

AXI ASYNCHRONUS FIFO V1.0

IP User Guide(Beta Release)



November 26, 2023

Contents

IP Specifications	2
Licensing	3
IP Specification	4
Overview	4
IP Support Details	5
Port List	5
Parameters	6
Resource Utilization	7
Design Flow	8
IP Customization and Generation	8
Example Design	10
Simulating the Example Design	10
Synthesis and PnR	10
Release	11
Revision History	11

IP Summary

Introduction

The AXI Async FIFO is an AXI full compliant customize-able asynchronous FIFO. It can be used to store and retrieve ordered data at different clock domains, while using optimal resources. This core can be configured via Raptor's IP Catalog GUI interface.

Features

- 32-bit AXI4 slave interface
- Data width can be configured to 32, 64, 128, 256 or 512 bits for AXI4
- Configurable FIFO depth ranging from 8 to 32k locations

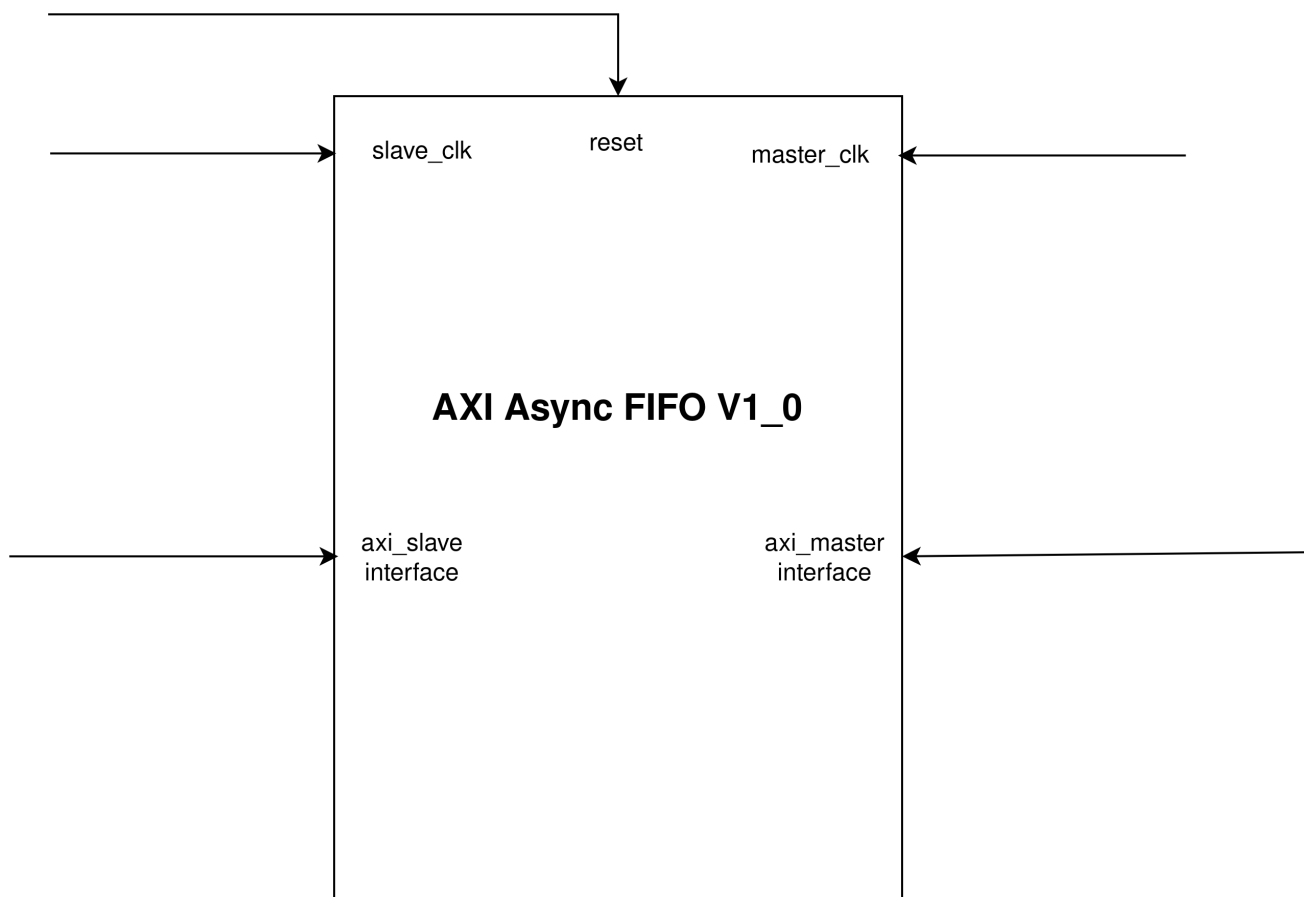


Figure 1. AXI_Async_FIFO block level diagram

Licensing

COPYRIGHT TEXT:

Copyright (c) 2022 RapidSilicon

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

IP Specification

Overview

The AXI Async FIFO Generator core has five channels and creates a native FIFO for every channel. The Write Channels consist of a Write Address Channel, Write Data Channel, and Write Response Channel, while the Read Channels include a Read Address Channel and Read Data Channel. The core integrates three FIFOs for Write and two FIFOs for Read Channels, five independent FIFOs are integrated.

The figure 2 shows all five channels and the native FIFOs for exchange of data.

