

TimingDesigner

The Timing's Right

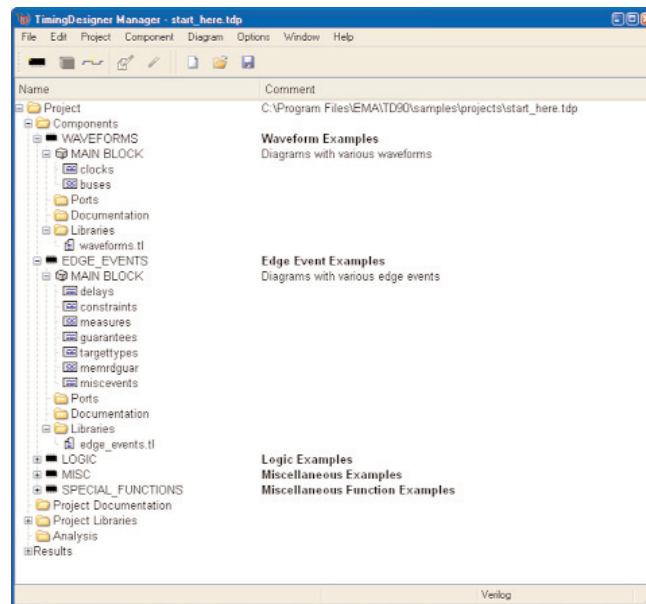
Capabilities

Features and Capabilities

This overview introduces the key features in TimingDesigner, the leading interactive timing specification and analysis tool used for timing-critical designs. TimingDesigner is straightforward to use, providing a clear and consistent methodology for communicating vital timing information. Users depend on TimingDesigner to evaluate comprehensive sets of timing alternatives and provide accurate direction to complex timing challenges.

Intuitive Timing Diagram Approach

TimingDesigner includes an extensive set of capabilities to specify, analyze, and document timing for embedded function interfaces on an ASIC or programmable IC, or between devices on a circuit board. Incorporating the use of highly interactive and accurate timing diagrams, allows the designer to focus on complex timing challenges and effectively communicate their resolution.



Manage

Organization for Timing Analysis Projects

TimingDesigner's Project Manager gives users the ability to better manage growing design complexity by organizing multiple timing diagrams into logical components within a single project. These organizational aspects will enable users to better manage the timing specification and analysis of their high performance designs and simplify the exchange of critical timing information among project team members.

Component-based Interface Diagram Organization

Components and blocks are arranged and displayed in a single tree, with a summary list of all constraint violations in the project diagrams. Each diagram component includes a port list noting all port connections associated with the device interface. It also includes a project netlist that shows component interconnectivity which TimingDesigner uses to determine what signals are controlling

interface protocol operations.

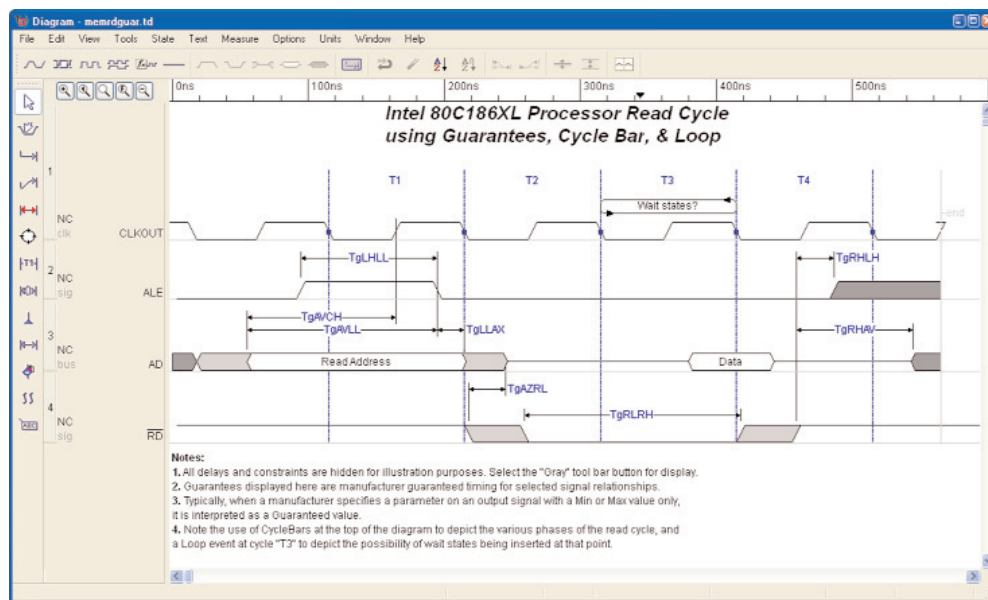
Auto-merge Diagrams for Quick Interface Evaluations

Using the port list and netlist information, TimingDesigner can automatically merge designated component interface diagrams to greatly simplify the creation of interface timing scenarios. You can merge two diagrams from different components, automatically creating an interface timing result that accounts for component connectivity, as well as managing signal duplication and propagation delays.

