

Intro. to Synopsys Verdi Debug System

Presenter: Jui-Huang Tsai

Advisor: Prof. Chen-Yi Lee

2023.10.04



Outline

- Startup Verdi with Simulation Tool
- Verdi's Feature
 - nTrace
 - nWave
 - nMemory
 - nState



Startup

- `./01_run_vcs_rtl` for dumping `.fsdb`
- `./04_verdi` for starting Verdi

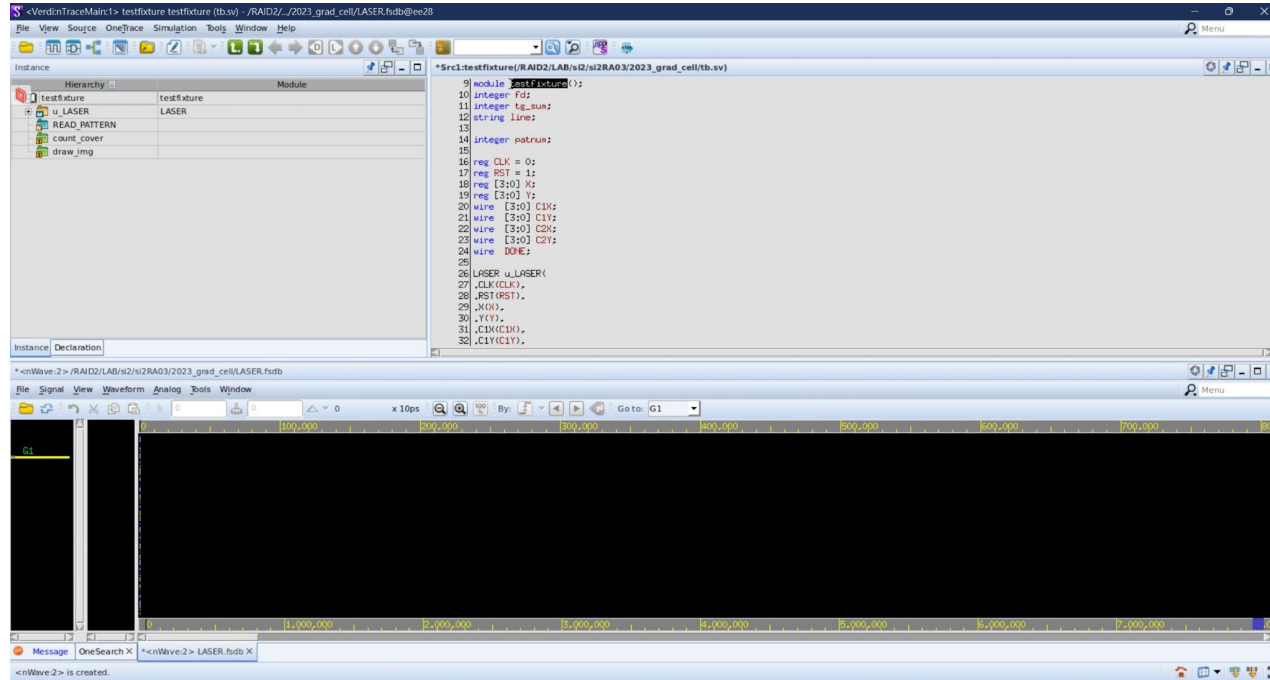


Verdi's Features

- RTL debug tool developed by Synopsys, it supports
 - Verilog
 - SystemVerilog
 - VHDL
- Features
 - nTrace
 - nWave
 - nMemory
 - nState



Startup



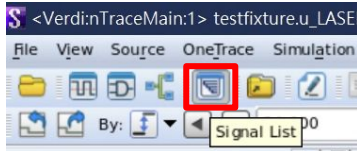
nTrace: Show Signal's Value

- Show the value of signal (Command: “x”)

```
50 always@(posedge CLK) begin
51     if(RST)
52         c_state <= IDLE; /* initial state */
53     else
54         c_state <= n_state;
55 end
56
57 always@(*) begin
58     case(c_state)
59         IDLE: begin
60             if(!RST && !DONE)
```



