Digital Microelectronics

Politecnico di Torino

Dipartimento di Elettronica e Telecomunicazioni

Final Project: Standard Cell layout of an inverter and a half-adder

Group 5:
Marco Andorno (247222)
Michele Caon (253027)
Matteo Perotti (251453)

Date: June 20, 2018

1 Introduction

The goal of this final project is to design and characterize two logic gates of a standard cell library. These are in particular an inverter whose transistors have width four times the minimum, and therefore a 4 times the minimum drive strength (`INV_X4` in the following), and a half adder with minimum drive strength (`HA_X1` in the following).

The design starts from the schematic drawing, using an available `.png` image as a reference. This schematic will be simulated as an initial condition. Then, the layout of the cell must be drawn and the parasitic elements must be extracted from it. Finally, those parasitics must be taken into account during the final characterization. These results necessary to fill in the Liberty file of the standard cell library, replacing the old ones.

The files generated by the software tools for these two gates are inside the following paths on the server:

`/home/md18.5/project/INV_X4/`

and

`/home/md18.5/project/HAX1/`

while the current report file and the final Liberty files filled with the simulated timing parameters can be found in

`/home/md18.5/project/report_files/`

Notice that all the produced material, like scripts, plots and the current report files, can be found in the following GitHub repository:

`https://github.com/mksoc/standard-cell-design`

2 Inverter

2.1 Schematic

Starting from the schematic of the reference `.png` (figure 1), we copied it using Virtuoso Schematic Editor, obtaining the result shown in figure 2. Note that the four PMOS and NMOS transistors in parallel will be implemented as one large transistor in the layout, with the source/drain diffusions fingered.

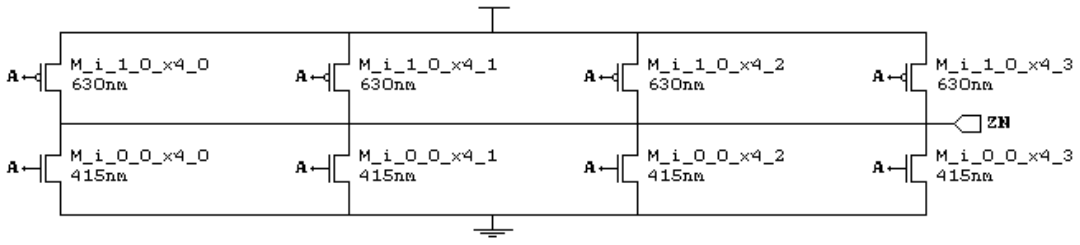


Figure 1: Reference schematic of the inverter

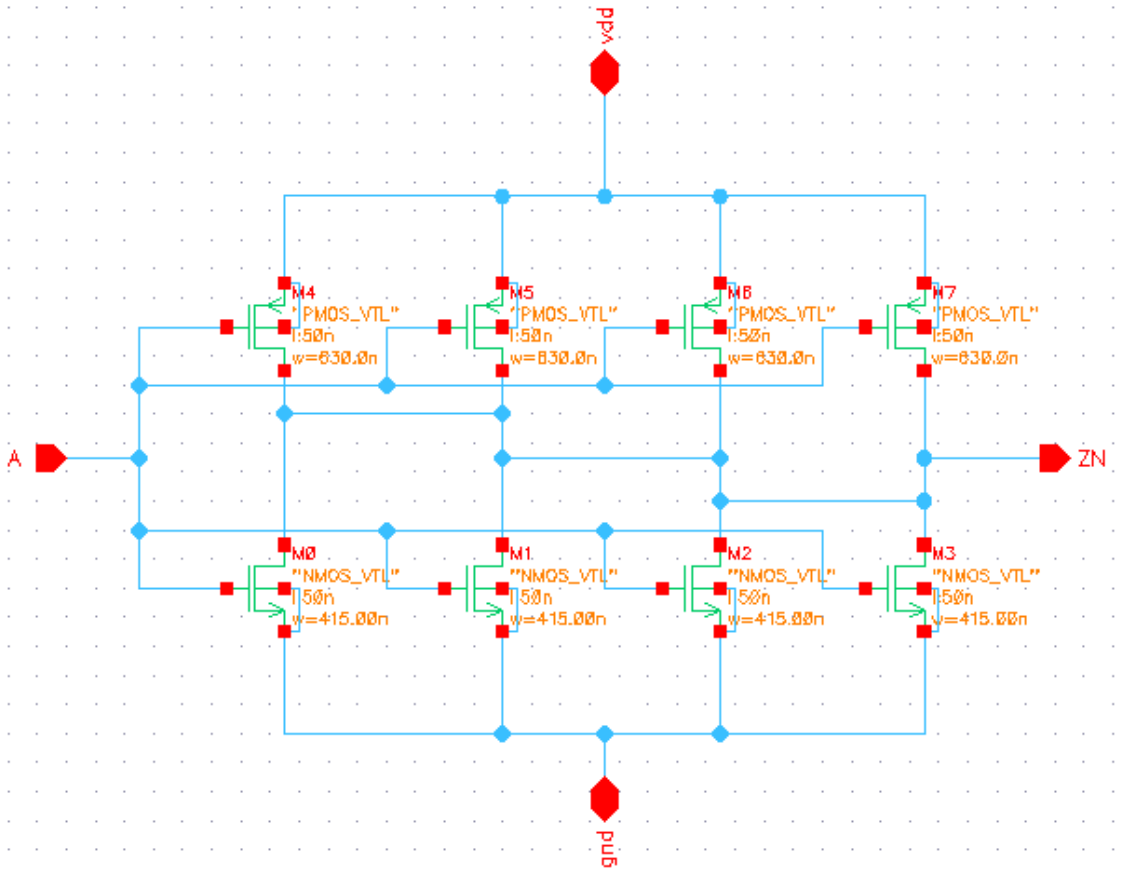


Figure 2: Final schematic of `INV_X4`

We then carried out some preliminary simulations using a test bench, to verify that the circuit worked correctly. We measured rise and fall times, as well as propagation delays, that will be compared with the actual characterization later on. We also measured the transfer characteristic, shown in figure 3.

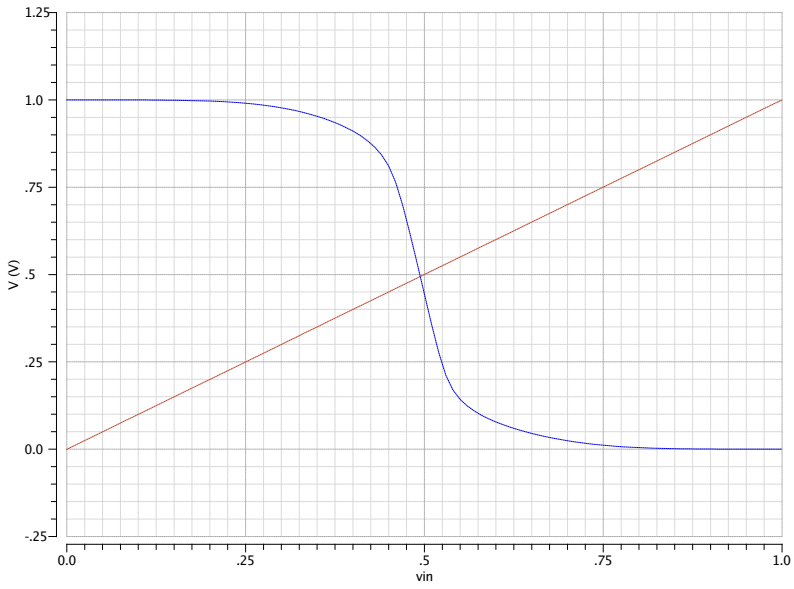


Figure 3: Inverter transfer characteristic

2.2 Layout

The final layout of `INV_X4` is shown in figure 4. The main trick to note here is how we used transistor fingering extensively, by sharing as much as possible the source and drain diffusions between the different gates. This is generally done to reduce the total area of the standard cell, but in this case it was actually mandatory, because a single transistor large 4 times the minimum width would not fit inside the well height, at least for the PMOS pull-up network.

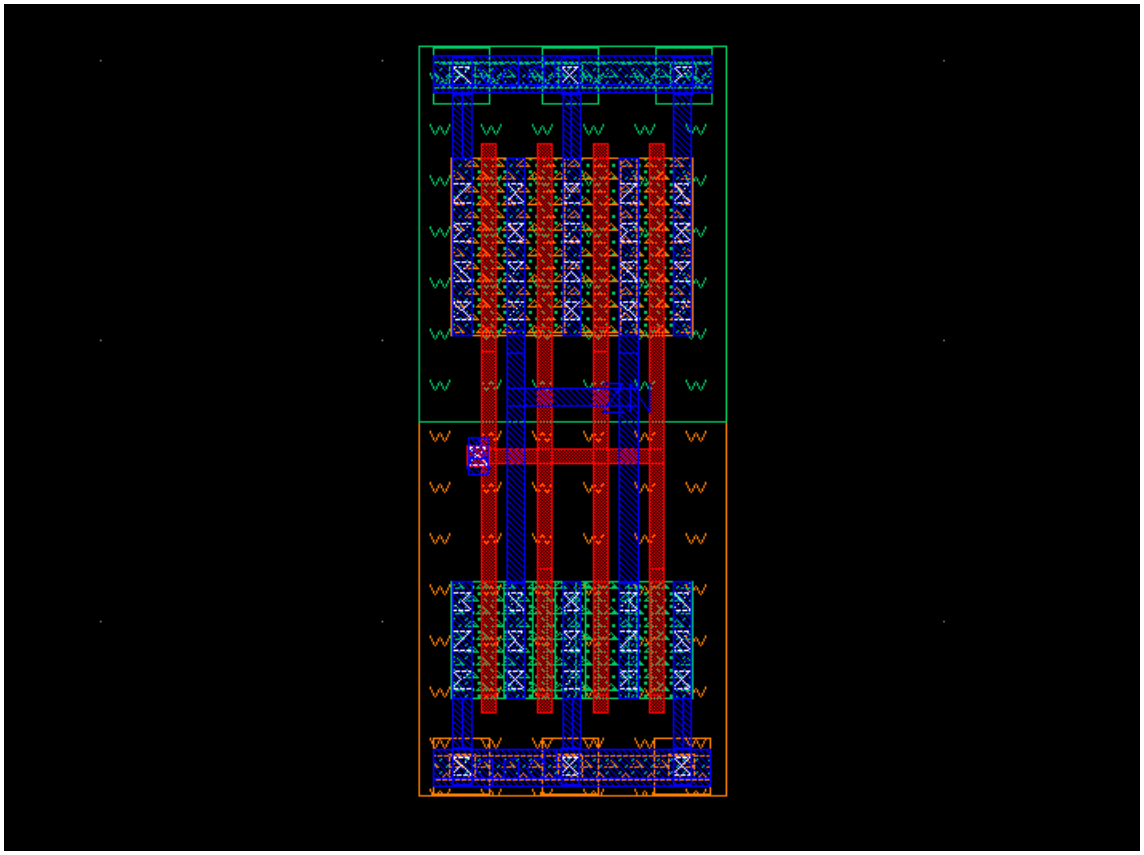


Figure 4: Layout of `INV_X4`

For PMOS transistors we were forced to finger four different transistor, while regarding the NMOS, probably, two fingered transistors with double width would have fit, but since the cell width would have been the same, we decided to go with four single width transistors anyway to improve symmetry.

