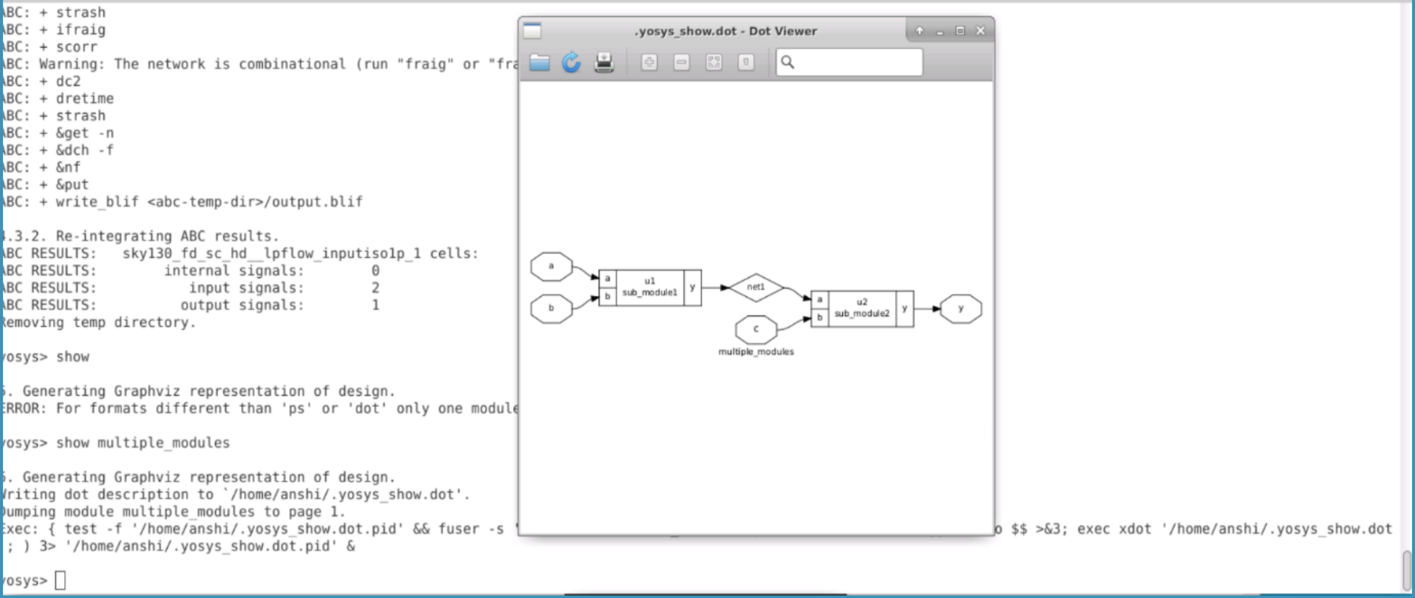
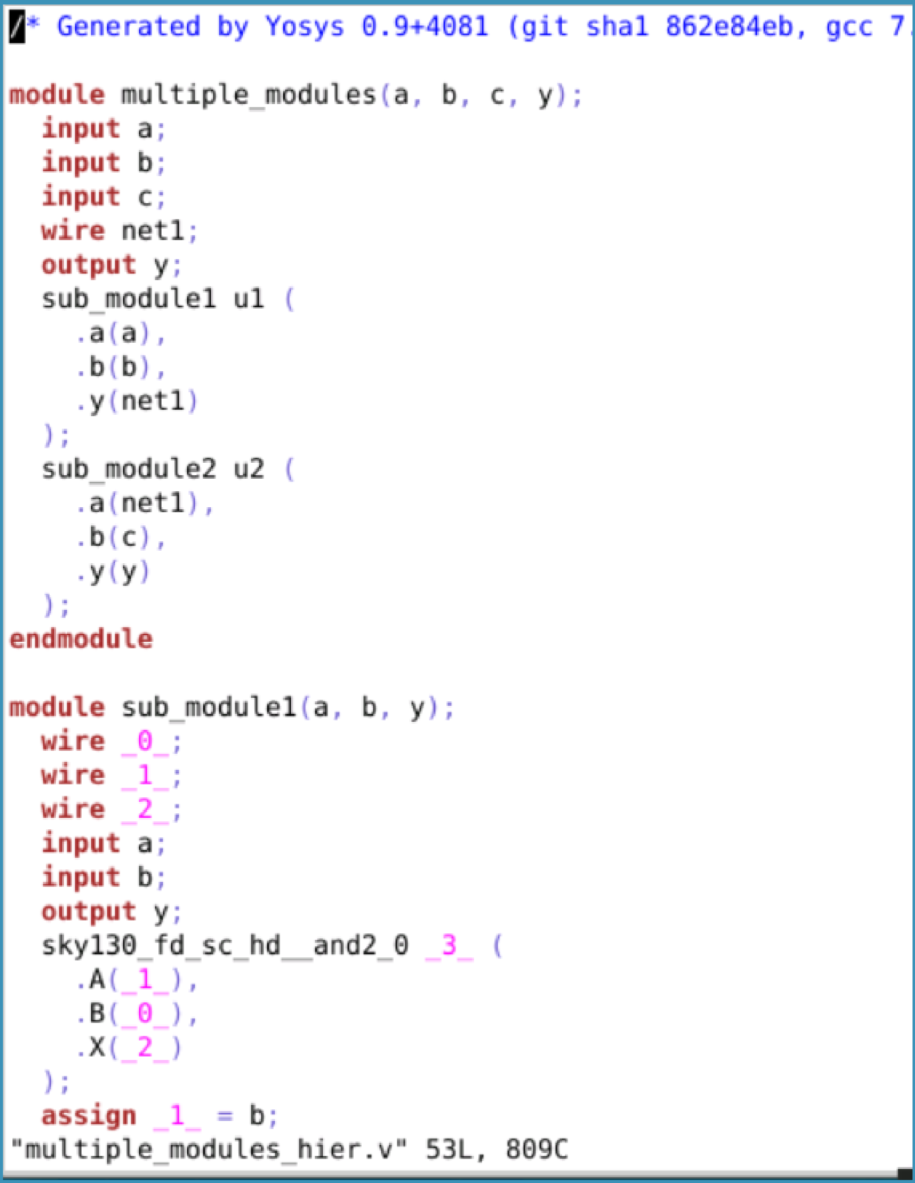
Day-2

Hierarchical Vs Flat Synthesis and Efficient Flip-Flop Coding Styles

Hierarchy Is Preserved



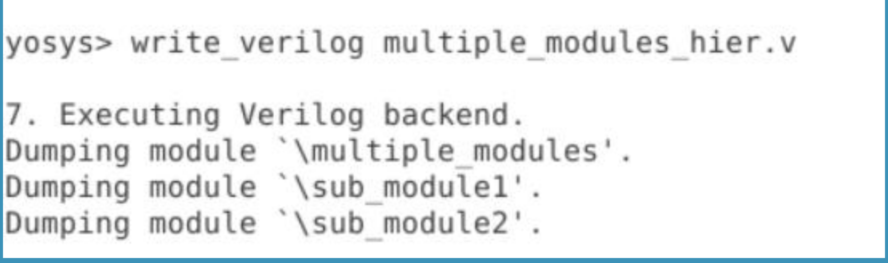
```
\BC: + strash
\BC: + ifraig
\BC: + scorr
\BC: Warning: The network is combinational (run "fraig" or "fra
\BC: + dc2
\BC: + dretime
\BC: + strash
\BC: + &get -n
\BC: + &dch -f
\BC: + &nf
\BC: + &put
\BC: + write_blif <abc-temp-dir>/output.blif

.3.2. Re-integrating ABC results.
\BC RESULTS:   sky130_fd_sc_hd__lpflow_inputiso1p_1 cells:
\BC RESULTS:        internal signals:        0
\BC RESULTS:           input signals:        2
\BC RESULTS:          output signals:        1
Removing temp directory.

/osys> show

. Generating Graphviz representation of design.
ERROR: For formats different than 'ps' or 'dot' only one module

/osys> show multiple_modules

. Generating Graphviz representation of design.
/riting dot description to `/home/anshi/.yosys_show.dot'.
Dumping module multiple_modules to page 1.
Exec: { test -f '/home/anshi/.yosys_show.dot.pid' && fuser -s '                                                  o $$ >&3; exec xdot '/home/anshi/.yosys_show.dot
 ; ) 3> '/home/anshi/.yosys_show.dot.pid' &

/osys> 
```



```
/* Generated by Yosys 0.9+4081 (git sha1 862e84eb, gcc 7.

module multiple_modules(a, b, c, y);
  input a;
  input b;
  input c;
  wire net1;
  output y;
  sub_module1 u1 (
    .a(a),
    .b(b),
    .y(net1)
  );
  sub_module2 u2 (
    .a(net1),
    .b(c),
    .y(y)
  );
endmodule

module sub_module1(a, b, y);
  wire _0_;
  wire _1_;
  wire _2_;
  input a;
  input b;
  output y;
  sky130_fd_sc_hd__and2_0 _3_ (
    .A(_1_),
    .B(_0_),
    .X(_2_)
  );
  assign _1_ = b;
"multiple_modules_hier.v" 53L, 809C
```

