XAPP419 (v1.0) October 28, 2002

What is the Pinout Area Constraints Editor (PACE)

Summary

This application note discusses the fundamental flows of the Pinout Area Constraints Editor (PACE) tool. The PACE tool was created to simplify constraining tasks that are performed relatively early in the design process: I/O Pin assignment and Area Group creation. Widespread PACE usage is anticipated, especially for I/O Pin assignment, as all users must perform this task for every design. Rapidly increasing package sizes and I/O counts make PACE a particularly vital tool.

Pin assignments may be made in the Pin Package View and Architecture View using Drag and Drop functionality or in the Design Object List View (for I/Os) by either using drop-down menus or entering the information manually. The Area Groups are used in incremental design and generic design applications. This application note will discuss how this new tool fits into the Xilinx flow, and how a user may best utilize it to simplify the task of creating physical constraints.

Tool Background

The PACE tool is launched via a command line or the Project Navigator GUI. The command line is “pace” (without quotation marks). In Project Navigator, the project must include a UCF file; you may select "Assign Package Pins" or "Create Area Constraints" as illustrated in Figure 1. PACE writes all physical constraints into the UCF file.

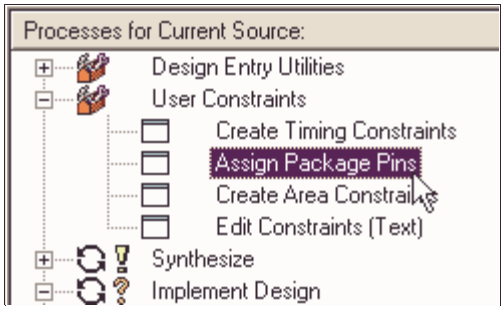

Figure 1: **Launching PACE from Project Navigator**

You may launch the PACE tool with two different versions of your design. If you are in the beginning phases of your design, and your ports are only declared in the hardware description language, you may run the tool to assign pins only. (This does require at least one piece of logic, usually a global buffer.) If your design contains hierarchy, you may run the tool to perform both pin assignments and area constraints.

After the tool has been launched from Project Navigator, it will automatically load the NGD and UCF files. If your design contains signals that are dangling or are not connected, the tool will ask you if you want them to be displayed, as illustrated in Figure 2. The most common

response is "No", but there are exceptions to the rule — if only the ports are declared, you would answer "Yes.

Figure 2: **PACE to List Dangling Nets**

The primary windows that appear after you launch the PACE tool are the Package Pin view, the Package Pin Legend, the Device Architecture view, the Design Object List (DOL), and the Hierarchy view.

The Package Pin view displays all pins for the design and the specific package, as well as the different banks associated with the package. The Package Pin Legend window is a companion to the Package Pin window, and it displays the color-coding of the banks and other predefined pins. If you select *Show Differential Pairs* from the *IOBs* menu, the differential pairs will be displayed with red lines, as illustrated in Figure 3. You may also change the perspective from Top to Bottom by using either the Preferences dialog box or a pop-up menu. You may also disable the use of I/Os and place I/Os in this view.