Regarding pins, the output `ZN` did not cause any troubles and was placed on the rightmost metal strip connecting the drains of the PMOS and NMOS. As for the input `A`, instead, our first idea had been to place its via in the middle of the horizontal poly-silicon strip, but then we noticed that this way the external connections to that pin would cross the `metal1` layer, thus requiring `metal2` to be used instead. So eventually we settled with placing the via for the input on the leftmost part of the poly-silicon, avoiding any `metal1` crossing and also accurately mirroring the topology of the schematic.

While developing the layout we ran the DRC tool many times to ensure all the design rules were met. After completing it we got the following final statement:

```
================================================================================
=== CALIBRE::DRC-H SUMMARY REPORT
===
Execution Date/Time:       Wed May 23 17:11:38 2018
Calibre Version:           v2011.4_14.13    Tue Nov 15 15:18:31 PST 2011
Rule File Pathname:        /home/md18.5/project/INV_X4/_calibreDRC.rul_
Rule File Title:
Layout System:             GDS
Layout Path(s):            INV_X4.calibre.db
Layout Primary Cell:       INV_X4
Current Directory:         /home/md18.5/project/INV_X4
User Name:                 md18.5
Maximum Results/RuleCheck: 1000
Maximum Result Vertices:   4096
DRC Results Database:      INV_X4.drc.results (ASCII)
Layout Depth:              ALL
Text Depth:                PRIMARY
Summary Report File:       INV_X4.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
---
--------------------------------------------------------------------------------
--- ORIGINAL LAYER STATISTICS
---
LAYER pwell ...... TOTAL Original Geometry Count = 4  (4)
.
.
.
LAYER via9 ....... TOTAL Original Geometry Count = 0  (0)
--------------------------------------------------------------------------------
--- RULECHECK RESULTS STATISTICS
---
RULECHECK Well.1 ............ TOTAL Result Count = 0 (0)
.
.
.
RULECHECK Antenna.metal10 ... TOTAL Result Count = 0 (0)
--------------------------------------------------------------------------------
--- RULECHECK RESULTS STATISTICS (BY CELL)
---
--------------------------------------------------------------------------------
--- SUMMARY
---
TOTAL CPU Time:                  0
TOTAL REAL Time:                 0
TOTAL Original Layer Geometries: 171 (171)
TOTAL DRC RuleChecks Executed:   167
TOTAL DRC Results Generated:     0 (0)
```

Moreover, we compared the layout and the schematic net-lists using LVS, that succeeded on the first try with its reassuring smiling smiley-face:

```
                ################################################
                ##                                            ##
                ##        C A L I B R E   S Y S T E M         ##
                ##                                            ##
                ##            L V S   R E P O R T             ##
                ##                                            ##
                ################################################
REPORT FILE NAME:         INV_X4.lvs.report
LAYOUT NAME:              /home/md18.5/project/INV_X4/INV_X4.sp ('INV_X4')
SOURCE NAME:              /home/md18.5/project/INV_X4/INV_X4.src.net ('INV_X4')
RULE FILE:                /home/md18.5/project/INV_X4/_calibreLVS.rul_
RULE FILE TITLE:          LVS Rule File for FreePDK45
CREATION TIME:            Tue May 29 15:46:49 2018
CURRENT DIRECTORY:        /home/md18.5/project/INV_X4
USER NAME:                md18.5
CALIBRE VERSION:          v2011.4_14.13    Tue Nov 15 15:18:31 PST 2011
                               OVERALL COMPARISON RESULTS
                           #       ###################       _   _
                          #        #                 #       *   *
                      #   #        #     CORRECT     #         |
                       # #         #                 #       \___/
                        #          ###################
********************************************************************************
                                    CELL  SUMMARY
********************************************************************************
  Result         Layout                      Source
  -----------    -----------                 --------------
  CORRECT        INV_X4                      INV_X4
********************************************************************************
                                   LVS PARAMETERS
********************************************************************************
o LVS Setup:
.
.
.
                    CELL COMPARISON RESULTS ( TOP LEVEL )
                          #       ###################       _   _
                         #        #                 #       *   *
                    #   #         #     CORRECT     #         |
                     # #          #                 #       \___/
                      #           ###################
LAYOUT CELL NAME:         INV_X4
SOURCE CELL NAME:         INV_X4
--------------------------------------------------------------------------------
INITIAL NUMBERS OF OBJECTS
--------------------------
                Layout    Source         Component Type
                ------    ------         --------------
 Ports:              4         4
 Nets:               4         4
 Instances:          4         4         MN (4 pins)
                     4         4         MP (4 pins)
                ------    ------
 Total Inst:         8         8
NUMBERS OF OBJECTS AFTER TRANSFORMATION
---------------------------------------
                Layout    Source         Component Type
                ------    ------         --------------
 Ports:              4         4
 Nets:               4         4
 Instances:          1         1         _invv (4 pins)
```

```
                ------    ------
 Total Inst:         1         1
********************************************************************************
                              INFORMATION AND WARNINGS
********************************************************************************
                  Matched    Matched    Unmatched    Unmatched    Component
                   Layout     Source       Layout       Source    Type
                  -------    -------    ---------    ---------    ---------
   Ports:               4          4            0            0
   Nets:                4          4            0            0
   Instances:           1          1            0            0    _invv
                  -------    -------    ---------    ---------
   Total Inst:          1          1            0            0
o Statistics:
   8 layout mos transistors were reduced to 2.
     6 mos transistors were deleted by parallel reduction.
   8 source mos transistors were reduced to 2.
     6 mos transistors were deleted by parallel reduction.
o Layout Names That Are Missing In The Source:
   Ports:        GND! VDD!
   Nets:         GND! VDD!
o Initial Correspondence Points:
   Ports:        A ZN
********************************************************************************
                                       SUMMARY
********************************************************************************
Total CPU Time:      0 sec
Total Elapsed Time:  0 sec
```

Just a remark on the warning that says that nets `GND!` and `VDD!` are present in the layout but missing in the source (schematic). We tried to follow the steps of the laboratory sessions where the supply nets had the exclamation mark at the end in the layout view but not in the schematic one, and in fact we had no problems with this method for the `INV_X4`. However, we had a lot of issues for `HA_X1` when running LVS as it complained about missing power nets. We finally came to the conclusion that schematic and layout must have the exact same names for supply nets and that the exclamation mark can be avoided if the schematic does not exploit global nets to make the drawing cleaner and clearer. Even then, to this day it remains unclear why the inverter cell worked just as well with different names for power nets (disregarding this warning), while the half adder with the same set-up caused so much trouble. We can only accept this and give in to the mighty secrets of Virtuoso.

