DIGITAL_IO Configurable Digital IO Module

Summary

This document describes how to place and use the configurable DIGITAL IO virtual instrument in an FPGA design.

Core Reference
CR0179 (v2.1) August 11, 2008

The DIGITAL_IO device is a configurable digital I/O instrument for use in an FPGA design. It provides separated inputs and outputs, allowing you to monitor and display signal levels, as well as define control signals for use elsewhere in the design.

Unlike its legacy counterparts (IOB_x family of devices), with the DIGITAL_IO device you are not constrained to limited signals of 8- or 16-bits. Instead, any number of signals may be added, and any number of bits can be assigned to a single signal. You may also have different numbers of input and output signals.

The DIGITAL_IO device also brings more variety in the way the inputs and outputs are displayed.

For information specific to the legacy Digital I/O instruments, IOB_x, refer to the *CR0102 IOB_x Digital IO Module* core reference.

Features

- Supports any number of input and output signals
- Each signal can be configured to any number of bits (typically 8, 16, 32)
- Customizable naming of input and output signals
- Defined signals can be reordered, graphically, as required
- Two-level display of inputs
  - Hexadecimal value
  - configurable graphic display – Numeric, LEDs, LED Digits, Bar
- Two-level control of outputs
  - Hexadecimal entry of entire value
  - Configurable graphic control – Numeric, LEDs, LED Digits, Slider
- Ability to set initial value for each output signal

Available Devices

The DIGITAL_IO device can be found in the FPGA Instruments integrated library (`FPGA Instruments.IntLib`), located in the `\Library\Fpga` folder of the installation.

Functional Description

Symbol

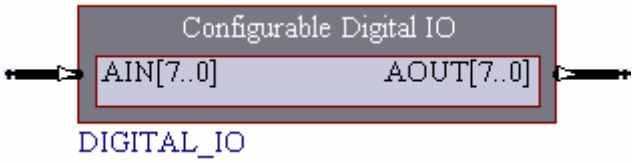


Figure 1. DIGITAL_IO Symbol.

Note: The image of Figure 1 represents the device when initially placed from the integrated library. The actual number of signals, name and width of those signals, will vary depending on how the device is configured.

Pin Description

The DIGITAL_IO device is very simple in its pinout. It can have any number of input signals and any number of output signals. A signal can be any number of bits wide, but this will typically be 8-, 16- or 32-bit for most designs. Each signal can also be customized in terms of its name.

For more information, see the section *Configuring the DIGITAL_IO instrument*, later in this document.

Placing a DIGITAL_IO Instrument in a Design

The configurable DIGITAL_IO instrument provides an efficient and uncomplicated means by which to monitor/generate digital signals in a design. Figure 2 shows a simple example of how the device is wired into an FPGA design.

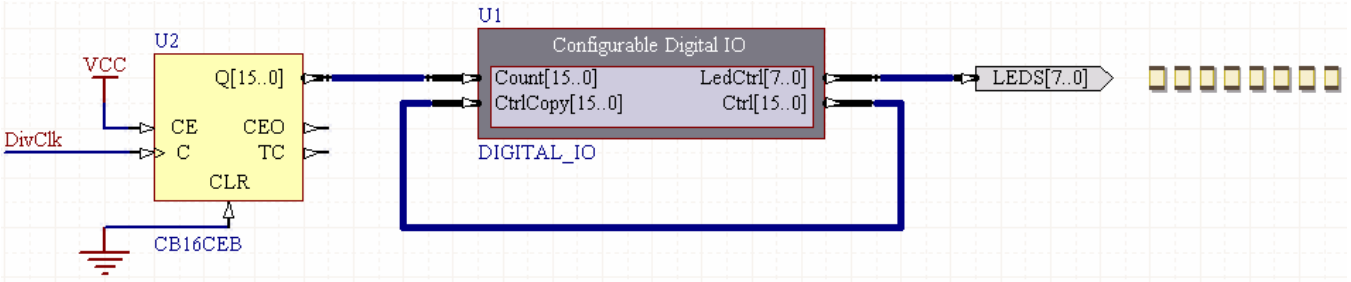


Figure 2. Using a DIGITAL_IO instrument to monitor and control areas of a design.

In the example circuit above, the DIGITAL_IO device has been configured with two inputs and two outputs. The use of these signals can be summarized as follows:

- The 16-bit input signal `Count` is used to monitor the output from a 16-bit cascadable binary counter.
- The 8-bit output signal `LedCtrl` is connected to the User LEDs on the Desktop NanoBoard NB2DSK01.
- The 16-bit output signal `Ctrl` is directly connected to the 16-bit input signal `CtrlCopy`. Whatever is set on the output side will be fed back and displayed on the input side.

