AXIS RAM Switch v1.0

IP User Guide (Beta Release)

December 5, 2023

RapidSilicon

Contents

IP Summary

Introduction

The AXIS RAM Switch core is an AXI4-Streaming compliant customizable switch that is designed to be used in applications that require configurable buffered routing between masters and slaves with multiple arbitration options. The TDEST based routing uses RTL parameters configured before synthesis to control the routing by assigning a base TDEST pair to each master interface that is then used to generate a decode table which is then decoded by the respective slave interface based on the valid TDEST value. This transfer routes a request to an arbiter of one of the master interfaces.
The arbiter responds with a grant based on input from a priority encoder, and then the slave proceeds with the transfer. Arbitration can be performed at either transfer level or the transaction level. The TDEST based routing requires that the signal has at least log_2 number of bits. The embedded RAM makes sure that the flow of data is not interrupted between different pairs of master/slave interfaces while also making sure only desired traffic is routed filtering out the non-required chunks of data.

Features

- Supports 1-16 slaves.
- Supports 1-16 masters.
- Supports Round-Robin and LSB High Priority arbitration choices.
- Supports routing based on TDEST base/high pairs.
- Configurable data width.
- Configurable slave/master register types.
- Configurable offset in address arbitration.
- Configurable RAM pipeline stages.
- Configurable data masking options.

Overview

AXIS RAM Switch

The AXIS RAM switch is a hardware device and it allows multiple computers or devices to access the same data or resources over a LAN or WAN. It is configured via its AXI-streaming slave interface. It captures the number of slaves and masters and their respective addressing offsets and generate address configuration for each master-slave pair. The arbitration occurs via an embedded priority encoder. The embedded RAM in turn checks for bad frame markers and drops them based on the routing information configured at boot time. The Figure 1 shows the AXIS RAM Switch connected to multiple masters and slaves routing data in between them.

