SYNOPSYS®

DesignWare Cores MIPI CSI-2 Host Controller

Databook

DWC_mipi_csi2_host

Version 1.02a
May 2011

Copyright Notice and Proprietary Information

Destination Control Statement

All technical data contained in this publication is subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the reader's responsibility to determine the applicable regulations and to comply with them.

Disclaimer

SYNOPSYS, INC., AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Registered Trademarks (®)

Synopsys, AMPS, Astro, Behavior Extracting Synthesis Technology, Cadabra, CATS, CRITIC, Certify, CHIPit, Design Compiler, DesignWare, Formality, HDL Analyst, HSIM, HSPICE, Identify, Leda, MAST, ModelTools, NanoSim, OpenVera, PathMill, Physical Compiler, PrimeTime, SCOPE, Simply Better Results, SiVL, SNUG, SolvNet, Syndicated, Synplicity, the Synplicity Logo, Synplify, Synplify Pro, Synthesis Constraints Optimization Environment, TetraMAX, UMRBus, VCS, Vera, and YIELDirector are registered trademarks of Synopsys, Inc.

Trademarks (™)

AFGen, Apollo, Astro, Astro-Rail, Astro-Xtalk, Aurora, AvanWaves, BEST, Columbia, Columbia-CE, Confirma, Cosmos, CosmosLE, CosmosScope, CRITIC, CustomExplorer, CustomSim, DC Expert, DC Professional, DC Ultra, Design Analyzer, Design Vision, DesignerHDL, DesignPower, DFTMAX, Direct Silicon Access, Discovery, Eclypse, Encore, EPIC, Galaxy, Galaxy Custom Designer, HANEX, HAPS, HapsTrak, HDL Compiler, Hercules, Hierarchical Optimization Technology, High-performance ASIC Prototyping Ssytem, HSIMplus, i-Virtual Stepper, IICE, in-Sync, iN-Tandem, Jupiter, Jupiter-DP, JupiterXT, JupiterXT-ASIC, Liberty, Libra-Passport, Library Compiler, Magellan, Mars, Mars-Rail, Mars-Xtalk, Milkyway, ModelSource, Module Compiler, MultiPoint, Physical Analyst, Planet, Planet-PL, Polaris, Power Compiler, Raphael, Saturn, Scirocco, Scirocco-i, Star-RCXT, Star-SimXT, StarRC, Taurus, TotalRecall, TSUPREM-4, VCS Express, VCSi, VHDL Compiler, VirSim, and VMC are trademarks of Synopsys, Inc.

Service Marks (SM)

MAP-in, SVP Café, and TAP-in are service marks of Synopsys, Inc.

SystemC is a trademark of the Open SystemC Initiative and is used under license.

ARM and AMBA are registered trademarks of ARM Limited.

Saber is a registered trademark of SabreMark Limited Partnership and is used under license.

PCI Express is a trademark of PCI-SIG.

All other product or company names may be trademarks of their respective owners.

Synopsys, Inc.
700 E. Middlefield Road
Mountain View, CA 94043

www.synopsys.com

Contents

Revision History

The following table provides the history of changes to this databook.

Date	Version	Description
March 2010	1.00a	Initial release
November 2010	1.01a	Updated the configuration parameters, signals, and registers.
May 2011	1.02a	• Minimum frequency of RXBYTECLKHS is changed to 10 in Table 1-1. • A table showing errors that can be identified by the CSI-2 Host IP is added to section 2.4 Error Detection. • Section 3.1 Image Data Interface is updated.

Preface

This databook describes the DesignWare Cores MIPI CSI-2 Host Controller, which along with Synopsys DWC HDMI TX PHYis a part of a complete MIPI CSI-2 interface solution.

Throughout this databook, MIPI CSI-2 is used to reference the DWC_mipi_csi2_host controller.

Databook Organization

The chapters of this databook are organized as follows:

- "Product Overview" provides an introduction to the MIPI CSI-2 core, including a block diagram, supported features, deliverables, supported standards, and so on.
- "Architecture Overview" describes the MIPI CSI-2 core's general architecture, startup sequence, and interrupt mechanism.
- "Timing Interfaces" provides information on timing and data format of the image data output of MIPI CSI-2 .
- "Hardware Configuration Parameters" describes the hardware configuration parameters.
- "Signals" provides descriptions of the MIPI CSI-2 's inputs/outputs.
- "Software Registers" provides the memory map of the MIPI CSI-2 and descriptions of the programmable software registers.

Related Documentation

Refer to the following documentation:

- *coreConsultant User's Guide*
- *coreAssembler User's Guide*

Web Resources

The following web links are various Synopsys online resources you may find useful:

- DesignWare IP product information: http://www.designware.com
- Your custom DesignWare IP page: http://www.mydesignware.com
- Documentation through SolvNet: http://solvnet.synopsys.com (Solvnet ID required)
- Synopsys Common Licensing (SCL): http://www.synopsys.com/keys

Customer Support

To obtain support for your product, choose one of the following:

- First, prepare the following debug information, if applicable:
  - For environment setup problems or failures with configuration, simulation, or synthesis that occur within coreConsultant or coreAssembler, use the following menu entry:

    File > Build Debug Tar-file

    Check all the boxes in the dialog box that apply to your issue. This menu entry gathers all the Synopsys product data needed to begin debugging an issue and writes it to the file <*core tool startup directory*>/debug.tar.gz.
  - For simulation issues outside of coreConsultant or coreAssembler:
    - Create a waveforms file (such as VPD or VCD)
    - Identify the hierarchy path to the DesignWare instance
    - Identify the timestamp of any signals or locations in the waveforms that are not understood
- Then, contact Support Center, with a description of your question and supplying the above information, using one of the following methods:
  - *For fastest response*, use the SolvNet website. If you fill in your information as explained below, your issue is automatically routed to a support engineer who is experienced with your product. The **Sub Product** entry is critical for correct routing.

    Go to http://solvnet.synopsys.com/support/open_case.action.
    Provide the requested information, including:
    - Product: DesignWare Cores
    - Sub Product: MIPI Controller
    - Version: 1.02a
    - Problem Type:
    - Priority:
    - Title: <insert dwc_mipi_csi2_host>
    - Description: For simulation issues, include the timestamp of any signals or locations in waveforms that are not understood

    After creating the case, attach any debug files you created in the previous step.
  - Or, send an e-mail message to support_center@synopsys.com (your e-mail will be queued and then, on a first-come, first-served basis, manually routed to the correct support engineer):
    - Include the Product name, Sub Product name, and Tool Version number in your e-mail (as identified above) so it can be routed correctly.
    - For simulation issues, include the timestamp of any signals or locations in waveforms that are not understood
    - Attach any debug files you created in the previous step.
  - or, telephone your local support center:
    - North America:

      Call 1-800-245-8005 from 7 AM to 5:30 PM Pacific time, Monday through Friday.
    - All other countries:

      http://www.synopsys.com/Support/GlobalSupportCenters

1 Product Overview

This chapter includes the following topics:

1.1 General Product Description

The DWC_mipi_csi2_host implements CSI-2 protocol. The CSI-2 link protocol specification is a part of group of communication protocols defined by MIPI Alliance standards intended for mobile system chip-to-chip communications. The CSI-2 specification is specifically targeted for Camera to Image application processor communication.

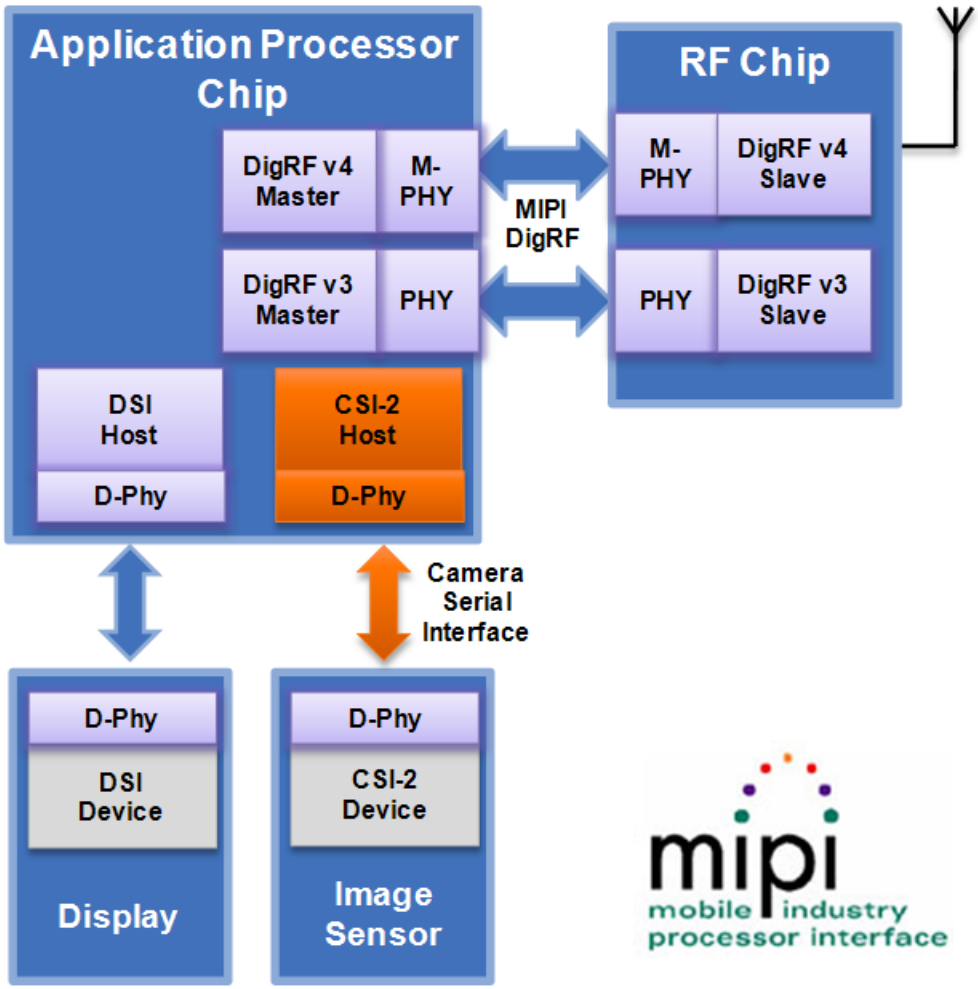
DesignWare MIPI CSI-2 Host Controller is used for the reception of data from a CSI-2 compliant camera sensor. A D-PHY configured as Slave ensures the physical layer.

The DesignWare Cores MIPI CSI-2 Host Controller includes the DWC_mipi_csi2_host design and its verification environment. Synopsys also provides the coreConsultant tool for automated configuration, simulation, and synthesis of the core.

The DWC_mipi_csi2_host is designed to integrate a Synopsys MIPI D-PHY. There is a broad range of D-PHY that include bifurcational or slave only PHYs with 2 and 4 Lanes for several technologies. For more information on MIPI D-PHY, visit Synopsys DesignWare MIPI D-PHY IP Solution page.

Figure 1-1 shows the DWC_mipi_csi2_host in an example system on chip design.

Figure 1-1 DWC_mipi_csi2_host in System on Chip Example



1.1.1 Applications

Typical applications built with the DWC_mipi_csi2_host Controller core are Mobile SoC, Application processors, and co-processors, targeting:

- Handheld devices
- Smartphone
- Multimedia tablets
- MID
- Navigation
- DSC
- DVC

1.1.2 Standards Compliance

MIPI CSI-2 conforms to the following standards:

- MIPI Alliance Standard for Camera Serial Interface 2 (CSI-2), Version 1.00 - 29 November 2005
- MIPI Alliance Specification for D-PHY, Version 1.00.00 - 14 May 2009
- AMBA 2.0 Specification (APB) from ARM

1.1.3 Unsupported Features and Exceptions

The following feature is not supported in this version of DWC_mipi_csi2_host:

DWC_mipi_csi2_host does not directly include support for Camera Control Interface (CCI) as defined in CSI-2 specification since this is an I2C compliant control link. If CCI/I2C compliant link is required (in the absence of a preexisting one), Synopsys DesignWare APB/I2C bridge (DW_apb_i2c) is recommended.

1.1.4 Operational Model Overview

The CSI-2 Host Controller is a digital core that implements all protocol functions defined in the MIPI CSI-2 Specification, providing an interface between the System and the MIPI D-PHY, allowing the communication with a MIPI CSI-2 compliant Camera Sensor.

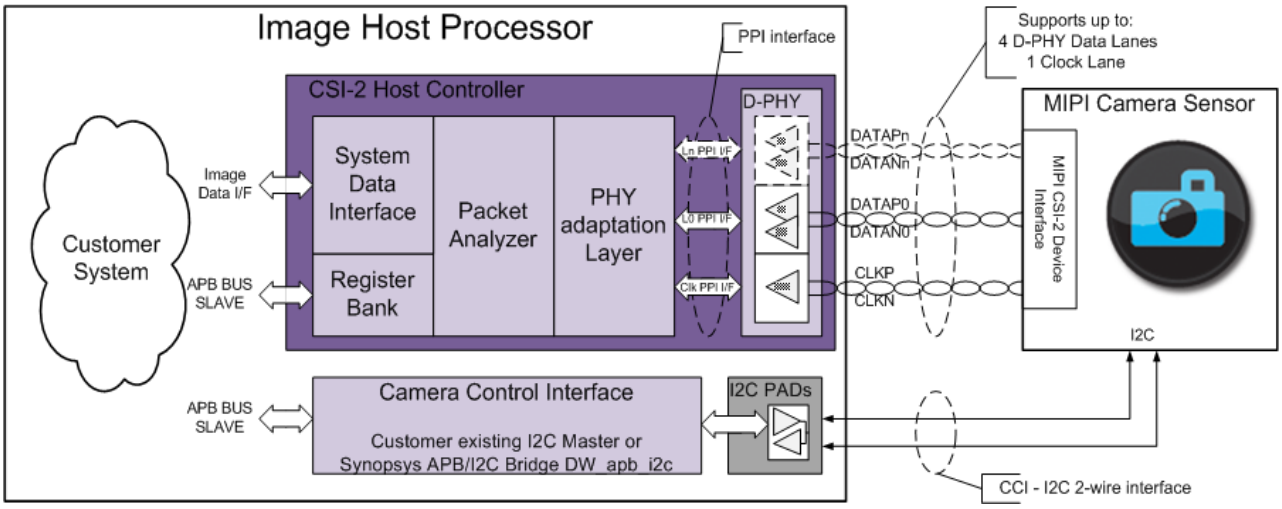
The CSI-2 Host controller provides means for seamless integration with Synopsys D-PHYs through coreConsultant. Optionally, the core can be configured for a non-Synopsys D-PHY. In such a configuration it exhibits a PPI compliant interface to connect to D-PHY.

The CSI-2 main functional blocks are as follows:

- D-PHY macro: Implements the physical link layer.
- PHY Adaptation Layer: Manages the D-PHY PPI interface.
- Packet Analyzer: Merges the data from the different lanes.
- Image Data Interface: Reorders pixels into 32-bit data for memory storage and generates timing accurate video synchronization signals.
- AMBA-APB Register Bank: Provides access to configuration and control registers.
- CCI/I2C control link: For camera configuration (optional).

Figure 1-2 shows the system level block diagram of MIPI CSI-2 .

Figure 1-2 Block Diagram of MIPI CSI-2 Host Controller Functions



1.2 Interfaces

MIPI CSI-2 has the following interfaces:

- Image Data Interface that provides:
  - 32-bit image data formatted for memory storage as recommended in CSI-2 specification
  - Vertical and Horizontal timing accurate video synchronization signals
  - Information regarding CSI-2 Data Type and Virtual Channel
- AMBA APB Slave bus for core configuration purposes
- D-PHY PPI interface as recommended in D-PHY specification

1.3 Features

MIPI CSI-2 includes the following features:

- Compliant with MIPI Alliance Standard for Camera Serial Interface 2 (CSI-2), Version 1.00 - 29 November 2005
- Optional support for Camera Control Interface (CCI) through the use of DesignWare Core (DW_apb_i2c)
- Interface with MIPI D-PHY following PHY Protocol Interface (PPI), as defined in MIPI Alliance Specification for D-PHY, Version 1.00.00 - 14 May 2009
- Supports up to 4 D-PHY Rx Data Lanes
- Dynamically configurable multi-lane merging
- Long and Short packet decoding
- Timing accurate signaling of Frame and Line synchronization packets
- Support for several frame formats such as:
  - General Frame or Digital Interlaced Video with or without accurate sync timing
  - Data type (Packet or Frame level) and Virtual Channel interleaving
- 32-bit Image Data Interface delivering data formatted as recommended in CSI-2 Specification
- Supports all primary and secondary data formats:
  - RGB, YUV and RAW color space definitions
  - From 24-bit down to 6-bit per pixel
  - Generic or user-defined byte-based data types
- Error detection and correction:
  - PHY level
  - Packet level
  - Line level
  - Frame level

1.4 Speed and Clock Requirements

DWC_mipi_csi2 host has two clock domains for its operation.

For most part, the core works in High Speed Byte clock provided by the D-PHY (pin RXBYTECLKHS). RXBYTECLKHS is, by specification, 1/4 of the DDR clock on the D-PHY clock lane.

The register bank relies on AMBA APB PCLK clock. APB PCLK clock should be a free running clock. If PCLK is gated for APB operations, a free running synchronized copy of PCLK must be provided in FPCLK pin. Using FPCLK comes as an option when doing core configuration in coreConsultant. Figure 1-3 illustrates clock domain distribution in the DWC_mipi_csi2 host core.

Figure 1-3 Clock Domain Distribution

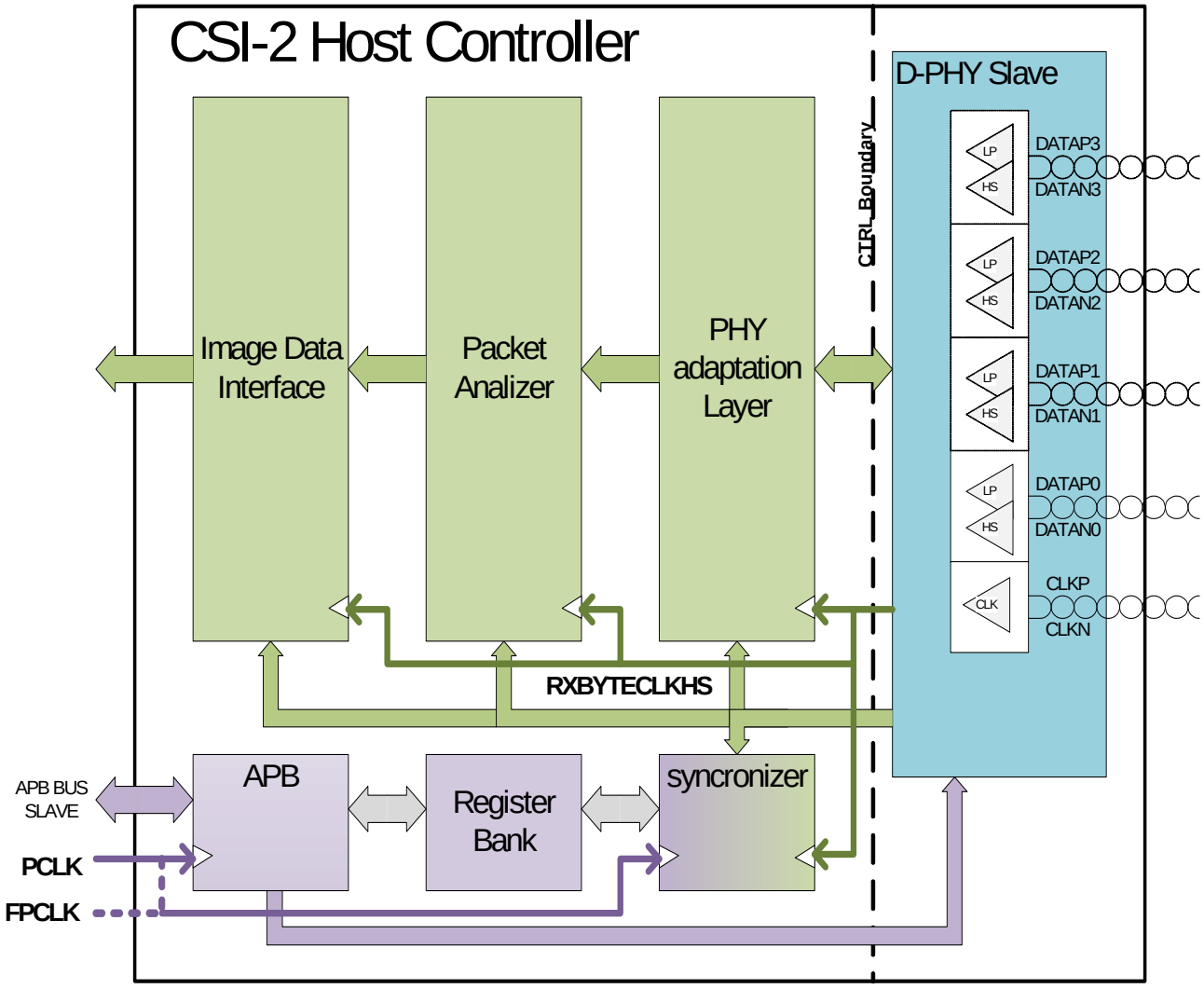


Table 1-1 shows the clock frequency of the DWC_mipi_csi2_host core.

Table 1-1 DWC_mipi_csi2_host Clock Frequency

Clock Domain	Minimum Frequency	Maximum Frequency
RXBYTECLKHS	10 MHz (per spec)	125 MHz (per spec)
PCLK (and FPCLK)	15 MHz	N/A

1.5 Area

Table 1-2 shows the area of the DWC_mipi_csi2_host core for 2 Lanes and 4 Lanes configuration. D-PHY area is not considered in these numbers.

