

Final Project: Configurable Logic Block

Team : Siddhant Mathur and Ruturaj A. Nanoti

1 DESIGN OF 2:1 MUX

1.1 Circuit Schematic

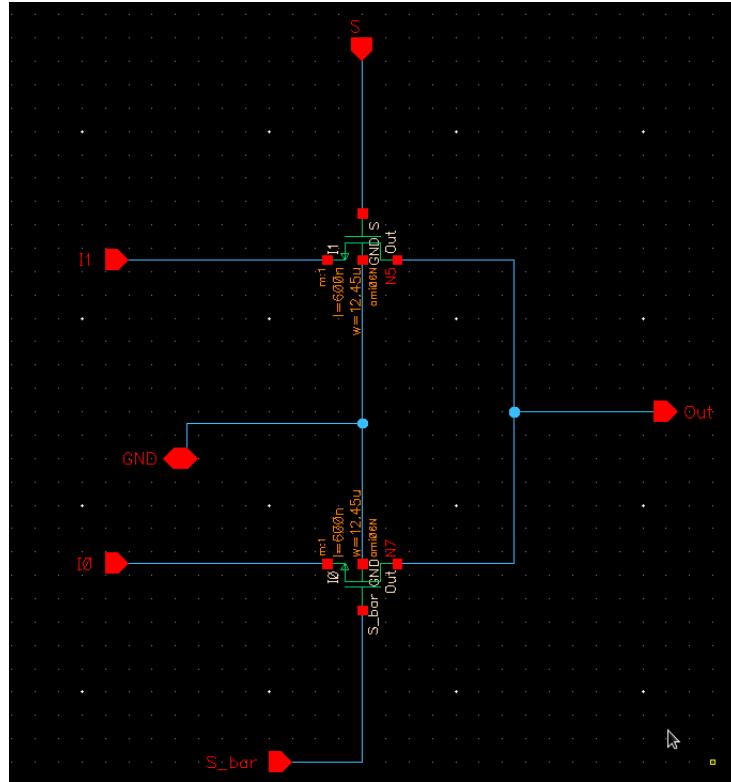


Figure 1: Mux 2:1 - Circuit Schematic

We have used pass transistor logic for our 2:1 Mux design because it uses lesser inverters compared to transmission logic. We have removed the inverter from the select line of the 2:1 Mux and have made it common for the select lines for each of the 4 stages in our 16:1 Mux so that we can cut down on the number of transistors in our 16:11 Mux. This makes our base design relatively more optimized in terms of area compared to having an inverter in each of the 2:1 Mux's.

The optimized sizings we used for the LUT were (we have explained how we arrived at those values in the later sections):

Width of PMOS: 13.5 μm

Width of NMOS: 12.5 μm

These sizings were obtained using a combination of our *Elmore Delay* model for the LUT and the *parametric analysis* performed on Cadence to assist us in reaching the optimal size. The circuit was optimized for delay , while keeping in mind the area that the LUT will contribute to the design of our Configurable Logic Block. Fig. 1 shows the circuit schematic for the 2:1 Mux, which forms a basic block in the design of our LUT.

1.2 Symbol For 2:1 Mux

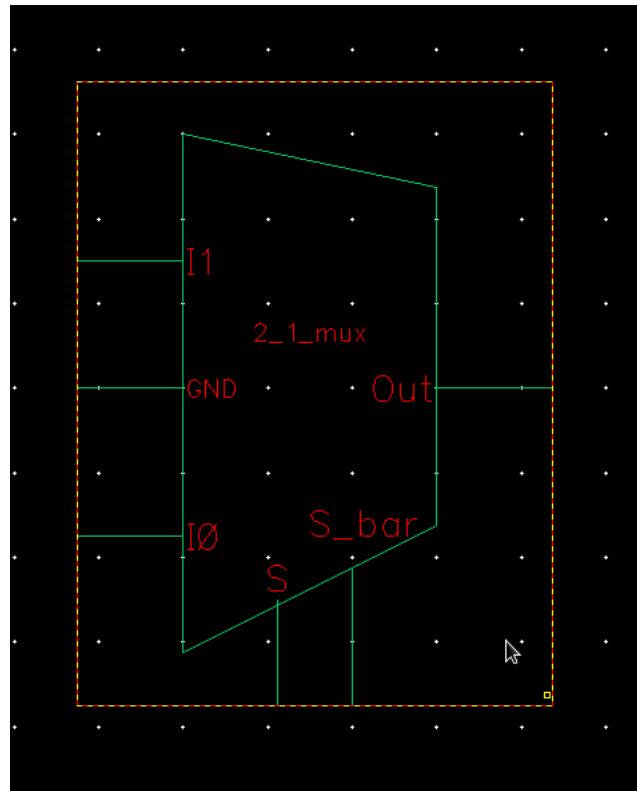


Figure 2: Mux 2:1 - Symbol

1.3 Test Schematic Using Symbol

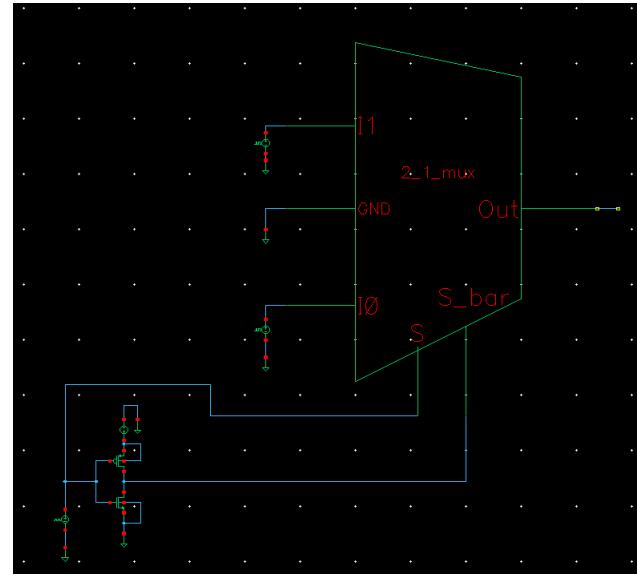


Figure 3: Verification using the symbol

Fig. 3 shows the test schematic that was setup to verify the functionality of our 2:1 Mux. For testing our design, we have cycled through the *address bit* which will make the output follow the input based on the state of the address bit. The inputs are being fed by two different *vpulse* sources, where one has a period that is double the other, that allows us to test the possible cases.

1.4 Functionality Verification of 2:1 Mux

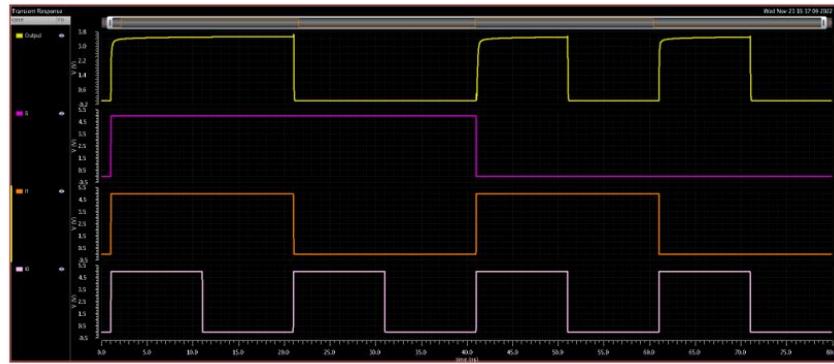


Figure 4: Verification Graph using the symbol

This waveform proves that the 2:1 Mux functions accurately. According to the design of our 2:1 Mux, the output should follow I1 when S=1 and I0 when S=0. This is exactly what is replicated in the above waveforms which confirms the functionality of our 2:1 Mux.

2 16:1 MUX DESIGN USING 2:1 MUX

In the 16:1 Mux design , we have used an inverter for the select lines for each of the 4 stages. This was done to avoid having an inverter in each of the 2:1 Mux's as they would have been redundant in the 16:1 Mux design.

The optimized sizings we considered were (explained in the later sections):

Width of PMOS: 13.5 μ m

Width of NMOS: 12.5 μ m

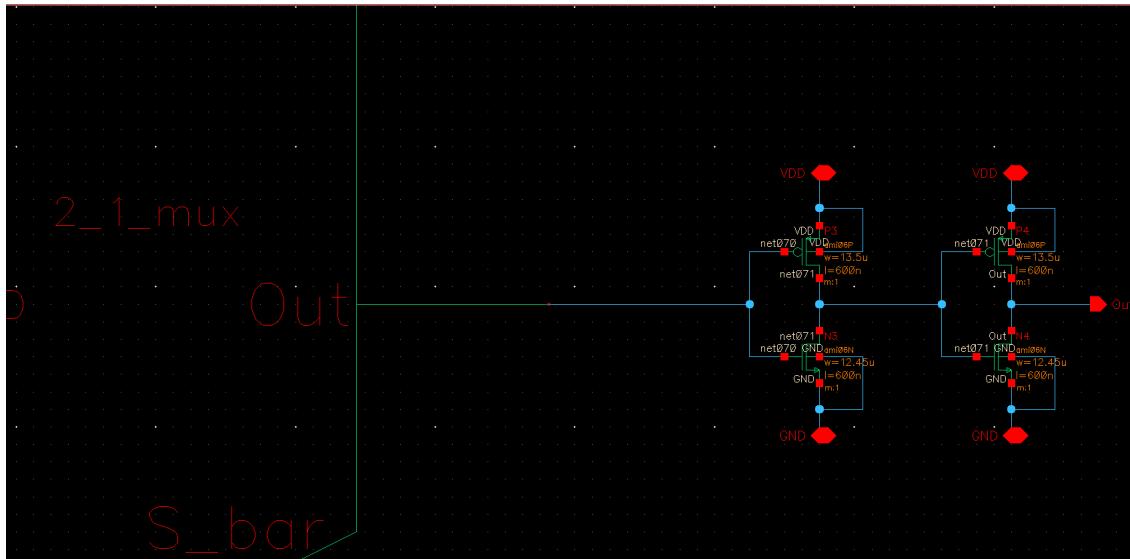


Figure 5: Zoomed in view of the 16:1 Mux

Fig.5 denotes the sizings we have used for the 16:1 Mux as mentioned above. Here , we can see that the sizings have also been maintained for the restoration buffer after the last stage of our 16:1 Mux .This ensures rail to rail output since, the pass transistors cause a V_T drop.

2.1 Circuit Schematic

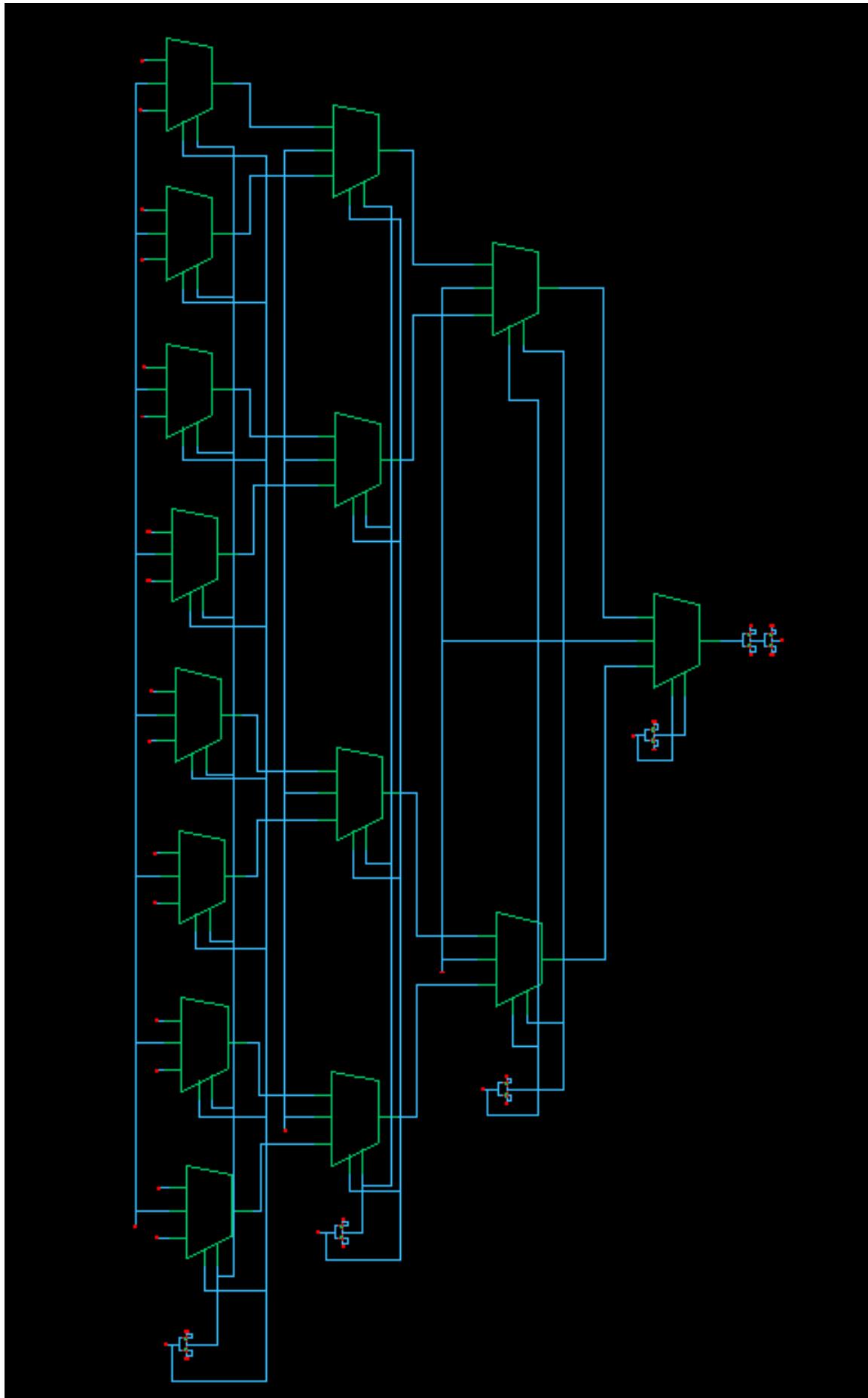


Figure 6: Circuit Schematic - 16:1 Mux

2.2 Symbol for the 16:1 Mux

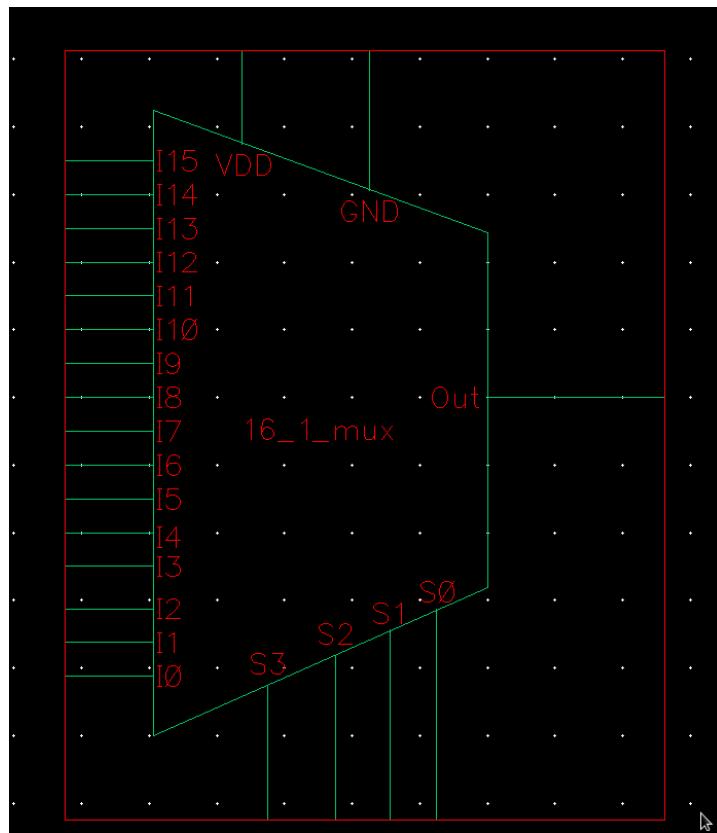


Figure 7: Symbol - 16:1 Mux

2.3 Test Schematic Using Symbol

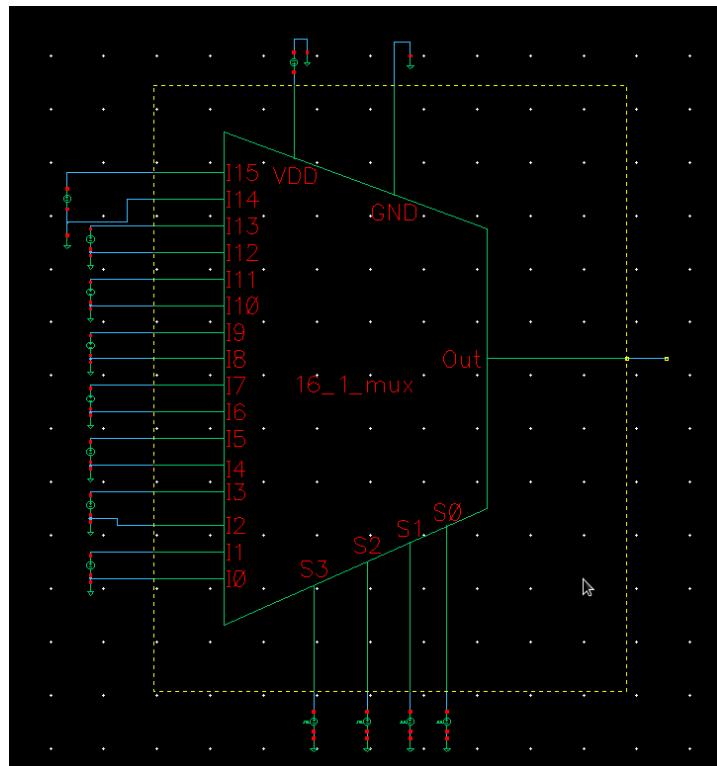


Figure 8: Test Schematic - Symbol 16:1 Mux

Fig. 8 shows the test schematic that we setup to verify the functionality of our 16:1 Mux design. For this, we have connected the alternate bits from I_{15} to I_0 to a 5V supply. This enables us to test switching input cases and verify if our design conforms to all possible input cases.

2.4 Functionality Verification of 16:1 Mux



Figure 9: Functionality Verification Graph

As can be seen in the verification plot (Fig. 9), the inputs are alternating between 1 and 0 and the address bits are being cycled through with different pulse widths and periods to test for all possible 16 cases. Thus, the output follows all the bits at the input and verifies the working of our 16:1 Mux.

2.5 Maximum Worst Case Delay

We measure the delay between the S_3 select line and the output before the load. The S_3 select line (for the leftmost stage) is our LSB and the S_0 select line (for the rightmost stage) is our MSB. We have identified the case that contributes to the maximum worst case delay as when S_3 is switching and the other 3 select lines are kept at a logical high (VDD).

