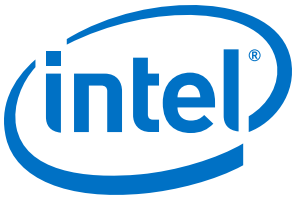


eCPRI Intel® Stratix® 10 FPGA IP Design Example User Guide

Updated for Intel® Quartus® Prime Design Suite: **19.4**

IP Version: **1.0.0**

Subscribe
Send Feedback

UG-20278 | 2020.04.13
Latest document on the web: **PDF** | **HTML**

Contents

1. Quick Start Guide

The eCPRI Intel FPGA IP for Intel® Stratix® 10 devices provides a simulation testbench and a hardware design example that supports compilation and hardware testing. When you generate the design example, the parameter editor automatically creates the files necessary to simulate, compile, and test the design example in hardware.

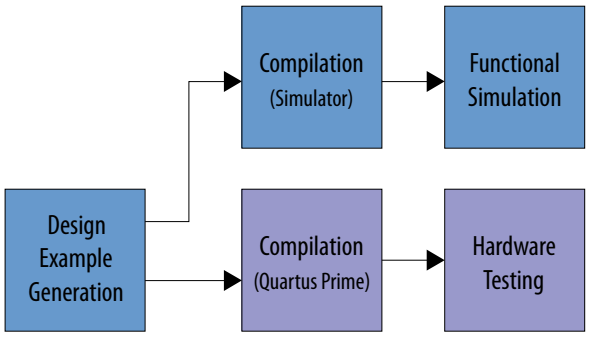
In addition, you can download the compiled hardware design and run it on the Intel Stratix 10 GX Transceiver Signal Integrity Development Kit for the H-tile design examples and Intel Stratix 10 TX Transceiver Signal Integrity Development Kit for the E-tile design examples. Intel provides a compilation-only example project that you can use to quickly estimate IP core area and timing.

The testbench and design example supports Intel Stratix 10 H- or E-tile device variations of the eCPRI IP.

The eCPRI Intel FPGA IP core design example supports the following features:

- Internal TX to RX serial loopback mode
- Traffic generator and checker
- Basic packet checking capabilities
- Ability to use System Console to run the design and reset the design for re-testing purpose

Figure 1. Development Steps for the Design Example



Related Information

eCPRI Intel FPGA IP User Guide

1.1. Hardware and Software Requirements

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

- Intel Quartus® Prime Pro Edition software version 19.4
- System Console
- Modelsim-SE*, VCS*, VCS MX*, NCSim*, Aldec Riviera*, and Xcelium Parallel Simulator*
- Intel Stratix 10 GX Transceiver Signal Integrity Development Kit for the H-tile device variation design example or Intel Stratix 10 TX Transceiver Signal Integrity Development for the E-tile device variation design example for hardware testing

1.2. Generating the Design

Prerequisite: Once you receive the eCPRI web-core IP, save the web-core installer to the local area. Run the installer with Windows/Linux. When prompt, install the web-core to the same location as Intel Quartus Prime folder. The eCPRI Intel FPGA IP now appears in the IP Catalog.

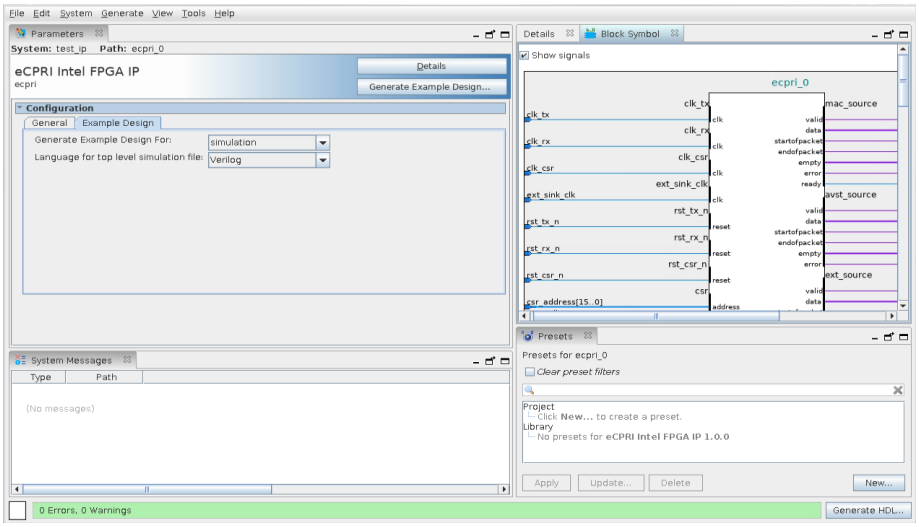
If you do not already have an Intel Quartus Prime Pro Edition project in which to integrate your eCPRI Intel FPGA IP core, you must create one.