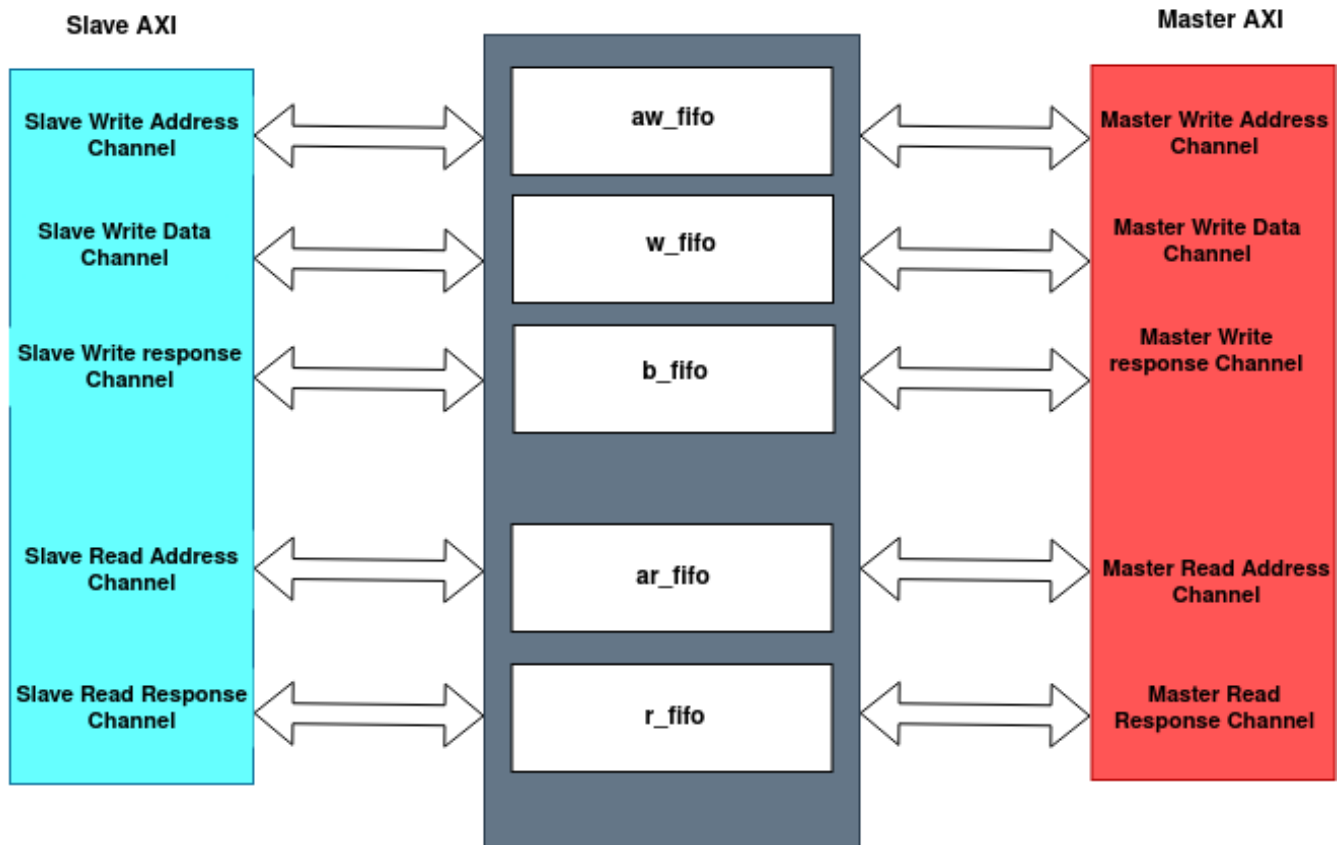


Figure 2. AXI Async FIFO internal channel division and FIFO connections.

IP Support Details

Compliance		IP Resources					Tool Flow		
Device	Interface	Source Files	Constraint File	Testbench	Simulation Model	Software Driver	Analyze and Elaboration	Simulation	Synthesis
GEMINI	AXI4	Systemverilog	SDC	Systemverilog	-	-	Raptor	Raptor	Raptor

Ports

Table 2 lists the top interface ports of the AXI Async FIFO.

Signal Name	I/O	Description
AXI Clock and Reset		
i_s_axi_clk	I	AXI4 Clock
i_s_axi_rese	I	AXI4 RESET
i_m_axi_clk	I	AXI4 Clock
AXI WRITE ADDRESS CHANNEL		
s_axil_awvalid	I	AXI4 Write address valid
s_axil_awready	O	AXI4 Write address ready
s_axil_awaddr	I	AXI4 Write address
s_axil_awprot	I	AXI4 Protection type
AXI WRITE DATA CHANNEL		
s_axil_wvalid	I	AXI4 Write valid
