1:2 and 1:1 MIPI CSI-2 Camera Interface Bridge Soft IP

User Guide

FPGA-IPUG-02019-1.1

April 2019

Contents

Figures

Tables

1. Introduction

The Mobile Industry Processor Interface (MIPI®) provides specifications for standardization in consumer mobile devices. MIPI Camera Serial Interface (CSI-2) and MIPI D-PHY specifications were developed to create a standardized interface for all cameras used in the mobile industry. New uses for image sensors continue to develop within the mobile, IoT, Industrial and automotive markets.

A CSI-2 to CSI-2 repeater or a CSI-2 to CSI-2 splitter along with some programmability can solve the following problems and enable new solutions:

- When cameras need to be placed longer distances away from the application processor, a repeater may be required to improve signal integrity across the PCBs and connectors of the system.
- Some systems may have a processor that has certain restrictions on timing between short and long packets and need special adjustments that the camera cannot perform by itself.
- Updating software that is in the processor can be expensive if extensive qualification is required, and a device that can translate the video stream to behave like a previous generation camera can provide a lower fixed cost alternative.
- Some new systems leverage both an application processor and a specialty image signal processor, and benefit from having the same image sensor be transmitted to the two processors through a splitter.

The Lattice Semiconductor MIPI CSI-2 to CSI-2 Camera Interface Bridge IP allows you to resolve these interfacing problems with the Lattice Semiconductor CrossLink™ programmable device.

This user guide is for 1:2 and 1:1 MIPI CSI-2 Camera Interface Bridge Soft IP design version 1.x.

1.1. Quick Facts

Table 1.1 provides quick facts about the MIPI CSI-2 to CSI-2 Camera Interface Bridge IP for CrossLink device.

Table 1.1. MIPI CSI-2 to CSI-2 Camera Interface Bridge IP Quick Facts

		MIPI CSI-2 to CSI-2 Camera Interface Bridge IP Configuration			
		4-Lane Gear 16 Continuous Rx Clock	4-Lane Gear 8 Continuous Rx Clock	4-Lane Gear 16 Non-continuous Rx Clock	4-Lane Gear 8 Non-continuous Rx Clock
IP Requirements	FPGA Families Supported	CrossLink			
Resource Utilization	Target Device	LIF-MD6000-6MG81I			
	LUTs	2013	1514	2085	1598
	EBRs	8	4	8	4
	Registers	1474	1020	1489	1035
	Programmable I/O	13	13	14	14
Design Tool Support	Lattice Implementation	Lattice Diamond® 3.8 or later			
	Synthesis	Lattice Synthesis Engine			
		Synopsys® Synplify Pro® L-2016.03L			
	Simulation	Aldec® Active HDL™ 10.3 Lattice Edition			

1.2. Features

The key features of the MIPI CSI-2 to CSI-2 Camera Interface Bridge IP are:

- Interfaces one or two MIPI CSI-2 compliant receivers to a MIPI CSI-2 transmitter
- Supports up to 6 Gb/s per MIPI CSI-2 interface.
- Supports 1, 2, 3 or 4 data lanes and one clock lane per MIPI CSI-2 interface
- Supports all MIPI CSI-2 compatible video formats (RAW, YUV, RGB and User Defined)
- Compliant with MIPI D-PHY v1.1 and MIPI CSI-2 v1.1 specifications

1.3. Conventions

1.3.1. Nomenclature

The nomenclature used in this document is based on Verilog HDL. This includes radix indications and logical operators.

1.3.2. Data Ordering and Data Types

The highest bit within a data bus is the most significant bit.

Single-bit data stream from each MIPI D-PHY data lane is deserialized into 8-bit or 16-bit parallel data where bit 0 is the first received bit. The size of the parallel data bus depends on the Rx gear setting.

When gear setting is 16, the byte in the lower 8 bits of the 16-bit parallel data is the first byte.

Byte arrangement within data words is illustrated in Figure 1.1.

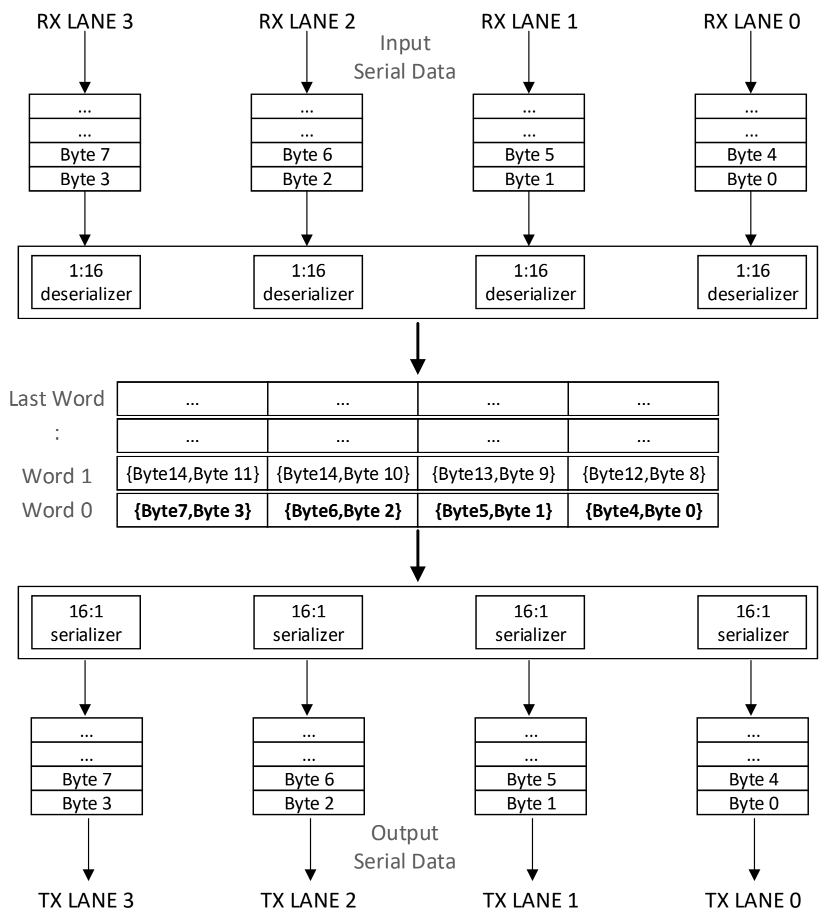


Figure 1.1. Data Ordering for a Gear 16 x4 Configuration

1.3.3. Signal Names

Signal names that end with:

- _*i* are input pins.
- _*o* are output pins.
- _*io* are bi-directional pins.
- _*n_i* are active low input signals.

2. Functional Description

Figure 2.1 shows the top level block diagram of the 1:2 MIPI CSI-2 Camera Interface Bridge IP.

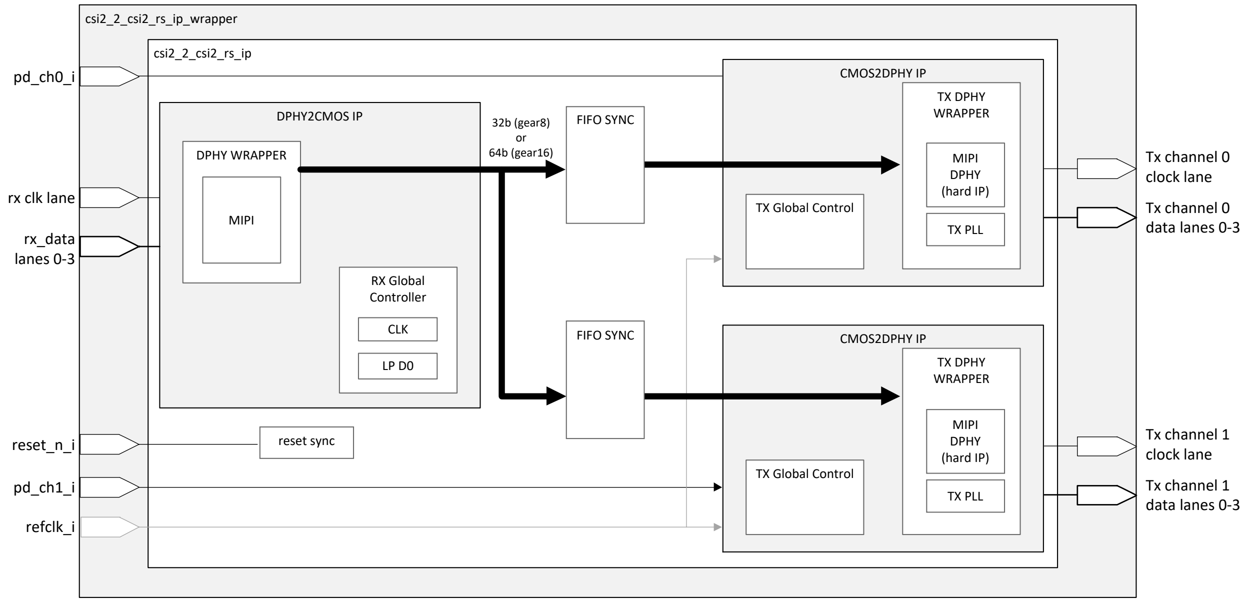


Figure 2.1. MIPI CSI-2 to CSI-2 Camera Interface Bridge IP Block Diagram

Table 2.1 lists the ports of the 1:2 MIPI CSI-2 Camera Interface Bridge IP.