Table 1-2 DWC_mipi_csi2_host Gate Count

Configuration	Area (gates)
2 Data Lanes configuration	7.2 K
4 Data Lanes configuration	8.3 K

2 Architecture Overview

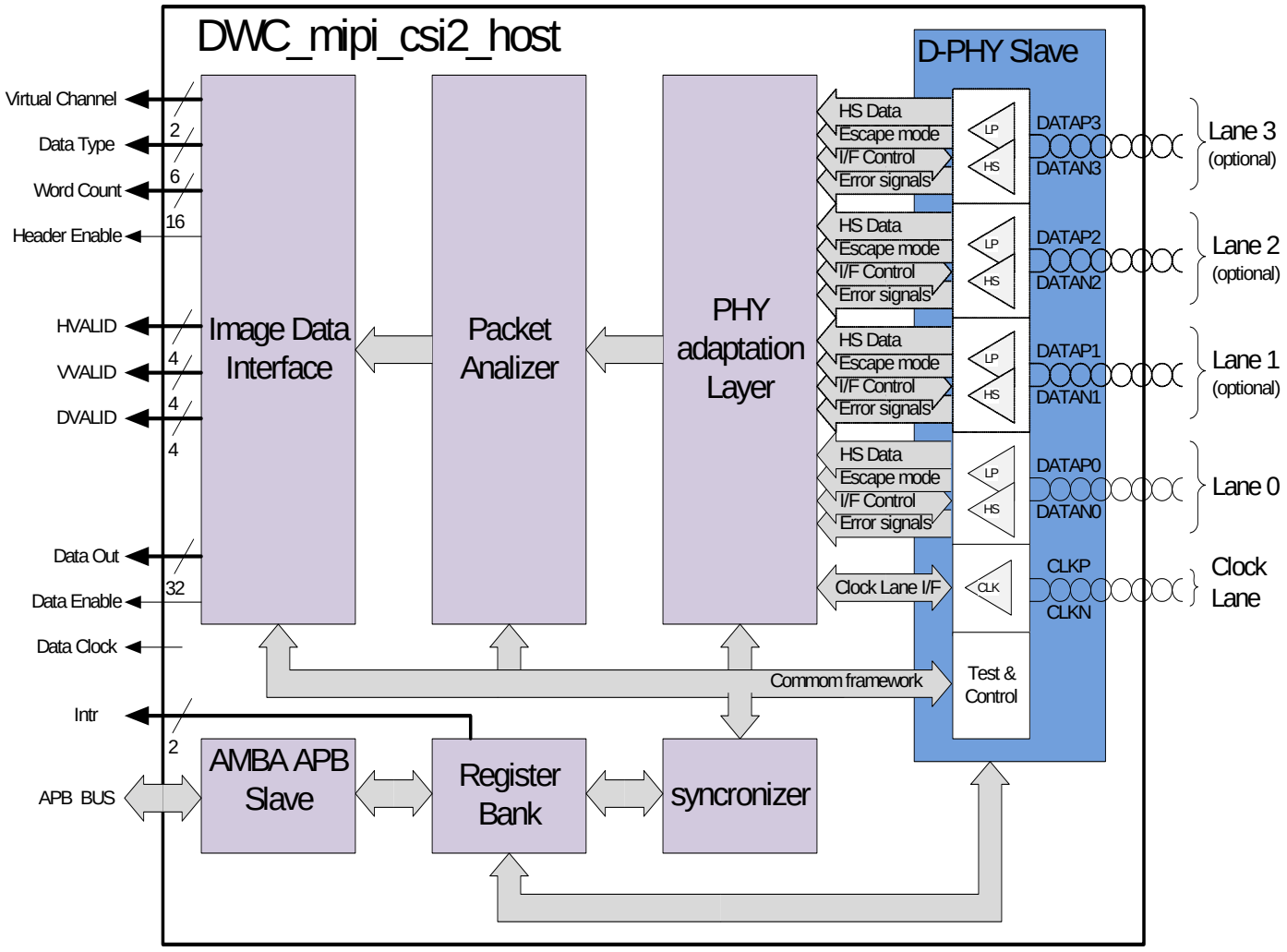
This chapter describes the MIPI CSI-2 core's general architecture, startup sequence, interrupt mechanism, and protocol error detection.

2.1 Architecture

Figure 2-1 shows the overall architecture of the CSI-2 Host Controller. The main blocks are as follows:

- PHY Adaptation Layer: It is responsible for managing the D-PHY interface, including PHY error handling.
- Packet Analyzer: Here data lane merging is implemented if required, together with header decoding, error detection and correction, frame size error detection and CRC error detection.
- Image Data Interface: This block separates CSI-2 packet header information and reorders data according to memory storage format, and it also generates timing accurate video synchronization signals. Several error detections are also performed at frame-level and line-level.
- Register Bank: This block is accessible through a standard AMBA-APB slave interface, providing access to the CSI-2 Host Controller registers for configuration and control. There is also a fully programmable interrupt generator to inform the system upon certain events.

Figure 2-1 Architecture of CSI-2 Host Controller

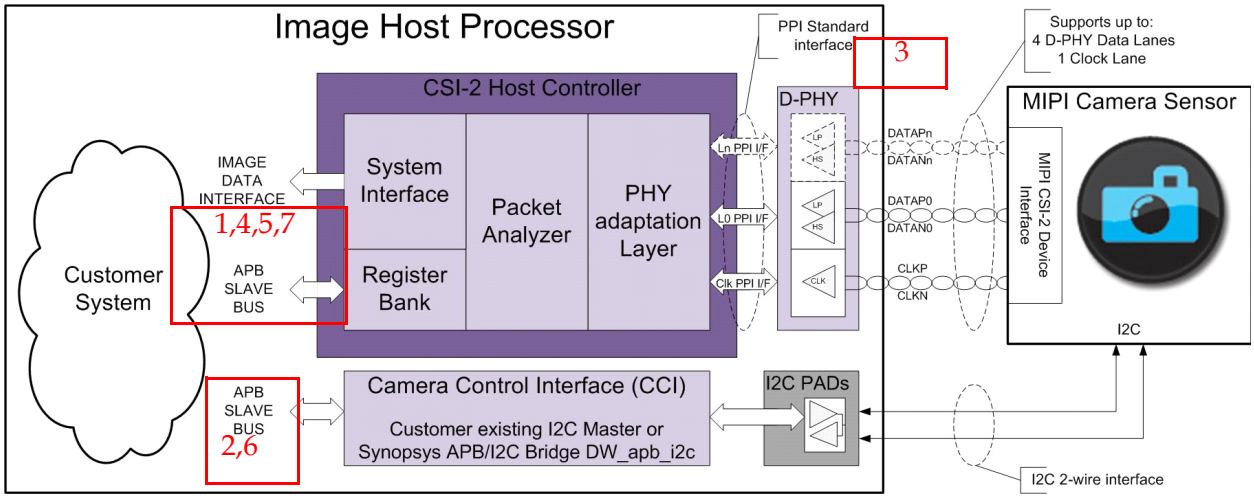


2.2 Initialization Sequence

The following information is provided as a guideline to allow a safe startup of the system operation.

Figure 2-2 shows how the DWC_mipi_csi2_host core can be integrated into an SoC.

Figure 2-2 Block Diagram Showing the Recommended Startup Sequence



The startup steps are as follows:

1. De-assert CSI-2 presetn signal (global reset).
2. Configure MIPI Camera Sensor to have all Tx lanes in LP-11 state (STOPSTATE). According to D-PHY Specification, the D-PHY master should be initialized at LP-11 state (STOPSTATE). However, a CCI command may be required to switch-on the MIPI interface.
3. D-PHY initialization - Access the D-PHY programming interface through registers PHY_TST_CRTL0 and PHY_TST_CTRL1 of AMBA APB to initialize and program the D-PHY. This is D-PHY dependent, and this programming should be carried out according to the D-PHY databook.
4. CSI-2 Controller programming - Program the CSI-2 Host Controller registers based on the operating mode required:
   a. Number of Lanes (register N_LANES)
   b. De-assert PHY shutdown (register PHY_SHUTDOWNZ)
   c. De-assert PHY reset (register PHY_RSTZ)
   d. De-assert CSI reset (register CSI2_RESETN)
   e. Program Data IDs for matching line error reporting (registers DATA_IDS_1 and DATA_IDS_2) (Optional)
   f. Program the interrupt masks (registers MASK1 and MASK2) (Optional)
5. CSI-2 Controller programming - Read the PHY status register (PHY_STATE) to confirm that all data and clock lanes of the D-PHY are in Stop State (that is, ready to receive data).
6. Configure the MIPI Camera Sensor - Access Camera Sensor using CCI interface to initialize and configure the Camera Sensor to start transmitting a clock on the D-PHY clock lane.

7. CSI-2 Controller programming - Read the PHY status register (PHY_STATE) to confirm that the D-PHY is receiving a clock on the D-PHY clock lane.

Note Additional steps may be required to correctly configure the specific D-PHY and MIPI Camera Sensors that are part of the system, as well as any other requirements that is integral part of the relevant MIPI Specifications.

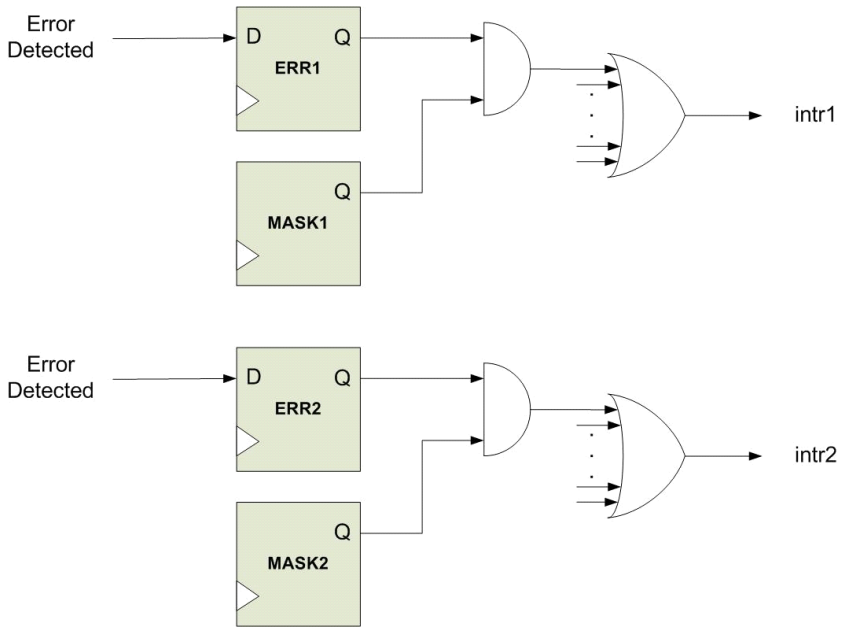
2.3 Interrupt mechanism

The CSI-2 Host Controller provides an interrupt mechanism that can be used mostly for monitoring errors and debugging.

There are two interrupt signals, intr1 and intr2, which are synchronous with the AMBA-APB clock signal. Registers MASK1 and MASK2 are used to select which bits of registers ERR1 and ERR2 are able to generate interrupts by asserting signals intr1 and intr2, respectively. Both ERR1 and ERR2 always contain the information of events, irrespective of the state of MASK1 and MASK2. Registers ERR1 and ERR2 will self-clear after a read access. Interrupt signals intr1 and intr2 are de-asserted upon read access of register ERR1 and ERR2, respectively.

Figure 2-3 shows the main parts of the interrupt mechanism:

Figure 2-3 Interrupt Mechanism



2.4 Error Detection

The DWC_mipi_csi2_host analyzes the received packets and determines if there are protocol errors. It is possible to monitor the following errors:

- Frame errors such as incorrect Frame sequence, reception of a CRC error in the most recent frame, and the mismatch between Frame Start and Frame end.
- Line errors such as incorrect line sequence and mismatch between Line Start and Line end.
- Packet errors such as ECC or CRC mismatch.
- PHY errors such as synchronization pattern mismatch.

Table 2-1 shows all the errors that can be identified by the CSI-2 Host IP:

Table 2-1 Errors Identified by the CSI-2 Host IP

Error	Description	Level	Action
phy_errsotsynchs_*	Start of Transmission Error on data lane * with no synchronization achieved	PHY	Packets with this error are not delivered in IDI interface
phy_erresc_*	Escape Entry Error (ULPM) on data lane *	PHY	Informative only. Error is acknowledged in register and Interrupt pin is raised
phy_errsoths_*	Start of Transmission Error on data lane * but synchronization can still be achieved	PHY	Informative only since PHY can recover from this error. Error is acknowledged in register and Interrupt pin is raised
vc*_err_crc	Checksum Error detected on Virtual Channel *	Packet	Informative only. Error is acknowledged in register and Interrupt pin is raised
vc*_err_ecc_corrected	Header ECC contains 1 error detected on Virtual Channel *	Packet	Informative only since controller can recover the correct header. Error is acknowledged in register and Interrupt pin is raised
err_ecc_double	Header ECC contains 2 errors. Unrecoverable	Packet	Packets with this error are not delivered in IDI. For debug purposes delivery can be unblocked by setting bypass_2ecc_tst
err_id_vc*	Unrecognized or unimplemented data type detected in Virtual Channel *	Packet	Informative only. Error is acknowledged in register and Interrupt pin is raised
err_l_bndry_match_di*[1]	Error matching Line Start with Line End for vc* and dt*	Line	Informative only. Error is acknowledged in register and Interrupt pin is raised
err_l_seq_di*1	Error in the sequence of lines for vc* and dt*	Line	Informative only. Error is acknowledged in register and Interrupt pin is raised
err_f_bndry_match_vc*	Error matching Frame Start with Frame End for Virtual Channel *	Frame	Informative only. Error is acknowledged in register and Interrupt pin is raised if not masked
err_f_seq_vc*	Incorrect Frame Sequence detected in Virtual Channel *	Frame	Informative only. Error is acknowledged in register and Interrupt pin is raised if not masked

Table 2-1 Errors Identified by the CSI-2 Host IP

Error	Description	Level	Action
err_frame_data_vc*	Last received frame, in Virtual Channel *, had at least one CRC error	Frame	Informative only. Error is acknowledged in register and Interrupt pin is raised

1. The line level errors are optional and can exist if the parameter CSI2_HOST_N_DATA_IDS is defined as 1 or 2. Otherwise, the IP will not have the logic to identify them.

2.4.1 Data ID Monitors

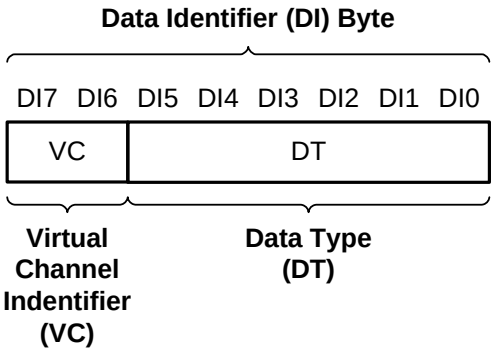
According to CSI-2 specification, each CSI-2 line contains information regarding Data Type and Virtual Channel identification. Nevertheless, each CSI-2 frame is identified uniquely by its Virtual Channel. In fact, a Frame can contain different Lines using different Data Types.

This fact imposes some restriction on how errors are monitored for CSI-2 lines. To have a mechanism to identify these errors it is required to have knowledge of the Data Type and Virtual Channel pairs that are active on the CSI-2 link. The DWC_mipi_csi2_host can optionally include registers to store this information. This is controlled by parameter CSI2_HOST_N_DATA_IDS as follows:

- Selecting 1 for CSI2_HOST_N_DATA_IDS adds 32 bit DATA_IDS_1 register to register map, creating the possibility to program up to four Data IDs (VC/DT pairs) for monitoring simultaneously.
- Selecting 2 for CSI2_HOST_N_DATA_IDS adds additional DATA_IDS_2 register, besides DATA_IDS_1, enabling up to eight different Data IDs that can be identified simultaneously.
- Keeping 0 for CSI2_HOST_N_DATA_IDS removes any logic associated with this functionality. This means that the register DATA_IDS_1 and DATA_IDS_2 will not be present on the register bank and that errors err_l_bndry_match_di* and err_l_seq_di* will not be monitored.

To configure the options mentioned above, on coreConsultant Configuration window, under “Select the number of Data IDs to monitor”, select None, 4, or 8 to define the parameter CSI2_HOST_N_DATA_IDS as 0, 1, or 2 respectively.

Figure 2-4 Definition of Data ID



The DATA_IDS registers, if selected during core configuration, are to be programmed during core initialization only (Step e in section 2.2).

For further details, protocol layer errors are detected and errors are reported as follows:

- Analyze the Data Type field of the headers of short and long packets and raise an error whenever an unrecognized or unimplemented one is found. This corresponds to err_id, and is reported for each Virtual Channel, independently (Register ERR2, fields err_id_vc*).
- Take note of the occurrence of CRC error detections in any of the packets that are received between a Frame Start and its corresponding Frame End. When the Frame End is received, err_frame_data error is raised, for the Virtual Channel where the frame was received, if at least one CRC error has been detected in some packet of that frame (Register ERR1, fields err_frame_data_vc*)
- Analyze the headers of short packets, aiming to match Frame Start with Frame End and Line Start with Line End packets by the corresponding Frame Number or Line Number.
  - If there are two consecutive Frame Start packets or two consecutive Frame End ones, or even if the Frame Number of a Frame End packet is not the same as for the last received Frame Start (all verified in the same Virtual Channel), err_f_bndry_match error is raised (Register ERR1, fields err_f_bndry_match _vc*).
  - If the Frame Number in a Frame Start packet is not incremented by 1 relatively to the previous one, for some Virtual Channel, err_f_seq error is raised. The exception is that if the Frame Number is 0, it is considered that frame numbering is inactive and if it is 1, it is considered a restart of frame numbers, so the error is not raised in these exceptional cases (Register ERR1, fields err_f_seq_vc*).
  - If there are two consecutive Line Start packets or two consecutive Line End ones, or even if the Line Number of a Line End packet is not the same as for the last received Line Start (all verified in the same Virtual Channel), err_l_bndry_match error is raised for each Data ID of Long Packets that have been received in between and that match the Data IDs that are programmed in the register bank (Register ERR1, err_l_bndry_match_di* where di* corresponds to the Data ID programmed in DATA_IDS_1 and DATA_IDS_2 registers).
  - The Line Number of each Line Start packet is saved, according to its Virtual Channel. Later, when a Long Packet is received for the same Virtual Channel, if its Data ID is programmed in the register bank and if that Line Number is out of the expected sequence for that Data ID, an err_l_seq error is raised for that Data ID. Prior to raising any err_l_seq error, the increment between successive lines, for a given Data ID has to be determined. For that, at least two consecutive lines of a given Data ID must be received without errors after each frame starts, so that the difference between those two Line Numbers can be used to determine the increment of the Line Numbers to expect for that Data ID (Register ERR1, err_l_seq_di* where di* corresponds to the Data ID programmed in DATA_IDS_1 and DATA_IDS_2 registers).

All the errors described above can trigger one of the two interrupt pins. Any of these errors can be independently masked to inhibit the generation of interrupts as described in "Interrupt mechanism" on page 22.

3 Timing Interfaces

This chapter describes the timing interfaces of the DWC MIPI CSI-2 controller.

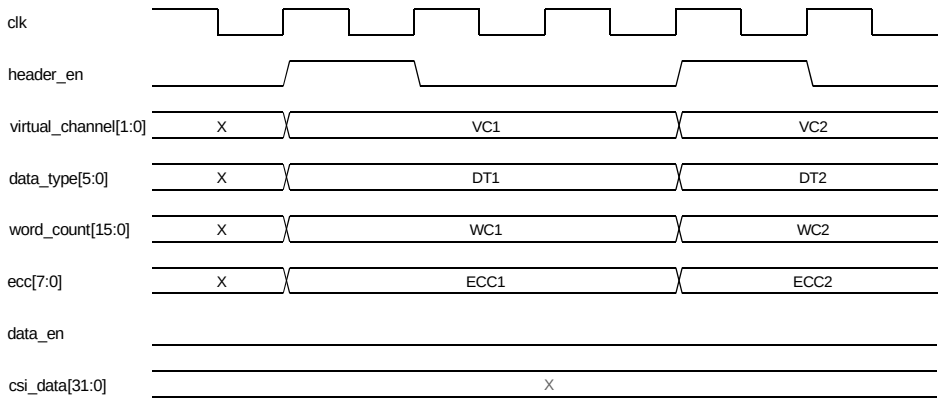
The interfaces described are:

3.1 Image Data Interface

At the Image Data Interface, signal header_en is used to indicate that new data is being transferred. It rises when a new packet becomes available at the interface and falls as soon as the packet finishes. Between two consecutive packets, there is always a fall and a rise of header_en, since the CSI-2 transmitter must enter Low-Power State between an End of Transmission and the following Start of Transmission.

An example of transferring two short packets is presented in the following Figure. The fields of the header packet become available simultaneously to the rise of header_en, which falls after one clock cycle, as no new data is to be transferred. This is the behavior of the circuit, independently of the number of active lanes.

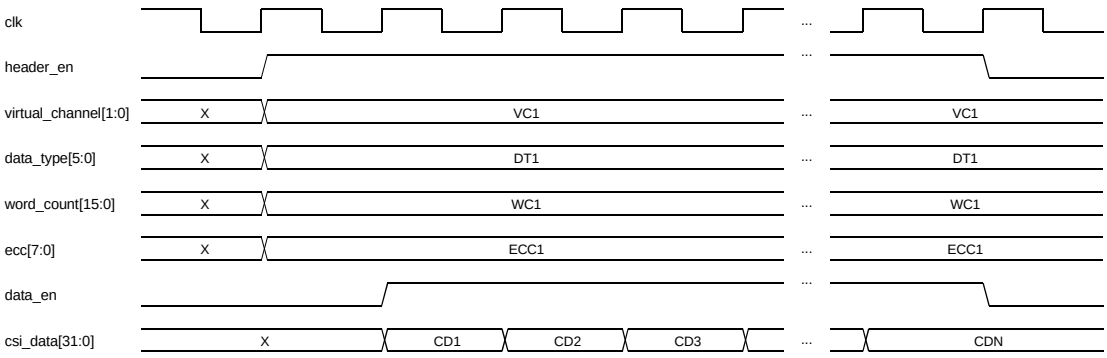
Figure 3-1 Image Data Interface example with two short packets



When transferring long packets, the data provided by the payload on all lanes is transferred separately on csi_data bus, while the header fields remain stable until the transfer of the packet has been completed.