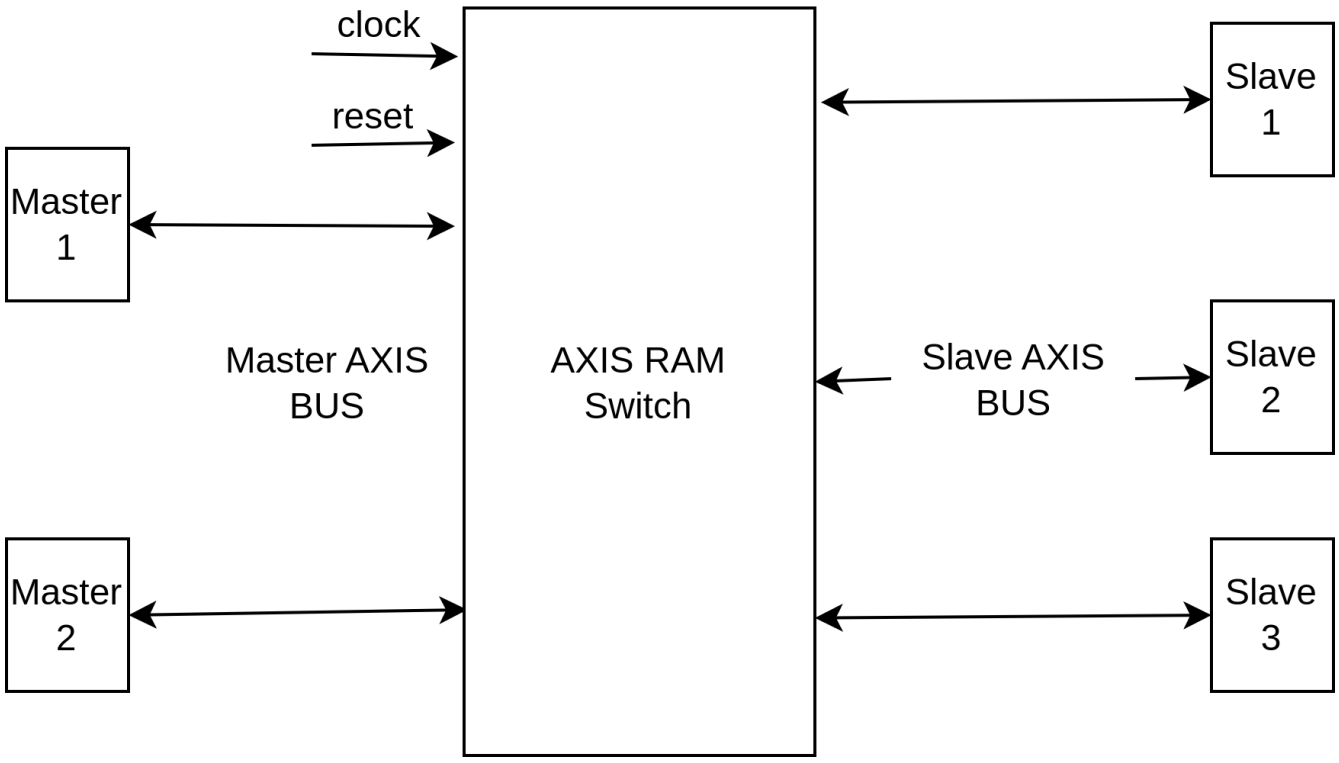


Figure 1. AXIS RAM Switch

Licensing

COPYRIGHT TEXT:

IP Specification

The Figure 2 shows the internal block diagram of AXIS RAM Switch. It consists of a priority encoder, an arbiter and an axis register module to be used on both slave and master sides.
The AXIS RAM Switch is configured from the AXI interface on the rising edge of the AXI bus clock. All the modules interfaced with this AXIS RAM Switch run on the same AXI bus clock and hence follow a pipelined structure. The Switch generates the address configuration based on the configuration and then the arbitration module routes the traffic between the slave-master pairs. This Switch forms the base for an AXIS Interconnect.
The whole module combines sequential and combinational logic to make sure it operates effectively without breaking. The inclusion of a shared RAM enables the switch to have a buffered filter between the routed traffic. This buffered filter allows to filter unwanted chunks of data and makes sure that the the traffic is routed as efficiently as possible by scaling the data by a speedup factor that comes innately with the RAM. This RAM also enables some status flags so a coupled DUT can know when the RAM is overflown or when there has been bad frames of data. This essentially eliminates the need to add FIFO or other buffered memory to a normal AXIS Switch. The embedded **Bus Width Adaptor** makes sure that the bus widths remain coherent around the scaling of the data caused by the RAM buffer.