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 **Stratix 10 (GX/SX/MX/TX/DX)** and select a device that meets all of these requirements:
   - Transceiver tile is E-tile or H-tile
   - Transceiver speed grade is 2
   - Core speed grade is 2
3. Click **Finish**.
4. In the IP Catalog, locate and double-click **eCPRI Intel FPGA IP**. The **New IP Variant** window appears.

Follow these steps to generate the eCPRI IP hardware design example and testbench:

1. In the IP Catalog, locate and double-click **eCPRI Intel FPGA IP**. The **New IP Variant** window appears.
2. Click **OK**. The parameter editor appears.

Figure 2. Example Design Tab in the eCPRI Intel FPGA IP Parameter Editor



3. 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.
4. Click **OK**. The parameter editor appears.
5. On the **General** tab, specify the parameters for your IP core variation.
6. On the **Example Design** tab, select the **simulation** option to generate the testbench, select the **synthesis** option to generate the hardware example design, and select **synthesis and simulation** option to generate both the testbench and the hardware design example.
7. For **Language for top level simulation file**, select **Verilog** or **VHDL**.

   Note: This option is available only when you select **Simulation** option for your example design.
8. For **Language for top level synthesis file**, select **Verilog** or **VHDL**.

   Note: This option is available only when you select **Synthesis** option for your example design.
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 (`ecpri_0_testbench`), browse to the new path and type the new design example directory name.
11. Click **OK**.

Related Information

eCPRI Intel FPGA IP User Guide

1.3. Directory Structure

The eCPRI IP core design example file directories contain the following generated files for the design example.