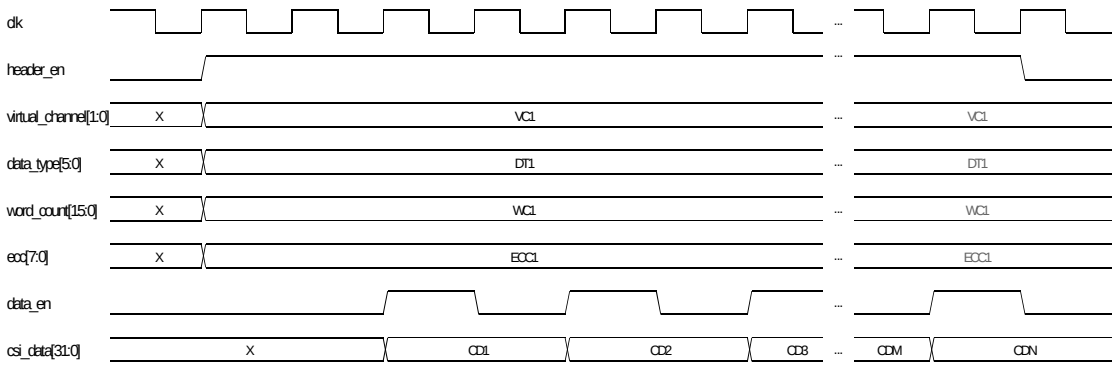
Signal data_en is used to indicate that a new 32-bit word is available in csi_data, and it can only be set if header_en is also set. The following Figure shows an example of transferring a long packet received from 4 data lanes. Since a new 32-bit word is transferred at each clock cycle, signal data_en remains set until all data is transferred.

Figure 3-2 Timing interface for a long packet, receiving data from 4 lanes



If less than 4 lanes are used, a 32-bit word might take more than one clock cycle to be received. In this case, data_en is set only when a new word has been fully received and becomes available at the interface. As shown in Figure 3-3, when data is received from two lanes, a new word is released every two clock cycles. Both header_en and data_en return to 0 as soon as all the data has been transferred.

Figure 3-3 Timing interface for a long packet, receiving data from 2 lanes



If data is transferred using only 1 lane, a new 32-bit word becomes available every four clock cycles. In the case of 3 lanes, a new word is available at all but fourth clock cycles.

Figure 3-4 Timing interface for a long packet, receiving data from 1 lane

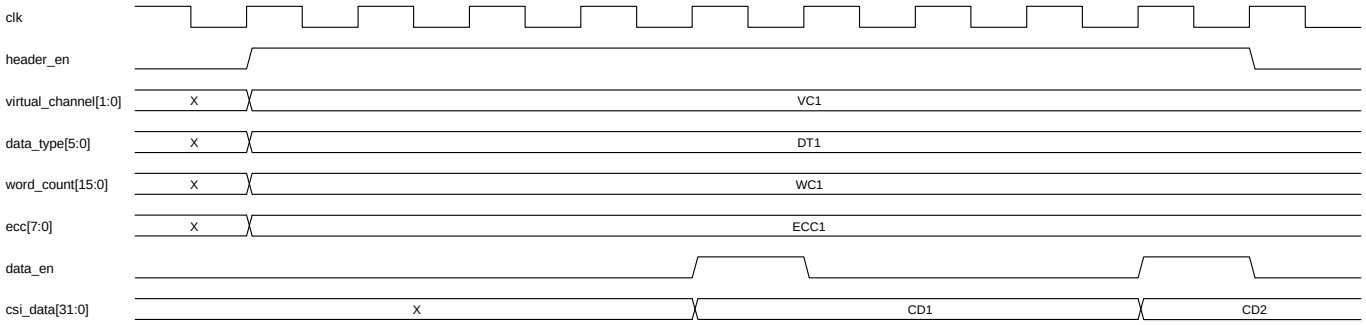
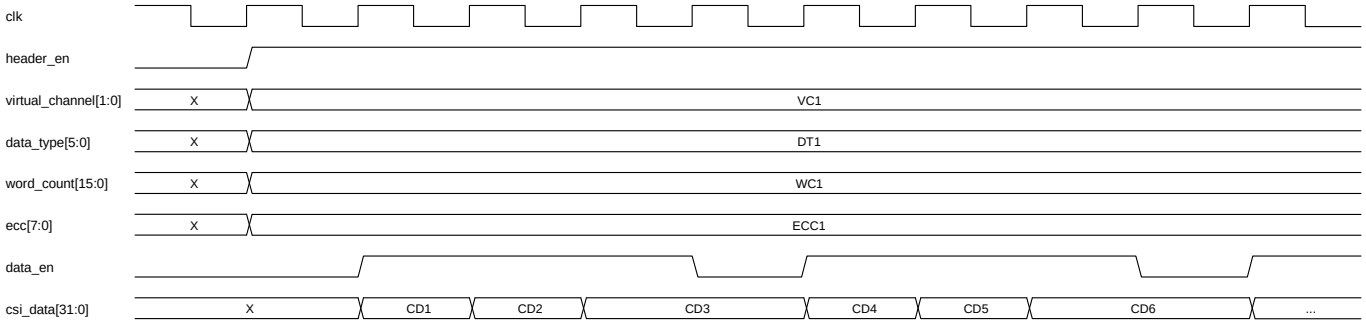


Figure 3-5 Timing interface for a long packet, receiving data from 3 lanes



A CSI-2 packet size is always an integer number of bytes but that number is not necessarily a multiple of 4. Since csi_data is 32-bit wide, at the end of a packet, part of the bytes may not contain valid packet data. The

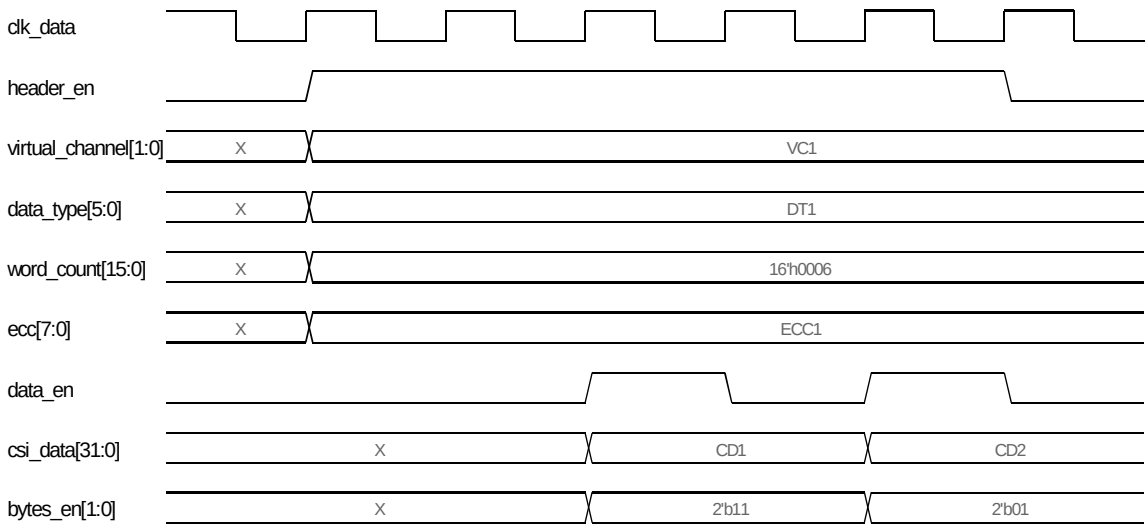
signal bytes_en[1:0] indicates how many bytes in the csi_data output signal are valid bytes as shown in Table 3-1.

Table 3-1 Valid csi_data Bits

	Bits From the csi_data Bus			
bytes_en[1:0]	**[31:24]**	**[23:16]**	**[15:8]**	**[7:0]**
2'b 00	Don't care	Don't care	Don't care	Valid
2'b 01	Don't care	Don't care	Valid	Valid
2'b 10	Don't care	Valid	Valid	Valid
2'b 11	Valid	Valid	Valid	Valid

While the payload data is being transferred, bytes_en is always kept at 2'b11, indicating that the 4 bytes in csi_data are from packet's payload. The exception is when the last word becomes available, as bytes_en might change according to the number of bytes that belong to payload. As an example, a timing diagram of the interface is presented in Figure 3-7, for a packet with 6 bytes of payload, transferred in two lanes. Bits in the range [15:0] of the value CD2 are payload, while bits in the range [31:16] of CD2 are meaningless and should be discarded.

Figure 3-6 Timing interface for a 6 bytes long packet



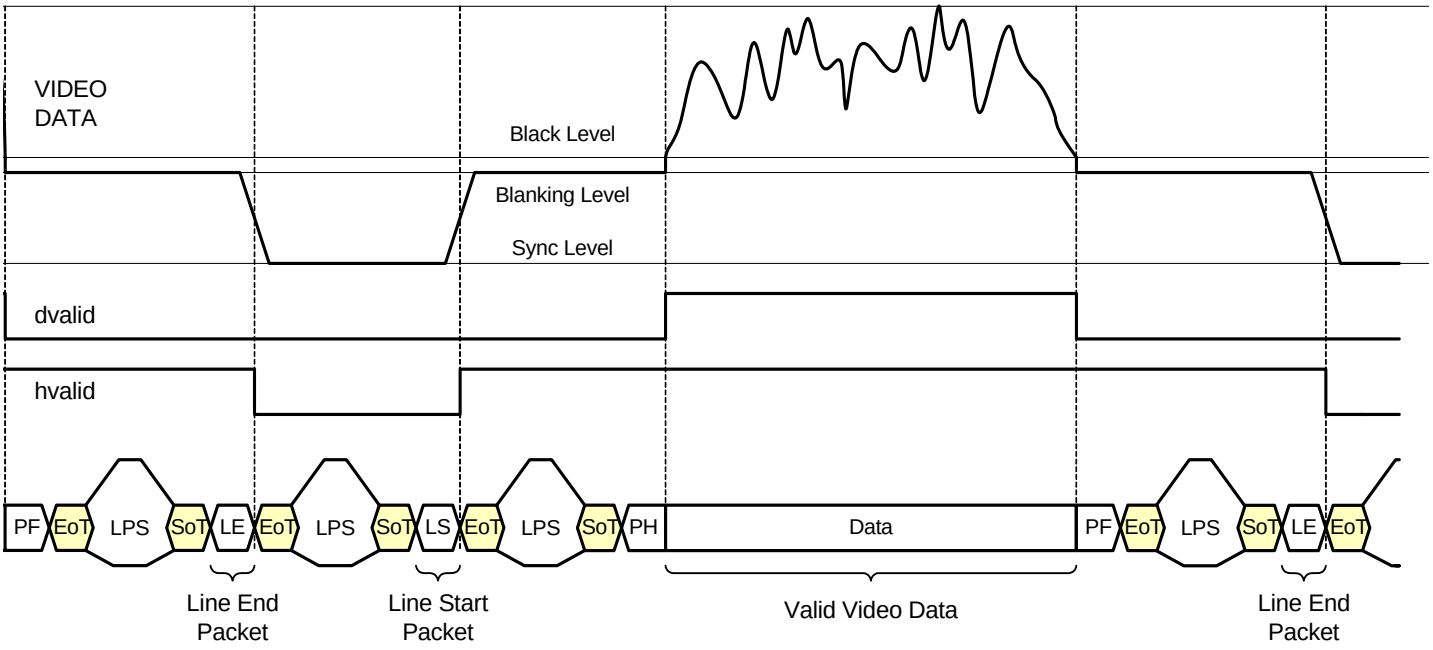
Signals dvalid, hvalid, and vvalid, are used to provide information about Frame Start, Frame End, Line Start and Line End packets. These signals are included in the interface for synchronization purposes.

dvalid is used to indicate when data is being transferred, excluding header information. Data sent through Blanking Data or Null packets does not activate dvalid.

hvalid is set on detection of Line Start packets and unset by Line End ones. These packets can be used as a reference for synchronization, even though relevant data can be more or less delayed between them, as it might be surrounded by Blanking Periods. Because Line Start and Line End packets are optional, in case they are not available, hvalid adopts the same behavior of dvalid, and is not activated by Blank or Null

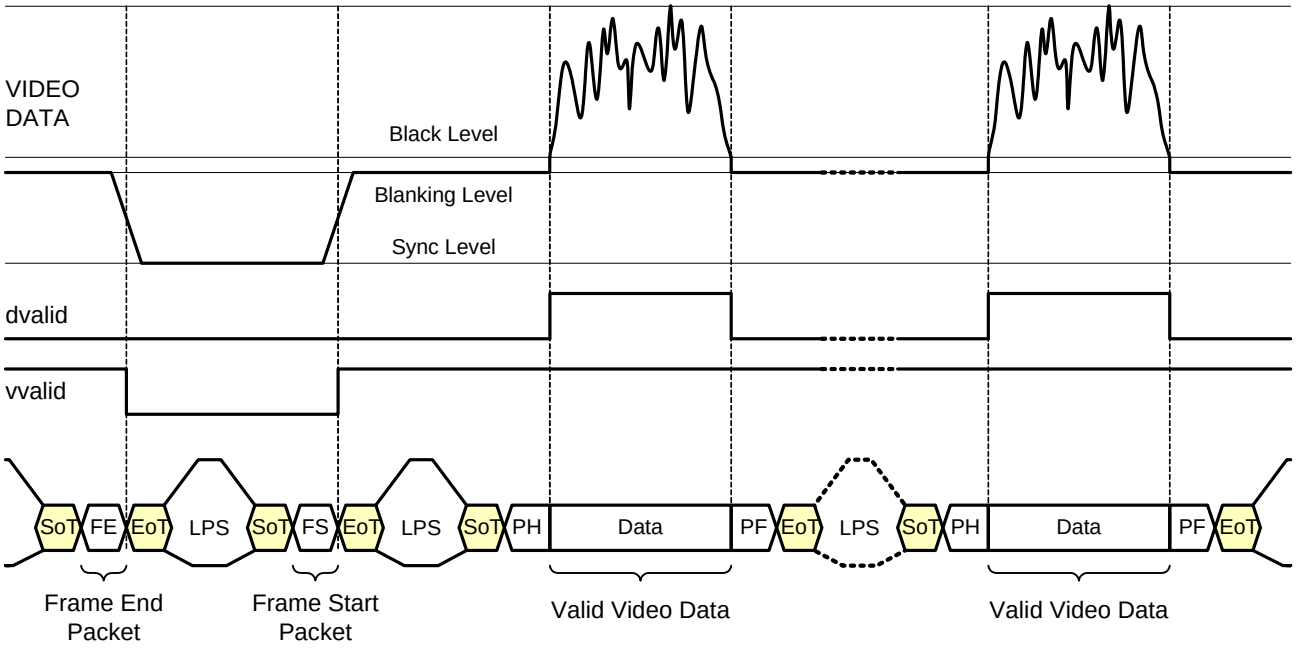
packets. In case a new Line Start packet is received without a Line End indicating the end of previous line, a pulse will be generated in hvalid to signal that a new Line Start packet was received.

Figure 3-7 HVALID synchronization signal



vvalid is set on detection of Frame Start packets and unset by Frame End ones. As these packets are mandatory, vvalid is always reliable for video applications to synchronize frame updating. In case a new Frame Start packet is received without a Frame End indicating the end of previous frame, a pulse is generated in vvalid to signal that a new Frame Start packet was received.

Figure 3-8 VVALID synchronization signal



3.2 Payload Data Output Format

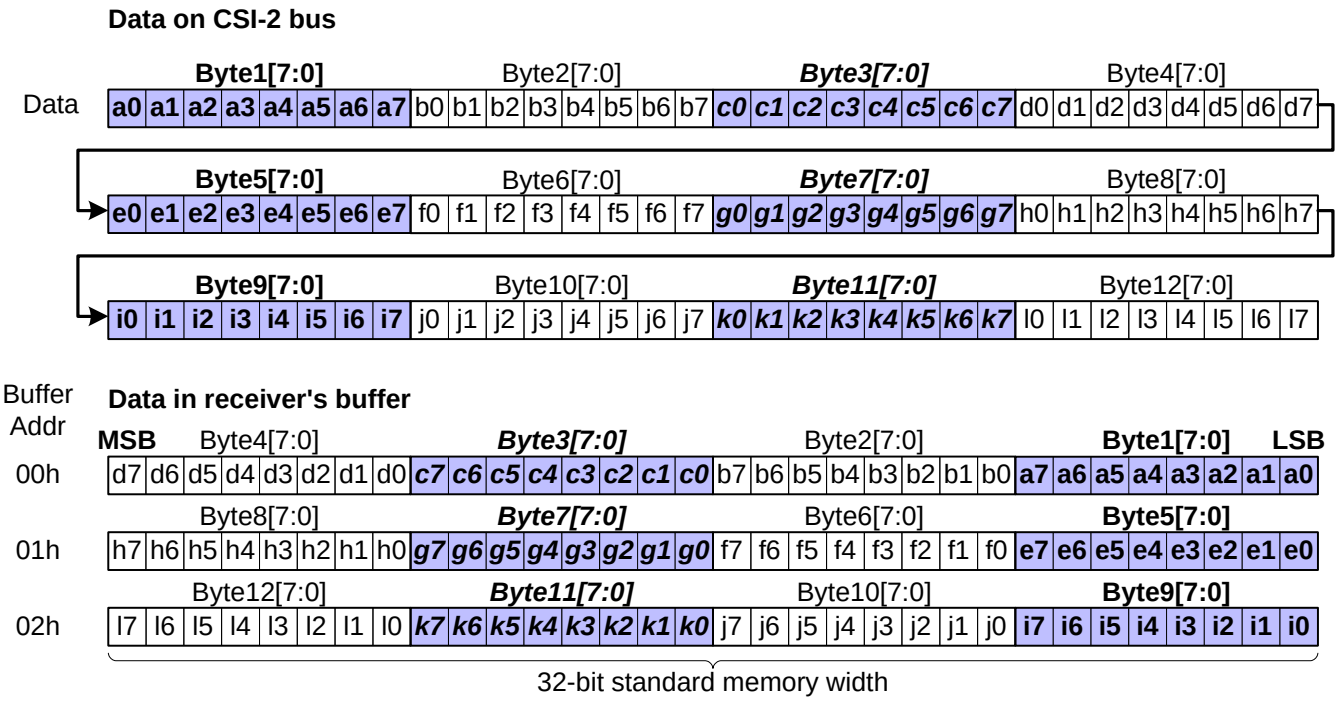
The Image Data Interface delivers payload data in a common data storage format, as suggested in CSI-2 Specification and described below. The following sections describe how different data formats are transferred in output bus csi_data.

3.2.1 General/Arbitrary Data Reception

In the generic case and for arbitrary data, the first byte of payload data transmitted maps the least significant byte of the 32-bit memory word and the fourth byte of payload data transmitted maps to the most significant byte of the 32-bit memory word.

The following Figure illustrates the generic CSI-2 byte to 32-bit memory word mapping rule.

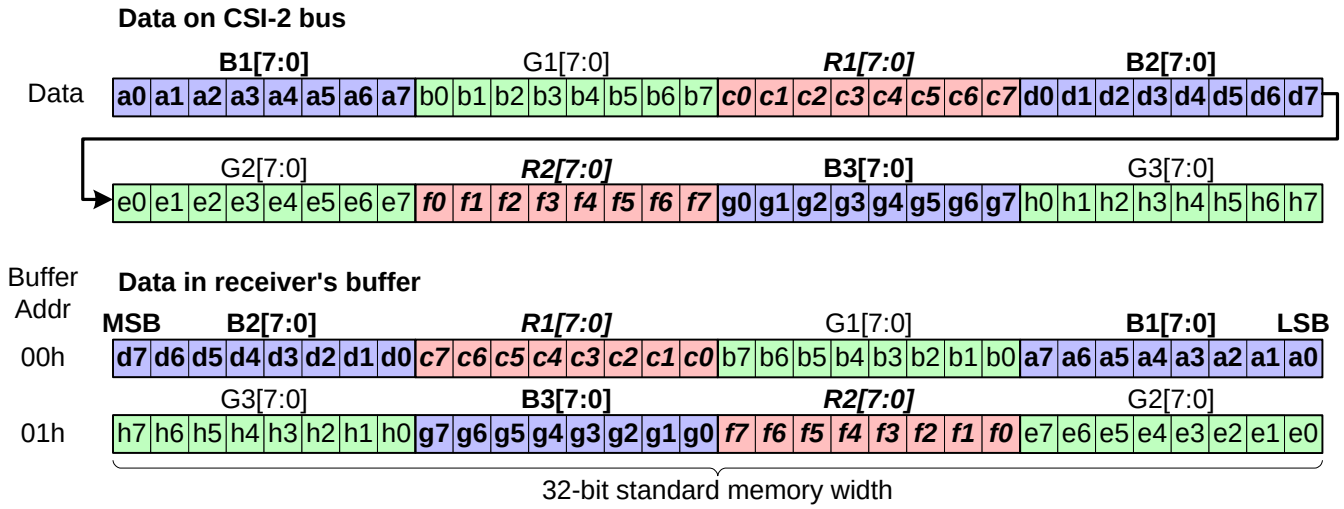
Figure 3-9 General/Arbitrary Data Reception



3.2.2 RGB888 Data Reception

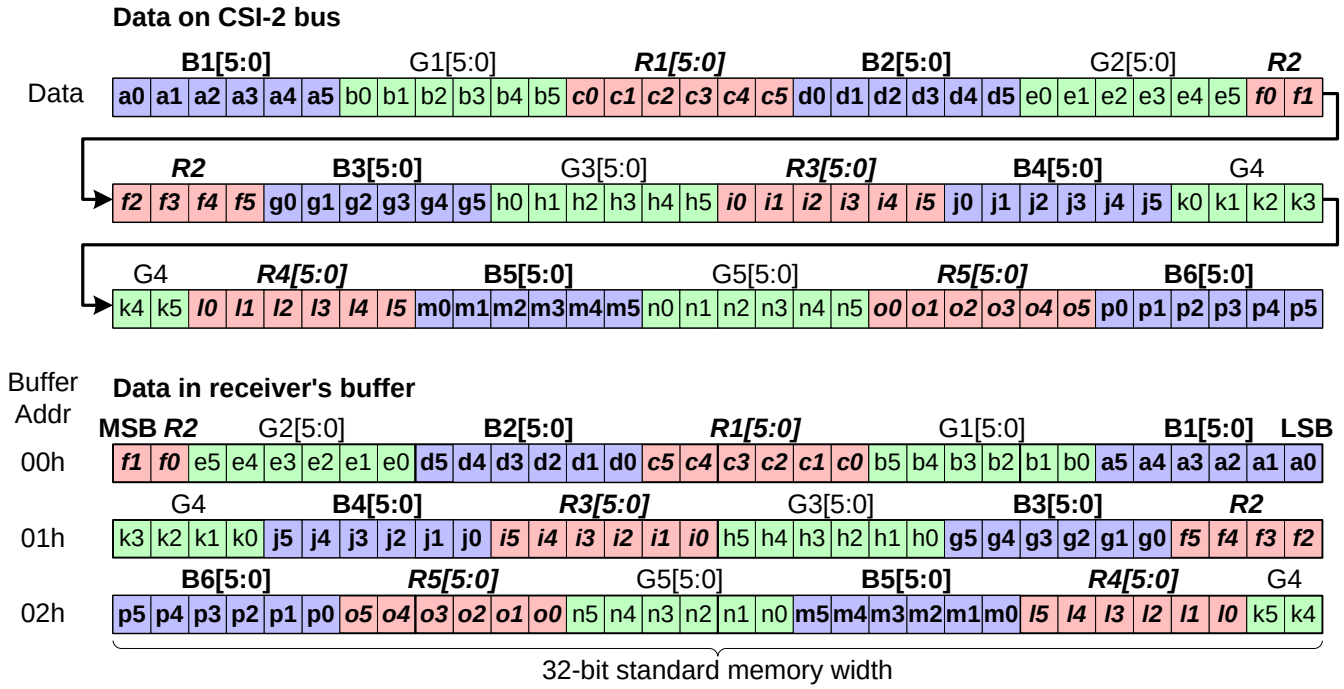
The RGB888 data format byte to 32-bit memory word mapping follows the generic CSI-2 rule.

