Extending Embedded System into PL

Zynq
Vivado 2013.4 Version

Objectives

- **After completing this module, you will be able to:**
  - Identify the IP supplied as part of Vivado
  - Describe how to add hardware to an existing Vivado project
  - Explain how IP is added to extend processing system functionality

Outline

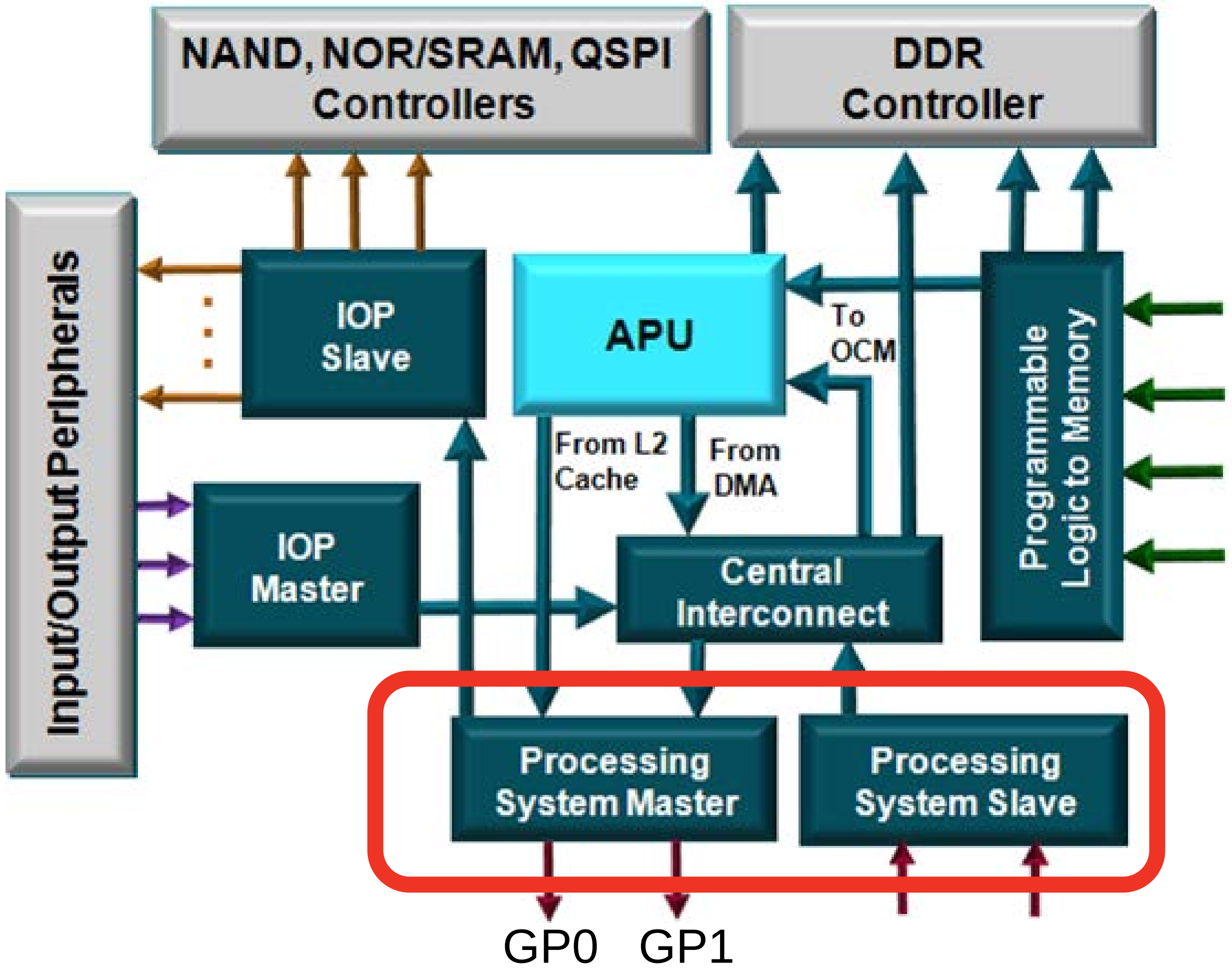
The PS and the PL

- **The Zynq-7000 AP SoC architecture consists of two major sections**
  - PS: Processing system
    - Dual ARM Cortex-A9 processor based
    - Multiple peripherals
    - Hard silicon core
  - **PL: Programmable logic**
    - Shares the same 7 series programmable logic as
      - Artix™-based devices: Z-7010, Z-7015 and Z-7020 (high-range I/O banks only)
      - Kintex™-based devices: Z-7030 and Z-7045, and Z-7100 (mix of high-range and high-performance I/O banks)

- **This sections focuses on the PL**

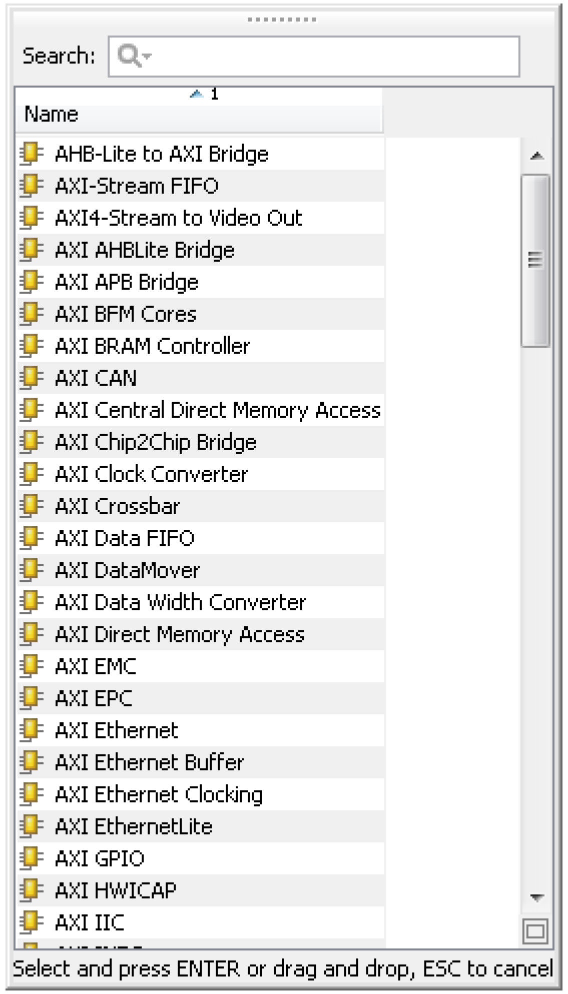
Communicating with PL

- **Processing system master**
  - Two ports from the processing system to programmable logic
  - Connects the CPU block to common peripherals through the central interconnect
- **Processing system slave**
  - Two ports from programmable logic to the processing system
- **Slave PL peripherals address range**
  - 4000_0000 and 7FFF_FFFF (connected to GP0) and
  - 8000_0000 and BFFF_FFFF (connected to GP1)



IP Catalog

- **The IP Catalog contain a collection of IP that can be used to assemble the (embedded) system**
- **Supported by IPI**
- **Facilitates quick system construction**
- **Each IP block has its own configuration parameters**
- **Most of the IP is free, some require licenses**
- **Stored as source code in the install directory**
  - Always synthesized with the latest tools
  - Some proprietary source code is encrypted
- **Peripherals in the PS are always present and can be dynamically enabled or disabled through PS Configuration wizard**


Search:

Name

AHB-Lite to AXI Bridge
AXI-Stream FIFO
AXI4-Stream to Video Out
AXI AHBLite Bridge
AXI APB Bridge
AXI BFM Cores
AXI BRAM Controller
AXI CAN
AXI Central Direct Memory Access
AXI Chip2Chip Bridge
AXI Clock Converter
AXI Crossbar
AXI Data FIFO
AXI DataMover
AXI Data Width Converter
AXI Direct Memory Access
AXI EMC
AXI EPC
AXI Ethernet
AXI Ethernet Buffer
AXI Ethernet Clocking
AXI EthernetLite
AXI GPIO
AXI HWICAP
AXI IIC

