MIPI SPI2DSI IP V1.0

IP User Guide(Beta Release)

February 20, 2024

RapidSilicon

Contents

Licensing

COPYRIGHT TEXT:

IP Specification

Overview

MIPI SPI2DSI IP

As the realm of display technology continues its rapid evolution, High-Resolution LCD/TFT LCD panels are becoming increasingly prevalent. The surge in resolution not only elevates picture quality but also introduces content with heightened bandwidth requirements for efficient transmission from the display controller to the display itself.

The MIPI SPI2DSI Bridge, serves as a MIPI DSI (Display Serial Interface) transmitter, enabling seamless integration between host processors and MIPI DSI display panels. This sophisticated bridge translates Serial Peripheral Interface (SPI) signals from the host device into MIPI DSI signals, facilitating a smooth and efficient communication link between the host and DSI displays. With its advanced capabilities, our MIPI SPI2DSI Bridge empowers designers and engineers to effortlessly connect standard SPI interfaces to MIPI DSI displays, unlocking a world of possibilities for innovative and high-performance display applications across a wide range of electronic devices. This MIPI SPI2DSI Bridge stands as a versatile and robust solution, offering the flexibility and performance needed to meet the evolving demands of modern display technologies.

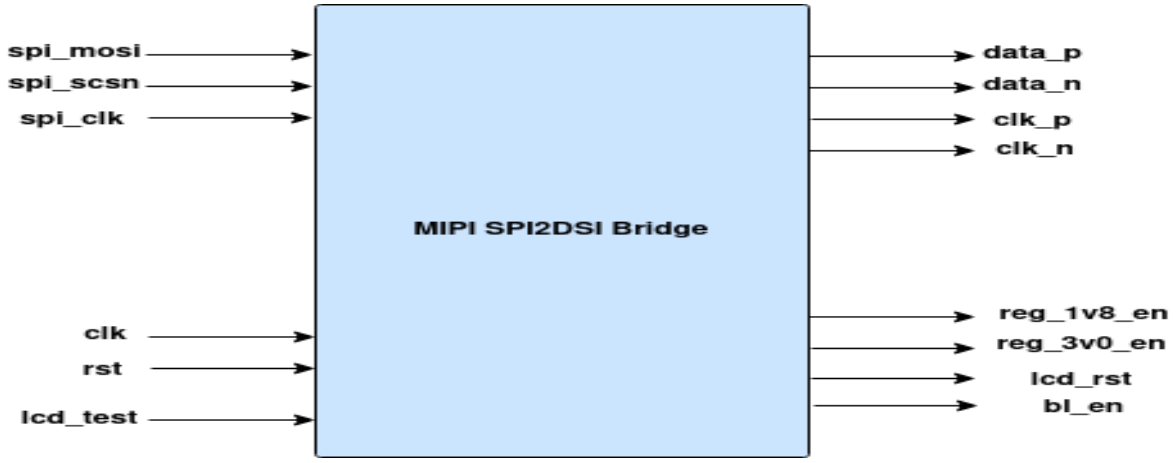


Figure 1. MIPI SPI2DSI IP Block

Features:

- Supports High speed transmission
- Standard SPI interface for incoming data
- Compatibile with MIPI DPHY standard.

Overview

Core Description

The MIPI SPI2DSI bridge IP is composed of crucial components integrated to operate as a MIPI transmitter. Key building blocks include the SPI slave, PLL, FIFO, ROM, main controller, and a custom DPHY equiped with a transmitter interface. The block diagram below shows the connectivity of the blocks.

SPI Slave: Responsible for receiving data from the processor through the SPI interface, enabling communication between the processor and Main controller.

PLL (Phase-Locked Loop): Generates precise, high-speed clocks with phase shifts, ensuring synchronization and stability for optimal performance for the bridge.

FIFO (First-In-First-Out): Buffers data from the SPI Slave, maintaining a consistent flow and preventing data loss during transitions.

Main Controller: The core module orchestrates system functions, generating triggers for the custom DPHY, issuing commands, and coordinating communication between components for seamless operation.

Custom DPHY: Mimics select features of the standard MIPI DPHY, generating high-speed frames for transmission, tailored to the specific requirements of the system.

