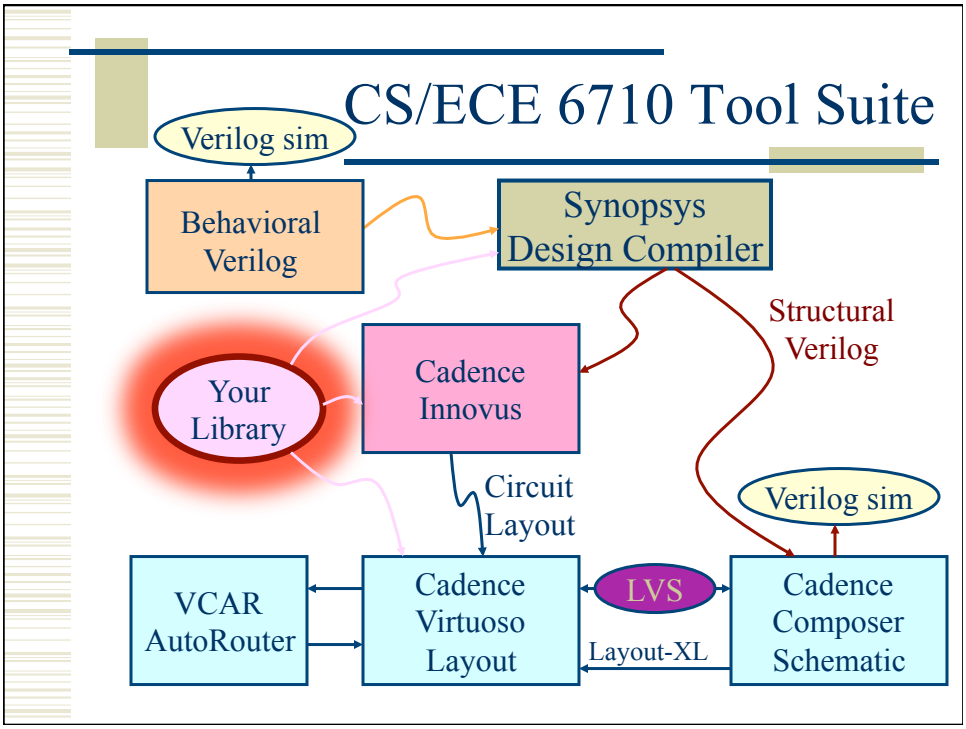
CAD5

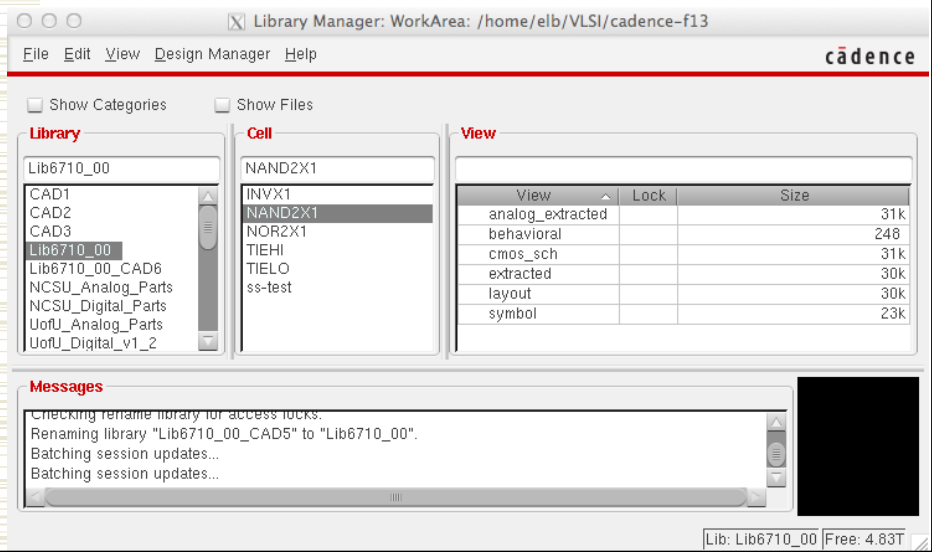
- Designing the first five cells in your library
  - Multiple cell views
- Liberate library characterizer
- Abstract generator
- Synopsys database generation
  - Using the cells in synthesis



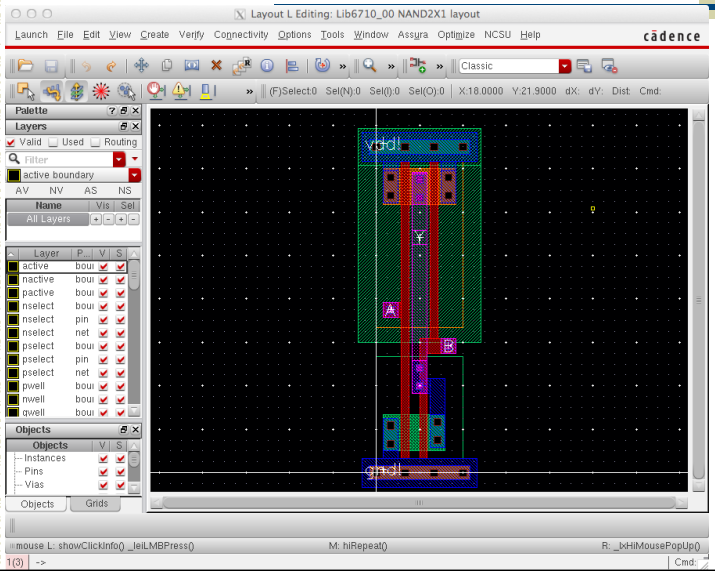
In the CAD Book

- Chapter 8 on Cell Characterization
  - Section 8.1 describes .lib format
  - Section 8.2 describes ELC – that tool is gone…
    - Instead we'll use Cadence Liberate
  - Section 8.3 describes characterization by hand with Spectre (don't do it!)
  - Section 8.4 describes converting from .lib to .db format (used by Synopsys Design Compiler)
    - Use syn-lc instead of syn-dc…
    - (.lib is used also by other tools…)

