Virtual Input/Output v3.0

LogiCORE IP Product Guide

Vivado Design Suite

PG159 April 4, 2018

XILINX
ALL PROGRAMMABLE™

Table of Contents

IP Facts

Chapter 1: Overview

Chapter 2: Product Specification

Chapter 3: Designing with the Core

Chapter 4: Design Flow Steps

Chapter 5: Example Design

Chapter 6: Test Bench

Appendix A: Migrating and Upgrading

Appendix B: Debugging

Appendix C: Additional Resources and Legal Notices

Introduction

The LogiCORE™ IP Virtual Input/Output (VIO) core is a customizable core that can both monitor and drive internal FPGA signals in real time. The number and width of the input and output ports are customizable in size to interface with the FPGA design. Because the VIO core is synchronous to the design being monitored and/or driven, all design clock constraints that are applied to your design are also applied to the components inside the VIO core. Run time interaction with this core requires the use of the Vivado® logic analyzer feature.

Features

- Provides virtual LEDs and other status indicators through input ports.
- Includes optional activity detectors on input ports to detect rising and falling transitions between samples.
- Provides virtual buttons and other controls through output ports.
- Includes custom output initialization that allows you to specify the value of the VIO core outputs immediately following device configuration and start-up.
- Run time reset of the VIO core to its initial values.

LogiCORE™ IP Facts Table	
Core Specifics	
Supported Device Family[1]	UltraScale+™ Families, UltraScale™ Architecture, Zynq®-7000 All Programmable SoC, 7 Series
Supported User Interfaces	N/A
Resources	Performance and Resource Utilization web page
Provided with Core	
Design Files	Encrypted RTL
Example Design	Verilog or VHDL
Test Bench	Not Provided
Constraints File	XDC
Simulation Model	Not Provided
Supported S/W Driver	N/A
Tested Design Flows[2]	
Design Entry	Vivado® Design Suite
Simulation	Not Provided
Synthesis[3]	Vivado Synthesis
Support	
Provided by Xilinx at the Xilinx Support web page	

Notes:

1. For a complete listing of supported devices, see the Vivado IP catalog.
2. For the supported versions of the tools, see the Xilinx Design Tools: Release Notes Guide.
3. The standard synthesis flow for Synplify is not supported for the core.

Overview

The VIO core is a customizable core that can both monitor and drive internal FPGA signals in real time. Unlike the ILA core, no on-chip or off-chip RAM is required.

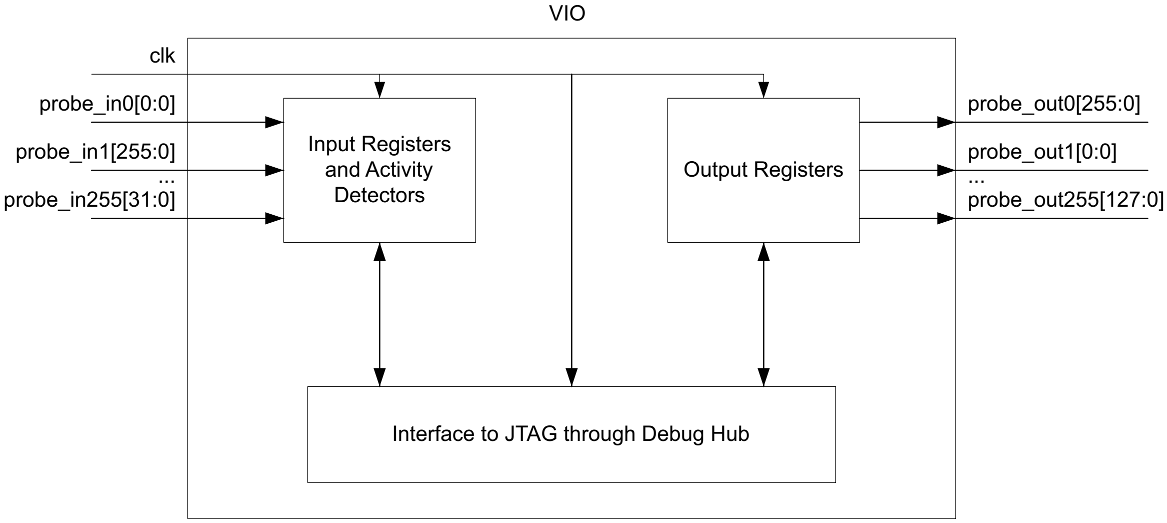


Figure 1-1: **VIO Block Diagram**

Feature Summary

Two types of signals are available in the VIO core:

- Input probes
- Output probes

Input Probes

These inputs to the VIO core registered using the design clock that is attached to the `CLK` input port. The input values are read back periodically and displayed in the Vivado® logic analyzer feature.

