|  |  |  |  |
| --- | --- | --- | --- |
| Lab2 | | | |
| Project name |  | | |
| Document ref |  | | |
| Version |  | | |
| Release date |  | | |
| Author | Wu Fei | | |
| Classification |  | | |
| Distribution List |  | | |
| Approved by | Name | Signature | Date |
|  |  |  |  |



**Ming Hsieh Department of Electrical Engineering**

**University of Southern California, Los Angeles, CA 90089**

**Spring 2022**

**EE 477**

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | List of changes | Author + Signature |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

C A T A L O G

[1 Motivation 1](#_Toc96377647)

[2 XOR DESIGN 2](#_Toc96377648)

[3 1\_bit full adder design 7](#_Toc96377649)

[3.1 Draw a 1b full adder schematic as shown. 7](#_Toc96377650)

[3.2 Design a delay measurement circuit 10](#_Toc96377651)

[3.2.1 (A, B, Cin) = (0,1,0) -> (0,1,1). This gives τPLH for Cout and τPHL for Sum. 11](#_Toc96377652)

[3.2.2 (A, B, Cin) = (0,1,1) -> (0,1,0). This gives τPHL for Cout and τPLH for Sum. 11](#_Toc96377653)

[3.3 Delay test 11](#_Toc96377654)

[3.3.1 P = NAND2\_1X, Q = NAND2\_1X 13](#_Toc96377655)

[3.3.2 P = NAND2\_4X, Q = NAND2\_1X 14](#_Toc96377656)

[3.3.3 P = NAND2\_4X, Q = NAND2\_4X 15](#_Toc96377657)

[3.4 Layout 15](#_Toc96377658)

[4 4\_bit ripper carry adder 20](#_Toc96377659)

[4.1.1 Test the functionality 20](#_Toc96377660)

[4.1.2 Worst part 21](#_Toc96377661)

[4.1.3 Layout 24](#_Toc96377662)

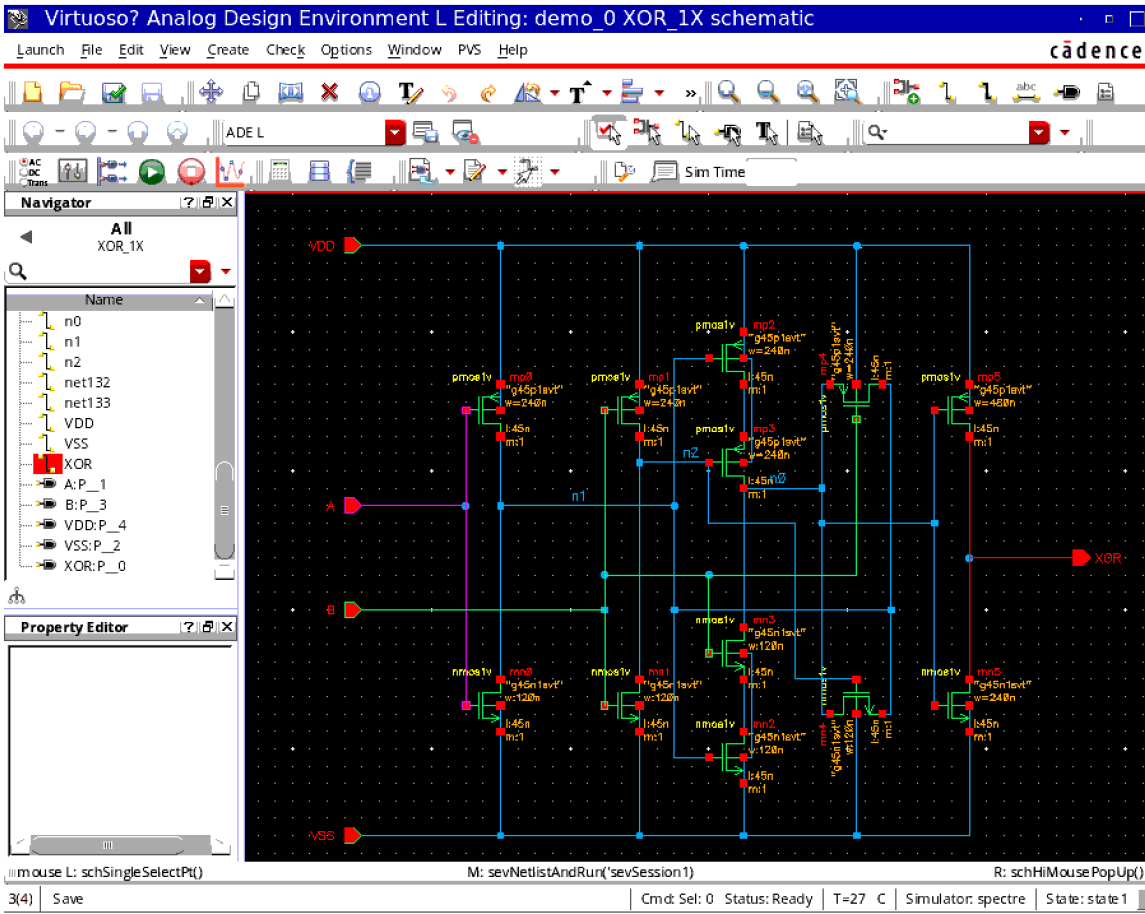
# Motivation

[Motivation](https://blackboard.usc.edu/webapps/blackboard/content/listContent.jsp?course_id=_294497_1&content_id=_8502445_1)

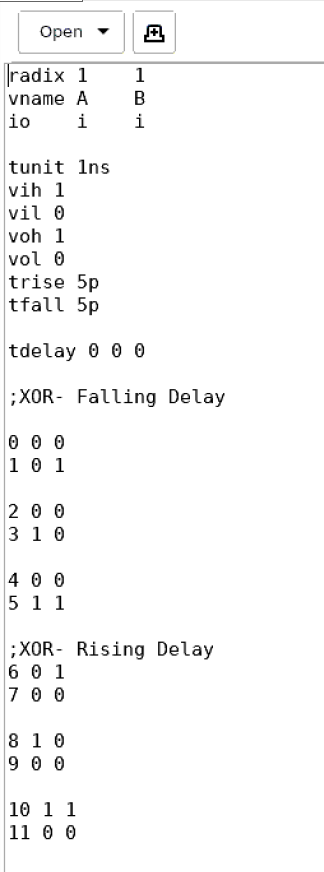
Design an adder as a practice of circuit design. This adder and/or the design ideas developed in this assignment would be applicable to the final project.

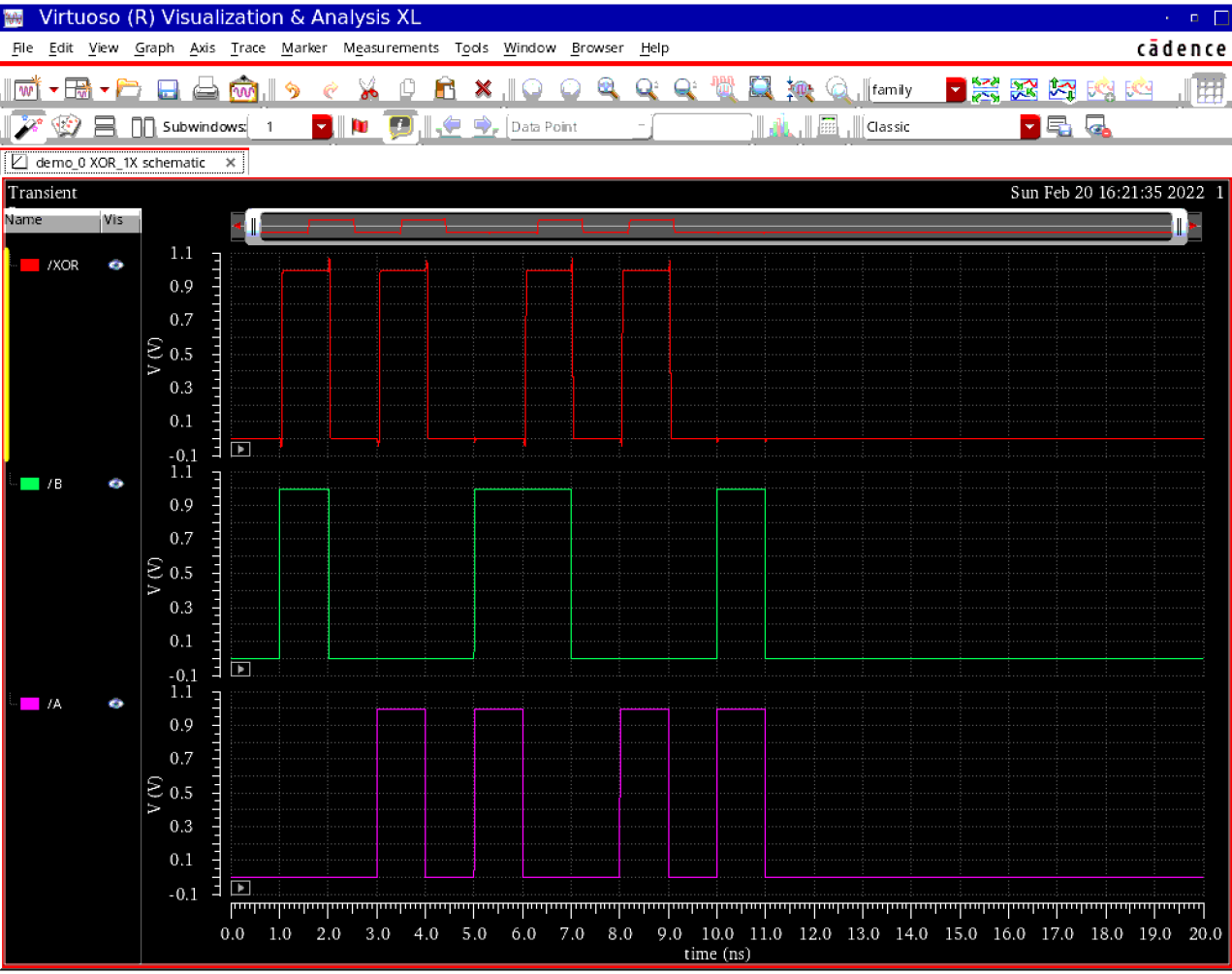
# XOR DESIGN

Design the schematic for a 2-input XOR gate using any method you like.