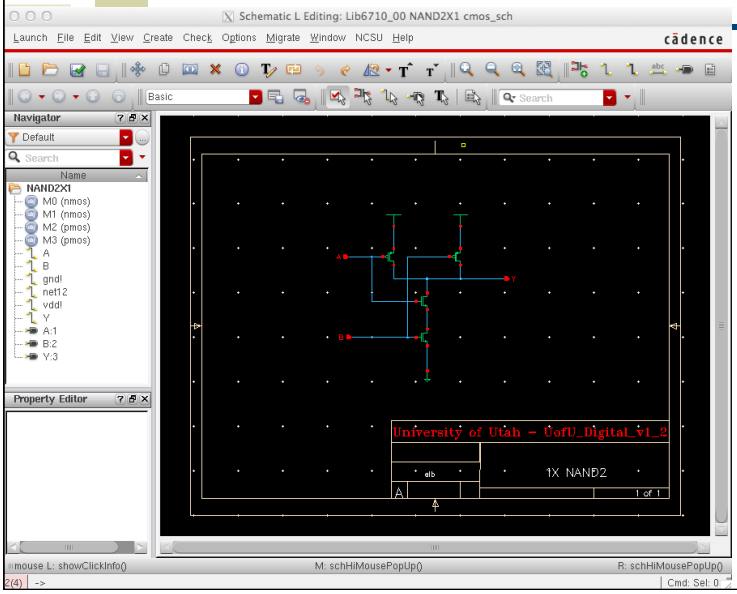
Start with Cells



Layout View



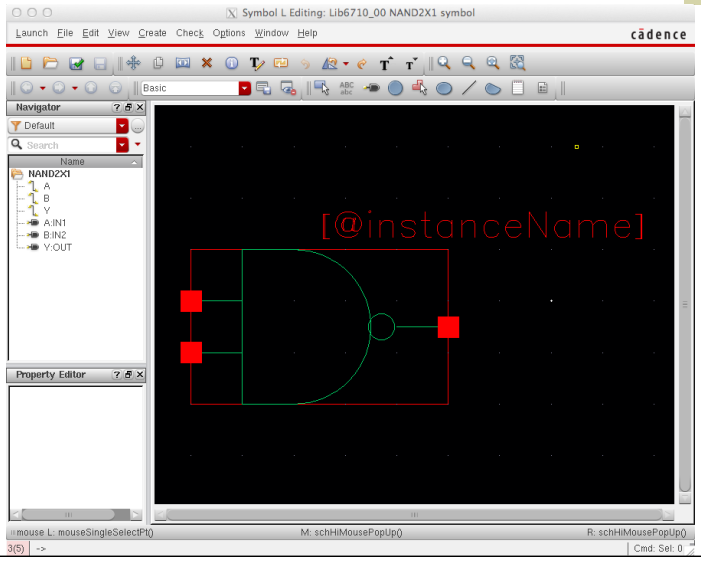
Schematic View



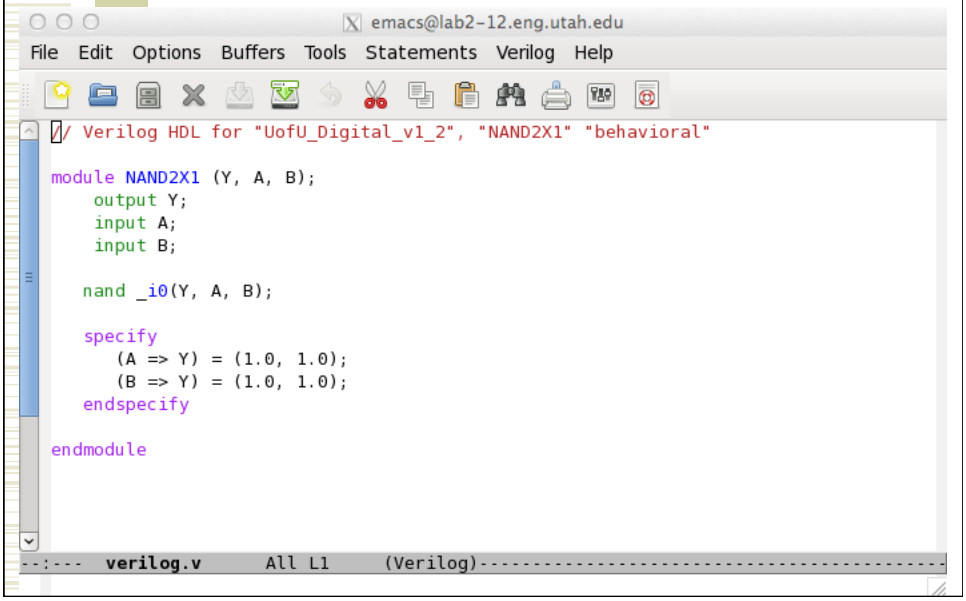
Use cmos_sch as the view type for your cell schematics

This is to differentiate them from schematics that use the cells.

Symbol View



Behavioral View



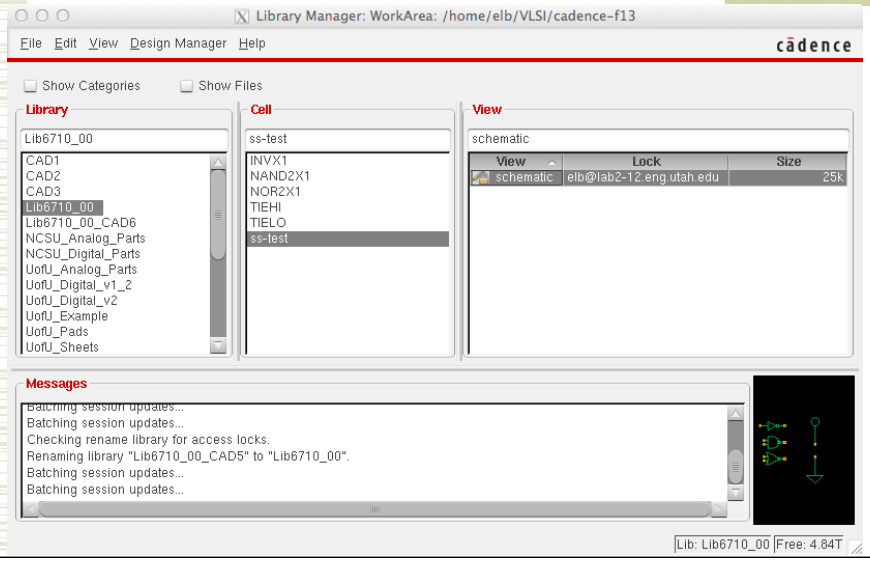
Start with six views of each cell

1. layout - make sure to use the template!
2. cmos_sch – use this view type for a schematic
3. symbol – Make them look nice
4. behavioral – Having this view makes simulation go much faster (if you use it)
5. extracted – generated from the extract processes
6. analog_extracted – after LVS

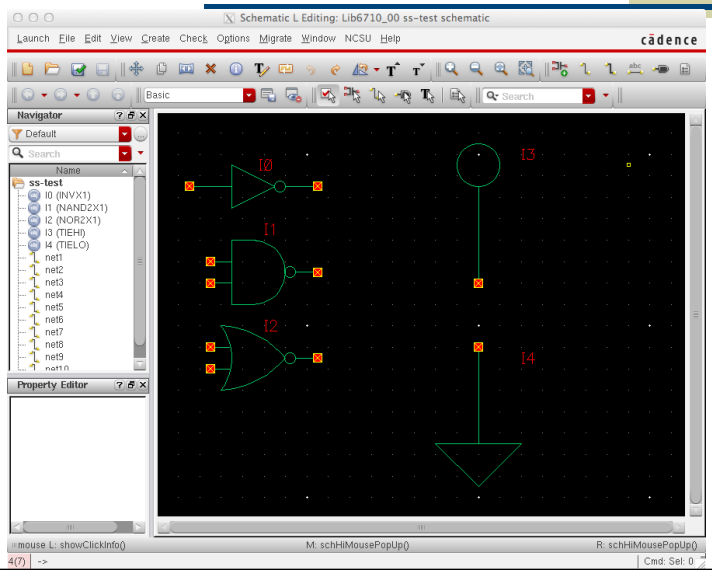
Characterize the cells

- Run a set of analog (Spectre) simulations that builds a table of delay values
  - Input drive vs. output load
  - For all outputs
  - Plus rise and fall times
  - And some power information
- Cadence Liberate
  - cad-lib from the bin directory
  - But, don't call it directly… use additional scripts…