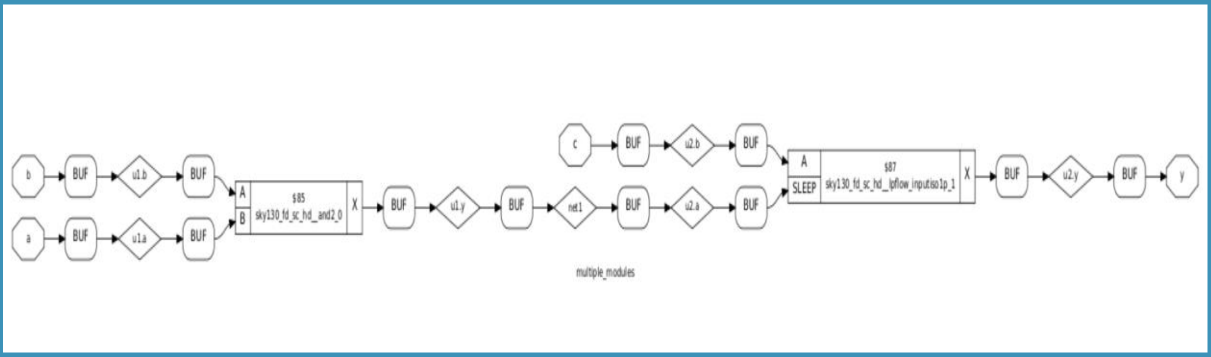
Modules are instantiated by u1 and u2 whereas submodule

Write Verilog Multiple Modules Hierarchy



```
yosys> write_verilog multiple_modules_hier.v

7. Executing Verilog backend.
Dumping module `\multiple_modules'.
Dumping module `\sub_module1'.
Dumping module `\sub_module2'.
```

We can directly see the structure completely when we flatten -

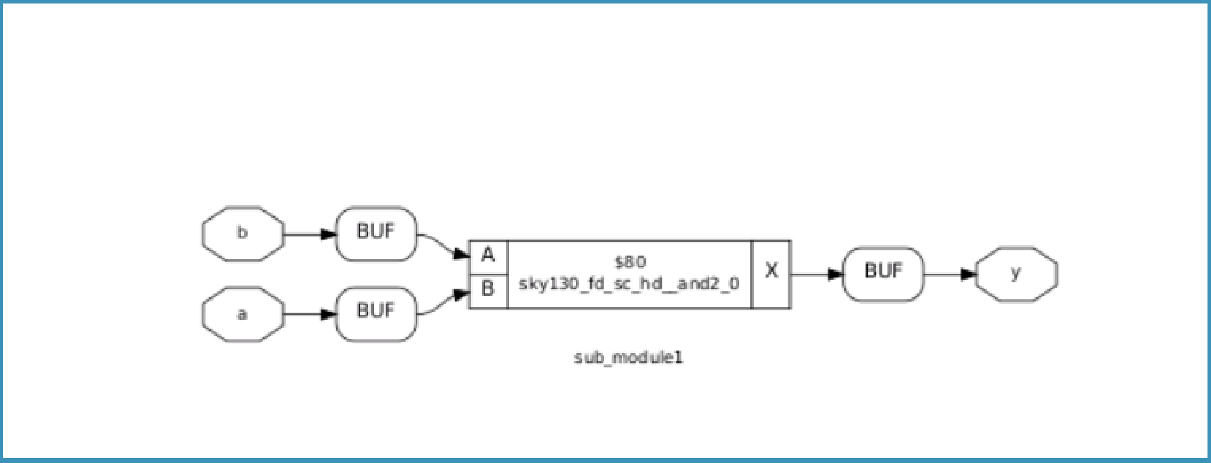


Sub-Module

Module Level Synthesis

I preferred when we have multiple instances of same module.

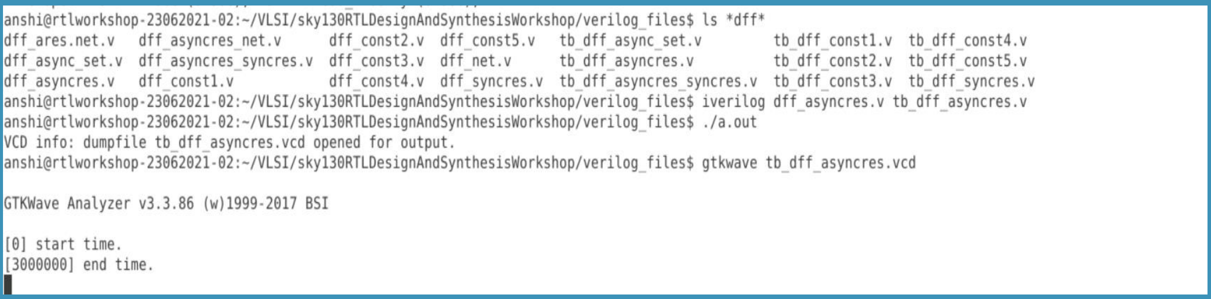
Or Using Divide and Conquer Approach in massive designs.



Using Synth -top command we can control the module to be synthesized.

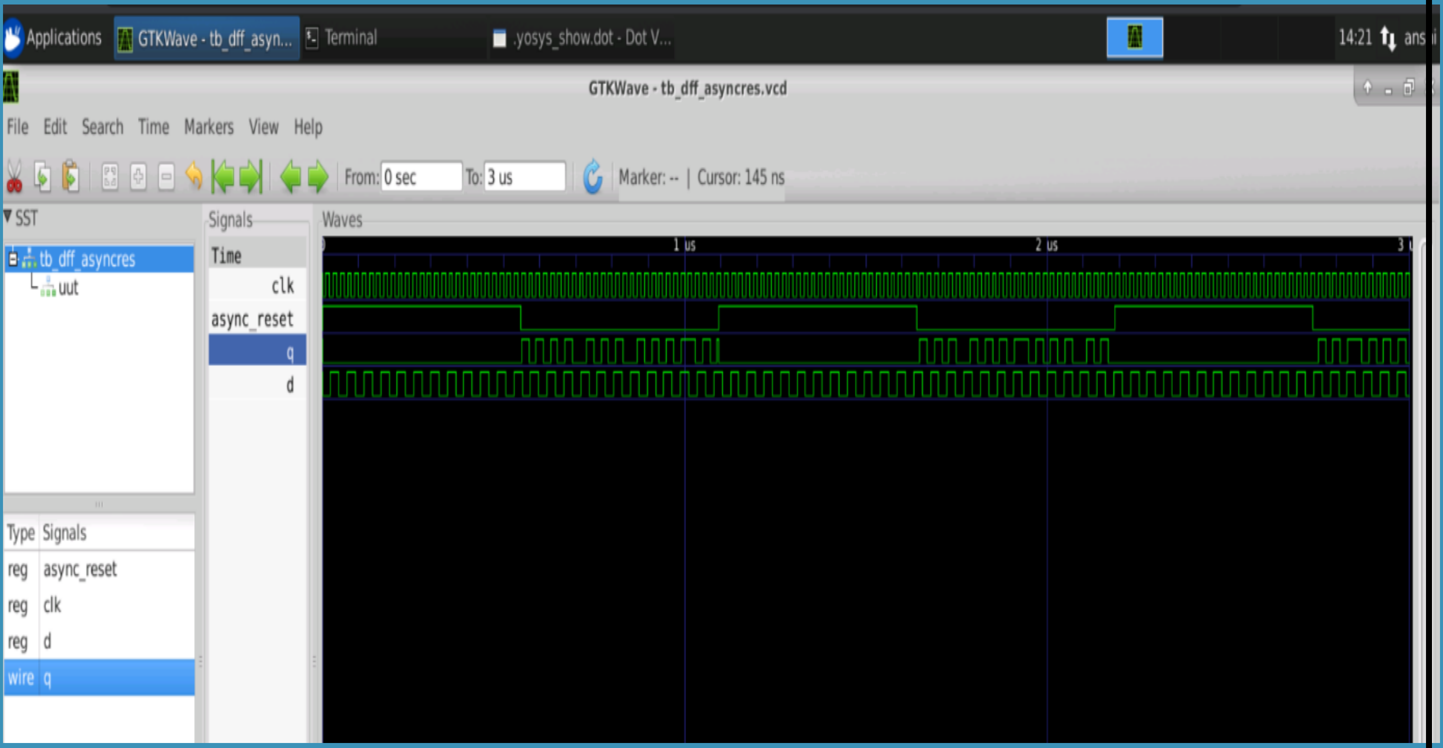
Various Flop Coding Styles and Optimization

