Designer Migration from Timer to SmartTime

Introduction

Actel has introduced the SmartTime static timing analysis tool with the release of the Libero® Integrated Design Environment (IDE) and Designer v6.2 software tools. SmartTime enables you to run static timing analysis for the following families: SX-A/RTSX-S, eX, ProASIC®, ProASIC PLUS®, ProASIC3/E, and Axcelerator®/RTAX-S. In Libero IDE / Designer v6.1, Timer was the static timing analysis tool used for these families; however, Timer no longer works with these families. This application note describes the differences between Timer and SmartTime and migration from Timer to SmartTime.

The differences between Timer and SmartTime can be grouped into five categories:

- Timer vs. SmartTime GUI
  - External setup/hold (SmartTime) versus in-to-reg (Timer)
  - Clock to Output (SmartTime) versus reg-to-out (Timer)
  - Filters / User Sets (SmartTime) and set-of-paths (SmartTime, Timer)
- Timing analysis features
  - Explicit clocks (SmartTime) versus potential clocks (SmartTime, Timer)
  - Maximum delay constraint priority (SmartTime, Timer)
  - Maximum delay constraint on registers (SmartTime, Timer)
  - Inter-clocks domain (SmartTime, Timer)
  - Constraint conversion from Timer GUI to SmartTime
- SDC support
- DCF support
- TCL command

Timer vs. SmartTime GUI

Timer uses the same window for applying constraints and for doing timing analysis. The windows are arranged in tabs. SmartTime, however, provides multiple views for editing timing constraints and performing minimum or maximum timing analyses. In the SmartTime Constraints Editor under SmartTime, you can edit your timing requirements and timing exceptions using easy-to-use visual dialogs. The Timing Analysis View enables you to browse through the design's various clock domains to examine the timing paths and identify any violation of the timing requirements. Figure 1 and Figure 2 on page 2 show the default Timer and SmartTime GUI.