Table 2.1. Top Level Ports

Signal	Direction	Functional Description
clk_p_i	I	Positive differential Rx D-PHY input clock
clk_n_i	I	Negative differential Rx D-PHY input clock
d0_p_io	I/O	Positive differential Rx D-PHY input data 0
d0_n_io	I/O	Negative differential Rx D-PHY input data 0
d1_p_i	I	Positive differential Rx D-PHY input data 1
d1_n_i	I	Negative differential Rx D-PHY input data 1
d2_p_i	I	Positive differential Rx D-PHY input data 2
d2_n_i	I	Negative differential Rx D-PHY input data 2
d3_p_i	I	Positive differential Rx D-PHY input data 3
d3_n_i	I	Negative differential Rx D-PHY input data 3
clk_ch0_p_o	O	Positive differential Tx D-PHY output clock, channel 0
clk_ch0_n_o	O	Negative differential Tx D-PHY output clock, channel 0
d0_ch0_p_io	I/O	Positive differential Tx D-PHY output data 0, channel 0
d0_ch0_n_io	I/O	Negative differential Tx D-PHY output data 0, channel 0
d1_ch0_p_o	O	Positive differential Tx D-PHY output data 1, channel 0
d1_ch0_n_o	O	Negative differential Tx D-PHY output data 1, channel 0
d2_ch0_p_o	O	Positive differential Tx D-PHY output data 2, channel 0
d2_ch0_n_o	O	Negative differential Tx D-PHY output data 2, channel 0
d3_ch0_p_o	O	Positive differential Tx D-PHY output data 3, channel 0
d3_ch0_n_o	O	Negative differential Tx D-PHY output data 3, channel 0
clk_ch1_p_o	O	Positive differential Tx D-PHY output clock, channel 1
clk_ch1_n_o	O	Negative differential Tx D-PHY output clock, channel 1
d0_ch1_p_io	I/O	Positive differential Tx D-PHY output data 0, channel 1

Signal	Direction	Functional Description
d0_ch1_n_io	I/O	Negative differential Tx D-PHY output data 0, channel 1
d1_ch1_p_o	O	Positive differential Tx D-PHY output data 1, channel 1
d1_ch1_n_o	O	Negative differential Tx D-PHY output data 1, channel 1
d2_ch1_p_o	O	Positive differential Tx D-PHY output data 2, channel 1
d2_ch1_n_o	O	Negative differential Tx D-PHY output data 2, channel 1
d3_ch1_p_o	O	Positive differential Tx D-PHY output data 3, channel 1
d3_ch1_n_o	O	Negative differential Tx D-PHY output data 3, channel 1
ref_clk_i	I	Input reference clock in Non-continuous Rx clock mode
reset_n_i	I	Asynchronous active low system reset
pd_ch0_i	I	Power down pin of Tx channel 0
pd_ch1_i	I	Power down pin of Tx channel 1
Miscellaneous Debug Ports		
tx0_tinit_done	O	Indicates the Initialization delay counter from Tx channel 0 is done
tx0_pll_lock	O	PLL lock indicator for Tx channel 0
tx0_byteclock	O	Tx channel 0 output byte clock
tx0_lp_clk_en	O	Low-power clock enable of Tx channel 0
fifo0_empty	O	Indicates that synchronizing FIFO for Tx channel 0 is empty
tx1_tinit_done	O	Indicates the Initialization delay counter from Tx channel 1 is done
tx1_pll_lock	O	PLL lock indicator for Tx channel 1
tx1_byteclock	O	Tx channel 1 output byte clock
tx1_lp_clk_en	O	Low-power clock enable of Tx channel 1
fifo1_empty	O	Indicates that synchronizing FIFO for Tx channel 1 is empty

2.1. Design and Module Description

Within the csi2_2_csi2_rs_ip block are instances of:

- dphy2cmos.v
  - Instantiates the MIPI D-PHY Rx soft IP wrapper. The D-PHY wrapper uses Snow DDR I/O and fabric to receive MIPI serial data. This converts the incoming serial data from the D-PHY data lanes to 32-bit (gear8) or 64-bit (gear 16) words.
  - Instantiates the Rx global controller which contains finite state machines (FSMs) that detects the state transitions of the clock and data lanes.
- fifo_sync.v
  - The design uses two instances of 32-word deep FIFO used to pass the Rx data to the Tx clock domains, one for each Tx channel.
  - The purpose of these FIFOs are for clock domain synchronization only and not for data buffering. The MIPI D-PHY specification does not allow for flow control, therefore, the Rx and the Tx byte clock must equal.
- cmos2dphy.v
  - There are two instances of this IP block, one for each Tx channel. Each instance contains the hard MIPI D-PHY Tx wrapper that serializes the packet data. Each wrapper contains its own Tx PLL.
  - This also contains the Tx Global Control module that controls the transitions of the Tx clock and data lanes.

3. Parameter Settings

Table 3.1 lists the user parameters used to generate the design.

Table 3.1. MIPI CSI-2 to CSI-2 Camera Interface Bridge IP Parameter Settings

Parameter	Attribute	Options	Description
Number of Rx Lanes	User - configurable	1, 2, 3 or 4	This selects the number of MIPI D-PHY data lanes. The number of Tx data lanes is the same as the number of Rx lanes.
Rx Gear	User - configurable	Gear8 or Gear16	Input serial bits are converted into 8-bit or 16-bit data bus. Gear8 is automatically selected for line rates lower than 250 Mb/s, and Gear16 is automatically selected for line rates above 900 Mb/s.
Rx D-PHY IP	User - configurable	Hard D-PHY or Soft D-PHY	The Hard IP option is only available for single CSI-2 to single CSI-2 configuration. Selecting Hard IP makes use of one of the hardened D-PHY modules as the Rx D-PHY. Selecting the Soft D-PHY IP option generates a Soft D-PHY Rx logic. The maximum line rate when Soft IP is selected is 1.2 Gb/s.
Lane Aligner FIFO Type	User - configurable	EBR or LUT	If lane aligner is enabled, you have the option to choose FIFO implementation for resource optimization.
Enable Lane Aligner	User - configurable	ON or OFF	Word aligner is sufficient to handle data lane skews within 1 byte clock period. Option to use lane aligner is retained in case the customer's system have more than 1 byte clock period of skew.
Number of Tx Channels	Read-Only	1 or 2	Rx D-PHY IP should be set to Soft D-PHY to enable this option. This option configures the bridge to have two transmission channels. The two channels are asynchronous with each other.
Rx Line Rate	User - configurable	192–1440	Data rate per lane in Mb/s. It must be noted that T_{LPX}, the D-PHY low-power state period, must be twice the byte clock period.
D-PHY Clock Frequency	Read-Only	96–720	The D-PHY Clock Frequency is half of the line rate, in MHz.
Rx D-PHY Clock Mode	User - configurable	Continuous or Non-Continuous	In continuous mode, the input D-PHY clock lanes are always in high-speed mode. The CSI-2 to CSI-2 bridge utilizes this clock to generate the byte clock for internal logic. In non-continuous mode, the input D-PHY clock lanes go to low-power states. Therefore, an external clock, refclk_i, is needed.
Tx D-PHY Clock Mode	User - configurable	Continuous or Non-Continuous	In continuous mode, the output D-PHY clock lanes are always in high speed mode. In non-continuous mode, the output D-PHY clock lanes go to low power states.
Byte Clock Frequency	Read-Only	(Line rate) / (Rx Gear)	This is the frequency that the internal logic operates at.
Reference Clock Frequency	Read-Only	(Line rate) / (Rx Gear)	The frequency of the reference clock for the Tx PLL. This clock is also used to clock the Rx byte clock domain if the D-PHY Clock Mode is non-continuous.