DFF Asynchronous Reset

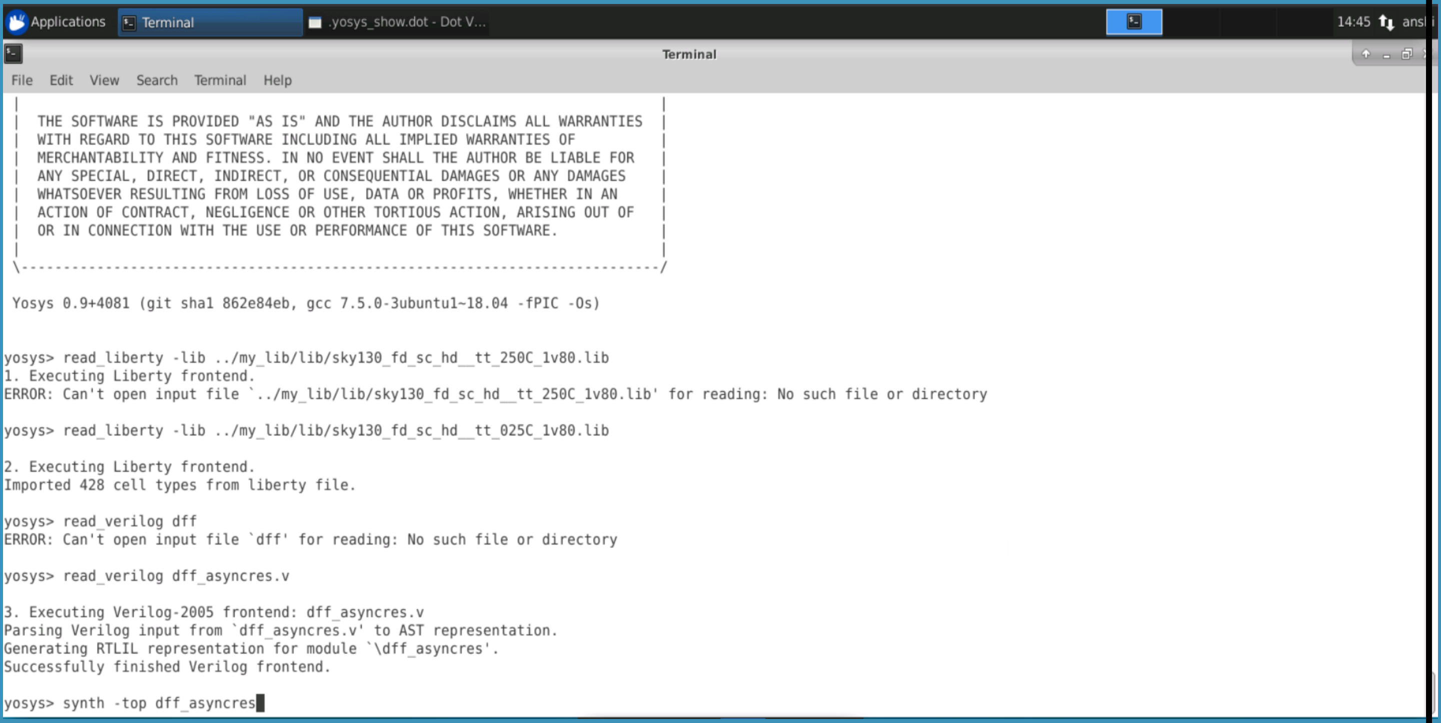


```
anshi@rtlworkshop-23062021-02:~/VLSI/sky130RTLDesignAndSynthesisWorkshop/verilog_files$ ls *dff*
dff_ares.net.v   dff_asyncres_net.v      dff_const2.v  dff_const5.v   tb_dff_async_set.v        tb_dff_const1.v  tb_dff_const4.v
dff_async_set.v  dff_asyncres_syncres.v  dff_const3.v  dff_net.v      tb_dff_asyncres.v         tb_dff_const2.v  tb_dff_const5.v
dff_asyncres.v   dff_const1.v            dff_const4.v  dff_syncres.v  tb_dff_asyncres_syncres.v  tb_dff_const3.v  tb_dff_syncres.v
anshi@rtlworkshop-23062021-02:~/VLSI/sky130RTLDesignAndSynthesisWorkshop/verilog_files$ iverilog dff_asyncres.v tb_dff_asyncres.v
anshi@rtlworkshop-23062021-02:~/VLSI/sky130RTLDesignAndSynthesisWorkshop/verilog_files$ ./a.out
VCD info: dumpfile tb_dff_asyncres.vcd opened for output.
anshi@rtlworkshop-23062021-02:~/VLSI/sky130RTLDesignAndSynthesisWorkshop/verilog_files$ gtkwave tb_dff_asyncres.vcd

GTKWave Analyzer v3.3.86 (w)1999-2017 BSI

[0] start time.
[3000000] end time.
```

Waveform of Dff Asynchronous Reset-Q follows d only when async_reset is low otherwise low.

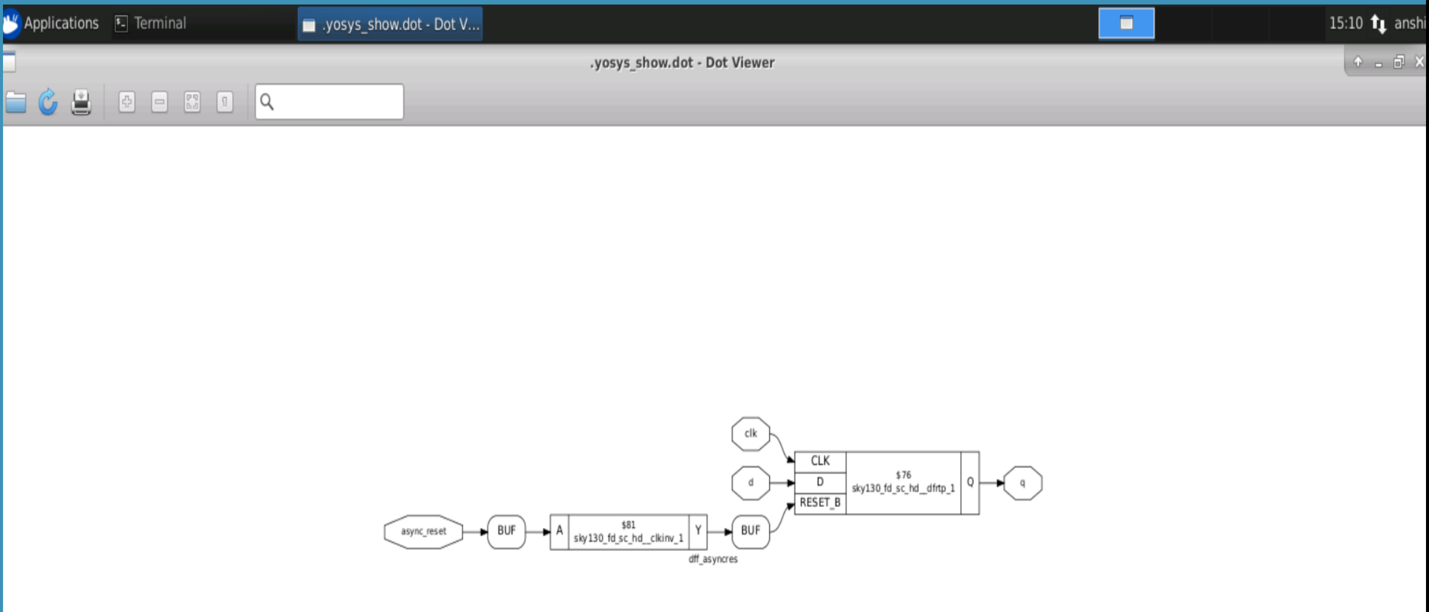


Synthesize

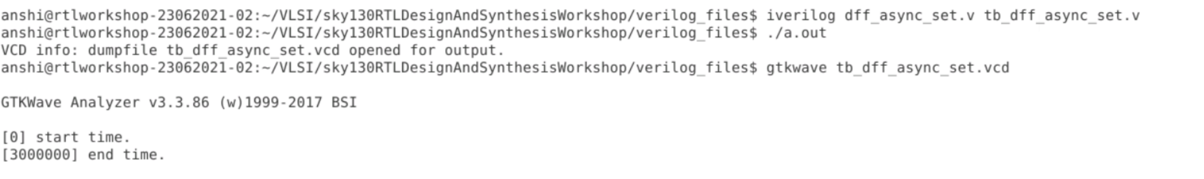

Applications
Terminal
.yosys_show.dot - Dot V...
14:45
ansl
Terminal
File Edit View Search Terminal Help

```
 |                                                                            |
 |  THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES  |
 |  WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF          |
 |  MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR   |
 |  ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES    |
 |  WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN     |
 |  ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF   |
 |  OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.            |
 |                                                                            |
 \----------------------------------------------------------------------------/

 Yosys 0.9+4081 (git sha1 862e84eb, gcc 7.5.0-3ubuntu1~18.04 -fPIC -Os)


yosys> read_liberty -lib ../my_lib/lib/sky130_fd_sc_hd__tt_250C_1v80.lib
1. Executing Liberty frontend.
ERROR: Can't open input file `../my_lib/lib/sky130_fd_sc_hd__tt_250C_1v80.lib' for reading: No such file or directory

yosys> read_liberty -lib ../my_lib/lib/sky130_fd_sc_hd__tt_025C_1v80.lib

2. Executing Liberty frontend.
Imported 428 cell types from liberty file.

yosys> read_verilog dff
ERROR: Can't open input file `dff' for reading: No such file or directory

yosys> read_verilog dff_asyncres.v

3. Executing Verilog-2005 frontend: dff_asyncres.v
Parsing Verilog input from `dff_asyncres.v' to AST representation.
Generating RTLIL representation for module `\dff_asyncres'.
Successfully finished Verilog frontend.

yosys> synth -top dff_asyncres
```