This would mean that the output keeps toggling between In_{14} and In_{15} since the select lines would either be 1110(In_{14}) or 1111(In_{15}).

We have switched S_3 since this select line is driving the most number of 2:1 MUXs and would result in higher parasitic capacitance as compared to the other 3 stages. Thus, it makes sense to switch the select line(S_3) that has to drive the highest load and is associated with the highest capacitance. The inputs to all the select lines are given through inverters and the out pin of the 2:1 Mux is connected to In_{15} of the 16:1 Mux.

2.6 Parametric Sweep Graph

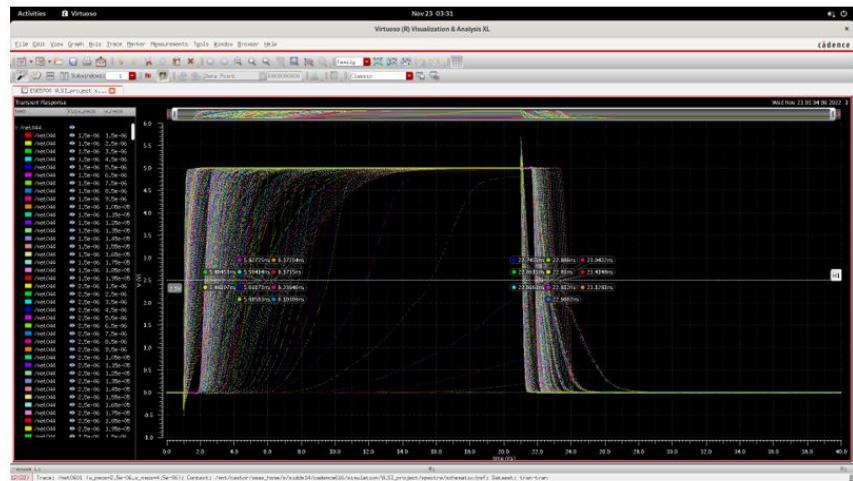


Figure 10: Overall Parametric Sweep

2.7 Delay Using Parametric Sweep



Figure 11: Delay using chosen sizing from Parametric Sweep

For the PLH and PHL values,

$$\tau_{PLH} = 2.3561ns - 1.20239ns = 1.15371ns$$

$$\tau_{PHL} = 21.9131ns - 21.1371ns = 0.776ns$$

So, the worst-case delay comes out to be **1.15371 ns**.

The optimized sizings we considered were:

Width of PMOS: 13.5 μm

Width of NMOS: 12.5 μm

The sizing for the inverter that drives the our 16:1 mux for the worst delay case, the transistors for 2:1 mux and the buffer used for restoration at the end before the load are as mentioned above. The inverters driving the select lines of our DUT need to have the same drive strength as a 2:1 mux; the buffer used for restoration also needs to be sized the same since it is being driven by a single 2:1 mux.

The transistors in the inverters of the load are sized as to act as a $100C_g$ load.

Width of the PMOS in the inverter of the load: 75 μm

Width of the NMOS in the inverter of the load: 75 μm

We could have reduced the worst-case delay further by increasing the sizes of the PMOS and NMOS even more, however, we did not prefer increasing the sizes beyond a point as that would lead to an increase in the total area. Thus, we settled at the above mentioned sizings for our design.

3 SRAM CELL

3.1 Circuit Schematic

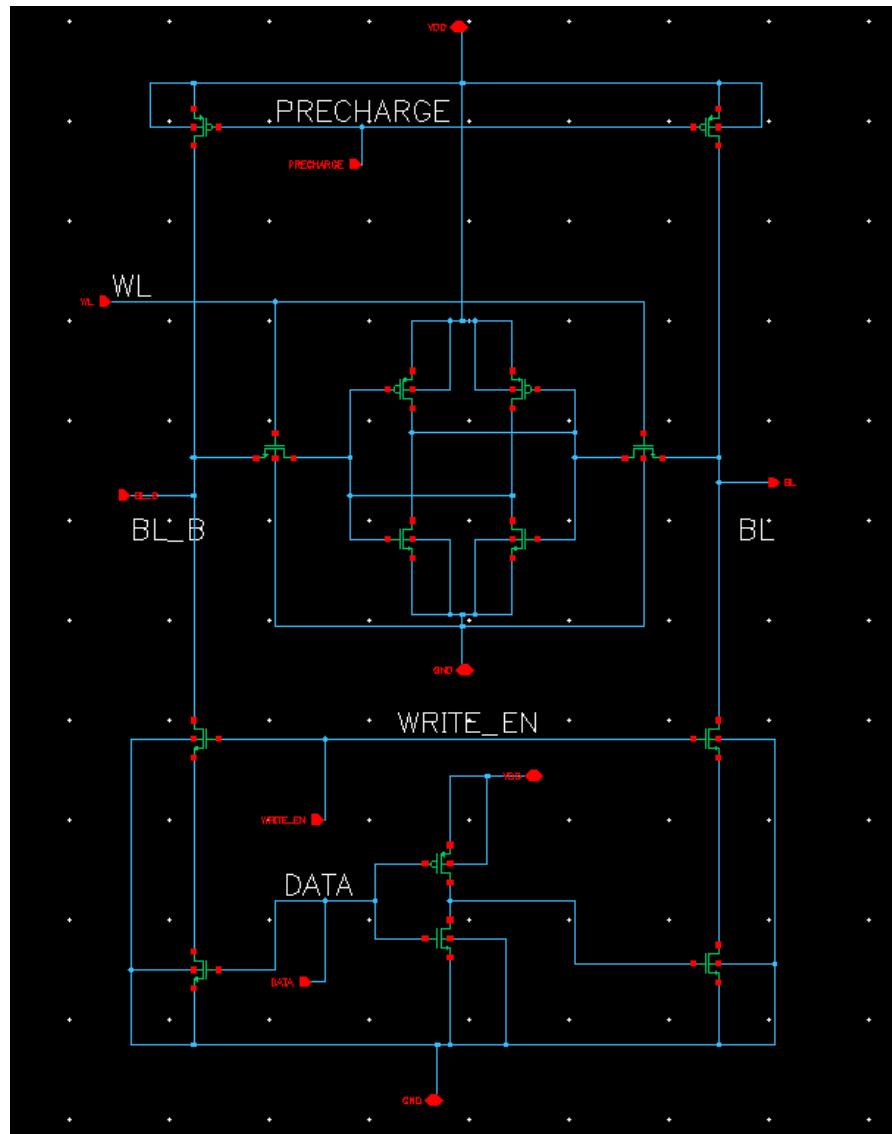


Figure 12: 6-T SRAM Cell

We have designed a 6-T SRAM cell which stores data in cross-coupled inverters and is used in most commercial chips.

3.2 Proof of sizings

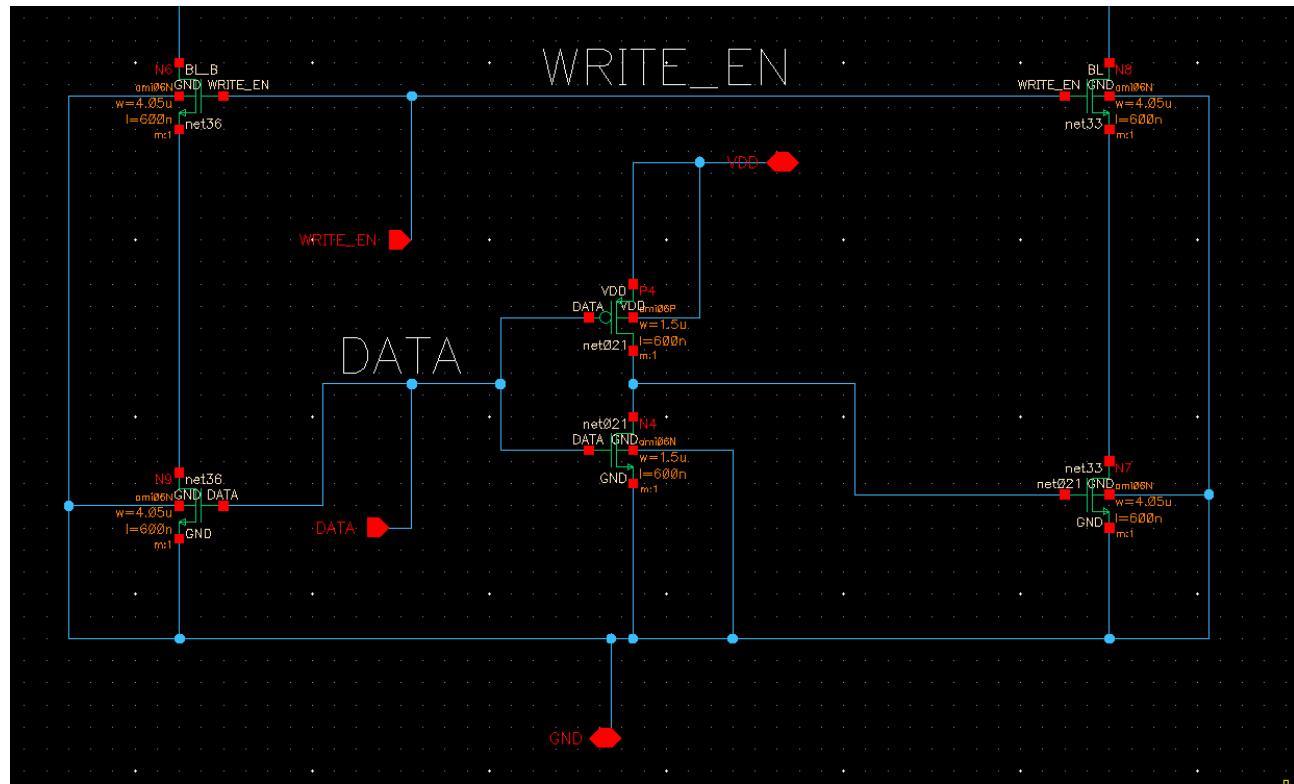


Figure 13: Zoomed in View of the SRAM Cell - 1

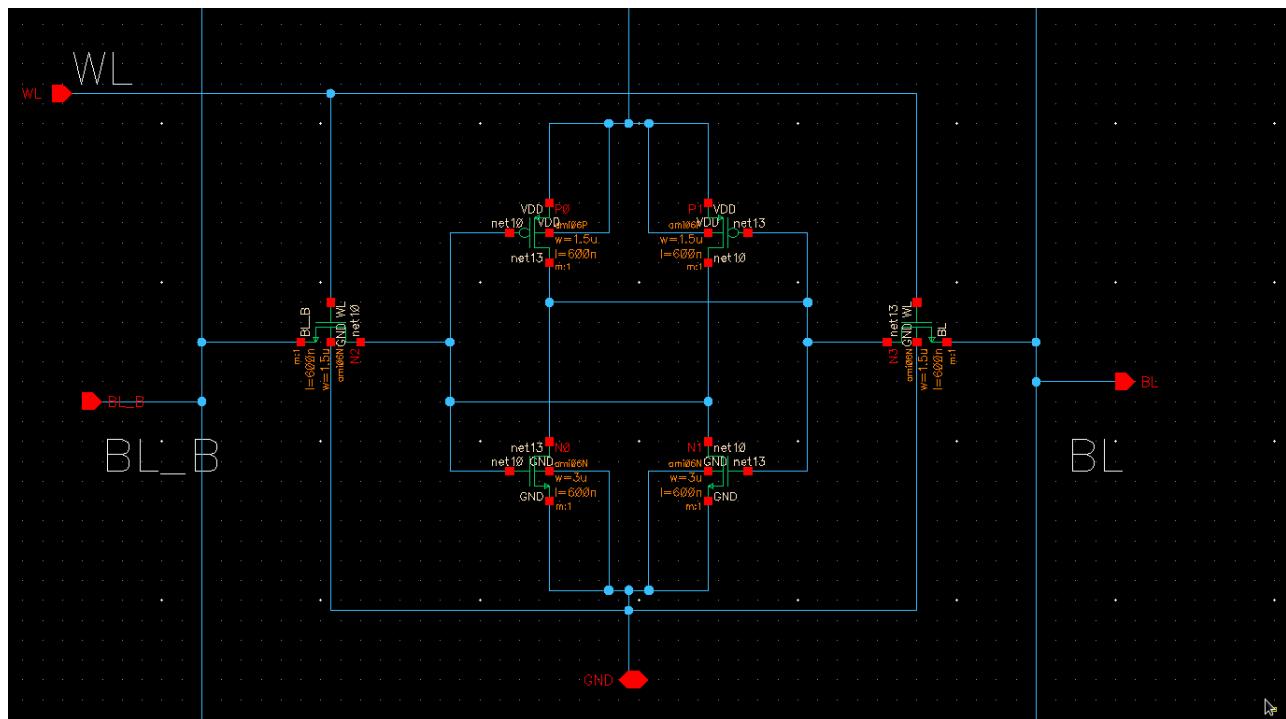


Figure 14: Zoomed in View of the SRAM Cell - 2

3.3 Sizings used for the SRAM cell

We have designed a traditional 6-T SRAM cell and set the size of the NMOS' in the cross coupled inverters to $3\mu\text{m}$.

Shown below is a representation of the 6-T SRAM :

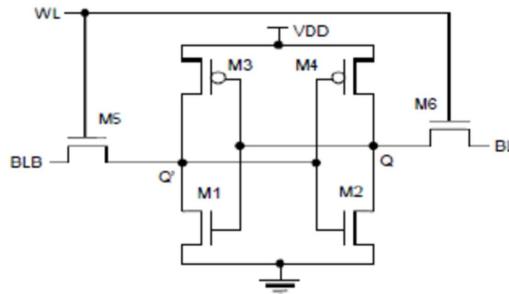


Figure 15: 6T - SRAM Cell

We have set the sizes of the NMOS M_1 and M_2 as $3\mu\text{m}$ because it is important for the NMOS M_1 to be stronger than M_5 as during discharging , the bitline bar discharges through M_5 and M_1 . Thus, we have sized the M_1 NMOS to be greater than NMOS M_5 . The same applies to the NMOS' M_2 and M_6 where $M_2 \gg M_6$ for the above mentioned reason. This is done to ensure read stability.

The reason why we did not increase the sizes of the NMOS' of the cross coupled inverter beyond $3\mu\text{m}$ is that all bit cells of a single SRAM cell should ideally be small to make sure that the memory cell does not end up becoming too bulky. The NMOS' in the write circuitry are set to $4\mu\text{m}$ to ensure write stability. These sizings were not arrived at through parametric analysis but through ratios suggested for the general functionality of a 6-T SRAM cell.

3.4 Symbol



Figure 16: 6T - SRAM Cell Symbol

3.5 Testing of a single SRAM cell

The SRAM cell is tested with a pattern that makes sure the data stored in the cell does not get corrupted when performing a read. This is tested when storing both '1' and '0'. A write is performed by driving the data onto the bitline and then raising the word line, and then a read operation is performed twice to make sure that the read doesn't corrupt the data.

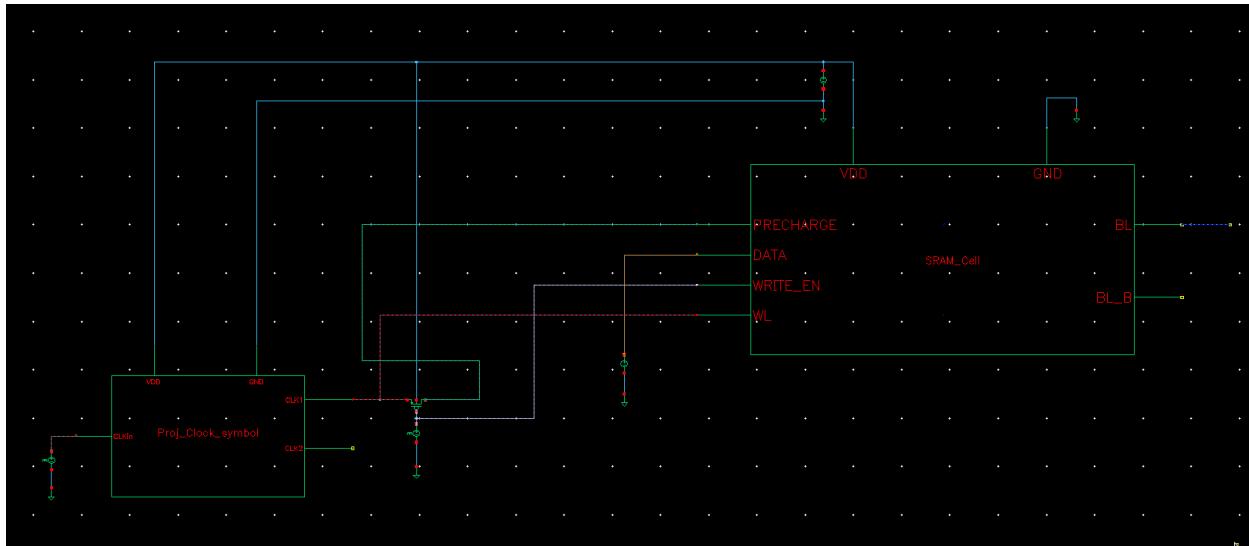


Figure 17: 6T - SRAM Cell Circuit Schematic : Testing with storing "1"

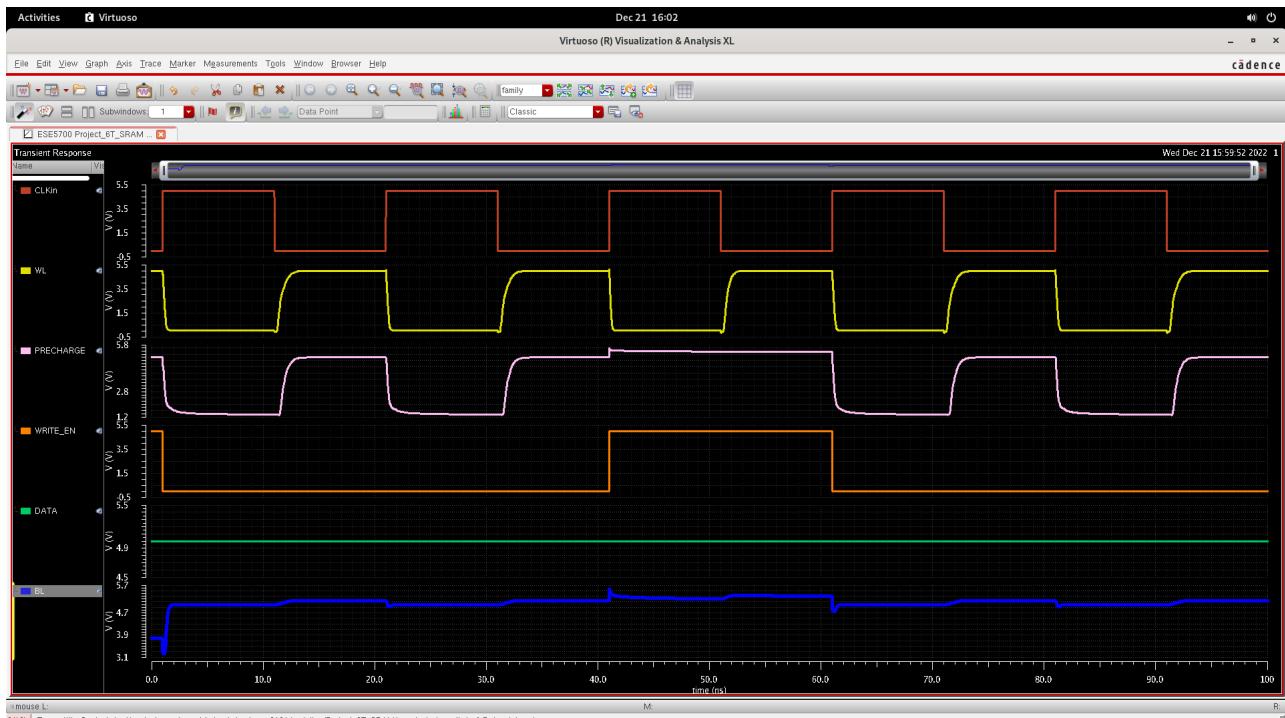


Figure 18: Verification Graph - 1

Fig. 17 and 18 depict that our SRAM is functioning just as expected. In this, we performed a read operation twice (for testing the nature of the read operation - destructive or non-destructive) and then wrote a 1 into the cell by driving the data onto the bitline, and then raising the word line. Upon writing a 1 into the cell , we then perform the read operation twice to make sure that our read is non-destructive, i.e. the data stored in the SRAM cell is not corrupted. The BL waveform shows that the SRAM is working properly.

