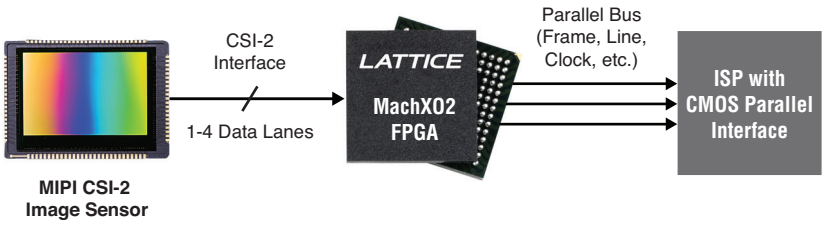
# MIPI CSI2-to-CMOS Parallel Sensor Bridge

**December 2016** **Reference Design RD1146**

## Introduction

The majority of image sensors in the consumer market use the MIPI CSI2 interface. The Mobile Industry Processor Interface (MIPI) has become the interface standard for the majority of components in consumer mobile devices. CSI2 (Camera Serial Interface 2) is the MIPI interface specification focused specifically on cameras. Because of the high volume and cost optimized design of these image sensors, many embedded application designers are interested in using these CSI2 image sensors. The challenge is that traditional ISPs (Image Signal Processors) do not have a CSI2 interface. Many have a traditional CMOS bus for camera interfaces. ISPs with a parallel CMOS interface bus must use an external bridge to convert from CSI2 to parallel CMOS. The Lattice MIPI CSI2-to-CMOS Parallel Sensor Bridge reference design performs this conversion in the ultra-low density MachXO2™-1200HC FPGA.

***Figure 1. CSI2 to CMOS Parallel Functional Block Diagram***



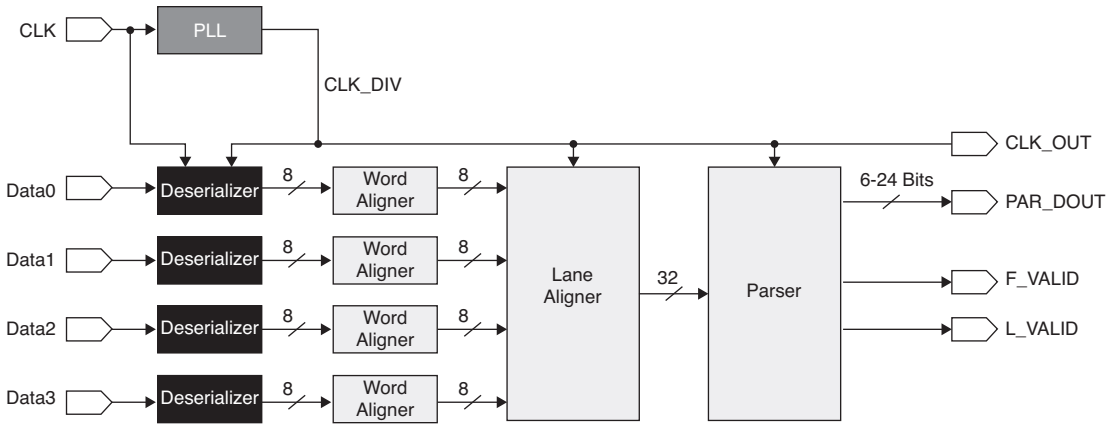

## Key Features

- Supports CSI2 high-speed differential signaling
  - — From one to four lanes, running at up to ~800 Mbps
  - — Supports 200 mV common mode voltage in High-Speed (HS) mode
- Provides parallel sensor output
  - — Output bus widths of 6 to 24 bits RAW, RGB and YUV
- Bridge device offered in space-saving 8x8mm 132-ball csBGA package. TQFP packages also available.
- Requires no external PROM
- Tested using the Sony IMX169 image sensor device on the Lattice CSI2 to Parallel Bridge Board and the HDR-60 Base Board
  - — 1080p30 2-lane and 4-lane configurations can be demonstrated
- Parallel interface can be configured for 1.8V, 2.5V or 3.3V LVCMOS levels
- MachXO2 is available in commercial and industrial temperature grades

## Functional Description

The CSI2 Bridge converts the CSI2 interface to a parallel sensor interface for an ISP. True LVDS input pads on the MachXO2 device handle the 200 mV common mode voltage of the MIPI DPHY high-speed interface. The CSI2 interface from the image sensor can be 1, 2 or 4 data lanes. To keep the FPGA density small, the CSI2 bridge is typically synthesized for a single CSI2 format. In most embedded applications the image sensor is typically configured for a single CSI2 output format at all times. However, multiple CSI2 formats can be supported for “on the fly” switching by adding multiple instantiations of the mipi_csi2_serial2parallel NGO in each desired format. This may increase the device density needed depending on the quantity and types of formats that must be supported. To request a CSI2 bridge meeting your specific design format, visit the CSI2 Bridge Request page on the Lattice website. See CSI2 Bridge Design Request Page section for details.

***Figure 2. CSI2 Bridge System Block Diagram***




The MIPI CSI2 to CMOS Parallel Sensor Bridge’s design modules follow the PHY and Protocol layer definitions described in the MIPI Alliance Specification for CSI2 Version 1.01.00.

*Note: It is advised to configure the lower clip value of the image sensor to 0x05 to ensure proper operation of the reference design.*

*The control capture module in the design may interpret the middle of the data payload as a sync code if byte values of 0x0 are transmitted. Using a lower clip value of 0x5 ensures that there will be a non-zero value for each byte of the payload during transmission of all data types.*

### PHY Layer

Each CSI2 serial data lane is sampled and converted into an 8-bit data bus using a dedicated IDDRx4 (Input Double Data Rate with x4 gearing) MachXO2 and MachXO3™ (specifically the L version of the family) fabric primitives. The word aligner module then performs the 8-bit word alignment so the SoT (Start of Transmission) leader sequence can easily be recognized on a single serial_clk/4-clock cycle. The Lane Aligner synchronizes the SoT leader sequences on each data lane. The result is an 8-bit output for each data lane that is synchronized by word and lane.

