ECE 429 Laboratory 5: Hierarchical Design and Formal Verification

Alexander Lukens

Illinois Institute of Technology

October 24, 2020

Acknowledgement: I acknowledge all of the work (including figures and codes) belongs to me and/or persons who are referenced.
Signature: *Alexander Lukens*

1 Introduction

In this laboratory, students will be introduced to hierarchical design and verification techniques for CMOS circuits. Students will be required to implement a physical design layout for a 2-input NAND gate, and then use this layout with the inverter layout created in Lab 3 to make a 2-input AND gate. A new cell will be created in Cadence Virtuoso to hold the AND gate, and create a testing circuit to ensure correct functionality. A netlist of the physical layout cellview will be extracted, and the functionality will be compared to a Verilog file containing the behavioral description of an AND gate. Students will then calculate the rising and falling propagation delay of the AND gate using methods from previous laboratory sessions.

2 Theory\Pre-lab

In this lab, students will use hierarchical design to implement an AND gate. Hierarchical design refers to the design of a CMOS circuit using a combination of multiple cells. Furthermore, in hierarchical design, more complex cells are made from simplified cells. This allows some abstraction in a VLSI design, as the complex cells can be implemented instead of each single transistor. This drastically simplifies the design process of complex circuits, while retaining full detail retention of the components in the design.

In order for hierarchical design to be effective, a circuit verification and equivalence check must be utilized. One type of verification is the checking of a layout design functionality versus the corresponding schematic cellview. One tool that implements this verification is Calibre Layout Versus Schematic (LVS). This quickly ensures that the design is compliant with the port names and functionality implemented in the schematic cellview. Another verification technique is Calibre Design Rule Checking (DRC), which allows users to test their physical layout design for compliance with the design rules of a specific process technology library. This allows designers to ensure that their design will be producible on the specified process node.

A third type of verification is equivalence checking. Using equivalence checking, the CMOS design implemented in Cadence Virtuoso can be tested for functionality against a Verilog hardware description language file. This finally allows the designer to confirm that their design will carry out the intended function when implemented in hardware.

For the Prelab for Lab 5, students were required to sketch the stick diagram of a 2-input NAND gate, and a logic gate diagram for a 2-input AND gate. These diagrams are shown below.

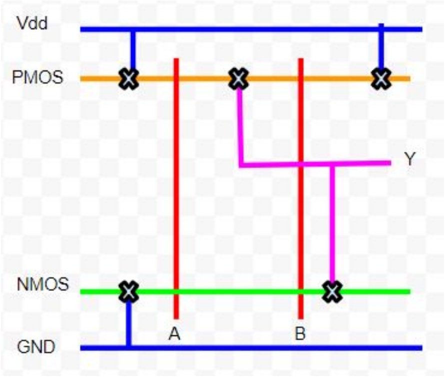


Figure 1: Stick Diagram for a 2-input NAND gate

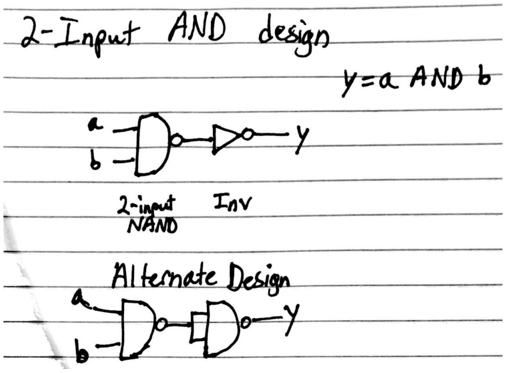


Figure 2: Logic Gate Diagram for a 2-input AND gate

3 Implementation

To create the 2-input AND gate, a physical layout must be created for the 2-input NAND gate. This involves first setting the schematic of the NAND gate to use transistors of length=50nm and width=90nm. Then, the layout cellview was created according to the design of the NAND gate stick diagram.

Using this diagram, the physical layout will be implemented as follows:

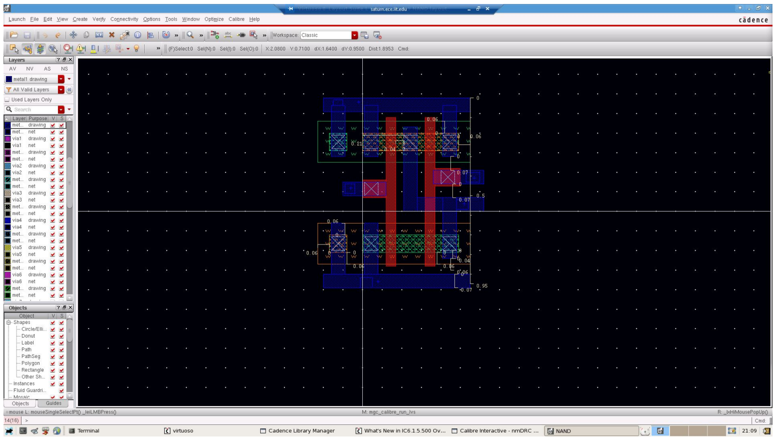
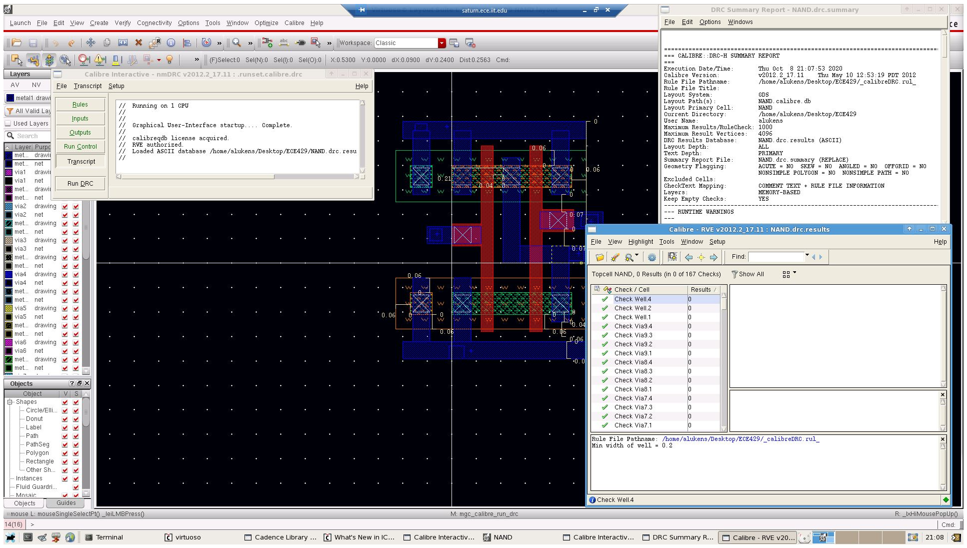

