16-bit (4x4) Dual Port SRAM

Benjamin Ryle, Sri Harsha Patnala. Jayadeep Tunuguntla
Department of Electrical & Computer Engineering, NC State University

Introduction

The study of SRAM (Static Random Access Memory) implementation precludes an exploration of the varieties of SRAM cells available for implementation, and an objective for optimization. For the purposes of this design, our primary objective was stability, followed in descending order with density, speed, energy consumption, and design complexity.

The dominant hurdle towards the stability of our design was the read upset problem, when "the output voltage of the cell under a read operation is disrupted from a charge or discharge of the bitline through the NMOS gate connected in series to it."[3] This property of reading our SRAM causing the stored value to corrupt is normally mitigated by a high ratio of the size of the nmos connecting our WBL (word bit line) to our stored value Q, when compared to the PMOS driving the value Q. Additionally though, we found that a high degree of stability could be achieved by introducing a pair of transmission gates to our SRAM design, bringing us from 8 transistors for a standard dual port SRAM to a 10T design .

This effort in the research of stability enhancing designs resulted in a decision to exclude the bitline precharge from our design, as this methodology of removing read issues would be superfluous, and it's exclusion would benefit our second objective of layout density. This finally led us to a non-precharge 10T SRAM cell with TSPC capture flip flops. Our specific 10T SRAM cell design consumes half the amount of power a 6T SRAM cell would during a read operation, given readout data is random and bitline capacitance is constant.[2]

Design description

The 4x4 dual port SRAM presented in this design performs simultaneous read & write operations to different addresses and consists of 4 address locations with each location 4 bits wide. The R/W ports are dedicated exclusively to their assigned operation. Read and Write wordlines (one of RWL0-3 , one of WWL0-3) are selected by Read and Write address decoders based on the address inputs(AW0-1,AR0-1) fed to each decoder.

The logic for selecting the appropriate word line for the write port of the SRAM row resembles a static decoder, with the WENB (active low write enable) signal as the enable pin and the write address lines (AW1, AW0) serving as the inputs. The read decoder implements a similar structure with inputs as address lines (AR1, AR0). The read bit lines are captured using negative-edge triggered flip-flops to hold the read values until the next sequence.

The idea to proceed with the rudimentary static CMOS combinational logic is mainly influenced by the reduced complexity of the design requirements.

With the exclusion of pre-charge for the read bit lines, this design heavily focuses on an accurate representation of a read operation by including flip-flops to hold the values steady until the next read cycle. Thus, it manifests our objectives to choose a TSPC flip-flop which could eliminate clock overlaps rather than a conventional master-slave and also reduces the area with fewer transistors. The flip-flops are negative edge-triggered to allow sufficient time to successfully drive the bit lines with the inputs being sampled during the positive cycle of clock. The gates are minimum sized to allow for more density.

Layout description

Extensive research showed that running the M1/M3 transistors, connecting WWL to the WBL and BLB lines, perpendicular to the SRAM's capture transistors would allow for a much more compact implementation of the SRAM cell, resulting in less wasted area, and allowing the sharing of a gate between both of these transistors. This layout methodology was selected for it's "ubiquitous use in sub 65nm technologies"[1], but upon implementation, found to not be a possibility with respect to either our specific design process. This setback was the most major hurdle to our SRAM design layout. Our solution was to modify the design to have all transistor gates run vertically, which cost us valuable design space.

After this setback, it was found that increasing the sizing of our Word Line PMOS transistors would not cost us extra space due to the loss of density from the previous setback, and would further increase our design stability, so M1 and M3 were run parallel to the SRAM inverter, and increased in size.

We were additionally able to exclude AIL connections from our SRAM cell layout, instead implementing a singular AIL connection at the end of each row of SRAM cells, and have all SRAM cells in a row, plus the inverter the feeding RWLN, share 1 connected NW, NIM, and PIM layer, allowing all of the pmos and nmos transistors for a row of SRAM cells share a singular base connection.

Lastly, research into the 10T SRAM layout found using a minimum of 56nm wide instances of MINT1, and MINT2 lines to supply voltage, ground, and data signals to our design results in a smaller area, lower power consumption, higher stability, and less layout complexity for our SRAM array than an attempt to route signals in a lower overall number of layers would have resulted in.