## Protocol Layer

The parser looks for the leader sequence on the first data lane. It then looks across the data lanes and extracts packet-specific information from the Packet Header. Information contained within the Packet Header includes the Virtual Channel Number (VC), Data Type (DT), Word Count (WC) and Error Correction Code (ECC). All four of these are optional outputs from the parser if they are desired for custom applications. The Data Type information is used to determine if the parser should utilize the data. The parser module can support any of the following data types:

- RAW6
- RAW7
- RAW8
- RAW10
- RAW12
- RAW14
- RGB444
- RGB555
- RGB565
- RGB666
- RGB888
- YUV420 8-bit legacy
- YUV420 8-bit
- YUV420 10-bit
- YUV420 8-bit (CSPS)
- YUV420 10-bit (CSPS)
- YUV422 8-bit
- YUV422 10-bit

If the data type value matches the format and pixel size indicated in the parameters format and bus_width, the parser will utilize the data. The Word Count indicator is also used to determine the length of data being transmitted. The parser utilizes the “Data Formats” section of the MIPI Alliance Specification for CSI2 for the appropriate output format and pixel size to unpackage the pixels from across the data lanes. The unpackaged pixels are stored in a FIFO where they are then read out one by one using the converted pixel clock from the PLL. The speed of the pixel clock is dependent on the number of data lanes and the pixel bus width. See Top Level Design Modules section for details.

## Reference Design Package

The MIPI CSI2-to-CMOS Parallel Sensor Bridge reference design package is available free of charge. The design package contains everything you need to get started.

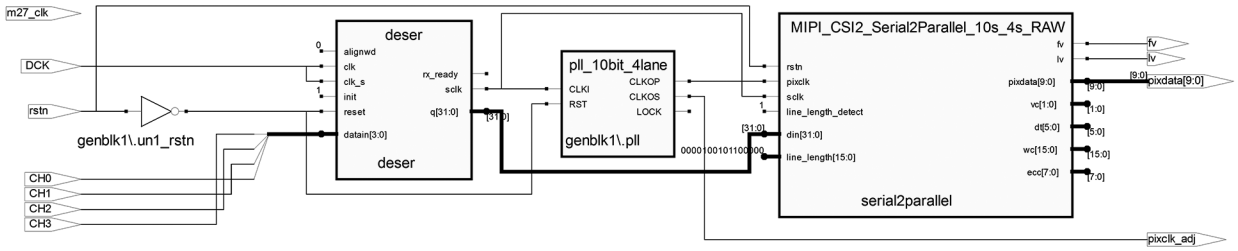
- ./doc/* – This document and Readme file
- ./source/* – Top evel design module, NGO black box modules and NGOs targeted to MachXO2 and MachXO3L devices for requested serial device mode. This folder also includes IPexpress™ modules for PLLs and IDDRx4 primitive wrappers.
- ./project/* – Lattice Diamond® 3.1 project targeted to MachXO2, MachXO3L and Bitstream for demo
- ./simulation/* – Aldec Active-HDL Simulation environment files, do files, simulation wizard script files and .vo files for simulation
- ./testbench/* – Verilog simulation test bench

## Top Level Design Modules

The top level design contains three modules:

- deser.v – IDDRx4 IPexpress primitive wrappers for each lane
- pll_*bit_*lane.ipx – PLL for conversion from serial_clk/4 to pixel clock
- mipi_csi2_serial2parallel.v – NGO wrapper module

***Figure 3. Top Level RTL Block Diagram***




The deser.v module takes in the serial clock and up to four data lanes. It converts the 1-bit DDR (double data rate) input on each data lane to an 8-bit SDR (single data rate) output at a reduced clock speed. This sclk clock is available at the output of the deser.v module along with the 8-bit * 4 data bus.

sclk = DCK/4

Gearing the data in the FPGA allows the design to internally run at a much slower clock rate, which increases performance, decreases power and decreases cost.

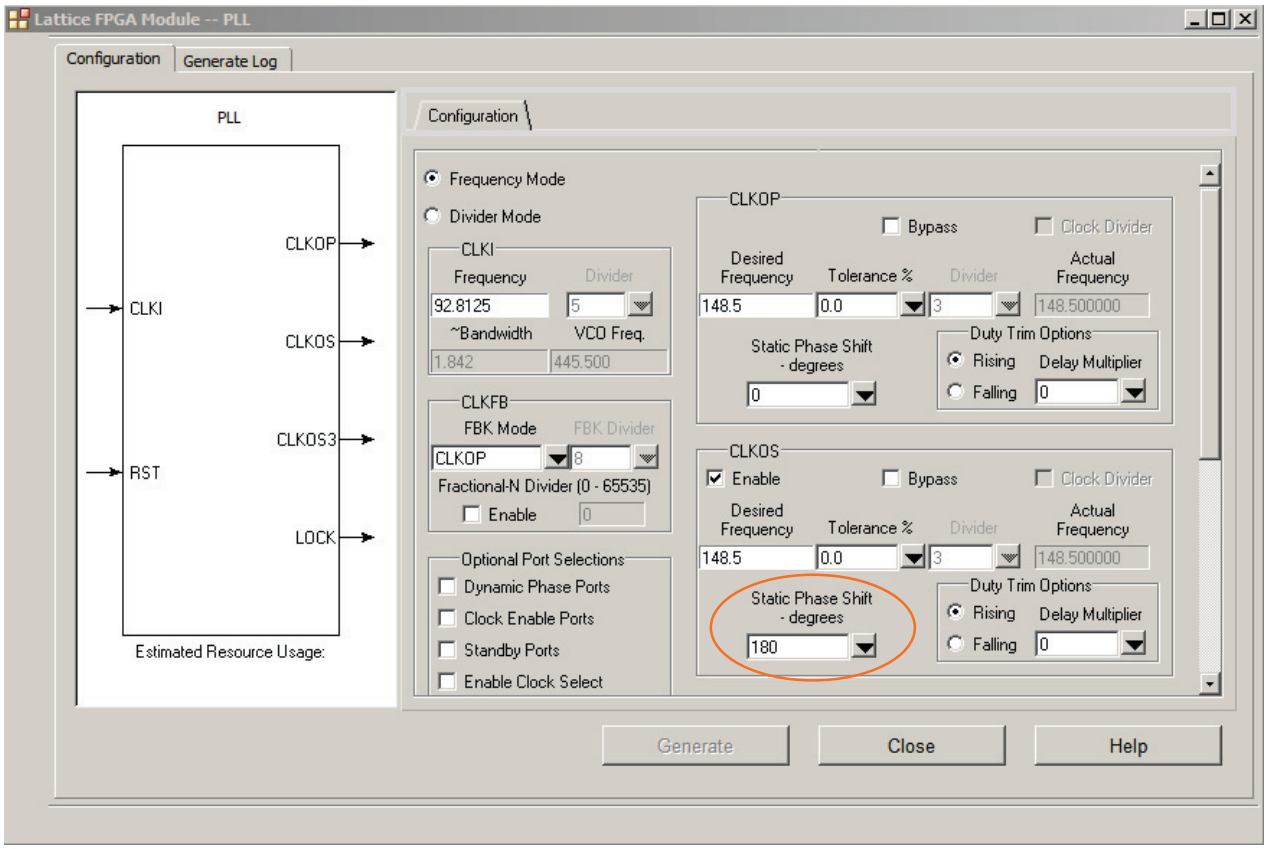
The pll_*bit_*lane.ipx module is a PLL primitive wrapper module generated in IPexpress. It is used to convert the serial_clk/4 clock to the pixel clock. The PLL ratio used for this conversion is dependent on the number of data lanes used and the pixel bus width.