Launch File Edit View Create Verify Connectivity Options Tools Window Optimize Calibre Help
cadence
Workspace Classic
Layers
Objects
Terminal
virtuoso
Cadence Library Manager
NAND

Figure 3: Layout Cellview for 2-input NAND gate

This layout was then verified for compliance with the design rules using Calibre Design Rule Checking (DRC), and for functionality compared to the schematic cellview using Calibre Layout vs Schematic (LVS)


Calibre Interactive - nmDRC v2012.2_17.11 : .runset.calibre.drc
Rules
Inputs
Outputs
Run Control
Transcript
Run DRC
DRC Summary Report - NAND.drc.summary
Calibre - RVE v2012.2_17.11 : NAND.drc.results

```
//  Running on 1 CPU
//
//
//  Graphical User-Interface startup.... Complete.
//
//  calibreqdb license acquired.
//  RVE authorized.
//  Loaded ASCII database /home/alukens/Desktop/ECE429/NAND.drc.resu
//
```

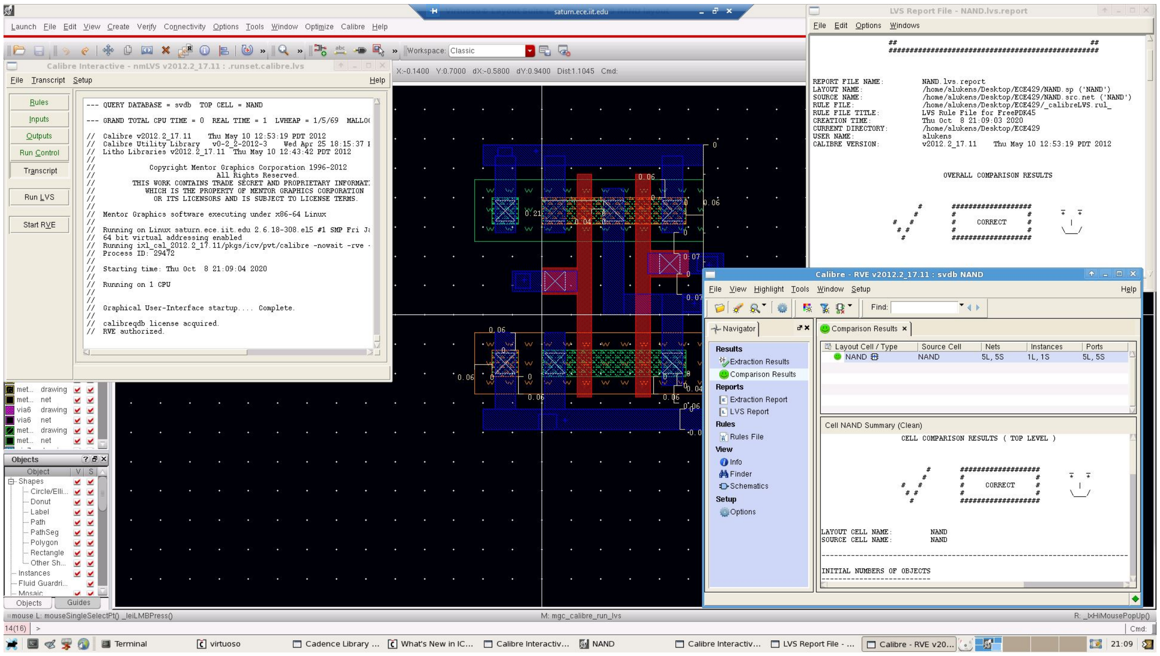
```
==================================================================================
=== CALIBRE::DRC-H SUMMARY REPORT
===
Execution Date/Time:        Thu Oct  8 21:07:53 2020
Calibre Version:            v2012.2_17.11    Thu May 10 12:53:19 PDT 2012
Rule File Pathname:         /home/alukens/Desktop/ECE429/_calibreDRC.rul_
Rule File Title:
Layout System:              GDS
Layout Path(s):             NAND.calibre.db
Layout Primary Cell:        NAND
Current Directory:          /home/alukens/Desktop/ECE429
User Name:                  alukens
Maximum Results/RuleCheck:  1000
Maximum Result Vertices:    4096
DRC Results Database:       NAND.drc.results (ASCII)
Layout Depth:               ALL
Text Depth:                 PRIMARY
Summary Report File:        NAND.drc.summary (REPLACE)
Geometry Flagging:          ACUTE = NO  SKEW = NO  ANGLED = NO  OFFGRID = NO
                            NONSIMPLE POLYGON = NO  NONSIMPLE PATH = NO
Excluded Cells:
CheckText Mapping:          COMMENT TEXT + RULE FILE INFORMATION
Layers:                     MEMORY-BASED
Keep Empty Checks:          YES
----------------------------------------------------------------------------------
--- RUNTIME WARNINGS
---
```

Topcell NAND, 0 Results (in 0 of 167 Checks)

Check / Cell	Results
✓ Check Well.4	0
✓ Check Well.2	0
✓ Check Well.1	0
✓ Check Via9.4	0
✓ Check Via9.3	0
✓ Check Via9.2	0
✓ Check Via9.1	0
✓ Check Via8.4	0
✓ Check Via8.3	0
✓ Check Via8.2	0
✓ Check Via8.1	0
✓ Check Via7.4	0
✓ Check Via7.3	0
✓ Check Via7.2	0
✓ Check Via7.1	0

```
Rule File Pathname: /home/alukens/Desktop/ECE429/_calibreDRC.rul_
Min width of well = 0.2
```

Figure 4: NAND DRC Result



```
--- QUERY DATABASE = svdb  TOP CELL = NAND

--- GRAND TOTAL CPU TIME = 0  REAL TIME = 1  LVHEAP = 1/5/69  MALLOC

//  Calibre v2012.2_17.11    Thu May 10 12:53:19 PDT 2012
//  Calibre Utility Library   v0-2_2-2012-3    Wed Apr 25 18:15:37 I
//  Litho Libraries v2012.2_17.11  Thu May 10 12:43:42 PDT 2012
//
//            Copyright Mentor Graphics Corporation 1996-2012
//                         All Rights Reserved.
//         THIS WORK CONTAINS TRADE SECRET AND PROPRIETARY INFORMATI
//            WHICH IS THE PROPERTY OF MENTOR GRAPHICS CORPORATION
//              OR ITS LICENSORS AND IS SUBJECT TO LICENSE TERMS.
//
//  Mentor Graphics software executing under x86-64 Linux
//
//  Running on Linux saturn.ece.iit.edu 2.6.18-308.el5 #1 SMP Fri Ja
//  64 bit virtual addressing enabled
//  Running ixl_cal_2012.2_17.11/pkgs/icv/pvt/calibre -nowait -rve -
//  Process ID: 29472
//
//  Starting time: Thu Oct  8 21:09:04 2020
//
//  Running on 1 CPU
//
//
//  Graphical User-Interface startup.... Complete.
//
//  calibreqdb license acquired.
//  RVE authorized.
```

```
                   ##                                                 ##
                   ####################################################



REPORT FILE NAME:          NAND.lvs.report
LAYOUT NAME:               /home/alukens/Desktop/ECE429/NAND.sp ('NAND')
SOURCE NAME:               /home/alukens/Desktop/ECE429/NAND.src.net ('NAND')
RULE FILE:                 /home/alukens/Desktop/ECE429/_calibreLVS.rul_
RULE FILE TITLE:           LVS Rule File for FreePDK45
CREATION TIME:             Thu Oct  8 21:09:03 2020
CURRENT DIRECTORY:         /home/alukens/Desktop/ECE429
USER NAME:                 alukens
CALIBRE VERSION:           v2012.2_17.11    Thu May 10 12:53:19 PDT 2012


                               OVERALL COMPARISON RESULTS



                         #        ###################       _   _
                        #         #                 #       *   *
                   #   #          #     CORRECT     #         |
                    # #           #                 #       \___/
                     #            ###################
```