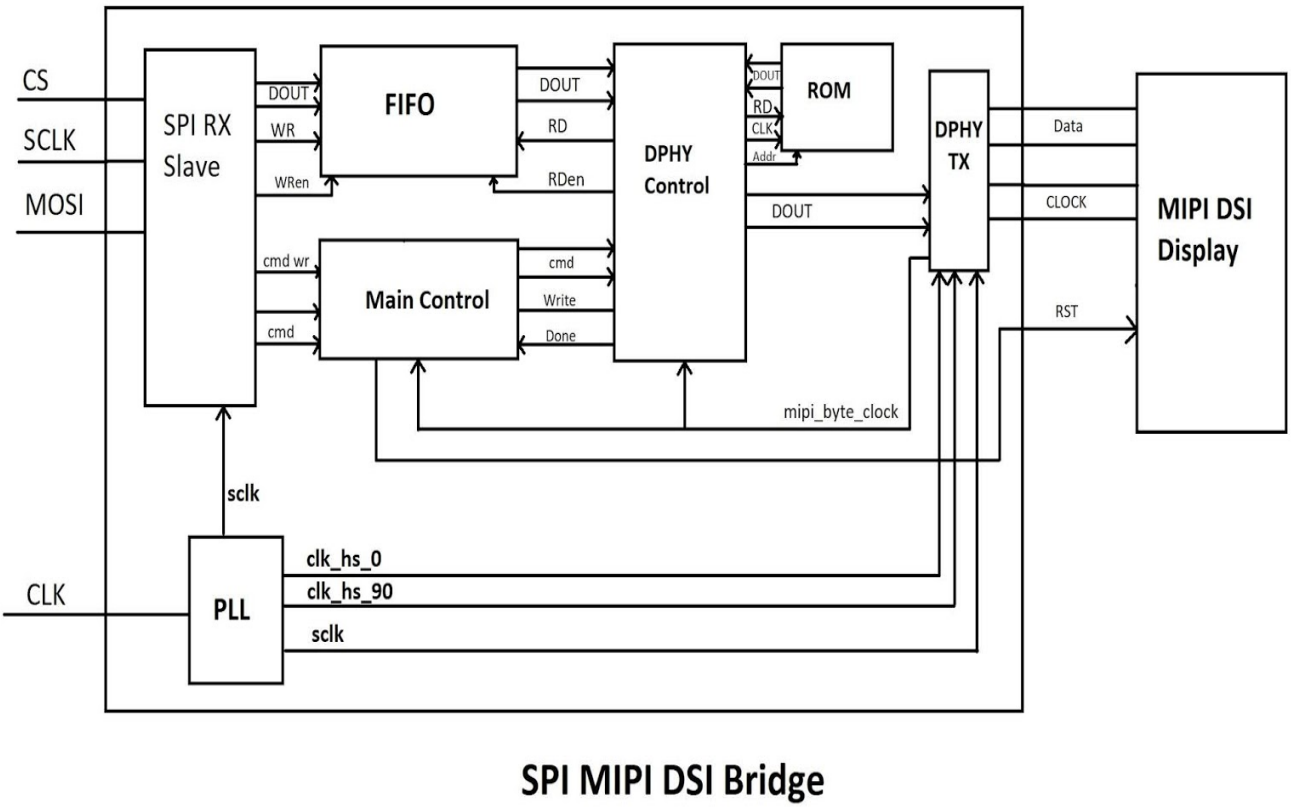


Figure 2. MIPI SPI2DSI IP Block

The IP supports basic High speed transmission as per MIPI Alliance Specification for D-PHY Version 1.00. The transmitter is required to generate a specific sequence for entering and exiting High-Speed (HS) mode. The image below illustrates the steps to enter HS modes:

