ModelSim

6.5 Update 2009

Walter Gude

Senior Applications Engineer

Mentor Graphics®

ModelSim

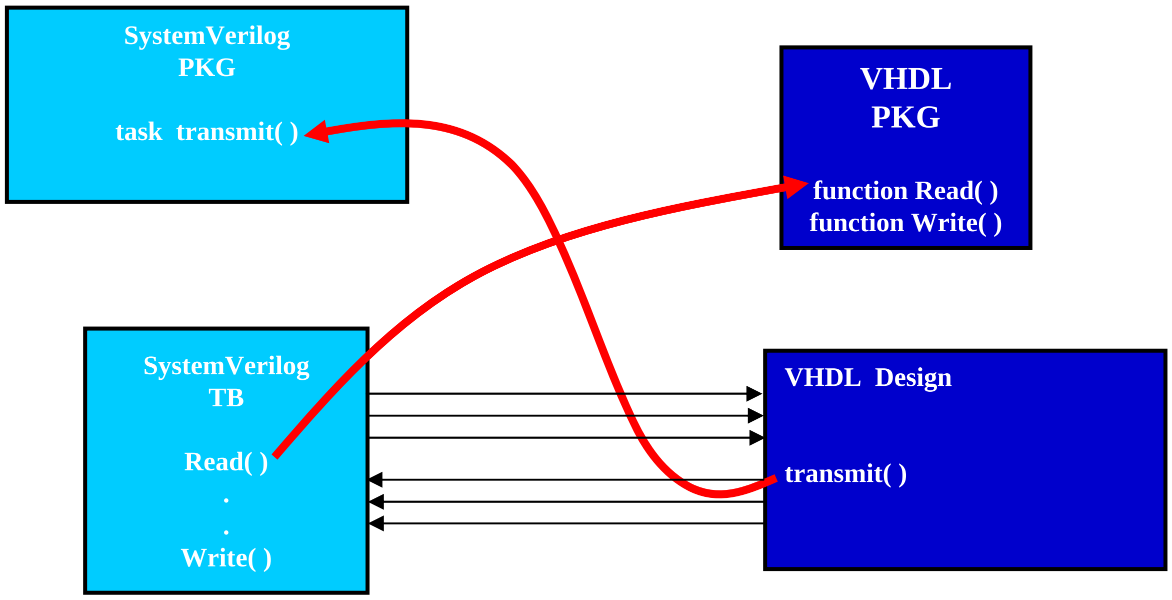
Leading Single & Mixed Language Simulation

- **Native single kernel verification environment**
  - **Verilog 1995, 2001, 2005**
  - **VHDL 1987/1993/2002/2008**
  - **SystemVerilog for design**
  - **SystemC with SCV and TLM, C, C++ (option)**

- **Broadest type support at language boundaries**
  - **Component/module instantiation**
  - **SignalSpy™**
  - **SC control, observe and connect methods**
  - **Only simulator able to share type definitions written in one package in both VHDL and SystemVerilog**
    - **Preserving full benefits of strong type checking**

- **Integrated debug capabilities**
  - **Commands/GUI consistent across languages, HW platforms and abstraction levels**

True Mixed Language Environment



- **Complexity and size drive the need for multi-abstraction verification**
  - **Transaction to gate**

- **Need common environment to debug mixed language simulations**

Agenda

- Performance
- Debug & Analysis
- Coverage

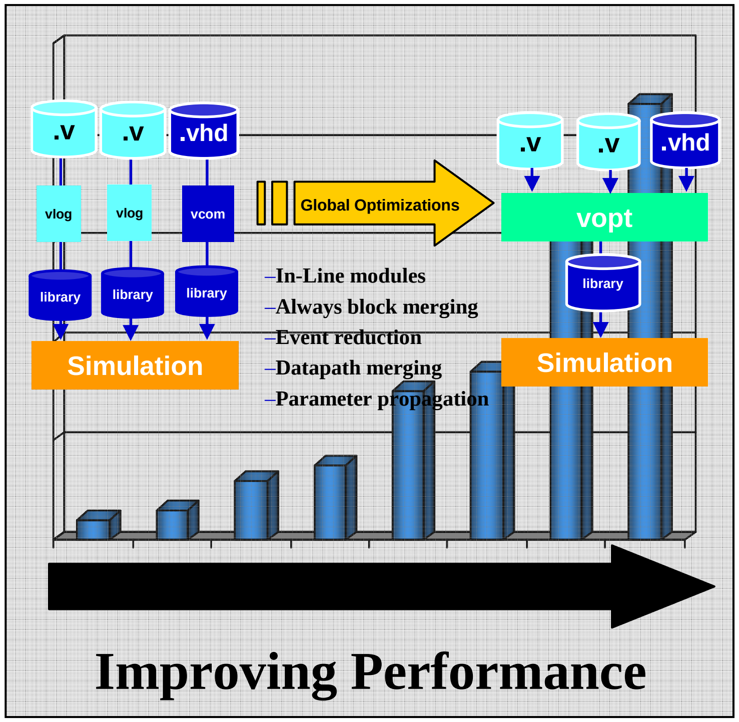
Performance & Capacity

- Native Single Kernel for all languages
- Global optimizations
- Best Verilog gate-level performance
- Best time to next simulation performance
- Multi-core capable

Performance improvements in every release

Verification Throughput

- Native simulator architecture
  - Continual opportunities for performance improvements
- vopt performance flow (SE only)
  - All abstract levels
  - Run time optimization
- Regression suite throughput
  - "time to next simulation"
- Best gate-level performance
  - Best capacity
  - VHDL/Verilog compiled SDF



See Questa/ModelSim User's Manual
Optimizing designs with vopt

6.5 Performance Update

- **SystemVerilog**
  - **30% on active benchmark designs vs 6.4**
- **Verilog/SystemVerilog**
  - **Race abatement algorithm**
  - **Gate Level**
    - **Design load/elaboration**
      - **3x improvement on customer designs**
    - **SDF timing options**
      - **Optimize and elaborate once run any sdfmin<u>r</u>/sdftyp<u>r</u>/sdfmax<u>r</u> settings**
        - **Optimization time and disk space reduction**

6.5 Performance Update

- **VHDL**
  - **15% on target designs vs 6.4**
- **Time to next simulation (introduced in 6.4)**
  - **Pre-optimize unchanged blocks for regression suite throughput**
  - **Mix optimized and non-optimized blocks**
  - **Disk space savings**

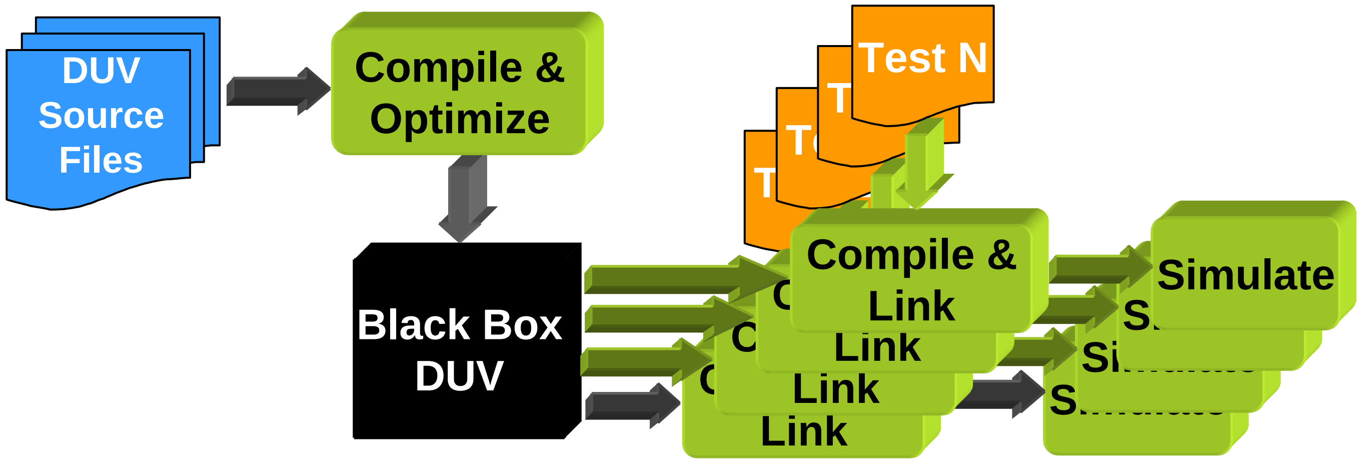
Possible Black Box Flow

```
vlib work
vlib asic_lib
vlog -work asic_lib cell_lib.v
vlog netlist.v
vopt -L asic_lib -debugCellOpt +nocheckALL \
     -bbox netlist -o optnet
vlog tb.v test1.v
vsim -c tb -do sim.do
```

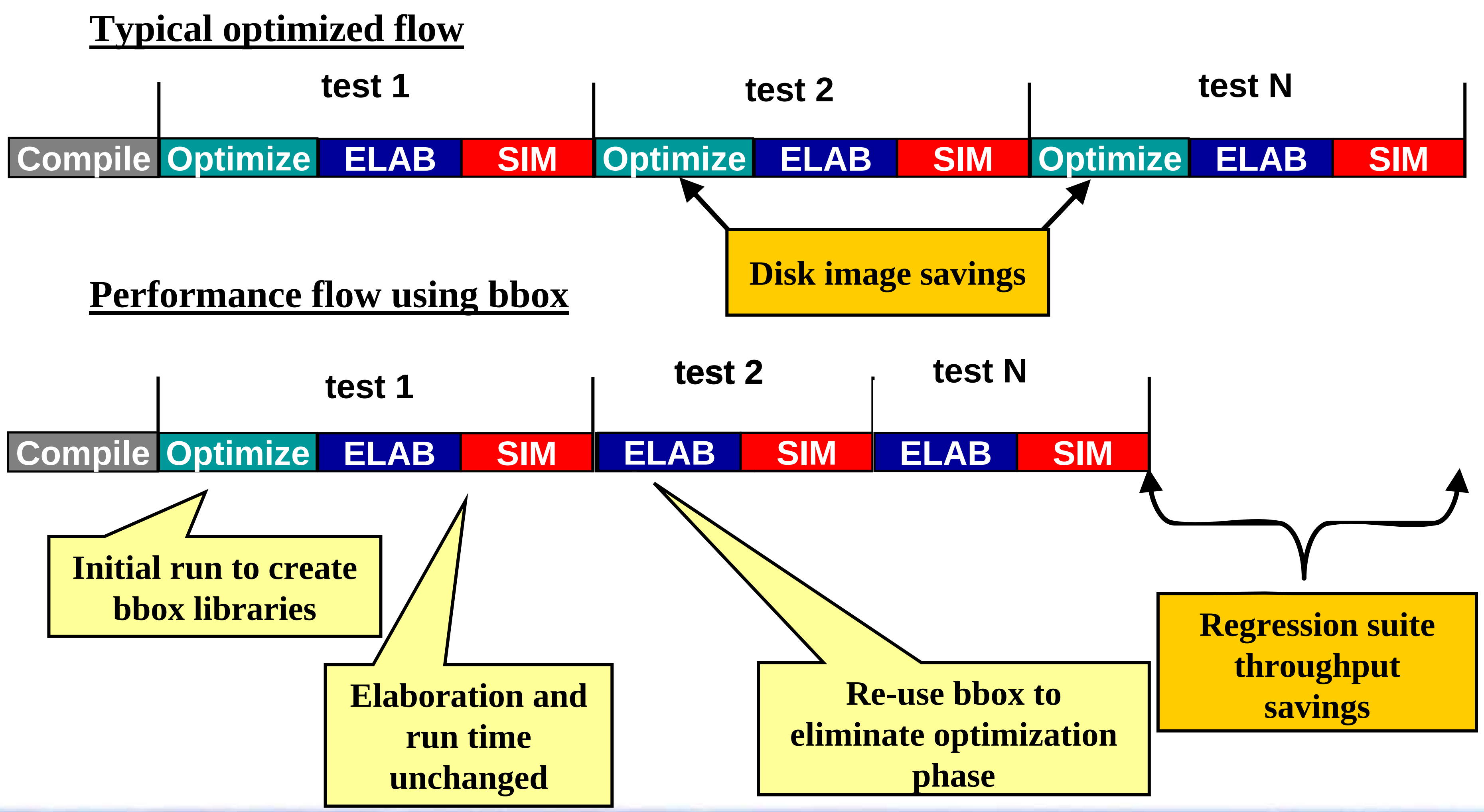
```
vlog tb.v test2.v
vsim -c tb -do sim.do
```

```
vlog test3.v
vsim -c tb -do sim.do
```

```
vlog testN.v
vsim -c tb -do sim.do
```



Verification Throughput


Typical optimized flow
test 1
test 2
test N
Compile
Optimize
ELAB
SIM
Optimize
ELAB
SIM
Optimize
ELAB
SIM
Disk image savings
Performance flow using bbox
test 1
test 2
test N
Compile
Optimize
ELAB
SIM
ELAB
SIM
ELAB
SIM
Initial run to create bbox libraries
Elaboration and run time unchanged
Re-use bbox to eliminate optimization phase
Regression suite throughput savings

Compiled SDF
Significantly Reduces GLS Load Times!

- SDF can be compiled using sdfcom
  - Significant time savings during elaboration for SDF files used repetitively without modification
  - Use: sdfcom <sdf_file> <compiled_name>

- When vopt is run, sdfcom will be run implicitly if it detects one of the following:
  - $sdf_annotate
  - -sdfmin/max/typ
  - Disable with the “modelsim.ini” file setting:
    - VoptAutoSDFCompile

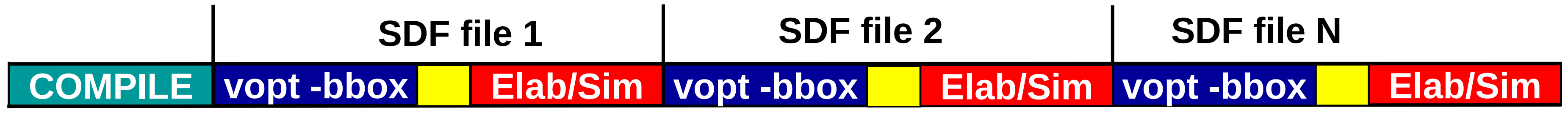
Verification Throughput
Simulating with Different Timing Corners

- Originally changing SDF files required creation of the bbox object for each SDF file
  - Reduced throughput by requiring many ***vopt*** runs
  - Required extra disk space to keep multiple bbox libraries
- Using SDF replacement increases throughput and reduces needed disk space
  - ***vopt -bbox*** once, ***vsim*** many times with different timing corners

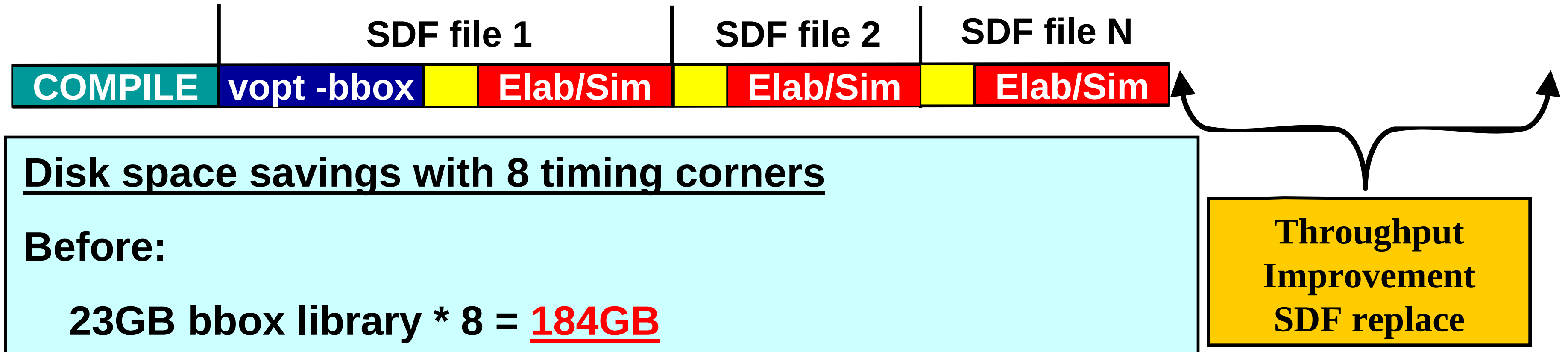
```
vopt -bbox core -o core_bb

vsim top -sdftypr /top/dut=tc-1.sdf
vsim top -sdftypr /top/dut=tc-2.sdf
vsim top -sdftypr /top/dut=tc-3.sdf
   ...
vsim top -sdftypr /top/dut=tc-i.sdf
```

Case Study: SDF Replacement

Run time for traditional SDF flow with bbox



Using SDF replacement flow with bbox



<u>**Disk space savings with 8 timing corners**</u>

Before:

23GB bbox library * 8 = <u>184GB</u>

After:

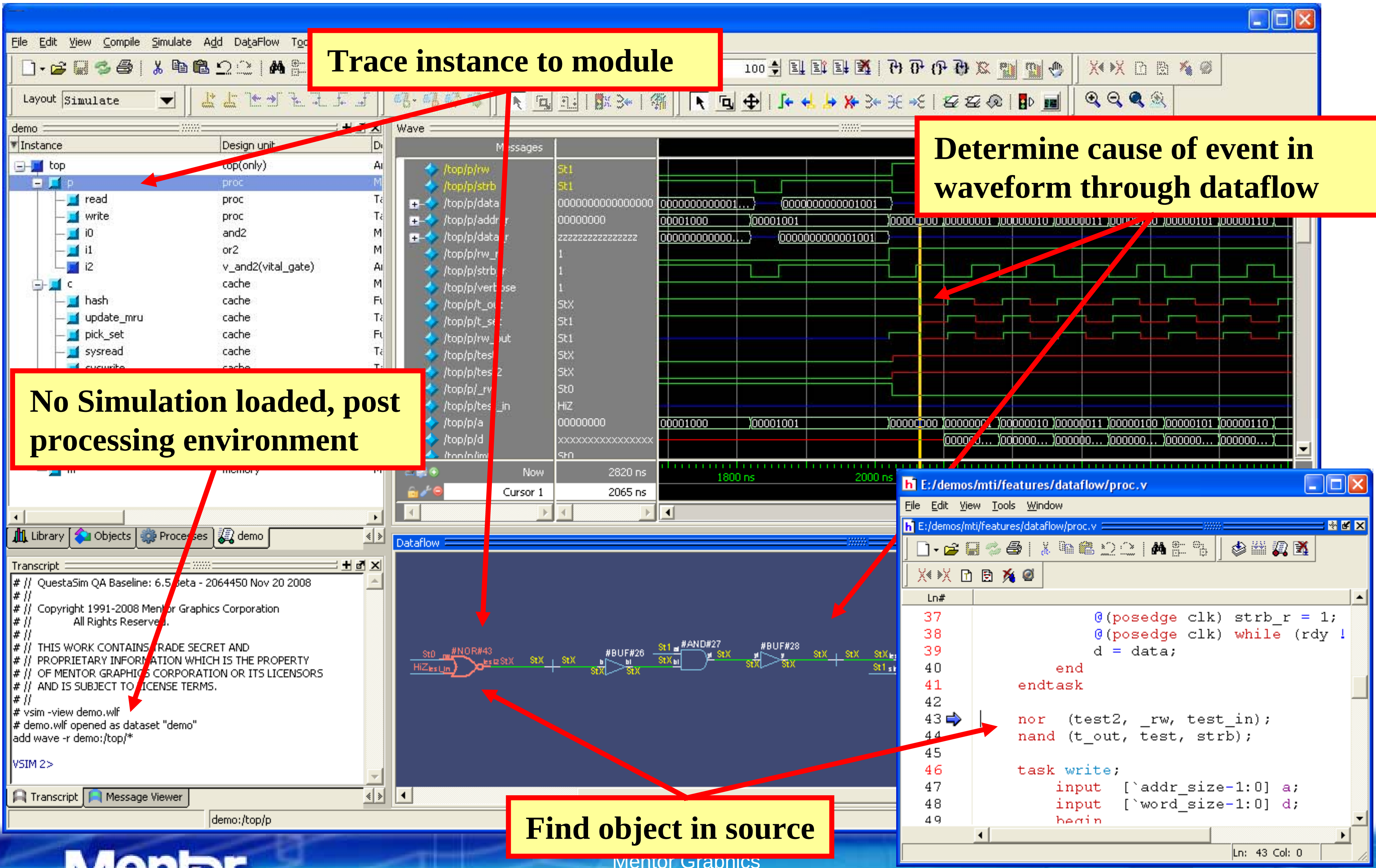
23GB bbox library + (8 * 800MB Compiled SDF) = <u>29.4GB</u>

Exploiting Multicore Platforms

- **Ability to enable multi-threaded logging of simulation results**
  - **Can improve performance up to 2x**

- **SystemC compile**
  - **Faster SC compilation when enabled**

Optional Post Processing Debug



Trace instance to module

Determine cause of event in waveform through dataflow

No Simulation loaded, post processing environment

Find object in source

Instance	Design unit
top	top(only)
p	proc
read	proc
write	proc
i0	and2
i1	or2
i2	v_and2(vital_gate)
c	cache
hash	cache
update_mru	cache
pick_set	cache
sysread	cache

```
# //  QuestaSim QA Baseline: 6.5 Beta - 2064450 Nov 20 2008
# //
# //  Copyright 1991-2008 Mentor Graphics Corporation
# //              All Rights Reserved.
# //
# //  THIS WORK CONTAINS TRADE SECRET AND
# //  PROPRIETARY INFORMATION WHICH IS THE PROPERTY
# //  OF MENTOR GRAPHICS CORPORATION OR ITS LICENSORS
# //  AND IS SUBJECT TO LICENSE TERMS.
# //
# vsim -view demo.wlf
# demo.wlf opened as dataset "demo"
add wave -r demo:/top/*

VSIM 2>
```

E:/demos/mti/features/dataflow/proc.v

```
                @(posedge clk) strb_r = 1;
                @(posedge clk) while (rdy !
                d = data;
            end
        endtask

        nor  (test2, _rw, test_in);
        nand (t_out, test, strb);

        task write;
            input  [`addr_size-1:0] a;
            input  [`word_size-1:0] d;
            begin
```