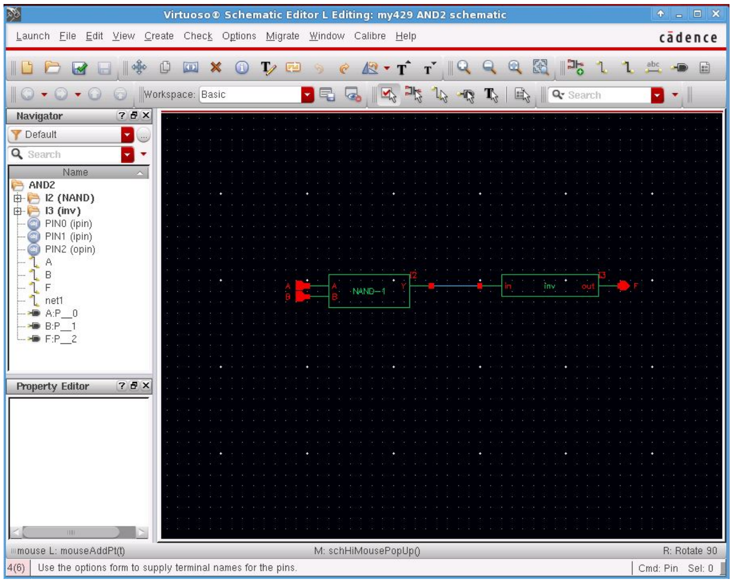
Layout Cell / Type	Source Cell	Nets	Instances	Ports
NAND	NAND	5L, 5S	1L, 1S	5L, 5S

Cell NAND Summary (Clean)

```
                    CELL COMPARISON RESULTS ( TOP LEVEL )



                         #        ###################       _   _
                        #         #                 #       *   *
                   #   #          #     CORRECT     #         |
                    # #           #                 #       \___/
                     #            ###################



LAYOUT CELL NAME:          NAND
SOURCE CELL NAME:          NAND

--------------------------------------------------------------------------

INITIAL NUMBERS OF OBJECTS
--------------------------
```

Figure 5: NAND LVS Result

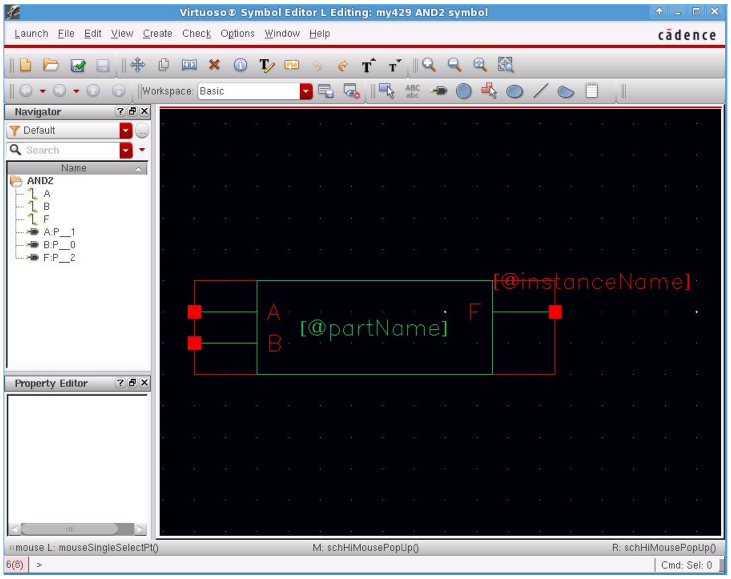
After successfully implementing the NAND layout cellview, a schematic cell was created for the 2-input AND gate. From this schematic, a symbol cellview was automatically generated



mouse L: mouseAddPt(t) M: schHiMousePopUp() R: Rotate 90

4(6) Use the options form to supply terminal names for the pins. Cmd: Pin Sel: 0

Figure 6: Schematic Cellview for 2-input AND gate



mouse L: mouseSingleSelectPt() M: schHiMousePopUp() R: schHiMousePopUp()

6(8) > Cmd: Sel: 0

Figure 7: Symbol Cellview for 2-input AND gate

The symbol cellview for the AND gate was used to create a testing circuit for this cell. This test circuit will be used to manipulate the input values to the AND gate to determine the rising and falling propagation delay of the gate. This will use similar methods to previous laboratories.