Parameter	Attribute	Options	Description
tINIT_SLAVE Value	User - configurable	16-bit non-zero decimal value	This parameter sets the period needed to meet the required initialization time of the CSI-2 slave. The value is in terms of byte clock cycles. The D-PHY specification places a minimum period of 100 µs, but this parameter may be increased depending on the receiver requirement. During this period, all incoming data is ignored by the bridge.
Bypass tINIT Counter	User - configurable	ON or OFF	User can bypass tINIT counter since PLL lock time takes around 15 ms, more than enough to meet D-PHY tINIT requirement.
HS-SKIP Parameter*	User - configurable	0–20	This sets the $T_{HS\text{-}SKIP}$ in terms of number of byte clock cycles. This is the time interval during which the design neglects the high-speed data lane transitions before data lane0 goes to LP-11 state. It is suggested that the minimum input trail be at least 3 byte clock cycles to give the design enough allowance to trim the glitches and still have enough trail bytes.
t_HS-PREPARE	User - configurable	1–99	This sets the $T_{HS\text{-}PREPARE}$ counter in terms of number of byte clock cycles. It is the time the transmitter drives the Data Lane LP-00 Line state immediately before the HS-0 Line state starting the HS transmission.
t_HS-ZERO	User - configurable	1–99	This sets the $T_{HS\text{-}ZERO}$ counter in terms of number of byte clock cycles. It is the time the transmitter drives the HS-0 state prior to transmitting the Sync sequence. In gear 8, the actual $T_{HS\text{-}ZERO}$ has ~2.5 cycles more than the specified value due to the register delays when converting data from parallel to serial. In gear 16, the actual value has ~3.5 cycles more.
t_CLK-PRE	User - configurable	1–99	This sets the $T_{CLK\text{-}PRE}$ counter in terms of number of byte clock cycles. This is the time that the transmitter drives the HS clock prior to any associated Data Lane beginning the transition from LP to HS mode. The actual $T_{CLK\text{-}PRE}$ has 2 more additional byte clock cycles due to register delays.
t_CLK-POST	User - configurable	1–99	This sets the $T_{CLK\text{-}POST}$ counter in terms of number of byte clock cycles. This is the time that the transmitter continues to send HS clock after the last associated Data Lane has transitioned to LP Mode. Interval is defined as the period from the end of $T_{HS\text{-}TRAIL}$ to the beginning of $T_{CLK\text{-}TRAIL}$. The actual $T_{CLK\text{-}POST}$ has one more additional byte clock cycle due to register delays.
Enable Miscellaneous Status Signals	User - configurable	ON or OFF	Enabling the miscellaneous signals ports out some internal signals for debug purpose.
Enable Tx0 tINIT Done Enable Tx1 tINIT Done Enable Tx0 PLL Lock Enable Tx1 PLL Lock Enable Tx0 Byte Clock Enable Tx1 Byte Clock Enable Tx0 LP Clock Enable	User - configurable	ON or OFF	Each miscellaneous signal can be enabled/disabled separately.

Parameter	Attribute	Options	Description
Enable Tx1 LP Clock Enable Enable Tx0 FIFO Empty Enable Tx1 FIFO Empty			

***Note**: EoT processing is not performed in the design. Thus, one or more additional bytes from data lanes may be present after the trail. Refer to *MIPI D-PHY Specification, Appendix A.3 High Speed Receive at the Slave Side.*

To clean up these unwanted bytes, it is recommended to extend the input trail and increase the HS-SKIP value.

4. IP Generation and Evaluation

This section provides information on how to generate MIPI CSI-2 to CSI-2 Camera Interface Bridge IP using the Diamond Clarity Designer, and how to run simulation, synthesis and hardware evaluation.

4.1. Licensing the IP

An IP-specific license is required to enable full, unrestricted use of the MIPI CSI-2 to CSI-2 Camera Interface Bridge IP in a complete, top-level design. The MIPI CSI-2 to CSI-2 Camera Interface Bridge IP license is available free of charge.

Request your license by going to the link http://www.latticesemi.com/en/Support/Licensing and request the free Lattice Diamond license. In this form, select the desired CrossLink IP for your design.

You may download and generate the MIPI CSI-2 to CSI-2 Camera Interface Bridge IP and fully evaluate through functional simulation and implementation (synthesis, map, place and route) without an IP license. The MIPI CSI-2 to CSI-2 Camera Interface Bridge IP also supports Lattice's IP hardware evaluation capability, which makes it possible to create versions of the IP that operate in hardware for a limited time (approximately four hours) without requiring an IP license. See the Hardware Evaluation section for further details.

HOWEVER, THE IP LICENSE IS REQUIRED TO ENABLE TIMING SIMULATION, TO OPEN THE DESIGN IN DIAMOND EPIC TOOL, OR TO GENERATE BITSTREAMS THAT DO NOT INCLUDE THE HARDWARE EVALUATION TIMEOUT LIMITATION.

4.2. Getting Started

The MIPI CSI-2 to CSI-2 Camera Interface Bridge IP is available for download from the Lattice IP Server using the Clarity Designer tool. The IP files are automatically installed using ispUPDATE technology in any customer-specified directory. After the IP has been installed, the IP is available in the Clarity Designer user interface as shown in Figure 4.1.

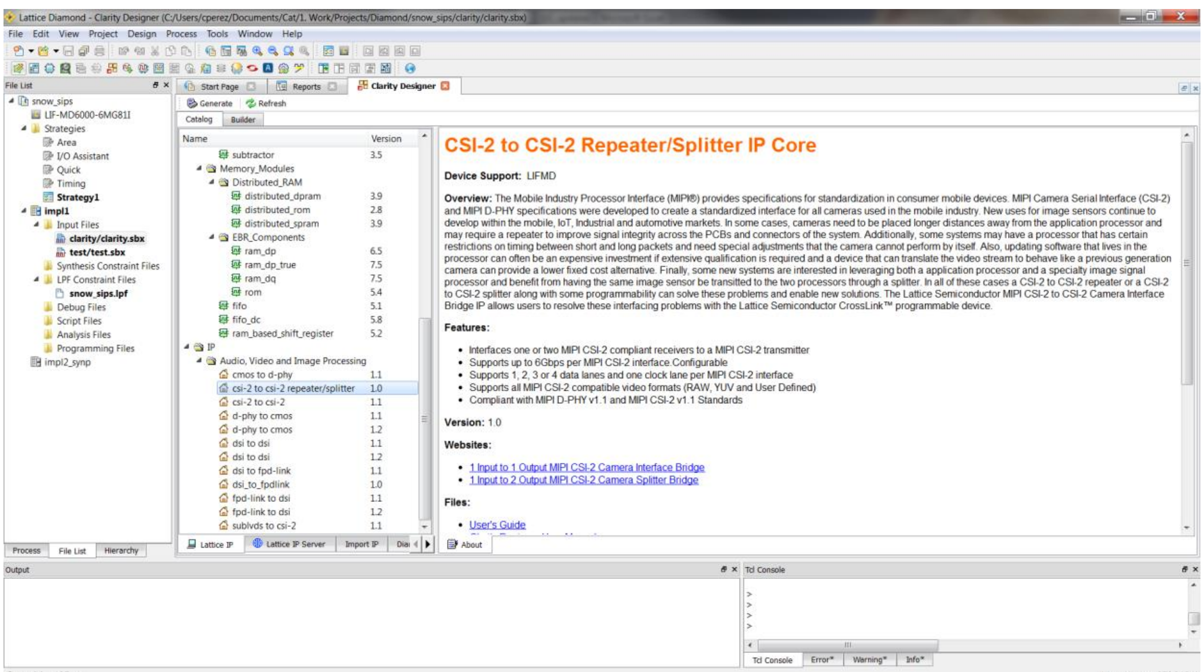


Figure 4.1. Clarity Designer Window

4.3. Generating IP in Clarity Designer

The Clarity Designer tool is used to customize modules and IPs and place them into the device's architecture. Besides configuration and generation of modules and IPs, Clarity Designer can also create a top module template in which all generated modules and IPs are instantiated.

The following describes the procedure for generating MIPI CSI-2 to CSI-2 Camera Interface Bridge IP in Clarity Designer.

Clarity Designer can be started from the Diamond design environment.

To start Clarity Designer:

1. Create a new empty Diamond project for CrossLink family devices.
2. From the Diamond main window, choose **Tools** > **Clarity Designer**, or click  in Diamond toolbox. The Clarity Designer project dialog box is displayed.
3. Select and or fill out the following items as shown in Figure 4.2.
   - **Create new Clarity design** – Choose to create a new Clarity Design project directory in which the MIPI CSI-2 to CSI-2 Camera Interface Bridge IP is generated.
   - **Design Location** – Clarity Design project directory path.
   - **Design Name** – Clarity Design project name.
   - **HDL Output** – Hardware Description Language Output Format (Verilog).