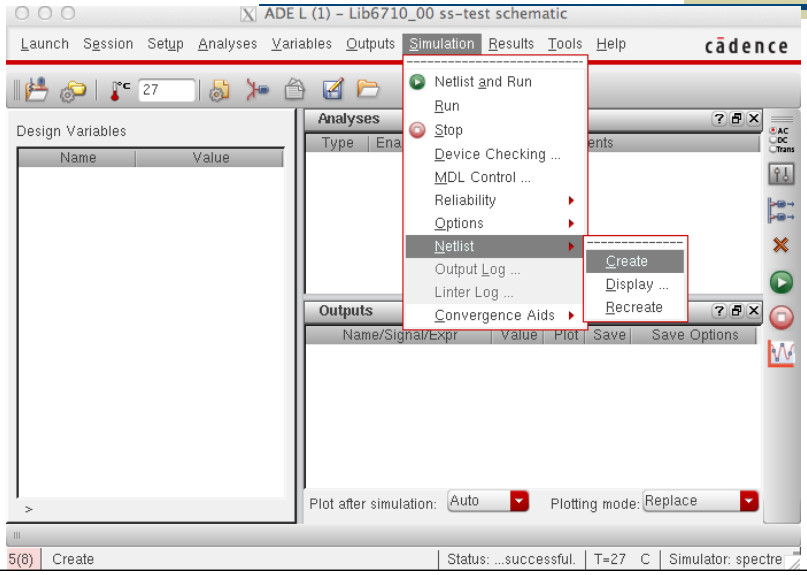
Single Schematic with All Cells



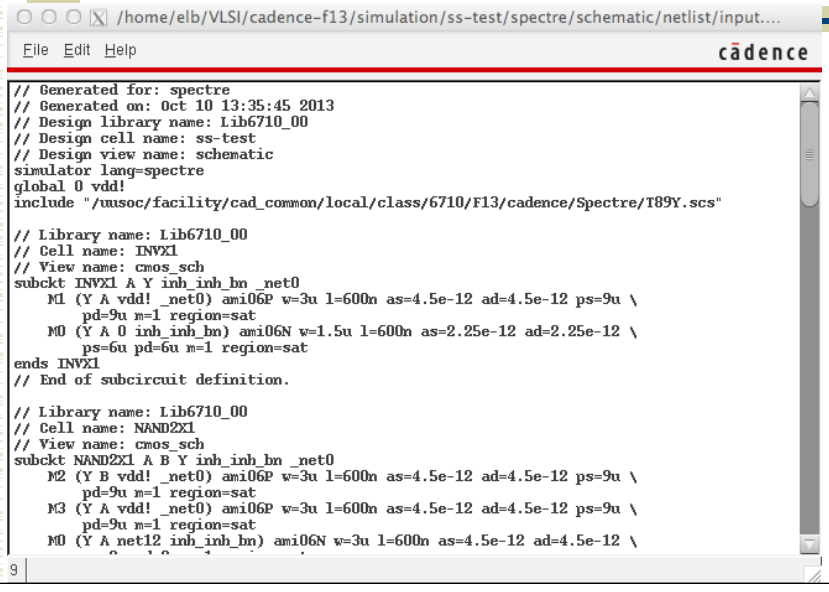
Single Schematic with All Cells



Create Netlist of that ss-test Cell



Call it foo.scs (for example)



`scs2liberate foo.scs libcells.scs`

Convert to Liberate format

```
simulator lang=spectre
global vss vdd

subckt INVX1 A Y
    M1 (Y A vdd vdd) ami06P w=6u l=600n as=9e-12 ad=9e-12 ps=15.0u \
        pd=15.0u m=1 region=sat
    M0 (Y A vss vss) ami06N w=3u l=600n as=4.5e-12 ad=4.5e-12 ps=9u \
        pd=9u m=1 region=sat
ends INVX1
subckt NAND2X1 A B Y
    M2 (Y A vdd vdd) ami06P w=6u l=600n as=9e-12 ad=9e-12 ps=15.0u \
        pd=15.0u m=1 region=sat
    M3 (Y B vdd vdd) ami06P w=6u l=600n as=9e-12 ad=9e-12 ps=15.0u \
        pd=15.0u m=1 region=sat
    M0 (Y B net12 vss) ami06N w=6u l=600n as=9e-12 ad=9e-12 \
        ps=15.0u pd=15.0u m=1 region=sat
    M1 (net12 A vss vss) ami06N w=6u l=600n as=9e-12 ad=9e-12 \
        ps=15.0u pd=15.0u m=1 region=sat
ends NAND2X1
```

Run Liberate

- Liberate Library Characterizer
  - Figures out what each cell is (logic)
  - Generates test inputs for Spectre
  - Runs Spectre
  - Checks output and extracts timings
  - Formats the output in .lib format
- I like to make a new VLSI/Liberate directory from which to run this tool…
  - cp –r /uusoc/facility/cad_common/local/class/6710/F17/cadence/Liberate .

Liberate setup…

- A **lib** directory
  - where the generated **<library>.lib** file will be generated.
- A **netlist** directory
  - put your **libcells.scs** file in this directory.

```
[elb@lab2-20 Liberate]$ ls
models/   README   tcl/            templates/      userdata/
netlist/  run.sh*  TechHeader.lef  UofU_Cells.tcl
[elb@lab2-20 Liberate]$
```

Liberate setup

- A **templates** directory
  - edit the **UofU_Cell_Defs.tcl** file in this directory to include cell descriptions for each of the cells you want to characterize. The **UofU_Templates.tcl** file has definitions for the timing and power templates that will be used for characterization. You probably don't need to modify these unless you are using a different technology than ON Semi C5N.


```
[elb@lab2-20 Liberate]$ ls
models/   README   tcl/            templates/      userdata/
netlist/  run.sh*  TechHeader.lef  UofU_Cells.tcl
[elb@lab2-20 Liberate]$
```

Liberate setup

- A **UofU_Cells.tcl** file
  - edit this file to make a list of the cells that you want to characterize in this run. This could be a list of every cell described in **templates/ UofU_Cell_Defs.tcl**, or it could be a subset if you just want to try a few.
- A **userdata** directory
  - edit the **userdata.lib** file to reflect the areas and footprints of each of the cells in your library.

```
[elb@lab2-20 Liberate]$ ls
models/   README   tcl/            templates/      userdata/
netlist/  run.sh*  TechHeader.lef  UofU_Cells.tcl
[elb@lab2-20 Liberate]$
```

Liberate setup

- A **tcl** directory
  - Edit the **UofU_Char.tcl** file in this directory to change the name of the library that the tool generates. If you don't modify this, the tool will generate a file named **Lib6710_XX.lib** by default. The **settings.tcl** file in this directory has configuration commands for the Spectre simulator. You won't need to modify this file at all.

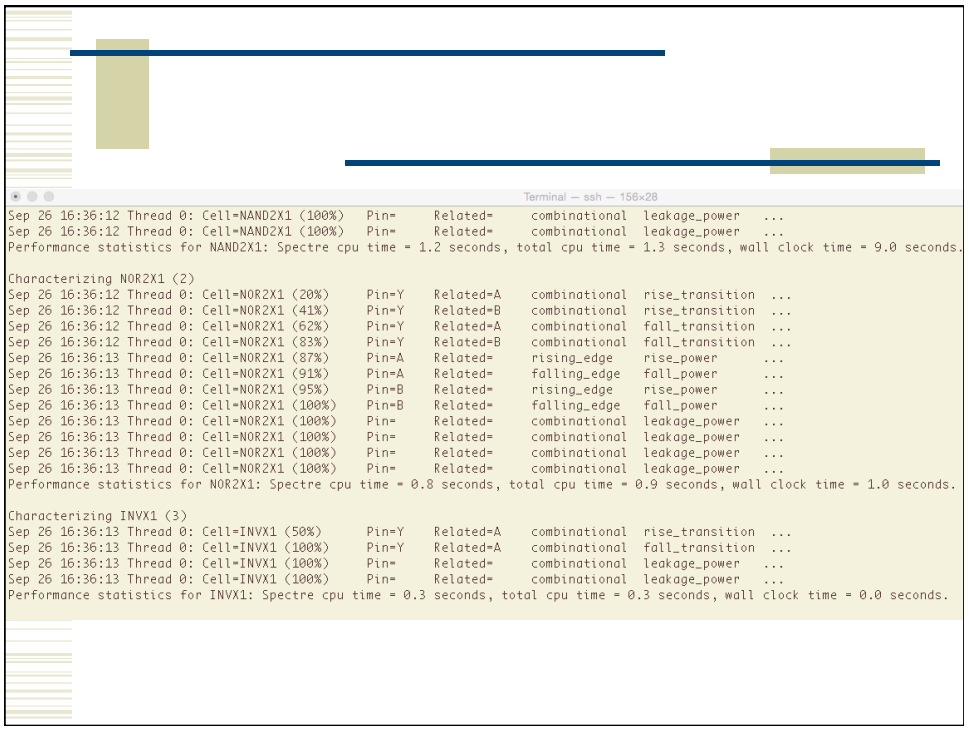
```
[elb@lab2-20 Liberate]$ ls
models/   README   tcl/            templates/      userdata/
netlist/  run.sh*  TechHeader.lef  UofU_Cells.tcl
[elb@lab2-20 Liberate]$
```