Enabling the Soft Devices JTAG Chain

Communications from the Altium Designer software environment to embedded processors and virtual instruments in an FPGA design, is carried out over a JTAG communications link. This is referred to on the Desktop NanoBoard NB2DSK01 as the Soft JTAG (or Nexus) chain. Within Altium Designer, such devices included in the chain are presented in the **Devices** view as part of the Soft Devices chain.

The Soft JTAG chain signals (`NEXUS_TMS`, `NEXUS_TCK`, `NEXUS_TDI` and `NEXUS_TDO`) are derived in the NB2DSK01's NanoTalk Controller (Xilinx Spartan-3). As part of the communications chain, these signals are wired to four pins of the daughter board FPGA. To interface to these pins, you need to place the `NEXUS_JTAG_CONNECTOR` design interface component (Figure 3). This can be found in the FPGA NB2DSK01 Port-Plugin integrated library (`\Library\Fpga\FPGA NB2DSK01 Port-Plugin.IntLib`).

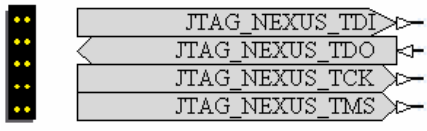


Figure 3. Nexus JTAG Connector.

This component 'brings' the Soft JTAG chain into the design. In order to wire all relevant Nexus-enabled devices (processors, virtual instruments) into this chain, you need to also place a `NEXUS_JTAG_PORT` component (Figure 4), and connect this directly to the `NEXUS_JTAG_CONNECTOR` (Figure 5). This component can be found in the FPGA Generic integrated library (`\Library\Fpga\FPGA Generic.IntLib`).

The presence of the `NEXUS_JTAG_PORT` component instructs the software to wire all components that possess the parameter `NEXUS_JTAG_DEVICE=True` into the Soft JTAG chain.

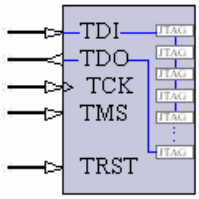


Figure 4. Nexus JTAG Port.

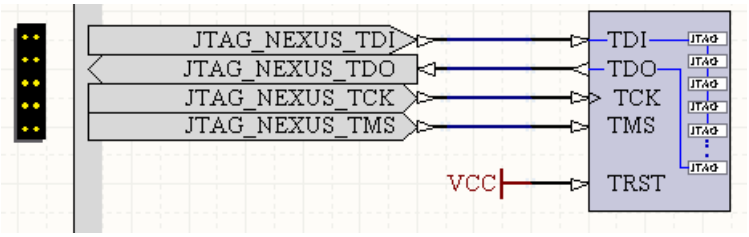


Figure 5. Connecting JTAG devices into the Soft JTAG chain.

Configuring the DIGITAL_IO Instrument

The DIGITAL_IO instrument can be configured after placement on the schematic sheet using the *Digital I/O Configuration* dialog (Figure 6). To access this dialog, simply right-click over the symbol for the device and choose the Configure command from the context menu that appears. Alternatively, click on the **Configure** button available in the *Component Properties* dialog for the device.

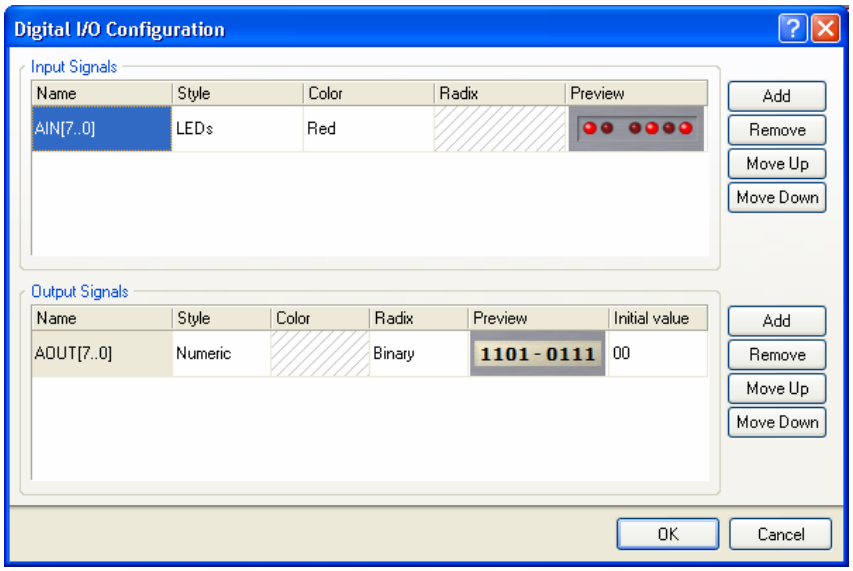


Figure 6. Configuration dialog for the DIGITAL_IO instrument.