Select and press ENTER or drag and drop, ESC to cancel

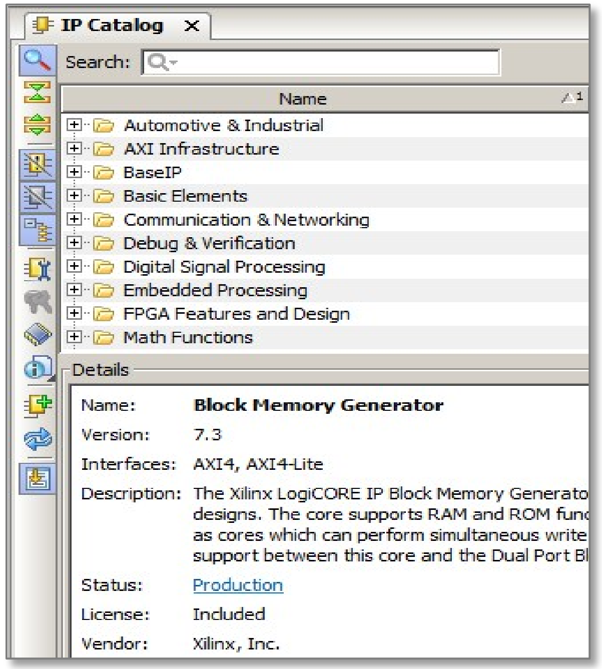
XILINX ALL PROGRAMMABLE.

IP Peripherals
Included as Source (Free)

- **Bus and bridge controllers**
  - AXI to AXI connector
  - Local Memory Bus (LMB)
  - AXI Chip to Chip
  - AHB-Lite to AXI
  - AXI4-Lite to APB
  - AXI4 to AHB-Lite
- **Debug cores**
  - Integrated Logic Analyzer
- **DMA and Timers**
  - Watchdog, fixed interval
- **Inter-processor communication**
- **External peripheral controller Memory and memory controller**
- **High-speed and low-speed communication peripherals**
  - AXI 10/100 Ethernet MAC controller
  - Hard-core tri-mode Ethernet MAC
  - AXI IIC
  - AXI SPI
  - AXI UART
- **Other cores**
  - System monitor
  - Xilinx Analog-to-Digital Converter (XADC)
  - Clock generator
  - System reset module
  - interrupt controller

Vivado IP Catalog

- **Integrated IP Support**
  - Instant access to IP customization
  - Vivado IP GUI look and feel
  - Support for Vivado synthesis and implementation
  - Selectable IP output products
  - Full Tcl support



IP Catalog

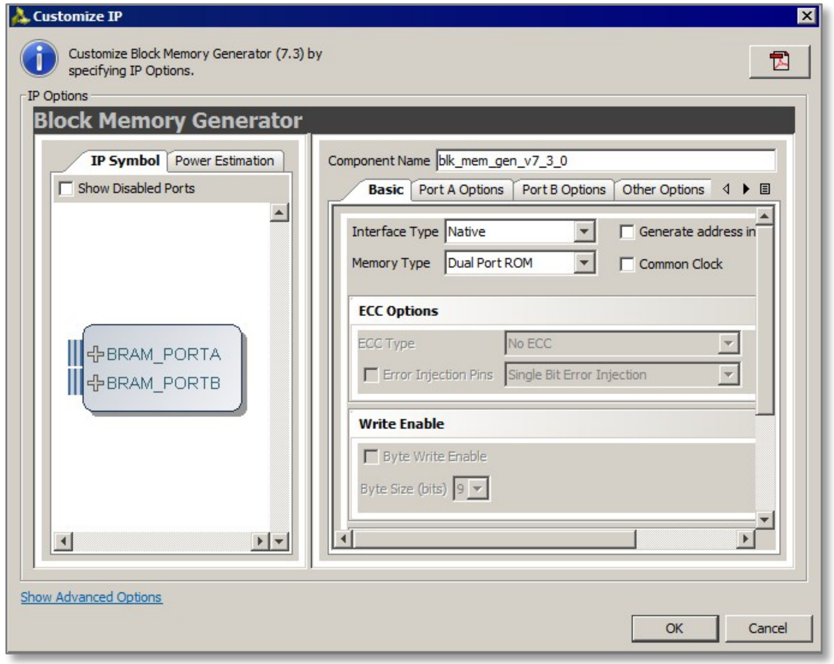
Search:

Name

- Automotive & Industrial
- AXI Infrastructure
- BaseIP
- Basic Elements
- Communication & Networking
- Debug & Verification
- Digital Signal Processing
- Embedded Processing
- FPGA Features and Design
- Math Functions

Details

Name:	**Block Memory Generator**
Version:	7.3
Interfaces:	AXI4, AXI4-Lite
Description:	The Xilinx LogiCORE IP Block Memory Generato designs. The core supports RAM and ROM func as cores which can perform simultaneous write support between this core and the Dual Port Bl
Status:	Production
License:	Included
Vendor:	Xilinx, Inc.



Customize IP

Customize Block Memory Generator (7.3) by specifying IP Options.

IP Options

Block Memory Generator

Component Name: blk_mem_gen_v7_3_0

Interface Type: Native
Memory Type: Dual Port ROM
☐ Generate address in
☐ Common Clock

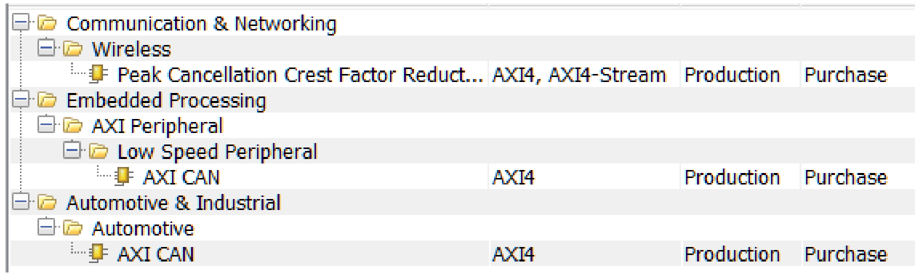
ECC Options
ECC Type: No ECC
☐ Error Injection Pins: Single Bit Error Injection

Write Enable
☐ Byte Write Enable
Byte Size (bits): 9

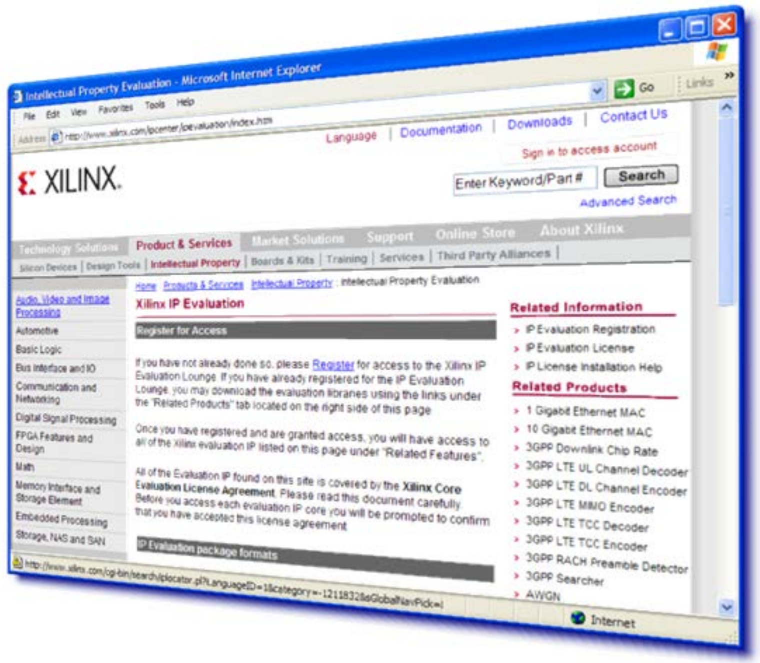
XILINX ALL PROGRAMMABLE.