pixclk = (1/bus_width) * #_lanes * 2 * 4 * sclk

The proper PLL ratio is provided in the reference design for the requested configuration. Pixclk drives the bridge modules internally in the FPGA. However, it is often desirable to have the output clock center aligned with the data. As a result, a second output from the PLL pixclk_adj drives the output pin at the same frequency as pixclk, but with a default 180-degree phase shift by default. This phase shift can be adjusted in the Diamond design software by

double-clicking the **pll_*bit_*lane.ipx** file. As seen in Figure 4, the phase can be adjusted within the encircled Static Phase shift - degrees drop-down box.

***Figure 4. pixclk_adj Phase Adjustment in IPexpress***




Parameters in the top level design module define the operation of the CSI2 bridge. After requesting a CSI2 bridge design, the bus_width, lane_width, and format parameters should remain unchanged. The line_length_detect and line_length parameters can be changed if the user desires to use the line_length_detect feature (see Important Design Considerations section).

***Table 1. Top Level NGO Descriptions***

| Parameter | Configurations | Description |
|---|---|---|
| bus_width | 6-24 | Width of pixel data. Only bus widths supported by CSI2 specification are included. |
| lane_width | 1, 2, 4 | Number of lanes to be used. |
| format | “RAW,” “RGB,” “YUV” | Output format type. |
| lp_mode | “ON,” “OFF” | Determines whether to base sclk, pixclk, and pixclk_adj off of DCK (the serial clock input) or sensor_clk (the sensor’s base clock, typically 27 MHz). See Important Design Considerations section for more details. |
| line_length_detect | 1 | Allows line valid to go active only for lines of size line_length. |
| line_length | 16’h0000 - 16’hFFFF | Expected line length to be recognized as a valid line from the word count (WC) in the CSI2 packet header (PH). Only valid if line_length_detect = 1. |

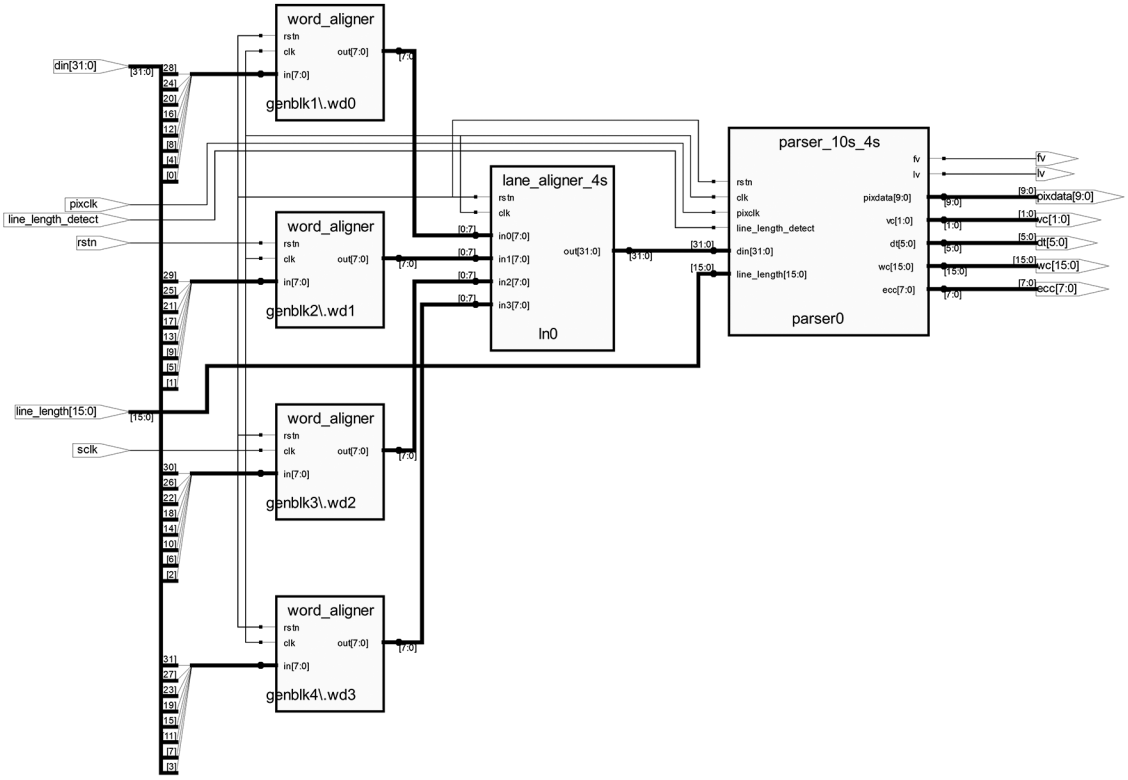
*Table 2. Top Level Design I/O*

| Signal | Direction | Description |
|---|---|---|
| Rstn | Input | Design reset |
| DCK | Input | Serial CSI2 input clock |
| CH0 | Input | Serial CSI2 data input |
| CH1 | Input | |
| CH2 | Input | |
| CH3 | Input | |
| sensor_clk | Input (optional) | Clock that drives the Image sensor. See Free Running Serial Clock in the Important Design Considerations section. |
| pixclk_adj | Output | Phase adjusted pixel clock |
| FV | Output | Frame valid |
| LV | Output | Line valid |
| pixdata[n-1:0] | Output | Pixel data; N = 6 to 24 bits. Pixels formatted MSB to LSB for RAW and YUV data types. RGB formatted as R, then G, then B from MSB to LSB respectively. |