Signals, Buses, Edges, and Clocks

- Signals, buses, edges, and clocks are easily added and moved with all dependencies automatically updated.
- Signal, buses and clocks can be drawn as single-ended or differentially-ended.
- Signals may include high, low, & high impedance signal states as well as valid & invalid bus states.
- Signal and bus names, initial states, and edge types are easily specified.
- Buses may include valid, invalid, and high-impedance signal states, and are easily expanded to show their elements with conversion from Virtual to Derived bus types.
- I/O signal direction can be changed on individual edges.
- Voltage threshold markers are supported for any edge so specific logic transition levels are clearly noted.
- Clocks are specified by period, frequency, duty cycle, offset, and rising/falling edge jitter.

Evaluating alternatives is essential to developing specifications that can accurately convey design details and timing budgets. TimingDesigner supports the early investigation of timing options and provides a straightforward means to clearly specify the sequence of events and timing relationships required for modules or subsystems to communicate as expected.



Specify

Derived Clocks and Signals

Derived clocks can be specified from a source clock or other derived clocks. Derived clocks and signals can be drawn as single-ended or differentially-ended. Accurate models of real world clocking schemes with separate rising and falling delays are fully supported.

Derived signals are the sequential or combinatorial outputs of other waveforms in the diagram. To use derived signals, logical signal relationships using VHDL or Verilog syntax are entered and resulting output signals are automatically created.

Use of derived signals (inverted signals, gated clocks, and any signal that tracks a clock) permits the rapid creation, evaluation, and modification of complicated specifications ranging from simple signal propagations to complex conditional equations. Changes made to derived signal inputs are reflected instantly in the affected outputs, including changes in

spreadsheet variables. Manual modification of signals is not required allowing quick analysis of "what-if" scenarios and execution of worst-case calculations.

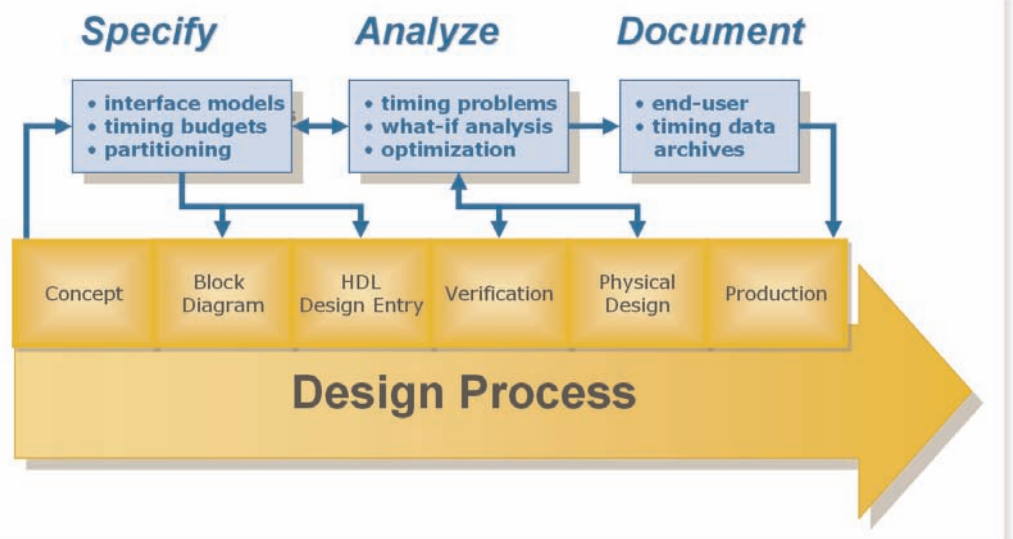
Insert and Delete Time and Clock Cycles

Time can be inserted and deleted anywhere in a diagram allowing the shift of waveform edges. Adding or removing time on an individual waveform, a group of waveforms, or the entire diagram is also supported.

Cycles on clocks or derived clocks are easily inserted and deleted to support wait states.