Liberate setup

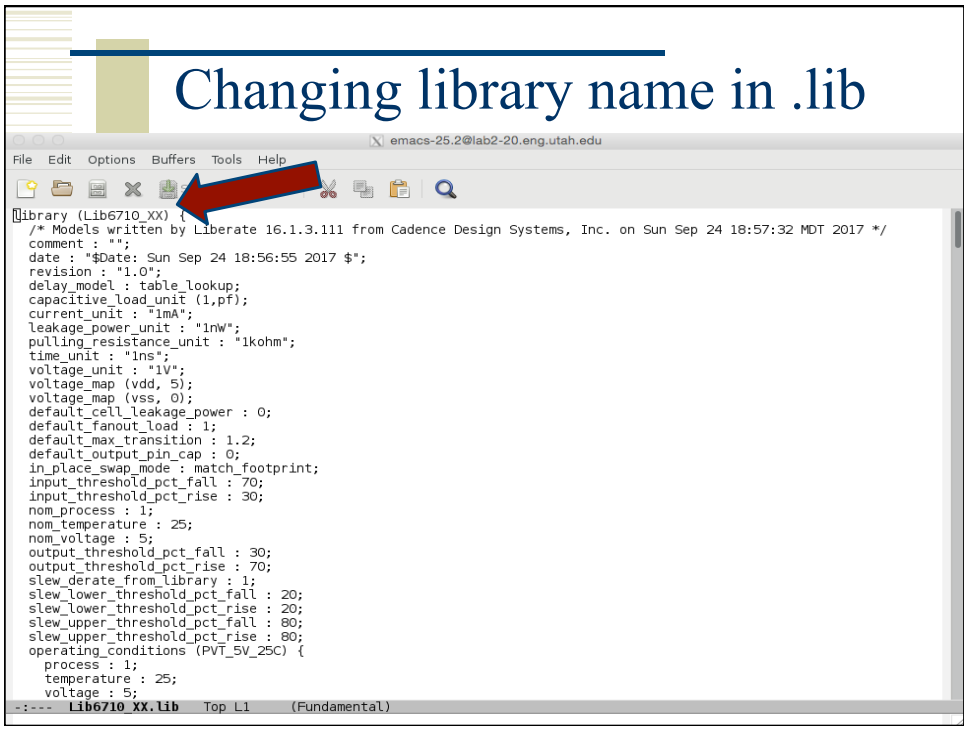
- A **models** directory
  - Has the Spectre model files for the ami06N and ami06P transistors used by in your netlist.
- A **run.sh** shell script
  - This calls the cad-lib Liberate script with the appropriate input files, and makes a copy of the log information in a **Liberate.log** file.

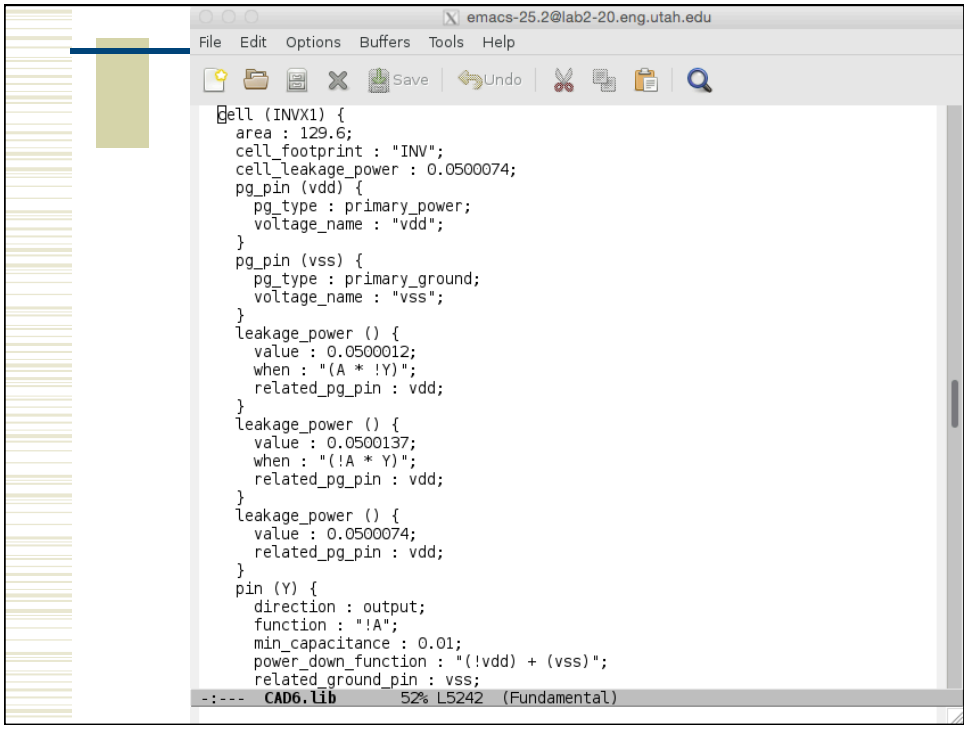
```
[elb@lab2-20 Liberate]$ ls
models/   README   tcl/            templates/      userdata/
netlist/  run.sh*  TechHeader.lef  UofU_Cells.tcl
[elb@lab2-20 Liberate]$
```

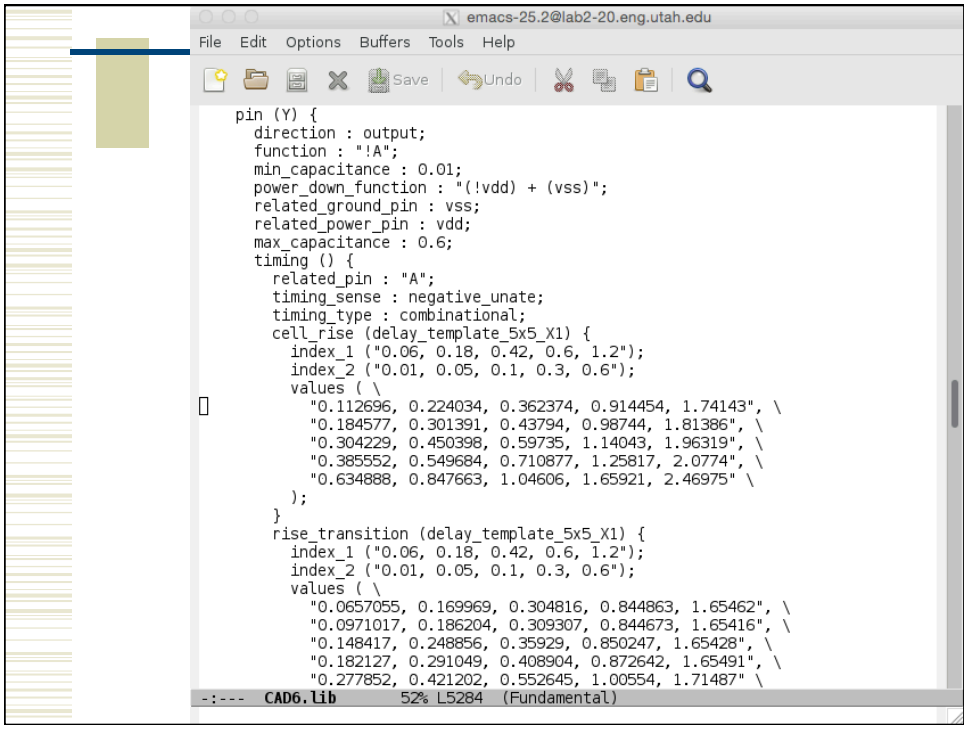
- Liberate tutorial on Canvas…



Changing library name in .lib




emacs-25.2@lab2-20.eng.utah.edu
File Edit Options Buffers Tools Help
Save Undo
cell (INVX1) {
area : 129.6;
cell_footprint : "INV";
cell_leakage_power : 0.0500074;
pg_pin (vdd) {
pg_type : primary_power;
voltage_name : "vdd";
}
pg_pin (vss) {
pg_type : primary_ground;
voltage_name : "vss";
}
leakage_power () {
value : 0.0500012;
when : "(A * !Y)";
related_pg_pin : vdd;
}
leakage_power () {
value : 0.0500137;
when : "(!A * Y)";
related_pg_pin : vdd;
}
leakage_power () {
value : 0.0500074;
related_pg_pin : vdd;
}
pin (Y) {
direction : output;
function : "!A";
min_capacitance : 0.01;
power_down_function : "(!vdd) + (vss)";
related_ground_pin : vss;
-:--- CAD6.lib 52% L5242 (Fundamental)