s_axil_wready	O	AXI4 Write ready.
s_axil_wdata	I	AXI4 Write data
s_axil_wstrb	I	AXI4 Write strobes
AXI WRITE RESPONSE CHANNEL		
s_axil_bvalid	O	AXI4 Write response valid
s_axil_bready	I	AXI4 Response ready
s_axil_bresp	O	AXI4 Write response
AXI READ ADDRESS CHANNEL		
s_axil_arvalid	I	AXI4 Read address valid
s_axil_arready	O	AXI4 Read address ready
s_axil_araddr	I	AXI4 Read address
s_axil_arprot	I	AXI4 Protection type
AXI READ DATA CHANNEL		
s_axil_rvalid	I	AXI4 Read valid
s_axil_rready	O	AXI4 Read ready
s_axil_rresp	I	AXI4 Read data
s_axil_rdata	O	AXI4 Read response
AXI WRITE ADDRESS CHANNEL		
m_axil_awvalid	I	AXI4 Write address valid
m_axil_awready	O	AXI4 Write address ready
m_axil_awaddr	I	AXI4 Write address
m_axil_awprot	I	AXI4 Protection type
AXI WRITE DATA CHANNEL		
m_axil_wvalid	I	AXI4 Write valid
m_axil_wready	O	AXI4 Write ready.