nTrace: Signal List



Filter String

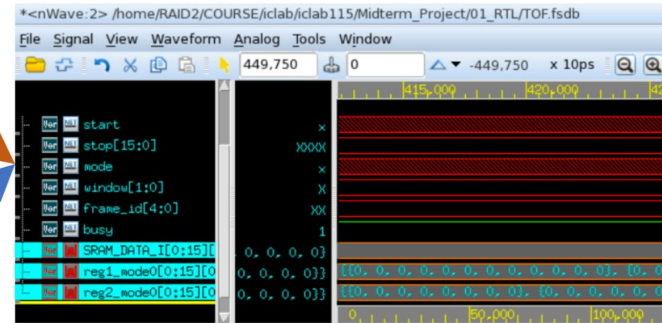
Signal	Value	Type
LASER		
CLK	0->1	wire(Port In)
RST	0	wire(Port In)
X[3:0]	6	wire(Port In)
Y[3:0]	f	wire(Port In)
C1X[3:0]	0	reg(Port Out)
C1Y[3:0]	0	reg(Port Out)
C2X[3:0]	0	reg(Port Out)
C2Y[3:0]	0	reg(Port Out)
DONE	0	reg(Port Out)
i	XXXXXXXX	integer
j	XXXXXXXX	integer
IDLE	0	parameter
DATAIN	1	parameter
ITERATE	2	parameter
DATAOUT	3	parameter
c_state[1:0]	1->2	reg
n_state[1:0]	2	reg
...



nTrace – Add Value Into Waveform

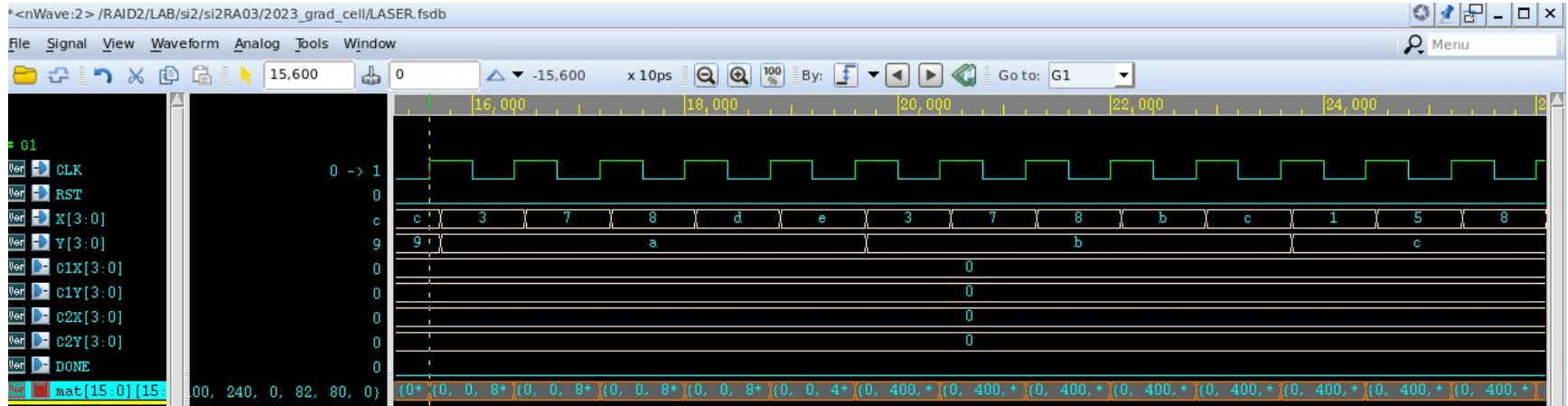
```
*Src1:TESTBED.U_TOF(/home/RAID2/COURSE/icla  
294 reg [1:0] window_r;  
295 reg mode_r;  
296 reg [4:0] frame_id_r;  
297  
298 reg [7:0] reg1_mode0 [0:15][0:11];  
299 reg [7:0] reg2_mode0 [0:15][0:7];  
300 reg [8:0] add_1 [0:15] [0:31];
```

Signal_List		
Filter String		
Signal	Value	Type
window_r[1:0]	3	reg
mode_r	1	reg
frame_id_r[4:0]	c	reg
reg1_mode0[0:15]	{ {6e, ac, 9...	reg
reg2_mode0[0:15]	{ {c9, 31, a...	reg
add_1[0:15]	{ {fa, e8, 9d...	reg
add_2[0:15]	{ {1e2, cf}, ...	reg
add_4[0:15]	{2b1, 38b, ...	reg
cur_max[0:15]	{537, 50b, ...	reg
cur_max_dist[0:15]	{83, 8, c, d, ...	reg
f_counter_256_3f	0	reg
mode0_AXI_las...	0	reg
mode0_create...	1	wire
state_after	1	wire