## Reference Design NGO

The mipi_csi2_serial2parallel NGO contains the core of the bridge design. As described earlier, the word_aligner modules synchronize the 8-bit data words. The lane_aligner module synchronizes each data lane to the same clock cycle. The parser module takes the 8-bit data words across each of the lanes and parses the n width pixel bus based on the mode requested.

*Figure 5. NGO Block Diagram*



## Important Design Considerations

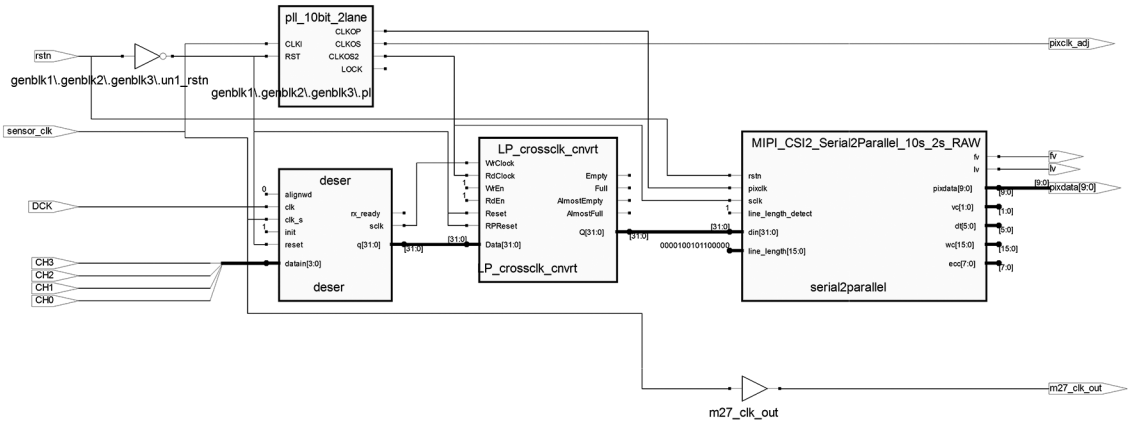
The majority of embedded designs that use CSI2 image sensors only utilize the data contained in the HS mode packets. To simplify the I/O on the input side of the MachXO2 device, the CSI2 to Parallel Sensor Bridge dismisses information received from the sensor in LP (Low Power) mode as it is not needed to convert video data available in HS (High Speed) mode. In fact, many image sensors do not utilize LP mode and often prefer HS short packets or additional long packets as a way to transfer additional information related to the video stream. Users should consult their image sensor's data sheet to determine the need to utilize LP mode. LP mode data can also be received by the MachXO2 device if desired, by using additional I/O. If it is determined that LP mode is necessary, Lattice suggests the following additional recommendations.

### Considerations for LP Mode on the CSI2 Clock Lane

In LP mode, the HS differential serial clock transitions to drive CMOS at 1.2V. As a result, the clocking inside the bridge stops as well as the pixclk_adj output. This may be problematic for some ISPs as they may require a pixel clock at all times. Most image sensors utilizing MIPI CSI2 either have the capability of placing the serial clock in a free running mode (always clocking) or turning off LP mode all together. It is recommended that designers utilize this feature for their design if possible. Users should consult their image sensor's data sheet to determine if their image sensor can run in free-running clock mode or turn off LP mode on all clock and data lanes.

If LP mode cannot be turned off, or if it is desired to utilize LP mode data on the clock lane, additional provisions must be made to the design. First, the image sensor's input clock (typically 27 MHz) must also run to the MachXO2. The lp_mode parameter must be switched to "ON." This will change the PLL; sclk, pixclk, and pixclk_adj will be based on the sensor's input clock rather than DCK/4 (the serial clock from the image sensor/4). In addition, a FIFO will be added to change the clock domain of the data coming in from the down-converted DCK/4 to the new sclk through the PLL. The result of these changes means the parallel bus output of the design will operate with a free-running clock. The serial clock from the image sensor will be used to clock HS data into the deserializer and FIFO. When in LP mode, the data on the clock lane can be used in the user's design for detection of mode changes or escape mode communication.

***Figure 6. Top-Level RTL Block Diagram with lp_mode = "ON"***



## Utilizing LP Mode Data with MachXO2 and MachXO3L

To utilize LP mode, traces must run to additional input pins on the MachXO2 device. A pin to determine the reference voltage for the LP pins also needs to be run to a regulator operating at ~0.55V. This pin needs to be on the same VCCIO bank as the LP input pins. Lastly, the LP input pins will need to be configured to HSTL18_I in the *.lpf file for the Diamond Project. Since image sensors between vendors can be different, it is recommended to choose a regulator that can have its voltage output adjusted via an external resistor network. Although the MachXO2 device has built-in termination in Bank 2, it is recommended to place external differential termination resistor footprints on the PCB for the serial clock and data lanes. Lattice recommends adding these resistor pads in case the differential voltage swing needs to be increased with a larger resistor value. This could occur if the operating voltage on the image sensor is low, which in turn causes the voltage driven from the sensor to not be within the CSI2 specification. The differential termination resistor pads should be put in the design but not populated on the board.

