MIPI D-PHY Receiver 1.2 IP Core User Guide

Revised September 12, 2018; Author Elod Gyorgy

1 Introduction

This user guide describes the Digilent MIPI D-PHY Receiver Intellectual Property. This IP is compatible with D-PHY 1.0 specifications and serves as the lowest layer of the high-speed source-synchronous interface defined by MIPI Alliance. It pairs up with a MIPI CSI-2 Receiver IP over the standard PHY Protocol Interface (PPI) to receive data from an image sensor and source a video subsystem. The physical interconnect for Xilinx 7-series FPGA relies on techniques outlined in XAPP894[1].

2 Features

- Single or dual lane support
- CIL-SFEN, CIL-SCNN lane implementation: unidirectional, Control and High-Speed modes
- Xilinx interfaces used: AXI4-Lite, rx_mipi_ppi_if_rtl:1.0
- Debug module

IP quick facts	
Supported device families	Zynq®-7000, 7 series
Supported user interfaces	Xilinx®: AXI4-Lite, rx_mipi_ppi
Provided with core	
Design files	VHDL
Simulation model	VHDL Behavioral
Constraints file	XDC
Software driver	standalone
Tested design flows	
Design entry	Vivado™ Design Suite 2017.4
Synthesis	Vivado Synthesis 2017.4

3 Performance

The IP has been tested in dual-lane configuration with 1344 Mbps total data rate, resulting in 84 MHz PPI high-speed byte clock (RxByteClkHS).

4 Resource Utilization

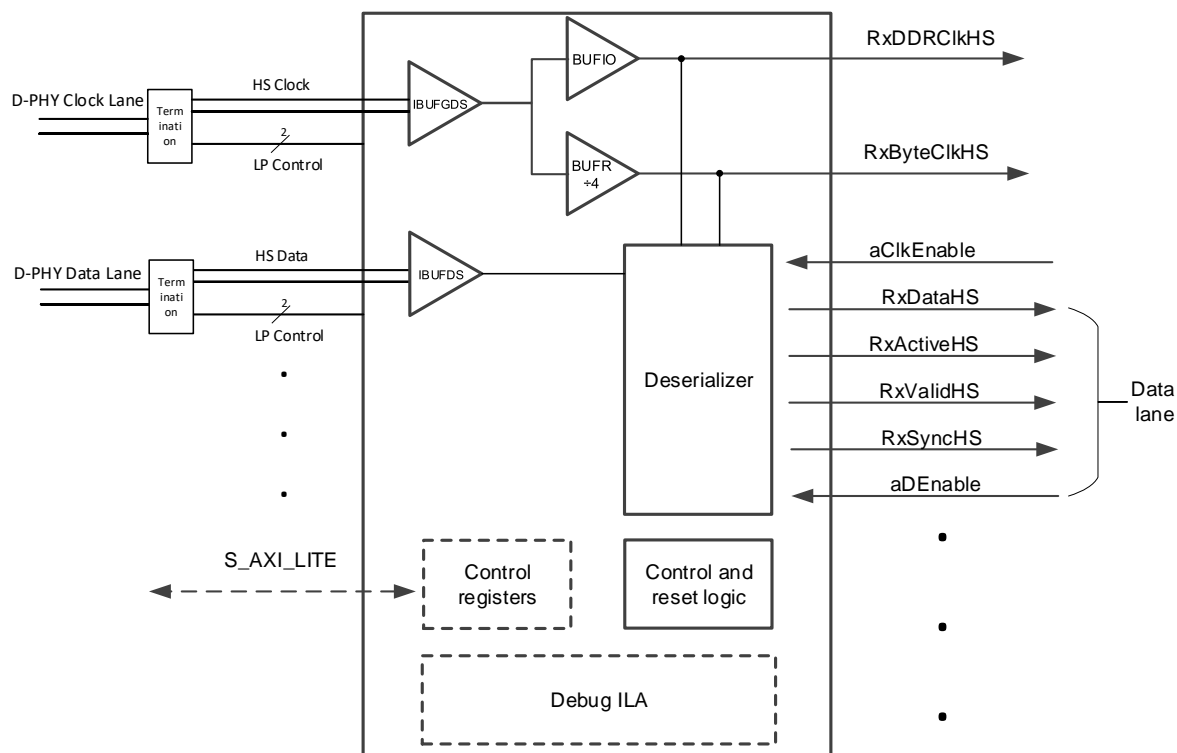| Device	Configuration	Resource				
		LUT	FF	BRAM	URAM	DSP
xc7z020clg400-1	AXI-Lite Interface	297	312	0	0	0

5 Overview

Figure 1. DVI to VGA converter block diagram.

The IP is built from multiple blocks: input buffers, clock buffers, de-serializer, control logic and optional debug modules.

6 Port Descriptions

7 Designing with the core

The IP expects to be connected directly to top-level ports, since input buffers are instantiated internally. Since the D-PHY I/O standard is not supported directly by FPGA pins, it implements the techniques described in [1] that separate the D-PHY lane into a differential high-speed bus (LVDS_25) and two low-power control signals (HSUL_12). It was verified as working with either passive or active termination. This implementation allows 3.3V-supplied HR banks to interface with D-PHY transmitters using external on-board terminations and internal voltage reference.

7.1 Constraints

See an example below on how to constrain the low-power (LP) and high-speed (HS) input pins. Banks hosting HSUL_12 pins need a 0.6V voltage reference, either internal or external. A primary clock with a frequency corresponding to the maximum expected data rate should be created on the clock input port.

```
set_property INTERNAL_VREF 0.6 [get_iobanks 35]
set_property -dict {PACKAGE_PIN J19 IOSTANDARD HSUL_12} [get_ports dphy_clk_lp_n]
set_property -dict {PACKAGE_PIN H20 IOSTANDARD HSUL_12} [get_ports dphy_clk_lp_p]
set_property -dict {PACKAGE_PIN M18 IOSTANDARD HSUL_12} [get_ports
{dphy_data_lp_n[0]}]
set_property -dict {PACKAGE_PIN L19 IOSTANDARD HSUL_12} [get_ports
{dphy_data_lp_p[0]}]
set_property -dict {PACKAGE_PIN L20 IOSTANDARD HSUL_12} [get_ports
{dphy_data_lp_n[1]}]
set_property -dict {PACKAGE_PIN J20 IOSTANDARD HSUL_12} [get_ports
{dphy_data_lp_p[1]}]

set_property -dict {PACKAGE_PIN H18 IOSTANDARD LVDS_25} [get_ports
dphy_hs_clock_clk_n]
set_property -dict {PACKAGE_PIN J18 IOSTANDARD LVDS_25} [get_ports
dphy_hs_clock_clk_p]
# 672Mbps/lane = 336 MHz HS_Clk
create_clock -period 2.976 -name dphy_hs_clock_p -waveform {0.000 1.488} [get_ports
dphy_hs_clock_clk_p]

set_property -dict {PACKAGE_PIN M20 IOSTANDARD LVDS_25} [get_ports
{dphy_data_hs_n[0]}]
set_property -dict {PACKAGE_PIN M19 IOSTANDARD LVDS_25} [get_ports
{dphy_data_hs_p[0]}]
set_property -dict {PACKAGE_PIN L17 IOSTANDARD LVDS_25} [get_ports
{dphy_data_hs_n[1]}]
set_property -dict {PACKAGE_PIN L16 IOSTANDARD LVDS_25} [get_ports
{dphy_data_hs_p[1]}]
```

7.2   Customization

8 Debugging
9 References

1. Xilinx Inc., *XAPP894: D-PHY Solutions*, v1.0, August 25, 2014.