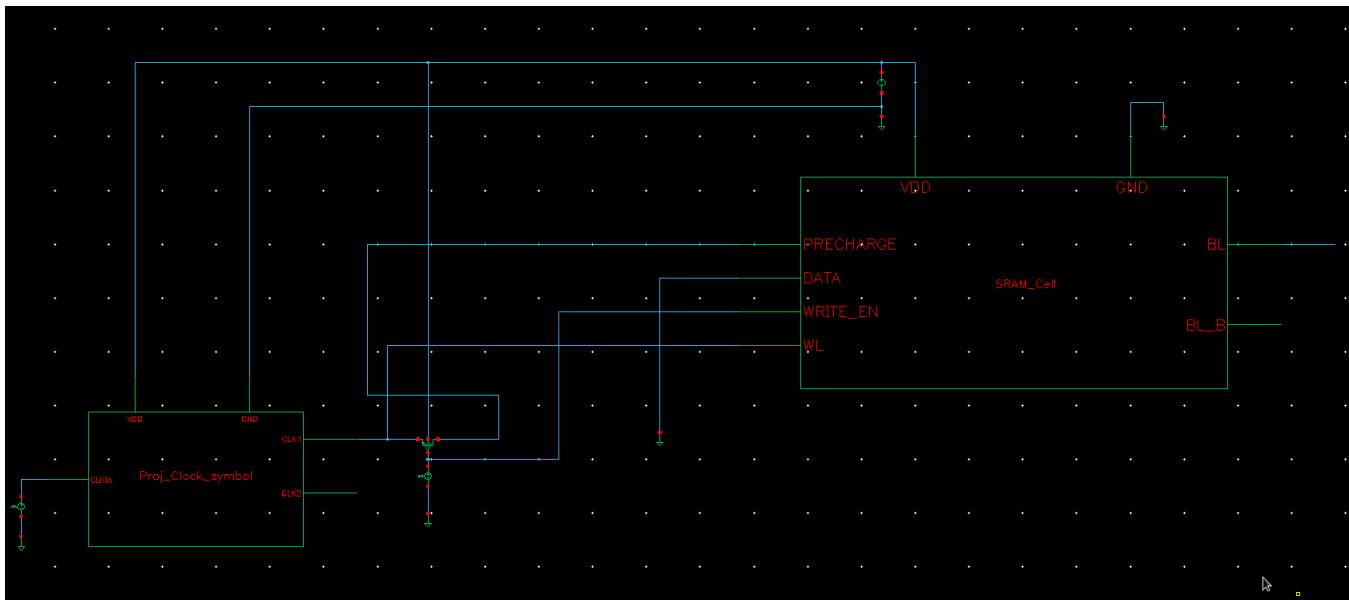


Figure 19: 6T - SRAM Cell Circuit Schematic : Testing with storing "0"

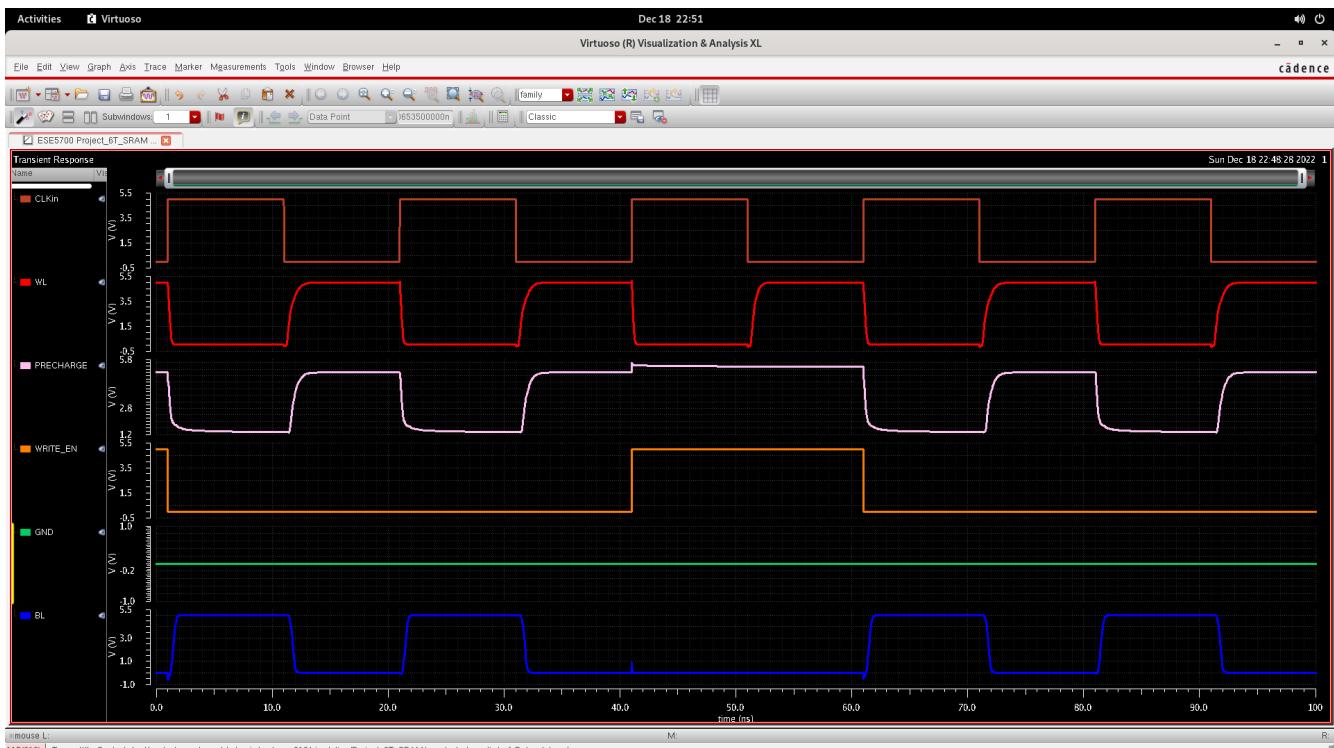


Figure 20: Verification Graph - 2

Fig. 19 and 20 depict that our SRAM is functioning just as expected. In this, we performed a read operation twice (for testing the nature of the read operation - destructive or non-destructive) and then wrote a 0 into the cell by driving the data onto the bitline, and then raising the word line. Upon writing a 0 into the cell , we then perform the read operation twice to make sure that our read is non-destructive, i.e. the data stored in the SRAM cell is not corrupted. The BL waveform shows that the SRAM is working properly.

3.6 Layout

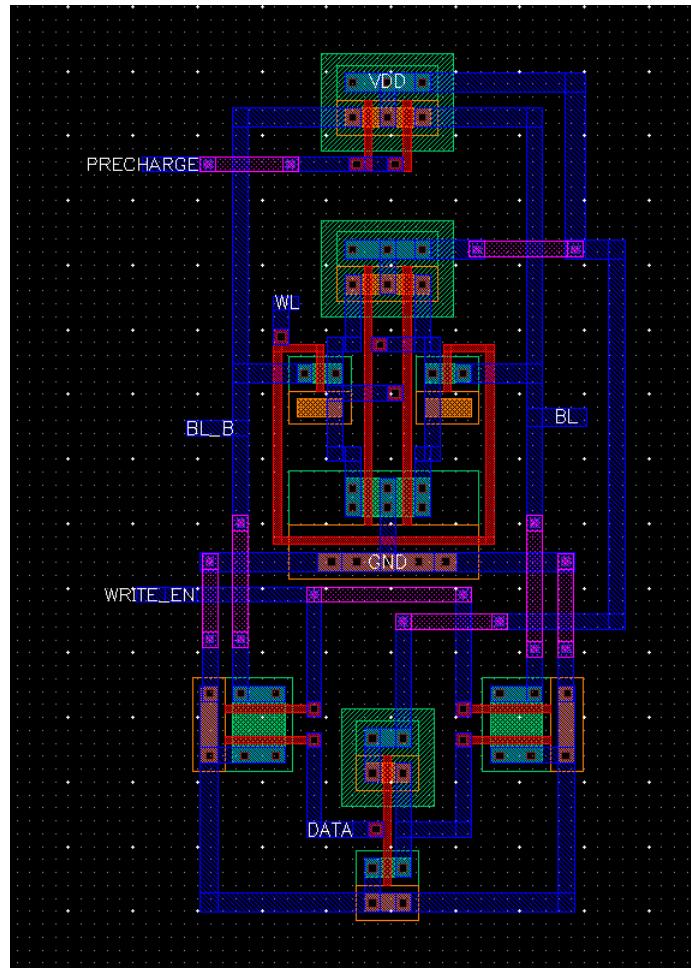


Figure 21: 6T - SRAM Cell Layout

3.7 Proof of Passing DRC

Figure 22: 6T - SRAM Cell DRC

3.8 Proof of Passing LVS

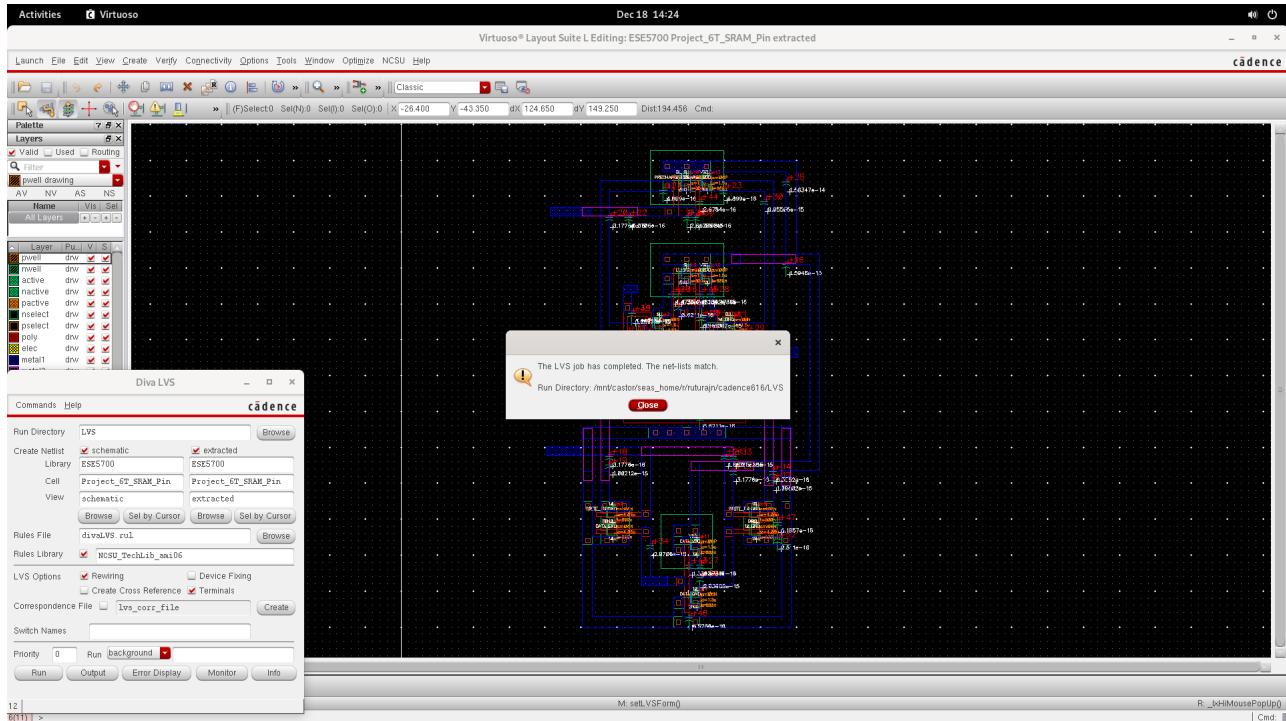


Figure 23: 6T - SRAM Cell LVS

```

Activities  @ Virtuoso          Dec 18 14:25
File Edit View Help
/mnt/castor/seas_home/r/ruturajn/cadence616/LVS/si.out
cadence

#DIV/6T-SRAM_LVS version 6.1.0.09/01/2015 15:36 (ejfnl105) #
Command line: /opt/pellicc/software/ic-design/cadence/ICE16/tools/lrv86/dfl1/bin/32bit/LVS -dir /mnt/castor/seas_home/r/ruturajn/cadence616/LVS -l -s -t /mnt/castor/seas_home/r/ruturajn/cadence616/LVS/layout /mnt/castor/seas_home/r/ruturajn/cadence616/LVS/schematic
Line watching is enabled.
Net swapping is enabled.
Using table correspondence points.
Compiling Diva LVS rules...
Net-list summary for /mnt/castor/seas_home/r/ruturajn/cadence616/LVS/layout/netlist
  count
  13   nets
  5   terminals
  5   pmos
  9   nmos

Net-list summary for /mnt/castor/seas_home/r/ruturajn/cadence616/LVS/schematic/netlist
  count
  13   nets
  5   terminals
  5   pmos
  9   nmos

Terminal correspondence points
N13  M1  EL
N12  M9  EL_B
N10  M8  DATA
N8   M11 GND
N7   M5  PRECHARGE
N4   M3  VDD
N9   M2  VSS
N11  M6  WRITE_EN

Devices in the netlist but not in the rules:
  scope netlist
Devices in the rules but not in the netlist:
  cap nfet pfet nmos4 pmos4

The net-lists match.
      layout  schematic
un-watched      0      0
  revised       0      0
size across     0      0
  pruned       0      0
  active       14     14
  total        14     14

      nets
un-watched      0      0
  merged       0      0
  pruned       0      0
  active       13    13
  total        13    13

      terminals
un-watched      0      0
  watched but different type  0      0
  total         8      8

Probe files from /mnt/castor/seas_home/r/ruturajn/cadence616/LVS/schematic
devbad.out:
netbad.out:
wrgenout.out:
termbad.out:
13 |
```

Figure 24: 6T - SRAM Cell LVS Output 1

```

Activities  Virtuoso  Dec 18 14:25
/mnt/castor/seas_home/r/ruturajn/cadence616/LVS/si.out  cadence
File Edit View Help

Terminal correspondence points
N1 M9 EL_B
N12 M8 DATA
N10 M1 GND
N5 M11 PRECHARGE
N7 M5 VDD
N14 M3 WL
N11 M6 WRITX_EN

Devices in the netlist but not in the rules:
probe.out:
Devices in the rules but not in the netlist:
cap nfet pmos4 pmos4

The net-lists match.

layout schematic instances
un-matched 0 0
resolved 0 0
size errors 0 0
pruned 0 0
active 14 14
total 14 14

nets
un-matched 0 0
merged 0 0
pruned 0 0
active 13 13
total 13 13

terminals
un-matched 0 0
matched but different type 0 0
total 8 8

Probe files from /mnt/castor/seas_home/r/ruturajn/cadence616/LVS/schematic
devbad.out:
netbad.out:
wergnet.out:
terbad.out:
prunenet.out:
prunedev.out:
audit.out:

Probe files from /mnt/castor/seas_home/r/ruturajn/cadence616/LVS/layout
devbad.out:
netbad.out:
wergnet.out:
terbad.out:
prunenet.out:
prunedev.out:
audit.out:
13

```

Figure 25: 6T - SRAM Cell LVS Output 2

4 CLOCK

This circuit provides us with two out of phase clocks (generated using a single clock) that are used to run our configurable logic block. The clock was designed using NOR gates and inverters. All transistors used to design the clock are minimum sized.

