F-Tile Interlaken Intel® FPGA IP Design Example User Guide

Updated for Intel® Quartus® Prime Design Suite: **21.3**

IP Version: **3.0.0**

Online Version
Send Feedback

UG-20334

ID: 683069
Version: 2021.10.04

Contents

1. Quick Start Guide

The F-Tile Interlaken Intel® FPGA IP core provides a simulation testbench. A hardware design example that supports compilation and hardware testing will be available in the Intel Quartus® Prime Pro Edition software version 21.4. When you generate the design example, the parameter editor automatically creates the files necessary to simulate, compile, and test the design.

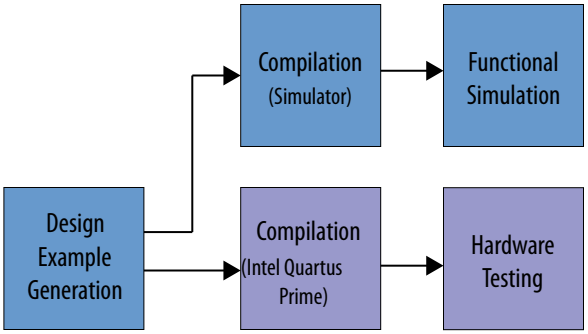
The testbench and design example supports NRZ and PAM4 mode for F-tile devices. The F-Tile Interlaken Intel FPGA IP core generates design examples for the following supported combinations of number of lanes and data rates.

Table 1. IP Supported Combinations of Number of Lanes and Data Rates

The following combinations are supported in the Intel Quartus Prime Pro Edition software version 21.3. All other combinations will be supported in a future version of the Intel Quartus Prime Pro Edition.

Number of Lanes	Lane Rate (Gbps)				
	6.25	10.3125	12.5	25.78125	53.125
4	Yes	-	Yes	Yes	-
6	-	-	-	Yes	Yes
8	-	-	Yes	Yes	-
10	-	-	Yes	Yes	-
12	-	Yes	Yes	Yes	-

Figure 1. Development Steps for the Design Example

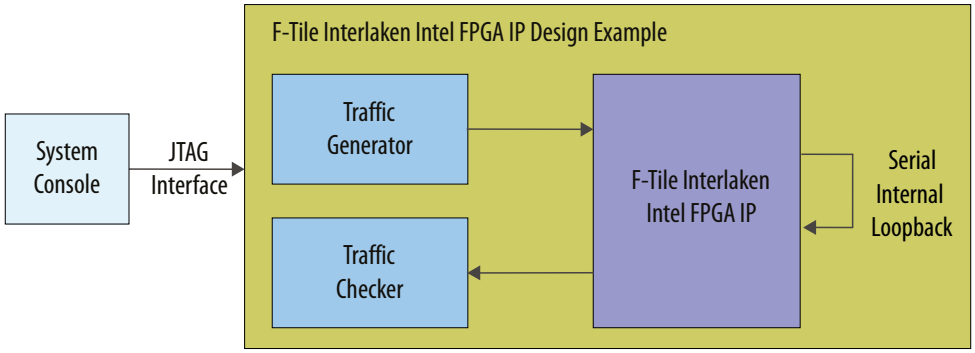


Note: Hardware Compilation and Testing will be available in the Intel Quartus Prime Pro Edition software version 21.4.

The F-Tile Interlaken Intel FPGA IP core design example supports the following features:

- Internal TX to RX serial loopback mode
- Automatically generates fixed size packets
- Basic packet checking capabilities
- Ability to use System Console to reset the design for re-testing purpose

Figure 2. High-level Block Diagram



Related Information

- F-Tile Interlaken Intel FPGA IP User Guide
- F-Tile Interlaken Intel FPGA IP Release Notes

1.1. Hardware and Software Requirements

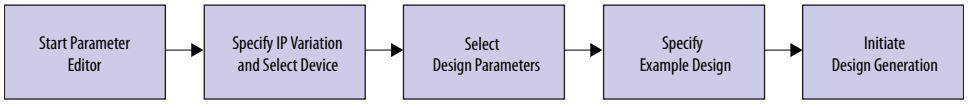
To test the example design, use the following hardware and software:

- Intel Quartus Prime Pro Edition software version 21.3
- System Console
- Supported Simulator:
  - Synopsys* VCS*
  - Synopsys VCS MX
  - Siemens* EDA ModelSim* SE or Questa*

Note: Hardware support for design example will be available in the Intel Quartus Prime Pro Edition software version 21.4.

