AXI4-Lite GPIO
(Beta Release)

Version 0.1

May 4, 2023

Contents

IP Summary

Introduction

GPIO stands for General-Purpose Input/Output. It refers to a type of interface found on microcontrollers and single-board computers that allows them to connect to and interact with a wide variety of external devices. GPIO pins can be configured to function as either inputs or outputs, and they can be used to read digital signals or generate digital signals, respectively. This allows them to be used for a wide range of applications, such as controlling LEDs, reading buttons, and communicating with sensors. This GPIO IP is AXILite compliant and hence can be used in a bunch of AXI based systems.

Features

- Configurable data width selection between 32 and 64 bits.
- Configurable address width from 8 to 16.
- Modular and independent read and write embedded modules.
- Supports the AXI4-Lite interface specification.

Overview

AXIL GPIO

AXIL GPIO provides the flexibility of configuring each pin as either an input or an output port depending on the usability of the application. The IP core also supports interrupts, which can be used to notify the processor when an input signal changes state. The AXI GPIO IP core is a commonly used IP block in FPGA-based designs and this IP core is delivered in a hardware description language (HDL), Verilog, which is be used to customize the core to meet specific design requirements. A block diagram for the AXI-Lite GPIO IP is shown in Figure 1.

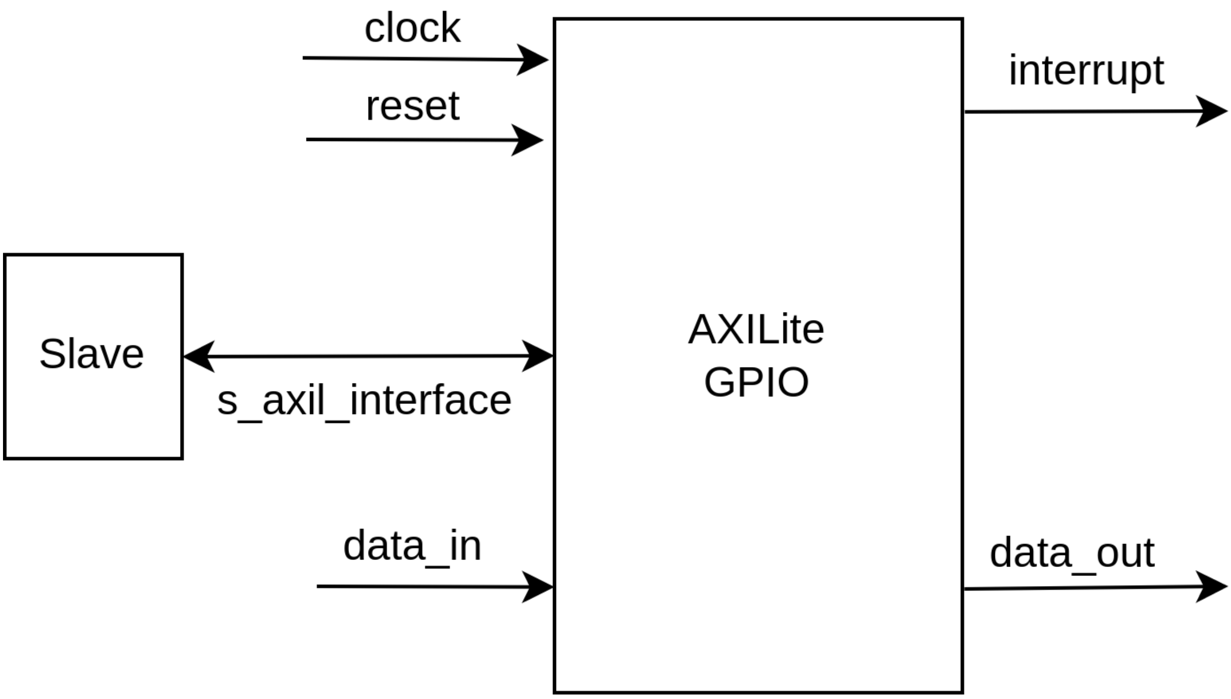

Figure 1: AXIL GPIO Block Diagram

IP Specification