IP Cores Included as Evaluation

- **AXI CAN controller**
- **AXI USB2 device**
- **Video IP**
- **Telecoms/ Wireless IP**



Communication & Networking			
Wireless			
Peak Cancellation Crest Factor Reduct...	AXI4, AXI4-Stream	Production	Purchase
Embedded Processing			
AXI Peripheral			
Low Speed Peripheral			
AXI CAN	AXI4	Production	Purchase
Automotive & Industrial			
Automotive			
AXI CAN	AXI4	Production	Purchase


Intellectual Property Evaluation - Microsoft Internet Explorer
Language | Documentation | Downloads | Contact Us
Sign in to access account
XILINX
Enter Keyword/Part #
Search
Advanced Search
Technology Solutions
Product & Services
Market Solutions
Support
Online Store
About Xilinx
Silicon Devices | Design Tools | Intellectual Property | Boards & Kits | Training | Services | Third Party Alliances
Audio, Video and Image Processing
Automotive
Basic Logic
Bus Interface and IO
Communication and Networking
Digital Signal Processing
FPGA Features and Design
Math
Memory Interface and Storage Element
Embedded Processing
Storage, NAS and SAN
Related Information
IP Evaluation Registration
IP Evaluation License
IP License Installation Help
Related Products
1 Gigabit Ethernet MAC
10 Gigabit Ethernet MAC
3GPP Downlink Chip Rate
3GPP LTE UL Channel Decoder
3GPP LTE DL Channel Encoder
3GPP LTE MIMO Encoder
3GPP LTE TCC Decoder
3GPP LTE TCC Encoder
3GPP RACH Preamble Detector
3GPP Searcher
AWGN
Internet

Home Products & Services Intellectual Property : Intellectual Property Evaluation

Xilinx IP Evaluation

Register for Access

If you have not already done so, please Register for access to the Xilinx IP Evaluation Lounge. If you have already registered for the IP Evaluation Lounge, you may download the evaluation libraries using the links under the "Related Products" tab located on the right side of this page.

Once you have registered and are granted access, you will have access to all of the Xilinx evaluation IP listed on this page under "Related Features".

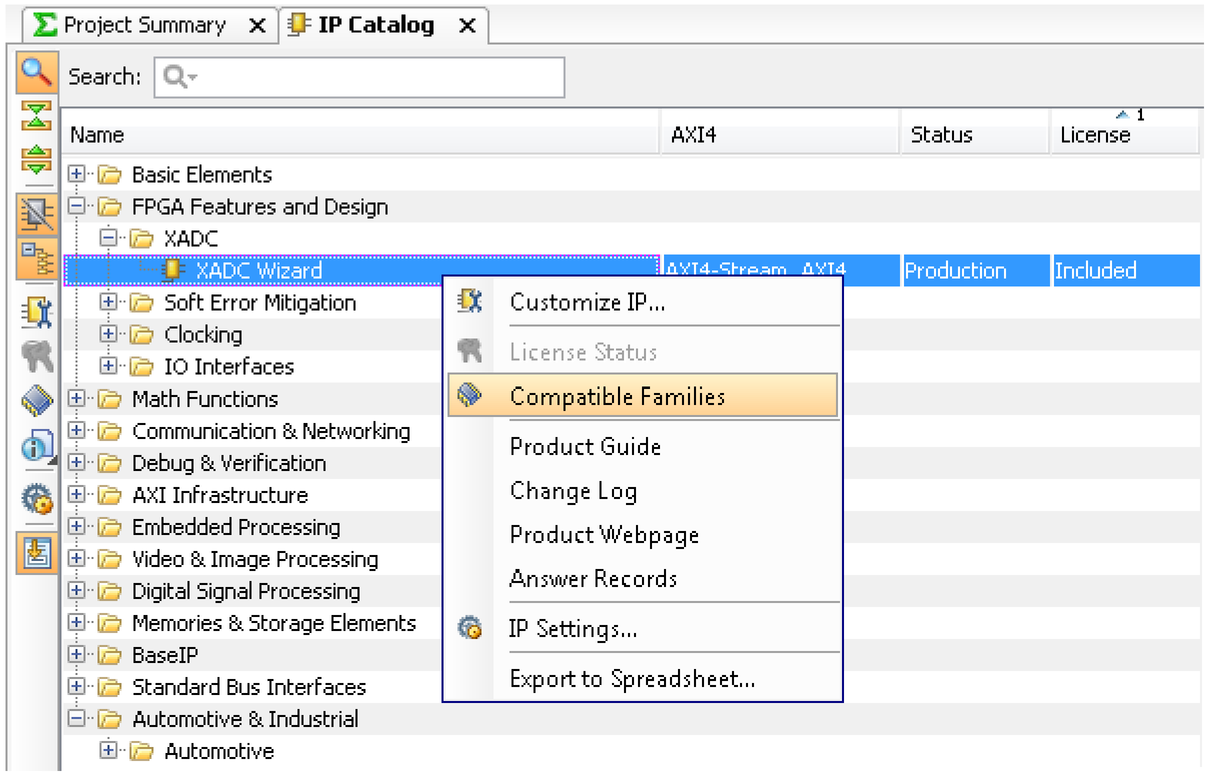
All of the Evaluation IP found on this site is covered by the **Xilinx Core Evaluation License Agreement**. Please read this document carefully. Before you access each evaluation IP core you will be promoted to confirm that you have accepted this license agreement.

IP Evaluation package formats

Xilinx developed, delivered, and supported
Evaluation IP installs with a 90-day evaluation license

IP Cores

- **Right click to -**
  - Add/customize
  - Determine compatibility
  - Product Guide (datasheet) > Document Navigator
  - Change Log
  - Product Webpage
  - Answer record
- **Export complete IP Catalog to excel**



Project Summary | IP Catalog

Search:

Name	AXI4	Status	License
Basic Elements			
FPGA Features and Design			
XADC			
XADC Wizard	AXI4-Stream, AXI4	Production	Included
Soft Error Mitigation			
Clocking			
IO Interfaces			
Math Functions			
Communication & Networking			
Debug & Verification			
AXI Infrastructure			
Embedded Processing			
Video & Image Processing			
Digital Signal Processing			
Memories & Storage Elements			
BaseIP			
Standard Bus Interfaces			
Automotive & Industrial			
Automotive			

IP Core Information

Data sheet provided for each core (right-click on core in IP catalog to access)

- **The size of each core is available in the data sheet**
- **For example, the axi_timer_v2_00_a data sheet contains the following table:**