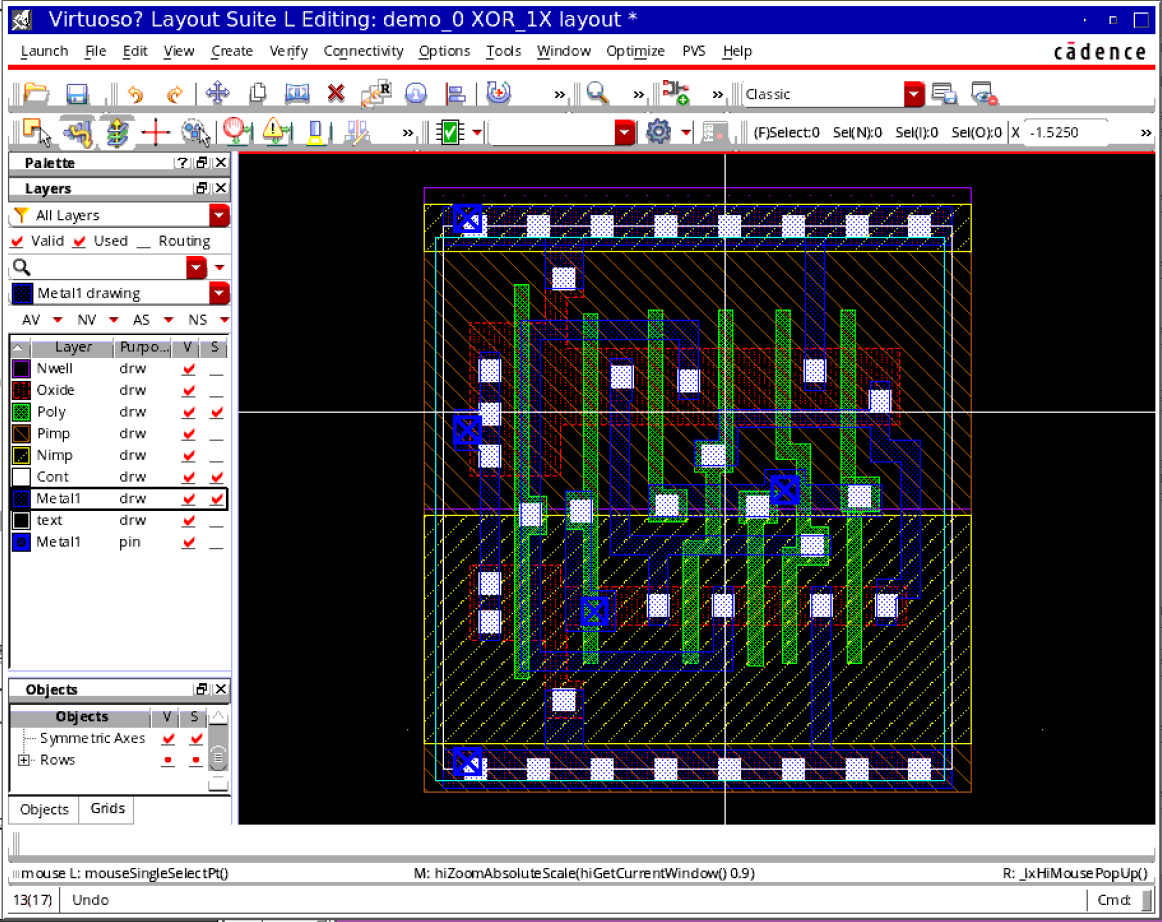


Test functionality for all possible input combinations: 00, 01, 10, 11.

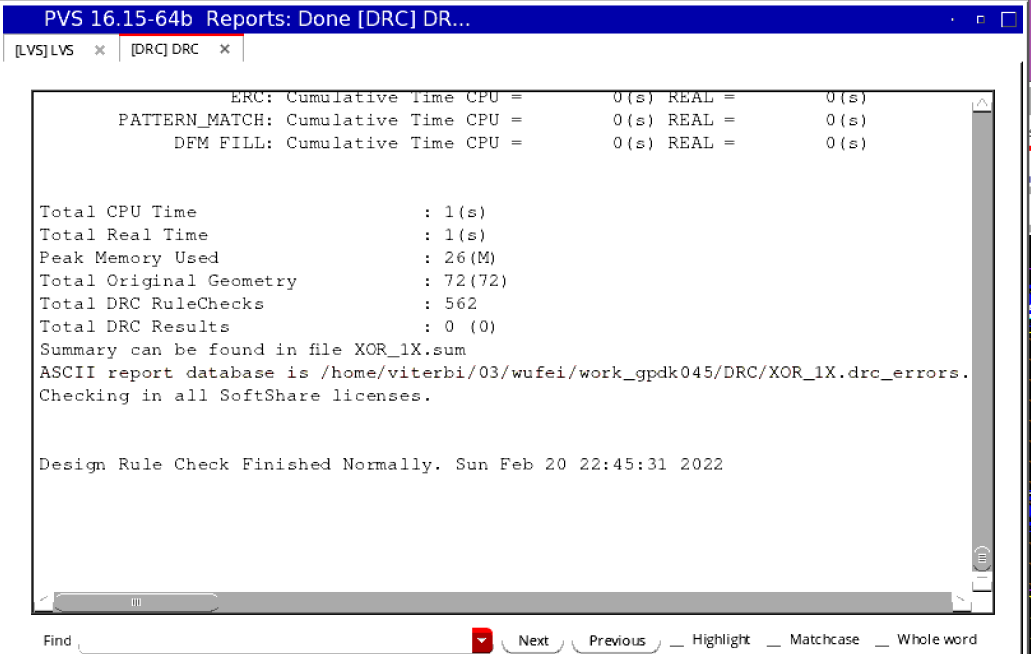




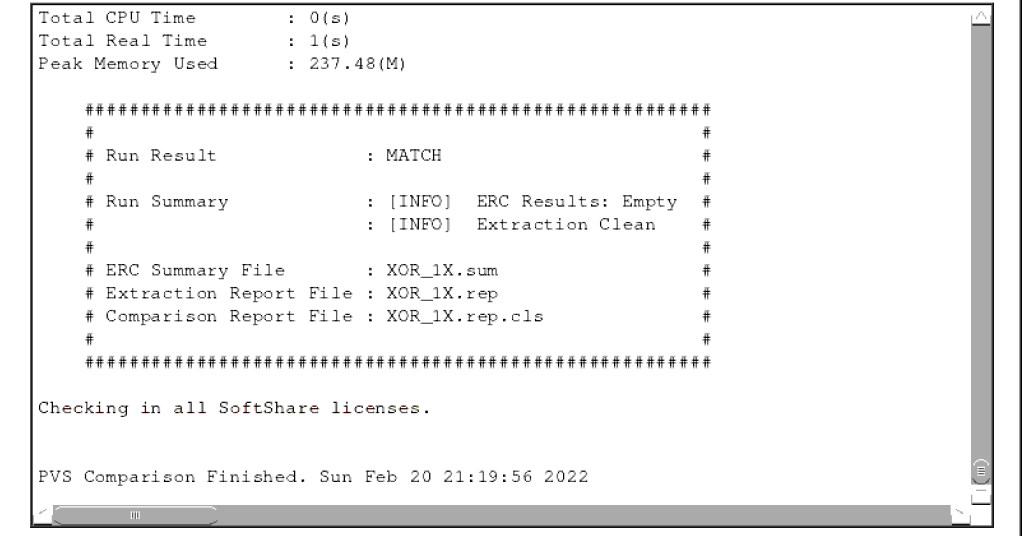
Draw the layout for the method you chose.