1.2. Generating the Design

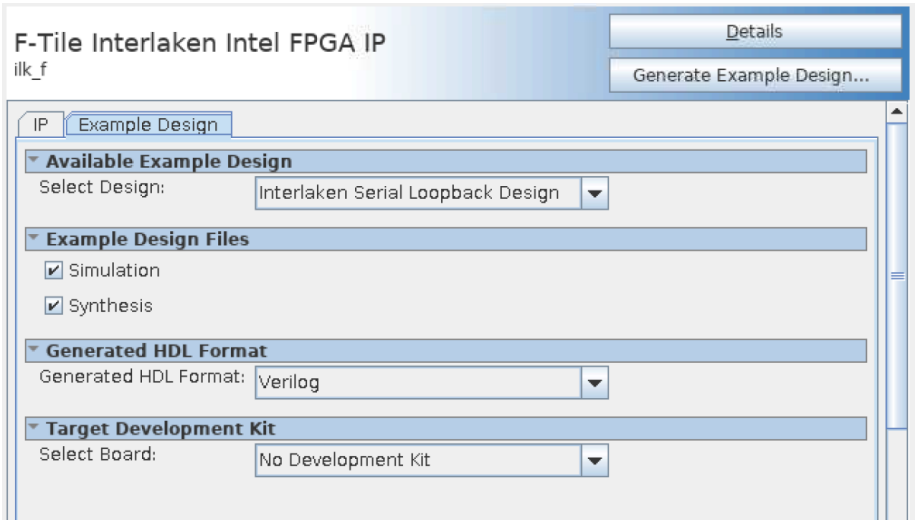
Figure 3. Procedure



Follow these steps to generate the design example and testbench:

1. In the Intel Quartus Prime Pro Edition software, click **File ➤ New Project Wizard** to create a new Intel Quartus Prime project, or click **File ➤ Open Project** to open an existing Intel Quartus Prime project. The wizard prompts you to specify a device.
2. Specify the device family **Agilex** and select device with F-Tile for your design.
3. In the IP Catalog, locate and double-click **F-Tile Interlaken Intel FPGA IP**. The **New IP Variant** window appears.
4. Specify a top-level name <*your_ip*> for your custom IP variation. The parameter editor saves the IP variation settings in a file named <*your_ip*>.ip.
5. Click **OK**. The parameter editor appears.

Figure 4. Example Design Tab



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.

   Note: **Synthesis** option is for hardware example design, which will be available in the Intel Quartus Prime Pro Edition software version 21.4.
8. For **Generated HDL Format**, both **Verilog** and **VHDL** option is available.
9. Click **Generate Example Design**. The **Select Example Design Directory** window appears.
10. If you want to modify the design example directory path or name from the defaults displayed (`ilk_f_0_example_design`), browse to the new path and type the new design example directory name.
11. Click **OK**.

Note: In the F-Tile Interlaken Intel FPGA IP design example, a SystemPLL is instantiated automatically, and connected to F-Tile Interlaken Intel FPGA IP core. The SystemPLL hierarchy path in the design example is:

```
example_design.test_env_inst.test_dut.dut.pll
```

The SystemPLL in the design example shares the same 156.26 MHz reference clock as the Transceiver.

1.3. Directory Structure

The F-Tile Interlaken Intel FPGA IP core generates the following files for the design example:

Figure 5. Directory Structure

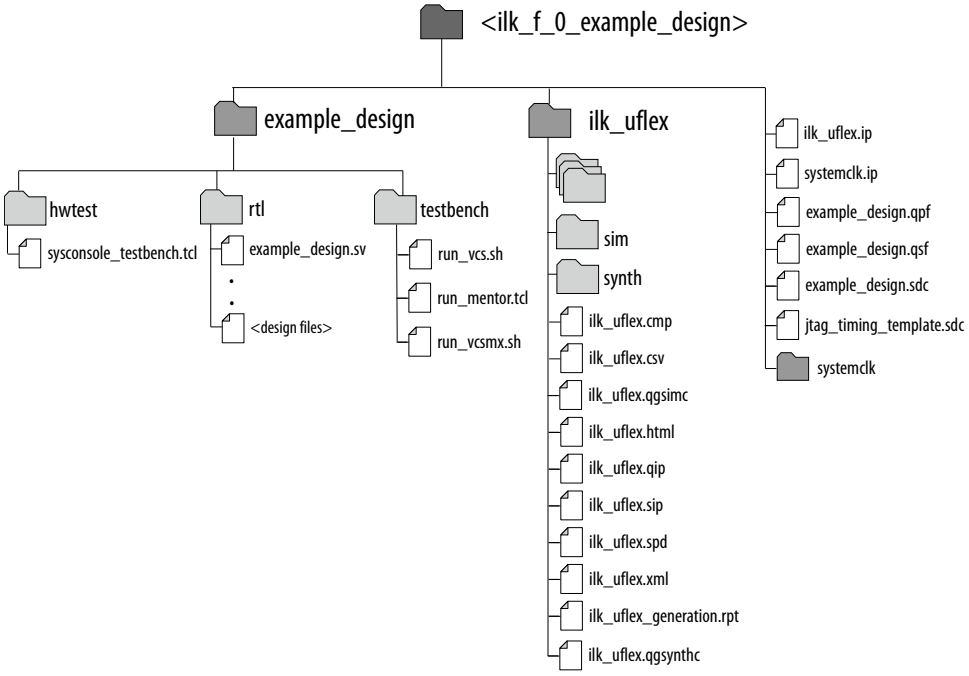


Table 2. Hardware Design Example File Descriptions

These files are in the *<design_example_installation_dir>*/ilk_f_0_example_design directory.

File Names	Description
example_design.qpf	Intel Quartus Prime project file.
example_design.qsf	Intel Quartus Prime project settings file
example_design.sdc jtag_timing_template.sdc	Synopsys Design Constraint file. You can copy and modify for your own design.
sysconsole_testbench.tcl	Main file for accessing System Console

Note: Hardware support for design example will be available in the Intel Quartus Prime Pro Edition software version 21.4.

Table 3. Testbench File Description

This file is in the <*design_example_installation_dir*>/ilk_f_0_example_design/example_design/rtl directory.

File Name	Description
top_tb.sv	Top-level testbench file.

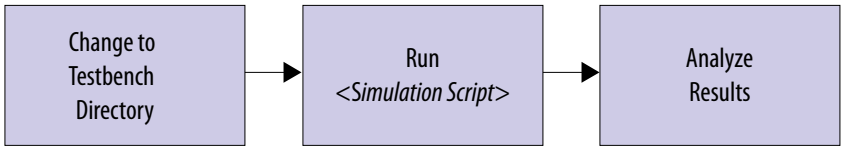
Table 4. Testbench Scripts

These files are in the <*design_example_installation_dir*>/ilk_f_0_example_design/example_design/testbench directory.

File Name	Description
run_vcs.sh	The Synopsys VCS script to run the testbench.
run_vcsmx.sh	The Synopsys VCS MX script to run the testbench.
run_mentor.tcl	The Siemens EDA ModelSim SE or Questa script to run the testbench.

1.4. Simulating the Design Example Testbench

Figure 6. Procedure



Follow these steps to simulate the testbench:

1. At the command prompt, change to the testbench simulation directory. The directory path is <*design_example_installation_dir*>/example_design/testbench.
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.

Table 5. Steps to Run Simulation

Simulator	Instructions
VCS	In the command line, type: `sh run_vcs.sh`
VCS MX	In the command line, type: `sh run_vcsmx.sh`

ModelSim SE or Questa	In the command line, type:
	`vsim -do run_mentor.tcl`
	If you prefer to simulate without bringing up the ModelSim GUI, type:
	`vsim -c -do run_mentor.tcl`

3. Analyze the results. A successful simulation sends and receives packets, and displays "Test PASSED".

The testbench for the design example completes the following tasks:

- Instantiates the F-Tile Interlaken Intel FPGA IP core.
- Prints PHY status.
- Checks metaframe synchronization (SYNC_LOCK) and word (block) boundaries (WORD_LOCK) .
- Waits for individual lanes to be locked and aligned.
- Starts transmitting packets.
- Checks packet statistics:
  - CRC24 errors
  - SOPs
  - EOPs

The following sample output illustrates a successful simulation test run:

```
*********************************************
                INFO: Waiting for lanes to be aligned
                      All of the receiver lanes are aligned and are ready
to receive traffic.
*************************************************

*************************************************
                INFO: Start transmitting packets
*************************************************

*************************************************
                INFO: Stop transmitting packets
*************************************************

*************************************************
                INFO: Checking packets statistics
*************************************************

                CRC 24 errors reported: 0
                SOPs transmitted: 100
                EOPs transmitted: 100
                SOPs received: 100
                EOPs received: 100
                ECC error count: 0

*************************************************
                INFO: Test PASSED
*************************************************
```

Note: The Interlaken design example simulation testbench sends 100 packets and receives 100 packets.