Output Probes

These outputs from the VIO core are driven to the surrounding user design. The output values are driven to combinations of 1s and 0s by the Vivado logic analyzer feature. Output probes can also be initialized to any desired value during generation time.

Activity Detectors

Every VIO core input has additional cells to capture the presence of transitions on the input. Because the design clock is most likely much faster than the sample period of the Analyzer, it is possible for the signal being monitored to transition many times between successive samples. The activity detectors capture this behavior and the results are displayed along with the value in the Vivado logic analyzer.

Licensing and Ordering

This Xilinx LogiCORE™ IP module is provided at no additional cost with the Xilinx® Vivado® Design Suite under the terms of the Xilinx End User License.

Information about other Xilinx LogiCORE IP modules is available at the Xilinx Intellectual Property page. For information on pricing and availability of other Xilinx LogiCORE IP modules and tools, contact your local Xilinx sales representative.

License Checkers

If the IP requires a license key, the key must be verified. The Vivado design tools have several license check points for gating licensed IP through the flow. If the license check succeeds, the IP can continue generation. Otherwise, generation halts with error. License checkpoints are enforced by the following tools:

- Vivado synthesis
- Vivado implementation
- write_bitstream (Tcl command)

IMPORTANT: *IP license level is ignored at checkpoints. The test confirms a valid license exists. It does not check IP license level.*

Product Specification

The VIO core is used to drive data into your design and read data from your design through the JTAG port. The core is designed to replace or augment board-level I/O components such as status indicators (for example, LEDs) and low-bandwidth controls (for example, buttons or DIP switches).

Performance

Maximum Frequency

The VIO core is designed to run at design clock frequencies up to 250 MHz, but maximum clock frequency could be limited by other factors in the design such as overall utilization or routing congestion.

Resource Utilization

For full details about performance and resource utilization, visit the Performance and Resource Utilization web page.

Port Descriptions

Table 2-1 shows the VIO core I/O port signals.

Table 2-1: **VIO I/O Signals**

Signal Name	I/O	Description
clk	I	Design clock used to register input ports and output ports. This is required.
probe_in<m>[<n> – 1:0]	I	Input probe port number <m> (where <m> can be 0 to 256) of width <n> (where <n> can be 1 to 256). For a 1-bit wide port, use probe_in<m>[0:0].
probe_out<m>[<n> – 1:0]	O	Output probe port number <m> (where <m> can be 0 to 256) of width <n> (where <n> can be 1 to 256). For a 1-bit wide port, use probe_out<m>[0:0].

Chapter 3

Designing with the Core

This chapter includes guidelines and additional information to facilitate designing with the core.

General Design Guidelines

The VIO core can be used to replace or augment any board-level status indicators (such as LEDs) and controls (such as buttons or DIP switches).

Clocking

The `clk` input port is the clock used by the VIO core to register input and output probe values. For best results, it should be the same clock signal that is synchronous to the design logic that is attached to the input and output probe ports of the VIO core.

Resets

The VIO core output probes can only be reset to their initial values by using the Vivado® logic analyzer feature. Similarly, the VIO core input probe activity detectors can only be reset using the Vivado logic analyzer.

Chapter 4

Design Flow Steps

This chapter describes customizing and generating the core, constraining the core, and the simulation, synthesis and implementation steps that are specific to this IP core. More detailed information about the standard Vivado® design flows and the IP integrator can be found in the following Vivado Design Suite user guides:

- *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994) [Ref 1]
- *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 2]
- *Vivado Design Suite User Guide: Getting Started* (UG910) [Ref 3]
- *Vivado Design Suite User Guide: Logic Simulation* (UG900) [Ref 4]

Customizing and Generating the Core

This section includes information about using Xilinx tools to customize and generate the core in the Vivado Design Suite.