   The Clarity Designer project dialog box also allows you to open an existing Clarity Designer project by selecting the following:
   - **Open Clarity design** – Open an existing Clarity Design project.
   - **Design File** – Name of existing Clarity Design project file with .sbx extension.
4. Click the **Create** button. A new Clarity Designer project is created.

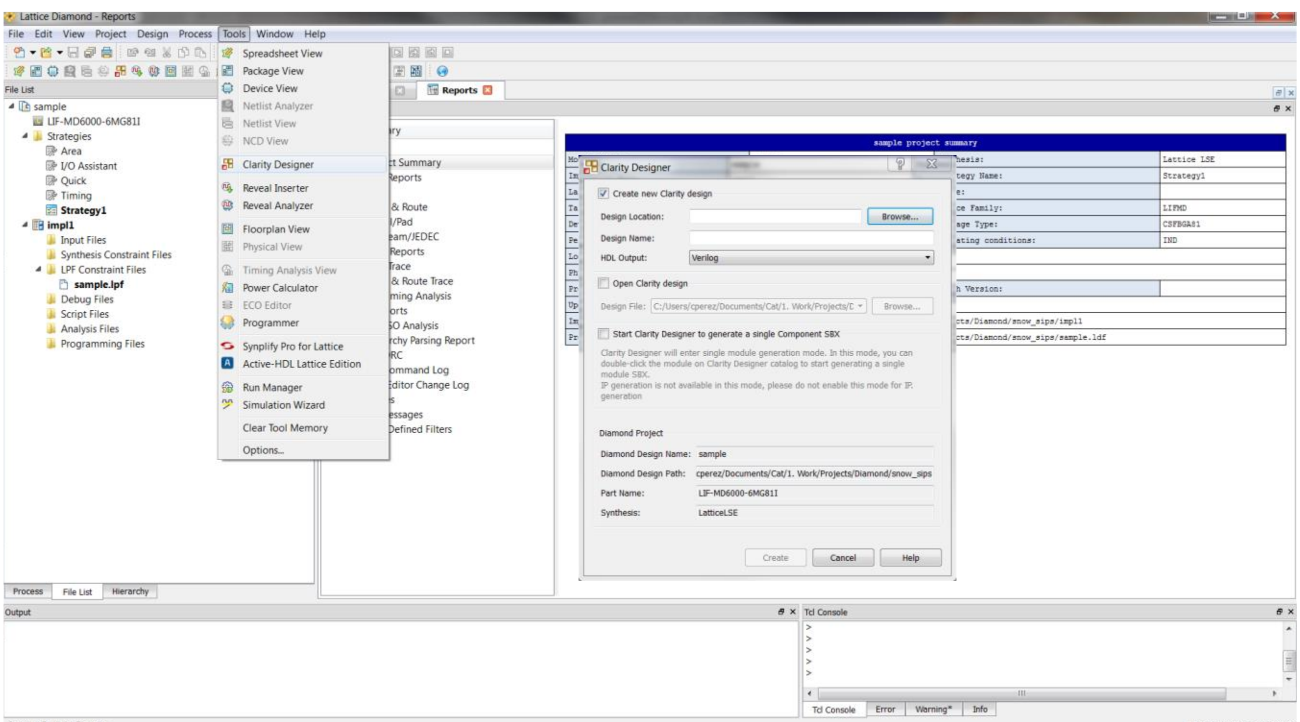


Figure 4.2. Starting Clarity Designer from Diamond Design Environment

To configure the MIPI CSI-2 to CSI-2 Camera Interface Bridge IP in Clarity Designer:

1. Double-click **csi-2 to csi-2 repeater/splitter** in the IP list of the Catalog view. The **csi-2 to csi-2 repeater/splitter** dialog box is displayed as shown in Figure 4.3.

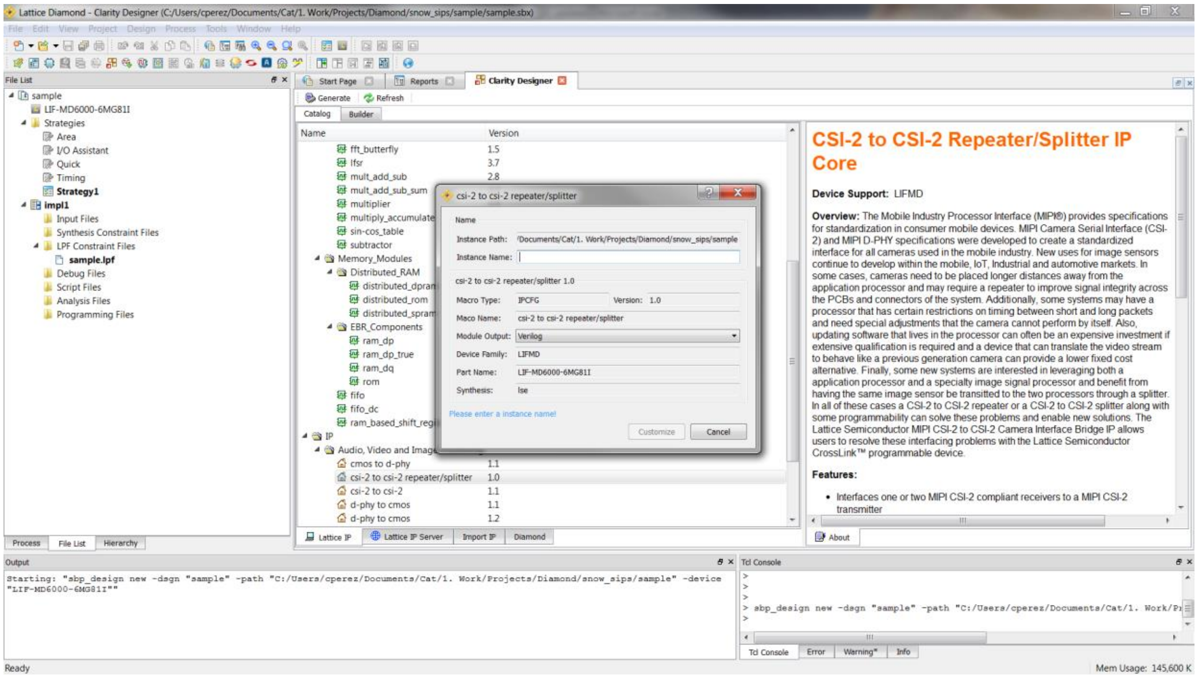


Figure 4.3. Configuring MIPI CSI-2 to CSI-2 Camera Interface Bridge IP in Clarity Designer

2. Enter the **Instance Name**.
3. Click the **Customize** button. An IP configuration interface is displayed as shown in Figure 4.4 to Figure 4.7. From this dialog box, you can select the IP parameter options specific to your application.