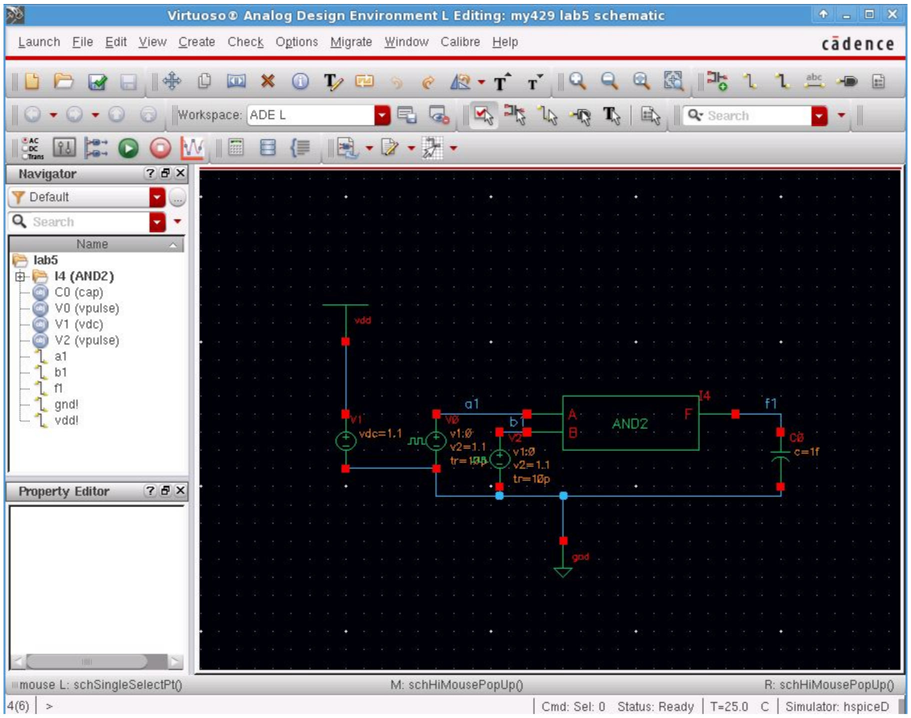


Figure 8: 2-input AND gate testing circuit

The next step in the implementation of the AND cell is to create a layout cellview that implements the physical design of the 2-input AND gate. This was completed by utilizing the 2-input NAND gate created previously, and then connecting the output of the NAND gate to an inverter, also created previously. The final layout is shown below.

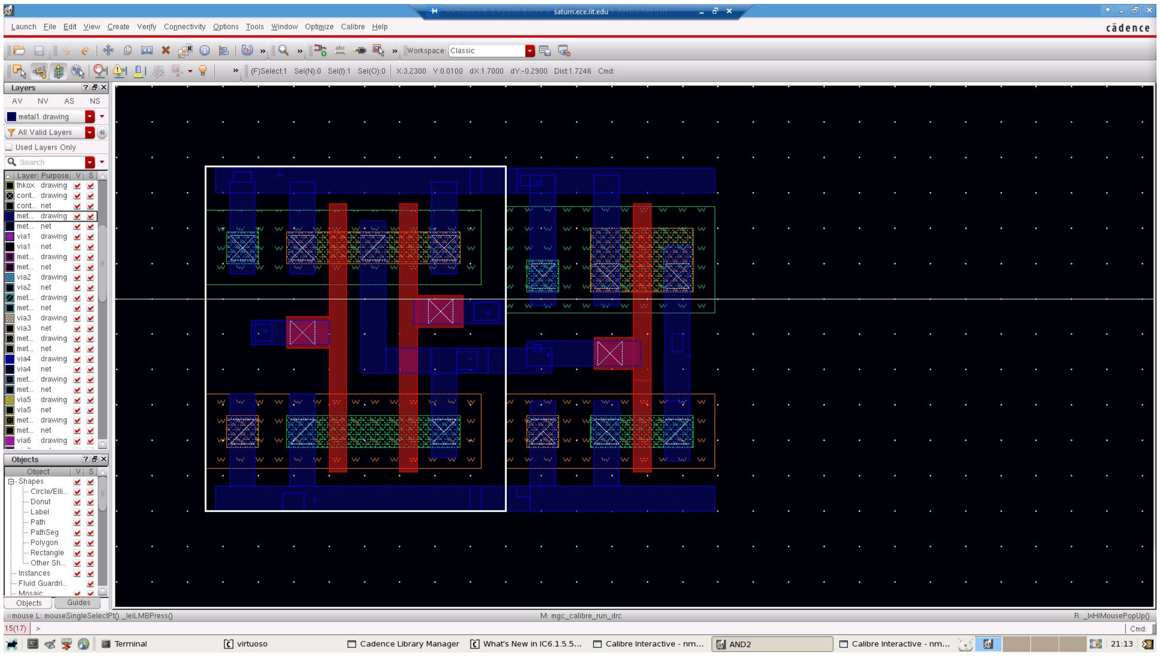
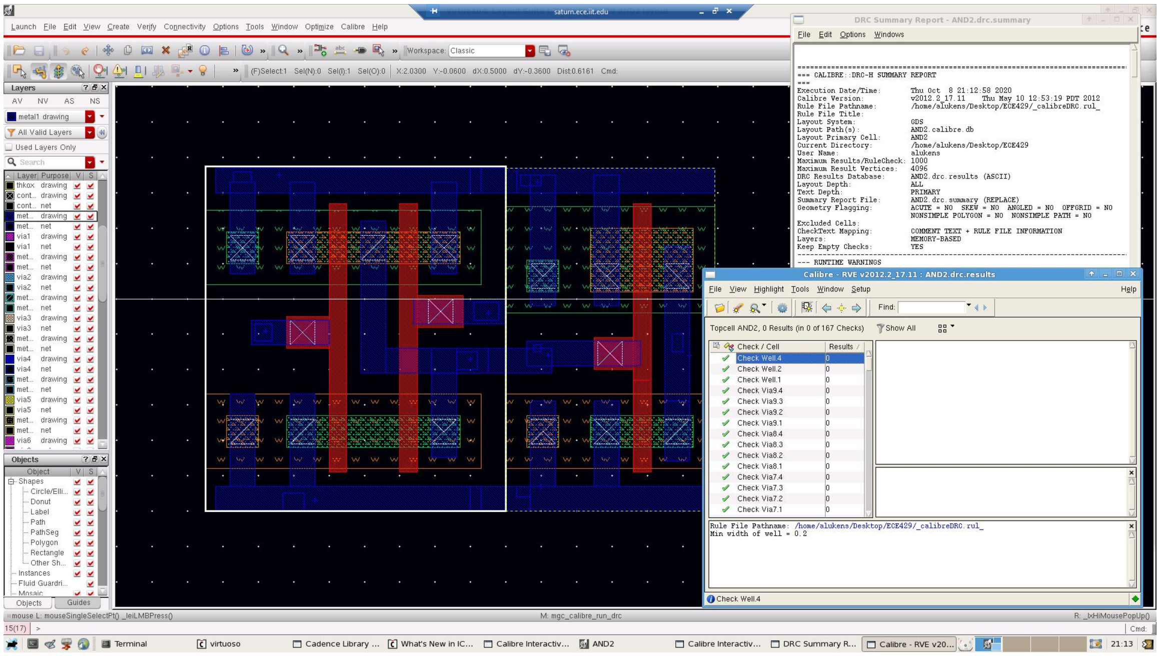


Figure 9: 2-input AND gate layout cellview

To ensure that this design could actually be implemented on the FreePDK45 process, the design was ran through Calibre DRC to enforce the FreePDK design rules on the layout. After compliance was ensured, the layout was tested against the schematic cellview to ensure functionality. The results of both of these tests are shown below.