m_axil_wdata	I	AXI4 Write data
m_axil_wstrb	I	AXI4 Write strobes
AXI WRITE RESPONSE CHANNEL		
m_axil_bvalid	O	AXI4 Write response valid
m_axil_bready	I	AXI4 Response ready
m_axil_bresp	O	AXI4 Write response
AXI READ ADDRESS CHANNEL		
m_axil_arvalid	I	AXI4 Read address valid
m_axil_arready	O	AXI4 Read address ready
m_axil_araddr	I	AXI4 Read address
m_axil_arprot	I	AXI4 Protection type
AXI READ DATA CHANNEL		
m_axil_rvalid	I	AXI4 Read valid
m_axil_rready	O	AXI4 Read ready
m_axil_rresp	I	AXI4 Read data
m_axil_rdata	O	AXI4 Read response

AXI Async FIFO Interface

Parameters

Table 3 lists the parameters of the AXI Async FFIFO.

Parameter	Values	Default Value	Description
DATA WIDTH	8,16,32,64,128	32	Data width of data being transferred.
ADDR WIDTH	32-64	32	FIFO address width.
ID WIDTH	1-32	8	FIFO ID width.
AWUSER ENABLE	0-1	0	Depth of internal FIFO.
AWUSER WIDTH	1-1024	32	Data width of data being transferred.
WUSER ENABLE	1-64	32	FIFO address width.
WUSER WIDTH	1-1024	8	FIFO ID width.
BUSER ENABLE	8-8192	4096	Depth of internal FIFO.
BUSER WIDTH	1-1024	32	Data width of data being transferred.
ARUSER ENABLE	1-64	32	FIFO address width.
ARUSER WIDTH	1-1024	8	FIFO ID width.
RUSER ENABLE	8-8192	4096	Depth of internal FIFO.
RUSER WIDTH	1-1024	32	Data width of data being transferred.
WRITE FIFO DEPTH	1-32515	32	FIFO address width.
READ FIFO DEPTH	1-32515	8	FIFO ID width.
WRITE FIFO DELAY	0-1	4096	Depth of internal FIFO.
READ FIFO DELAY	0-1	32	Data width of data being transferred.

Parameters

Resource Utilization

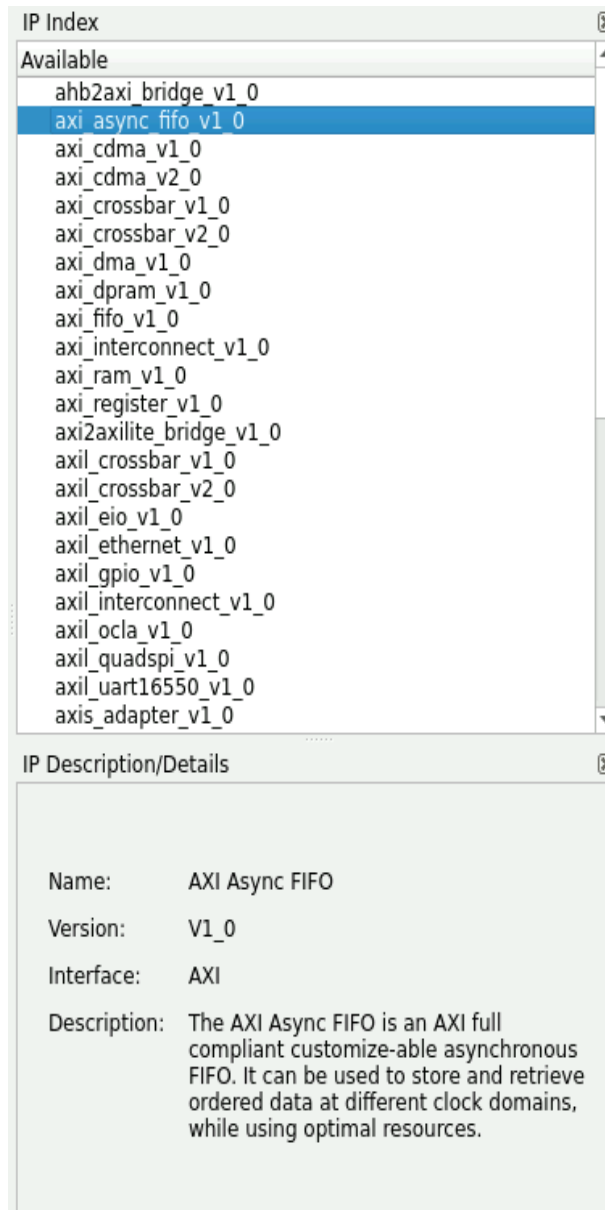
Please note that the utilization and timing figures provided in this section for the Processor System Reset IP core should be considered as estimates, as they are based on its usage in conjunction with other design modules in the FPGA. Once integrated with other designs in the system, the FPGA resource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