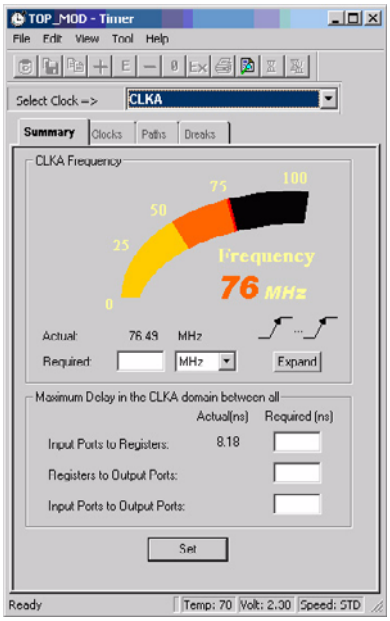

Figure 1 • **Timer GUI**

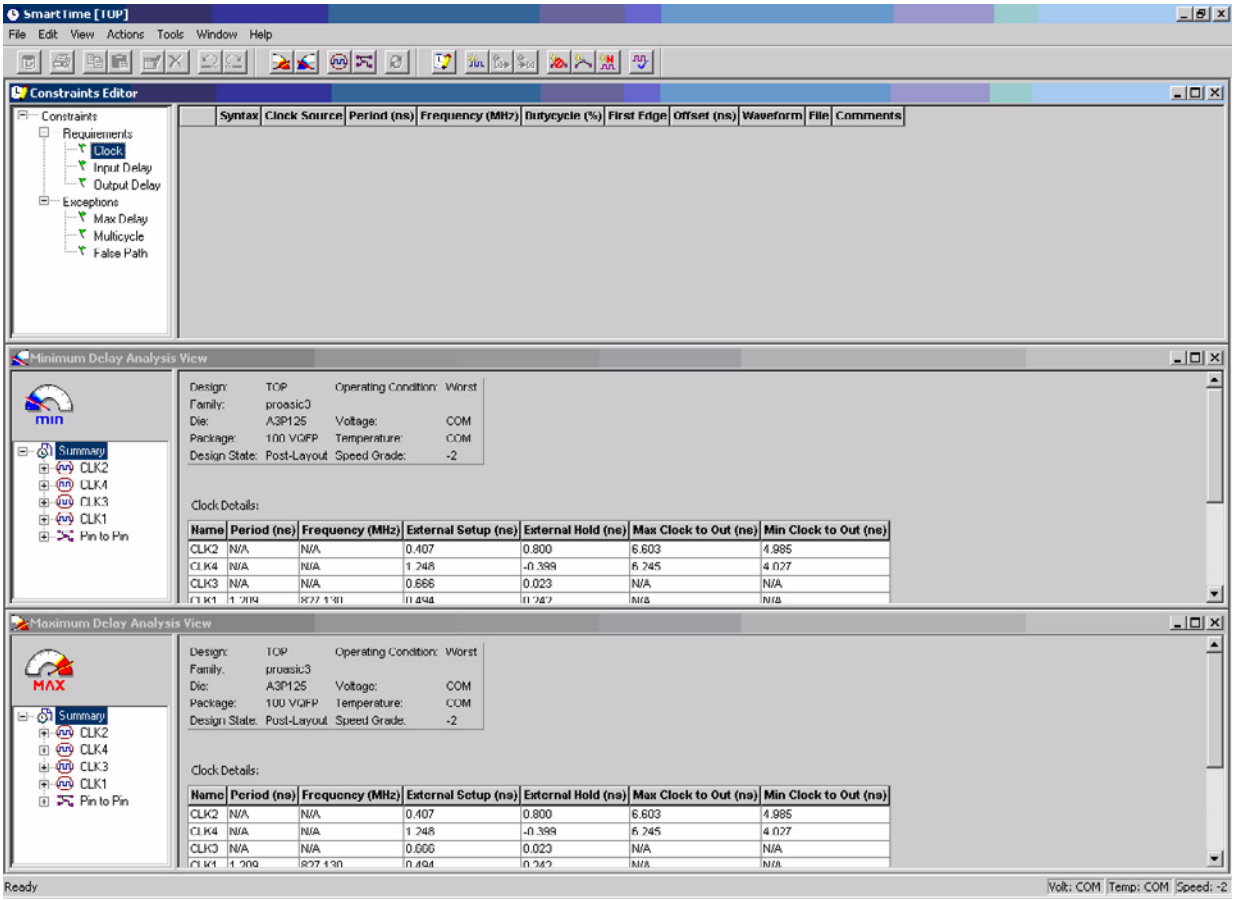

Figure 2 • **SmartTime GUI**

As you can see, the GUI has been completely redesigned in SmartTime. The titles of some of the default paths set in Timer have been changed to make it more meaningful for timing analysis. These GUI changes are described in the following sections.

External Setup/Hold (SmartTime) Versus Input to Register (Timer)

Timer shows input to register as a default path set. This includes the delay from the input pad to the input pin of the register. It does not take setup and clock insertion delay into account. Also, these paths are sorted by the input to register delay and not by the external setup/hold value. This is true for maximum and minimum delay analysis. Figure 3 shows the input to register path set.

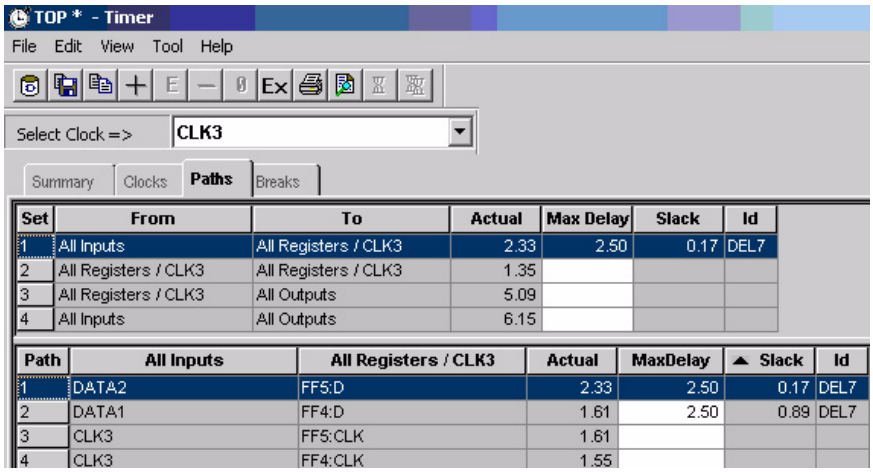

Figure 3 • **Inputs to Registers Path Set in Timer**

SmartTime uses the external setup/hold domain browser to show paths in the same predefined set. The external setup uses the delay from input to register, clock insertion delay, and register setup time. EQ 1 shows the formula used to calculate the external setup.

$$\text{External setup} = \text{input to register} + \text{setup} - \text{clock insertion delay} \quad \textit{EQ 1}$$

The same formula applies to external hold. The paths are arranged using the external setup/hold delay value. The paths may be sorted differently in SmartTime than in Timer. Figure 4 shows the external setup domain browser in SmartTime.