DRC

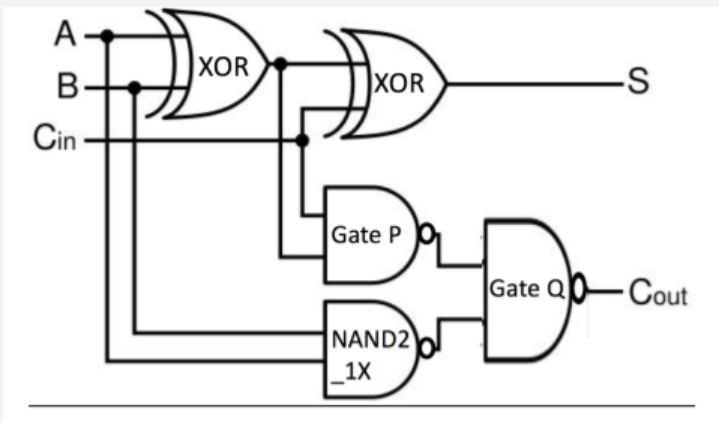


LVS



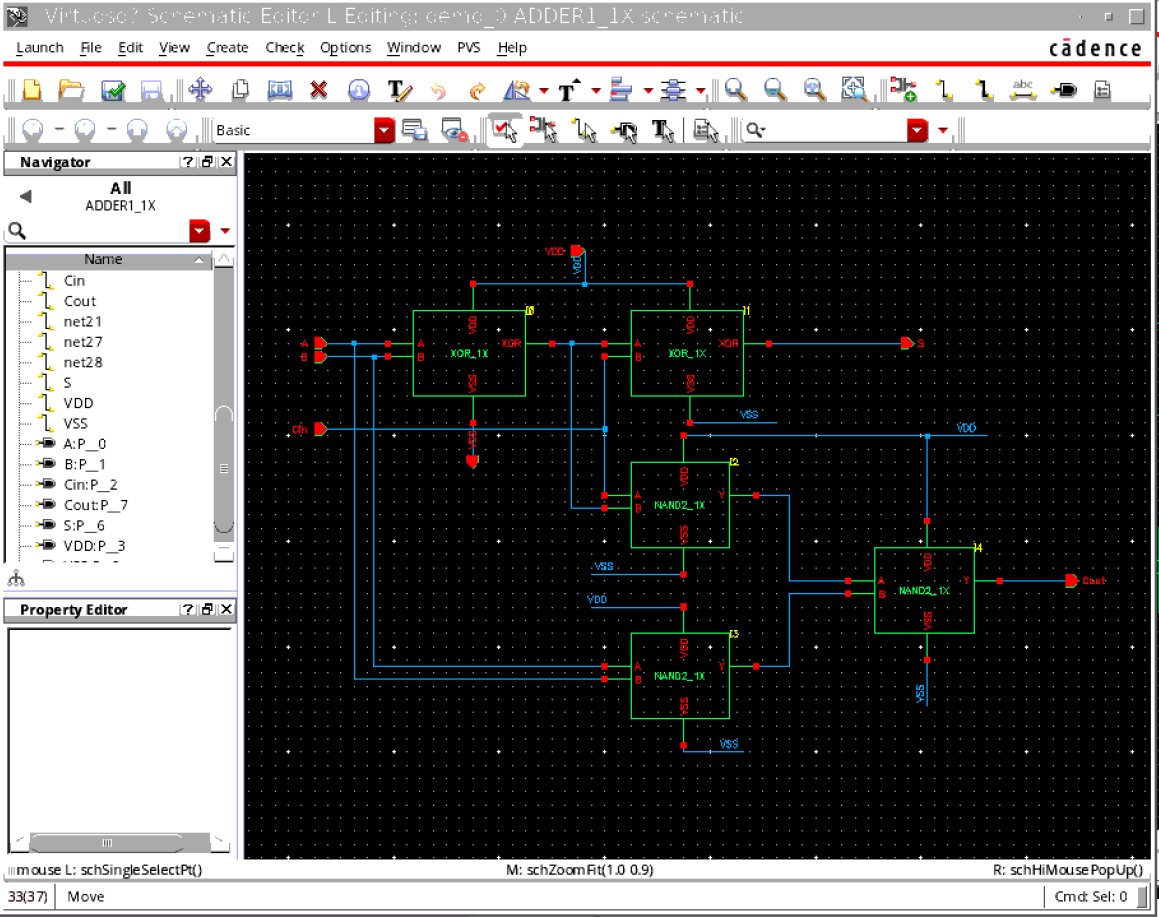
# 1\_bit full adder design

## Draw a 1b full adder schematic as shown.

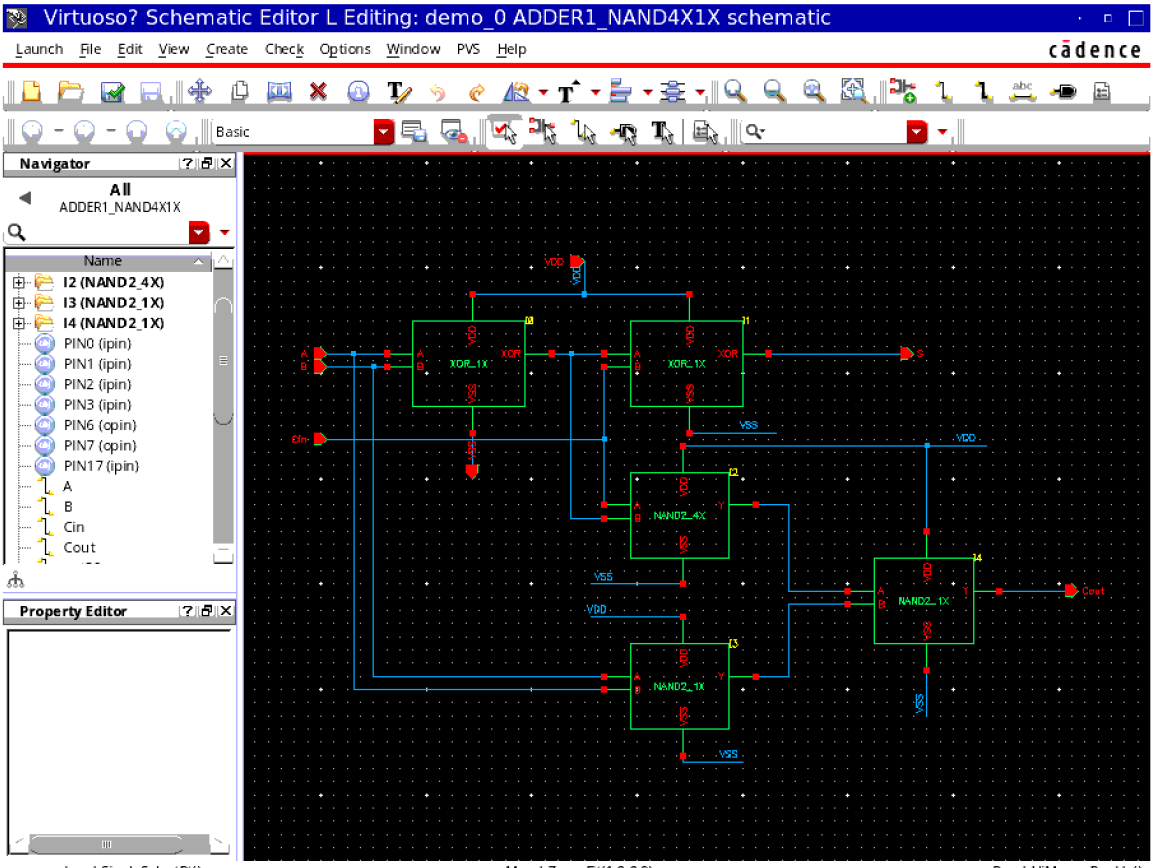


For gates P and Q, try 3 combinations:

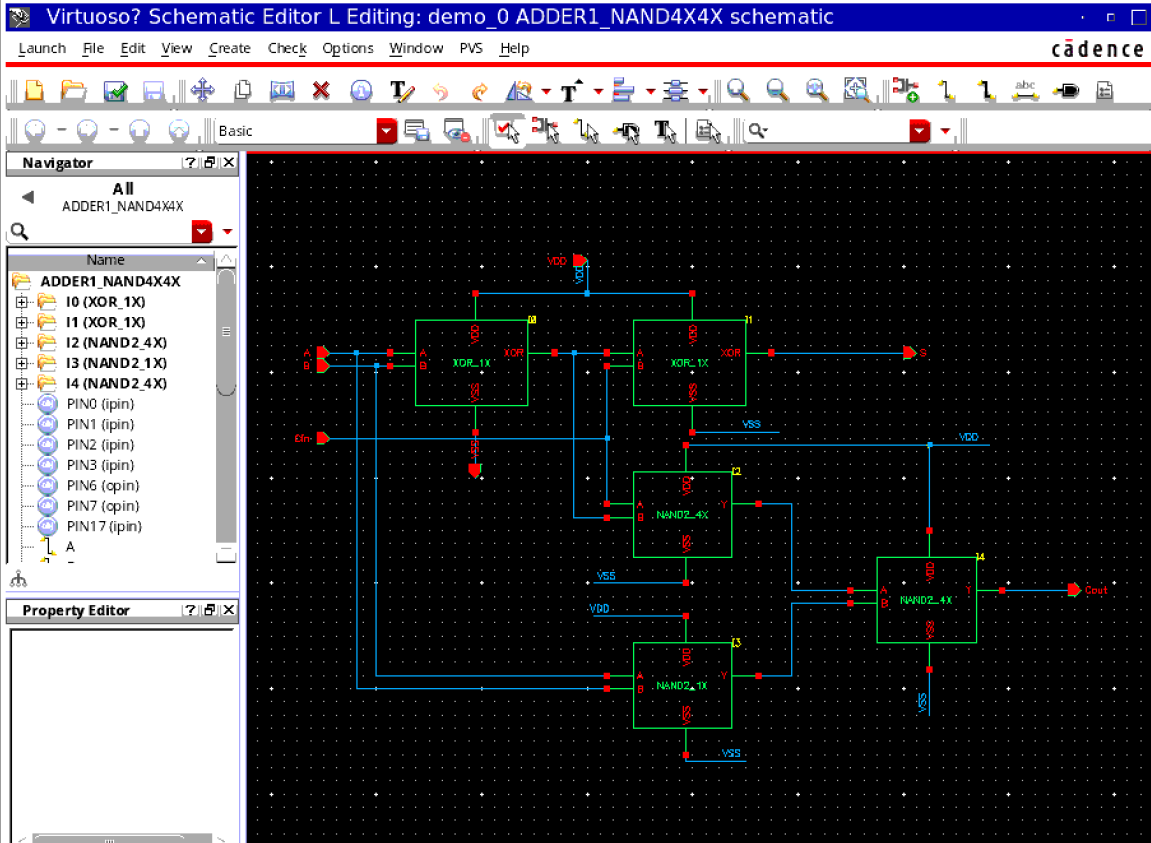
P = NAND2\_1X, Q = NAND2\_1X



P = NAND2\_4X, Q = NAND2\_1X



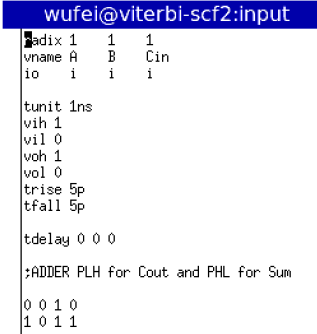
P = NAND2\_4X, Q = NAND2\_4X



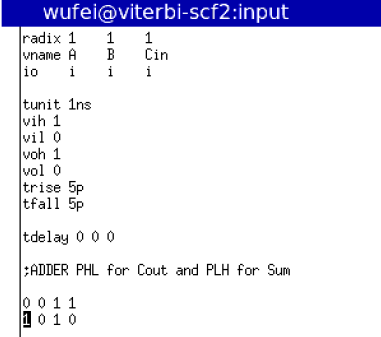
## Design a delay measurement circuit

with INV\_1X connected to all 3 inputs A, B, Cin and INV\_4X connected to both outputs S and Cout. Measure the delay from Cin to each output using the following transitions:

### (A, B, Cin) = (0,1,0) -> (0,1,1). This gives τPLH for Cout and τPHL for Sum.



### (A, B, Cin) = (0,1,1) -> (0,1,0). This gives τPHL for Cout and τPLH for Sum.

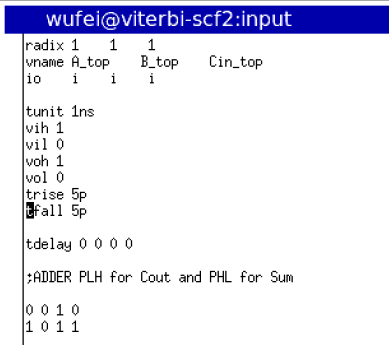


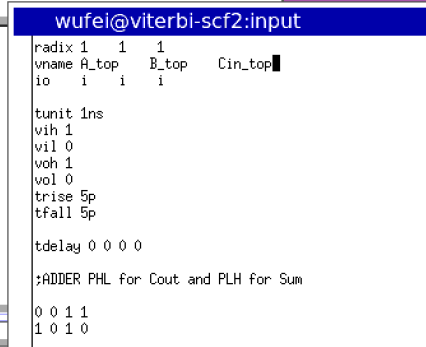
## Delay test

As usual, delay will be the time difference between input=0.5Vdd and output=0.5Vdd.

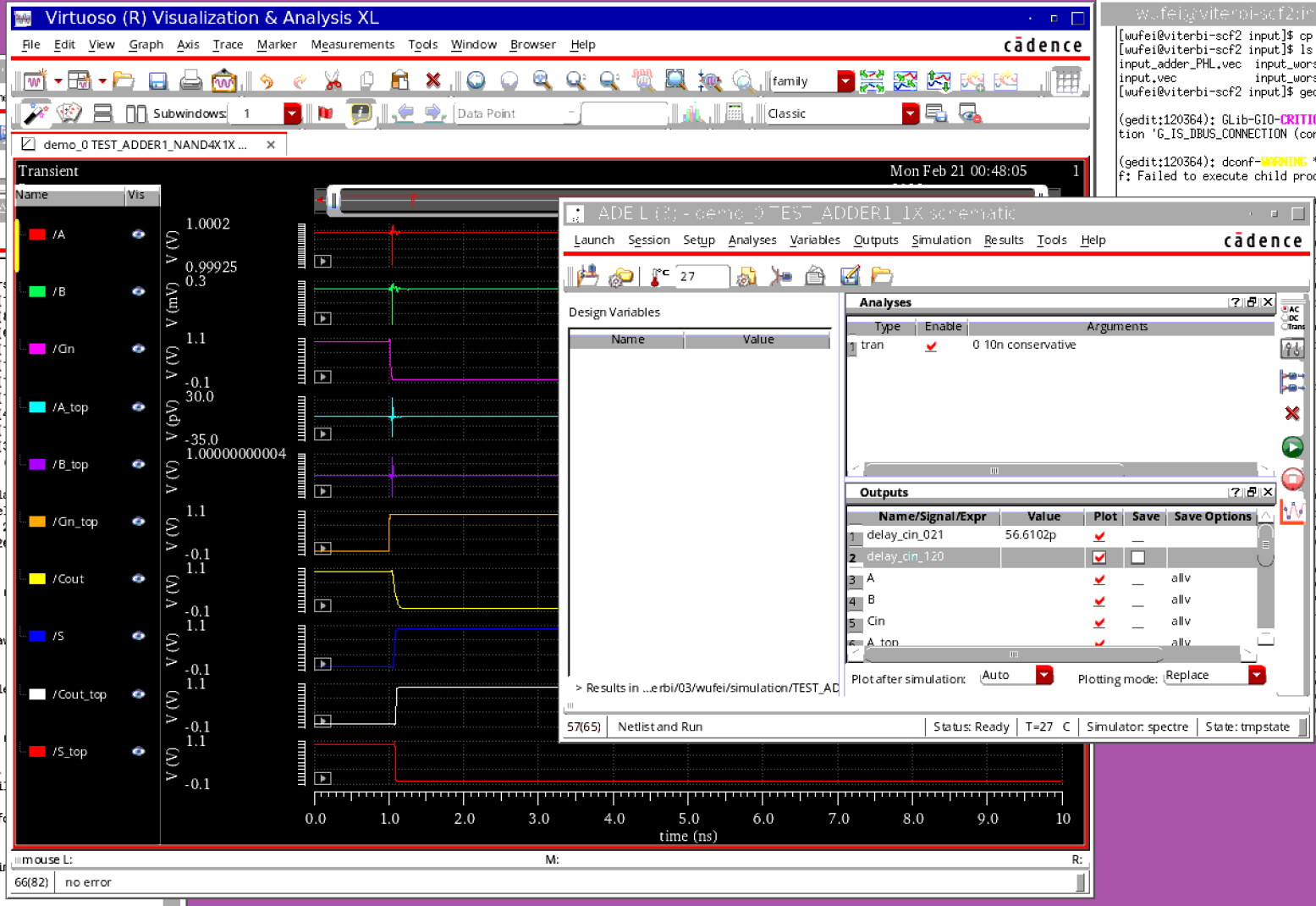
Find average delay from Cin to Cout = 0.5\*(τPLH+τPHL) and pick the (P,Q) combination which gives the least average delay for Cout. This is your final design.

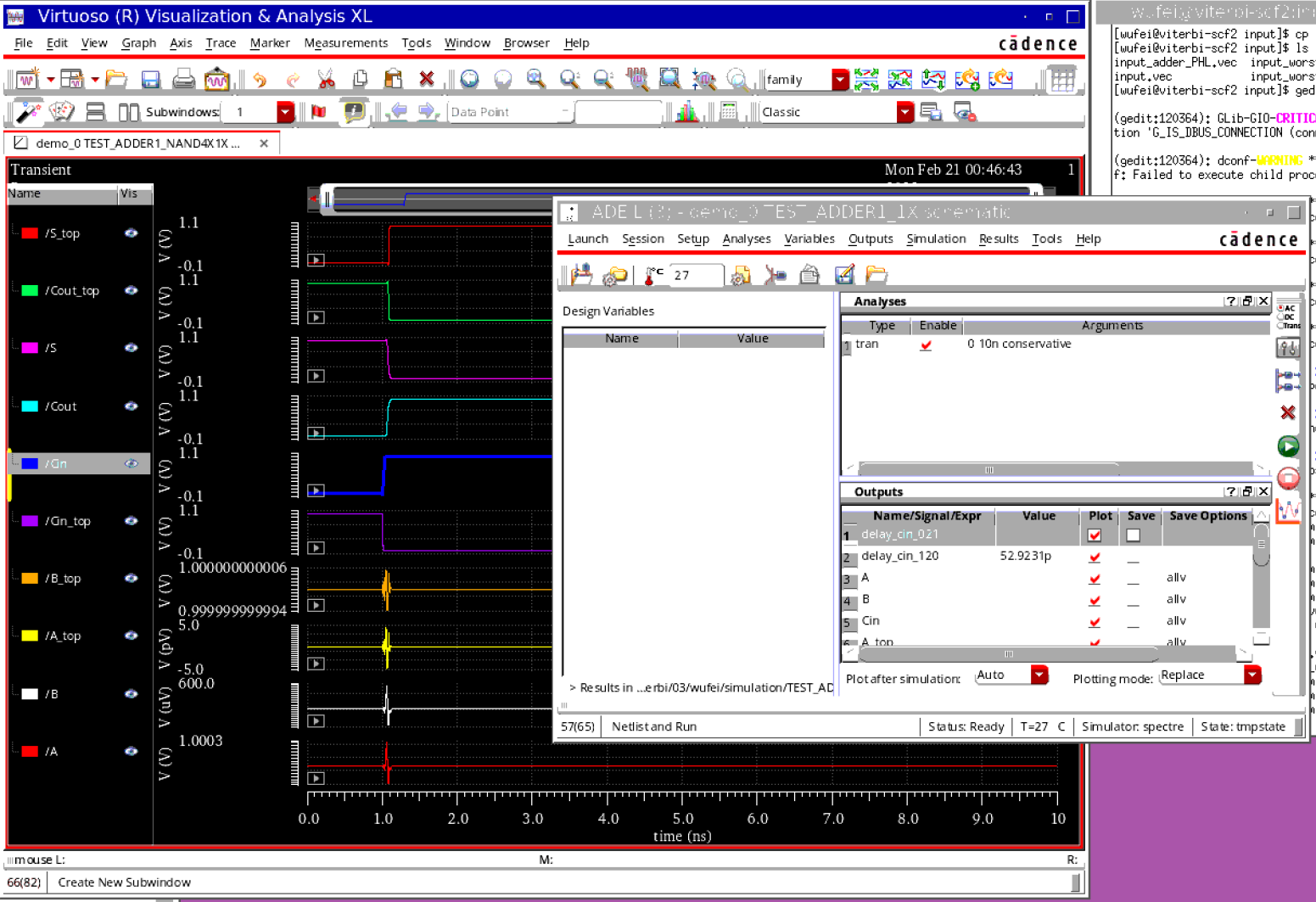
Draw the layout for this design.