In the scope of this project, we did not consider dynamic flip flop styles, instead opted for a negative edge triggered TSPC (True Single Phase Flip Flop). This would benefit our primary objective of design stability, because it avoids a common issue for large designs, clock overlap. Considering the size of our complete design, clock overlap would not have been an issue, but at the point of this implementation decision, stability issues that result from a large layout size were still a factor for consideration. It didn't hurt that this decision also marginally benefited our secondary directive of layout area, because the TSPC style flip flop is known to use a lesser number of transistors compared to other styles of flip flop such as master slave.

Results

The end result of this exploration shows a fully operational design layout, with excess design stability resulting in a 120 ps clock period, which, while not outstanding, is a genuine benefit to our design. Additionally, the decision to remove precharge lines for space reduction has resulted in a design with very low power consumption of 0.66 pJ for a supply voltage of 0.8 V. A lack of energy wasted on the creation or purging of pre charge in our word lines, results in a tightly spaced lower power design with moderate speed advantages.

Conclusion

Further improvements could be made towards the minimization of layout area, where the creation an SRAM cell layout where the word line transistors of cells on different rows sit beside each other, would both half the number of AIL connections used on the end of transistor rows, and minimize the mass of empty space on each side of the SRAM's word line transistors caused by the orientation setback.

Finally, we have identified a high threshold voltage, and static decoding logic as the current bottleneck to our design speed, and as such would consider using dynamic decoding to increase our designs speed at the cost of power consumption. Furthermore, we would stand to gain in design speed, stability, and power consumption by attempting to reduce the threshold voltage of the design lower than .8V. Or conversely, a study to derive the most compact decoding logic available for implementation, or the most stable threshold voltage for our design may be in order.

References

R. W. Mann and B. H. Calhoun, "New category of ultra-thin notchless 6T SRAM cell layout topologies for sub-22nm," *2011 12th International Symposium on Quality Electronic Design*, 2011, pp. 1-6.[1]

H. Noguchi *et al.*, "A 10T Non-Precharge Two-Port SRAM for 74% Power Reduction in Video Processing," *IEEE Computer Society Annual Symposium on VLSI (ISVLSI '07)*, 2007, pp. 107-112. [2]

Q. Zheng *et al.*, "Read Static Noise Margin Decrease of 65-nm 6-T SRAM Cell Induced by Total Ionizing Dose," in *IEEE Transactions on Nuclear Science*, vol. 65, no. 2, pp. 691-697, Feb. 2018[3]