Figure 3-10 RGB888 Data Format Reception



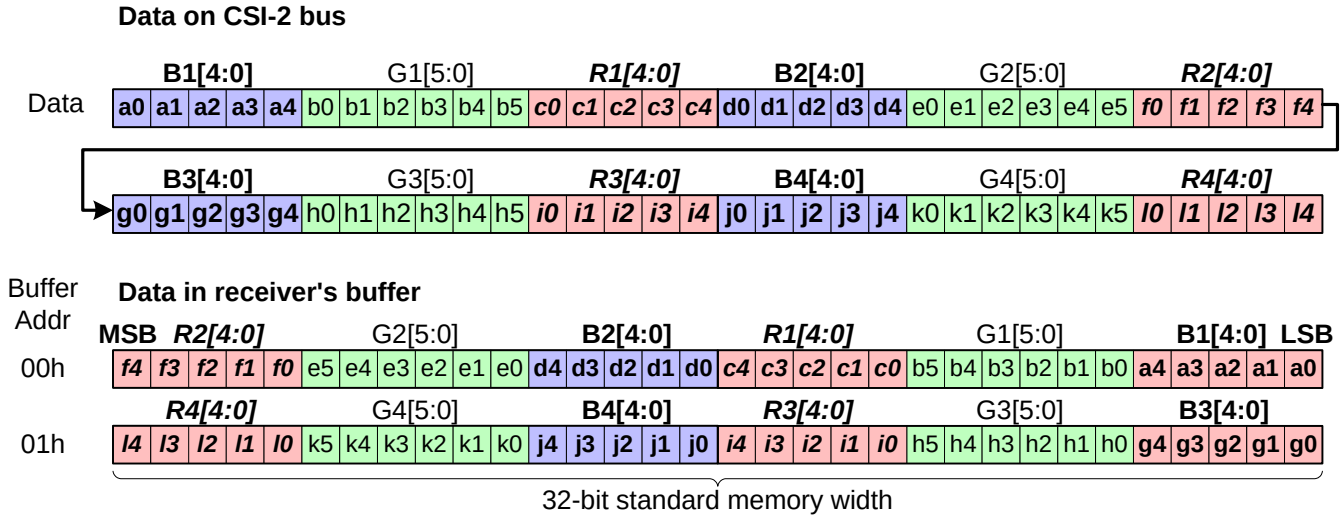
3.2.3 RGB666 Data Reception

Figure 3-11 RGB666 Data Format Reception



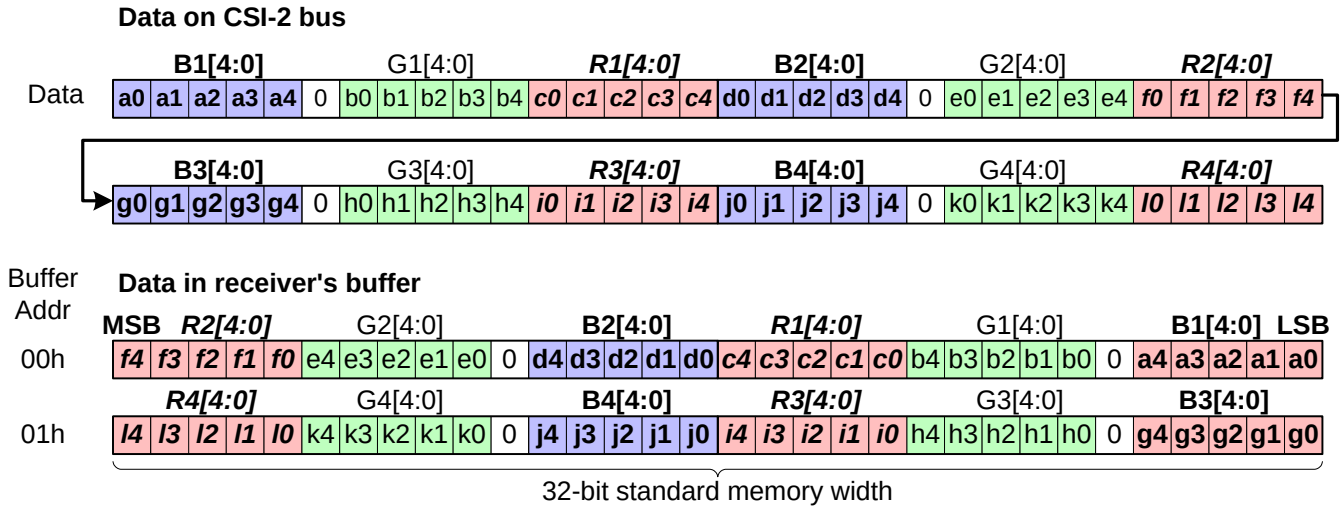
3.2.4 RGB565 Data Reception

Figure 3-12 RGB565 Data Format Reception



3.2.5 RGB555 Data Reception

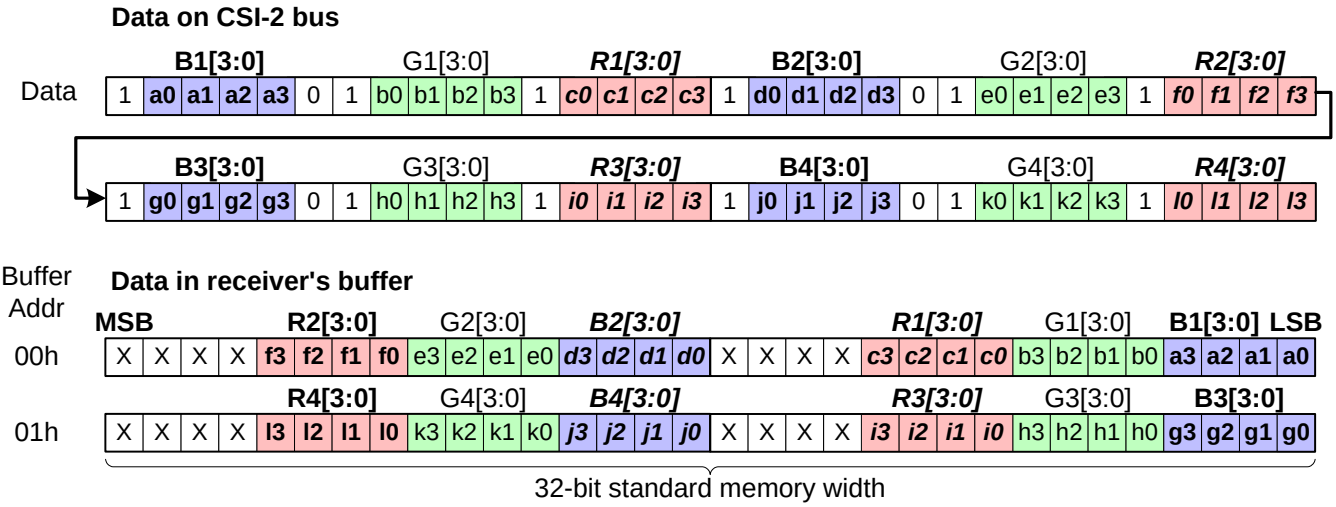
Figure 3-13 RGB555 Data Format Reception



3.2.6 RGB444 Data Reception

The RGB444 data format byte to 32-bit memory word mapping has a special transform as shown in the following Figure:

Figure 3-14 RGB444 Data Format Reception

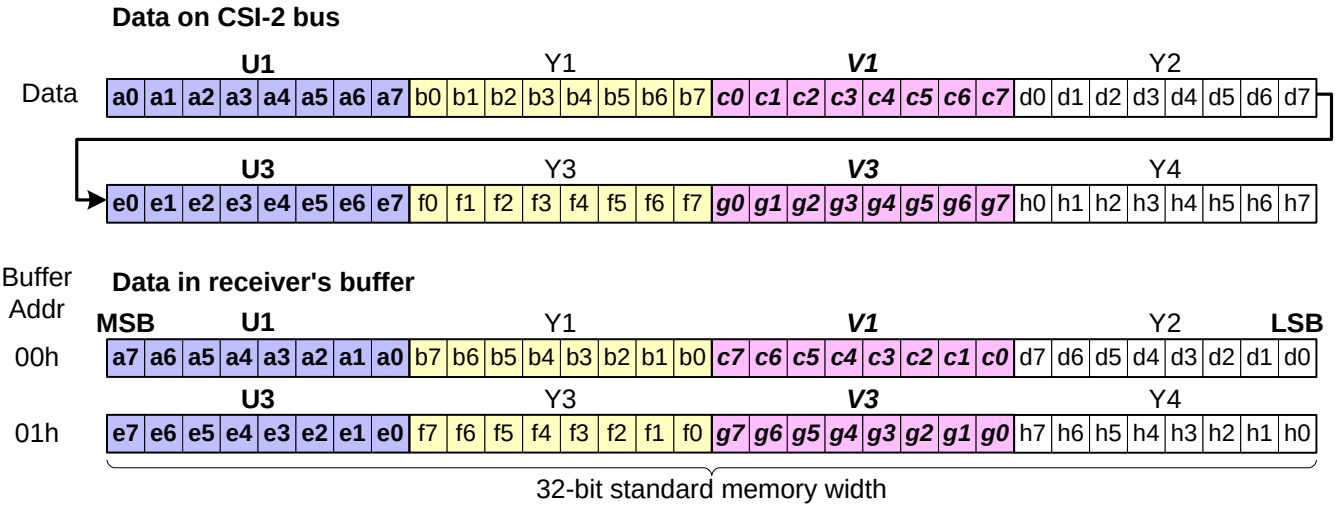


3.2.7 YUV422 8-bit Data Reception

The YUV422 8-bit data format the byte to 32-bit memory word mapping does not follow the generic CSI-2 rule.

For YUV422 8-bit data format the first byte of payload data transmitted maps the most significant byte of the 32-bit memory word and the fourth byte of payload data transmitted maps to the least significant byte of the 32-bit memory word.

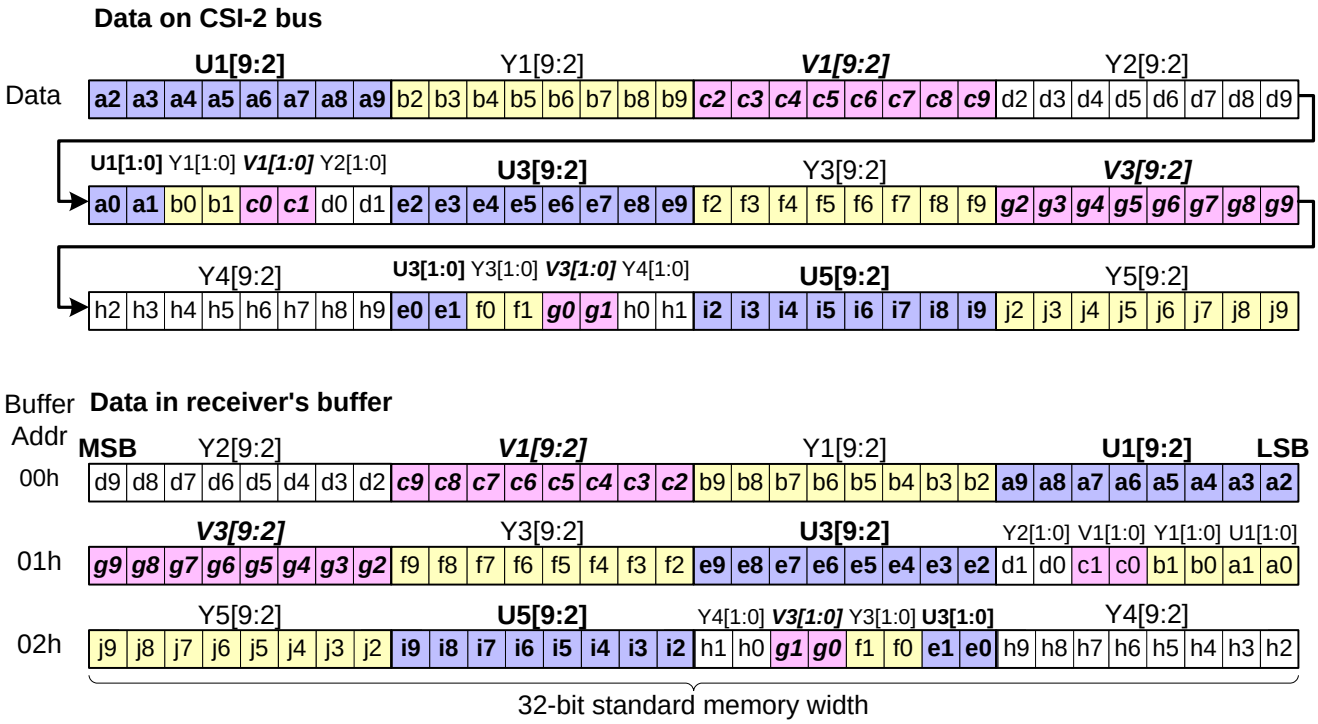
Figure 3-15 YUV422 8-bit Data Format Reception



3.2.8 YUV422 10-bit Data Reception

The YUV422 10-bit data format the byte to 32-bit memory word mapping follows the generic CSI-2 rule.

Figure 3-16 YUV422 10-bit Data Format Reception

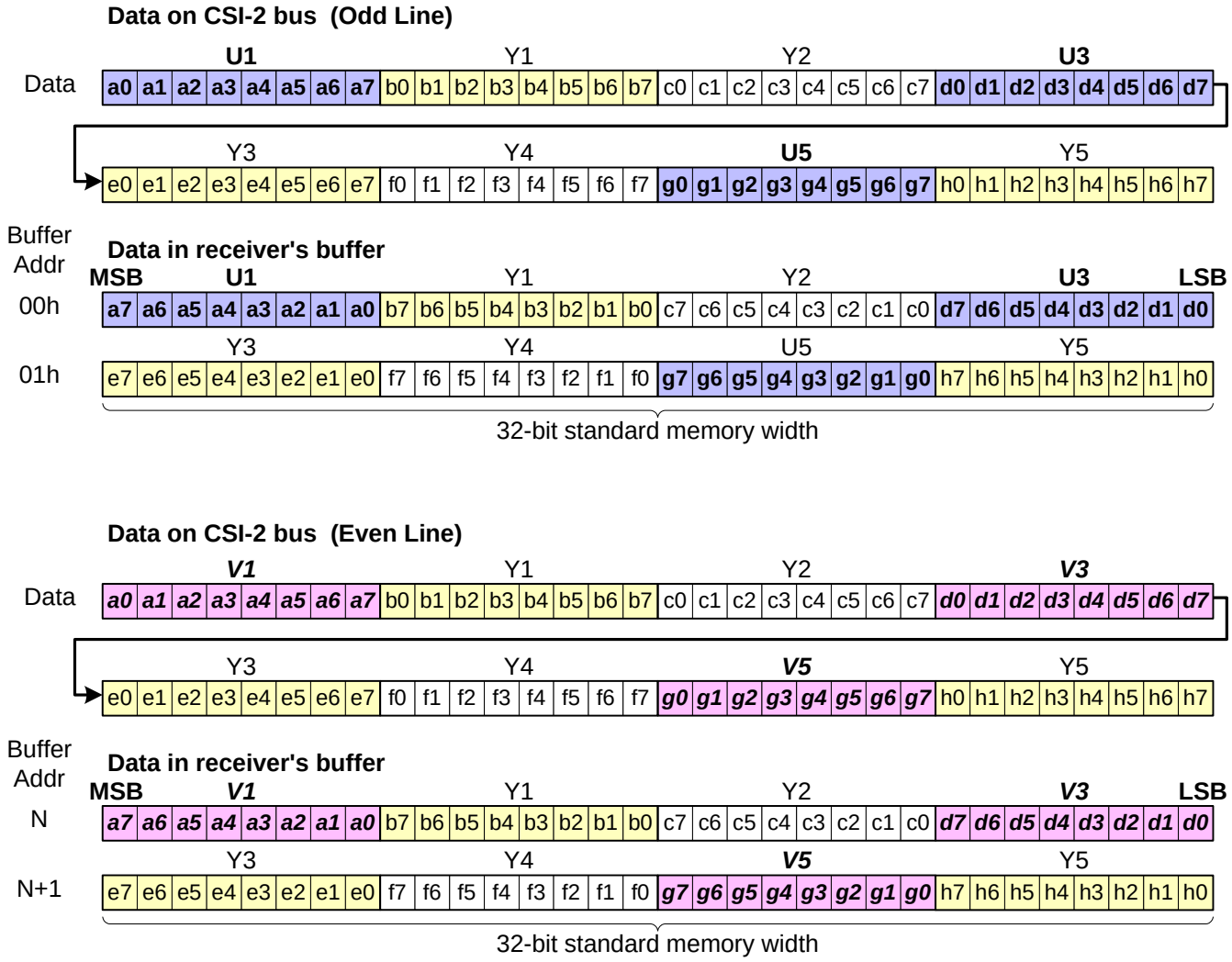


3.2.9 YUV420 8-bit (Legacy) Data Reception

The YUV420 8-bit (legacy) data format the byte to 32-bit memory word mapping does not follow the generic CSI-2 rule.

For YUV422 8-bit (legacy) data format the first byte of payload data transmitted maps the MS byte of the 32-bit memory word and the fourth byte of payload data transmitted maps to the LS byte of the 32-bit memory word.

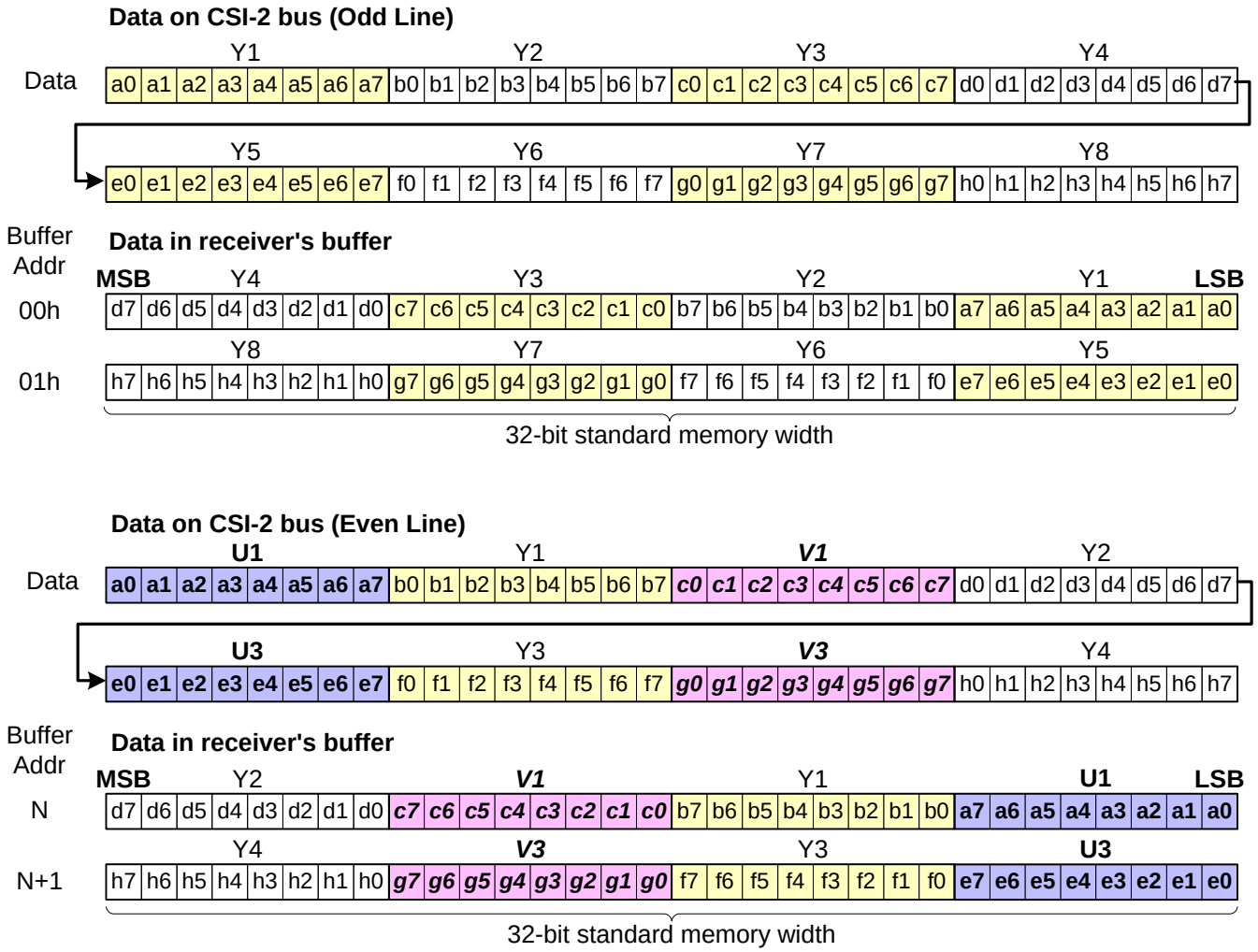
Figure 3-17 YUV420 8-bit (Legacy) Data Format Reception



3.2.10 YUV420 8-bit Data Reception

The YUV420 8-bit data format the byte to 32-bit memory word mapping follows the generic CSI-2 rule.