DRC Summary Report - AND2.drc.summary
Calibre - RVE v2012.2_17.11 : AND2.drc.results

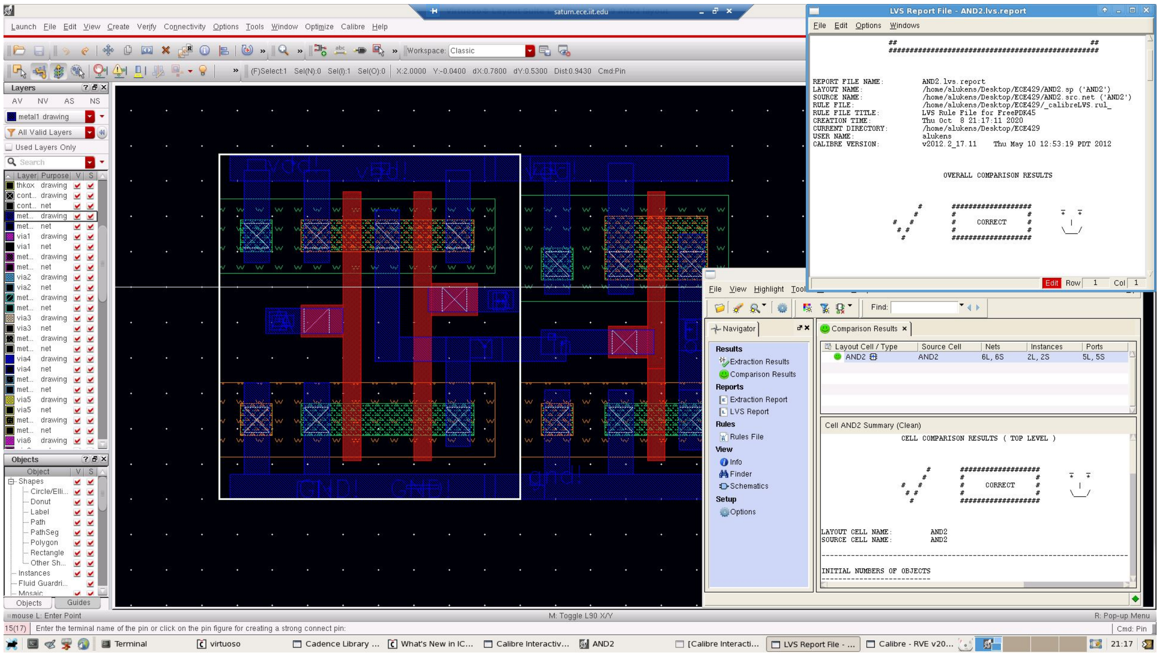
```
================================================================================
=== CALIBRE::DRC-H SUMMARY REPORT
===
Execution Date/Time:        Thu Oct  8 21:12:58 2020
Calibre Version:            v2012.2_17.11    Thu May 10 12:53:19 PDT 2012
Rule File Pathname:         /home/alukens/Desktop/ECE429/_calibreDRC.rul_
Rule File Title:
Layout System:              GDS
Layout Path(s):             AND2.calibre.db
Layout Primary Cell:        AND2
Current Directory:          /home/alukens/Desktop/ECE429
User Name:                  alukens
Maximum Results/RuleCheck:  1000
Maximum Result Vertices:    4096
DRC Results Database:       AND2.drc.results (ASCII)
Layout Depth:               ALL
Text Depth:                 PRIMARY
Summary Report File:        AND2.drc.summary (REPLACE)
Geometry Flagging:          ACUTE = NO  SKEW = NO  ANGLED = NO  OFFGRID = NO
                            NONSIMPLE POLYGON = NO  NONSIMPLE PATH = NO
Excluded Cells:
CheckText Mapping:          COMMENT TEXT + RULE FILE INFORMATION
Layers:                     MEMORY-BASED
Keep Empty Checks:          YES
--------------------------------------------------------------------------------
--- RUNTIME WARNINGS
```

Topcell AND2, 0 Results (in 0 of 167 Checks)

Check / Cell	Results
✓ Check Well.4	0
✓ Check Well.2	0
✓ Check Well.1	0
✓ Check Via9.4	0
✓ Check Via9.3	0
✓ Check Via9.2	0
✓ Check Via9.1	0
✓ Check Via8.4	0
✓ Check Via8.3	0
✓ Check Via8.2	0
✓ Check Via8.1	0
✓ Check Via7.4	0
✓ Check Via7.3	0
✓ Check Via7.2	0
✓ Check Via7.1	0

```
Rule File Pathname: /home/alukens/Desktop/ECE429/_calibreDRC.rul_
Min width of well = 0.2
```

Figure 10: 2-input AND gate Design Rule Checking (DRC) Results



```
                ##                                                ##
                ####################################################


REPORT FILE NAME:         AND2.lvs.report
LAYOUT NAME:              /home/alukens/Desktop/ECE429/AND2.sp ('AND2')
SOURCE NAME:              /home/alukens/Desktop/ECE429/AND2.src.net ('AND2')
RULE FILE:                /home/alukens/Desktop/ECE429/_calibreLVS.rul_
RULE FILE TITLE:          LVS Rule File for FreePDK45
CREATION TIME:            Thu Oct  8 21:17:11 2020
CURRENT DIRECTORY:        /home/alukens/Desktop/ECE429
USER NAME:                alukens
CALIBRE VERSION:          v2012.2_17.11    Thu May 10 12:53:19 PDT 2012


                          OVERALL COMPARISON RESULTS



                     #         ###################       _   _
                    #          #                 #       *   *
               #   #           #     CORRECT     #         |
                # #            #                 #       \___/
                 #             ###################
```

Comparison Results

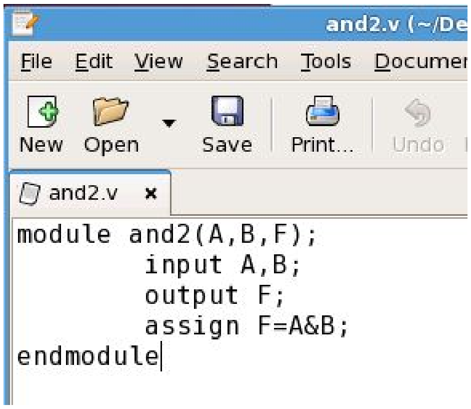
Layout Cell / Type	Source Cell	Nets	Instances	Ports
AND2	AND2	6L, 6S	2L, 2S	5L, 5S

Cell AND2 Summary (Clean)

```
                   CELL COMPARISON RESULTS ( TOP LEVEL )



                     #         ###################       _   _
                    #          #                 #       *   *
               #   #           #     CORRECT     #         |
                # #            #                 #       \___/
                 #             ###################


LAYOUT CELL NAME:         AND2
SOURCE CELL NAME:         AND2

--------------------------------------------------------------------------------

INITIAL NUMBERS OF OBJECTS
--------------------------
```

