LabVIEW™ FPGA Module Release and Upgrade Notes

Version 2010

These release notes contain instructions for installing the LabVIEW FPGA Module, introduce new features, and provide upgrade information. Refer to the resources listed at the end of this document for information about developing applications with LabVIEW and the FPGA Module.

Contents

NATIONAL INSTRUMENTS™

System Requirements

The development computer is a PC or PXI system on which you install the LabVIEW development system and the LabVIEW FPGA Module.

Table 1 describes the requirements the development computer must meet to run the FPGA Module. The FPGA Module system requirements are in addition to the LabVIEW system requirements listed in the *LabVIEW Release Notes*.

Table 1. System Requirements for the FPGA Module

Supported Platforms	**Media and System Requirements**	**Important Notes**
• Windows 7[*] • Windows XP Pro (Service Pack 1, 2 or 3) • Windows Vista (32-bit) • Windows Vista (64-bit) running LabVIEW 32-bit only • Windows Server 2003 R2 (32-bit)[‡] • Windows Server 2008 R2 (64-bit)[‡]	• 1.2 GHz Pentium processor or a compatible processor of equal or higher speed • 2 GB memory[†] • At least 11 GB additional available disk space for a complete installation • LabVIEW 2010 Full or Professional Development System • FPGA target hardware and driver software, such as an NI Reconfigurable I/O target and the NI-RIO 3.5 software version.	[*] **Windows 7**—The FPGA Module uses Xilinx tools that do not officially support Windows 7. National Instruments obtained permission from Xilinx to allow FPGA Module customers to use the tools on this platform, with the disclaimer that Xilinx will not be able to fix any bugs found that are specific to this platform. National Instruments tested the Xilinx tools that the FPGA Module uses, and did not find any issues. If you encounter problems with the Xilinx tools and Windows 7, you might be required to compile using Windows XP or Vista 32-bit. In such cases, you might want to install the compilation tools on a remote computer. Refer to Table 2 for requirements to install the compilations tools on a remote computer. National Instruments will not be liable for any problems or issues related to the use of Xilinx tools with Windows 7. [†] **Memory Requirements**—Memory requirements vary by FPGA target. Also, although these memory requirements apply to typical FPGA VIs, the unique characteristics of each FPGA design can affect whether LabVIEW can compile the design using the available memory. You can monitor the memory usage of the `xst.exe` process in the **Windows Task Manager** to determine if you need to install additional memory. [‡] **Windows Server**—The FPGA Module does not support the Windows Server non-R2 editions.

If the FPGA design or target requires more than 2 GB of memory, National Instruments recommends that you install the LabVIEW FPGA compilation tools on a remote computer. Refer to `ni.com/info` and enter the Info Code `fpgakb1rcs` for more information about installing the compilation tools on a remote computer. Table 2 describes the recommended operating systems for installing the compilation tools on a remote computer.

Table 2. Recommended Systems for Installing the Compilation Tools on a Remote Computer

Supported Platform	Important Notes
• Windows XP Pro (Service Pack 1, 2, or 3) • Windows Vista (32-bit) • Windows Vista (64-bit)	**Memory Requirements**—On a remote computer with Windows Vista 64-bit, the compilation tools can take advantage of 4 GB or more of memory. The complete set of compilation tools require at least 11 GB of disk space for a complete installation.

Installing the LabVIEW FPGA Module

This section includes information about installing the FPGA Module on a development computer.

Note **(Upgrade Users)** You must install the FPGA Module 2010 as well as LabVIEW 2010 before you mass compile existing VIs. If you mass compile existing FPGA VIs before you install the FPGA Module, some FPGA-specific VIs will have mutation issues.

Complete the following steps to install LabVIEW and the FPGA Module.

1. Log in as an administrator or as a user with administrative privileges.
2. Insert the LabVIEW Platform DVD 1.

   To request additional LabVIEW 2010 Platform DVDs, refer to the National Instruments Web site. If you purchased this product with an NI Software Suite or NI Product Bundle, use the installation media that shipped with your purchase to install this product.

