



Re-targeting the Design to the Production Board

Summary

Application Note
AP0104 (v2.0) February 28, 2008

This application note covers the process of adding the configuration and constraint file information required to perform synthesis and run the vendor place and route.

Once the design has been captured as schematics and VHDL source files, the first thing you will want to do is to see if it can be synthesized and if a place and route can be performed.

What is required to target a design to your board?

The design captured in the schematics and VHDL defines the behavior of the circuit. To map this design into an FPGA which is part of a larger PCB design, requires the addition of all the necessary implementation information. This information could include:

- the target PCB project
- the target FPGA
- FPGA net-to-physical device pin assignments
- pin configuration information, such as output voltage settings
- design timing constraints, such as allocating a specific net to a special function FPGA pin
- FPGA place and route constraints.

How the nets in the design connect to the FPGA pins

Any net that you want connected to a physical pin on the FPGA must be wired to a port on the top schematic sheet. When the design is compiled, the top sheet is scanned and all nets that connect to ports are assumed to connect to physical pins on the FPGA.

In the FPGA design domain, these connection points are also referred to as *ports*. While the ports are defined on the schematic, the pins they are assigned to are defined in a constraint file.


As well as placing a port to identify a net (or bus) as connecting to a physical pin (or set of pins), you can also use a component, as long as it includes the parameter `PortComponent = True`. When the design is synthesized, each pin in the Port Component is converted to a port, with the port named the same as the pin designator.


How the implementation information is included

The implementation information is stored in *constraint* files. A constraint file is a set of constraint records, where each record (or constraint group) defines one or more constraints to be applied to a target in the FPGA project. The constraint file system supports storing all this information in one constraint file, or splitting it into multiple files.

The minimum constraint information required to move to synthesis is the device specification.

Like source documents, constraint files are added to the project. They are then mapped to the FPGA project by creating a suitable *Configuration*, where the configuration is a set of constraints that map the FPGA design to a target implementation (the actual device on a PCB).

 For a detailed description of configurations and constraint files, see the [Design Portability, Configurations and Constraints](#) article.

 For detailed information on constraint syntax, see the [Constraint File Reference](#).

Creating a constraint file and specifying the device

1. To add a new constraint file to your project, right-click on the project name in the **Projects** panel, and select **Add New to Project » Constraint File** from the menu that appears.
2. The new constraint file will appear. To specify the target device, select **Design » Add/Modify Constraint » Part** from the constraint file editor menu. The *Choose Physical Device* dialog will appear, from where you can browse for and select the

Re-targeting the design to the Production Board

required target device. Once selected, click **OK** – a record will be added to the constraint file, specifying the device, something like:

```
Record=Constraint | TargetKind=Part | TargetId=XC2S300E-6PQ208C
```

3. Save the constraint file. Typically it would be named to reflect its role – for example if the target device was an Altera Cyclone mounted on your project PCB you might call the constraint file `MyProject_Cyclone.Constraint`.

Assigning nets to FPGA pins in the constraint file

The net (or port)-to-physical pin assignments also need to be defined in a constraint file. You can manually define the assignments, or let the place and route tools assign them and then import the assignments back into the constraint file.

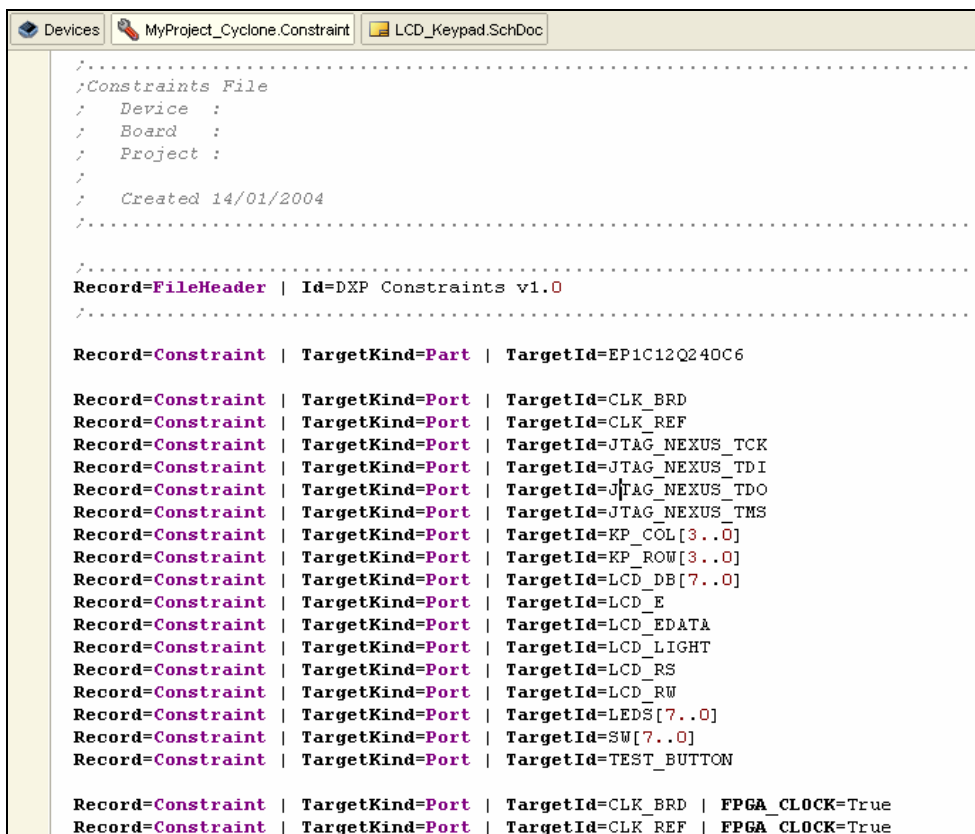
Typically there will be certain ports that you will want to assign, such as clock nets, then let the place and route process assign the rest. Once the FPGA is placed on the PCB, pin assignments can be changed to optimize the PCB routing and the changes back-annotated to the FPGA project.

