Timer User's Guide

Actel Corporation,

955 E. Arques Ave., Sunnyvale, CA 94086

Printed in the United States of America

Part Number: 5029123-5

Release: July 2003

Table Of Contents

Table Of Contents

Timer

Welcome to Timer

Timer is Actel's static timing analysis tool. Timing analysis is a convenient and thorough method of analyzing, debugging and validating the timing performance of a design. This is achieved by breaking down the design into sets of paths. Delays for each path are then calculated and every path is checked for timing violations.

You can only use Timer after you open a compiled design (*.adb file), or after compiling a netlist in designer. If you invoke Timer before compiling your netlist, Designer guides you through the compile.

There are three ways to start Timer:

1. Choose Timer from the Tools menu, or
2. Click the Timer icon in Designer's toolbar, or
3. Click the Timer button in Designer's design flow.

Timer Interface

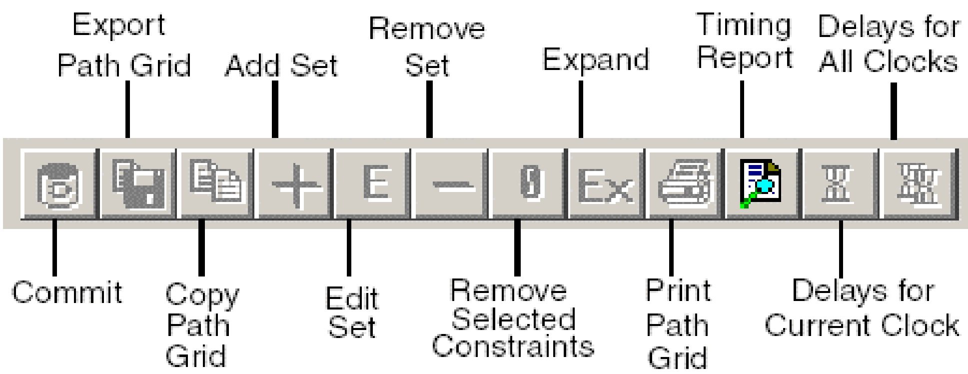
Timer user interface

Timer's four tab screens organize and display static timing information according to the timing analysis preferences you set in the **Preferences** dialog box.

Timer consists of four tab screens: **Summary**, **Clocks**, **Paths**, and **Breaks** (Timer does not display the Clocks tab screen if the device you are using has no clock).

Timer Toolbar

The Timer toolbar contains commands for performing common Timer operations on your designs. Tool tips are available for each button.



Timer Toolbar

Status Bar

Timer's status bar displays information on menu commands, error messages, your selected temperature, voltage, and speed grade. In addition, Timer displays the following:

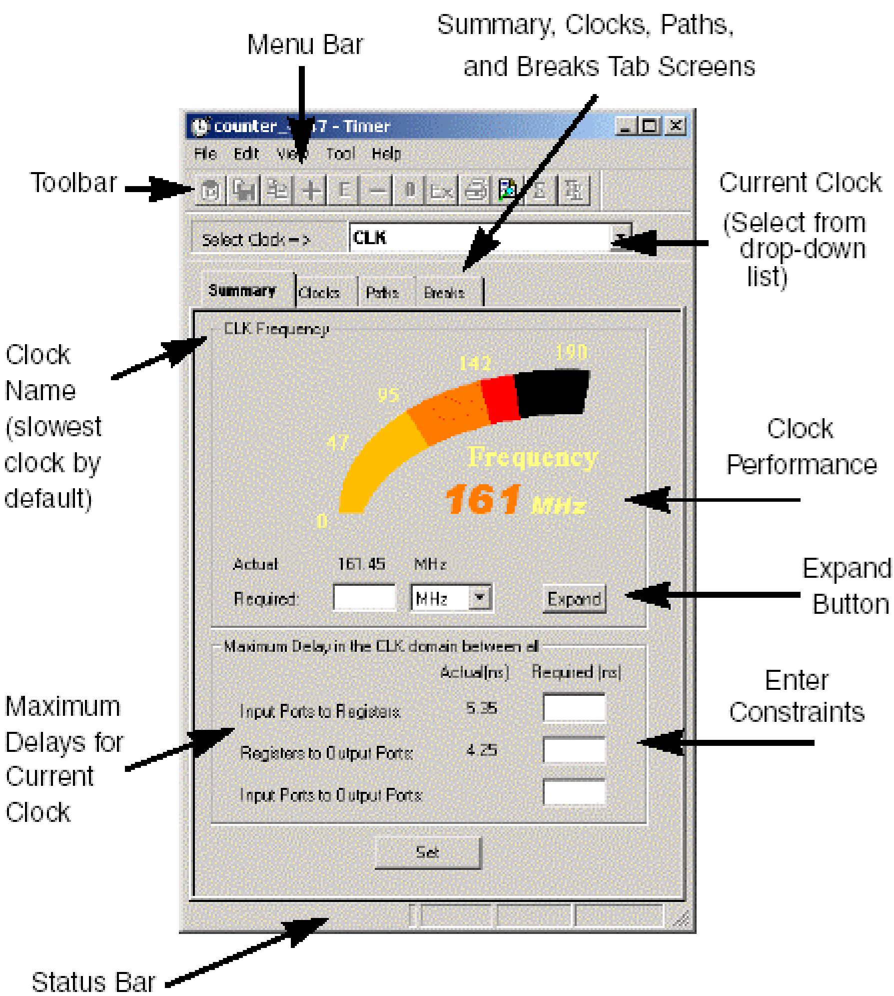
Temp: Displays the temperature consistent with the operating conditions selected.

Volt: Displays the voltage consistent with the operating conditions selected.

Speed Grade: The speed grade of the selected device.

Summary tab

By default, Timer's Summary tab screen displays the maximum frequency for the current clock selected in the Select Clock drop-down list box. If you have multiple clock groups or gated clocks, please add timing constraints in Timer to get the correct frequency of your design.



Timer Summary Tab Screen - PC Only

To change the default clock, select one from the Select Clock list.

Click the **Expand** button Timer to display the details of the path that determined the maximum clock frequency in the Expanded Path window.

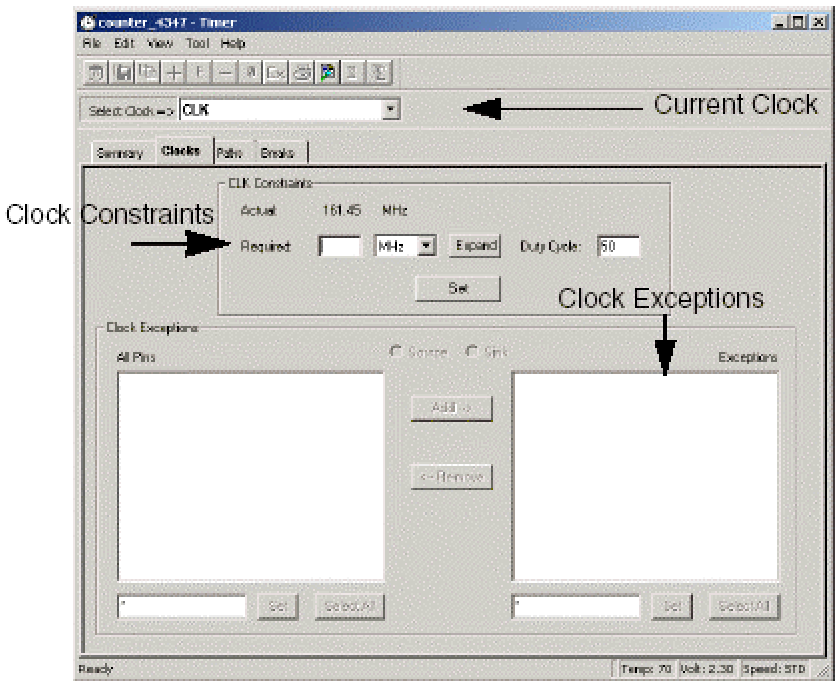
The Summary tab displays the actual longest/shortest delay between all Input ports to registers, Registers to Output Ports, and Input Ports to Output Ports.

Enter your new delay values in the Required input boxes and click Set to recalculate your delays.

IMPORTANT:By adding constraints to the set of paths listed in this tab, be careful that you do not over-constrain the design. This may degrade the quality of the Timing Driven Layout and increase the overall run time.

Clocks tab

The Clocks tab allows you to enter constraint information and set clock exceptions. Select the default clock from the Select Clocks list.



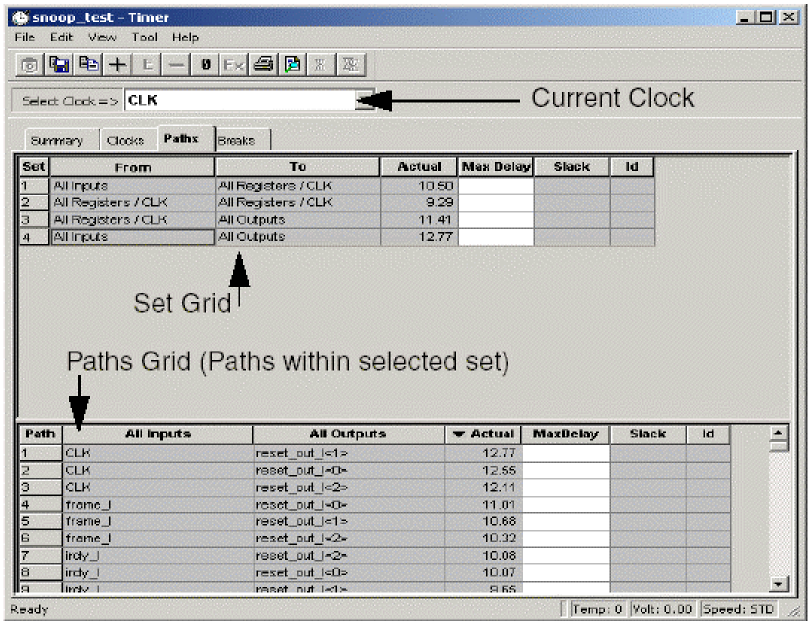
Clocks Tab

Enter constraint information in the **Constraints** area and click **Set**.