Figure 3. Directory Structure of the Generated Example Design

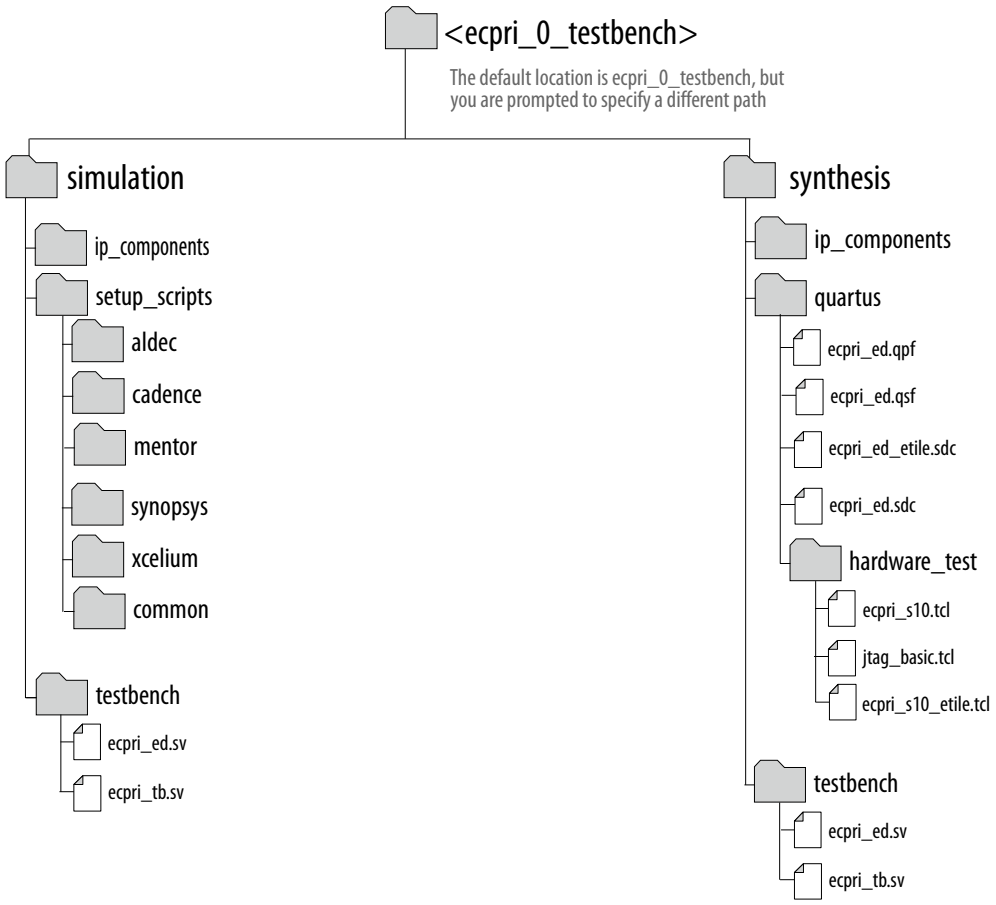


Table 1. eCPRI Intel FPGA IP Core Testbench File Descriptions

File Names	Description
Key Testbench and Simulation Files	
`<design_example_dir>/simulation/testbench/ecpri_tb.sv`	Top-level testbench file. The testbench instantiates the DUT wrapper and runs Verilog HDL tasks to generate and accept packets.
`<design_example_dir>/simulation/testbench/ecpri_ed.sv`	DUT wrapper that instantiates DUT and other testbench components.
Testbench Scripts	
`<design_example_dir>/simulation/setup_scripts/mentor/run_vsim.do`	The Mentor Graphics script to run the testbench.
`<design_example_dir>/simulation/setup_scripts/synopsys/vcs/run_vcs.sh`	The Synopsys VCS script to run the testbench.
`<design_example_dir>/simulation/setup_scripts/synopsys/vcsmx/run_vcsmx.sh`	The Synopsys VCS MX script (combined Verilog HDL and SystemVerilog with VHDL) to run the testbench.

continued...

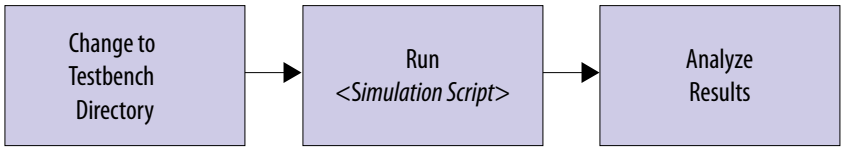
File Names	Description
<design_example_dir>/simulation/setup_scripts/cadence/run_ncsim.sh	The Cadence NCSim script to run the testbench.
<design_example_dir>/simulation/setup_scripts/aldec/run_rivierapro.tcl	The Aldec Riviera script to run the testbench.
<design_example_dir>/simulation/setup_scripts/xcelium/run_xcelium.sh	The script to run the testbench.

Table 2. eCPRI Intel FPGA IP Core Hardware Design Example File Descriptions

File Names	Descriptions
<design_example_dir>/synthesis/quartus/ecpri_ed.qpf	Intel Quartus Prime project file.
<design_example_dir>/synthesis/quartus/ecpri_ed.qsf	Intel Quartus Prime project setting file.
<design_example_dir>/synthesis/quartus/ecpri_ed.sdc	Synopsys Design Constraints files. You can copy and modify these files for your own Intel Stratix 10 design.
<design_example_dir>/synthesis/testbench/ecpri_ed_top.sv	Top-level Verilog HDL design example file.
<design_example_dir>/synthesis/testbench/ecpri_ed.sv	DUT wrapper that instantiates DUT and other testbench components.
<design_example_dir>/synthesis/quartus//ecpri_s10.tcl	Main file for accessing System Console (Only available in H-tile device variations).
<design_example_dir>/synthesis/quartus//ecpri_s10_etile.tcl	Main file for accessing System Console (Only available in E-tile device variations).

1.4. Simulating the Design Example Testbench

Figure 4. Procedure



Follow these steps to simulate the testbench:

1. At the command prompt, change to the testbench simulation directory *<design_example_dir>*/simulation/setup_scripts.
2. Run the simulation script for the supported simulator of your choice. The script compiles and runs the testbench in the simulator. Refer to the table *Steps to Simulate the Testbench*.

   Note: The VHDL language support for simulation is only available with ModelSim and VCS MX simulators. The Verilog language support for simulation is available for all simulators listed in *Table: Steps to Simulate the Testbench*.
3. Analyze the results. The successful testbench sends and receives packets, and displays "PASSED".