Robust Toolset

- **Delay tool:** Is used to add gate and path delays which are automatically accumulated to determine the earliest and latest possible occurrence for every edge in a delay chain. Delay changes automatically update the entire diagram.
- **Constraint tool:** Is used to add setups, holds, and pulse widths. Timing margins are automatically calculated when a diagram is changed and violations are instantly highlighted in red.
- **Guarantee tool:** Is used to add guarantees (known times between edges) and overrides the normally calculated min/max time between edges.
- **Pulse tool:** Is used to add signal pulses with adjustable widths.
- **Measure tool:** Is used to display the inside and outside time measurement between two edges. This measurement dynamically adjusts as changes are made to the diagram.
- **Cycle tool:** Is used to automate the labeling of different cycles on timing diagrams. It simplifies documentation of major and minor cycles, number clock cycles, and creates rulers with customized numbering systems.
- **Cause and Effect tool:** Is used to add a simple visual event to aid in descriptive clarification of your timing diagram. The event is displayed as a curved arrow between two edges whose spacing, line width, and color are user-selectable.
- **Text tool:** Is used to add text to any element on the timing diagram. When the diagram is updated, the text remains linked as originally specified.



Comprehensive Event Specifications

Sample events...

are used to sample a data signal relative to an edge on a control signal, allowing specification of latched data values. Sampled values are logged in the Parameter Spreadsheet.

Wait events...

are used to signify that a signal transition is dependent on another signal, allowing the specification of wait states.

Time break events...

are used to designate a finite, user-defined time break in the interface specification. Time breaks can be attached to any time on the ruler or any edge of a waveform, and will serve to collapse “uninteresting” spans of time. This is especially useful for circuits with large time scales, such as video chips.

End diagram events...

are added to specify the end of a timing diagram at which point all signals are automatically extended and aligned.

Loops...

are used to specify repeated portions of a diagram, such as with page mode & EDO DRAMs, and serial-to-parallel & parallel-to-serial interfaces.

Dynamic Text Substitution

Keywords can be embedded in labels and text annotations. Customized information shown may include diagram name, revision number, designer’s name, earliest and latest times, source and target edges, uncertainty, as well as other properties.

Keywords can be substituted in comments, allowing automatic generation of concise, descriptive comments for timing parameters and edge relationships.

Keywords can also be used with value decode for easy display of state labels associated with state machines.

A Dynamic Text Window is provided to create reports and constraint information that dynamically links to timing analysis parameters within a diagram. Customized reports can be generated and resulting timing constraint information can be used to drive other tool flows.

User-defined Display Options

- Global and selective labeling options for diagram objects including name, formula, and user-defined text are fully supported.
- Edge and state labels can display information including time, state, user-defined text, and properties.
- Diagram objects and uncertainties can be shown or hidden.
- Groups of waveforms can be separated from each other with a Divider waveform object that allows descriptive text display and a user-specified line style.
- Waveform type (signal, clock, dclock, dsig) can be designated next to the row number.

SmartLayout & SmartSnap

- SmartLayout automates spacing to minimize overlap of delays, constraints, and guarantees for clear, readable diagrams.
- SmartSnap automates proper alignment of text annotations by placing annotations on or above the horizontal plane of the associated waveform as diagram changes are made.

Type	Name	Formula	Min	Nom	Max	Margin	Comment
1 V	Freq	50	50	50	50		Frequency variable for clocks
2 V	Period	1/Freq*1000	20	20	20		Period calculation for clocks
3 M	tm1	[9,11]				{9,11}{10,10}{9.5,10.5}	Measurement for pulse width
4 M	tm2	[19,21]				{19,21}{20,20}	Measurement for cycle time
5 C	tcPW	[,[(max(Period))/2]]			10	<-,1><0><-,0.5>	Constraint for maximum pulse width limitation
6 C	tcPer	[,max(Period)]			20	<-,1><0>	Constraint for maximum cycle time limitation
7 D	tDly1	[10,12.5,15]	10	12.5	15		Delay from Signal A to Signal B
8 D	tDly2	[45,47.5,50]	45	47.5	50		Delay from Signal A to Signal C
9 C	tCnst1	[10,43]	10		43	<20,3>	Constraint from Signal B to Signal C
10 M	tMeas	[30,40]	25		45	[25,45]	Measure without common uncertainty removed (unadjusted)

Analyze

Additional Editing and User Interface Features

Zoom... selections include Zoom-in, Zoom-out, Zoomrange, Zoom-back, and Zoom-full and are menu or keyboard driven. Zooming with a mouse selectable time range in the ruler or a user-defined time range is also available to support viewing and documenting specific time ranges within your design.

Ctrl-key keyboard commands... support single key navigation and editing allowing power users to design their diagrams with minimal mouse contact.

Context-sensitive menus... allow quick access to the most commonly used TimingDesigner commands by clicking the right mouse button. The contents of the menu change depending on which objects are selected.