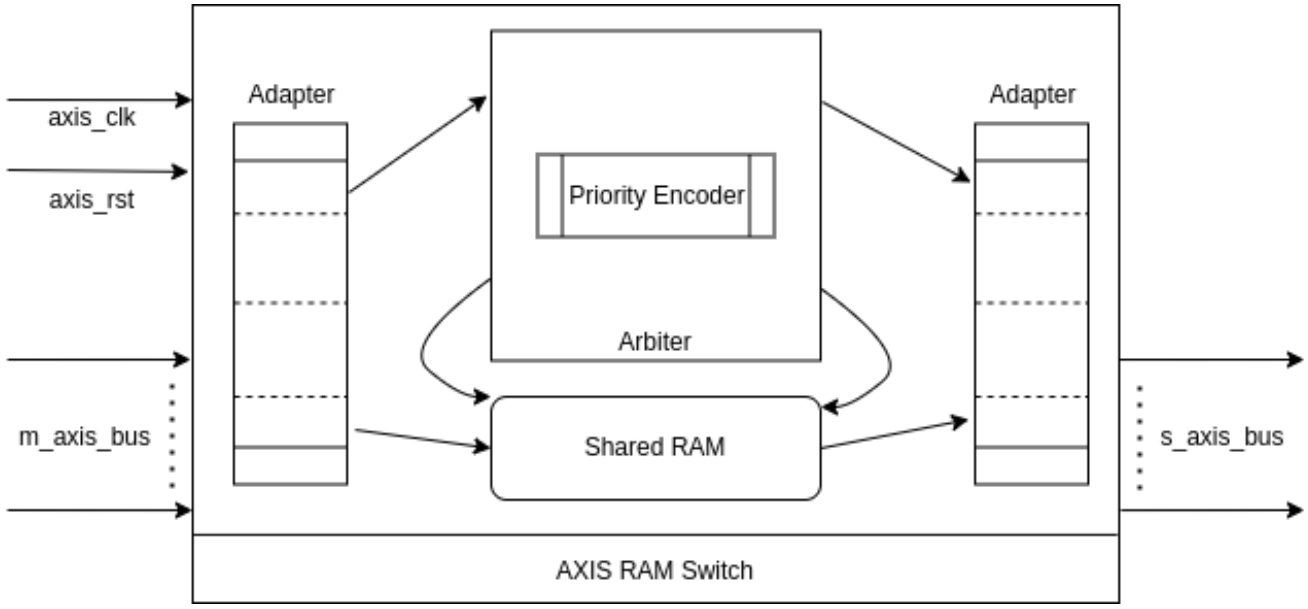


Figure 2. Top Module

Standards

The AXI4-Stream Slave interface is compliant with the AMBA® AXI Protocol Specification.

IP Support Details

The Table 1 gives the support details for AXIS RAM Switch.

Compliance		IP Resources					Tool Flow		
Device	**Interface**	**Source Files**	**Constraint File**	**Testbench**	**Simulation Model**	**Software Driver**	**Analyze and Elaboration**	**Simulation**	**Synthesis**
GEMINI	AXI4-Stream	Verilog	SDC	Python	Cocotb	Verilator	Raptor	Raptor	Raptor

Table 1. IP Details

Resource Utilization

The parameters for computing the maximum and minimum resource utilization are given in Table 2, remaining parameters have been kept at their default values.

Tool	Raptor Design Suite			
FPGA Device	GEMINI			
Configuration			**Resource Utilization**	
Minimum Resource	**Options**	**Configuration**	**Resources**	**Utilized**
	Master Interfaces	1	LUT	758
	Slave Interfaces	1	Registers	1383
	S ID Width	1	BRAM	3
	LSB High Priority	0	-	-
	Speedup	5	-	-
	Type Round Robin	0	-	-
	FIFO Depth	32	-	-
Maximum Resource	**Options**	**Configuration**	**Resources**	**Utilized**
	Master Interfaces	3	LUT	5091
	Slave Interfaces	8	Registers	7868
	ID En	ON	BRAM	29
	TID En	ON	DSP	3
	Type Round Robin	OFF	-	-
	CMD FIFO Depth	16	-	-
	RAM Pipeline	6	-	-

Table 2. AXIS RAM Switch Resource Utilization

Ports

Table 3 lists the top interface ports of the AXIS RAM Switch.

Signal Name	I/O	Description
AXI Clock and Reset		
clk	I	AXI4-Stream Clock
rst	I	AXI4-Stream RESET
AXI Slave Interface		
s_axis_tdata	I	AXI4-Stream data
s_axis_tkeep	I	AXI4-Stream keep data qualifier
s_axis_tvalid	I	AXI4-Stream valid transfer
s_axis_tready	O	AXI4-Stream transfer ready
s_axis_tlast	I	AXI4-Stream boundary of transfer packet
s_axis_tid	I	AXI4-Stream data stream identifier
s_axis_tdest	I	AXI4-Stream data routing information
s_axis_tuser	I	AXI4-Stream user defined sideband information
AXI Master Interface		
m_axis_tdata	O	AXI4-Stream data
m_axis_tkeep	O	AXI4-Stream keep data qualifier
m_axis_tvalid	O	AXI4-Stream valid transfer
m_axis_tready	I	AXI4-Stream transfer ready
m_axis_tlast	O	AXI4-Stream boundary of transfer packet
m_axis_tid	O	AXI4-Stream data stream identifier
m_axis_tdest	O	AXI4-Stream data routing information
m_axis_tuser	O	AXI4-Stream user defined sideband information
Status Interface		
status_overflow	O	Overflow Signal
status_bad_frame	O	Bad Frame Signal
status_good_frame	O	Good Frame Signal

Table 3. AXIS RAM Switch Interface

Parameters

Table 4 lists the parameters of the AXIS RAM Switch.

