AXI Register
(Beta Release)

Version 0.1

May 11, 2023

Copyright

Trademarks

All Rapid Silicon trademarks are as listed at www.rapidsilicon.com. Synopsys and Synplify Pro are trademarks of Synopsys, Inc. Aldec and Active-HDL are trademarks of Aldec, Inc. Modelsim and Questa are trademarks or registered trademarks of Siemens Industry Software Inc. or its subsidiaries in the United States or other countries. All other trademarks are the property of their respective owners.

Disclaimers

NO WARRANTIES: THE INFORMATION PROVIDED IN THIS DOCUMENT IS “AS IS” WITHOUT ANY EXPRESS OR IMPLIED WARRANTY OF ANY KIND INCLUDING WARRANTIES OF ACCURACY, COMPLETENESS, MERCHANTABILITY, NONINFRINGEMENT OF INTELLECTUAL PROPERTY, OR FITNESS FOR ANY PARTICULAR PURPOSE. IN NO EVENT WILL RAPID SILICON OR ITS SUPPLIERS BE LIABLE FOR ANY DAMAGES WHATSOEVER (WHETHER DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL, INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION, OR LOSS OF INFORMATION) ARISING OUT OF THE USE OF OR INABILITY TO USE THE INFORMATION PROVIDED IN THIS DOCUMENT, EVEN IF RAPID SILICON HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. BECAUSE SOME JURISDICTIONS PROHIBIT THE EXCLUSION OR LIMITATION OF CERTAIN LIABILITY, SOME OF THE ABOVE LIMITATIONS MAY NOT APPLY TO YOU.

Rapid Silicon may make changes to these materials, specifications, or information, or to the products described herein, at any time without notice. Rapid Silicon makes no commitment to update this documentation. Rapid Silicon reserves the right to discontinue any product or service without notice and assumes no obligation to correct any errors contained herein or to advise any user of this document of any correction if such be made. Rapid Silicon recommends its customers obtain the latest version of the relevant information to establish that the information being relied upon is current and before ordering any products.

Contents

IP Summary

Introduction

An AXI Register is an IP core that provides a simple interface to read and write registers in a system-on-chip (SoC) design. The AXI protocol is a widely-used, industry-standard bus interface that enables high-speed communication between different modules in a SoC. This IP Core acts as a bridge between the AXI bus and the registers in the design. It can be customized to include any number of registers, each with a specific address and width. The IP core supports both read and write operations, allowing software running on a processor or other modules in the SoC to access the registers. The benefits of it include reduced design time, increased reliability, and improved system performance.

Features

- AXI4 (memory mapped) one master and one slave interface
- Configurable data width 8, 16, 32, 64, 128, 256, 512 and 1024 bits
- Configurable address width up to 64 bits
- Support ID width up to 32 bits
- Register options i.e. bypass, simple buffer or skid buffer for each channel
- Compatible with AXI4 Interconnect

Overview

AXI Register

The AXI Register IP Core is a part of Raptor Design Suite that can be integrated into a larger SoC design to provide AXI register functionality. It typically includes a set of memory-mapped control registers that are accessed by the processor using memory access instructions. The IP core also includes the necessary logic to interface with the AXI bus and respond to register accesses. The IP core can also be customized and optimized for specific design requirements, such as reducing power consumption or improving performance. It offers flexibility in customization and optimization to meet specific design requirements, which can help reduce the time and cost of development. Additionally, as a pre-designed and pre-verified block of logic, it can speed up the design process and ensure reliability.

Figure 1: AXI Register Block Diagram

IP Specification

The AXI Register is a IP core based on AMBA AXI Protocol which is a widely used, industry-standard bus interface that enables high-speed communication between different modules in an SoC. It has configurable register width and the width of the registers may range from 8 bits to 1024 bits, depending on the specific requirements of the SoC design. It support both read and write operations, allowing software running on a processor or other modules in the SoC to access the registers. It also supports up to 1024 bits for user signals. The IP core is comply with the AXI protocol and be compatible with other AXI-compliant modules in the SoC.

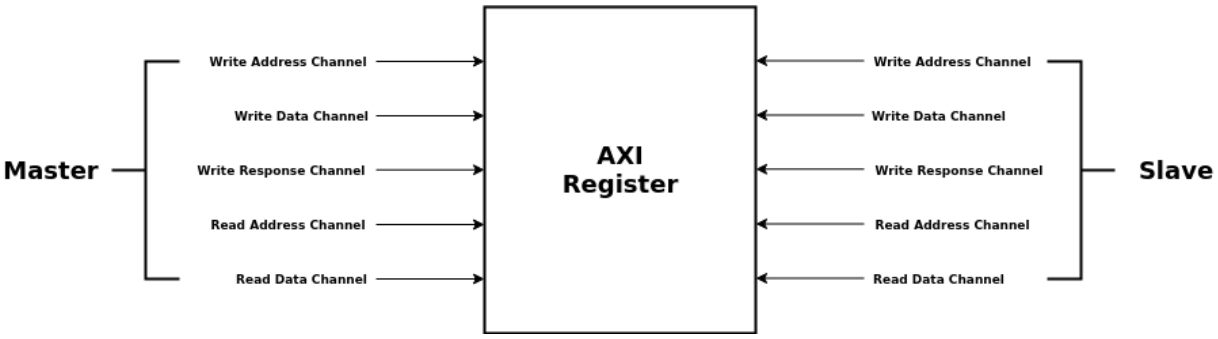

Figure 2: Top Module

Type of Registers

The "<channel>_REG_TYPE" parameter in an AXI Register IP Core specifies how the register behaves when data is written to it. There are three types of registers that can be selected: Bypass, Simple Buffer, and Skid Buffer.