emacs-25.2@lab2-20.eng.utah.edu
File Edit Options Buffers Tools Help
Save Undo
pin (Y) {
direction : output;
function : "!A";
min_capacitance : 0.01;
power_down_function : "(!vdd) + (vss)";
related_ground_pin : vss;
related_power_pin : vdd;
max_capacitance : 0.6;
timing () {
related_pin : "A";
timing_sense : negative_unate;
timing_type : combinational;
cell_rise (delay_template_5x5_X1) {
index_1 ("0.06, 0.18, 0.42, 0.6, 1.2");
index_2 ("0.01, 0.05, 0.1, 0.3, 0.6");
values (\
"0.112696, 0.224034, 0.362374, 0.914454, 1.74143", \
"0.184577, 0.301391, 0.43794, 0.98744, 1.81386", \
"0.304229, 0.450398, 0.59735, 1.14043, 1.96319", \
"0.385552, 0.549684, 0.710877, 1.25817, 2.0774", \
"0.634888, 0.847663, 1.04606, 1.65921, 2.46975" \
);
}
rise_transition (delay_template_5x5_X1) {
index_1 ("0.06, 0.18, 0.42, 0.6, 1.2");
index_2 ("0.01, 0.05, 0.1, 0.3, 0.6");
values (\
"0.0657055, 0.169969, 0.304816, 0.844863, 1.65462", \
"0.0971017, 0.186204, 0.309307, 0.844673, 1.65416", \
"0.148417, 0.248856, 0.35929, 0.850247, 1.65428", \
"0.182127, 0.291049, 0.408904, 0.872642, 1.65491", \
"0.277852, 0.421202, 0.552645, 1.00554, 1.71487" \
-:--- CAD6.lib 52% L5284 (Fundamental)

syn-lc

Converting .lib to .db

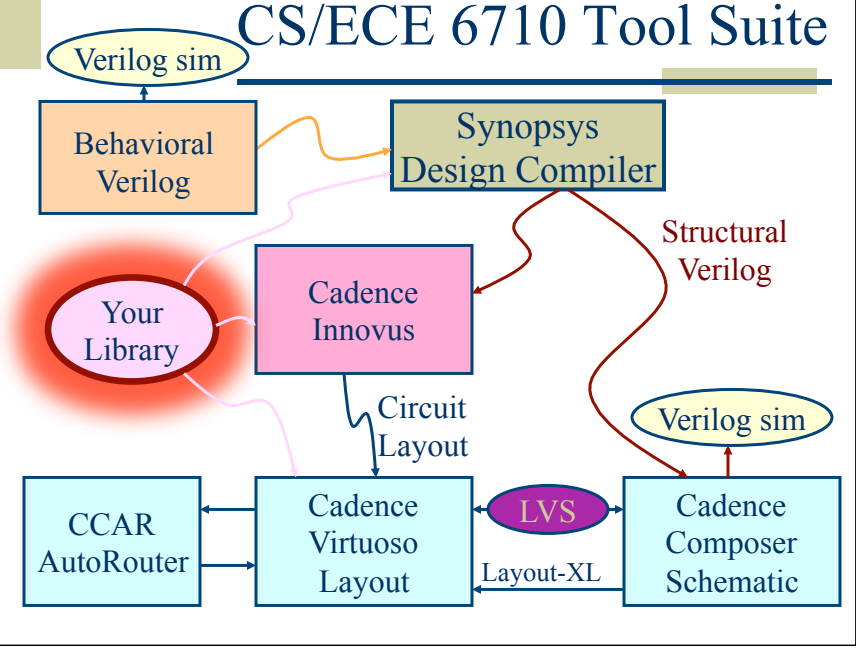
```
[elb@lab2-20 lib]$ syn-lc
Using setup-synopsys from S17
Assuming your OS is amd64
You are now set up to run the synopsys tools.

Working directory is /home/elb/VLSI/Liberate/lib

                                Library Compiler (TM)
                                DesignWare (R)
                Version M-2016.12-SP3 for linux64 - Apr 13, 2017
                        Copyright (c) 1988 - 2017 Synopsys, Inc.
    This software and the associated documentation are proprietary to Synopsys,  Inc.
This software may only be used in accordance with the terms and conditions
  of a written license agreement with Synopsys, Inc. All other use, reproduction,
                or distribution of this software is strictly prohibited.

Initializing...
lc_shell> read_lib Lib6710_XX.lib
Reading '/home/elb/VLSI/Liberate/lib/Lib6710_XX.lib' ...
Warning: Line 1, The 'default_inout_pin_cap' attribute is not specified. Using 1.00. (LBDB-172)
Warning: Line 1, The 'default_input_pin_cap' attribute is not specified. Using 1.00. (LBDB-172)
Warning: Line 1, The 'default_leakage_power_density' attribute is not specified. Using 0.00. (LBDB-172)
Warning: Line 105, Cell 'INVX1', The cell_leakage_power attribute of the 'INVX1' cell is redundant
        and not used in the leakage_power modeling. (LBDB-644)
Warning: Line 215, Cell 'INVX1', pin 'A', The pin 'A' does not have a internal_power group. (LBDB-607)
Warning: Line 229, Cell 'NAND2X1', The cell_leakage_power attribute of the 'NAND2X1' cell is redundant
        and not used in the leakage_power modeling. (LBDB-644) Technology library
'Lib6710_XX' read successfully
1
```

syn-lc

Converting .lib to .db

```
lc_shell> write_lib Lib6710_XX -o Lib6710_XX.db Wrote the
'Lib6710_XX' library to '/home/elb/VLSI/Liberate/lib/Lib6710_XX.db' successfully
1
lc_shell> exit Memory usage for this session 19 Mbytes.
CPU usage for this session 0 seconds ( 0.00 hours ).

Thank you...
[elb@lab2-20 lib]$
```