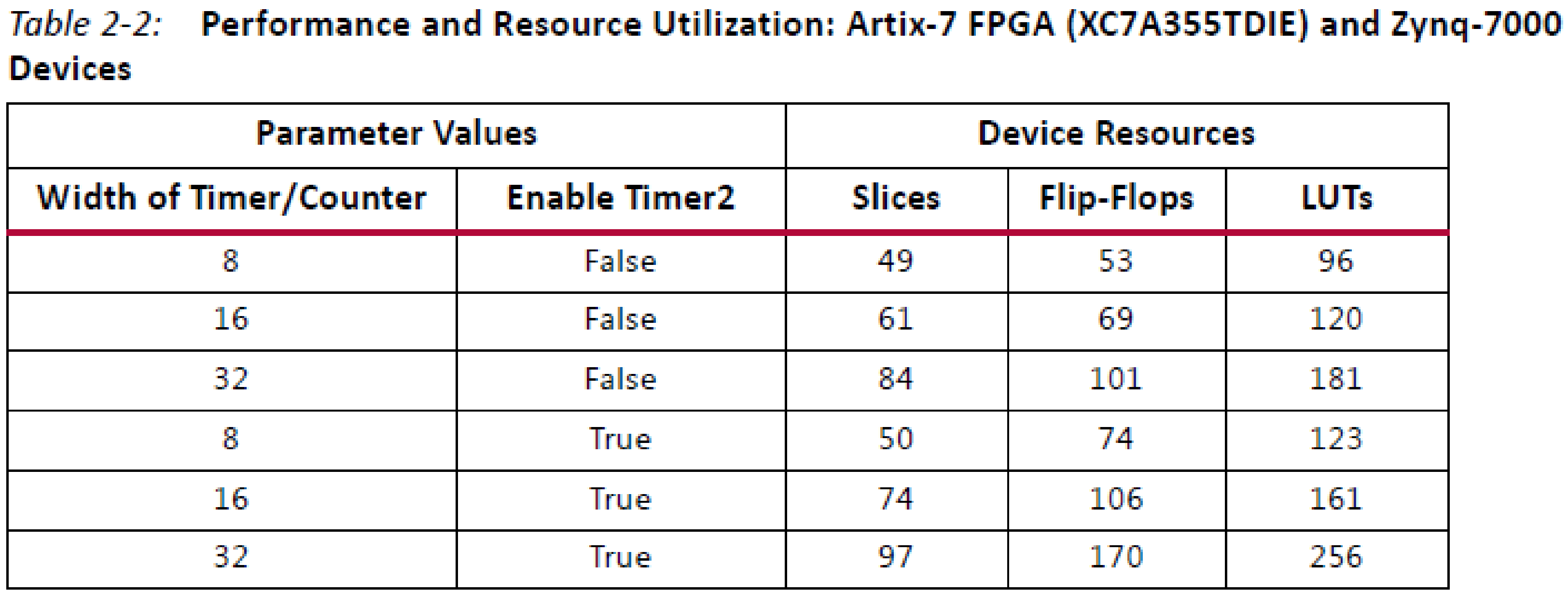


Table 2-2: **Performance and Resource Utilization: Artix-7 FPGA (XC7A355TDIE) and Zynq-7000 Devices**

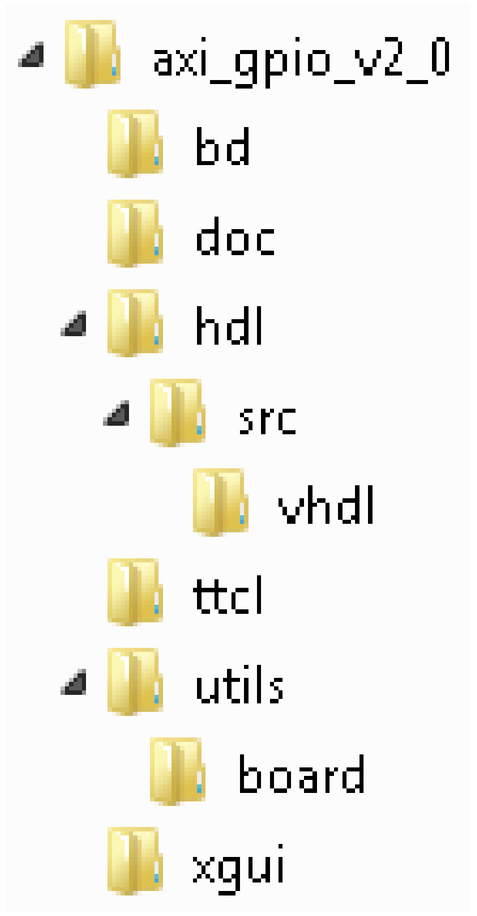
Parameter Values		Device Resources		
Width of Timer/Counter	**Enable Timer2**	**Slices**	**Flip-Flops**	**LUTs**
8	False	49	53	96
16	False	61	69	120
32	False	84	101	181
8	True	50	74	123
16	True	74	106	161
32	True	97	170	256

Outline

Peripheral Storage

User peripherals can be located in the project directory or a peripheral repository

- **IP Core directory (located in the project directory)**
  - – {component}.xml
  - – MyProcessorIPLib directory (user defined)
    - Repository Directory listed using **Project** → **Project Options** → **Device and Repository Search** tab
  - – *%XILINX_INSTALL%\Vivado\2013.X\data\ip*



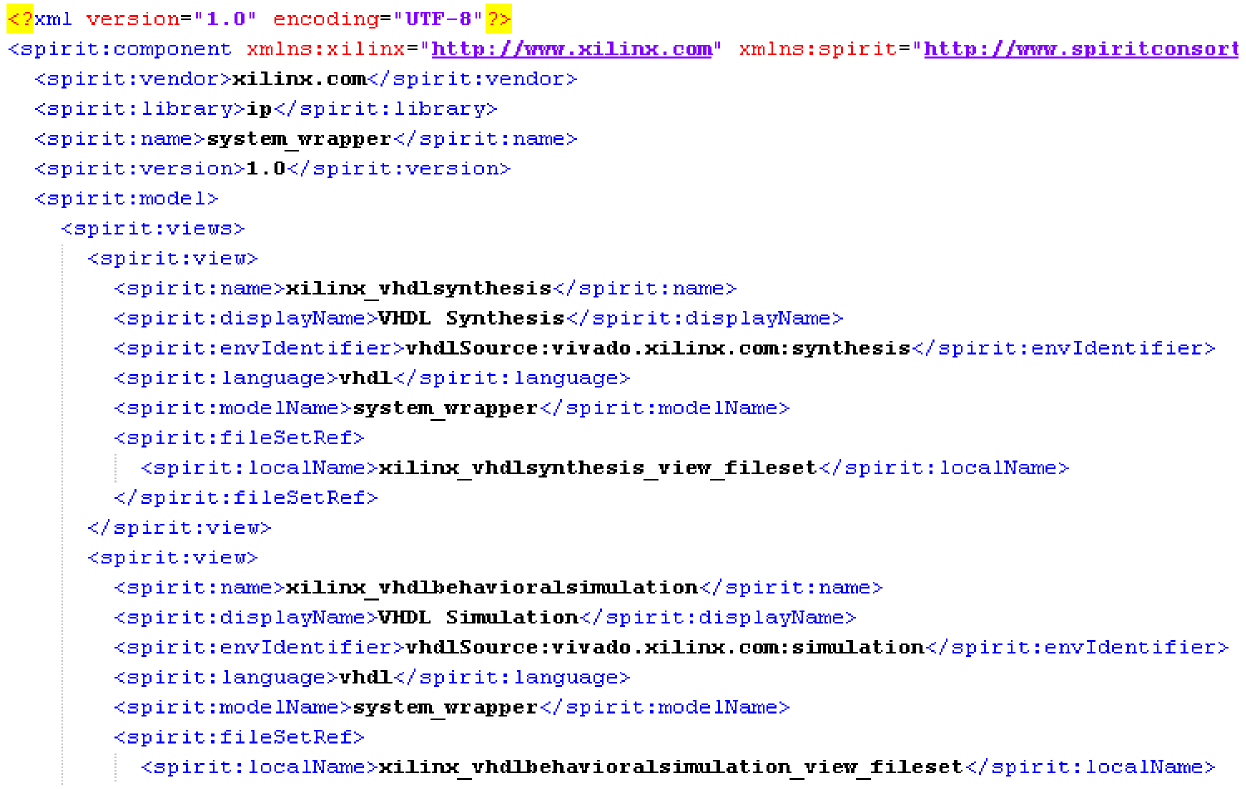
XILINX ALL PROGRAMMABLE.

