TestBencher Pro

The Next Generation in Test Bench Automation

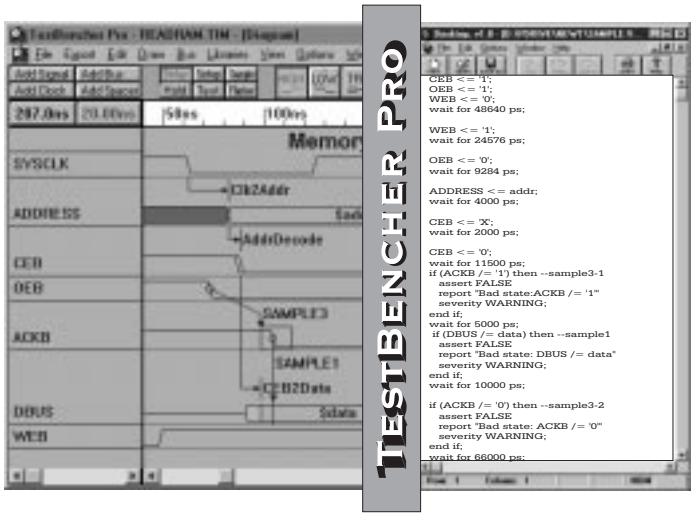
by SynaptiCAD

Blaze ahead of the competition with **TestBencher Pro**, the only customizable, reactive test bench generator!

Accelerate your design cycle to warp speed

Recent surveys of ASIC/FPGA designers have indicated that creation and verification of HDL test benches typically consumes over 30% of the entire ASIC/FPGA design cycle. **TestBencher Pro** drastically reduces test bench coding and verification time by automatically generating self-checking, HDL test benches from graphically-entered, reusable timing diagrams.

Generate VHDL and Verilog test benches from a single source



TestBencher Pro generates reactive Verilog and VHDL test benches from language-independent timing diagrams drawn by the user. The user creates a bench script that controls the sequence in which timing diagrams are applied during HDL simulation. Diagrams can also conditionally launch other diagrams based on simulation activity and can run in parallel and asynchronously with other diagrams.

Output signals on a timing diagram are translated into HDL signal assignment statements which drive the inputs to the simulation model under test. Variables can be used for state values of a signal, so that multiple test vector sequences can be generated from a single timing diagram by varying variable values of the timing diagram. This is especially useful for specifying bus transactions, where address and data values typically vary over multiple transactions.

Sample windows are placed on input signals to indicate expected condition checks that are to made on signals being output from the model under test. Conditional checks are expressed as if-then-else statements. A simple conditional check could test the value of a given signal over a given time range. More complex conditional checks can verify the occurrence and sequential order of state transitions on several signals at different points in time. Predefined and user-defined actions can conditionally be triggered in response to the success or failure of a conditional check.

Obtain complete control over your HDL code generation

TestBencher Pro is the only test bench generator with customizable HDL output. Other testbench generators allow direct entry of HDL code (TestBencher Pro does too), but direct entry of HDL code has two big weaknesses: (1) the code is specific to a given HDL language and (2) each use of an HDL code fragmented must be individually entered (code generation is not automated). TestBencher Pro users have direct access to the HDL code generation routines, so users can add their own routines (or modify existing routines) to generate HDL code that meets their own unique test bench requirements. Language independence can be achieved by writing separate code generation routines for VHDL and Verilog.

Another advantage of TestBencher Pro's customizability is the ability to specify custom mappings between values entered in a timing diagram and the HDL code generated for the diagram. A custom mapping could be used, for instance, to map between microprocessor instruction names in a timing diagram to opcodes in the HDL test bench.

Free evaluation copy available for download via Internet

Download a free working copy of **TestBencher Pro** from our web site (http://www.syncad.com). Our web site also contains technical papers and information on other EDA products offered by SynaptiCAD. Or contact **SynaptiCAD** directly at 1-800-804-7073. Don't waste another day - call today to shift to warp speed test generation!

web: http://www.syncad.com
phone: 800-804-7073 or 540-953-3390
fax: 540-953-3078
email: sales@syncad.com
mail: SynaptiCAD
PO Box 10608
Blacksburg VA 24062

TestBencher Pro Simplifies Reactive VHDL/Verilog Test Bench Generation

Surveys of ASIC/FPGA designers indicate that creation and verification of HDL test benches consumes over 30% of the entire ASIC/FPGA design cycle. This is not surprising since test bench code typically emulates a complex, system-specific environment for which few pre-written models exist. To simplify the creation of HDL test bench code, SynaptiCAD Inc. has released a new EDA tool, TestBencher Pro, that automatically generates HDL test benches from graphical specifications.