Clock exceptions are terminals in a synchronous network that should be excluded from the specified clock analysis.

Paths tab

The Paths tab displays timing analysis information for categories of paths, known as “sets,” and the paths within each set. The Paths tab displays the sets in the set spreadsheet (at top) and the paths within each set (at bottom).



Paths Tab

The Paths tab default setting displays four path sets:

From All Inputs TO All Registers / CLK

All paths from the input ports of the design to the input pins of all the registers in the current clock domain. In this instance, CLK is an example of the current clock domain.

From All Registers / CLK TO All Registers / CLK

All paths from the clock, clear, and pre pins of registers in the current clock domain to the input pins of all the registers in the current clock domain; in this instance, CLK is an example of the current clock domain. To view the register setup and clock skew, right-click the desired path in the paths grid and select Expand Path from the shortcut menu, or click the Expand Path button.

From All Registers / CLK TO All Outputs

All paths from the clock pin of registers in the current clock domain to the primary outputs of the design.

All Inputs TO All Outputs

All input ports to all output ports in the design. This set is completely asynchronous (independent of the clock).

All the sets default to display the longest path in the category. You can change this default setting by selecting Preferences from the File menu. When you select a set, Timer displays the paths within the set in the lower spreadsheet labeled “Paths.” The spreadsheet displays a sorted list of paths (the number of paths it displays is controlled in the Preferences dialog box). Double-click the column headings to sort the columns.

Note: The run-time required to compute the content of the spreadsheet is a function of the number of paths you wish to display. Select Preferences from the File menu to change the default settings.

The timing information displayed for sets and paths includes:

Actual: The actual delay calculated by Timer for each path.

Slack: The difference between the maximum required delay and the actual delay.

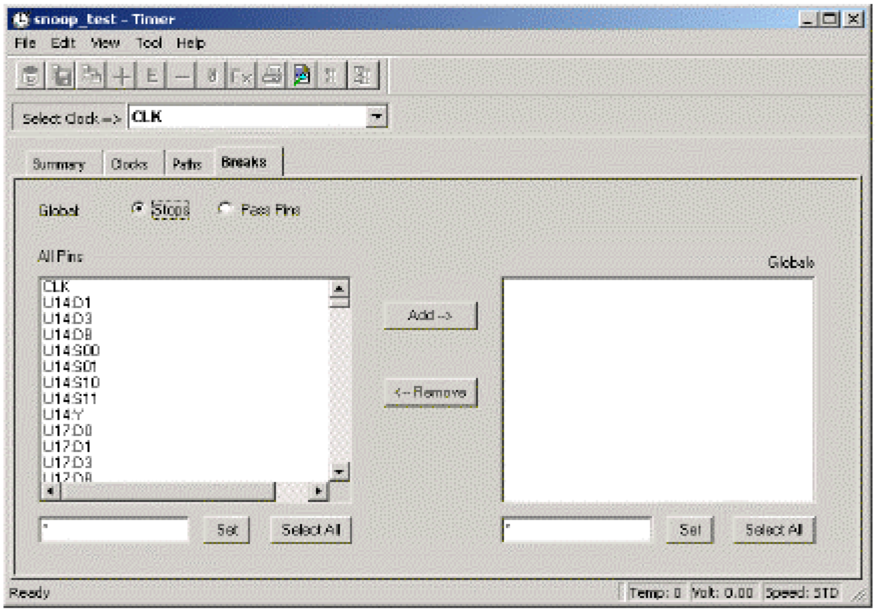
Max Delay: The maximum required delay specified. Do not interpret this value as the clock frequency. To set clock frequency, input on the Summary tab, or on the Clocks tab.

ID: The constraint ID for the path.

Breaks tab

Use the Breaks tab to enter global stops and pass pins. 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. Setting a pass pin on a module pin enables you to see a path through individual pins, which you are not normally allowed to view a path through.

Note: If you are not careful when you set a pass pin on a module pin, you may set a false path in your design.



Breaks Tab

Calculating Delays

Delays, PLLs, RAMs, and FIFOs

Actel Timer uses two different timing models, "pin-to-pin" and "input-to-input". The first type uses a “pin-to-pin" timing model, because Timer reports a pin-to-pin delay. The second type uses an “input-to-input” timing model, because Timer reports the delays from an input gate to the input of the next gate by lumping the gate and net delays together.

ACT1, ACT2, ACT3, DX, MX, and SX devices use the input-to-input timing model, while the 54SX-A, RTSX-S, eX, Axcelerator, and Flash devices use the pin-to-pin timing model. Some timing analysis features are specific to the different timing models; exceptions are noted in the help.

The delay for pin-to-pin devices is reported until the input pins of the registers. Therefore, setup time is not included in the delay. However, the register setup and hold, as well as the clock skew, are taken into account during the analysis of setup check and hold check when identifying timing violations. Setup, hold, and clock skew are also taken into account during clock frequency estimation.

For information on the setup and hold process in Timer, see the Expanded Paths window. It enables you to view clock network insertion delay and clock skew information.

PLLs

The timing tool sees a PLL as a register and a clock generator. Any clock output port in a PLL is a potential clock (and appears in the list of potential clocks for the design). Like all other potential clocks, you can constrain these PLL output clocks by setting any clock constraint independently. The input clock of the PLL on which you set the constraint is not the clock input port of the PLL but the clock driving this clock input port. The driving clock will be a Primary port of the design, a register's output, or another PLL's output.

PLLs for Axcelerator

By default, when you set a clock constraint on the clock source connected to the clock input of the PLL, Timer automatically computes the clock constraints on the outputs of the PLL (according to the PLLs configuration). Thus, the value of the clock output is equivalent to the clock input multiplied, divided, or shifted by the value of your static configuration.

If you specify a clock constraint for the output clock(s), the PLL ignores the static configuration value and delivers a clock frequency according to your constraints. Timer reports this value accurately. In addition, if you remove your constraints on the output clock(s), the Timer tool recalculates your output frequency according to your static configuration value. For more information on generating PLLs and their logic characteristics, please refer to the *ACTgen Macros Reference Guide* in the online help or in .pdf format.

Note: For ProASIC^{PLUS} devices, the PLL is only considered as a register; there is no output clock computation.

RAMs and FIFOs

The Timer tool displays blocks of RAM and FIFO as a single "black box," (you have as many black boxes as you have instantiations of RAMs and FIFOs in your design). Thus, if you construct a RAM or FIFO cell out of several RAM blocks, Timer sees and treats the cell as a single black box. Timer does not display timing information within individual black boxes, because all the delays are reported using the interface of the RAM. Timer displays timing information between black boxes and other logic in the design.

Timer treats RAMs and FIFOs as registers, and like any register, they have clock signals. For more information about RAMs and FIFOs, please refer to the *ACTgen Macros Reference Guide*, in the online help or in .pdf format.

FIFO: Timer displays the paths to the FIFO flags depending on their clock. Timer shows paths to Empty and Almost Empty with respect to the Read clock; paths to Full and Almost Full are displayed with respect to the Write clock.

Using Timer

Determining your clock frequency

Because a design's performance is often measured through the clock frequency, determining the clock frequency is the most common use of static timing analysis.

To obtain a specific clock frequency:

1. Click the **Summary** tab in the **Timer** window.
2. Select a clock from the **Select Clock** list. The selected clock becomes your current clock. The frequency is displayed under the speedometer.

The clocks listed in the pull-down menu are defined as signals which drive the clock or gated input of two or more adjacent registers. For pin-to-pin delay families, one register is enough to have the clock listed as a potential clock. Timer does not support virtual clocks.

In frequency calculations, values for latency is assumed to be 0, the duty cycle is 50%, and the clock edge is rising. (You can set the duty cycle in the Clocks tab.) For pin-to-pin timing model families, Timer takes into account the register setup and the clock skew when estimating the maximum clock frequency. However, the setup value is not included in the actual delay reported between the clock pin of a source register and the data pin of a sink register. For more information on calculating delays, please refer to Calculating Delays.