Ln: 43 Col: 0

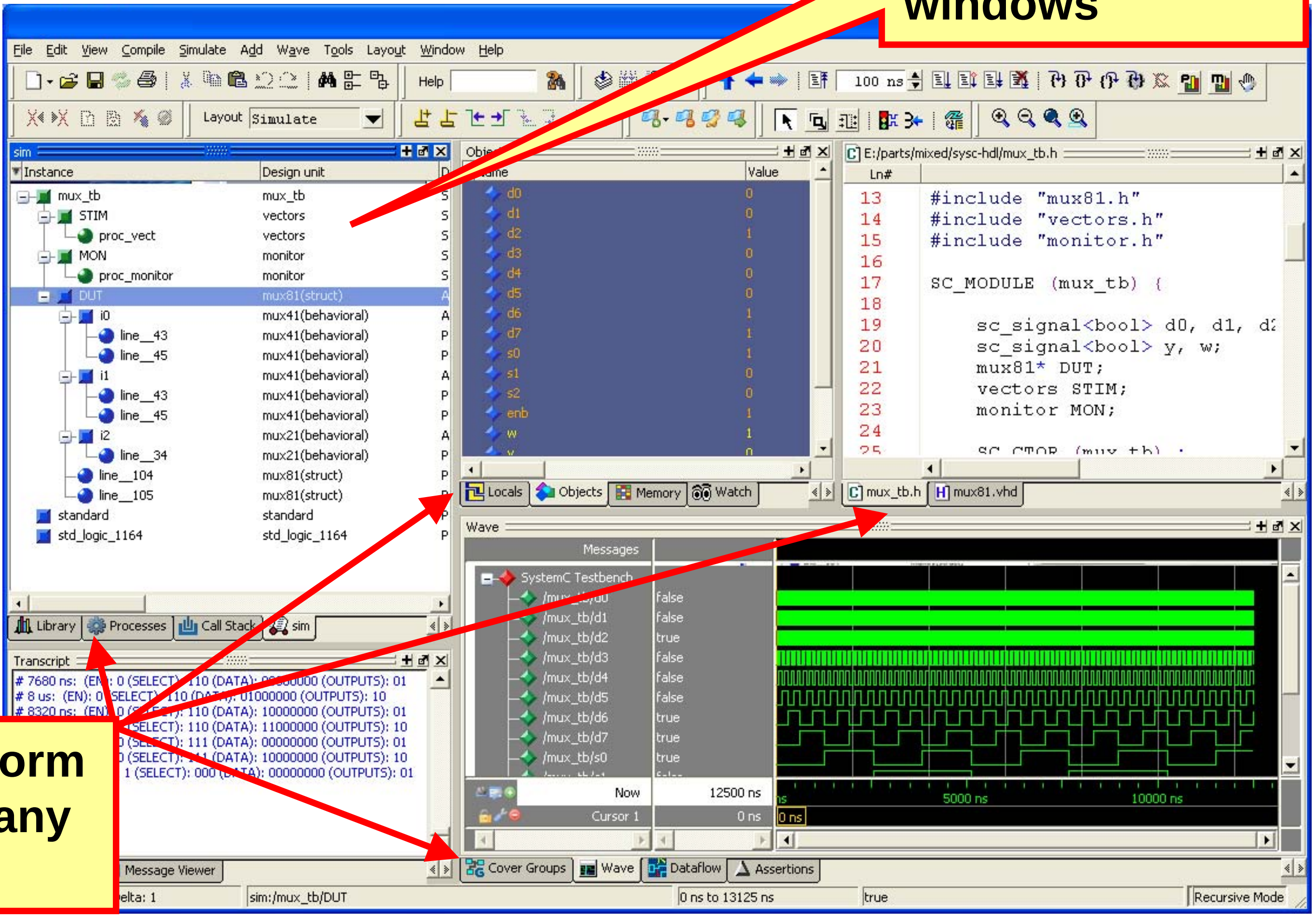
Agenda

- **Performance**
- **Debug & Analysis**
- **Coverage**

Window Manager

- Consistent look and feel
- Improved quality

Common control & behavior for all windows



Instance	Design unit
mux_tb	mux_tb
STIM	vectors
proc_vect	vectors
MON	monitor
proc_monitor	monitor
DUT	mux81(struct)
i0	mux41(behavioral)
line__43	mux41(behavioral)
line__45	mux41(behavioral)
i1	mux41(behavioral)
line__43	mux41(behavioral)
line__45	mux41(behavioral)
i2	mux21(behavioral)
line__34	mux21(behavioral)
line__104	mux81(struct)
line__105	mux81(struct)
standard	standard
std_logic_1164	std_logic_1164

Name	Value
d0	0
d1	0
d2	1
d3	0
d4	0
d5	0
d6	1
d7	1
s0	1
s1	0
s2	0
enb	1
w	1

E:/parts/mixed/sysc-hdl/mux_tb.h

```
#include "mux81.h"
#include "vectors.h"
#include "monitor.h"

SC_MODULE (mux_tb) {

    sc_signal<bool> d0, d1, d2
    sc_signal<bool> y, w;
    mux81* DUT;
    vectors STIM;
    monitor MON;
```

Transcript

```
# 7680 ns:  (EN): 0 (SELECT): 110 (DATA): 00000000 (OUTPUTS): 01
# 8 us:  (EN): 0 (SELECT): 110 (DATA): 01000000 (OUTPUTS): 10
# 8320 ns:  (EN): 0 (SELECT): 110 (DATA): 10000000 (OUTPUTS): 01
(SELECT): 110 (DATA): 11000000 (OUTPUTS): 10
0 (SELECT): 111 (DATA): 00000000 (OUTPUTS): 01
0 (SELECT): 111 (DATA): 10000000 (OUTPUTS): 10
1 (SELECT): 000 (DATA): 00000000 (OUTPUTS): 01
```

Wave

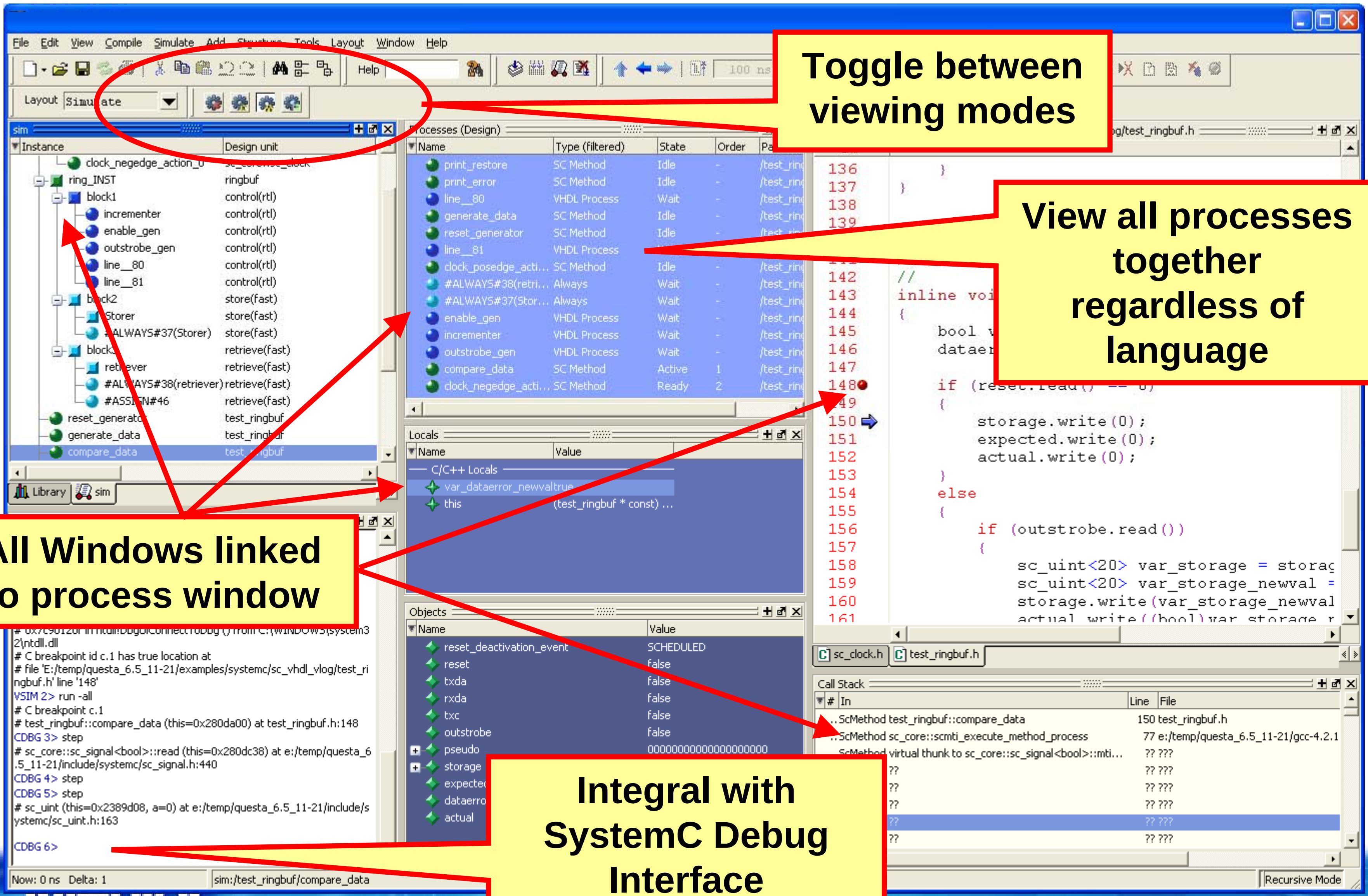
Messages	
SystemC Testbench	
/mux_tb/d0	false
/mux_tb/d1	false
/mux_tb/d2	true
/mux_tb/d3	false
/mux_tb/d4	false
/mux_tb/d5	false
/mux_tb/d6	true
/mux_tb/d7	true
/mux_tb/s0	true
Now	12500 ns
Cursor 1	0 ns

5000 ns 10000 ns 0 ns

elta: 1 | sim:/mux_tb/DUT | 0 ns to 13125 ns | true | Recursive Mode

Any window can form a tab group with any other window

Easing Processes Debug



Toggle between viewing modes

View all processes together regardless of language

All Windows linked to process window

Integral with SystemC Debug Interface

Instance	Design unit
clock_negedge_action_0	
ring_INST	ringbuf
block1	control(rtl)
incrementer	control(rtl)
enable_gen	control(rtl)
outstrobe_gen	control(rtl)
line__80	control(rtl)
line__81	control(rtl)
block2	store(fast)
Storer	store(fast)
#ALWAYS#37(Storer)	store(fast)
block3	retrieve(fast)
retriever	retrieve(fast)
#ALWAYS#38(retriever)	retrieve(fast)
#ASSIGN#46	retrieve(fast)
reset_generator	test_ringbuf
generate_data	test_ringbuf
compare_data	test_ringbuf

Name	Type (filtered)	State	Order	Pa
print_restore	SC Method	Idle	-	/test_ring
print_error	SC Method	Idle	-	/test_ring
line__80	VHDL Process	Wait	-	/test_ring
generate_data	SC Method	Idle	-	/test_ring
reset_generator	SC Method	Idle	-	/test_ring
line__81	VHDL Process			
clock_posedge_acti…	SC Method	Idle	-	/test_ring
#ALWAYS#38(retri…	Always	Wait	-	/test_ring
#ALWAYS#37(Stor…	Always	Wait	-	/test_ring
enable_gen	VHDL Process	Wait	-	/test_ring
incrementer	VHDL Process	Wait	-	/test_ring
outstrobe_gen	VHDL Process	Wait	-	/test_ring
compare_data	SC Method	Active	1	/test_ring
clock_negedge_acti…	SC Method	Ready	2	/test_ring

Name	Value
C/C++ Locals	
var_dataerror_newvaltrue	
this	(test_ringbuf * const) …

Name	Value
reset_deactivation_event	SCHEDULED
reset	false
txda	false
rxda	false
txc	false
outstrobe	false
pseudo	00000000000000000000
storage	
expected	
dataerro	
actual	

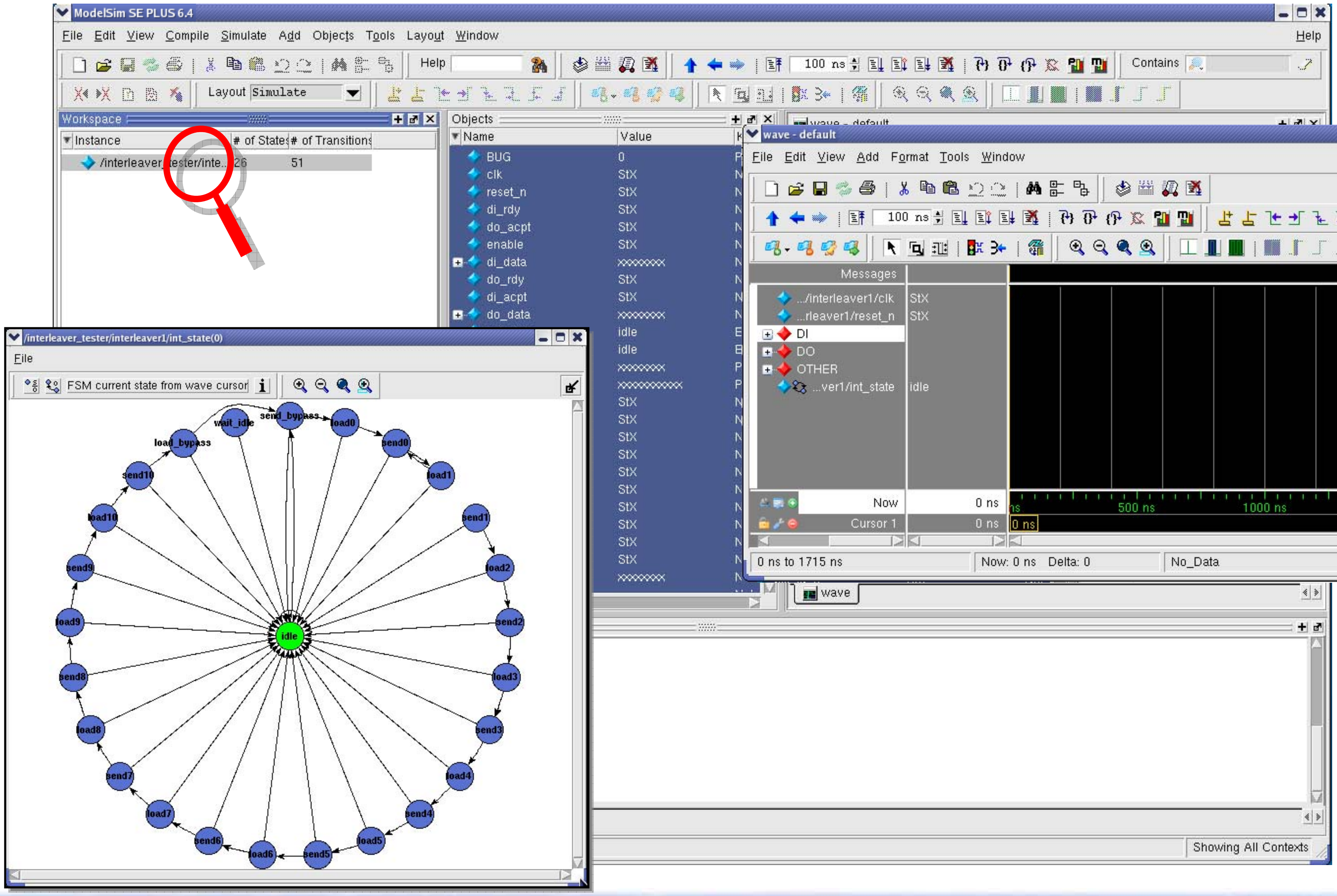
```
136         }
137     }
138
139
142     //
143     inline voi
144     {
145         bool v
146         dataer
147
148         if (re
149         {
150             storage.write(0);
151             expected.write(0);
152             actual.write(0);
153         }
154         else
155         {
156             if (outstrobe.read())
157             {
158                 sc_uint<20> var_storage = storag
159                 sc_uint<20> var_storage_newval =
160                 storage.write(var_storage_newval
```

#	In	Line	File
	..ScMethod test_ringbuf::compare_data	150	test_ringbuf.h
	..ScMethod sc_core::scmti_execute_method_process	77	e:/temp/questa_6.5_11-21/gcc-4.2.1
	ScMethod virtual thunk to sc_core::sc_signal<bool>::mti…	??	???
	??	??	???
	??	??	???
	??	??	???
	??	??	???
	??	??	???

```
2\ntdll.dll
# C breakpoint id c.1 has true location at
# file 'E:/temp/questa_6.5_11-21/examples/systemc/sc_vhdl_vlog/test_ri
ngbuf.h' line '148'
VSIM 2> run -all
# C breakpoint c.1
# test_ringbuf::compare_data (this=0x280da00) at test_ringbuf.h:148
CDBG 3> step
# sc_core::sc_signal<bool>::read (this=0x280dc38) at e:/temp/questa_6
.5_11-21/include/systemc/sc_signal.h:440
CDBG 4> step
CDBG 5> step
# sc_uint (this=0x2389d08, a=0) at e:/temp/questa_6.5_11-21/include/s
ystemc/sc_uint.h:163

CDBG 6>
```

Now: 0 ns Delta: 1   sim:/test_ringbuf/compare_data

Debug an FSM



Instance	# of States	# of Transitions
/interleaver_tester/inte...	26	51

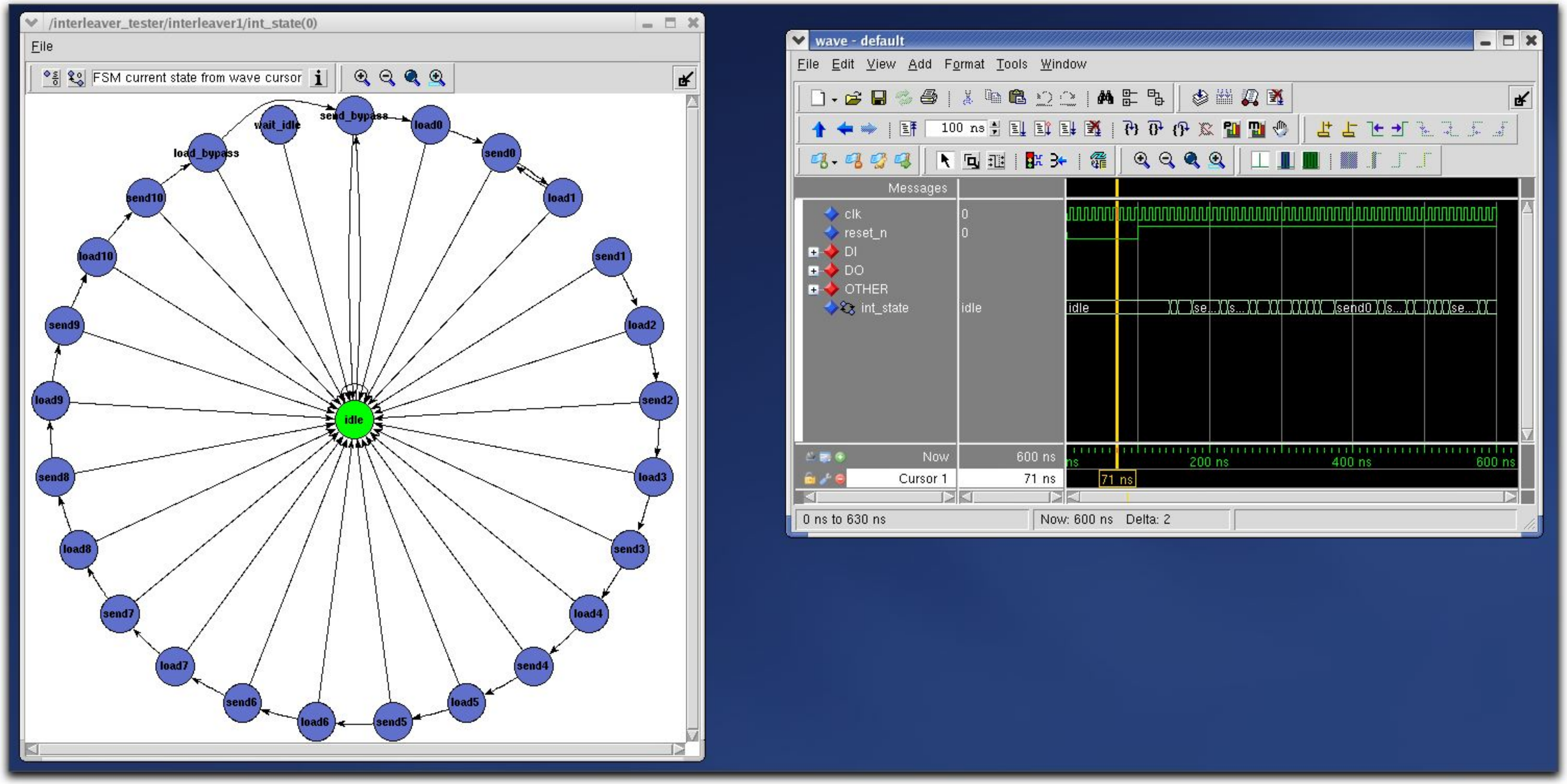
Name	Value
BUG	0
clk	StX
reset_n	StX
di_rdy	StX
do_acpt	StX
enable	StX
di_data	xxxxxxxx
do_rdy	StX
di_acpt	StX
do_data	xxxxxxxx
	idle
	idle
	xxxxxxxx
	xxxxxxxxxxx
	StX
	StX
	StX
	StX
	StX
	StX
	StX
	StX
	StX
	StX
	xxxxxxxx

Messages	
.../interleaver1/clk	StX
...rleaver1/reset_n	StX
DI	
DO	
OTHER	
...ver1/int_state	idle
Now	0 ns
Cursor 1	0 ns

```
ns        500 ns        1000 ns        1500 ns
0 ns
```

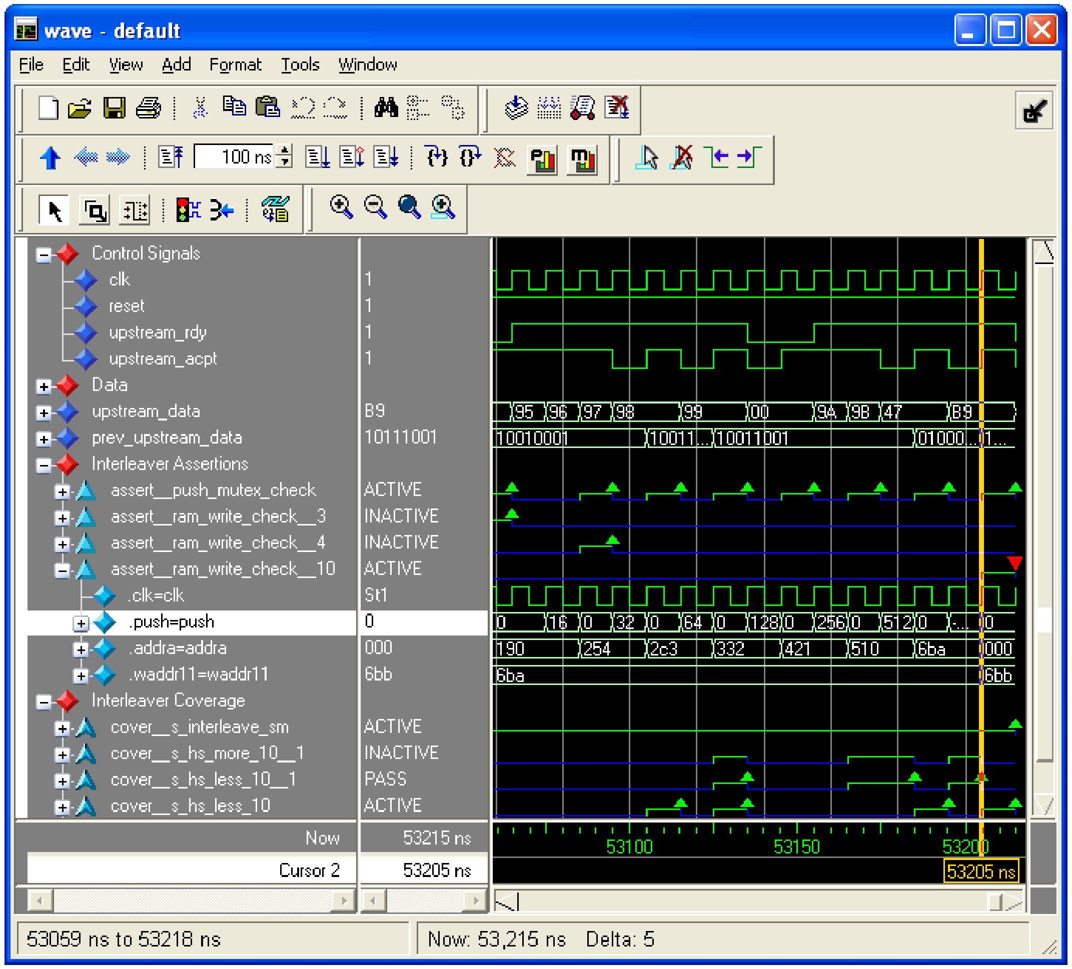
0 ns to 1715 ns | Now: 0 ns Delta: 0 | No_Data

FSM analysis



Enhanced Wave Window

- **Full capabilities**
  - **HDL, SystemC, TLM and Assertion Debug**
- **Cross linked to other windows**
- **Waveform Compare**
- **Virtual Objects**
  - **Signals, Functions and regions**
- **Waveform management**
  - **Dataset snapshot, subset, clear, save, stats**