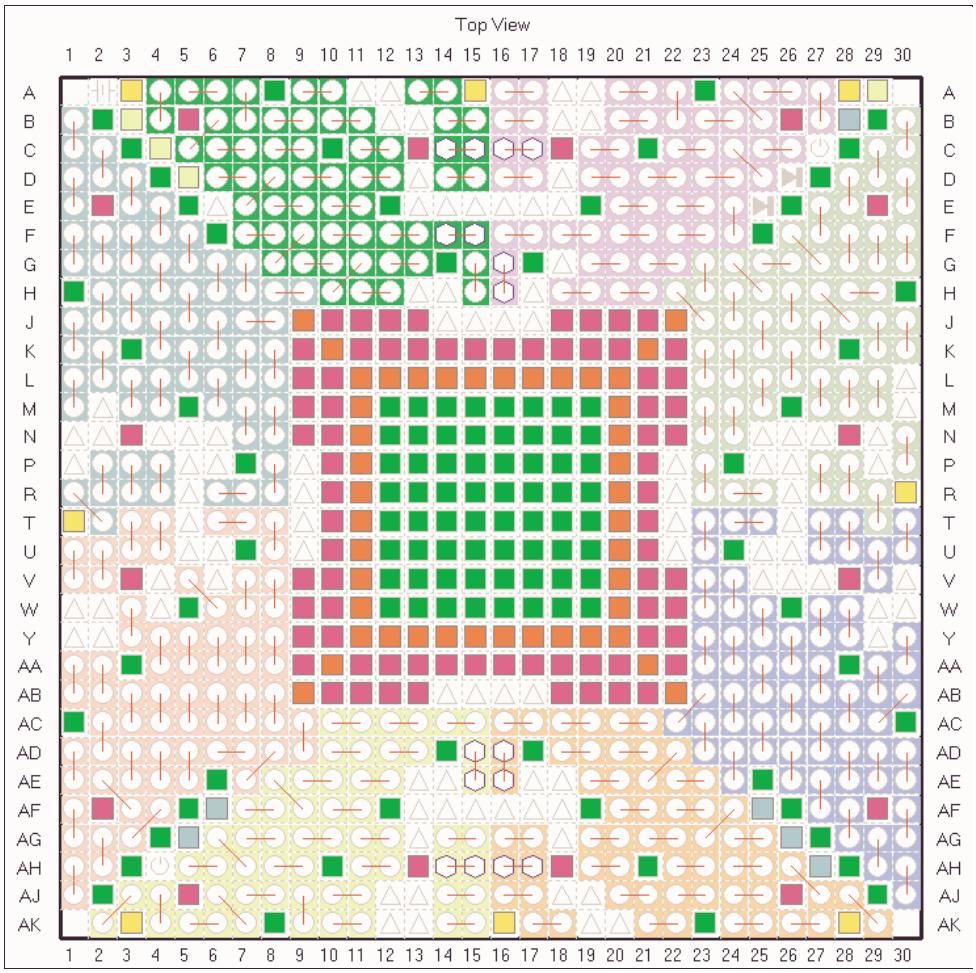

Figure 3: **Package Pin View with Differential Pairs**

The Device Architecture view shows all IOBs around the edge of and the Slices/BRAMs/Multipliers in the FPGA array. It also displays the various banks and the color-coding associated with each. The differential pairs can be turned on the same way as in the Package Pin view and will be displayed with red arcs, as illustrated in Figure 4. You may disable the use of I/Os and array features and place I/Os and area constraints in this view.

