Verilog HDL

Default Session: mux.v - Kate — root@slax.example.net: /r — GTKWave - fifo.vcd

Default Session: mux.v - Kate

File Edit Document View Bookmarks Tools Sessions Settings
Window Help

Documents
Filesystem Browser

```
module MUX (TIME_DATA, ALARM_DATA, SHOW_A, DISPLAY);

input  [3:0] TIME_DATA, ALARM_DATA;
input        SHOW_A;
output [3:0] DISPLAY;
reg    [3:0] DISPLAY;

  // DISP_MUX procedure
  always @(TIME_DATA or ALARM_DATA or SHOW_A)
   begin : DISP_MUX
      if (SHOW_A == 1'b1)
        DISPLAY = ALARM_DATA;
      else
        DISPLAY = TIME_DATA;
   end

endmodule
```

Line: 1 Col: 1 INS NORM mux.v

```
                550  0 26
                570  0 27
                590  0 28
                610  0 29
                630  0 30
                650  0 31
                650   0 1 0
                750 255 0 0
                770 254 0 0
                790 253 0 0
                810 252 0 0
** VVP Stop(0) **
** Current simulation time is 82000 ticks.
> finish
** Continue **
root@slax:~/sample_projects/verilog_samples/fifo_sample # ls
fifo.v  fifo.vcd  sim  tb_fifo.v
root@slax:~/sample_projects/verilog_samples/fifo_sample #
```

Find in Files | Terminal

root@slax.example.net: /root/sample_projects/verilog_samples/fifo_sample - Shell - Konsole

```
root@slax:~/sample_projects/verilog_samples #
root@slax:~/sample_projects/verilog_samples # cd fifo_sample/
root@slax:~/sample_projects/verilog_samples/fifo_sample # ls
fifo.v  fifo.vcd  sim  tb_fifo.v
root@slax:~/sample_projects/verilog_samples/fifo_sample # gtkwave fi
fifo.v    fifo.vcd
root@slax:~/sample_projects/verilog_samples/fifo_sample # gtkwave fifo.vcd &
[1] 6108
root@slax:~/sample_projects/verilog_samples/fifo_sample #
GTKWave Analyzer v3.1.13 (w)1999-2008 BSI

[0] start time.
[810000] end time.
root@slax:~/sample_projects/verilog_samples/fifo_sample #
```

GTKWave - fifo.vcd

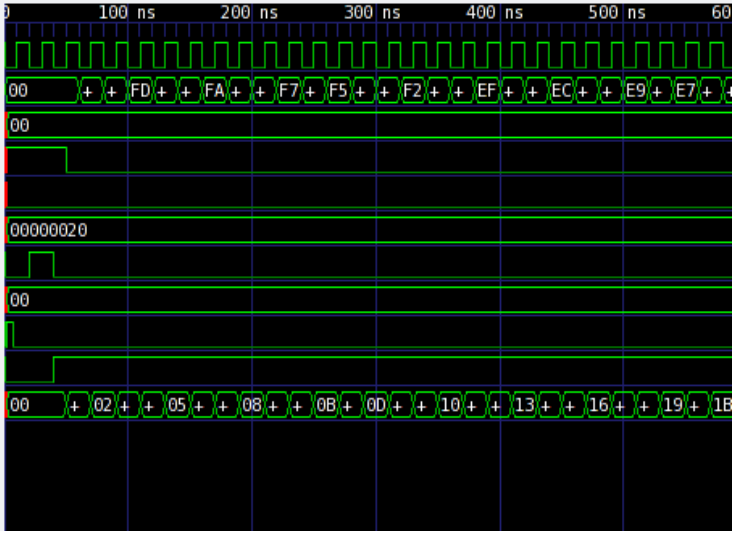
File Edit Search Time Markers View Help

From: 0 sec To: 810 ns Marker: -- | Cursor: 3 ns

SST
- fifo_tb
  - dut

Signals
clock
data_in[7:0]
data_out[7:0]
f_empty
f_full
i[31:0]
rd_en
rd_ptr[4:0]
reset
wr_en
wr_ptr[4:0]

Waves



Verilog Hardware Description Language

Textual modelling language (logisim is visual)

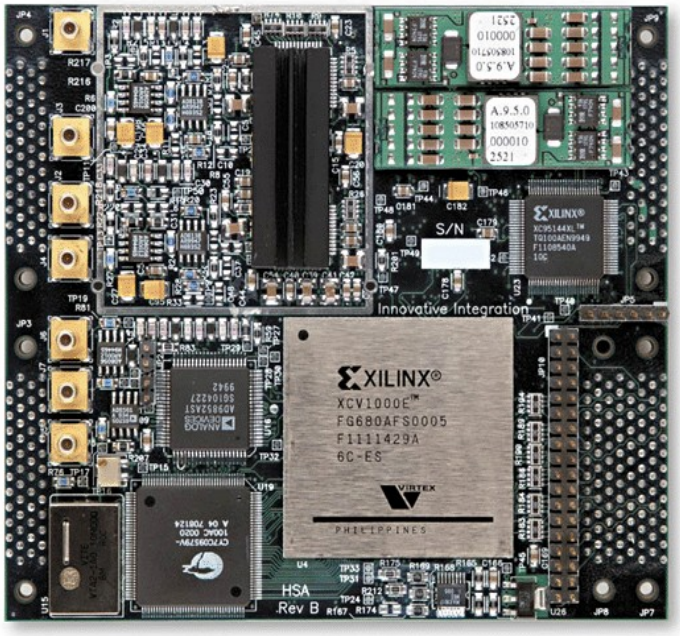
System Under Study **modelling** at **different levels of abstraction**:

- Behavioural
- Register-Transfer Level (RTL)
- Structural/Gate Level (logisim's level)

Simulation (SUS + **testbench** using behavioural description)

Synthesis (for FPGA for example):

- Logic synthesis (higher to lower level)
- Partitioning and
  mapping to primitive elements
- Element placement
- Wire routing



Verilog Hardware Description Language

Verilog vs. VHDL
 VHSIC (Very High Speed Integrated Circuit)
 Hardware Description Language

Hardware description IS NOT programming!