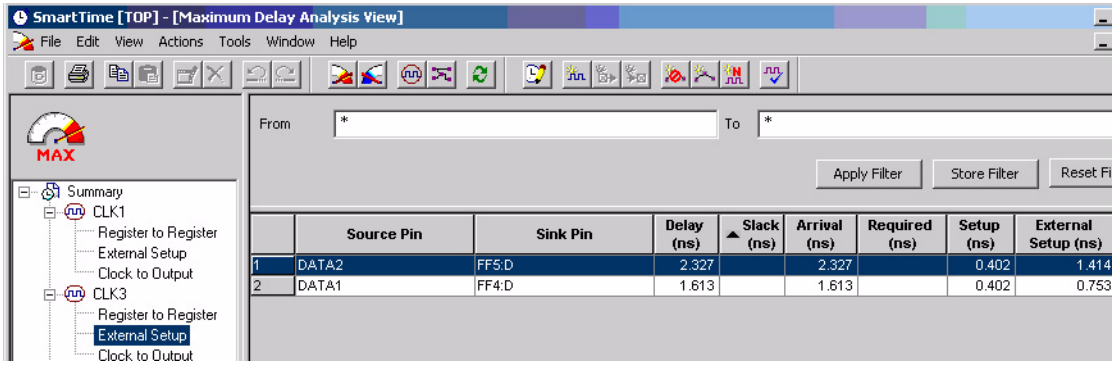

Figure 4 • **External Setup Domain Browser in SmartTime**

Clock to Output (SmartTime) Versus Register to Output (Timer)

The register to output set is shown as a default path set in Timer. This includes the delay from the clock pin of the register to the output port. Figure 5 shows the register to output path set. These paths are sorted by the actual delay, not taking into account the clock insertion delay from the clock port to the clock pin of the register.

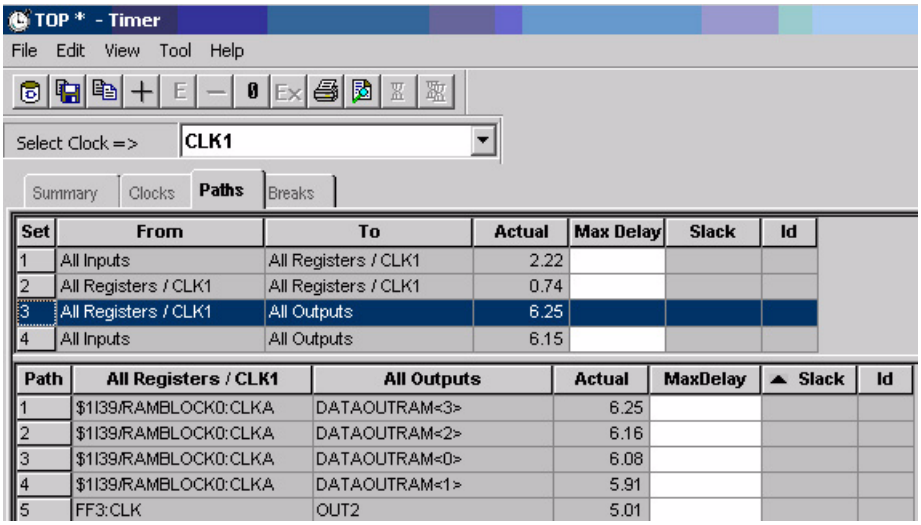

Figure 5 • **Registers to Outputs Path Set in Timer**

SmartTime shows the same paths under Clock to Output in the Domain Browser (Figure 6). This includes the delay from the clock source through the clk pin of the register to the output port. So, the paths may be sorted differently in SmartTime than Timer.