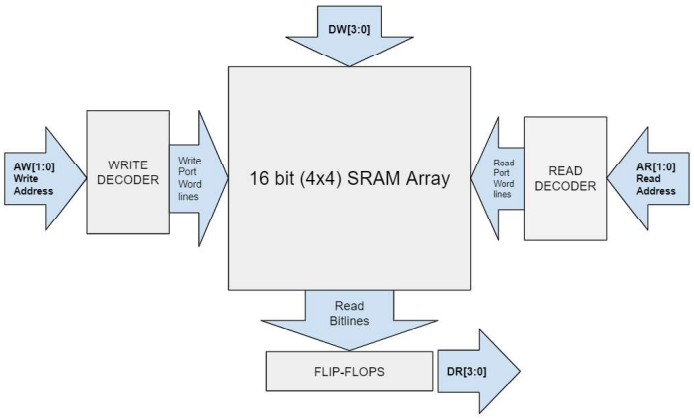

Figure.1 Block Diagram

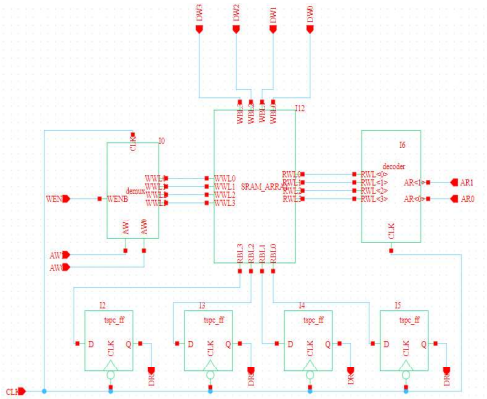

Figure.2 Design Schematic

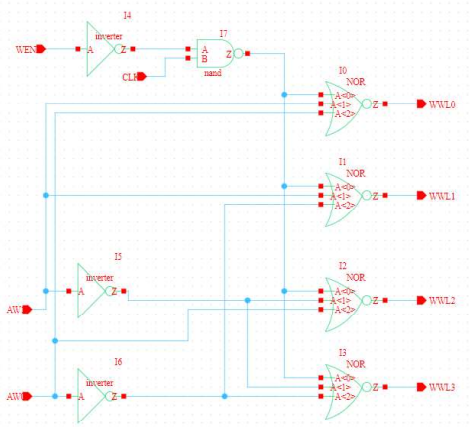

Figure.3 Write Decoder Schematic

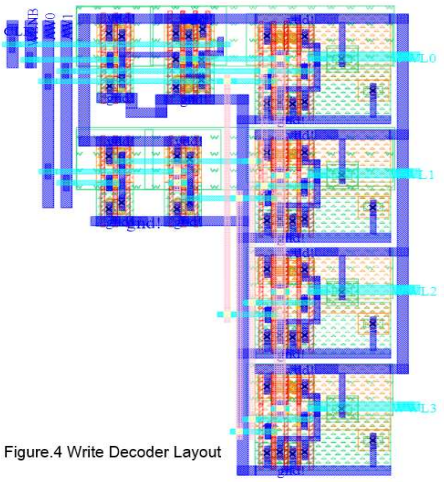

Figure.4 Write Decoder Layout

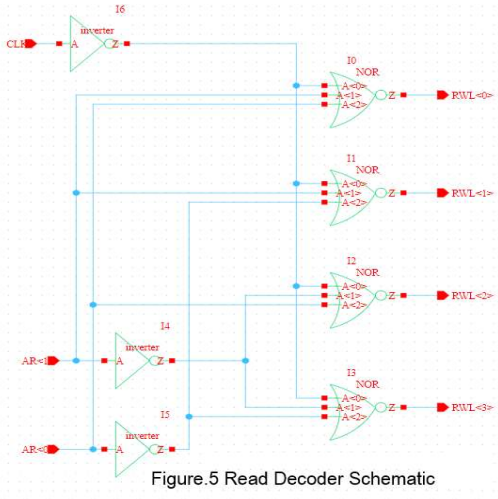

Figure.5 Read Decoder Schematic

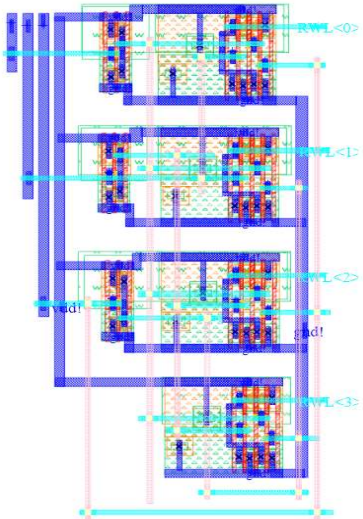

Figure.6 Read Decoder Layout

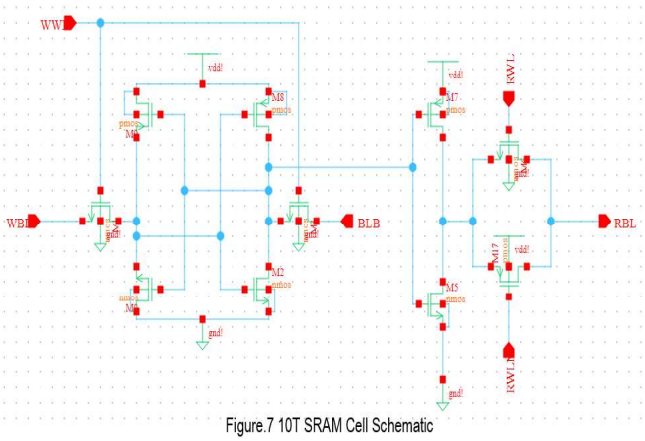

Figure.7 10T SRAM Cell Schematic

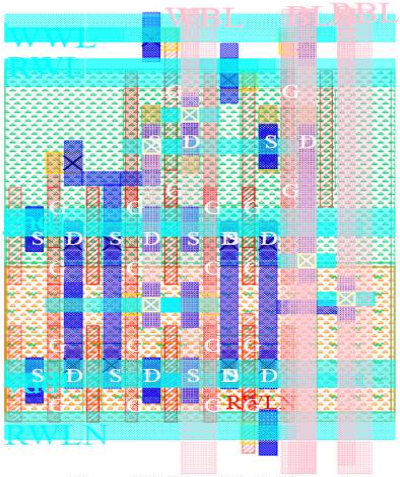

Figure.8 10T SRAM Cell Layout

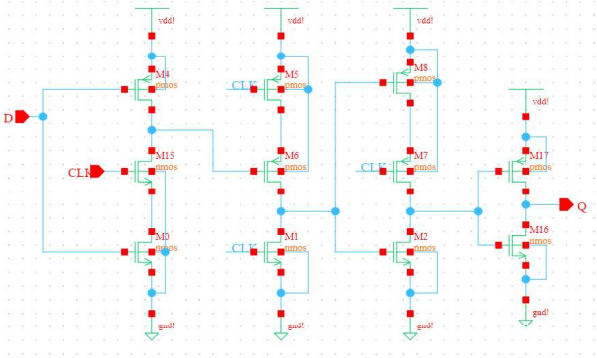

Figure.9 TSPC Flip-flop Schematic