Parameter	Values	Default Value	Description
S COUNT	1 - 16	4	Number of Slave Interfaces
M COUNT	1 - 16	4	Number of Master Interfaces
S DATA WIDTH	1 - 1024	8	Data Width for each transfer on slave interface
M DATA WIDTH	1 - 1024	8	Data Width for each transfer on master interface
USER WIDTH	1 - 1024	1	Data Width for user defined sideband information
S ID WIDTH	1 - 16	8	Slave side ID Width
M DEST WIDTH	1 - 8	1	Master side Destination Width
M BASE	0 - 15	0	Address configuration for Master Interface from Base
M TOP	0 - 15	0	Address configuration for Master Interface from Top
ID EN	0 / 1	0	ID Enable
USER EN	0 / 1	1	User Data Enable
LSB HIGH PRIORITY	0 / 1	1	LSB Priority Selection
TYPE ROUND ROBIN	0 / 1	1	Round Robin Architecture
TID	0 / 1	0	Enable update of Transfer ID
FIFO DEPTH	8, 16, 32, ..., 32768	4096	Depth of FIFO
CMD FIFO DEPTH	8, 16, 32, ..., 1024	32	Depth of Command FIFO
SPEEDUP	0 - 100	32	Data Width Scaling factor
CMD FIFO DEPTH	8, 16, 32, ..., 1024	32	Depth of Command FIFO
RAM PIPELINE	0 - 32	2	Pipeline stages for RAM
BAD FRAME VALUE	0 - 99	1	Value for bad frame marker
BAD FRAME MASK	0 - 99	1	Mask for bad frame marker
DROP WHEN FULL	0 / 1	0	Drop incoming frames when FIFO is full
DROP BAD FRAME	0 / 1	0	Drop frames that are marked bad
IP TYPE	-	RAMSWTCH	Type of Peripheral
IP VERSION	-	<ip_version>	Version of Peripheral
IP ID	-	<date_and_time>	Date and Time of the generated Peripheral

Table 4. Parameters

Design Flow

IP Customization and Generation

AXIS RAM Switch IP core is a part of the Raptor Design Suite Software. A customized AXIS RAM Switch can be generated from the Raptor's IP configurator window as shown in Figure 3.