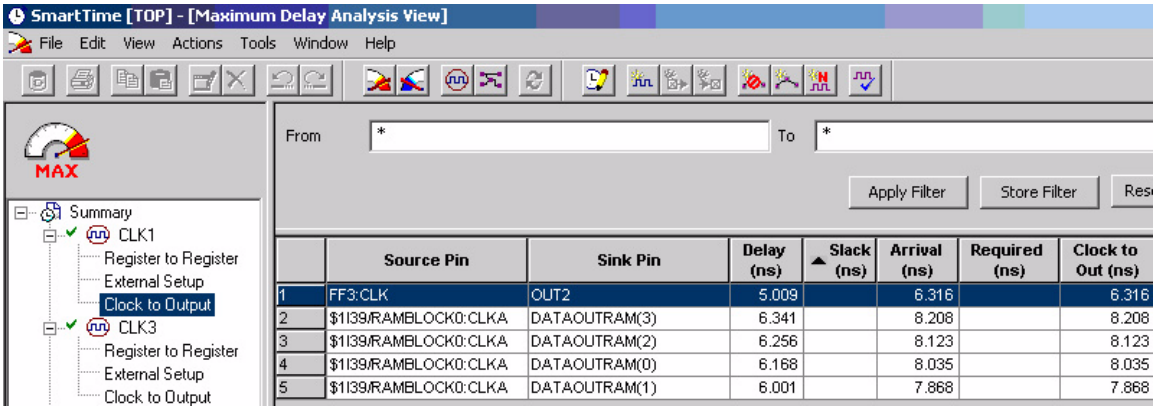

Figure 6 • **Clock to Output Domain Browser in SmartTime**

Filters / User Sets (SmartTime) and Add Set (SmartTime, Timer)

In Timer, you can add custom path sets using the Add Set dialog box. You can add the path set in SmartTime using the Add Path Analysis Set dialog box. If you already have a custom path set in Timer, it will be added under User Sets in SmartTime as shown in Figure 7.

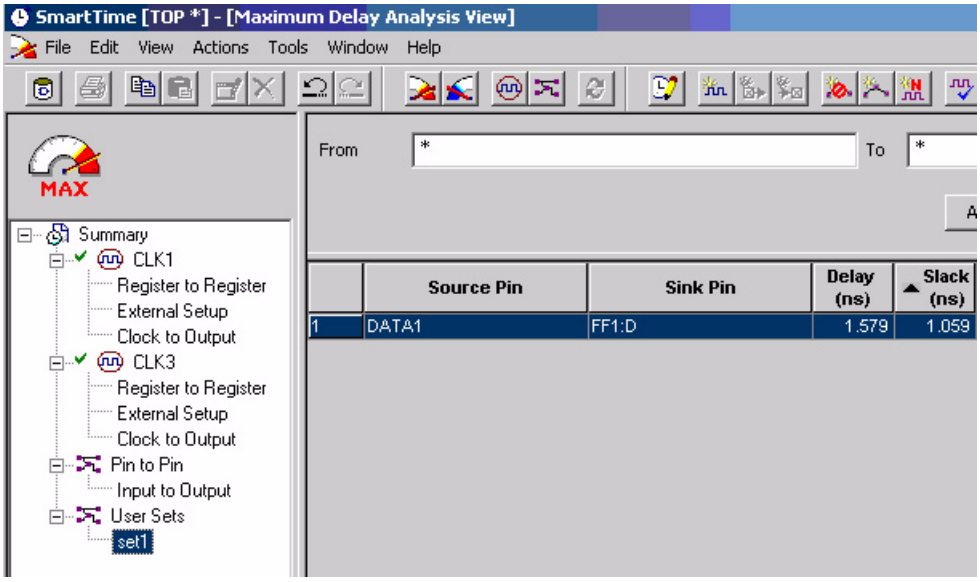

Figure 7 • **Added Path Set in SmartTime**

SmartTime has a new feature called Filters that represents an efficient way to add User Sets. You can create a filtered list on the source and sink pin names. You can save these filters one level below the set under which they have been created.

Timing Analysis Features

SmartTime includes a combination of advanced analysis features and compatibility with industry-standard formats and workflows. There are some differences in the Timing analysis feature between SmartTime and Timer. They are explained below.