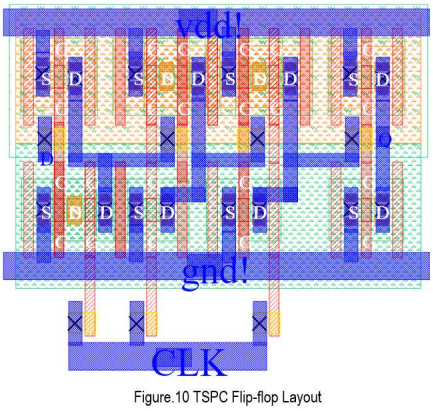

Figure.10 TSPC Flip-flop Layout

Maximum Clock Frequency	8.33 GHz
Supply Voltage	0.8 V
Area	47.2 μm^2
Energy	0.66 pJ
Number of transistors	163
Density (Number of transistors/Area)	3.45 per μm^2
EDA Product	3.73824 pJ-ns-μm^2
Estimated design time	70 hours

Table.1 Design Statistics

Benjamin Ryle	SRAM bit cell & array schematic and layout, area optimizations, report
Sri Harsha Patnala	Write decoder schematic and layout, TSPC flip-flop schematic and layout, overall DRC and LVS checks, report
Jayadeep Tunuguntla	Read decoder schematic and layout, design integration, simulation analysis, DRC and LVS checks, report