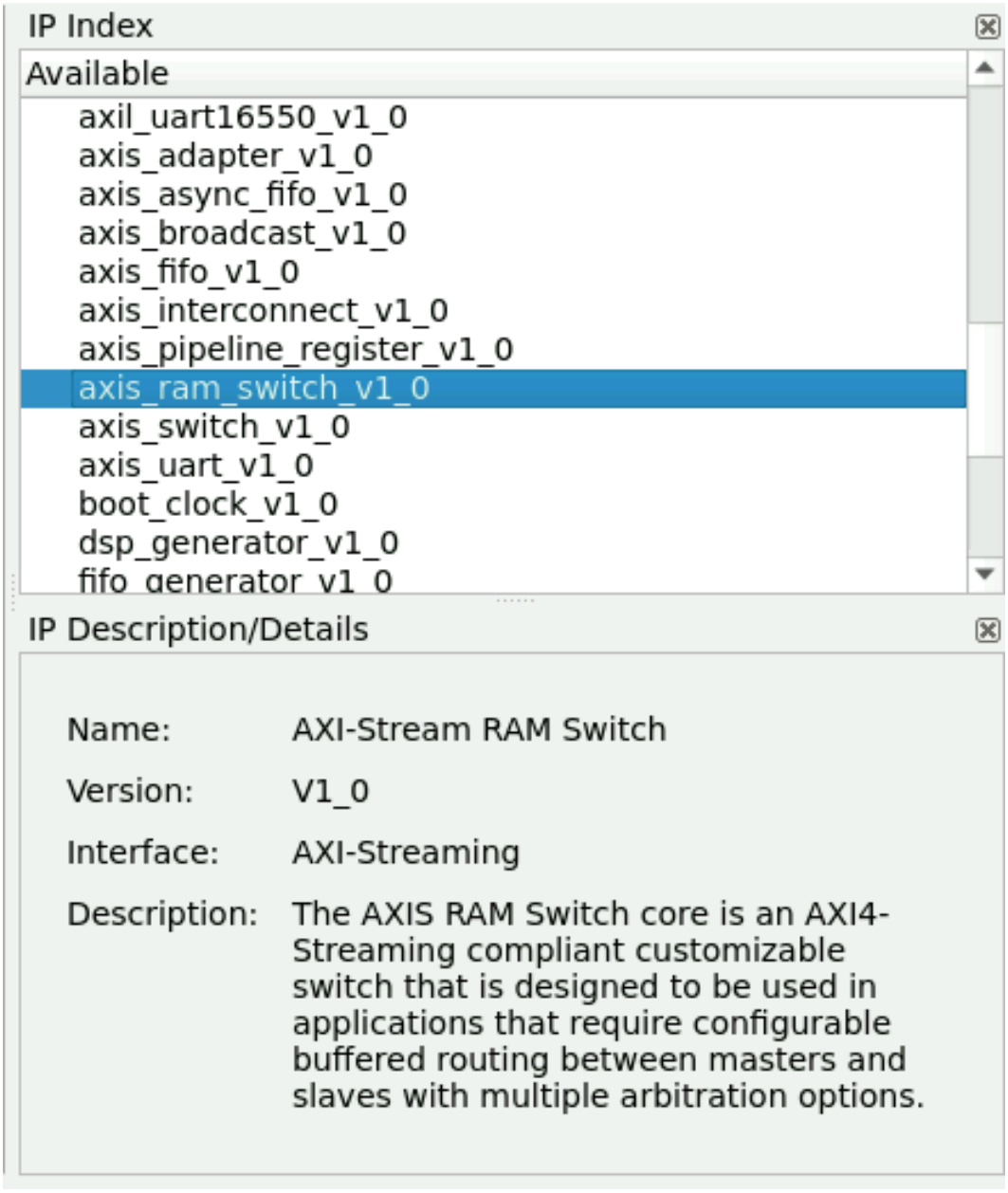


Figure 3. IP list

Parameters Customization: From the IP configuration window, the parameters of the RAM Switch can be configured and the RAM Switch features can be enabled for generating a customized Switch IP core that suits the user application requirement as shown in Figure 4. After IP Customization, all the source files are made available to the user with a top wrapper that instantiates a parameterized instance of the AXIS RAM Switch.