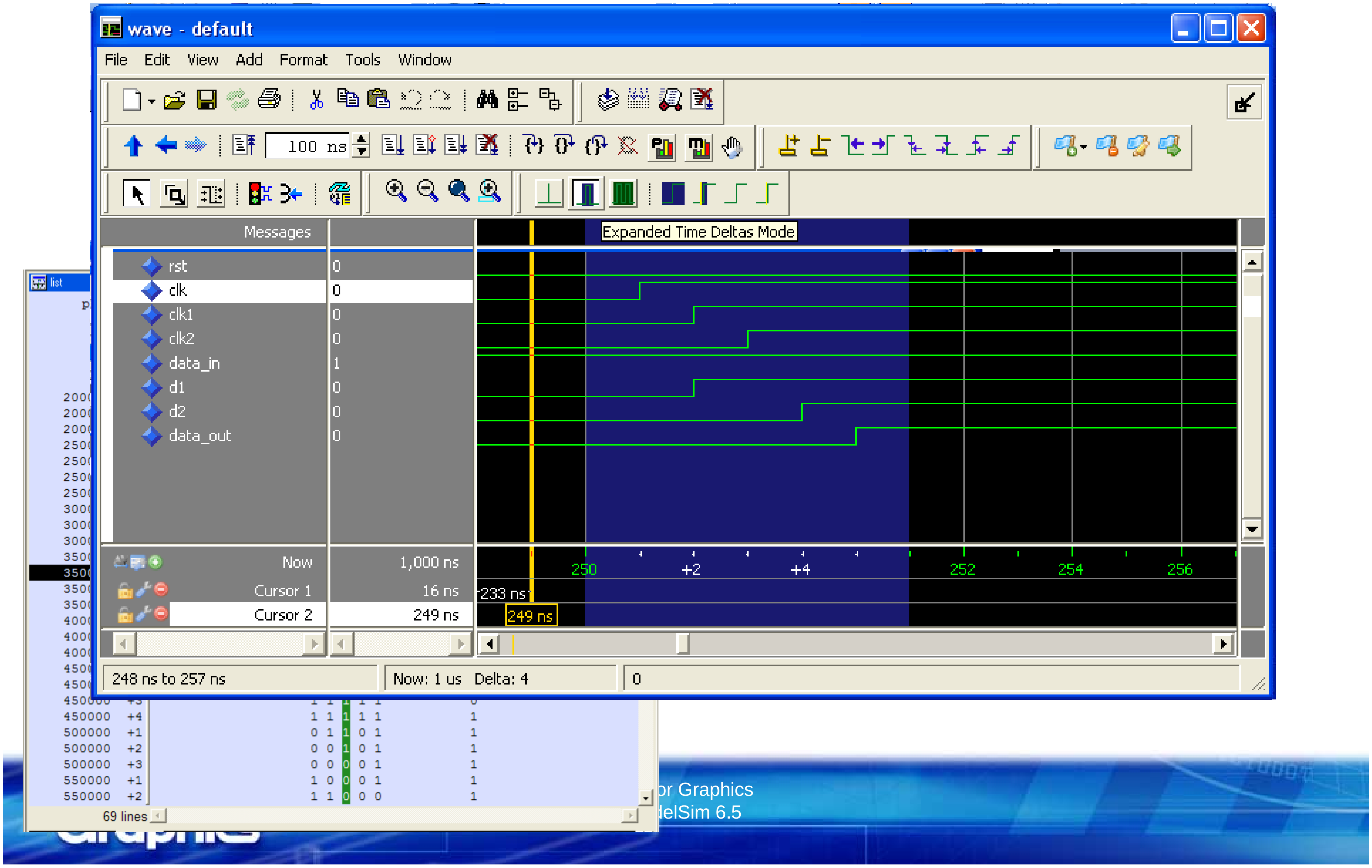


Control Signals	
clk	1
reset	1
upstream_rdy	1
upstream_acpt	1
Data	
upstream_data	B9
prev_upstream_data	10111001
Interleaver Assertions	
assert__push_mutex_check	ACTIVE
assert__ram_write_check__3	INACTIVE
assert__ram_write_check__4	INACTIVE
assert__ram_write_check__10	ACTIVE
.clk=clk	St1
.push=push	0
.addra=addra	000
.waddr11=waddr11	6bb
Interleaver Coverage	
cover__s_interleave_sm	ACTIVE
cover__s_hs_more_10__1	INACTIVE
cover__s_hs_less_10__1	PASS
cover__s_hs_less_10	ACTIVE
Now	53215 ns
Cursor 2	53205 ns

53059 ns to 53218 ns

Now: 53,215 ns Delta: 5

Delta Cycle Debugging



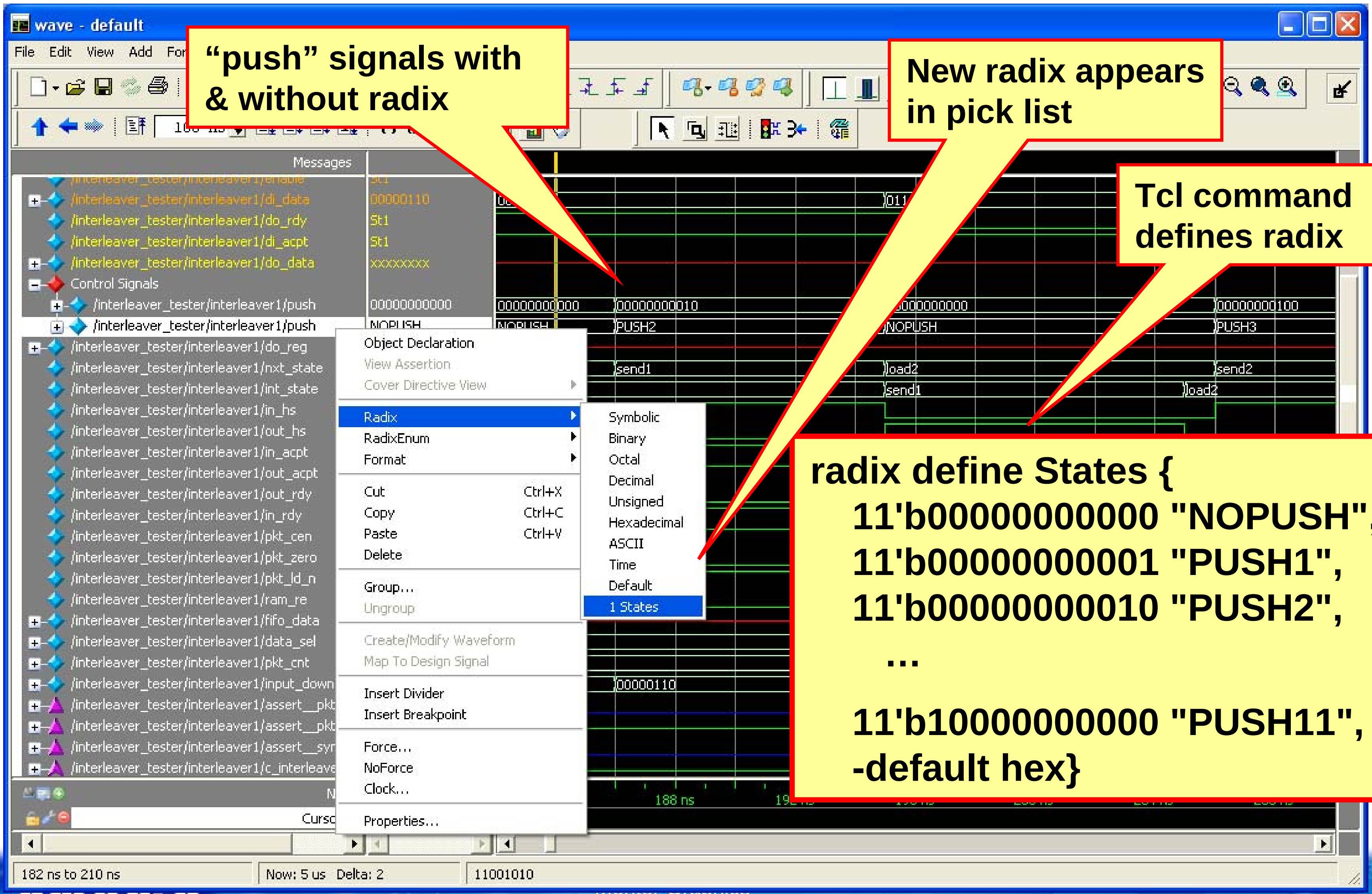
```
rst       0
clk       0
clk1      0
clk2      0
data_in   1
d1        0
d2        0
data_out  0

Now       1,000 ns
Cursor 1  16 ns
Cursor 2  249 ns
```

```
248 ns to 257 ns    Now: 1 us  Delta: 4    0
```

```
450000  +4   1 1 1 1 1   1
500000  +1   0 1 1 0 1   1
500000  +2   0 0 1 0 1   1
500000  +3   0 0 0 0 1   1
550000  +1   1 0 0 0 1   1
550000  +2   1 1 0 0 0   1
```

or Graphics
delSim 6.5

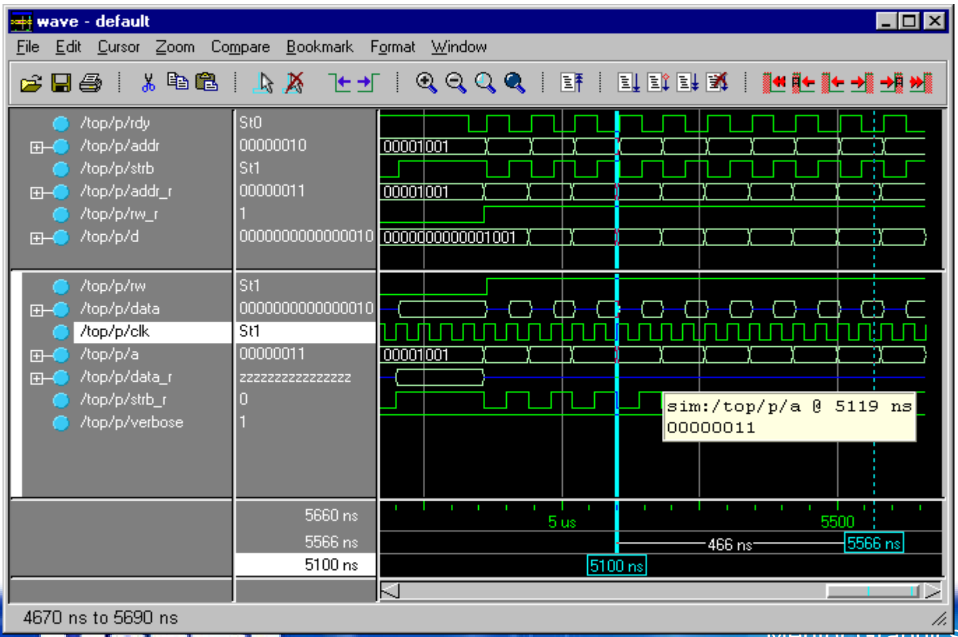
User Defined Radix


wave - default
File Edit View Add For
Messages
/interleaver_tester/interleaver1/di_data 00000110
/interleaver_tester/interleaver1/do_rdy St1
/interleaver_tester/interleaver1/di_acpt St1
/interleaver_tester/interleaver1/do_data xxxxxxxx
Control Signals
/interleaver_tester/interleaver1/push 00000000000
/interleaver_tester/interleaver1/push NOPUSH
/interleaver_tester/interleaver1/do_reg
/interleaver_tester/interleaver1/nxt_state
/interleaver_tester/interleaver1/int_state
/interleaver_tester/interleaver1/in_hs
/interleaver_tester/interleaver1/out_hs
/interleaver_tester/interleaver1/in_acpt
/interleaver_tester/interleaver1/out_acpt
/interleaver_tester/interleaver1/out_rdy
/interleaver_tester/interleaver1/in_rdy
/interleaver_tester/interleaver1/pkt_cen
/interleaver_tester/interleaver1/pkt_zero
/interleaver_tester/interleaver1/pkt_ld_n
/interleaver_tester/interleaver1/ram_re
/interleaver_tester/interleaver1/fifo_data
/interleaver_tester/interleaver1/data_sel
/interleaver_tester/interleaver1/pkt_cnt
/interleaver_tester/interleaver1/input_down
Object Declaration
View Assertion
Cover Directive View
Radix
RadixEnum
Format
Cut Ctrl+X
Copy Ctrl+C
Paste Ctrl+V
Delete
Group...
Ungroup
Create/Modify Waveform
Map To Design Signal
Insert Divider
Insert Breakpoint
Force...
NoForce
Clock...
Properties...
Symbolic
Binary
Octal
Decimal
Unsigned
Hexadecimal
ASCII
Time
Default
1 States
00000000010
PUSH2
NOPUSH
00000000100
PUSH3
send1
load2
send2
188 ns
182 ns to 210 ns
Now: 5 us Delta: 2
11001010
"push" signals with & without radix
New radix appears in pick list
Tcl command defines radix

```
radix define States {
   11'b00000000000 "NOPUSH",
   11'b00000000001 "PUSH1",
   11'b00000000010 "PUSH2",
     …
   11'b10000000000 "PUSH11",
   -default hex}
```

Wave Window

- Create multiple panes and drag and drop signals from one pane to the other.
- Powerful Edit and Search Capabilities under the Edit Menu.
- Cursors - multiple, jump to edge and measurement.
- Bookmarks for marking multiple waveform views.
- Balloon popup to display values

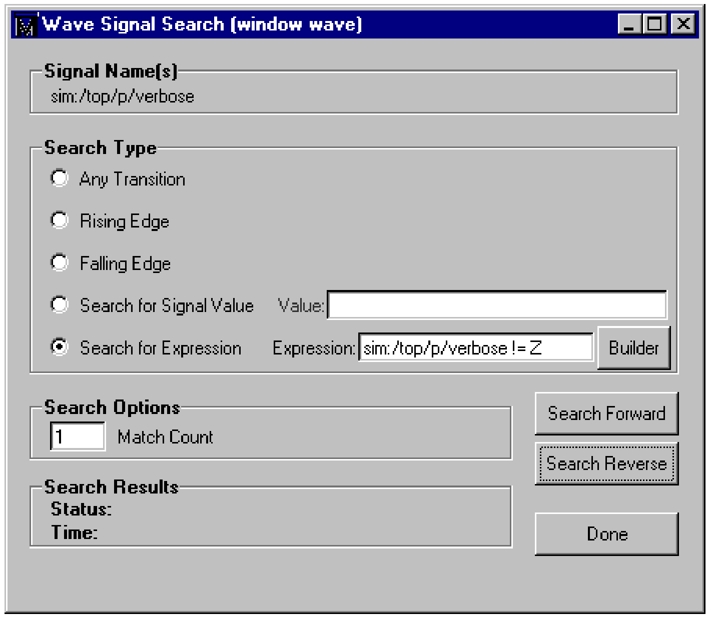


/top/p/rdy	St0
/top/p/addr	00000010
/top/p/strb	St1
/top/p/addr_r	00000011
/top/p/rw_r	1
/top/p/d	0000000000000010
/top/p/rw	St1
/top/p/data	0000000000000010
/top/p/clk	St1
/top/p/a	00000011
/top/p/data_r	zzzzzzzzzzzzzzzz
/top/p/strb_r	0
/top/p/verbose	1

```
sim:/top/p/a @ 5119 ns
00000011
```

5660 ns
5566 ns
5100 ns

4670 ns to 5690 ns



Signal Name(s): sim:/top/p/verbose

Search Type
- ○ Any Transition
- ○ Rising Edge
- ○ Falling Edge
- ○ Search for Signal Value   Value:
- ● Search for Expression   Expression: sim:/top/p/verbose != Z

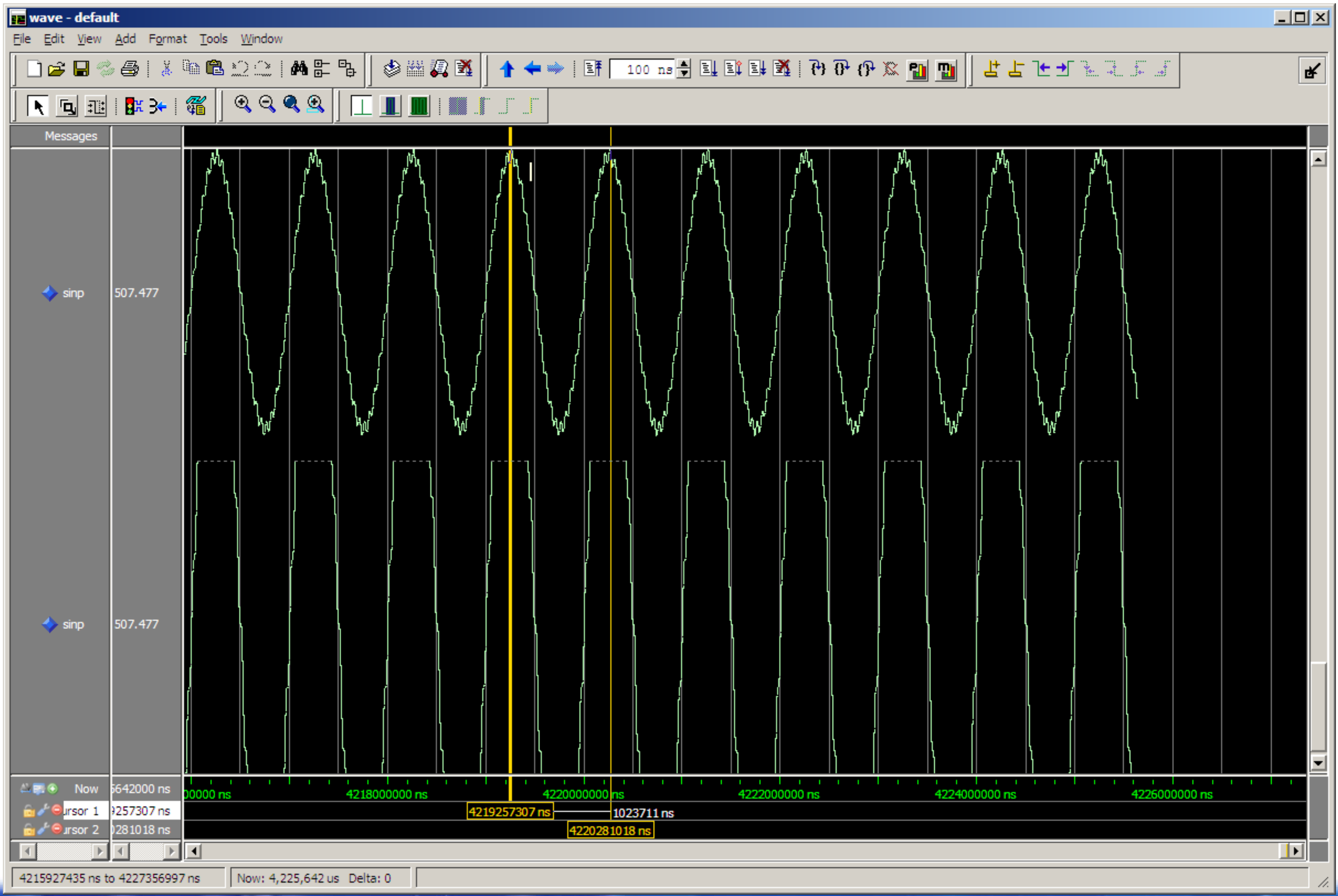
Search Options: 1 Match Count

Search Results
Status:
Time:

Wave Window

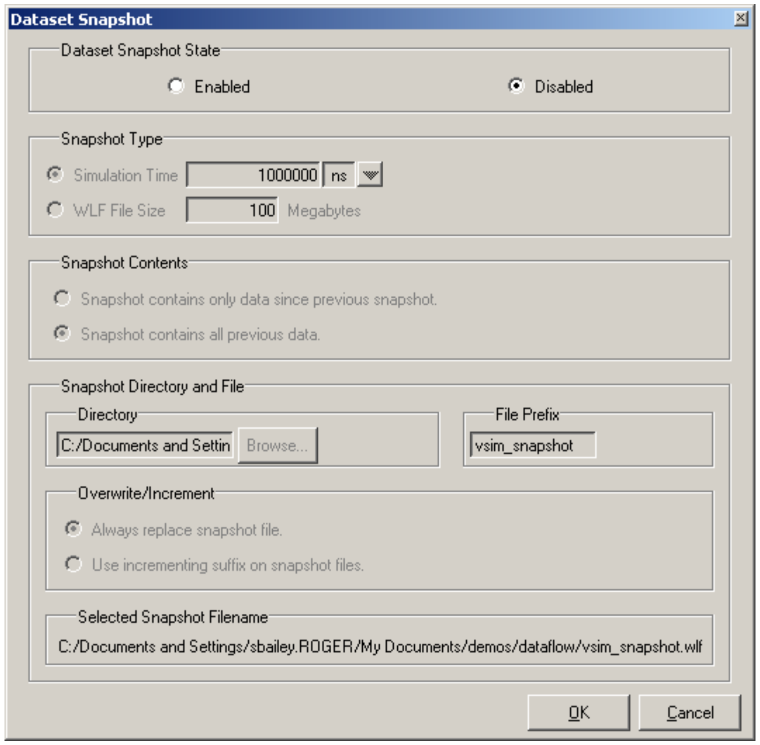
- Wave window reload
- *Write format restart*” command saves a single .do file that can be used to restore the current state of all wave, list and source windows as well as all line and signal (when) breakpoints

Wave Window Analog Display



Dataset snapshot

- Extract waveforms from current simulation
  - Can be based on time
  - Can be based on size
    - Use to minimize waveform file size
  - Use for monitoring batch jobs



Dataset Snapshot

Dataset Snapshot State: Enabled [✓ Disabled]

Snapshot Type:
[✓ Simulation Time] 1000000 ns
WLF File Size 100 Megabytes

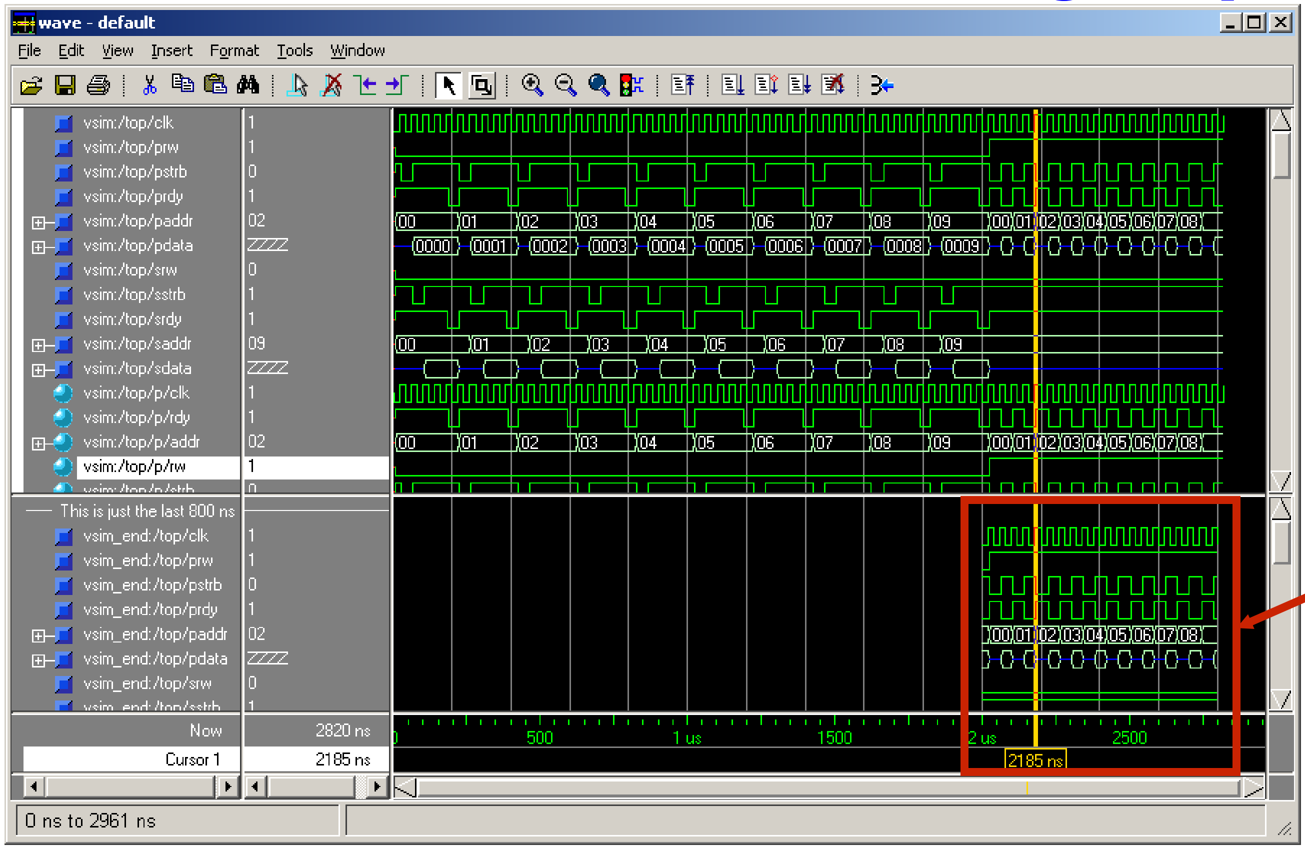
Snapshot Contents:
Snapshot contains only data since previous snapshot.
[✓ Snapshot contains all previous data.]