Outline

- IP Catalog
- IP directory
- *IP device files*
- GP Interfaces
- Adding IP to extend PS into PL
- Bitstream generation
- Summary

IP Core files

- **component.xml**
  - XML format
  - Top level folder
  - Provides ports description, parameters and options for IP
  - Links to source files
- **xgui folder**
  - .tcl file for IPI GUI



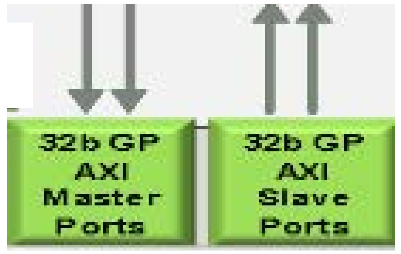
```
<?xml version="1.0" encoding="UTF-8"?>
<spirit:component xmlns:xilinx="http://www.xilinx.com" xmlns:spirit="http://www.spiritconsort
  <spirit:vendor>xilinx.com</spirit:vendor>
  <spirit:library>ip</spirit:library>
  <spirit:name>system_wrapper</spirit:name>
  <spirit:version>1.0</spirit:version>
  <spirit:model>
    <spirit:views>
      <spirit:view>
        <spirit:name>xilinx_vhdlsynthesis</spirit:name>
        <spirit:displayName>VHDL Synthesis</spirit:displayName>
        <spirit:envIdentifier>vhdlSource:vivado.xilinx.com:synthesis</spirit:envIdentifier>
        <spirit:language>vhdl</spirit:language>
        <spirit:modelName>system_wrapper</spirit:modelName>
        <spirit:fileSetRef>
          <spirit:localName>xilinx_vhdlsynthesis_view_fileset</spirit:localName>
        </spirit:fileSetRef>
      </spirit:view>
      <spirit:view>
        <spirit:name>xilinx_vhdlbehavioralsimulation</spirit:name>
        <spirit:displayName>VHDL Simulation</spirit:displayName>
        <spirit:envIdentifier>vhdlSource:vivado.xilinx.com:simulation</spirit:envIdentifier>
        <spirit:language>vhdl</spirit:language>
        <spirit:modelName>system_wrapper</spirit:modelName>
        <spirit:fileSetRef>
          <spirit:localName>xilinx_vhdlbehavioralsimulation_view_fileset</spirit:localName>
```

Outline

- IP Catalog
- IP directory
- IP device files
- *GP Interfaces*
- Adding IP to extend PS into PL
- Bitstream generation
- Summary

GP Ports

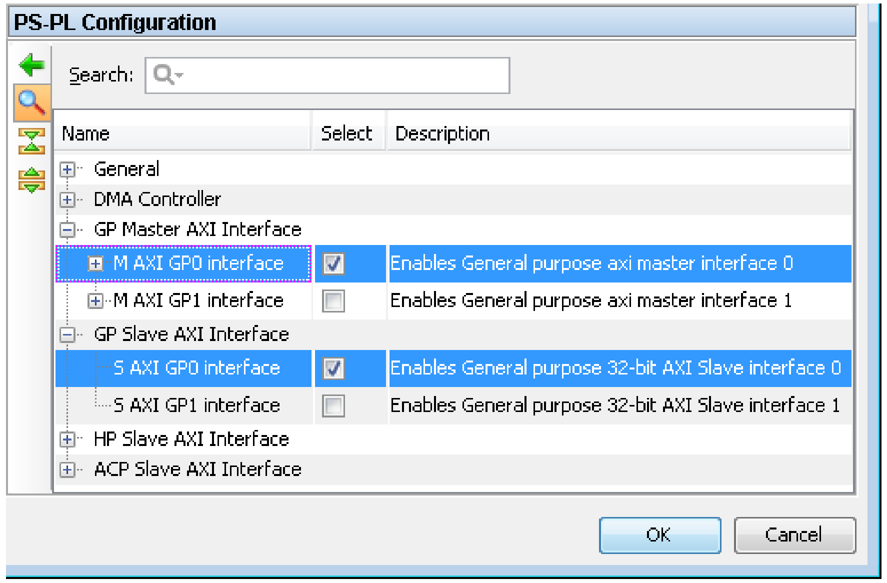
- **By default, GP Slave and Master ports are disabled**



- **Enable GP Master and/or Slave ports depending on whether a slave or a master peripheral is going to be added in PL**
- **axi_interconnect block is required to connect IP to a port with different protocols**
  - Automatically convert Protocols
  - Can be automatically added when using Block Automation in IPI

Configuring GP Ports

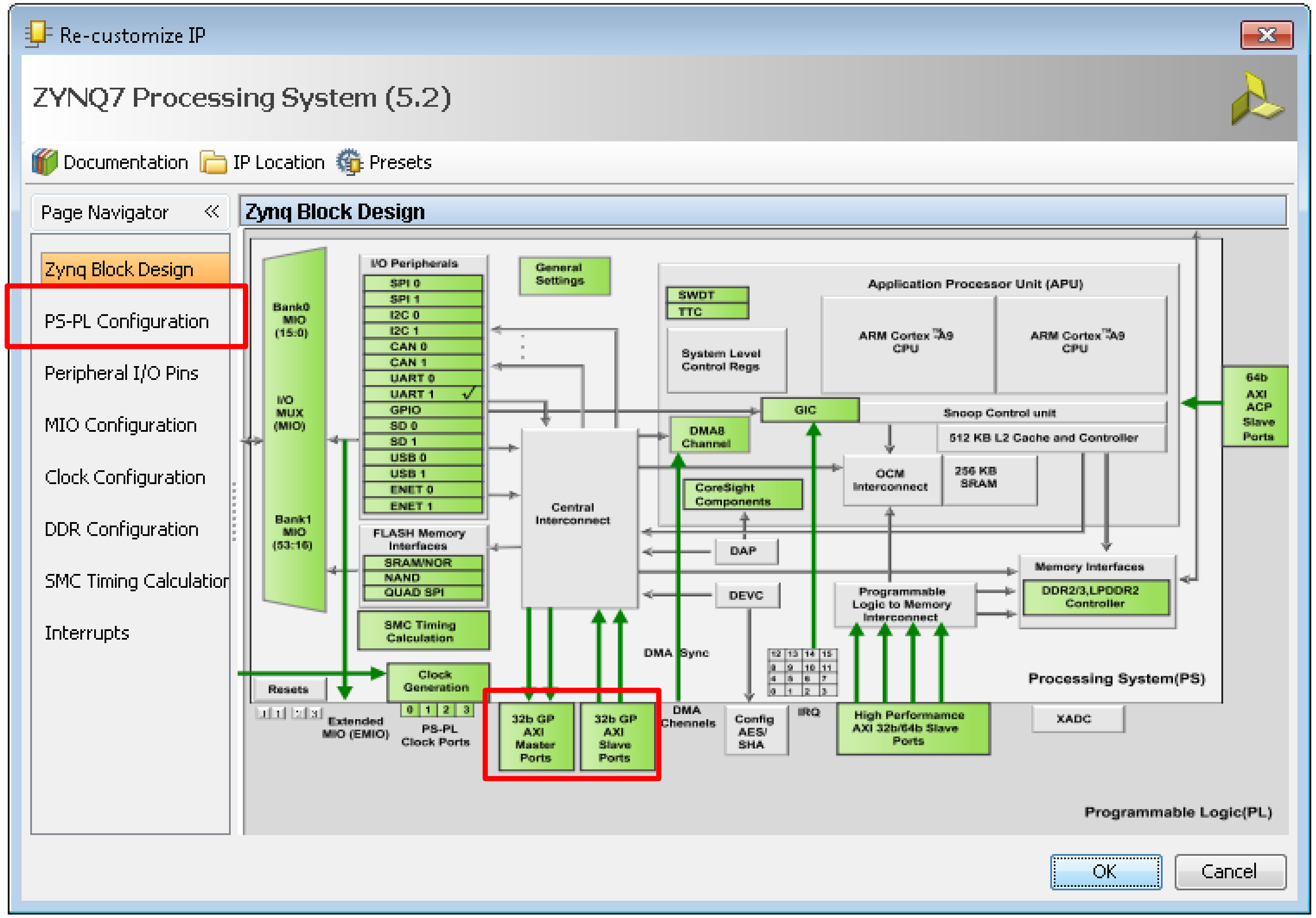
- **Click on the menu or green GP Blocks to configure**