Tip If the installer does not start automatically, you can double-click `setup.exe` from the media to launch the installer.

3. Follow the instructions on the screen to install and activate the following software.
   - **LabVIEW**
   - **FPGA Module**
   - **Compilation Tools for Devices other than Virtex-II**—You must install these compilation tools locally to create simulation exports, configure the IP Integration Node, or use the Configuring CLIP wizard.
   - (Optional) **Compilation Tools for Virtex-II FPGA Devices**—You must install these compilation tools only if your hardware contains a Virtex-II or Virtex-II Pro FPGA.
   - (Optional) **FPGA Compile Farm Toolkit**

- (Optional) **Real-Time Module**
- **Device Drivers**—NI-RIO is the device driver software for most FPGA targets. NI-RIO includes driver software for R Series, FlexRIO, and CompactRIO devices. You might need to install different or additional driver software for the FPGA target you use. Refer to the specific hardware documentation for information about the appropriate drivers and for information about installing and configuring the FPGA target.

Note You must install the NI-RIO 3.5 drivers to use NI-RIO with the FPGA Module 2010. Earlier versions of the NI-RIO drivers do not support the FPGA Module 2010.

In addition to installing program files and documentation in the LabVIEW directory, the installer also puts files from Xilinx ISE in the `x:\NIFPGA` directory, where *x* is the drive on which you installed LabVIEW. Xilinx ISE is third-party software that the FPGA Module uses to compile FPGA VIs into code that runs on an FPGA target.

Activating the LabVIEW FPGA Module

The FPGA Module relies on licensing activation. You have a temporary license for a 30-day evaluation period that includes both the FPGA Module and the Xilinx tools that the FPGA Module uses. When the evaluation period expires, you must activate a valid FPGA Module license to continue using the FPGA Module. Activating the FPGA Module license also activates the license for the Xilinx tools.

You can use the NI License Manager, available by selecting **Start»All Programs»National Instruments»NI License Manager**, to activate National Instruments products. Refer to the *National Instruments License Manager Help*, available by selecting **Help»Contents** in the NI License Manager, for more information about activating NI products.

LabVIEW 2010 FPGA Module New Features

The FPGA Module includes the following new features to help you better manage, implement, and validate your FPGA application. Refer to the *LabVIEW Help*, available by selecting **Help»LabVIEW Help** in LabVIEW, for more information about these features.

Compilation Process Improvements

Build Specifications

You now use build specifications to configure compilations options for your FPGA VI. LabVIEW uses the build specification options you set to generate the HDL and bitfile from the block diagram. You can configure multiple build specifications for the same FPGA VI to switch between different compilation configurations while developing your FPGA VIs.

Early Feedback on FPGA Resource Usage

For some FPGA target families, such as the Virtex-5, you can estimate how many FPGA resources your FPGA VI uses without compiling the FPGA VI. LabVIEW uses a Xilinx tool to estimate the number of slices, LUTs, and flip-flops that FPGA VI will use. Estimating resources before compiling allows you to decide if your FPGA design is close to fitting on the FPGA without waiting minutes to hours for the compilation to complete.

Simultaneous Compilation of FPGA VIs with the FPGA Compile Farm

If you purchase and install the FPGA Compile Farm, you can compile multiple FPGA VIs simultaneously as well as monitor and manage the compilation queue remotely. With this add-on you can install compile workers on different computers and let the FPGA compile farm manage all available compile workers on the network and load balance compilation requests. In addition, the new compile system has built-in fault tolerance, so if a compile worker becomes inoperable during a compilation, the compile server can reschedule the compilation job to a different compile worker.

Reusable Host SubVIs Using Dynamic FPGA References

You can now create reusable subVIs that work with interfaces from different FPGA VIs or bitfiles as long as the relevant portions of the communication hardware interface are consistent. The communication hardware interface includes all of the controls, indicators, DMA FIFOs, peer-to-peer FIFOs, and supported FPGA interface methods in the FPGA VI or bitfile. These subVIs can work with any FPGA reference that implements the specified interface, even if the reference is to VIs or bitfiles on different target types. For example, you can use a subVI for both a PCI and a PXI target. This feature is available only for targets using NI-RIO drivers.