Snapshot Directory and File
Directory: C:/Documents and Settin
File Prefix: vsim_snapshot

Overwrite/Increment:
[✓ Always replace snapshot file.]
Use incrementing suffix on snapshot files.

Selected Snapshot Filename: C:/Documents and Settings/sbailey.ROGER/My Documents/demos/dataflow/vsim_snapshot.wlf

Waveform File Manager (wlfman)



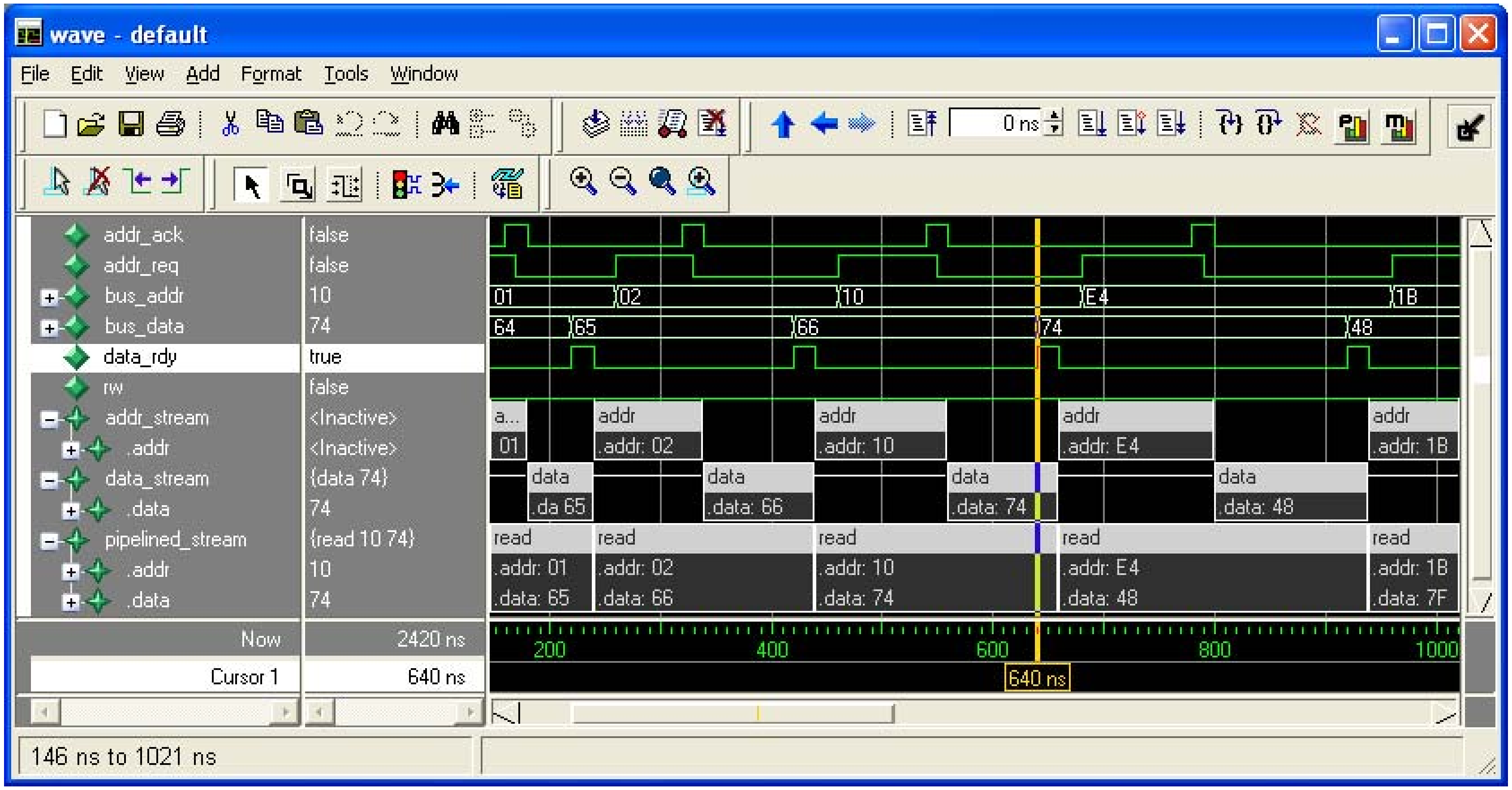
vsim:/top/clk	1
vsim:/top/prw	1
vsim:/top/pstrb	0
vsim:/top/prdy	1
vsim:/top/paddr	02
vsim:/top/pdata	ZZZZ
vsim:/top/srw	0
vsim:/top/sstrb	1
vsim:/top/srdy	1
vsim:/top/saddr	09
vsim:/top/sdata	ZZZZ
vsim:/top/p/clk	1
vsim:/top/p/rdy	1
vsim:/top/p/addr	02
vsim:/top/p/rw	1
This is just the last 800 ns	
vsim_end:/top/clk	1
vsim_end:/top/prw	1
vsim_end:/top/pstrb	0
vsim_end:/top/prdy	1
vsim_end:/top/paddr	02
vsim_end:/top/pdata	ZZZZ
vsim_end:/top/srw	0
Now	2820 ns
Cursor 1	2185 ns

0 ns to 2961 ns

View portion of original waveform file

- **Utility to manipulate existing wlf files**
  - **Reduce amount of information to display**

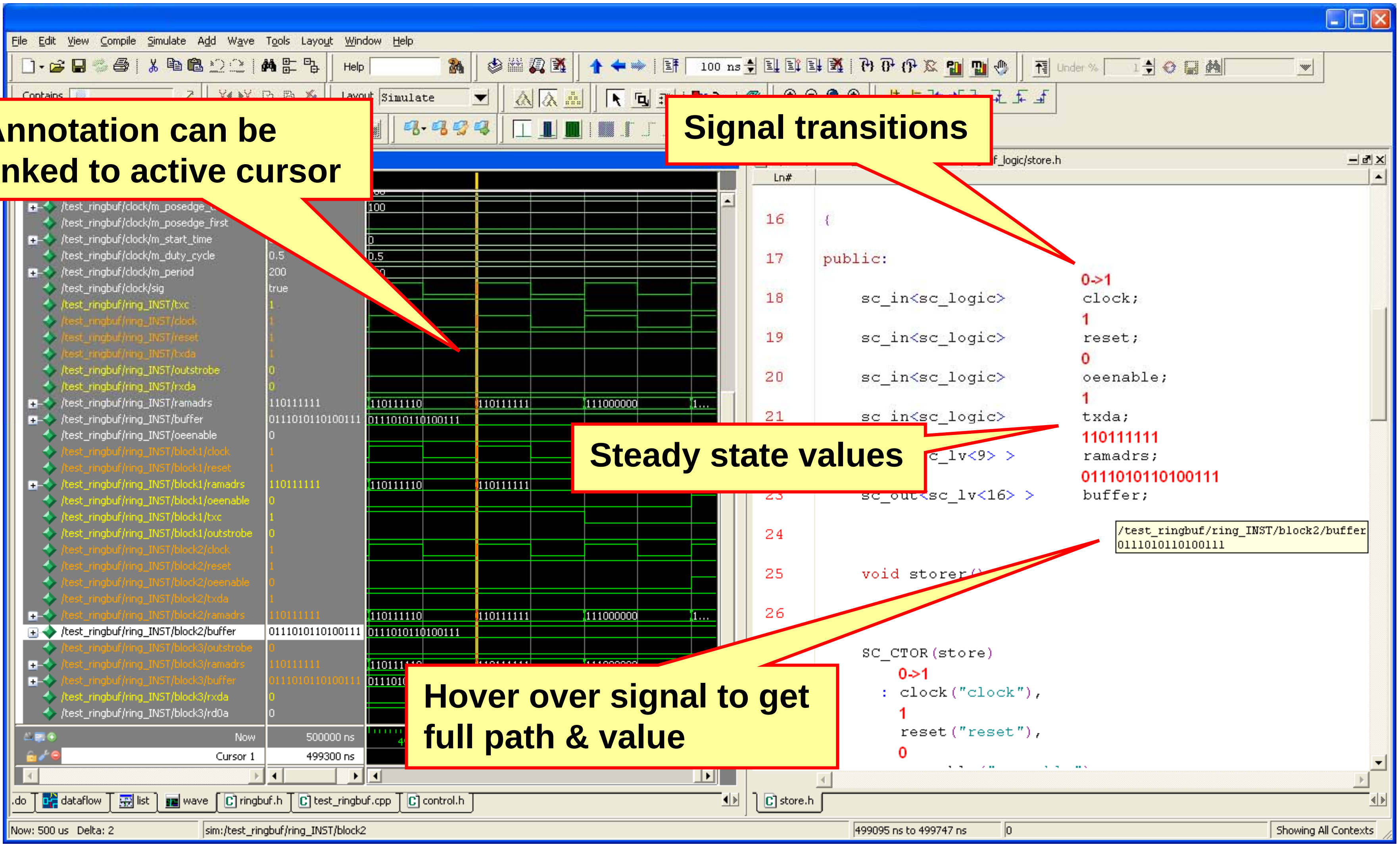
Transaction Display



addr_ack	false
addr_req	false
bus_addr	10
bus_data	74
data_rdy	true
rw	false
addr_stream	<Inactive>
.addr	<Inactive>
data_stream	{data 74}
.data	74
pipelined_stream	{read 10 74}
.addr	10
.data	74
Now	2420 ns
Cursor 1	640 ns

146 ns to 1021 ns

Source Annotation

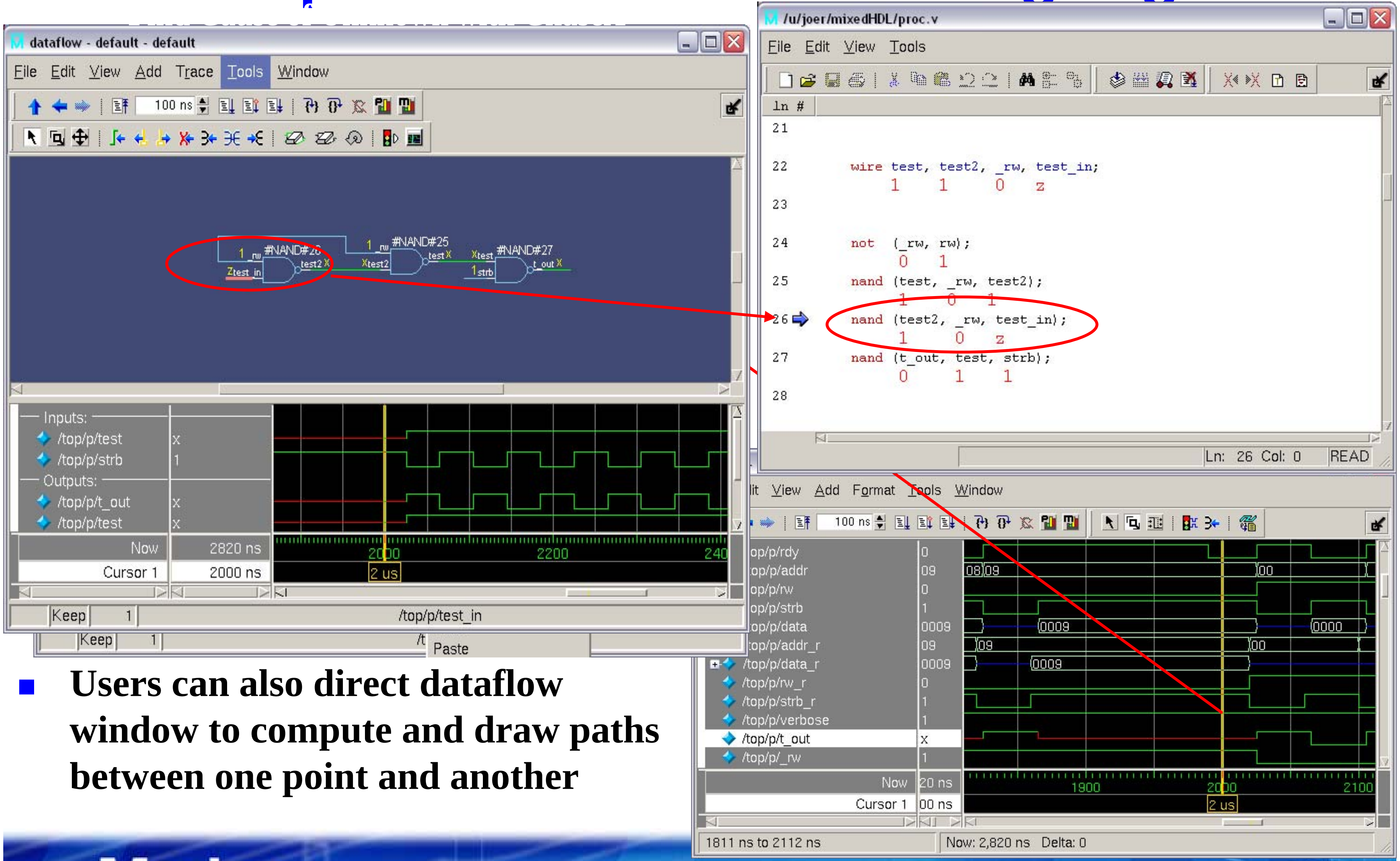

File Edit View Compile Simulate Add Wave Tools Layout Window Help
Annotation can be linked to active cursor
Signal transitions
Steady state values
Hover over signal to get full path & value
.do
dataflow
list
wave
ringbuf.h
test_ringbuf.cpp
control.h
store.h

/test_ringbuf/clock/m_posedge_	
/test_ringbuf/clock/m_posedge_first	
/test_ringbuf/clock/m_start_time	
/test_ringbuf/clock/m_duty_cycle	0.5
/test_ringbuf/clock/m_period	200
/test_ringbuf/clock/sig	true
/test_ringbuf/ring_INST/txc	1
/test_ringbuf/ring_INST/clock	1
/test_ringbuf/ring_INST/reset	1
/test_ringbuf/ring_INST/txda	1
/test_ringbuf/ring_INST/outstrobe	0
/test_ringbuf/ring_INST/rxda	0
/test_ringbuf/ring_INST/ramadrs	110111111
/test_ringbuf/ring_INST/buffer	0111010110100111
/test_ringbuf/ring_INST/oeenable	0
/test_ringbuf/ring_INST/block1/clock	1
/test_ringbuf/ring_INST/block1/reset	1
/test_ringbuf/ring_INST/block1/ramadrs	110111111
/test_ringbuf/ring_INST/block1/oeenable	0
/test_ringbuf/ring_INST/block1/txc	1
/test_ringbuf/ring_INST/block1/outstrobe	0
/test_ringbuf/ring_INST/block2/clock	1
/test_ringbuf/ring_INST/block2/reset	1
/test_ringbuf/ring_INST/block2/oeenable	0
/test_ringbuf/ring_INST/block2/txda	1
/test_ringbuf/ring_INST/block2/ramadrs	110111111
/test_ringbuf/ring_INST/block2/buffer	0111010110100111
/test_ringbuf/ring_INST/block3/outstrobe	0
/test_ringbuf/ring_INST/block3/ramadrs	110111111
/test_ringbuf/ring_INST/block3/buffer	0111010110100111
/test_ringbuf/ring_INST/block3/rxda	0
/test_ringbuf/ring_INST/block3/rd0a	0
Now	500000 ns
Cursor 1	499300 ns

```
{
public:
                                         0->1
    sc_in<sc_logic>                      clock;
                                         1
    sc_in<sc_logic>                      reset;
                                         0
    sc_in<sc_logic>                      oeenable;
                                         1
    sc_in<sc_logic>                      txda;
                                         110111111
        c_lv<9> >                        ramadrs;
                                         0111010110100111
    sc_out<sc_lv<16> >                   buffer;
                                         /test_ringbuf/ring_INST/block2/buffer
                                         0111010110100111
    void storer()

    SC_CTOR(store)
      0->1
    : clock("clock"),
      1
      reset("reset"),
      0
```

Now: 500 us Delta: 2   sim:/test_ringbuf/ring_INST/block2   499095 ns to 499747 ns   0   Showing All Contexts

Graphical Dataflow: Tracing Signals



```
wire test, test2, _rw, test_in;
     1     1      0    z

not  (_rw, rw);
      0    1
nand (test, _rw, test2);
      1     0    1
nand (test2, _rw, test_in);
      1      0    z
nand (t_out, test, strb);
      0      1     1
```

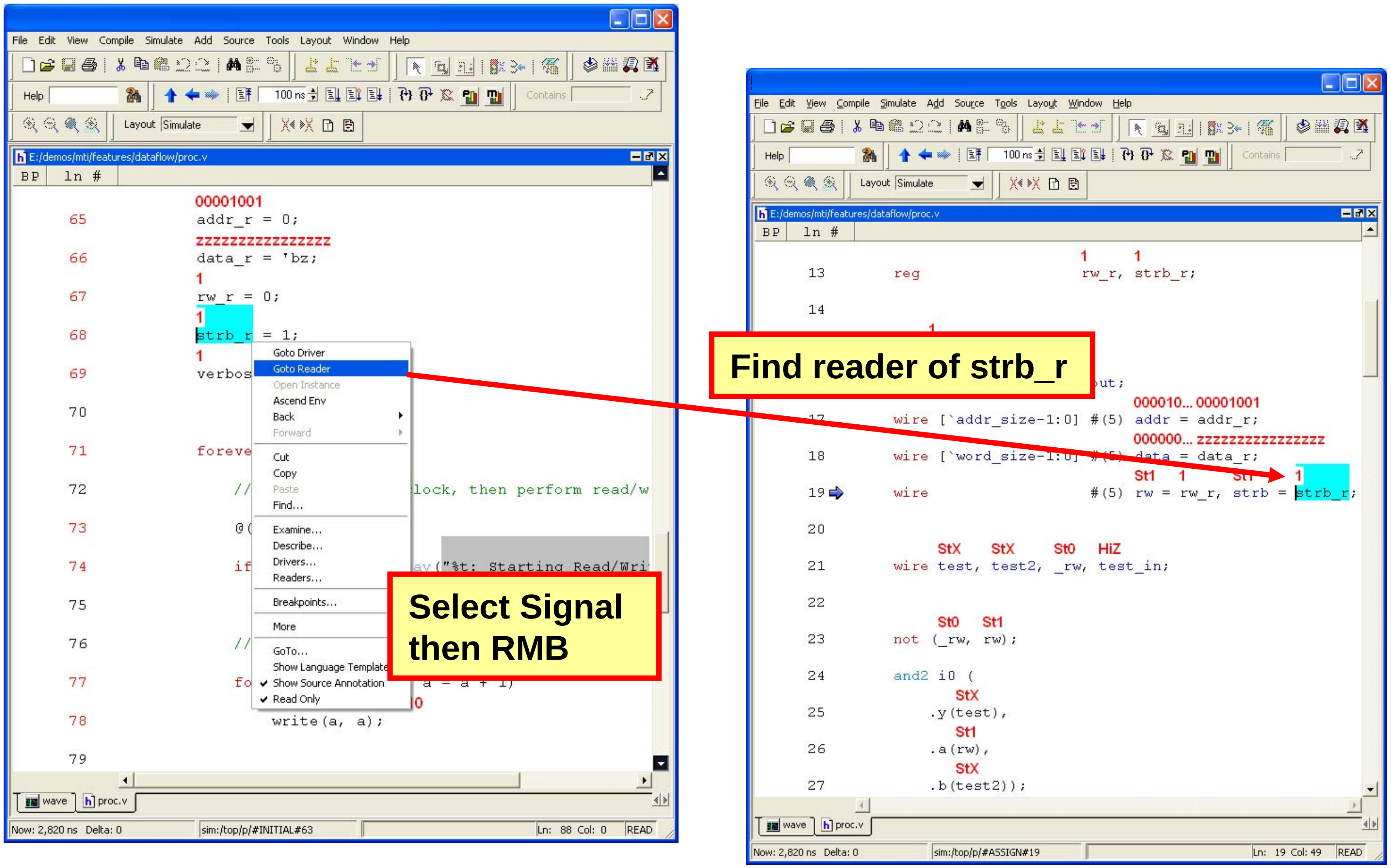
```
Ln: 26 Col: 0   READ
```

```
Inputs:
  /top/p/test      x
  /top/p/strb      1
Outputs:
  /top/p/t_out     x
  /top/p/test      x
Now       2820 ns
Cursor 1  2000 ns
Keep  1   /top/p/test_in
```

```
op/p/rdy          0
op/p/addr         09
op/p/rw           0
op/p/strb         1
top/p/data        0009
top/p/addr_r      09
/top/p/data_r     0009
/top/p/rw_r       0
/top/p/strb_r     1
/top/p/verbose    1
/top/p/t_out      x
/top/p/_rw        1
Now       20 ns
Cursor 1  00 ns
1811 ns to 2112 ns    Now: 2,820 ns  Delta: 0
```

- Users can also direct dataflow window to compute and draw paths between one point and another

Textual Dataflow: Tracing Signals



```
00001001
addr_r = 0;
zzzzzzzzzzzzzzzz
data_r = 'bz;
1
rw_r = 0;
1
strb_r = 1;
1
verbos

forever
    //                        lock, then perform read/w
    @(
    if                        ay("%t: Starting Read/Wri

    //
    fo                         a = a + 1)
                          0
        write(a, a);
```