1. To add port assignments to the constraint file, select **Design » Import Port Constraints from Project** from the constraint file editor menu. A constraint record will be added for each port in the FPGA project. At this stage there will be no actual pins assigned to the ports.
2. To manually assign a port to an FPGA pin, you add the `FPGA_PINNUM` constraint to the record. For example, to assign `CLK_REF` to pin 185 in a Xilinx device, the constraint would look like:

```
Record=Constraint | TargetKind=Port | TargetId=CLK_REF | FPGA_PINNUM=P185
```

3. For a clock you may prefer to instruct the place and route tools which ports are to be assigned to clock pins, then let the place and route tool choose from the available clock pins on the target device. To constrain the `CLK_REF` port to an FPGA clock resource, the constraint would look like:

```
Record=Constraint | TargetKind=Port | TargetId=CLK_REF | FPGA_CLOCK=TRUE
```



```
Devices | MyProject_Cyclone.Constraint | LCD_Keypad.SchDoc
/.....
;Constraints File
; Device :
; Board :
; Project :
;
; Created 14/01/2004
/.....

Record=FileHeader | Id=DXP Constraints v1.0
/.....

Record=Constraint | TargetKind=Part | TargetId=EP1C12Q240C6

Record=Constraint | TargetKind=Port | TargetId=CLK_BRD
Record=Constraint | TargetKind=Port | TargetId=CLK_REF
Record=Constraint | TargetKind=Port | TargetId=JTAG_NEXUS_TCK
Record=Constraint | TargetKind=Port | TargetId=JTAG_NEXUS_TDI
Record=Constraint | TargetKind=Port | TargetId=JTAG_NEXUS_TDO
Record=Constraint | TargetKind=Port | TargetId=JTAG_NEXUS_TMS
Record=Constraint | TargetKind=Port | TargetId=KP_COL[3..0]
Record=Constraint | TargetKind=Port | TargetId=KP_ROW[3..0]
Record=Constraint | TargetKind=Port | TargetId=LCD_DB[7..0]
Record=Constraint | TargetKind=Port | TargetId=LCD_E
Record=Constraint | TargetKind=Port | TargetId=LCD_EDATA
Record=Constraint | TargetKind=Port | TargetId=LCD_LIGHT
Record=Constraint | TargetKind=Port | TargetId=LCD_RS
Record=Constraint | TargetKind=Port | TargetId=LCD_RW
Record=Constraint | TargetKind=Port | TargetId=LEDS[7..0]
Record=Constraint | TargetKind=Port | TargetId=SW[7..0]
Record=Constraint | TargetKind=Port | TargetId=TEST_BUTTON

Record=Constraint | TargetKind=Port | TargetId=CLK_BRD | FPGA_CLOCK=True
Record=Constraint | TargetKind=Port | TargetId=CLK_REF | FPGA_CLOCK=True
```

Figure 1. Sample constraint file, with two clock nets constrained by the `FPGA_CLOCK=TRUE` constraint. Note that the pin allocations have not been defined yet.

When the place and route process is complete, the pin assignments can be imported into the constraint file.

Creating a Configuration

Constraints can be divided over a number of constraint files, and a project can include a number of different constraint files that are used to target the design to different implementations. To allow you to group constraint files and to move easily from one implementation to another, you define configurations.

A configuration is a named list of constraint files. Configurations are stored in the project file.

1. To define a configuration, right-click on the project name in the **Projects** panel, and select **Configuration Manager** from the menu that appears.

The *Configuration Manager* dialog lists all constraint files in the project down the left and all current configurations across the top. Constraint files are assigned to a configuration by enabling the appropriate checkbox. Figure 2 shows the *Configuration Manager* dialog with three constraint files listed on the left, the two default constraints and one created specifically for the FPGA project. Three configurations have been defined — two targeting the NanoBoard and one the project board.