Table.2 Workload distribution

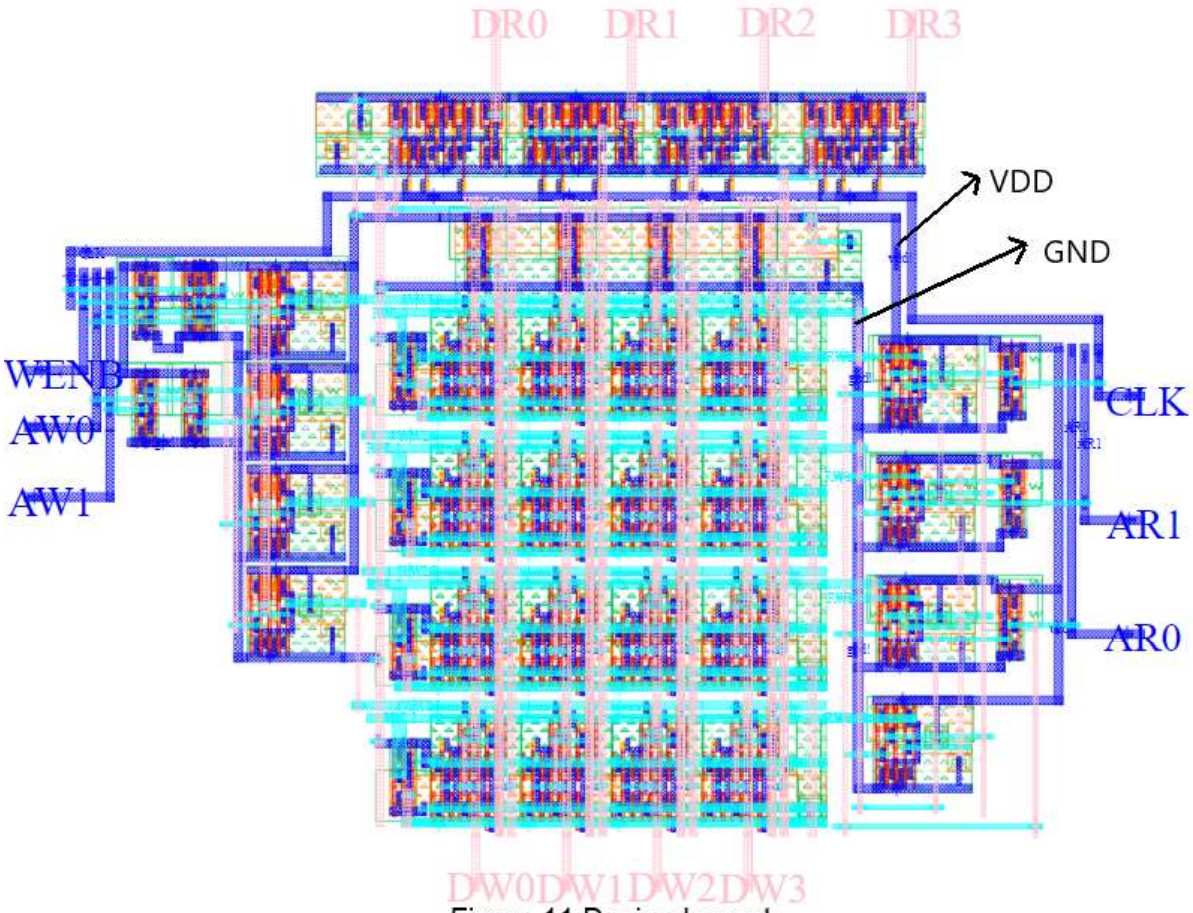

Figure.11 Design Layout

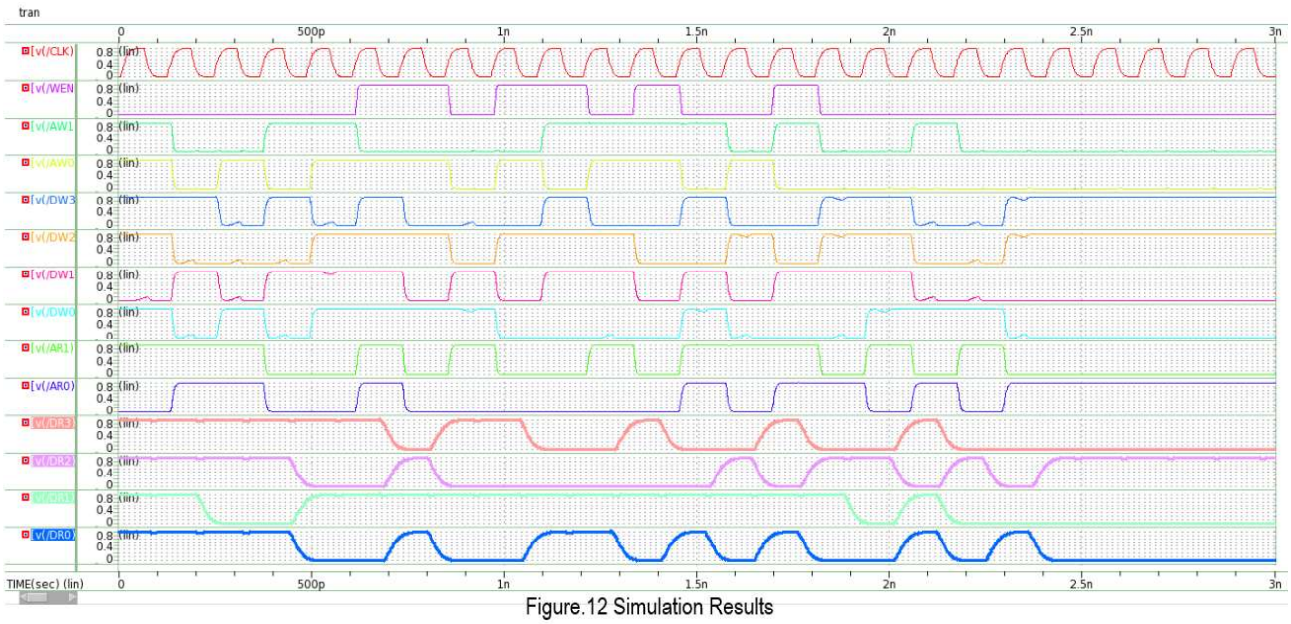

Figure.12 Simulation Results

Appendix

DRC Report Summary

```
                    RESULTS: CLEAN

                 #### #    #####  ###  #  #
                #    #    #    #  # ## #
                #    #    ####  ##### ###
                #    #    #    #  # #  ##
                 #### ##### ##### #  # #  #

================================================================================


--------------------------------------------------------------------------------
ICV Execution
--------------------------------------------------------------------------------


                          IC Validator

              Version R-2020.09 for linux64 - Aug 19, 2020 cl#5783960

                   Copyright (c) 1996 - 2020 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.

Called as: icv -f gdsii -i /afs/unity.ncsu.edu/users/s/spatnal/ece546project/synopsys_custom/memory_design.icv.drc/
memory_design.custom_compiler.gds -c memory_design -oa_dm6 -vue /afs/unity.ncsu.edu/users/s/spatnal/ece546pr
oject/synopsys_custom/memory_design.icv.drc/freepdk15_DRC.drc.rs

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

User name:       jtunugu
Layout format:   GDSII
Input file name:  /afs/unity.ncsu.edu/users/s/spatnal/ece546project/synopsys_custom/memory_design.icv.drc/memor
y_design.custom_compiler.gds
Top cell name:   memory_design
Time started:    2021/04/29 11:40:39PM
Time ended:      2021/04/29 11:40:53PM

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

--------------------------------------------------------------------------------
Results Summary
--------------------------------------------------------------------------------

Rule and DRC Error Summary

699 total rules were run.
0 rules NOT EXECUTED.
0 rules have violations.
There are 0 total violations.
Refer to memory_design.LAYOUT_ERRORS
```

LVS Report Summary

```
+--------------------------------------------------------------------------+
|                 ICV_Compare LVS Comparison Report                        |
+--------------------------------------------------------------------------+

ICV_Compare (R) Hierarchical Layout Vs. Schematic