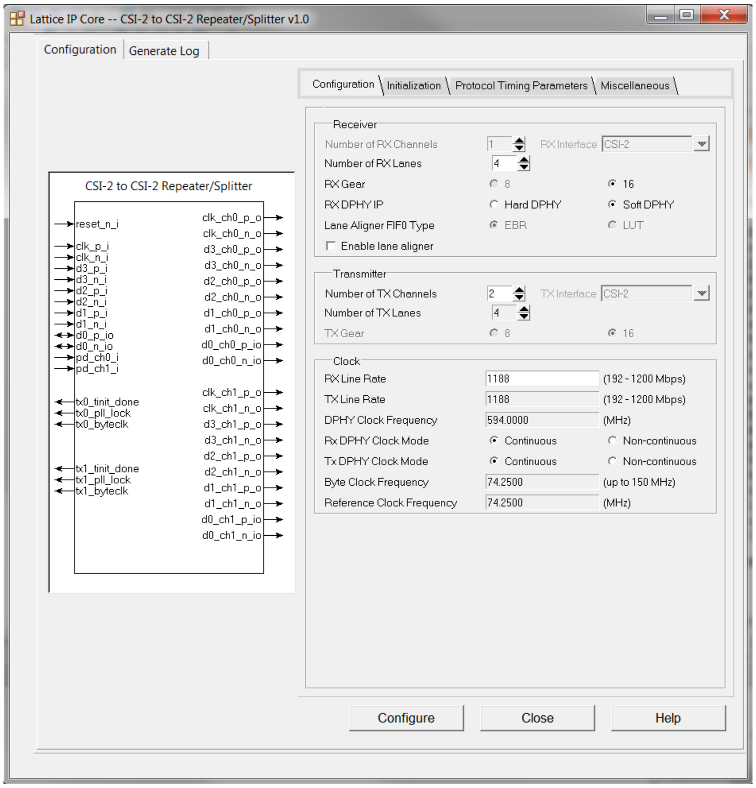


Figure 4.4. Configuration Tab in IP User Interface

4. To configure Initialization parameters, click the **Initialization** tab as shown in Figure 4.5.

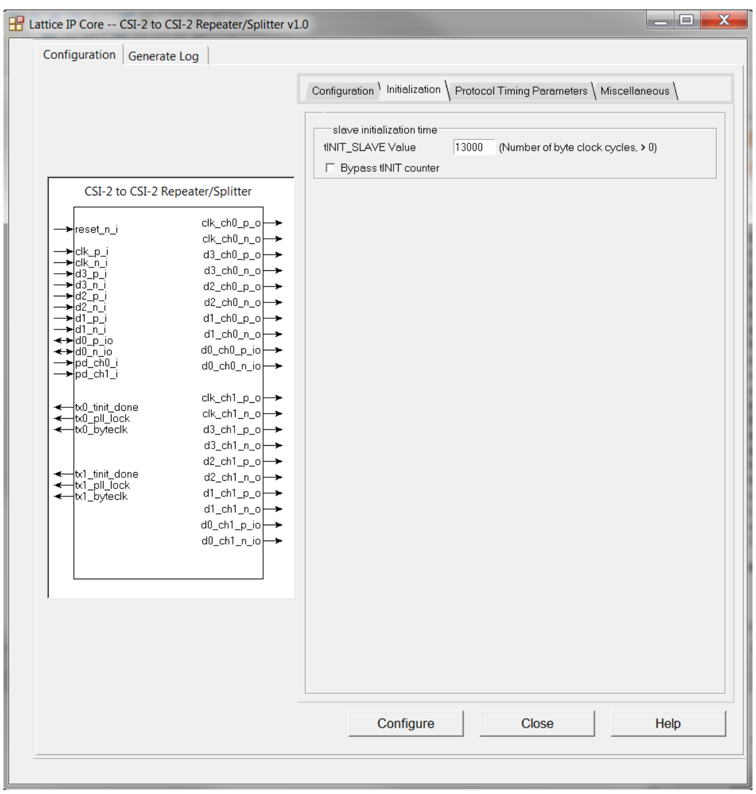


Figure 4.5. Initialization Tab in IP User Interface

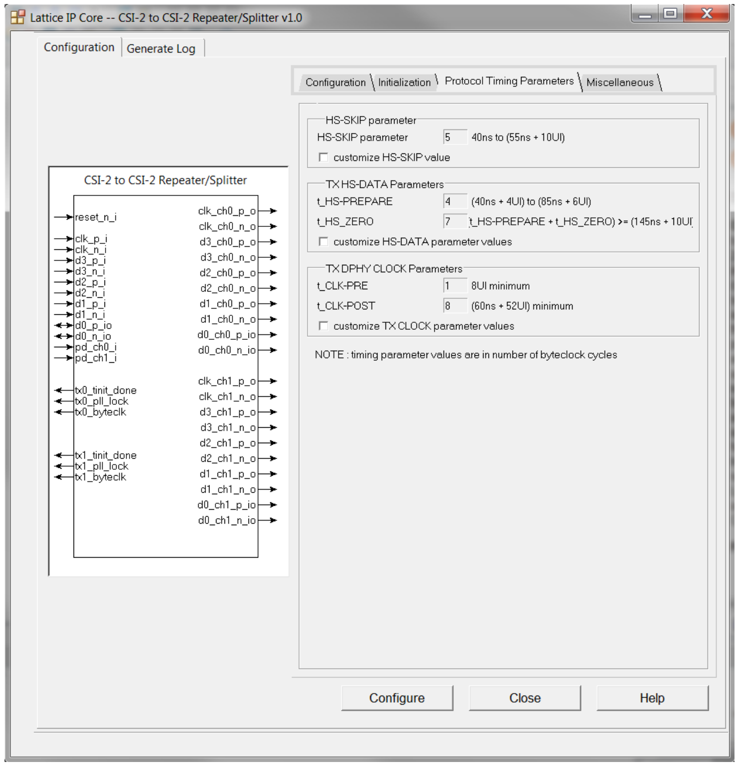


Figure 4.6. Protocol Timing Parameters Tab in IP User Interface

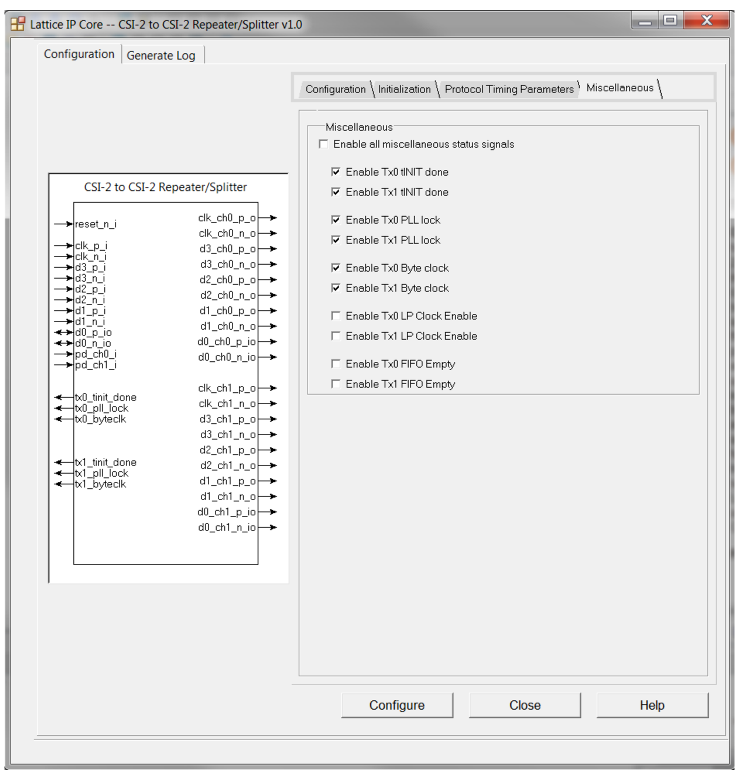


Figure 4.7. Miscellaneous Tab in IP User InterfaceI

5. Select the required parameters, and click the **Configure** button.
6. Click **Close**.

7. Click Generate in the toolbox. Clarity Designer generates all the IPs and modules, and creates a top module to wrap them.

For detailed instructions on how to use the Clarity Designer, refer to the Lattice Diamond software user guide.

4.4. Generated IP Directory Structure and Files

The directory structure of generated IP files is shown in Figure 4.8.

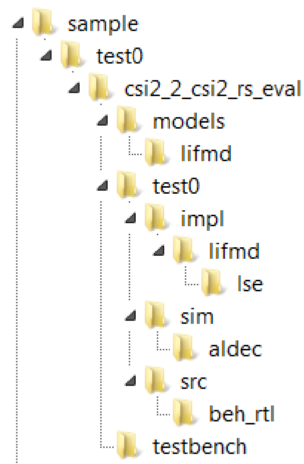


Figure 4.8. MIPI CSI-2 to CSI-2 Camera Interface Bridge IP Directory Structure

The design flow for the IP created with Clarity Designer uses post-synthesized modules (NGO) of IP core modules for synthesis and uses protected models for simulation. The post-synthesized modules are customized when you configure the IP and created automatically when the IP is generated. The protected models are common to all configurations. Other files are also provided to enable functional simulation and implementation.

Table 4.1 provides a list of key files and directories created by Clarity Designer with details on where they are located and how they are used.

Table 4.1. List of Generated Files

File	Description
<instance_name>.v	Verilog top-level module of MIP D-PHY to CMOS IP used for both synthesis and simulation.
<instance_name>_*.v	Verilog submodules for simulation. Files that do not have equivalent black box modules are also used for synthesis.
<instance_name>_inst.v/vhd	Template for instantiating the design in another user-created top module.
<instance_name>_*_beh.v	Protected Verilog models for simulation.
<instance_name>_*_bb.v	Verilog black box modules for synthesis.
<instance_name>_*.ngo	User interface configured and synthesized modules for synthesis.
<instance_name>_params.v	Verilog parameters file which contains required compiler directives to successfully configure the IP during synthesis and simulation.
<instance_name>.lpc	Lattice Parameters Configuration file. This file records all the IP configuration options set through Clarity Designer. It is used by IP generation script to generate configuration-specific IP. It is also used to reload parameter settings in the IP user interface in Clarity Designer when it is being reconfigured.

All IP files are generated inside *<project_dir>* directory (sample in Figure 4.8). The *<project_dir>* is *<design_location>\\<design_name>\\<instance_name>*, see the Generating IP in Clarity Designer section. A separate *<project_dir>* is created each time MIPI CSI-2 to CSI-2 Camera Interface Bridge IP is created with a different IP instance name.