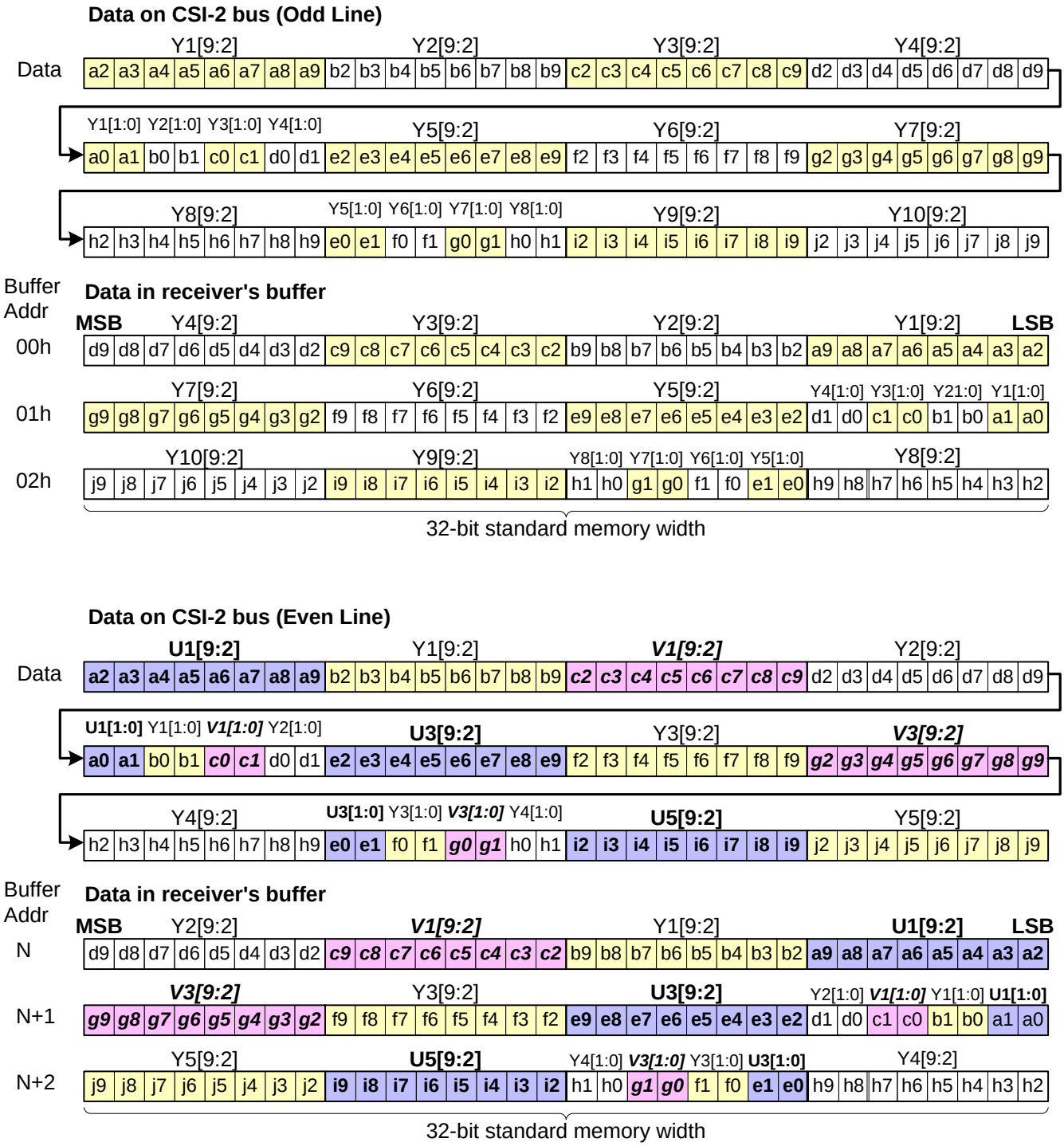
Figure 3-18 YUV420 8-bit Data Format Reception



3.2.11 YUV420 10-bit Data Reception

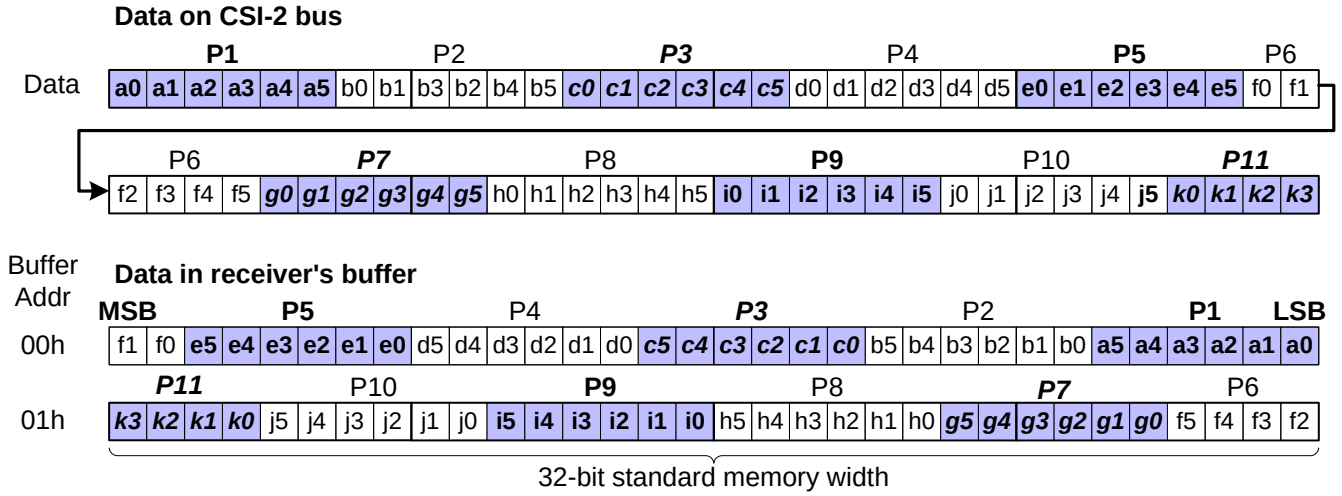
The YUV420 10-bit data format the byte to 32-bit memory word mapping follows the generic CSI-2 rule.

Figure 3-19 YUV420 10-bit Data Format Reception



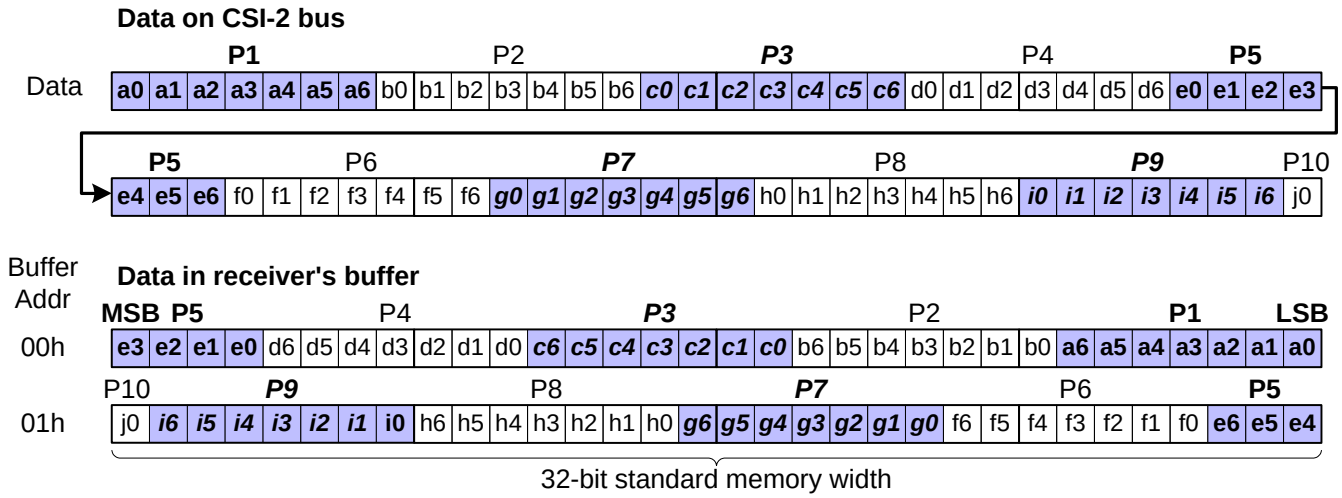
3.2.12 RAW6 Data Reception

Figure 3-20 RAW6 Data Format Reception



3.2.13 RAW7 Data Reception

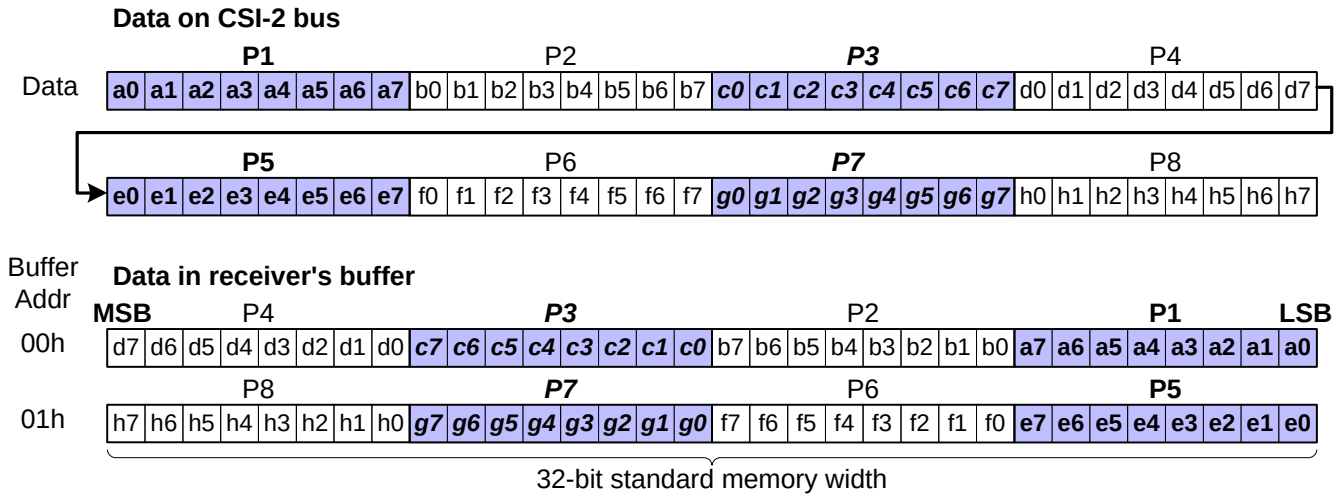
Figure 3-21 RAW7 Data Format Reception



3.2.14 RAW8 Data Reception

The RAW8 data format the byte to 32-bit memory word mapping follows the generic CSI-2 rule.

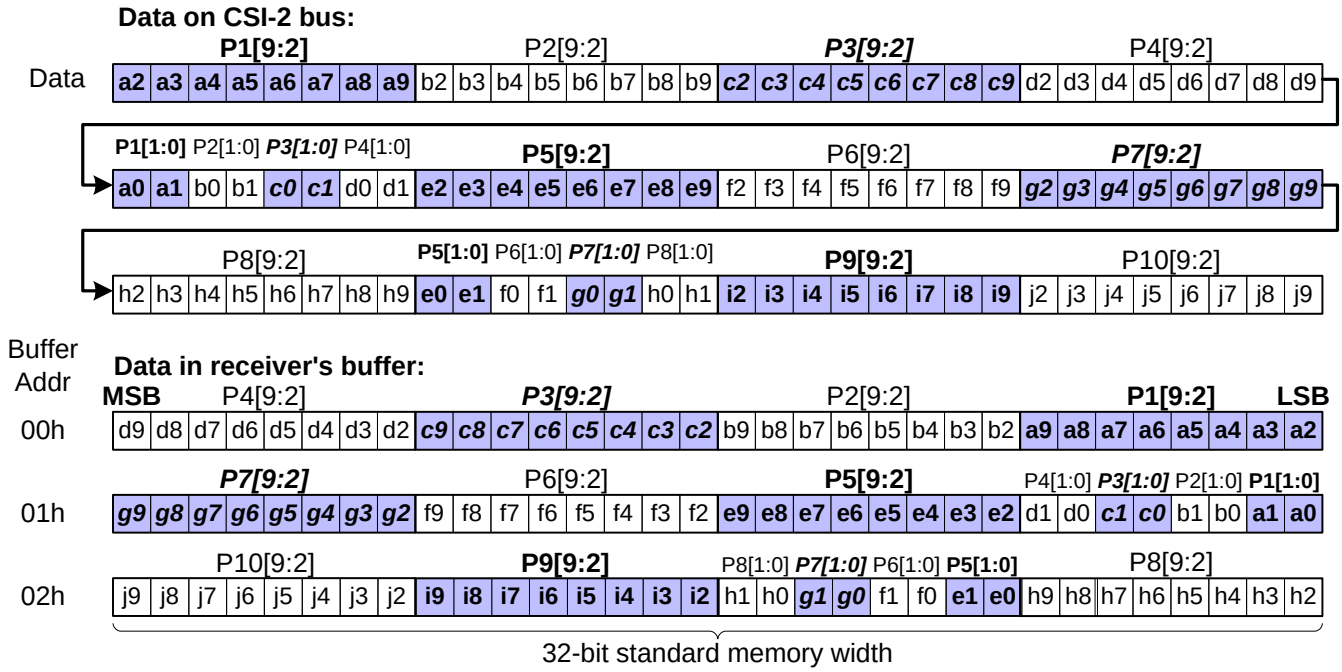
Figure 3-22 RAW8 Data Format Reception



3.2.15 RAW10 Data Reception

The RAW10 data format the byte to 32-bit memory word mapping follows the generic CSI-2 rule.

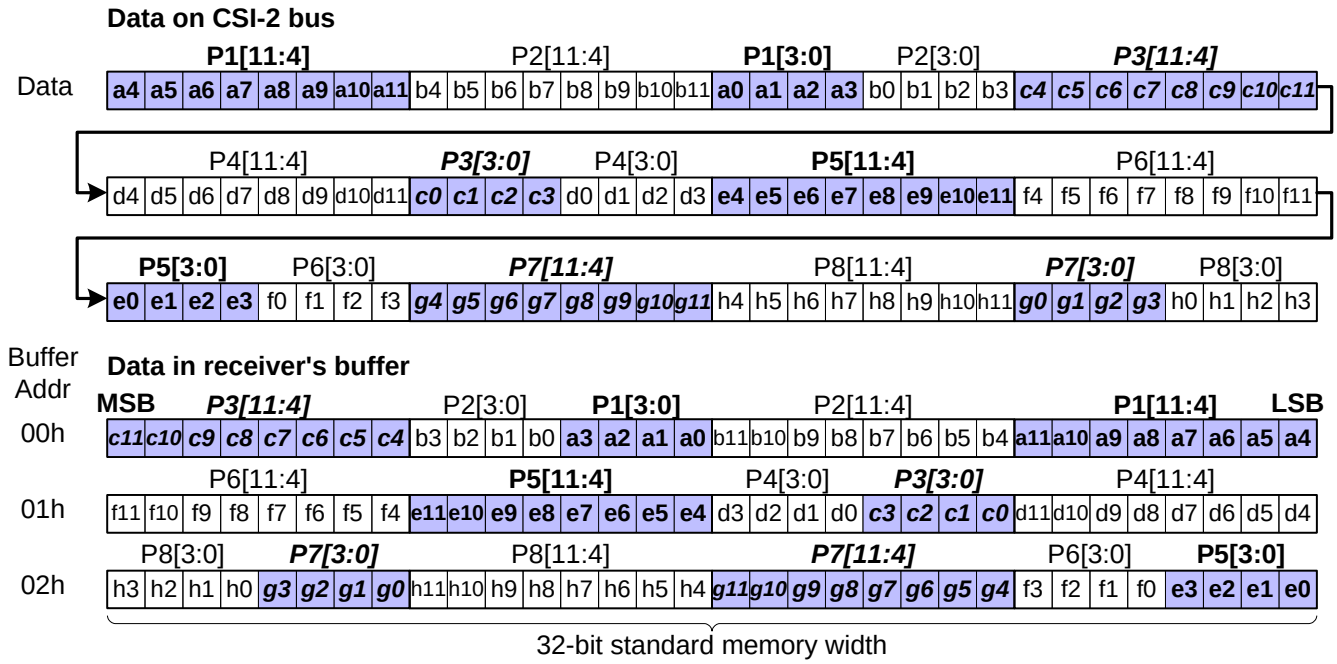
Figure 3-23 RAW10 Data Format Reception



3.2.16 RAW12 Data Reception

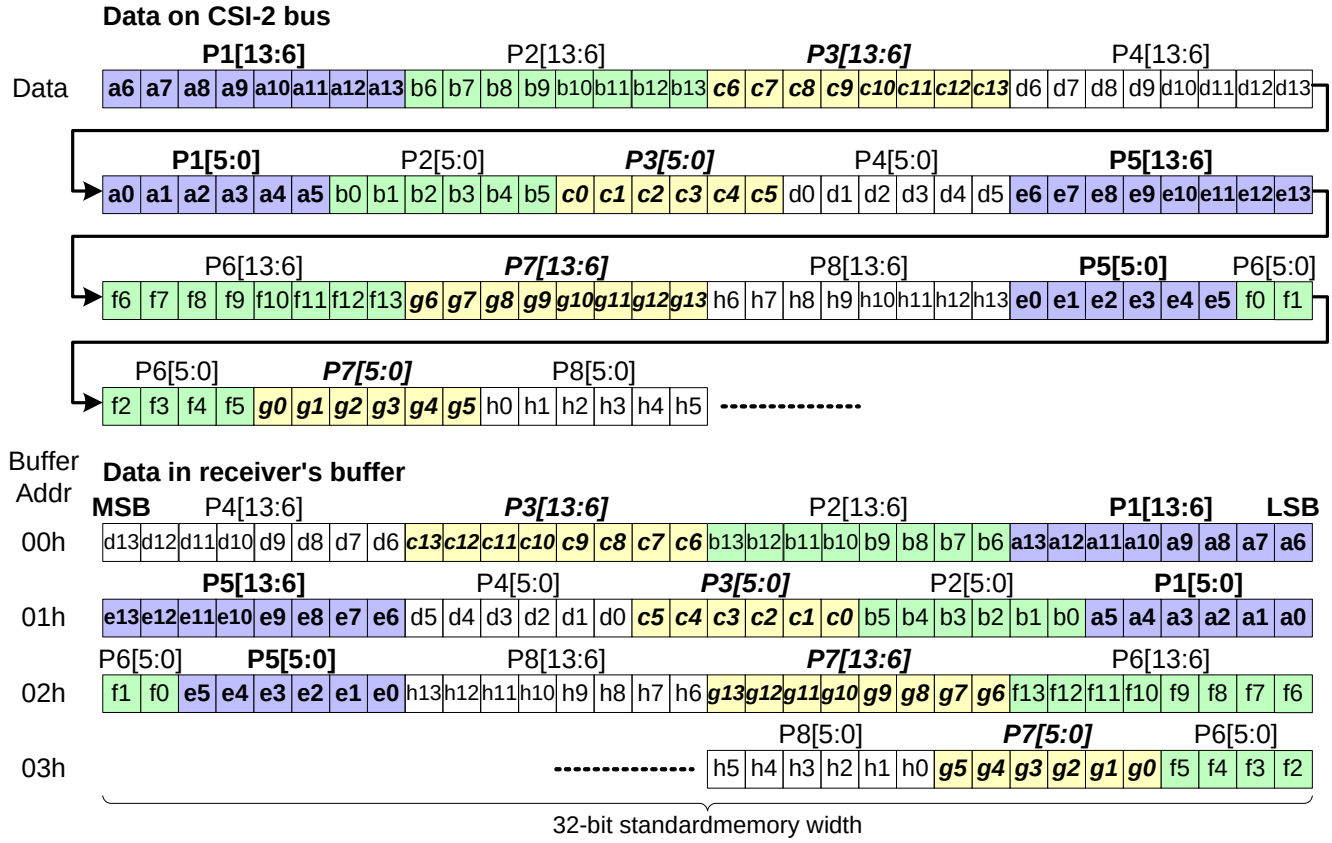
The RAW12 data format the byte to 32-bit memory word mapping follows the generic CSI-2 rule.

Figure 3-24 RAW12 Data Format Reception



3.2.17 RAW14 Data Reception

Figure 3-25 RAW14 Data Format Reception



4
Hardware Configuration Parameters

This chapter provides a description of the hardware configuration parameters available for the DWC_mipi_csi2_host. You use either the coreConsultant or coreAssembler GUI to specify the configuration parameters. This chapter also describes how to configure your core using the Specify Configuration dialog in the coreConsultant GUI. coreConsultant is your user interface for configuration, standalone verification, and synthesis of the DWC_mipi_dsi_host core.

4.1 Overview

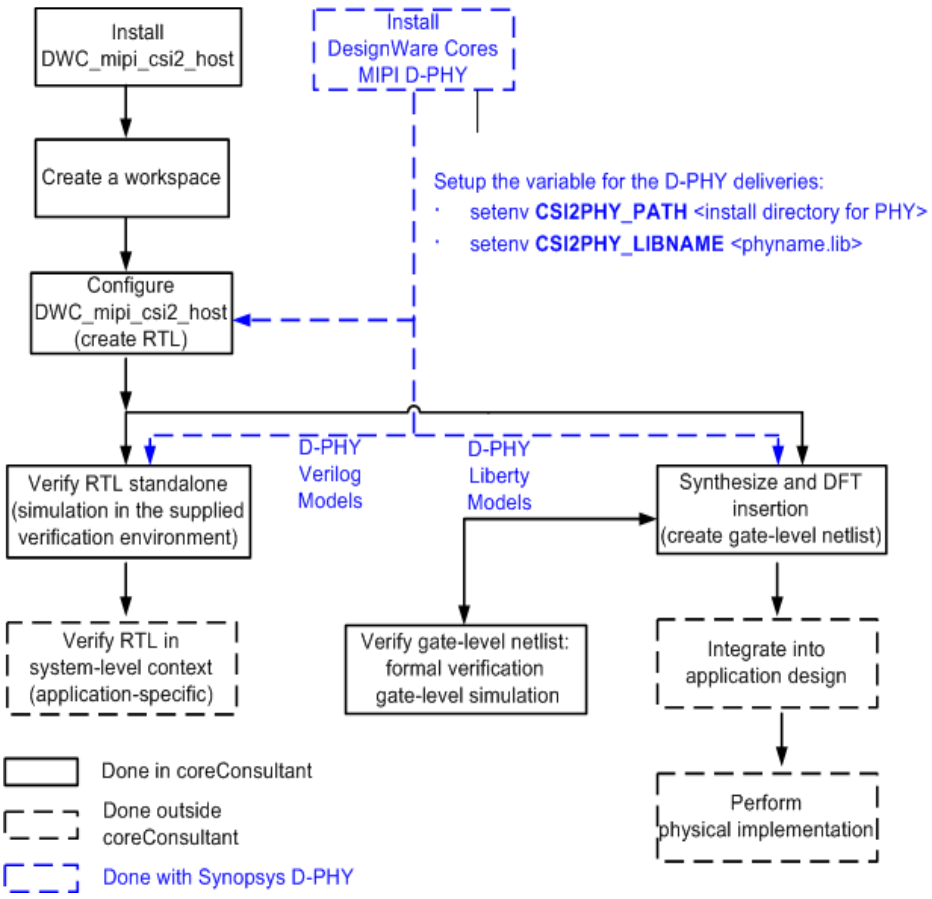
To configure your core using coreConsultant, follow these steps:

1. Specify your configuration - Select options to enable or disable features as appropriate for your design. You can use default values for an initial simulation and synthesis trial. However, you must select or enter specific values required to implement your design.
2. Generate RTL - After you have specified values for all configuration options, click Apply to generate configured RTL code for DWC_mipi_csi2_host. coreConsultant then checks your parameter values, generates configured RTL code in the <workspace>/src/ directory, and displays a configuration report.
3. Reconfigure your core - You can return to the Specify Configuration activity to reconfigure DWC_mipi_csi2_host (create new RTL) at any time. If you do so, you must re-complete any downstream activities.