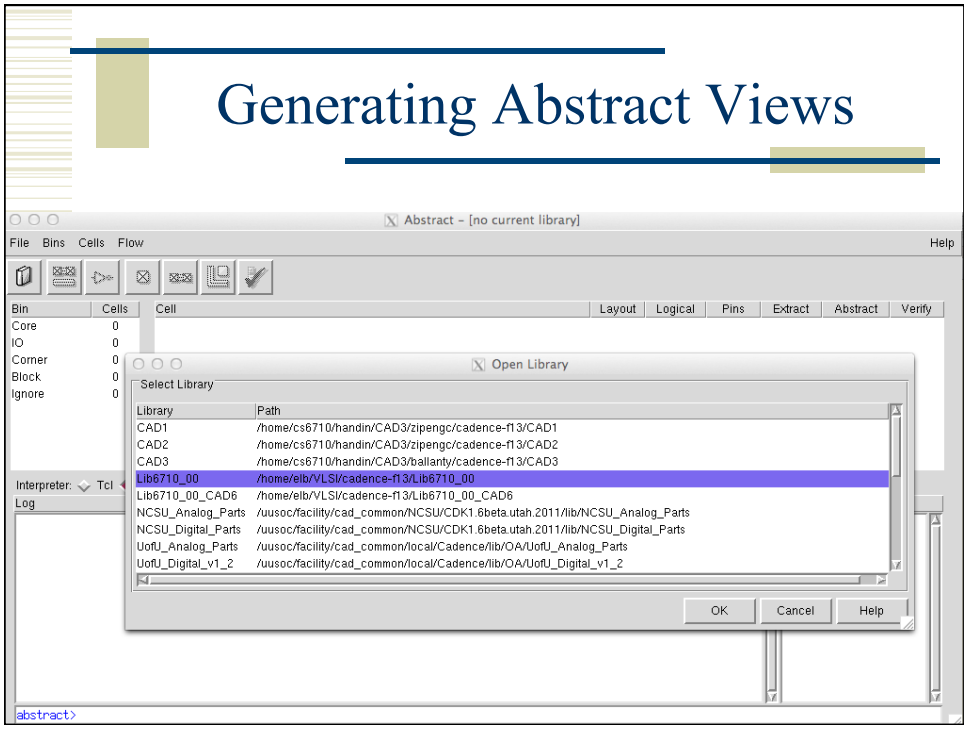
CS/ECE 6710 Tool Suite



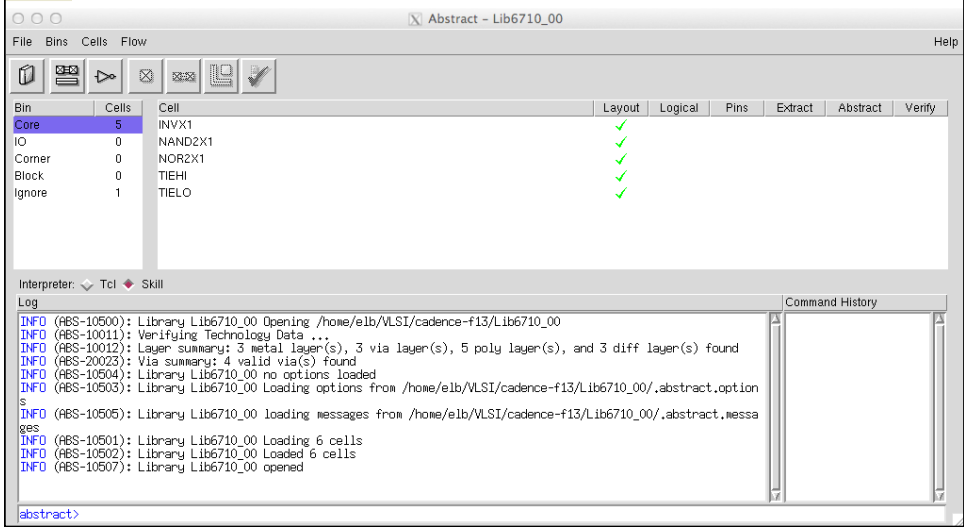
In the CAD Book

- Chapter 10 – Abstract Generation
  - Abstract views are used by place and route
  - Eventually they are captured in a .lef file that describes the place and route geometry

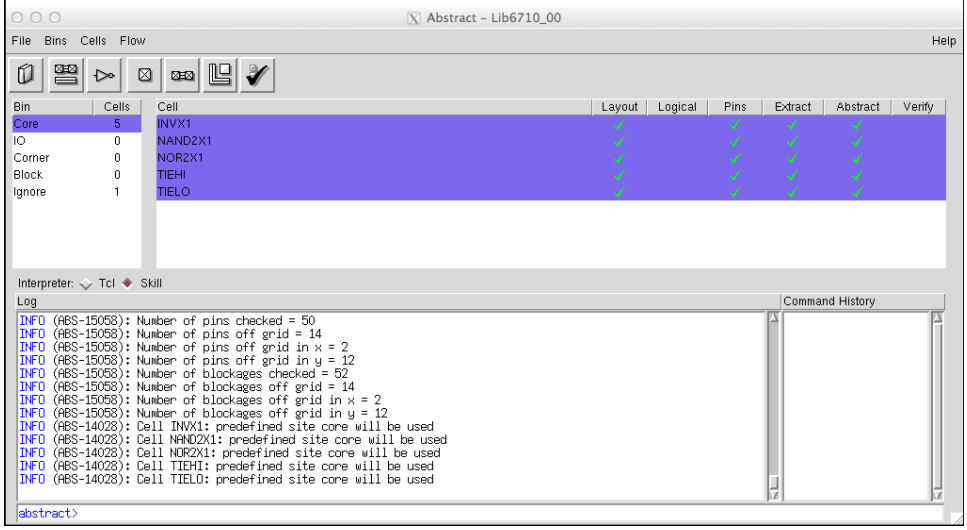
Generating Abstract Views



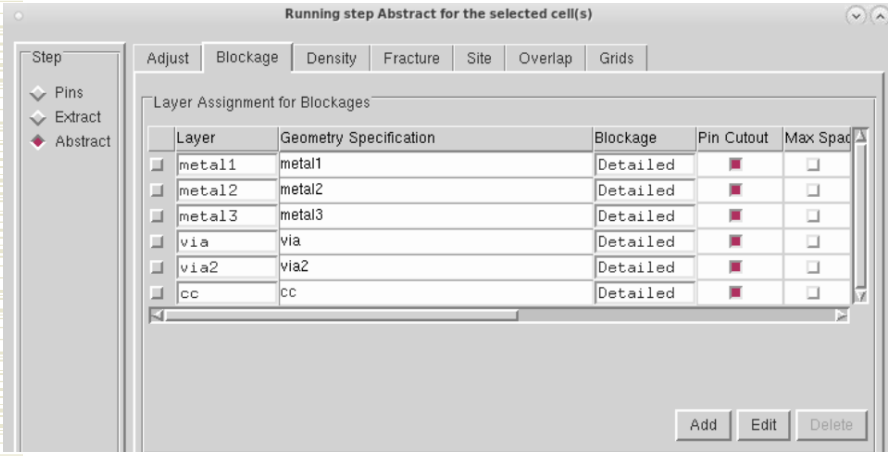
Imported library in cad-abstract



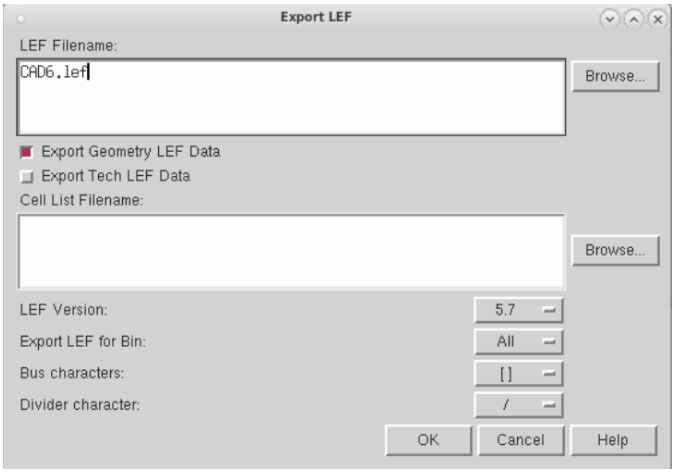
Green checks are good!



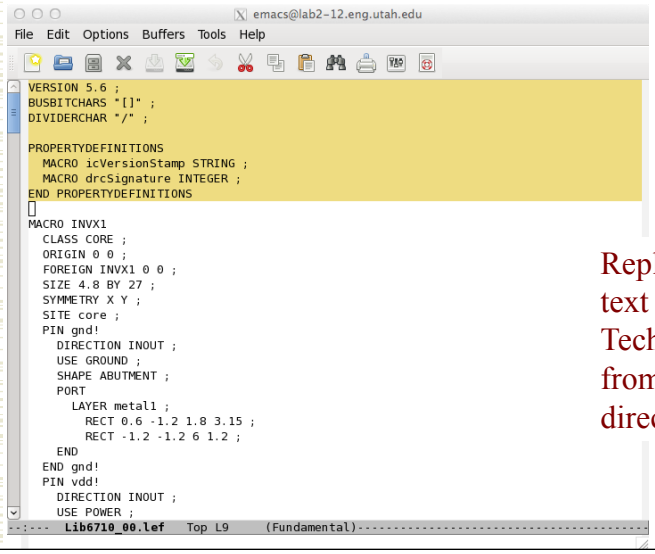
Abstract Generation



Export Lib6710_00.lef file

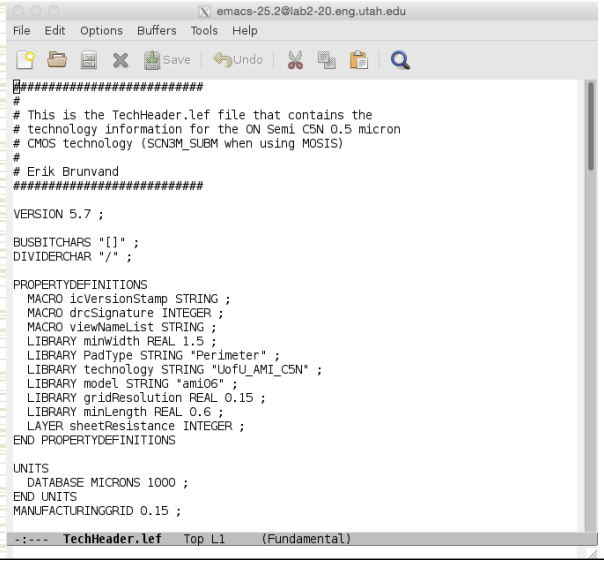


"geometry lef"