Now: 2,820 ns Delta: 0 sim:/top/p/#INITIAL#63 Ln: 88 Col: 0 READ

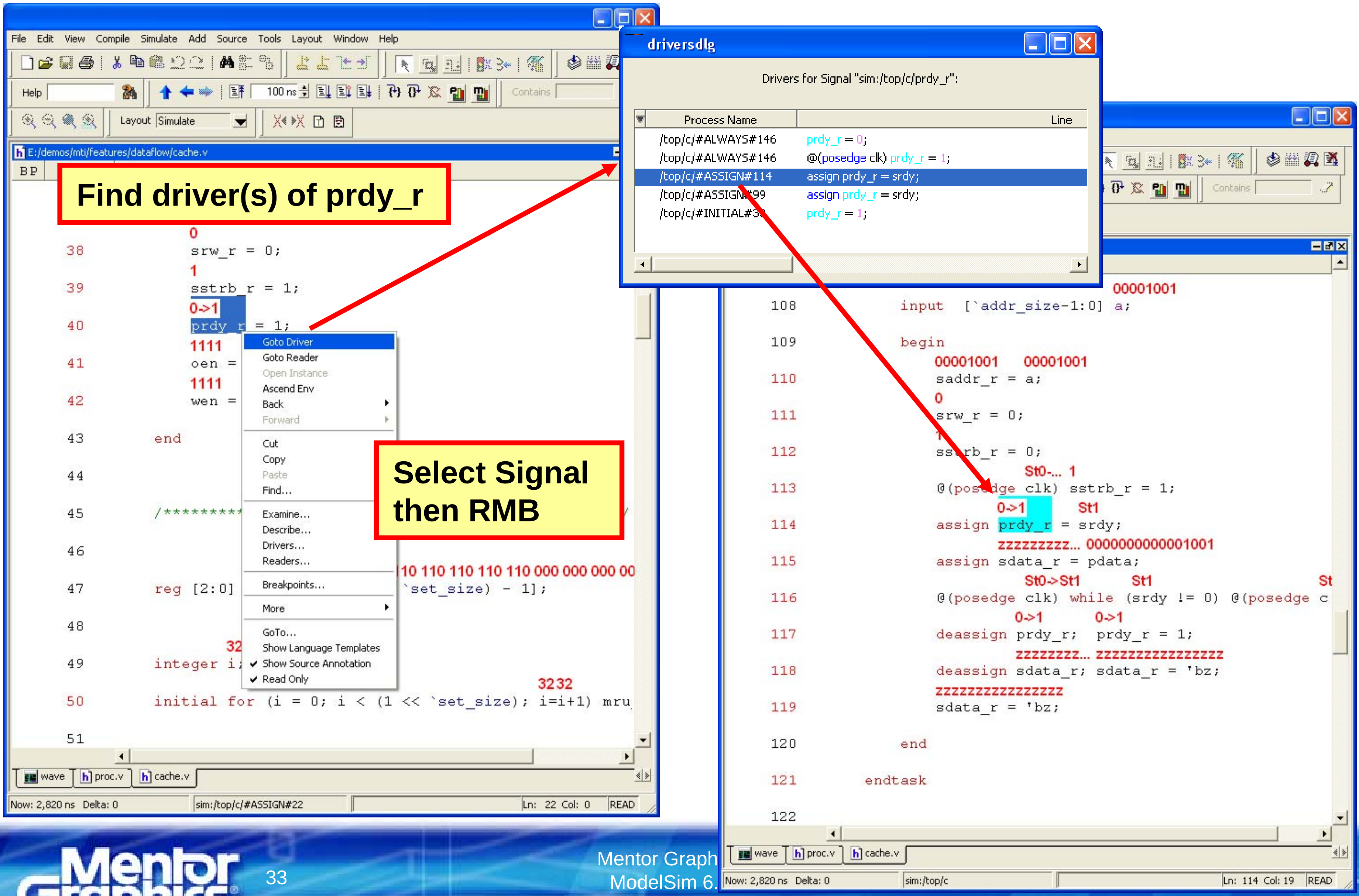
Select Signal then RMB

```
                                  1      1
reg                               rw_r, strb_r;

             1
                                        ut;
                                  000010... 00001001
wire [`addr_size-1:0] #(5) addr = addr_r;
                                  000000... zzzzzzzzzzzzzzzz
wire [`word_size-1:0] #(5) data = data_r;
                                  St1   1      St1    1
wire                       #(5) rw = rw_r, strb = strb_r;

     StX     StX     St0    HiZ
wire test, test2, _rw, test_in;

     St0   St1
not (_rw, rw);

and2 i0 (
       StX
    .y(test),
       St1
    .a(rw),
       StX
    .b(test2));
```

Now: 2,820 ns Delta: 0 sim:/top/p/#ASSIGN#19 Ln: 19 Col: 49 READ

Find reader of strb_r

Textual Dataflow: Tracing Signals



Find driver(s) of prdy_r

Select Signal then RMB

```
            0
            srw_r = 0;
            1
            sstrb_r = 1;
            0->1
            prdy_r = 1;
            1111
            oen =
            1111
            wen =
        end

        /*********
                              10 110 110 110 110 000 000 000 00
        reg [2:0]               `set_size) - 1];
                   32
        integer i;
                                                     3232
        initial for (i = 0; i < (1 << `set_size); i=i+1) mru
```

Now: 2,820 ns Delta: 0   sim:/top/c/#ASSIGN#22   Ln: 22 Col: 0   READ

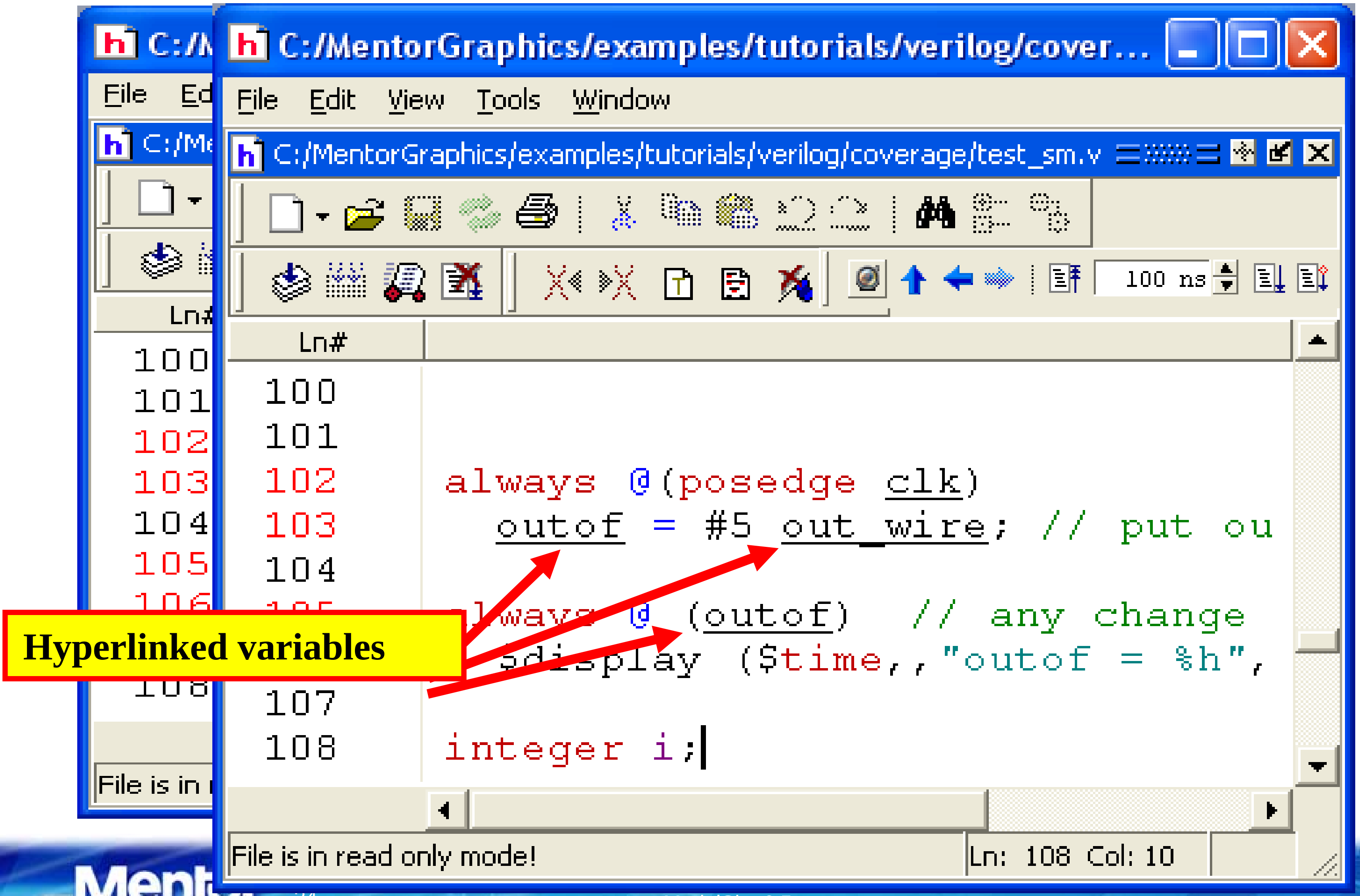
Drivers for Signal "sim:/top/c/prdy_r":

Process Name	Line
/top/c/#ALWAYS#146	prdy_r = 0;
/top/c/#ALWAYS#146	@(posedge clk) prdy_r = 1;
/top/c/#ASSIGN#114	assign prdy_r = srdy;
/top/c/#ASSIGN#99	assign prdy_r = srdy;
/top/c/#INITIAL#3	prdy_r = 1;

```
                                                  00001001
        input  [`addr_size-1:0] a;
        begin
            00001001    00001001
            saddr_r = a;
            0
            srw_r = 0;
            1
            sstrb_r = 0;
                  St0-...  1
            @(posedge clk) sstrb_r = 1;
                 0->1      St1
            assign prdy_r = srdy;
                 zzzzzzzzz... 0000000000001001
            assign sdata_r = pdata;
                  St0->St1      St1                       St
            @(posedge clk) while (srdy != 0) @(posedge c
                   0->1       0->1
            deassign prdy_r;  prdy_r = 1;
                   zzzzzzzz... zzzzzzzzzzzzzzzz
            deassign sdata_r; sdata_r = 'bz;
            zzzzzzzzzzzzzzzz
            sdata_r = 'bz;
        end
    endtask
```

Now: 2,820 ns Delta: 0   sim:/top/c   Ln: 114 Col: 19   READ

Easing Causality Tracing
Source Code Hyperlinking


C:/MentorGraphics/examples/tutorials/verilog/cover...
File Edit View Tools Window
C:/MentorGraphics/examples/tutorials/verilog/coverage/test_sm.v
100 ns
Ln#

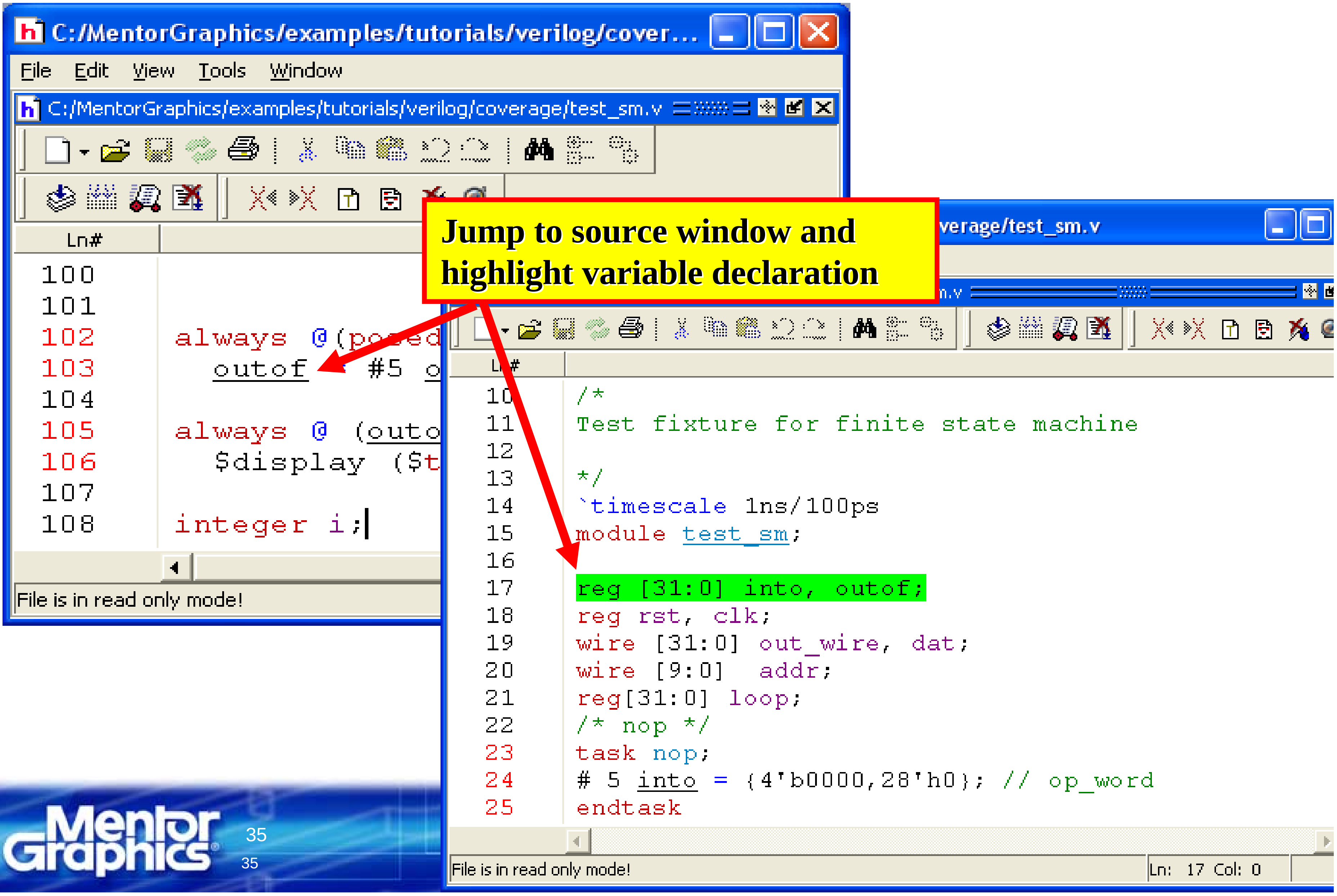
```
always @(posedge clk)
   outof = #5 out_wire; // put ou

always @ (outof)   // any change
   $display ($time,,"outof = %h",

integer i;
```

File is in read only mode!

Ln: 108 Col: 10

Hyperlinked variables

Easing Causality Tracing
Source Code Hyperlinking



C:/MentorGraphics/examples/tutorials/verilog/cover...

C:/MentorGraphics/examples/tutorials/verilog/coverage/test_sm.v

```
always @(posed
  outof   #5  o

always @  (outo
  $display ($t

integer i;
```

File is in read only mode!

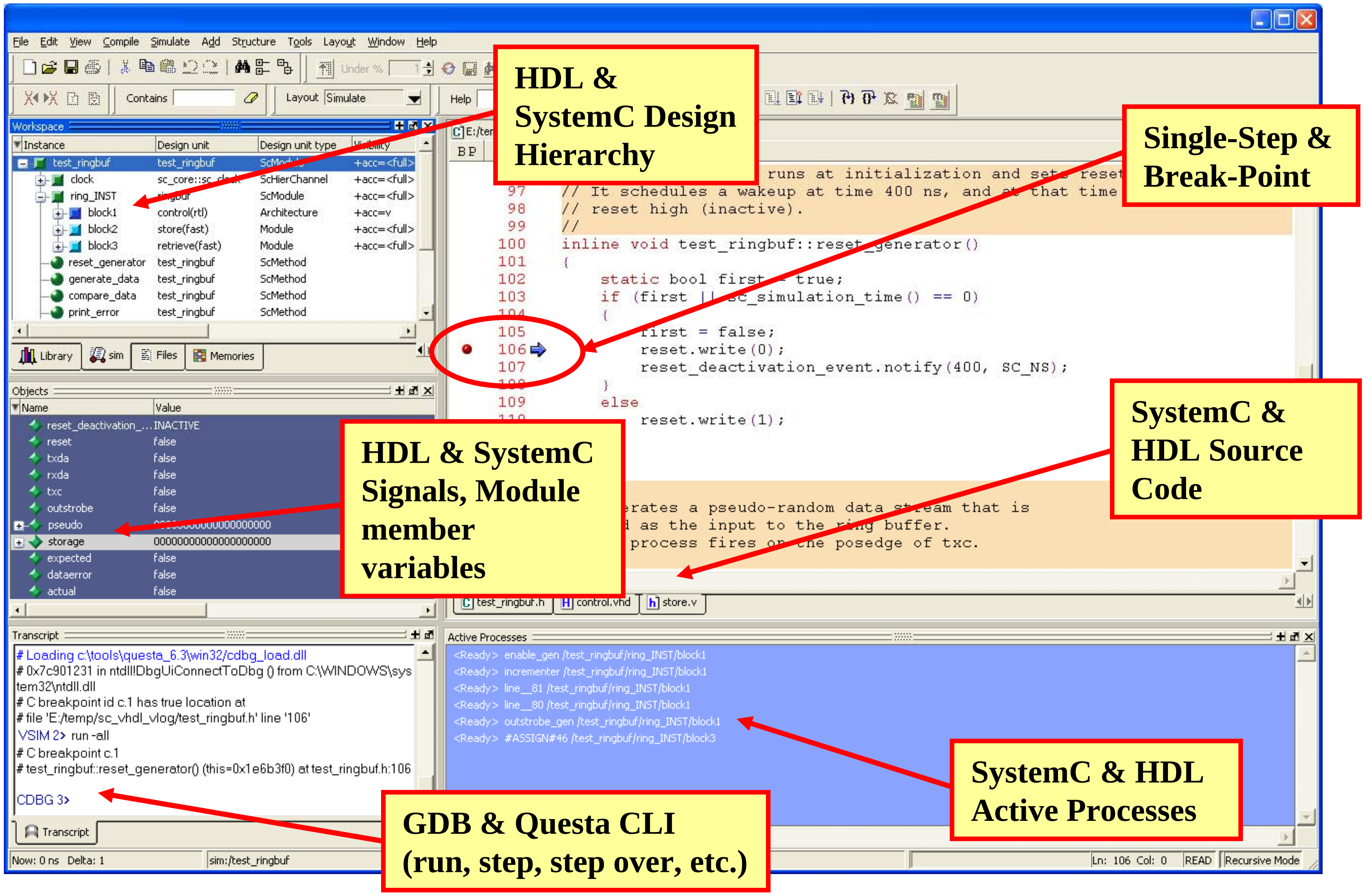
Jump to source window and highlight variable declaration

verage/test_sm.v

```
/*
Test fixture for finite state machine

*/
`timescale 1ns/100ps
module test_sm;

reg [31:0] into, outof;
reg rst, clk;
wire [31:0] out_wire, dat;
wire [9:0]  addr;
reg[31:0] loop;
/* nop */
task nop;
# 5 into = {4'b0000,28'h0}; // op_word
endtask
```

File is in read only mode! Ln: 17 Col: 0

Debug of SystemC & HDLs



File Edit View Compile Simulate Add Structure Tools Layout Window Help
Contains
Layout Simulate
Help
Workspace
Library
sim
Files
Memories
Objects
Transcript
Active Processes
HDL & SystemC Design Hierarchy
Single-Step & Break-Point
SystemC & HDL Source Code
HDL & SystemC Signals, Module member variables
GDB & Questa CLI (run, step, step over, etc.)
SystemC & HDL Active Processes

Instance	Design unit	Design unit type	Visibility
test_ringbuf	test_ringbuf	ScModule	+acc=<full>
clock	sc_core::sc_clock	ScHierChannel	+acc=<full>
ring_INST	ringbuf	ScModule	+acc=<full>
block1	control(rtl)	Architecture	+acc=v
block2	store(fast)	Module	+acc=<full>
block3	retrieve(fast)	Module	+acc=<full>
reset_generator	test_ringbuf	ScMethod	
generate_data	test_ringbuf	ScMethod	
compare_data	test_ringbuf	ScMethod	
print_error	test_ringbuf	ScMethod	

Name	Value
reset_deactivation_...	INACTIVE
reset	false
txda	false
rxda	false
txc	false
outstrobe	false
pseudo	00000000000000000000
storage	00000000000000000000
expected	false
dataerror	false
actual	false

```
C] E:/ter
BP
          runs at initialization and sets reset
97      // It schedules a wakeup at time 400 ns, and at that time
98      // reset high (inactive).
99      //
100     inline void test_ringbuf::reset_generator()
101     {
102         static bool first = true;
103         if (first || sc_simulation_time() == 0)
104         {
105             first = false;
106 ⇨           reset.write(0);
107             reset_deactivation_event.notify(400, SC_NS);
108         }
109         else
110             reset.write(1);

          erates a pseudo-random data stream that is
          d as the input to the ring buffer.
          process fires on the posedge of txc.
```

test_ringbuf.h | control.vhd | store.v

```
# Loading c:\tools\questa_6.3\win32/cdbg_load.dll
# 0x7c901231 in ntdll!DbgUiConnectToDbg () from C:\WINDOWS\sys
tem32\ntdll.dll
# C breakpoint id c.1 has true location at
# file 'E:/temp/sc_vhdl_vlog/test_ringbuf.h' line '106'
VSIM 2> run -all
# C breakpoint c.1
# test_ringbuf::reset_generator() (this=0x1e6b3f0) at test_ringbuf.h:106

CDBG 3>
```

```
<Ready> enable_gen /test_ringbuf/ring_INST/block1
<Ready> incrementer /test_ringbuf/ring_INST/block1
<Ready> line__81 /test_ringbuf/ring_INST/block1
<Ready> line__80 /test_ringbuf/ring_INST/block1
<Ready> outstrobe_gen /test_ringbuf/ring_INST/block1
<Ready> #ASSIGN#46 /test_ringbuf/ring_INST/block3
```

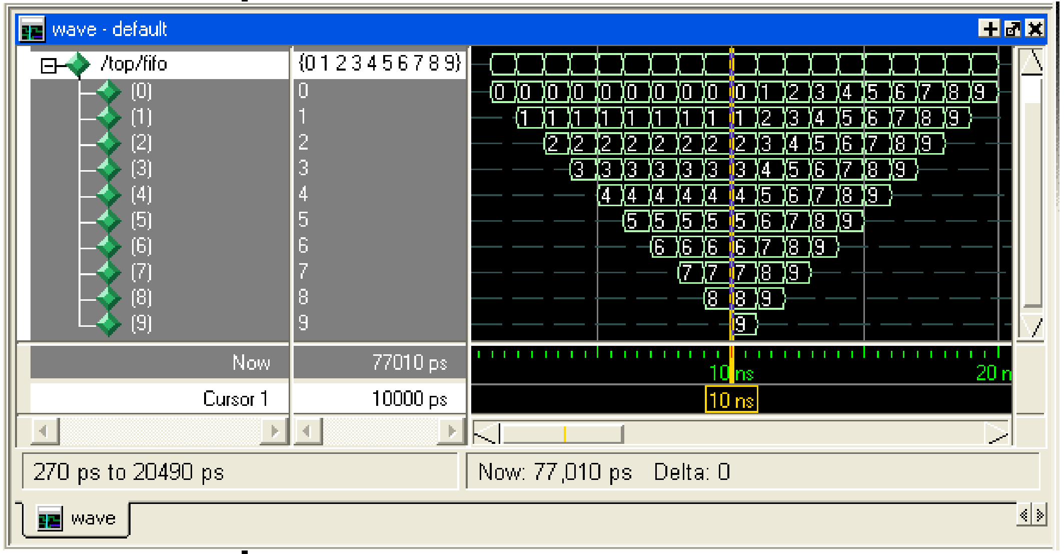
Now: 0 ns Delta: 1 | sim:/test_ringbuf | Ln: 106 Col: 0 | READ | Recursive Mode

Debug Example: `sc_fifo`

- **Writing to & Reading from `sc_fifo<long>` with 10 Elements**

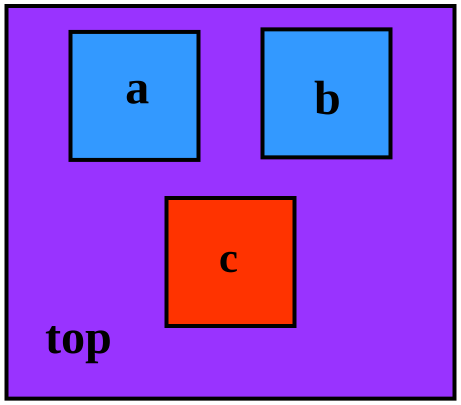
```
# 100: writing 0
# Executing 'examine fifo' yields
# {{0}}
# 200: writing 1
# Executing 'examine {fifo(0 to 1)}'
yields
# {{0 1}}
# 300: writing 2
# 400: writing 3
# 500: writing 4
# 600: writing 5
# 700: writing 6
# Executing 'examine {fifo(0 to 7)}'
yields
# {{0 1 2 3 4 5 6}}
# Executing 'examine {fifo(7)}' yields
# -Unused-
# 800: writing 7
# 900: writing 8
# 1000: writing 9
# 1100: Available: 10
```

```
# 1100: reading 0
# 1200: Available: 9
# 1200: reading 1
# {{2 3 4 5 6 7 8 9}}
# 1300: Available: 8
# 1300: reading 2
# 1400: Available: 7
# 1400: reading 3
# 1500: Available: 6
# 1500: reading 4
# 1600: Available: 5
# 1600: reading 5
# 1700: Available: 4
# 1700: reading 6
# {{7 8 9}}
```



Failure Isolation – Time To Debug

- Identify a sub-block from a chip/system environment
  - Save extended VCD data from sub-block of system/chip environment
- Improved time to debug
  - Transfer sub-block to design team with most expertise
  - Re-simulate sub-block with extended VCD file