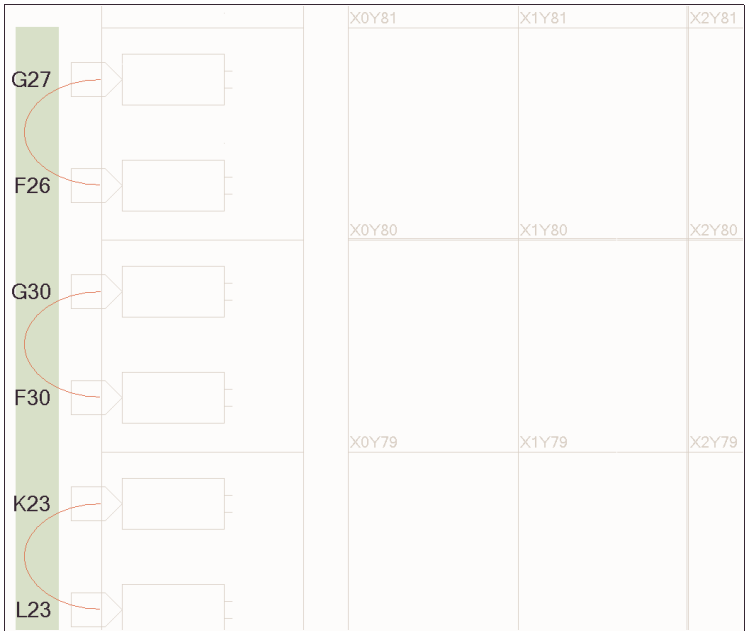

Figure 4: **Device Architecture View with Differential Pairs**

The Hierarchy view displays the folders of I/O Pins, Global Logic, and Logic. The I/O Pins folder includes all of a design's I/Os. The Global Logic folder includes the block RAMs, DCM/DLLs, Multipliers, Global Buffers, GTs, and PowerPCs. The Logic folder includes the hierarchy of the

design. By expanding each folder, you can drag and drop specific elements into the Package Pin view and/or the Device Architecture view, as illustrated in Figure 5.

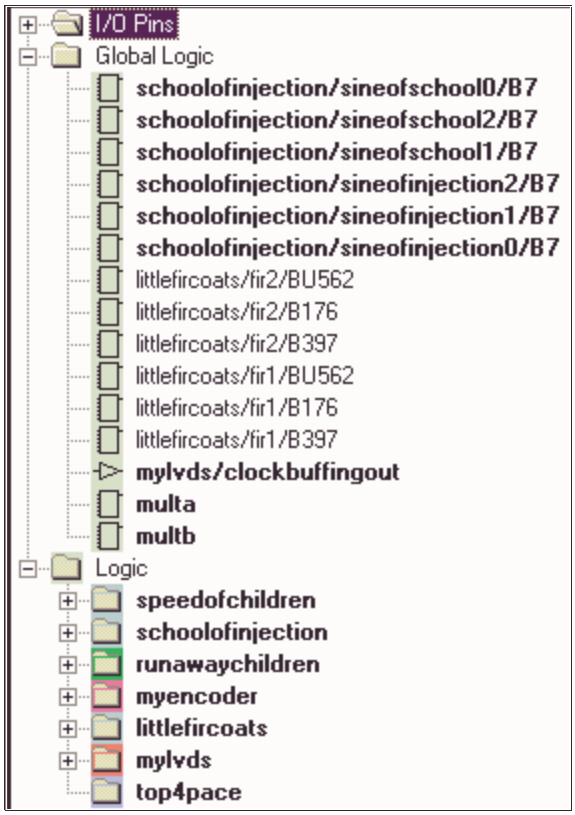

Figure 5: **Design Hierarchy with Expanded Folders**

The Design Object List displays the elements associated with the folder selected in the Hierarchy view. If you select the I/O Pins folder in the Hierarchy view, the DOL displays all of a design's I/Os, along with the specific direction, I/O Standard, and pin location, as shown in Figure 6. In the Location column, you may chose to type in the specific pin location (A15), the side of the device (TR = Top Right, R = Right, etc.), or the bank of the device (Bank1). The Bank column is read-only and represents the bank on which the pin is placed. You may group I/Os and type the bank/side location, or drag and drop the I/Os into the Package Pin view or Device Architecture view.

	I/O Name	I/O Direction	Location	Bank	I/O Std.
	Injectschool<0>	Output			
	Injectschool<1>	Output			
	Mary<7>	Input			
	Mary<6>	Input			
	Mary<5>	Input			
	Mary<3>	Input			
	Mary<2>	Input			
	Mary<1>	Input			
	Reset	Input			
	Play_P<71>	Output			LVDS_25

	Group	I/O Direction	Bank	I/O Std.
	Lamb_Grp	Output		
	laugh1019_grp	Output		LVDS_25

Figure 6: **DOL of I/O Pins with Groups**

If you select the Global Logic folder in the Hierarchy view, the DOL displays the high-level components. You may drag and drop the component into the Device Architecture view. If you select the Logic folder in the Hierarchy view, the DOL displays the hierarchical breakdown of the design. If you select a sub-module of the hierarchy, the DOL displays the hierarchy below that sub-module. You may also create area constraints based upon the hierarchical boundaries in the Device Architecture view.