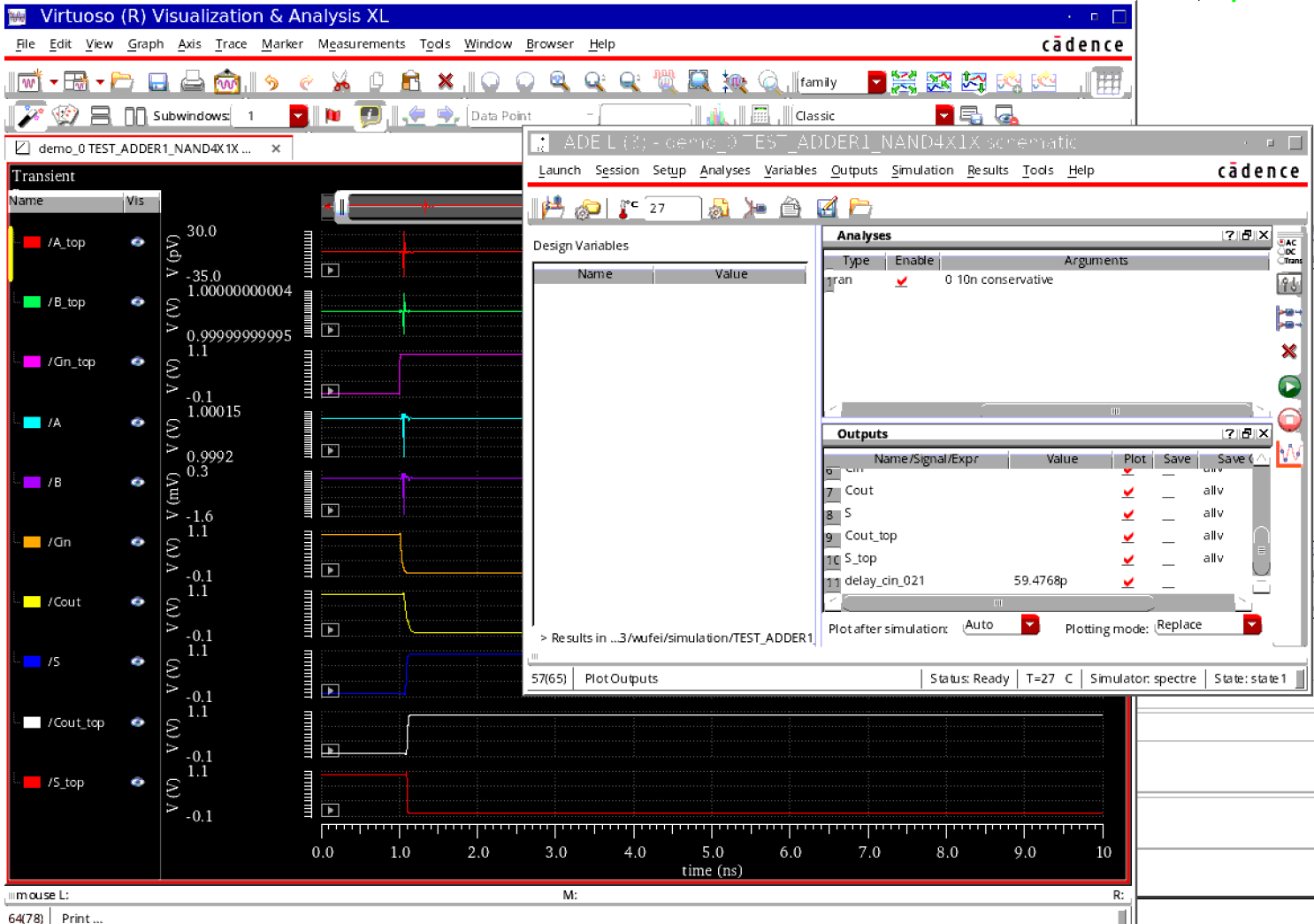


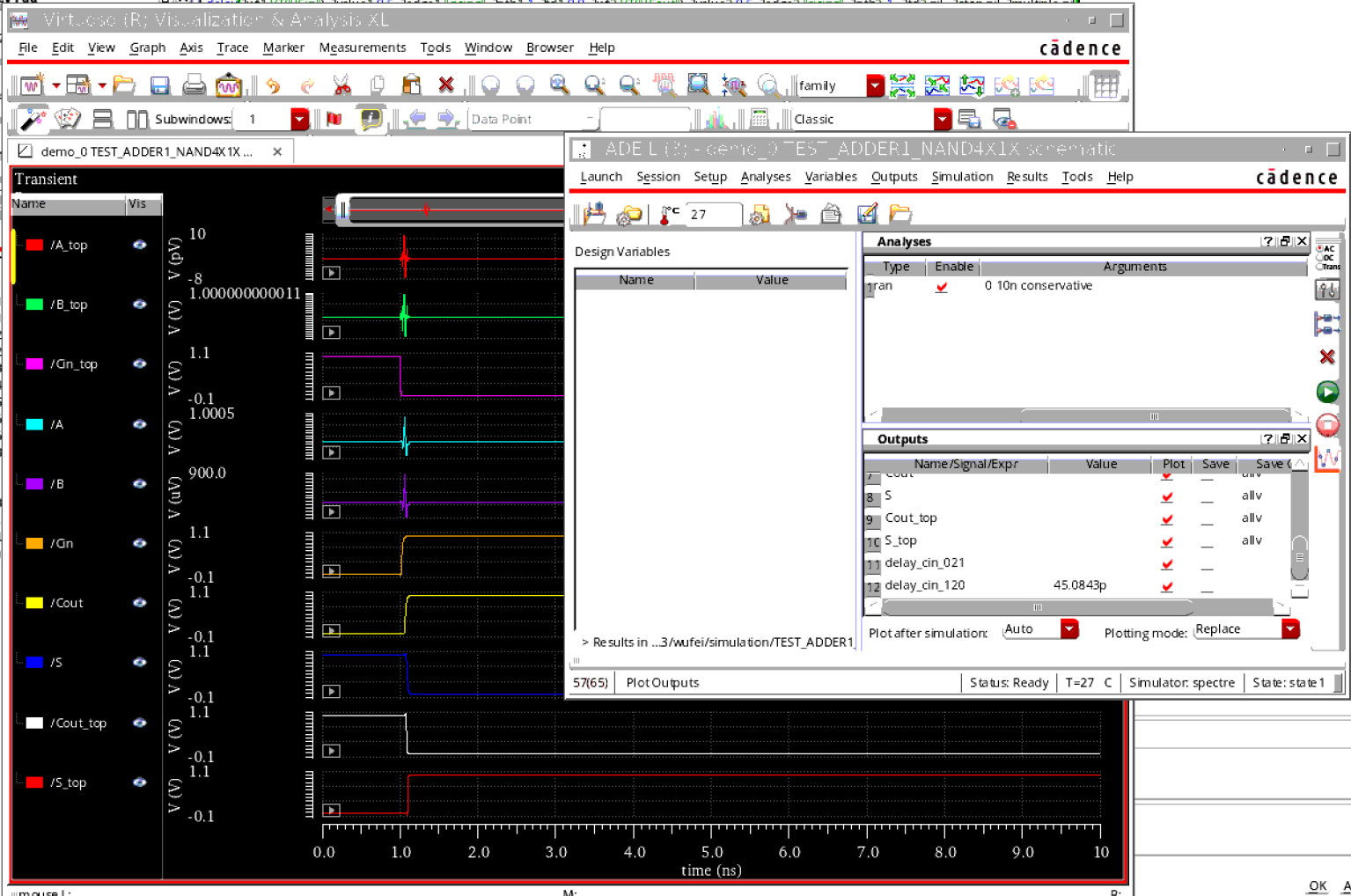
### P = NAND2\_1X, Q = NAND2\_1X



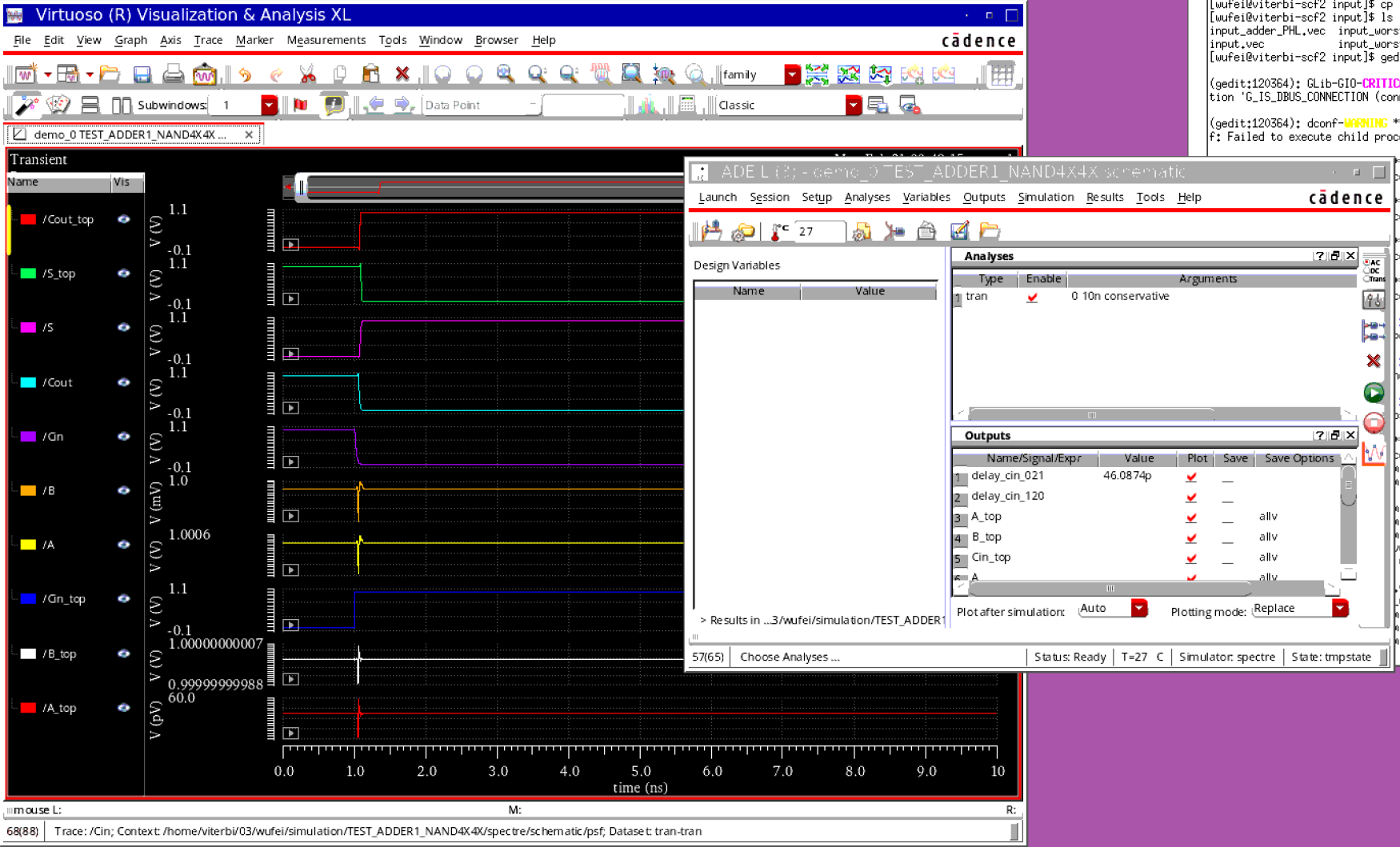


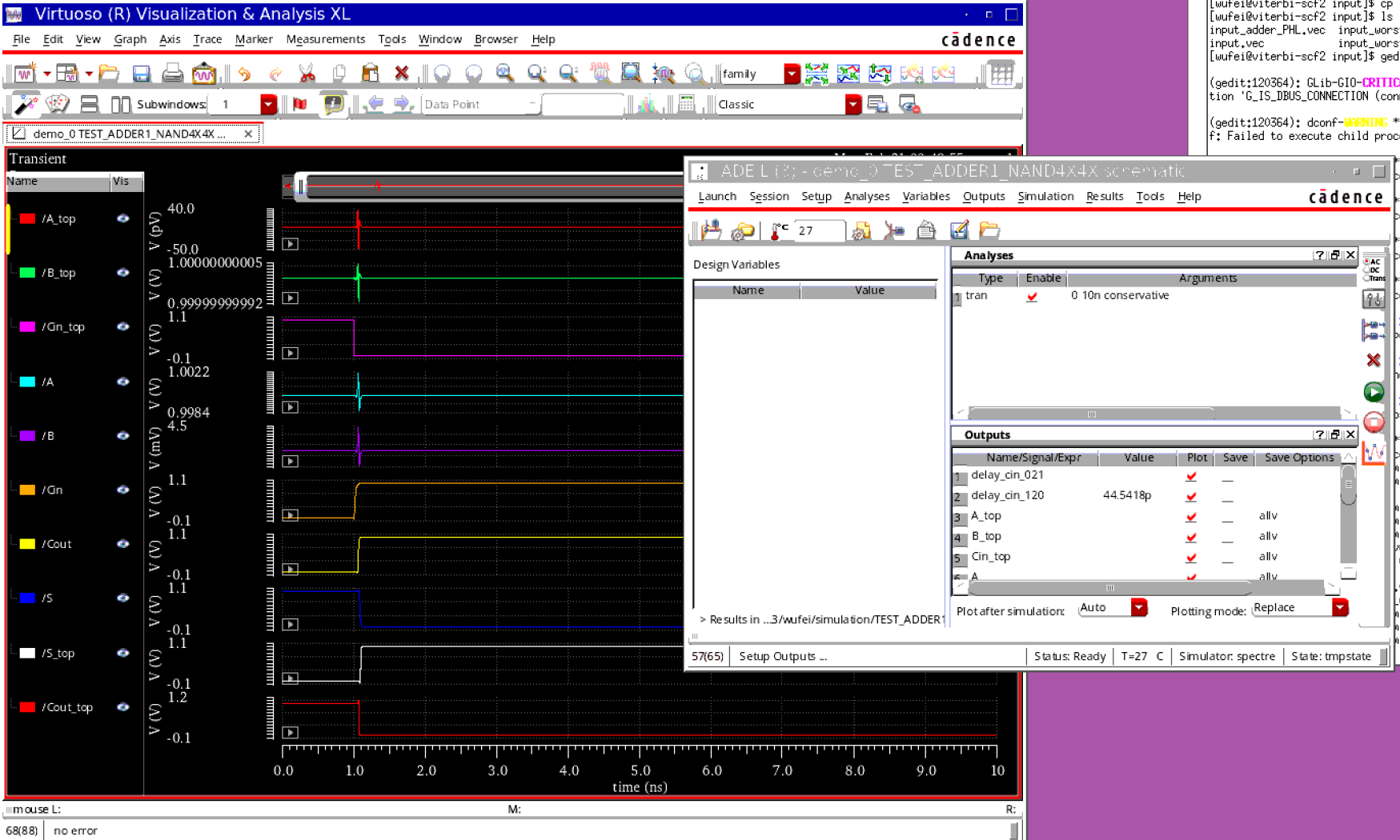
### P = NAND2\_4X, Q = NAND2\_1X





### P = NAND2\_4X, Q = NAND2\_4X

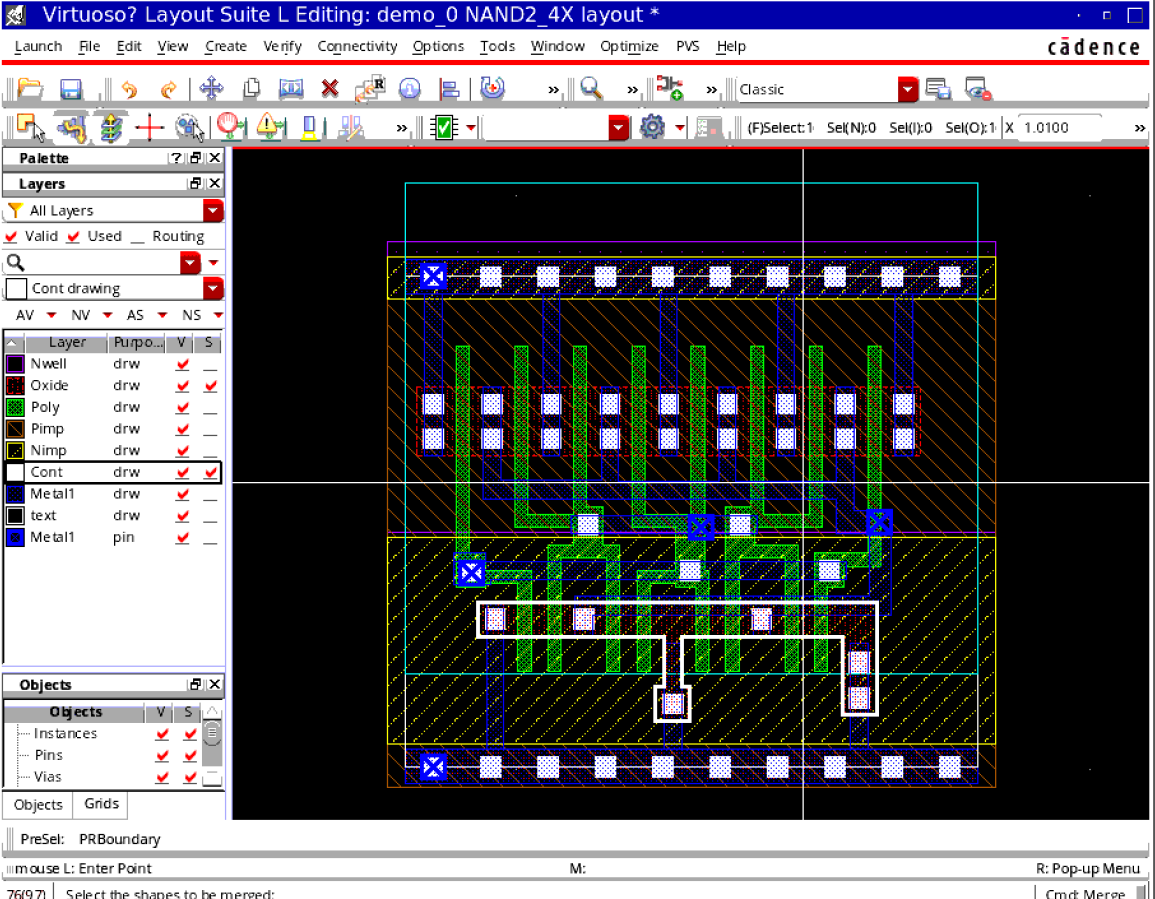


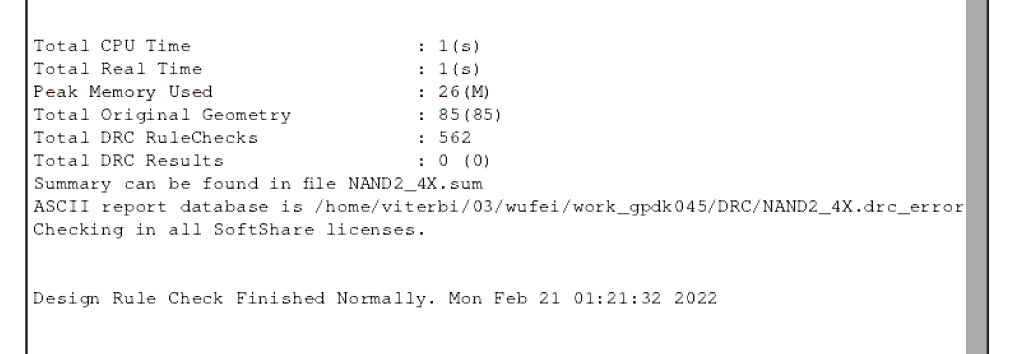
ich

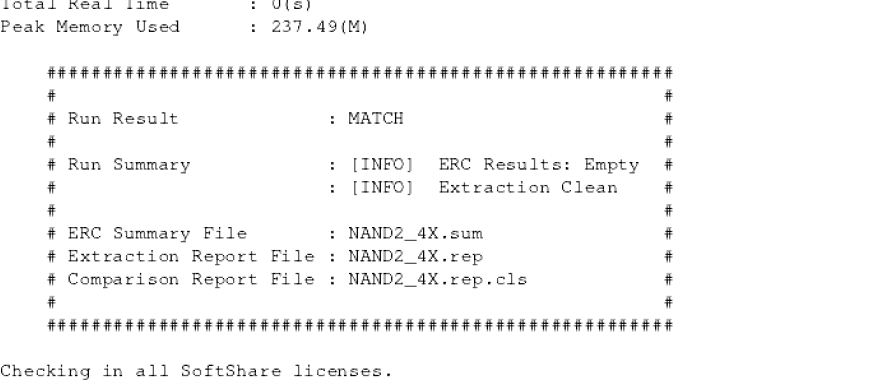
## Layout

Base on the test above, the c option which consist of NAND4X will be the best choose. So first we design the NAND4X.