4.1 Schematic

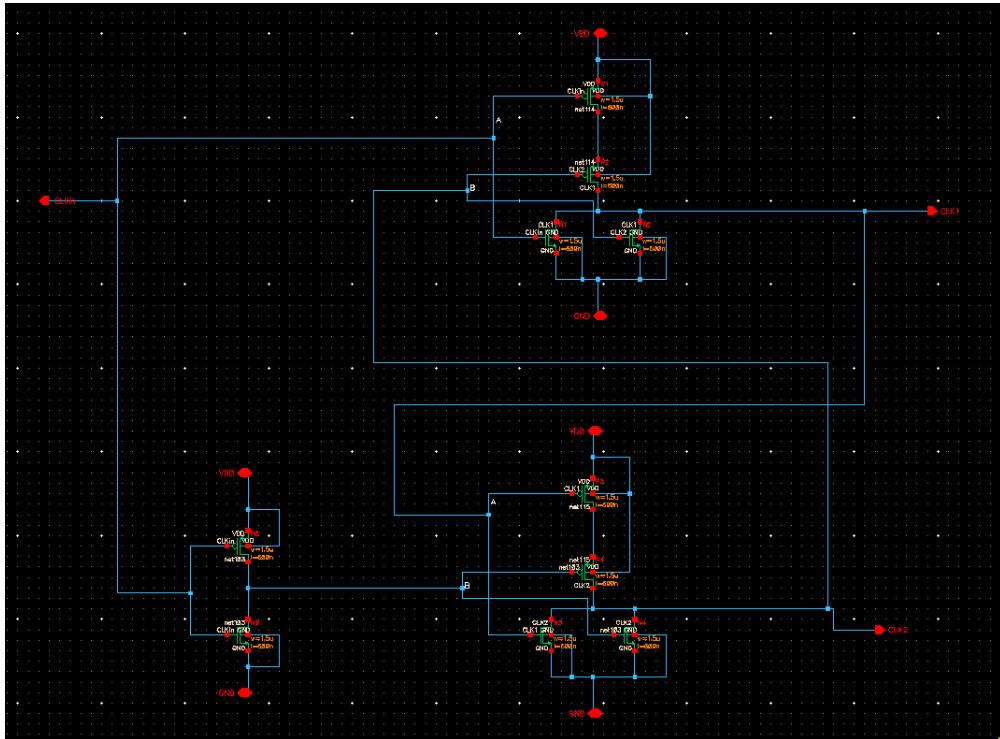


Figure 26: Circuit Schematic for Clock

4.2 Symbol

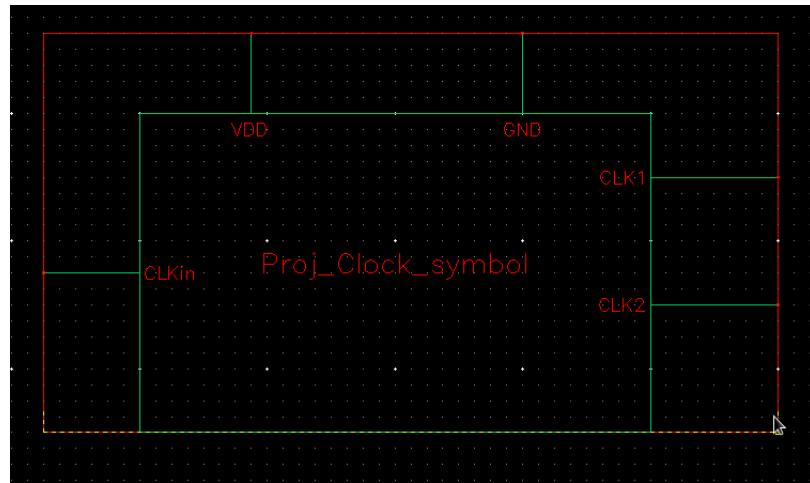


Figure 27: Circuit Schematic for Clock

4.3 Test schematic using symbol

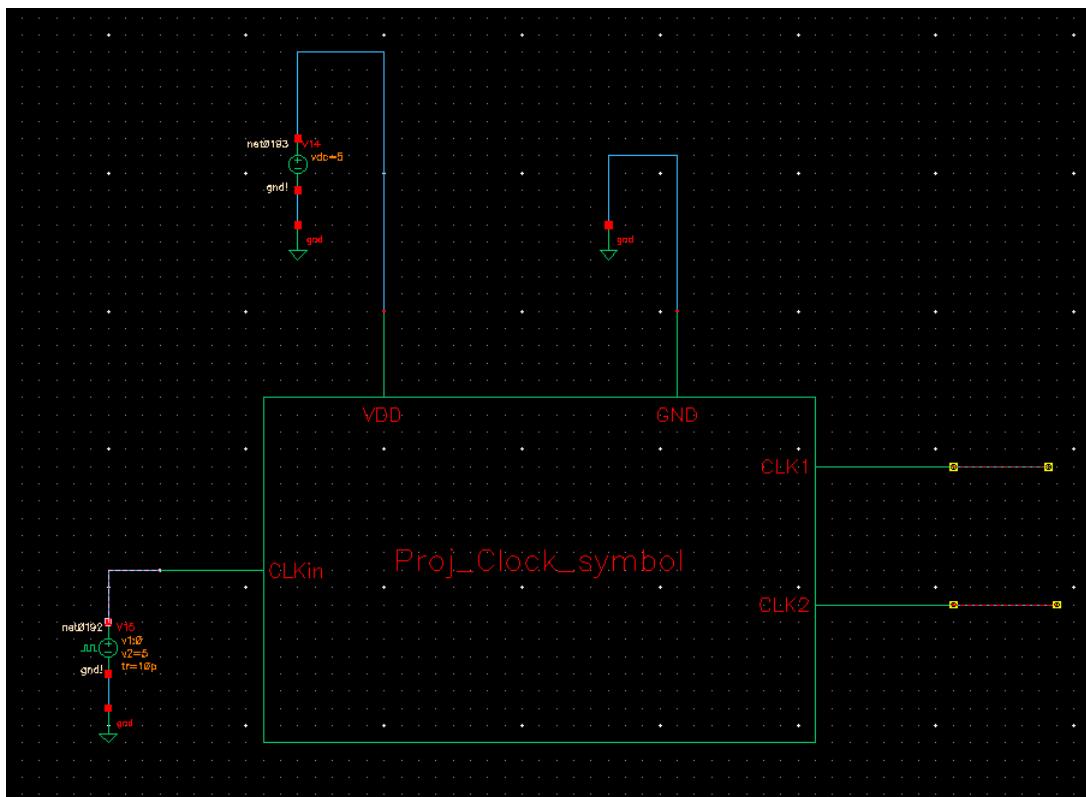


Figure 28: Test Schematic for Clock

4.4 Verification using Symbol

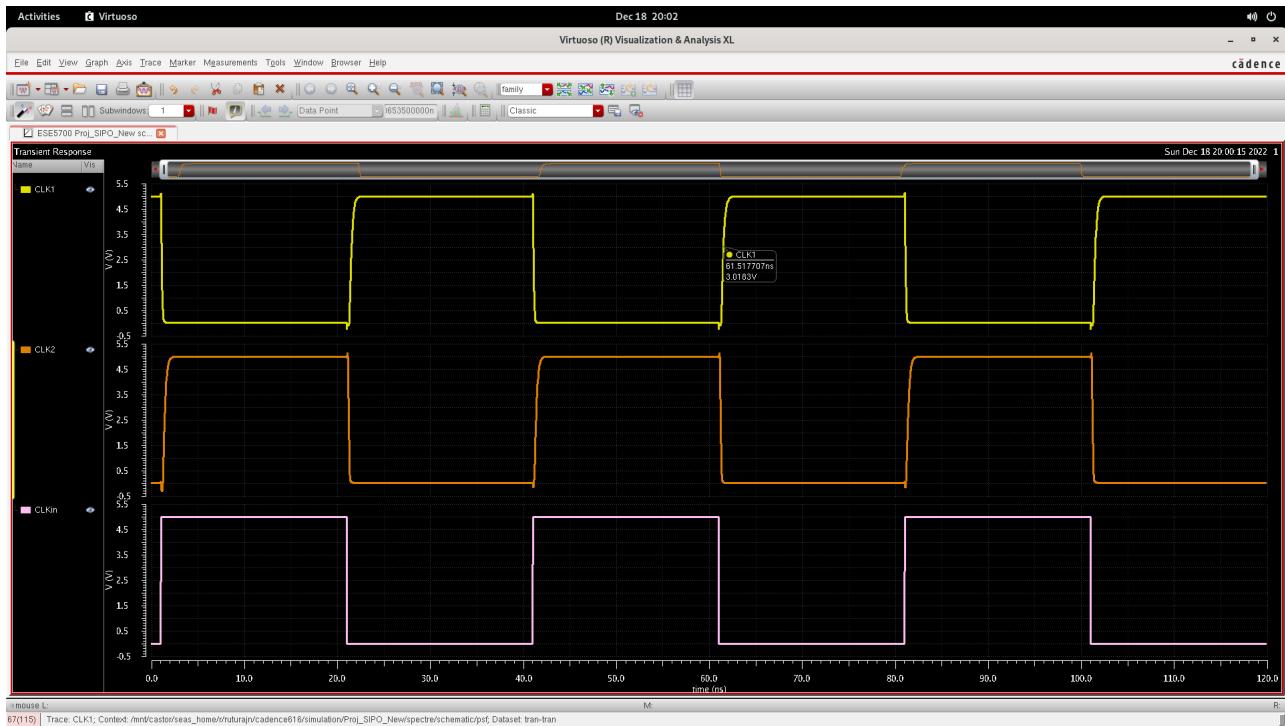


Figure 29: Verification Graph - Clock

From this plot , it is evident that we are able to generate two out of phases clocks using a single clock. Here CLK_2 follows our input CLK_{in} and CLK_1 is the opposite / out of phase.

4.5 Layout

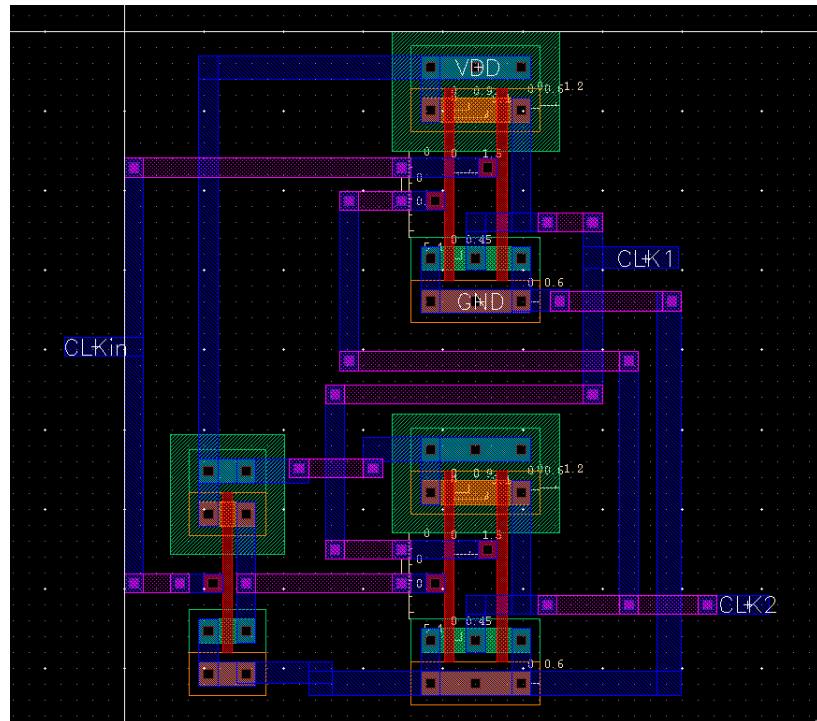


Figure 30: Layout - Clock

4.6 Proof of Passing DRC

```

Activities   Virtuoso  Dec 18 20:35
Virtuoso® 6.1.6 - Log:/mnt/castor/seas_home/r/ruturajn/CDS.log  cädence
File Tools Options Help

executing: drc(geomEdgEdge("elec") geomNetEdge("metal2") (sep < (lambda * 2.0)) errMsg)
executing: saveDerived(geomNetEdge(netsAll elec diffNet) errMsg)
executing: saveDerived(geomNetEdge(netsAll elec diffNet) (sep < (lambda * 2.0)) errMsg)
executing: saveDerived(geomNetEdge(netsAll elec diffNet) (width < (lambda * 2.0)) errMsg)
executing: drc(TransistorEleEdge (sep < (lambda * 3.0)) errMsg)
drc(TransistorEleEdge (width < (lambda * 3.0)) errMsg)
executing: drc(TransistorEleEdge activeEdge (sep < (lambda * 2.0)) errMsg)
executing: drc(TransistorEleEdge activeEdge (sep < (lambda * 1.0)) errMsg)
executing: drc(TransistorEleEdge polyEdge (sep < (lambda * 3.0)) errMsg)
executing: drc(TransistorEleEdge polyEdge (sep < (lambda * 2.0)) errMsg)
executing: saveDerived(geomNetTransistorEle (sep < (lambda * 3.0)) errMsg)
executing: saveDerived(geomNetTransistorEle (sep < (lambda * 2.0)) errMsg)
executing: saveDerived(geomNetTransistorEle cs) errMsg)
executing: drc(codEdge (width < (lambda * 2.0)) errMsg)
executing: drc(codEdge (notch < (lambda * 3.0)) errMsg)
executing: drc(codEdge (notch < (lambda * 2.0)) errMsg)
executing: drc(codEdge (area > ((lambda * 2.0 * (lambda * 2.0)) + (lambda * 0.1 * (lambda * 0.1)))) ... errMsg)
executing: drc(CompactEleEdge codEdge (enc < (lambda * 2.0)) errMsg)
executing: saveDerived(geomNetNet("elec") "(SONS Rule 13 3.13.4) electrode enclosure of cont... errMsg)
executing: drc(codEdge polyEdge (sep < (lambda * 3.0)) errMsg)
executing: saveDerived(geomNetNet("elec") "(SONS Rule 13 3.13.4) electrode enclosure of cont... errMsg)
executing: drc(codEdge activeEdge (sep < (lambda * 3.0)) errMsg)
executing: saveDerived(geomNetNet("active") (sep < (lambda * 3.0)) errMsg)
executing: drc(via2Edge (width < (lambda * 2.0)) errMsg)
executing: drc(via2Edge (area > ((lambda * 2.0 * (lambda * 2.0)) + (lambda * 0.1 * (lambda * 0.1)))) ... errMsg)
executing: drc(metal2Edge via2Edge (enc < (lambda * 1.0)) errMsg)
executing: drc(metal2Edge via2Edge (width < (lambda * 5.0)) errMsg)
executing: saveDerived(geomNetNet("via2_metal3") "(SONS Rule 13 3.13.4) electrode enclosure of cont... errMsg)
executing: drc(metal2Edge (sep < (lambda * 3.0)) errMsg)
drc(metal2Edge (notch < (lambda * 3.0)) errMsg)
executing: drc(chipresEdge (width < (lambda * 4.0)) errMsg)
executing: drc(chipresEdge (notch < (lambda * 4.0)) errMsg)
executing: drc(chipresEdge codEdge (sep < (lambda * 2.0)) errMsg)
executing: drc(chipresEdge codEdge (sep < (lambda * 2.0)) errMsg)
executing: saveDerived(geomNetNet("cap") errMsg)
executing: drc(chipresEdge activeEdge (sep < (lambda * 4.0)) errMsg)
executing: saveDerived(geomNetNet("elecHighres") (sep < (lambda * 2.0)) errMsg)
executing: saveDerived(geomNetNet("elecHighres") (sep < (lambda * 2.0)) errMsg)
executing: saveDerived(geomNetNet("elecHighres") (ignore == 2)) errMsg)
executing: saveDerived(geomNetNet("elecHighres") (overall)) "(SONS Rule 27.6) resistor must be outside w... errMsg)
executing: saveDerived(geomNetNet("elecHighres") (overall)) "(SONS Rule 27.6) resistor must be outside w... errMsg)
executing: drc(elelighresEdge (sep < (lambda * 5.0)) errMsg)
drc(elelighresEdge (notch < (lambda * 7.0)) errMsg)
executing: drc(elelighresEdge (width < (lambda * 2.0)) errMsg)
DRC started ... Sun Dec 18 20:35:28 2022
completed ... Sun Dec 18 20:35:28 2022
CPU TIME = 00:00:00 TOTAL TIME = 00:00:00
*****Summary of rule violations for cell "Proj_Clock layout" *****
Total errors Found: 0

```

Figure 31: DRC - Clock

4.7 Proof of Passing LVS

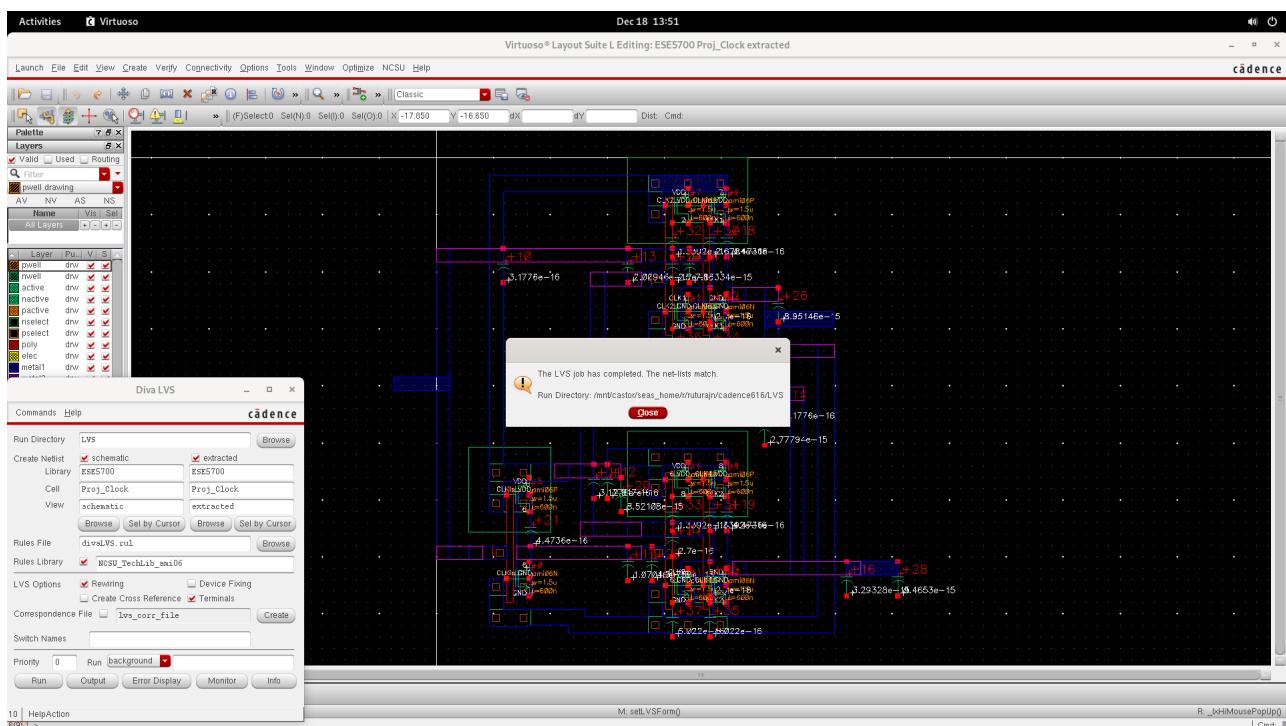


Figure 32: LVS - Clock

```

Activities Virtuoso
Dec 18 13:52
/mnt/castor/seas_home/r/ruturajn/cadence616/LVS/si.out
File Edit View Help
cadence

5      rmos

Terminal correspondence points
N5      N5      CLK1
N4      N4      CLK2
N5      N7      GNDin
N3      N2      GND
N7      N1      VDD

Devices in the netlist but not in the rules:
  popcap
Devices in the rules but not in the netlist:
  cap nfet pfet rmos4 pmos4

The net-lists match.
          layout schematic
instances
un-matched          0      0
reserved           0      0
size errors        0      0
pruned             0      0
active             10     10
total              10     10

nets
un-matched          0      0
merged              0      0
pruned              0      0
active              8      8
total               8      8

terminals
un-matched          0      0
matched but different type  0      0
total               5      5

Probe files from /mnt/castor/seas_home/r/ruturajn/cadence616/LVS/schematic
devbad.out:
netbad.out:
mergenet.out:
terabad.out:
prunenet.out:
prunedev.out:
audit.out:

Probe files from /mnt/castor/seas_home/r/ruturajn/cadence616/LVS/layout
devbad.out:
netbad.out:
mergenet.out:
terabad.out:
prunenet.out:
prunedev.out:
audit.out:

```

Figure 33: LVS Output 1 - Clock

```

Activities Virtuoso
Dec 18 13:52
/mnt/castor/seas_home/r/ruturajn/cadence616/LVS/si.out
File Edit View Help
cadence

R@FUSCUS: LVS version 5.1.6 09/01/2015 15:36 (aifol105) $.
Command line: /mnt/pulux/software/ic-design/cadence/IC616/tools/lrv86/dfl1/bin/32bit/LVS -dir /mnt/castor/seas_home/r/ruturajn/cadence616/LVS -l -s -t /mnt/castor/seas_home/r/ruturajn/cadence616/LVS /mnt/castor/seas_home/r/ruturajn/cadence616/LVS/schematic
Like matching is enabled
Net swapping is enabled
Using 1000000 correspondence points.
Compiling Diffs LVS rules...
Net-list summary for /mnt/castor/seas_home/r/ruturajn/cadence616/LVS/layout/netlist
count
  5      nets
  5      terminals
  5      pmos
  5      rmos

Net-list summary for /mnt/castor/seas_home/r/ruturajn/cadence616/LVS/schematic/netlist
count
  5      nets
  5      terminals
  5      pmos
  5      rmos

Terminal correspondence points
N5      N5      CLK1
N4      N4      CLK2
N5      N7      GNDin
N3      N2      GND
N7      N1      VDD

Devices in the netlist but not in the rules:
  popcap
Devices in the rules but not in the netlist:
  cap nfet pfet rmos4 pmos4

The net-lists match.
          layout schematic
instances
un-matched          0      0
reserved           0      0
size errors        0      0
pruned             0      0
active             10     10
total              10     10

nets
un-matched          0      0
merged              0      0
pruned              0      0
active              8      8
total               8      8

terminals
un-matched          0      0
matched but different type  0      0
total               5      5

Probe files from /mnt/castor/seas_home/r/ruturajn/cadence616/LVS/schematic
devbad.out:
netbad.out:
mergenet.out:
terabad.out:
prunenet.out:
prunedev.out:
audit.out:

```

Figure 34: LVS Output 2 - Clock

5 D FLIP FLOP

5.1 Schematic

We have designed a positive edge triggered D Flip Flop using Pass transistors and inverters. The D Flip Flop is of the Master - Slave configuration. All the transistors used in the design of the D Flip Flop are minimum sized.