Use the dialog to configure the device as required. The input and output signals are separate. You can define any number of signals for input or output, and the number of signals for each can differ. The following section takes a closer look at signal configuration.

Configuring Signals

Configuration of an input or output signal consists of defining the name and width of the signal, as well as the graphical representation for that signal on the device's instrument panel. Each signal will have the following associated fields in the dialog, with which to configure it.

- **Name** – this is the name of the signal, as it will appear on the schematic symbol and the instrument panel. Click on this field to edit the default signal name and replace it with a more meaningful name, in context with the area of the circuit being monitored/controlled. Set the width of the signal – in terms of bits – as required. Although the width specified can be any number of bits, this will typically be kept to 8, 16 or 32 bits.

- **Style** – this is the graphical display style used to represent the input signal (LEDs, Numeric, LED Digits, or Bar) or output signal (LEDs, Numeric, LED Digits, or Slider) on the instrument panel. For more detailed information on each style, see the next section – *Graphical display styles*.
- **Color** – the color to use for the chosen graphical display style. This field is not applicable to the Numeric or Slider display styles.
- **Radix** – the number system to use for the chosen graphical display style. This field is not applicable to the LEDs, Bar, or Slider display styles.
- **Preview** – shows an example image of how the chosen display style will look when viewed on the instrument panel.
- **Initial Value** – applicable to an output only, this field allows you to control the initial value for an output when the program is first downloaded to the FPGA and the instrument starts running.

In addition, the following four buttons are associated with both the **Input Signals** and **Output Signals** regions of the dialog:

- **Add** and **Remove** – add a new signal to the device or remove an existing one
- **Move Up** and **Move Down** – use these buttons to control the order of the signals, as they appear on the schematic symbol.

Graphical Display Styles

The following tables illustrate the various display combinations available for each supported display style. In each case, an example of the resulting display on the instrument panel is shown. For consistency, each display on the instrument panel is set to show the 16-bit value `1010_0000_1111_0101`.

Table 1. Display combinations for style LEDs.

Style	Color	Example of display on instrument panel	Comments
LEDs	Mixed		0 = Red, 1 = Green
	Red		0 = OFF, 1 = ON
	Green		0 = OFF, 1 = ON

Table 2. Display combinations for style Numeric.

Style	Radix	Example of display on instrument panel
Numeric	Binary	
	Octal	
	Decimal	
	Hexadecimal	

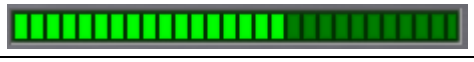
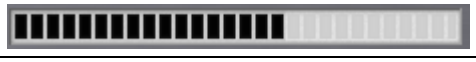
Table 3. Display combinations for style Slider.

Style	Example of display on instrument panel
Slider	

Table 4. Display combinations for style LED Digits.

Style	Color	Radix	Example of display on instrument panel
LED Digits	Red	Binary	
		Octal	
		Decimal	
		Hexadecimal	
	Green	Binary	
		Octal	
		Decimal	
		Hexadecimal	
	Orange	Binary	
		Octal	
		Decimal	
		Hexadecimal	
	Black	Binary	
		Octal	
		Decimal	
		Hexadecimal	

Table 5. Display combinations for style Bar.

Style	Color	Example of display on instrument panel
Bar	Red	
	Green	
	Orange	
	Black	

Operating the DIGITAL_IO Instrument

Once the DIGITAL_IO device has been configured and the design processed and downloaded into the physical FPGA device, the instrument can be used. Displays and controls for the instrument can be found on the device's associated instrument panel. This panel enables you to effectively use the instrument in your design.

The following sections detail how the instrument panel is accessed, and subsequent use of the panel to monitor and generate digital signals.

Accessing the DIGITAL_IO Module's Instrument Panel

The host computer is connected to the target DIGITAL_IO instrument using the IEEE 1149.1 (JTAG) standard interface. This is the physical interface, providing connection to physical pins of the FPGA device in which the instrument has been embedded.

The Nexus 5001 standard is used as the protocol for communications between the host and all devices that are debug-enabled with respect to this protocol. This includes the digital I/O modules, as well as other Nexus-compliant devices such as debug-enabled processors, frequency generators, logic analyzers, counters and cross-point switches.