Table 3. Steps to Simulate the Testbench

Simulator	Instructions
Mentor Graphics ModelSim	In the command line, type `vsim -do run_vsim.do` If you prefer to simulate without bringing up the ModelSim GUI, type `vsim -c -do run_vsim.do` *Note:* The simulator does not have the capacity to simulate this IP core. You must use another supported ModelSim simulator such as SE.
Cadence NCSim	In the command line, type `sh run_ncsim.sh`
Synopsys VCS	In the command line, type `sh run_vcs.sh`
Synopsys VCSMX	In the command line, type `sh run_vcsmx.sh`
Aldec Riviera	In the command line, type `vsim -c -do run_rivierapro.tcl`
Xcelium	In the command line, type `sh run_xcelium.sh`

The following sample output illustrates a successful simulation test run for H-tile variation of the eCPRI IP design example:

```
     INFO: Out of reset status
_______________________________________________________________


         eCPRI TX SOPs count : 0
         eCPRI TX EOPs count : 0
         eCPRI RX SOPs count : 0
         eCPRI RX EOPs count : 0
         External PTP TX SOPs count : 0
         External PTP TX EOPs count : 0
         External MISC TX SOPs count : 0
         External MISC TX EOPs count : 0
         External RX SOPs count : 0
         External RX EOPs count : 0
_______________________________________________________________
     INFO: Start transmitting packets
_______________________________________________________________


_______________________________________________________________
     INFO: Stop transmitting packets
_______________________________________________________________


_______________________________________________________________
     INFO: Checking packets statistics
_______________________________________________________________


         eCPRI SOPs transmitted: 1000
         eCPRI EOPs transmitted: 1000
         eCPRI SOPs received: 1000
         eCPRI EOPs received: 1000
         External PTP SOPs transmitted: 128
         External PTP EOPs transmitted: 128
         External MISC SOPs transmitted: 100
         External MISC EOPs transmitted: 100
         External SOPs received: 228
         External EOPs received: 228
```

1.5. Compiling the Compilation-Only Project

To compile the compilation-only example project, follow these steps:

1. Ensure compilation design example generation is complete.
2. In the Intel Quartus Prime Pro Edition software, open the Intel Quartus Prime Pro Edition project <*design_example_dir*>/synthesis/quartus/ecpri_ed.qpf.
3. On the Processing menu, click **Start Compilation**.
4. After successful compilation, reports for timing and for resource utilization are available in your Intel Quartus Prime Pro Edition session. Go to **Processing ➤ Compilation Report** to view the detailed report on compilation.

Related Information

Block-Based Design Flows

1.6. Compiling and Configuring the Design Example in Hardware

To compile the hardware design example and configure it on your Intel Stratix 10 device, follow these steps:

1. Ensure hardware design example generation is complete.
2. In the Intel Quartus Prime Pro Edition software, open the Intel Quartus Prime project <*design_example_dir*>/synthesis/quartus/ecpri_ed.qpf.
3. On the Processing menu, click **Start Compilation**.
4. After successful compilation, a .sof file is available in <*design_example_dir*>/synthesis/quartus/output_files directory. Follow these steps to program the hardware design example on the Intel Stratix 10 device:
   a. Connect Development Kit to the host computer.
   b. Launch the Clock Control application, which is part of the development kit, and set the new frequencies for the design example. Below is the frequency setting in the Clock Control application:
      - If you are targeting your design on Intel Stratix 10 GX SI Development Kit:
        — U5, OUT8- 100 MHz
        — U6, OUT3- 322.265625 MHz
      - If you are targeting your design on Intel Stratix 10 TX SI Development Kit:
        — U1, CLK4- 322.265625 MHz
        — U3, OUT3- 100 MHz
   c. On the **Tools** menu, click **Programmer**.
   d. In the Programmer, click **Hardware Setup**.
   e. Select a programming device.
   f. Select and add the Intel Stratix 10 GX/TX Transceiver Signal Integrity Development to which your Intel Quartus Prime Pro Edition session can connect.

g. Ensure that **Mode** is set to **JTAG**.

h. Select the Intel Stratix 10 device and click **Add Device**. The Programmer displays a block diagram of the connections between the devices on your board.

i. Load the `.sof` file to your Intel Stratix 10 device

j. In the row with your `.sof`, check the **Program/Configure** box for the `.sof` file.

k. Click **Start**.

Related Information

- Block-Based Design Flows
- Programming Intel FPGA Devices
- Analyzing and Debugging Designs with System Console

1.7. Testing the eCPRI Intel FPGA IP Design Example

After you compile the eCPRI Intel FPGA IP core design example and configure it on your Intel Stratix 10 device, you can use the System Console to program the IP core and its embedded Native PHY IP core registers.