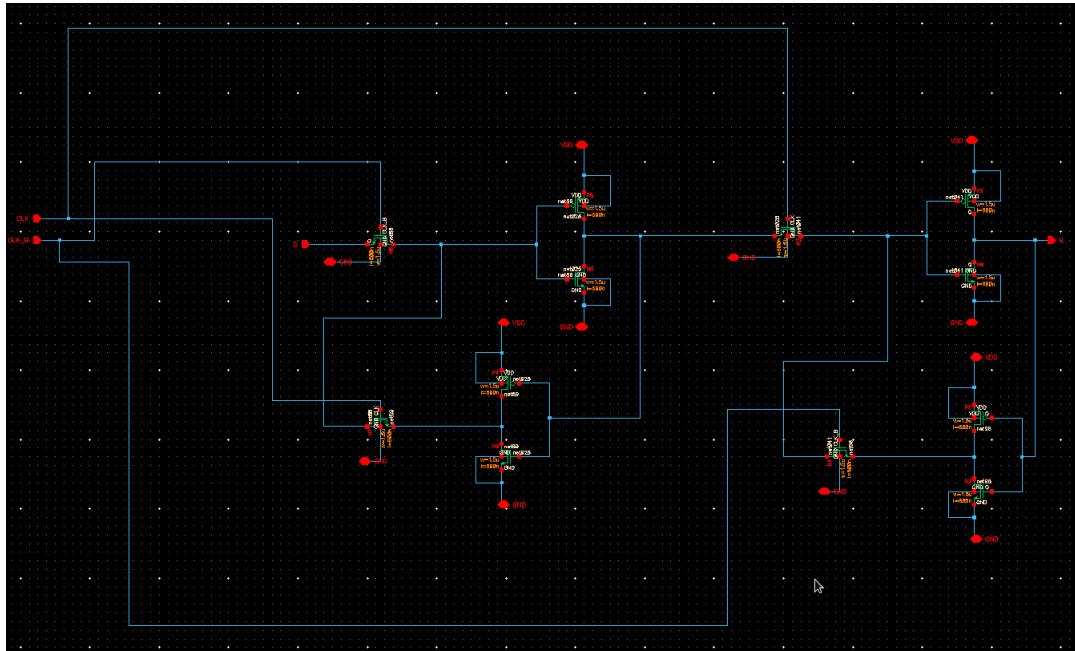


Figure 35: Circuit Schematic for D Flip Flop

5.2 Symbol

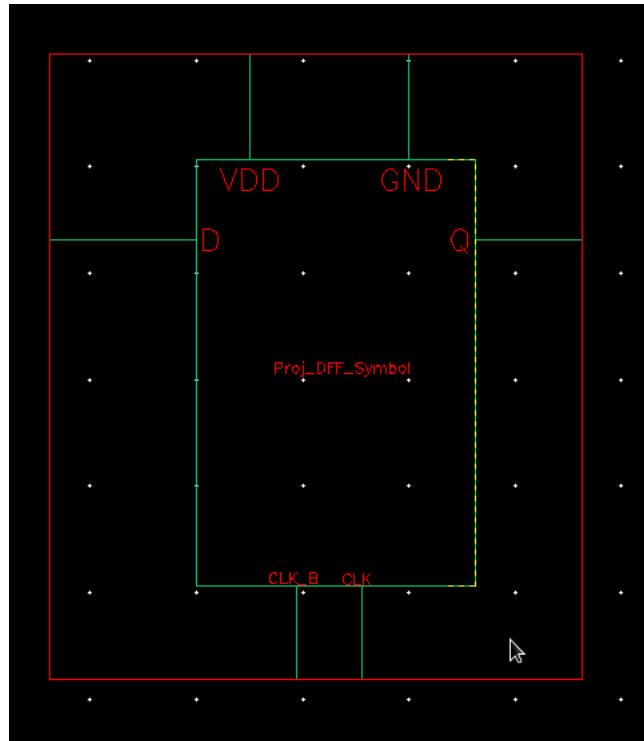


Figure 36: Symbol for D Flip Flop

5.3 Test schematic using Symbol

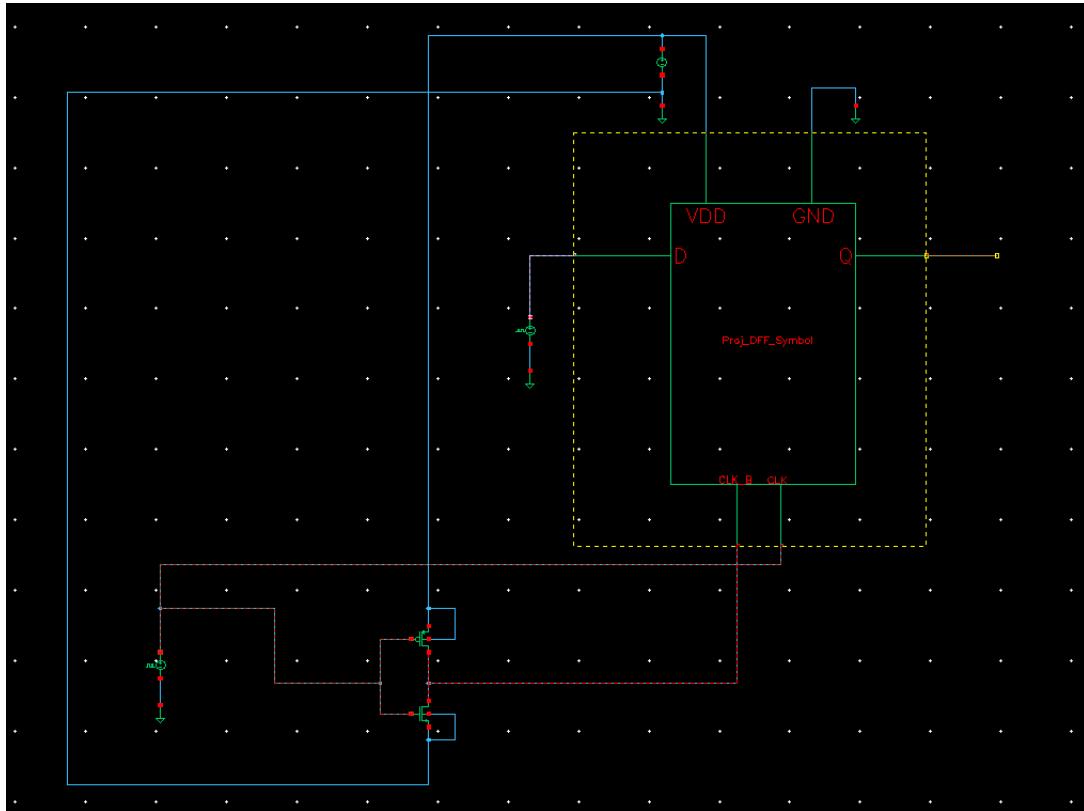


Figure 37: Symbol for D Flip Flop

For testing our D Flip Flop, we provided a *vpulse* to the *CLK*, and its inverted output to *CLK_B*. We have also set a *vpulse* as our input data D.

5.4 Functionality Verification



Figure 38: Functionality Verification for the D Flip Flop

From this plot , it is evident that the output (Q) follows the input data (D) when a positive edge of the clock is encountered since we have designed a positive edge triggered D Flip Flop. This justifies the working of our D Flip Flop.

5.5 Layout

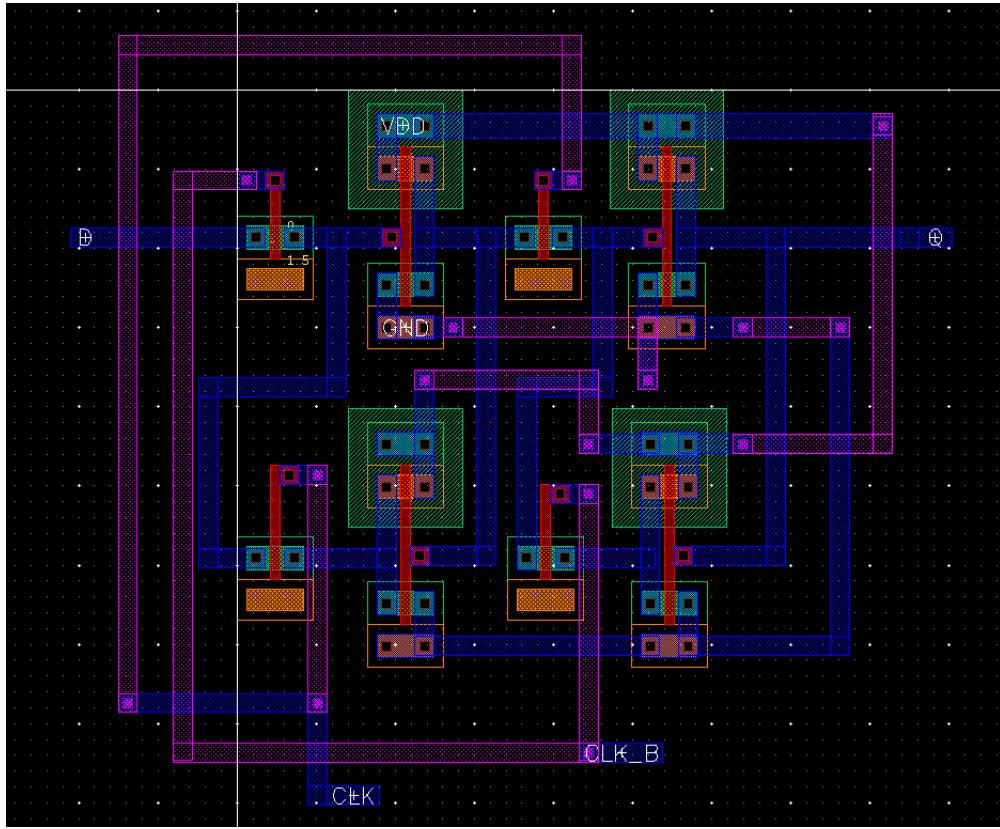


Figure 39: Layout for the D Flip Flop

5.6 Proof of Passing DRC

Figure 40: D Flip Flop - DRC

5.7 Proof of Passing LVS

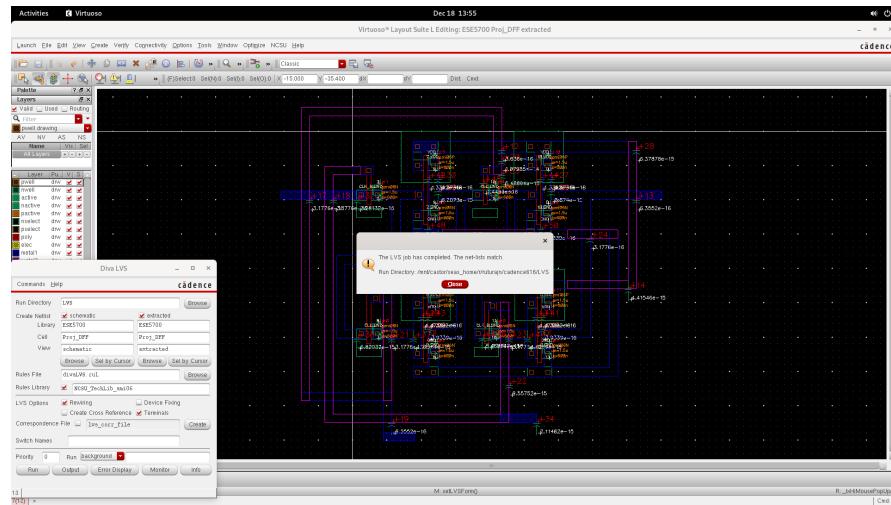


Figure 41: D Flip Flop - LVS

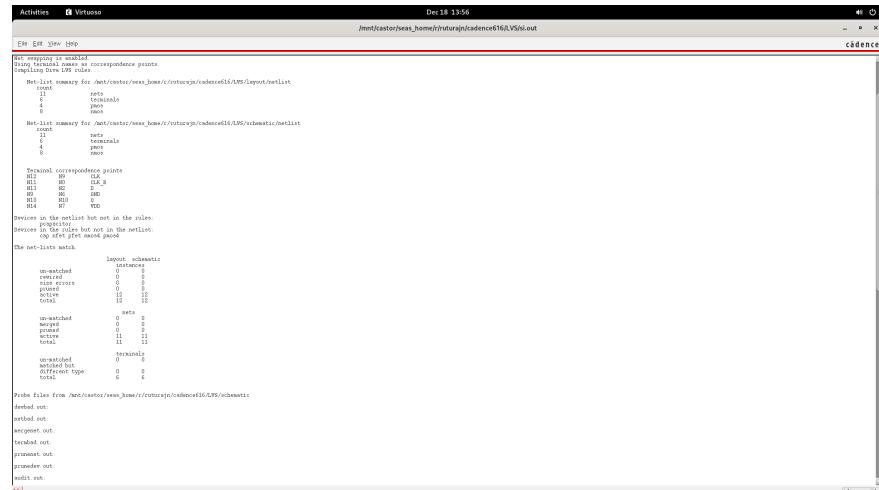


Figure 42: D Flip Flop - LVS Output 1

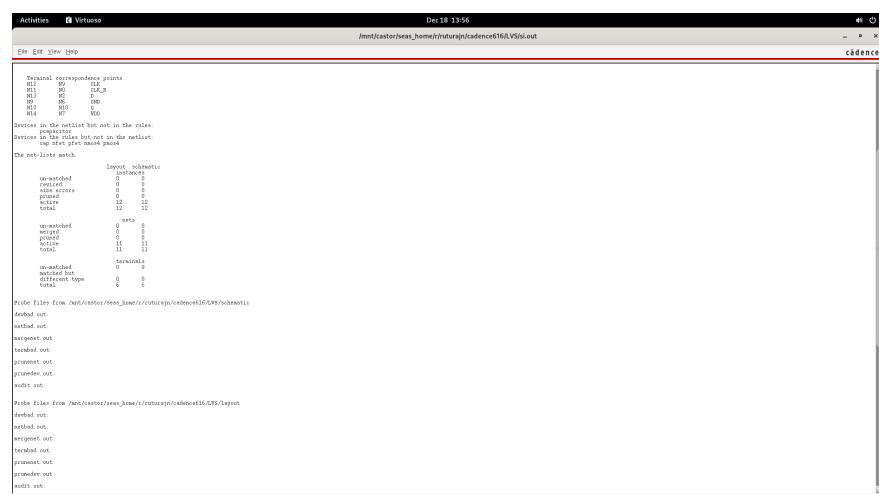


Figure 43: D Flip Flop - LVS Output 2

6 SERIAL IN PARALLEL OUT REGISTER

We are using a serial in parallel out shift register to which we input a serial bitstream . The output from the serial in parallel out shift register is fed to the SRAM array. This is done by cascading 16 D Flip Flops in series. The output at each stage is in parallel given to the SRAM array(16 SRAM's cascaded together).

