

**SIMULATING WITH**

# **Xilinx Foundation Series**

**..... it's just so easy!**



## About This Document

This document has extensive cross referencing to pages in the online documentation. If you want to know how to do 'such and such' an operation, and haven't yet come across the menu option, have a look in here . It may point you to the appropriate page in the on-line docs.

Slide 4 outlines the basic simulation process, and is followed by an index to sections 1->4. Section 5 is a collection of slides that highlight features worth mentioning during a demonstration.

Page numbers to the online documentation are given throughout the presentation. The objective is to supplement the online documentation, not rewrite it!

-----

Foundation approaches simulation with the analogy of testing a printed circuit board.

∩

Each library element and user defined macro is considered as a 'chip', to which test signals from a 1000 channel, 100GHz signal generator can be applied. These signals are test vectors which can be created and applied to the circuit using some very quick and easy methods.

The output is monitored by a 1000 channel 100Ghz software logic analyzer, and operations such as Breakpoint analysis and step by step cross point probing with the schematic are implemented with ease.

To help grasp some ideas of the simulator, try to think of it as a PCB test environment.



## Documents referenced by this “Cheat Sheet”

Extensive reference is made to two sources of online documentation.

- 1) The Foundation online simulation document \ACTIVE\DOC\MANSIM.PDF
- 2) The online help file \ACTIVE\EXE\xlxguide.hlp

All page numbers refer to the MANSIM.PDF file. It will be useful to have this file loaded into the Adobe Acrobat reader whilst reading through this document.

To Load the MANSIM.PDF file.....

- o Press the  icon, in your Foundation Series program group. This will invoke Adobe Acrobat and load in the ACTIVE\DOC\ACTIVE.PDF file.
- o Select the Logic Synthesis Book.
- o To navigate directly to a page number, double click on the  icon at the bottom of the Acrobat window. This will bring up a window in which you can enter the page number.

To load the XLXGUIDE.PDF file.....

- o Invoke Windows File Manager
- o Navigate to \ACTIVE\EXE
- o Double click on xlxguide.hlp

## Contents

There are four basic steps to simulating your design with Foundation:

- 1) Creating a logic representation of your design.
- 2) Selecting test points for monitoring
- 3) Creating and applying design stimulus signals
- 4) Analyzing the simulation results.

The following index identifies features associated with each step above.

	<u>Slide No:</u>
1) <b>CREATING A LOGIC REPRESENTATION OF YOUR DESIGN</b>	6
- How do I build a simulation model of my design?	7
2) <b>SELECTING TEST POINTS FOR MONITORING</b>	8
- From the Schematic Editor	9
- From the Simulator Waveform Viewer	10
3) <b>CREATING AND APPLYING DESIGN STIMULUS SIGNALS</b>	11
- How do I define my test vectors?	12
- The Stimulus Selection window	13
- Software Counter ( <b>Virtual Stimulator</b> )	14
- Mapping stimulus signals to the <b>keyboard keys</b> .	15
- Describing test vectors using <b>formulae</b>	16
- <b>Drawing</b> waveforms to input as test vectors	17
- <b>Viewlogic</b> command files.	18
....example XC5200 Command file	20

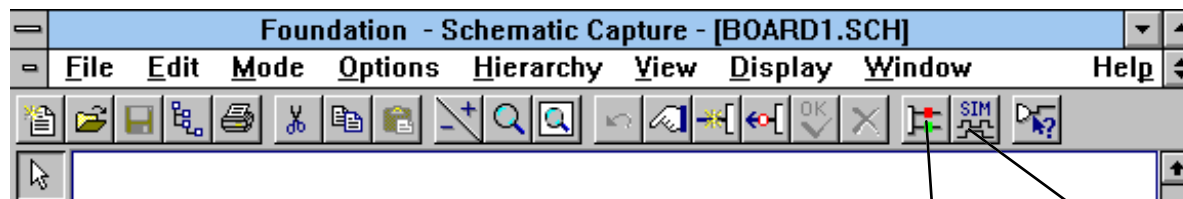
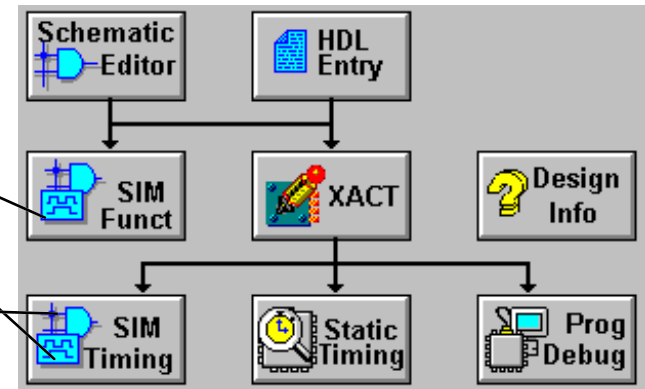
	<u>Slide No:</u>
4) <b>ANALYZING THE SIMULATION RESULTS</b>	21
- <b>Breakpoints</b> ..... What is one? When would I use one?	23
- How to create a Breakpoint	24
- How to use a Breakpoint	28
- <b>TAGs</b> ..... What is one? When would I use one?	29
- How to create a TAG	30
- How to use a TAG	31
- <b>Milestones</b> ..... What is one? When would I use one? How do I create one?	32
- <b>Presets</b> ..... What is one? When would I use one? How do I create one?	33
- <b>Selective Simulation</b> What is it?	35
- How to invoke selective simulation	36
- Cross Point Probing ..... What is it?	37
5) <b>MISCELLANEOUS</b>	
- What do I do if this presentation doesn't tell me what I want to know?	38
- Topics: - Signal Strengths in the simulator	39 & 40
- Simulator Precision	39
- Changing the simulation <b>short step</b> and <b>long step</b> duration	39 & 42
- <b>Creating a Bus</b> in the Waveform display window	39
- <b>Zooming</b> in and out of the waveform display	39
- Using multiple asynchronous clocks in a simulation	39
- <b>Simulating Tri-state</b> devices	39
- Simulation <b>Modes. Functional, Timing and Glitch</b>	39
- Locating simulation data on schematic sheets	39
- Locating components on schematic sheets	39
- Running a simulation for hours!	39 & 43
- Backing up data on long simulation runs	39
- Error reporting	39
- Resetting the design, what actually happens?	39
- Tracking Errors through a design netlist	39
- <b>Annotating measurements</b> and comments onto the waveform	39 & 44
- What do the <b>Simulation window Icons</b> do?	41 & 42

## Creating a Logic Representation of the Design

## Creating a Simulation Model to load into the simulator

(A) From the project manager window press either of these two buttons.

(If you do not have a back annotated XNF file, but have requested a timing simulation, XACT 6 is automatically launched so that the design can be implemented).



or

(B) Launch the simulator from the schematic editor by pressing this button.

Either (A) or (B) will launch the simulator.

Slide 41 is a quick reference chart to the various icons on the simulation window.

With or without probes? See slide 9

Select this button and then click on the net names/pads/nodes that you are interested in stimulating and monitoring. These will be automatically loaded into the simulator wave viewer window.

## Selecting Test Points For Monitoring



## Selecting Test Points For Monitoring

Loading the simulation window with the signals you wish to stimulate and monitor is as easy as pointing and clicking.



### WHY?

The Foundation Simulator and Schematic Editor are written with software that is OLE v2.0 compliant.


Object Linking Embedded (OLE) v2.0 is a Microsoft standard that defines how data can be linked between Windows applications. Any changes to the data caused by either application will be reflected in the other.

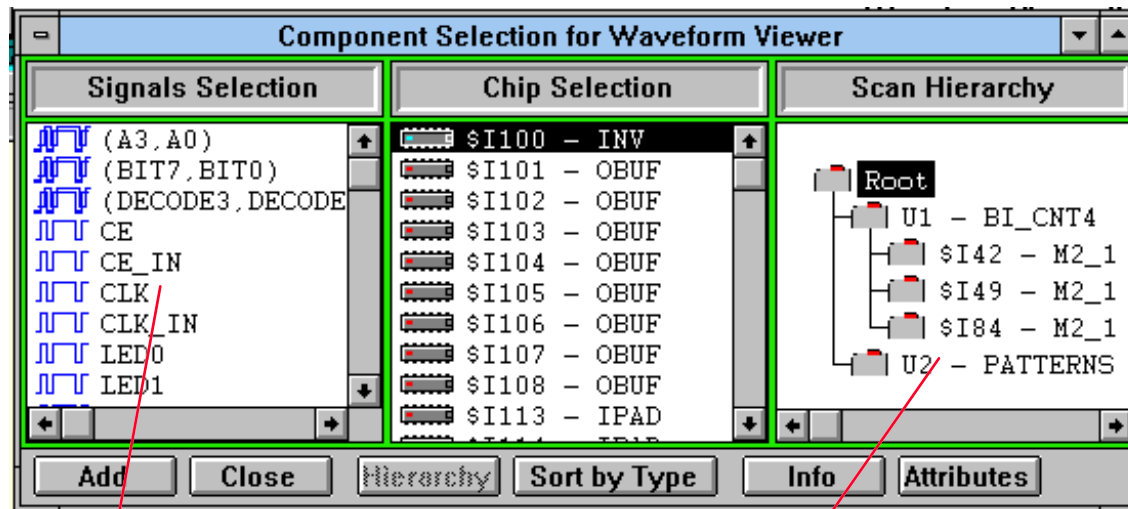
The Foundation Schematic Editor and Simulator use OLE to share a common view of selected signals. This means that signals can be added to the simulation window from the schematic editor or the simulator, and that the status of every signal in the simulator will always be back annotated to the Schematic Editor. This cross probing is a VERY useful feature.