All such devices are connected in a chain – the Soft Devices chain – which is determined when the design has been implemented within the target FPGA device and presents within the **Devices** view (Figure 7). It is not a physical chain, in the sense that you can see no external wiring – the connections required between the Nexus-enabled devices are made internal to the FPGA itself.

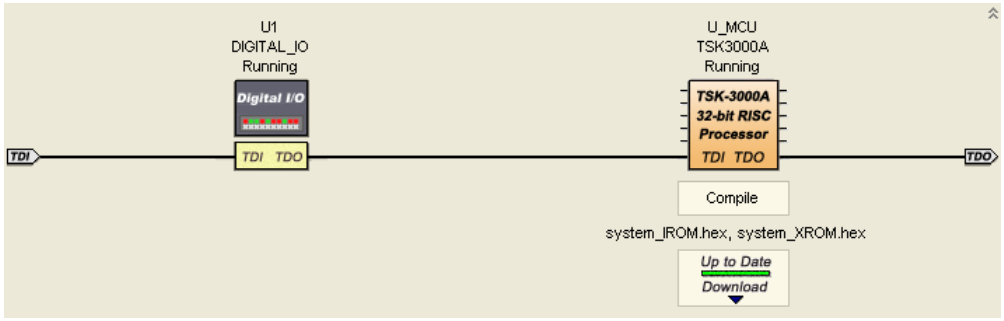


Figure 7. Nexus-enabled devices appearing in the Soft Devices chain.

For instruments such as the DIGITAL_IO device, the Nexus protocol enables you to access the registers used for controlling the device. These registers are not exposed as such, rather input to them is provided through an instrument panel, which allows you to view digital inputs and set digital outputs.

The controls for a DIGITAL_IO instrument used in a design can be accessed from the **Devices** view. Simply double-click on the icon representing the DIGITAL_IO device whose controls you wish to access, in the Soft Devices region of the view. The **Instrument Rack – Soft Devices** panel will appear, with the chosen instrument added to the rack (Figure 8).

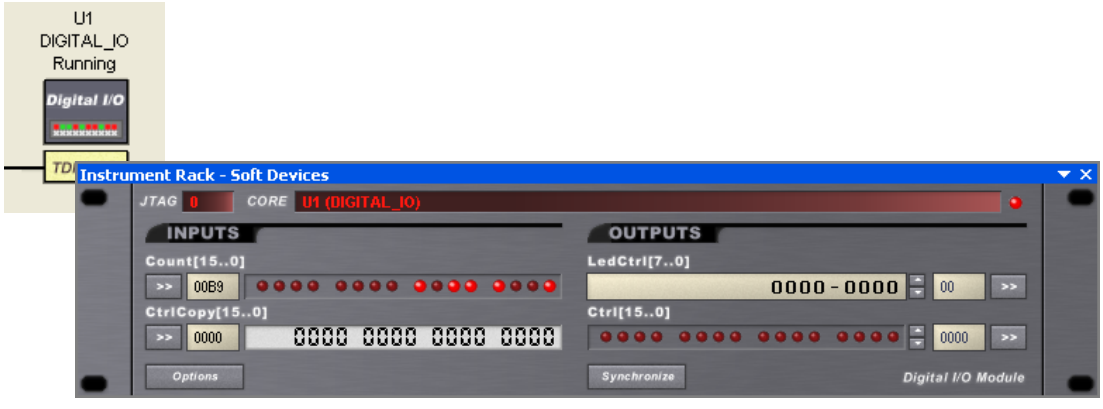


Figure 8. Accessing the Digital I/O module instrument panel.

Note: Each DIGITAL_IO device that you have included in the design will appear, when double-clicked, as an instrument in the rack (along with any other Nexus-enabled devices).

Monitoring Digital Signals – Inputs

The **INPUTS** region of the instrument panel allows you to monitor the digital signals that have been wired to the input channels of the DIGITAL_IO instrument.

For each input signal defined for the instrument, the incoming digital value will be presented in hexadecimal format, as well as being displayed using the graphical display style defined for that input when configuring the device.

Any changes to input signals will be reflected on the panel in real-time, in accordance with defined polling (see *Polling*).

Generating Digital Signals – Outputs

The **OUTPUTS** region of the instrument panel allows you to generate digital signals for output to other areas of the design.

For each output signal defined for the instrument, the digital value to be generated will be presented in hexadecimal format, as well as being displayed using the graphical display style defined for that output when configuring the device.

To specify a new signal value, either enter the full hexadecimal value directly, or use the graphical representation. As you make a change to one output representation (e.g. graphical), the second representation (e.g. hexadecimal) will update accordingly to match the value specified.