2.3 Characterization

The first step toward the complete characterization of our logic gate was the extraction of the parasitic elements from the layout. To do this, we ran the PEX tool from the layout editor and got the following report as well as a complete new net-list that takes into account those parasitics in the schematic.

```
                  ##################################################
                  ##                                              ##
                  ##                 Calibre xRC                  ##
                  ##                                              ##
                  ##          Export Lumped Parameters            ##
                  ##                                              ##
                  ##################################################
LAYOUT NAME:              INV_X4
RULE FILE NAME:           rules
CREATION TIME:            Thu May 31 18:19:25 2018
UNITS:                    Resistance  = ohm
```

```
                        Capacitance = farad
                        Time        = ns
--------------------------------------------------------------------------------
CELL NAME:              INV_X4
Netid      R(UpperBound)      Cvalue      %Coupled    RC(UpperBound)   Netname
---------+---------------+-------------+-------------+--------------+-----------
1                   0.0   5.50637e-16       29.1819            0.0  A
           - Coupled nets
              MM3:d,MM3:d,MM3:d,MM3:d,MM3:d,MM3:d,MM7:d,c_43:n,c_43:n,c_43:n,MM1:d,MM1:d,
              MM1:d,MM1:d,MM1:d,MM5:d,c_52:n,MM3:s,c_70:n,c_70:n,c_70:n,MM2:s,MM2:s,MM2:s,
              c_76:n,c_76:n,MM0:s,MM7:s,c_90:n,c_90:n,MM6:s,MM6:s,c_94:n,c_94:n,MM4:s
           - Intrinsic Capacitance
              3.89951e-16
           - Coupled capacitance
              8.86343e-18   netid: MM3:d
                6.347e-18   netid: MM3:d
              4.99255e-18   netid: MM3:d
               4.8594e-18   netid: MM3:d
              1.08097e-18   netid: MM3:d
              1.78248e-17   netid: MM3:d
              4.09779e-18   netid: MM7:d
              4.17774e-18   netid: c_43:n
                9.749e-19   netid: c_43:n
              4.32339e-18   netid: c_43:n
               4.8594e-18   netid: MM1:d
              1.08097e-18   netid: MM1:d
              6.73195e-18   netid: MM1:d
              9.48261e-18   netid: MM1:d
              1.06043e-17   netid: MM1:d
              4.14482e-18   netid: MM5:d
              4.99255e-18   netid: c_52:n
              8.27127e-18   netid: MM3:s
              1.34578e-18   netid: c_70:n
              1.35056e-18   netid: c_70:n
              2.57507e-19   netid: c_70:n
              2.18914e-18   netid: MM2:s
              4.78238e-19   netid: MM2:s
              2.18914e-18   netid: MM2:s
              1.35056e-18   netid: c_76:n
               1.4614e-18   netid: c_76:n
              8.27127e-18   netid: MM0:s
              1.21306e-17   netid: MM7:s
              1.15069e-18   netid: c_90:n
              1.11503e-18   netid: c_90:n
              3.18871e-18   netid: MM6:s
              3.16413e-18   netid: MM6:s
              7.07137e-19   netid: c_94:n
              4.92603e-19   netid: c_94:n
              1.21341e-17   netid: MM4:s
2                   0.0   4.97637e-16       73.1851            0.0  ZN
           - Coupled nets
              MM3:g,c_2:p,c_3:p,MM2:g,MM6:g,c_6:p,MM6:g,MM6:g,c_9:p,MM5:g,MM1:g,MM5:g,
              c_13:p,MM0:g,MM4:g,MM5:g,MM7:g,MM3:s,c_70:n,MM2:s,MM2:s,MM2:s,c_76:n,c_76:n,
              MM0:s,MM7:s,c_90:n,c_90:n,MM6:s,MM6:s,MM6:s,c_94:n,MM4:s
           - Intrinsic Capacitance
              1.33441e-16
           - Coupled capacitance
              8.86343e-18   netid: MM3:g
                6.347e-18   netid: c_2:p
              4.99255e-18   netid: c_3:p
               4.8594e-18   netid: MM2:g
              1.08097e-18   netid: MM6:g
              1.78248e-17   netid: c_6:p
```

```
                4.09779e-18   netid: MM6:g
                4.17774e-18   netid: MM6:g
                  9.749e-19   netid: c_9:p
                4.32339e-18   netid: MM5:g
                 4.8594e-18   netid: MM1:g
                1.08097e-18   netid: MM5:g
                6.73195e-18   netid: c_13:p
                9.48261e-18   netid: MM0:g
                1.06043e-17   netid: MM4:g
                4.14482e-18   netid: MM5:g
                4.99255e-18   netid: MM7:g
                2.54537e-17   netid: MM3:s
                3.92759e-18   netid: c_70:n
                 2.3559e-17   netid: MM2:s
                6.89683e-19   netid: MM2:s
                 2.3559e-17   netid: MM2:s
                2.07068e-19   netid: c_76:n
                3.92759e-18   netid: c_76:n
                2.54537e-17   netid: MM0:s
                3.86405e-17   netid: MM7:s
                2.77537e-18   netid: c_90:n
                2.29436e-19   netid: c_90:n
                3.66791e-17   netid: MM6:s
                3.35031e-18   netid: MM6:s
                3.48897e-17   netid: MM6:s
                2.77537e-18   netid: c_94:n
                3.86405e-17   netid: MM4:s
3                     0.0   2.69349e-16       50.1268            0.0  GND
            - Coupled nets
                MM3:g,MM3:g,MM2:g,c_6:p,MM2:g,c_9:p,MM1:g,MM1:g,MM0:g,MM0:g,MM3:d,MM3:d,MM3:d,
                c_43:n,MM1:d,c_43:n,MM1:d,MM1:d,MM7:s,MM4:s
            - Intrinsic Capacitance
                1.34333e-16
            - Coupled capacitance
                8.27127e-18   netid: MM3:g
                1.34578e-18   netid: MM3:g
                1.35056e-18   netid: MM2:g
                2.57507e-19   netid: c_6:p
                2.18914e-18   netid: MM2:g
                4.78238e-19   netid: c_9:p
                2.18914e-18   netid: MM1:g
                1.35056e-18   netid: MM1:g
                 1.4614e-18   netid: MM0:g
                8.27127e-18   netid: MM0:g
                2.54537e-17   netid: MM3:d
                3.92759e-18   netid: MM3:d
                 2.3559e-17   netid: MM3:d
                6.89683e-19   netid: c_43:n
                 2.3559e-17   netid: MM1:d
                2.07068e-19   netid: c_43:n
                3.92759e-18   netid: MM1:d
                2.54537e-17   netid: MM1:d
                5.04182e-19   netid: MM7:s
                5.69945e-19   netid: MM4:s
4                     0.0   3.55081e-16       54.3925            0.0  VDD
            - Coupled nets
                MM7:g,MM7:g,MM6:g,MM6:g,MM5:g,MM5:g,MM4:g,MM4:g,MM7:d,MM7:d,c_43:n,MM7:d,c_43:n,
                MM5:d,MM5:d,MM5:d,MM3:s,MM0:s
            - Intrinsic Capacitance
                1.61944e-16
            - Coupled capacitance
                1.21306e-17   netid: MM7:g
                1.15069e-18   netid: MM7:g
```

```
1.11503e-18   netid: MM6:g
3.18871e-18   netid: MM6:g
3.16413e-18   netid: MM5:g
7.07137e-19   netid: MM5:g
4.92603e-19   netid: MM4:g
1.21341e-17   netid: MM4:g
3.86405e-17   netid: MM7:d
2.77537e-18   netid: MM7:d
2.29436e-19   netid: c_43:n
3.66791e-17   netid: MM7:d
3.35031e-18   netid: c_43:n
3.48897e-17   netid: MM5:d
2.77537e-18   netid: MM5:d
3.86405e-17   netid: MM5:d
5.04182e-19   netid: MM3:s
5.69945e-19   netid: MM0:s
```

Then, we generated a new `config` view for our cell and linked it to the calibre view, which contained the aforementioned net-list with the parasitics. We then set up the old test bench to use this new view and ran the automatic simulations needed to compile the Liberty file and to perform the comparison with the schematic simulations.

In order to fill in the Liberty file faster we developed a Matlab script that converts the traces exported in `.csv` from Virtuoso into a suitable format for the `INV_X4.lib` file. As already mentioned in section 1, this file can be found in the report directory on GitHub or on the server.

We also developed another Matlab script to plot the results of the simulations, shown in the following figures.

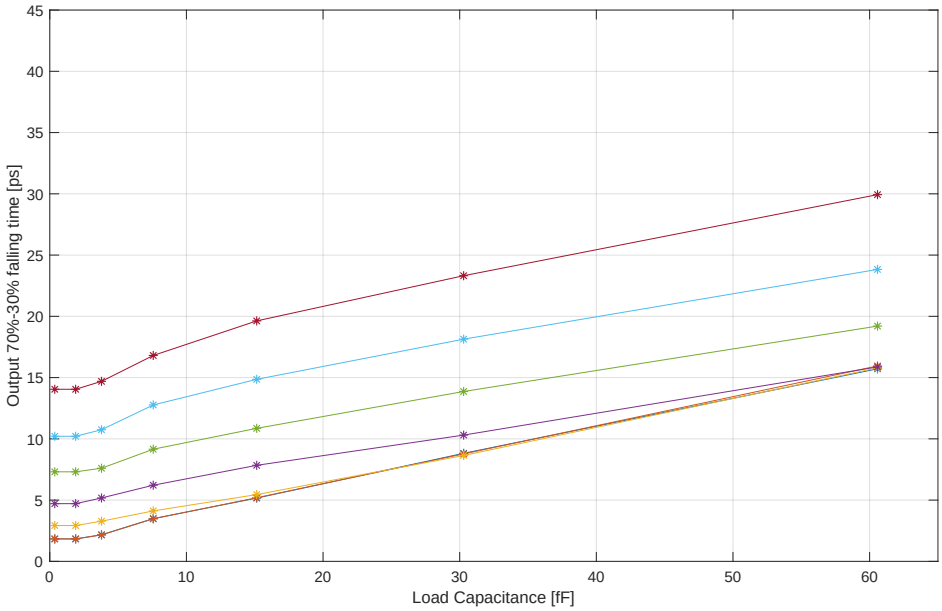


Figure 5: Fall time vs load capacitance, higher curves are for longer input rise times.

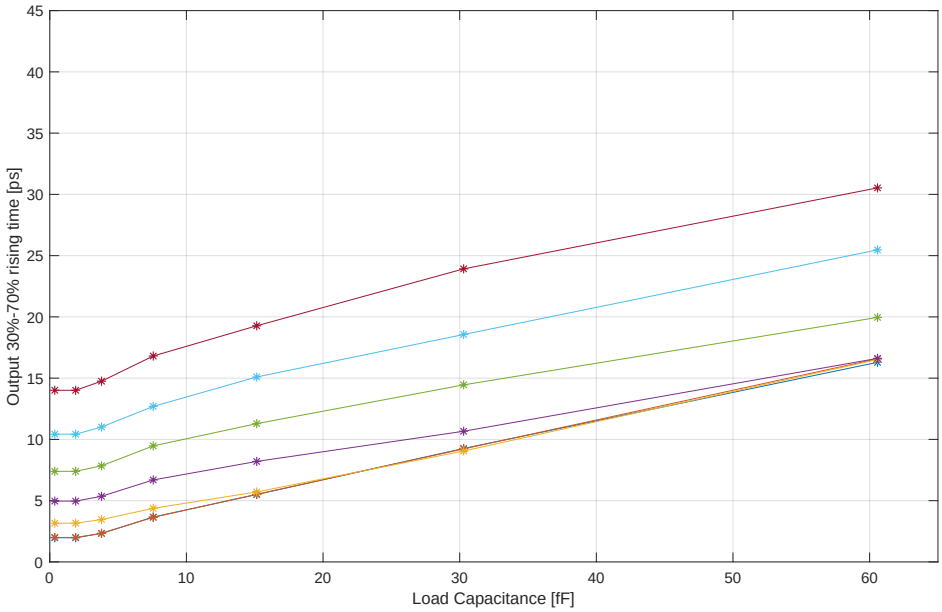


Figure 6: Rise time vs load capacitance, higher curves are for longer input fall times.

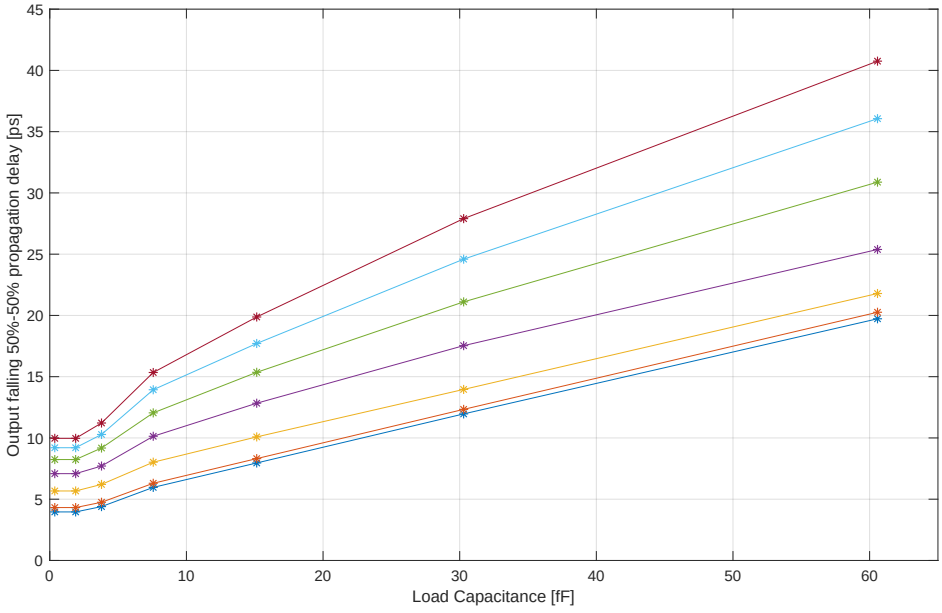


Figure 7: Output falling propagation delay vs load capacitance, higher curves are for longer input rise times.