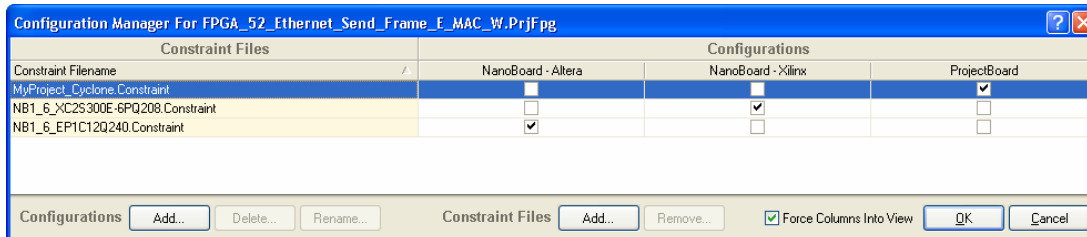


Figure 2. The configuration manager showing three configurations, each with one constraint file assigned.

2. Click the Configurations **Add** button at the bottom left to add a new configuration.
3. Enable the appropriate checkbox to allocate the constraint file to the configuration.

You are now ready to synthesize and place and route the design.

Design Synthesis


Synthesis is the process of translating the schematic and behavioural VHDL descriptions of the design into a low-level form suitable for the vendor place and route tools. The synthesis engine first creates an intermediate VHDL description of the design, and then synthesizes this into EDIF.

Synthesis can be launched in a number of ways:

- right-click on the project file and select **Synthesis** from the menu that appears
- open the **Devices** view and click the **Synthesize** button.

Note that whenever a synthesis is performed, a suitable configuration must be available. If there are multiple configurations to choose from, a selection dialog will appear. In the **Devices** view, the project and configuration combination are selected in the drop down list.

During synthesis, use the **Messages** panel to monitor the progress and double-click to explore any errors that occur.

 For information on using the **Devices** view to process the design, see the [Processing the Captured FPGA Design](#) application note.

Vendor place and route software

The task of implementing the design in the actual FPGA is carried out by specialised tools, referred to as place and route tools. These software tools are provided by the FPGA vendors, who with their intimate knowledge of the features available within each FPGA, can write software that can take a low-level description of the design and work out how to arrange, or *place*, the required logic inside the FPGA, then create the interconnections (or *route* it).

The appropriate place and route tools must be installed, most FPGA vendors have freely downloadable versions available on their website.

It is not necessary to actually run the place and route software, this is done automatically from the **Devices** view. When you click the **Build** button in the **Devices** view, the place and route software is run remotely. All feedback from the place and route software is displayed in the **Output** panel.


Back-annotation of pin assignments

Any pins (ports) that were not assigned in the constraint file prior to place and route are automatically assigned a physical pin during place and route. These assignments are needed before the FPGA design can be linked to the PCB design. To import the pin assignments from the place and route tool:

1. Open the constraint file, then select **Design » Import Pin File » Select File** from the menu.

Re-targeting the design to the Production Board

2. Navigate to the appropriate vendor-generated pin file. It will be located in a folder with a name like `MyFPGAProject\ProjectOutputs\ProjectBoard`.
3. After selecting the pin file and clicking **OK**, the *Constraint Editor Preferences* dialog will appear. If you enable an option in the dialog, that constraint information will be extracted from the file (if it is available) and included in your constraint file.
4. Save and close the Constraint file. The design is now ready to be linked to the target PCB.

 For more information on linking the FPGA project to the PCB project and transferring design changes back and forth, see the [Linking an FPGA Project to a PCB Project](#) application note.

Converting Parts to Ports

If you have used NanoBoard port plug-in components in your design, you will probably prefer to display them as ports when the design is moved to your own board. Use the **Tools » Convert » Convert Part To Ports** to achieve this. If you change port names as well you will need to update the Constraint file.

Revision History

Date	Version No.	Revision
19-Dec-2003	1.0	New product release
06-Jul-2005	1.1	Updated for Altium Designer SP4
28-Feb-2008	2.0	Updated for Altium Designer Summer 08

Software, hardware, documentation and related materials:

Copyright © 2008 Altium Limited.

All rights reserved. You are permitted to print this document provided that (1) the use of such is for personal use only and will not be copied or posted on any network computer or broadcast in any media, and (2) no modifications of the document is made. Unauthorized duplication, in whole or part, of this document by any means, mechanical or electronic, including translation into another language, except for brief excerpts in published reviews, is prohibited without the express written permission of Altium Limited. Unauthorized duplication of this work may also be prohibited by local statute. Violators may be subject to both criminal and civil penalties, including fines and/or imprisonment. Altium, Altium Designer, Board Insight, Design Explorer, DXP, LiveDesign, NanoBoard, NanoTalk, P-CAD, SimCode, Situs, TASKING, and Topological Autorouting and their respective logos are trademarks or registered trademarks of Altium Limited or its subsidiaries. All other registered or unregistered trademarks referenced herein are the property of their respective owners and no trademark rights to the same are claimed.