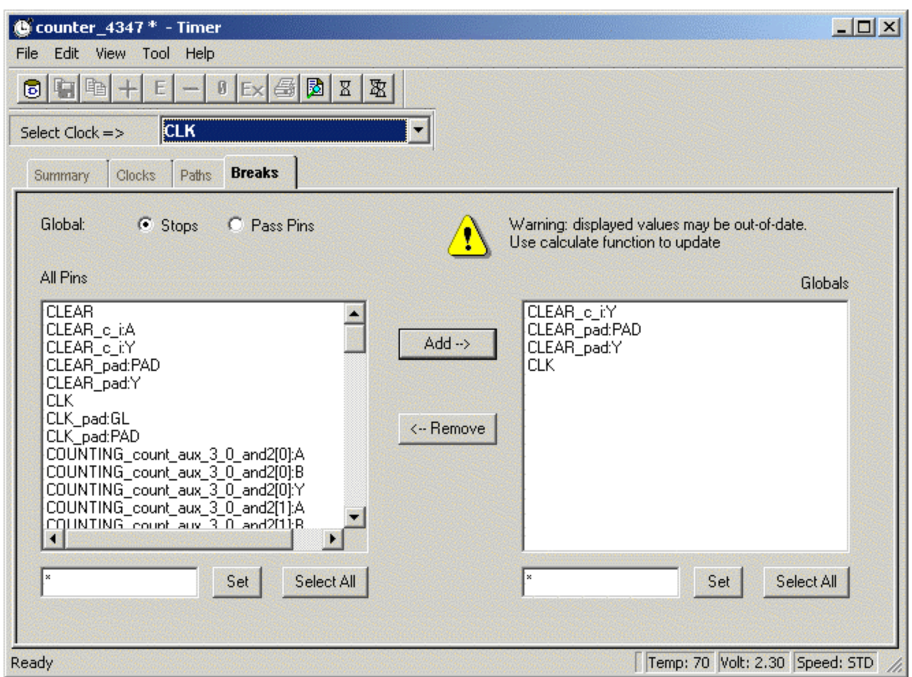
Adding and Removing Break Points

The Timer default behavior is to break paths at clocks. You can change this default behavior in the Timer Preferences dialog box. Without stop points (or break points), Timer reports all the paths from pad to pad in the design. If you do not want to see the paths going through registers clock pins, you could specify these as break points. The path running through those pins is not displayed.

Setting a pass on a module pin enables you to see a path through individual pins. Additionally, you can set a global pass on all Clk/G and Clr/Pre pins in the Preferences dialog box, which is available by choosing Preferences from the File menu.

To add break points:

1. Click the **Breaks** tab.
2. Select **Global Stops** or **Pass Pins**. The **All Pins** list box displays the pins.
3. Select the pin(s). The **All Pins** list box defaults to show all pins. Text boxes are provided below the list boxes to help you limit the list for consideration. Enter a value and click **Set**. The * character is used as a wildcard. To select multiple pins, hold down the CTRL key while selecting with your mouse. Click **Select All** to select all pins displayed in the **All Pins** list box.
4. Click **Add**. The **Stops** or **Pass Pins** will be added to the **Global** list box as break points.



Breaks Tab - Stops Selected

To remove break points:

1. Click the **Breaks** tab.
2. Select **Global Stops** or **Pass Pins**. The break points are displayed in the Global list box.
3. Select break **Points** to remove. To select multiple breaks, hold down the CTRL key while selecting with your mouse. Click **Select All** to select everything displayed in the **Globals** list box.
4. Click **Remove**. The pin(s) will be removed from the **Globals** list box.

Path Analysis

Path Analysis

Timer organizes and displays data based on your timing analysis preferences. Timer assists you in analyzing critical paths, paths with the greatest delay, and by expanding paths so you can trace delays along paths. The Expanded Path window provides delay information for the path that is in greatest violation.

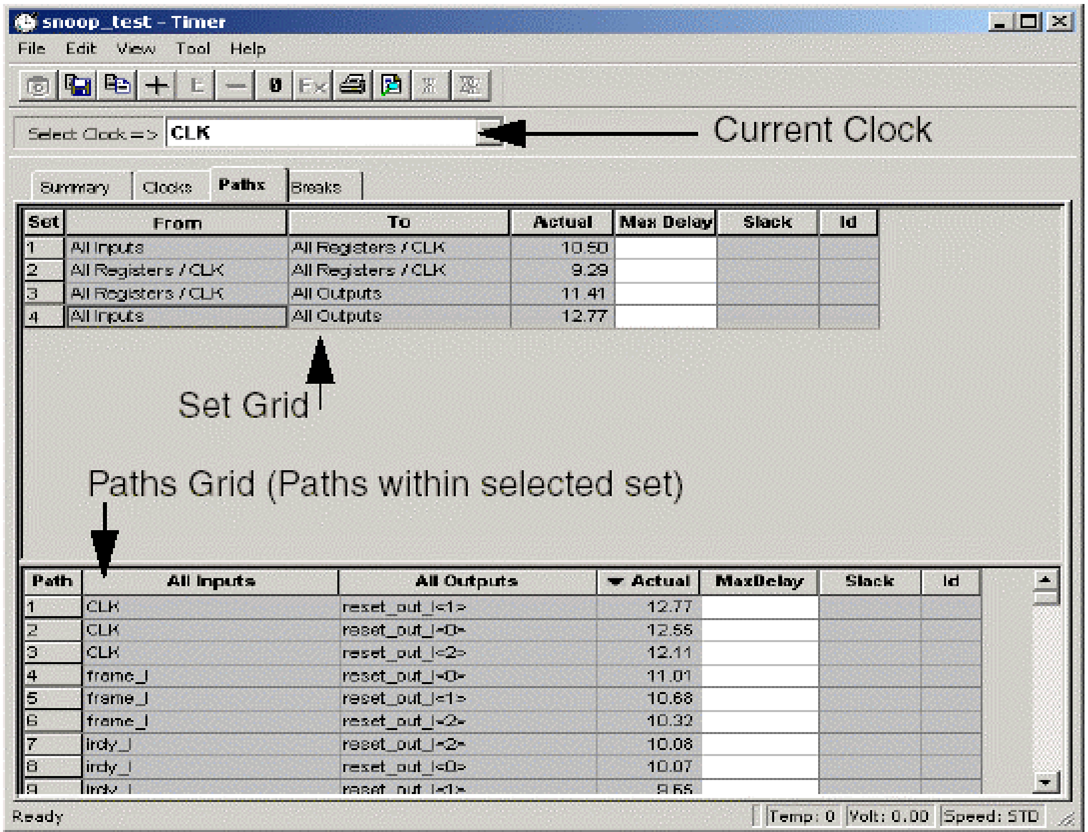
You can change your preferences to control how you display paths, add path sets, and expand paths.

Display paths

Path sets (groups) and paths within each set are displayed on the Paths tab. You can create your own sets and add them to the paths tab (see Adding path sets). Also, Timer displays all previously entered sets that have a constraint in the Set grid.

To display paths:

1. Click the **Paths** tab. By default, Timer displays four path sets in the set grid.
2. Click a set. Timer displays the paths in the path grid.



Timer Paths Tab

When you set a clock constraint for a pin-to-pin timing model family, it is mapped into specific register to register max delay values; these values appear in the max delay of each specific path in the spreadsheet. Timer takes into account register setup and clock skew when computing max delay values for these pin-to-pin model families.

The register-to-register selections are based on the clock domain selected in the Clocks tab. (To select another clock, choose the Select Clock menu from the Toolbar.) See the index for a list of Paths tab information.

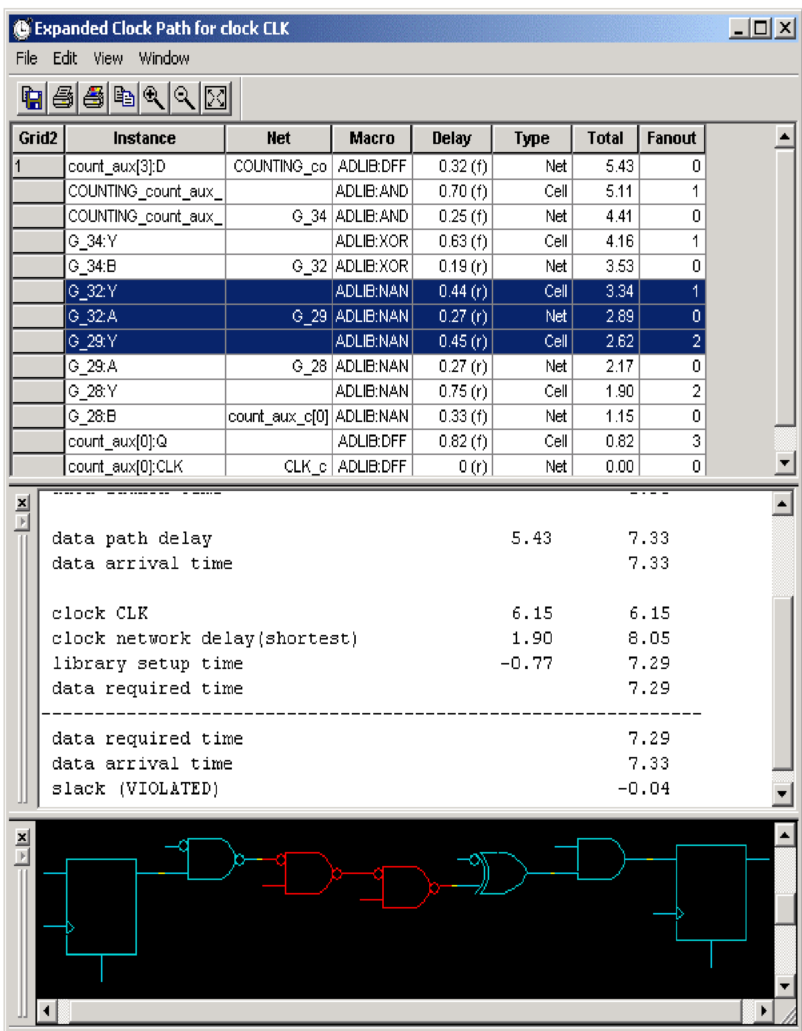
All delays shown are worst-case by default. To change this setting, see Case Analysis in the index.