Replace highlighted text with TechHeader.lef from Liberate directory

"technology lef"

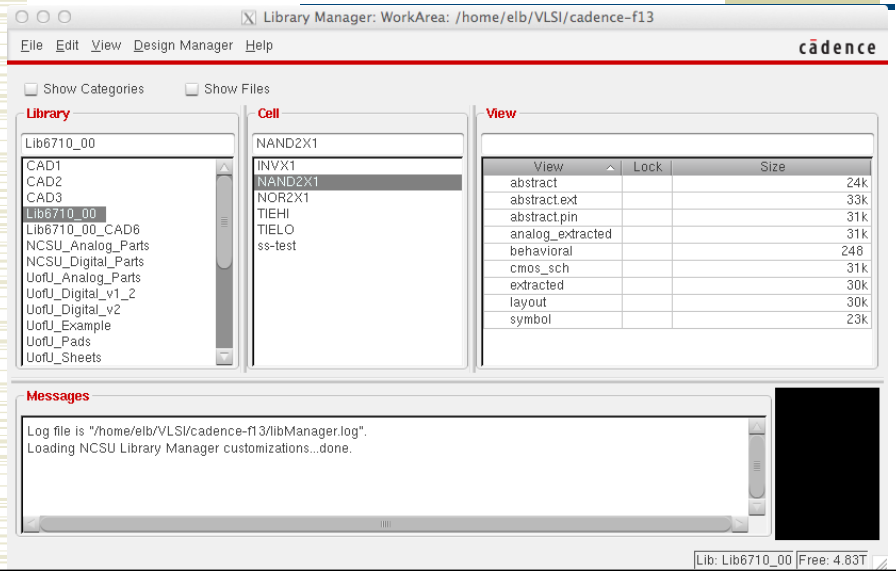


Replace highlighted text with TechHeader.lef from class directory

Final CAD5 Files…

- *Nine* views of *every* cell
  - abstract, abstract.ext, abstract.pin, analog_extracted, behavioral, cmos_sch, extracted, layout, symbol
  - DRC and LVS-checked, and simulated
- Three versions of the library description
  - Lib6710_00.lib // timing information
  - Lib6710_00.db // design compiler target lib
  - Lib6710_00.lef // place and route info

All Nine Views…



Test with beh2str

- `beh2str addsub.v addsub_dc.v Lib6710_00.db`
- **Results in addsub_dc.v and addsub_dc.v.rep**
- *NOTE! Your 5-cell library has no DFF, so you can't synthesize any sequential circuits!*
  - *Combinational circuits only at this point…*

addsub_dc.v

```
module addsub ( a, b, addnsub, result );
  input [7:0] a;
  input [7:0] b;
  output [8:0] result;
  input addnsub;
  wire   n26, n27, n28, n29, n30, n31, n32, n33, n34, n35, n36, n37, n38, n39,
         n40, n41, n42, n43, n44, n45, n46, n47, n48, n49, n50, n51, n52, n53,
         n54, n55, n56, n57, n58, n59, n60, n61, n62, n63, n64, n65, n66, n67,
...

NAND2X1 U33 ( .A(n26), .B(n27), .Y(result[8]) );
NAND2X1 U34 ( .A(b[7]), .B(n28), .Y(n27) );
NAND2X1 U35 ( .A(n29), .B(n30), .Y(n28) );
NOR2X1 U36 ( .A(n31), .B(n32), .Y(n29) );
...
NOR2X1 U216 ( .A(n188), .B(a[0]), .Y(n175) );
NAND2X1 U217 ( .A(a[0]), .B(n188), .Y(n202) );
INVX1 U218 ( .A(b[0]), .Y(n188) );
endmodule
```

addsub_dc.v.rep

```
Operating Conditions: typical   Library: Lib6710_00
Wire Load Model Mode: top
  Startpoint: b[1] (input port)
  Endpoint: result[8] (output port)
  Path Group: (none)
  Path Type: max
  Point                                    Incr       Path
  -----------------------------------------------------------
  input external delay                     0.00       0.00 r
  b[1] (in)                                0.00       0.00 r
  U198/Y (NOR2X1)                          0.50       0.50 f
  U194/Y (NOR2X1)                          0.36       0.86 r
...
  U34/Y (NAND2X1)                          0.25      10.16 f
  U33/Y (NAND2X1)                          0.24      10.40 r
  result[8] (out)                          0.00      10.40 r
  data arrival time                                  10.40
  -----------------------------------------------------------
  (Path is unconstrained)
```

addsub_dc.v.rep

```
Library(s) Used:

    Lib6710_00 (File: /home/elb/VLSI/cadence-f13/syn-f13/CAD5test/
Lib6710_00.db)

Number of ports:                          26
Number of nets:                          203
Number of cells:                         186
Number of combinational cells:           186
Number of sequential cells:                0
Number of macros:                          0
Number of buf/inv:                        37
Number of references:                      3
```

Summary

- You now have a library that is fully functional
  - BUT – only on combinational circuits
  - No DFF yet!
- Every step of the way requires extreme care to get things exactly right
  - No trick to finding the right answer
  - The point is to practice working with the data & tools

Look ahead to Innovus

```
[elb@lab2-12 addsub]$ ll -t
total 132
-rw-r--r-- 1 elb cs-rsrch  1522 Oct 10  2013 addsub.globals
-rw-r--r-- 1 elb cs-rsrch  3814 Oct 10  2013 top.tcl
-rw-r--r-- 1 elb cs-rsrch  2618 Oct 10  2013 mmmc.tcl
-rw-r--r-- 1 elb cs-rsrch   280 Oct 10 16:39 Lib6710_00.v
-rw-r--r-- 1 elb cs-rsrch 22736 Oct 10 16:39 Lib6710_00.lib
-rw-r--r-- 1 elb cs-rsrch 31744 Oct 10 16:39 Lib6710_00.db
-rw-r--r-- 1 elb cs-rsrch 11919 Oct 10 16:39 Lib6710_00.lef
-rw-r--r-- 1 elb cs-rsrch  3413 Oct 10 16:39 addsub_struct.sdc
-rw-r--r-- 1 elb cs-rsrch 11742 Oct 10 16:39 addsub_struct.v
-rw-r--r-- 1 elb cs-rsrch  2016 Oct 10 16:37 verify.tcl
-rw-r--r-- 1 elb cs-rsrch  3005 Oct 10 16:37 route.tcl
-rw-r--r-- 1 elb cs-rsrch  2394 Oct 10 16:37 pplan.tcl
-rw-r--r-- 1 elb cs-rsrch  1940 Oct 10 16:37 place.tcl
-rw-r--r-- 1 elb cs-rsrch   971 Oct 10 16:37 fplan.tcl
-rw-r--r-- 1 elb cs-rsrch  1641 Oct 10 16:37 cts.tcl
[elb@lab2-12 addsub]$
```