Ports Declared: Pin Assignment

Pin assignment can be performed in the beginning design stages when only ports have been declared, or for a completely finished design. The main focus of assigning pins is specifying how your design's pins will be assigned to IOB sites as well as disabling the use of special or reserved I/Os and for I/O placement. You can disable the use of I/Os that are not pin-compatible with your device, special purpose pins, and any I/Os that you do not wish to use. You may also place your I/Os through various means, ranging from typing in a location to dragging and dropping the I/O to a specific location, and constraining a group of them to a bank.

One of the most common uses associated with pin assignment is the ability to make your design pin-compatible with devices within the same architecture family. For example, if your device is an XC2V1500-ff896-6, and you wish to make it pin-compatible with an XC2V1000-ff896-6, it is possible for a pin assignment to be the same between devices, as shown in Figure 7 and Figure 8. You can do this by selecting *Make Pin Compatible With* from the *IOBs* menu, and select the new device with which you wish the design to be pin-compatible. A prohibit command will be placed in the UCF for each I/O, and a gray box will be placed on I/Os that are not pin-compatible.

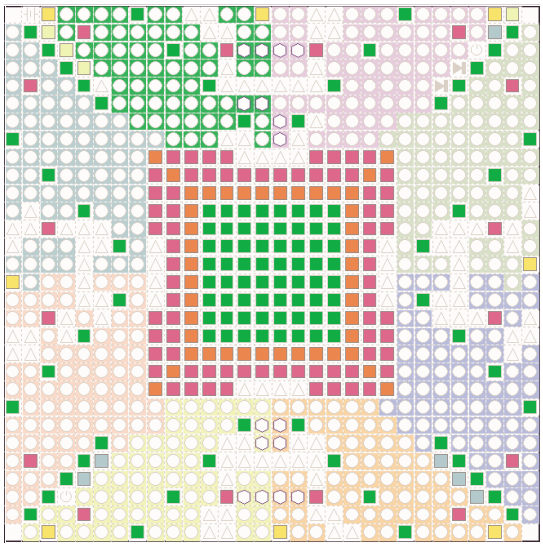

Figure 7: **Original Pinout — XC2V1500ff896**

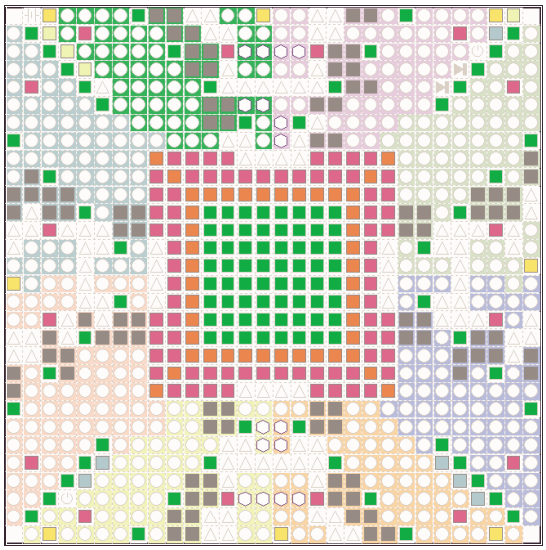

Figure 8: **Design Pin-Compatible with XC2V1000ff896**

Sometimes, you may wish to disable special purpose pins (including VREF and VR pins for each I/O Bank, as well as configuration mode pins) as shown in Figure 9. This is also done via the prohibit command in the UCF. The VREF and VR pins can be prohibited based on the I/O Standard that will be used in a particular bank. The configuration mode pins may also be prohibited based upon the way the device will be programmed or configured. To do this, select *Prohibit Special Pins* from the *IOBs* menu, and select the pins to be prohibited. The prohibited I/Os will be displayed as gray boxes.