Explicit Clocks (SmartTime) Versus Potential Clocks (SmartTime, Timer)

In SmartTime, timing paths are organized by clock domains. Timer shows, by default, all clocks in the Domain Browser. SmartTime only shows the explicit clocks by default. Explicit clocks are pins or ports connected to the clock pin of one or more sequential components, where each clock is one of the following:

- The output of a PLL
- An input port that is not gated between the source and the clock pins it drives
- The output pin of a sequential element that is not gated between the source and the clock pins it drives
- Any pin or port on which a clock constraint was specified

Note that you can add other clock domains if needed, as shown in Figure 8. For details on how to add a clock domain, refer to the SmartTime online help.

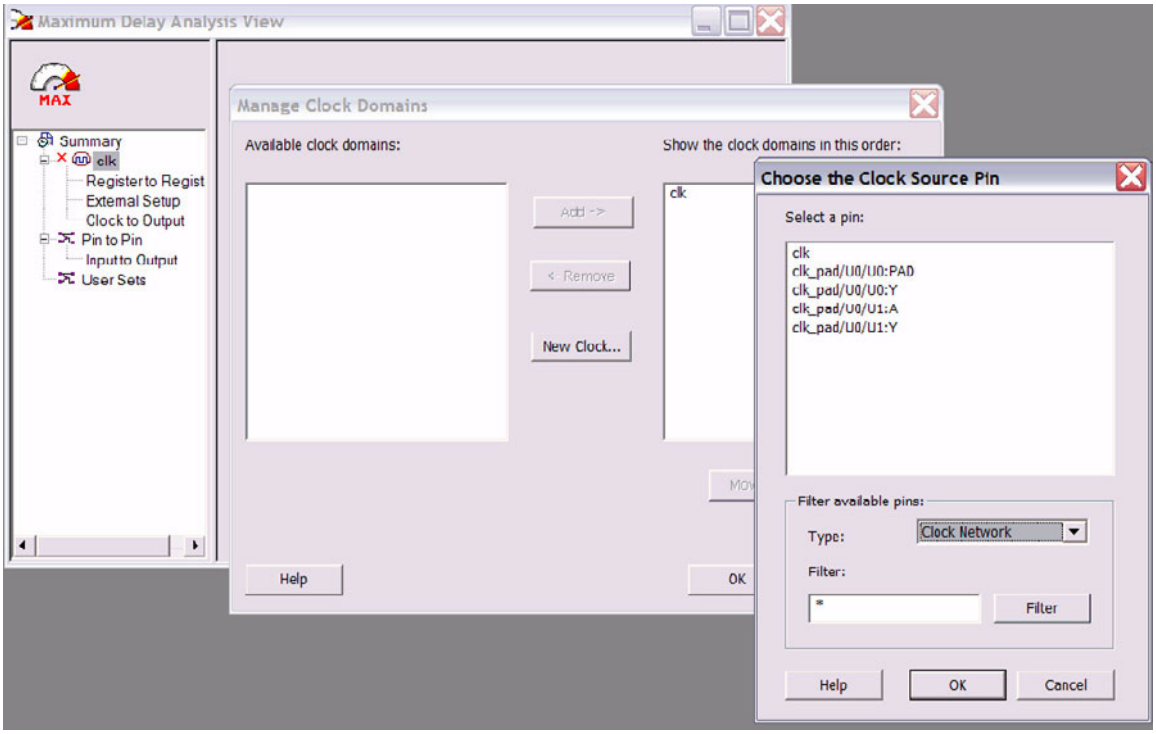

Figure 8 • **Adding a Clock Domain in SmartTime**

Maximum Delay Constraint Priority (Timer vs. SmartTime)

In the presence of a conflict between a clock constraint and a maximum delay constraint, Timer uses the tightest requirement between the two. In SmartTime, a maximum delay is seen as an exception to the clock constraint and will systematically overwrite the latter. Therefore, a maximum delay constraint always has a higher priority than the clock constraint. The slack report may be different in SmartTime than in Timer when opening an existing design with both maximum delay and clock constraints.

Maximum Delay Constraint on Registers (Timer vs. SmartTime)

Timer uses the maximum delay constraint as a timing budget from the source pin to the sink pin of the constraint. However, in SmartTime the same maximum delay constraint on a path that involves registers will automatically take into account the setup and the clock skew during calculation for the timing budget. Therefore, in SmartTime a maximum delay constraint between all registers is equivalent to a clock constraint and a maximum delay constraint between the input and a register is equivalent to an external setup constraint. This may cause SmartTime to behave differently than Timer.