Expanding paths

Each path has one or more logic macros and nets that contribute to its total delay. By expanding the path, you can view detailed delay information for parallel paths. Note that with the exception of parallel edges, parallel paths are not available for families that use the pin-to-pin timing model.

To expand a path:

1. Click the **Paths** tab.
2. Select a path set. Paths within that set are displayed in the path grid.
3. Select the path you wish to expand.
4. Expand the path by double-clicking the path, right-click and select **Expand Path** from the shortcut menu, or in the **Edit** menu, click **Expand Path**. The **Expanded Path** window opens and displays a single path in the **Expanded Path Grid** and a graphical representation of the paths in the **Schematic Window**.



Timer Expanded Path Window

Clock frequency should be the inverse of reg-reg delay plus setup time. However, this is not true with clock skew.

For pin-to-pin timing model families (54SX-A, eX, Axcelerator, and Flash devices), Timer takes into account the register setup and the clock skew (starting in R1-2003) when estimating the maximum clock frequency. However, the setup value is not included in the actual delay reported between the clock pin of a source register and the data pin of a sink register. The Timer Expanded Path window shows Setup Check / Hold Check information. Set the Show option (in Preferences) to "Longest" to view a detailed analysis of the Setup Check and set the Show option to "Shortest" to view a detailed analysis of the Hold Check. (These analyses include clock insertion delay information.)

The images below show the correlation between the Expanded Path Grid (at top), the Setup Check or Hold Check report (at middle) and the Schematic window (at bottom).

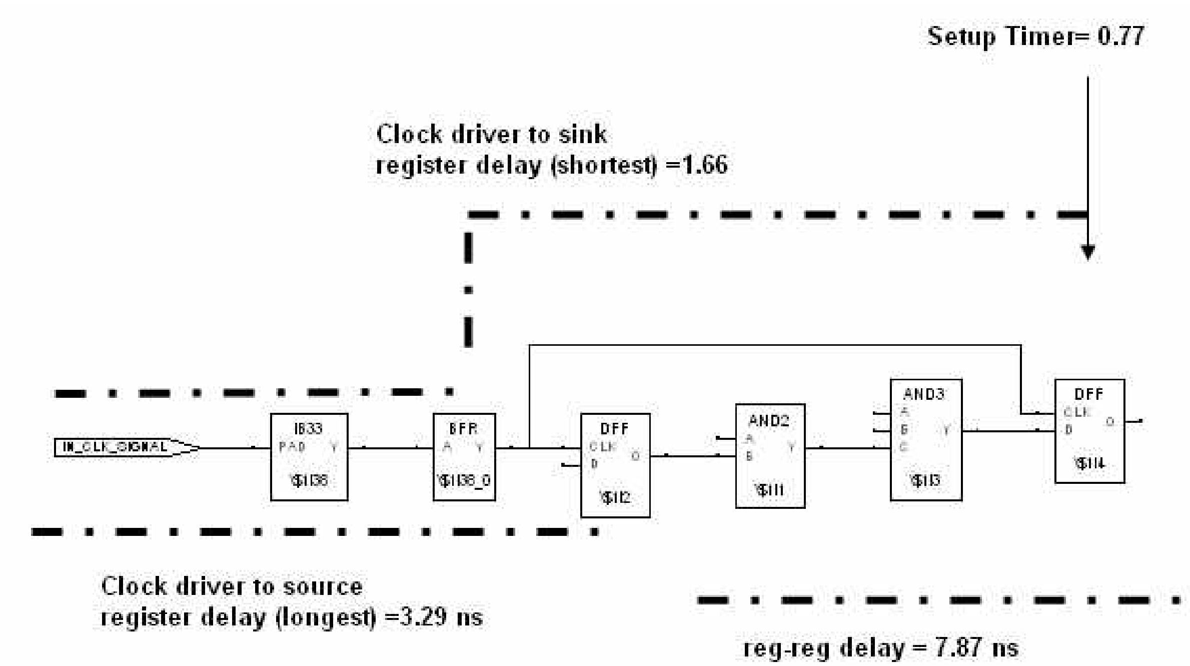
Note that the values displayed in the Setup Check / Hold Check report are affected by rounding in calculations, so you may see a very slight discrepancy in the difference between the data required time and the data arrival time (the slack time).

Grid1	Instance	Net	Macro	Delay	Type	Total	Fanout
1	$1I4:D	$1N9	ADLIB:DFF	1.18 (f)	Net	7.87	0
	$1I3:Y		ADLIB:AND	0.44 (f)	Cell	6.68	1
	$1I3:C	$1N7	ADLIB:AND	2.51 (f)	Net	6.25	0
	$1I1:Y		ADLIB:AND	0.73 (f)	Cell	3.74	1
	$1I1:B	$1N5	ADLIB:AND	2.20 (f)	Net	3.01	0
	$1I2:Q		ADLIB:DFF	0.82 (f)	Cell	0.81	1
	$1I2:CLK	clk1	ADLIB:DFF	0 (r)	Net	0.00	0

```
                              Setup Check
Name                                          Delay(ns)    Total(ns)
------------------------------------------------------------------
clock IN_CLK_SIGNAL                               0.00         0.00
clock network delay(longest)                      3.29         3.29
data launch time                                               3.29

data path delay                                   7.87        11.16
data arrival time                                             11.16

clock IN_CLK_SIGNAL                               8.33         8.33
clock network delay(shortest)                     1.66         9.99
library setup time                               -0.77         9.22
data required time                                             9.22
------------------------------------------------------------------
data required time                                             9.22
data arrival time                                             11.16
slack (VIOLATED)                                              -1.93
```



Slack = data required time - data arrival time = (8.33+1.66-0.77)-(3.29+7.87) = -1.93ns

Clock Skew Analysis

The difference in the arrival times of the clock signals between two sequentially-adjacent registers (clock skew) may cause a design to malfunction with short data paths. The most efficient method to eliminate the short data path problem is to minimize the clock skew by using the low-skew global routing resources for clock signals.

Please refer to the Actel website (http://www.actel.com) for an application note on clock skew analysis.

To measure clock skew:

1. **Specify clock frequency.** In order to obtain hold margin calculations, Timer requires that you specify a clock frequency. Timer uses the frequency to calculate the period, which it needs to evaluate the margin for adjacent flip-flops with alternate clock edges. If you do not enter a clock frequency in the summary tab, you will not get any results.

   To enter a frequency, select the desired clock under "select clock", and enter a frequency under "required" in the Summary tab. Click **Set** when you are done.

2. **Set operating conditions.** To measure clock skew, perform hold time analysis for **BEST** case in **Shortest** path mode. Set the **Case** in **File -> Preferences**.
3. **Run Violations Report.** A report is available from Timer that provides a summary of timing margins for all paths in the design. In Timer, go to **Tool -> Report Violations**. This report lists the following categories:

   ```
   · Section Clock constraints violation
   · Section Max Delay constraints violation
   · Section Min Delay constraints violation
   ```

   To find a summary of hold time margin in your design for the given operating conditions, refer to the timing paths listed under **Section Min Delay constraints violation**.

   The first column defines the slack for each path. Positive values represent margin, negative values represent a violation.

4. **Detailed analysis.** To see the details of a given path, go to the **Paths** tab in Timer. You can look for a specific path by creating a new path set for the specific path(s) you are interested in. To create a path set, go to **Edit -> Add set of paths**. Refer to the Paths tab

for information on how to add a set of paths.

Once you have defined the new path set, click the set to display the path list in the lower spreadsheet. Then highlight the path you are interested in, right click and select **Expand Path**.

5. **Review Expanded Path window.** The expand path window shows details of the calculations performed in the clock skew analysis.

   The margin is calculated by adding the clock propagation delay of the master register to the data path delay between the two registers. This is the data arrival time. Then the clock propagation to the slave register is subtracted from the sum, giving the final slack value. If alternate clock edges are used for adjacent registers, Timer considers the clock period accordingly.