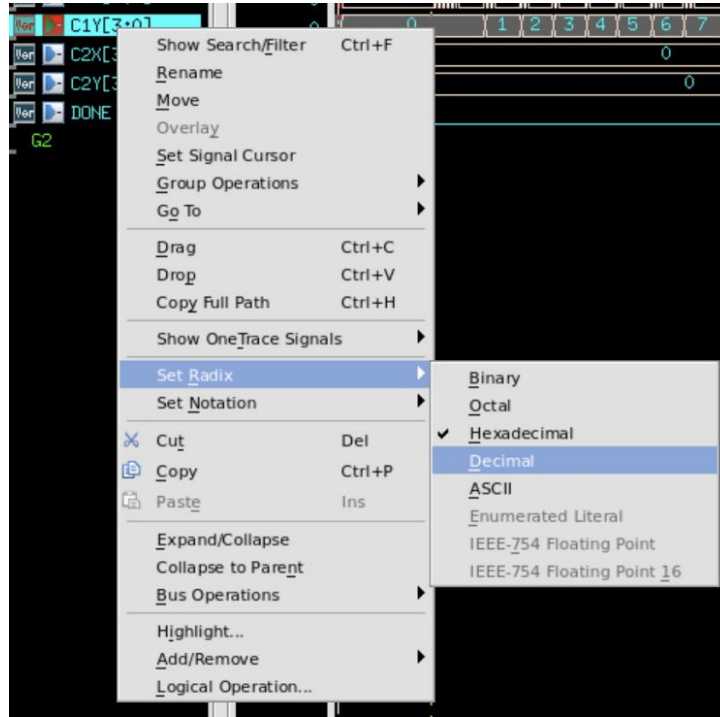


Ctrl + W

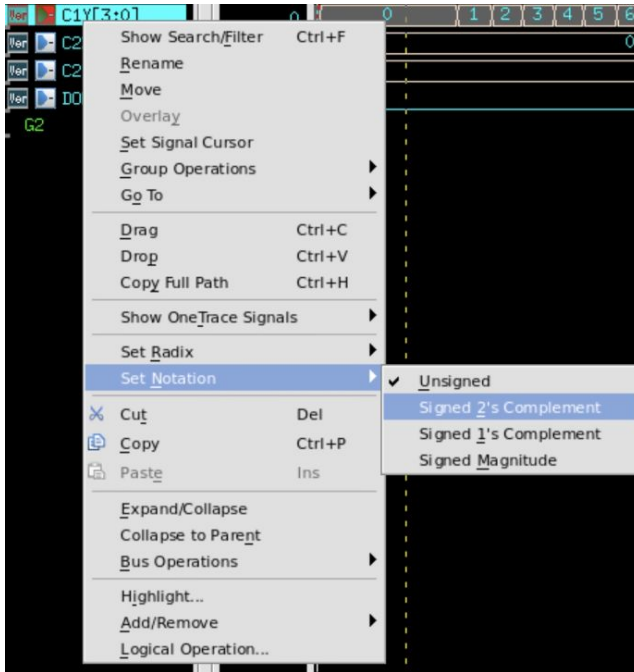
nWave



nWave - Change Radix of Signal



nWave - 2's Complement

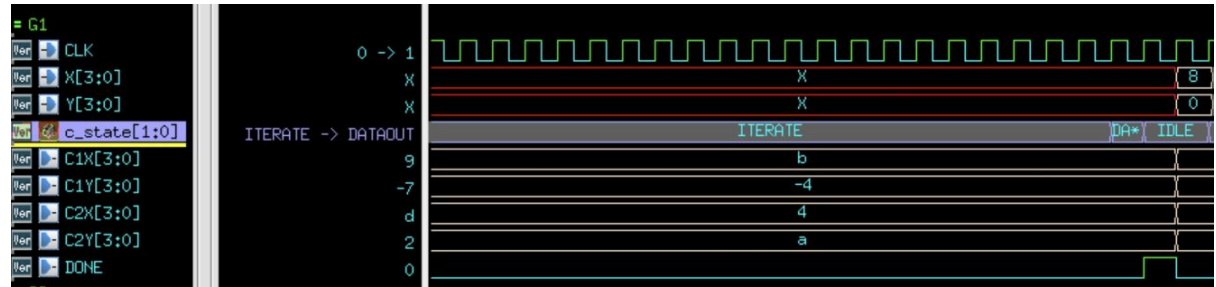
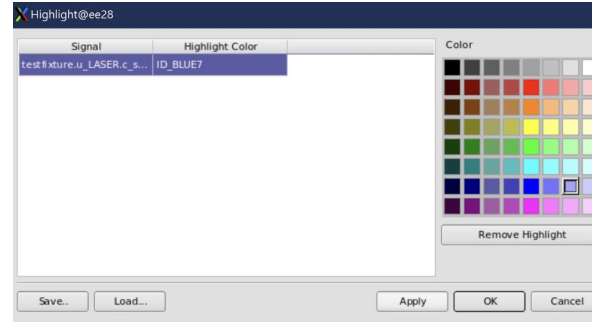
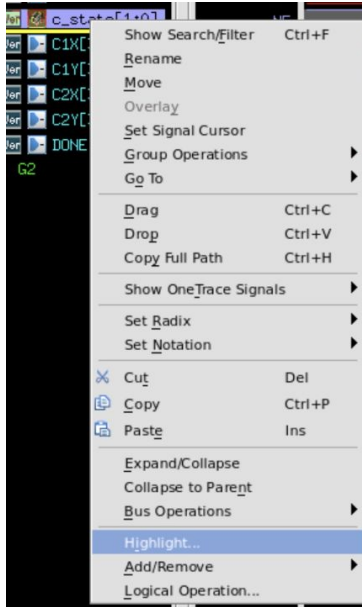


Eight-bit integers

Bits	Unsigned value	Signed value (Two's complement)
0000 0000	0	0
0000 0001	1	1
0000 0010	2	2
0111 1110	126	126
0111 1111	127	127
1000 0000	128	-128
1000 0001	129	-127
1000 0010	130	-126
1111 1110	254	-2
1111 1111	255	-1

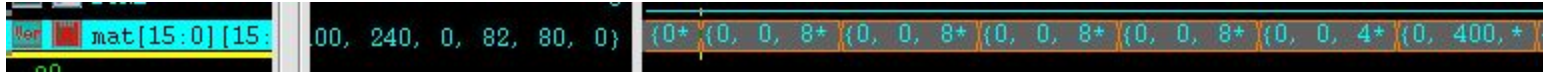
Source:
[Two's complement - Wikipedia](https://en.wikipedia.org/wiki/Two%27s_complement)