AXIL GPIO IP is an intellectual property (IP) core that enables designers to easily integrate General Purpose Input/Output (GPIO) pins into their FPGA and SoC designs. GPIO pins are commonly used in digital systems to connect peripheral devices such as buttons, sensors, and actuators.
The AXIL GPIO IP provides a simple interface to control GPIO pins through the Advanced eXtensible Interface (AXI) bus, a widely-used interface for connecting IP cores in FPGAs and SoCs. The IP core provides flexible configurations that allow GPIO pins to be configured for a wide range of applications. For example, designers can configure GPIO pins to operate as pulse generators, level detectors, or edge detectors.
The AXIL GPIO IP also supports interrupt generation, which allows the designer to configure interrupts based on various events, such as a rising or falling edge on a particular GPIO pin. Interrupts are an essential feature for many applications, especially those that require real-time processing. The AXIL GPIO IP is easy to integrate into a variety of FPGA and SoC designs and supports software programming interfaces such as C/C++ and Verilog when implemented within an SoC via bare-metal firmwares. This allows the designer to develop software that can interface with the GPIO pins, making it straightforward to control and monitor the state of the pins. The internal block diagram can be seen in Figure 2.

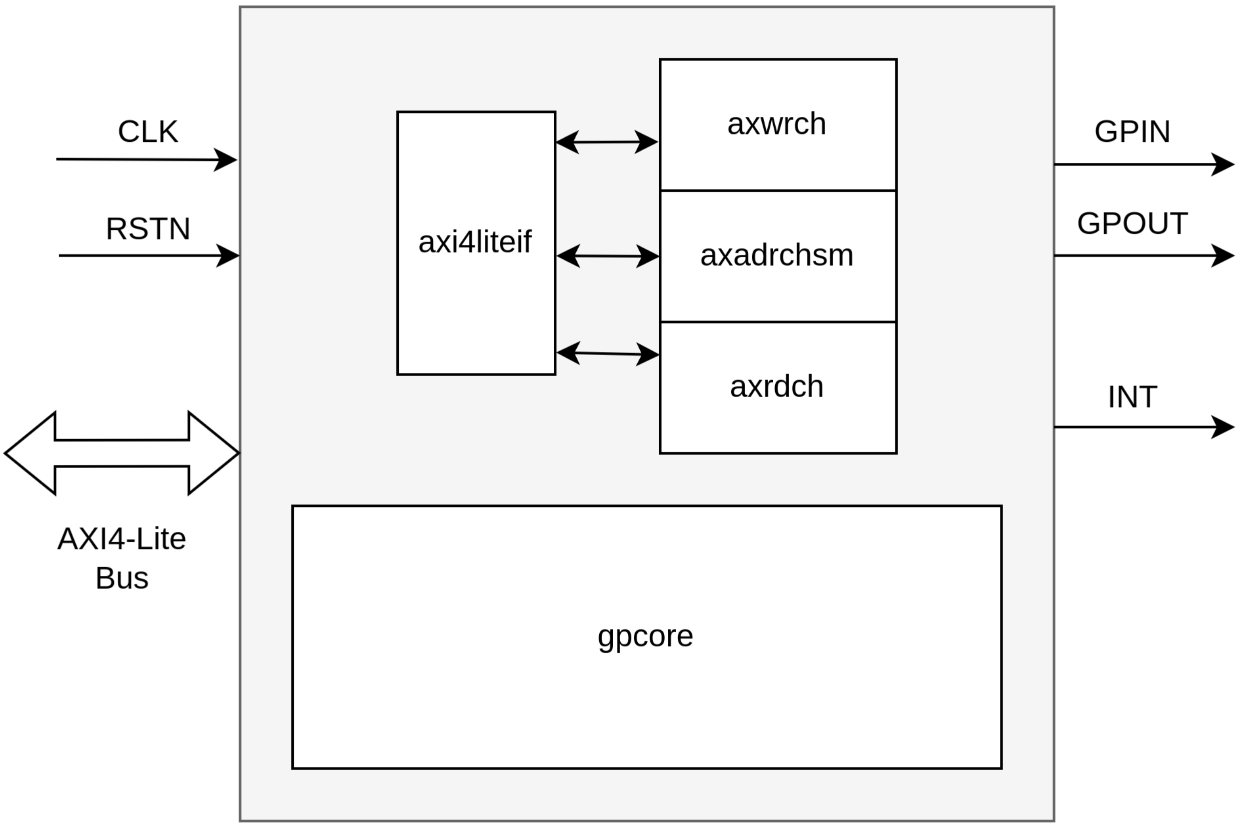

Figure 2: AXIL GPIO Internal Diagram

Standards

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

IP Support Details

The Table 1 gives the support details for AXIL GPIO.