Inter-Clock Domain (Timer vs. SmartTime)

Timer does not support inter-clock domain analysis. You can add max delay on registers between two clock domains to analyze those paths. SmartTime allows the inter-clock domain analysis by setting a specific option. Actel recommends that you remove these Max Delay constraints emulating the inter-clock domain and use the built-in inter-clock domain, if activated in SmartTime. Choose **Options** from the **Tools** menu and check **Include inter-clock domains in calculations for timing analysis**. The inter-clock domain in SmartTime, if activated, will use the clock constraints for analysis (Figure 9).

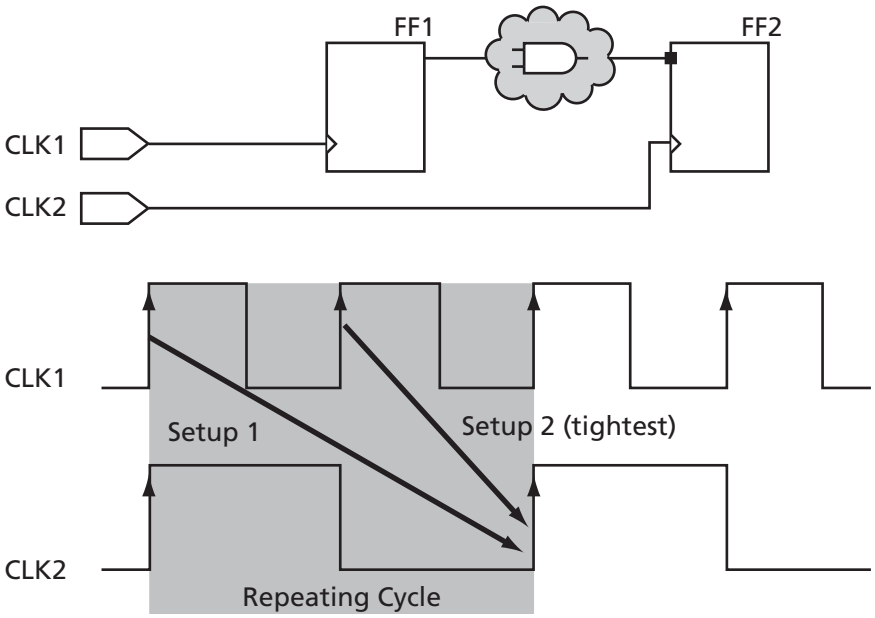

Figure 9 • **Inter-Clock Domain Analysis**

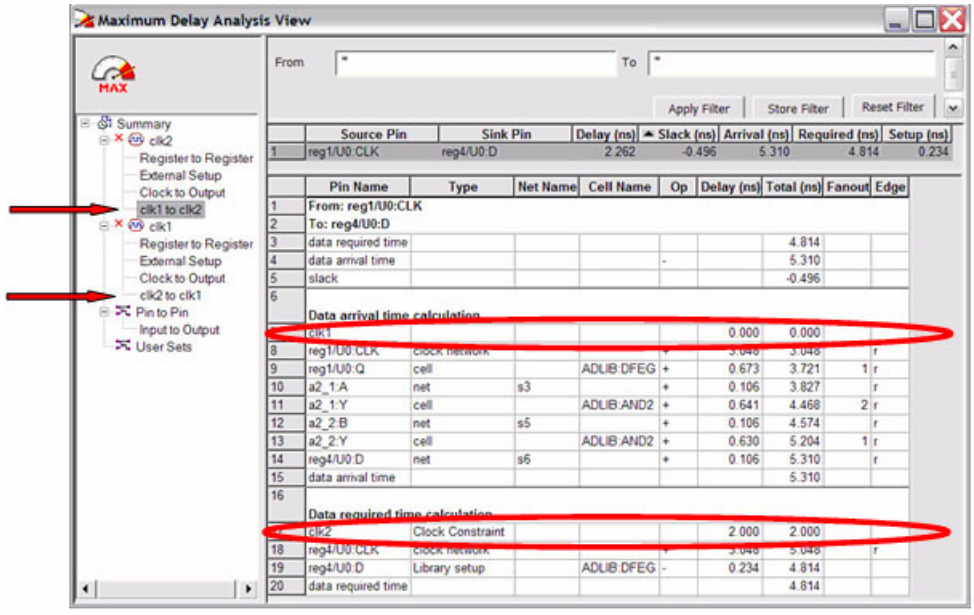

Figure 10 • **Inter-Clock Domain Analysis Using SmartTime**

Constraint Conversion from Timer GUI to SmartTime

You can apply constraints using the Timer GUI. When you open an existing design in SmartTime, the constraint already applied using Timer will be converted. The following sub-sections will describe how some of the constraints applied to Timer will be converted in SmartTime.