To turn on the System Console and test the hardware design example, follow these steps:

1. After the hardware design example is configured on the Intel Stratix 10 device, in the Intel Quartus Prime Pro Edition software, on the **Tools** menu, click **System Debugging Tools ➤ System Console**.
2. In the Tcl Console pane, change directory to `<design_example_dir>/synthesis/quartus/hardware_test`.
3. Type `source ecpri_s10.tcl` (for H-tile design example) or `source ecpri_s10_etile.tcl` (for E-tile design example) to open a connection to the JTAG master and start the test.

   If you use 25G Ethernet Intel Stratix 10 Intel FPFA IP for your Intel Stratix 10 H-tile device variations, use the following commands to program the loopback:

   - `loop_on`: Enables TX to RX internal serial loopback.
   - `loop_off`: Disables TX to RX internal serial loopback.

   If you use E-tile Hard IP for Ethernet for your Intel Stratix 10 E-tile device variations, use the following command to program the serial loopback:

   Note: For proper transceiver calibration in E-tile device variations, you must perform either an internal or external loopback command once after you program the `sof` file.

   - `sl4_link_init_int_lpbk`: Enables TX to RX internal serial loopback within the transceiver and performs the transceiver calibration flow.
   - `sl4_link_init_ext_lpbk`: Enables TX to RX external loopback and performs the transceiver calibration flow.

- You can program the IP core with the following additional design example commands:
  - `traffic_gen_enable`: Resets the entire design system, and enables the traffic generator and checker.
  - `traffic_gen_disable`: Disables the traffic generator and checker.
  - `chkmac_stats`: Displays the statistics for the Ethernet MAC.
  - `read_test_statistics`: Display the error statistics for traffic generator and checkers.
  - `ext_continuous_mode_en`: Resets the entire design system, and enables the traffic generator to generate continuous traffic packets.

The following sample output illustrates a successful test run:

```
================================================================================
==============
                        STATISTICS FOR BASE 0x20000000 (Tx)
================================================================================
==============
Fragmented Frames               : 0
Jabbered Frames                 : 0
Any Size with FCS Err Frame     : 0
Right Size with FCS Err Fra     : 0
Multicast data  Err Frames      : 0
Broadcast data Err  Frames      : 0
Unicast data Err  Frames        : 0
Multicast control  Err Frame    : 0
Broadcast control Err  Frame    : 0
Unicast control Err  Frames     : 0
Pause control Err  Frames       : 0


64 Byte Frames                  : 3072
65 - 127 Byte Frames            : 259867903
128 - 255 Byte Frames           : 25986817
256 - 511 Byte Frames           : 0
512 - 1023 Byte Frames          : 0
1024 - 1518 Byte Frames         : 0
1519 - MAX Byte Frames          : 0
> MAX Byte Frames               : 0
Tx Frame Starts                 : 285857794
Multicast data  OK  Frame       : 285857792
Broadcast data OK   Frame       : 0
Unicast data OK   Frames        : 0
Multicast Control Frames        : 0
Broadcast Control Frames        : 0
Unicast Control Frames          : 0
Pause Control Frames            : 0
Payload Octets OK               : 23648123502
Frame Octets OK                 : 28897511044

================================================================================
==============
                        STATISTICS FOR BASE 0x20000000 (Rx)
================================================================================
==============
Fragmented Frames               : 0
Jabbered Frames                 : 0
Any Size with FCS Err Frame     : 0
Right Size with FCS Err Fra     : 0
Multicast data  Err Frames      : 0
Broadcast data Err  Frames      : 0
Unicast data Err  Frames        : 0
Multicast control  Err Frame    : 0
Broadcast control Err  Frame    : 0
Unicast control Err  Frames     : 0
```

```
Pause control Err  Frames        : 0


64 Byte Frames                   : 14080
65 - 127 Byte Frames             : 1204663009
128 - 255 Byte Frames            : 120466430
256 - 511 Byte Frames            : 0
512 - 1023 Byte Frames           : 0
1024 - 1518 Byte Frames          : 0
1519 - MAX Byte Frames           : 0
> MAX Byte Frames                : 0
Rx Frame Starts                  : 1325143521
Multicast data  OK  Frame        : 1325143519
Broadcast data OK   Frame        : 0
Unicast data OK   Frames         : 0
Multicast Control Frames         : 0
Broadcast Control Frames         : 0
Unicast Control Frames           : 0
Pause Control Frames             : 0
Payload Octets OK                : 109624995810
Frame Octets OK                  : 133959444894

TX SOPs count: 1206190717
TX EOPs count: 1206271236
RX SOPs count: 1206357250
RX EOPs count: 1206427289
Checker Errors: 0
Checker Error Counts: 0
Ext PTP TX SOPs count: 128
Ext PTP TX EOPs count: 128
Ext MISC TX SOPs count: 120682090
Ext MISC TX EOPs count: 120690366
Ext RX SOPs count : 120713321
Ext RX EOPs count : 120720965
Ext Checker Errors: 0
Ext Checker Error Counts: 0
```