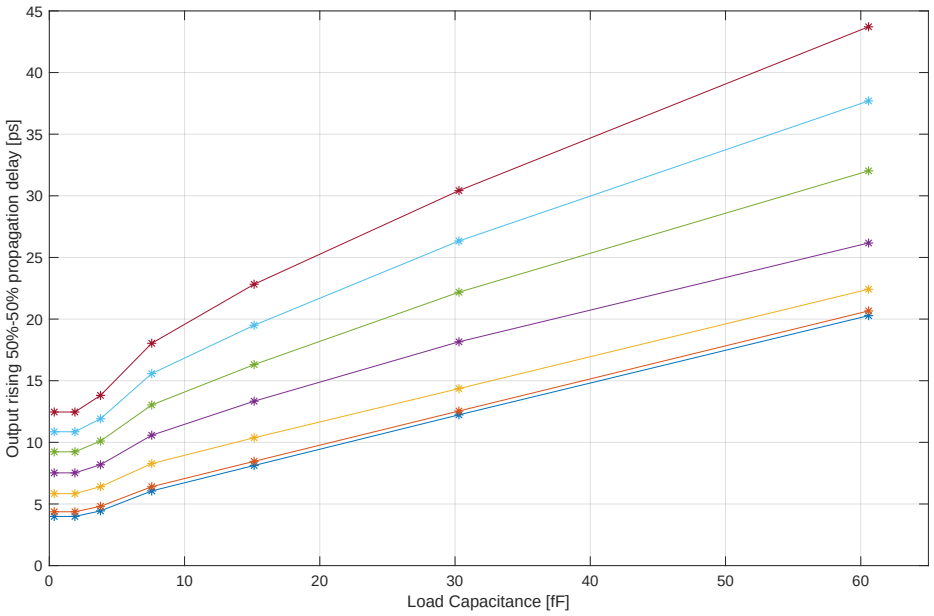


Figure 8: Output rising propagation delay vs load capacitance, higher curves are for longer input fall times.

We also compared these results with the simulation done with original schematic, computing the relative difference in the measured parameters for a given input delay (different input delay values gave similar results and thus were not meaningful to show).

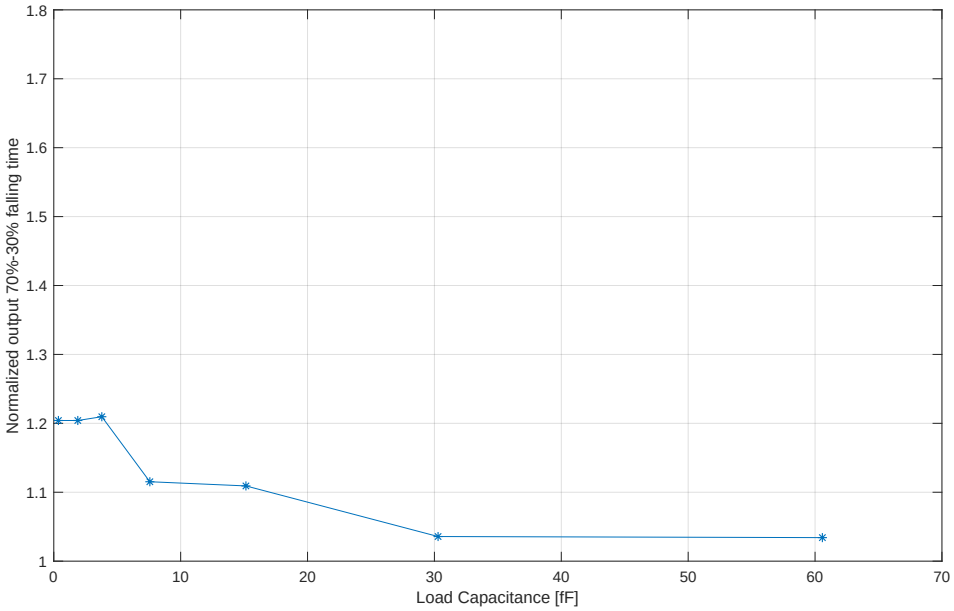


Figure 9: Relative fall time difference vs load capacitance

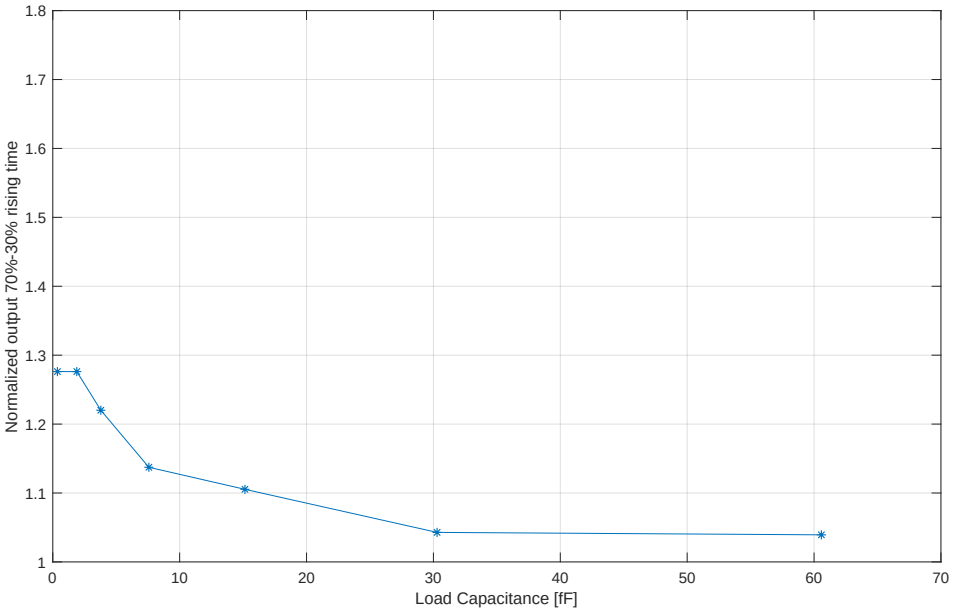


Figure 10: Relative rise time difference vs load capacitance

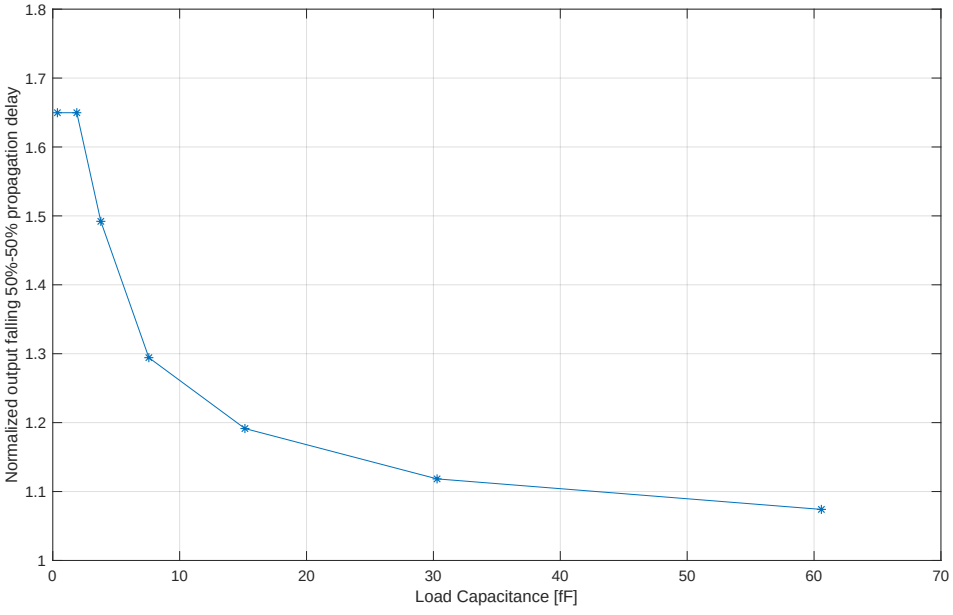


Figure 11: Relative falling propagation delay difference vs load capacitance

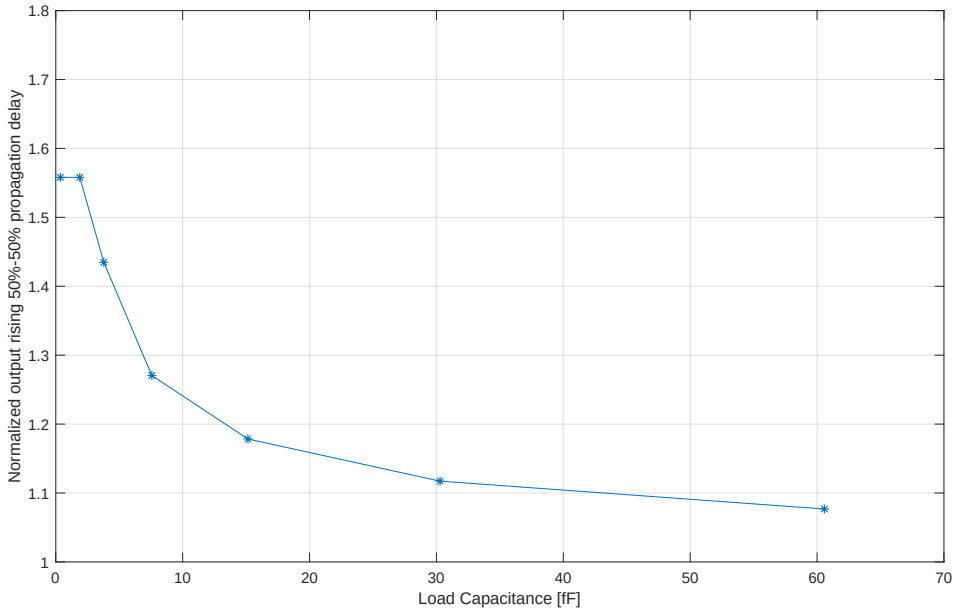


Figure 12: Relative rising propagation delay difference vs load capacitance

We can notice that for very small load capacitance values the simulations done on the extracted parasitics show steadily longer delays, from 20% to 45% longer than the schematic-only simulations. This is easily explained by the fact that the parasitic elements have a bigger impact on performance when the load capacitance is very small, while for larger values of load capacitance its effect becomes dominant and the weight of the parasitic contributions drastically reduces.