***Figure 7. Block Diagram of HS and LP Mode Board Design Provisions***

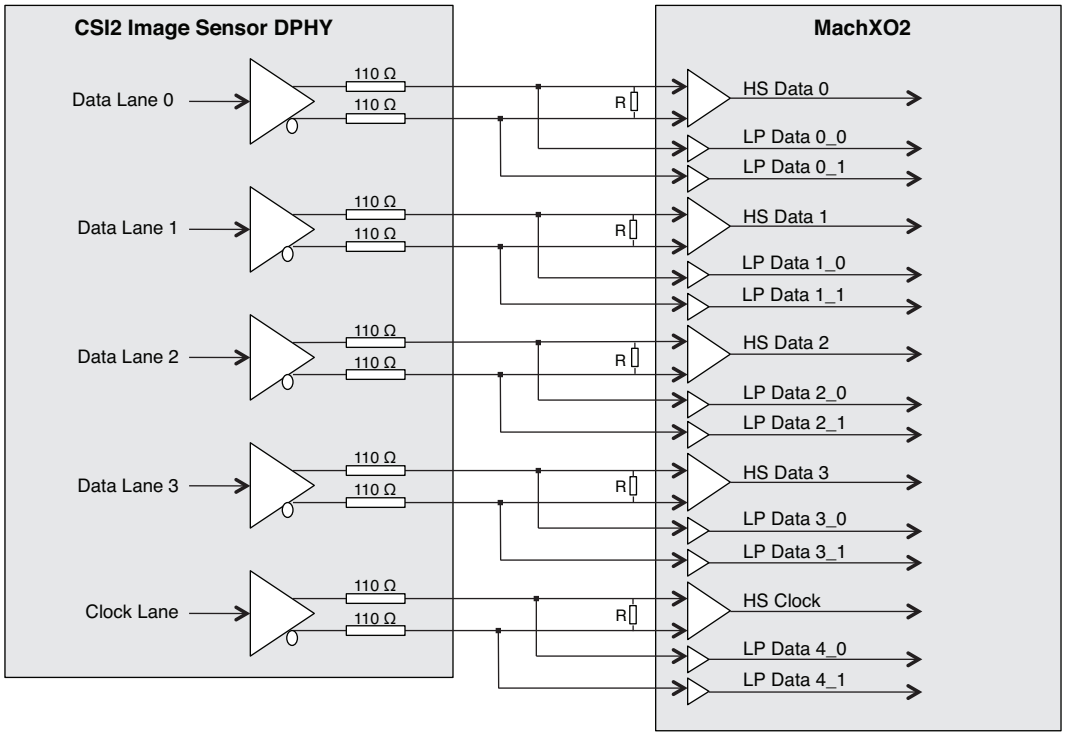
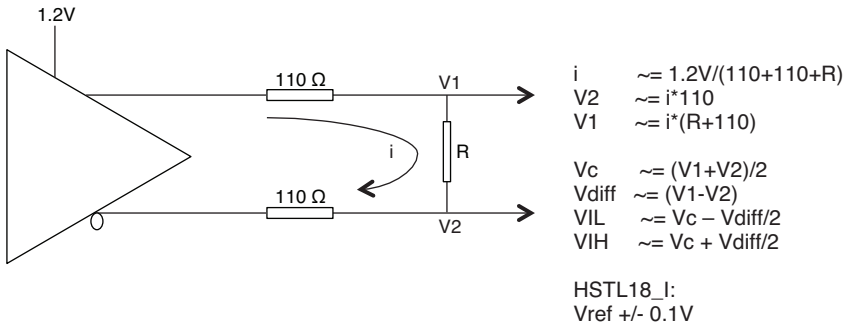



***Figure 8. Calculation of Termination Resistor when Utilizing LP Mode***




*Note: Equations for i, V1 and V2 are approximate as they do not include load of the MachXO2 input drivers, which varies by driver. As a result, the $V_C$ (common mode voltage) and $V_{DIFF}$ (differential voltage) may be slightly lower. If necessary, the $V_{DIFF}$ can be increased by the termination resistance value 'R'. $V_{REF}$ can be adjusted to place the CMOS threshold voltages in the center of $V_{IL}$ and $V_{IH}$.*

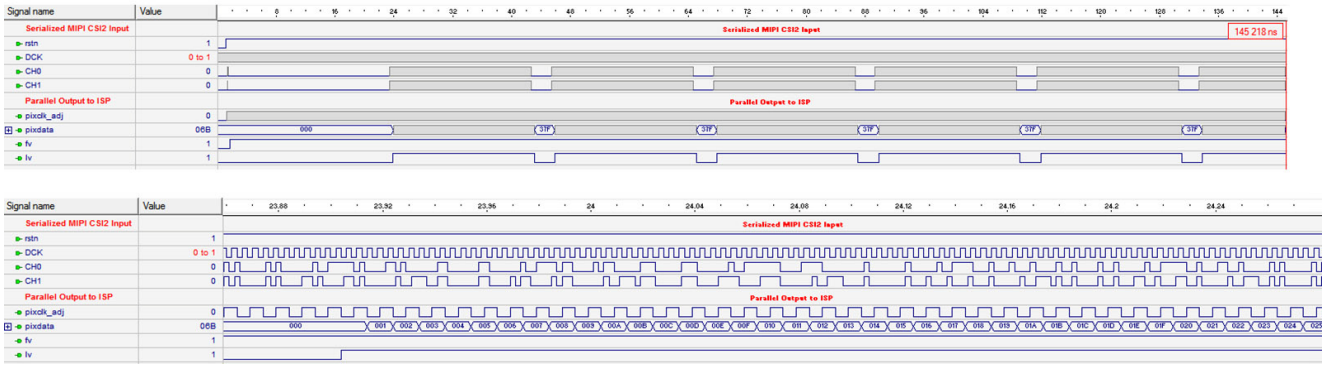
Example calculation using 100 Ohm termination:

| | | |
|---|---|---|
| I | = 1.2/(110+110+100) | = 3.75mA |
| V2 | = i*110 | = 0.4125V |
| V1 | = i*(110+100) | = 0.7875V |
| Vc | = (V1+V2)/2 | = 0.6V |
| Vdiff | = (V1-V2) | = 0.375V |

## Simulation

The included test bench tests the CSI2 bridge in 10-bit, 2-lane mode. The test bench should be used as a baseline to test other lane widths, pixel widths and output modes. The easiest way to run the simulation is through the Lattice Diamond design software and the pre-configured script (simulation_xxx.spf) provided in the *.ldf project. To do this, double-click on the script file under **Script Files** in the File List. Then click **Finish** in the Simulation Wizard. This will open the Active-HDL simulator. Compile the design, initialize the simulation, choose the signals you wish to view, run the simulation and move them to the Waveform Viewer. Follow the instructions for running a basic simulation in the simulator Help system if you are unfamiliar with the Active-HDL simulation environment.