Powerful Timing Engine and Patented Spreadsheet Deliver Accurate Results

Interfaces between embedded processors, memory, and logic functions on chip or between devices on a circuit board are often the source of difficult-to-locate timing violations. The timing engine with TimingDesigner uses a timing diagram specification to accurately analyze parameters and identify violations that may otherwise go undetected until late in the design process. The ability to quickly evaluate design alternatives and compute worst-case timing margins make it an excellent choice to help develop solutions for specific problem areas.

Innovative, Patented Parameter Spreadsheet Technology

All parameters are stored in a dynamically-

linked spreadsheet. Each row represents a delay, constraint, guarantee, skew, variable, user-defined function, alias, or part alias. Rows can be shared, to allow modification of multiple parameters at the same time. Columns are used for names, formulas, mins, noms, maxs, margins, and comments.

All value-encompassing parameter elements (delays, constraints, guarantees, measurements, and variables) can be referenced in the formula field of other parameter spreadsheet elements, producing unique and complex formula generation.

Timing analysis can be set for minimum value evaluations, maximum value evaluations, or min/max evaluations(default).

35 built-in analysis functions (such as square root, log, and cosine) are included for use in formulas to model complex parameters.

User-defined functions extend the spreadsheet analysis capabilities by enabling specific parameters to be used for computing delays based on external conditions such as loading and temperature.

Spreadsheet cells can be independently locked and colored, allowing designers to easily designate which parameters are fixed and which can be changed by other users.

Spreadsheet sort and search permits easy organization and analysis of parameter data according to row type, name, margin, and so on.

Instant Flagging of Constraint Violations

- The static timing engine automatically traces all delay paths specified in the timing diagram, selects the critical paths, and then computes the worst-case timing margins after each modification. The results are attained by comparing the total delay along each critical path to the min or max value specified in each constraint.

Automatic Removal of Common Delays

- Reconvergent Fanout: Common delays that occur when two constrained edges are derived from the same edge through timing chains are systematically removed to ensure that constraints will not fail due to overly-pessimistic static timing calculations.

Signal tracking (skews)

- Signal tracking acknowledges the relationships that exist between the delays driving two or more signals, clocks, or derived clocks and increases the timing margin in your design by returning uncertainty generated by gates within a common part. This allows design optimization without using overly-pessimistic delay values.

Timing Chains

Timing chains are automatically created when an edge that is a target of a delay is, in turn, used as the source for another delay. The accumulated values are maintained on each edge so that the manual calculation of edge positions in the design is not required.

Timing Calculations Viewer

Users gain insight to the calculation of margin and measure values in the Diagram Window. By selecting a measuring event, the Viewer will display a breakdown of the timing chains involved in the calculation, providing a quick and easy verification of the measurement in question. For measurements of multiple drivers or similar complexities, the Timing Calculations Viewer is extremely valuable. The Viewer can also graph the measurement chain in a directed graph saved as a graphical file for easier viewing.

Timing Violations Report Viewer

Displays a condensed report on the condition of all design constraints established in the Diagram Window in a convenient format. With a selectable report format, users can quickly determine which constraints need attention, and can quickly access them through highlight links in the Diagram Window. The report can also be saved as a file for later reference.

Timing Violations Report Viewer

Displays a condensed report on the condition of all design constraints established in the Diagram Window in a convenient format. With a selectable report format, users can quickly determine which constraints need attention, and can quickly access them through highlight links in the Diagram Window. The report can also be saved as a file for later reference.

Variables and Formulas

Variables may reference a min/nom/max value allowing design team members to change variable values and then automatically updating all impacted specifications.

Formulas, a patented TimingDesigner feature, are arithmetic combinations of constants and variables used for complex timing parameters. Special syntax supports min/nom/max constants where min, nom, max, or all three may be unspecified. Formulas may be used to model varying operating conditions for specification creation and are easily updated after layout/routing with real values, which confirm the final design remains within specification.

Interfacing with Other Design Tools

Interfacing is provided for several third-party tool flows such as the Cadence® OrCAD® and Cadence Allegro® products and the Mentor Graphics® Tau® timing tool.

Two methods are provided for SDF import. Create variables in your parameter or Library Window from all or part of an SDF file, or map existing waveforms to ports present in the file. All import settings can be saved to a designated diagram, automating future imports.

CSV import and export are supported in either the Parameter Window or the Library Window, allowing interaction with other spreadsheet programs.

Viewing and Analyzing Simulation Results

WLF, VCD, and FSDB, the export formats used by industry HDL simulators, are fully supported and allow the import of simulation waveforms for view and analysis. An intuitive dialog box is used to select specific waveforms and specify time ranges for importing waveforms.