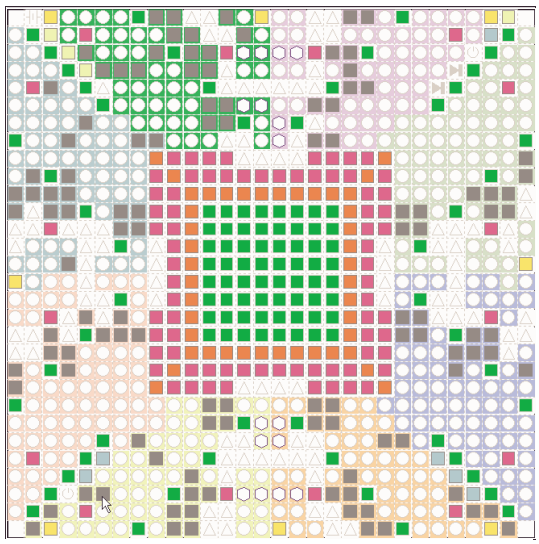

Figure 9: **Prohibit Special and Configuration Pins**

If the design might expand and require more pins in the future, a section of pins can be disabled. You may disable the use of IOBs, Slices, Multipliers, BRAMs, etc. The I/Os can be disabled in either the Package Pins view or the Device Architecture view; other array elements can only be disabled in Device Architecture view. Select *Prohibit Mode* from the *Tools* menu, and draw a box around the region you wish to prohibit in the Device Architecture view or

Package Pins view. The disabled array components and/or I/Os will be displayed as gray boxes, as shown Figure 10 and Figure 11.

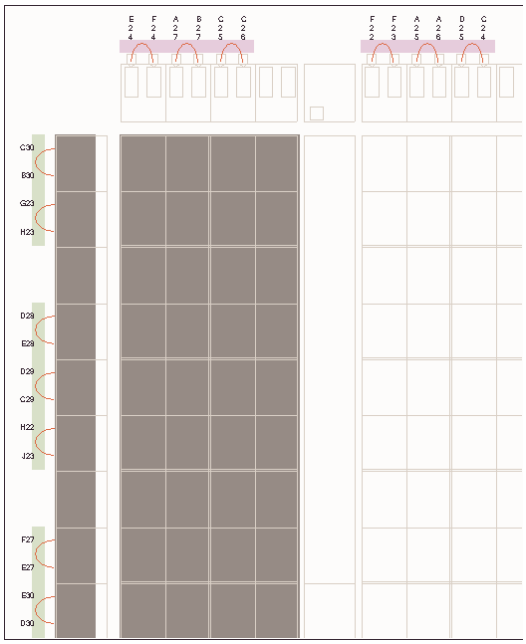

Figure 10: **Prohibit a Region in Device Architecture**

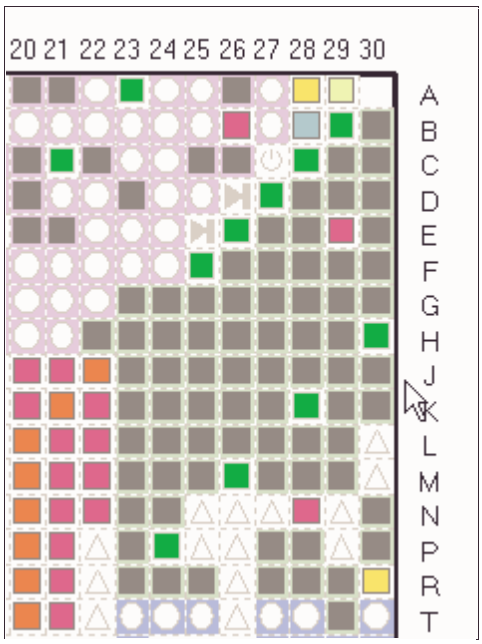

Figure 11: **Prohibit a Region in Package Pins**

If you elect to allow the use of these prohibited I/Os and/or array components, select *Allow Mode* from the *Tools* menu and draw a box around a region to remove the prohibits. By disabling usage of the pins that are not needed for the design, the actual pin assignment is carried out.