How the value is changed when using the graphical representation, depends on the representation used. For example, if using the LEDs style, simply click on a LED to toggle its state and change the output of the corresponding bit to '0' or '1'. If using the Slider style, simply move the slider bar to the position required – an example of where this style of representation is appropriate could be in the control of image brightness/contrast.

Any change made will be output in real-time, in accordance with defined polling (see *Polling*). Consider, for example, a DIGITAL_IO device with an 8-bit output signal hooked up to the User LEDs on the Desktop NanoBoard NB2DSK01. We will assume the graphical representation used is Numeric, with a Radix of Binary, and that the value on the panel for this output is initially `00h`. Changing a bit in the output value simply requires clicking on a bit in the representation. Changing bit `0` from '0' to '1' would instantly light the corresponding LED on the board (`LED0`).

To extend this example a little further, you could also feed the output signal back as an input signal to the instrument, as shown in Figure 9.

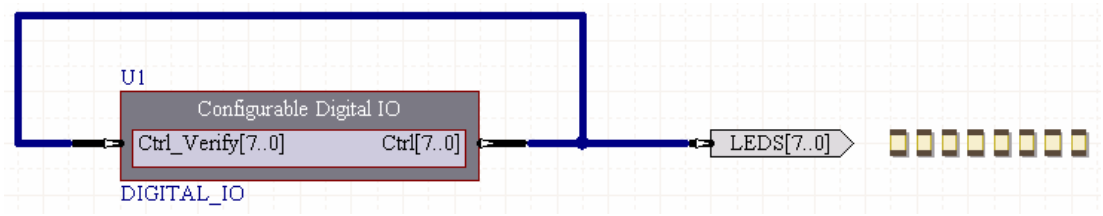


Figure 9. Generating and monitoring the same digital signal.

In this case, changing the output signal `Ctrl` will reflect not only in the LEDs on the NanoBoard, but also in the display for the input signal `Ctrl_Verify` – on the device's instrument panel. In Figure 10, setting bits `0`, `2`, `4` and `6` of the `Ctrl` output signal results in bits `0`, `2`, `4` and `6` of the input signal `Ctrl_Verify` also becoming set. You can see that the hexadecimal value for each signal is the same. On the NanoBoard, `LED0`, `LED2`, `LED4` and `LED6` would also be lit.

Figure 10. Direct feedback of output signal Ctrl to input signal Ctrl_Verify.

When processing a design for the first time, and downloading to the FPGA, the initial value specified for an output signal will appear displayed in the instrument panel, the first time it is accessed. Any changes to the output in the instrument panel will override this initial value immediately.

The value specified for an output signal in the instrument panel is persistent during the same session of Altium Designer. Should you reload the FPGA with the design, the initial value setting will be immediately replaced with the value current for the output in the panel. This happens if the panel is open. If it is closed, the initial value will be used for the output until subsequent access of the instrument panel, whereby the output will be set to whatever value is present for it in the panel.

Polling

Click on the Options button, at the bottom left of the panel, to access the *Digital I/O Module – Options* dialog (Figure 11). Input values to, and output values from, the Digital I/O Module, are automatically refreshed (updated) in accordance with the value specified in the **Update Display From Core Every** field of this dialog. In other words, this is the defined polling interval for the instrument. By default, the values will be refreshed every `250ms`.

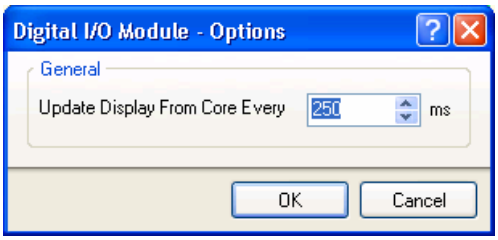


Figure 11. Accessing instrument options.

You can, if desired, manually refresh the IO values to/from the instrument – manual polling if you will. Such polling can be engaged at the global level, affecting all input and outputs simultaneously, or at the individual input/output level:

- *Global IO update* – simply click the Synchronize button, at the bottom-center of the panel.
- *Individual input/output update* - click on the >> button associated with the input/output signal you wish to update.

Note: Use of the Synchronize or >> buttons is only logical if the Update value in the *Digital I/O Module – Options* dialog is set to either `0ms`, or some longer time interval (e.g. `2000ms` or more). If the IO values are left to be automatically refreshed at the default `250ms` interval, this will be too quick to make use of these manual polling-related buttons!

Revision History

Date	Version No.	Revision
29-Aug-2007	1.0	Initial release
06-Mar-2008	2.0	Updated for Altium Designer Summer 08
11-Aug-2008	2.1	Information on use of Options and Synchronize buttons made clearer and moved into the new section – Polling.