 RHEL64 R-2020.09.5783960 2020/08/19
Copyright (C) Synopsys, Inc. All rights reserved.

--------------------------------------------------------------------------
LVS error file    = memory_design.LVS_ERRORS
Layout error file = memory_design.LAYOUT_ERRORS
Schematic netlist = /afs/unity.ncsu.edu/users/s/spatnal/ece546project/synopsys_custom/memory_design.icv.lvs/me
mory_design.icv.sp
Layout netlist    = /afs/unity.ncsu.edu/users/s/spatnal/ece546project/synopsys_custom/memory_design.icv.lvs/mem
ory_design.net
Runset file       = /afs/unity.ncsu.edu/users/j/jtunugu/layout1/synopsys_custom/nand.icv.lvs/freepdk15_LVS.lvs.rs
Working directory = /afs/unity.ncsu.edu/users/s/spatnal/ece546project/synopsys_custom/memory_design.icv.lvs
Compare directory = run_details/compare
Compare start time = 2021-04-29 23:41:41


--------------------------------------------------------------------------
Final comparison result:PASS

  #####   ##   ##### #####
  #   #  #  #  #     #
  ##### ###### ##### #####
  #     #    #     #     #
  #     #    # ##### #####


TOP equivalence point:
 [memory_design, MEMORY_DESIGN]

Comparison summary

   3 Successful equivalence points
   0 Failed equivalence points

TOP-level Post-compare summary (* = unmatched devices, nets or ports):

    Matched   Unmatched   Unmatched      Instance types
              schematic    layout       [schematic, layout]
   ---------  ---------  ---------   --------------------
       66         0          0        GATE[_INV(nmos/pmos), _INV(NMOS/PMOS)]
        1         0          0        GATE[_NAND2(nmos/pmos), _NAND2(NMOS/PMOS)]
        8         0          0        GATE[_NOR3(nmos/pmos), _NOR3(NMOS/PMOS)]
        4         0          0        GATE[_S_NMOS2_(nmos), _S_NMOS2_(NMOS)]
        8         0          0        GATE[_S_PMOS2_(pmos), _S_PMOS2_(PMOS)]
       56         0          0        NMOS[nmos, NMOS]
       20         0          0        PMOS[pmos, PMOS]
   ---------  ---------  ---------   --------------------


      163         0          0        Total instances

      103         0          0        Total nets

       16         0          0        Total ports


Schematic and layout agree at all user-intended equivalent points involved
in compare.
```