6.1 Schematic

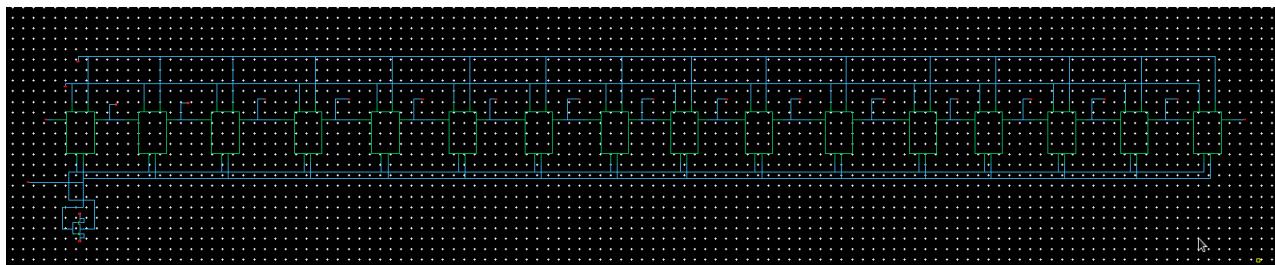


Figure 44: Circuit Schematic - Serial In Parallel Out register

6.2 Symbol

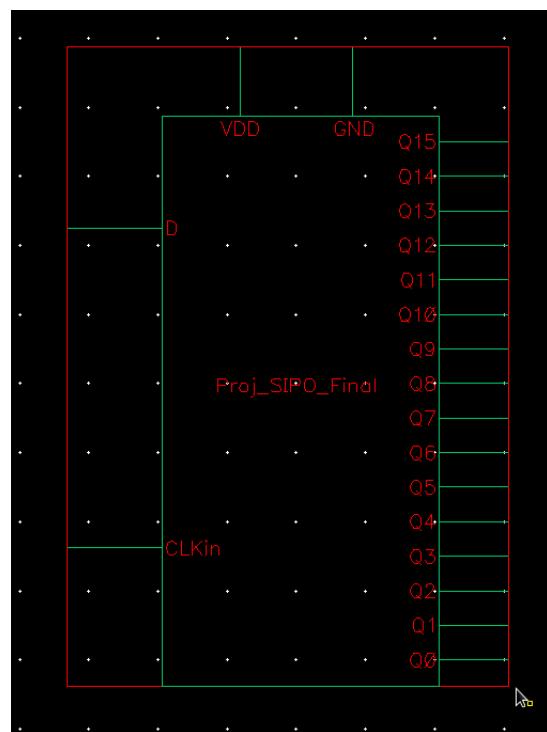


Figure 45: Symbol - Serial In Parallel Out register

6.3 Test schematic using symbol

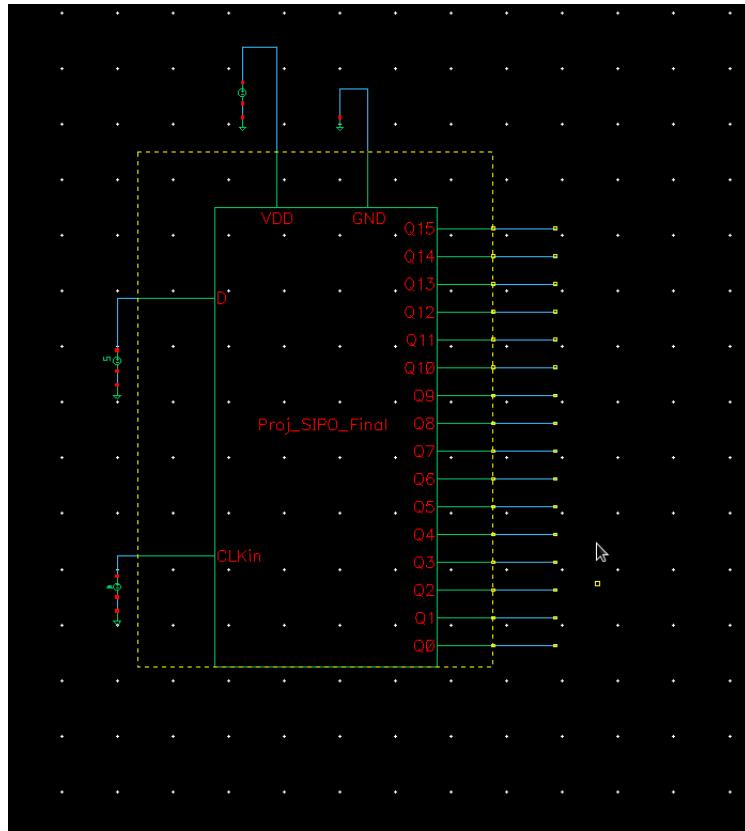


Figure 46: Test Schematic - Serial In Parallel Out register

Fig. 46 shows the test schematic for the Serial In Parallel Out register. The input is fed with a *vbit* source that gives out binary data 1101111010111010.

6.4 Functionality Verification

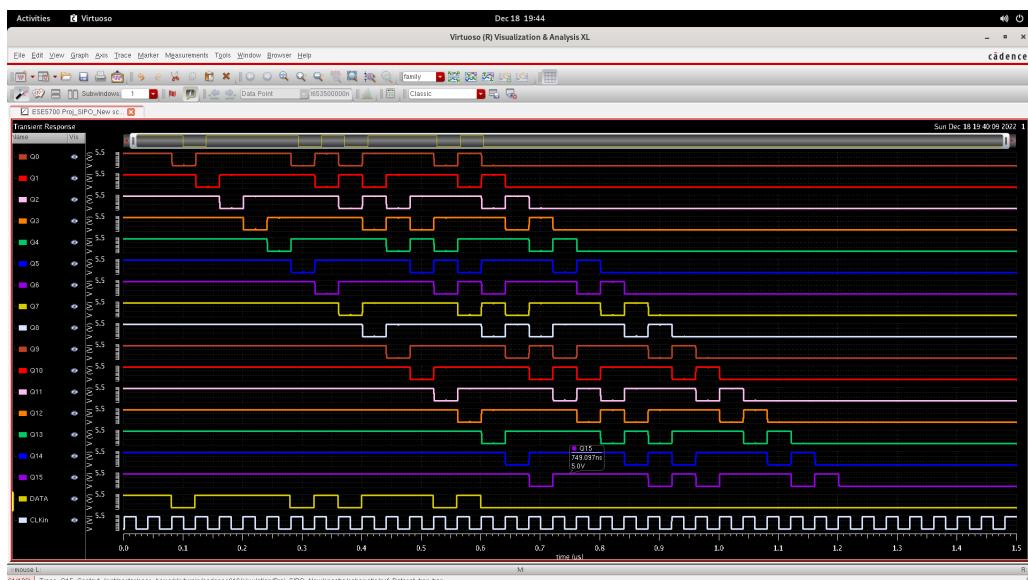


Figure 47: Verification of Functionality - Graph

As expected, Fig. 47 shows the input data being shifted into the register serially, and each bit is available at the output of each flip flop. The output at each stage of the Serial In Parallel Out register is slightly delayed with respect to the stage before it which confirms that the register is working properly.