### HOW?

- From the schematic window tool bar, use this icon, (  ), to select the probe tool. Now click on the pads/net names/nodes that you want to display in the simulation window. A gray box will appear next to the selected item. Launch the simulator using this icon(  ). The waveform viewer now appears with all probed signals loaded into it.
- If it is necessary to view additional signals, these can be added from the simulator and a gray 'probe' box will appear on the schematic! ...see next slide.

## Loading the Waveform Viewer With Signals From the Netlist

Click on the (  ) icon to launch the netlist viewer below. (see slide 41).



This viewer shows a list of nets and buses accessible to the simulator.

Note that as you navigate the design hierarchy in **this** window, the available signals change in **this** window. To load a net into the simulator wave display, double click on a signal. A red tick will appear on the signal to indicate that it is now in the waveform viewer. Alternatively select a group of signals using *shift* click and drag, then press the 'Add' button.

The middle window shows all of the library elements at the current level of hierarchy. Note that chip icons are used to reinforce the idea of the simulation model being a PCB!  
(See pg. 106)

## Creating and Applying Design Stimulus Signals

## Creating and Applying Design Stimulus Signals

This can be the most tedious of all simulation tasks due to the effort required in defining all of the test vectors used to emulate real world inputs.

Most PC based Xilinx customers will approach the Foundation simulator with experience of OrCAD or Viewlogic simulation techniques. The principles are still valid with Foundation, but this software provides a number of methods to accelerate the simulation and debugging process.

Before the design outputs are analyzed, input test vectors need to be defined. Foundation provides 5 ways of interactively applying stimulus to the design. The waveform results of these stimuli executed over a number of clock cycles can be saved as a test vector file. These test vector files can then be used in more complex simulation processes when invoked by COMMAND FILES and BREAKPOINTS (slide 23).

Each of the following is a quick and easy way to stimulate the inputs of your design.

- 1) Application of ready made test vectors using the virtual stimulator (pg. 117) & (pg. 19) ( *slide 14*).  
This is a 16 bit software-driven binary counter that counts at a preset clock rate. It is possible to assign any true/compliment bit of this counter to any schematic test point so enabling the automatic toggling of the selected signal line at a fixed rate.
- 2) Toggling signal lines with keyboard keys (pg. 117) & (pg. 21) ( *slide 15*).  
Signals in the waveform viewer can be mapped to keys on the keyboard so enabling real time user control.
- 3) Formulae (pg. 120) & (pg. 23) ( *slide 16*).  
16 signals in the stimulator selection tool can each be assigned a formula. These can then be mapped to any signal in the waveform viewer.
- 4) Drawing test vector waveforms (pg. 118) ( *slide 17*).  
An editor permits waveforms to be drawn in the waveform viewer window.
- 5) Using an existing Viewlogic command file (pg. 189). ( *slide 18*)  
The Foundation Simulator will read ViewSim .CMD files because the command language syntax is the same.



# The Stimulus Selection Window

This window is used to control how each stimulus signal is created. To invoke the Stimulus Selector, click on this icon, (  ), from the Simulation Waveform viewer window.

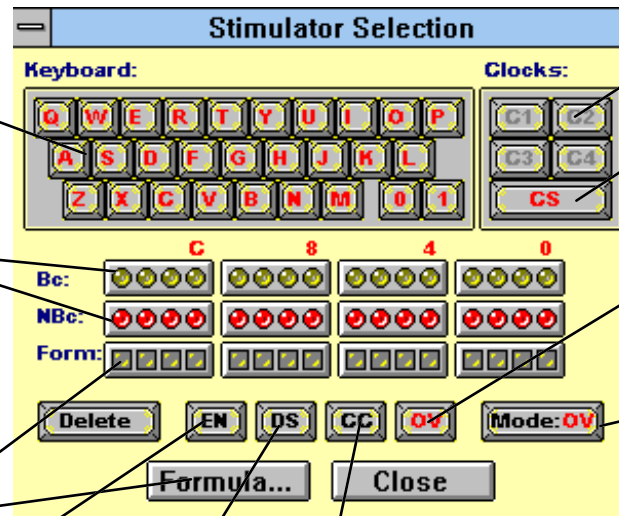
Assign keyboard keys nodes in your schematic (slide 15)

Constantly running 16 bit counter. True and Compliment outputs available (slide 14)

Test vectors defined by formula can be assigned to these LED's, and in turn assigned to test signals (slide 16)

Enables a previously disabled signal stimulator. pg. 116

Disconnects the assigned stimulator without deleting it. pg. 116



asynchronous clocks defined and assigned using these buttons.

Forces an existing signal waveform to act as an input waveform

when assigned to an output it will override the chip generated function.

This button displays the current default stimulation mode. Click on it and it will toggle between 0V and CC (pg.116)

When simulating a bi-directional I/O, a test vector will be applied to the net connecting the IOPAD output -> IBUF input->OBUFT output. By asserting a test vector to drive the IBUF input net, the vector signal strength will override the OBUFT output, even when it is enabled.

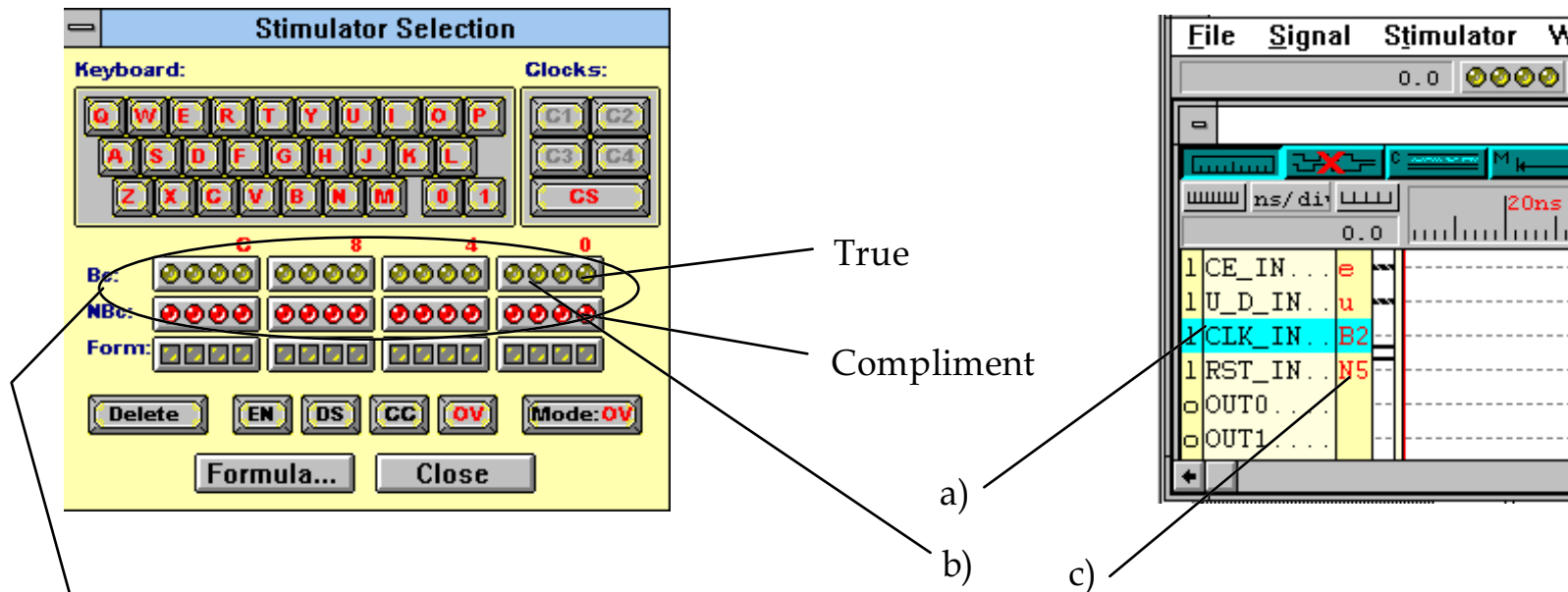
Applying this CC key to the test vector that is driving the OBUFT output, will inform the simulator that when the OBUFT component is enabled, it can drive the net with the same signal strength as the test vector. When the OBUFT is not enabled, the test vector can override the weak Hi-Z OBUFT output state.

So, **Use this key when simulating Three-State Functions.**

For more detailed description, refer to the on-line help file.