Adding Path Sets

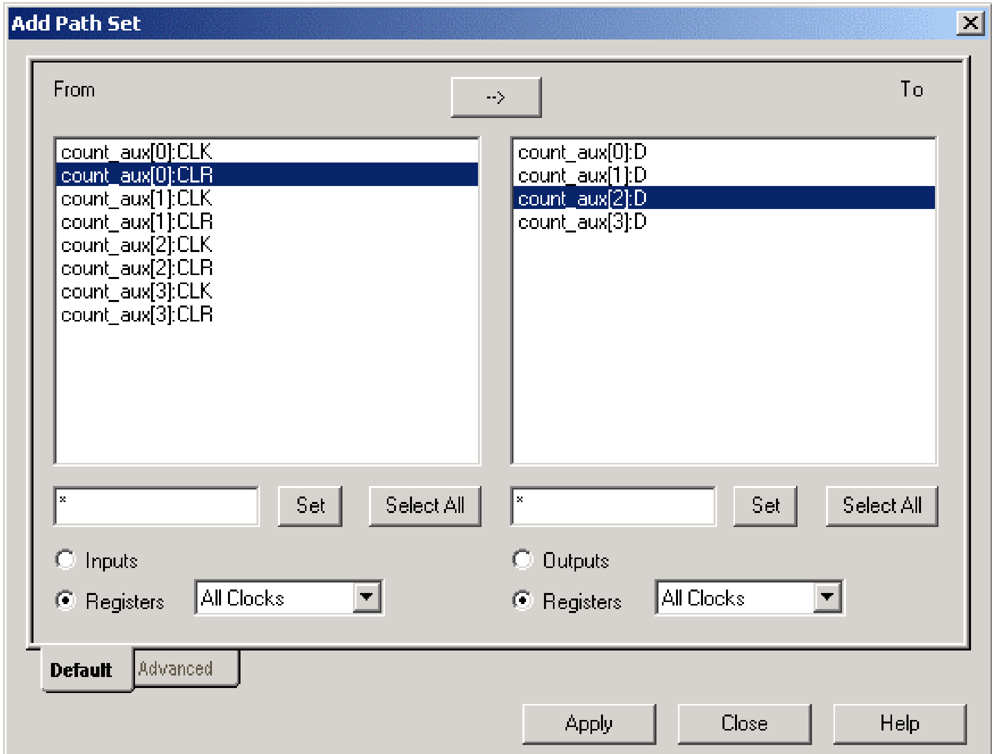
Adding path sets

Create and add path sets to the Paths tab to determine delay information and enter constraints. User-defined sets enable you to customize the sets that are available for analysis. By creating custom sets, you can simplify timing analysis and constraint setting for specific blocks or paths in your design.

For example, if you are concerned about timing of the lower-level block “sub_block_1” in your design, you can create a set that only includes timing paths in that block.

To add a set:

1. Click the **Paths** tab.
2. From the **Edit** menu, select **Add Set of Paths**. The **Add Path Set** dialog box consists of two screens, **Default** and **Advanced**. The **Advanced** tab enables you to use keywords to create a set. For more information on using keywords consult the Index.



Add Path Set Dialog Box

3. Select the desired clock.

4. Click the directional button to select path direction.

5. Select the desired **Inputs** (all input pad pins) or **Registers** (all input pins on the flip-flops and latches).

Use the pull-down menus to choose the active clock nets. Choose **All Clocks** for both to find delays for all register-to-register paths. Choose the **Outputs** radio buttons to filter the **From** and **To** list boxes (it limits the **From** and **To** list boxes to all output pad pins).

Select your desired starting and ending points in the **From** and **To** list boxes. Naming filters are provided to limit the list of terminals in the display. The naming filters use the * character as a wildcard and the / character to delimit levels of hierarchy.

For example, use * to filter for all terminals; *:E to filter for all terminals with pin E; U1/* to filter for all terminals in block U1; and U1/*:E to filter for all terminals in block U1 with a pin E. You can also use multiple wildcards such as */U1/*:E. After entering your naming filter, click the Set or Select All button.

If the directional button is pointing right:

1. Select a starting point in the **From** list. The **To** list box displays all corresponding endpoints.
2. Select one or more endpoints in the **To** list box that complete the path set. Click the **Select All** button to select all endpoints.

If the directional button is pointing left:

1. Select the endpoint in the **To** list box. The **From** list box displays all corresponding starting points.
2. Select one or more starting points in the **From** list box. Click the **Select All** button to select all.

Click **Apply** to add the path set to the **Paths** tab. Continue creating and adding sets. When you are done, click **Close** to close the **Add Path Set** dialog box.

Add a "one input to all outputs" path set

To show one input to all outputs, you must add the set to the **Paths** tab. You can then view delay details and set constraints.

To show one input to all outputs:

1. In the **Paths** tab, choose **Add Set of Paths** from the **Edit** menu. The Add Path Set dialog box appears.
2. Select the **Inputs** and **Outputs** radio buttons.

   Make sure the directional arrow is pointing to the right, from **Inputs** to **Outputs**. Click the arrow to change its direction.
3. Choose the desired input starting point in the **From** list box.
4. Click the **Select All** button to select all outputs.
5. Click OK. The set showing one input to all outputs is added to the Paths tab.

Add an "all inputs to one output" path set

To show all inputs to one output you must configure and add the set to the **Paths** tab.

To create a set showing all inputs to one output:

1. In the **Paths** tab, from the **Edit** menu, click **Add Set of Paths**. The **Add Path Set** dialog box appears.
2. Select the **Inputs** and **Outputs** radio buttons.
3. Click the directional arrow to point it left.
4. Choose the desired output endpoint in the **To** list box.
5. Select all inputs (all starting points), by Clicking on the **Select All** button under the **From** list box.
6. Click **OK**. The set showing all inputs to one output is added to the **Paths** tab.

Edit or Remove a Path Set

To edit a path set:

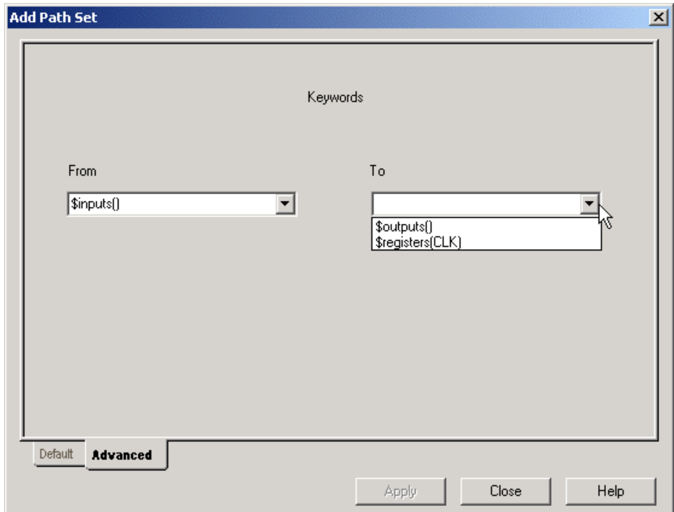
1. Select the set in the **Paths** tab.
2. In the **Edit** menu, click **Edit Set of Paths**, or right-click and choose **Edit Set**. This displays the Edit Set dialog box.
3. Edit the **Path Set** and click **OK**.

To remove a path set:

1. Select the set in the **Paths** tab.
2. In the **Edit** menu, click **Remove Set of Paths**, or right-click and choose **Remove Set** from the shortcut menu. The set is removed.

Adding/Removing sets with keywords

The **Advanced** tab in the **Add Set** dialog box enables you to use keywords (macros that represent various sets of terminals) to create a set. Timer does not save constraints on the **Advanced** tab in the **Add Set** dialog box. You must **Commit** your changes in the main window to save your constraints.



Add Path Set - Advanced Tab

Keywords for the SX-A, eX, Axcelerator, and Flash families are explained in the Keyword Filters section.

Supported keywords include:

`$inputs()`
All input and bi-directional pins.

`$outputs()`
All output and bi-directional pins.

`$registers(clock_name)`
The pins of all registers driven by the clock whose name is clock_name.

To create a set using keywords:

1. Click the **Paths** tab.
2. In the **Edit** menu, click **Add Set of Paths**.
3. Click the **Advanced** tab.
4. Enter the **From** keyword or choose a keyword from the **From** drop-down list box to define the **From** set.
5. Enter the **To** keyword or choose a keyword from the drop-down list box to define to set.

6. Click **OK**. This displays paths for the keyword set in the paths tab.

Timing Constraints

Timing constraints

Timer enables you to specify timing constraints and requirements for clocks and paths. Use these constraints to generate timing reports and in Timing Driven Layout. In order to run timing driven layout, you must enter and commit your constraints in Timer before you exit the tool.

For Flash, some of the constraints must be entered via GCF and SDC constraints. For Axcelerator, you can run timing driven place-and-route even if you have not set any user constraints.

Constraint guidelines

Constraint Guidelines

Delay constraints control the Timing Driven Layout engine. You can define these constraints using Timer or by importing an external DCF, SDC, or GCF (for Flash) file. The timing-driven layout engine considers the defined delays when allocating silicon resources with the goal of meeting or beating all constraints if possible. The timing-driven layout engine evaluates the performance criticality of one function versus another when allocating device resources. Because resources are limited, use the following guidelines to ensure the defined constraints meet the needs of the design without impairing device resources.

General Guidelines

Set Sufficient Constraints - All constraints for the design should be defined to ensure correct operation of the Timing Driven Layout engine. Timing Driven Layout considers networks that have not been defined as “don't care” paths, which have a low priority for resource allocation. If these undefined paths are actually critical, they may fail to meet performance demands.

Avoid Unnecessary Constraints - Describe “don’t care” paths to free high performance device resources. Not defining a path is one mechanism for doing this. However, it is difficult to avoid defining some “don't care” paths, so Designer provides clock exceptions and global stop sets to enhance this capability (see Clock exceptions).

Avoid Over-Constraining - The Timing Driven Layout engine is designed to achieve or exceed the delay constraint defined (less than or equal). Defining a constraint shorter than is actually required for margin can have a negative impact on the performance of the device because of competition for device resources.

Specifying clock constraints

Use the **Clocks** tab to assign values to each clock network in your design.

To assign clock constraints:

1. Click the **Clocks** tab.
2. Select the clock of interest in the **Select Clock** pull-down menu in the toolbar.
3. Specify the timing requirements. In the **Constraints** area, define the **Required** and **Duty Cycle** areas. Select **MHz** or **ns** from the pull-down menu.
4. Click **Set**.