PS-PL Configuration

Search:

Name	Select	Description
⊞ General		
⊞ DMA Controller		
⊟ GP Master AXI Interface		
⊞ M AXI GP0 interface	☑	Enables General purpose axi master interface 0
⊞ M AXI GP1 interface	☐	Enables General purpose axi master interface 1
⊟ GP Slave AXI Interface		
S AXI GP0 interface	☑	Enables General purpose 32-bit AXI Slave interface 0
S AXI GP1 interface	☐	Enables General purpose 32-bit AXI Slave interface 1
⊞ HP Slave AXI Interface		
⊞ ACP Slave AXI Interface		

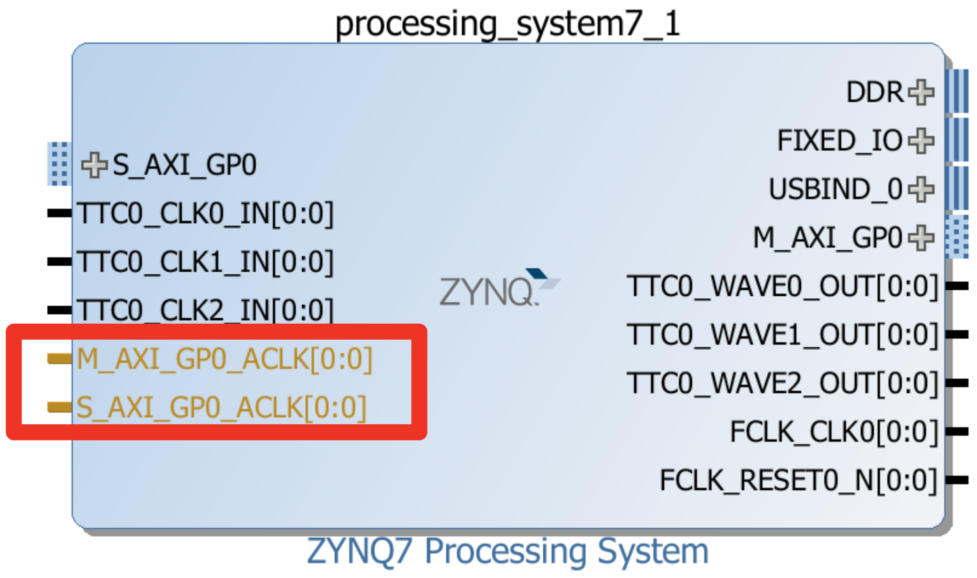


XILINX ALL PROGRAMMABLE.

Outline

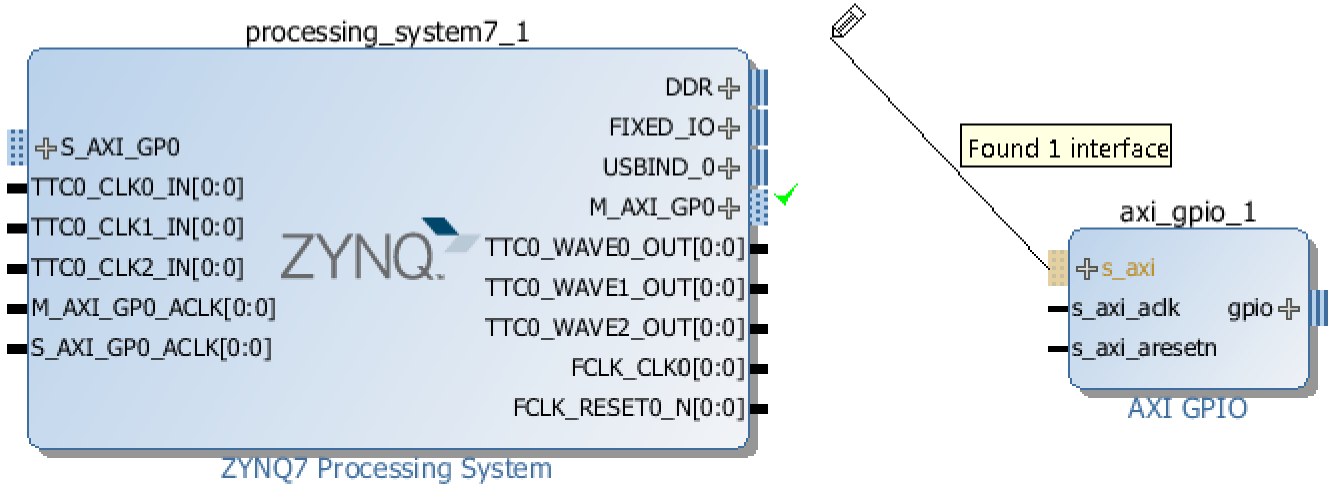
Add IP in the PL

- **Configure GP ports from PS Customization GUI**
- **PS-PL Configuration**
  - E.g. Enable M_AXI_GP0/1 or S_AXI_GP0/1
- **Ports will then be available on Zynq Block Diagram**
- **Connect the added IP to the appropriate port**
- **Assign address to the added IP, if unmapped**
- **Configure the IP, if necessary**
- **Make external connections, if needed**
  - – Add external ports/interfaces if the added IP is interacting with external devices