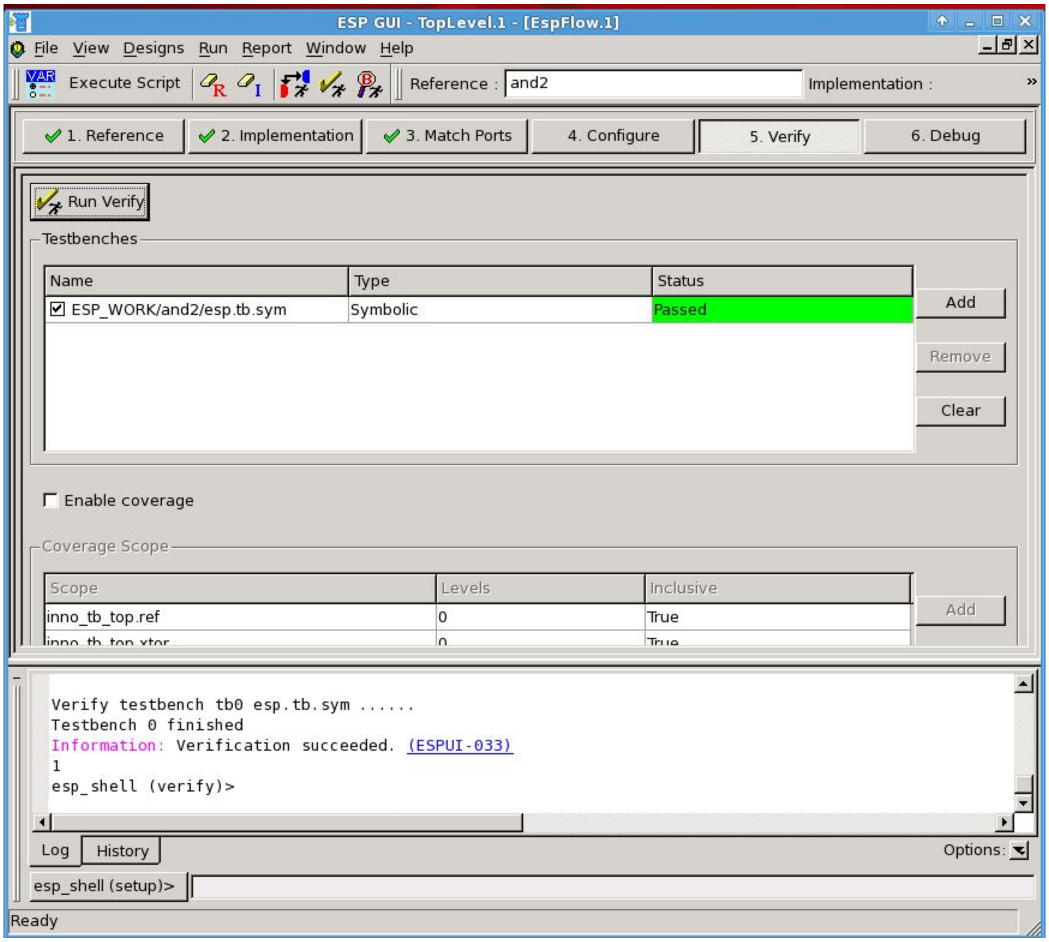
Figure 11: 2-input AND gate Layout vs Schematic (LVS) Results

After completing the LVS test, the functionality is ensured against the schematic cellview in Cadence Virtuoso. The next step is to ensure that the design actually works as intended. This can be done by comparing the functionality of the design in Virtuoso against a Verilog module that performs an AND operation. This is done using Cadence ESP, a tool used to check design equivalence.


and2.v (~/De
File Edit View Search Tools Documer
New Open Save Print... Undo
and2.v

```
module and2(A,B,F);
        input A,B;
        output F;
        assign F=A&B;
endmodule
```

Figure 12: AND gate design using Verilog



Name	Type	Status
☑ ESP_WORK/and2/esp.tb.sym	Symbolic	Passed

☐ Enable coverage

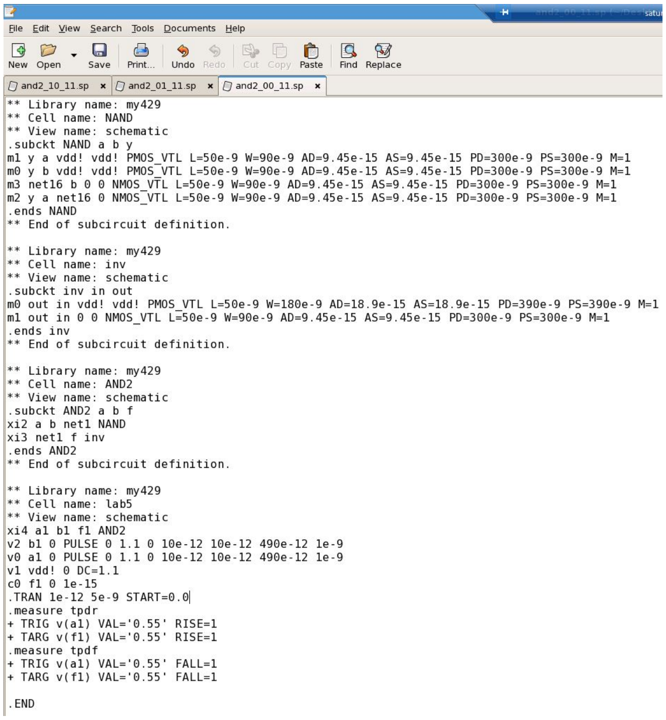
Scope	Levels	Inclusive
inno_tb_top.ref	0	True
inno_tb_top.xtor	0	True

```
Verify testbench tb0 esp.tb.sym ......
Testbench 0 finished
Information: Verification succeeded. (ESPUI-033)
1
esp_shell (verify)>
```

Figure 13: 2-input AND gate ESP Equivalence Results

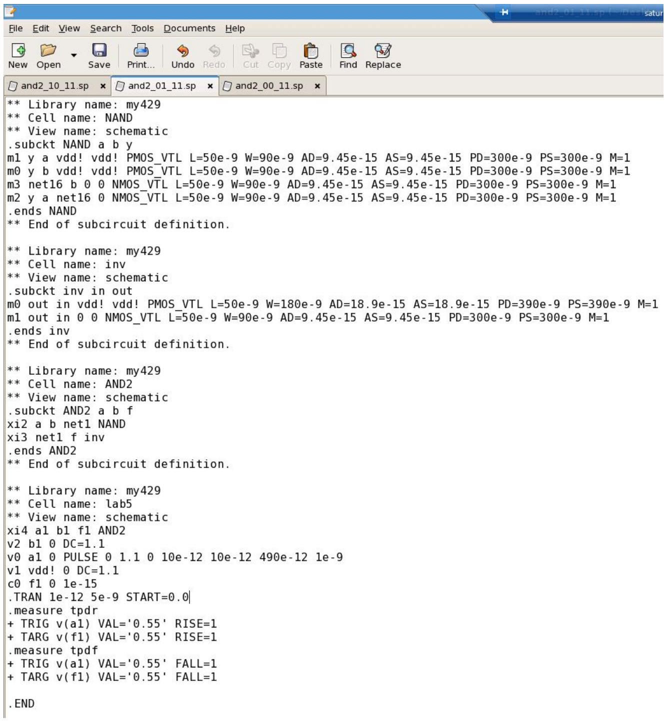
Now that the design functionality has been completely confirmed, the design can be tested for performance. This is done similarly to Laboratory 4, where several propagation delay tests are performed using

SPICE netlists that determine the rising and falling delay of the gate based on certain input values.