4.2 Basic Design Flow

Figure 4-1 shows the basic sequence of tasks.The coreConsultant GUI guides you through the DWC_mipi_csi2_host design flow in your workspace. The design flow includes the default set of coreConsultant design activities described in the coreConsultant User Guide (available through the coreConsultant Help facility), plus activities specific to the DWC_mipi_csi2_host core that are described in this chapter.

Figure 4-1 DWC_mipi_csi2_host Design Flow



4.3 Support for D-PHY integration

As previously stated the DWC_mipi_csi2_host can perform D-PHY integration whenever there is available a Synopsys D-PHY delivery. The coreConsultant flow uses files provided in Synopsys D-PHY to perform:

- Functional validation using D-PHY Verilog behavioral models
- Synthesis timing closure using the D-PHY liberty timing models

After downloading and installing the Synopsys D-PHY you need to setup the following variable in your environment to allow coreConsultant to locate the required files:

- Set up the variable CSI2PHY_PATH to locate required PHY files as follows:

  ```
  setenv CSI2PHY_PATH <install directory for PHY>
  ```

- Set up the variable CSI2PHY_LIB. The general name of the .lib file is mipi_x<n>_dphy + process corner + .lib

  ```
  setenv CSI2PHY_LIB <phyname.lib>
  ```

The DWC_mipi_csi2_host can also be configured to interface with other D-PHY models. If you choose not to use a Synopsys D-PHY, coreConsultant configures the RTL with a PPI standard interface. For validation, proposed models for the D-PHY are instantiated directly on the Verilog testbench.

You can execute the entire DWC_mipi_csi2_host design flow through coreConsultant, as shown in Figure 4-2, except for the following application-specific activities:

- Integrating the DWC_mipi_csi2_host into your application design
- Verifying the DWC_mipi_csi2_host in the context of your application design

4.4 Invoking coreConsultant and Creating a Workspace

A workspace is a local copy of your DWC_mipi_csi2_host installation in which you can configure, verify, and synthesize your own DWC_mipi_csi2_host implementation. After you install, you must create a workspace to begin working. You can create several workspaces so that you can experiment with different design alternatives.

To create a workspace:

1. Invoke coreConsultant:

   ```
   % coreConsultant &
   ```

2. Select File > New Workspace in the coreConsultant console, then enter the requested information in the New Workspace dialog. For more information about answering the New Workspace options, see the coreConsultant online help.

4.5 Configuration: Creating the RTL

The coreConsultant configuration tool enables you to configure the core and write out RTL according to your application requirements. Make sure that you understand the definition of each parameter and select the default configuration only when it is suitable for your application/chip. Figure 4-2 shows the Specify Configuration activity selected in coreConsultant.

The Specify Configuration activity provides an interactive mechanism for configuration:

- The parameters are grouped by common function and labeled with English language expressions for the parameters functions.
- You can access detailed information about each parameter by right-clicking on the parameter label and selecting What's This or by selecting the Help tab.

- The coreConsultant enforces the parameter interdependencies interactively. For example, when you select 'Use SNPS PHY', coreConsultant enables the parameter selection for 'Use BIDIR phy'.

Note Use the Set Design Prefix activity when you plan to instantiate DWC_mipi_csi2_host more than once in your design. The default state of Set Design Prefix is completed because, for most DWC_mipi_csi2_host users, it is not a required activity.

To configure the DWC_mipi_csi2_host interactively, go to the Specify Configuration dialog and select your configuration options. There are options available to enable or disable certain features. You can use the default values for an initial simulation and synthesis trial. Otherwise, you must select or enter the values required for your design.

See Table 4-1 for detailed descriptions of all configuration options. After you have specified values for all configuration options, click Apply to generate configured RTL code for DWC_mipi_csi2_host. coreConsultant then checks your parameter values, generates configured RTL code in the <workspace>/src/ directory, and displays a configuration report. You can return to the Specify Configuration activity to reconfigure DWC_mipi_csi2_host (create new RTL) at any time. If you do so, you need to re-complete any downstream activities.

Figure 4-2 The coreConsultant Specify Configuration Dialog

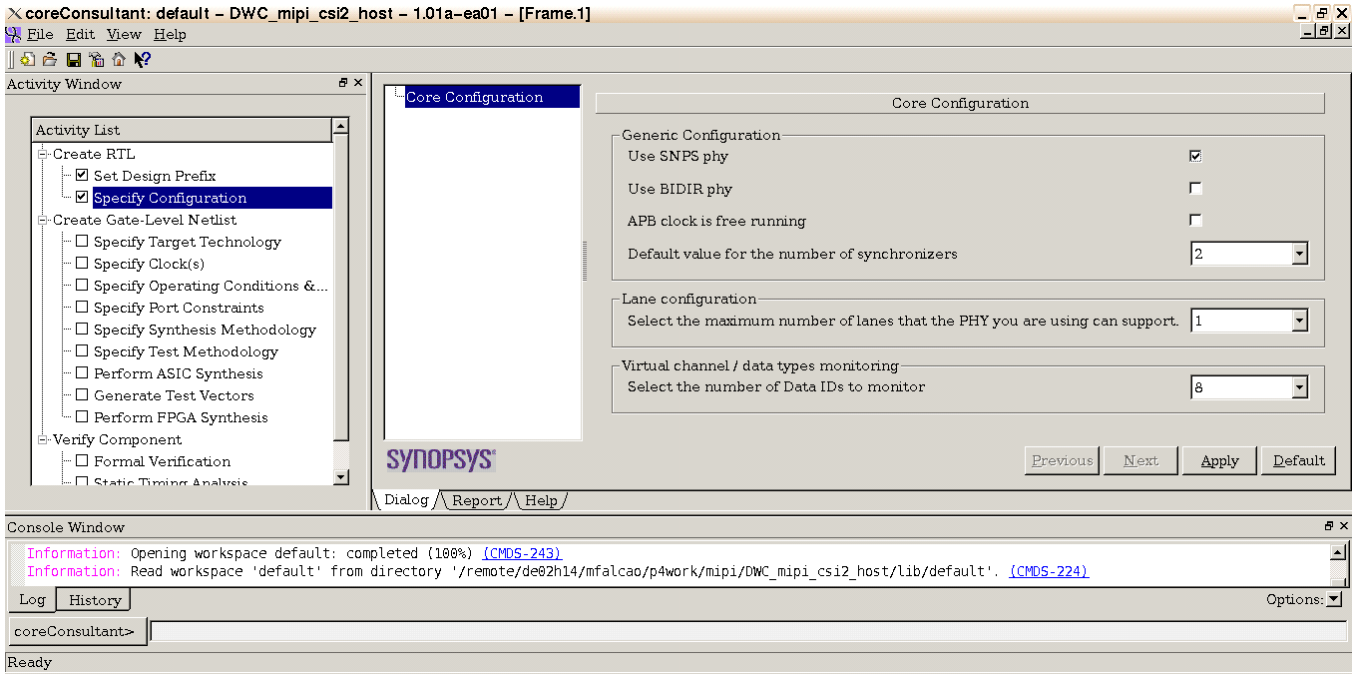


Table 4-1 DWC_mipi_csi2_Host Configuration Parameters

Label	Parameter Definition
Use SNPS phy	**Parameter Name:** CSI2_HOST_SNPS_PHY **Values:** 0, 1 **Default Value:** 0 **Dependencies:** N/A **Description:** By selecting this you want to use the SNPS PHY as an internal submodule in the design. This implies you have access to the PHY package. You need to setup the following variables: • setenv CSI2PHY_PATH <path where PHY is installed> • setenv CSI2PHY_LIB <.lib file name>
Use BIDIR phy	**Parameter Name:** CSI2_HOST_BIDIR_PHY **Values:** 0, 1 **Default Value:** 0 **Dependencies:** N/A **Enabled:** CSI2_HOST_SNPS_PHY **Description:** Only selectable when you have chose to use the SNPS PHY. Select this if you want to use the SNPS BIDIR D-PHY. Do not select this if you want to use the SNPS SLave D-PHY.
APB clock is free running	**Parameter Name:** CSI2_PCLK_FREE **Values:** 0, 1 **Default Value:** 0 **Dependencies:** N/A **Description:** Select if PCLK is a free running or gated clock. If not set, an additional input pin will be created. This pin needs to receive a free running version of PCLK.
Default value for the number of synchronizers	**Parameter Name:** CSI2_HOST_DFLT_F_SYNC_TYPE **Values:** 2, 3, and 4 **Default Value:** 2 **Dependencies:** N/A **Description:** Select the number of synchronization stages used for clock domain crossing. All stages are capturing data on the rising edge of the clock.
Select the maximum number of lanes that the PHY you are using can support	**Parameter Name:** CSI2_HOST_NUMBER_OF_LANES **Values:** 1 -4 **Default Value:** 1 **Dependencies:** N/A **Description:** Select the maximum number of lanes to be supported.
Select the number of Data IDs to monitor	**Parameter Name:** CSI2_HOST_N_DATA_IDS **Values:** None, 4, 8 (0, 1, 2) **Default Value:** None **Dependencies:** N/A **Description:** Select the number of Data IDs (Virtual Channel / Data type) for which the design is able to detect errors related to line boundary packets.

5 Signals

This chapter helps you to understand MIPI CSI-2 signals and their properties. It describes the naming conventions, I/O mapping, width, dependencies, and their behavior with various interfaces.

The topics include:

5.1 Naming and Description Conventions

5.1.1 Signal Name

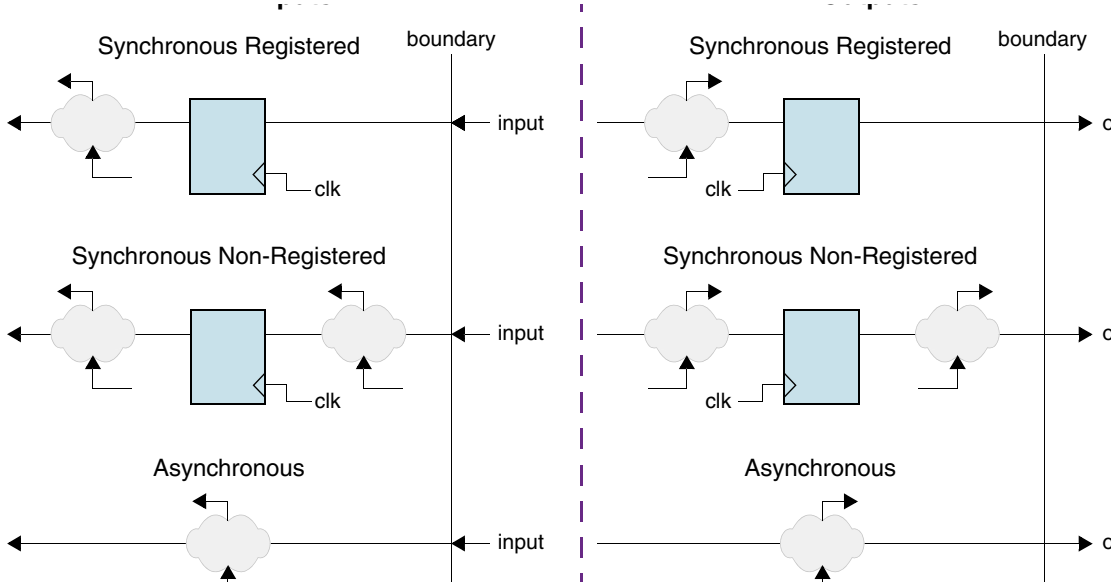
Signal have the same names as in AMBA and D-PHY specifications for APB and PPI interfaces.

5.1.2 Signal Description

The signal name description describes the function of each signal, and the type, that can be:

- **Synchronous:** The signal is asserted or deasserted with respect to a clock edge.
- **Asynchronous:** The signal is not asserted or deasserted with respect to a clock edge.
- **Registered:** The signal is captured (or launched) directly at the macro boundary with no intermediate logic between the core boundary and the capturing (or launching) flip-flop.

Figure 5-1 Synchronous and Asynchronous signals



5.2 Signal Descriptions

The following sections provide a detailed description of the DWC_mipi_csi2_host signals.

5.2.1 Image Data Interface (IDI) Signals

The Image Data Interface is a 32-bit Synopsys proprietary interface that delivers Image data on csi_data[31:0] ready for Memory storage as proposed in chapter 12 of CSI-2 specification.

Since more than one video sequence can be conveyed on CSI-2 link, additional information is provided related to data type data_type[5:0] and Virtual Channel virtual_channel[1:0]. This information that is provided on the CSI-2 packet header is maintained through out the complete line transmission.

Timing accurate video synchronization signals are also provided for proper image reconstruction through vvalid[3:0], hvalid[3:0], and dvalid[3:0], where the signal's index identifies the virtual channel that it relates to.

Figure 5-2 Image Data Interface Signals

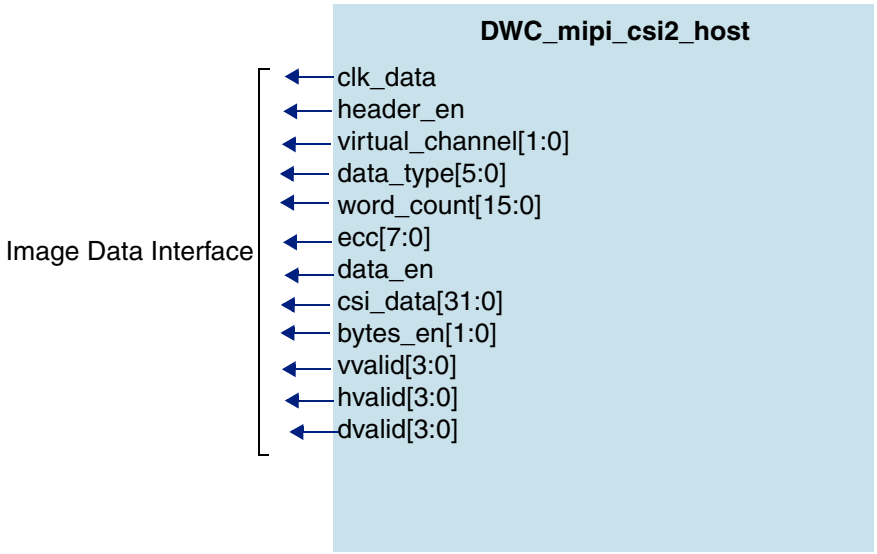


Table 5-1 Image Data Interface Signal Description

Name	Width	I/O	Description
clk_data	1 bit	O	Clock output for IDI. All signals are synchronous with this clock. This clock can vary between 10 MHz (D-PHY lane operating at minimum speed - 80Mbps) and 125 MHz (D-PHY lane operating at maximum speed - 1Gbps). **Active State**: N/A **Registered**: N/A **Synchronous to**: Follows rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None
header_en	1 bit	O	Informs that the header data at the output (signals virtual_channel, data_type, word_count, and ecc) is valid for the packet being transferred. This signal stays HIGH during the complete packet transfer. **Active State**: High **Registered**: Yes **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None
virtual_channel [1:0]	2 bits	O	Virtual Channel Identifier (VC) value, which is part of the Data Identifier (DI) byte: • 00: Virtual Channel 0 • 01: Virtual Channel 1 • 10: Virtual Channel 2 • 11: Virtual Channel 3 **Active State**: N/A **Registered**: Yes **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None
data_type[5:0]	6 bits	O	Data Type (DT) value, which is part of the Data Identifier (DI) byte: • 0x00 - 0x07: Synchronization Short Packet Data Types • 0x08 - 0x0F: Generic Short Packet Data Types • 0x10 - 0x17: Generic Long Packet Data Types • 0x18 - 0x1F: YUV Data • 0x20 - 0x27: RGB Data • 0x28 - 0x2F: RAW Data • 0x30 - 0x37: User Defined Byte-based Data • 0x38 - 0x3F: Reserved **Active State**: N/A **Registered**: Yes **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None

Table 5-1 Image Data Interface Signal Description (Continued)

Name	Width	I/O	Description
word_count[15:0]	16 bits	O	16-bit Word Count (WC) information from the Packet Header. This signal indicates the number of bytes and remains stable during the entire packet transfer. **Active State**: N/A **Registered**: Yes **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None
ecc[7:0]	8 bits	O	8-bit Error Correction Code (ECC) for the Packet Header. **Active State**: N/A **Registered**: Yes **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None
data_en	1 bit	O	Informs that new payload data is present at the output (signals csi_data and bytes_en). Signal header_en is also asserted when data_en is asserted. This signal is only asserted when receiving long packets. **Active State**: High **Registered**: Yes **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None
csi_data[31:0]	32 bits	O	Payload data output, formatted according to CSI-2 recommended data storage format. **Active State**: N/A **Registered**: Yes **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None
bytes_en[1:0]	2 bits	O	Informs how many bytes in the csi_data output signal are valid bytes: • 00: 1 valid byte in csi_data[7:0] • 01: 2 valid bytes in csi_data[15:0] • 10: 3 valid bytes in csi_data[23:0] • 11: 4 valid bytes in csi_data[31:0] **Active State**: N/A **Registered**: Yes **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None

Table 5-1 Image Data Interface Signal Description (Continued)

Name	Width	I/O	Description
vvalid[3:0]	4 bits	O	This signal is asserted when Frame Start is detected, and de-asserted when Frame End is detected. There is one valid signal per Virtual Channel: • vvalid[0]: vvalid for Virtual Channel 0 • vvalid[1]: vvalid for Virtual Channel 1 • vvalid[2]: vvalid for Virtual Channel 2 • vvalid[3]: vvalid for Virtual Channel 3 **Active State**: High, per bit **Registered**: Yes **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None
hvalid[3:0]	4 bits	O	This signal is asserted when Line Start is detected, and de-asserted when Line End is detected. Since Line Start and Line End are optional, in case they are not available, hvalid has same behavior as dvalid. There is one valid signal per Virtual Channel: • hvalid[0]: hvalid for Virtual Channel 0 • hvalid[1]: hvalid for Virtual Channel 1 • hvalid[2]: hvalid for Virtual Channel 2 • hvalid[3]: hvalid for Virtual Channel 3 **Active State**: High, per bit **Registered**: Yes **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None
dvalid[3:0]	4 bits	O	Used to signal when valid data is available. There is one valid signal per Virtual Channel: • dvalid[0]: dvalid for Virtual Channel 0 • dvalid[1]: dvalid for Virtual Channel 1 • dvalid[2]: dvalid for Virtual Channel 2 • dvalid[3]: dvalid for Virtual Channel 3 **Active State**: High, per bit **Registered**: Yes **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None

5.2.2 AMBA Slave Interface Signals

The AMBA APB slave interface module is compatible with the AMBA 3 APB Protocol Specification, revision 1.0. The APB slave interface is used in the DWC_mipi_csi2_host for register configuration. According to the APB specification, all signal transitions are only related to the rising edge of the clock to enable the integration of APB peripherals easily into any design flow. Every transfer takes at least two cycles.