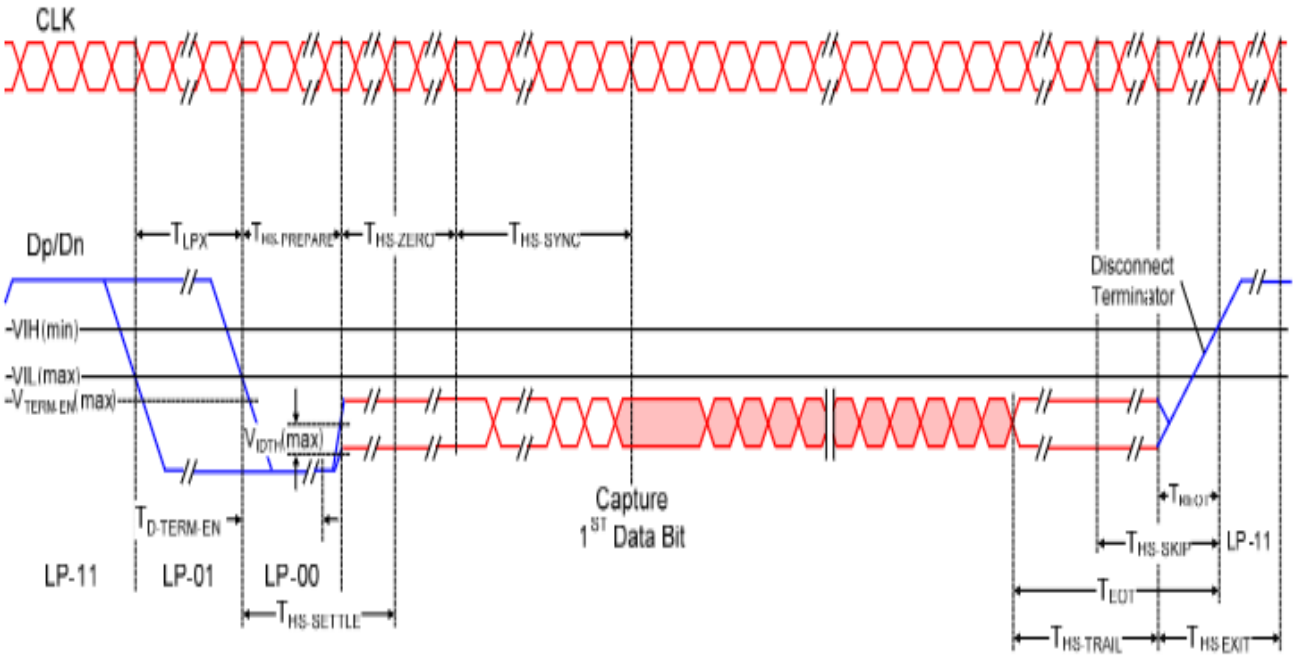


Figure 3. MIPI SPI2DSI IP Block

Stage 0: The LP-11 state in the image represents the Low-Power (LP) state, also referred to as stop state.
Stage 1: To transition into HS mode, the transmitter should pull LP dp low for a minimum of Tlpx (50ns) and remain in LP-01 (with the HS driver in tristate mode during LP-01).
Stage 2: Pull LPdn low for Ths-prepare (minimum 95ns) and remain in LP-00.
Stage 3: With the target now in HS mode, activate the HS driver and start sending mandatory zeros.
Stage 4: Send the mandatory 0xB8 sync byte, followed by the payload.

IP Support Details

Compliance		IP Resources					Tool Flow		
Device	**Interface**	**Source Files**	**Constraint File**	**Testbench**	**Simulation Model**	**Software Driver**	**Analyze and Elaboration**	**Simulation**	**Synthesis**
Raptor FPGA Device	SPI	Verilog	-	Verilog	-	-	Raptor	Raptor	Raptor

Resource Utilization

Please note that the utilization and timing figures provided in this section for the MIPI SPI2DSI IP core should be considered as estimates, as they are based on its usage in conjunction with other design modules in the FPGA. Once integrated with other designs in the system, the FPGA resource utilization and core timing may differ from the reported results.

Tool	Raptor Design Suite			
FPGA Device	GEMINI			
Configuration			**Resource Utilization**	
	Options	**Configuration**	**Resources**	**Utilized**
Minimum Resource			LUT	469
			Registers	280
			BRAM	0
			PLL	1

Ports

Table 2 lists the top interface ports of the MIPI SPI2DSI IP Core.