Timing Parameter Libraries Provide Flexibility and Reliability

User-defined hierarchical library support enables direct access of large, hierarchically-organized libraries of timing variables from a timing diagram and library sharing between project team members.

Aliases and part aliases may be used to carry out quick performance trade-offs between different parts. Speed versions, manufacturers, or logic families are changed with a single entry and the resulting diagram is automatically updated with data for new parts.

FPGA Design Flow Integration

- Constraint information used to drive FPGA timing-drivenplace and route tools is created in the Dynamic Text Window. The Dynamic Text feature allows access to timing measurements and other results to create vendor-specific constraints. Constraints are easily cut and pasted into popular FPGA design tools, or saved to a text file for later use.
- Import support of Altera® (*.tao and *.alt files) and Xilinx® (*.twx files) timing report files allows access to post-route timing information for verification of FPGA timing. Selected port information from the timing files are mapped to diagram waveforms for auto creation/update of post-routetiming information to confirm design success.
- Automatic library creation of timing variables for Xilinxparts is provided using Xilinx's SpeedPrint utility. Part-specific timing data from SpeedPrint is imported into TimingDesigner's Library Window, providing quick and easy access to Xilinx timing data.

Document

Simplifies Communication of Design Details

TimingDesigner delivers the ability to clearly and accurately communicate design details by exporting or linking timing diagram files generated during the design process through OLE support or in its native format. The standard, easy-to-interpret format of timing diagram specifications improves communication of complex design information.

Object Linking and Embedding (OLE)

Linking... automatically updates timing diagram modifications when linked within documentation applications that support the OLE standard. No re-importation is required.

Embedding... allows timing diagram data to be stored within documentation applications supporting OLE, eliminating the requirement for multiple files to co-exist with specifications and other documents.

Extensive File Format Support

FrameMaker Maker Interchange Format (MIF), Joint Photographic Experts Group format (JPEG), Graphics Interchange Format (GIF), Portable Network Graphics (PNG), Portable Document Format (PDF), Scaled Vector Graphics (SVG), Encapsulated PostScript (EPS) with TIFF Preview, Enhanced MetaFile for Windows (EMF), and Tagged Image File (TIFF) formats are supported.

TDML (Timing Diagram Markup Language), a standard for exchanging timing diagram information, was jointly developed by Forte and Si2. The SGML-based standard is part of the ECIX (Electronic Component Information Exchange) architecture and makes it possible to exchange timing diagram information with other applications, including automated publishing systems and other EDA tools.

ASCII file formats ease custom integration with other tools. The diagram file format is compatible between UNIX and Windows platforms.

Command Line Switches

Automated publishing processes as well as batch export/import, and file conversion processes can be invoked and controlled through command line switches.

Batch Mode Operation

A limited subset of TimingDesigner commands can be executed from a Tcl based batch script, regardless of whether the TimingDesigner windows are displayed. Commands exist to export diagrams and spreadsheet, add and edit waveforms and edges, change diagram settings, and modify spreadsheet values.

Diagram Styles

Multiple graphical settings enable the change and refinement of timing diagrams to meet an individual user's preference for display and export.

User-defined styles can be customized for different uses including specification, analysis, and publication. Diagram style elements include separate color, weight, height, and pattern for edge relationships or waveform types; separate font size, style, and typeface control for waveform names, edge labels, edge relationship labels, text annotations, cycle bars, and the ruler; waveform height and edge slope angles; and default label settings for each type of diagram object.

Call Us Today!

For additional information, visit us at www.timingdesigner.com or call 877.362.3321.

System Specifications

Licensing Options and Network Support

Single user and floating licenses are administered through the FLEXnet license manager. Supports all TCP/IP-based networks.

Platform Support

Windows requirements

- Microsoft Windows Vista/XP/2000/Server 2003
- Minimum 22MB RAM, 50MB disk space
- Internet access

Linux requirements

- Red Hat Enterprise (RHEL) 4
- Minimum 26MB RAM, 80MB disk space
- X11R6+ for most windows managers
- Internet access

Solaris requirements

- Solaris 10 and later
- Minimum 26MB RAM, 80MB disk space
- Open Windows 3.x or X11R6+ for most Window Managers, including OpenLook Window Manager (olwm) and Motif Window Manager (mwm)
- Internet access



EMA Design Automation, Inc.
225 Tech Park Drive
Rochester, New York 14623

Phone: 585.334.6001
Fax: 585.334.6693
eMail: info@ema-eda.com
Web: www.ema-eda.com
www.timingdesigner.com