The placement of individual I/Os can be performed in several ways. You may drag and drop the I/Os from the DOL or Design Hierarchy into the Device Architecture or Package Pins view. You can also specify the exact location, bank, or side by typing it in, or choosing from the pull-down

list in the DOL for each I/O. You may also select the I/O Standard for each I/O. If your design contains differential pairs, placement of the P side of the pair to a correct P location will cause the N side to be automatically placed in the correct location.

If you have buses or many I/Os, you may group them and place the group. Drag and drop the group from the DOL into the Device Architecture or Package Pins view. You can also specify the bank/side and the I/O Standard for the group, as shown in Figure 12.

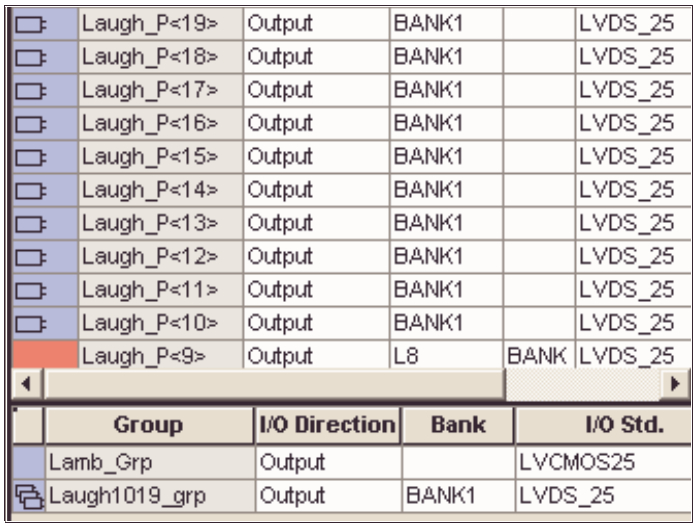

Figure 12: **Grouping of Buses in DOL**

After placing all I/Os and disabling the usage of any pins, you can run a Design Rules Check (DRC) on your pin assignments by selecting *Check Pin Assignment* from the *Tools* menu. PACE checks for IO Banking Rule violations. This will open a window that displays any errors discovered by the DRC on the placed I/Os. Selecting an Error message selects the corresponding IO, if applicable. You may fix the errors and perform a re-check on the I/Os from this window.

Hierarchical Design – Area Constraints and Global Logic

Once the design is partially or completely finished, you may create area constraints and place high-level design components. Area constraints are based upon the hierarchy of your design and are located under the Logic folder. The placement of high-level components is based upon the more critical components of the design, which are located under the Global Logic folder.

After pin assignment, the creation of area constraints is one of the most commonly performed functions. Area constraints are placed on different levels of hierarchy of the design and are used to restrict placement of the associated logic. (The PACE tool will not allow you to create your own groups.)

The area constraint is based upon the number of LUTs, flip-flops, and carry chain heights. The tool estimates the minimal area needed to fit the number of LUTs, flip-flops, and carry chain heights and adds a small amount of padding. The minimal area snaps to CLB boundaries and will cross over BRAM/Multiplier columns. You can change the padding from the default value by selecting *Area Padding* from the *Area* menu and increasing the value, as shown in Figure 13.

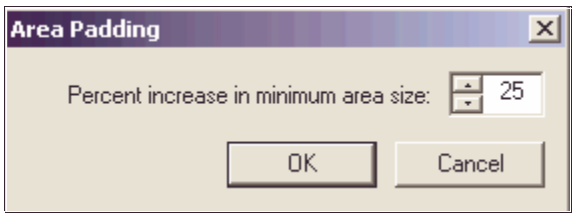

Figure 13: **Changing the Padding of the Area Constraint**

You may select a portion of the hierarchy in the Design Hierarchy by expanding the Logic folder. Then, select the folder associated with the piece of logical hierarchy and drag it to the Device Architecture view. If you select the Logic folder or a lower piece of the hierarchy, the DOL will reflect the lower levels of the hierarchy. You may then drag and drop from the DOL to the Device Architecture view. Either method will create a rectangle with the approximate minimal area, as illustrated in Figure 14.