-> Windows File manager, double click on <active/exe/xlxguide.hlp> navigate to [Functional Simulation](#)-> [Overriding device pins](#)

## The Stimulator Selection Window : Mapping Signals to The Counter

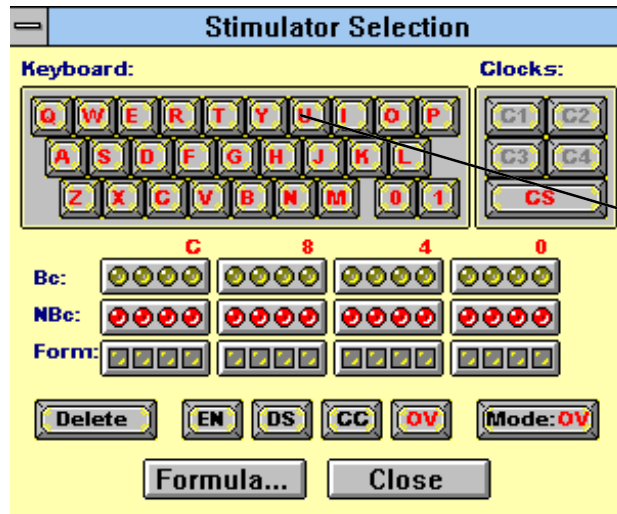


These 32 LED's represent the true and compliment outputs of a software driven 16 bit binary counter. To quickly assign a toggling stimulus signal to any node in the design:-

- Click on the signal to be stimulated from the waveform viewer (it turns blue)
- Click on the counter output you wish to assign to it.
- Note that the column next to the signal name identifies the source of stimulus.
- Alternatively click on the LED in the stimulator window and with the left mouse button still depressed, drag and drop it onto a signal in the waveform viewer.

The above example shows that signal CLK\_IN is stimulated by B2 and RST\_IN is stimulated by the compliment of bit 5. (See pg. 117 for further details)

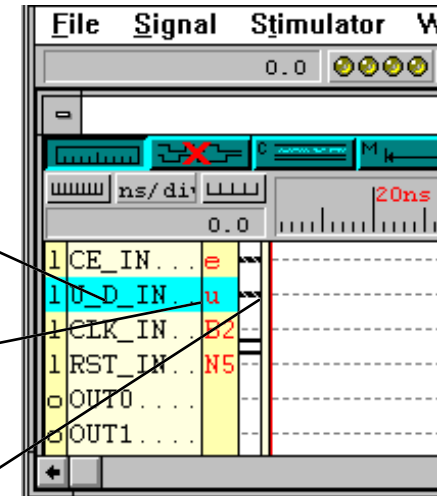
## The Stimulator Selection Window : Mapping Signals to The Keyboard



a)

c)

d)

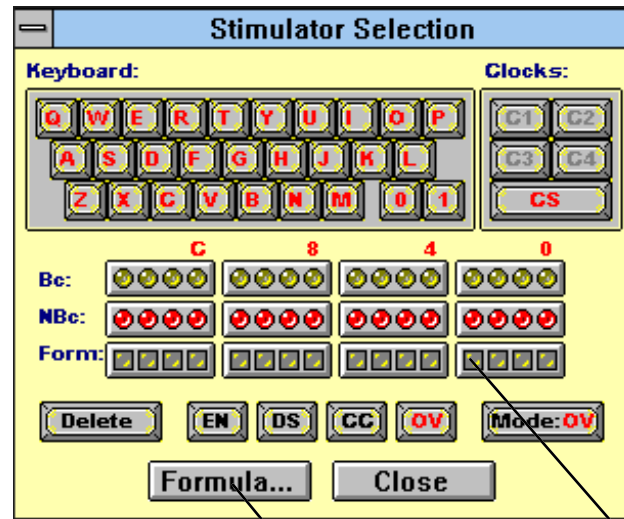


This feature makes full use on the interactive nature of the Foundation software.

Signals selected in the waveform viewer window can be associated with a keyboard key, and then toggled whilst the simulator is being stepped through a set of test vectors.

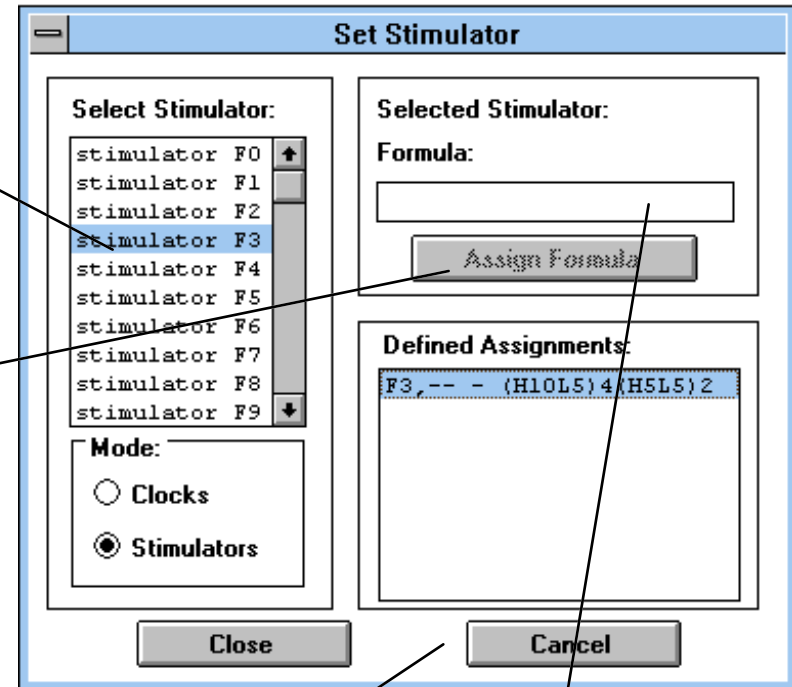
- a) Click on a signal in the waveform viewer window, (turns blue).
- b) Click on a keyboard key in the Stimulator selection window.
- c) The column next to the signal name shows that the selected keyboard key is now mapped to that signal.
- d) Alternatively click on the key in the stimulator window and with the left mouse button still depressed, drag and drop it onto a signal in the waveform viewer.
- e) This column displays the status of the stimulus input. Immediately after making the allocation, the signal is assigned a Hi-Z state. You will need to press the key once or twice to set a high strength '1' or '0' status.

## The Stimulator Selection Window : Applying Formulas



a)

d)



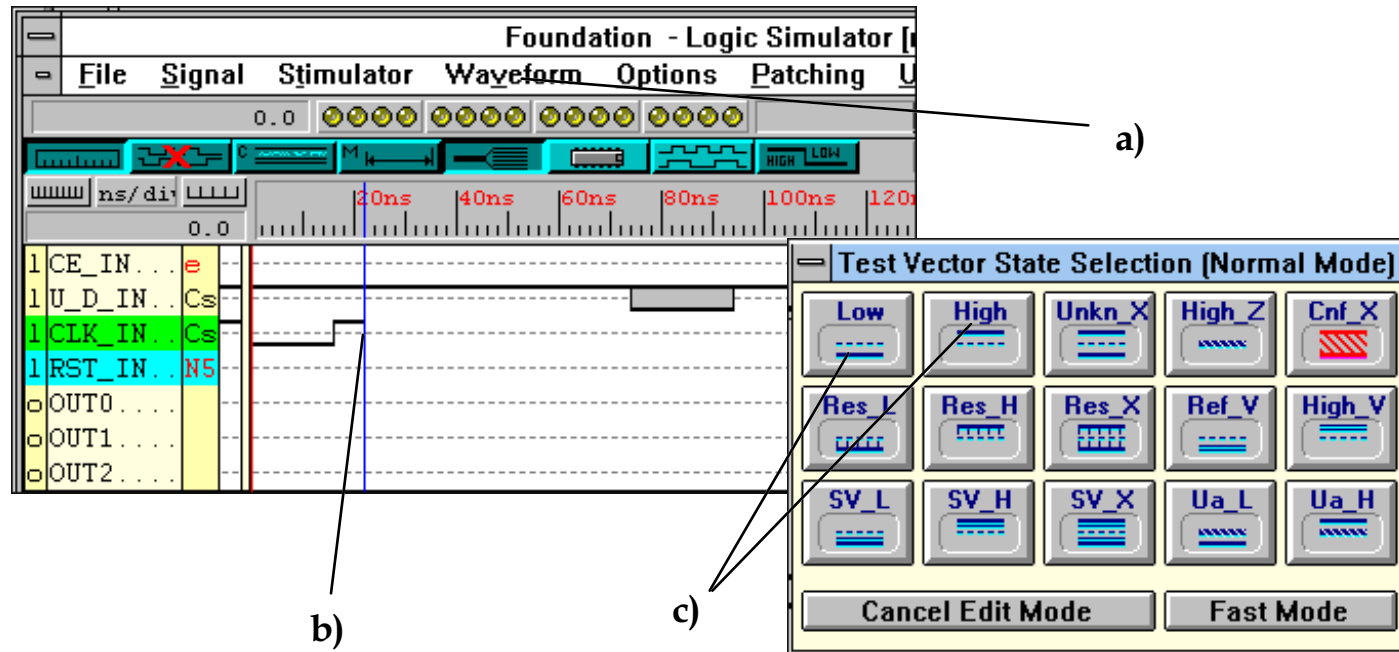
b)

c)

- To define a formula, press this button to give this window.
- Double click on one of the 16 stimulator labels.  
(The label will turn blue when selected and a cursor will appear in the formula window).
- Enter the formula and then press Assign Formula. That formula is now mapped to one of the 16 formula buttons in the Stimulator Selection Window.
- Assign a formula to a signal in the Waveform viewer window by clicking on a signal, and then clicking on the desired formula button. (See pg. 120->122)



## The Stimulator Selection Window : Drawing Waveforms



- a) From the Simulator Waveform Viewer, select EDIT from the WAVEFORM menu. This will bring up the Test Vector State Selection window.
- b) Select a waveform and position the blue cursor to where you want the next waveform transition to take place
- c) Select the state you would like the waveform to assume up to that point, i.e. High, Low, Unknown, etc.
- d) Continue to select and draw other input vectors. (See pg. 118 to 120)

## Command Files pg. 189

### What is a Command File?

A Command file is the equivalent of a DOS batch file in that it will execute a list of instructions in sequence. In the Foundation environment the command file has a .CMD extension, and each instruction is known as a macro. A list of macros supported by the Foundation simulator is discussed on pages 189->227 of the MANSIM.PDF file.