1.5. Compiling the Design Example

1. Ensure the example design generation is complete.
2. In the Intel Quartus Prime Pro Edition software, open the Intel Quartus Prime project <design_example_installation_dir>/example_design.qpf>.
3. On the **Processing** menu, click **Start Compilation**.

2. Design Example Description

The design example demonstrates the functionalities of the Interlaken IP core.

2.1. Design Example Components

The example design connects system and PLL reference clocks and required design components. 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 received and transmitted match.

The F-Tile Interlaken Intel IP design example includes the following components:

1. F-Tile Interlaken Intel FPGA IP core
2. Packet Generator and Packet Checker
3. F-Tile Reference and System PLL Clocks Intel FPGA IP core

2.2. Interface Signals

Table 6. Design Example Interface Signals

Port Name	Direction	Width (Bits)	Description
mgmt_clk	Input	1	System clock input. Clock frequency must be 100 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.
rx_pin_n(1)	Input	Number of lanes	Receiver SERDES data pin.
tx_pin_n(1)	Output	Number of lanes	Transmit SERDES data pin.
mac_clk_pll_ref	Input	1	This signal must be driven by a PLL and must use the same clock source that drives the pll_ref_clk. This signal is only available in PAM4 mode device variations.
usr_pb_reset_n	Input	1	System reset.

(1) Only available in PAM4 variants.

2.3. Register Map

Note:

- Design Example register address starts with 0x20** while the Interlaken IP core register address starts with 0x10**.
- F-tile PHY register address starts with 0x30** while the F-tile FEC register address starts with 0x40**. FEC register is only available in PAM4 mode.
- Access code: RO—Read Only, and RW—Read/Write.
- System console reads the design example registers and reports the test status on the screen.

Table 7. 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: • 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 bit [0] to enable the generator signal.
8'h10	Checker error		Indicates the checker error. (SOP data error, Channel number error, and PLD data error)
8'h11	System PLL lock	RO	Bit [0] indicates PLL lock indication.
8'h14	TX SOP count	RO	Indicates number of SOP generated by the packet generator.
8'h15	TX EOP count	RO	Indicates number of EOP generated by the packet generator.
8'h16	Continuous packet	RW	Write 1 to bit [0] to enable the continuous packet.

continued...

Offset	Name	Access	Description
8'h39	ECC error count	RO	Indicates number of ECC errors.
8'h40	ECC corrected error count	RO	Indicates number of corrected ECC errors.
8'h50	tile_tx_rst_n	WO	Tile reset to SRC for TX.
8'h51	tile_rx_rst_n	WO	Tile reset to SRC for RX.
8'h52	tile_tx_rst_ack_n	RO	Tile reset acknowledge from SRC for TX.
8'h53	tile_rx_rst_ack_n	RO	Tile reset acknowledge from SRC for RX.

2.4. Reset

In the F-Tile Interlaken Intel FPGA IP core, you initiate the reset (`reset_n=0`) and hold until the IP core returns a reset acknowledge (`reset_ack_n=0`). After the reset is removed (`reset_n=1`), the reset acknowledge returns to its initial state (`reset_ack_n=1`). In the design example, a `rst_ack_sticky` register holds the reset acknowledge assertion and then triggers the removal of the reset (`reset_n=1`). You can use alternative methods that fit your design needs.

Important: In any scenario where the internal serial loopback is required, you must release TX and RX of the F-tile separately in a specific order. Refer to the system console script for more information.