nWave - Highlight Signal



Vector

- How do we debug vector signal easier?



nMemory

<nMemory:3> /RAID2/.../2023_grad_cell/LASER.fsdb : testfixture.u_LASER.mat[15:0][15:0]

File Search Time Options Tools Window

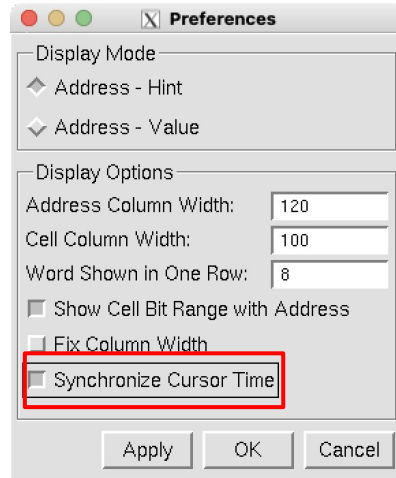
Time: 15,600 x 10ps By: Display Range Written Time:

Display Range: [f : 0] Cell: [f : 0]

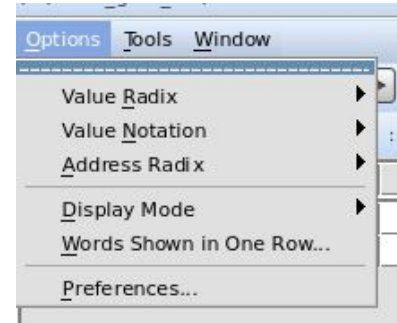
Addr/Hint	[f] [f:0]	[e] [f:0]	[d] [f:0]	[c] [f:0]	[b] [f:0]	[a] [f:0]	[9] [f:0]	[8] [f:0]
[f] [f:0]	0000	0000	0080	0200	0001	0202	0080	0082
[7] [f:0]	0180	0200	0100	0240	0000	0082	0080	0000



nMemory (Setting)