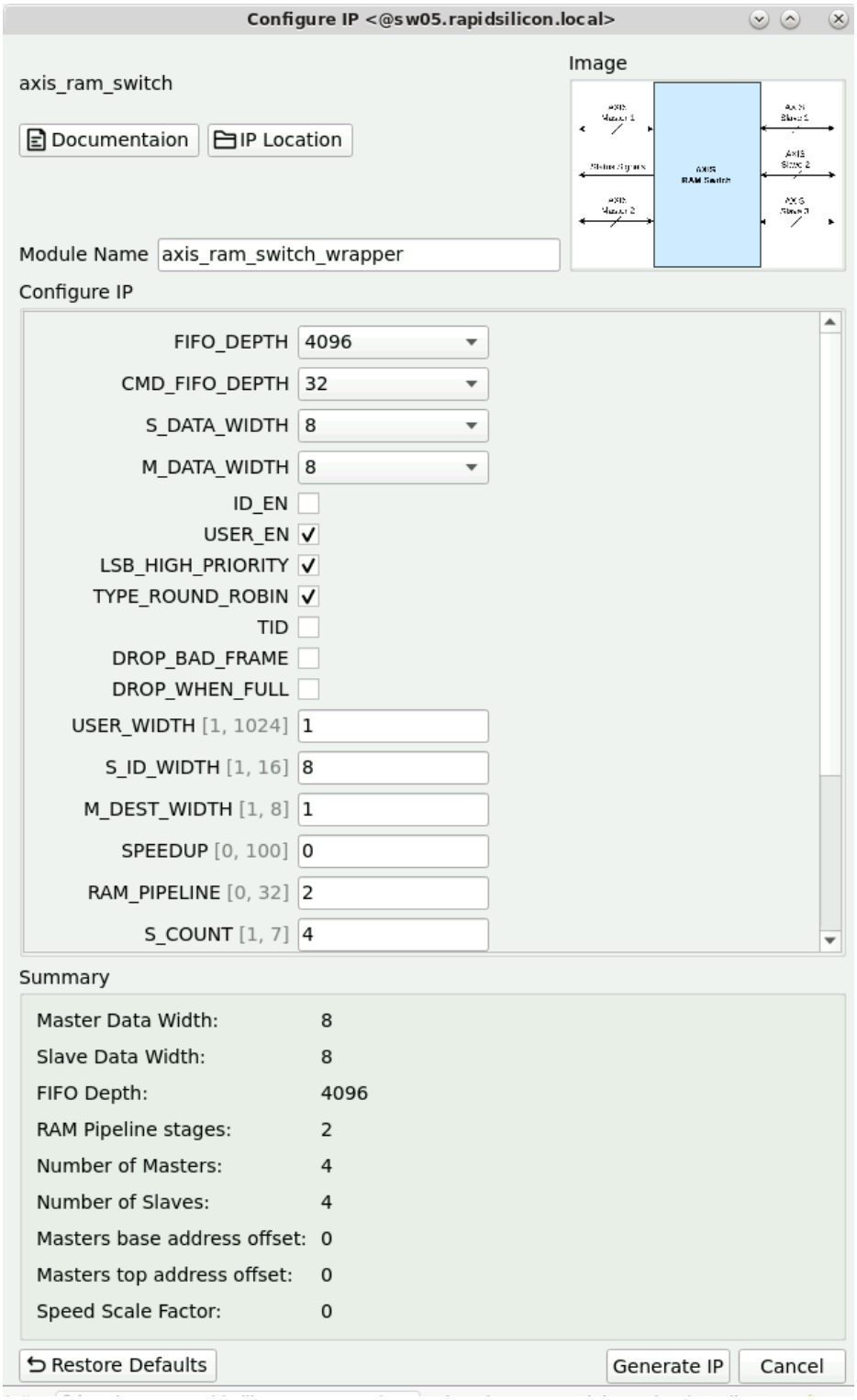


Figure 4. IP Configuration

Example Design

Overview

This AXIS RAM Switch can be utilized in any system that has multiple master and slave pairs and there is a need to route the traffic between them. This RAM Switch can also be utilized to make an Interconnect without adding FIFOs due to the included RAM. The included RAM eradicates the need of any external buffer all the while increasing the frequency of the data routing. Macro block diagram of one such design where a RAM Switch is utilized in an Interconnect can be visualized in Figure 5.