Clock exceptions

Timer enables you to specify global clock constraints. If you have paths that are not required to meet the global constraint (for example, multi-cycle paths), then list them as exceptions. The Clocks Exceptions area on the Clocks tab provides a mechanism for doing this. A terminal specified as a clock exception will cause all paths beginning or ending at this terminal to be unconstrained by the global timing constraint.

To add or remove terminals from the Clock Exception List:

1. Click the **Clocks** tab.
2. Select the **Clock** name from the drop down menu.
3. Enter a constraint in the **Constraints** area and click **Set**.
4. Select **Source** or **Sink** in the **Clock Exceptions** area. The **Clock Exceptions** area displays the Pins. The terminals of the sequential device are displayed using an <instance_name>:<pin_name> format.

   For example, a DFM with an instance name of U1\FF1 will have a single source terminal, U1\FF1:CLK, and 3 three sink terminals: U1\FF1:A, U1\FF1:B, and U1\FF1:S.

5. Use the **Filter** field to further sort the list of clock pins. Naming filters are provided to limit the list of terminals for consideration. The naming filters use the * character as a wildcard and the / character to delimit levels of hierarchy.

   For example, use * to filter for all terminals; *:E to filter for all terminals with pin E; U1/* to filter for all terminals in block U1; and U1/*:E to filter for all terminals in block U1 with a pin E. After entering your naming filter, click the **Set** or **Select All** button. Multiple * and / characters may be used.

6. **Add** or **Remove** the clock exception. To add a clock exception, highlight the desired entry from the **Clock Pins** list and click **Add**. To remove an exception, highlight it in the **Exceptions** list and click **Remove**.

7. From the **File** menu, select **Commit** to commit changes.

Path constraints - specifying or removing

You can specify a timing constraint on a specific path or groups (sets) of paths.

To specify a timing constraint for a path set:

1. Click the **Summary** or **Paths** tab.

2. Select a **Path** set.

3. Enter the timing constraint. On the **Summary** tab enter the constraint in the **Required** text box and click **Set**. On the **Paths** tab enter the constraint in the **Max Delay** column.

To specify a timing constraint for a specific path:

1. Click the **Paths** tab.

2. Click the corresponding set in the **Set** grid.

3. Select the path in the path grid.

4. Enter the timing constraint in the **Max Delay** column.

Removing Constraints

You can remove all constraints or just selected constraints. To remove all constraints choose **Remove All Constraints** from the **Edit** menu.

To remove select constraints on the Paths tab:

1. Click the **Paths** tab.
2. Select the path set with the constraint you wish to remove.
3. In the **Edit** menu, click **Remove Selected Constraints**.

To remove select constraints on the **Summary** tab, delete the constraint in the **Required** text box and click **Set**.

Timing Results

Export Results

From the **Paths** tab, you can export the path or set grids in a text file.

To save your results to a text file:

1. Click the **Paths** tab.
2. In the **File** menu, click **Export Path Grid** or **Export Set Grid**. This displays the **Save As** dialog box.
3. Choose a destination on your disk, enter a **File Name** and click **Save**.

Commit before you exit

If you wish to save the constraint requirements entered into Timer, you must **Commit** your Timing results before exiting Timer.

To commit your results choose **Commit** from the **File** menu before exiting, or click **Yes** when asked if you would like to commit your results before exiting. Timer saves your timing constraints to Designer's temporary design database.

Generate reports

The timing report enables you to quickly determine if any timing problems exist in your design. The timing report lists the following information:

Delay from input I/O to output I/O (longest or shortest, depending on your Preferences).

Delay from input I/O to internal registers (longest or shortest, depending on your Preferences).

Delay from internal registers to output I/O (longest or shortest).

Delays for each clock network (longest or shortest).

Delays for interaction between clocks networks (longest or shortest).

To generate a timing report:

1. In the **Tool** menu, click **Report Paths**. The **Timing Report** dialog appears. The External Setup-hold Timing Check box and the Slack Threshold text box are explained in the Timer Report dialog box section.
2. Click **Options** to specify more settings for your report. This displays the **Preferences** dialog box.
3. Verify your timing analysis preferences. Timer uses these preferences to generate your report.

Maximum List Size:

Longest/Shortest Path(s) - Defines the number of paths to display in the report (default is 1)

Expanded Path(s) in List - Defines the number of expanded paths to display in the report (default is 15)

Sort By:

Actual - Sort paths by actual delays.

Slack - Sort paths by slack delays.

This option is available only if you have entered timing constraints.

Case: specifies your operating conditions, **Best** (0 degrees centigrade), **Typical** (25 degrees centigrade), and **Worst** (70 degrees centigrade).

Break Path at Register - The default timing paths break at all clock, gate, clear, and preset pins. Uncheck the boxes in Break Path at Register to generate a timing report that displays paths that pass through (and do not "break") at all Clock, Clear, Gate, and Preset pins.

Once you are satisfied with your selections, click **OK** in the **Preferences** dialog box and then click **OK** in the **Timing Report** dialog box. The timing report is displayed in a separate window.

Violations Report

For families that use the pin-to-pin timing model, the Violations report enables you to obtain constraint results sorted by slack. You can now view Max Delay violations as well as Min Delay violations in the report.

Keyword Filters

Keywords

Keywords enable you to filter out any unwanted paths or instances, making it easier to view critical paths in the design and limiting the paths displayed for a particular set. The use of keywords is only supported for pin-to-pin delay families.

Use keywords to create custom sets for Timer's Paths tab screen.

The use of keywords is only supported for pin-to-pin delay families (for a an explanation of pin-to-pin and input-to-input delay families, see Delays, PLLs, RAMs, and FIFOs). Use keywords to create custom sets for Timer's **Paths** tab screen. Refer to Adding/Removing sets with keywords for details on how to enter keywords.

Keywords enable you to filter out any unwanted paths or instances, making it easier to view critical paths in the design and limiting the paths displayed for a particular set. Timer uses two types of keywords, first- and second-level.

Levels of Keywords

The first-level keywords enable access to the main objects of the design, such as registers, while the second-level keywords enable access to a sub-list of these main objects. For instance,

$registers() is a first-level keyword that enables access to all the registers of the design. This list includes clock pins, data pins, enable pins and, asynchronous pins.

If the *$registers()* keyword is combined with the second-level keyword *$datapins()*, the related command is applied only to the data pins of the registers. You can use a second-level keyword only with a first-level; second-level keywords may not be used alone. In Timer, only the first-level keyword *$registers()* may be combined with the second-level keywords. Use the colon ":" without any spaces to combine first- and second level keywords. Keywords and filters are case insensitive.

Filtering

Filter keywords with brackets [ ]. The filter is a string that is used as an identifier (it may contain wild cards). [ ] with an empty string is not accepted in the macro language. The user can enter $registers(), $registers()[*filterString*], but not $registers()[ ].

Functions

Sometimes you may want to locate objects of the design by defining or identifying other objects. For instance, you might want to analyze delays of all the registers driven by a specific primary clock. Functions can help you locate the registers (objects) by defining the primary clock (identifier).

To use functions, the identifier of the object has to be reported between parentheses ( ). This identifier may contain wild cards and can also be another keyword. For example:

`$registers(clock1)`	returns all the registers driven by the primary clock "clock1"

Supported Keywords

Timer supports the keywords listed below.

First-Level Keywords	Second-Level Keywords

$registers() $inputs() $outputs() $clocks() $ports	$datapins() $clockpins() $asyncpins() $enablepins() $outputpins() $inputpins $allpins

First-Level Keywords

Each keyword has two identifiers, a long version and a short version. They both have exactly the same function. The first-level keywords are defined below.

`$registers(ClockName)` or `$reg(ClockName)`

The keywords above only display the registers (edge-triggered flip-flops and level-sensitive latches) controlled by the clock *ClockName*. If no *ClockName* is specified, this keyword will cause all the registers of the design to be displayed.

`$inputs()` or `$in()`

This keyword only displays all the primary inputs of the design.

`$outputs()` or `$out()`

This keyword only displays the primary inputs of the design.

"$Clocks()" or "$CK()" only displays the primary clocks of the design.

`$ports(`*InstanceName*`)` or `$po(`*InstanceName*`)`

This macro replaces all the primary inputs and outputs of the design.

Second-Level Keywords

While first-level keywords allow access to the main objects of the design, such as registers, second-level keywords give access to a sub-list of these main objects.

Currently, second-level keywords can only be used with the first-level keyword *$registers()*. A first-level keyword is separated from a second-level keyword with the colon ":" character, without any white space.

As with first-level keywords, most second-level keyword have two identifiers, a long version and a short version. Each has the exact same function. In the following examples, it is assumed that the notion of event and pin are implicit.

“$DataPins()” or “$dp ()” indicates all the data pins of a register. For example:

`$registers(CLK):$dp()`	displays the data pins of all the registers controlled by CLK

“$OutputPins()” or “$qp()” indicates all the output pins of a register. For example:.

`$registers(CLK):$qp()`	displays the output pins of all the registers controlled by the primary clock CLK

“$ClockPins()” or “$cp()” indicates all the clock pins of a register. For example:.

`$registers(CLK):clockpins()`	displays the output pins of all the registers controlled by the primary clock CLK.

“$AsyncPins()” or “$ap()” indicates all the asynchronous pins of a register (preset and clear).

“$EnablePins()” or “$ep()” indicates all the enable pins of a register. For example:

`$registers(CLK):$ep()`	displays the enable pins of all the registers controlled by the primary clock CLK.

$inputpins() or $ip() indicates all the input pins of a register. For example:.