Design of the D-ff

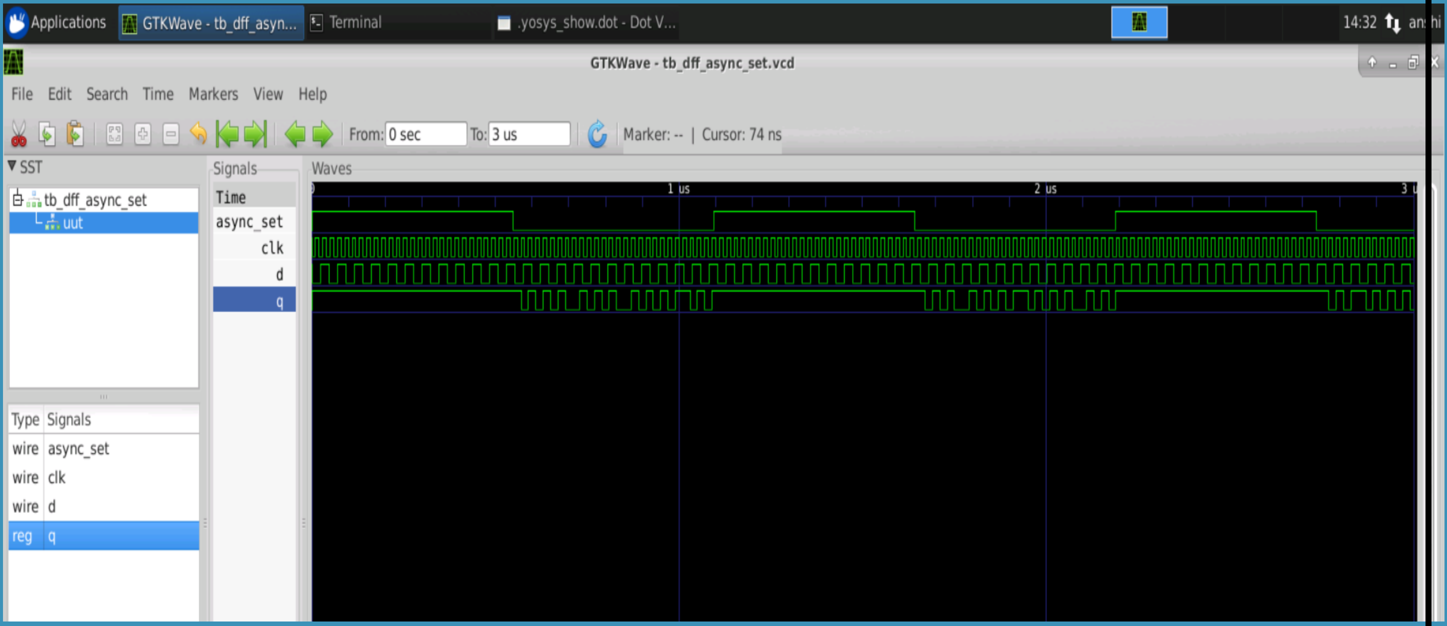


Asynchronous Set DFF

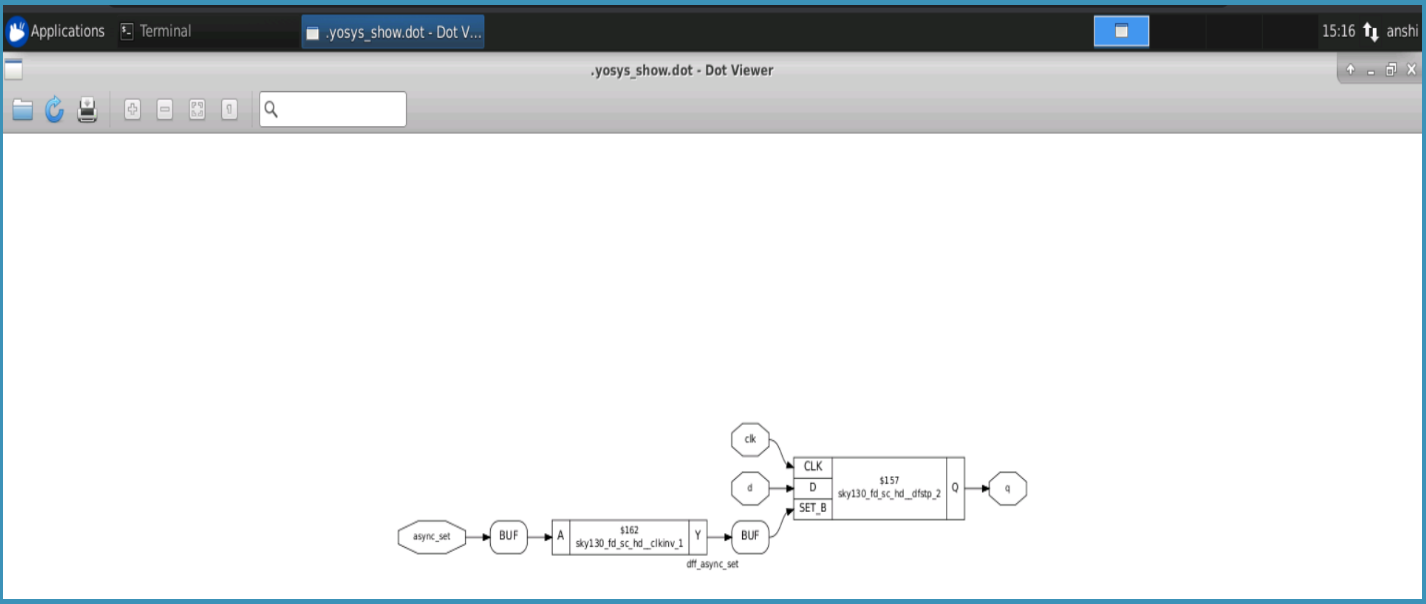


```
anshi@rtlworkshop-23062021-02:~/VLSI/sky130RTLDesignAndSynthesisWorkshop/verilog_files$ iverilog dff_async_set.v tb_dff_async_set.v
anshi@rtlworkshop-23062021-02:~/VLSI/sky130RTLDesignAndSynthesisWorkshop/verilog_files$ ./a.out
VCD info: dumpfile tb_dff_async_set.vcd opened for output.
anshi@rtlworkshop-23062021-02:~/VLSI/sky130RTLDesignAndSynthesisWorkshop/verilog_files$ gtkwave tb_dff_async_set.vcd

GTKWave Analyzer v3.3.86 (w)1999-2017 BSI

[0] start time.
[3000000] end time.
```