- **Bypass Register:**
  A Bypass register is the simplest type of register. When data is written to it, the register immediately transfers the data to the output without storing it. This type of register is useful when there is no need to store the data and it can be directly passed on to the next module. The bypass register provides the fastest and most efficient data transfer but does not store the data.
- **Simple Buffer:**
  A Simple Buffer register stores the data temporarily and releases it when requested. When data is written to a simple buffer register, it is stored in the register and is available for reading until it is overwritten. This type of register is useful when the data needs to be processed before being passed on to the next module. For example, a simple buffer register could be used to store data that needs to be processed by a CPU or a DSP core.
- **Skid Buffer:**
  A Skid Buffer register is similar to a Simple Buffer register, but it has an additional feature called "skidding." Skidding means that if new data is written to the register before the old data is read, the new data overwrites the old data. This behavior can be useful when only the most recent data needs to be kept. Skid buffer registers are often used in real-time applications, such as video or audio processing, where it is important to have the most recent data.

Standards

The AXI4 Master and Slave interfaces are compliant with the AMBA® AXI Protocol Specification.

IP Support Details

The Table 1 gives the support details for AXI Register.

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

Table 1: Support Details

Resource Utilization

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

Tool	Raptor Design Suite			
FPGA Device	GEMINI			
Configuration			**Resource Utilization**	
Minimum Resource	**Options**	**Configuration**	**Resources**	**Utilized**
	DATA_WIDTH	8	BRAMS	1
	ADDR_WIDTH	1	REGISTERS	94
	ID_WIDTH	1	LUTS	42
	AW_REG_TYPE	Bypass	-	-
Maximum Resource	**Options**	**Configuration**	**Resources**	**Utilized**
	DATA_WIDTH	1024	BRAMS	1
	ADDR_WIDTH	64	REGISTERS	4854
	ID_WIDTH	32	LUTS	2365
	AW_REG_TYPE	Skid_Buffer	-	-

Table 2: Resource Utilization

Ports

Table 3 lists the top interface ports of the AXI Register.

Signal Name	I/O	Description
clk	I	Clock Signal of Register
rst	I	Active High Synchronous Reset Signal
Slave Write Address Channel		
s_axi_awid	I	Write address ID
s_axi_awaddr	I	Write address
s_axi_awlen	I	Burst length
s_axi_awsize	I	Burst size
s_axi_awburst	I	Burst type
s_axi_awlock	I	Lock type
s_axi_awcache	I	Memory type
s_axi_awprot	I	Protection type
s_axi_awvalid	I	Write address valid
s_axi_awready	O	Write address ready
Slave Write Data Channel		
s_axi_wdata	I	Write data
s_axi_wstrb	I	Write strobe
s_axi_wlast	I	Write last
s_axi_wvalid	I	Write valid
s_axi_wready	O	Write ready
Slave Write Response Channel		
s_axi_bid	O	Response ID tag
s_axi_bresp	O	Write response
s_axi_bvalid	O	Write response valid
s_axi_bready	I	Write response ready
Slave Read Address Channel		
s_axi_arid	I	Read address ID
s_axi_araddr	I	Read address
s_axi_arlen	I	Burst length
s_axi_arsize	I	Burst size
s_axi_arburst	I	Burst type
s_axi_arlock	I	Lock type
s_axi_arcache	I	Memory type
s_axi_arprot	I	Protection type
s_axi_arvalid	I	Read address valid
s_axi_arready	O	Read address ready
Slave Read Data Channel		
s_axi_rid	O	Read ID tag
s_axi_rdata	O	Read data
s_axi_rresp	O	Read response