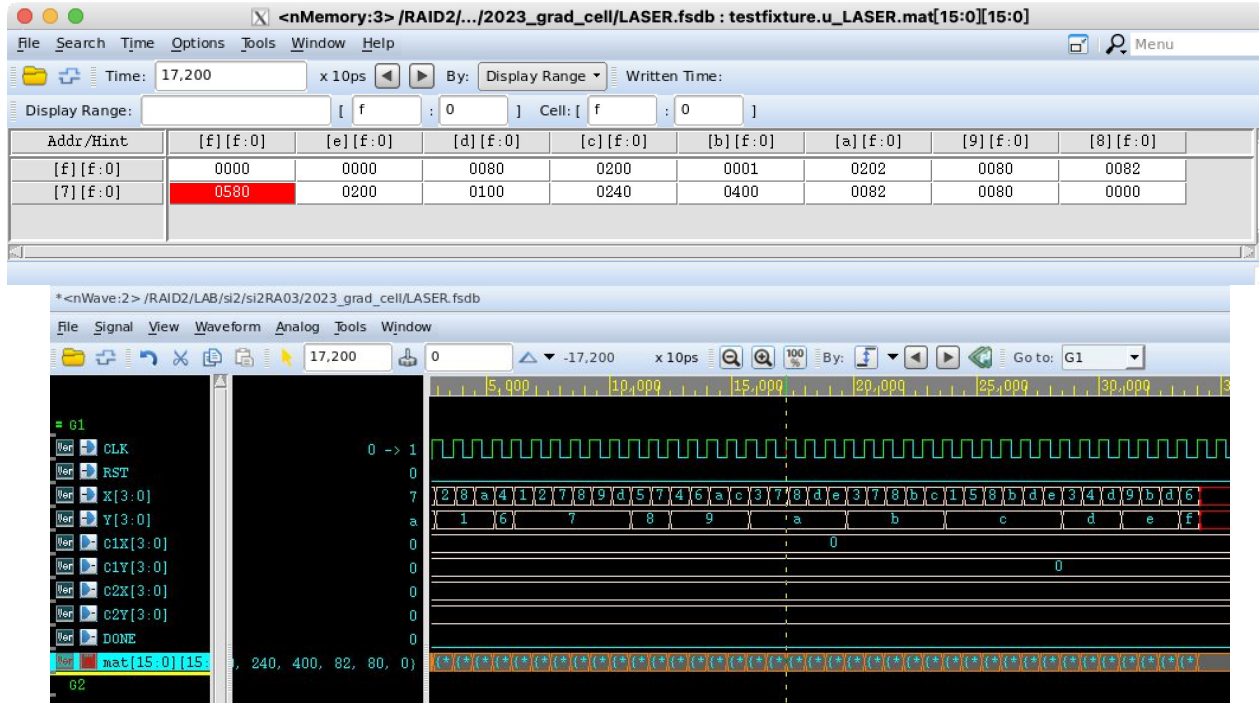


Synchronize with
waveform



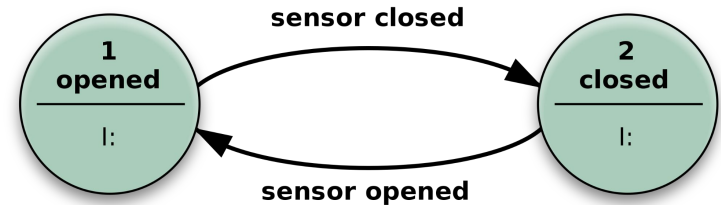
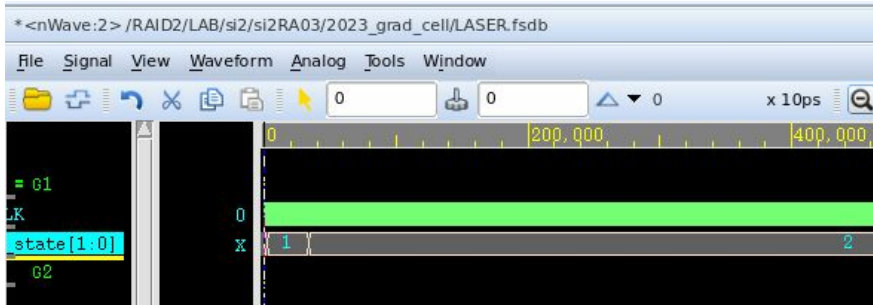
Set value's radix, notation
and address radix

nMemory (Sync with Waveform)