The *csi2_to_csi2_rs_eval* and subdirectories provide files supporting push-button IP evaluation through functional simulations, design implementation (synthesis, map, place and route) and hardware evaluation. Inside *csi2_to_csi2_rs_eval is* *<instance_name>* folder (test0 in Figure 4.8) which contains protected behavioral files in *<instance_name>**src**beh_rtl* and a pre-built Diamond project in *<instance_name>**impl**lifmd**<synthesis_tool>*. The *<instance_name>* is the IP instance name you specified in Clarity Designer. The simulation part of user evaluation provides testbench and test cases supporting RTL simulation for Active-HDL simulator under *<project_dir>**testbench*. Separate directories located at *<project_dir>**csi2_to_csi2_rs_eval**<instance_name>**sim**aldec* are provided and contain specific
pre-built simulation script files. See the Running Functional Simulation section below for details.

4.5. Running Functional Simulation

To run simulations using Active HDL:

1. Modify the ***.do** file located in **\<project_dir>\csi2_to_csi2_rs_eval\<instance_name>\sim\aldec**.
   a. Specify working directory (sim_working_folder). For example:
      set sim_working_folder **C:/my_design**.
   b. Specify workspace name that is created in working directory. For example:
      set workspace_name **design_space**.
   c. Specify design name. For example:
      set design_name **DesignA**.
   d. Specify design path where the IP Core generated using Clarity Designer is located. For example:
      set design_path **C:/my_designs/DesignA**.
   e. Specify design instance name (same as the instance name specified in Clarity Designer). For example:
      set design_inst **DesignA_inst**.
   f. Specify Lattice Diamond Primitive path (diamond_dir) to where it is installed. For example:
      set diamond_dir **C:/lscc/diamond/3.8_x64**.
2. Update testbench parameters to customize data size, clock and/or other settings. See Table 4.2 and Table 4.3 for the list of valid testbench compiler directives.
3. From the **Tools** menu, select **Active-HDL**.
4. In the Active-HDL window, under the **Tools** tab, select **Execute Macro**.
5. Select the **do** file **\<project_dir>\csi2_to_csi2_rs_eval\<instance_name>\sim\aldec\csi2csi.do**:
6. Click **OK**.
7. Wait for the simulation to finish.

Table 4.2 is a list of testbench directives which can be modified by setting the define in the vlog command in the *csi2csi.do* file.

Example:

```
vlog \
+define+NUM_FRAMES=60 \
+define+NUM_LINES=1080 \
```

Table 4.2. Testbench Directives for CSI-2

Directive	Description
NUM_FRAMES	Used to set the number of video frames.
NUM_LINES	Used to set the number of lines per frame.
VIRTUAL_CHANNEL	Used to set the virtual channel number.
DPHY_DEBUG_ON	Used to enable or disable debug messages. 0 – Debug messages are disabled 1 – Debug messages are enabled
DPHY_CLK	Used to set the D-PHY clock period (in ps). By default, the testbench automatically calculates the D-PHY clock period. However, you can override/change the clock period by defining DPHY_CLK in the do file. Example: +define+DPHY_CLK=1684
PD_CH0	Used to control the power-down pin of Tx channel 0. 0 – Channel 0 is enabled 1 – Channel 0 is enabled
PD_CH1	Used to control the power-down pin of Tx channel 1. 0 – Channel 1 is enabled 1 – Channel 1 is enabled
FIXED_DATA	Used to drive fixed video data values (alternate 8'h0 and 8'hF) if set to 1. Else, video data is random. 0 – Random video data (default) 1 – Video data is alternate 8'h0 and 8'hF
PLL_DURATION, TINIT_DURATION	Used when miscellaneous signals are off (for example debug output ports for PLL lock, and/or tINIT done are not included in the generated design). This directive is used to set the duration (in ps) of PLL lock, and/or tINIT done before the D-PHY model in the testbench transmits input data to the design. Example: +define+PLL_DURATION=15000000 +define+TINIT_DURATION=100000000
REF_CLK	By default, the testbench automatically calculates the Reference clock period (for HS_LP clock mode). But you can change/override the clock period by defining this directive. Used to set the Reference clock period (HS_LP clock mode) input to the design (in ps). Example: +define+REF_CLK=6736
CSI2_LPS_GAP	Used to set low power state delay between HS transactions (in ps).
CSI2_NUM_PIXELS	Used to set the number of pixels per line.
CSI2_LS_LE_EN	Used to enable/disable D-PHY model transmission of line start and line end packets. 0 – No Line start and Line end packets 1 – Line start and Line end packets enable
RAW8, RAW10, RAW12, YUV420_10 , YUV420_8, YUV422_10, YUV422_8, LEGACY_YUV420_8, YUV420_10_CSPS, YUV420_8_CSPS, RGB888 (default)	Testbench supports the data types below. User can define one of the following data types in the do file. Example: +define+RAW8

The testbench has default setting for the D-PHY timing parameters listed in Table 4.3. Refer to *MIPI D-PHY Specification version 1.1, Table 14* for information regarding D-PHY timing requirements. To modify the D-PHY timing parameters, you can set the following directives.

Table 4.3. Testbench Directives for D-PHY Timing Parameters

Directive	Description
DPHY_LPX	Used to set T-LPX (in ps).
DPHY_CLK_PREPARE	Used to set T-CLK-PREPARE (in ps).
DPHY_CLK_ZERO	Used to set T-CLK-ZERO (in ps).
DPHY_CLK_PRE	Used to set T-CLK-PRE (in ps).
DPHY_CLK_POST	Used to set T-CLK-POST (in ps).
DPHY_CLK_TRAIL	Used to set T-CLK-TRAIL (in ps).
DPHY_HS_PREPARE	Used to set T-HS-PREPARE (in ps).
DPHY_HS_ZERO	Used to set T-HS-ZERO (in ps).
DPHY_HS_TRAIL	Used to set T-HS-TRAIL (in ps).
FRAME_LPM_DELAY	Used to set the low-power mode delay between frames (in ps).

4.6. Simulation Strategies

This section describes the simulation environment which demonstrates basic MIPI CSI-2 to CSI-2 Camera Interface Bridge IP functionality. Figure 4.9 shows the block diagram of simulation environment.

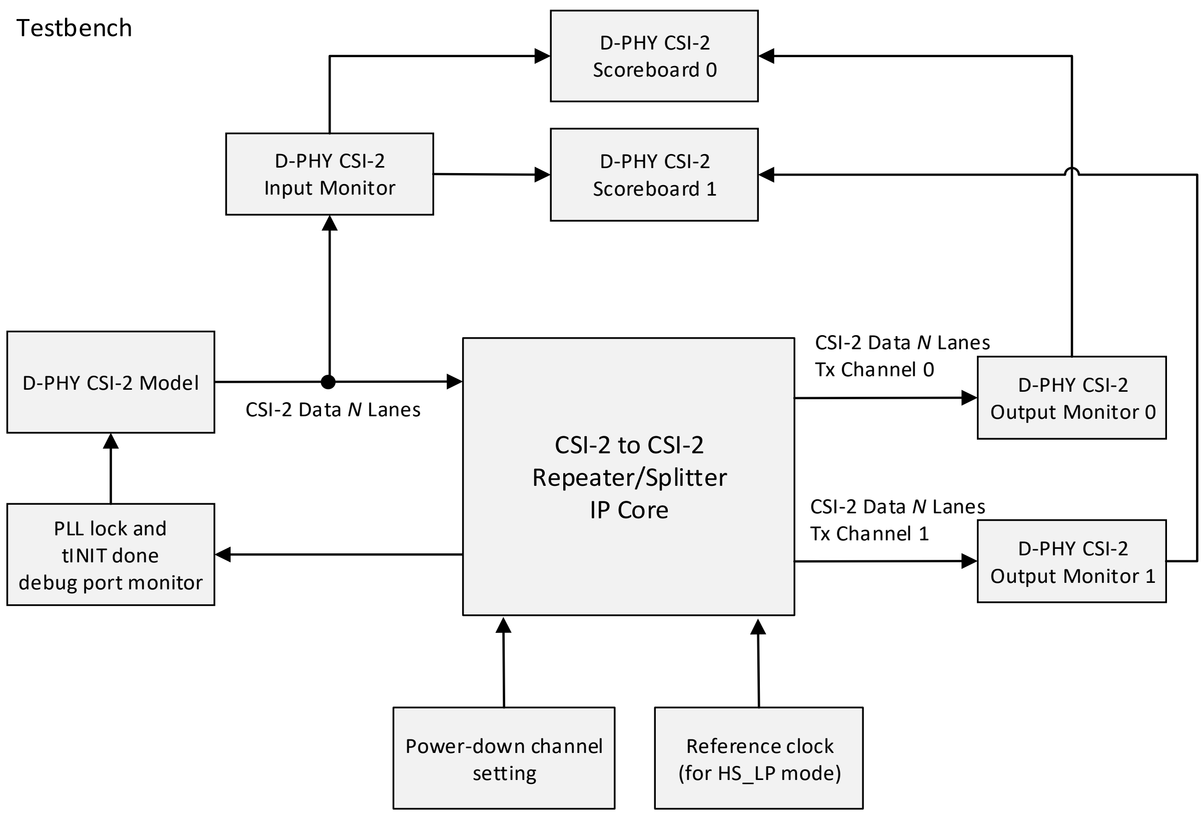


Figure 4.9. Simulation Environment Block Diagram

4.7. Simulation Environment

The simulation environment is made up of the D-PHY CSI-2 model instance connected to the IP core instance in the testbench. There are also D-PHY CSI-2 monitors connected to the input and outputs of the IP core. The monitors gather transaction information passing through the input and outputs of the IP core. The transaction information is then passed from the monitor to the scoreboard for data checking. The scoreboard only performs data checks (protocol/timing checks are not included).