Summary Tab

If you apply a clock constraint in the Summary tab of Timer, it will be transferred to a clock constraint in SmartTime with the default duty cycle and offset. You can view this constraint in the SmartTime Constraints Editor and modify the constraint if needed. The Input Ports to Registers, Register to Output Ports, and the Input Ports to Output Ports constraint applied in the Summary tab of Timer will be transferred to the Max Delay constraint. Figure 11 shows a 2.50 ns constraint applied to Input Ports to Registers for the CLK3 domain, and Figure 12 shows the constraint will be converted to the equivalent set_max_delay SDC constraint in SmartTime.

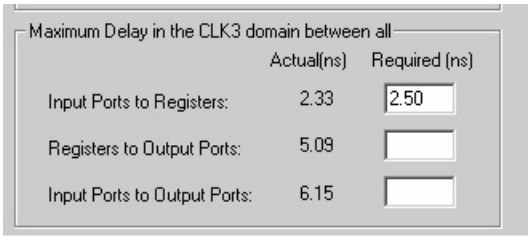

Figure 11 • **Constraints in Timer Constraint Editor**

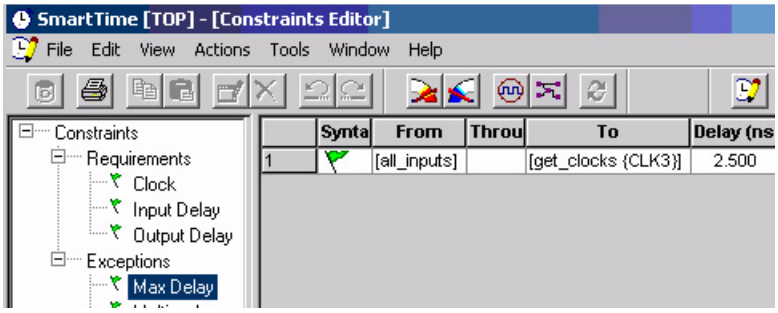

Figure 12 • **Conversion of Max Delay Constraint**

Clock Exception

In Timer, clock exceptions are terminals in a synchronous network that should be excluded from the specified clock analysis (Figure 13). If you apply a clock exception for a pin using Timer during the conversion, the SmartTime Timing Analyzer translates this constraint to the set_multicycle_path SDC command with a path multiplier of 100 (Figure 14).