Signal Name	I/O	Description
s_axi_rlast	O	Read last
s_axi_rvalid	O	Read valid
s_axi_rready	I	Read ready
Master Write Address Channel		
m_axi_awid	O	Write address ID
m_axi_awaddr	O	Write address
m_axi_awlen	O	Burst length
m_axi_awsize	O	Burst size
m_axi_awburst	O	Burst type
m_axi_awlock	O	Lock type
m_axi_awcache	O	Memory type
m_axi_awprot	O	Protection type
m_axi_awvalid	O	Write address valid
m_axi_awready	I	Write address ready
Master Write Data Channel		
m_axi_wdata	O	Write data
m_axi_wstrb	O	Write strobe
m_axi_wlast	O	Write last
m_axi_wvalid	O	Write valid
m_axi_wready	I	Write ready
Master Write Response Channel		
m_axi_bid	I	Response ID tag
m_axi_bresp	I	Write response
m_axi_bvalid	I	Write response valid
m_axi_bready	O	Write response ready
Master Read Address Channel		
m_axi_arid	O	Read address ID
m_axi_araddr	O	Read address
m_axi_arlen	O	Burst length
m_axi_arsize	O	Burst size
m_axi_arburst	O	Burst type
m_axi_arlock	O	Lock type
m_axi_arcache	O	Memory type
m_axi_arprot	O	Protection type
m_axi_arvalid	O	Read address valid
m_axi_arready	I	Read address ready
Master Read Data Channel		
m_axi_rid	I	Read ID tag
m_axi_rdata	I	Read data
m_axi_rresp	I	Read response
m_axi_rlast	I	Read last
m_axi_rvalid	I	Read valid

Signal Name	I/O	Description
m_axi_rready	O	Read ready

Table 3: Port List

Parameters

Table 4 lists the parameters of the AXI Register.

Parameter	Values	Default Value	Description
DATA_WIDTH	8, 16, 32, 64, 128, 256, 512, 1024	32	Data Width of Register
ADDR_WIDTH	1 - 64	32	Address Width of Register
ID_WIDTH	1 - 32	32	ID field of Register
AW_USER_WIDTH	1 - 1024	1	User Field for AW Channel
W_USER_WIDTH	1 - 1024	1	User Field for W Channel
B_USER_WIDTH	1 - 1024	1	User Field for B Channel
AR_USER_WIDTH	1 - 1024	1	User Field for AR Channel
R_USER_WIDTH	1 - 1024	1	User Field for R Channel
AW_REG_TYPE	Bypass, Simple_Buffer, Skid_Buffer	Simple_Buffer	Register Type for AW Channel
W_REG_TYPE	Bypass, Simple_Buffer, Skid_Buffer	Skid_Buffer	Register Type for W Channel
B_REG_TYPE	Bypass, Simple_Buffer, Skid_Buffer	Simple_Buffer	Register Type for B Channel
AR_REG_TYPE	Bypass, Simple_Buffer, Skid_Buffer	Simple_Buffer	Register Type for AR Channel
R_REG_TYPE	Bypass, Simple_Buffer, Skid_Buffer	Skid_Buffer	Register Type for R Channel

Table 4: Parameters

Design Flow

IP Customization and Generation

AXI Register IP core is a part of the Raptor Design Suite Software. A customized register can be generated from the Raptor's IP configuration window as shown in figure 3.

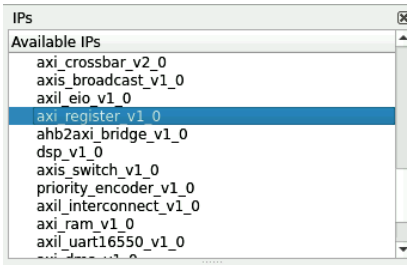

Figure 3: IP List

Parameters Customization

From the IP configuration window, the parameters of the AXI Register can be configured and it's features can be enabled for generating a customized IP core that suits the user application requirements. All parameters are shown in figure 4. In Figure 4, the module name specifies the name of both the Verilog file and the top-level IP name that will be generated based on above configured parameters. The Output Dir is a directory option that allows the user to specify where they want the generated IP to be saved.

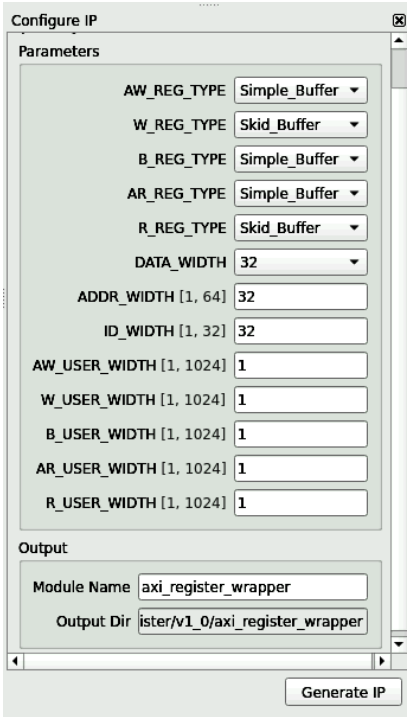

Figure 4: IP Configuration

Test Bench

The AXI Register IP is provided with a testbench which is based upon Cocotb verification environment. In this test, slave interface is connected to AXI Master and master interface is connected with AXI RAM. The input data is generated using a test data generator module. Input data is routed from master to slave through register. The output data is compared with the expected output data to verify the correctness of the IP core's operation. The dump file is generated to view the output of the test. In the end, there is status for passing or failure of the test.

Revision History

Date	Version	Revisions
May 11, 2023	0.1	Initial version AXI Register User Guide