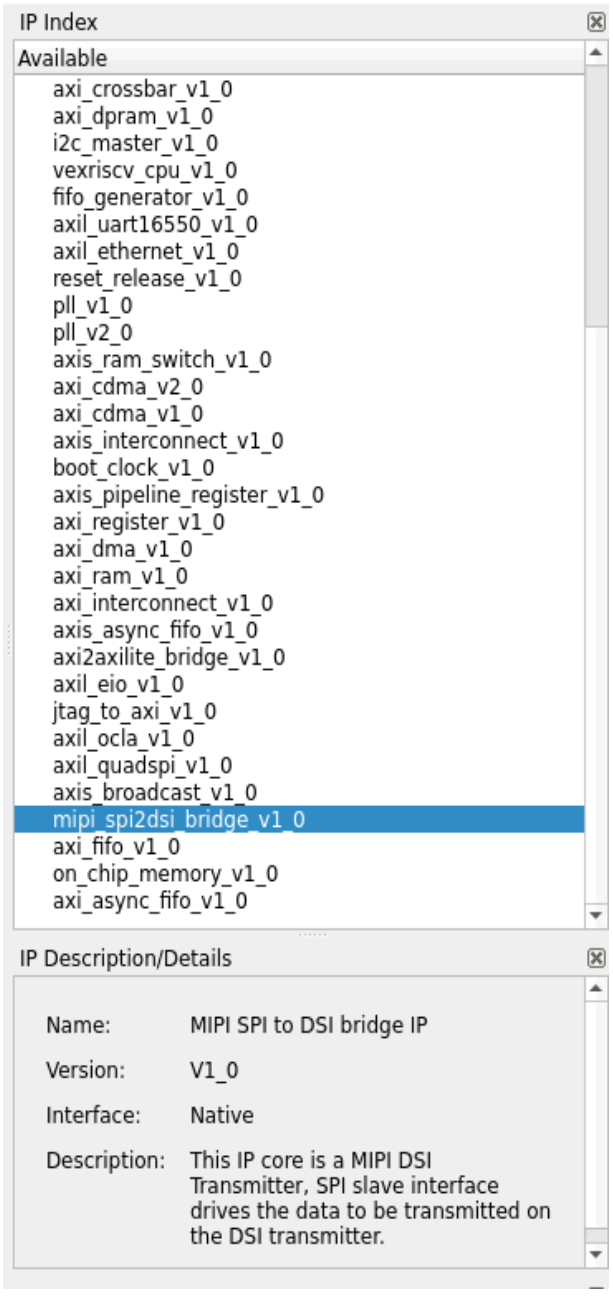
Signal Name	I/O	Description
clk	I	System clock
rst	I	External reset of the system
spi_mosi_i	I	SPI main out, subnode in (MOSI)
spi_csn_i	I	PPI chip select
spi_clk_i	I	SPI clock
lcd_test_i	I	Test input from LCD, shows that the LCD is reset
reg_1v8_en	O	1.8 V output signal for LCD
reg_3v0_en	O	3.0 V output signal for LCD
lcd_rst	O	Active low Reset for LCD
bl_en	O	Enable for the LCD
data_p	O	Data positive differential line
data_n	O	Data negative differential line
clock_p	O	Clock positive differential line
clock_n	O	Clock negative differential line