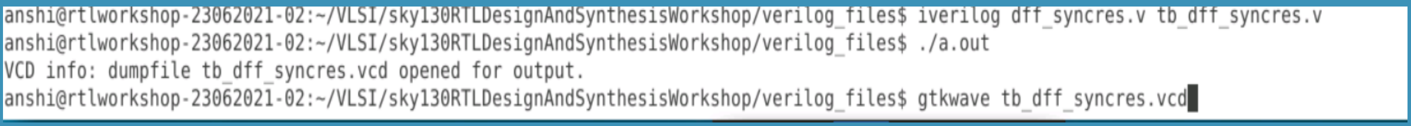
GtkWave Output



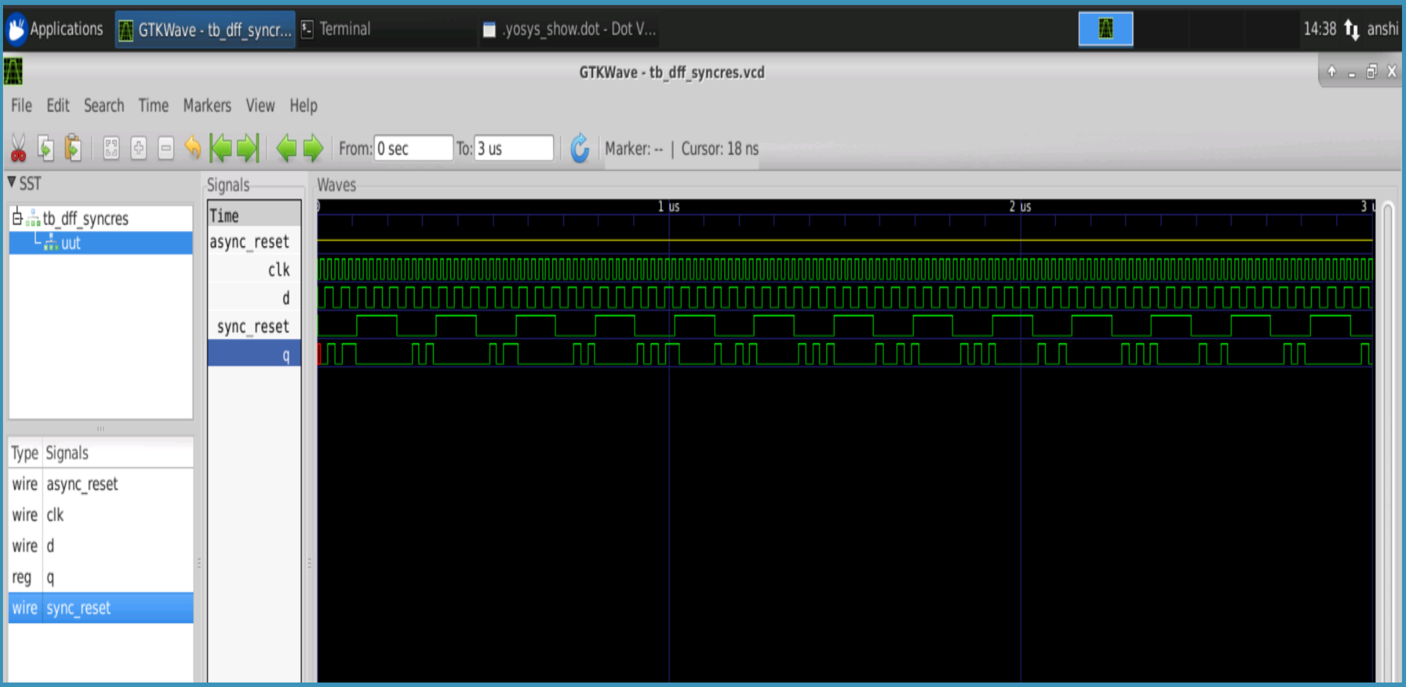
Sythesizer

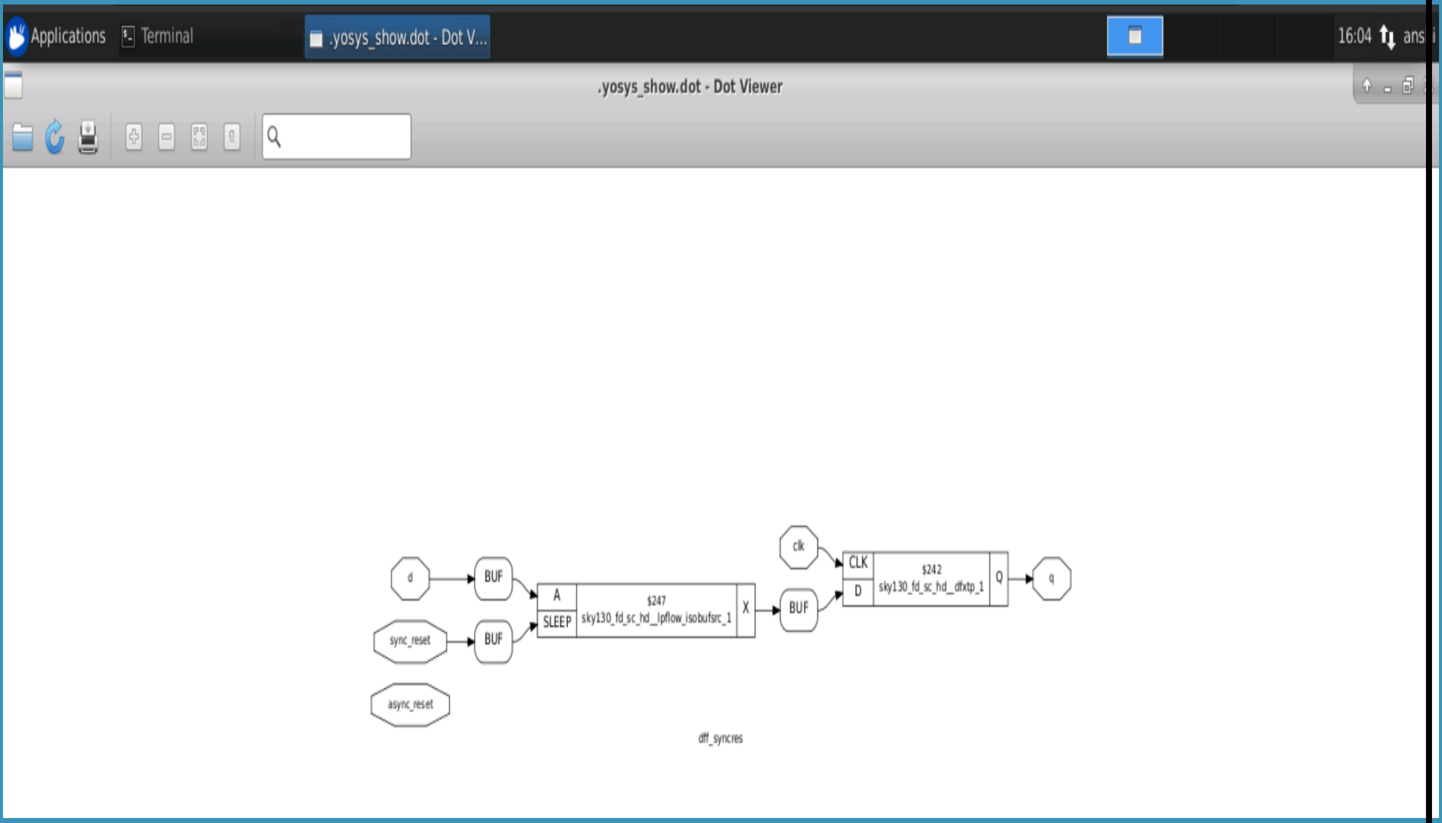


Synchronous Reset-



```
anshi@rtlworkshop-23062021-02:~/VLSI/sky130RTLDesignAndSynthesisWorkshop/verilog_files$ iverilog dff_syncres.v tb_dff_syncres.v
anshi@rtlworkshop-23062021-02:~/VLSI/sky130RTLDesignAndSynthesisWorkshop/verilog_files$ ./a.out
VCD info: dumpfile tb_dff_syncres.vcd opened for output.
anshi@rtlworkshop-23062021-02:~/VLSI/sky130RTLDesignAndSynthesisWorkshop/verilog_files$ gtkwave tb_dff_syncres.vcd
```

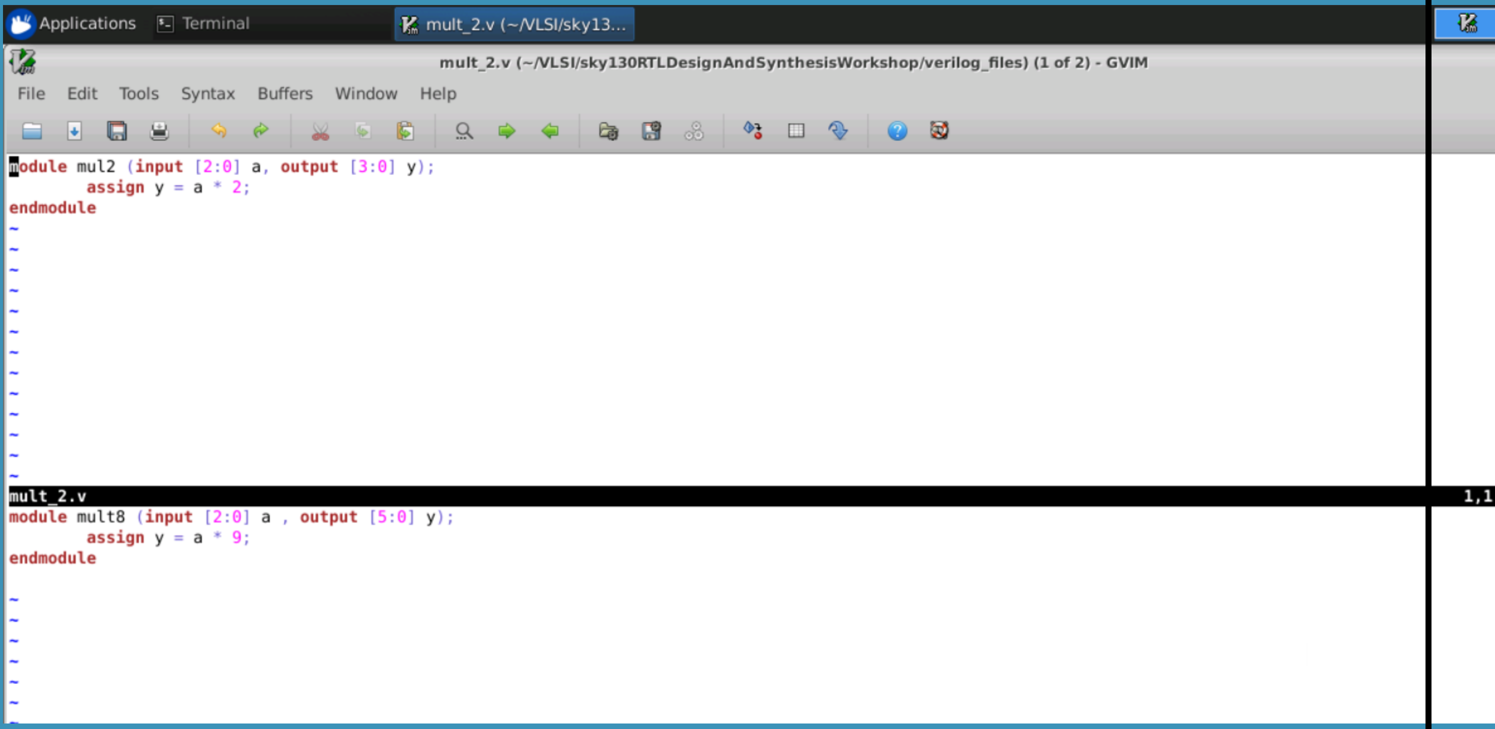




Optimization

Multiplication

Initial


Applications
Terminal
mult_2.v (~/VLSI/sky13...
mult_2.v (~/VLSI/sky130RTLDesignAndSynthesisWorkshop/verilog_files) (1 of 2) - GVIM
File Edit Tools Syntax Buffers Window Help