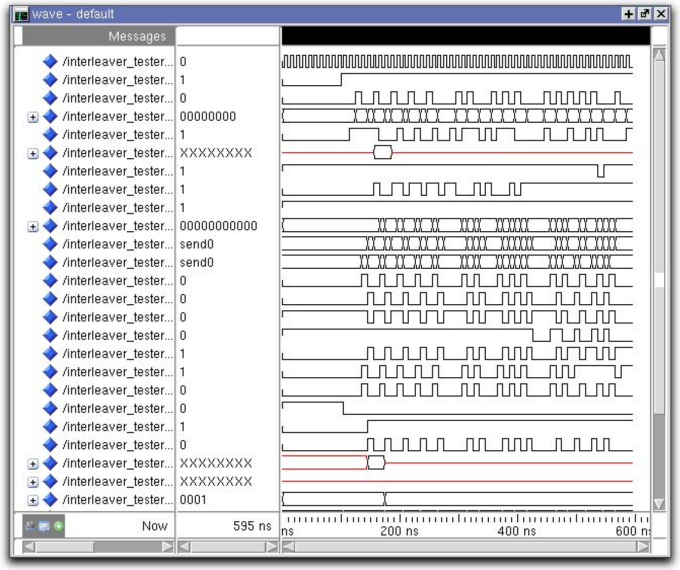


```
VSIM 1> vcd dumpports –file c.vcd /top/c/*
VSIM 2> run -all
```

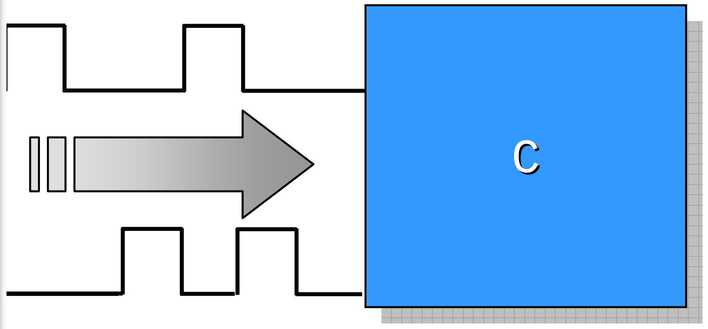
```
vsim c –vcdstim c.vcd
```

Using VCD as Stimulus

VCD file



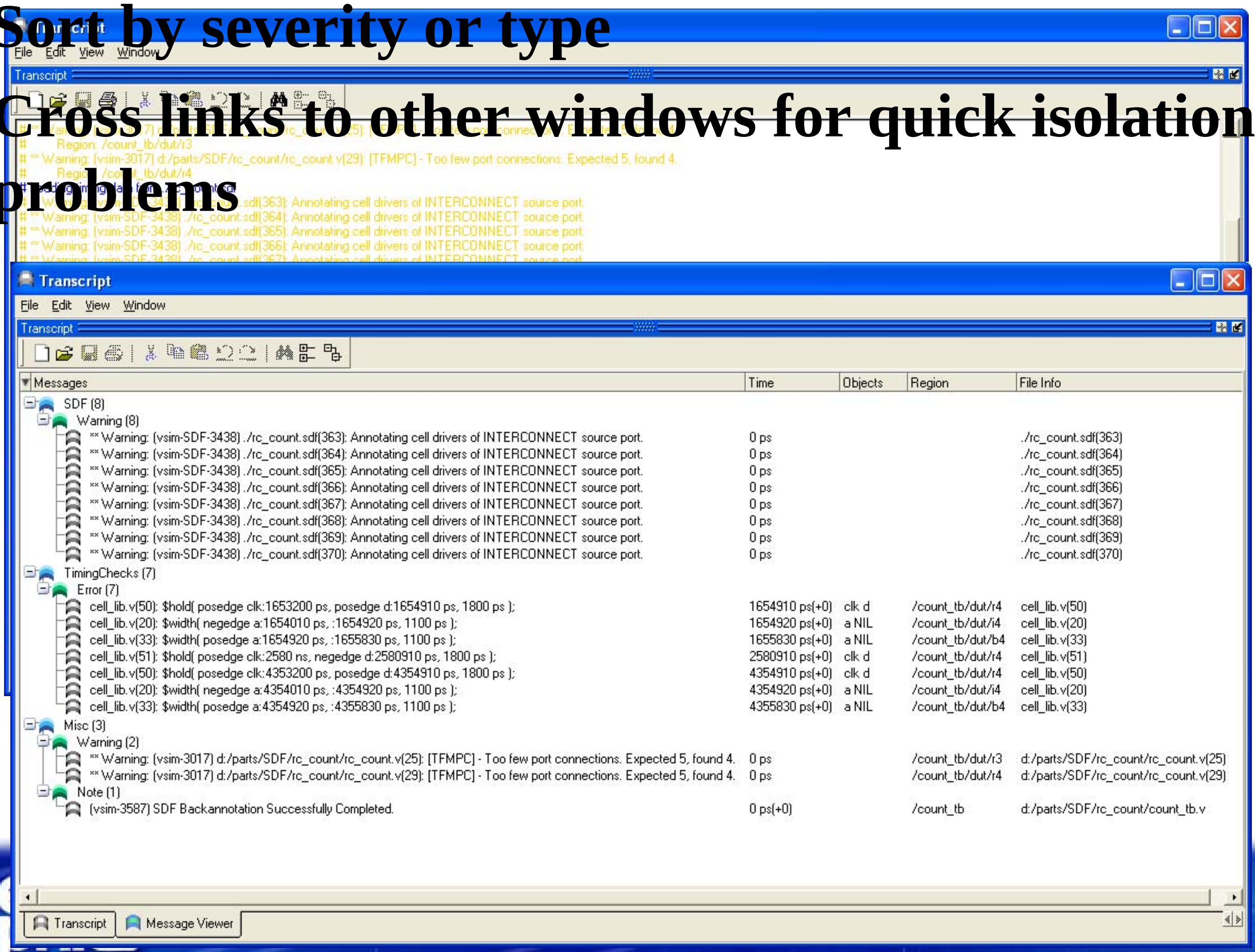
```
/interleaver_tester... 0
/interleaver_tester... 1
/interleaver_tester... 0
/interleaver_tester... 00000000
/interleaver_tester... 1
/interleaver_tester... XXXXXXXX
/interleaver_tester... 1
/interleaver_tester... 1
/interleaver_tester... 1
/interleaver_tester... 000000000000
/interleaver_tester... send0
/interleaver_tester... send0
/interleaver_tester... 0
/interleaver_tester... 0
/interleaver_tester... 0
/interleaver_tester... 0
/interleaver_tester... 1
/interleaver_tester... 1
/interleaver_tester... 0
/interleaver_tester... 0
/interleaver_tester... 1
/interleaver_tester... 0
/interleaver_tester... XXXXXXXX
/interleaver_tester... XXXXXXXX
/interleaver_tester... 0001
Now 595 ns
```



$ vsim c –vcdstim c.vcd

Expanded Data requires Data Management Message Viewer

- Organize all simulation messages
- Sort by severity or type
- Cross links to other windows for quick isolation of problems



```
# Region: /count_tb/dut/r3
# ** Warning: (vsim-3017) d:/parts/SDF/rc_count/rc_count.v(29): [TFMPC] - Too few port connections. Expected 5, found 4.
# Region: /count_tb/dut/r4
# ** Warning: (vsim-SDF-3438) ./rc_count.sdf(364): Annotating cell drivers of INTERCONNECT source port.
# ** Warning: (vsim-SDF-3438) ./rc_count.sdf(365): Annotating cell drivers of INTERCONNECT source port.
# ** Warning: (vsim-SDF-3438) ./rc_count.sdf(366): Annotating cell drivers of INTERCONNECT source port.
```

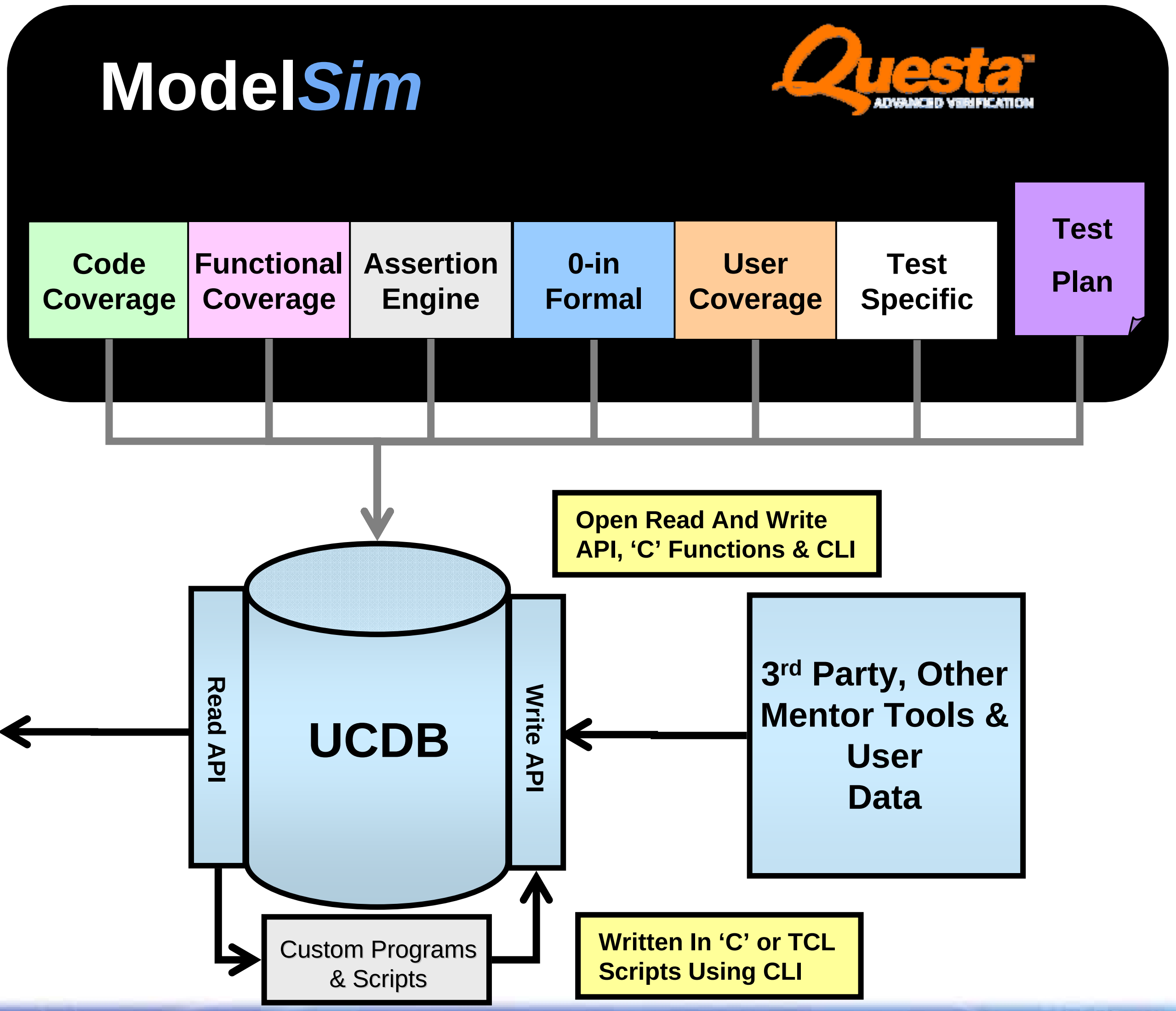
Messages	Time	Objects	Region	File Info
SDF (8)				
Warning (8)				
** Warning: (vsim-SDF-3438) ./rc_count.sdf(363): Annotating cell drivers of INTERCONNECT source port.	0 ps			./rc_count.sdf(363)
** Warning: (vsim-SDF-3438) ./rc_count.sdf(364): Annotating cell drivers of INTERCONNECT source port.	0 ps			./rc_count.sdf(364)
** Warning: (vsim-SDF-3438) ./rc_count.sdf(365): Annotating cell drivers of INTERCONNECT source port.	0 ps			./rc_count.sdf(365)
** Warning: (vsim-SDF-3438) ./rc_count.sdf(366): Annotating cell drivers of INTERCONNECT source port.	0 ps			./rc_count.sdf(366)
** Warning: (vsim-SDF-3438) ./rc_count.sdf(367): Annotating cell drivers of INTERCONNECT source port.	0 ps			./rc_count.sdf(367)
** Warning: (vsim-SDF-3438) ./rc_count.sdf(368): Annotating cell drivers of INTERCONNECT source port.	0 ps			./rc_count.sdf(368)
** Warning: (vsim-SDF-3438) ./rc_count.sdf(369): Annotating cell drivers of INTERCONNECT source port.	0 ps			./rc_count.sdf(369)
** Warning: (vsim-SDF-3438) ./rc_count.sdf(370): Annotating cell drivers of INTERCONNECT source port.	0 ps			./rc_count.sdf(370)
TimingChecks (7)				
Error (7)				
cell_lib.v(50): $hold(posedge clk:1653200 ps, posedge d:1654910 ps, 1800 ps);	1654910 ps(+0)	clk d	/count_tb/dut/r4	cell_lib.v(50)
cell_lib.v(20): $width(negedge a:1654010 ps, :1654920 ps, 1100 ps);	1654920 ps(+0)	a NIL	/count_tb/dut/i4	cell_lib.v(20)
cell_lib.v(33): $width(posedge a:1654920 ps, :1655830 ps, 1100 ps);	1655830 ps(+0)	a NIL	/count_tb/dut/b4	cell_lib.v(33)
cell_lib.v(51): $hold(posedge clk:2580 ns, negedge d:2580910 ps, 1800 ps);	2580910 ps(+0)	clk d	/count_tb/dut/r4	cell_lib.v(51)
cell_lib.v(50): $hold(posedge clk:4353200 ps, posedge d:4354910 ps, 1800 ps);	4354910 ps(+0)	clk d	/count_tb/dut/r4	cell_lib.v(50)
cell_lib.v(20): $width(negedge a:4354010 ps, :4354920 ps, 1100 ps);	4354920 ps(+0)	a NIL	/count_tb/dut/i4	cell_lib.v(20)
cell_lib.v(33): $width(posedge a:4354920 ps, :4355830 ps, 1100 ps);	4355830 ps(+0)	a NIL	/count_tb/dut/b4	cell_lib.v(33)
Misc (3)				
Warning (2)				
** Warning: (vsim-3017) d:/parts/SDF/rc_count/rc_count.v(25): [TFMPC] - Too few port connections. Expected 5, found 4.	0 ps		/count_tb/dut/r3	d:/parts/SDF/rc_count/rc_count.v(25)
** Warning: (vsim-3017) d:/parts/SDF/rc_count/rc_count.v(29): [TFMPC] - Too few port connections. Expected 5, found 4.	0 ps		/count_tb/dut/r4	d:/parts/SDF/rc_count/rc_count.v(29)
Note (1)				
(vsim-3587) SDF Backannotation Successfully Completed.	0 ps(+0)		/count_tb	d:/parts/SDF/rc_count/count_tb.v

Agenda

- **Performance**
- **Debug & Analysis**
- **Coverage**

Unified Coverage DataBase (UCDB)

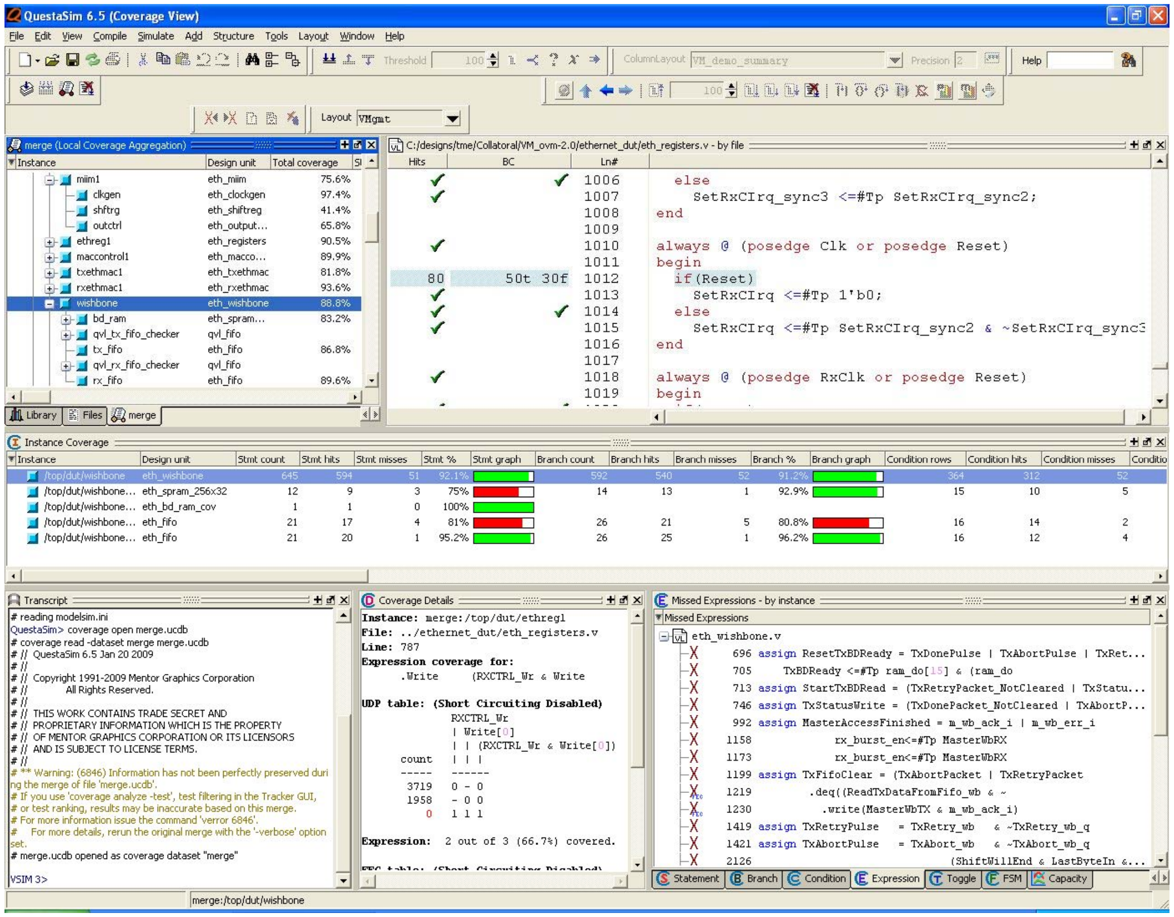
- Best capacity and performance
- Most comprehensive
- Open and flexible
- Base technology for Questa Verification Management



ModelSim
Questa
ADVANCED VERIFICATION
Code Coverage
Functional Coverage
Assertion Engine
0-in Formal
User Coverage
Test Specific
Test Plan
Open Read And Write API, 'C' Functions & CLI
Verification Coverage Analysis & Reporting
Read API
UCDB
Write API
3rd Party, Other Mentor Tools & User Data
Database Toolset, load, copy, merge, rank, report, analyze
Custom Programs & Scripts
Written In 'C' or TCL Scripts Using CLI

Code Coverage

- Measures language coverage
  - Have you executed each:
    - Statement
    - Branch
    - Condition
    - Expression
    - Or Toggled each bit
- Best used at block level
  - Easier to exercise code aspects
  - Ensures blocks are tested & ready for integration
- Built-in
  - Low overhead
    - compatible with most optimizations
  - Easy to use
  - High capacity and performance UCDB
- Improve verification throughput
  - Rank UCDB test files and eliminate regression tests that do not contribute to coverage metrics



QuestaSim 6.5 (Coverage View)
File Edit View Compile Simulate Add Structure Tools Layout Window Help
Threshold 100
ColumnLayout VM_demo_summary
Precision 2
Help
Layout VMgmt
Library
Files
merge
Statement
Branch
Condition
Expression
Toggle
FSM
Capacity

merge (Local Coverage Aggregation)

Instance	Design unit	Total coverage
miim1	eth_miim	75.6%
clkgen	eth_clockgen	97.4%
shftrg	eth_shiftreg	41.4%
outctrl	eth_output...	65.8%
ethreg1	eth_registers	90.5%
maccontrol1	eth_macco...	89.9%
txethmac1	eth_txethmac	81.8%
rxethmac1	eth_rxethmac	93.6%
wishbone	eth_wishbone	88.8%
bd_ram	eth_spram...	83.2%
qvl_tx_fifo_checker	qvl_fifo	
tx_fifo	eth_fifo	86.8%
qvl_rx_fifo_checker	qvl_fifo	
rx_fifo	eth_fifo	89.6%

C:/designs/tme/Collatoral/VM_ovm-2.0/ethernet_dut/eth_registers.v - by file

```
Hits     BC         Ln#
 ✓           ✓     1006      else
 ✓                 1007        SetRxCIrq_sync3 <=#Tp SetRxCIrq_sync2;
                   1008    end
                   1009
 ✓                 1010    always @ (posedge Clk or posedge Reset)
                   1011    begin
 80    50t 30f     1012      if(Reset)
 ✓                 1013        SetRxCIrq <=#Tp 1'b0;
 ✓           ✓     1014      else
 ✓                 1015        SetRxCIrq <=#Tp SetRxCIrq_sync2 & ~SetRxCIrq_sync3
                   1016    end
                   1017
 ✓                 1018    always @ (posedge RxClk or posedge Reset)
                   1019    begin
```

Instance Coverage

Instance	Design unit	Stmt count	Stmt hits	Stmt misses	Stmt %	Stmt graph	Branch count	Branch hits	Branch misses	Branch %	Branch graph	Condition rows	Condition hits	Condition misses
/top/dut/wishbone	eth_wishbone	645	594	51	92.1%	🟩	592	540	52	91.2%	🟩	364	312	52
/top/dut/wishbone...	eth_spram_256x32	12	9	3	75%	🟥	14	13	1	92.9%	🟩	15	10	5
/top/dut/wishbone...	eth_bd_ram_cov	1	1	0	100%	🟩								
/top/dut/wishbone...	eth_fifo	21	17	4	81%	🟥	26	21	5	80.8%	🟥	16	14	2
/top/dut/wishbone...	eth_fifo	21	20	1	95.2%	🟩	26	25	1	96.2%	🟩	16	12	4

Transcript

```
# reading modelsim.ini
QuestaSim> coverage open merge.ucdb
# coverage read -dataset merge merge.ucdb
# //  QuestaSim 6.5 Jan 20 2009
# //
# //  Copyright 1991-2009 Mentor Graphics Corporation
# //            All Rights Reserved.
# //
# //  THIS WORK CONTAINS TRADE SECRET AND
# //  PROPRIETARY INFORMATION WHICH IS THE PROPERTY
# //  OF MENTOR GRAPHICS CORPORATION OR ITS LICENSORS
# //  AND IS SUBJECT TO LICENSE TERMS.
# //
# ** Warning: (6846) Information has not been perfectly preserved duri
ng the merge of file 'merge.ucdb'.
# If you use 'coverage analyze -test', test filtering in the Tracker GUI,
# or test ranking, results may be inaccurate based on this merge.
# For more information issue the command 'verror 6846'.
#    For more details, rerun the original merge with the '-verbose' option
set.
# merge.ucdb opened as coverage dataset "merge"

VSIM 3>
```

Coverage Details

```
Instance: merge:/top/dut/ethreg1
File: ../ethernet_dut/eth_registers.v
Line: 787
Expression coverage for:
      .Write      (RXCTRL_Wr & Write

UDP table: (Short Circuiting Disabled)
              RXCTRL_Wr
              | Write[0]
              | | (RXCTRL_Wr & Write[0])
      count   | | |
      -----   ------
       3719   0 - 0
       1958   - 0 0
          0   1 1 1

Expression:  2 out of 3 (66.7%) covered.
```