MIPI SPI2DSI Interface

Design Flow

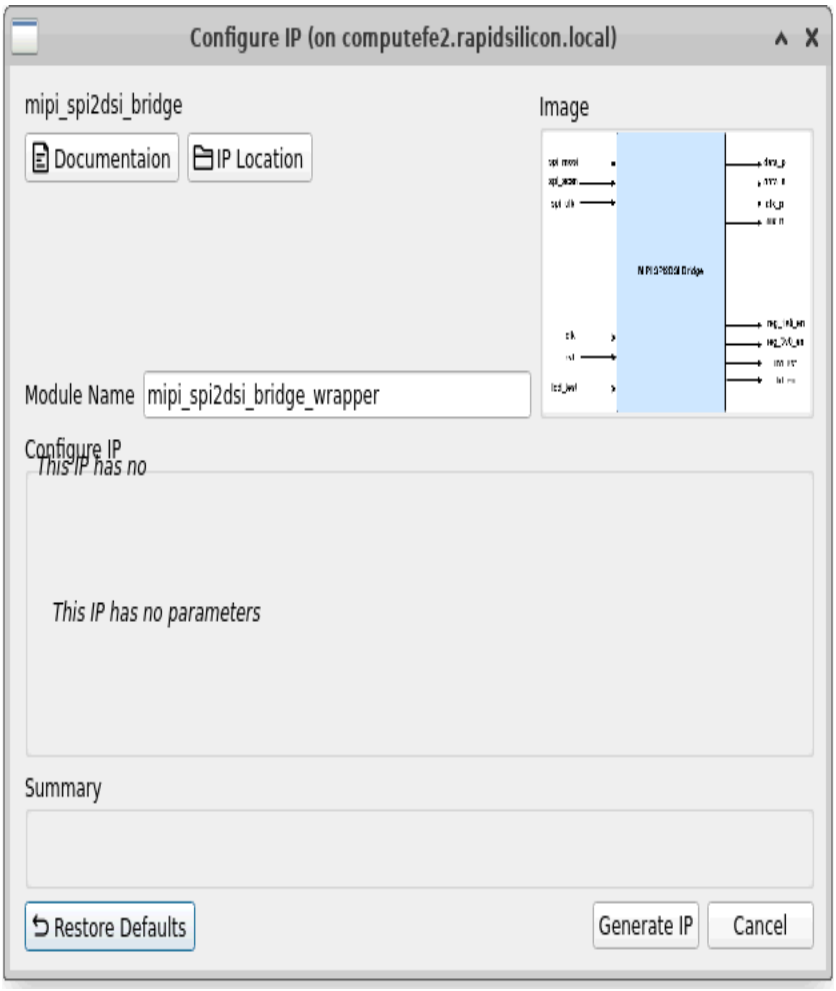
IP Customization and Generation

MIPI SPI2DSI IP core is a part of the Raptor Design Suite Software. A customized reset release can be generated from the Raptor's IP configurator window.



IP list

Parameters Customization: From the IP configuration window, the parameters of the MIPI SPI2DSI IP can be configured and MIPI SPI2DSI IP features can be enabled for generating a customized MIPI SPI2DSI IP core that suits the user application requirement.



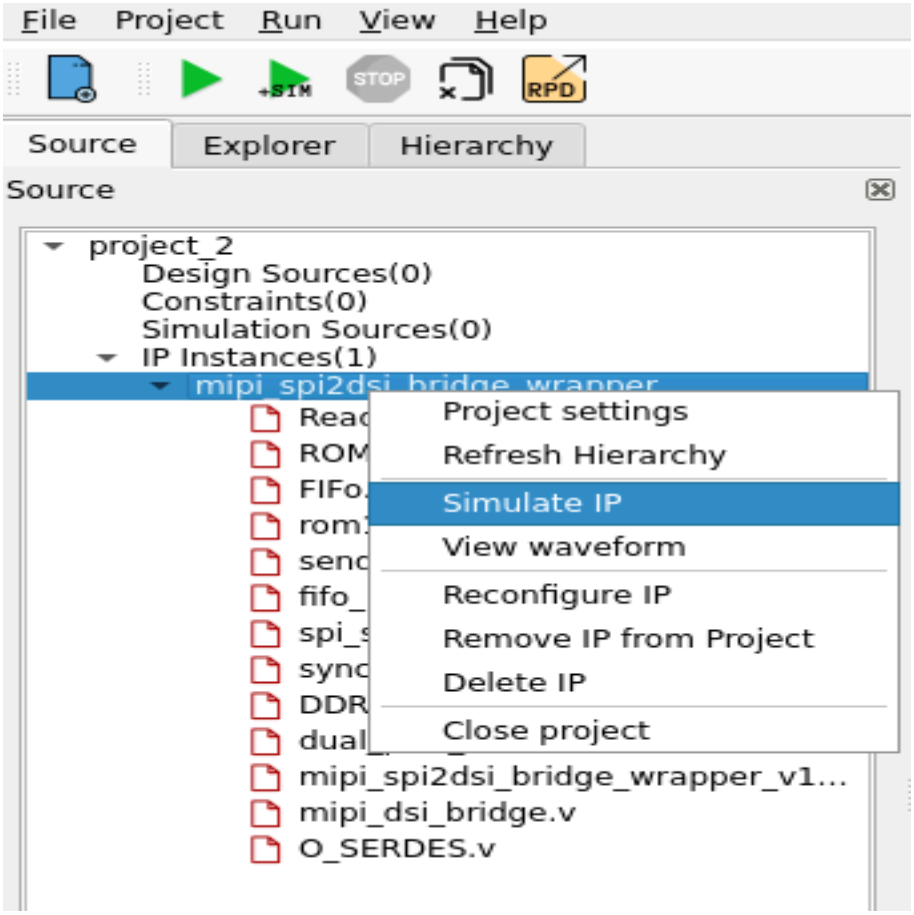
IP Configuration

Synthesis and PnR

Raptor Suite is armed with tools for Synthesis along with Post and Route capabilities and the generated post-synthesis and post-route and place netlists can be viewed and analyzed from within the Raptor. The generated bitstream can then be uploaded on an FPGA device to be utilized in hardware application.

Test Bench

The IP package hosts a simple Verilog based test bench that validates design functionality. It can be simulated using Iverilog, Verilator or other simulators. The simulation can be easily run by clicking the "Simulate IP" button as shown in figure 6. The waveforms are also dumped for in-depth analysis of the whole operation which can be seen by clicking the "View Waveform" button.



Simulate IP Window

Revision History

+

Date	Version	Revisions
February 20, 2024	0.01	Initial version MIPI SPI2DSI IP User Guide