Third-Party IP and HDL Integration Improvements

New IP Integration Node

The LabVIEW 2010 FPGA Module replaces the HDL Interface Node with the IP Integration Node. Use this node, available on the **Functions** palette when editing an FPGA VI, to integrate third-party IP code into FPGA VIs. The IP Integration Node supports most of what the HDL Interface Node supported except for implementation outside of a single-cycle Timed Loop. In addition, the IP Integration Node adds support for the fixed-point data type and Xilinx CORE Generator IP.

CLIP Wizard

Reduce errors when creating or modifying the CLIP declaration XML file by using the **Configure Component-Level IP** wizard (CLIP wizard). The CLIP wizard helps you define the IP interface without editing the declaration XML file by hand. You can use the CLIP wizard to create or modify a declaration XML file. The wizard also helps you define simulation models for CLIP so that you can export FPGA VIs containing CLIP for cycle-accurate simulation.

Generics Support for CLIP

CLIP now supports VHDL generics. Use the **Configure Component-Level IP** wizard to define generics in the IP interface.

Fixed-Point Data Type Support for CLIP

CLIP now supports the fixed-point data type. Use the **Configure Component-Level IP** wizard to define these data types in the IP interface.

Cycle-Accurate Simulation for Virtex-5 FPGA VIs

Now you can use LabVIEW to generate the necessary simulation files to simulate a Virtex-5 FPGA VI in a third-party simulator. Along with other debugging techniques, cycle-accurate simulation can help you test the timing behavior of FPGA VI components. Using cycle-accurate simulation in a hardware simulator, you can follow the states of individual signals that the FPGA VI components create.

FIFO and Memory Improvements

Support for Custom Data Types FIFOs and Memory

Now you can pass clusters and 1D fixed-size arrays of supported data types in and out of in target-scoped FIFOs and memory using custom controls. With this new functionality you can break the 64-bit barrier by joining smaller data types into larger data types while keeping resources separate. Using clusters and arrays, you can push related data into a FIFO without the risk of that FIFO timing out before it receives all data.

Improved Access to DRAM Memory through Memory Items in the Project

Some targets, such as those in the NI FlexRIO family, contain dynamic random access memory (DRAM) external to the FPGA. In LabVIEW 2009 and earlier, the only way to access DRAM was though socketed CLIP. In LabVIEW 2010 you can access the DRAM through memory items in a LabVIEW project. Accessing DRAM through memory items makes creating FPGA applications using DRAM easier.

Access to Built-In FIFO Control Logic on Supported Targets

The FPGAs on some hardware targets, such as those in the Virtex-5 family, provide FIFO control logic that is built into block memory. Built-in FIFO control logic saves significant FPGA resources by using dedicated control circuits in the block memory. In addition, these dedicated circuits can support higher clock rates than the control logic implemented in previous versions of LabVIEW.

Reducing Memory Resource Usage with Dual-Port Read Access

For some applications, where the values in memory do not change after compilation, you may be able to reduce the usage of block memory resources and/or improve execution time. For example, if coefficients relate to the same sine or cosine function, you can save the function once in a memory block and use two addresses to read the different phases. To take advantage of using dual-port read access, configure the second interface on the **Interfaces** page of the **Memory Properties** dialog box.

Math and Analysis VIs and Functions Improvements

The following functions are new for 2010.

- **Mean, Variance, and Standard Deviation Express VI**—Calculates the mean, variance, and/or standard deviation of an input signal.
- **High Throughput Complex Multiply Function**—Computes the product of two complex numbers with the same advantages of other High Throughput Math functions.
- **Reinterpret Number Function**—Shifts the decimal point of the bit pattern of x and returns y, where y has the same bit pattern as x but a different fixed-point configuration and value.
- **Basic Elements Functions**—Perform low-level FPGA operations such as accumulating totals, delaying signals, performing binary arithmetic, and accessing a DSP48E slice.