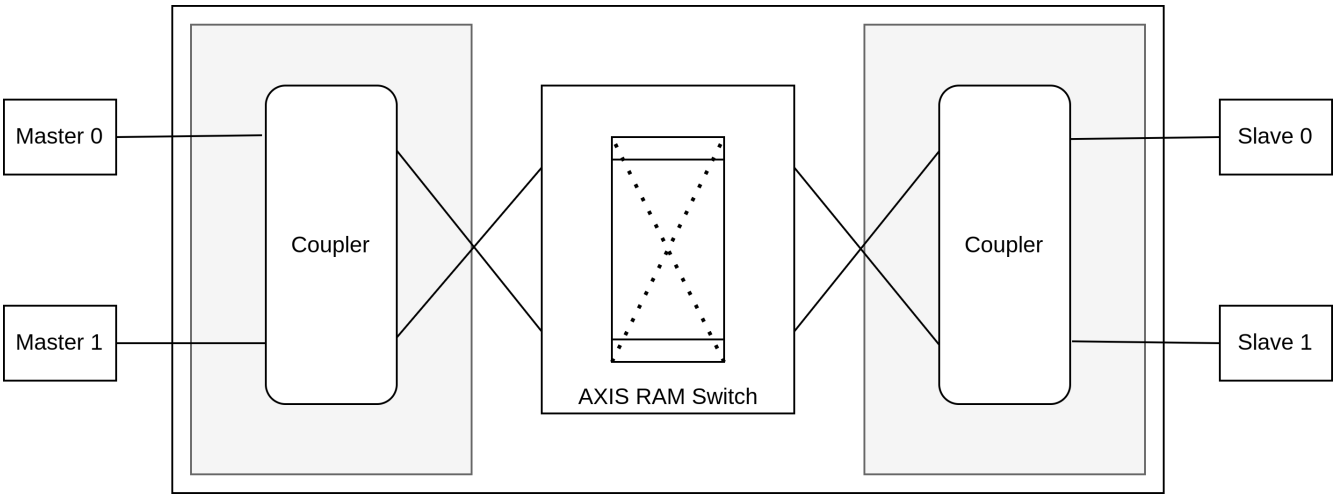


Figure 5. AXIS RAM Switch in Interconnect

Simulating the Example Design

The IP being Verilog HDL, can be simulated via a bunch of industry standard stimulus. For instance, it could be simulated via writing a Verilog Test-bench, or incorporating a soft processor that can stimulate this RAM Switch. The bundled example design is stimulated via a Coco-tb based environment that iteratively stimulates all the master/slave pairs while also stress testing the data routing between them.

Synthesis and PnR

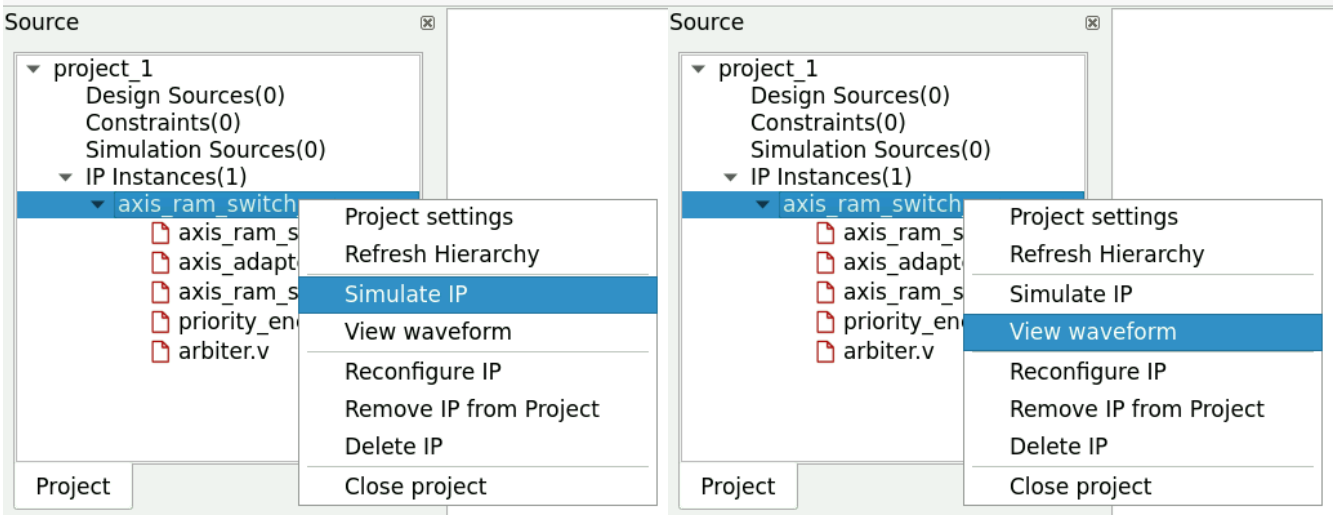
Raptor Suite is armed with tools for **Synthesis** along with **Post and Route** capabilities and the generated post-synthesis and post-route and place netlists can be viewed and analyzed from within the Raptor. The generated bitstream can then be uploaded on an FPGA device to be utilized in hardware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