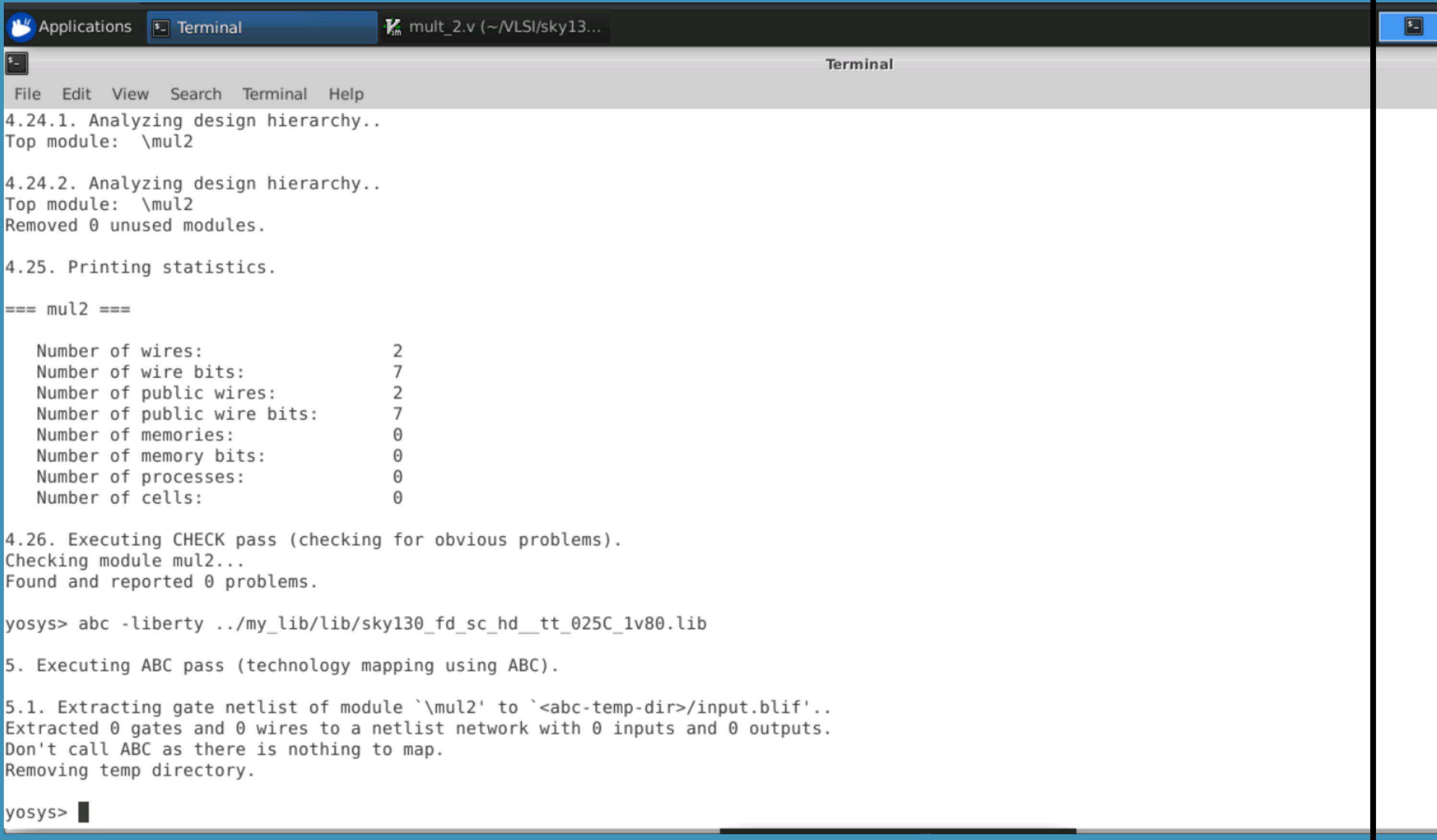
```
module mul2 (input [2:0] a, output [3:0] y);
        assign y = a * 2;
endmodule
```

```
mult_2.v                                                    1,1
```

```
module mult8 (input [2:0] a , output [5:0] y);
        assign y = a * 9;
endmodule
```

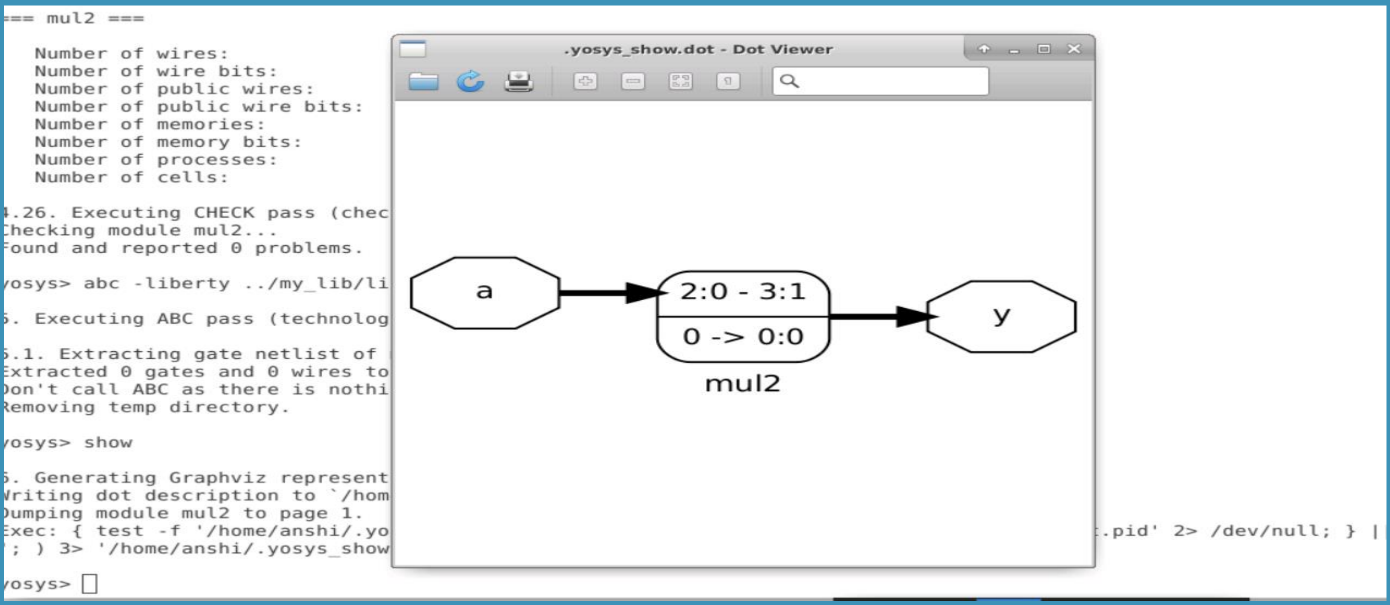
Zero memories, processes and no cells to be synthesized