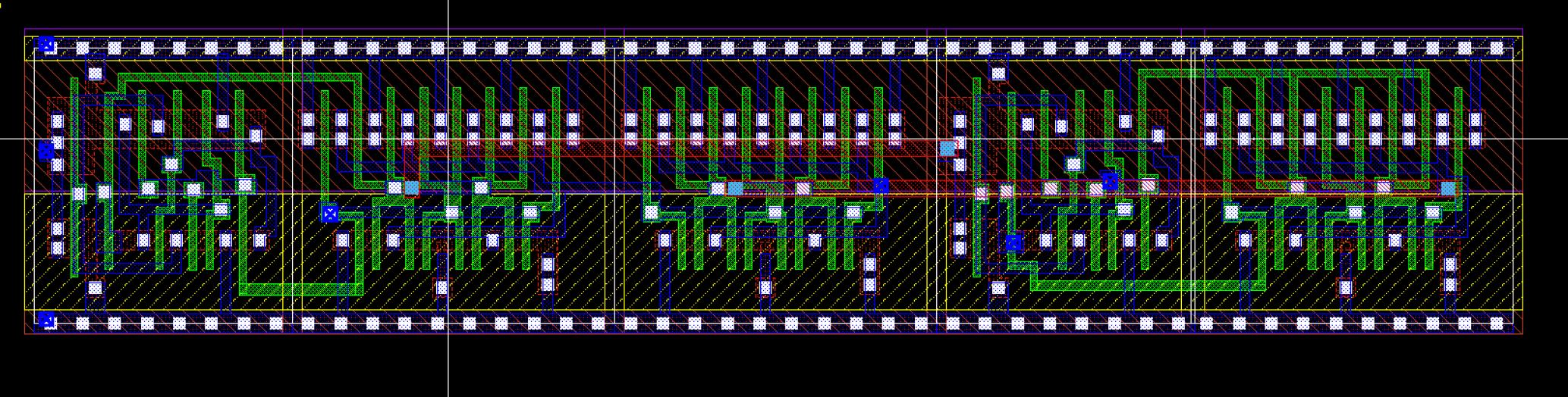


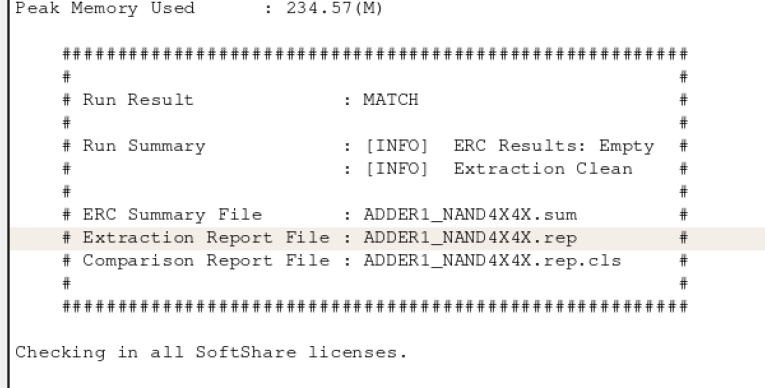


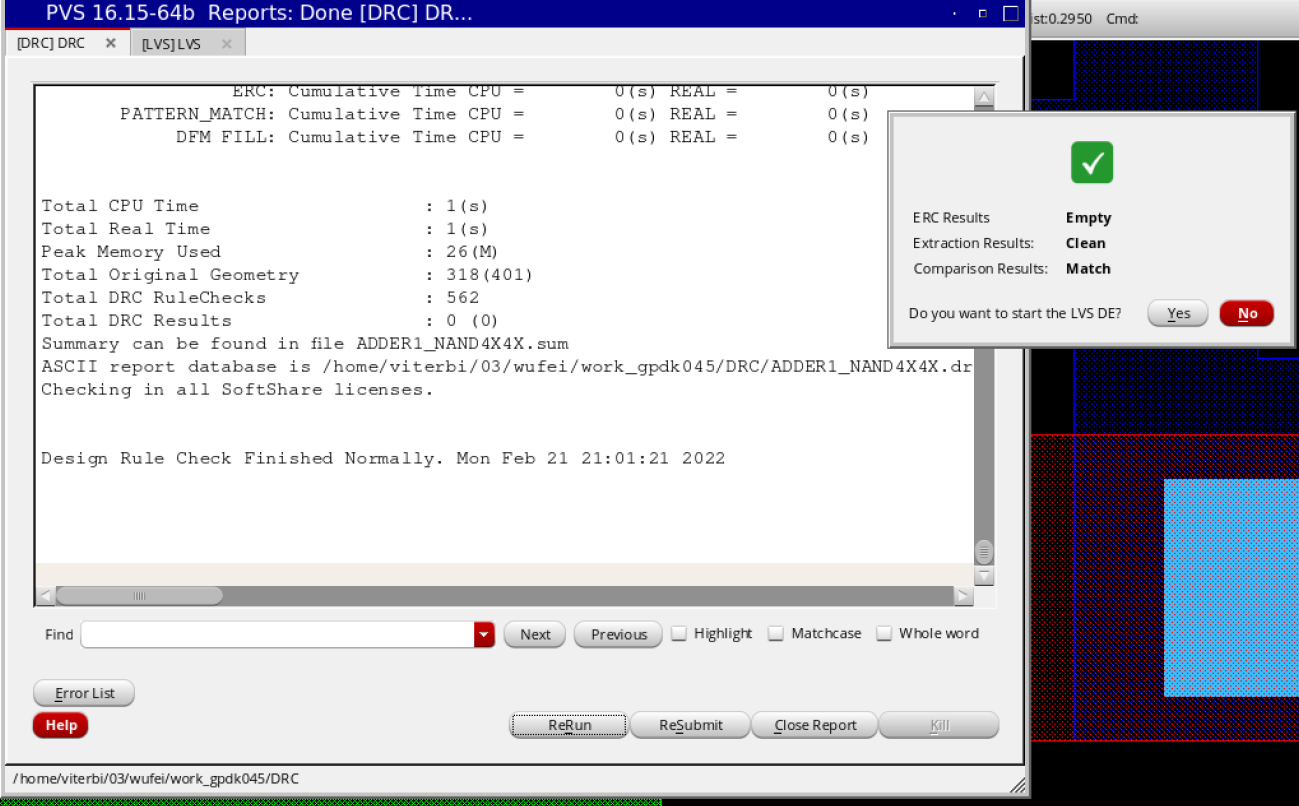




Then, I start to build the 1bit adder







# 4\_bit ripper carry adder

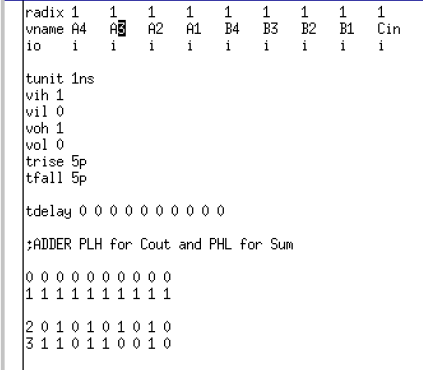


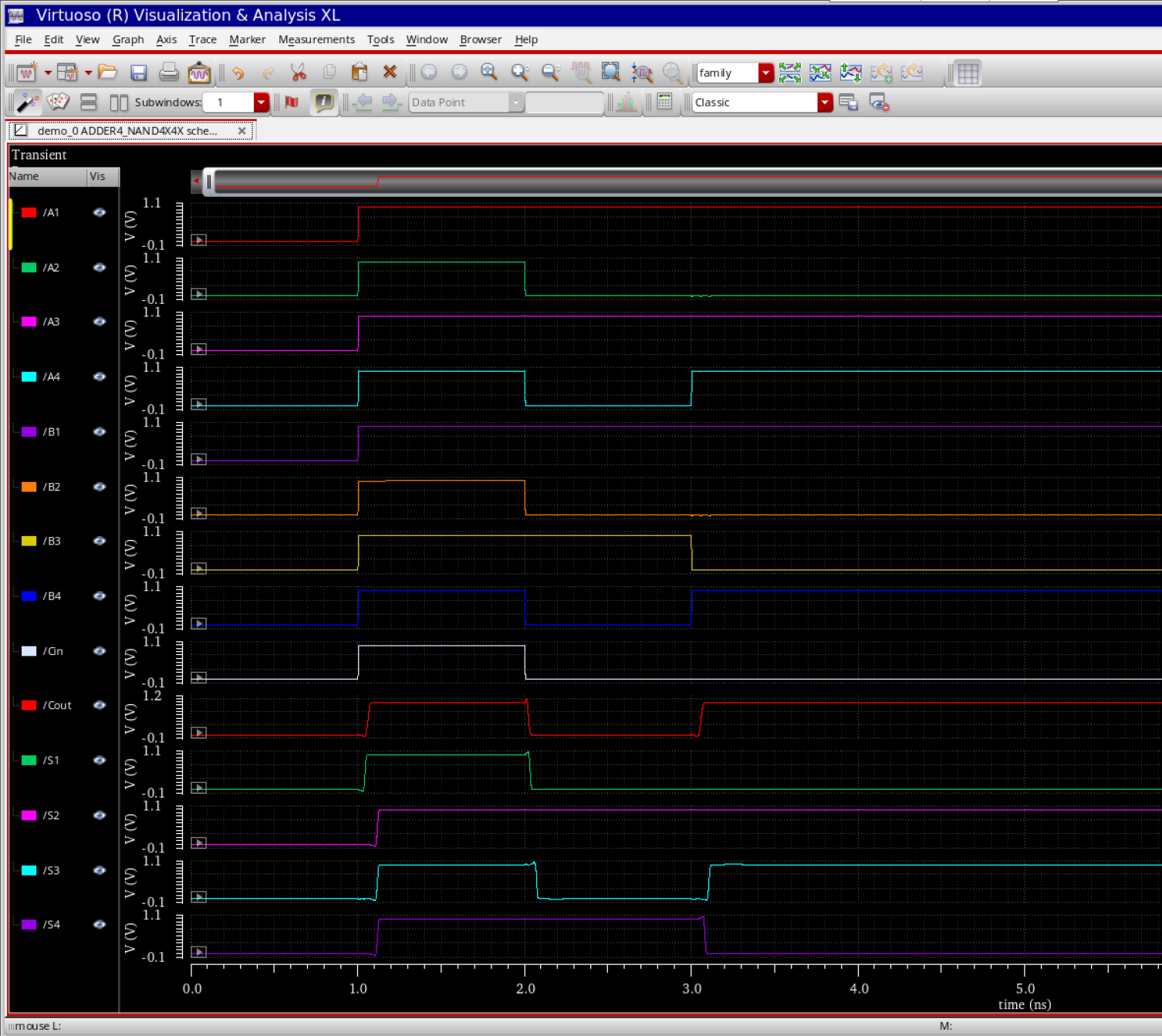
### Test the functionality

A = 15, B = 15, C0 = 1

A = 5, B = 5, C0 = 0

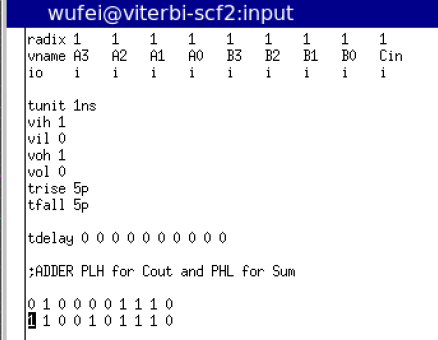
A = 13, B = 9, C0 = 0

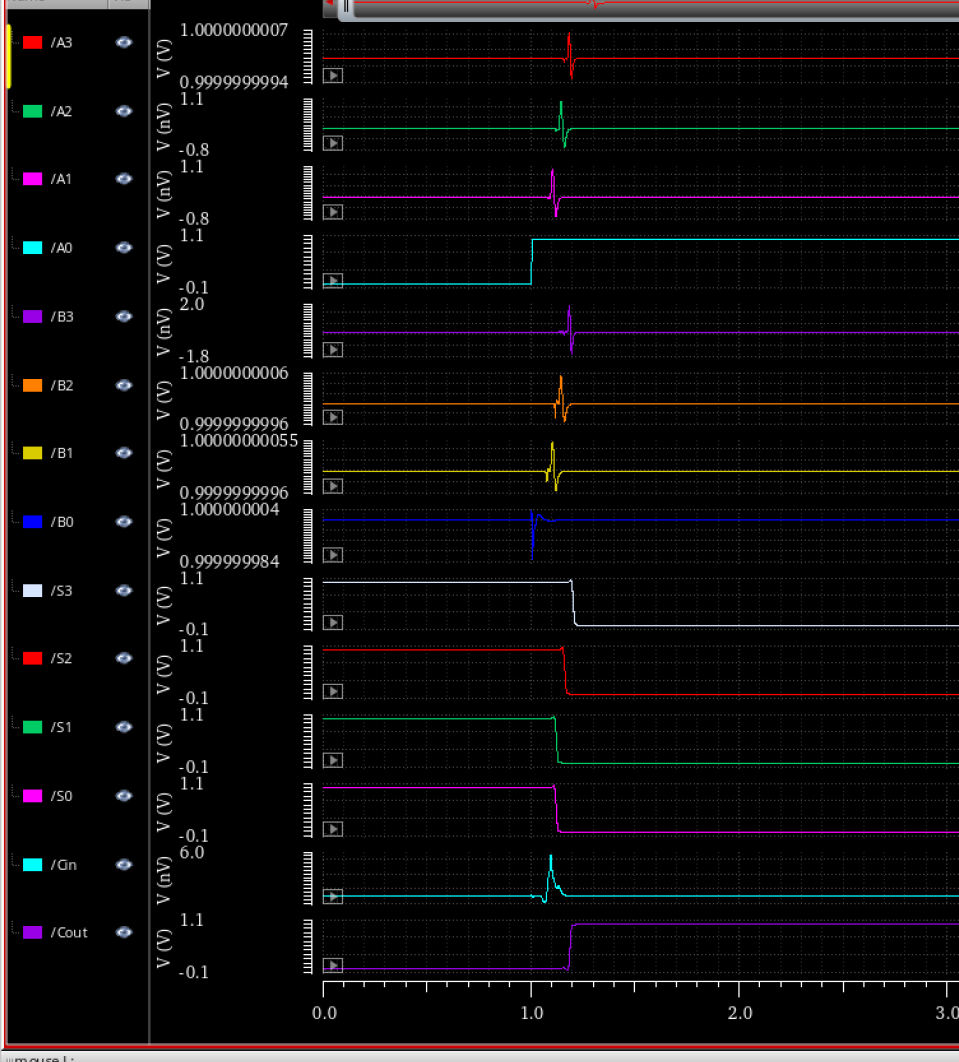