3 Half adder

3.1 Schematic

The schematic was drawn using Virtuoso Schematic Editor (figure 14), following the reference `.png` shown in figure 13.

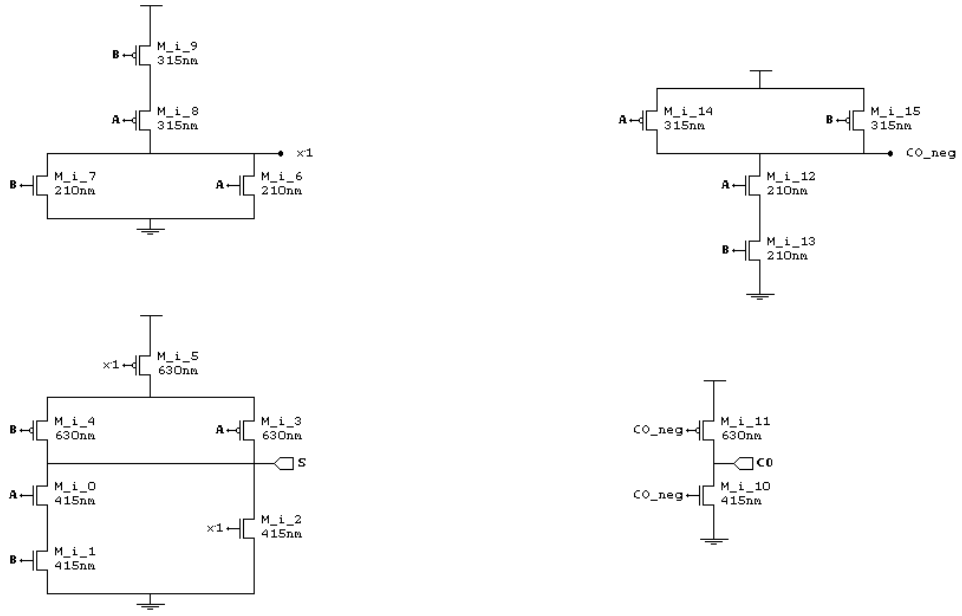


Figure 13: Reference schematic of the half-adder

Even though the AOI21 gate has the pull-up network made of a series PMOS connected through the power supply and two PMOS in parallel, we decided to switch the two subnets to obtain better performances. There are two main reasons for this choice:

1. If A and B are both at '0' and x1 is at '1', the intermediate node is discharged. When a transition of x1 happens then the node has to be charged from only one R_{eq}. With two PMOS between the supply and the other transistor the situation could be hypothetically better: in the worst case we could have the same situation as before, but if both A and B are asserted then the capacitance takes less time to complete the charging.

2. The capacitance of the intermediate node is greater if there are 2 PMOS before the output instead of one.

We report here the final schematic used to create the layout.

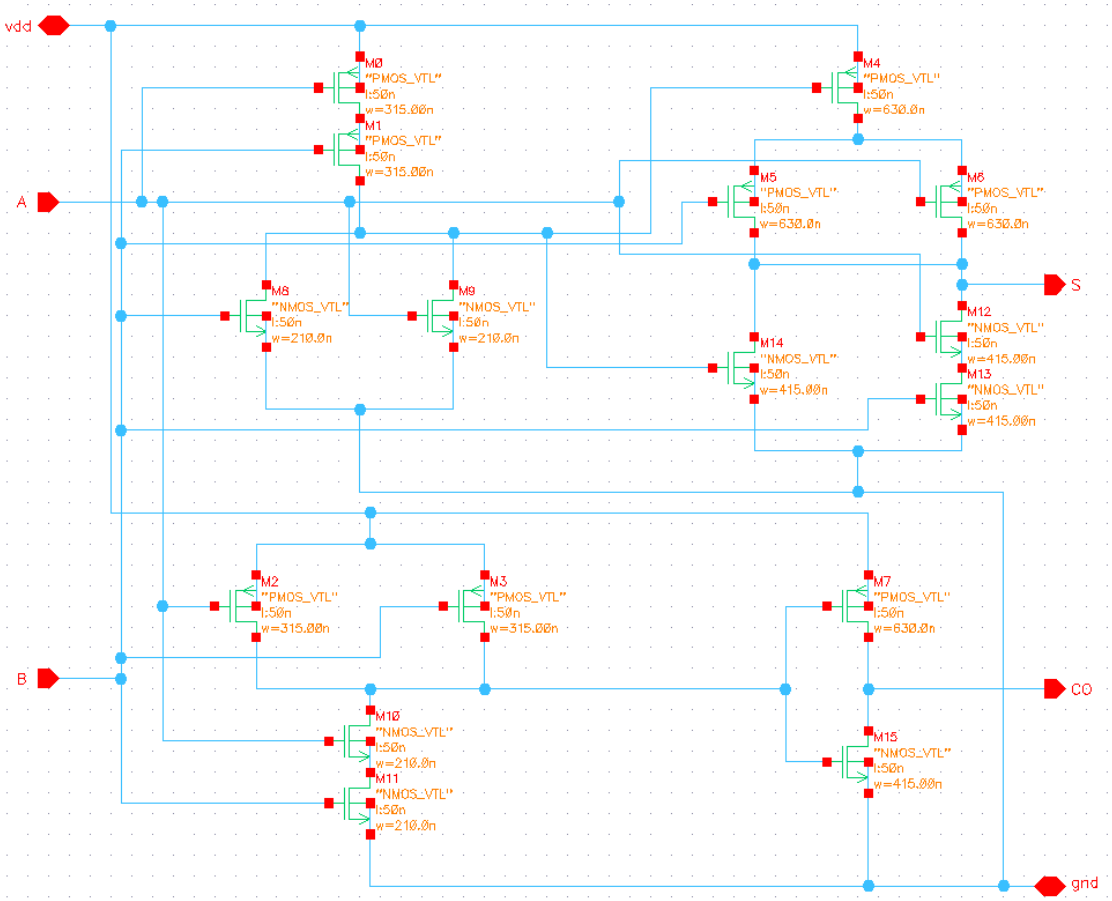


Figure 14: `HA_X1` schematic

At this point we created a new testbench and we simulated the circuit schematic as we did for the `INV_X4`. The cell behaved as it was supposed to. Great!

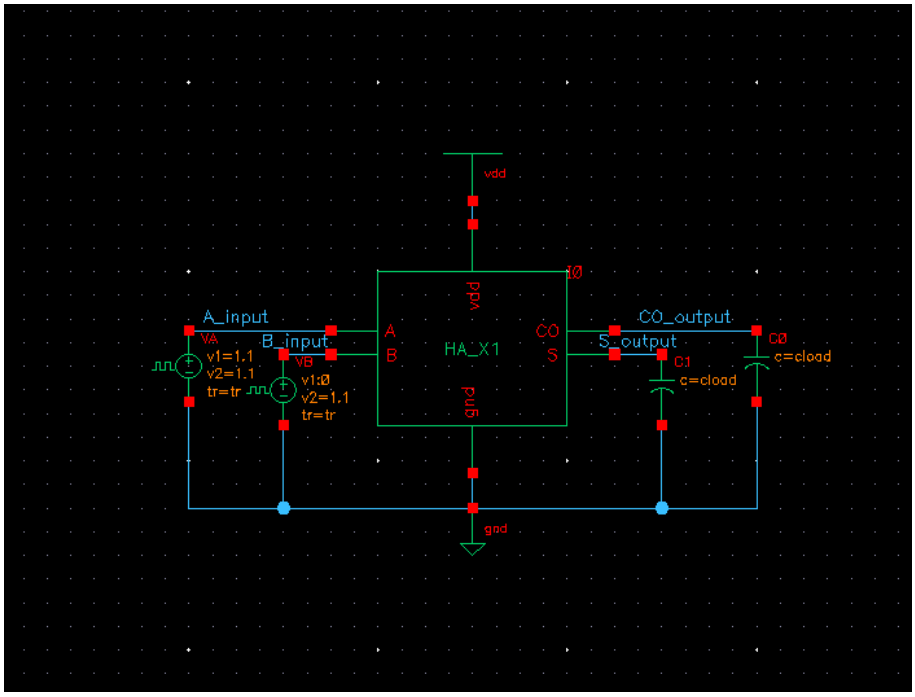


Figure 15: HAX1 testbench

3.2 Layout

Given that the layout of the half adder is quite more complex than the one of the inverter, we started by deeply analysing the possible approaches using the pen-and-paper method. The final solution we came to consisted in sharing source/drain diffusions as much as possible, in order to minimize the area in the horizontal direction, similarly to what we did in the inverter design.

When actually drawing the design in Virtuoso, we made sure to run the Design Rule Checker (DRC) very often, in order to avoid problems in advance. The final result of this tool was totally similar to the one we obtained for the `INV_X4` in section 2.1, and thus it is omitted.

The layout design started by looking at the schematic. First of all we tried to put the two outputs on the opposite sides of the cell. This is because this way, if the inputs are brought over it in the middle with the `metal2` and linked with vias, the input-output paths are more probably balanced in length, keeping the input poly-silicon lines shorter and therefore reducing these lines' propagation delays. This assumption lead us to place the carry-out generation network (a NAND gate followed by an inverter) on the left of the layout, and the sum generation network (NOR gate followed by an AOI12 gate) on the right side. Thus we started drawing on a paper the first NMOS of the inverter whose output was CO. With a view to share the sources/drains as much as possible we began putting all the transistors one near the other drawing only the strictly necessary metal. We used different colours to have an easier readability and go on drawing only the `metal1` for the sources/drains and the `polysilicon` for the gates until the end of the port and left the routing at last. Then we completed the layout linking together the various elements coherently with the schematic view. The resulting pencil draw is reported in figure 16.