Tool	Raptor Design Suite			
FPGA Device	GEMINI			
Configuration			Resource Utilization	
Minimum Resource	Options	Configuration	Resources	Utilized
	FIFO DEPTH	8	LUT	119
	DATA WIDTH	8	Registers	217
	ADDR WIDTH	8	BRAM	5
	ID WIDTH	8	DSP	0
Minimum Resource	Options	Configuration	Resources	Utilized
	FIFO DEPTH	8096	LUT	610
	DATA WIDTH	64	Registers	717
	ADDR WIDTH	32	BRAM	22
	ID WIDTH	8	DSP	0

Design Flow

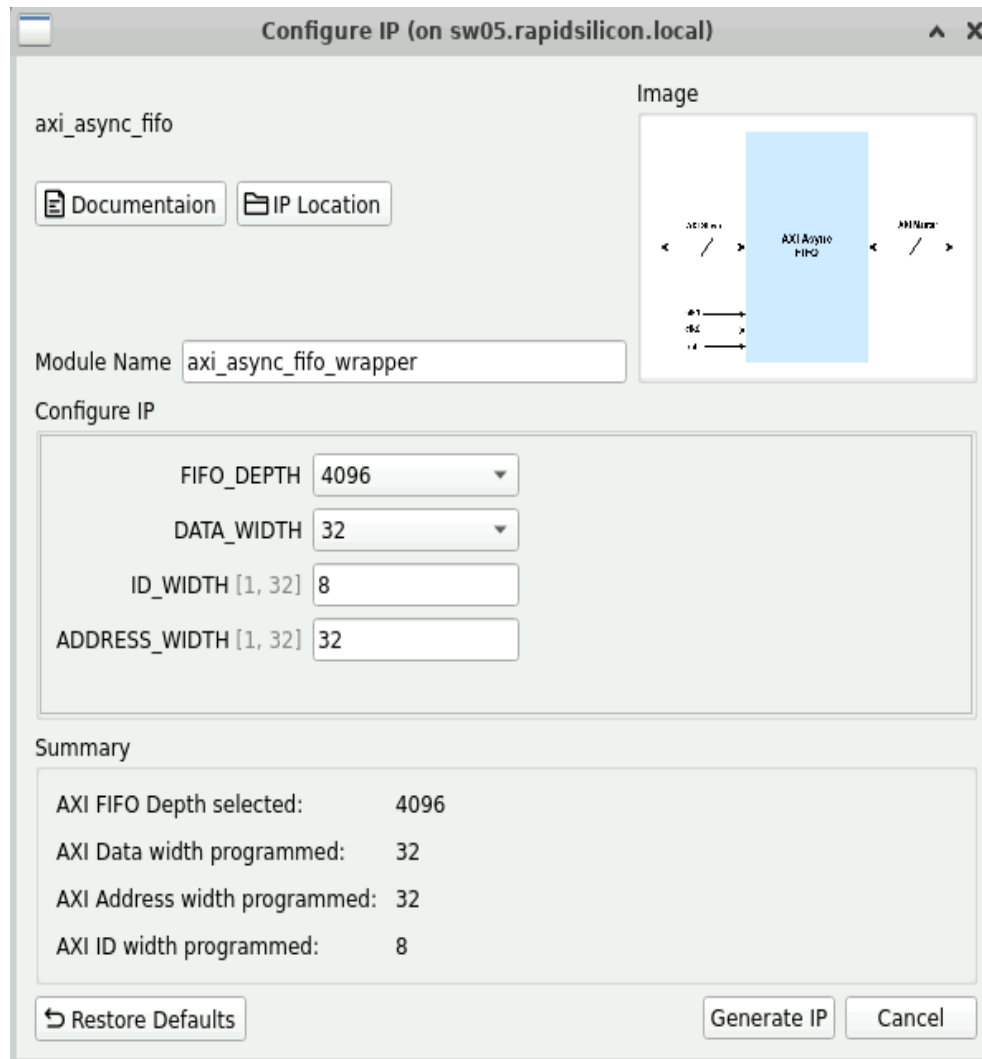
IP Customization and Generation

AXI Async FIFO IP core is a part of the Raptor Design Suite Software. A customized AXI Async FIFO can be generated from the Raptor's IP configurator window.



Selecting AXI Async FIFO from IP Catalog List

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



Configure IP (on sw05.rapidsilicon.local)

axi_async_fifo

[Documentation](#) [IP Location](#)

Module Name: axi_async_fifo_wrapper

Configure IP

FIFO_DEPTH: 4096

DATA_WIDTH: 32

ID_WIDTH [1, 32]: 8