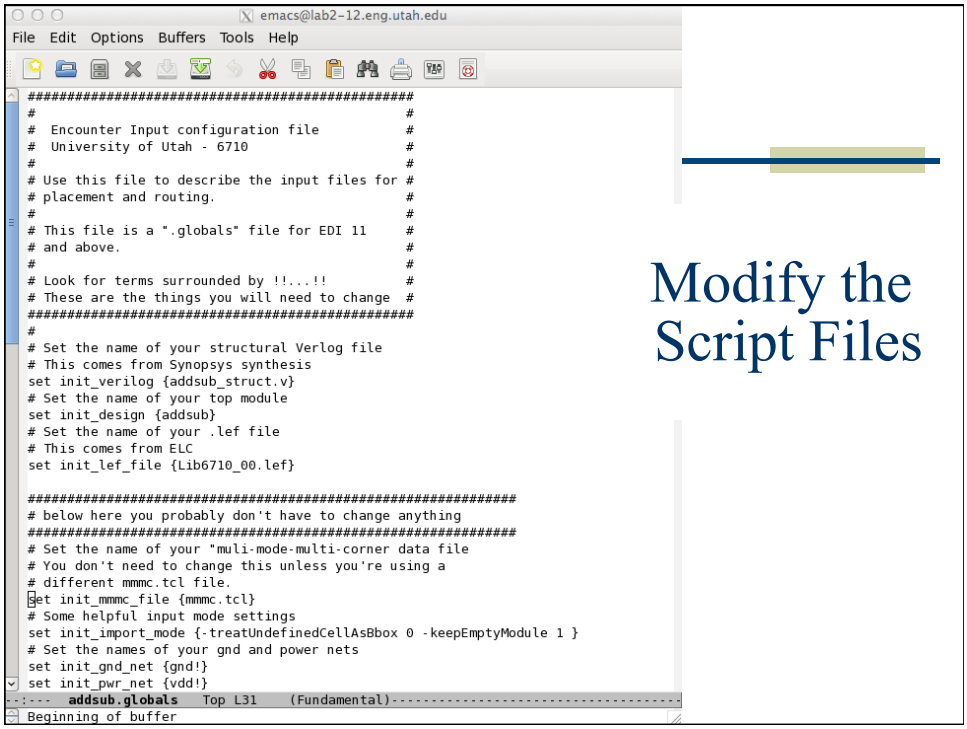
Modify the Script Files

emacs@lab2-12.eng.utah.edu

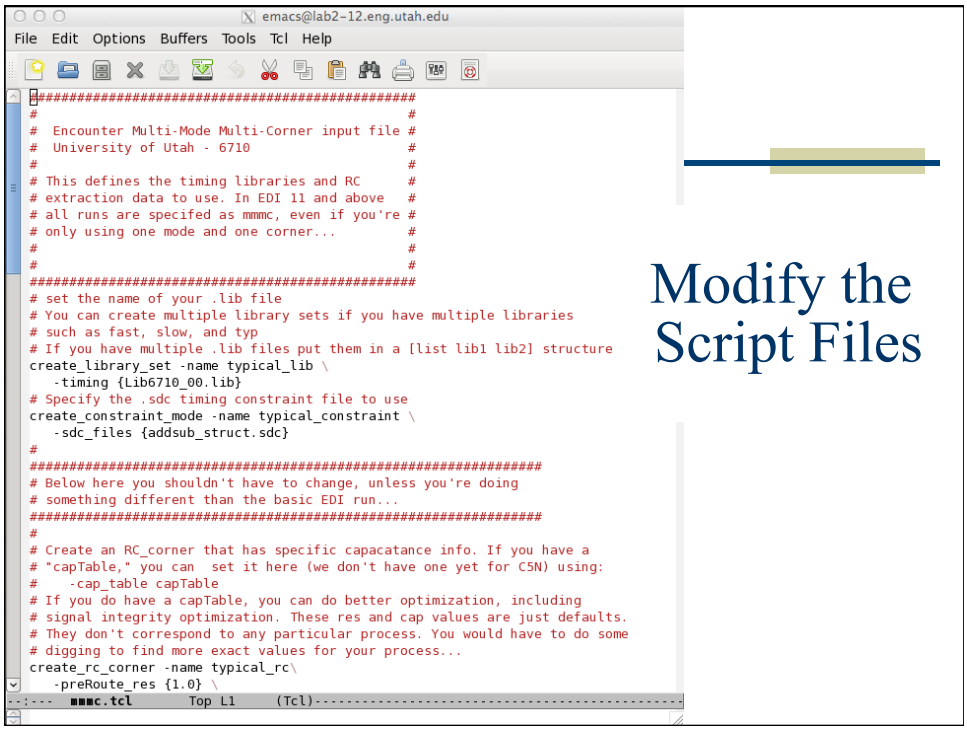
File Edit Options Buffers Tools Tcl Help

```
# configuration file should be named <BASENAME>.globals.
set BASENAME "addsub"

# The following variables are used in fplan.tcl...
#
# These set the percent utilization target (how dense should
# the cells be placed), and the gap for routing between rows.
# These are good starting values for small macros. Larger or
# more complex macros will likely need a lowered usepct or
# larger rowgap or both.
#
# Note that rowgap and coregap should be divisible by the basic
# grid unit of 0.3 that our AMIC5N/F process uses.
#
set usepct 0.70   ;# percent utilization in placing cells
set rowgap 15     ;# gap (microns) between pairs of std cell rows
set coregap 30.0  ;# gap (microns) between the core and the power rails
set aspect 0.60   ;# aspect ratio of overall cell (1 is square,
                   # <1 is landscape, >1 is portrait

# The following variables are used in pplan.tcl...
#
# These numbers control the power and ground grid.
# Note that all these numbers should be divisible by 0.3 so
# that they fit on the lambda grid
set pwidth 9.9     ;# power rail width
set pspace 1.8     ;# power rail space
set swidth 4.8     ;# power stripe width
set sspace 123     ;# power stripe spacing
set soffset 120    ;# power stripe offset to first stripe
set powernet vdd!  ;# the name of the power net
set groundnet gnd! ;# the name of the ground net
```

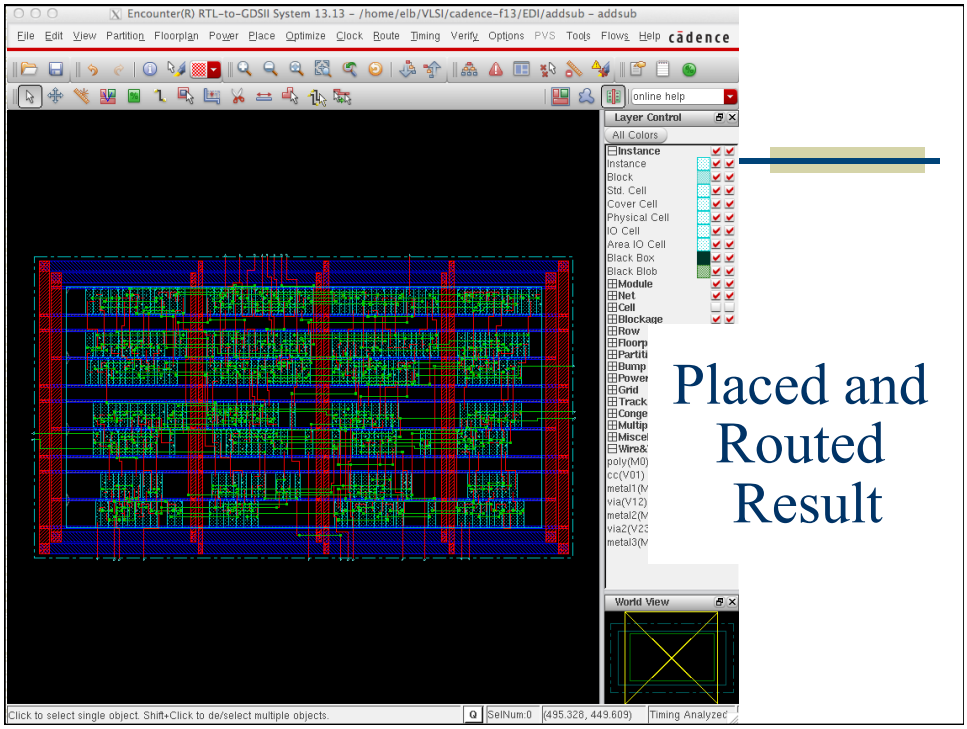
--:--- top.tcl 25% L39 (Tcl)----------



Modify the Script Files



Modify the Script Files



Placed and Routed Result