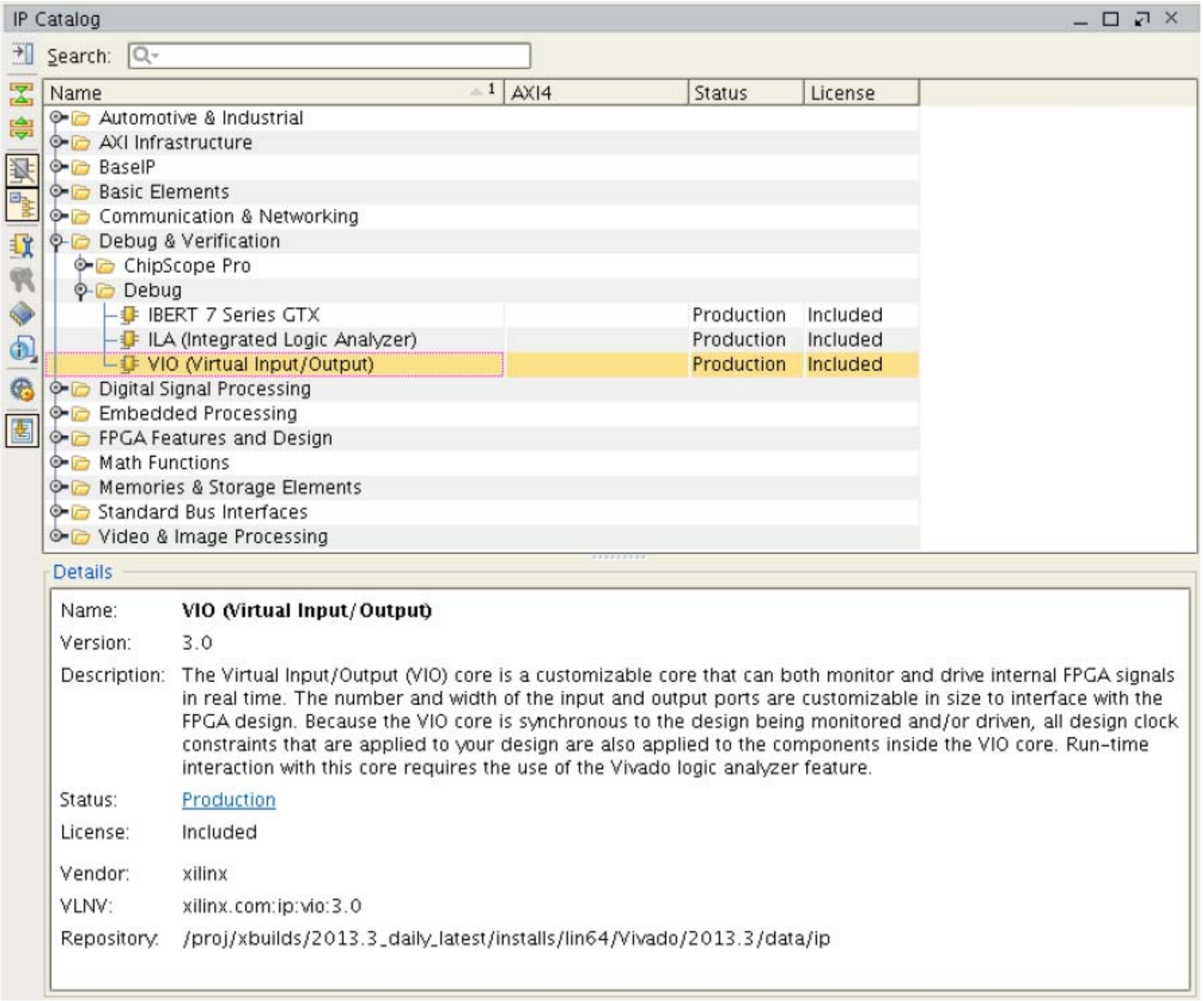
If you are customizing and generating the core in the Vivado IP integrator, see the *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994) [Ref 1] for detailed information. IP integrator might auto-compute certain configuration values when validating or generating the design. To check whether the values do change, see the description of the parameter in this chapter. To view the parameter value, run the `validate_bd_design` command in the Tcl console.

You can customize the IP for use in your design by specifying values for the various parameters associated with the IP core using the following steps:

1. Select the IP from the Vivado IP catalog.
2. Double-click the selected IP or select the **Customize IP** command from the toolbar or right-click menu.

For details, see the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 2] and the *Vivado Design Suite User Guide: Getting Started* (UG910) [Ref 3].

Note: Figure in this chapter is an illustration of the Vivado Integrated Design Environment. This layout might vary from the current version.



