

*******DEFAULT*******

Initial steps to convert ppio to tlv

10/10/20

*******DEFAULT*******

```
"java -jar /home/devel/SandPiper_1.9-2018_02_11-beta_distro/target/sandpiper.jar --verbose --  
dhtml --stats --viz --svg --graphTrans -i top.m4 -o top.sv"
```

INFORM(0) (PROD_INFO):

—— SandPiper(TM) 1.9-2018/02/11-beta from Redwood EDA

—— Run as: "java -jar /home/devel/SandPiper_1.9-2018_02_11-beta_distro/target/sandpiper.jar
--verbose --dhtml --stats --viz --svg --graphTrans -i top.m4 -o top.sv

—— For help, including product info, run with -h.

INFORM(0) (LICENSE):

—— Licensed to "Edward Vidal Jr." as: Full Edition.

INFORM(0) (FILES):

—— From directory: /home/devel/sandpiper_test/test2b/TLVerilog

—— Reading "top.m4" to produce:

—— Translated HDL File: "top.sv"

—— Generated HDL File: "top_gen.sv"

—— HTML TLX File: "top.html"

—— Transaction Flow Graph (Dot) File: "top_trans.dot"

—— Navigation Environment: "top_graph.svg"

—— Simulation Visualization File: "top_viz.json"

—— Statistics Directory: "top_stats"

Exiting due to DEFERRED_ERROR(s).

Note: Can not use -dhtml with -svg

```
"java -jar /home/devel/SandPiper_1.9-2018_02_11-beta_distro/target/sandpiper.jar --verbose  
--stats --viz --svg --graphTrans -i top.m4 -o top.sv"
```

INFORM(0) (PROD_INFO):

SandPiper(TM) 1.9-2018/02/11-beta from Redwood EDA

Run as: "java -jar

/home/devel/SandPiper_1.9-2018_02_11-beta_distro/target/sandpiper.jar --verbose --stats --viz
--svg --graphTrans -i top.m4 -o top.sv

For help, including product info, run with -h.

INFORM(0) (LICENSE):

Licensed to "Edward Vidal Jr." as: Full Edition.

INFORM(0) (FILES):

From directory: /home/devel/sandpiper_test/test2b/TLVerilog

Reading "top.m4" to produce:

Translated HDL File: "top.sv"

Generated HDL File: "top_gen.sv"

Transaction Flow Graph (Dot) File: "top_trans.dot"

Navigation Environment: "top_graph.svg"
Simulation Visualization File: "top_viz.json"
Statistics Directory: "top_stats"

WARNING(1) (UNUSED-SIG): File 'top.tlv' Line 25 (char 4)

Preprocessed as 'top.m4':25(ch4):

+---vvvvvvv-----

> \$in_dir = *in_dir

+---^^^-----

Signal \$in_dir is assigned but never used.

To silence this message use "`BOGUS_USE(\$in_dir)".

INFORM(0) (FILE): File 'top.tlv'

Writing file "top.sv".

INFORM(0) (FILE):

Writing file "top_gen.sv".

INFORM(0) (FILE):

Writing file "top_trans.dot".

INFORM(0) (FILE):

Writing file "top_stats/tlx.css".

INFORM(0) (FILE):

Writing file "top_stats/tlx.css".

INFORM(0) (FILE):

Writing file "top_stats/tlx.css".

INFORM(0) (FILE):

Writing file "top_stats/tlx.css".

INFORM(0) (FILE):

Writing file "top_stats/index.html".

INFORM(0) (FILE):

Writing file "top_stats/tlx.css".

INFORM(0) (FILE):

Writing file "top_stats/stats.csv".

INFORM(0) (STATS):

SandPiper generated 16% of your HDL code.

This includes: 1 signal declarations, 0 flops/latches, and 0 conditional clock signals.

View "top_stats" for more details.

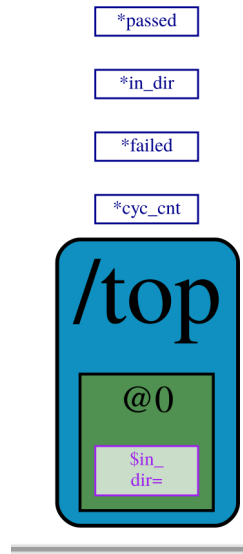
INFORM(0) (FILE):

Writing file "top_viz.json".