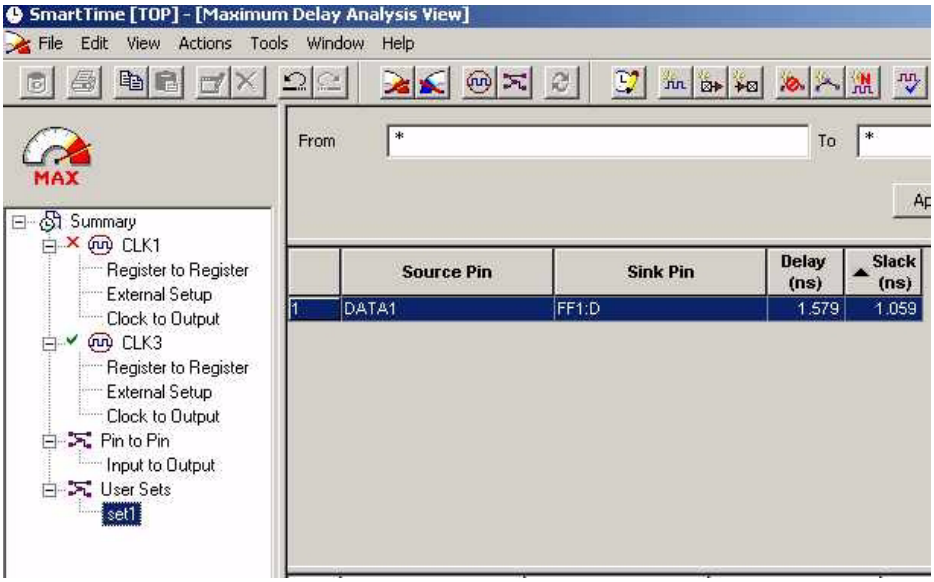

Figure 13 • **Clock Exception in Timer**

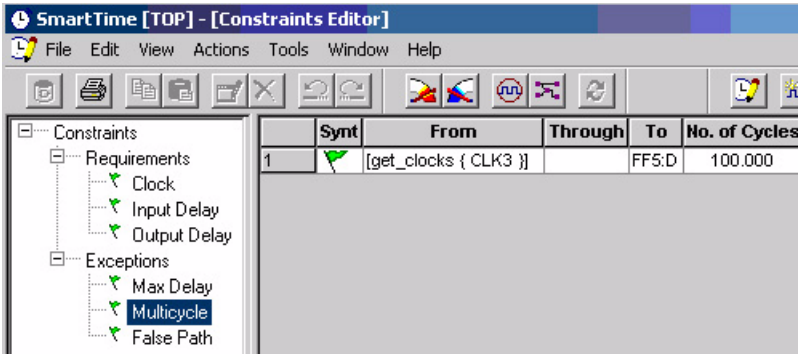

Figure 14 • **Clock Exception Converted as Multicycle in SmartTime**

Global Stops

In Timer, a global stop is a defined intermediate point in a network that forces all paths through the defined point to be "don't care" paths regardless of any constraint assignment. The SmartTime Timing Analyzer tool translates this constraint to the set_false_path SDC constraint.

Global Pass

In Timer, setting a pass pin on a module pin enables you to see a path through individual pins that do not normally allow a view. This was used to compute Clock to Output on the design, starting from the clock port to an output port. The SmartTime Timing Analyzer tool ignores this constraint because it is not needed. You can obtain the same information directly from the predefined set Clock to Output.

SDC

SmartTime supports all the SDC constraints supported by Timer. In addition, it supports some new SDC constraints. Refer to the SmartTime online help or Libero IDE online help for details.

DCF

SmartTime supports the DCF file format for the antifuse families SX-A, eX, and RTSX-S/RTSX-SU. Even though DCF is supported in SX-A, eX, and RTSX-S/RTSX-SU, Actel recommends that you use SDC in SmartTime for all your timing constraints.

Tcl

SmartTime supports most of the clock constraints supported by Timer. Refer to the SmartTime online help for more details. Actel strongly recommends the use of SDC commands to apply timing constraints wherever applicable.

TDPR and Timing Constraints for SX-A, eX, and RTSX-S/RTSX-SU

SmartTime will support all documented constraints for analysis. Like Timer, only clock, maximum delay, and false paths are supported by TDPR for SX-A, eX, and RTSX-S/RTSX-SU. For better interaction with TDPR, you may need to convert some of the SDC constraints not supported by TDPR to max delay constraints. You can create a set_input_delay SDC constraint for analyzing the external setup. However, this constraint will not be passed to Layout and you may not achieve the desired result. To work around the problem, you can create an equivalent Max Delay constraint, and Layout will honor that constraint.

Conclusion

SmartTime is a gate-level static timing analysis tool with powerful capabilities, fully integrated within the Actel software solution. SmartTime is an easy-to-use tool, which includes a combination of advanced analysis features and compatibility with industry-standard formats and workflows. SmartTime is very useful, very reliable, and provides a larger number of work-saving features compared to Timer.

www.actel.com

Actel Corporation

2061 Stierlin Court
Mountain View, CA
94043-4655 USA
Phone 650.318.4200
Fax 650.318.4600

Actel Europe Ltd.

Dunlop House, Riverside Way
Camberley, Surrey GU15 3YL
United Kingdom
Phone +44 (0) 1276 401 450
Fax +44 (0) 1276 401 490

Actel Japan
www.jp.actel.com

EXOS Ebisu Bldg. 4F
1-24-14 Ebisu Shibuya-ku
Tokyo 150 Japan
Phone +81.03.3445.7671
Fax +81.03.3445.7668

Actel Hong Kong
www.actel.com.cn

Suite 2114, Two Pacific Place
88 Queensway, Admiralty
Hong Kong
Phone +852 2185 6460
Fax +852 2185 6488

51900098-0/7.05