IP Catalog

Search:

Name	AXI4	Status	License
Automotive & Industrial			
AXI Infrastructure			
BaseIP			
Basic Elements			
Communication & Networking			
Debug & Verification			
ChipScope Pro			
Debug			
IBERT 7 Series GTX		Production	Included
ILA (Integrated Logic Analyzer)		Production	Included
VIO (Virtual Input/Output)		Production	Included
Digital Signal Processing			
Embedded Processing			
FPGA Features and Design			
Math Functions			
Memories & Storage Elements			
Standard Bus Interfaces			
Video & Image Processing			

Details

Name: **VIO (Virtual Input/Output)**

Version: 3.0

Description: The Virtual Input/Output (VIO) core is a customizable core that can both monitor and drive internal FPGA signals in real time. The number and width of the input and output ports are customizable in size to interface with the FPGA design. Because the VIO core is synchronous to the design being monitored and/or driven, all design clock constraints that are applied to your design are also applied to the components inside the VIO core. Run-time interaction with this core requires the use of the Vivado logic analyzer feature.

Status: Production

License: Included

Vendor: xilinx

VLNV: xilinx.com:ip:vio:3.0

Repository: /proj/xbuilds/2013.3_daily_latest/installs/lin64/Vivado/2013.3/data/ip

Figure 4-1: **VIO Core in Vivado IP Catalog**

General Options Panel

- **Component Name** – Use this text field to provide a unique module name for the VIO core.
- **Input Probe Count** – Use this text field to select the number of input probe ports on the VIO core. The valid range used in the Vivado IDE is 0 to 64. If you need more than 64 input probe ports, you need to use the following Tcl command to configure the VIO core before generating it. For example, `set_property -dict [list CONFIG.C_NUM_PROBE_IN {250}] [get_ips vio_0]`.

  Note: At least one input or output probe port needs to be specified.

- **Output Probe Count** – Use this text field to select the number of output probe ports on the VIO core. The valid range used in the Vivado IDE is 0 to 64. If you need more than 64 output probe ports, you need to use the following Tcl command to configure the VIO core before generating it. For example, `set_property -dict [list CONFIG.C_NUM_PROBE_OUT {250}] [get_ips vio_0]`.

  Note: At least one input or output probe port needs to be specified.
- **probe_in Ports Panels** – Each probe_in ports panel has up to 16 ports.
- **Probe Width** – Use the Probe Width field to set the width of each probe port. The valid width range is 1 to 256 bits wide.
- **probe_out Ports Panels** – Each probe_out ports panel has up to 16 ports.
- **Probe Width** – Use the Probe Width field to set the width of each probe port. The valid width range is 1 to 256 bits wide.
- **Initial Value** – Use the Initial Value field to set the initial value of the output probe port to a specific value. The value is to be specified in hexadecimal format with a “0x” prefix.

Output Generation

For more information, see the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 2].

Constraining the Core

This section contains information about constraining the core in the Vivado Design Suite.

Required Constraints

The VIO core includes an XDC file that contains appropriate false path constraints to prevent the over-constraining of clock domain crossing synchronization paths. It is also expected that the clock signal connected to the `clk` input port of the VIO core is properly constrained in your design constraints.

Device, Package, and Speed Grade Selections

This section is not applicable for this IP core.

Clock Frequencies

This section is not applicable for this IP core.

Clock Management

This section is not applicable for this IP core.

Clock Placement

This section is not applicable for this IP core.

Banking

This section is not applicable for this IP core.

Transceiver Placement

This section is not applicable for this IP core.

I/O Standard and Placement

This section is not applicable for this IP core.

Simulation

For comprehensive information about Vivado simulation components, as well as information about using supported third-party tools, see the *Vivado Design Suite User Guide: Logic Simulation* (UG900) [Ref 4].

IMPORTANT: *For cores targeting 7 series or Zynq-7000 devices, UNIFAST libraries are not supported. Xilinx IP is tested and qualified with UNISIM libraries only.*

Synthesis and Implementation

For details about synthesis and implementation, see the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 2].

IMPORTANT: *The standard synthesis flow for Synplify is not supported for the core.*

Chapter 5

Example Design

This chapter contains information about the provided example design in the Vivado® Design Suite environment.

Directory and File Contents

- **<component name>_example/<component name>_example.srcs/**
  Top-level project directory; name is user-defined
  - constrs_1/imports/<component name>/
    - example_<component name>.xdc
  - sources_1/imports/<component name>/
    - example_<component name>.v/.vhd
  - sources_1/ip/<component name>

<component name>_example/<component name>_example.srcs/

This directory contains the source files needed to synthesize the VIO core whose name is `<component name>`.