Figure 5-3 AMBA Slave Interface Signals

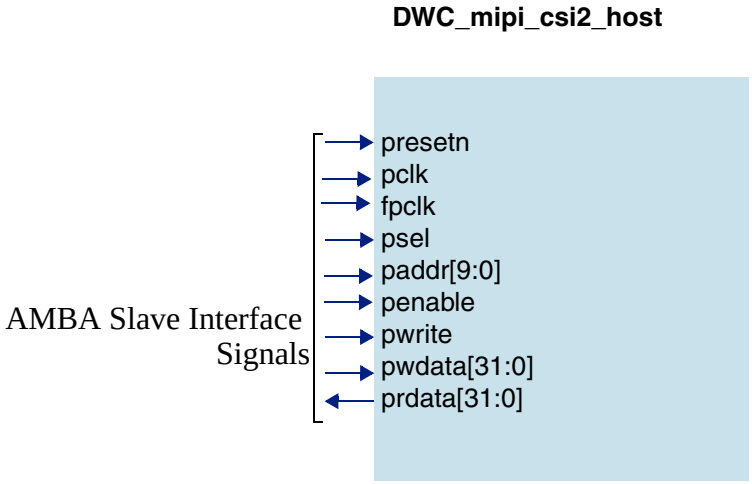


Table 5-2 AMBA Slave Interface Signals

Name	Width	I/O	Description
presetn	1 bit	I	Global reset of the controller including all registers. Active LOW. **Active State**: Low **Registered**: N/A **Synchronous to**: Rising edge must be synchronous to pclk **External Input Delay**: N/A **Dependencies**: None
pclk	1 bit	I	APB bus clock. The rising edge of pclk is used to time all transfers on the APB bus. The minimum required frequency is 15 MHz. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None

Table 5-2 AMBA Slave Interface Signals

Name	Width	I/O	Description
fpclk	1 bit	I	(OPTIONAL) Free running version of APB bus clock pclk. In case clock signal pclk is not always available during normal operation (for example, gating of pclk when not using APB interface, in order to save power consumption), then a free running version of pclk clock is required. If pclk is a free running clock, then this input is not required and should be removed via the associated coreConsultant option. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: Undefine CSI2_PCLK_FREE
psel	1 bit	I	APB select **Active State**: High **Registered**: N/A **Synchronous to**: pclk **External Input Delay**: N/A **Dependencies**: None
paddr[9:0]	10 bit	I	APB address bus **Active State**: N/A **Registered**: N/A **Synchronous to**: pclk **External Input Delay**: N/A **Dependencies**: None
penable	1 bit	I	APB strobe **Active State**: High **Registered**: N/A **Synchronous to**: pclk **External Input Delay**: N/A **Dependencies**: None
pwrite	1 bit	I	APB transfer direction. When HIGH this signal indicates an APB write access, and when LOW a read access. **Active State**: High **Registered**: N/A **Synchronous to**: pclk **External Input Delay**: N/A **Dependencies**: None
pwdata[31:0]	32 bits	I	APB write data bus **Active State**: N/A **Registered**: N/A **Synchronous to**: pclk **External Input Delay**: N/A **Dependencies**: None

Table 5-2 AMBA Slave Interface Signals

Name	Width	I/O	Description
prdata[31:0]	32 bits	O	APB read data bus **Active State**: N/A **Registered**: Yes **Synchronous to**: pclk **External Input Delay**: N/A **Dependencies**: None

5.2.3 Interrupt Signals

Figure 5-4 Interrupt Signals

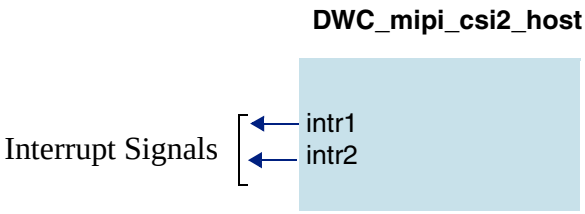


Table 5-3 Interrupt Signals

Name	Width	I/O	Description
intr1	1 bit	O	Interrupt function 1. This signal is synchronous with AMBA APB bus clock (pclk). **Active State**: High **Registered**: Yes **Synchronous to**: fpclk/pclk **External Input Delay**: N/A **Dependencies**: None
intr2	1 bit	O	Interrupt function 2. This signal is synchronous with AMBA APB bus clock (pclk). **Active State**: High **Registered**: Yes **Synchronous to**: fpclk/pclk **External Input Delay**: N/A **Dependencies**: None

5.2.4 Scan Chain Signals

Figure 5-5 Scan Chain Signals

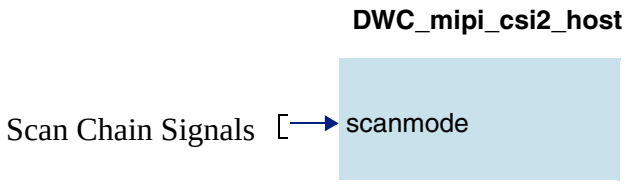


Table 5-4 Scan Chain Signals

Name	Width	I/O	Description
scanmode	1 bit	I	Scan Mode selection. Assert HIGH to configure the controller for Scan-Chain operation. Assert LOW for normal operation. **Active State**: High **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None

5.2.5 Interface with D-PHY Clock Lane Following PHY Protocol Interface (PPI) Signals

Sections 5.2.5 through 5.2.11 describe the signals that interface with the D-PHY by a standard PPI interface. Signals on these tables only appear as pins on the top RTL macro, if on coreConsultant GUI no Synopsys D-PHY is selected. This option is available when integration is done without a Synopsys D-PHY.

When configuring the core, there is also an option to select the number of Lane in the core. A Clock Lane and a Data Lane (Lane0) with TX HS features are always required. Data Lanes can be up to a maximum of 4. Pins associated with the Lanes that are not configured are also removed.

Figure 5-6 Interface with D-PHY Clock Lane Following PHY Protocol Interface Signals

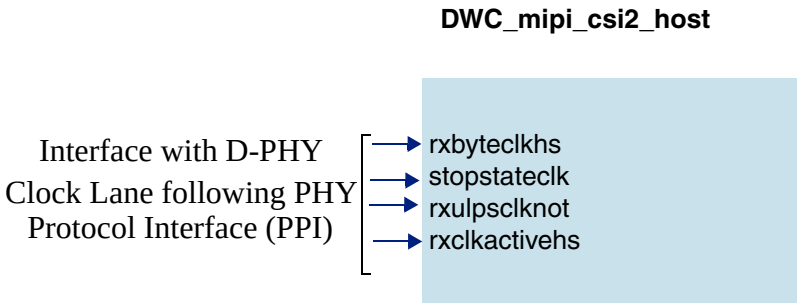


Table 5-5 Interface With D-PHY Clock Lane Following PHY Protocol Interface Signals Description

Name	Width	I/O	Description
rxbyteclkhs	1 bit	I	High Speed Receive byte clock. This is used to synchronize PPI signals in the High-Speed receive clock domain. The frequency of this clock is 1/8th of the PHY bit rate. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
stopstateclk	1 bit	I	Clock Lane in Stop state. This signal indicates that the clock lane is in Stop state, and is asynchronous to any clock in the PPI interface. **Active State**: High **Registered**: N/A **Synchronous to**: Asynchronous **External Input Delay**: N/A **Dependencies**: None
rxulpsclknot	1 bit	I	This signal indicates that the Clock Lane has entered Ultra-Low Power State. This signal is kept LOW until a Stop State is sent or detected on the Lane interconnect. **Active State**: Low **Registered**: N/A **Synchronous to**: Asynchronous **External Input Delay**: N/A **Dependencies**: None
rxclkactivehs	1 bit	I	Indicates that the Clock Lane is actively receiving a DDR clock. **Active State**: High **Registered**: N/A **Synchronous to**: Asynchronous **External Input Delay**: N/A **Dependencies**: None

5.2.6 Interface with D-PHY Data Lane 0 Following PHY Protocol Interface (PPI) Signals

Figure 5-7 Interface with D-PHY Data Lane 0 Following PHY Protocol Interface (PPI) Signals

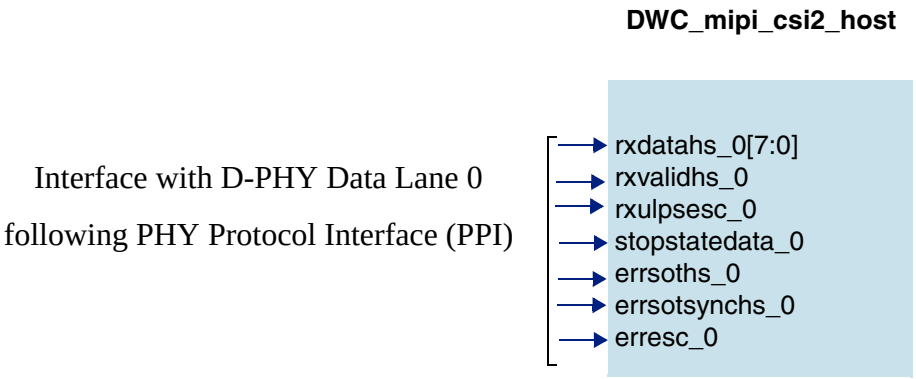


Table 5-6 Interface with D-PHY Data Lane 0 Following PHY Protocol Interface (PPI) Signals

Name	Width	I/O	Description
rxdatahs_0[7:0]	8 bits	I	High-Speed receive data. Bit 0 was received first in the lane interconnect. Data is transferred on the rising edges of rxbyteclkhs. **Active State**: N/A **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None
rxvalidhs_0	1 bit	I	High-Speed receive data valid. This signal indicates that the lane module is driving valid data to protocol layer on the rxdatahs_0 bus. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None

Table 5-6 Interface with D-PHY Data Lane 0 Following PHY Protocol Interface (PPI) Signals

Name	Width	I/O	Description
rxulpsesc_0	1 bit	I	Escape Ultra-Low Power Receive mode. This signal is asserted to indicate that the lane module has entered the Ultra-Low Power State. The lane module remains in this mode with rxulpsesc_0 asserted until a Stop state is detected on the lane interconnect. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None
stopstatedata_0	1 bit	I	Lane is in Stop State. This signal is asynchronous to any clock in the PPI interface. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None
errsoths_0	1 bit	I	Start of Transmission Error. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None
errsotsynchs_0	1 bit	I	Start of Transmission Synchronization Error. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None
erresc_0	1 bit	I	Escape Entry Error. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: None

5.2.7 Interface with D-PHY Data Lane 1 Following PHY Protocol Interface (PPI) Signals

Figure 5-8 Interface with D-PHY Data Lane 1 Following PHY Protocol Interface (PPI)

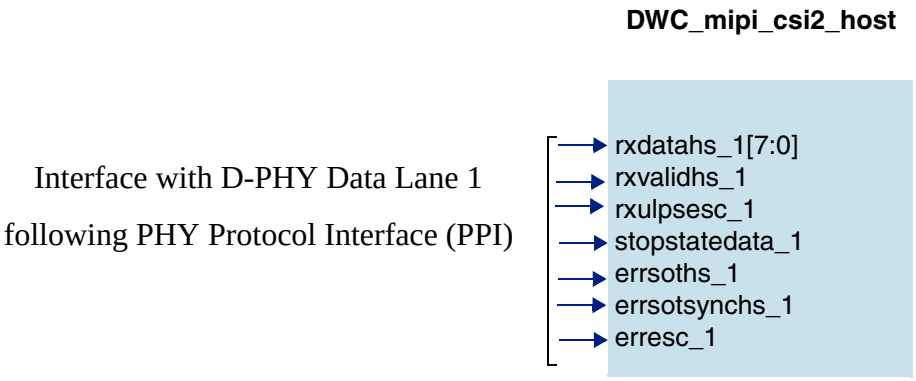


Table 5-7 Interface with D-PHY Data Lane 1 Following PHY Protocol Interface (PPI)

Name	Width	I/O	Description
rxdatahs_1[7:0]	8 bit	I	High-Speed receive data. Bit 0 was received first in the lane interconnect. Data is transferred on the rising edges of rxbyteclkhs. **Active State**: N/A **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_2
rxvalidhs_1	1 bit	I	High-Speed receive data valid. This signal indicates that the lane module is driving valid data to protocol layer on the rxdatahs_0 bus. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_2

Table 5-7 Interface with D-PHY Data Lane 1 Following PHY Protocol Interface (PPI) (Continued)

Name	Width	I/O	Description
rxulpsesc_1	1 bit	I	Escape Ultra-Low Power Receive mode. This signal is asserted to indicate that the lane module has entered the Ultra-Low Power State. The lane module remains in this mode with rxulpsesc_1 asserted until a Stop state is detected on the lane interconnect. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_2
stopstatedata_1	1 bit	I	Lane is in Stop State. This signal is asynchronous to any clock in the PPI interface. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_2
errsoths_1	1 bit	I	Start of Transmission Error. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_2
errsotsynchs_1	1 bit	I	Start of Transmission Synchronization Error. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_2
erresc_1	1 bit	I	Escape Entry Error. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_2

5.2.8 Interface with D-PHY Data Lane 2 Following PHY Protocol Interface (PPI) Signals

Figure 5-9 Interface with D-PHY Data Lane 2 Following PHY Protocol Interface (PPI)

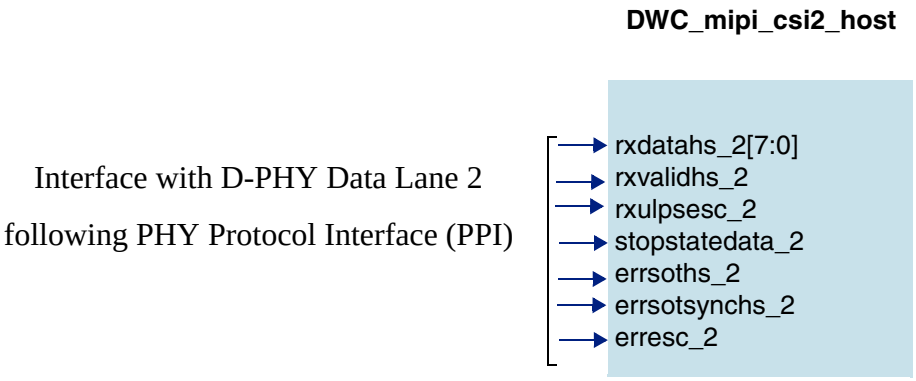


Table 5-8 Interface with D-PHY Data Lane 2 Following PHY Protocol Interface (PPI)

Name	Width	I/O	Description
rxdatahs_2[7:0]	8 bits	I	High-Speed receive data. Bit 0 was received first in the lane interconnect. Data is transferred on the rising edges of rxbyteclkhs. **Active State**: N/A **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_3
rxvalidhs_2	1 bit	I	High-Speed receive data valid. This signal indicates that the lane module is driving valid data to protocol layer on the rxdatahs_0 bus. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_3

Table 5-8 Interface with D-PHY Data Lane 2 Following PHY Protocol Interface (PPI) (Continued)

Name	Width	I/O	Description
rxulpsesc_2	1 bit	I	Escape Ultra-Low Power Receive mode. This signal is asserted to indicate that the lane module has entered the Ultra-Low Power State. The lane module remains in this mode with rxulpsesc_0 asserted until a Stop state is detected on the lane interconnect. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_3
stopstatedata_2	1 bit	I	Lane is in Stop State. This signal is asynchronous to any clock in the PPI interface. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_3
errsoths_2	1 bit	I	Start of Transmission Error. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_3
errsotsynchs_2	1 bit	I	Start of Transmission Synchronization Error. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_3
erresc_2	1 bit	I	Escape Entry Error. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_3

5.2.9 Interface with D-PHY Data Lane 3 Following PHY Protocol Interface (PPI) Signals

Figure 5-10 Interface with D-PHY Data Lane 3 Following PHY Protocol Interface (PPI)

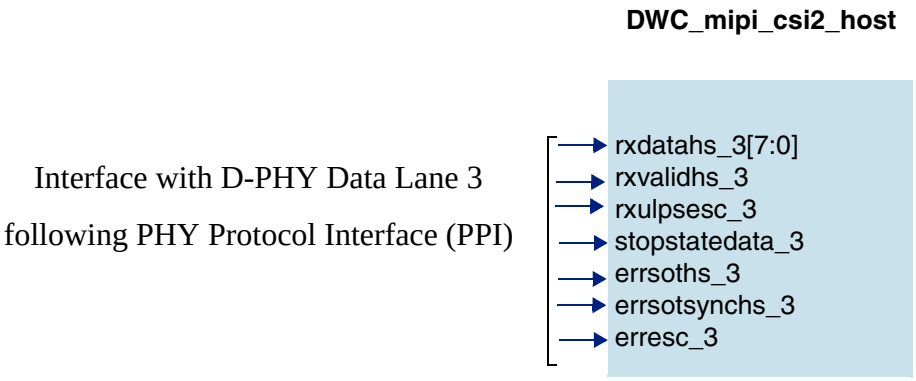


Table 5-9 Interface with D-PHY Data Lane 3 Following PHY Protocol Interface (PPI)

Name	Width	I/O	Description
rxdatahs_3[7:0]	8 bits	I	High-Speed receive data. Bit 0 was received first in the lane interconnect. Data is transferred on the rising edges of rxbyteclkhs. **Active State**: N/A **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_4
rxvalidhs_3	1 bit	I	High-Speed receive data valid. This signal indicates that the lane module is driving valid data to protocol layer on the rxdatahs_0 bus. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_4

Table 5-9 Interface with D-PHY Data Lane 3 Following PHY Protocol Interface (PPI) (Continued)

Name	Width	I/O	Description
rxulpsesc_3	1 bit	I	Escape Ultra-Low Power Receive mode. This signal is asserted to indicate that the lane module has entered the Ultra-Low Power State. The lane module remains in this mode with rxulpsesc_0 asserted until a Stop state is detected on the lane interconnect. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_4
stopstatedata_3	1 bit	I	Lane is in Stop State. This signal is asynchronous to any clock in the PPI interface. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_4
errsoths_3	1 bit	I	Start of Transmission Error. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_4
errsotsynchs_3	1 bit	I	Start of Transmission Synchronization Error. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_4
erresc_3	1 bit	I	Escape Entry Error. **Active State**: High **Registered**: N/A **Synchronous to**: rxbyteclkhs **External Input Delay**: N/A **Dependencies**: Define CSI2_HOST_INCLUDE_LANE_4

5.2.10 D-PHY Control Signals

Figure 5-11 D-PHY Control Signals

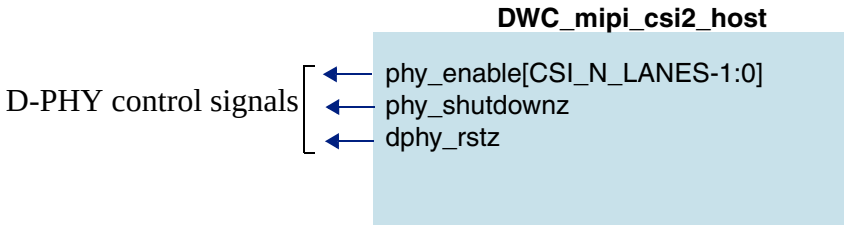


Table 5-10 D-PHY Control Signals

Name	Width	I/O	Description
phy_enable[CSI_N_LANES-1:0]	CSI_N_LANES	O	D-PHY data lane enable signal: • phy_enable [0]: Enable for lane 0. • phy_enable [1]: Enable for lane 1, if existent. • phy_enable [2]: Enable for lane 2, if existent. • finagle [3]: Enable for lane 3, if existent. **Active State**: High, per bit **Registered**: Yes **Synchronous to**: pclk **External Input Delay**: N/A **Dependencies**: Width depends on the number of lanes
phy_shutdownz	1 bit	O	D-PHY shutdown signal. This signal is directly controlled by register PHY_SHUTDOWNZ. **Active State**: Low **Registered**: Yes **Synchronous to**: pclk **External Input Delay**: N/A **Dependencies**: None
dphy_rstz	1 bit	O	D-PHY reset signal. This signal is directly controlled by register DPHY_RSTZ. **Active State**: Low **Registered**: Yes **Synchronous to**: pclk **External Input Delay**: N/A **Dependencies**: None

5.2.11 Parallel Port for PHY Configuration Signals

The signals described in this section configure a parallel port for general purpose configuration of a D-PHY. These pins are directly mapped on registers PHY_TST_CRTL0 and PHY_TST_CTRL1.

Figure 5-12 Parallel Port for PHY Configuration Signals

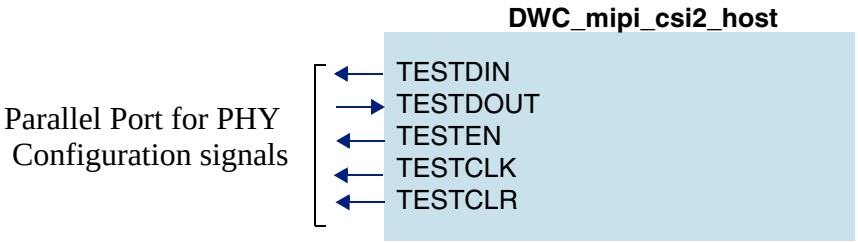


Table 5-11 Parallel Port for PHY Configuration Signals

Name	Width	I/O	Description
TESTDIN	<7:0>	O	D-PHY test data output port. **Active State**: N/A **Registered**: Yes **Synchronous to**: pclk **External Input Delay**: N/A **Dependencies**: None
TESTDOUT	<7:0>	I	D-PHY test data input port. **Active State**: N/A **Registered**: Yes **Synchronous to**: pclk **External Input Delay**: N/A **Dependencies**: None
TESTEN	1 bit	O	D-PHY test enable. **Active State**: High **Registered**: N/A **Synchronous to**: pclk **External Input Delay**: N/A **Dependencies**: None
TESTCLK	1 bit	O	D-PHY test clock signal. **Active State**: Low **Registered**: Yes **Synchronous to**: pclk **External Input Delay**: N/A **Dependencies**: None

Table 5-11 Parallel Port for PHY Configuration Signals

Name	Width	I/O	Description
TESTCLR	1 bit	O	D-PHY test clear signal, active high. **Active State**: Low **Registered**: Yes **Synchronous to**: pclk **External Input Delay**: N/A **Dependencies**: None

5.2.12 D-PHY External Signals

Table 5-12 shows D-PHY pin list with the pins expected when an internal Synopsys D-PHY is selected. The PPI interface described in Table 16 is handled inside the core and the connections between the D-PHY and the CSI-2 protocol layer no longer exist at core pinout.

Similar to what is presented for the for PPI interface, only the Lane differential signal configured in coreConsultant will appear as pins.

The Scan interface for the D-PHY is dependent on the number of lane presented on the Synopsys PHY.