Compliance		IP Resources				Tool Flow		
Device	Interface	Source Files	Constraint Files	Testbench	Simulation Model	Analyze and Elaboration	Simualtion	Synthesis
Gemini	AXI4-Lite	System Verilog	-	-	-	Verific (Raptor)	Icarus (Raptor)	Raptor

Table 1: IP Details

Parameters

Table 2 lists the parameters of the AXIL GPIO.

Parameter	Values	Default Value	Description
ADDR WIDTH	8 - 16	16	Address Width for GPIO
DATA WIDTH	32, 64	32	Data Width for GPIO

Table 2: Parameters

Port List

Table 3 lists the top interface ports of the AXIL GPIO.

Signal Name	I/O	Description
AXI Clock and Reset		
clk	I	System Clock
rstn	I	Active Low Reset
Write Address Channel		
s_axil_awaddr	I	AXI4-Lite write address
s_axil_awprot	I	AXI4-Lite protection data qualifier
s_axil_awvalid	I	AXI4-Lite valid write address
s_axil_awready	O	AXI4-Lite write address ready
Write Data Channel		
s_axil_wdata	I	AXI4-Lite data
s_axil_wstrb	I	AXI4-Lite data stream identifier
s_axil_wvalid	I	AXI4-Lite data valid
s_axil_wready	O	AXI4-Lite data ready
Write Response Channel		
s_axil_bresp	O	AXI4-Lite write response

s_axil_bvalid	O	AXI4-Lite write valid response
s_axil_bready	I	AXI4-Lite write ready response
Read Address Channel		
s_axil_araddr	I	AXI4-Lite read address
s_axil_arprot	I	AXI4-Lite protection data qualifier
s_axil_arvalid	I	AXI4-Lite read address valid
s_axil_arready	O	AXI4-Lite read address ready
Read Data Channel		
s_axil_rdata	O	AXI4-Lite read data
s_axil_rresp	O	AXI4-Lite read response
s_axil_rvalid	O	AXI4-Lite read data valid
s_axil_rready	I	AXI4-Lite read data ready
GPIO Signals		
gpin	O	Serial Input Signal
gpout	I	Serial Output Signal
int_1	O	Interrupt Output

Table 3: AXIL GPIO Interface

Resource Utilization

The parameters for computing the maximum and minimum resource utilization are given in Table 4, remaining parameters have been kept at their default values.

Tool	Raptor Design Suite			
FPGA Device	GEMINI			
Configuration			**Resource Utilized**	
Minimum Resource	**Options**	**Configuration**	**Resources**	**Utilized**
	DATA WIDTH	32	LUTs	220
	ADDR WIDTH	8	Registers	159
Maximum Resource	**Options**	**Configuration**	**Resources**	**Utilized**
	DATA WIDTH	64	LUTs	396
	ADDR WIDTH	16	Registers	287