Missed Expressions - by instance

```
Missed Expressions
eth_wishbone.v
  X      696 assign ResetTxBDReady = TxDonePulse | TxAbortPulse | TxRet...
  X      705     TxBDReady <=#Tp ram_do[15] & (ram_do
  X      713 assign StartTxBDRead = (TxRetryPacket_NotCleared | TxStatu...
  X      746 assign TxStatusWrite = (TxDonePacket_NotCleared | TxAbortP...
  X      992 assign MasterAccessFinished = m_wb_ack_i | m_wb_err_i
  X     1158            rx_burst_en<=#Tp MasterWbRX
  X     1173            rx_burst_en<=#Tp MasterWbRX
  X     1199 assign TxFifoClear = (TxAbortPacket | TxRetryPacket
  X FEC 1219        .deq((ReadTxDataFromFifo_wb & ~
  X FEC 1230          .write(MasterWbTX & m_wb_ack_i)
  X     1419 assign TxRetryPulse   = TxRetry_wb   & ~TxRetry_wb_q
  X     1421 assign TxAbortPulse   = TxAbort_wb   & ~TxAbort_wb_q
  X     2126                             (ShiftWillEnd & LastByteIn &...
```

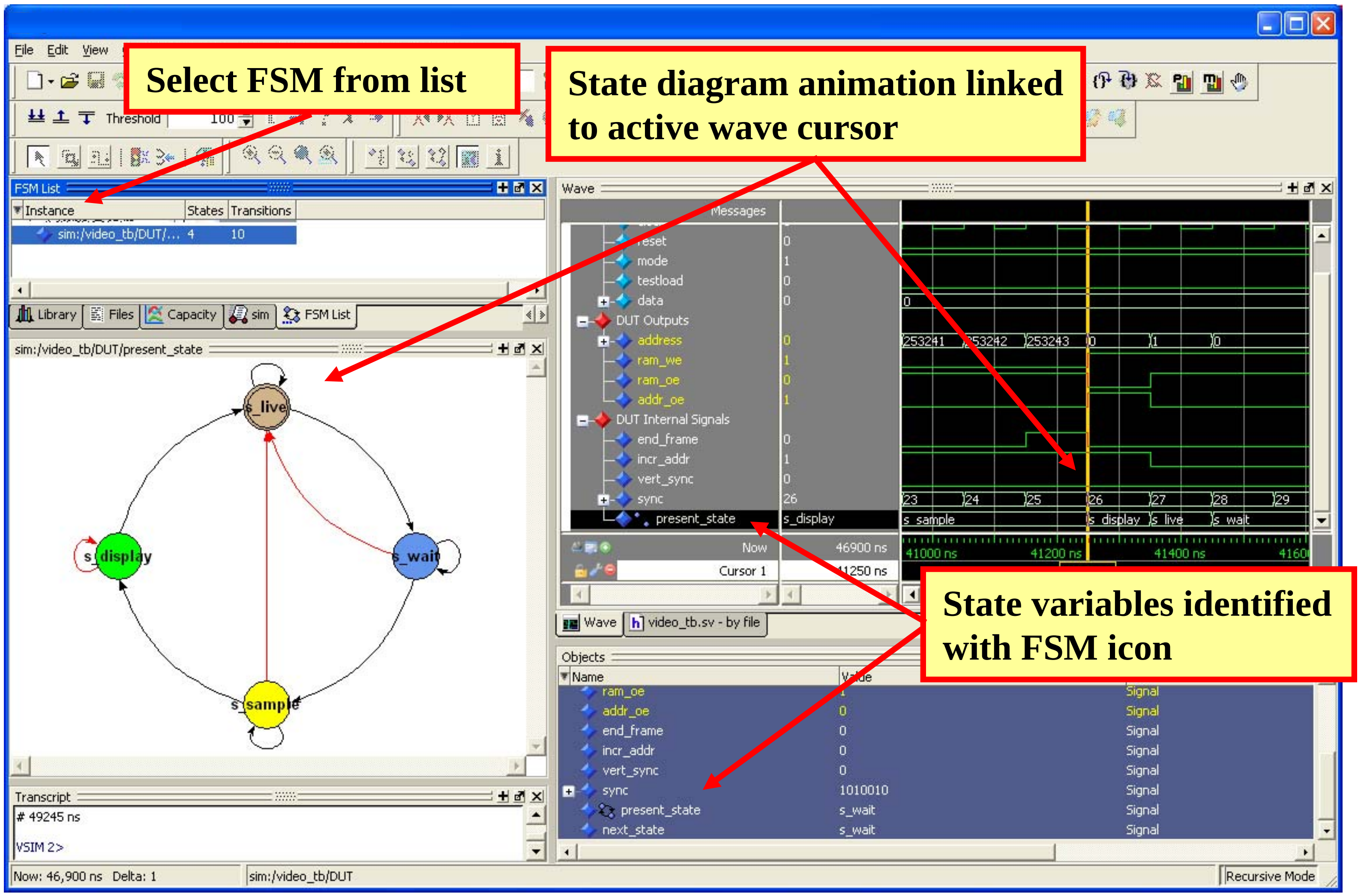
merge:/top/dut/wishbone

6.5 Coverage Update

- Improved FSM recognition
  - Must use vopt
- UCDB capacity
  - 20-30% memory footprint reduction
- Coverage remove
  - Elimination of test specific data from merged UCDB file
- Fine-grained exclusions
- UCDB preserve count merge
  - Enables detailed analysis of merged coverage data

Expanding Coverage Usability

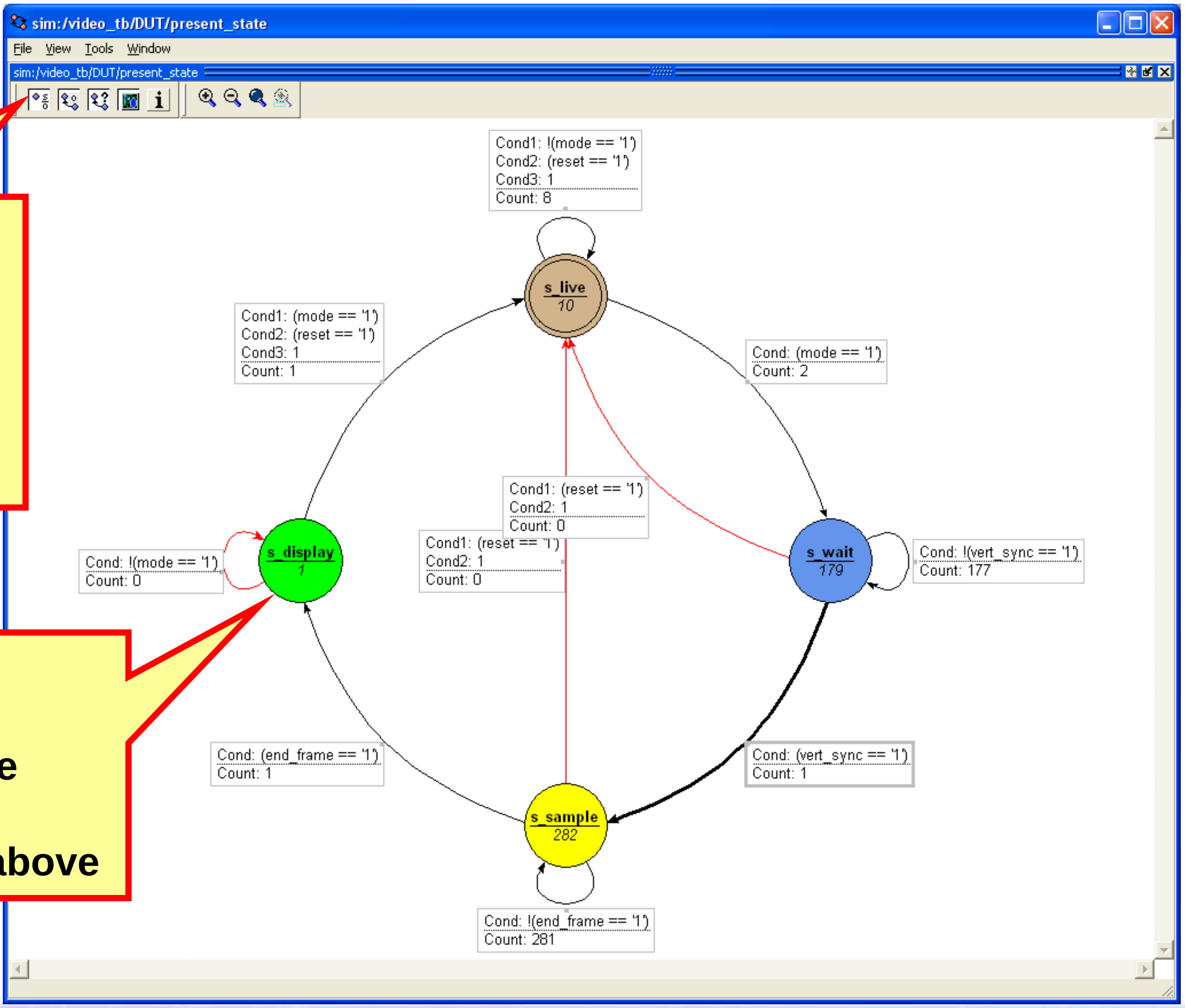
Comprehensive FSM Debug Solution



```
# 49245 ns

VSIM 2>
```

Name	Value	
ram_oe	1	Signal
addr_oe	0	Signal
end_frame	0	Signal
incr_addr	0	Signal
vert_sync	0	Signal
sync	1010010	Signal
present_state	s_wait	Signal
next_state	s_wait	Signal

FSM Debug with transition expressions



Cond1: !(mode == '1')
Cond2: (reset == '1')
Cond3: 1
Count: 8

Cond1: (mode == '1')
Cond2: (reset == '1')
Cond3: 1
Count: 1

Cond: (mode == '1')
Count: 2

Cond1: (reset == '1')
Cond2: 1
Count: 0

Cond1: (reset == '1')
Cond2: 1
Count: 0

Cond: !(mode == '1')
Count: 0

Cond: !(vert_sync == '1')
Count: 177

Cond: (end_frame == '1')
Count: 1

Cond: (vert_sync == '1')
Count: 1

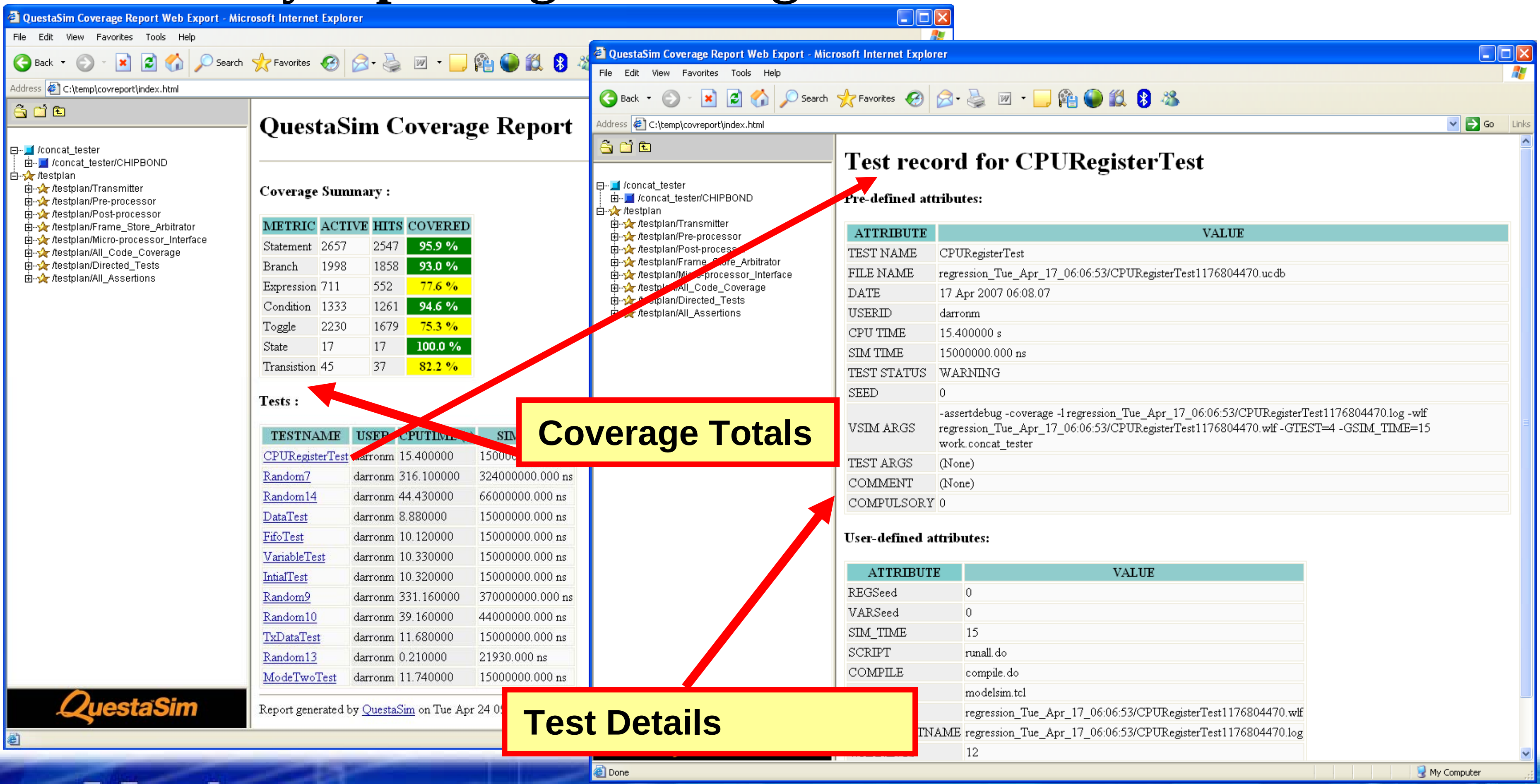
Cond: !(end_frame == '1')
Count: 281

Control display of:
- * Transition counts
- * State counts
- * Conditional paths
- * Balloon popups
- * Wave cursor linking

FSM states:
- * Brown - Reset state
- * Yellow - Previous state
- * Green - Present state
- * Blue - Not one of the above

Code Coverage Reporting Efficiencies

- Includes complete coverage result details
- Easy reporting to management



QuestaSim Coverage Report

Coverage Summary :

METRIC	ACTIVE	HITS	COVERED
Statement	2657	2547	95.9 %
Branch	1998	1858	93.0 %
Expression	711	552	77.6 %
Condition	1333	1261	94.6 %
Toggle	2230	1679	75.3 %
State	17	17	100.0 %
Transition	45	37	82.2 %

Tests :

TESTNAME	USER	CPUTIME	SIM
CPURegisterTest	arronm	15.400000	15000
Random7	darronm	316.100000	324000000.000 ns
Random14	darronm	44.430000	66000000.000 ns
DataTest	darronm	8.880000	15000000.000 ns
FifoTest	darronm	10.120000	15000000.000 ns
VariableTest	darronm	10.330000	15000000.000 ns
IntialTest	darronm	10.320000	15000000.000 ns
Random9	darronm	331.160000	370000000.000 ns
Random10	darronm	39.160000	44000000.000 ns
TxDataTest	darronm	11.680000	15000000.000 ns
Random13	darronm	0.210000	21930.000 ns
ModeTwoTest	darronm	11.740000	15000000.000 ns

Report generated by QuestaSim on Tue Apr 24 0

Test record for CPURegisterTest

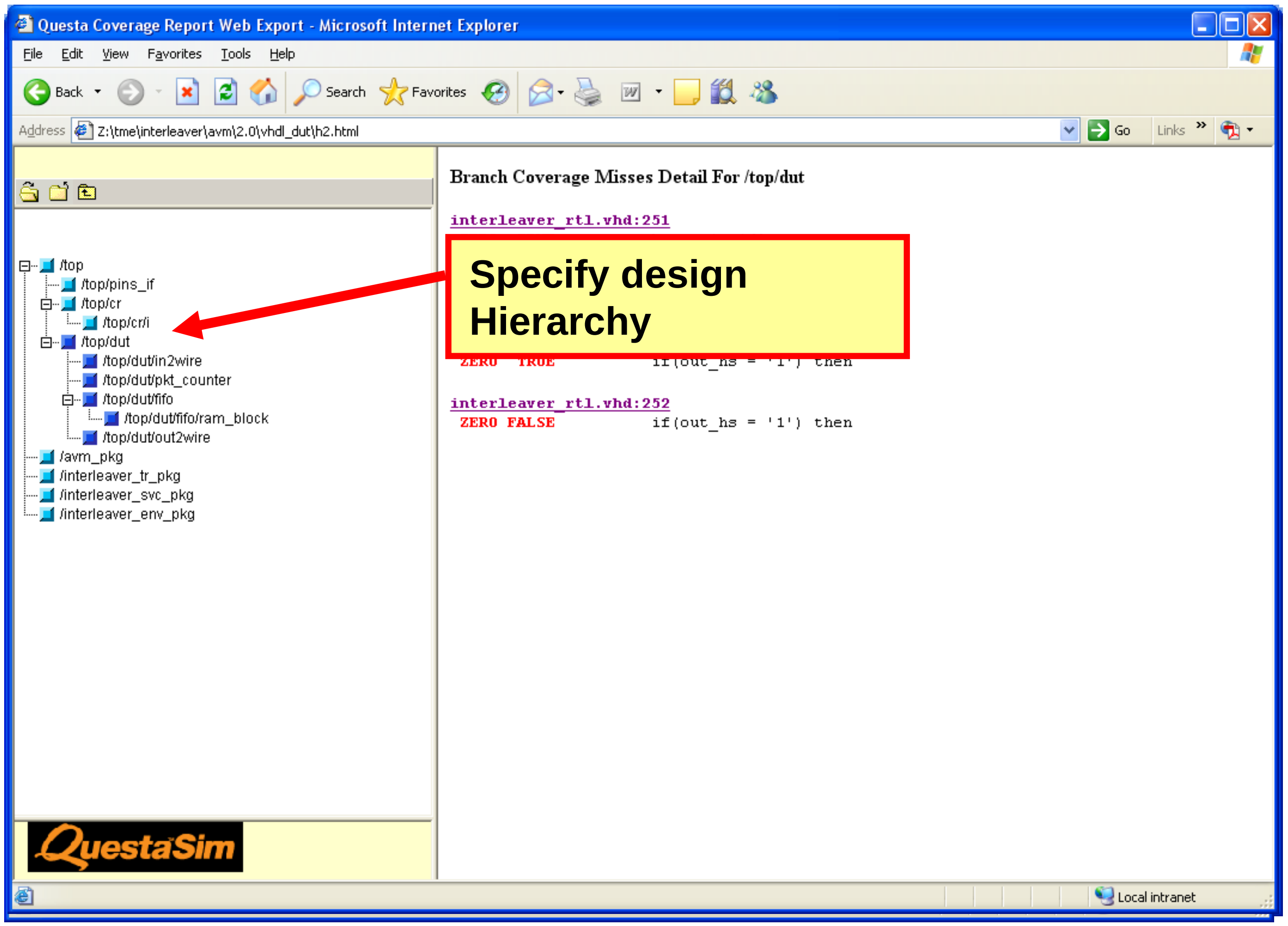
Pre-defined attributes:

ATTRIBUTE	VALUE
TEST NAME	CPURegisterTest
FILE NAME	regression_Tue_Apr_17_06:06:53/CPURegisterTest1176804470.ucdb
DATE	17 Apr 2007 06:08.07
USERID	darronm
CPU TIME	15.400000 s
SIM TIME	15000000.000 ns
TEST STATUS	WARNING
SEED	0
VSIM ARGS	-assertdebug -coverage -l regression_Tue_Apr_17_06:06:53/CPURegisterTest1176804470.log -wlf regression_Tue_Apr_17_06:06:53/CPURegisterTest1176804470.wlf -GTEST=4 -GSIM_TIME=15 work.concat_tester
TEST ARGS	(None)
COMMENT	(None)
COMPULSORY	0

User-defined attributes:

ATTRIBUTE	VALUE
REGSeed	0
VARSeed	0
SIM_TIME	15
SCRIPT	runall.do
COMPILE	compile.do
	modelsim.tcl
	regression_Tue_Apr_17_06:06:53/CPURegisterTest1176804470.wlf
TNAME	regression_Tue_Apr_17_06:06:53/CPURegisterTest1176804470.log
	12

HTML Coverage Viewing



Branch Coverage Misses Detail For /top/dut

interleaver_rtl.vhd:251

```
ZERO  TRUE            if(out_hs = '1') then
```

interleaver_rtl.vhd:252

```
ZERO  FALSE           if(out_hs = '1') then
```

Specify design Hierarchy

VHDL

- VHDL 2008 support introduced
  - Predefined operations on scalar (5.1.5) and array (5.2.1.3) types
  - External Names (8.7)
  - Package STANDARD (16.3)
  - Standard environment package (16.5)
  - VHDL Encryption (24.1)
- Enable by setting vhdl93 = 2008 in modelsim.ini

ModelSim 6.5 Functional Verification

- **The best execution**
  - **Integrated platform available today**
- **The best technology**
  - **High capacity, high performance and throughput**
- **The right strategy**
  - **Make every verification cycle count!**

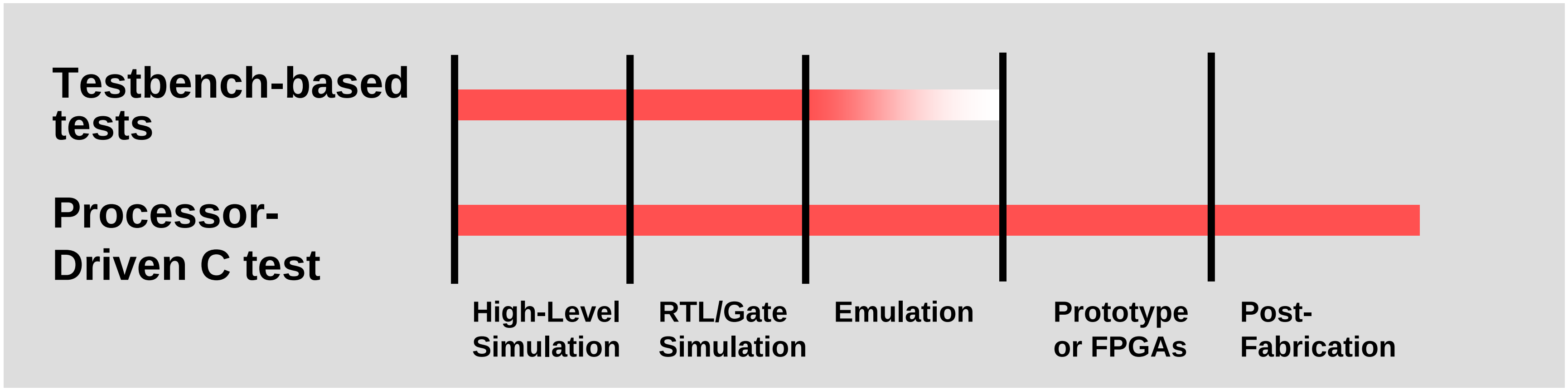
Mentor Graphics®

www.mentor.com

Additional Mentor Products

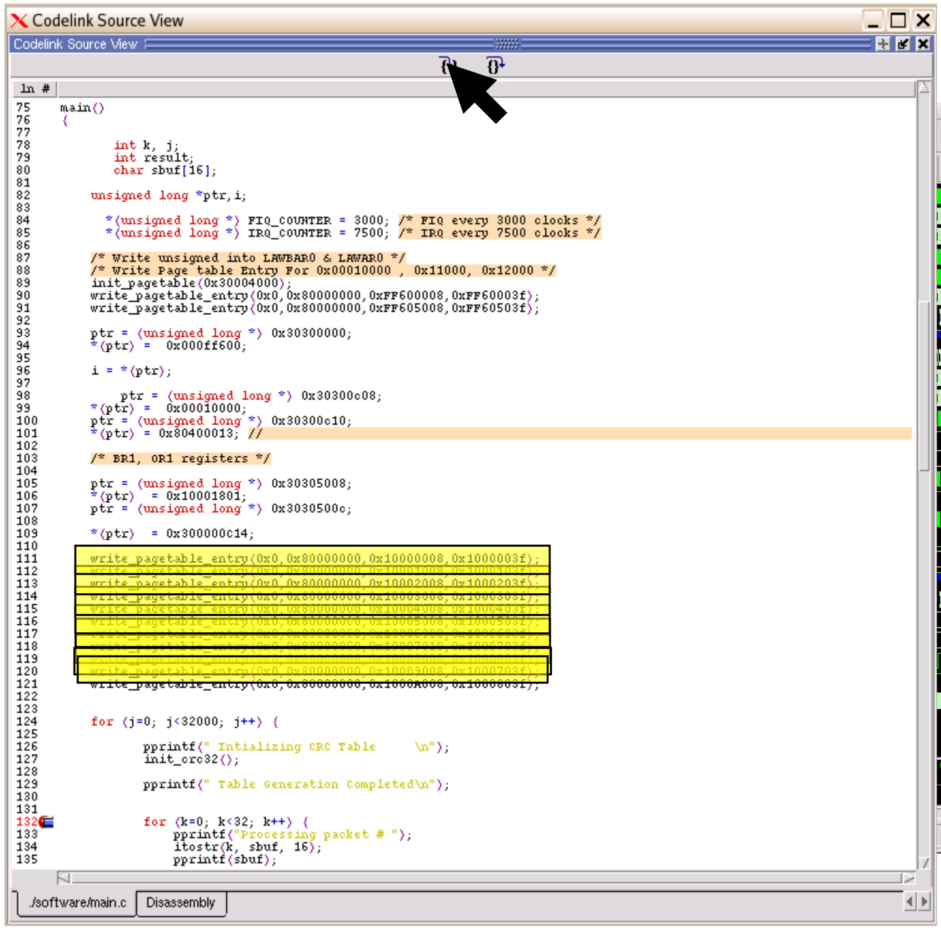
Processor-driven Verification

- Allows test efforts to span multiple stages of the design - test reuse across the project