The following function has changed in 2010.

- **DC and RMS Express VI**—Now supports use inside a single-cycle Timed Loop.

Accessing DSP48E Slices on Supported FPGAs

Use the new DSP48E function to configure and execute a DSP48E slice on a Virtex-5 FPGA.

Control VIs Improvements

The following VIs and Express VI are new for 2010.

- **Discrete FP Transfer Function to FXP VI**—Converts a discrete, floating-point transfer function model to the fixed-point data type.
- **Discrete Transfer Function Direct VI**—Implements a fixed-point transfer function model in FPGA control applications.
- **Matrix*Vector Express VI**—Multiplies a matrix by an input vector. This functionality is useful for implementing state-space system models in FPGA control applications.

FPGA Device and Compiler Information Available in Software

You can find the following information about your FPGA device through a LabVIEW dialog box:

- Target class
- FPGA family
- FPGA type
- Compiler version used to compile the FPGA VI

Right-click an FPGA target in the **Project Explorer** window and select **Properties** from the shortcut menu. The FPGA device information appears on the **General** page of the **FPGA Target Properties** dialog box.

Upgrade and Compatibility Issues

The following sections provide information about upgrade and compatibility issues specific to upgrading from the FPGA Module 2009 to the FPGA Module 2010. For information about upgrading from previous versions of the FPGA Module, refer to the previous editions of the *LabVIEW FPGA Module Release and Upgrade* document available on `ni.com/manuals`.

Changes to the Compilation Process

In 2010 you must use build specifications to compile FPGA VIs. If a build specification does not exist already, LabVIEW automatically creates a build specification for the FPGA VI when you click the **Run** button on the top-level FPGA VI. You can update the options in the build specification by right-clicking the build specification and selecting **Properties** from the shortcut menu.

Changes to Dialog Boxes and Windows

- The **Compilation Status** window appears slightly different than it did in LabVIEW 2009 but maintains most of the same functionality. In addition to the functionality in LabVIEW 2009, you can monitor multiple compilations in LabVIEW 2010.
- The **Xilinx Options** page moved from the **FPGA Target Properties** dialog box to the **Compilation Properties** dialog box, available by right-clicking a build specification and selecting **Properties** from the shortcut menu.
- The **Target-Specific VI Properties** dialog box is deprecated. You can find compilation information in the **Compilation Status** window, available by right-clicking a build specification and selecting **Display Compilation Results** from the shortcut menu. To determine if you need to recompile an FPGA VI, right-click the build specification and select **Check Signature** from the shortcut menu.
- The following dialog boxes and windows are deprecated:
  - **LabVIEW FPGA Compile Server** window—You can find the status of compilation jobs in the **Compilation Status** window or the **Compile Worker** window.

  - **Manage All Server Records** dialog box—You can find information about past compilation jobs in the **Compilation Status** window.
  - **Configure Server** dialog box—You cannot specify changes to the working directory in LabVIEW 2010. The working directory is `c:\NIFPGA`.

Changes to FPGA VI Shortcut Menus

The following options that used to appear in an FPGA VI shortcut menu have moved or changed:

- **Compile**—Renamed to **Build** and moved to the shortcut menu of the build specification.
- **Reconnect to Compilation**—Renamed to **Display Compilation Results** and moved to the shortcut menu of the build specification. The **Display Compilation Results** menu item displays the **Compilation Status** window.
- **Download**—Moved to the shortcut menu of the build specification.
- **Analyze Timing Violation**—No longer available. Instead, select **Display Compilation Results** from the shortcut menu of the build specification and then click the **Investigate Timing Violation** button in the **Compilation Status** window to display the **Timing Violation Analysis** window. The **Investigate Timing Violation** button appears only if the compile server encounters timing violations while trying to compile an FPGA VI.