Table 4: Resource Utilization

Design Flow

IP Customization and Generation

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

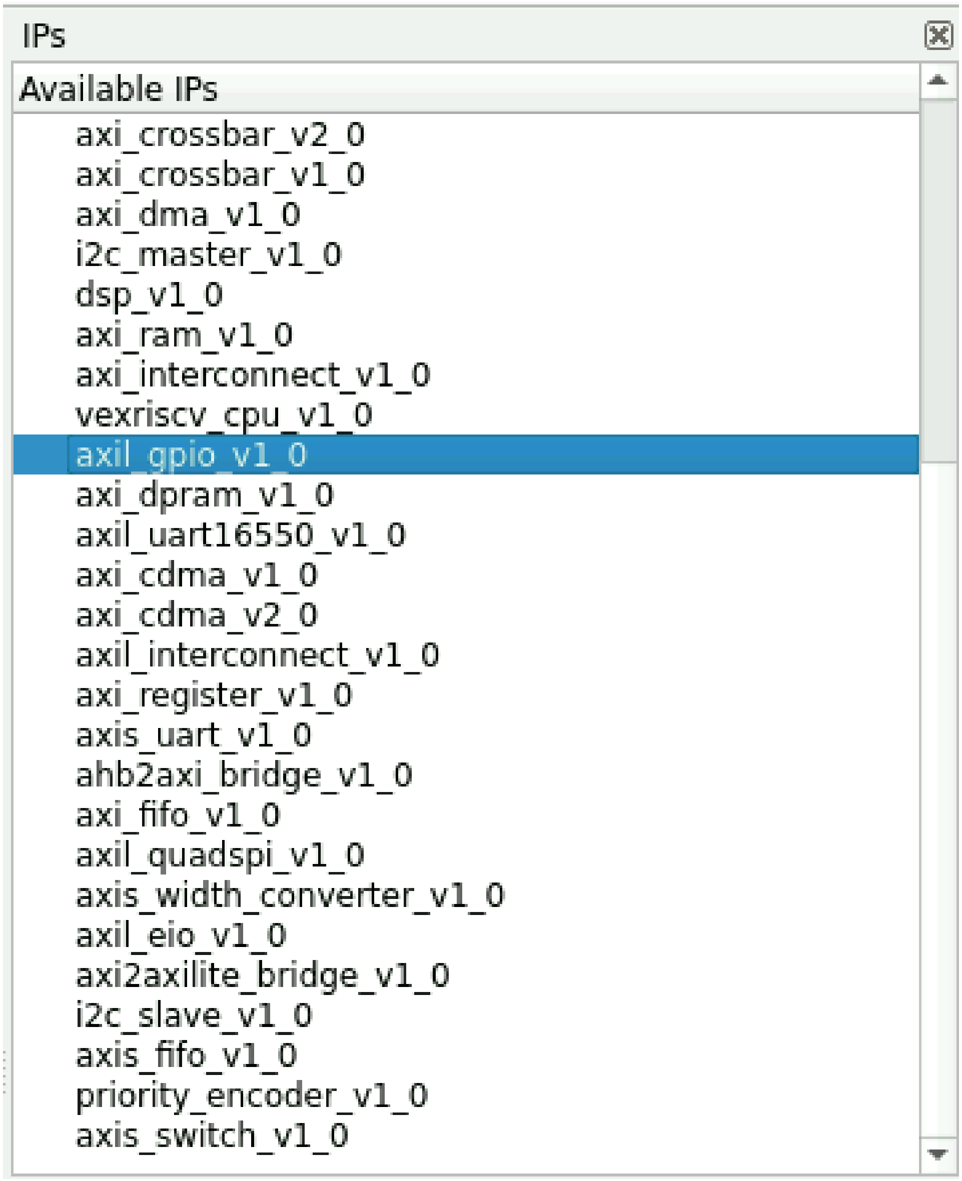

Figure 3: IP list

Parameters Customization

From the IP configuration window, the parameters of the GPIO can be configured and GPIO features can be enabled for generating a customized GPIO 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 AXIL GPIO.