Figure 5-13 D-PHY External Signals

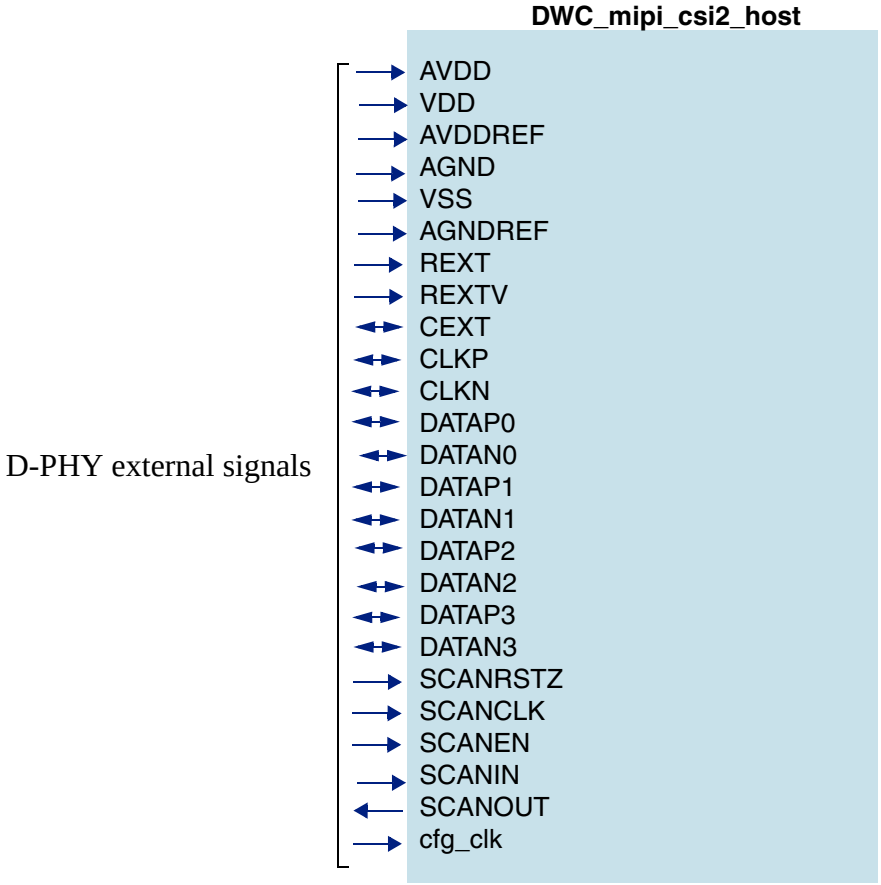


Table 5-12 D-PHY External Signals

Name	Width	I/O	Description
AVDD	1 bit	I	D-PHY analog power supply. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
VDD	1 bit	I	D-PHY digital power supply. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
AVDDREF	1 bit	I	D-PHY analog supply for reference generator. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
AGND	1 bit	I	D-PHY analog supply ground return. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
VSS	1 bit	I	D-PHY digital supply ground return. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
AGNDREF	1 bit	I	D-PHY analog supply ground return for reference generator. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None

Table 5-12 D-PHY External Signals

Name	Width	I/O	Description
REXT	1 bit	I	D-PHY external resistor connection (REXT and REXTV should be shorted at the chip pad connection). **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
REXTV	1 bit	I	D-PHY external resistor connection (REXT and REXTV should be shorted at the chip pad connection). **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
CEXT	1 bit	I	D-PHY external connection for integrated LDO (only applicable to some SNPS D-PHYs). **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
CLKP	1 bit	I/O	Positive D-Phy differential clock line transceiver output. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
CLKN	1 bit	I/O	Negative D-Phy differential clock line transceiver output. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
DATAP0	1 bit	I/O	Positive D-Phy differential data line transceiver output, Lane 0. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None

Table 5-12 D-PHY External Signals

Name	Width	I/O	Description
DATAN0	1 bit	I/O	Negative D-Phy differential data line transceiver output, Lane 0. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
DATAP1	1 bit	I/O	Positive D-Phy differential data line transceiver output, Lane 1. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
DATAN1	1 bit	I/O	Negative D-Phy differential data line transceiver output, Lane 1. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
DATAP2	1 bit	I/O	Positive D-Phy differential data line transceiver output, Lane 2. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
DATAN2	1 bit	I/O	Negative D-Phy differential data line transceiver output, Lane 2. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
DATAP3	1 bit	I/O	Positive D-Phy differential data line transceiver output, Lane 3. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None

Table 5-12 D-PHY External Signals

Name	Width	I/O	Description
DATAN3	1 bit	I/O	Negative D-Phy differential data line transceiver output, Lane 3. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
SCANRSTZ	1 bit	I	D-PHY reset signal for scan mode. Active Low. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
SCANCLK	1 bit	I	D-PHY scan Clock source for scan mode. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
SCANEN	1 bit	I	D-PHY scan enable: • 1 – shift mode • 0 – capture mode **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
SCANIN	N_SCAN_CHAINS	I	D-PHY scan in serial data stream input, when macro is in Scan mode. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None
SCANOUT	N_SCAN_CHAINS	I	D-PHY scan out serial data stream output, when macro is in Scan mode. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None

Table 5-12 D-PHY External Signals

Name	Width	I/O	Description
cfg_clk	1 bit	I	D-PHY configuration clock used for the initialization of the PHY. It is also used for exiting ULPS state. **Active State**: N/A **Registered**: N/A **Synchronous to**: N/A **External Input Delay**: N/A **Dependencies**: None

6 Software Registers

The following subsections describe the MIPI CSI-2 software registers:

6.1 Register Memory Map

All registers are addressable on 32-bit boundaries; each unused bit or address location is reserved for future use and read back as 0.

Table 6-1 summarizes the register memory map for MIPI CSI-2 .

Table 6-1 MIPI CSI-2 Memory Map

Name	Address Offset	Width	R/W	Description	Value after Reset
VERSION	0x000	32 bits	R	Version of the CSI-2 host controller	CSI_VERSION_ID
N_LANES	0x004	2 bits	R/W	Number of active data lanes	CSI_N_LANES
PHY_SHUTDOWNZ	0x008	1 bit	R/W	PHY shutdown control	0
DPHY_RSTZ	0x00C	1 bit	R/W	DPHY reset control	0
CSI2_RESETN	0x010	1 bit	R/W	CSI-2 controller reset	0
PHY_STATE	0x014	12 bits	R	General settings for all blocks	0
DATA_IDS_1	0x018	32 bits	R/W	List of Data Ids for which IDI reports line boundary matching errors (CSI2_HOST_N_DATA_IDS > 0)	0
DATA_IDS_2	0x01C	32 bits	R/W	List of Data Ids for which IDI reports line boundary matching errors (CSI2_HOST_N_DATA_IDS > 1)	0
ERR1	0x020	29 bits	R	Error state register 1	0
ERR2	0x024	24 bits	R	Error state register 2	0
MASK1	0x028	29 bits	R/W	Masks for errors 1	0
MASK2	0x02C	24 bits	R/W	Masks for errors 2	0
PHY_TST_CRTL0	0x030	2 bits	R/W	D-PHY Test interface control 0	0
PHY_TST_CRTL1	0x034	17 bits	R/W	D-PHY Test interface control 1	0

6.2 Register and Field Descriptions

The following subsections describe the data fields of the MIPI CSI-2 registers.

6.2.1 VERSION

- **Name:** Controller Version Identification
- **Address Offset:** 0x000
- **Size:** 32 bits
- **Value after Reset:** CSI_VERSION_ID
- **Access:** Read

Table 6-2 Controller Version Identification Register

Bits	Name	R/W	Description
31:0	VERSION	R	Version of the CSI-2 Host Controller

6.2.2 N_LANES

- **Name:** Number of Active Data Lanes

Note If the core configuration is only with 1 Lane, this register gets removed.

- **Address Offset:** 0x004
- **Size:** 2 bits
- **Value after Reset:** CSI_N_LANES
- **Access:** Read/Write

Table 6-3 Number of Active Data Lanes Register

Bits	Name	R/W	Description
1:0	N_LANES	R/W	Number of Active Data Lanes. • 00: 1 Data Lane (Lane 0) • 01: 2 Data Lanes (Lane 0, and 1) • 10: 3 Data Lanes (Lane 0,1 and 2) • 11: 4 Data Lanes (All) Can only be updated when the PHY lane is in stopstate.

6.2.3 PHY_SHUTDOWNZ

- **Name:** Phy shutdown control
- **Address Offset:** 0x008
- **Size:** 1 bit
- **Value after Reset:** 0
- **Access:** Read/Write

Table 6-4 Phy Shutdown Control Register

Bits	Name	R/W	Description
0	PHY_SHUTDOWNZ	R/W	Shutdown input. This line is used to place the complete macro in power down. All analog blocks are in power down mode and digital logic is cleared. Active Low.

6.2.4 DPHY_RSTZ

- **Name:** Phy reset control
- **Address Offset:** 0x00C
- **Size:** 1 bit
- **Value after Reset:** 0
- **Access:** Read/Write

Table 6-5 Phy Reset Control Register

Bits	Name	R/W	Description
0	DPHY_RSTZ	R/W	DPHY reset output. Active Low.

6.2.5 CSI2_RESETN

- **Name:** CSI-2 controller reset
- **Address Offset:** 0x010
- **Size:** 1 bit
- **Value after Reset:** 0
- **Access:** Read/Write

Table 6-6 CSI-2CSI-2 Controller Reset Register

Bits	Name	R/W	Description
0	CSI2_RESETN	R/W	CSI-2 controller reset output. Active Low.

6.2.6 PHY_STATE

- **Name:** General settings for all blocks
- **Address Offset:** 0x014
- **Size:** 12 bits
- **Value after Reset:** 0
- **Access:** Read (with the exception of bit 11 which is Read/Write)

Table 6-7 General Settings for All Blocks Register

Bits	Name	R/W	Description
0	phy_rxulpsesc_0	R	Lane module 0 has entered the Ultra Low Power mode
1	phy_rxulpsesc_1	R	Lane module 1 has entered the Ultra Low Power mode
2	phy_rxulpsesc_2	R	Lane module 2 has entered the Ultra Low Power mode
3	phy_rxulpsesc_3	R	Lane module 3 has entered the Ultra Low Power mode
4	phy_stopstatedata_0	R	Data Lane 0 in Stop state
5	phy_stopstatedata_1	R	Data Lane 1 in Stop state
6	phy_stopstatedata_2	R	Data Lane 2 in Stop state
7	phy_stopstatedata_3	R	Data Lane 3 in Stop state
8	phy_rxclkactivehs	R	Indicates that the clock lane is actively receiving a DDR clock
9	phy_rxulpsclknot	R	Active Low. This signal indicates that the Clock Lane module has entered the Ultra Low Power state
10	phy_stopstateclk	R	Clock Lane in Stop state
11	bypass_2ecc_tst	R/W	Payload Bypass test mode for double ECC errors

6.2.7 DATA_IDS_1

- **Name:** Data IDs for which IDI reports line boundary matching errors

Note If the core configuration has no Data IDs configured, then this register gets removed.

- **Address Offset:** 0x018
- **Size:** 32 bits
- **Value after Reset:** 0
- **Access:** Read/Write

Table 6-8 Data IDs for Which IDI Reports Line Boundary Matching Errors

Bits	Name	R/W	Description
5:0	di0_dt	R/W	Data ID 0 Data Type
7:6	di0_vc	R/W	Data ID 0 Virtual channel
13:8	di1_dt	R/W	Data ID 1 Data Type
15:14	di1_vc	R/W	Data ID 1 Virtual channel
21:16	di2_dt	R/W	Data ID 2 Data Type
23:22	di2_vc	R/W	Data ID 2 Virtual channel
29:24	di3_dt	R/W	Data ID 3 Data Type
31:30	di3_vc	R/W	Data ID 3 Virtual channel

6.2.8 DATA_IDS_2

- **Name:** Data IDs for which IDI reports line boundary matching errors

Note If the core configuration has four or less Data IDs configured, then this register gets removed.

- **Address Offset:** 0x01C
- **Size:** 32 bits
- **Value after Reset:** 0
- **Access:** Read/Write

Table 6-9 Data IDs for Which IDI Reports Line Boundary Matching Errors

Bits	Name	R/W	Description
5:0	di4_dt	R/W	Data ID 4 Data Type
7:6	di4_vc	R/W	Data ID 4 Virtual channel
13:8	di5_dt	R/W	Data ID 5 Data Type
15:14	di5_vc	R/W	Data ID 5 Virtual channel
21:16	di6_dt	R/W	Data ID 6 Data Type
23:22	di6_vc	R/W	Data ID 6 Virtual channel
29:24	di7_dt	R/W	Data ID 7 Data Type
31:30	di7_vc	R/W	Data ID 7 Virtual channel

6.2.9 ERR1

- **Name:** Error state register 1
- **Address Offset:** 0x020
- **Size:** 29 bits
- **Value after Reset:** 0
- **Access:** Read

Table 6-10 Error State Register 1

Bits	Name	R/W	Description
0	phy_errsotsynchs_0	R	Start of Transmission Error on data lane 0 (no synchronization achieved)
1	phy_errsotsynchs_1	R	Start of Transmission Error on data lane 1 (no synchronization achieved)
2	phy_errsotsynchs_2	R	Start of Transmission Error on data lane 2 (no synchronization achieved)
3	phy_errsotsynchs_3	R	Start of Transmission Error on data lane 3 (no synchronization achieved)
4	err_f_bndry_match_vc0	R	Error matching Frame Start with Frame End for Virtual Channel 0
5	err_f_bndry_match_vc1	R	Error matching Frame Start with Frame End for Virtual Channel 1
6	err_f_bndry_match_vc2	R	Error matching Frame Start with Frame End for Virtual Channel 2
7	err_f_bndry_match_vc3	R	Error matching Frame Start with Frame End for Virtual Channel 3
8	err_f_seq_vc0	R	Incorrect Frame Sequence detected in Virtual Channel 0
9	err_f_seq_vc1	R	Incorrect Frame Sequence detected in Virtual Channel 1
10	err_f_seq_vc2	R	Incorrect Frame Sequence detected in Virtual Channel 2
11	err_f_seq_vc3	R	Incorrect Frame Sequence detected in Virtual Channel 3
12	err_frame_data_vc0	R	Last received frame, in Virtual Channel 0, had at least one CRC error
13	err_frame_data_vc1	R	Last received frame, in Virtual Channel 1, had at least one CRC error
14	err_frame_data_vc2	R	Last received frame, in Virtual Channel 2, had at least one CRC error
15	err_frame_data_vc3	R	Last received frame, in Virtual Channel 3, had at least one CRC error
16	err_l_bndry_match_di0	R	Error matching Line Start with Line End for vc0 and dt
17	err_l_bndry_match_di1	R	Error matching Line Start with Line End for vc1 and dt1
18	err_l_bndry_match_di2	R	Error matching Line Start with Line End for vc2 and dt2
19	err_l_bndry_match_di3	R	Error matching Line Start with Line End for vc3 and dt3
20	err_l_seq_di0	R	Error in the sequence of lines for vc0 and dt0
21	err_l_seq_di1	R	Error in the sequence of lines for vc1 and dt1
22	err_l_seq_di2	R	Error in the sequence of lines for vc2 and dt2
23	err_l_seq_di3	R	Error in the sequence of lines for vc3 and dt3
24	vc0_err_crc	R	Checksum Error detected on Virtual Channel 0
25	vc1_err_crc	R	Checksum Error detected on Virtual Channel 1
26	vc2_err_crc	R	Checksum Error detected on Virtual Channel 2

Table 6-10 Error State Register 1

Bits	Name	R/W	Description
27	vc3_err_crc	R	Checksum Error detected on Virtual Channel 3
28	err_ecc_double	R	Header ECC contains 2 errors. Unrecoverable

6.2.10 ERR2

- **Name:** Error state register 2
- **Address Offset:** 0x024
- **Size:** 24 bits
- **Value after Reset:** 0
- **Access:** Read

Table 6-11 Error State Register 2

Bits	Name	R/W	Description
0	phy_erresc_0	R	Escape Entry Error (ULPM) on data lane 0
1	phy_erresc_1	R	Escape Entry Error (ULPM) on data lane 1
2	phy_erresc_2	R	Escape Entry Error (ULPM) on data lane 2
3	phy_erresc_3	R	Escape Entry Error (ULPM) on data lane 3
4	phy_errsoths_0	R	Start of Transmission Error on data lane 0 (synchronization can still be achieved)
5	phy_errsoths_1	R	Start of Transmission Error on data lane 1 (synchronization can still be achieved)
6	phy_errsoths_2	R	Start of Transmission Error on data lane 2 (synchronization can still be achieved)
7	phy_errsoths_3	R	Start of Transmission Error on data lane 3 (synchronization can still be achieved)
8	vc0_err_ecc_corrected	R	Header error detected and corrected on Virtual Channel 0
9	vc1_err_ecc_corrected	R	Header error detected and corrected on Virtual Channel 1
10	vc2_err_ecc_corrected	R	Header error detected and corrected on Virtual Channel 2
11	vc3_err_ecc_corrected	R	Header error detected and corrected on Virtual Channel 3
12	err_id_vc0	R	Unrecognized or unimplemented data type detected in Virtual Channel 0
13	err_id_vc1	R	Unrecognized or unimplemented data type detected in Virtual Channel 1
14	err_id_vc2	R	Unrecognized or unimplemented data type detected in Virtual Channel 2
15	err_id_vc3	R	Unrecognized or unimplemented data type detected in Virtual Channel 3
16	err_l_bndry_match_di4	R	Error matching Line Start with Line End for vc4 and dt4
17	err_l_bndry_match_di5	R	Error matching Line Start with Line End for vc5 and dt5

Table 6-11 Error State Register 2

Bits	Name	R/W	Description
18	err_l_bndry_match_di6	R	Error matching Line Start with Line End for vc6 and dt6
19	err_l_bndry_match_di7	R	Error matching Line Start with Line End for vc7 and dt7
20	err_l_seq_di4	R	Error in the sequence of lines for vc4 and dt4
21	err_l_seq_di5	R	Error in the sequence of lines for vc5 and dt5
22	err_l_seq_di6	R	Error in the sequence of lines for vc6 and dt6
23	err_l_seq_di7	R	Error in the sequence of lines for vc7 and dt7

6.2.11 MASK1

- **Name:** Masks for errors 1
- **Address Offset:** 0x028
- **Size:** 29 bits
- **Value after Reset:** 0
- **Access:** Read/Write

Table 6-12 Masks for Error State Register 1

Bits	Name	R/W	Description
0	mask_phy_errsotsynchs_0	R/W	Mask for phy_errsotsynchs_0
1	mask_phy_errsotsynchs_1	R/W	Mask for phy_errsotsynchs_1
2	mask_phy_errsotsynchs_2	R/W	Mask for phy_errsotsynchs_2
3	mask_phy_errsotsynchs_3	R/W	Mask for phy_errsotsynchs_3
4	mask_err_f_bndry_match_vc0	R/W	Mask for err_f_bndry_match_vc0
5	mask_err_f_bndry_match_vc1	R/W	Mask for err_f_bndry_match_vc1
6	mask_err_f_bndry_match_vc2	R/W	Mask for err_f_bndry_match_vc2
7	mask_err_f_bndry_match_vc3	R/W	Mask for err_f_bndry_match_vc3
8	mask_err_f_seq_vc0	R/W	Mask for err_f_seq_vc0
9	mask_err_f_seq_vc1	R/W	Mask for err_f_seq_vc1
10	mask_err_f_seq_vc2	R/W	Mask for err_f_seq_vc2
11	mask_err_f_seq_vc3	R/W	Mask for err_f_seq_vc3
12	mask_err_frame_data_vc0	R/W	Mask for err_frame_data_vc0
13	mask_err_frame_data_vc1	R/W	Mask for err_frame_data_vc1
14	mask_err_frame_data_vc2	R/W	Mask for err_frame_data_vc2
15	mask_err_frame_data_vc3	R/W	Mask for err_frame_data_vc3
16	mask_err_l_bndry_match_di0	R/W	Mask for err_l_bndry_match_di0
17	mask_err_l_bndry_match_di1	R/W	Mask for err_l_bndry_match_di1

Table 6-12 Masks for Error State Register 1

Bits	Name	R/W	Description
18	mask_err_l_bndry_match_di2	R/W	Mask for err_l_bndry_match_di2
19	mask_err_l_bndry_match_di3	R/W	Mask for err_l_bndry_match_di3
20	mask_err_l_seq_di0	R/W	Mask for err_l_seq_di0
21	mask_err_l_seq_di1	R/W	Mask for err_l_seq_di1
22	mask_err_l_seq_di2	R/W	Mask for err_l_seq_di2
21	mask_err_l_seq_di3	R/W	Mask for err_l_seq_di3
22	mask_vc0_err_crc	R/W	Mask for vc0_err_crc
23	mask_vc1_err_crc	R/W	Mask for vc1_err_crc
24	mask_vc2_err_crc	R/W	Mask for vc2_err_crc
25	mask_vc3_err_crc	R/W	Mask for vc3_err_crc
26	mask_err_ecc_double	R/W	Mask for err_ecc_double

6.2.12 MASK2

- **Name:** Masks for errors 2
- **Address Offset:** 0x02C
- **Size:** 24 bits
- **Value after Reset:** 0
- **Access:** Read/Write

Table 6-13 Masks for Error State Register 2

Bits	Name	R/W	Description
0	mask_phy_erresc_0	R/W	Mask for phy_erresc_0
1	mask_phy_erresc_1	R/W	Mask for phy_erresc_1
2	mask_phy_erresc_2	R/W	Mask for phy_erresc_2
3	mask_phy_erresc_3	R/W	Mask for phy_erresc_3
4	mask_phy_errsoths_0	R/W	Mask for phy_errsoths_0
5	mask_phy_errsoths_1	R/W	Mask for phy_errsoths_1
6	mask_phy_errsoths_2	R/W	Mask for phy_errsoths_2
7	mask_phy_errsoths_3	R/W	Mask for phy_errsoths_3
8	mask_vc0_err_ecc_corrected	R/W	Mask for vc0_err_ecc_corrected
9	mask_vc1_err_ecc_corrected	R/W	Mask for vc1_err_ecc_corrected
10	mask_vc2_err_ecc_corrected	R/W	Mask for vc2_err_ecc_corrected
11	mask_vc3_err_ecc_corrected	R/W	Mask for vc3_err_ecc_corrected
12	mask_err_id_vc0	R/W	Mask for err_id_vc0
13	mask_err_id_vc1	R/W	Mask for err_id_vc1
14	mask_err_id_vc2	R/W	Mask for err_id_vc2
15	mask_err_id_vc3	R/W	Mask for err_id_vc3
16	mask_err_l_bndry_match_di4	R/W	Mask for err_l_bndry_match_di4
17	mask_err_l_bndry_match_di5	R/W	Mask for err_l_bndry_match_di5

Table 6-13 Masks for Error State Register 2

Bits	Name	R/W	Description
18	mask_err_l_bndry_match_di6	R/W	Mask for err_l_bndry_match_di6
19	mask_err_l_bndry_match_di7	R/W	Mask for err_l_bndry_match_di7
20	mask_err_l_seq_di4	R/W	Mask for err_l_seq_di4
21	mask_err_l_seq_di5	R/W	Mask for err_l_seq_di5
22	mask_err_l_seq_di6	R/W	Mask for err_l_seq_di6
23	mask_err_l_seq_di7	R/W	Mask for err_l_seq_di7

6.2.13 PHY_TST_CRTL0

- **Name:** D-PHY Test interface control 0
- **Address Offset:** 0x030
- **Size:** 2 bits
- **Value after Reset:** 0
- **Access:** Read/Write

Table 6-14 PHY Configuration and Test Control 0 Registers

Bits	Name	R/W	Description
0	phy_testclr	R/W	PHY test interface clear. Used when active performs vendor specific interface initialization (Active High).
1	phy_testclk	R/W	PHY test interface strobe signal. Used to clock TESTDIN bus into the D-PHY. In conjunction with TESTEN signal controls the operation selection.

6.2.14 PHY_TST_CRTL1

- **Name:** D-PHY Test interface control 1
- **Address Offset:** 0x034
- **Size:** 17 bits
- **Value after Reset:** 0
- **Access:** Read/Write

Table 6-15 PHY Configuration and Test Control 1 Registers

Bits	Name	R/W	Description
0:7	phy_testdin	R/W	PHY test interface input 8-bit data bus for internal register programming and test functionalities access
8:15	phy_testdout	R	PHY output 8-bit data bus for read-back and internal probing functionalities.
16	phy_testen	R/W	PHY test interface operation selector: • 1: Configures address write operation on the falling edge of TESTCLK • 0: Configures a data write operation on the rising edge of TESTCLK