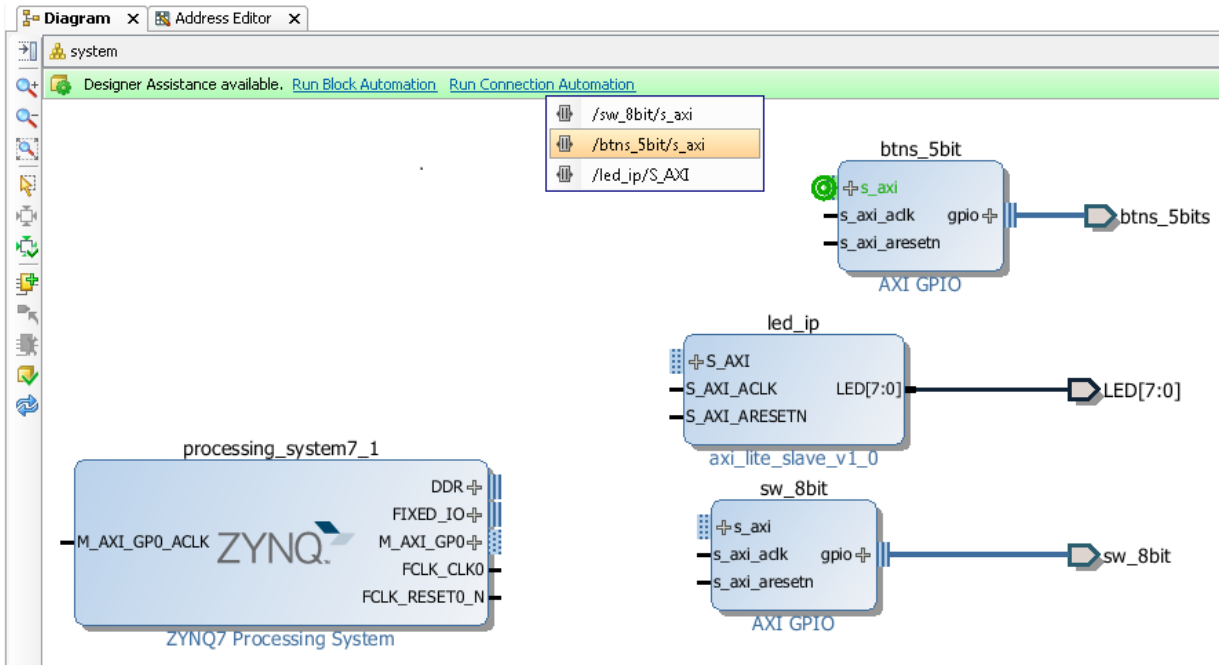
Connecting IP

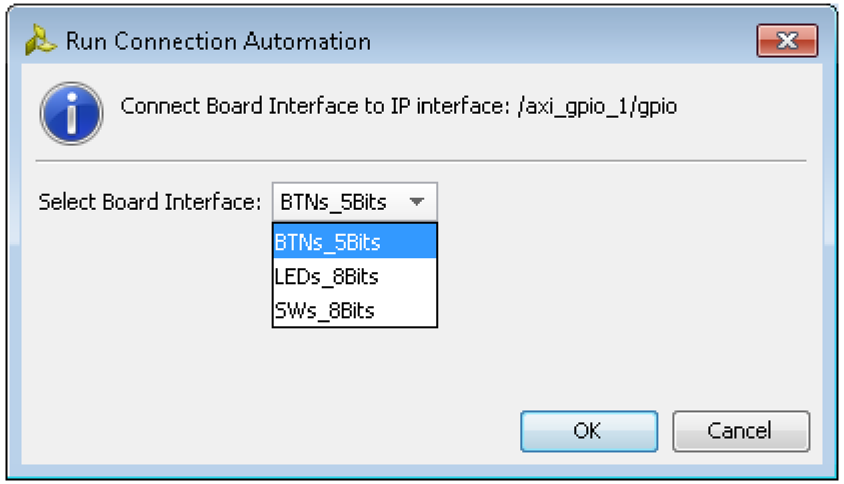
- **Add IP from the IP Catalog**
- **Click and drag to find connections**
- **Valid connections Highlighted**
- **Designer Assistance, Connection automation**
  - If Board Support available, IP can be connected to external pins
- **Or manually create and connect (external) ports**



Designer assistance; Block Automation, Connection Automation

- Block, Connection
- Can automatically connect IP blocks
- Automatically insert required blocks
- E.g. Add BRAM; Automation will insert and connect BRAM controller and Reset logic
- If Board Support is available, IP can automatically be connected to top level ports




Run Connection Automation

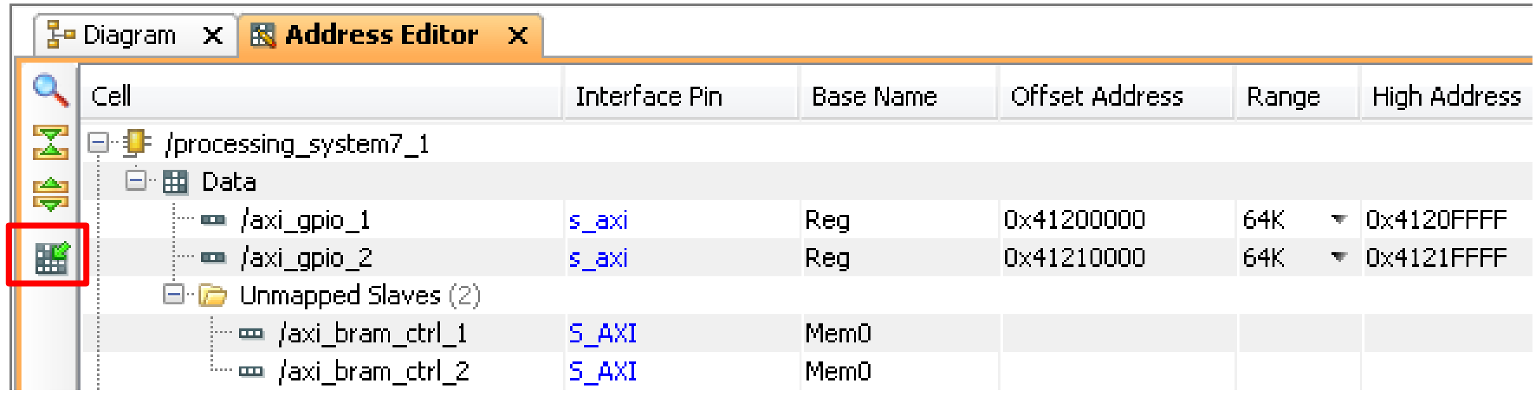
Connect Board Interface to IP interface: /axi_gpio_1/gpio

Select Board Interface: BTNs_5Bits

XILINX ALL PROGRAMMABLE.

Assign Addresses

- Peripherals in the Zynq™ AP SoC PS have fixed addresses and do not appear in the address map when an IP is added to the system
- For PS peripherals Click on the Auto Assign Addresses button

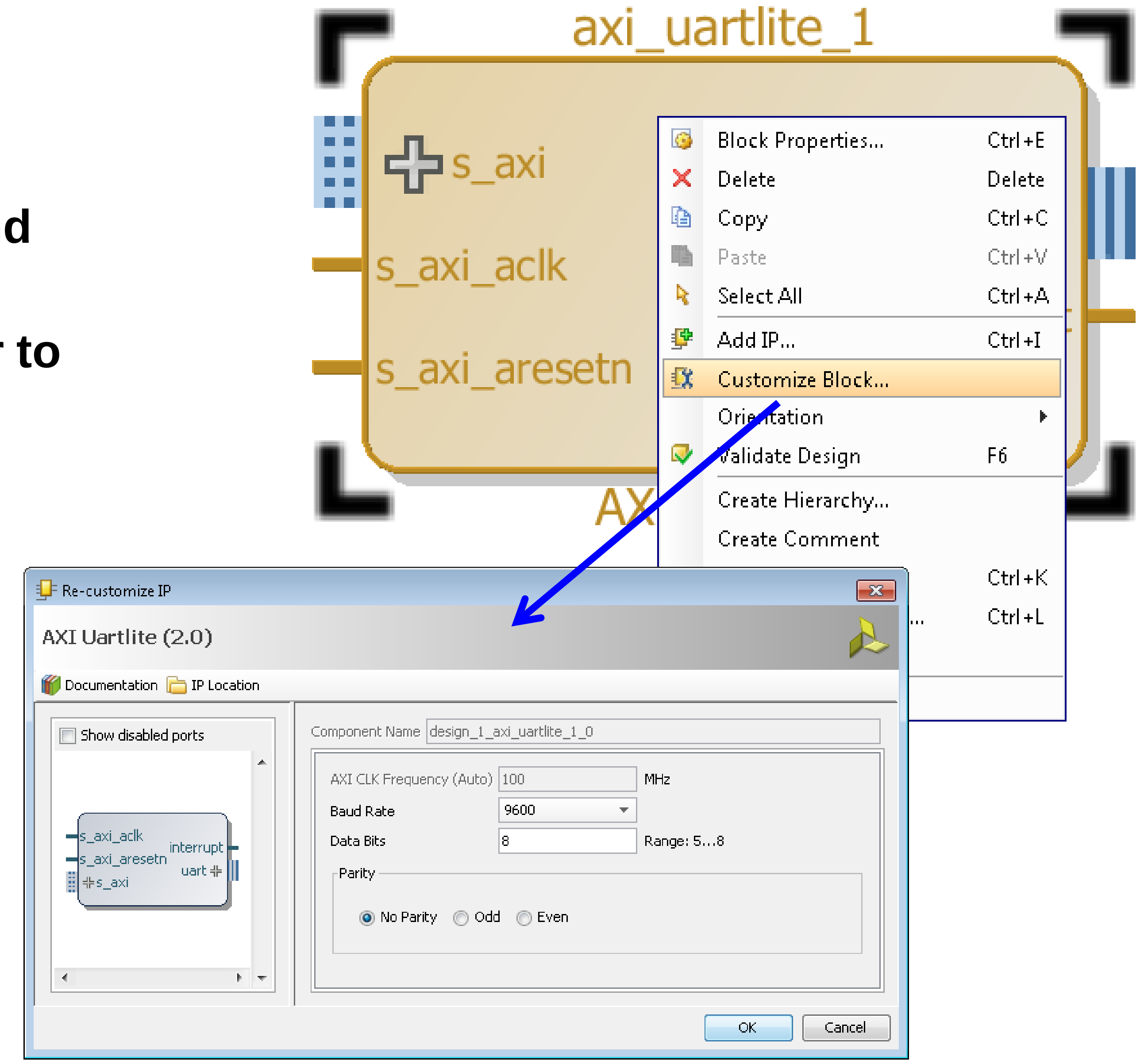


Cell	Interface Pin	Base Name	Offset Address	Range	High Address
/processing_system7_1					
Data					
/axi_gpio_1	s_axi	Reg	0x41200000	64K	0x4120FFFF
/axi_gpio_2	s_axi	Reg	0x41210000	64K	0x4121FFFF
Unmapped Slaves (2)					
/axi_bram_ctrl_1	S_AXI	Mem0			
/axi_bram_ctrl_2	S_AXI	Mem0			