Test Bench

A Coco-tb based test bench can be found in the **/sim** repository formed after the generation of the IP. It generates a Switch IP with the following parameters: -

- ID EN = 1
- TID = 1
- S ID WIDTH = 16
- M DEST WIDTH = 8
- S COUNT = 4
- M COUNT = 4

This test environment can be simulated with any Verilog HDL simulator of choice e.g., Verilator or Icarus. This simulates the generated IP under various test conditions including stimulating all master and slave interfaces individually and then some stress test cases where all the interfaces are stimulated together. This makes up for a total of 25 tests upon passing of which the IP can be verified functionally. The simulation can be easily run by clicking the "Simulate IP" button as shown in figure 6a. The waveform can also be viewed via the built-in wave-veiwer by clicking the "View waveform" button as shown in figure 6b.



(a) Simulate IP Window **(b)** View Waveform Window

Figure 6. IP Source Window

The simulation results are also displayed in the console window a glimpse of which can be seen in figure 7.

```
**************************************************************************
** TEST                                                STATUS  SIM TIME (ns)
**************************************************************************
** test_axis_ram_switch.run_test_001                   PASS       11530.00
** test_axis_ram_switch.run_test_002                   PASS       11540.00
** test_axis_ram_switch.run_test_003                   PASS       11540.00
** test_axis_ram_switch.run_test_004                   PASS       11540.00
** test_axis_ram_switch.run_test_005                   PASS       28450.00
** test_axis_ram_switch.run_test_006                   PASS       28450.00
** test_axis_ram_switch.run_test_007                   PASS       28450.00
** test_axis_ram_switch.run_test_008                   PASS       28450.00
** test_axis_ram_switch.run_test_009                   PASS       27160.00
** test_axis_ram_switch.run_test_010                   PASS       27160.00
** test_axis_ram_switch.run_test_011                   PASS       27160.00
** test_axis_ram_switch.run_test_012                   PASS       27160.00
** test_axis_ram_switch.run_test_013                   PASS       44090.00
** test_axis_ram_switch.run_test_014                   PASS       44090.00
** test_axis_ram_switch.run_test_015                   PASS       44090.00
** test_axis_ram_switch.run_test_016                   PASS       44090.00
** test_axis_ram_switch.run_test_tuser_assert_001      PASS         850.00
** test_axis_ram_switch.run_test_tuser_assert_002      PASS         850.00
** test_axis_ram_switch.run_test_tuser_assert_003      PASS         850.00
** test_axis_ram_switch.run_test_tuser_assert_004      PASS         850.00
** test_axis_ram_switch.run_arb_test_001               PASS        1170.00
** test_axis_ram_switch.run_stress_test_001            PASS       14030.00
** test_axis_ram_switch.run_stress_test_002            PASS       46730.00
** test_axis_ram_switch.run_stress_test_003            PASS       45280.00
** test_axis_ram_switch.run_stress_test_004            PASS       47450.00
**************************************************************************
** TESTS=25 PASS=25 FAIL=0 SKIP=0                                603010.03
**************************************************************************
```

Figure 7. Simulation Results

Revision History

Date	Version	Revisions
December 5, 2023	0.01	Initial version AXIS RAM Switch User Guide Document