***Figure 9. Simulation of CSI2 Input and Parallel Output with Included Test Bench***




## Reference Design Versus Demo Design

To simplify discrepancies between the Sony IMX169 demo design versus the standalone CSI2 reference design, both utilize the same project. The IMX169 demo design available on the Lattice website is defaulted to the demo. The reference design requested through the CSI2 sensor bridge request page defaults to the requested reference design.

Additional features in the demo design include:

- Pass-through signals for $I^2C$ from the HDR-60 Base Board
- Delayed XCLR signal for proper power-up of the Sony IMX169 Sensor
- Pass-through of the 27 MHz clock on the MachXO2 Dual Sensor Interface Board for synchronized clocks with LP mode
- LP inputs on Data Lane 0 are passed through to LED1 and LED2 on the MachXO2 Dual Sensor Interface Board

To switch to the IMX169 demo design, follow these steps:

- In the Lattice Diamond File List, under Input Files:
  — Right-click on the file **../source/mipi_csi2_serial2parallel_bridge.v**
  — Choose **Exclude from Implementation**
- In the Lattice Diamond File List, under Input Files:
  — Right-click on the file **../source/mipi_csi2_serial2parallel_bridge_demo.v**
  — Choose **Include in Implemention**
- In the Lattice Diamond File List, under Input Files:
  — Right-click on the file **../source/mipi_csi2_serial2parallel_bridge_demo.lpf**
  — Choose **Set as Active Preference File**
- In the Lattice Diamond File List:
  — Double-click on **LCMXO2-1200HC-6MG132C**
  — In Part Names, choose **LCMXO2-4000HE-6MG132C**

To utilize the reference design, removing all demo-specific features, reverse these steps by excluding the mipi_csi2_serial2parallel_bridge_demo.v and including mipi_csi2_serial2parallel_bridge.v. Set mipi_csi2_serial2parallel_bridge.lpf as the active preference file and switch the device from LCMXO2-4000HE-6MG132C to the device you wish to use.

## Recommended MachXO2 Pinout

*Table 3. Recommended MachXO2 Pinout*

| Signal | MachXO2 132-Ball csBGA, Speed Grade -6 |
|---|---|
| rstn | C1 |
| DCK_p | N6 |
| DCK_n | P6 |
| CH0_p | M11 |
| CH0_n | P12 |
| CH1_p | P8 |
| CH1_n | M8 |
| CH2_p | P2 |
| CH2_n | N2 |
| CH3_p | M7 |
| CH3_n | N8 |
| DCK_LP_p | N3 |
| DCK_LP_n | P4 |
| CH0_LP_p | N5 |
| CH0_LP_n | M5 |
| CH1_LP_p | P7 |
| CH1_LP_n | N7 |
| CH2_LP_p | P9 |
| CH2_LP_n | N9 |
| CH3_LP_p | M9 |
| CH3_LP_n | N10 |
| sensor_clk | C8 |
| pixclk_adj | A11 |
| FV | B7 |
| LV | C4 |
| pixdata_0 | C6 |
| pixdata_1 | B3 |
| pixdata_2 | C11 |
| pixdata_3 | A12 |
| pixdata_4 | A7 |
| pixdata_5 | B5 |
| pixdata_6 | A9 |
| pixdata_7 | A10 |
| pixdata_8 | A2 |
| pixdata_9 | B12 |
| pixdata_10 | C12 |
| pixdata_11 | B13 |
| pixdata_12 | B9 |
| pixdata_13 | A13 |

## Design Utilization and Timing

***Table 4. Design Performance***

| Device Family | Synthesis Engine | Configuration | Speed Grade - 4 (MHz) | | Speed Grade - 5 (MHz) | | Speed Grade - 6 (MHz) | |
|---|---|---|---|---|---|---|---|---|
| | | | sclk | pixclk | sclk | pixclk | sclk | pixclk |
| MachXO2 | LSE | 10-bit, 1-Lane | 63.865 | 150.01 | 73.708 | 164.690 | 78.198 | 182.548 |
| | | 10-bit, 2-Lane | 76.249 | 150.015 | 86.655 | 164.690 | 102.124 | 182.548 |
| | | 10-bit, 4-Lane | 63.865 | 150.01 | 75.763 | 164.690 | 77.791 | 182.548 |
| | Synplify Pro | 10-bit, 1-Lane | 94.832 | 110.939 | 164.69 | 102.501 | 105.208 | 182.548 |
| | | 10-bit, 2-Lane | 78.74 | 87.436 | 164.69 | 86.296 | 99.880 | 182.548 |
| | | 10-bit, 4-Lane | 80.978 | 77.954 | 164.69 | 80.88 | 81.934 | 182.548 |

| Device Family | Synthesis Engine | Configuration | Speed Grade - 5 (MHz) | | Speed Grade - 6 (MHz) | |
|---|---|---|---|---|---|---|
| MachXO3L | LSE | 10-bit, 2-Lane | 75.672 | 150.015 | 102.955 | 182.548 |
| | Synplify Pro® | | 83.126 | 150.015 | 95.193 | 182.548 |

| Device Family | Synthesis Engine | Speed Grade - 6 (MHz) | | Speed Grade - 7 (MHz) | | Speed Grade - 8 (MHz) | |
|---|---|---|---|---|---|---|---|
| | | sclk | pixclk | sclk | pixclk | sclk | pixclk |
| ECP5™ | LSE | 61.155 | 226.706 | 54.20 | 313.873 | 89.158 | 347.826 |
| | Synplify Pro | 52.565 | 239.063 | 61.095 | 276.625 | 82.781 | 345.185 |