After ABC pass, it shows there nothing to map

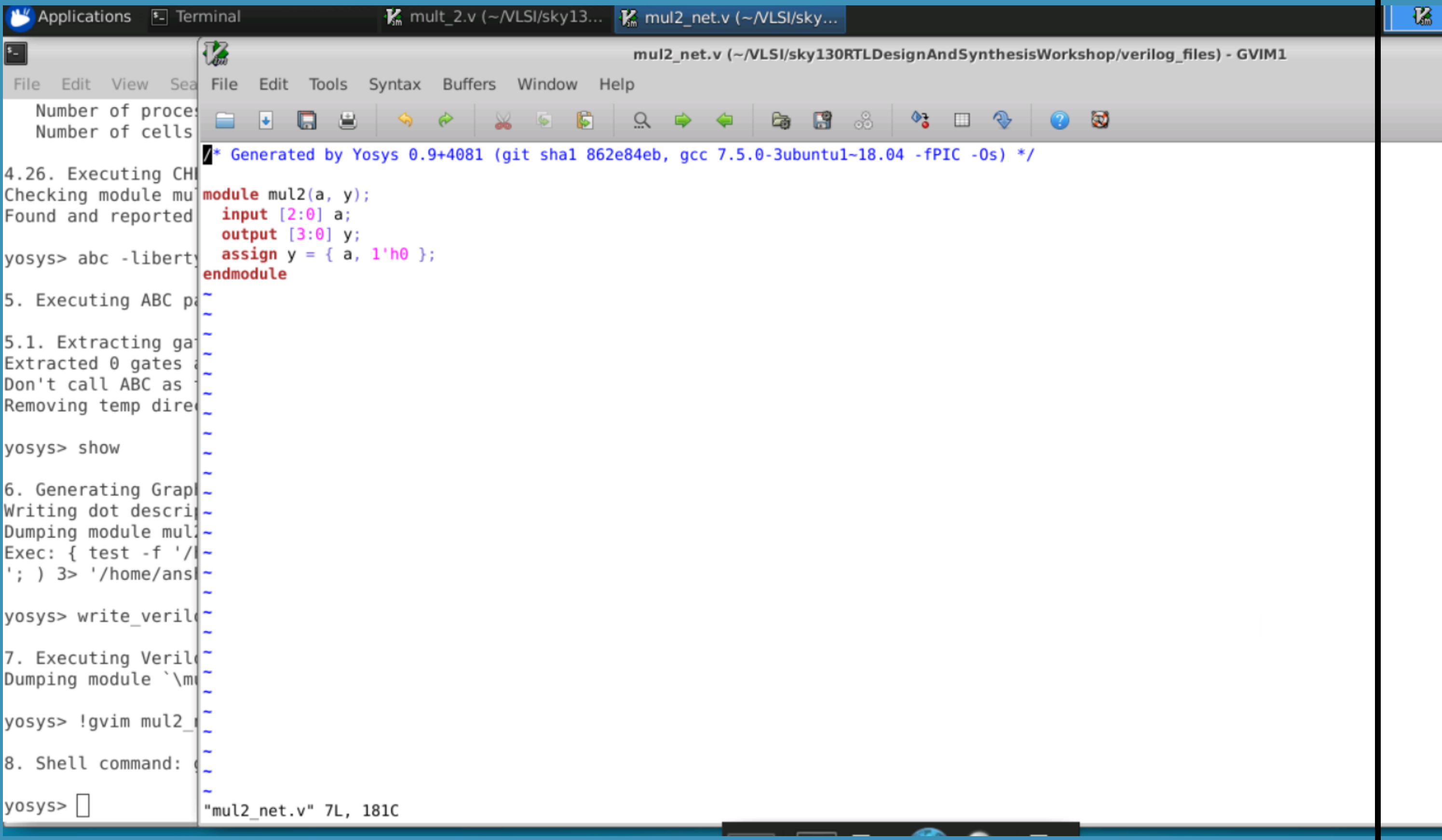

Applications
Terminal
mult_2.v (~/VLSI/sky13...
Terminal
File Edit View Search Terminal Help

```
4.24.1. Analyzing design hierarchy..
Top module:  \mul2

4.24.2. Analyzing design hierarchy..
Top module:  \mul2
Removed 0 unused modules.

4.25. Printing statistics.

=== mul2 ===

   Number of wires:                 2
   Number of wire bits:             7
   Number of public wires:          2
   Number of public wire bits:      7
   Number of memories:              0
   Number of memory bits:           0
   Number of processes:             0
   Number of cells:                 0

4.26. Executing CHECK pass (checking for obvious problems).
Checking module mul2...
Found and reported 0 problems.

yosys> abc -liberty ../my_lib/lib/sky130_fd_sc_hd__tt_025C_1v80.lib

5. Executing ABC pass (technology mapping using ABC).

5.1. Extracting gate netlist of module `\mul2' to `<abc-temp-dir>/input.blif'..
Extracted 0 gates and 0 wires to a netlist network with 0 inputs and 0 outputs.
Don't call ABC as there is nothing to map.
Removing temp directory.

yosys> █
```

Here no gate is synthesized. Only 2-input a is converted to y by appending a zero in the end.



```
=== mul2 ===

   Number of wires:
   Number of wire bits:
   Number of public wires:
   Number of public wire bits:
   Number of memories:
   Number of memory bits:
   Number of processes:
   Number of cells:

.26. Executing CHECK pass (chec
hecking module mul2...
ound and reported 0 problems.

osys> abc -liberty ../my_lib/li

. Executing ABC pass (technolog

.1. Extracting gate netlist of
xtracted 0 gates and 0 wires to
on't call ABC as there is nothi
emoving temp directory.

osys> show

. Generating Graphviz represent
riting dot description to `/hom
umping module mul2 to page 1.
xec: { test -f '/home/anshi/.yo                                              .pid' 2> /dev/null; } |
; ) 3> '/home/anshi/.yosys_show

osys> □
```



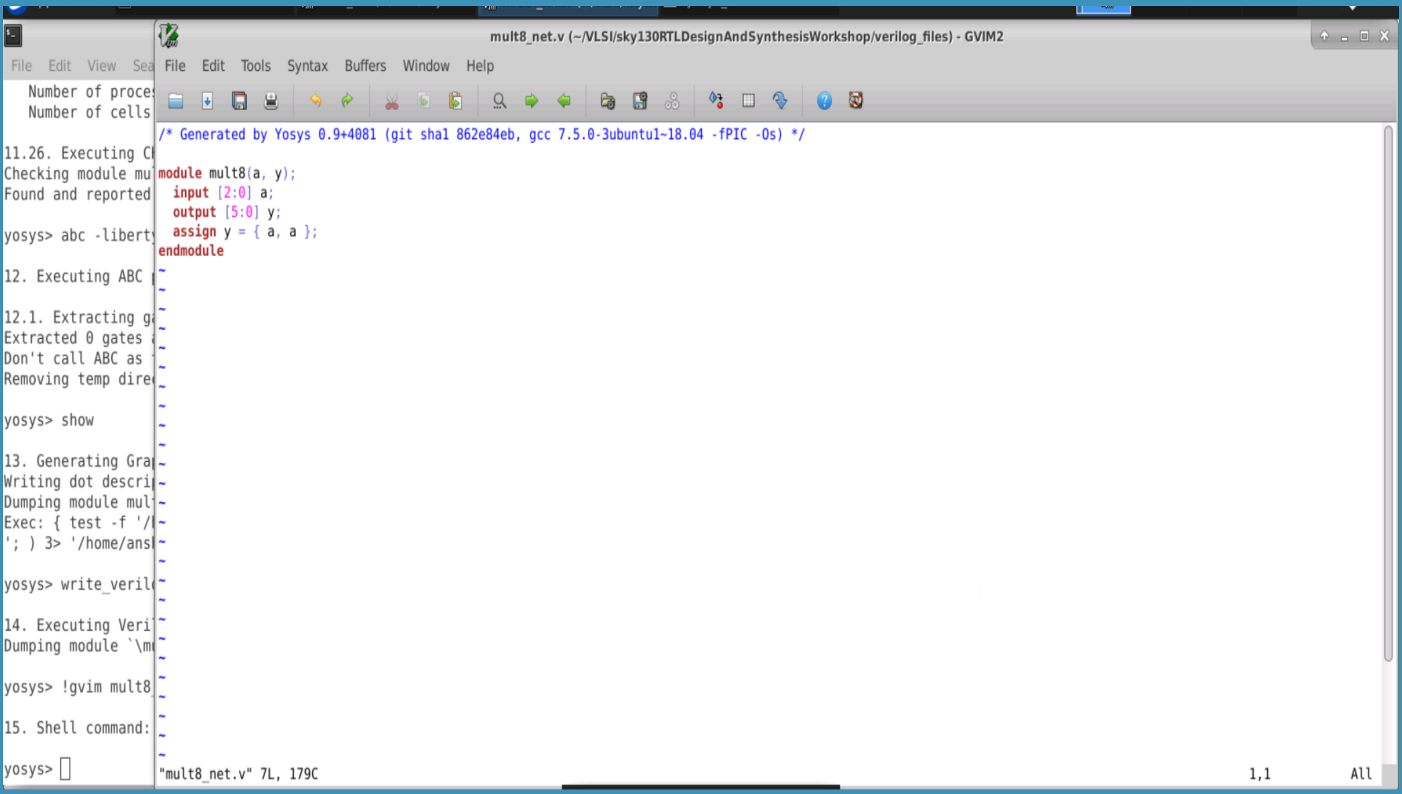
Applications
Terminal
mult_2.v (~/VLSI/sky13...
mul2_net.v (~/VLSI/sky...
mul2_net.v (~/VLSI/sky130RTLDesignAndSynthesisWorkshop/verilog_files) - GVIM1
File Edit Tools Syntax Buffers Window Help
File Edit View Sea

```
  Number of proce
  Number of cells

4.26. Executing CHI
Checking module mu
Found and reported

yosys> abc -liberty

5. Executing ABC pa

5.1. Extracting ga
Extracted 0 gates a
Don't call ABC as
Removing temp dire

yosys> show

6. Generating Grapl
Writing dot descrip
Dumping module mul2
Exec: { test -f '/
'; ) 3> '/home/ansl

yosys> write_verilo

7. Executing Veril
Dumping module `\mu

yosys> !gvim mul2_n

8. Shell command: g

yosys>
```

```
/* Generated by Yosys 0.9+4081 (git sha1 862e84eb, gcc 7.5.0-3ubuntu1~18.04 -fPIC -Os) */

module mul2(a, y);
  input [2:0] a;
  output [3:0] y;
  assign y = { a, 1'h0 };
endmodule
"mul2_net.v" 7L, 181C
```

