof**, check the box for the **.sof**.
   h. Check the box in the **Program/Configure** column.
   i. Click **Start**.

5. After the hardware example design is configured on the Stratix 10 device, in the Quartus Prime software, on the Tools menu, click **System Debugging Tools** > **System Console**.
6. In the Tcl Console pane, type `sysconsole_testbench.tcl`.
7. Type `run_example_design`.

2 Design Example Description

The example design demonstrate the functionalities of the Interlaken IP core. You can generate the design from the **Example Design** tab of the Interlaken IP core graphical user interface (GUI) in the IP parameter editor.

2.1 Features

- Internal TX to RX serial loopback mode.
- Automatically generates fixed size packets.
- Basic packet checking capabilities.
- You can use system console to reset the design for re-testing purpose.

2.2 Hardware and Software Requirements

To test the example design, use the following hardware and software:

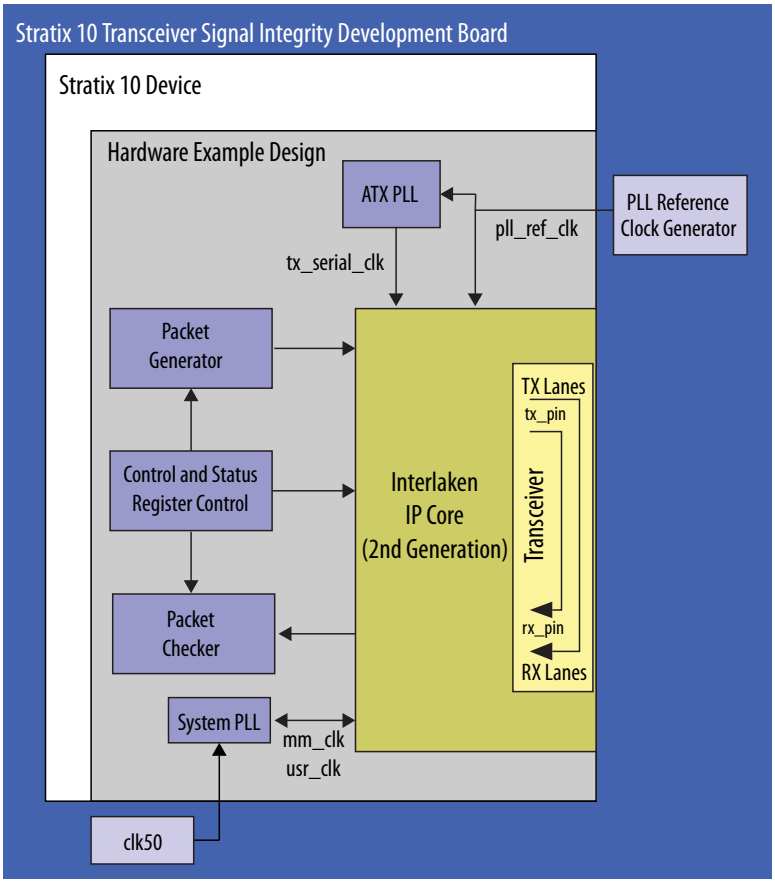
- Quartus Prime Pro Edition software
- System Console
- Modelsim-SE or VCS simulator
- Stratix 10 Transceiver Signal Integrity Development Kit for hardware testing

2.3 Functional Description

The hardware example design connects system and PLL reference clocks and required design components. After you program the device on the Stratix 10 transceiver signal integrity development board, the example design configures the IP core in internal loopback mode and generates packets on the IP core TX user data transfer interface. The IP core sends these packets on the internal loopback path through the transceiver.

After the IP core receiver receives the packets on the loopback path, it processes the Interlaken packets and transmits them on the RX user data transfer interface. The example design checks that the packets it receives on the IP core RX user data transfer interface are consistent with the packets sent in.

Figure 8. Interlaken IP Core Design Example Block Diagram



The hardware example design includes external PLLs. You can examine the clear text files to view sample code that implements one possible method to connect external PLLs to the Interlaken IP core.

The hardware example design packs six Interlaken lanes in a transceiver block, and connects all of the channels in the same transceiver block to a single ATX PLL. IP core connects the ATX PLL to the `tx_pll_locked` and `tx_pll_powerdown` ports. This simple connection model is only one of many options available to you for configuring and connecting the external PLLs in your Interlaken design.

2.4 Design Example Behavior

Immediately following configuration on the Stratix 10 device, when you type `run_example_design` in system console, the Interlaken IP core hardware example design performs the following actions:

1. Resets the Interlaken IP core.
2. Configures the Interlaken IP core in internal loopback mode.
3. Sends a sequence of 100 256-byte Interlaken packets with predefined data in the payload to the TX user data transfer interface of the IP core.

Note: For single segment Interleaved mode, the hardware example design sends 128-byte bursts.

4. Checks the received packets and reports the status.

The packet checker included in the hardware example design provides the following basic packet checking capabilities:

- Checks that the transmitted packet sequence is not violated.
- Checks that the received data matches expected values.

2.5 Interface Signals

Table 2. Design Example Interface Signals

Port Name	Direction	Width (Bits)	Description
clk50	Input	1	System clock input. Clock frequency must be 50 MHz.
pll_ref_clk	Input	1	Transceiver reference clock. Drives the RX CDR PLL.
rx_pin	Input	Number of lanes	Receiver SERDES data pin.
tx_pin	Output	Number of lanes	Transmit SERDES data pin.
usr_pb_reset_n	Input	1	System reset.

2.6 Register Map

Table 3. Design Example Register Map

Offset	Name	Access	Description
8'h00	Reserved		
8'h01	Reserved		
8'h02	System PLL reset	RO	Following bits indicates system PLL reset request and enable value: • Bit [0] - sys_pll_rst_req • Bit [1] - sys_pll_rst_en
8'h03	RX lane aligned	RO	Indicates the RX lane alignment.
8'h04	WORD locked	RO	[NUM_LANES–1:0] – Word (block) boundaries identification.
8'h05	Sync locked	RO	[NUM_LANES–1:0] – Metaframe synchronization.
8'h06 - 8'h09	CRC32 error count	RO	Indicates the CRC32 error count.
8'h0A	CRC24 error count	RO	Indicates the CRC24 error count.
8'h0B	Overflow/Underflow signal	RO	Following bits indicate: • Bit [3] - TX underflow signal • Bit [2] - TX overflow signal • Bit [1] - RX overflow signal
8'h0C	SOP count	RO	Indicates the number of SOP.
8'h0D	EOP count	RO	Indicates the number of EOP
8'h0E	Error count	RO	Indicates the number of following errors:

continued...

Offset	Name	Access	Description
			• Loss of lane alignment • Illegal control word • Illegal framing pattern • Missing SOP or EOP indicator
8'h0F	send_data_mm_clk	RW	Write 1 to enable the generator signal.
8'h10	Reserved		
8'h11	System PLL lock	RO	PLL lock indication.

Notes:

- Design Example register address starts with 0x20** while the Interlaken IP core register address starts with 0x10**.
- Access code: RO—Read Only, and RW—Read/Write.
- System console reads the example design registers and reports the test status on the screen.

A Document Revision History

Table 4. Revision History

Date	Changes
2016.10.31	Initial release