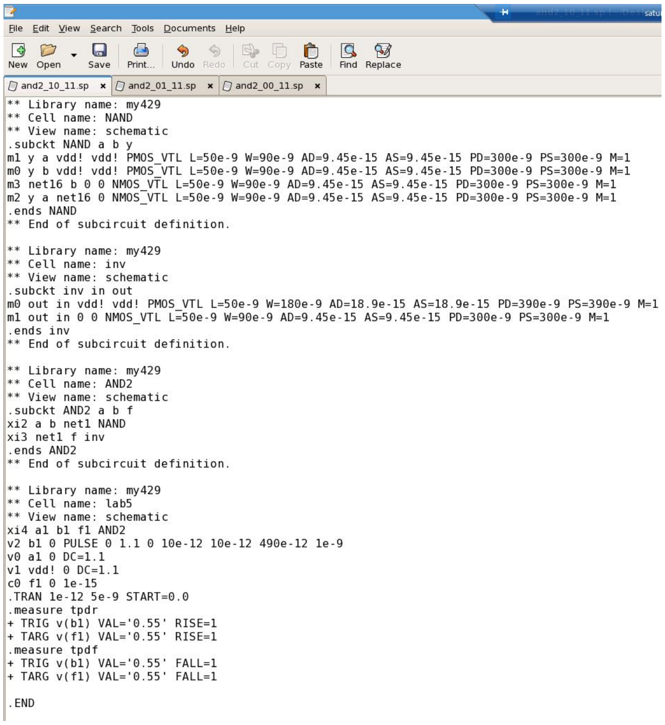
```
** Library name: my429
** Cell name: NAND
** View name: schematic
.subckt NAND a b y
m1 y a vdd! vdd! PMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
m0 y b vdd! vdd! PMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
m3 net16 b 0 0 NMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
m2 y a net16 0 NMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
.ends NAND
** End of subcircuit definition.

** Library name: my429
** Cell name: inv
** View name: schematic
.subckt inv in out
m0 out in vdd! vdd! PMOS_VTL L=50e-9 W=180e-9 AD=18.9e-15 AS=18.9e-15 PD=390e-9 PS=390e-9 M=1
m1 out in 0 0 NMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
.ends inv
** End of subcircuit definition.

** Library name: my429
** Cell name: AND2
** View name: schematic
.subckt AND2 a b f
xi2 a b net1 NAND
xi3 net1 f inv
.ends AND2
** End of subcircuit definition.

** Library name: my429
** Cell name: lab5
** View name: schematic
xi4 a1 b1 f1 AND2
v2 b1 0 PULSE 0 1.1 0 10e-12 10e-12 490e-12 1e-9
v0 a1 0 PULSE 0 1.1 0 10e-12 10e-12 490e-12 1e-9
v1 vdd! 0 DC=1.1
c0 f1 0 1e-15
.TRAN 1e-12 5e-9 START=0.0
.measure tpdr
+ TRIG v(a1) VAL='0.55' RISE=1
+ TARG v(f1) VAL='0.55' RISE=1
.measure tpdf
+ TRIG v(a1) VAL='0.55' FALL=1
+ TARG v(f1) VAL='0.55' FALL=1

.END
```

Figure 14: Netlist for 00→11 transition



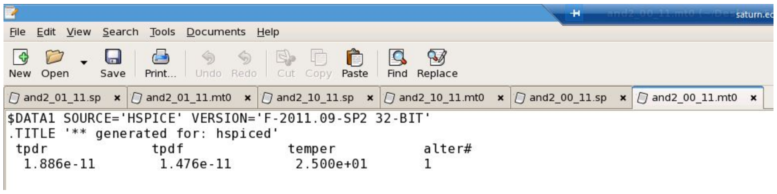
```
** Library name: my429
** Cell name: NAND
** View name: schematic
.subckt NAND a b y
m1 y a vdd! vdd! PMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
m0 y b vdd! vdd! PMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
m3 net16 b 0 0 NMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
m2 y a net16 0 NMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
.ends NAND
** End of subcircuit definition.

** Library name: my429
** Cell name: inv
** View name: schematic
.subckt inv in out
m0 out in vdd! vdd! PMOS_VTL L=50e-9 W=180e-9 AD=18.9e-15 AS=18.9e-15 PD=390e-9 PS=390e-9 M=1
m1 out in 0 0 NMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
.ends inv
** End of subcircuit definition.

** Library name: my429
** Cell name: AND2
** View name: schematic
.subckt AND2 a b f
xi2 a b net1 NAND
xi3 net1 f inv
.ends AND2
** End of subcircuit definition.

** Library name: my429
** Cell name: lab5
** View name: schematic
xi4 a1 b1 f1 AND2
v2 b1 0 DC=1.1
v0 a1 0 PULSE 0 1.1 0 10e-12 10e-12 490e-12 1e-9
v1 vdd! 0 DC=1.1
c0 f1 0 1e-15
.TRAN 1e-12 5e-9 START=0.0
.measure tpdr
+ TRIG v(a1) VAL='0.55' RISE=1
+ TARG v(f1) VAL='0.55' RISE=1
.measure tpdf
+ TRIG v(a1) VAL='0.55' FALL=1
+ TARG v(f1) VAL='0.55' FALL=1

.END
```

Figure 15: Netlist for 01→11 transition


File Edit View Search Tools Documents Help
New Open Save Print... Undo Redo Cut Copy Paste Find Replace
and2_10_11.sp
and2_01_11.sp
and2_00_11.sp