Hence, no hardware required for Mult2

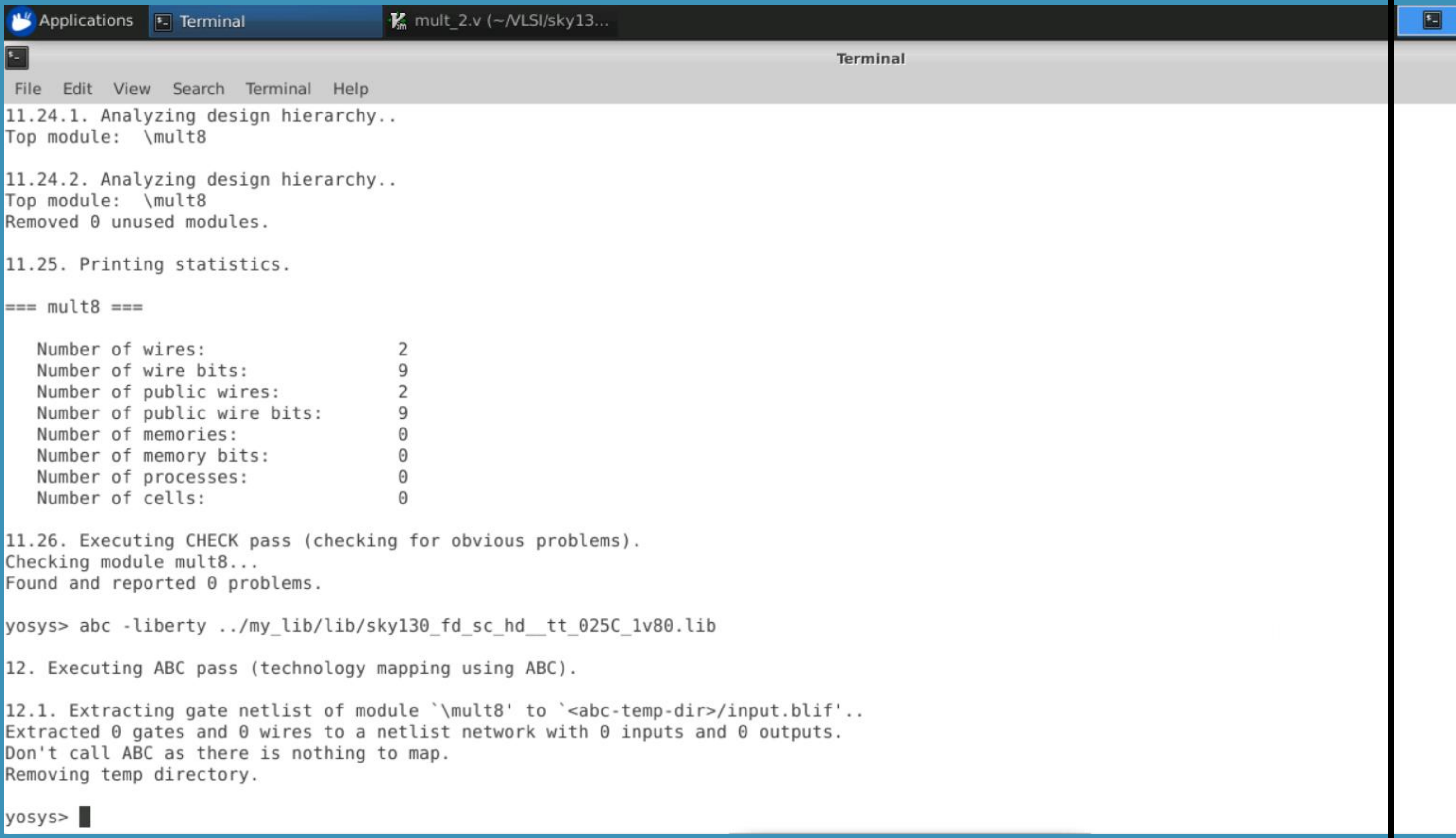
Multiply by 9



mult8_net.v (~/VLSI/sky130RTLDesignAndSynthesisWorkshop/verilog_files) - GVIM2
File Edit Tools Syntax Buffers Window Help
File Edit View Sea

```
  Number of proce
  Number of cells

11.26. Executing CI
Checking module mu
Found and reported

yosys> abc -liberty

12. Executing ABC

12.1. Extracting ga
Extracted 0 gates a
Don't call ABC as
Removing temp dire

yosys> show

13. Generating Gra
Writing dot descri
Dumping module mul
Exec: { test -f '/
'; ) 3> '/home/ansl

yosys> write_veril

14. Executing Veri
Dumping module `\m

yosys> !gvim mult8

15. Shell command:

yosys>
```

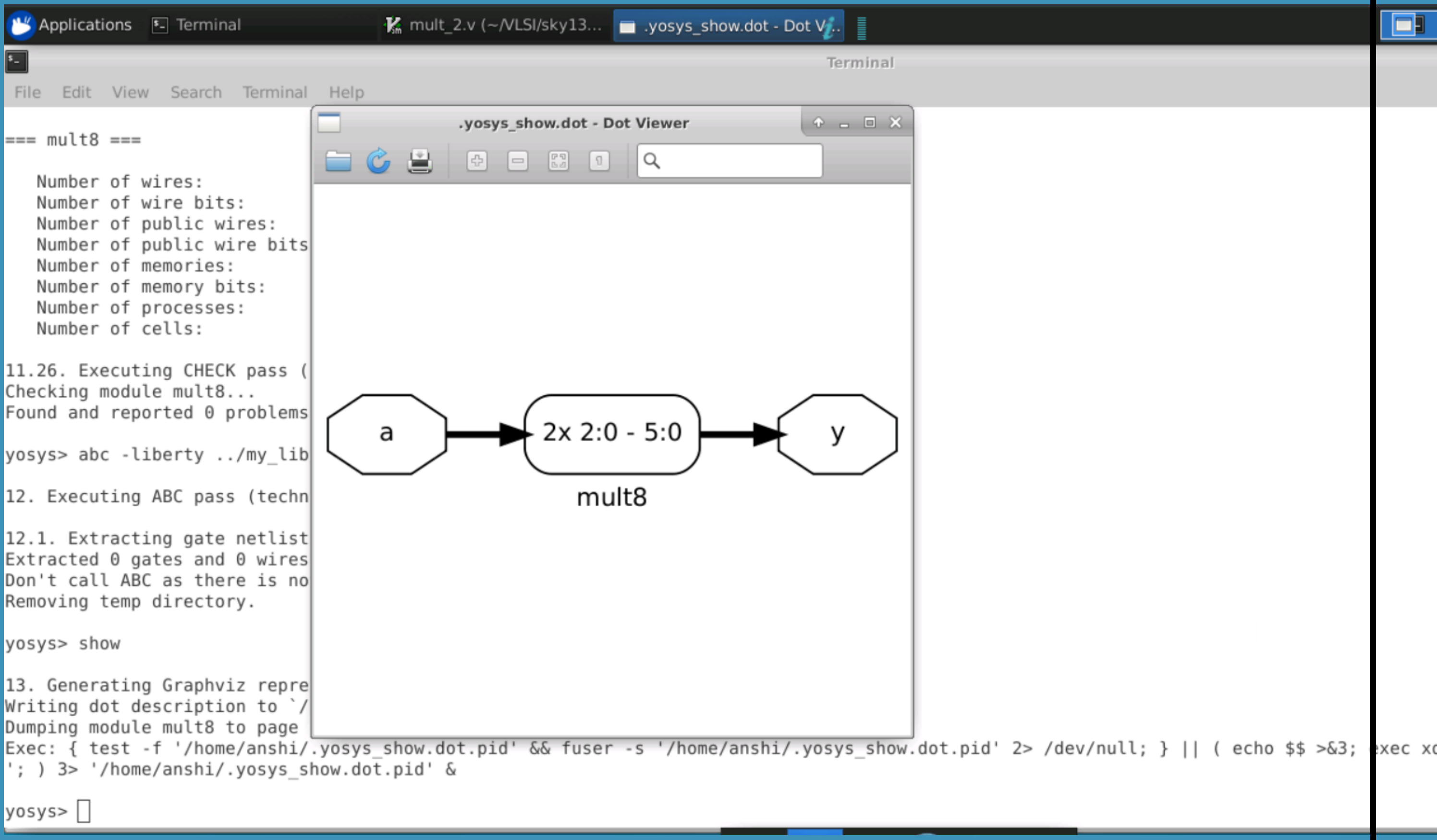
```
/* Generated by Yosys 0.9+4081 (git sha1 862e84eb, gcc 7.5.0-3ubuntu1~18.04 -fPIC -Os) */

module mult8(a, y);
  input [2:0] a;
  output [5:0] y;
  assign y = { a, a };
endmodule
"mult8_net.v" 7L, 179C                                        1,1          All
```



Applications
Terminal
mult_2.v (~/VLSI/sky13...
Terminal
File Edit View Search Terminal Help

```
11.24.1. Analyzing design hierarchy..
Top module:  \mult8

11.24.2. Analyzing design hierarchy..
Top module:  \mult8
Removed 0 unused modules.

11.25. Printing statistics.

=== mult8 ===

   Number of wires:                 2
   Number of wire bits:             9
   Number of public wires:          2
   Number of public wire bits:      9
   Number of memories:              0
   Number of memory bits:           0
   Number of processes:             0
   Number of cells:                 0

11.26. Executing CHECK pass (checking for obvious problems).
Checking module mult8...
Found and reported 0 problems.

yosys> abc -liberty ../my_lib/lib/sky130_fd_sc_hd__tt_025C_1v80.lib

12. Executing ABC pass (technology mapping using ABC).

12.1. Extracting gate netlist of module `\mult8' to `<abc-temp-dir>/input.blif'..
Extracted 0 gates and 0 wires to a netlist network with 0 inputs and 0 outputs.
Don't call ABC as there is nothing to map.
Removing temp directory.

yosys> 
```

Show



Applications
Terminal
mult_2.v (~/VLSI/sky13...
.yosys_show.dot - Dot V...
Terminal
File Edit View Search Terminal Help
.yosys_show.dot - Dot Viewer
a
2x 2:0 - 5:0
mult8
y

```
=== mult8 ===

   Number of wires:
   Number of wire bits:
   Number of public wires:
   Number of public wire bits
   Number of memories:
   Number of memory bits:
   Number of processes:
   Number of cells:

11.26. Executing CHECK pass (
Checking module mult8...
Found and reported 0 problems

yosys> abc -liberty ../my_lib

12. Executing ABC pass (techn

12.1. Extracting gate netlist
Extracted 0 gates and 0 wires
Don't call ABC as there is no
Removing temp directory.

yosys> show

13. Generating Graphviz repre
Writing dot description to `/
Dumping module mult8 to page
Exec: { test -f '/home/anshi/.yosys_show.dot.pid' && fuser -s '/home/anshi/.yosys_show.dot.pid' 2> /dev/null; } || ( echo $$ >&3; exec xd
'; ) 3> '/home/anshi/.yosys_show.dot.pid' &

yosys> 
```

So, 2 a [2:0] are mapped to the 6 bits of y

So,we don't need any hardware just by rewiring the signals we can implement the logic