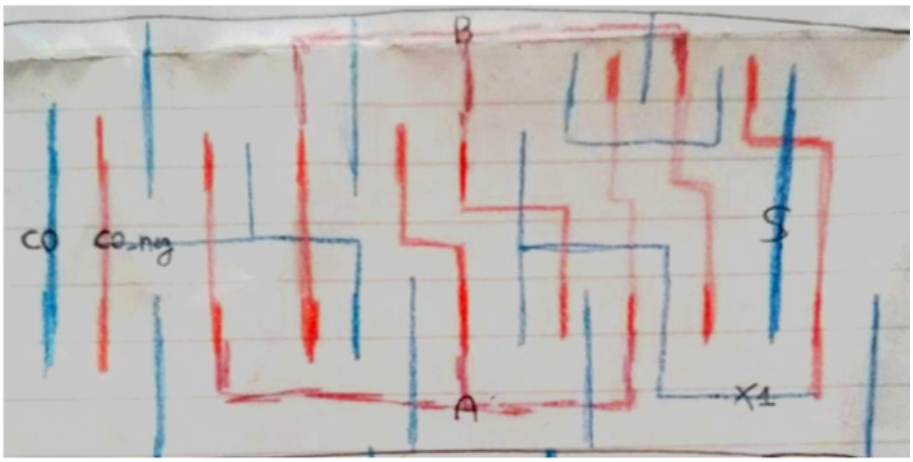


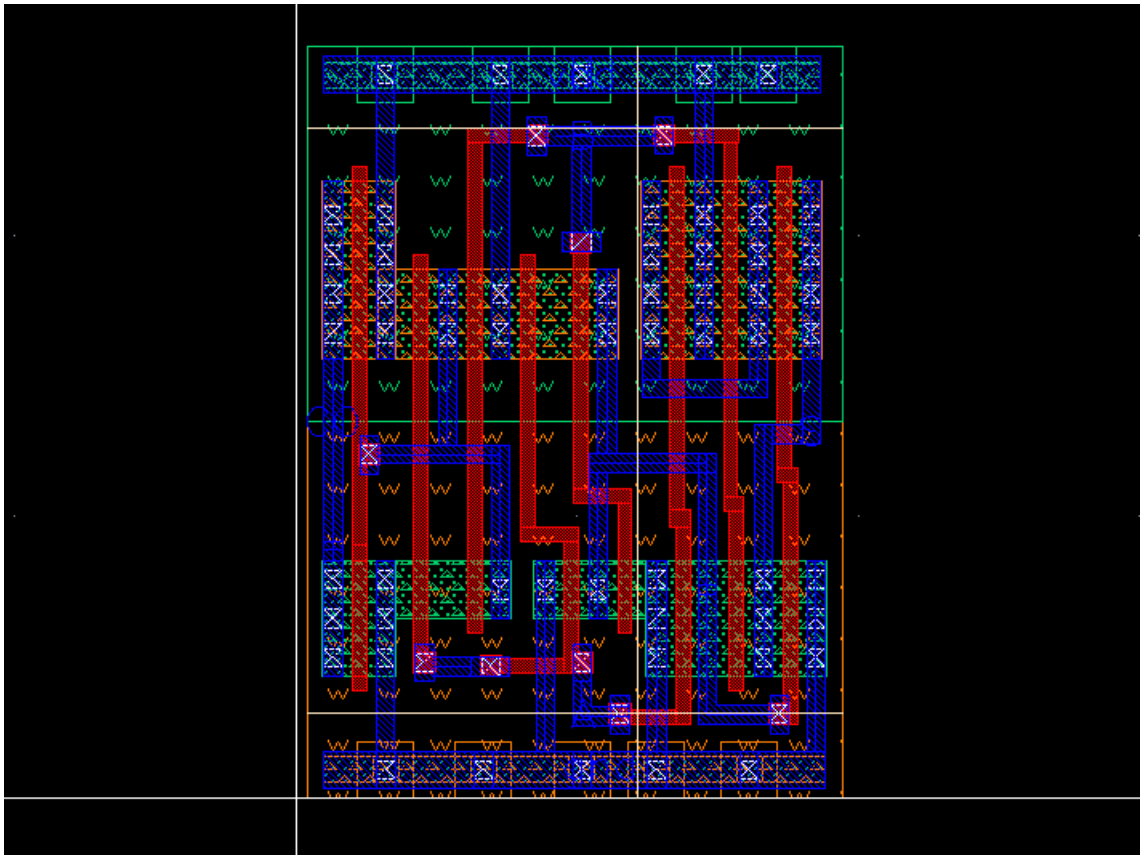
Figure 16: First sketch of the layout. Blue: metal1, red: poly-silicon. Labels do not match the position of the corresponding vias.

Then we converted this sketch in a real layout design using the Virtuoso embedded layout editor. During our sessions we paid attention to reduce every distance to the minimum with the aid of the meter tool and in accordance with the Design Rules. We made a lot of effort to have a cell with the minimum area because the area is directly linked to the yield of the process and to logic ports density, capacitance and therefore delay and power cost per commutation. The less is the area, the more ports can be fitted in a single chip or the more the smaller the chip will be, and it is harder to find defects on a single chip if it is small because the density

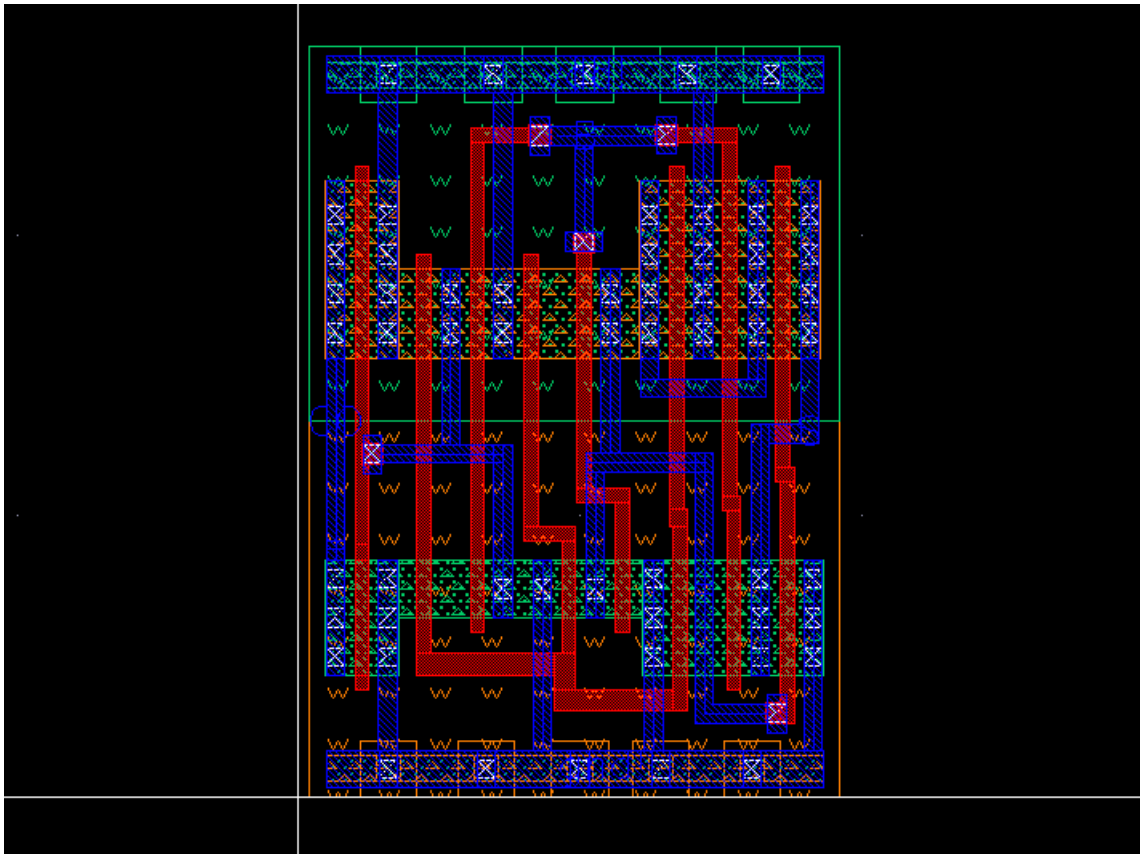
of the defects per unit area is about constant (higher yield for the process). The delay and the power costs rise with the area because if more material is used, higher resistance/capacitance is added. For this reason we also tried to reduce the length of every wire.

The height of the cell is fixed to 2.66 µm because of the standard-cell format requirements. Its top and ground lines, whose purpose is to power the transistors, are made of `metal1`. Many entity like this one can be put one next to the other to form a row of cells packed together and fed with global V_{DD} and GND voltage lines.

When we drew the layout with the editor we changed the connections routing a little bit. Instead of putting a single wire of `polysilicon` to connect each gate with the related input A/B, we used vias and `metal1` pieces of line. It would have been interesting to compare the performances of the two implementations, to see whether longer poly-silicon lines perform better or worse than shorter ones connected through metal lines and vias. Both the layouts are reported in figure 17, though only the poly-silicon and metal version (figure 17a) has been simulated and characterized due to the lack of time.



(a) Polysilicon and metal input lines



(b) Polysilicon only input lines

Figure 17: Different layouts

Notice how the final layout in figure 17a appears divided in two different active region both for the NMOS and PMOS networks. This is necessary whenever two metal lines are connected to the active region without a gate between them. In fact, during the manufacturing process, the poly-silicon layer is lay before the active one. Therefore, the gates prevent the active regions to extend below them, even if the layout masks do. This automatically isolates the drain and source implants and the related metal lines. When no gate is lay between two metal lines, the isolation must be considered as part of the active layer mask. In the early stages of the development this wasn't clear to us, as proved by figure 17b.

Finally we launched the LVS and then extracted the parasitic elements. The PEX operation gave us a schematic with all the MOS with the relative parasitics. Again, the output log and schematic listed all the extracted resistances and capacitances similarly to what has been reported for the `INV_X4` in section 2.3. The net is implicit because no wire is drawn, but every terminal has its own label. The following figures show the group of the 16 MOS of the cell, a detail of one MOS to highlight the labels on the terminals and a parasitic capacitance between the drain of that MOS and `gnd`.