Most HDL test bench coding consists of writing bus-functional models. Code in HDL test benches generally falls into one of three categories: raw data (test patterns/RAM data), functional models which emulate the interface and internal computations of components, and bus-functional models (BFMs) which emulate a component's interface without emulating its full computational capabilities. BFMs like the kind generated by TestBencher Pro are often used instead of full models to model complex components such as microprocessors for three reasons: (1) BFMs are easier to code than full models, (2) component interface information is generally publicly available whereas information about internal component operation is usually vendor proprietary, and (3) they execute faster. Full functionality is generally not required in test bench components since it is the functioning of the model under test that is of interest.

Although BFM's are easier to create than full models, they still require considerable coding effort because they usually model very complex parts. A typical BFM might be used to model a microprocessor's interactions with an IO device or memory subsystem. This type of BFM would need to generate control and data signals from the microprocessor for each type of microprocessor bus transaction (e.g. read cycle, write cycle, interrupt processing) with exact timing requirements. BFMs also perform checks on outputs from the model under test so that simulation waveform results don't have to checked manually by designers. In addition, BFMs are typically reactive models; that is their output depends on the activity of the model under test. For example, a microprocessor BFM might have to wait for a data valid signal from a memory subsystem before completing a read cycle. Most BFMs are hand-coded in HDL from signal interface and timing information (timing diagrams) published by component vendors. This type of HDL code is difficult to create and maintain because it is difficult to visualize the waveforms generated by the HDL code and compare them to vendor timing diagrams.

TestBencher Pro generates reactive bus-functional models directly from timing diagrams. To simplify the task of creating BFMs, SynaptiCAD has introduced TestBencher Pro, the first customizable reactive test bench generator. Users create re-usable timing diagrams with TestBencher Pro's built-in timing diagram editor that describe the types of protocol transactions between the model under test (MUT) and the outside world. These timing diagrams contain stimulus to apply to the MUT and expected output data from the MUT. Signal states values can be specified as hard-coded Boolean or Hex values or they can be variables which are set to different values each time a diagram is executed. Users place "samples" on MUT outputs to control how the test bench reacts to output activity from the MUT. For example, a sample on an interrupt line could stop execution of a diagram and begin execution of an interrupt acknowledge diagram. Time markers can also be placed in a diagram to indicate areas of a diagram that should repeat or times at which user-written HDL sequences should be executed. The overall execution sequence of the timing diagrams is controlled by a bench script written by the user which contains calls to execute diagrams, loops, and conditional statements. Bench scripts can be written with a combination of native HDL and Perl code.

TestBencher generates small, efficient HDL code similar to manually coded test benches. This means the generated code is easy to understand and integrate with existing HDL code and doesn't suffer from code bloating problems that could slow down test bench execution. A TestBencher test bench consists of a component model for each timing diagram, a sequencer component generated from the primary bench script that controls diagram execution order, and a top-level structural component that connects test bench components and the model(s) under test.

TestBencher Pro can import/export waveforms from simulators and test equipment. TestBencher also supports import of waveforms from popular logic simulators and logic analyzers. These waveforms can be edited as desired in TestBencher and then used to generate HDL test bench code. TestBencher can also generate waveform stimulus to drive SPICE and many popular gate-level simulators. In addition, TestBencher Pro's built-in Perl scripting language allows users to import/export waveforms and other timing diagram data to any simulator or test equipment waveform format, including internal formats used by customers.

TestBencher Pro's test bench code generation is user-customizable. One unique aspect of TestBencher is that the Perl source code that generates HDL code from the timing diagrams is shipped with the product, giving end users the ability to customize HDL output to meet any specialized requirements for their testing environment.

To download a FREE functional evaluation of TestBencher Pro, visit SynaptiCAD at www.syncad.com. You can also get a free evaluation kit by contacting SynaptiCAD directly:

Sales: (800)804-7073 Email: sales@syncad.com
Phone: (540)953-3390 Web: http://www.syncad.com
Fax: (540)953-3078