Complex command files can be used to setup the simulator, define clock periods and clocks, load and save test vector files and check simulation output against known timing files.

In essence, command files are simply a way of driving the simulator in an automated fashion.

It is not essential to use a command file in order to drive the Foundation simulator, but this method of input is supported because it is an excellent way of scripting a complex simulation process.

### Can I use my existing Viewlogic Command File?

YES. Foundation supports the command file syntax used by Viewsim. Consequently most Viewlogic '.CMD' file statements will be accepted by the Foundation simulator.

Be aware that some Viewsim macros are not supported by the Foundation simulator.

To determine if your Viewsim file contains any macros that are not supported by Foundation, check pages 195->217.

... more next slide



## Command Files

### How do I run a command file?

- 1) Copy an existing command file into your current working design directory.
- 2) You must now inform the Foundation environment that there is a .CMD file available.  
There are two ways to do this;

Either

a) Link the .CMD to the project as described by the bottom four bullet points of page 193.

or

b) Select *Utilities -> Macro -> Edit -> Browse ->* select desired .CMD file, then save it.

- 3) To run the command file select *Utilities -> Macro -> Run ->* select .CMD file.

..... a window will pop up and display the .CMD file as it is being executed. Any errors or nodes that could not be found, will be displayed in this window.

TIP! It is possible to create waveforms using any of the methods defined on slide X.

To turn these waveforms into stimulus test vectors that can be called from a Command file, do the following.

Select *File -> Save Test vectors.... ->* Enter a name -> *Save.*

In your Command file, use the macro "load\_timing      *test\_vector\_name*" to load in the vectors.  
(see pg. 224 for a description of this macro). However, note that once you have run a command file, any signal controlled by the command file will have to be reassigned using the Stimulus Window, (slide 13), if you wish to re-engage say keyboard control of the signal.

# Example Command File

File = ACTIVE\PROJECTS\CALC5K.CMD

```
| Filename calc_fun.cmd. -PGH
| The reset sequence is different from the 3k version.
| This is a sample Viewsim command file. It steps through
| the sequence of commands used in the Calc Viewsim
| Tutorial.
| You can create a command file with any text editor, and
| run it in Viewsim simply by typing the name of the
| file. The default extension is .cmd.
```

```
| Echo the command file to the screen, display time
| Re-initialize the circuit
restart
```

```
|-----VECTOR DEFINITION-----
| This file uses the convention that vector names are
| capitalized; this is just a convenience.
| Use the name of the net between the pad and the buffer
| to drive the input at the pad. Net names at lower
| levels of hierarchy are in the format
| ...block_label\block_label\net_label.
| Use "+" to continue a long line
vector SW sw1/sw6_p sw1/sw5_p sw1/sw4_p sw1/sw3_p +
sw1/sw2_p sw1/sw1_p sw1/sw0_p
```

```
| You can also use bus syntax when defining vectors
vector ALU alu[3:0]
vector STACK stack[3:0]
```

```
| Set radices for vectors
| The default radix is binary for input, hex for output
radix hex SW ALU
radix bin STACK
```

```
|-----SIMULATION OUTPUT DEFINITION-----
|-----
```

```
| Select the nets and vectors for display in Viewwave. If
| bus syntax is used for nets, nets are still displayed
| individually. If the vector name is used, the value
| of the whole vector is displayed.
wave scale.wfm clk SW exc_p ALU STACK we rst
```

```
| Save simulation values for these nodes
watch clk SW exc_p ALU STACK we rst
```

```
| Output the values of all watched signals each time
| "clk" goes high. Create tabular output.
break clk 1 do (print > scale.tab)
```

```
| Output node and vector transitions and simulation time
| whenever any of the nodes or vectors changes state
trace clk SW exc_p ALU STACK we rst > scale.trc
```

```
|-----CLOCK DEFINITION-----
clock clk 0 1
| Use a clock period of 100ns. Set stepsize=50ns
step 50ns

|-----GLOBAL RESET & INITIAL INPUT VALUES-----
| Set initial values for all inputs using the "H" and "L"
| commands for nets and "assign" for vectors
h exc_p
assign SW 00\h

| Initialize all flip-flops (globalreset is active high
| for 5k designs)
l reset
| Viewsim uses units of 0.1 ns, so this statement
| simulates for 100 ns.
sim 1000
h reset
sim 1000

|-----TESTING THE COMMANDS-----
| Exp_p going high is the execute command. OPCODE is a
| three-digit opcode which operates on the DATA. Opcode
| 111 activates an extended instruction set which uses
| the data field as an extended opcode field, as external
| data is not required to execute these commands.

| Execute opcode 110 to Load data of 1 to ALU Register
assign SW 61\h
| Set exc_p low with the "L" command
l exc_p
| The cycle command steps forward by one clock cycle
cycle 2
| Execute
| Set exc_p high with the "H" command
h exc_p
cycle 3
| Results: SW=61\h => ALU=1\h STACK=0000

| Opcode 000: Add 13 from Switches to ALU Register
assign SW 0D\h
l exc_p
| You can abbreviate cycle as "c"
c 2
| Execute
h exc_p
c 3
| Results: SW=0D\h => ALU=E\h STACK=0000

| Opcode 111-101x: Push Register value onto Stack
assign SW 7B\h
l exc_p
c 2
| Execute
```

## Analyzing The Simulation Results

## Analyzing Simulation Results

Having captured the schematic, invoked the simulator, setup the clock periods and created test vectors, all that remains is to analyze the waveforms. The designer will know what the outputs should do for a given set of test vector inputs, but often the difficulty lies in trying to find out where events occurred, and why.

This next section will cover some of the tools available for debugging with the simulator.

These include;

- Breakpoints
- Tags
- Milestones
- Presets
  
- Selective Simulation

The Simulator also has a method of speeding up the analysis of large designs by using a process of selective simulation. Components of the design are selectively disabled by the user, so enabling the simulator to process the reduced design faster.

- Cross Point Probing With The Schematic.

Simulation data is automatically back annotated to the schematic during a simulation run.

## **BREAKPOINTS**

### **What is a Breakpoint?** (pg. 228)

A Breakpoint is a software routine that checks for selected signal conditions in the design.

When these conditions are met, you have reached a breakpoint condition and may perform the following design operations:

- Stop simulation
- Place a marker on the screen
- Save test vectors
- Load a new test vector file.
- Modify the existing test vectors using the Append operation.

### **When Would I use a Breakpoint?**

Breakpoints are very handy in tracking major design activities and error conditions. They are used extensively both in hardware tools, (logic analyzers hardware emulators), and also in software development systems.

Lets say for example that you want to detect a certain value on a bus or selected bits on a bus. Breakpoint conditions could be defined for each bit of the bus such that when all bits assumed their respective conditions, the bus breakpoint condition has been met and a marker could be dropped onto the waveform display.

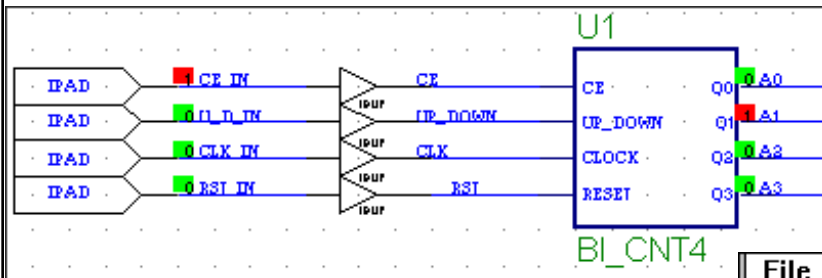
### **How Do I invoke a Breakpoint?**

See the next slide, for a description of defining a Breakpoint for a bus.

A more complete description can be obtained from the MANSIM.PDF document (pg. 228).

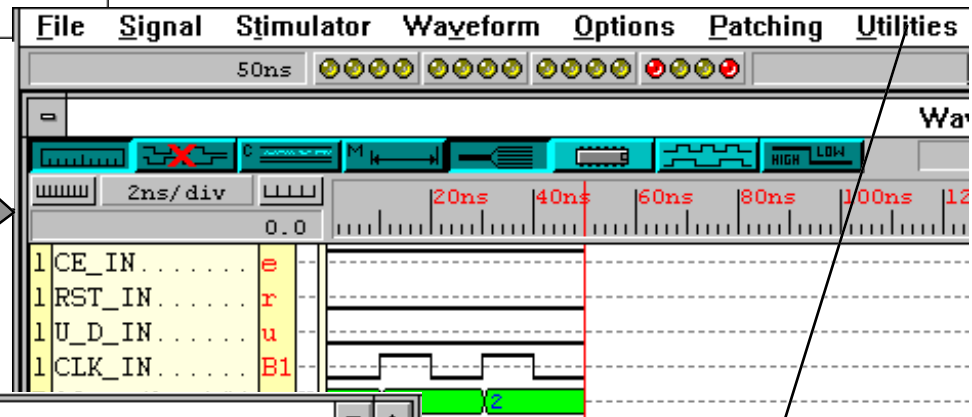


## How to Create A Breakpoint - 1



1) Assign probes to schematic

2) Invoke simulator.