***Table 5. I/O Timing Analysis of LVDS Input Bus***

| Device Family | Speed Grade - 4 (MHz) | | Speed Grade - 5 (MHz) | | Speed Grade - 6 (MHz) | |
|---|---|---|---|---|---|---|
| MachXO2 | Setup | Hold | Setup | Hold | Setup | Hold |
| | 0.222 | 0.337 | 0.222 | 0.254 | 0.222 | 0.183 |

| Device Family | Speed Grade - 5 (MHz) | | Speed Grade - 6 (MHz) | |
|---|---|---|---|---|
| MachXO3L | Setup | Hold | Setup | Hold |
| | 0.248 | 0.292 | 0.243 | 0.158 |

| Device Family | Speed Grade - 6 (MHz) | | Speed Grade - 7 (MHz) | | Speed Grade - 8 (MHz) | |
|---|---|---|---|---|---|---|
| ECP5 | Setup | Hold | Setup | Hold | Setup | Hold |
| | 0.224 | 0.051 | 0.145 | 0.051 | 0.055 | 0.051 |

*Table 6. Resource Utilization*

| Device Family | Synthesis Engine | Configuration | Registers | LUTs | EBRs | PLL | Gearbox | Clock Divider |
|---|---|---|---|---|---|---|---|---|
| MachXO2 [1] | LSE | 1 Data Lane (LP+HS) | 348 | 225 | 3 | 1 | 1 | 1 |
| | | 2 Data Lanes (LP+HS) | 626 | 436 | 5 | 1 | 2 | 1 |
| | | 4 Data Lanes (LP+HS) | 873 | 879 | 3 | 1 | 4 | 1 |
| | Synplify Pro | 1 Data Lane (LP+HS) | 365 | 279 | 3 | 1 | 1 | 1 |
| | | 2 Data Lanes (LP+HS) | 626 | 436 | 5 | 1 | 2 | 1 |
| | | 4 Data Lanes (LP+HS) | 877 | 828 | 3 | 1 | 4 | 1 |
| MachXO3L [2] | LSE | 10-bit, 2-Lane | 626 | 436 | 5 | 1 | 2 | 1 |
| | Synplify Pro | | 626 | 436 | 5 | 1 | 2 | 1 |
| ECP5 [3] | LSE | 10-bit, 4-Lane | 1210 | 1120 | 7 | 1 | 2 | 1 |
| | Synplify Pro | | 1209 | 1113 | 7 | 1 | 2 | 1 |

1. Performance and utilization characteristics are generated using LCMXO2-1200HC-4MG132C with Lattice Diamond 3.3 design software. When using this design in a different device, density, speed, or grade, performance and utilization may vary.
2. Performance and utilization characteristics are generated using LCMXO3L-4300C -6BG256C with Lattice Diamond 3.3 design software. When using this design in a different device, density, speed, or grade, performance and utilization may vary.
3. Performance and utilization characteristics are generated using LFE5U-85F-6MG285IES with Lattice Diamond 3.3 design soft ware. When using this design in a different device, density, speed, or grade, performance and utilization may vary.

## CSI2 Bridge Design Request Page

The CSI2 to parallel reference design is defaulted to support 10-bit raw output image data. This is because the Sony IMX169 is a 10-bit image sensor. If you are using a different image sensor or require a different mode, you can request a specific CSI2 bridge design on the Lattice website at www.latticesemi.com/csi2bridge.

## CSI2 Bridge Development Platform

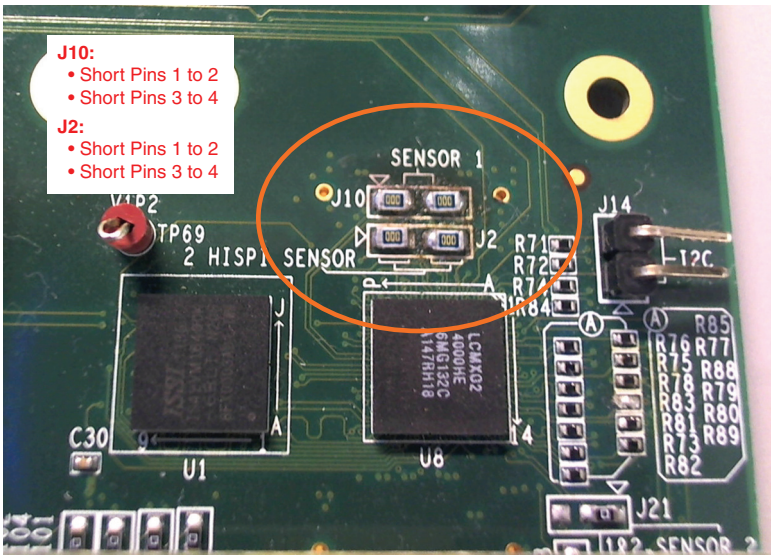
The CSI2 bridge design can be demonstrated using the following development boards:

- HDR-60 Video Camera Development Kit (part number: LFE3-70EA-HDR60-DKN)
- MachXO2 Dual Sensor Interface Board (part number: LCMXO2–4000HE–DSIB-EVN)
- CSI2 to Parallel Bridge Board (part number: LF-C2P-EVN)

A demonstration bitstream and reference design are available at www.latticesemi.com/csi2bridge that target the CSI2 to Parallel Bridge in 10-bit, 2-lane mode. The CSI2 to Parallel Bridge outputs a 1080p30 frame at 74.25 MHz. The HDR-60 Base Board uses a frame doubler to increase the frame rate to 1080p60. It then uses a simple debayer and SERDES to output the image via a DVI interface.