Table 5-1: **VIO Example Design Source Files**

Name	Description
constrs_1/imports/<component name>/	
example_<component name>.xdc	Constraints file for the example design
sources_1/imports/<component name>/	
example_<component name>.v/.vhd	Verilog (.v) or VHDL (.vhd) source file for the example design

Implementation

To implement the example design, select **Run Implementation** in the **Vivado Project Manager** window. For further details on setting up the implementation, see the *Vivado Design Suite User Guide: Implementation* (UG904) [Ref 5].

Chapter 6

Test Bench

There is no test bench for this IP core release.

Appendix A

Migrating and Upgrading

This appendix contains information about migrating a design from ISE® to the Vivado® Design Suite, and for upgrading to a more recent version of the IP core. For customers upgrading in the Vivado Design Suite, important details (where applicable) about any port changes and other impact to user logic are included.

Migrating to the Vivado Design Suite

For information on migrating to the Vivado® Design Suite, see *ISE to Vivado Design Suite Migration Guide* (UG911) [Ref 6].

Upgrading in the Vivado Design Suite

This section provides information about any changes to the user logic or port designations that take place when you upgrade to a more current version of this IP core in the Vivado Design Suite.

Port Changes

- The `clk` port is now required. The `async_in` and `async_out` ports are no longer available on the VIO v3.0 core.
- The `sync_in` and `sync_out` ports are now similar to the `probe_in<m>` and `probe_out<m>` ports, except that VIO v3.0 now allows up to 256 of each kind of port.
- The port names are changed from uppercase to lowercase and you have to take care of updating this in your design.

Functionality Changes

The VIO v3.0 core no longer has asynchronous input or output ports. All input and output probe ports are assumed to be synchronous to the `clk` input port.

IMPORTANT: *The VIO v3.0 core is not compatible with the legacy ChipScope™ Pro Analyzer tool. The VIO v3.0 core requires the Vivado logic analyzer feature for run time interaction.*

Appendix B

Debugging

This appendix includes details about resources available on the Xilinx Support website and debugging tools.

TIP: *If the IP generation halts with an error, there might be a license issue. See License Checkers in Chapter 1 for more details.*

Finding Help on Xilinx.com

To help in the design and debug process when using the VIO, the Xilinx Support web page contains key resources such as product documentation, release notes, answer records, information about known issues, and links for obtaining further product support.

Documentation

This product guide is the main document associated with the VIO. This guide, along with documentation related to all products that aid in the design process, can be found on the Xilinx Support web page or by using the Xilinx Documentation Navigator.

Download the Xilinx Documentation Navigator from the Downloads page. For more information about this tool and the features available, open the online help after installation.

Answer Records

Answer Records include information about commonly encountered problems, helpful information on how to resolve these problems, and any known issues with a Xilinx product. Answer Records are created and maintained daily ensuring that users have access to the most accurate information available.

Answer Records for this core can also be located by using the Search Support box on the main Xilinx support web page. To maximize your search results, use proper keywords such as:

- Product name
- Tool message(s)
- Summary of the issue encountered

A filter search is available after results are returned to further target the results.

Master Answer Record for the VIO

AR: 54606

Technical Support

Xilinx provides technical support at the Xilinx Support web page for this LogiCORE™ IP product when used as described in the product documentation. Xilinx cannot guarantee timing, functionality, or support if you do any of the following:

- Implement the solution in devices that are not defined in the documentation.
- Customize the solution beyond that allowed in the product documentation.
- Change any section of the design labeled DO NOT MODIFY.

To contact Xilinx Technical Support, navigate to the Xilinx Support web page.

Debug Tools

There are many tools available to address VIO design issues. It is important to know which tools are useful for debugging various situations.

Vivado Design Suite Debug Feature

The Vivado® Design Suite debug feature inserts logic analyzer and virtual I/O cores directly into your design. The debug feature also allows you to set trigger conditions to capture application and integrated block port signals in hardware. Captured signals can then be analyzed. This feature in the Vivado IDE is used for logic debugging and validation of a design running in Xilinx devices.

The Vivado logic analyzer is used with the logic debug IP cores, including:

- ILA 2.0 (and later versions)
- VIO 2.0 (and later versions)

See *Vivado Design Suite User Guide: Programming and Debugging* (UG908) [Ref 7].

