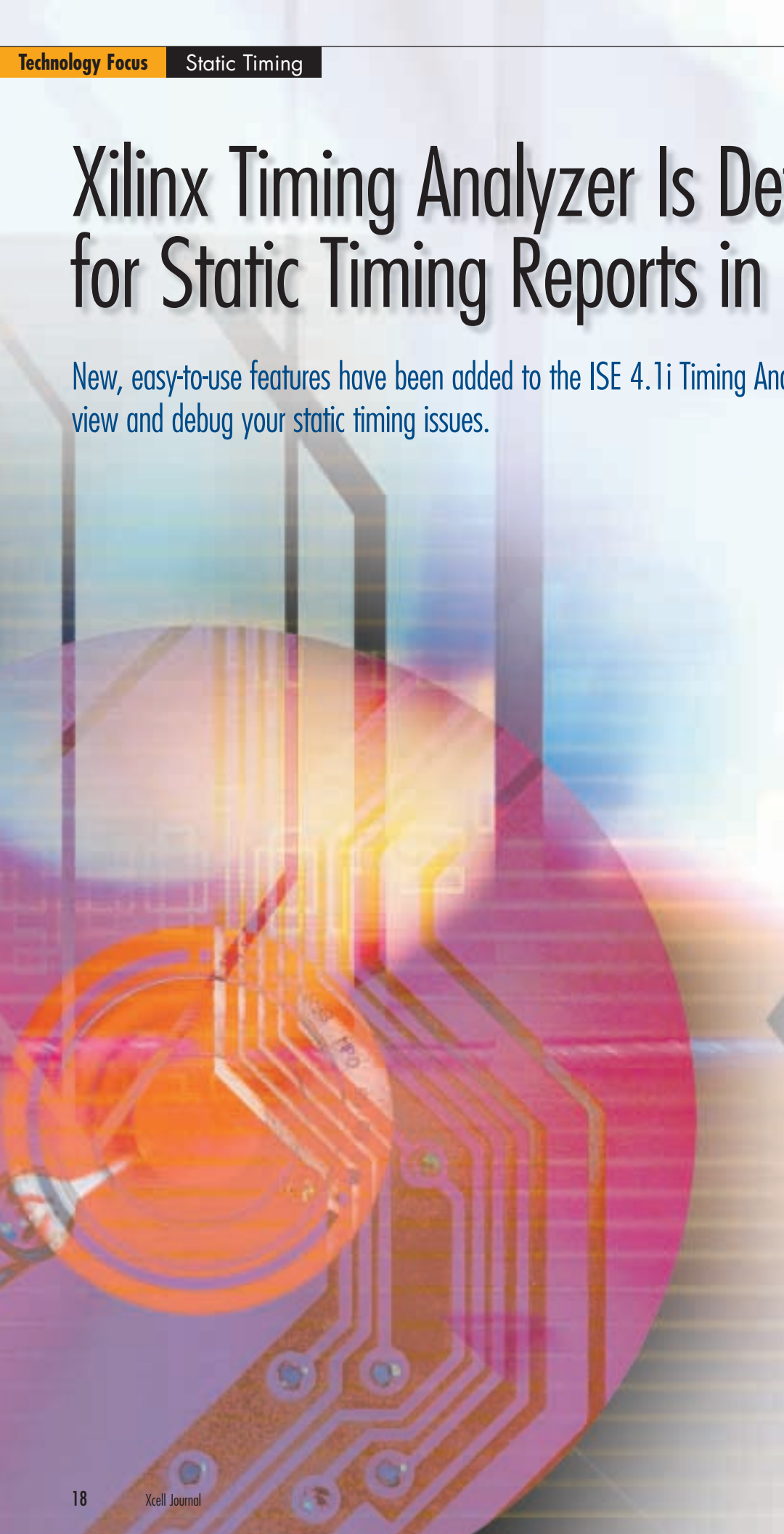
Xilinx Timing Analyzer Is Default Viewer for Static Timing Reports in ISE 4.1i

New, easy-to-use features have been added to the ISE 4.1i Timing Analyzer software to help you view and debug your static timing issues.



by Kate Meilicke
Technical Marketing Engineer
Kate.Meilicke@xilinx.com

Craig Cholvin
Technical Marketing Engineer
Craig.Cholvin@xilinx.com

The Xilinx Timing Analyzer software is a graphic user interface (GUI) for the static timing engine. In the past, Timing Analyzer was perceived as a tool for experts only. With the ISE 4.1i software release, the Timing Analyzer has become the default viewer for static timing reports – and a tool for every designer.

Timing Analyzer delivers reports that are easier to understand because important information is highlighted. Furthermore, Timing Analyzer has new ways to sort the data that help you find the critical paths and get suggestions on how to fix failing paths. Cross-probing timing paths to the Xilinx Floorplanner, Synplicity's Synplify™, and Exemplar Logic's LeonardoSpectrum™ design tools have also been enhanced. These and other new features greatly simplify the resolution of static timing issues.