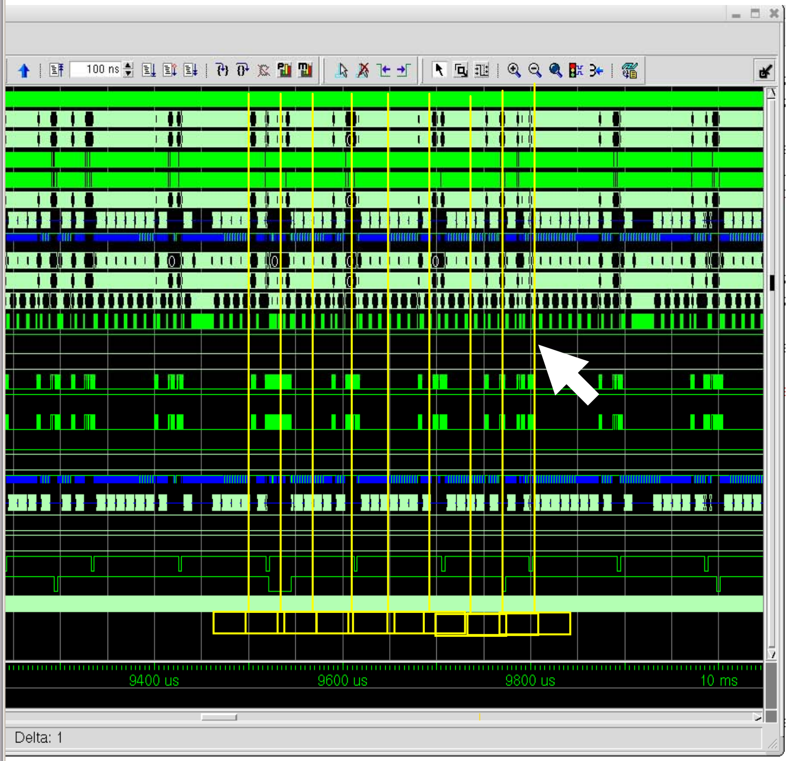


- Questa Codelink provides the critical features to support processor-driven, including multi-core verification

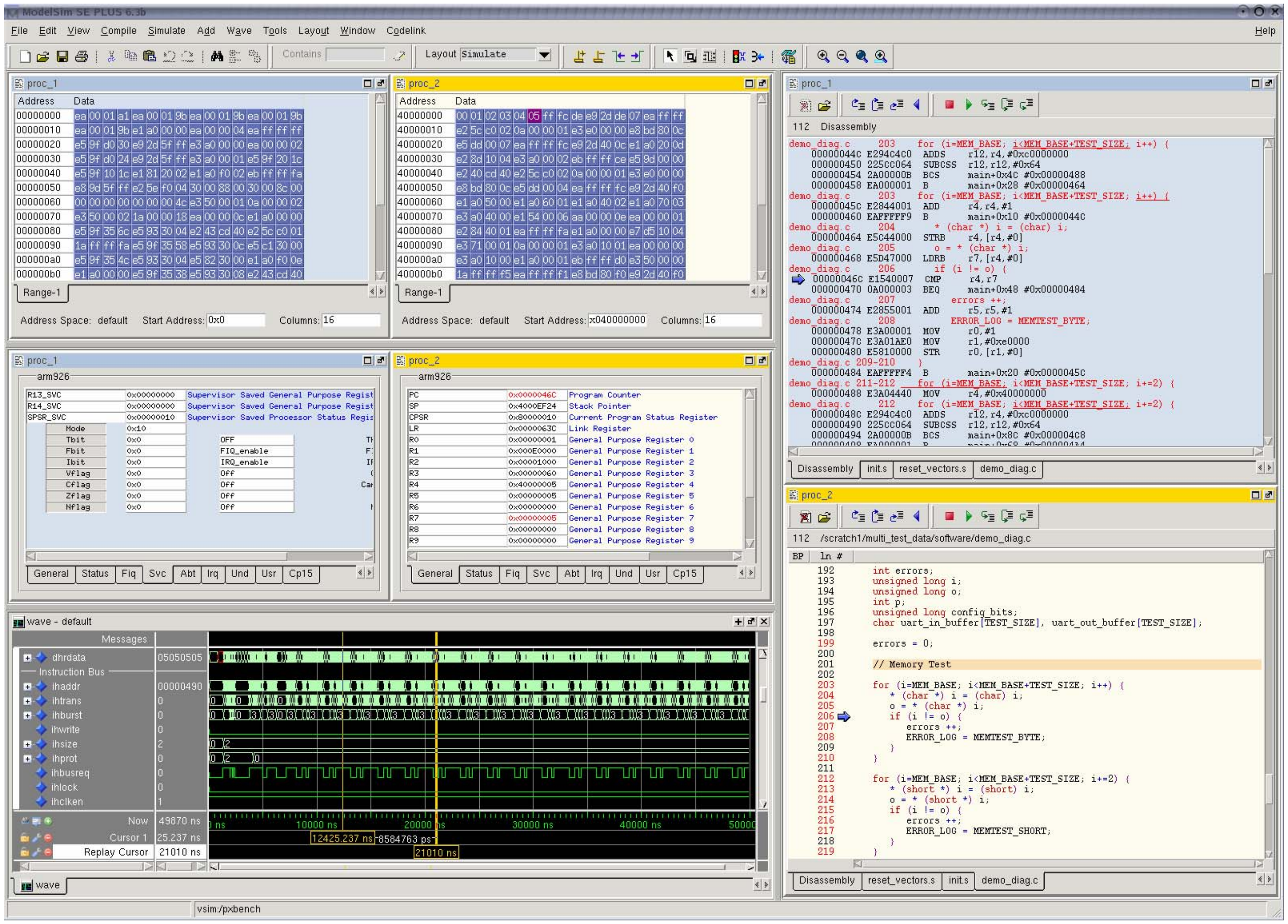
Hardware/Software Correlation



```
main()
{

        int k, j;
        int result;
        char sbuf[16];

    unsigned long *ptr,i;

      *(unsigned long *) FIQ_COUNTER = 3000; /* FIQ every 3000 clocks */
      *(unsigned long *) IRQ_COUNTER = 7500; /* IRQ every 7500 clocks */

    /* Write unsigned into LAWBAR0 & LAWAR0 */
    /* Write Page table Entry For 0x00010000 , 0x11000, 0x12000 */
    init_pagetable(0x30004000);
    write_pagetable_entry(0x0,0x80000000,0xFF600008,0xFF60003f);
    write_pagetable_entry(0x0,0x80000000,0xFF605008,0xFF60503f);

    ptr = (unsigned long *) 0x30300000;
    *(ptr) =  0x000ff600;

    i = *(ptr);

        ptr = (unsigned long *) 0x30300c08;
    *(ptr) =  0x00010000;
    ptr = (unsigned long *) 0x30300c10;
    *(ptr) = 0x80400013; //

    /* BR1, OR1 registers */

    ptr = (unsigned long *) 0x30305008;
    *(ptr)  = 0x10001801;
    ptr = (unsigned long *) 0x3030500c;

    *(ptr)  = 0x300000c14;

    write_pagetable_entry(0x0,0x80000000,0x10000008,0x1000003f);
    write_pagetable_entry(0x0,0x80000000,0x10001008,0x1000103f);
    write_pagetable_entry(0x0,0x80000000,0x10002008,0x1000203f);
    write_pagetable_entry(0x0,0x80000000,0x10003008,0x1000303f);
    write_pagetable_entry(0x0,0x80000000,0x10004008,0x1000403f);
    write_pagetable_entry(0x0,0x80000000,0x10005008,0x1000503f);
    write_pagetable_entry(0x0,0x80000000,0x10006008,0x1000603f);
    write_pagetable_entry(0x0,0x80000000,0x10007010,0x1000703f);
    [illegible]
    write_pagetable_entry(0x0,0x80000000,0x10009008,0x1000703f);
    write_pagetable_entry(0x0,0x80000000,0x1000A008,0x1000803f);


    for (j=0; j<32000; j++) {

            pprintf(" Intializing CRC Table     \n");
            init_crc32();

            pprintf(" Table Generation Completed\n");


            for (k=0; k<32; k++) {
                pprintf("Processing packet # ");
                itostr(k, sbuf, 16);
                pprintf(sbuf);
```



Multi-Core Processor Debug Environment


ModelSim SE PLUS 6.3b
File Edit View Compile Simulate Add Wave Tools Layout Window Codelink Help
Contains
Layout Simulate

proc_1

```
Address   Data
00000000  ea 00 01 a1 ea 00 01 9b ea 00 01 9b ea 00 01 9b
00000010  ea 00 01 9b e1 a0 00 00 ea 00 00 04 ea ff ff ff
00000020  e5 9f d0 30 e9 2d 5f ff e3 a0 00 00 ea 00 00 02
00000030  e5 9f d0 24 e9 2d 5f ff e3 a0 00 01 e5 9f 20 1c
00000040  e5 9f 10 1c e1 81 20 02 e1 a0 f0 02 eb ff ff fa
00000050  e8 9d 5f ff e2 5e f0 04 30 00 88 00 30 00 8c 00
00000060  00 00 00 00 00 00 00 4c e3 50 00 01 0a 00 00 02
00000070  e3 50 00 02 1a 00 00 18 ea 00 00 0c e1 a0 00 00
00000080  e5 9f 35 6c e5 93 30 04 e2 43 cd 40 e2 5c c0 01
00000090  1a ff ff fa e5 9f 35 58 e5 93 30 0c e5 c1 30 00
000000a0  e5 9f 35 4c e5 93 30 04 e5 82 30 00 e1 a0 f0 0e
000000b0  e1 a0 00 00 e5 9f 35 38 e5 93 30 08 e2 43 cd 40
```

Range-1

Address Space: default Start Address: 0x0 Columns: 16

proc_2

```
Address   Data
40000000  00 01 02 03 04 05 ff fc de e9 2d de 07 ea ff ff
40000010  e2 5c c0 02 0a 00 00 01 e3 e0 00 00 e8 bd 80 0c
40000020  e5 dd 00 07 ea ff ff fc e9 2d 40 0c e1 a0 20 0d
40000030  e2 8d 10 04 e3 a0 00 02 eb ff ff ce e5 9d 00 00
40000040  e2 40 cd 40 e2 5c c0 02 0a 00 00 01 e3 e0 00 00
40000050  e8 bd 80 0c e5 dd 00 04 ea ff ff fc e9 2d 40 f0
40000060  e1 a0 50 00 e1 a0 60 01 e1 a0 40 02 e1 a0 70 03
40000070  e3 a0 40 00 e1 54 00 06 aa 00 00 0e ea 00 00 01
40000080  e2 84 40 01 ea ff ff fa e1 a0 00 00 e7 d5 10 04
40000090  e3 71 00 01 0a 00 00 01 e3 a0 10 01 ea 00 00 00
400000a0  e3 a0 10 00 e1 a0 00 01 eb ff ff d0 e3 50 00 00
400000b0  1a ff ff f5 ea ff ff f1 e8 bd 80 f0 e9 2d 40 f0
```

Range-1

Address Space: default Start Address: x040000000 Columns: 16

proc_1

arm926

```
R13_SVC     0x00000000  Supervisor Saved General Purpose Regist
R14_SVC     0x00000000  Supervisor Saved General Purpose Regist
SPSR_SVC    0x00000010  Supervisor Saved Processor Status Regis
    Mode    0x10
    Tbit    0x0         OFF
    Fbit    0x0         FIQ_enable
    Ibit    0x0         IRQ_enable
    Vflag   0x0         Off
    Cflag   0x0         Off
    Zflag   0x0         Off
    Nflag   0x0         Off
```

General | Status | Fiq | Svc | Abt | Irq | Und | Usr | Cp15

proc_2

arm926

```
PC     0x0000046C  Program Counter
SP     0x4000EF24  Stack Pointer
CPSR   0x80000010  Current Program Status Register
LR     0x0000063C  Link Register
R0     0x00000001  General Purpose Register 0
R1     0x000E0000  General Purpose Register 1
R2     0x00001000  General Purpose Register 2
R3     0x00000060  General Purpose Register 3
R4     0x40000005  General Purpose Register 4
R5     0x00000005  General Purpose Register 5
R6     0x00000000  General Purpose Register 6
R7     0x00000005  General Purpose Register 7
R8     0x00000000  General Purpose Register 8
R9     0x00000000  General Purpose Register 9
```

General | Status | Fiq | Svc | Abt | Irq | Und | Usr | Cp15

proc_1

112 Disassembly

```
demo_diag.c      203    for (i=MEM_BASE; i<MEM_BASE+TEST_SIZE; i++) {
    00000044C E294C4C0  ADDS    r12, r4, #0xc0000000
    000000450 225CC064  SUBCSS  r12, r12, #0x64
    000000454 2A00000B  BCS     main+0x4C #0x00000488
    000000458 EA000001  B       main+0x28 #0x00000464
demo_diag.c      203    for (i=MEM_BASE; i<MEM_BASE+TEST_SIZE; i++) {
    00000045C E2844001  ADD     r4, r4, #1
    000000460 EAFFFFF9  B       main+0x10 #0x0000044C
demo_diag.c      204       * (char *) i = (char) i;
    000000464 E5C44000  STRB    r4, [r4, #0]
demo_diag.c      205       o = * (char *) i;
    000000468 E5D47000  LDRB    r7, [r4, #0]
demo_diag.c      206       if (i != o) {
    00000046C E1540007  CMP     r4, r7
    000000470 0A000003  BEQ     main+0x48 #0x00000484
demo_diag.c      207          errors ++;
    000000474 E2855001  ADD     r5, r5, #1
demo_diag.c      208          ERROR_LOG = MEMTEST_BYTE;
    000000478 E3A00001  MOV     r0, #1
    00000047C E3A01AE0  MOV     r1, #0xe0000
    000000480 E5810000  STR     r0, [r1, #0]
demo_diag.c 209-210       }
    000000484 EAFFFFF4  B       main+0x20 #0x0000045C
demo_diag.c 211-212    for (i=MEM_BASE; i<MEM_BASE+TEST_SIZE; i+=2) {
    000000488 E3A04440  MOV     r4, #0x40000000
demo_diag.c      212    for (i=MEM_BASE; i<MEM_BASE+TEST_SIZE; i+=2) {
    00000048C E294C4C0  ADDS    r12, r4, #0xc0000000
    000000490 225CC064  SUBCSS  r12, r12, #0x64
    000000494 2A00000B  BCS     main+0x8C #0x000004C8
```

Disassembly | init.s | reset_vectors.s | demo_diag.c

proc_2

112 /scratch1/multi_test_data/software/demo_diag.c

```
BP  ln #
    192      int errors;
    193      unsigned long i;
    194      unsigned long o;
    195      int p;
    196      unsigned long config_bits;
    197      char uart_in_buffer[TEST_SIZE], uart_out_buffer[TEST_SIZE];
    198
    199      errors = 0;
    200
    201      // Memory Test
    202
    203      for (i=MEM_BASE; i<MEM_BASE+TEST_SIZE; i++) {
    204         * (char *) i = (char) i;
    205         o = * (char *) i;
    206         if (i != o) {
    207            errors ++;
    208            ERROR_LOG = MEMTEST_BYTE;
    209         }
    210      }
    211
    212      for (i=MEM_BASE; i<MEM_BASE+TEST_SIZE; i+=2) {
    213         * (short *) i = (short) i;
    214         o = * (short *) i;
    215         if (i != o) {
    216            errors ++;
    217            ERROR_LOG = MEMTEST_SHORT;
    218         }
    219      }
```

Disassembly | reset_vectors.s | init.s | demo_diag.c

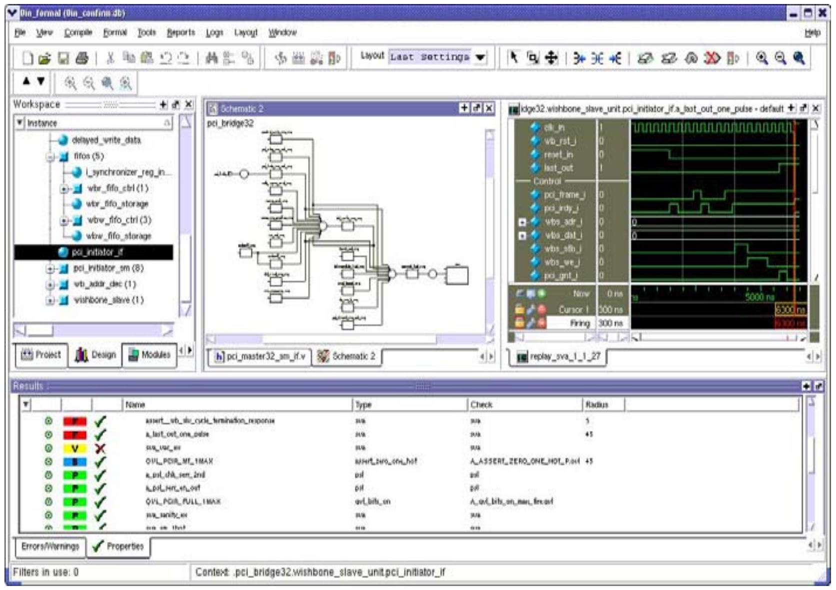
wave - default

```
Messages
dhrdata            05050505
Instruction Bus
ihaddr             00000490
ihtrans            0
ihburst            0
ihwrite            0
ihsize             2
ihprot             0
ihbusreq           0
ihlock             0
ihclken            1
Now                49870 ns
Cursor 1           25.237 ns
Replay Cursor      21010 ns
```

vsim:/pxbench

Mentor's 0-In® Formal Verification Solution Delivers …

- Highest capacity and performance
- Extensive Design Style Support
- Smart integration of formal verification with simulation



- The largest library of assertion IP in the industry
- Intuitive graphical analysis and debug
- Questa Coverage database enabled

The 'proven' formal verification solution

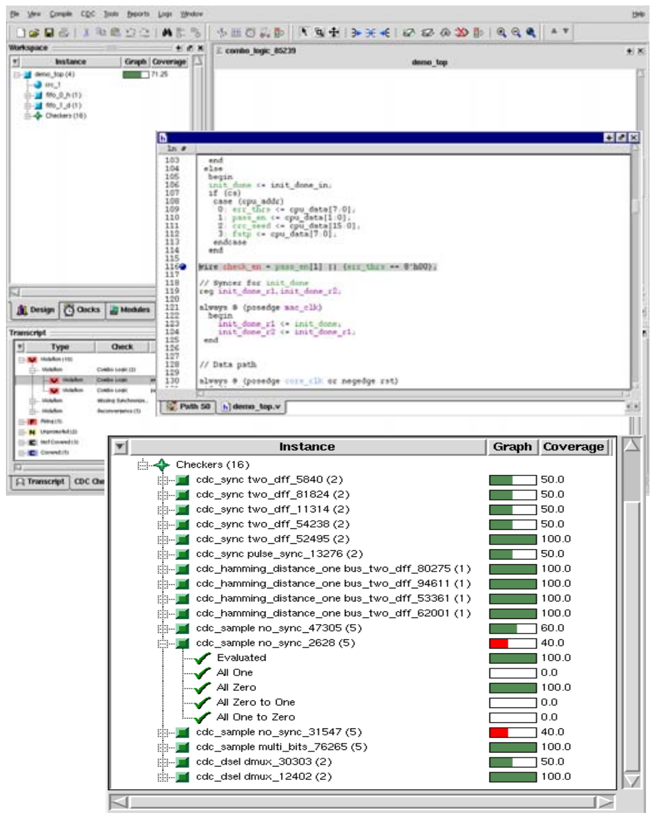
0-In® CDC Verificaton

✓ **Structural CDC verification**
- Automatically identifies all clocks and clock-domain crossings (CDCs)

✓ **Verification of CDC protocols**
- Automatically proves CDC Protocols
- Simulate CDC protocol assertions

✓ **Silicon-accurate RTL simulation**
- Mimics the metastability effects in synchronizers

✓ **Accurate Coverage metrics**
- Provides a measure of completeness for the testbench as related to metastability issues



```
   end
  else
   begin
   init_done <= init_done_in;
   if (cs)
    case (cpu_addr)
     0: err_thrs <= cpu_data[7:0];
     1: pass_en <= cpu_data[1:0];
     2: crc_seed <= cpu_data[15:0];
     3: fstp <= cpu_data[7:0];
    endcase
   end

wire check_en = pass_en[1] || (err_thrs == 8'h00);

// Syncer for init_done
reg init_done_r1,init_done_r2;

always @ (posedge mac_clk)
  begin
    init_done_r1 <= init_done;
    init_done_r2 <= init_done_r1;
 end


// Data path

always @ (posedge core_clk or negedge rst)
```

Instance	Graph	Coverage
Checkers (16)		
cdc_sync two_dff_5840 (2)	🟩	50.0
cdc_sync two_dff_81824 (2)	🟩	50.0
cdc_sync two_dff_11314 (2)	🟩	50.0
cdc_sync two_dff_54238 (2)	🟩	50.0
cdc_sync two_dff_52495 (2)	🟩	100.0
cdc_sync pulse_sync_13276 (2)	🟩	50.0
cdc_hamming_distance_one bus_two_dff_80275 (1)	🟩	100.0
cdc_hamming_distance_one bus_two_dff_94611 (1)	🟩	100.0
cdc_hamming_distance_one bus_two_dff_53361 (1)	🟩	100.0
cdc_hamming_distance_one bus_two_dff_62001 (1)	🟩	100.0
cdc_sample no_sync_47305 (5)	🟩	60.0
cdc_sample no_sync_2628 (5)	🟥	40.0
✓ Evaluated	🟩	100.0
✓ All One		0.0
✓ All Zero	🟩	100.0
✓ All Zero to One		0.0
✓ All One to Zero		0.0
cdc_sample no_sync_31547 (5)	🟥	40.0
cdc_sample multi_bits_76265 (5)	🟩	100.0
cdc_dsel dmux_30303 (2)	🟩	50.0
cdc_dsel dmux_12402 (2)	🟩	100.0

0-In® CDC – The Benchmark in CDC verification

Mentor Graphics®

www.mentor.com

Expanding Debug Feature Set

Improves Productivity

- **Advanced waveform viewer**
- **Schematic-based Dataflow window**
- **Integrated assertion browser**
- **Functional and code coverage including FSM**
- **Performance analyzer**
- **Memory viewer**
- **Waveform editor**
- **Signal Spy**
- **JobSpy**
- **Message viewer (6.2)**
- **Source annotation (6.2)**
- **TLM Viewing (6.2)**
- **Post Simulation Debug (6.3)**
- **Textual Dataflow (6.3)**
- **TLM Analysis (6.3)**
- **Verification Management (6.3)**
- **UCDB Browser (6.3)**
- **SV call stack window (6.3)**
- [illegible]

Rich native debug environment for all languages and abstraction levels