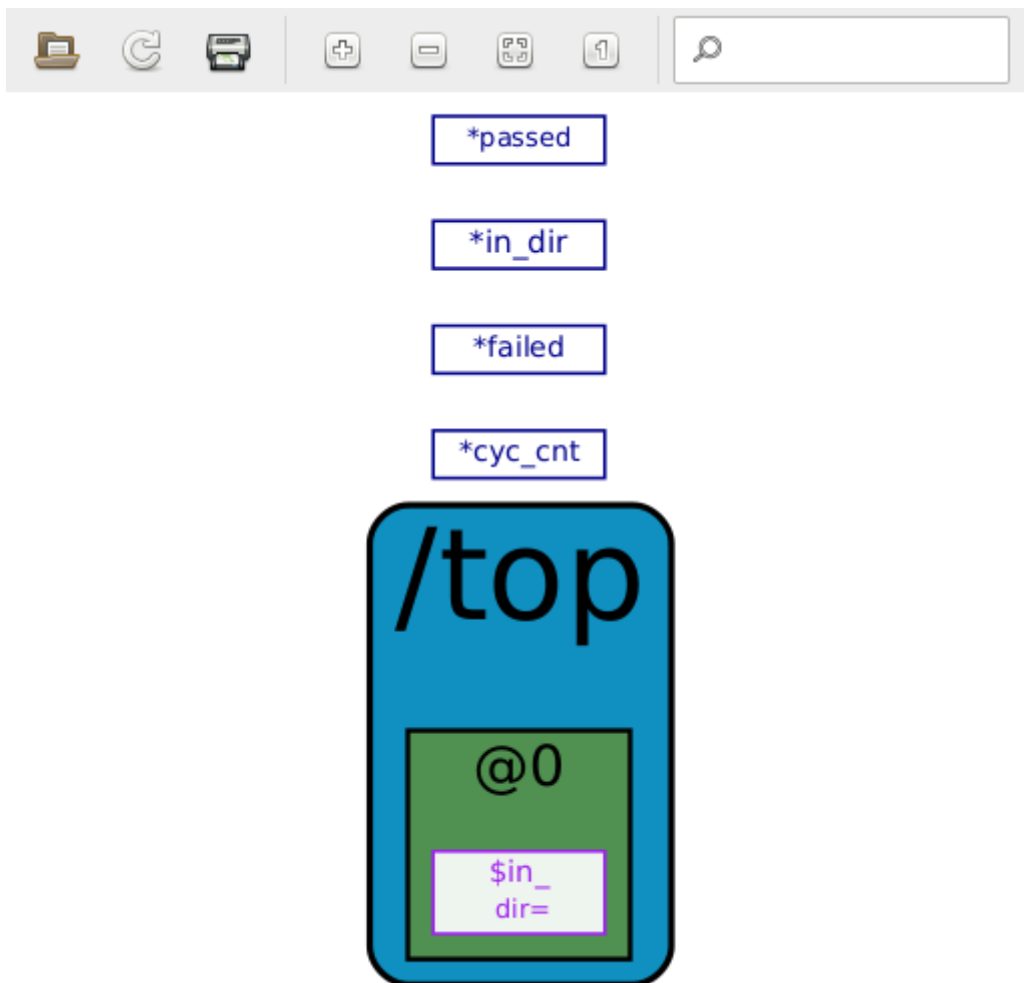
INFORM(0) (FILE): File 'top.tlv'

Writing file "top_graph.svg".

Used Inkscape to view top_graph.svg.