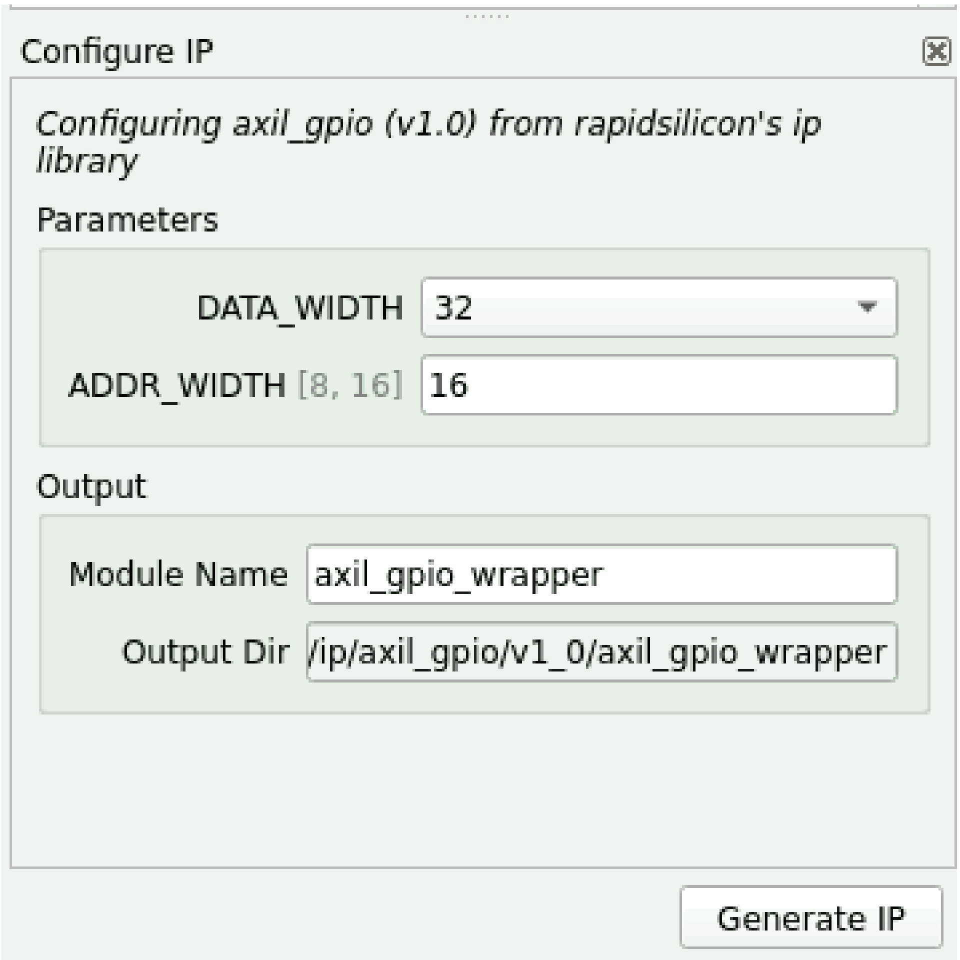

Figure 4: IP Configuration

Example Design

Overview

This AXIL GPIO IP can be utilized in a system that requires sequential transmission and reception of data from the outside world. GPIO is a crucial component in many electronic systems, enabling communication between the system and external devices through a serial interface. It can be embedded inside SoCs to enable two-way communication via the SoC. One such example design of this AXIL GPIO can be visualized in Figure 5.

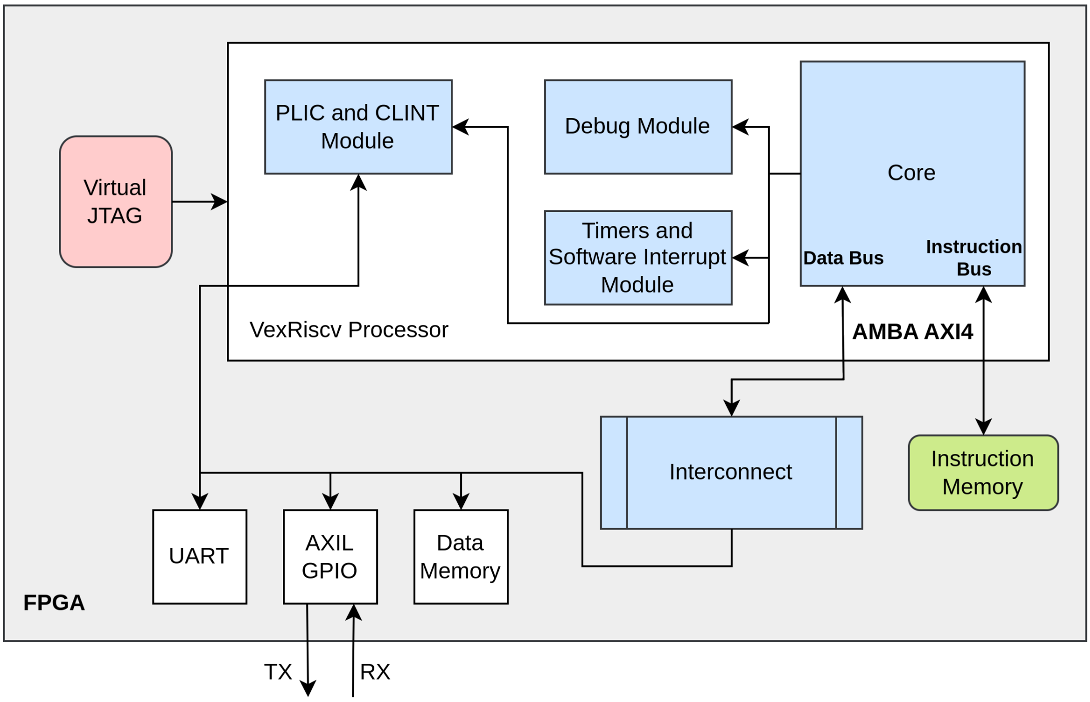

Figure 5: AXIL GPIO inside and SoC

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 GPIO. 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 PR

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

Test Bench

The AXIL GPIO is simulated via incorporating it in an SoC. The SoC is booted up via writing a bare-metal firmware in C / Assembly. The testbench for this AXIL GPIO is incorporating inside this bare-metal firmware in a loopback fashion to make sure that the received data is the same as the one that was transmitted. This firmware is then loaded onto the SoC and the GPIO starts its operation. The clock and reset is given externally via a Verilog testbench file. The bare metal testbench can be enhanced to cover different types of GPIO operations making sure all the GPIO registers are getting hit by the test, ensuring complete coverage and the usability of the GPIO by integration with other AXI based systems and peripherals.

Release

Release History

Date	Version	Revisions
May 4, 2023	0.1	Initial version AXI4-Lite GPIO User Guide Document