Breakpoint Conditions																
Signals	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
1 CE_IN .....																
1 RST_IN .....																
1 U_D_IN .....																
1 CLK_IN .....																
1 A3 .....	+															
1 A2 .....	+															
1 A1 .....	+															
1 A0 .....	*															

3) Select

Utilities -> Breakpoints

see pg. 229



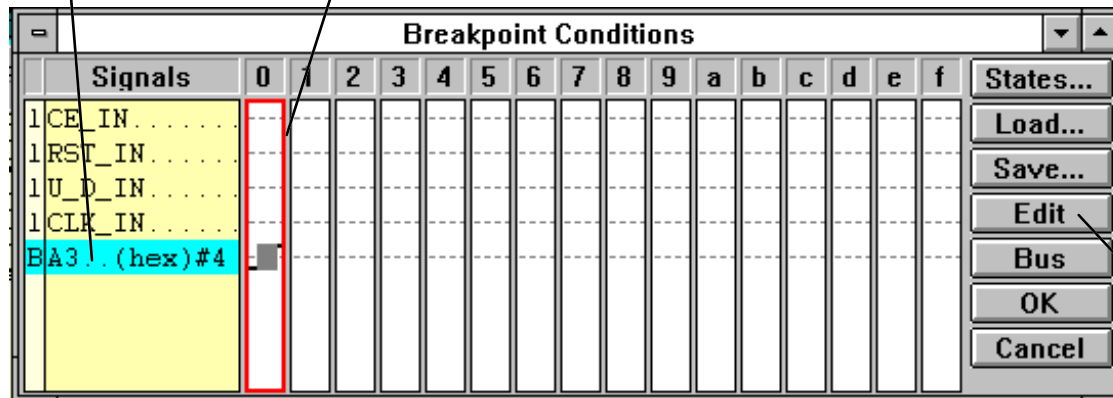
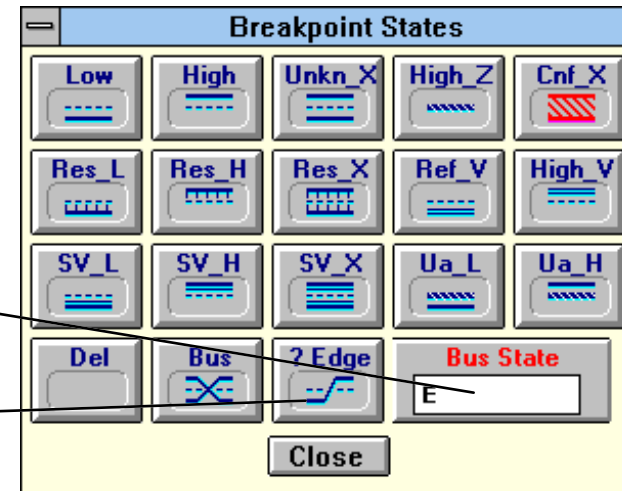
## How to Create A Breakpoint - 2

4) Select the bus (pg. 231)

5) Enter value on bus that you wish to detect. (Must be in Hex)

6) To detect the transition to this value, select Edge.

7) You have now declared the 'mask' that defines 'Condition 0'.



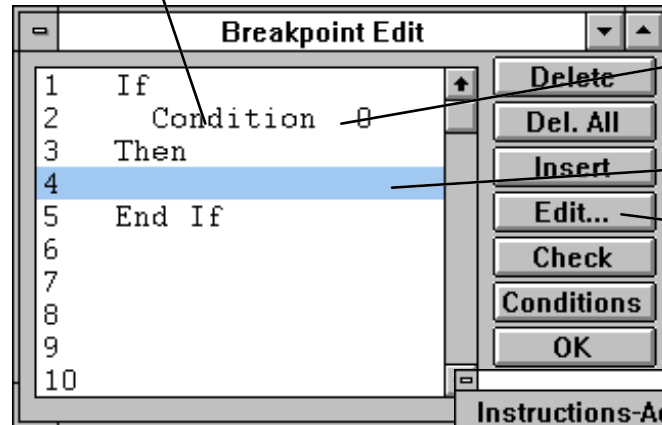
8) Having defined a 'Condition', you must now define the action to be taken by the simulator when that Condition is met.

Click on Edit

... see next slide

## How to Create A Breakpoint - 3

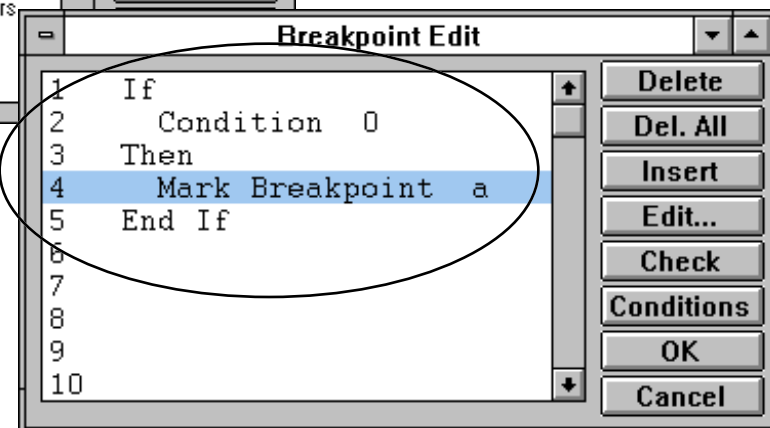
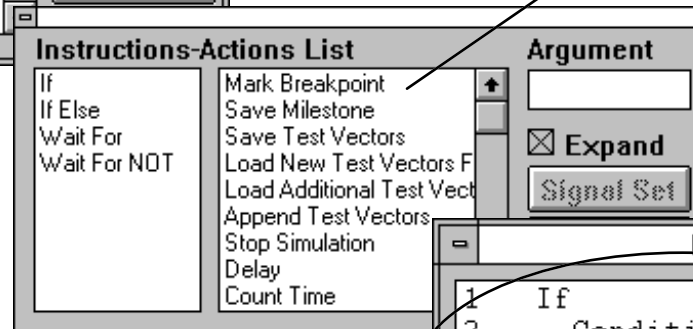
The 'Breakpoint Edit' Window contains Constructs , ( e.g. If-Then-Else ), which refer to Conditions , (0 -> F), for which Actions are specified when they occur.



9) Ensure that this is the Condition you want to edit.

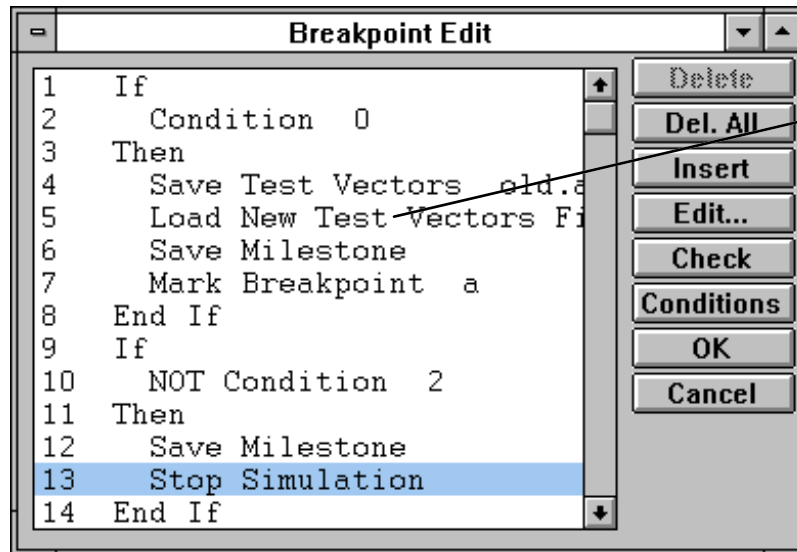
10) Select a blank line into which an action is to be defined.

11) Click on Edit and then select an 'Action'.



12) The Breakpoint Action is now defined

## How to Create A Breakpoint - 4



13) Note that multiple Actions can be defined for each construct,

.... and that multiple constructs can be used in the Breakpoint file.

Each 'If-Then-Else' construct is executed in the sequence in which it is listed in the file.

14) To add extra constructs, double click on an empty line and make a selection from this box.



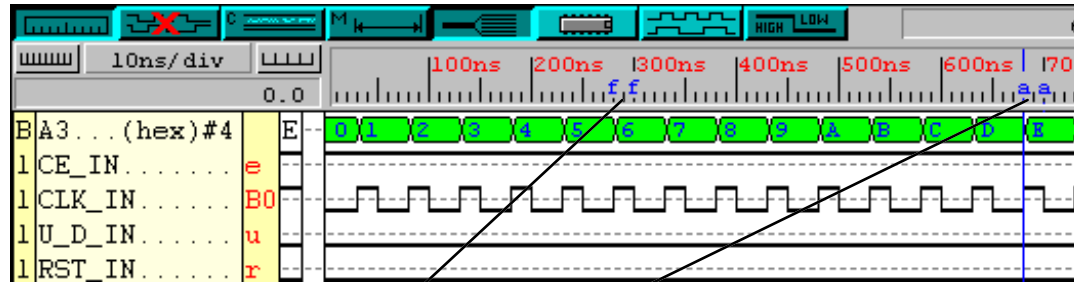
Each Breakpoint file can monitor the simulation output looking for the true and compliment of up to 16 Conditions.