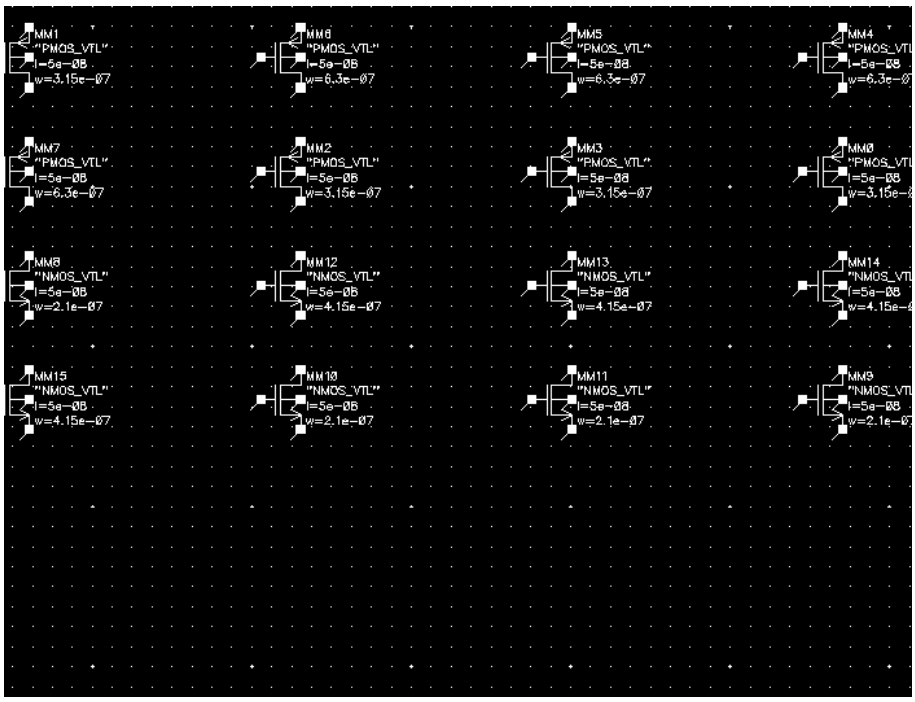


Figure 18: Detail of the extracted parasitics - group of MOS

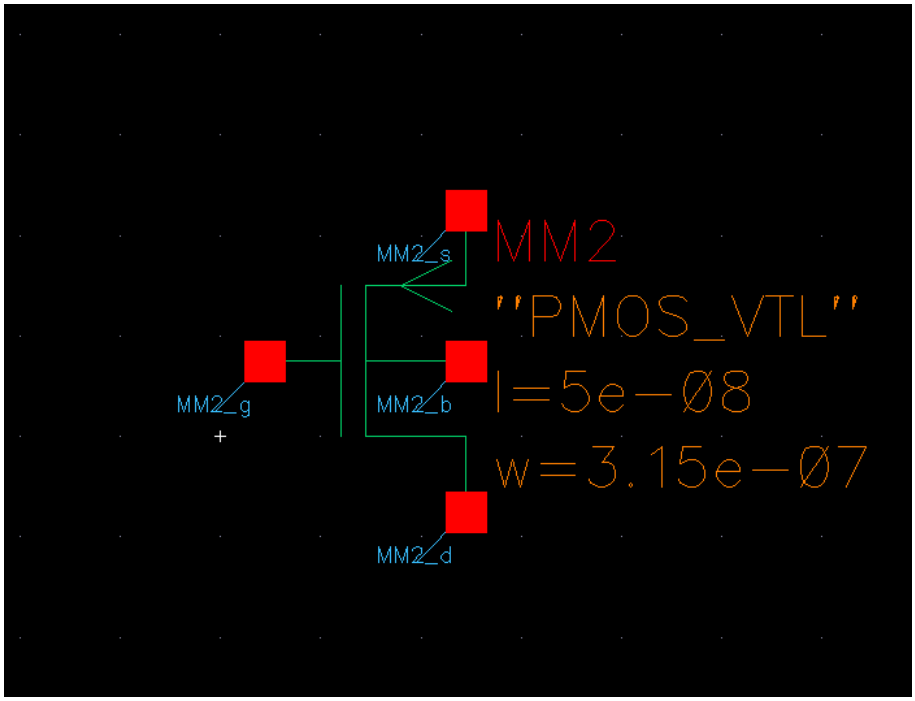


Figure 19: Detail of the extracted parasitics - MOS M2

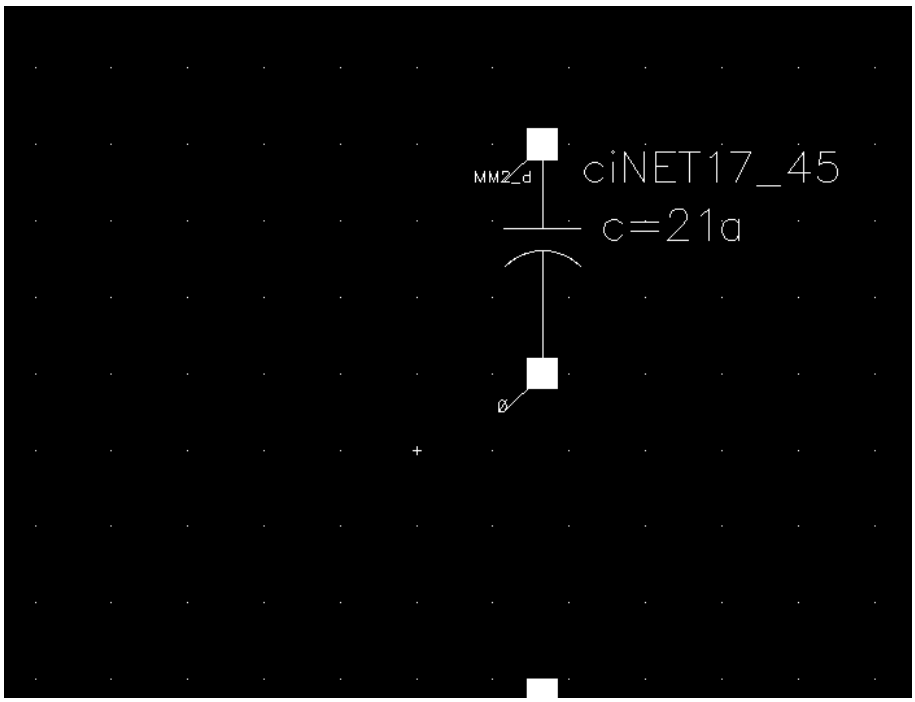


Figure 20: One parasitic capacitance, between the drain of M2 and gnd

Every reference can be spotted looking at the schematic and at the relative file. The PEX produced also the report and the net-list files. In the net-list all the informations about the net with the parasitics can be found. For example here we report the part regarding the previous pictures.

```
102
103 mr_ni "NET17" 287.286 1.72143e-16 2.2726e-16 '( "MM11_d" "MM2_d" "MM3_d" "MM15_g"
    "MM7_g" )
104 mr_pp 'c "ciNET17_41" '("NET17_26" "0") 0.00137581f
105 mr_pp 'c "ciNET17_42" '("c_104_n" "0") 0.00658474f
106 mr_pp 'c "ciNET17_43" '("c_91_n" "0") 0.0028344f
107 mr_pp 'c "ciNET17_44" '("c_102_n" "0") 0.00544915f
108 mr_pp 'c "ciNET17_45" '("MM2_d" "0") 0.0209999f
109 mr_pp 'c "ciNET17_46" '("c_98_n" "0") 0.00629705f
110 mr_pp 'c "ciNET17_47" '("MM11_d" "0") 0.0193463f
111 mr_pp 'c "ciNET17_48" '("MM7_g" "0") 0.0617933f
112 mr_pp 'c "ciNET17_49" '("MM15_g" "0") 0.0474626f
113 mr_pp 'r "rNET17_50" '("c_104_n" "NET17") 3.9
114 mr_pp 'r "rNET17_51" '("c_91_n" "c_104_n") 89.3538
115 mr_pp 'r "rNET17_52" '("NET17_18" "NET17_26") 0.160909
116 mr_pp 'r "rNET17_53" '("c_102_n" "NET17_19") 0.212317
117 mr_pp 'r "rNET17_54" '("c_102_n" "NET17_18") 0.716154
118 mr_pp 'r "rNET17_55" '("NET17_13" "NET17_26") 0.0418175
```

Figure 21: Part of the file 'HA-X1.pex.netlist'

3.3 Characterization

As done before, we proceeded with the simulation of the layout taking into account the lumped parameters extracted with the PEX tool. From the extracted data we created a new cell configuration (`config` view) for the `HA_X1`. At this point we created a new testbench cell, where we imported our `HA_X1` as DUT. Two pulse generators and two load capacitors with parametric values completed the testbench along with the V_{DD} and GND ports.

At this point we were ready to perform the simulations required to fill in the timing characterization tables in the Liberty file. The following simulations are needed:

- **Carry-out** Since the `CO` generation network is an `AND` gate of the two inputs `A` and `B`, it is necessary to have one of these inputs to be fixed at its high value for the output to switch. Therefore only eight measurements are required to fully characterize this output pin: output falling and rising propagation delays, output fall time and rise time, each of these related to both `A` and `B` inputs.
- **Sum** The `S` generation network is a `XOR` gate. Its output can thus switch whenever the inputs change from being equal to being different and vice versa. Therefore sixteen measurements are required. It is important to notice that unlike `CO`, this output shows an inverting behaviour with respect to the related input when the other input is high.

Having set properly the measurement parameters in the ADE L simulation environment we ran the parametric simulation with the same input transition times and load capacitance values used for the `INV_X4`.

Only the results regarding the output falling propagation delays and the output fall times related to the `A` input are reported. The output rising propagation delays and output rising time are slightly higher than these, of about 10 ns to 20 ns, but the behaviour is completely similar. Since the `A` and `B` paths are quite balanced, there's almost no difference between the related timing parameters. Therefore they are omitted for the sake of clarity and mercy for the reader. Instead, both the measurements related to `CO` and `S` are reported one next to the other, in order to make the comparison easier. In figure 22 the output falling propagation delays are shown, while the falling transition times can be found in figure 23.