`$reg(CLK):$inputpins()`	Displays the input pins of all the registers controlled by the CLK

$allpins() indicates all the pins of a register. For example:.

`$registers(CLK):$all-pins()`	Displays the pins of all the registers controlled by the CLK

Second-Level Exceptions

In order to provide more flexibility, the second level keywords can be coupled with exceptions. For instance, if the you want to select all the input pins of the registers except the clock pins, you can use the following macro:

```
$registers(clk):$inputpins(TmacEx_CLOCKPINS)
```

The available exceptions are listed in the following table:

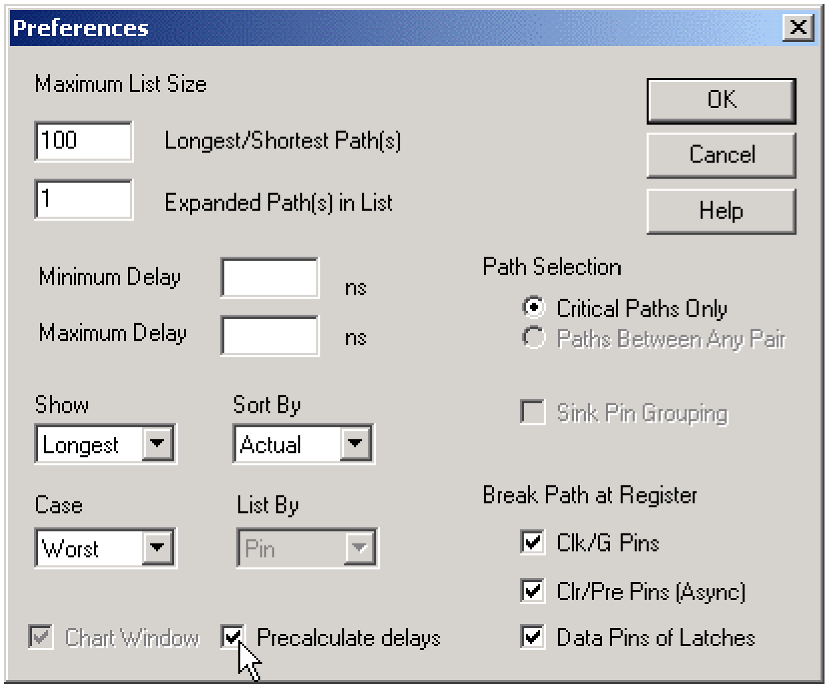
Exception	Result
TmacEX_CLOCKPINS	The clock pins will not be returned from the pins indicated by the 2nd level macro.
TmacEX_DATAPINS	The data pins will not be returned from the pins indicated by the 2nd level macro.
TmacEX_ASYNCPINS	The asynchronous pins will not be returned from the pins indicated by the 2nd level macro.
TmacEX_INPUTPINS	The input pins will not be returned from the pins indicated by the 2nd level macro.
TmacEX_ENABLEPINS	The enable pins will not be returned from the pins indicated by the 2nd level macro.
TmacEX_OUTPUTPINS	The output pins will not be returned from the pins indicated by the 2nd level macro.

Setting Preferences in Timer

Delay preferences

The Preferences dialog box controls your delay and timing analysis preferences. If you are using UNIX, consult the UNIX Command Summary for a list of the commands related to preferences.

You may wish to prevent Timer from calculating delays when the tool starts. (By default, Timer estimates all potential clock frequencies, as well as the 4 delays corresponding to the 4 primary sets (All inputs to All outputs, All inputs to registers, registers to All outputs, registers to registers) associated with the slowest clock when the tool opens.)



Preferences - Precalculate Delays Selected

To change your option for pre-calculating delays, from the **File** menu, select **Preferences** and deselect “Precalculate delays.” Alternatively, you may choose to modify the related variables. To do so, from the Designer GUI, in the Options menu, select “Set Variable.” In the Variable Name dialog box, enter:

```
TIMER_PRECALCULATE_DELAYS
```

And in the Value dialog box, enter

```
0
```

(default is “1”).

Changing and displaying paths

Use the **Preferences** dialog box to control the number of paths displayed in the **Paths** tab, **Expanded Paths** window, and **Timing Report**.

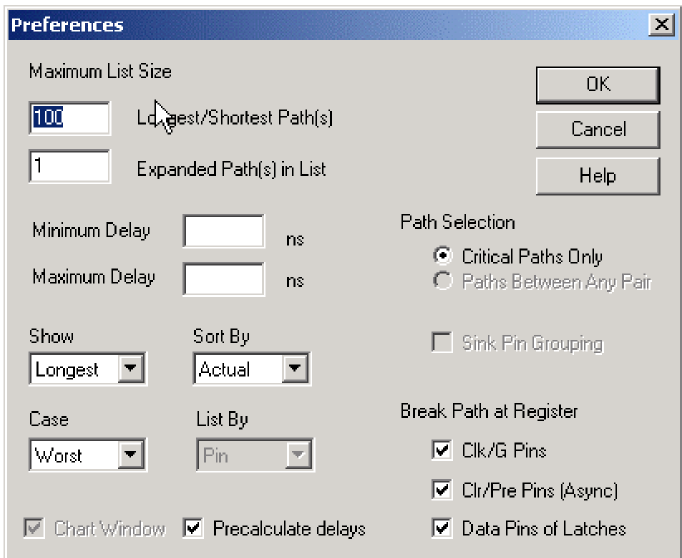
To change the number of paths that Timer displays:

1. In the **File** menu, click **Preferences**. The **Preferences** dialog box appears.

2. In the **Maximum List Size** area, enter your default preferences for the maximum number of **Longest/Shortest Path(s)** and **Expanded Path(s)** that you want to display

   The value for **Longest/Shortest Path(s)** is set to 100 by default; you can modify the value if you wish to see more paths in your report. However, the higher the value the longer it takes to invoke Timer.

3. Click **OK**.

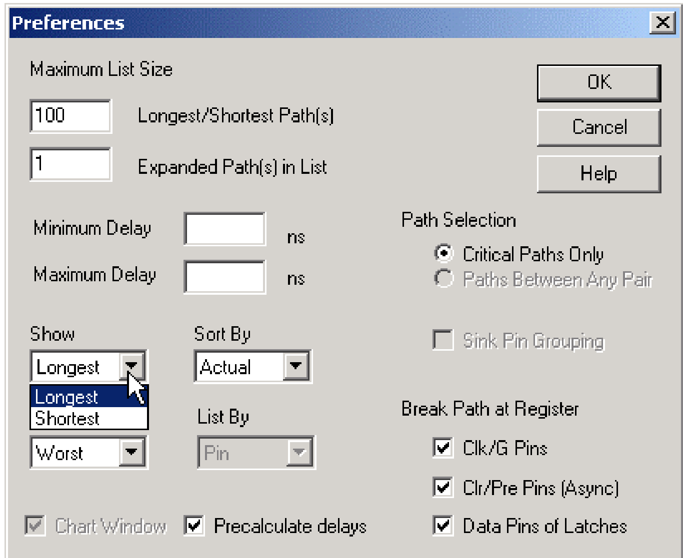


Displaying the Shortest Paths First

By default, Timer displays the first 100 paths from longest to shortest in the **Paths** tab and **Expanded Paths** window. When Timer displays paths from longest to shortest, it reports setup times for registers. To view hold times in the **Expanded Path** window, you must set the **Preferences** to **Show** the **Shortest** paths.

To display the shortest paths first:

1. In the **File** menu, click **Preferences**. The **Preferences** dialog box appears.
2. Select **Shortest** from the **Show** drop-down menu.
3. Click **OK**.



Setting Preferences in Timer to Display Shortest Paths

Delay filters (max. or min.) / Sorting by actual or slack delays

Setting Minimum or Maximum Delay Filters

Use the **Preferences** dialog box to filter paths for delays above, below, or between a specified value. Enter your display preferences in the **Maximum Delay** and **Minimum Delay** boxes and click **OK**.

Sorting and Displaying by Actual or Slack Delays

Timer can display delay information in two ways:

Actual delay values

Slack, which is the difference between actual delay and a user-specified delay (that is, user-specified constraint)

Displaying by Actual Delay

The actual delay is the path delay between two points in your design. This is the only way to sort your data if you do not have any timing constraints entered (for information on setting timing constraints, see Constraint Guidelines). If you have entered timing constraints, the actual delay report automatically displays the slack - even if you don't ask for it - but the data will always be listed from longest to shortest actual delay.

Actual delay measurements may be calculated before or after layout.

To display Actual delay:

1. In the **File** menu, click **Preferences**. This displays the **Preferences** dialog box
2. Select **Actual** in the **Sort By** pull-down menu.
3. Click **OK**.

Displaying by Slack Delay

Slack delay is the delay difference between a timing constraint entered in Timer and the actual delay of a path. For example, if a signal takes 20 ns to get from point A to point B, and you entered a timing constraint of 15 ns, the Timing Report would list -5 ns slack for that path. Thus, if the slack is negative, then the actual delay did not meet the desired timing by the absolute value of the slack (in ns). Conversely, if the slack value is positive, then the timing constraint was met, with the slack value (in ns) to spare. In a violations report, Timer sorts the data (by default) from longest to shortest slack.

When displaying slack, all the paths without timing constraints are filtered from the reported data. This enables you to quickly determine how well your design meets your timing requirements. This is especially useful for viewing critical delays like register-to-register, clock-to-out, and input-to-register.