If any one of these conditions occur, the simulator can be forced to execute several actions which include modifying the current test vector file, saving the status of all signals and halting the simulation, etc.

This feature is a powerful debugging tool... and it is easy to drive!!

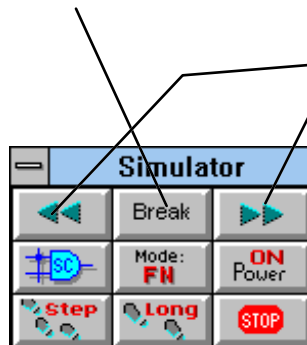
Refer to pages 228 -> 235 for more detail.

## How to use a Breakpoint



- 1) Load the edited Breakpoint file using *Utilities -> Breakpoint -> Load -> filename*
- 2) Run the simulation. If any Breakpoint conditions occur, blue letter markers will be placed on the display like this. These letters are the arguments that were assigned to the Set Marker 'Action' defined when editing the Breakpoint file.

To quickly locate Breakpoints in a large waveform display, ensure that this button is set to 'Break', .....



..... and use these buttons to jump left and right to the next Breakpoint in that location.

.... try it!!

## Tags      pg. 139

### What is a TAG?

A TAG is a combination or sequence of signals

### When is a TAG used?

Finding a desired signal combination by manually scrolling through the waveform diagram can be tedious. Using TAGs is a quick and easy way of automatically searching for a particular event through the whole waveform diagram.

### What is the difference between a TAG and a Breakpoint?

BREAKPOINTS operate whilst a simulation is running. They are software routines that monitor the simulation outputs looking for a particular condition. When that condition is met, they react as instructed.

TAGS are employed whilst the simulation is stopped, and are a quick and easy way of scrolling through the waveform looking for a particular condition.

### How do I create a TAG?

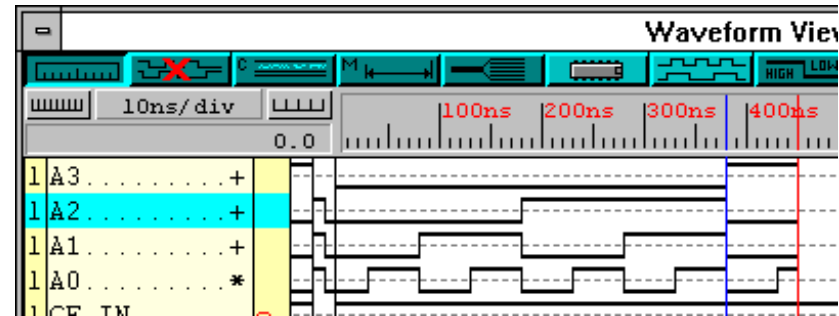
see next slide and also pg. 139.

## How to Create a TAG (pg 139)

How do I create a TAG?

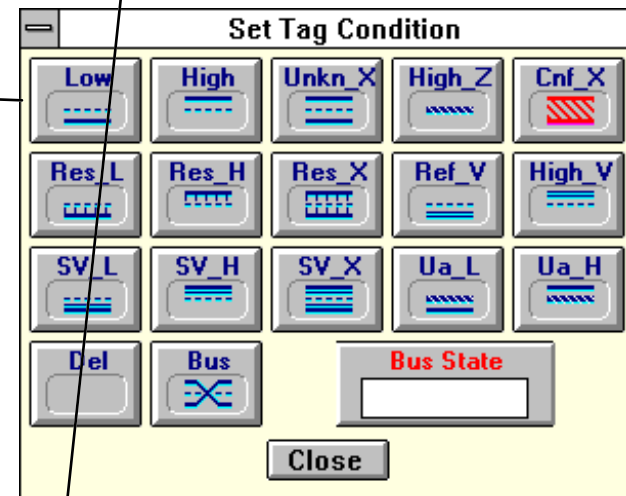
- 1) Select *Utilities -> View -> TAG*

This enables a second column next to the signal names. This column shows the state or transition that each signal contributes to the TAG.



- 2) Now select *Options -> Set TAG Conditions*.  
A state editor window is enabled.

- 3) Select a signal from the Waveform Viewer window and click on a state in the state Editor Window.

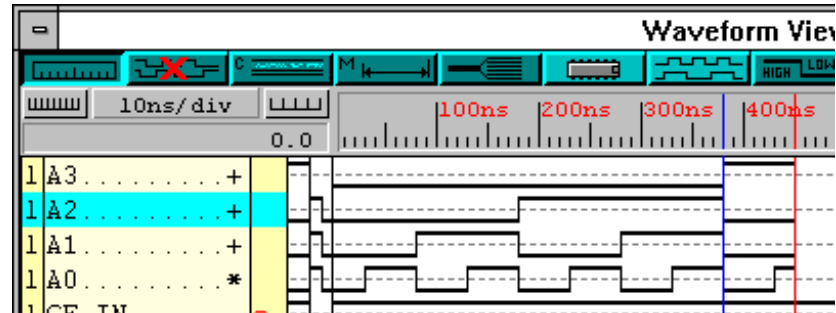


- 4) To implement a transition, say '1'->'0',
- select the signal
  - select the High state button,
  - select the same signal,
  - then select the Low state button.

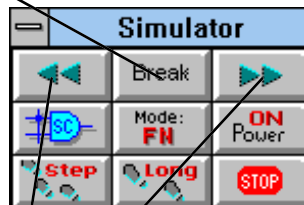
The TAG status column should now display a '1'->'0' transition.

## How to Use a TAG pg. 139

How do I find a TAG?



- 1) To quickly find a TAG during any stage of a simulation, click on this button to until it shows TAG,



Note that the current time of the simulation is marked in red, and the TAG cursor that scrolls around the screen is in blue.

- 2) ...and use these buttons to jump left or right between TAG'd events.  
The screen will jump to a blue cursor that marks the TAG'd event.

## **Milestones** *(pg. 152)*

### What is a milestone?

A milestone is used to store the complete design status that exists at a selected simulation cycle. This includes the timing at the current simulation cycle, and all internal registers, memories and flags etc., of all IC models.

### When would I use one?

Milestones allow extensive design analysis. You can re-simulate a modified design from a selected milestone, and request the simulator to automatically display the effect of the latest design or test vector changes on the selected signal.

It is also convenient to re-simulate the design from a certain cycle with modified test vectors, new timing delays or other design changes.

### How do I create a milestone?

*see page 153.*



## Presets pg. 127

### What is a Preset?

The simulator allows any signal or device pin to be preset to any logical state.

### When would I use a Preset?

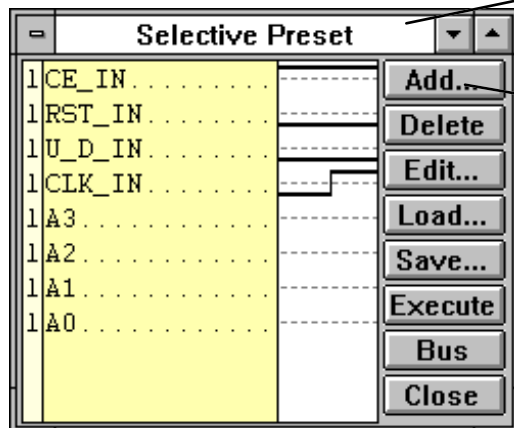
Preset conditions can be executed at any time during the simulation. Their main use is to test for some design situation that is difficult to generate, or for a design situation that takes a very long simulation time to create.

e.g., the end of count sequence on a 64 bit counter takes... a large number of cycles!

An alternative is to set preset the counter to FFFFFFFF0h

### How do I define a Preset?

Select *Utilities -> Selective Preset*



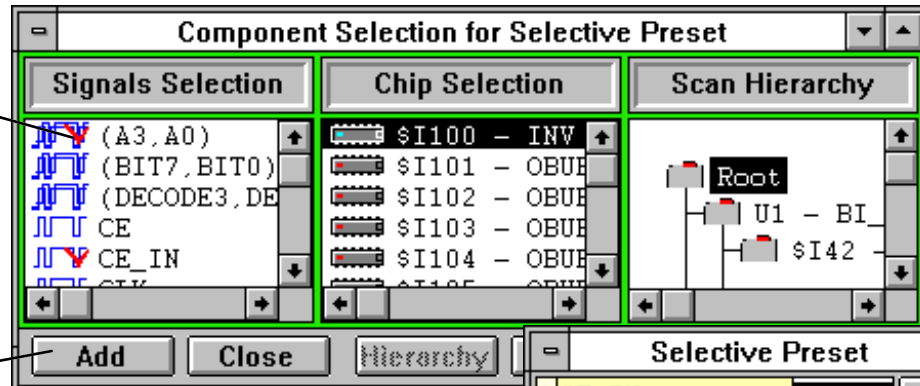
This window pops up and you define the presets as follows.

1) Click on Add in the 'Selective Preset' window

...see next slide.