Report: Just the Facts

In the past, too many customers complained that the ISE 3.1i static timing report had too much information and was too confusing. Responding to that feedback, we slimmed down the timing report to show, by default, only the most important information. Expert users who want to see all the details, however, can enable that information in the Timing Analyzer viewer.

Analyzing the Timing Report

Figure 1 shows the new timing report format with a period constraint that was created during the Translate (ngdbuild) step. TS_clk25 is the original TIMESPEC with a PERIOD constraint of 40 ns. In this case, clk25g has the same timing as the original clock (TS_clk25 / 1.000000), therefore, it also has a period of 40 ns.

The heading for each path has more information. The amount of slack and the equation used to calculate the slack is shown first. Next, you see the source and destination using the logical name (the name in the design). The source clock is clk50g, which is twice as fast as the destination clock, clk25g, in this design. Because clk50g rises at 20 ns and clk25g rises at 40 ns the new requirement is 20 ns (40 ns – 20 ns). Only negative clock skew is used in the equations when calculating setup times.

After the header, a hyperlink highlights the path in the Floorplanner. Following that is a simplified path. The physical CLB (configurable logic blocks) locations are not shown. Only logical names are used.

Looking for Details

The simplified static timing report omits certain details, but all the information is still there. If you're used to working with all the information in the ISE 3.1i report, you can find the details in the .twr file (ASCII report file), and you can turn on the details in Timing Analyzer under File > Preferences. The Timegroup information can be generated in the Timing Analyzer under Analyze -> Query Timegroup.

Improved Period Constraint

In ISE 4.1i, paths between "unrelated" clocks are not covered by a PERIOD constraint. Clocks can be identified as related in the timing constraints language (user constraints file) or through the Constraints Editor. Paths between clocks from a DLL/DCM are automatically related when the Translate (ngdbuild) creates PERIOD constraints for each output of the DCM when the input clock is constrained with a PERIOD constraint. There is a new constraint keyword, "PHASE", that defines the phase relationship between two clocks.

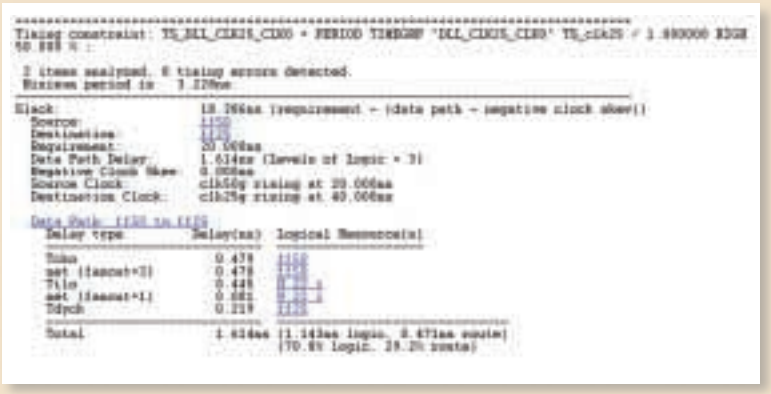


Figure 1 - New 4.1i Timing Report

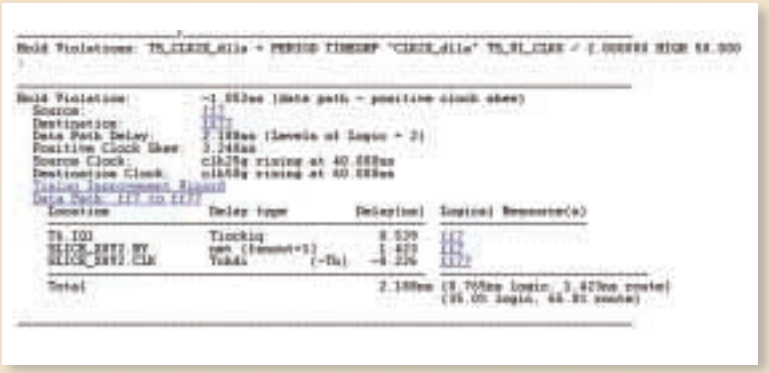


Figure 2 - Hold Violation Report

Related Clock Definition Examples

First clock definition is for the main clock:

```
TIMESPEC "TS_Clock1" = PERIOD
"Clock1" 20 ns HIGH 50 %;
```

The second clock is defined as 1/2 the period of Clock1:

```
TIMESPEC "TS_Clock2" = PERIOD
"Clock2" "TS_Clock1" / 2;
```

The third clock is defined as twice the period of Clock1 and offset by 5 ns:

```
TIMESPEC "TS_Clock3" = PERIOD "Clock3"
"TS_Clock1" * 2 PHASE + 5 ns;
```

Prevent Hold Violations

When the positive clock skew (delay to the destination clock – delay to the source clock) is greater than the data path delay, a "Hold Violation" is reported. The data from the source will no longer be valid at the destination register when the next clock edge arrives. See the example report in Figure 2.

Typically, a hold violation happens when a clock is not on a dedicated clock routing, such as the global clock networks or the low skew lines. The example in Figure 2 was created by routing the clock on local routing resource – which is poor design practice. All clocks must be on a clock resource to prevent hold violations.