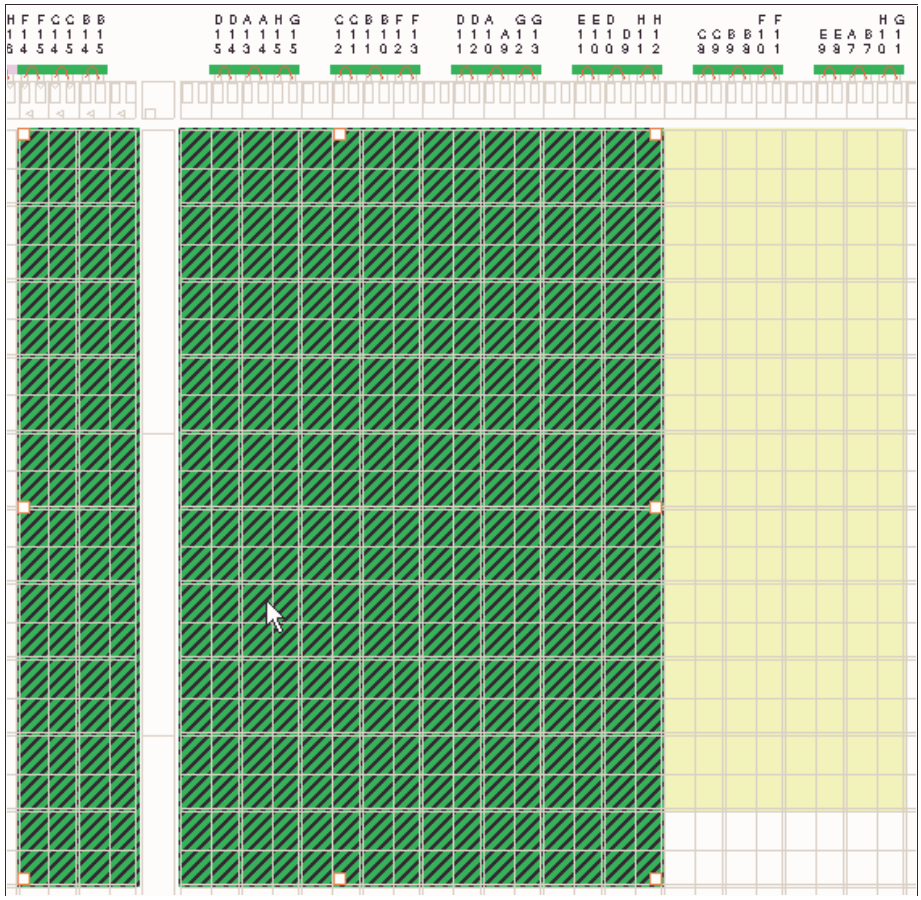

Figure 14: **Placing of Area Constraints in Device Architecture**

The shape and size of the area constraint may be changed. You can place the mouse over an area constraint, and the area constraint statistics will be displayed at the lower left-hand corner of the PACE window, as shown in Figure 15.

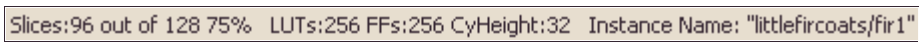

Figure 15: **Area Constraint Statistics**

You may also create non-rectangular area constraints by selecting an area constraint, pressing the right mouse button, and selecting *Add Rectangle*. A new rectangle will appear at the mouse point — place it next to the original area constraint. If a piece of your hierarchy interacts with

many I/Os, you could place an "L"-shaped area constraint next to several I/Os, as illustrated in Figure 16.