The propagation delay of `S` is lower than the one of `CO` for very low input transition time and low load capacitance, while it becomes higher at higher values of load capacitance. This behaviour can be explained taking a look at the transistor widths of the two output generation pull-down networks. These are not completely

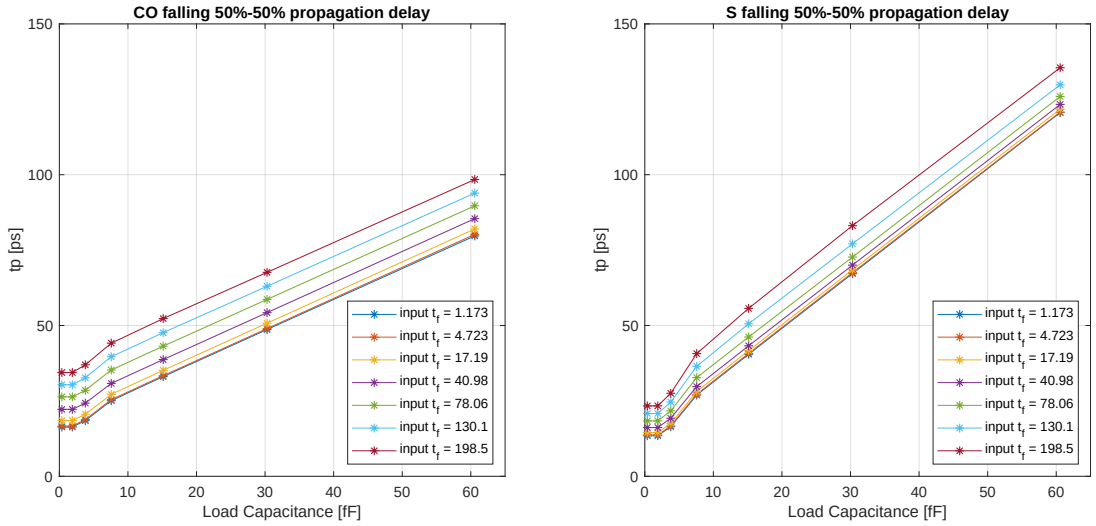


Figure 22: `HA_X1` output falling propagation delays

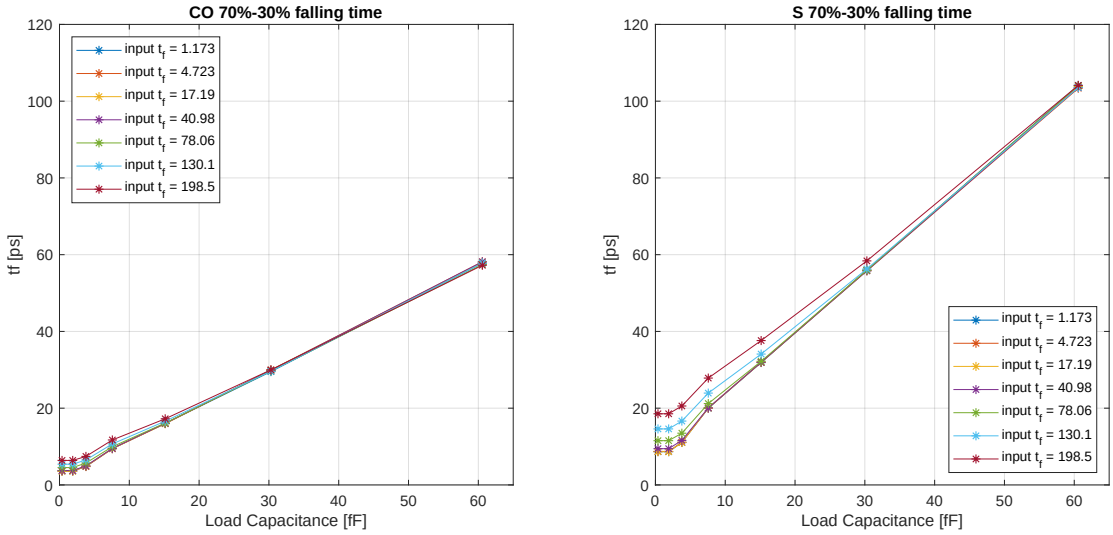


Figure 23: `HA_X1` output falling times

optimized, since two transistors in series have the same width as one transistor alone (see for example the `AOI12` gate in figure 14). Notice also how the outer gates (the inverter for `CO` and the `AOI21` for `S`) have a transistor width which is double the one of the previous gates. This kind of choice is the optimal one to optimize the logical effort of each stage in a sequence of the same kind of logic gates (i.e. an inverter cascade network). In this case, we know that the solution that optimises both the capacitive gain is doubling the transistor dimension every stage. This is not that case, since both the `S` and `CO` network are made up of different logic gates. Therefore both the networks are not the optimal solution in terms of area or propagation delay. On the other side, since the inverter in CMOS technology is the logic gate with the higher capacitive gain, the `CO` generation network is supposed to behave better (lower delays) with higher loads. This is also true for the output transition times, as shown in 23. Another conclusion can be reached from this second plot is that the input transition time has a greater impact with low load capacitances,

while for higher values the slope of the output signals is mainly determined by this last component, as confirmed by the converging curves and by the `INV_X4` analysis discussed before.

Having done the characterization of the `HA_X1` on the basis of its lumped parameters from the designed layout, we are interested in seeing how these affect its behaviour with respect to the schematic-based transistor model. This purpose is better achieved with 3D plots, that can easily show the relation among delay, input transition times and load capacitance. These plots are reported in figure 24 and 25.

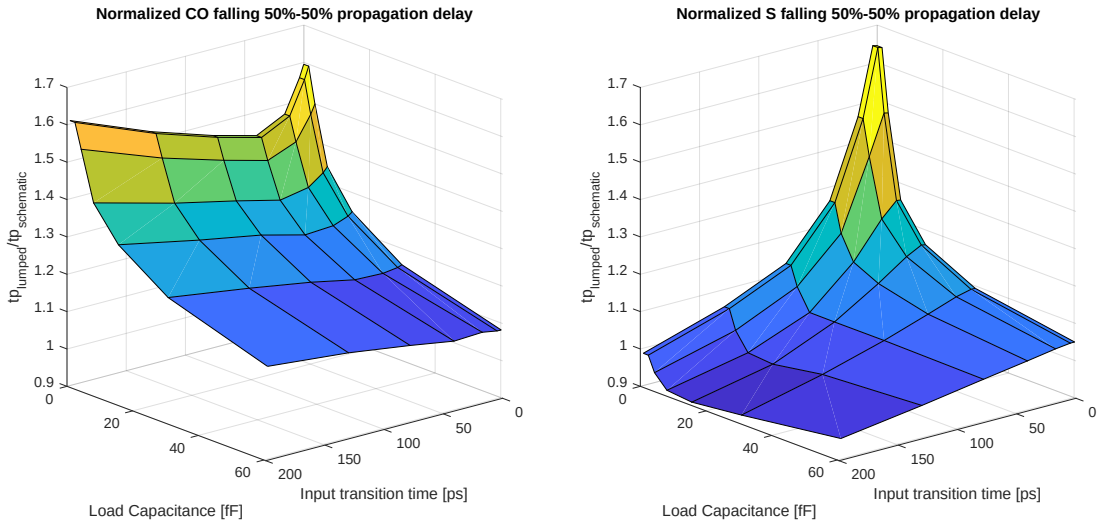


Figure 24: Relative output falling propagation delays vs. load capacitance and input transition time

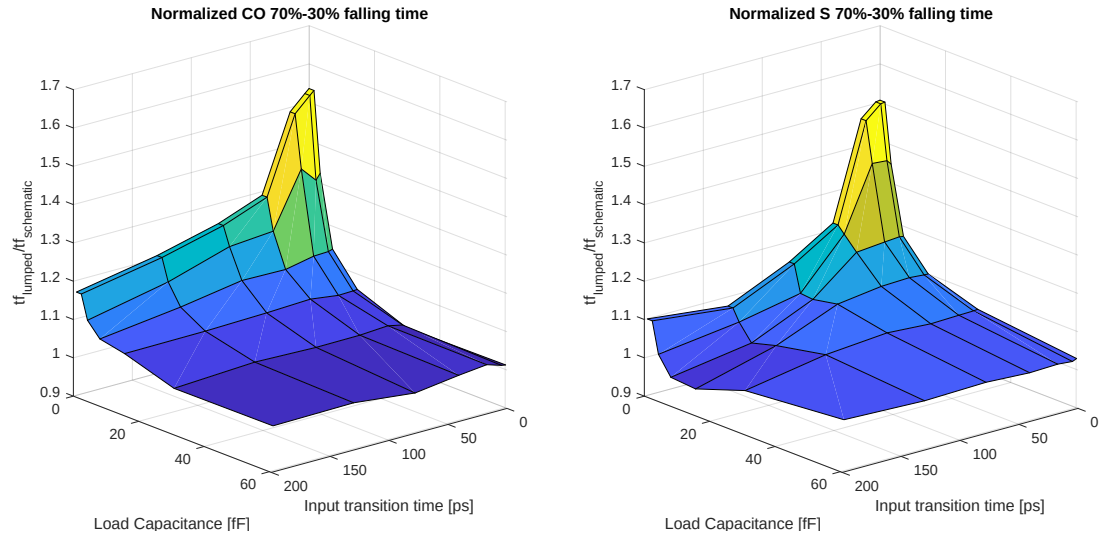


Figure 25: Relative output falling times vs. load capacitance and input transition time

As we expected, the impact of the lumped components is much grater at lower load capacitances and shorter input transition times, where the timing parameters are about 60% greater. However, it is interesting to observe how different this impact is when comparing the two output networks. For `CO` the propagation delay increases with the load capacitance, while it decreases for `S`. This latter case leads the difference to become even negative at the highest load capacitance and lower input

transition time (ratio $t_{p,par}/t_{p,sch} < 1$), where the lumped parameters simulations brought better timing performance with respect to the schematic ones. The reason for this is uncertain, but it can be related to the shared sources and drains of the different transistors, that could lead to overall lower lumped capacitance with respect to the ones taken into account by the schematic simulation models.

If we look at the plots, we can also see how there's no difference between the simulations at the lowest two values of the input signal transition times. This may be due to the simulation resolution.

3.4 Remarks

- **Working frequency** From the timing parameters (see the file `HA_X1.lib` for full characterization) we can see how the propagation delay of the `HA_X1` is always lower than 150 ps. This means that the component can be inserted in a circuit (for example a pipeline stage) that works at a maximum clock frequency of 6.7 GHz, if sequential components timing constraints are ignored.
- **Propagation delay** As it is possible to see in figure 22 the increment in propagation delay is independent on the input rising time. This means that a slower input transition induces an offset on the propagation delay, but the slope of the curve *tp* vs C_L is constant. A longer input transition time makes both the pull-up and pull-down transistor being concurrently opened for a longer time, increasing the delay before the moment in which all the current is directed to the load (because one of those two networks closes). Other than this, if the current is constant, the propagation delay should be almost linear with the load capacitance, as confirmed by the simulation.

4 Conclusions

- **Area** Due to the different input vector used for the characterization of the original library gates it is impossible to directly compare those cells with ours. However, a comparison can be done about the standard cell areas. The standard cell like design fixes the height of the gates to 2.67 µm for both the layouts.
  - For the `INV_X4` we were able to achieve a cell width of 1.33 µm, slightly larger than the 1.0975 µm of the original layout.
  - For the `HA_X1` we managed to achieve much better results, resulting in a width of 1.9025 µm versus the 2.66 µm of the original layout.
- **Further optimization** Better timing performance and smaller areas could be obtained changing the reference design transistor widths according to the best solution for the specific maximum load, setting every single gate width in order to balance the logical effort and driving strength. Nevertheless in this instance we focused on a first level optimization through source/drain sharing and metal/poly-silicon paths routing. The use of `metal1` only makes it possible to employ the upper metal levels to realize the interconnections between cells.