Sorting Speeds Up Report Reading

"Slack" is the default sorting algorithm of timing paths in the updated Timing Analyzer. Sorting by slack allows the path with the worst timing to be presented at the top of the report. To accommodate requests for alternate sorting mechanisms, we have implemented an "index" window in the Timing Analyzer where you can select one or all of the timing constraints to sort (or reverse sort) on source name, destination name, or slack for paths. The index is useful when you want to see unique failing paths. A right mouse

ISE 4.1i

click over a time constraint will show the sorting options.

Debugging Static Timing Issues

When a failing path is displayed in the Timing Analyzer, a link to the Timing Improvement Wizard is presented. As shown in Figure 3, the wizard gives suggestions on how to improve the failing path based on information in the path.

Cross Probing

Cross probing is now available between the Timing Analyzer and the editor/viewers of Xilinx, Synplicity, and Exemplar Logic. Simply clicking on the path or element in the Timing Analyzer timing report (.twx) will "hilite" the path in the active editor, as seen in Figure 4 of the Floorplanner.

Tips and Tricks for ISE 4.1i Timing Analyzer

1. If you are looking at the timing report, but also want to find out more information about a specific path, the design must be loaded. This is done under File > Open Design and browse the design. Once the design is loaded, click on Analyze > Against User Specified Paths > by Defining Endpoints. This window will allow you to select specific sources and destinations with which to evaluate paths in a new timing report.
2. If there are no time constraints in your design, a report can be generated to show the longest path for each clock. Again, the design must be opened (File > Open Design), and then you must run Analyze > Against Auto Generated Design Constraints. This will automatically generate a timing report organized by clocks and their associated inputs and outputs.
3. A datasheet report appears at the end of the timing report. The first section of the datasheet shows the setup and hold times for each signal, and it is organized by clocks. The setup and hold times are reported with respect to the pins of the chip – not at the registers themselves. This means the clock and data delays have been incorporated into the numbers. A negative setup typically indicates a long delay on the clock net and usually a hold requirement. This means the data can show up after the clock at the pins. Because there is a shorter delay on the data path, the data will show up at the synchronous element before the clock.

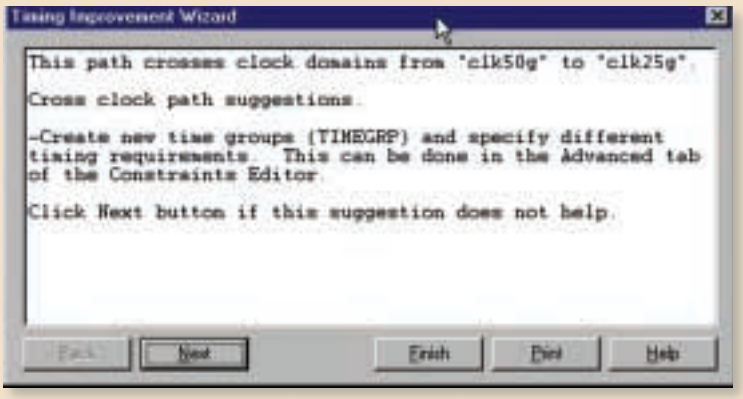


Figure 3 - Timing Improvement Wizard

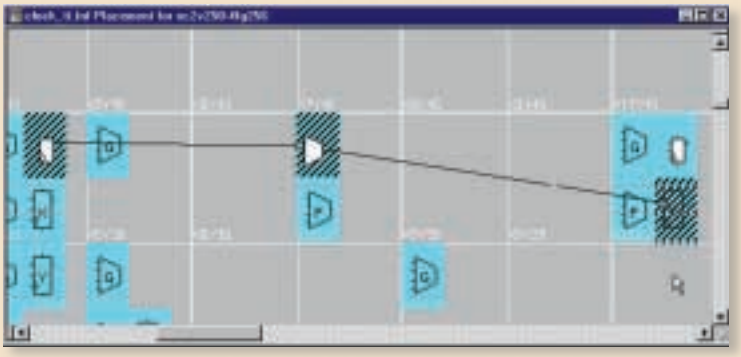


Figure 4 - Cross probing to the Floorplanner

The next section of the datasheet is the Clock to Pad . This shows the Clock to Out of all the output signals organized by clocks.

The last section of the datasheet is the Clock to Setup for each clock. All the source clocks that go to the destination clock appear in the first column. The setup time when both clocks are using the rising edge is shown in the second column. The next column shows when the source clock is falling, but the destination clock is rising and so on. This is a great place to see when data is crossing clock domains and/or half-cycle paths.

Paths are only reported in the datasheet if they are covered by constraints, or if unconstrained paths are included, or if a default or advanced analysis is run.

Conclusion

Whether you're a novice or an expert, the ISE 4.1i Timing Analyzer is very useful. Although several new functions, such as the Timing Improvement Wizard, are geared towards the new Xilinx user, we've kept all of the advanced features expert users like so much.

For more product training, please visit *www.xilinx.com/support/education-home.htm.* For more product information, please see the Help Topics in the Timing Analyzer tool.