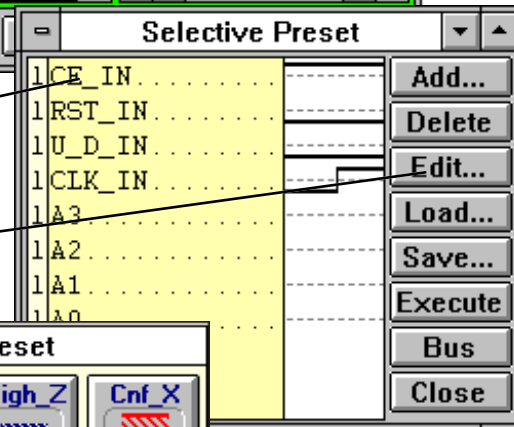
## Presets

2) Highlight nodes in this window,



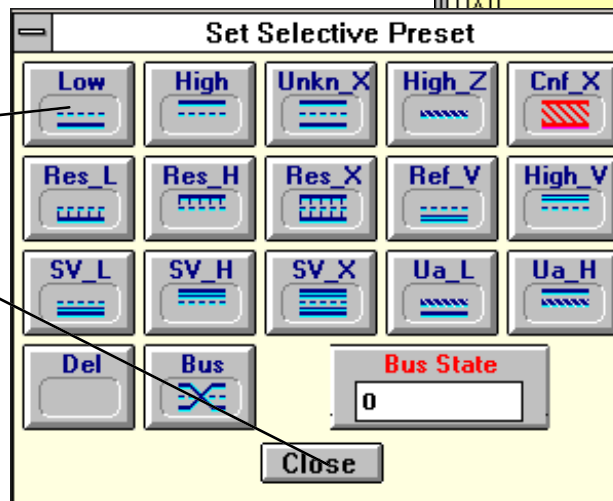
3) Click on Add

4) Nodes transferred to here



5) Select Edit

6) Enter state for each signal.  
Then select close.



7) Use Save and Load to choose between Preset groups.

8) Use Execute to run the preset.

see pg. 127 to 129  
for further details

## **Selective Simulation (pg. 56)**

Consider what a simulator does.

It operates on a model of the design and tries to calculate what all of the nodes are doing at any given moment in time for a given set of inputs. Thus it follows that the more nodes that exist in the design, the more time it takes for the simulator to complete one step of the simulation.

Running a large design for a long simulation period can produce... long simulation times!

More often than not, the simulation is only being carried out to exercise a portion of the overall design, (e.g. the design is targeted at a XC4013 and just the ABEL state machine needs to be tested.

Why should the simulator engage in calculating the status of nodes that are of no interest during this particular simulation run?

The answer is that it shouldn't. The Foundation simulator addresses this situation with a process called selective simulation (pg. 56)

Selective simulation allows the user to search through the design netlist clicking on icons that represent hierarchical blocks in the design. These parts of the design are closed down in the simulation model and their outputs forced to a Hi-Z state. Reducing the number of 'active' nodes in this manner results in faster simulation times.

The process of closing down areas of the design simulation model is no more difficult to achieve than selecting a file from the Windows File Manager!

...see next slide



## Invoking Selective Simulation

- 1) Select Waveform Viewer -> *Options* -> *Selective Simulation*.

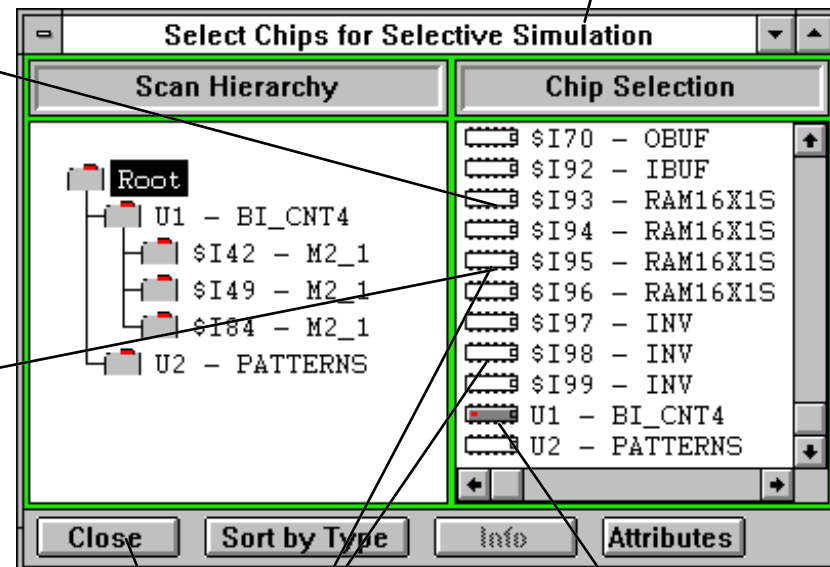
.....this window pops up

Note that in any given hierarchical level, the components are symbolized as 'chips'.

- 2) To disable a component, click on the chip icon, and it will change color from gray to white.

- 3) The outputs of selected components now assume a Hi-Z state in the simulation model, and this is reflected on the schematic.

- 4) click on close ...**it's as easy as that!**



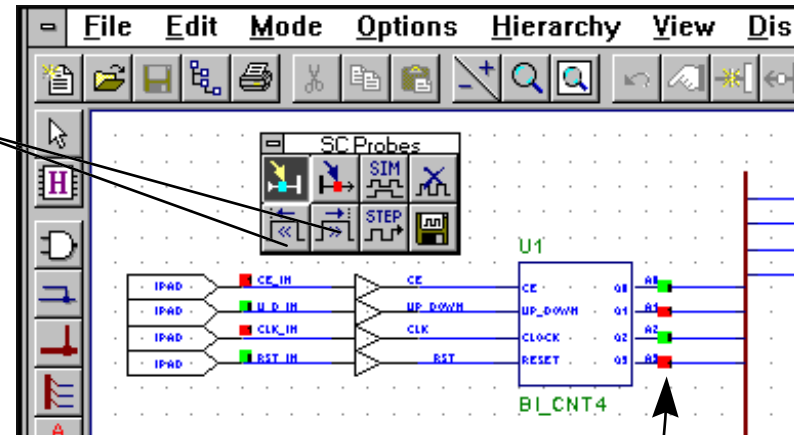
disabled

active.

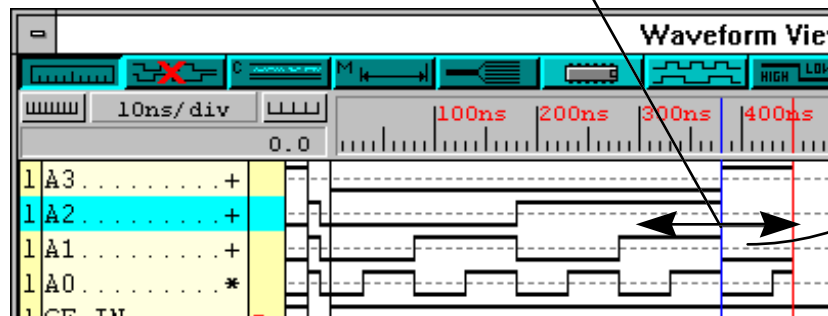
## Cross Point Probing

Due to the Foundation software's OLE v2.0 compliance, the Schematic Editor and simulator tool are very tightly coupled. This enables cross point probing. The status of all probed signals will be back annotated to the schematic window and the various logic states, represented with different colors.

Using these buttons, one can advance or retard the simulator through one clock transition. The Simulation cursor will scroll along the Waveform viewer window, and the annotated signal states, will change color with each transition.



Similarly, scrolling the waveform viewer cursor along the waveform, will back annotate the individual signal states to the schematic display.



## Miscellaneous

The remainder of this presentation covers miscellaneous aspects of the simulator operation

- i.e. - How do I zoom in and out of the waveform display?  
- How do I set the clock step size?  
- What do all the icons in the waveform display window do?

### What do I do if this presentation doesn't have what I want?

If you do not find what you are looking for in this presentation, load the MANSIM.PDF file into the acrobat reader, ensure this button is selected, and select the Chapter 1 triangle.

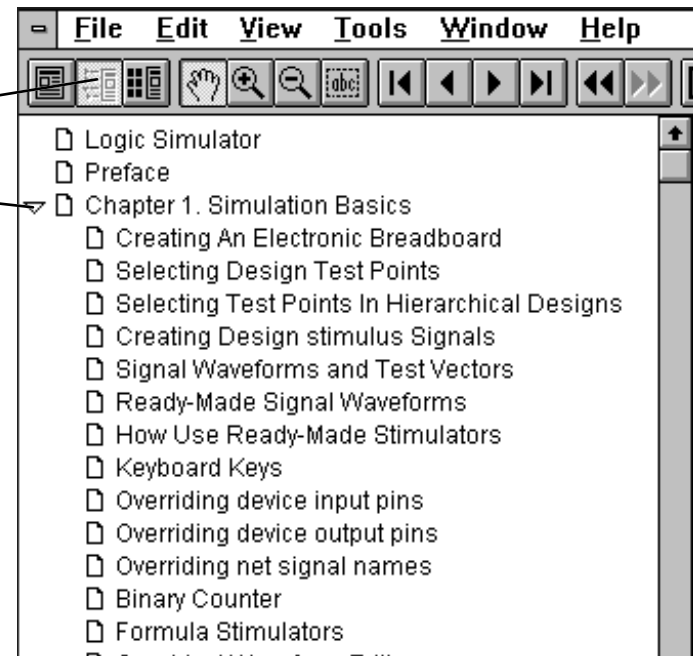
A long list of topics is presented from which it is easy to find the subject you are looking for.