- The address will be generated and show the generated addresses of the added IP
- The fixed addresses of the configured peripherals of the PS

Parameterize IP Instances

- **Double-click or right click the instance and select** Customize Block **to open the configurable parameters dialog box (refer to the datasheet if needed)**
- **Default values are shown**
  - Customize the parameters that you want



Component Name design_1_axi_uartlite_1_0

AXI CLK Frequency (Auto) 100 MHz

Baud Rate 9600

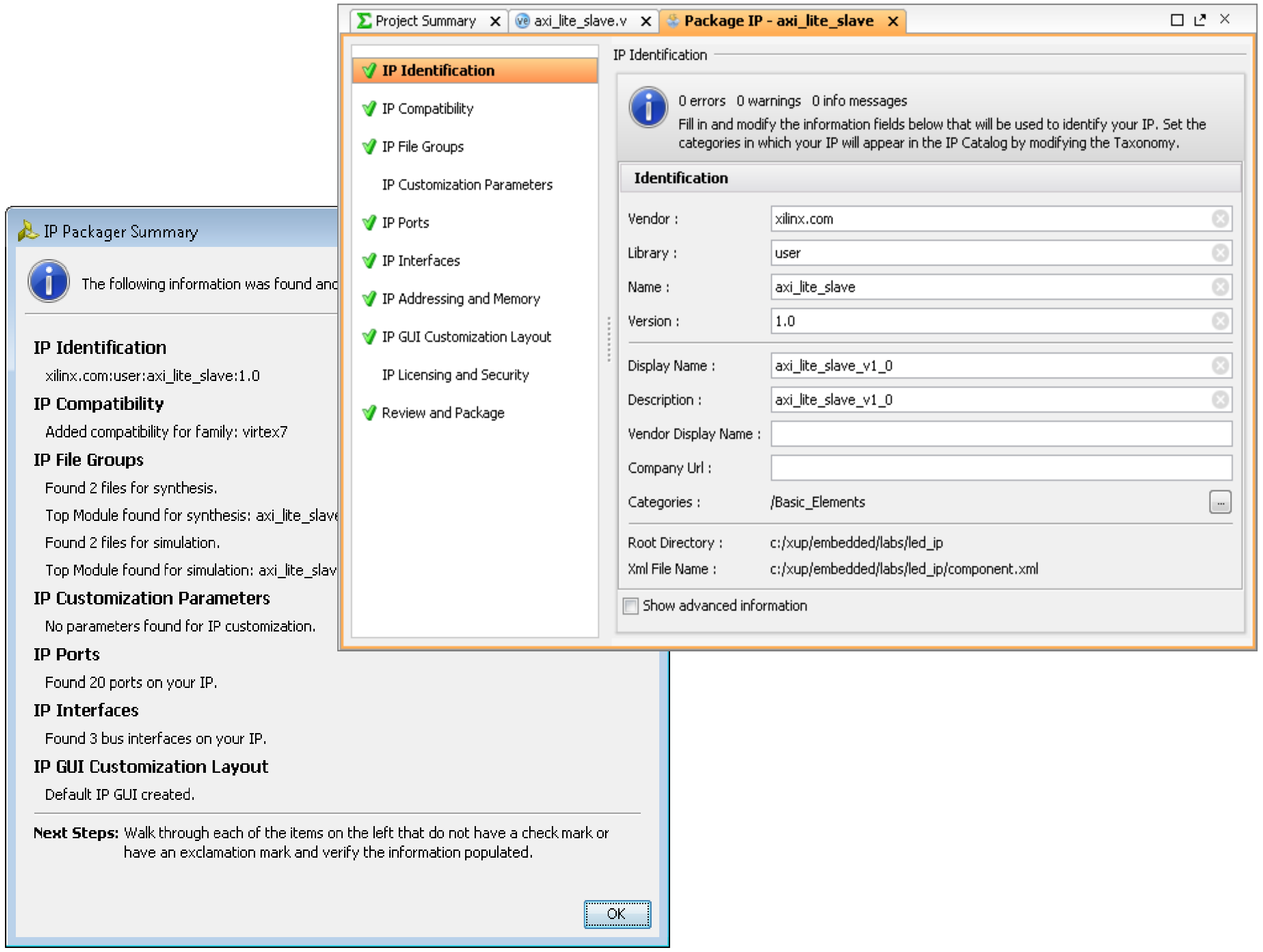
Data Bits 8 Range: 5...8

Parity

[● No Parity] Odd Even

Extending the System

- **Import custom IP**
- **IP Packager**
  - Packages into IP Integrator Format
- **Specify repository (local/global)**
- **IP can then be used in IP Integrator**
- **More on IP Packager later**



IP Identification

0 errors 0 warnings 0 info messages

Fill in and modify the information fields below that will be used to identify your IP. Set the categories in which your IP will appear in the IP Catalog by modifying the Taxonomy.

Identification

Vendor :	xilinx.com
Library :	user
Name :	axi_lite_slave
Version :	1.0
Display Name :	axi_lite_slave_v1_0
Description :	axi_lite_slave_v1_0
Vendor Display Name :	
Company Url :	
Categories :	/Basic_Elements
Root Directory :	c:/xup/embedded/labs/led_ip
Xml File Name :	c:/xup/embedded/labs/led_ip/component.xml

☐ Show advanced information

IP Packager Summary

The following information was found and

IP Identification

xilinx.com:user:axi_lite_slave:1.0

IP Compatibility

Added compatibility for family: virtex7

IP File Groups

Found 2 files for synthesis.

Top Module found for synthesis: axi_lite_slave

Found 2 files for simulation.

Top Module found for simulation: axi_lite_slav

IP Customization Parameters

No parameters found for IP customization.

IP Ports

Found 20 ports on your IP.

IP Interfaces

Found 3 bus interfaces on your IP.

IP GUI Customization Layout

Default IP GUI created.

Next Steps: Walk through each of the items on the left that do not have a check mark or have an exclamation mark and verify the information populated.

Outline

- IP Catalog
- IP directory
- IP device files
- GP Interfaces
- Adding IP to extend PS into PL
- ***Bitstream generation***
- Summary

Bitstream Generation

- **After defining the system hardware, the next step is to create hardware netlists if the system hardware has logic in PL**
- **A HDL wrapper for the block diagram must be generated**
  - Additional logic can be added to the HDL, or the Processor system can be used as a sub block in a HDL design
- **The design and block diagram must be open before synthesise and implementation can be carried out**
- **If the system contains hardware in the PL, the bitstream must be generated**
- **The PL (FPGA) must be programmed before application can be downloaded and executed**

Outline

Summary

- **PS functionality can be extended by instantiating peripherals in PL**
- **Adding IP in PL involves**
  - Enabling interface(s) in PS
  - Selecting IP from the IP Catalog and configuring IP for desired functionality
  - Connecting the (PL) IP to the PS using IP Integrator
  - Assigning address
  - Connecting IP ports to ports of other peripherals and/or to external pins
- **HDL Wrapper is needed for IP Integrator Block**
- **Bitstream must be generated when PL has any IP**
- **The FPGA must be programmed with the generated hardware bitstream before an application can be run**