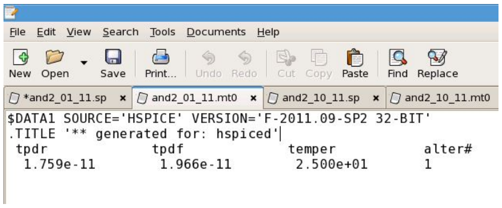
```
** Library name: my429
** Cell name: NAND
** View name: schematic
.subckt NAND a b y
m1 y a vdd! vdd! PMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
m0 y b vdd! vdd! PMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
m3 net16 b 0 0 NMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
m2 y a net16 0 NMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
.ends NAND
** End of subcircuit definition.

** Library name: my429
** Cell name: inv
** View name: schematic
.subckt inv in out
m0 out in vdd! vdd! PMOS_VTL L=50e-9 W=180e-9 AD=18.9e-15 AS=18.9e-15 PD=390e-9 PS=390e-9 M=1
m1 out in 0 0 NMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
.ends inv
** End of subcircuit definition.

** Library name: my429
** Cell name: AND2
** View name: schematic
.subckt AND2 a b f
xi2 a b net1 NAND
xi3 net1 f inv
.ends AND2
** End of subcircuit definition.

** Library name: my429
** Cell name: lab5
** View name: schematic
xi4 a1 b1 f1 AND2
v2 b1 0 PULSE 0 1.1 0 10e-12 10e-12 490e-12 1e-9
v0 a1 0 DC=1.1
v1 vdd! 0 DC=1.1
c0 f1 0 1e-15
.TRAN 1e-12 5e-9 START=0.0
.measure tpdr
+ TRIG v(b1) VAL='0.55' RISE=1
+ TARG v(f1) VAL='0.55' RISE=1
.measure tpdf
+ TRIG v(b1) VAL='0.55' FALL=1
+ TARG v(f1) VAL='0.55' FALL=1

.END
```

Figure 16: Netlist for 10→11 transition


File Edit View Search Tools Documents Help
New Open Save Print... Undo Redo Cut Copy Paste Find Replace
and2_01_11.sp
and2_01_11.mt0
and2_10_11.sp
and2_10_11.mt0
and2_00_11.sp
and2_00_11.mt0

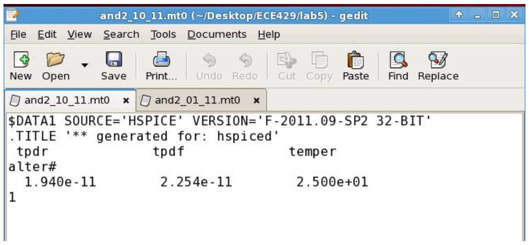
```
$DATA1 SOURCE='HSPICE' VERSION='F-2011.09-SP2 32-BIT'
.TITLE '** generated for: hspiced'
 tpdr             tpdf             temper           alter#
  1.886e-11        1.476e-11        2.500e+01       1
```

Figure 17: MT0 for 00→11 transition


File Edit View Search Tools Documents Help
New Open Save Print... Undo Redo Cut Copy Paste Find Replace
*and2_01_11.sp
and2_01_11.mt0
and2_10_11.sp
and2_10_11.mt0

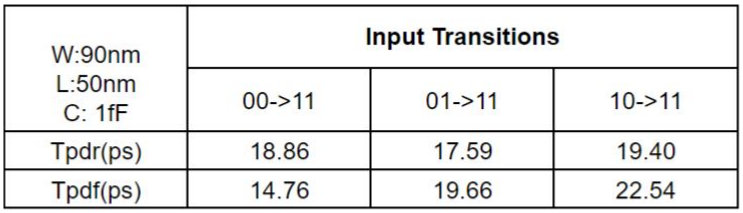
```
$DATA1 SOURCE='HSPICE' VERSION='F-2011.09-SP2 32-BIT'
.TITLE '** generated for: hspiced'
 tpdr             tpdf             temper           alter#
  1.759e-11        1.966e-11        2.500e+01       1
```

Figure 18: MT0 for 01→11 transition


and2_10_11.mt0 (~/Desktop/ECE429/lab5) - gedit
File Edit View Search Tools Documents Help
New Open Save Print... Undo Redo Cut Copy Paste Find Replace
and2_10_11.mt0
and2_01_11.mt0

```
$DATA1 SOURCE='HSPICE' VERSION='F-2011.09-SP2 32-BIT'
.TITLE '** generated for: hspiced'
 tpdr             tpdf             temper
alter#
  1.940e-11        2.254e-11        2.500e+01
1
```

Figure 19: MT0 for 10→11 transition



W:90nm L:50nm C: 1fF	**Input Transitions**		
	00->11	01->11	10->11
Tpdr(ps)	18.86	17.59	19.40
Tpdf(ps)	14.76	19.66	22.54

Figure 20: Propagation delay results

4 Deliverables

- ***Is there any other way to design an AND gate than combining an inverter with a NAND gate? Which one will you prefer?***

  Yes, an AND gate could also be created by using two NAND gates. The first NAND gates would take in the inputs A and B, and the second NAND gate would act as an inverter by attaching both inputs to the output of the first NAND gate. This design would be preferable over the NAND and inverter design, because multiples of a given gate (all NAND design) is easier to produce and implement over a design implementing multiple types of gates

- ***Do the input transitions leading to the propagation delays match your expectations? Why or why not?***

  Yes, the propagation delays meet my expections. All delays were determined to be larger than the NAND gate and Inverter gates used in previous labs. This makes sense, because the gates are connected in series. Specifically, it makes sense that the 10→11 transition will take the longest time as this will require charging of the most capacitance based on the NAND layout created previously (see figure 3).

- ***Is there any relationship between the propagation delays you measured in this lab for the AND gate and those you measured in Lab 2 for the inverter and in Lab 4 for the NAND gate?***

  The propagation delays measured in this laboratory is the sum of the propagation delays of the individual NAND and Inverter circuits. This makes sense, as in the design of the AND gate, both circuits were implemented in series, and the Inverter only reacts to the output of the NAND gate after that output is changed.

- ***What is the difference between a schematic and a Verilog model?***

  When designing a schematic, the designer has complete control over all parts of the schematic design. This includes individual transistor placement, method of implementation, etc. In a verilog model, all of these parameters are ”infered” by the compiler, such that the specifics of the implementation are not known to the designer. In this way, the schematic gives more specific control over the implementation of a circuit, and allows for more robust simulation (timing, power, etc) than the Verilog model.

5 Conclusions

To conclude, this laboratory session should be considered a success. Students were successfully able to design and implement a 2-input AND gate in Cadence Virtuoso. Students were able to create a schematic, layout, symbol, and testing circuit for the AND gate. Students tested the layout for functionality against the schematic cellview, and then ensured compliance with the design rules for the FreePDK45 process library. Then, students used a netlist produced by Cadence Virtuoso to ensure functionality of the design versus a Verilog file performing a logical AND operation using equivalence checking software.

Students will use the skills obtained in this laboratory to design and implement more complex CMOS circuits in future laboratory sessions. Furthermore, students will be able to ensure design compliance and functionality at all stages of the digital logic design process. Students will use the hierarchical design process implemented in this laboratory session to provide abstraction for increasingly complex logic cells, allowing for more efficient design in Cadence Virtuoso.

6 Resources

- Choi, Ken. “ECE 429 Laboratory 5 Manual.” Illinois Institute of Technology, October 24, 2020.
- Kim, Victoria. “ECE 429 Guideline for Writing Report & Grading Criteria.” Illinois Institute of Technology, October 24, 2020.