2. Design Example Description

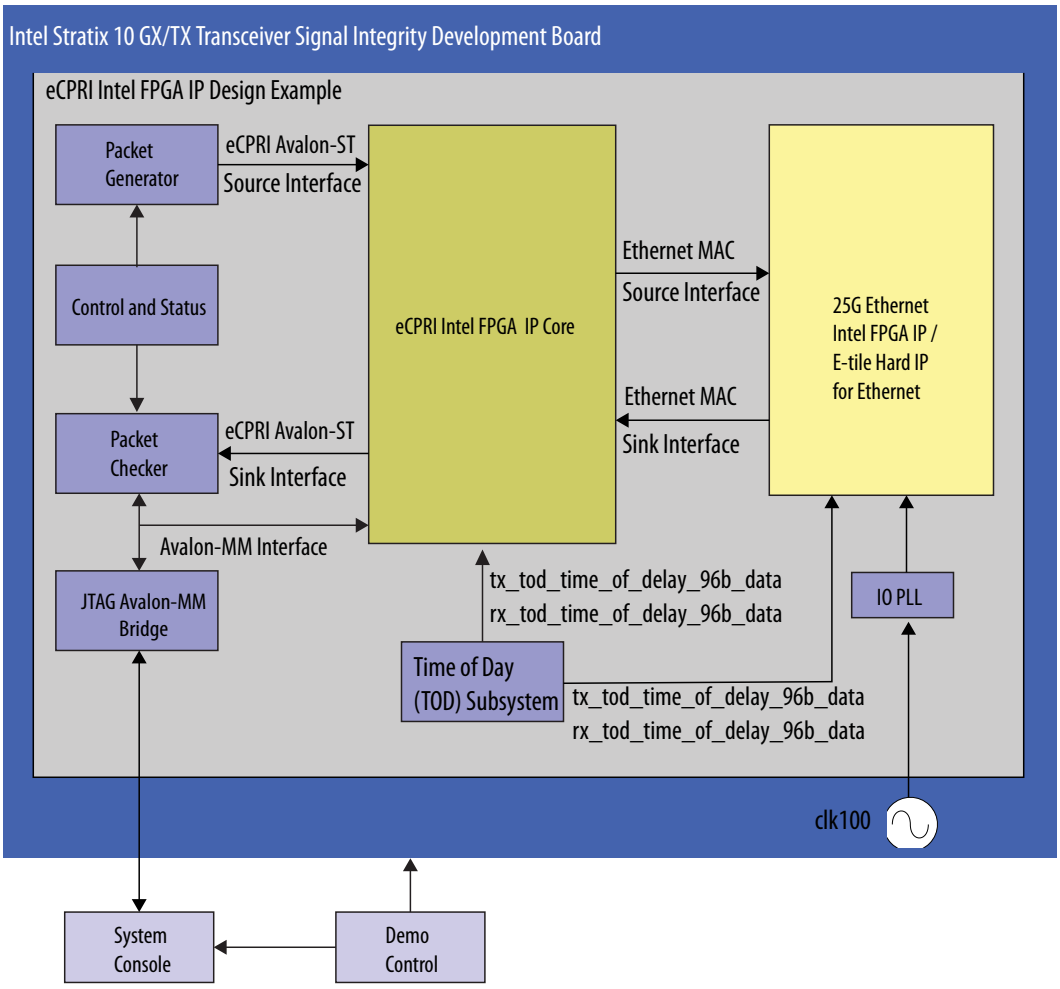
The design example demonstrates the basic functionality of the eCPRI IP core. You can generate the design from the Example Design tab in the eCPRI IP parameter editor.