Alternatively run the following program by double clicking on it in the Windows File Manager.

`\ACTIVE\EXE\xlxguide.hlp`

This is the Xilinx Foundation Interface guide

This has many sections specific to Xilinx users.


















- Reference documents :
- o A = MANSIM.PDF
  - o B = /ACTIVE/EXE/xlxguide.pdf

## Topics

Signal strengths .....	A (pg. 137) & slide 40
Simulator Precision .....	A (pg. 46)
Changing the clock 'short' and 'long' step size .....	A (pg. 125) & slide 42
Creating a bus in the waveform display window .....	A (pg. 38)
Zooming in and out of the waveform display .....	A (pg. 88)
Using multiple clocks in a simulation .....	A (pg. 74)
Simulating Three-state devices	B -> <a href="#">Functional Simulation</a> -> <a href="#">Simulating tri-state signals</a>
Simulation Modes      Functional .....	A (pg. 30) & ref. slide 42
Glitch .....	A (pg. 33) & ref. slide 42
Timing .....	A (pg. 35) & ref. slide 42
Locating Simulating data on schematic sheets .....	A (pg. 42)
Locating components on schematic sheets .....	A (pg. 42)
Running a simulation for hours! .....	A (pg. 47) & slide 43
Backing up data on long simulation runs .....	A (pg. 48)
Error reporting .....	A (pg. 49 & 145)
Resetting the design, what actually happens? .....	A (pg. 58)
Tracking Errors through a design netlist .....	A (pg. 79)
Annotating measurements onto the Waveform display .....	A (pg. 138/139) & slide 44

## Signal Strengths that could be observed in the Simulator Waveform Window

State Name	Symbol	Description
Low		Strong Logical Low state, e.g. output of a TTL gate.
High		Strong Logical High state, e.g. output of a TTL gate.
High Impedance		Tri-Stated output or unconnected input.
Unknown		Strong Undefined High or Low state, e.g. initial state of a TTL flip-flop.
Resistive Low		Weak Logical Low state, e.g. Open Emitter output in Low state.
Resistive High		Weak Logical High State, e.g. Open Collector output in High state.
Resistive Unknown		Undefined Resistive Low or Resistive High state.
Output Conflict		Indicates that there is a bus conflict in the node (High and Low at the same time)
High Voltage		This logical state indicates that there is a high voltage at the pin. E.g. +12V, -5V, etc. etc.
Reference Voltage		The Reference voltage used in ECL technology
Unknown Activity Low		Low or Resistive Low or High Impedance. This state is generated by tri-state ICs when the tri-state control pin is undefined.
Unknown Activity High		High, Resistive High or High Impedance. Similar to the above.
SV High		Power Voltage High (e.g. VCC)
SV Low		Power Voltage Low (e.g. GND)
SV X		Power Voltage Unknown



## The Simulation Window (See also pg. 88 )

Deletes all waveforms without resetting the simulation using Power-On

16 bit software counter LEDs. These represent the current count value

Enable ruler

Zoom Out

Display scale pg 124

Zoom In  
( Note to manually Zoom In, place cursor in here, click with left mouse and drag across the time period then release. During drag operation a light blue bar is displayed).

Use this button to toggle the comment display on and off.

Comments can be used to document important situations on the waveform diagram and can be both displayed at the specified screen locations and printed with the waveform diagram.

Annotates precise measurements between signal transitions on the waveform diagram. (pg 138/139)

This invokes the logical states selection window which allows one to select and assign any logical state to any signal or pin. Pg.

This invokes the stimulus window that is used to define and assign stimulators or test vectors to the selected signals See Slide

This invokes the Probes Selection Window which is used to select signals and IC pins in the Waveform window. See Slide 10 .

toggling this button alternately collapses and expands buses.

It is only meaningful only if you have defined some buses, either in the waveform display window, or applied a probe to a schematic bus. ( See pg. 38)

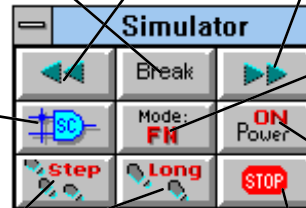
## Simulator Main Tool Box Window

Toggle this button to select the item to search for.

Breakpoint (pg. 228 & slide 23)	Presets (pg. 127 & slide 33)
TAGs (pg. 139 & slide 29)	Milestones (pg. 152 & slide 32)
Errors (pg. 145)	Events

...Toggle these buttons to Search

Launch the Schematic tool  
from the simulator!!



Toggle this button to select between  
Simulation modes.

Functional (pg. 30 & 132)

Glitch (pg. 33 & 133)

Timing (pg. 35 & 133)

Run the simulation for a short step  
or long step, the periods of which  
can be defined like this. (pg. 125)

Executes a Power On Reset  
(See pg. 58 & 60)

Stop a long simulation.  
Note that this button can be disabled.  
(see pg. 69 & pg. 64 , and next slide )

To Change the Simulation Long and Short step  
sizes, Select *Utilities -> Simulation Step* from the  
Simulators' Menu Bar. This box appears.

Select the desired values for each long and short  
step, then click on 'Set Step'.



## Running a Long Simulation (pg. 64)

The previous slide identified buttons used to invoke a short and long simulation step.

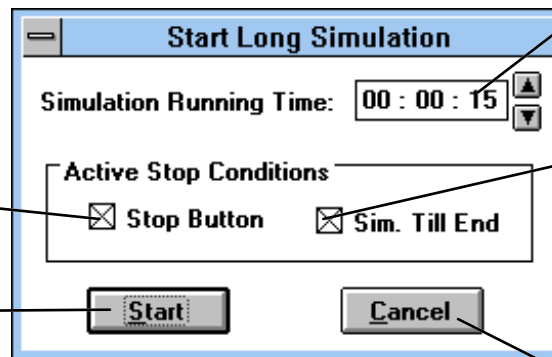
If the idea of writing a Command file to run for hours does not appeal, the simulator has the ability to define a long simulation period in terms of Hr-Min-Sec.

From the Simulator Waveform Viewer Menu bar, select *Options -> Simulation Stop...*

... this window appears and it operates as follows;-

This box disables the STOP button on the Simulation Main Toolbox window. (See Previous slide).

This makes coffee.  
... OK so it doesn't, but after selecting it, you can!



Define the required simulation period.

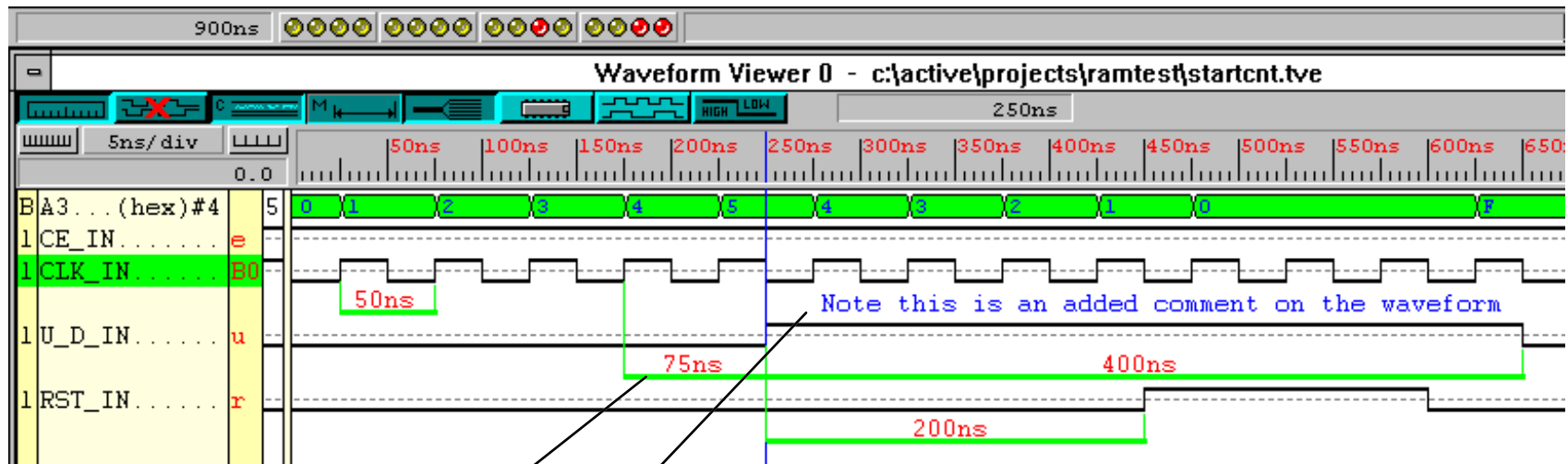
Select this box to enable simulation until the end of selected period.

If only this worked on my mortgage...

(... This has been a long presentation.)

Note: For simulations longer than 10 seconds, a window pops up with a display counting down in Hr-Min-Sec.

## Annotating Measurements and Comments Onto the Waveform Display



Measurements pg. 138/139

To activate Measurement annotation, select *Waveform -> Measurements -> Measurements on*.

Use the cursor to drop start points and endpoints for the measurement annotation.

To Turn Measurement display on and off, toggle the *Waveform -> Measurements -> Measurements on* option.

Comments pg. 141

To activate Comments, select *Waveform -> Comments -> Add*, ..... add comments in the box that appears, then select 'close'. The comments will be dropped where the blue cursor is positioned.

There is no option to hide comments, they are either there or not.