Hardware Debug

Hardware issues can range from link bring-up to problems seen after hours of testing. This section provides debug steps for common issues. The Vivado debug feature is a valuable resource to use in hardware debug. The signal names mentioned in the following individual sections can be probed using the debug feature for debugging the specific problems.

General Checks

Ensure that all the timing constraints for the core were properly incorporated from the example design and that all constraints were met during implementation.

- Does it work in post-place and route timing simulation? If problems are seen in hardware but not in timing simulation, this could indicate a PCB issue. Ensure that all clock sources are active and clean.
- If using MMCMs in the design, ensure that all MMCMs have obtained lock by monitoring the `LOCKED` port.

Additional Resources and Legal Notices

Xilinx Resources

For support resources such as Answers, Documentation, Downloads, and Forums, see Xilinx Support.

Documentation Navigator and Design Hubs

Xilinx® Documentation Navigator provides access to Xilinx documents, videos, and support resources, which you can filter and search to find information. To open the Xilinx Documentation Navigator (DocNav):

- From the Vivado® IDE, select **Help > Documentation and Tutorials**.
- On Windows, select **Start > All Programs > Xilinx Design Tools > DocNav**.
- At the Linux command prompt, enter `docnav`.

Xilinx Design Hubs provide links to documentation organized by design tasks and other topics, which you can use to learn key concepts and address frequently asked questions. To access the Design Hubs:

- In the Xilinx Documentation Navigator, click the **Design Hubs View** tab.
- On the Xilinx website, see the Design Hubs page.

Note: For more information on Documentation Navigator, see the Documentation Navigator page on the Xilinx website.

References

These documents provide supplemental material useful with this product guide:

1. *Vivado Design Suite User Guide: Designing IP Subsystems Using IP Integrator* (UG994)
2. *Vivado® Design Suite User Guide: Designing with IP* (UG896)
3. *Vivado Design Suite User Guide: Getting Started* (UG910)
4. *Vivado Design Suite User Guide: Logic Simulation* (UG900)
5. *Vivado Design Suite User Guide: Implementation* (UG904)
6. *ISE® to Vivado Design Suite Migration Guide* (UG911)
7. *Vivado Design Suite User Guide: Programming and Debugging* (UG908)
8. *Vivado Design Suite Tutorial: Programming and Debugging* (UG936)

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
04/04/2018	3.0	Added examples for Input Probe Count and Output Probe count bullets in General Options Panel section.
10/04/2017	3.0	Updated Resource Utilization section.
06/07/2017	3.0	Updated to latest product guide.
11/18/2015	3.0	Added support for UltraScale+ families.
12/18/2013	3.0	Added UltraScale support.
10/02/2013	3.0	Revision number advanced to 3.0 to align with core version number. • Added Zynq-7000 support. • Updated Vivado IDE. • Updated project directory in Output Generation. • Updated Migrating Appendix.
03/20/2013	1.0	Initial Xilinx release of the product guide and replaces DS284.

Please Read: Important Legal Notices

The information disclosed to you hereunder (the “Materials”) is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available "AS IS" and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of Xilinx’s limited warranty, please refer to Xilinx’s Terms of Sale which can be viewed at https://www.xilinx.com/legal.htm#tos; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in such critical applications, please refer to Xilinx’s Terms of Sale which can be viewed at https://www.xilinx.com/legal.htm#tos.

AUTOMOTIVE APPLICATIONS DISCLAIMER

AUTOMOTIVE PRODUCTS (IDENTIFIED AS “XA” IN THE PART NUMBER) ARE NOT WARRANTED FOR USE IN THE DEPLOYMENT OF AIRBAGS OR FOR USE IN APPLICATIONS THAT AFFECT CONTROL OF A VEHICLE (“SAFETY APPLICATION”) UNLESS THERE IS A SAFETY CONCEPT OR REDUNDANCY FEATURE CONSISTENT WITH THE ISO 26262 AUTOMOTIVE SAFETY STANDARD (“SAFETY DESIGN”). CUSTOMER SHALL, PRIOR TO USING OR DISTRIBUTING ANY SYSTEMS THAT INCORPORATE PRODUCTS, THOROUGHLY TEST SUCH SYSTEMS FOR SAFETY PURPOSES. USE OF PRODUCTS IN A SAFETY APPLICATION WITHOUT A SAFETY DESIGN IS FULLY AT THE RISK OF CUSTOMER, SUBJECT ONLY TO APPLICABLE LAWS AND REGULATIONS GOVERNING LIMITATIONS ON PRODUCT LIABILITY.