ADDRESS_WIDTH [1, 32]: 32

Summary

AXI FIFO Depth selected: 4096

AXI Data width programmed: 32

AXI Address width programmed: 32

AXI ID width programmed: 8

[Restore Defaults](#) [Generate IP](#) [Cancel](#)

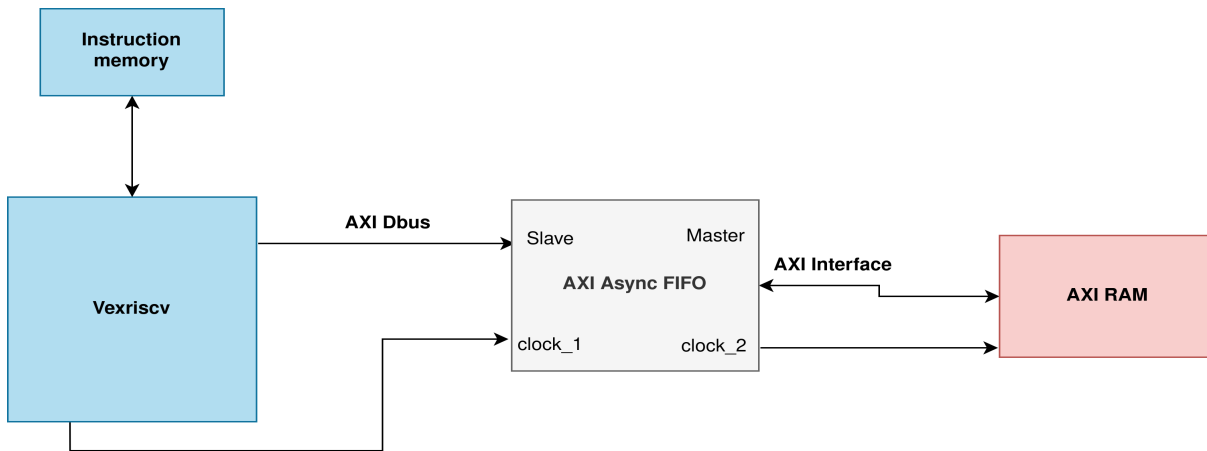
IP Configuration

Example Design

Simulating the Example Design

We are simulating AXI Async FIFO in an SoC based environment where we use it as a bridge between the CPU and peripherals where it takes care of different clocks. The test generates write transactions from the CPU for the AXI RAM and the transactions generated by the CPU are at a higher frequency while the RAM frequency is half than the CPU.

The simulation collateral is available in the sim directory.



AXI Async FIFO example design simulation

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.

Revision History

Date	Version	Revisions
November 26, 2023	0.01	Initial version AXI_Async_FIFO User Guide Document