6.5 Layout

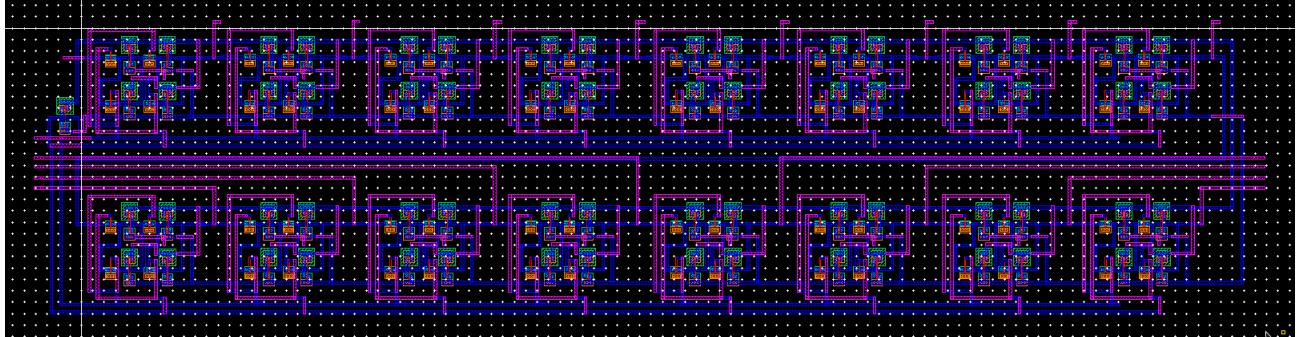


Figure 48: Layout - Serial In Parallel Out register

6.6 Proof of Passing DRC

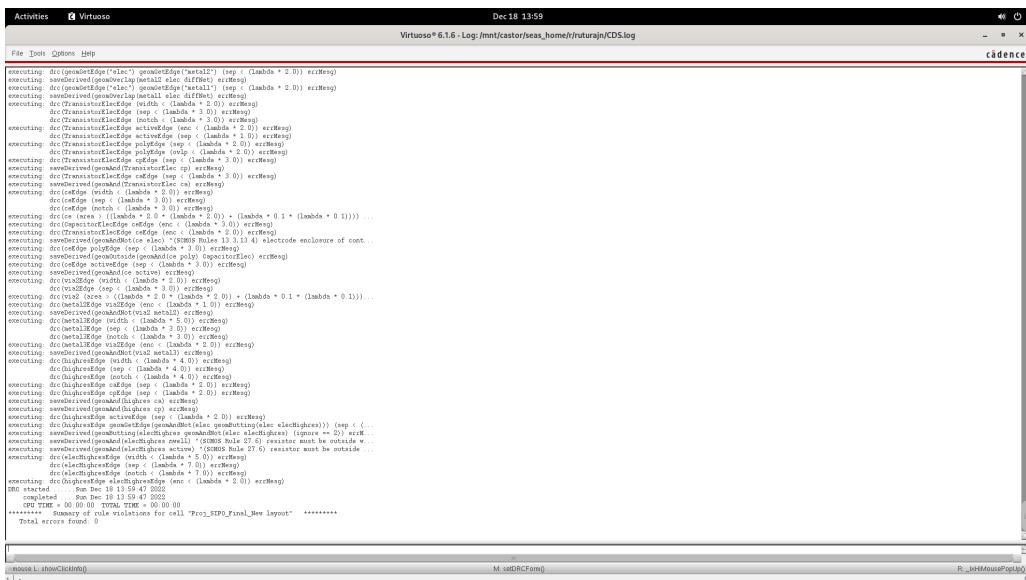


Figure 49: DRC - Serial In Parallel Out register

6.7 Proof of Passing LVS

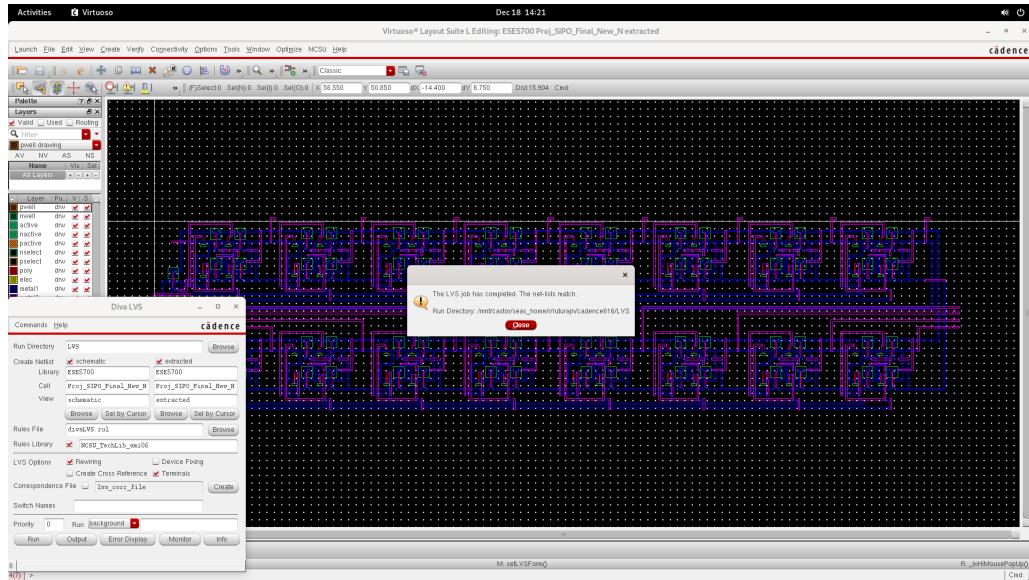


Figure 50: LVS - Serial In Parallel Out register

```
Activities  Virtuoso
Dec 18 14:21
/mnt/castor/seas_home/ruturaj/cadence61b/LVS/si.out
cadence

Edu View Help
#DIVSOS: LVS version 6.1.0 09/01/2011 18:10: (xjfallis) $ 
Command line: /usr/ppl00/nsoftice/ic-design/cadence/lvs6/tools/lxos/dfl1/bin/32bit/LVS -dir /mnt/castor/seas_home/c/ruturaj/cadence61b/LVS -l -t /mnt/castor/seas_home/c/ruturaj/cadence61b/LVS/layout /mnt/castor/seas_home/c/ruturaj/cadence61b/LVS/schematic
Using terminal names as correspondence points.
Correspondence file: /mnt/castor/seas_home/c/ruturaj/cadence61b/LVS/si.out

Net-list summary for /mnt/castor/seas_home/c/ruturaj/cadence61b/LVS/layout.netlist
  count   nets   terminals
  101     0       0
  65     100      100
  120     0       0

Net-list summary for /mnt/castor/seas_home/c/ruturaj/cadence61b/LVS/schematic.netlist
  count   nets   terminals
  101     0       0
  65     100      100
  120     0       0

Terminal correspondence points
W157  W1  Q1X1
W158  W11  Q1D
W159  W11  Q1D
W160  W11  Q1D
W161  W11  Q1D
W162  W11  Q1D
W163  W11  Q1D
W164  W11  Q1D
W165  W11  Q1D
W166  W11  Q1D
W167  W11  Q1D
W168  W11  Q1S
W169  W11  Q1S
W170  W11  Q1S
W171  W11  Q1S
W172  W11  Q1S
W173  W11  Q1S
W174  W11  Q1S
W175  W11  Q1S
W176  W11  Q1S
W177  W11  Q1S
W178  W11  Q1S
W179  W11  Q1S
W180  W11  Q1S
W181  W11  Q1S
W182  W11  Q1S
W183  W11  Q1S
W184  W11  Q1S
W185  W11  Q1S
W186  W11  Q1S
W187  W11  Q1S
W188  W11  Q1S
W189  W11  Q1S
W190  W11  Q1S
W191  W11  Q1S
W192  W11  Q1S
W193  W11  Q1S
W194  W11  Q1S
W195  W11  Q1S
W196  W11  Q1S
W197  W11  Q1S
W198  W11  Q1S
W199  W11  Q1S
W200  W11  Q1S
W201  W11  Q1S
W202  W11  Q1S
W203  W11  Q1S
W204  W11  Q1S
W205  W11  Q1S
W206  W11  Q1S
W207  W11  Q1S
W208  W11  Q1S
W209  W11  Q1S
W210  W11  Q1S
W211  W11  Q1S
W212  W11  Q1S
W213  W11  Q1S
W214  W11  Q1S
W215  W11  Q1S
W216  W11  Q1S
W217  W11  Q1S
W218  W11  Q1S
W219  W11  Q1S
W220  W11  Q1S
W221  W11  Q1S
W222  W11  Q1S
W223  W11  Q1S
W224  W11  Q1S
W225  W11  Q1S
W226  W11  Q1S
W227  W11  Q1S
W228  W11  Q1S
W229  W11  Q1S
W230  W11  Q1S
W231  W11  Q1S
W232  W11  Q1S
W233  W11  Q1S
W234  W11  Q1S
W235  W11  Q1S
W236  W11  Q1S
W237  W11  Q1S
W238  W11  Q1S
W239  W11  Q1S
W240  W11  Q1S
W241  W11  Q1S
W242  W11  Q1S
W243  W11  Q1S
W244  W11  Q1S
W245  W11  Q1S
W246  W11  Q1S
W247  W11  Q1S
W248  W11  Q1S
W249  W11  Q1S
W250  W11  Q1S
W251  W11  Q1S
W252  W11  Q1S
W253  W11  Q1S
W254  W11  Q1S
W255  W11  Q1S
W256  W11  Q1S
W257  W11  Q1S
W258  W11  Q1S
W259  W11  Q1S
W260  W11  Q1S
W261  W11  Q1S
W262  W11  Q1S
W263  W11  Q1S
W264  W11  Q1S
W265  W11  Q1S
W266  W11  Q1S
W267  W11  Q1S
W268  W11  Q1S
W269  W11  Q1S
W270  W11  Q1S
W271  W11  Q1S
W272  W11  Q1S
W273  W11  Q1S
W274  W11  Q1S
W275  W11  Q1S
W276  W11  Q1S
W277  W11  Q1S
W278  W11  Q1S
W279  W11  Q1S
W280  W11  Q1S
W281  W11  Q1S
W282  W11  Q1S
W283  W11  Q1S
W284  W11  Q1S
W285  W11  Q1S
W286  W11  Q1S
W287  W11  Q1S
W288  W11  Q1S
W289  W11  Q1S
W290  W11  Q1S
W291  W11  Q1S
W292  W11  Q1S
W293  W11  Q1S
W294  W11  Q1S
W295  W11  Q1S
W296  W11  Q1S
W297  W11  Q1S
W298  W11  Q1S
W299  W11  Q1S
W300  W11  Q1S
W301  W11  Q1S
W302  W11  Q1S
W303  W11  Q1S
W304  W11  Q1S
W305  W11  Q1S
W306  W11  Q1S
W307  W11  Q1S
W308  W11  Q1S
W309  W11  Q1S
W310  W11  Q1S
W311  W11  Q1S
W312  W11  Q1S
W313  W11  Q1S
W314  W11  Q1S
W315  W11  Q1S
W316  W11  Q1S
W317  W11  Q1S
W318  W11  Q1S
W319  W11  Q1S
W320  W11  Q1S
W321  W11  Q1S
W322  W11  Q1S
W323  W11  Q1S
W324  W11  Q1S
W325  W11  Q1S
W326  W11  Q1S
W327  W11  Q1S
W328  W11  Q1S
W329  W11  Q1S
W330  W11  Q1S
W331  W11  Q1S
W332  W11  Q1S
W333  W11  Q1S
W334  W11  Q1S
W335  W11  Q1S
W336  W11  Q1S
W337  W11  Q1S
W338  W11  Q1S
W339  W11  Q1S
W340  W11  Q1S
W341  W11  Q1S
W342  W11  Q1S
W343  W11  Q1S
W344  W11  Q1S
W345  W11  Q1S
W346  W11  Q1S
W347  W11  Q1S
W348  W11  Q1S
W349  W11  Q1S
W350  W11  Q1S
W351  W11  Q1S
W352  W11  Q1S
W353  W11  Q1S
W354  W11  Q1S
W355  W11  Q1S
W356  W11  Q1S
W357  W11  Q1S
W358  W11  Q1S
W359  W11  Q1S
W360  W11  Q1S
W361  W11  Q1S
W362  W11  Q1S
W363  W11  Q1S
W364  W11  Q1S
W365  W11  Q1S
W366  W11  Q1S
W367  W11  Q1S
W368  W11  Q1S
W369  W11  Q1S
W370  W11  Q1S
W371  W11  Q1S
W372  W11  Q1S
W373  W11  Q1S
W374  W11  Q1S
W375  W11  Q1S
W376  W11  Q1S
W377  W11  Q1S
W378  W11  Q1S
W379  W11  Q1S
W380  W11  Q1S
W381  W11  Q1S
W382  W11  Q1S
W383  W11  Q1S
W384  W11  Q1S
W385  W11  Q1S
W386  W11  Q1S
W387  W11  Q1S
W388  W11  Q1S
W389  W11  Q1S
W390  W11  Q1S
W391  W11  Q1S
W392  W11  Q1S
W393  W11  Q1S
W394  W11  Q1S
W395  W11  Q1S
W396  W11  Q1S
W397  W11  Q1S
W398  W11  Q1S
W399  W11  Q1S
W400  W11  Q1S
W401  W11  Q1S
W402  W11  Q1S
W403  W11  Q1S
W404  W11  Q1S
W405  W11  Q1S
W406  W11  Q1S
W407  W11  Q1S
W408  W11  Q1S
W409  W11  Q1S
W410  W11  Q1S
W411  W11  Q1S
W412  W11  Q1S
W413  W11  Q1S
W414  W11  Q1S
W415  W11  Q1S
W416  W11  Q1S
W417  W11  Q1S
W418  W11  Q1S
W419  W11  Q1S
W420  W11  Q1S
W421  W11  Q1S
W422  W11  Q1S
W423  W11  Q1S
W424  W11  Q1S
W425  W11  Q1S
W426  W11  Q1S
W427  W11  Q1S
W428  W11  Q1S
W429  W11  Q1S
W430  W11  Q1S
W431  W11  Q1S
W432  W11  Q1S
W433  W11  Q1S
W434  W11  Q1S
W435  W11  Q1S
W436  W11  Q1S
W437  W11  Q1S
W438  W11  Q1S
W439  W11  Q1S
W440  W11  Q1S
W441  W11  Q1S
W442  W11  Q1S
W443  W11  Q1S
W444  W11  Q1S
W445  W11  Q1S
W446  W11  Q1S
W447  W11  Q1S
W448  W11  Q1S
W449  W11  Q1S
W450  W11  Q1S
W451  W11  Q1S
W452  W11  Q1S
W453  W11  Q1S
W454  W11  Q1S
W455  W11  Q1S
W456  W11  Q1S
W457  W11  Q1S
W458  W11  Q1S
W459  W11  Q1S
W460  W11  Q1S
W461  W11  Q1S
W462  W11  Q1S
W463  W11  Q1S
W464  W11  Q1S
W465  W11  Q1S
W466  W11  Q1S
W467  W11  Q1S
W468  W11  Q1S
W469  W11  Q1S
W470  W11  Q1S
W471  W11  Q1S
W472  W11  Q1S
W473  W11  Q1S
W474  W11  Q1S
W475  W11  Q1S
W476  W11  Q1S
W477  W11  Q1S
W478  W11  Q1S
W479  W11  Q1S
W480  W11  Q1S
W481  W11  Q1S
W482  W11  Q1S
W483  W11  Q1S
W484  W11  Q1S
W485  W11  Q1S
W486  W11  Q1S
W487  W11  Q1S
W488  W11  Q1S
W489  W11  Q1S
W490  W11  Q1S
W491  W11  Q1S
W492  W11  Q1S
W493  W11  Q1S
W494  W11  Q1S
W495  W11  Q1S
W496  W11  Q1S
W497  W11  Q1S
W498  W11  Q1S
W499  W11  Q1S
W500  W11  Q1S
W501  W11  Q1S
W502  W11  Q1S
W503  W11  Q1S
W504  W11  Q1S
W505  W11  Q1S
W506  W11  Q1S
W507  W11  Q1S
W508  W11  Q1S
W509  W11  Q1S
W510  W11  Q1S
W511  W11  Q1S
W512  W11  Q1S
W513  W11  Q1S
W514  W11  Q1S
W515  W11  Q1S
W516  W11  Q1S
W517  W11  Q1S
W518  W11  Q1S
W519  W11  Q1S
W520  W11  Q1S
W521  W11  Q1S
W522  W11  Q1S
W523  W11  Q1S
W524  W11  Q1S
W525  W11  Q1S
W526  W11  Q1S
W527  W11  Q1S
W528  W11  Q1S
W529  W11  Q1S
W530  W11  Q1S
W531  W11  Q1S
W532  W11  Q1S
W533  W11  Q1S
W534  W11  Q1S
W535  W11  Q1S
W536  W11  Q1S
W537  W11  Q1S
W538  W11  Q1S
W539  W11  Q1S
W540  W11  Q1S
W541  W11  Q1S
W542  W11  Q1S
W543  W11  Q1S
W544  W11  Q1S
W545  W11  Q1S
W546  W11  Q1S
W547  W11  Q1S
W548  W11  Q1S
W549  W11  Q1S
W550  W11  Q1S
W551  W11  Q1S
W552  W11  Q1S
W553  W11  Q1S
W554  W11  Q1S
W555  W11  Q1S
W556  W11  Q1S
W557  W11  Q1S
W558  W11  Q1S
W559  W11  Q1S
W560  W11  Q1S
W561  W11  Q1S
W562  W11  Q1S
W563  W11  Q1S
W564  W11  Q1S
W565  W11  Q1S
W566  W11  Q1S
W567  W11  Q1S
W568  W11  Q1S
W569  W11  Q1S
W570  W11  Q1S
W571  W11  Q1S
W572  W11  Q1S
W573  W11  Q1S
W574  W11  Q1S
W575  W11  Q1S
W576  W11  Q1S
W577  W11  Q1S
W578  W11  Q1S
W579  W11  Q1S
W580  W11  Q1S
W581  W11  Q1S
W582  W11  Q1S
W583  W11  Q1S
W584  W11  Q1S
W585  W11  Q1S
W586  W11  Q1S
W587  W11  Q1S
W588  W11  Q1S
W589  W11  Q1S
W590  W11  Q1S
W591  W11  Q1S
W592  W11  Q1S
W593  W11  Q1S
W594  W11  Q1S
W595  W11  Q1S
W596  W11  Q1S
W597  W11  Q1S
W598  W11  Q1S
W599  W11  Q1S
W600  W11  Q1S
W601  W11  Q1S
W602  W11  Q1S
W603  W11  Q1S
W604  W11  Q1S
W605  W11  Q1S
W606  W11  Q1S
W607  W11  Q1S
W608  W11  Q1S
W609  W11  Q1S
W610  W11  Q1S
W611  W11  Q1S
W612  W11  Q1S
W613  W11  Q1S
W614  W11  Q1S
W615  W11  Q1S
W616  W11  Q1S
W617  W11  Q1S
W618  W11  Q1S
W619  W11  Q1S
W620  W11  Q1S
W621  W11  Q1S
W622  W11  Q1S
W623  W11  Q1S
W624  W11  Q1S
W625  W11  Q1S
W626  W11  Q1S
W627  W11  Q1S
W628  W11  Q1S
W629  W11  Q1S
W630  W11  Q1S
W631  W11  Q1S
W632  W11  Q1S
W633  W11  Q1S
W634  W11  Q1S
W635  W11  Q1S
W636  W11  Q1S
W637  W11  Q1S
W638  W11  Q1S
W639  W11  Q1S
W640  W11  Q1S
W641  W11  Q1S
W642  W11  Q1S
W643  W11  Q1S
W644  W11  Q1S
W645  W11  Q1S
W646  W11  Q1S
W647  W11  Q1S
W648  W11  Q1S
W649  W11  Q1S
W650  W11  Q1S
W651  W11  Q1S
W652  W11  Q1S
W653  W11  Q1S
W654  W11  Q1S
W655  W11  Q1S
W656  W11  Q1S
W657  W11  Q1S
W658  W11  Q1S
W659  W11  Q1S
W660  W11  Q1S
W661  W11  Q1S
W662  W11  Q1S
W663  W11  Q1S
W664  W11  Q1S
W665  W11  Q1S
W666  W11  Q1S
W667  W11  Q1S
W668  W11  Q1S
W669  W11  Q1S
W670  W11  Q1S
W671  W11  Q1S
W672  W11  Q1S
W673  W11  Q1S
W674  W11  Q1S
W675  W11  Q1S
W676  W11  Q1S
W677  W11  Q1S
W678  W11  Q1S
W679  W11  Q1S
W680  W11  Q1S
W681  W11  Q1S
W682  W11  Q1S
W683  W11  Q1S
W684  W11  Q1S
W685  W11  Q1S
W686  W11  Q1S
W687  W11  Q1S
W688  W11  Q1S
W689  W11  Q1S
W690  W11  Q1S
W691  W11  Q1S
W692  W11  Q1S
W693  W11  Q1S
W694  W11  Q1S
W695  W11  Q1S
W696  W11  Q1S
W697  W11  Q1S
W698  W11  Q1S
W699  W11  Q1S
W700  W11  Q1S
W701  W11  Q1S
W702  W11  Q1S
W703  W11  Q1S
W704  W11  Q1S
W705  W11  Q1S
W706  W11  Q1S
W707  W11  Q1S
W708  W11  Q1S
W709  W11  Q1S
W710  W11  Q1S
W711  W11  Q1S
W712  W11  Q1S
W713  W11  Q1S
W714  W11  Q1S
W715  W11  Q1S
W716  W11  Q1S
W717  W11  Q1S
W718  W11  Q1S
W719  W11  Q1S
W720  W11  Q1S
W721  W11  Q1S
W722  W11  Q1S
W723  W11  Q1S
W724  W11  Q1S
W725  W11  Q1S
W726  W11  Q1S
W727  W11  Q1S
W728  W11  Q1S
W729  W11  Q1S
W730  W11  Q1S
W731  W11  Q1S
W732  W11  Q1S
W733  W11  Q1S
W734  W11  Q1S
W735  W11  Q1S
W736  W11  Q1S
W737  W11  Q1S
W738  W11  Q1S
W739  W11  Q1S
W740  W11  Q1S
W741  W11  Q1S
W742  W11  Q1S
W743  W11  Q1S
W744  W11  Q1S
W745  W11  Q1S
W746  W11  Q1S
W747  W11  Q1S
W748  W11  Q1S
W749  W11  Q1S
W750  W11  Q1S
W751  W11  Q1S
W752  W11  Q1S
W753  W11  Q1S
W754  W11  Q1S
W755  W11  Q1S
W756  W11  Q1S
W757  W11  Q1S
W758  W11  Q1S
W759  W11  Q1S
W760  W11  Q1S
W761  W11  Q1S
W762  W11  Q1S
W763  W11  Q1S
W764  W11  Q1S
W765  W11  Q1S
W766  W11  Q1S
W767  W11  Q1S
W768  W11  Q1S
W769  W11  Q1S
W770  W11  Q1S
W771  W11  Q1S
W772  W11  Q1S
W773  W11  Q1S
W774  W11  Q1S
W775  W11  Q1S
W776  W11  Q1S
W777  W11  Q1S
W778  W11  Q1S
W779  W11  Q1S
W780  W11  Q1S
W781  W11  Q1S
W782  W11  Q1S
W783  W11  Q1S
W784  W11  Q1S
W785  W11  Q1S
W786  W11  Q1S
W787  W11  Q1S
W788  W11  Q1S
W789  W11  Q1S
W790  W11  Q1S
W791  W11  Q1S
W792  W11  Q1S
W793  W11  Q1S
W794  W11  Q1S
W795  W11  Q1S
W796  W11  Q1S
W797  W11  Q1S
W798  W11  Q1S
W799  W11  Q1S
W800  W11  Q1S
W801  W11  Q1S
W802  W11  Q1S
W803  W11  Q1S
W804  W11  Q1S
W805  W11  Q1S
W806  W11  Q1S
W807  W11  Q1S
W808  W11  Q1S
W809  W11  Q1S
W810  W11  Q1S
W811  W11  Q1S
W812  W11  Q1S
W813  W11  Q1S
W814  W11  Q1S
W815  W11  Q1S
W816  W11  Q1S
W817  W11  Q1S
W818  W11  Q1S
W819  W11  Q1S
W820  W11  Q1S
W821  W11  Q1S
W822  W11  Q1S
W823  W11  Q1S
W824  W11  Q1S
W825  W11  Q1S
W826  W11  Q1S
W827  W11  Q1S
W828  W11  Q1S
W829  W11  Q1S
W830  W11  Q1S
W831  W11  Q1S
W832  W11  Q1S
W833  W11  Q1S
W834  W11  Q1S
W835  W11  Q1S
W836  W11  Q1S
W837  W11  Q1S
W838  W11  Q1S
W839  W11  Q1S
W840  W11  Q1S
W841  W11  Q1S
W842  W11  Q1S
W843  W11  Q1S
W844  W11  Q1S
W845  W11  Q1S
W846  W11  Q1S
W847  W11  Q1S
W848  W11  Q1S
W849  W11  Q1S
W850  W11  Q1S
W851  W11  Q1S
W852  W11  Q1S
W853  W11  Q1S
W854  W11  Q1S
W855  W11  Q1S
W856  W11  Q1S
W857  W11  Q1S
W858  W11  Q1S
W859  W11  Q1S
W860  W11  Q1S
W861  W11  Q1S
W862  W11  Q1S
W863  W11  Q1S
W864  W11  Q1S
W865  W11  Q1S
W866  W11  Q1S
W867  W11  Q1S
W868  W11  Q1S
W869  W11  Q1S
W870  W11  Q1S
W871  W11  Q1S
W872  W11  Q1S
W873  W11  Q1S
W874  W11  Q1S
W875  W11  Q1S
W876  W11  Q1S
W877  W11  Q1S
W878  W11  Q1S
W879  W11  Q1S
W880  W11  Q1S
W881  W11  Q1S
W882  W11  Q1S
W883  W11  Q1S
W884  W11  Q1S
W885  W11  Q1S
W886  W11  Q1S
W887  W11  Q1S
W888  W11  Q1S
W889  W11  Q1S
W890  W11  Q1S
W891  W11  Q1S
W892  W11  Q1S
W893  W11  Q1S
W894  W11  Q1S
W895  W11  Q1S
W896  W11  Q1S
W897  W11  Q1S
W898  W11  Q1S
W899  W11  Q1S
W900  W11  Q1S
W901  W11  Q1S
W902  W11  Q1S
W903  W11  Q1S
W904  W11  Q1S
W905  W11  Q1S
W906  W11  Q1S
W907  W11  Q1S
W908  W11  Q1S
W909  W11  Q1S
W910  W11  Q1S
W911  W11  Q1S
W912  W11  Q1S
W913  W11  Q1S
W914  W11  Q1S
W915  W11  Q1S
W916  W11  Q1S
W917  W11  Q1S
W918  W11  Q1S
W919  W11  Q1S
W920  W11  Q1S
W921  W11  Q1S
W922  W11  Q1S
W923  W11  Q1S
W924  W11  Q1S
W925  W11  Q1S
W926  W11  Q1S
W927  W11  Q1S
W928  W11  Q1S
W929  W11  Q1S
W930  W11  Q1S
W931  W11  Q1S
W932  W11  Q1S
W933  W11  Q1S
W934  W11  Q1S
W935  W11  Q1S
W936  W11  Q1S
W937  W11  Q1S
W938  W11  Q1S
W939  W11  Q1S
W940  W11  Q1S
W941  W11  Q1S
W942  W11  Q1S
W943  W11  Q1S
W944  W11  Q1S
W945  W11  Q1S
W946  W11  Q1S
W947  W11  Q1S
W948  W11  Q1S
W949  W11  Q1S
W950  W11  Q1S
W951  W11  Q1S
W952  W11  Q1S
W953  W11  Q1S
W954  W11  Q1S
W955  W11  Q1S
W956  W11  Q1S
W957  W11  Q1S
W95
```

```

Activities  Virtuoso
Dec 18 14:21
/mnt/castor/seas_home/c/ruturajn/cadence616/LVS/si.out
cadence

Devices in the netlist but not in the rules:
  programmed
  Devices in the rules but not in the netlist:
  None
The net-lists match.

          layout schematic
un-satched      0
routed          0
size errors     0
pinroute        0
native          194   194
total           194   194

          nets
un-satched      0
mapped          0
printed         101   101
total           101   101

          terminals
un-satched      0
matched but different type  0   0
total           20   20

Probe files from /mnt/castor/seas_home/c/ruturajn/cadence616/LVS/schematic
deshad.out
netbad.out
segment.out
terminal.out
primnet.out
primdev.out
wunit.out

Probe files from /mnt/castor/seas_home/c/ruturajn/cadence616/LVS/Layout
desbad.out
netbad.out
segment.out
terminal.out
primnet.out
primdev.out
wunit.out
$|
```

Figure 52: LVS Output 2 - Serial In Parallel Out register

7 SRAM ARRAY

7.1 Schematic

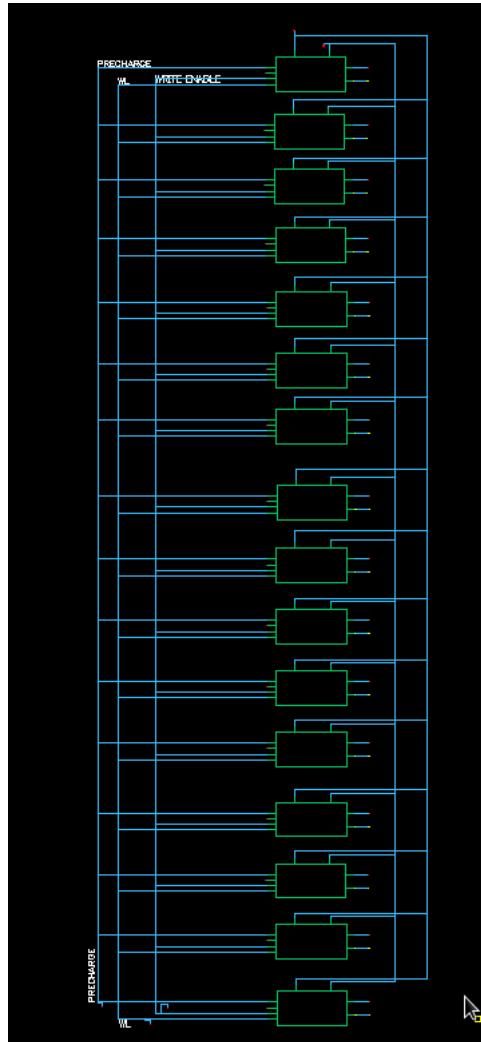


Figure 53: SRAM Array - Schematic

We have designed a SRAM array consisting of 16 SRAM cells cascaded together to feed inputs to the 16:1 Mux. The output BL from each of the individual SRAM cells are connected to the inputs of the 16:1 Mux.

7.2 Symbol

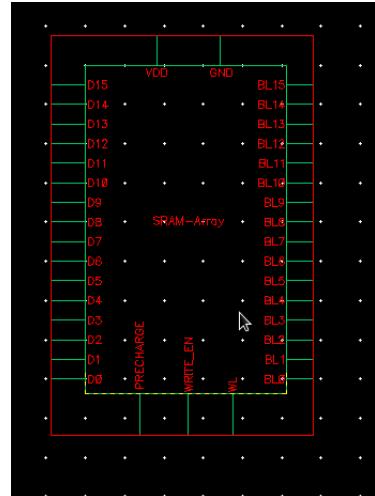


Figure 54: SRAM Array - Symbol

7.3 Test Schematic using symbol

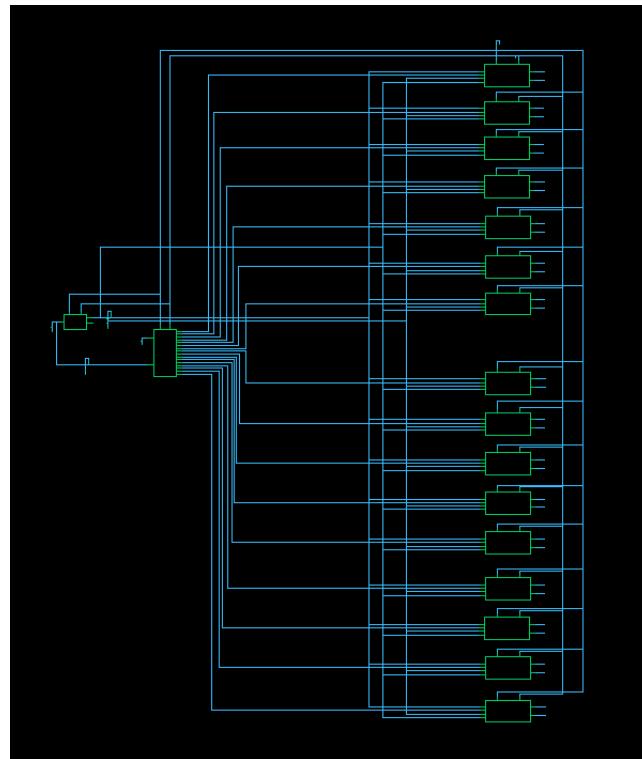


Figure 55: SRAM Array - Test Schematic

The input is fed with a *vbit* source that gives out binary data 1101111010111010. Then the output from the serial in parallel out register is given to each individual SRAM cell in the array.