Figure 7. Reset Sequence in NRZ Mode

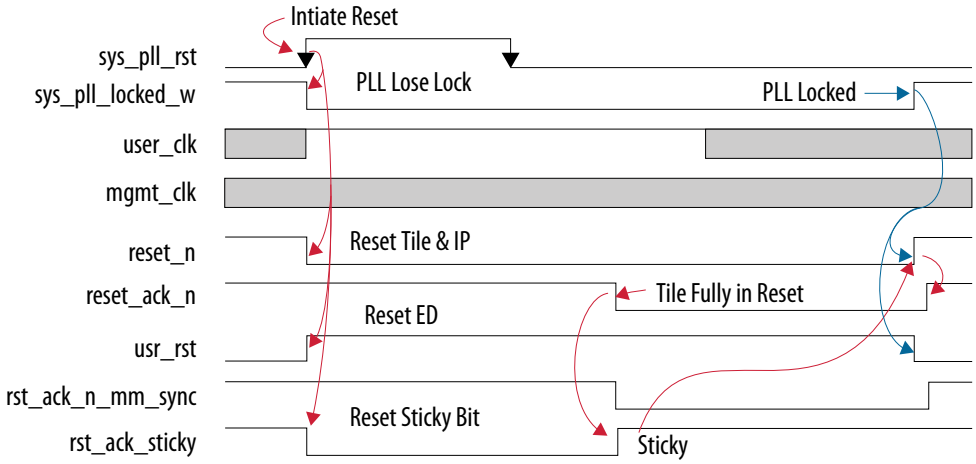
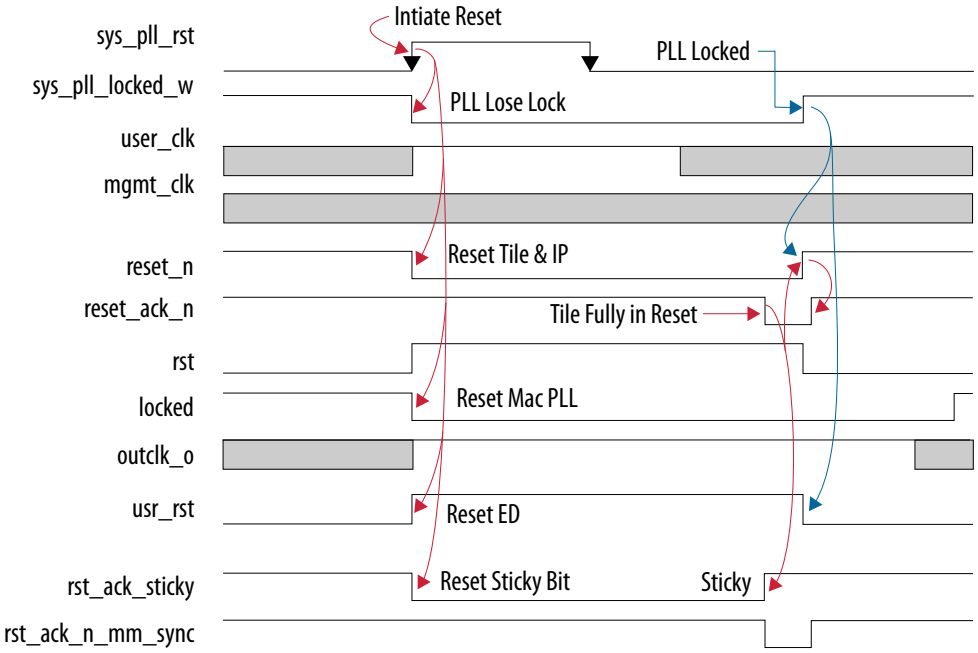


Figure 8. Reset Sequence in PAM4 Mode



3. F-Tile Interlaken Intel FPGA IP Design Example User Guide Archives

If an IP core version is not listed, the user guide for the previous IP core version applies.

Intel Quartus Prime Version	IP Core Version	User Guide
21.2	2.0.0	F-Tile Interlaken Intel FPGA IP Design Example User Guide

4. Document Revision History for F-Tile Interlaken Intel FPGA IP Design Example User Guide

Document Version	Intel Quartus Prime Version	IP Version	Changes
2021.10.04	21.3	3.0.0	• Added support for new lane rate combinations. For more information, refer to *Table: IP Supported Combinations of Number of Lanes and Data Rate*. • Updated the supported simulator list in section: *Hardware and Software Requirements*. • Added new reset registers in section: *Register Map*.
2021.06.21	21.2	2.0.0	Initial release.