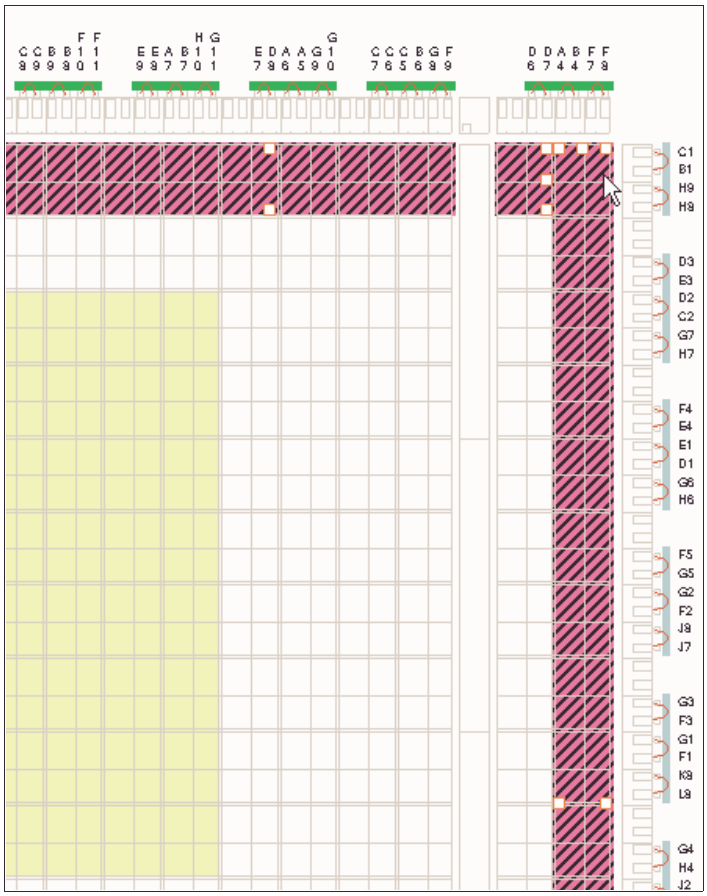

Figure 16: **Non-Rectangular Area Constraint**

After area constraints have been created, the next requested feature places critical components of the design in a specific location. The list of critical components includes PowerPC, GTs, Block RAMs, Multipliers, Global Buffers, and DCM/DLLs. When you select and expand the Global Logic folder within the Design Hierarchy view, the DOL reflects the

components in the design. You may drag and drop the components from either the Design Hierarchy or DOL into the Device Architecture view, as shown in Figure 17.

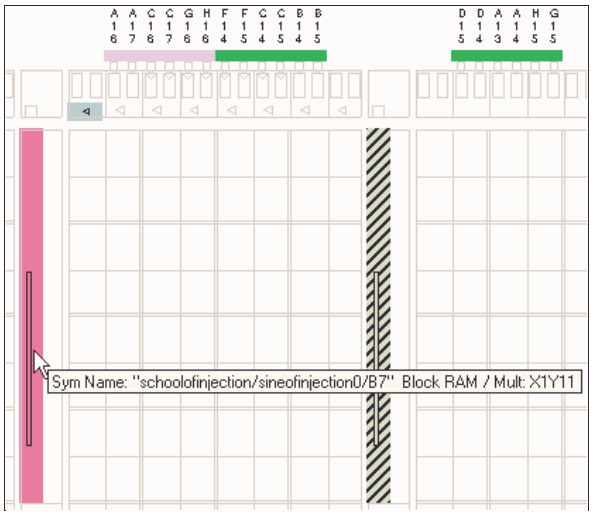

Figure 17: **Placing of Global Logic — Global Buffer and Block RAMs**

Conclusion – Move onto Implementation

Once you have disabled the usage of I/Os, placed I/Os, placed area constraints, and global logic, and saved your constraints to the UCF, your design is ready to go through the implementation process. The implementation process needs to start with NGDBUILD, to incorporate the new UCF file. Then MAP and PAR will use the new constraints from PACE on your design.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
10/28/02	1.0	Initial Xilinx release.