Finite State Machine

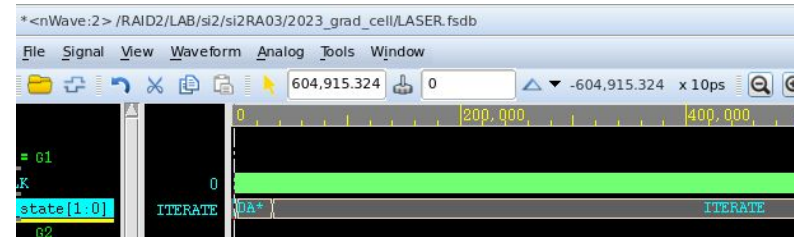
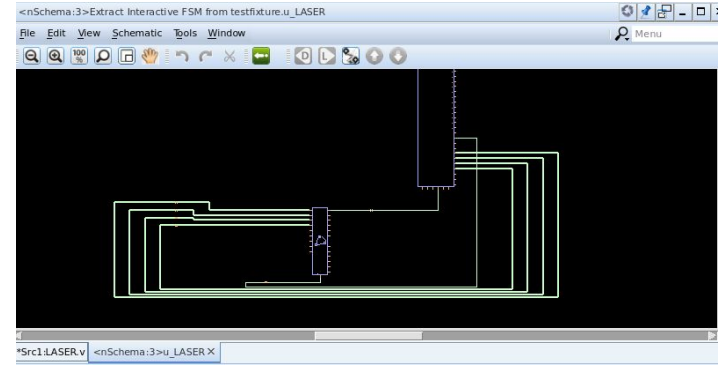
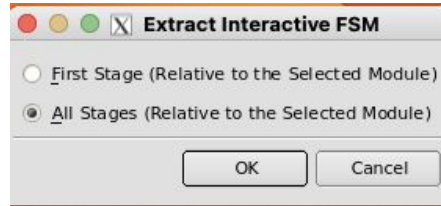
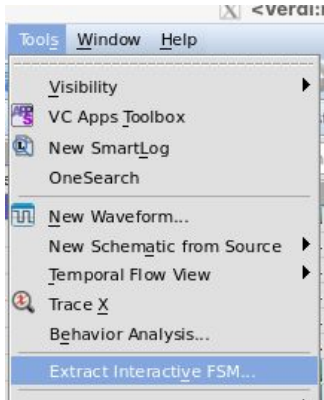
- How do we show the current state in “readable” format of the reg referred to FSM?



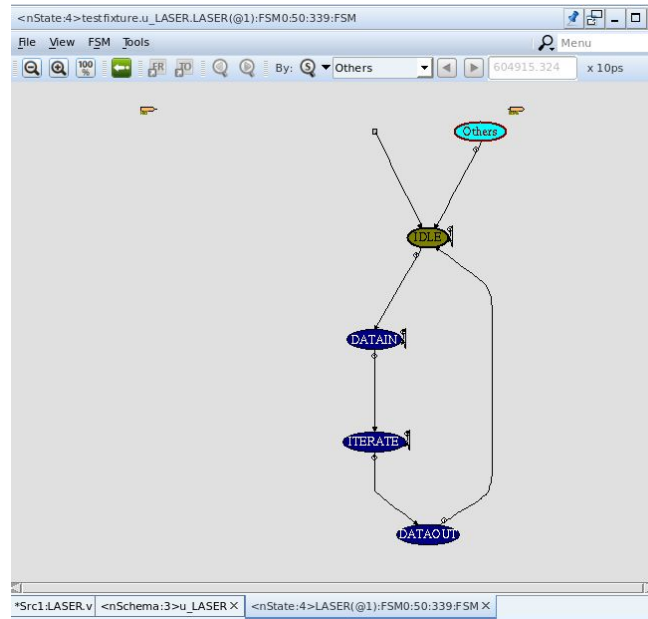
Source:

[Finite-state machine - Wikipedia](#)

nState (1/2)



nState (2/2)



Save Session



Conclusion

- nTrace showed the signal value and signal in list
- nWave displayed the waveform of signal
- nMemory eased the pain on observing value of 2d reg
- nState helped us dealing with FSM issues