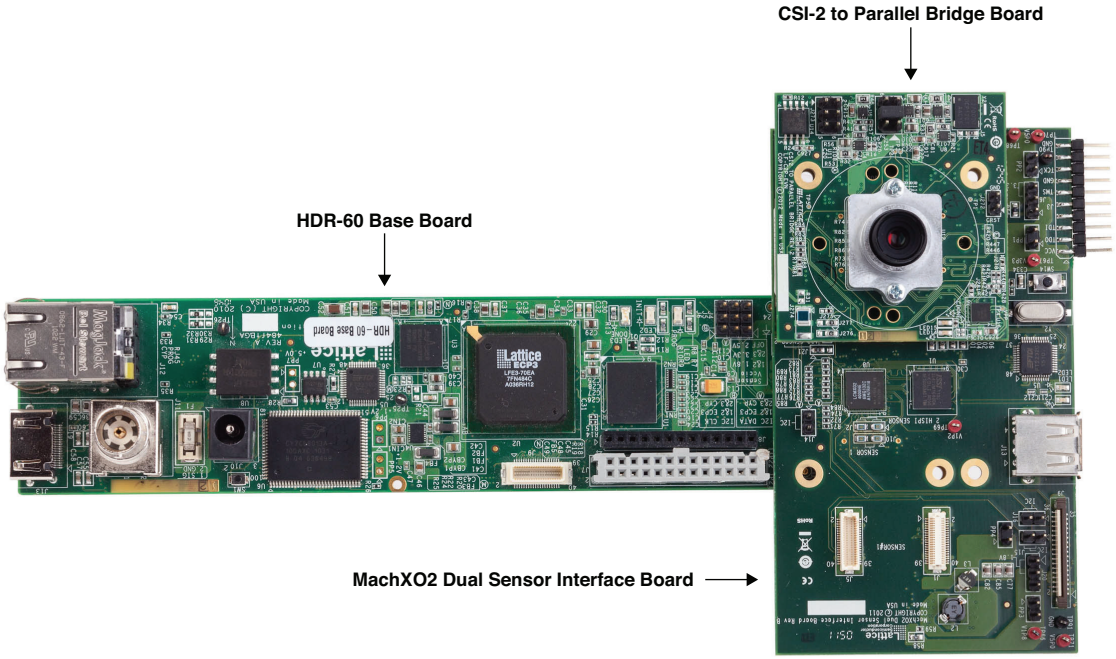
**To set up the demonstration platform two resistor modifications need to be performed to the MachXO2 Dual Sensor Interface Board from the default configuration.** J10 and J2 contain one 0 ohm resistor between pads 2 to 3 by default. This configuration needs to be changed to two resistors between pads 1 to 2 and pads 3 to 4 for both J10 and J2. On the CSI2 to Parallel Bridge Board, a jumper must be placed on J253 between pins 3 and 4 (see Figure 12). This jumper is populated by default on the board when purchased.

*Figure 10. Resistor Modifications to MachXO2 Dual Sensor Interface Board*




Each board connects together through the white Hirose connectors on each board. **Please note that the CSI2 to Parallel Bridge Board connects into sensor location 2 on the MachXO2 Dual Sensor Interface Board for the design demonstration** (see Figure 11).

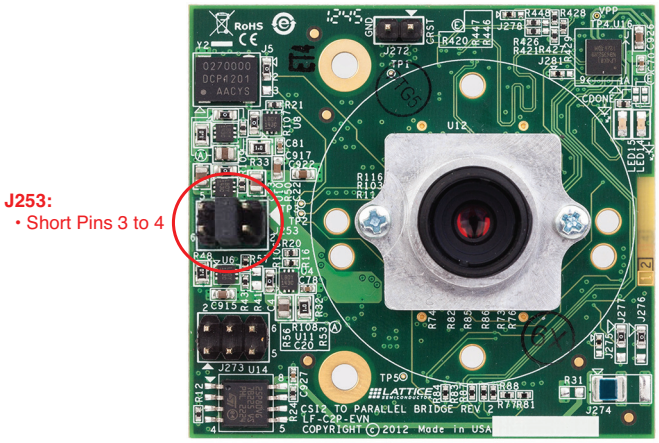
*Figure 11. Lattice MIPI CSI2 Development Setup*



## Tested Hardware

The CSI2 to Parallel Bridge Board contains the Sony IMX169 CSI2 sensor.

***Figure 12. CSI2 to Parallel Bridge Board with Sony IMX169 CSI2 Sensor***




## References

- MIPI Alliance Specification for Camera Serial Interface 2 (CSI2) V1.01
- MIPI Alliance Specification for D-PHY V1.1
- Sony IMX169CQK-C Data Sheet

## Technical Support Assistance

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

## Revision History

| Date | Version | Change Summary |
|---|---|---|
| December 2016 | 1.5 | Updated Added note to Functional Description section. Added note on configuring the lower clip value of the image sensor. |
| | | Updated Technical Support Assistance section. |
| January 2015 | 1.4 | Updated PHY Layer section. Product name/trademark adjustment. |
| | | Updated Design Utilization and Timing section.<br>Revised Table 4, Design Performance.<br>— Added support for ECP5 device family.<br>— Added values for LSE and Synplify Pro.<br>Revised Table 5, I/O Timing Analysis of LVDS Input Bus.<br>— Changed table title.<br>— Added support for ECP5 device family.<br>Revised Table 6, Resource Utilization.<br>— Added support for ECP5 device family.<br>— Added values for LSE and Synplify Pro.<br>— Added footnotes. |
| April 2014 | 01.3 | Updated Key Features section. Revised CSI2 high speed differential signaling. |
| | | Updated the following sections to reflect revised file names and folder structure based on RD guidelines.<br>— Functional Description<br>— Reference Design Package<br>— Top Level Design Modules<br>— Reference Design NGO<br>— Simulation<br>— Reference Design Versus Demo Design |
| | | Updated the following sections to add support for MachXO3L device family:<br>— PHY Layer<br>— Utilizing LP Mode Data with MachXO2 and MachXO3L |
| | | Updated Reference Design Package section. Updated folder structure according to the RD guidelines. |
| | | Updated the following tables to add support for MachXO3L device family:<br>— Table 4, Design Performance<br>— Table 5, I/O Timing Analysis of Sub-LVDS Parallel Input Bus<br>— Table 6, Resource Utilization |
| | | Updated Technical Support Assistance section. |
| December 2013 | 01.2 | Updated pixdata[n-1:0] signal description in Top Level Design I/O table. |
| December 2012 | 01.1 | CSI2 Bridge Development Platform section – Added information about two resistor modifications needed to perform default configuration of the MachXO2 Dual Sensor Interface Board. |
| | | Updated CSI2 to Parallel Bridge with Sony IMX169 CSI2 Sensor photo. |
| | | Added callouts to Lattice MIPI CSI2 Development Setup photo. |
| | 01.0 | Initial release. |