Changes to the Process for Compiling VIs Remotely

- The LabVIEW FPGA Compile Server has been modularized in 2010. In LabVIEW 2009 or earlier, the LabVIEW FPGA Compile Server application combined the compile server and the compile worker. In 2010, LabVIEW uses the compile server and the compile worker to compile FPGA VIs. The compile server is a system service that runs in the background on the computer where you install the compilation tools. The compile worker is a separate application. When you compile an FPGA VI, LabVIEW communicates with the compile server which then communicates with the compile worker.
- If you want to install the compile server and worker on a remote computer, install the Xilinx compilation tools from the LabVIEW 2010 Platform DVDs or other installation media on the remote computer.

- You still must launch the compilation tools on the remote computer before you can compile an FPGA VI remotely. For LabVIEW 2010, you must complete two steps before you can compile FPGA VIs remotely.
  - Select **Start»All Programs»National Instruments»FPGA» FPGA Compile Worker** to launch the compile worker.
  - Select **Start»All Programs»National Instruments»FPGA» FPGA Compile Server Configuration**. Place a checkmark in the **Allow users to connect remotely to this compile server** checkbox.

  Refer to the *LabVIEW Help* for more information about compiling an FPGA VI remotely.

Changes to Third-Party IP and HDL Integration

- The HDL Node is deprecated. You can use the IP Integration Node, available on the **Functions** palette, instead.
- LabVIEW 2010 does not support FPGA VIs that contain the IP Integration Node from `ni.com/labs`. If you have an FPGA VI that uses this version of the IP Integration Node, you must delete the node and re-add it from the **Functions** palette.
- Component-level IP declaration XML files created in LabVIEW 2009 or earlier might not be compatible with the LabVIEW 2010. National Instruments recommends saving a back-up of the old CLIP declaration XML file, then opening and modifying the file using the new **Configure Component-Level IP** wizard, available on the **Component-Level IP** page of the **FPGA Target Properties** dialog box.

Changes to the Utility VIs Palette

The Discrete Delay Express VI on the **Discrete Linear Systems** palette in LabVIEW 2009 has been renamed the Basic Discrete Delay Express VI.

Changes to FPGA Conditional Disable Symbols

The `FPGA_EMULATION` symbol is deprecated. Use the `FPGA_EXECUTION_MODE` symbol instead.

Where to Go from Here

National Instruments provides many resources to help you succeed with your NI products. Use the following resources as you start exploring LabVIEW and the FPGA Module.

Related Documentation and Examples

Use the following resources to learn more about using LabVIEW and the FPGA Module.

- **LabVIEW Help**—Available by selecting **Help»LabVIEW Help** in LabVIEW. Browse the **FPGA Module** book in the **Contents** tab for an overview of the FPGA Module and hardware-specific information. Browse the **FPGA Interface** book in the **Contents** tab for an overview of the FPGA Interface information.
- **Context Help Window**—Available by selecting **Help»Show Context Help**. Context help provides brief descriptions of VIs, functions, and dialog boxes. Context help for most VIs and functions include a link to the complete reference for a VI or function.
- **Hardware-Specific Documentation**—Some FPGA targets provide printed documentation as well as content in the *LabVIEW Help*. Use hardware-specific documentation for information about using the FPGA target with LabVIEW and for information about hardware specifications.
- **Examples**—The driver software for many FPGA targets includes corresponding examples. Refer to the specific hardware documentation for information about whether the FPGA target you use comes with corresponding examples.

  You can start with an existing example and use it as a starting point for developing FPGA VIs and host VIs. From LabVIEW, launch the NI Example Finder by selecting **Help»Find Examples**. Browse the examples by directory or by task.

NI Web Site

Visit ni.com/fpga for the latest NI Developer Zone articles, examples, and support information for the FPGA Module.

Refer to ni.com/info and enter the Info Code fpgatrn to access online training for the FPGA Module.

Known Issues

Refer to the National Instruments Web site at ni.com/info and enter the Info Code LVFPGA2010KI to access the known issues for the FPGA Module.