The D-PHY CSI-2 model and monitors are configured based from the IP core configurations and testbench configurations. The testbench can also be configured to set the power-down channel setting for the IP core Tx channels. The testbench also transmits reference clock to the IP core if clock mode is non-continuous (HS_LP).

If miscellaneous signals such as PLL lock and tINIT done debug ports are included in the IP core, the testbench monitors assertion of these signals before sending the CSI-2 data to the IP core.

Figure 4.10. shows an example simulation where PLL lock debug ports are included.

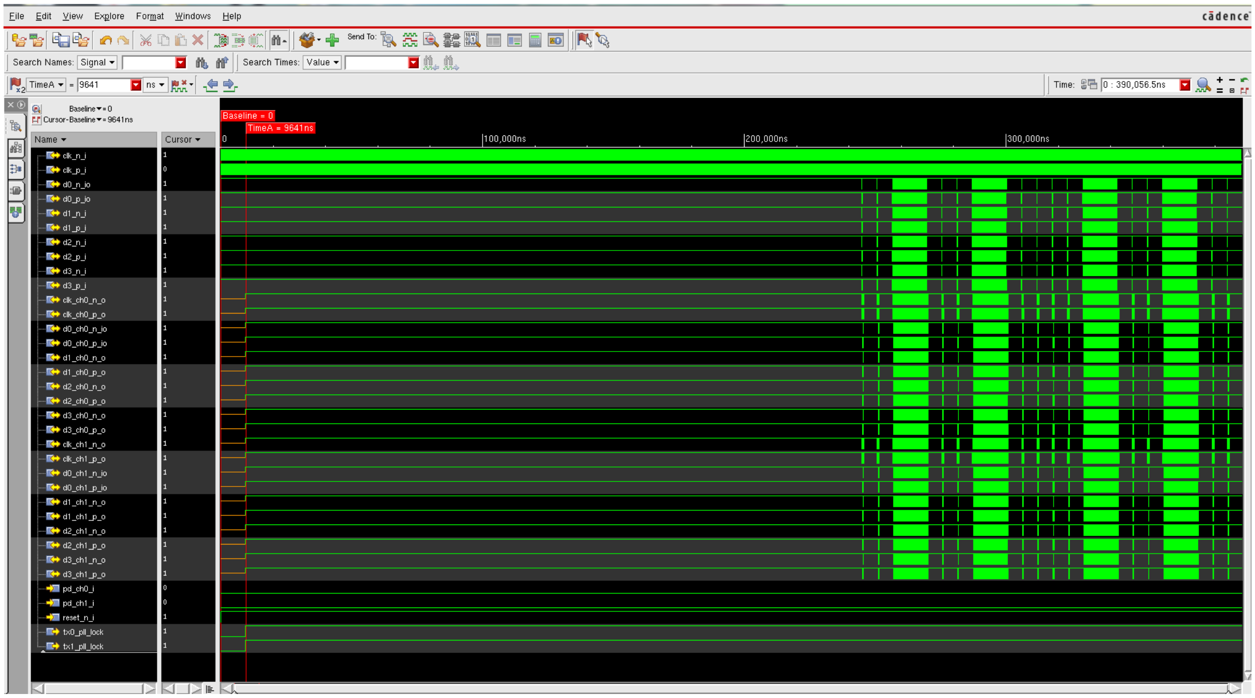


Figure 4.10. PLL Lock Miscellaneous Signals

The data transmitted by the D-PHY CSI-2 model can viewed in the waveform, see Figure 4.11:

- tb.csi2_ch0.data0 – Refers to the data bytes transmitted in D-PHY data lane 0
- tb.csi2_ch0.data1 – Refers to the data bytes transmitted in D-PHY data lane 1
- tb.csi2_ch0.data2 – Refers to the data bytes transmitted in D-PHY data lane 2
- tb.csi2_ch0.data3 – Refers to the data bytes transmitted in D-PHY data lane 3

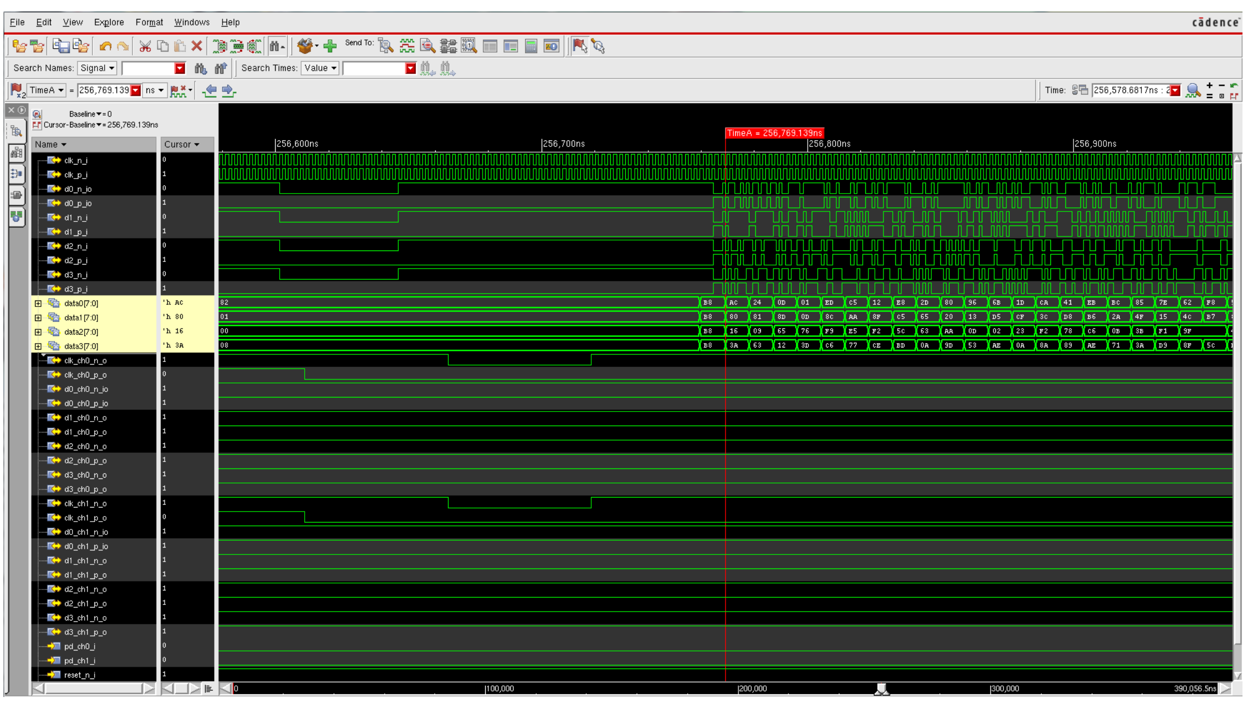


Figure 4.11. D-PHY CSI-2 Model Data

4.8. Instantiating the IP

The core modules of the MIPI CSI-2 to CSI-2 Camera Interface Bridge IP are synthesized and provided in NGO format with black box Verilog source files for synthesis. A Verilog source file named *<instance_name>_csi2_2_csi2_rs_ip.v* instantiates the black box of core modules. The top-level file *<instance_name>.v* instantiates *<instance_name>_csi2_2_csi2_rs_ip.v*.

The IP instances do not need to be instantiated one by one manually. The top-level file and the other Verilog source files are provided in *\<project_dir>*. These files are refreshed each time the IP is regenerated.

The MIPI CSI-2 to CSI-2 Camera Interface Bridge Soft IP is intended as a complete standalone solution. However, a Verilog instance template *<instancename>_inst.v* or VHDL instance template *<instancename>_inst.vhd* is also generated as a guide if the design is to be included in another top level module.