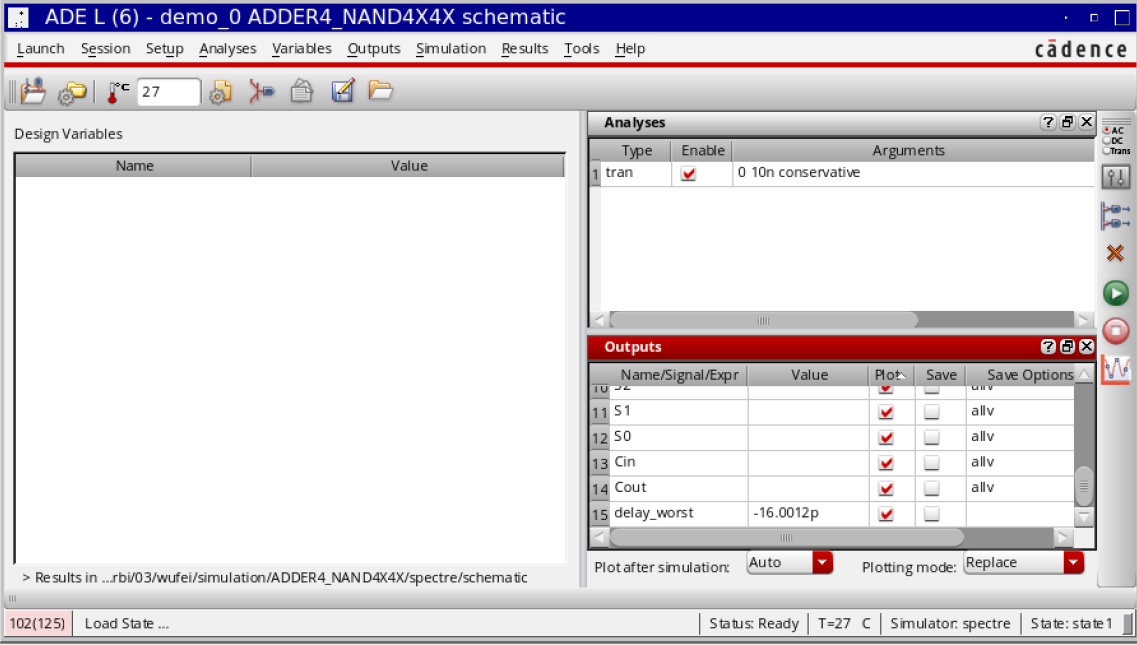


### Worst part

Find an input transition [A3,A2,A1,A0,B3,B2,B1,B0,C0] which results in worst case delay for outputs S3 and C4. The worst case delay case arises when all intermediate signals transition in the path from the changing input to a certain output. Hint: There is a single input transition which makes (C4,C3,C2,C1) go from (0,0,0,0)->(1,1,1,1) and S3 go from 1->0.







### Layout