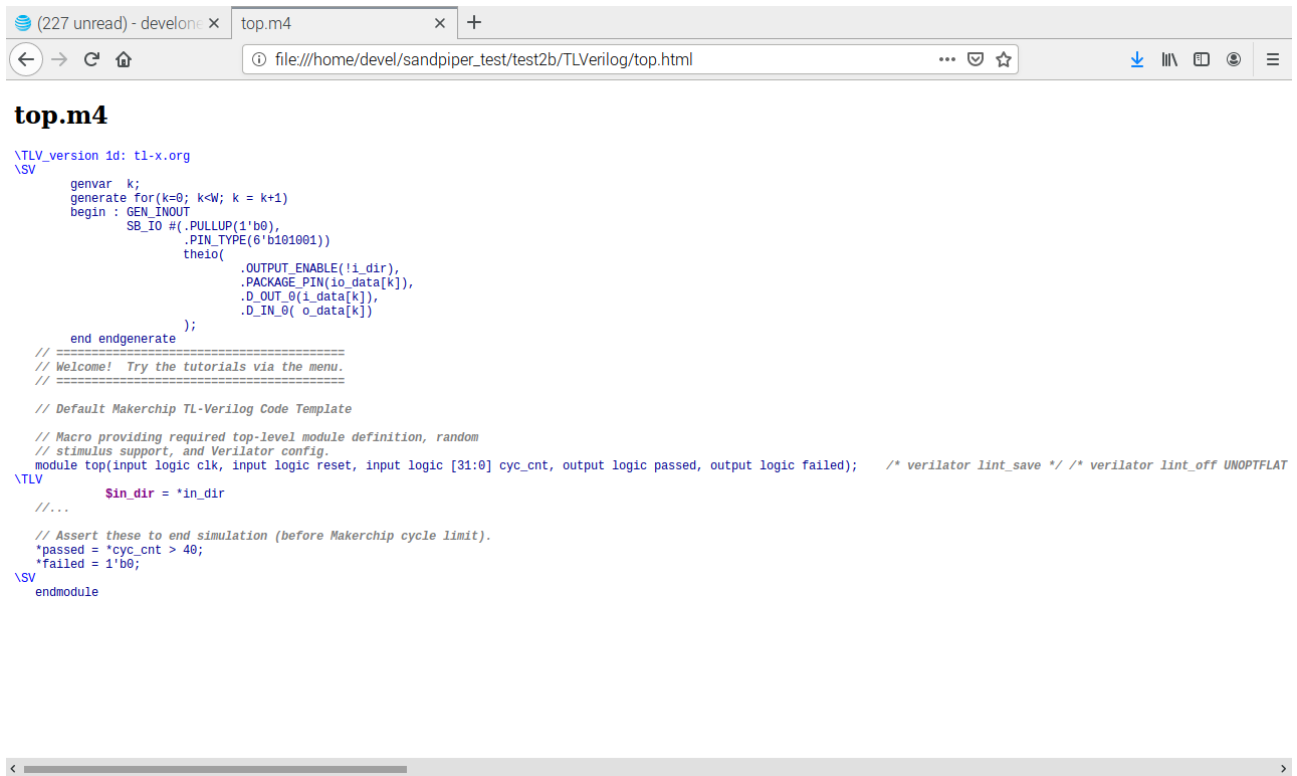


`xdot top_trans.dot`



removing `--svg` and adding `--dhtml` now provides `top.html`

`java -jar /home/devel/SandPiper_1.9-2018_02_11-beta_distro/target/sandpiper.jar --verbose --stats --viz --dhtml --graphTrans -i top.m4 -o top.sv`



```
\TLV_version id: tl-x.org
\SV
    genvar k;
    generate for(k=0; k<W; k = k+1)
    begin : GEN_INOUT
        SB_IO #(.PULLUP(1'b0),
                .PIN_TYPE(6'b101001))
            theio(
                .OUTPUT_ENABLE(!i_dir),
                .PACKAGE_PIN(io_data[k]),
                .D_OUT_0(i_data[k]),
                .D_IN_0( o_data[k])
            );
    end endgenerate
// =====
// Welcome! Try the tutorials via the menu.
// =====

// Default Makerchip TL-Verilog Code Template

// Macro providing required top-level module definition, random
// stimulus support, and Verilator config.
module top(input logic clk, input logic reset, input logic [31:0] cyc_cnt, output logic passed, output logic failed); /* verilator lint_save */ /* verilator lint_off UNOPTFLAT
\TLV
    $in_dir = *in_dir
    //...

    // Assert these to end simulation (before Makerchip cycle limit).
    *passed = *cyc_cnt > 40;
    *failed = 1'b0;
\SV
endmodule
```