4.9. Synthesizing and Implementing the IP

In Clarity Designer, the Clarity Designer project file (.sbx) is added to Lattice Diamond as a source file after IP is generated. All required Verilog source files for implementation are invoked automatically. The IP can be directly synthesized, mapped and placed/routed in the Diamond design environment after the IP is generated. Note that the default Diamond strategy (.sty) and default Diamond preference file (.lpf) are used. When using the .sbx approach, import the recommended strategy and preferences from *\<project_dir>\csi2_to_csi2_rs_eval\<instancename>\impl\lifmd\lse* or *\<project_dir>\csi2_to_csi2_rs_eval\<instancename>\impl\lifmd\synplify* directories. All required files are invoked automatically. The design can be directly synthesized, mapped, placed and routed (PAR) in the Diamond design environment after the cores are generated.

Push-button implementation of this top-level design with either Lattice Synthesis Engine (LSE) or Synopsys Synplify Pro RTL synthesis is supported via the Diamond project file *<instancename>_top.ldf* located in *\<project_dir>\csi2_to_csi2_rs_eval\<instancename>\impl\lifmd\<synthesis_tool>* directory.

To use the pre-built Diamond project file:

1. Choose **File > Open > Project**.
2. In the **Open Project** dialog box browse to **\<project_dir>\csi2_to_csi2_rs_eval\<instancename>\impl\lifmd\<synthesis_tool>**.
3. Select and open **<instancename>_top.ldf.** At this point, all of the files needed to support top-level synthesis and implementation are imported to the project.
4. Select the **Process** tab in the left-hand user interface window.
5. Implement the complete design via the standard Diamond user interface flow.

4.10. Hardware Evaluation

The MIPI CSI-2 to CSI-2 Camera Interface Bridge IP supports Lattice's IP hardware evaluation capability, which makes it possible to create versions of the IP that operate in hardware for a limited period of time (approximately four hours) without requiring the request of an IP license. It may also be used to evaluate the core in hardware in user-defined designs.

4.10.1. Enabling Hardware Evaluation in Diamond

If using LSE, choose **Project > Active Strategy > LSE Settings**. If using Synplify Pro, choose **Project > Active Strategy > Translate Design Settings**. The hardware evaluation capability may be enabled or disabled in the **Strategy** dialog box. It is enabled by default.

4.11. Updating/Regenerating the IP

The Clarity Designer allows you to update the local IPs from the Lattice IP server. The updated IP can be used to regenerate the IP instance in the design. To change the parameters of the IP used in the design, the IP must also be regenerated.

4.11.1. Regenerating an IP in Clarity Designer

To regenerate IP in Clarity Designer:

1. In the **Builder** tab, right-click the IP instance to be regenerated and select **Config** in the menu as shown in Figure 4.12.

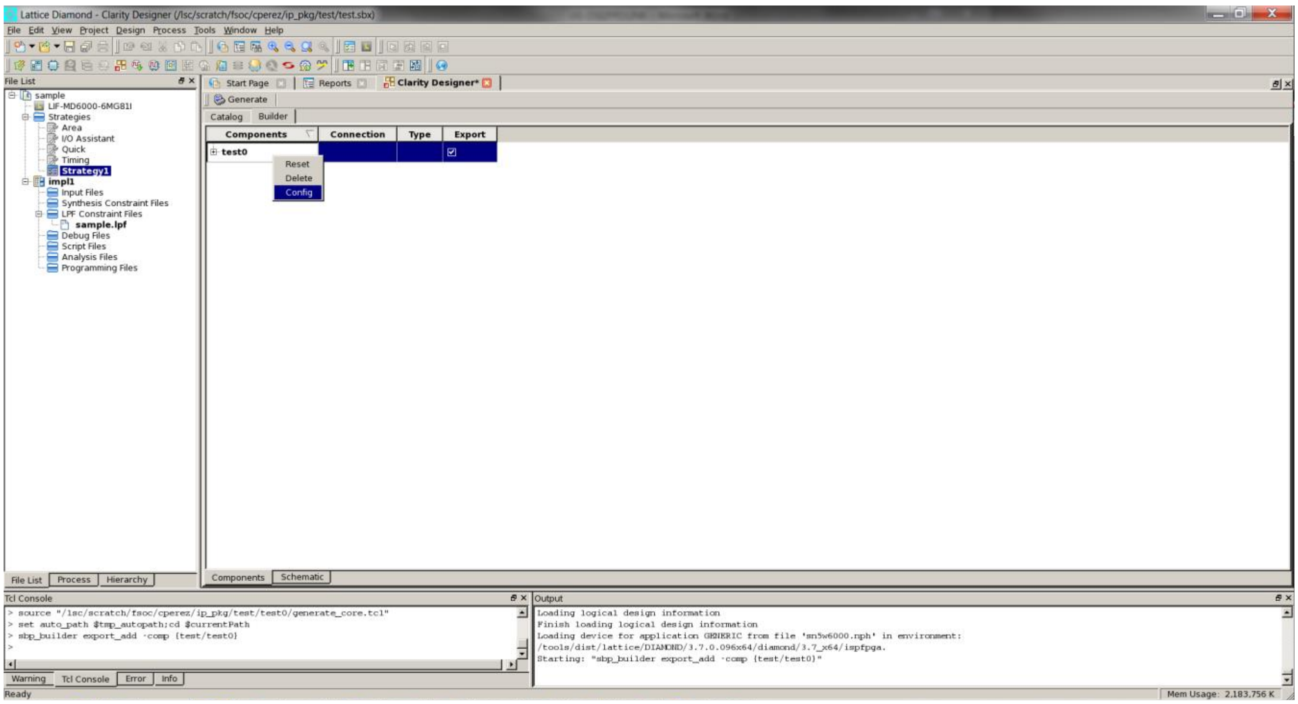
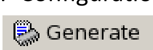


Figure 4.12. Regenerating IP in Clarity Designer

2. The IP Configuration user interface is displayed. Change the parameters as required and click the **Configure** button.
3. Click 

in the toolbox. Clarity Designer regenerates all the IP instances which are reconfigured.

References

For more information about CrossLink devices, refer to the CrossLink Family Data Sheet (FPGA-DS-02007).

For further information on interface standards, refer to:

- MIPI Alliance Specification for D-PHY, version 1.1, November 7, 2011, www.mipi.org
- MIPI Alliance Specification for Camera Serial Interface, version 1.1, November 22, 2011, www.mipi.org

Technical Support Assistance

Submit a technical support case through www.latticesemi.com/techsupport.

Appendix A. Resource Utilization

Table A.1 lists resource utilization for Lattice CrossLink FPGAs using the 1:2 MIPI CSI-2 Camera Interface Bridge IP. The performance and utilization data target an LIF-MD6000-6MG81I device with –6 speed grade using Lattice Diamond 3.9 and Lattice Synthesis Engine. Performance may vary when using a different software version or targeting a different device density or speed grade within the CrossLink family. The values of f_{MAX} shown are based on byte clock. The Target f_{MAX} column shows target byte clock frequency for each configuration.

Table A.1. Resource Utilization

IP User-Configurable Parameters	Slices	LUTs	Registers	sysMEM EBRs	Actual f_{MAX} (MHz)	Target f_{MAX} (MHz)
Continuous Rx Clock Mode, 4-Lane, Gear 16	1734	2013	1474	8	98.406	74.25
Continuous Rx Clock Mode, 4-Lane, Gear 8	1234	1514	1020	4	113.084	112.5
Non-continuous Rx Clock Mode, 4-Lane, Gear 16	1779	2085	1489	8	100.664	74.25
Non-continuous Rx Clock Mode, 4-Lane, Gear 8	1309	1598	1035	4	112.524	112.5

Appendix B. What is Not Supported

The IP does not support the following features:

- Cycling Redundancy Check (CRC) and Error Correction Code (ECC) checking and generation
- Bidirectional Communication
- Low-Level Protocol Error reporting
- Protocol Watchdog Timers

  The MIPI CSI-2 to CSI-2 Camera Interface Bridge IP has the following design limitations:
- Minimum duration of MIPI D-PHY low-power states (tLPX) should be at least two times the byte clock period
- Maximum byte clock frequency is 112.5 MHz, lower than maximum fabric speed due to heavy logic inside core modules

Revision History

Revision 1.1, IP Version 1.0, April 2019

Section	Change Summary
Introduction	Specified that this user guide can be used for IP design version 1.x.
IP Generation and Evaluation	In Licensing the IP, modified the instructions for requesting free license.
Revision History	Updated revision history table to new template.
All	Minor adjustments in style and formatting.

Revision 1.0, IP Version 1.0, February 2017

Section	Change Summary
All	Initial release

www.latticesemi.com