2.1. Features

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

2.2. Hardware Design Example

Figure 5. eCPRI Intel FPGA IP Hardware Design Examples High Level Block Diagram



The eCPRI Intel FPGA IP core hardware design example includes the following components:

- eCPRI Intel FPGA IP Core- accepts data from the traffic generators instantiated within the test wrapper and prioritize the data for transmission to the Ethernet IP.
- 25G Ethernet Intel Stratix 10 IP (for H-tile) / E-tile Hard IP for Ethernet (for E-tile)- instantiated with the eCPRI IP.
- IO PLL- generates the latency measurement input reference clock for the Ethernet IP and sampling clock for Time of Day (TOD) subsystem. Refer to the *25G Ethernet Intel Stratix 10 FPGA IP User Guide* and *E-tile Hard IP User Guide* for more information.

- Test Wrapper- consists of traffic generators and checkers which generates different set of data packets to the Avalon Streaming (Avalon-ST) interfaces of the eCPRI IP as below:
  - eCPRI packets to the Avalon-ST source/sink interfaces:
    - Only supports message type 2.
    - Back-to-back mode generation with incremental pattern mode generation and payload size of 72 bytes for each packet.
    - Configurable via CSR to run in either non-continuous or continuous mode
    - TX/RX packet statistic status available to access via CSR.
  - Precision Clock Synchronization Protocol (1588 PTP) packet and non-PTP miscellaneous packets to the External source/sink interfaces:
    - Static Ethernet header generation with pre-defined parameters: Ethertype- 0x88F7, Message type- Opcode 0 (Sync), and PTP version-0.
    - Pre-defined pattern mode generation with interpacket gap of 2 cycles and payload size of 57 bytes for each packet.
    - 128 packets are generated in the period of every one second.
    - Configurable via CSR to run in either non-continuous or continuous mode.
    - TX/RX packet statistic status available to access via CSR.
  - External non-PTP miscellaneous packets:
    - Static Ethernet Header generation with pre-defined parameter, Ethertype- 0x8100 (non-PTP).
    - PRBS pattern mode generation with interpacket gap of 2 cycles and payload size of 128 bytes for each packet.
    - Configurable via CSR to run in either non-continuous or continuous mode.
    - TX/RX packet statistic status available to access via CSR.
- Time of Day (TOD) subsystem- contains two IEEE 1588 TOD modules for both TX and RX, and one IEEE 1588 TOD Synchronizer module generated by Intel Quartus Prime software.
- System Console- provides a user-friendly interface for you to do first-level debugging and monitor status of the IP, and the traffic generators and checkers.
- Demo Control- This module consists of reset synchronizer modules, and In-system Source and Probe (ISSP) modules for design system debugging and initialization process.