Best\Typical\Worst Case Analysis

By default, Timer displays the worst case analysis.

To display the best or typical case analysis:

1. In the **File** menu, click **Preferences**. The **Preferences** dialog box appears.
2. Select **Best**, **Typical**, or **Worst** from the Case drop-down menu. If you change the setting, timing is recalculated for the entire design; this may take a few minutes.
3. Click **OK**.

Selecting paths - Adding or removing break paths

Normally, Timer displays only critical paths. Critical paths are the longest path between any of the starting points (terminals) and each ending terminal. If you would like to see the timing of all paths between any of the starting terminals and any of the ending terminals, select **Paths**

Between Any Pair (input-to-input timing model families only) in the **Path Selection** area of the **Preferences** dialog box. Selecting **Critical Paths** displays only critical paths.

Adding and Removing Break Paths

Asynchronous feedback paths in a design can cause paths to be reported as having excessive delays. The most common example is feedback paths through asynchronous Set or Reset pins to banks of flip-flops, like a state machine or a counter.

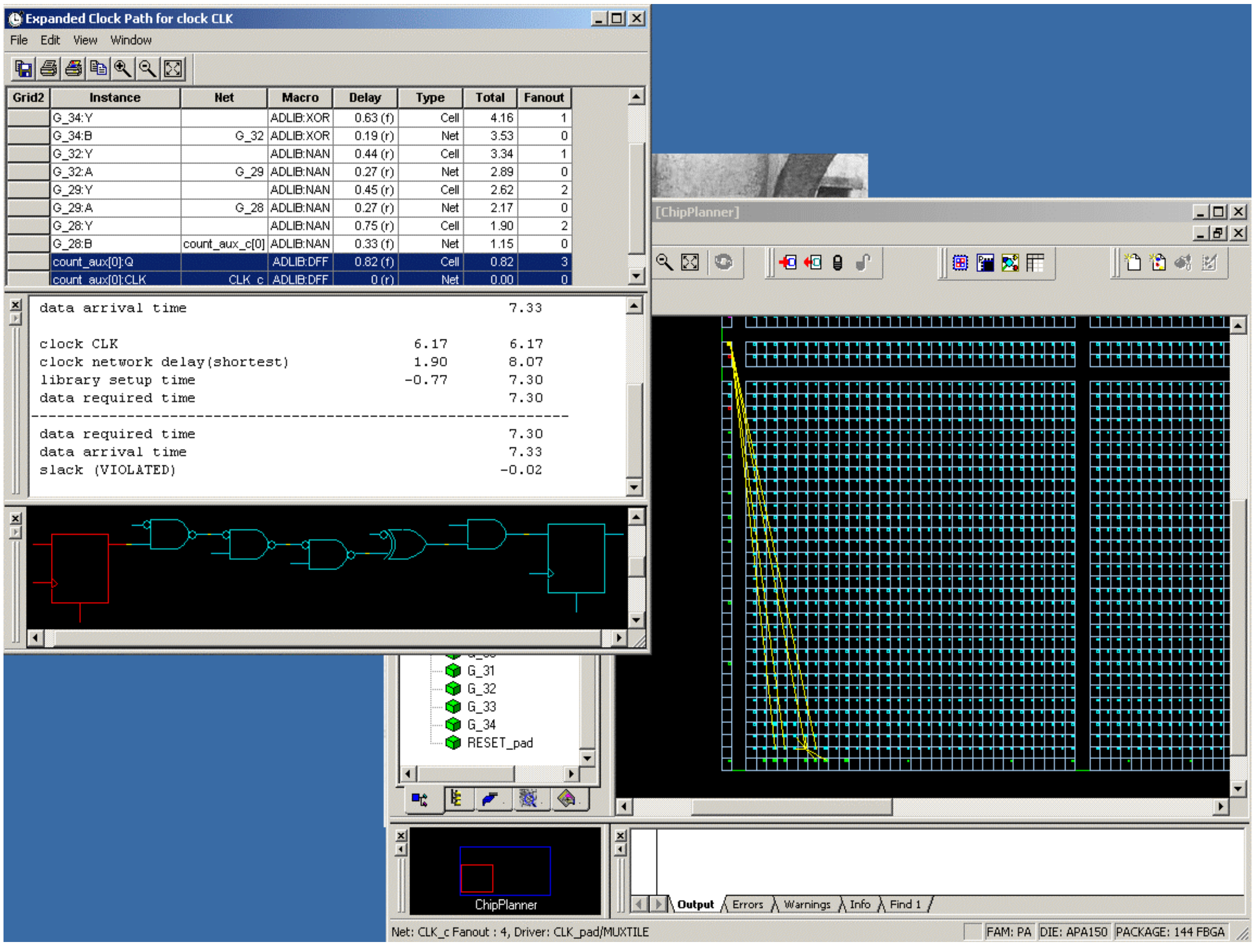
To exclude paths:

1. In the **File** menu, click **Preferences**. The **Preferences** dialog box appears
2. **Break Paths at Register**. Choose **Clk/G Pins**, **Clr/Pre Pins (Async)** or **Data Pins of Latches** to prevent displaying paths that pass through either clock, gated, clear, preset, or data pins of flip-flops or latches.
   Note: The **Break Paths at Register** option is selected by default, and the paths are excluded. Deselect the checkboxes in the **Timer Preferences** menu to display these paths.
3. Click **OK**.

Using ChipEdit with Timer

Using ChipPlanner/ChipEditor with Timer

Use ChipPlanner or ChipEditor and Timer together to view place-and-route of Timer paths.



Timer and ChipPlanner

To view critical paths:

1. Open **Timer** and **ChipEditor/ChipPlanner** from Designer (ChipEditor opens if you are using an ACT1, ACT2, ACT3, DX, MX, SX, SX-A, or eX device; all other devices use ChipPlanner).
2. In Timer, click the **Paths** tab.
3. Select a **Path** set in the path set grid. Paths within that set are displayed in the path grid.
4. Select the path you wish to expand.
5. Expand the path by double-clicking on the path, or in the **Edit** menu, click **Expand Path**. The **Expanded Paths** window opens and displays a single path in the **Expanded Paths Grid** and a graphical representation of the paths in the **Chart Window**.

6. Select a module or net in the **Expanded Paths** dialog box. The module or net is shown in **ChipEditor** or **ChipPlanner**.

Refer to the online help for more information on how to use the ChipEditor/ChipPlanner tools.

You can add and remove break points in Timer while you use the ChipEditor/ChipPlanner tool.

Timer Glossary of Terms

This glossary defines terms and concepts used in the Timer online help.

clock exception

A terminal in a synchronous network that should be excluded from the specified clock period. The exception can remain undefined (don't care) or can be assigned a unique value in the Path Constraint Editor.

critical path

The path within a design that dictates the fastest time at which an entire design can run. This path runs from the source to a sink node such that if any activity on the path is delayed by an amount t, then the entire circuit function is delayed by time t.

delay constraint

A delay constraint defines a fixed amount of time required for a signal to propagate from all starting terminals to all ending terminals for a network.

destination

An ending point, sink node, for a timing analysis path, often the data input of a synchronous element or pad.

don't care path

A signal path in which the delay is considered to be infinite.

Dynamic Timing Analysis

Dynamic timing analysis (simulation) has been the standard mechanism in verifying design functionality and performance. Both pre-layout and post-layout timing analysis can be performed via the SDF interface. Pre-layout timing analysis provides quick estimates of the designs performance. Post-layout timing simulation on the other hand provides accurate timing information that is appropriate for device or system level simulation.

filter

A set of limitations or options applied to the timing analysis to more specifically target important items of interest.

global Stop

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.

network

Network. A network can consist of 1 or more start terminals and 1 or more end terminals. All signal paths connecting any start terminal to any end terminal are included in the network. Only one delay value can be assigned to each defined network. Networks can be defined implicitly by a common clock (synchronous network) or explicitly by a defined set of terminals. Network and Paths are used interchangeably.

path

An ordered set of elements identifying a logic flow pathway through a circuit. A path may consist of a single net or a grouping of related nets and components. There can be multiple, or parallel paths (consisting of nets and components) between the two pins. When a component is selected as part of a path, both the input pin to the component and the output pin are included in the path. A path stops when it reaches the data input of a synchronous element (flip-flop), clock, or pad. A path usually starts at the output of a synchronous element, clock, or pad.

path delay

The path delay defines the sum of all the individual delays of the nets and the logic macros in the signal path.

path sets

In this manual we refer to groups or categories of paths as “sets.” Path sets are displayed on the Paths tab.

signal path

The signal path describes a consecutive sequence of logic and nets, the first net being driven by a start terminal, and the last net driving a macro input pin of the end terminal

slack

The difference between the constraint and the analyzed value, with negative slack indicating the analyzed value is greater than the constrained value.

Standard Delay Format (SDF)

Standard Delay Format is an industry-standard file format used for storing timing data generated by EDA tools. It is often used for simulation.

Static Timing Analysis (STA)

Static timing analysis is an exhaustive and convenient method of ensuring that the design meets its timing requirements. There are functions that are especially easy to analyze with the static approach. Complex functions such as a multiplier are much easier to analyze using the static approach because static analysis offers one hundred percent coverage with minimal effort compared to dynamic timing analysis. In addition, the static approach is faster for highly synchronous designs compared to dynamic timing analysis.

status bar

The area located at the bottom of an application window

Index