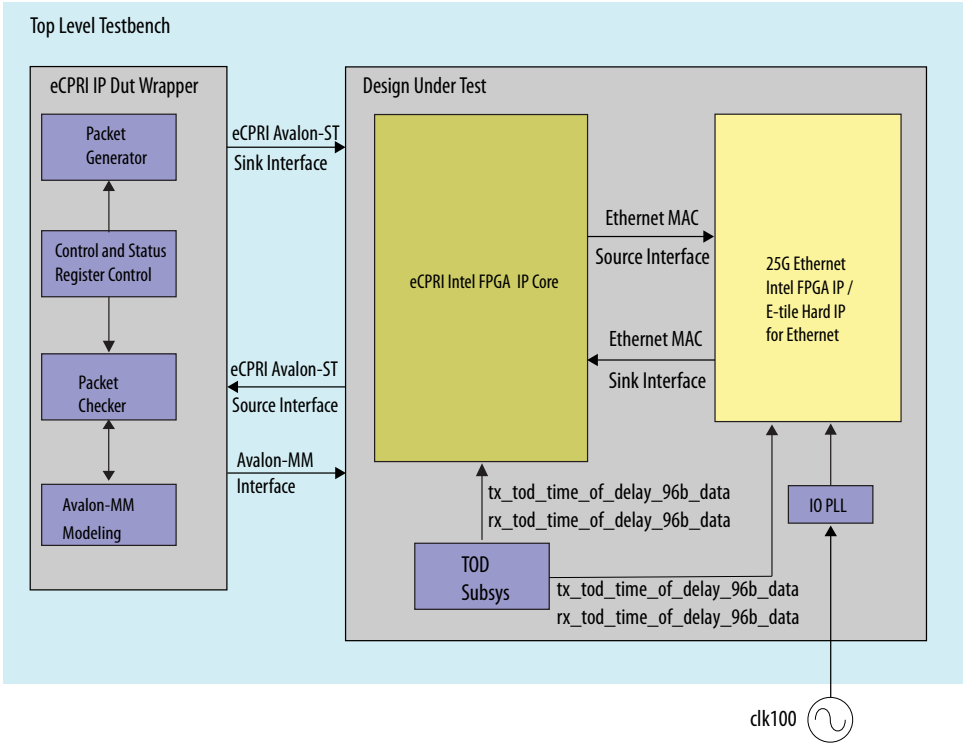
Related Information

- 25G Ethernet Intel Stratix 10 FPGA IP User Guide
- E-tile Hard IP User Guide
- eCPRI Intel FPGA IP User Guide
- 25G Ethernet Intel Stratix 10 FPGA IP Design Example User Guide
- E-tile Hard IP for Intel Stratix 10 Design Examples User Guide
- Intel Stratix 10 L- and H-Tile Transceiver PHY User Guide
- E-Tile Transceiver PHY User Guide

2.3. Simulation Design Example

The eCPRI design example generates a simulation testbench and simulation files that instantiates the eCPRI Intel FPGA IP core when you select the **Simulation** or **Synthesis & Simulation** option.

Figure 6. eCPRI Intel FPGA IP Simulation Block Diagram



In this design example, the simulation testbench provides basic functionality such as startup and wait for lock, transmit and receive packets.

The successful test run displays output confirming the following behavior:

1. The client logic resets the IP core.
2. The client logic waits for the RX datapath alignment.
3. The client logic transmits packets on the Avalon-ST interface.
4. Receive and checks for the content and correctness of the packets.
5. Display "Test PASSED" message.

2.4. Interface Signals

Table 4. Design Example Interface Signals

Signal	Direction	Description
`clk_ref`	Input	Reference clock for the Ethernet MAC. Drive at 322.2625 MHz.
`clk100`	Input	Management clock. This clock is used to generate `latency_clk` for PTP. Drive at 100 MHz.
`mgmt_reset_n`	Input	Global reset signal.
`tx_serial`	Output	TX serial pin.
`rx_serial`	Input	RX serial pin.

2.5. Design Example Register Map

Below is the register mapping for the eCPRI IP core design example:

Table 5. eCPRI Intel FPGA IP Design Example Register Mapping

Address	Register
0x20000000 – 0x3FFFFFFF	Ethernet MAC Avalon-MM Register
0x40000000 – 0x5FFFFFFF	eCPRI IP Avalon-MM Register
0x60000000 – 0x7FFFFFFF	Ethernet Frame Avalon-MM Register
0x80000000 – 0x9FFFFFFF	Ethernet Design Test Generator/Verifier Avalon-MM Register
0xA0000000 – 0xBFFFFFFF	Ethernet MAC Native PHY Avalon-MM Register

Note: You can access the Ethernet MAC and Ethernet MAC Native PHY AVMM registers using word offset instead of byte offset.

For detailed information on Ethernet MAC, Ethernet MAC Native PHY, and eCPRI IP core register maps, refer to the respective user guides.

Table 6. eCPRI Intel FPGA IP Hardware Design Example Register Map

Word Offset	Register Type
0x0001	Start send data
0x0002	Continuous packet enable
0x0003	Clear error
0x0004	Checker errors
0x0005	TX start of packet (SOP) count
0x0006	TX end of packet (EOP) count
0x0007	RX SOP count
0x0008	RX EOP count
0x0009	Total error count

continued...

Word Offset	Register Type
0x000A	External packets error
0x000B	External PTP packets TX SOP count
0x000C	External PTP packets TX EOP count
0x0010	External misc packets TX SOP count
0x0011	External misc packets TX EOP count
0x0012	External RX packets SOP count
0x0013	External RX packets EOP count
0x0014	External error count

Related Information

- Control, Status, and Statistics Register Descriptions
  Register information for the 25G Ethernet Stratix 10 FPGA IP
- Reconfiguration and Status Register Descriptions
  Register information for the E-tile Hard IP for Ethernet
- Registers
  Register information for the eCPRI Intel FPGA IP

3. Document Revision History for the eCPRI Intel Stratix 10 FPGA IP Design Example User Guide

Document Version	Intel Quartus Prime Version	IP Version	Changes
2020.04.13	19.4	1.0.0	Initial release.