7.4 Functionality verification

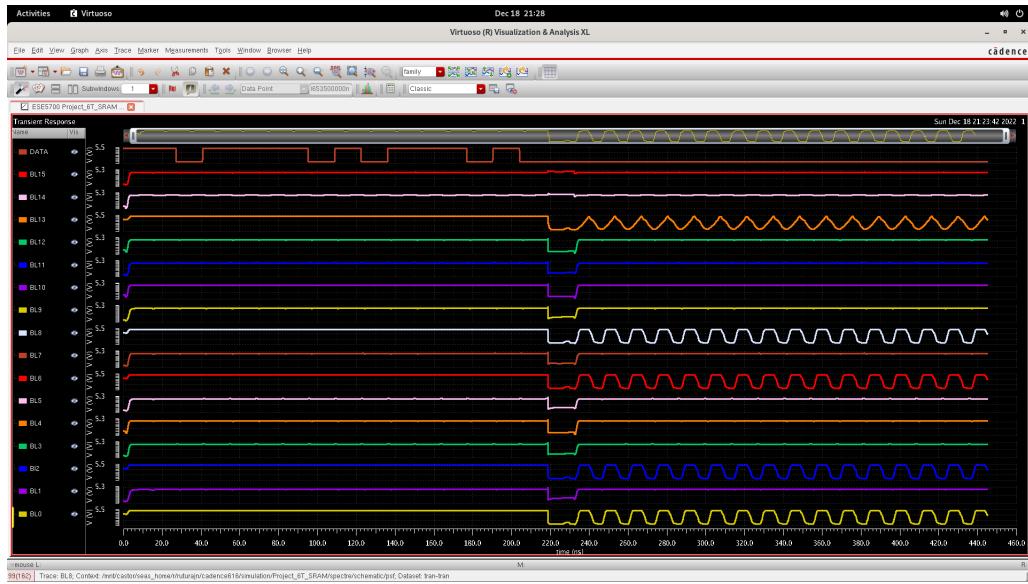


Figure 56: Functionality Verification Graph

Here, we can see that the output at each stage of the SRAM array exactly follows the input data which was fed 1101111010111010 using a vbit source. Reading a 0 produces an oscillating waveform which can be seen in the waveforms. Reading a 1 produces a constant 1. The graph shows that the output replicates the input data and this verifies the functionality of our SRAM array.

7.5 Layout

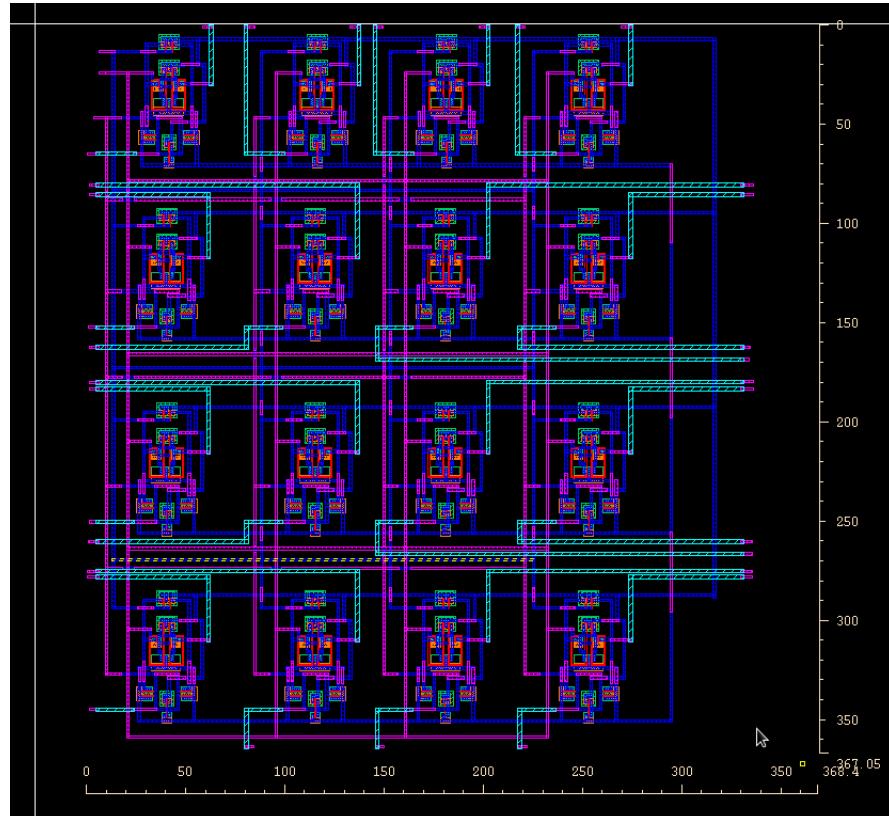


Figure 57: Layout - SRAM Array

7.6 Proof of Passing DRC

Figure 58: DRC - SRAM Array

7.7 Proof of Passing LVS

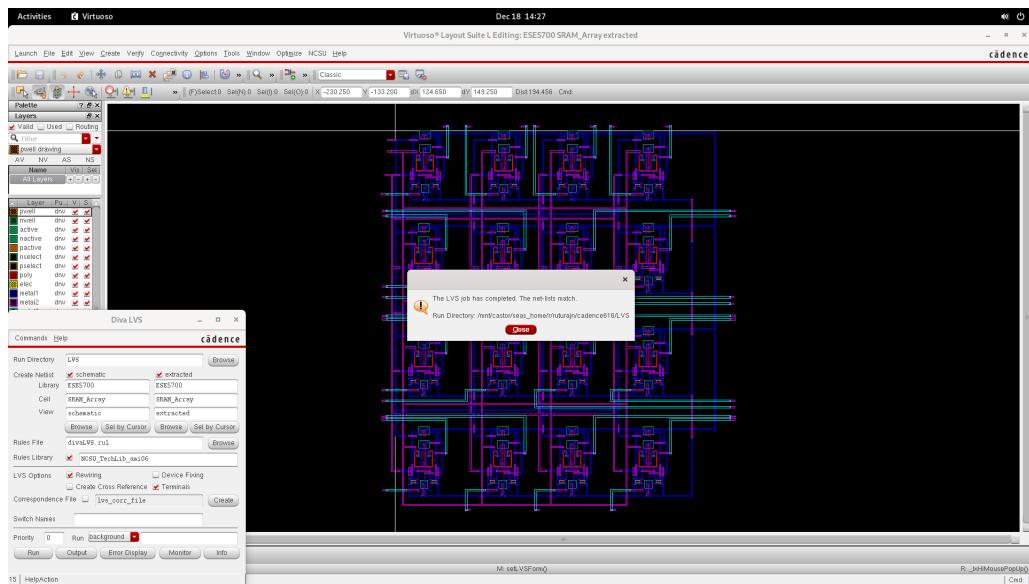


Figure 59: SRAM Array - LVS

```

Activities  | Virtuoso
Dec 18 14:27
/mnt/castor/seas_home/c/ruturajn/cadence616/LVS/si.out
File Edit View Help
#(X)1000: LVS Version 6.1.6 09/01/2015 15:38 (sjf01125) 8
Command line: /opt/pnlun/software/cd-cadence/lvs/tools/lx09v/dfl/bin/ldbt/LVS -dir /mnt/castor/seas_home/c/ruturajn/cadence616/LVS -l -o -t /mnt/castor/seas_home/c/ruturajn/cadence616/LVS/Layout /mnt/castor/seas_home/c/ruturajn/cadence616/LVS/schematic
Link analysis: 0 nodes
Net swapping: it enabled
Duplicating nets: 0
Correspondence points: 0
Compiling DRC LVS rules...
Net-list summary for /mnt/castor/seas_home/c/ruturajn/cadence616/LVS/layout/netlist
nodes 123
nets 123
terminals 123
pins 123
ports 123
DB68 123
144 144

Net-list summary for /mnt/castor/seas_home/c/ruturajn/cadence616/LVS/schematic/netlist
nodes 123
nets 123
terminals 123
pins 123
ports 123
DB68 123
144 144

Terminal correspondence points
N143 N5 H1
N143 N6 H10
N155 N51 H11
N155 N52 H12
N155 N53 H13
N155 N54 H14
N155 N55 H15
N128 N7 H15
N142 N8 H16
N141 N9 H13
N140 N10 H14
N138 N5 H15
N134 N6 H17
N129 N7 H15
N111 N60 H19
N152 N5 H1
N149 N81 H11
N146 N82 H12
N147 N83 H13
N145 N85 H14
N145 N33 H15
N151 N12 H16
N160 N17 H13
N150 N18 H14
N158 N46 H5
N157 N47 H5
N156 N14 H7
N155 N15 H8
N154 N13 H9
N150 N20 PREGND/NVDD
N159 N21 VDD
N154 N51 VDD
N152 N51 VDD
N153 N59 MNTRX_XN
N153 N60 MNTRX_XN

Devices in the netlist but not in the rules:
programmed
Devices in the rules but not in the netlist:
cap net pfet nand4 pmos4

The net-lists match:
      layout schematic
      instances instances
un-matched      0      0
reserved      0      0
  
```

Figure 60: SRAM Array - LVS Output 1

```

Activities  | Virtuoso
Dec 18 14:27
/mnt/castor/seas_home/c/ruturajn/cadence616/LVS/si.out
File Edit View Help
#(X)1000: LVS Version 6.1.6 09/01/2015 15:38 (sjf01125) 8
Command line: /opt/pnlun/software/cd-cadence/lvs/tools/lx09v/dfl/bin/ldbt/LVS -dir /mnt/castor/seas_home/c/ruturajn/cadence616/LVS -l -o -t /mnt/castor/seas_home/c/ruturajn/cadence616/LVS/Layout /mnt/castor/seas_home/c/ruturajn/cadence616/LVS/schematic
Link analysis: 0 nodes
Net swapping: it enabled
Duplicating nets: 0
Correspondence points: 0
Compiling DRC LVS rules...
Net-list summary for /mnt/castor/seas_home/c/ruturajn/cadence616/LVS/layout/netlist
nodes 123
nets 123
terminals 123
pins 123
ports 123
DB68 123
144 144

Net-list summary for /mnt/castor/seas_home/c/ruturajn/cadence616/LVS/schematic/netlist
nodes 123
nets 123
terminals 123
pins 123
ports 123
DB68 123
144 144

Terminal correspondence points
N157 N15 DB
N156 N16 DB
N155 N64 DB
N154 N65 DB
N150 N61 DND
N159 PREGND/NVDD
N154 N55 VDD
N155 N56 NVDD
N153 N59 MNTRX_XN
N153 N60 MNTRX_XN

Devices in the netlist but not in the rules:
programmed
Devices in the rules but not in the netlist:
cap net pfet nand4 pmos4

The net-lists match:
      layout schematic
      instances instances
un-matched      0      0
reserved      0      0
  
```

Figure 61: SRAM Array - LVS Output 2

8 CONFIGURABLE LOGIC BLOCK

8.1 Schematic

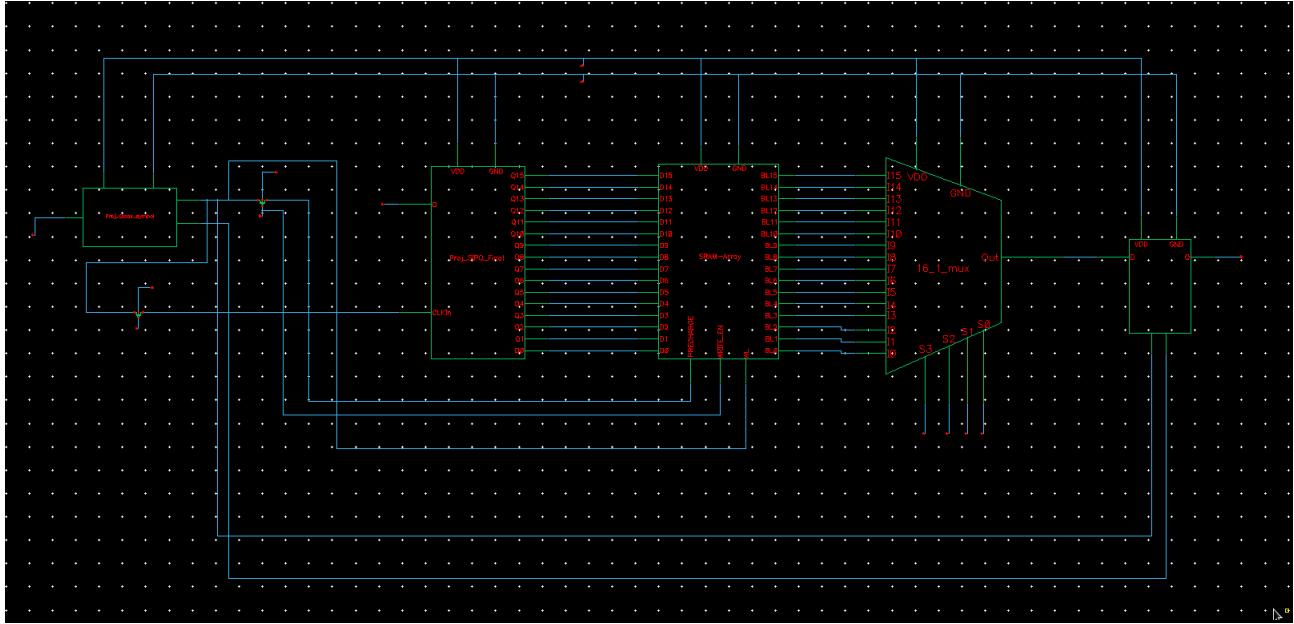


Figure 62: Circuit Schematic for Configurable Logic Block

This schematic represents our Configurable Logic Block and consists of all the individual components that have been designed. The clock module is connected to the Serial In Parallel Out register, to which a serial bitstream is fed. A *LOAD* signal is used to control the Serial In Parallel Out register, which as the name suggests helps us load data into the SIPO 16-bits at a time (i.e. we can switch the state of the Serial In Parallel Out register).

The output from the Serial In Parallel Out register is then given to our SRAM array, which stores the incoming data. Now that the data is stored in the SRAM array, any individual cell is read based on the select inputs of our 16:1 Mux. Finally, to eliminate transients a D Flip Flop is placed at the output.

All the pass transistors used in this schematic(Fig. 62) are minimum sized.

8.2 Symbol

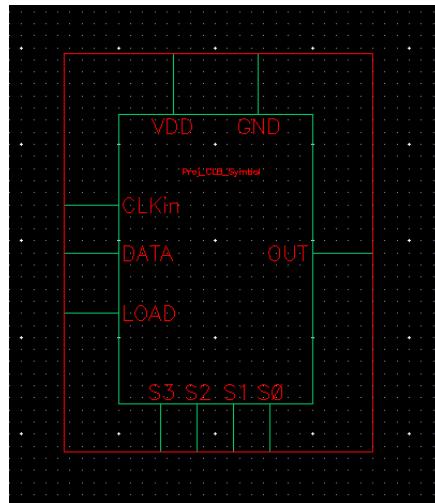


Figure 63: Symbol for Configurable Logic Block

8.3 Functionality at Maximum Frequency

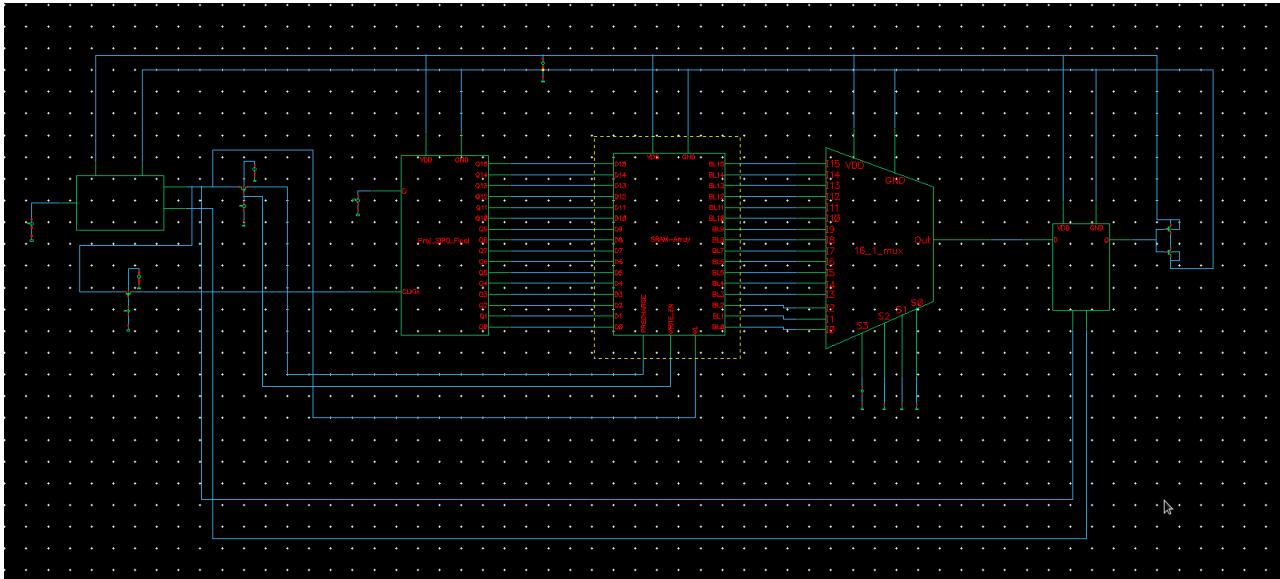


Figure 64: Test Schematic for Maximum Frequency

Verification plot



Figure 65: Verification of the Configurable Logic Block at Maximum Operating Frequency

Fig. 65 indicates that the output follows the input data provided using a *vbit* source depending on the state of the select lines of our 16:1 Mux. The address bits for the LUT are cycled through using *vpulse* sources with varying periods to make sure every combination of the select lines are tested.

We tested the Configurable Logic block for varying periods of CLK_{in} , however, it failed to operate below a time period of 13.6ns. Thus, we had to settle at a time period of 13.6ns for our main clock. This implies that the Maximum operating frequency that our Configurable Logic block operates at is 73.529MHz.

We could have attained higher operating frequency for our Configurable Logic block had we increased the sizes of transistors for most of our components. Since, our SRAM and 16:1 Mux are

not sized to a huge extent , our maximum operating frequency is slightly low, however, this was a conscious design choice keeping in the mind the area that our Configurable Logic block will take.

8.4 Layout

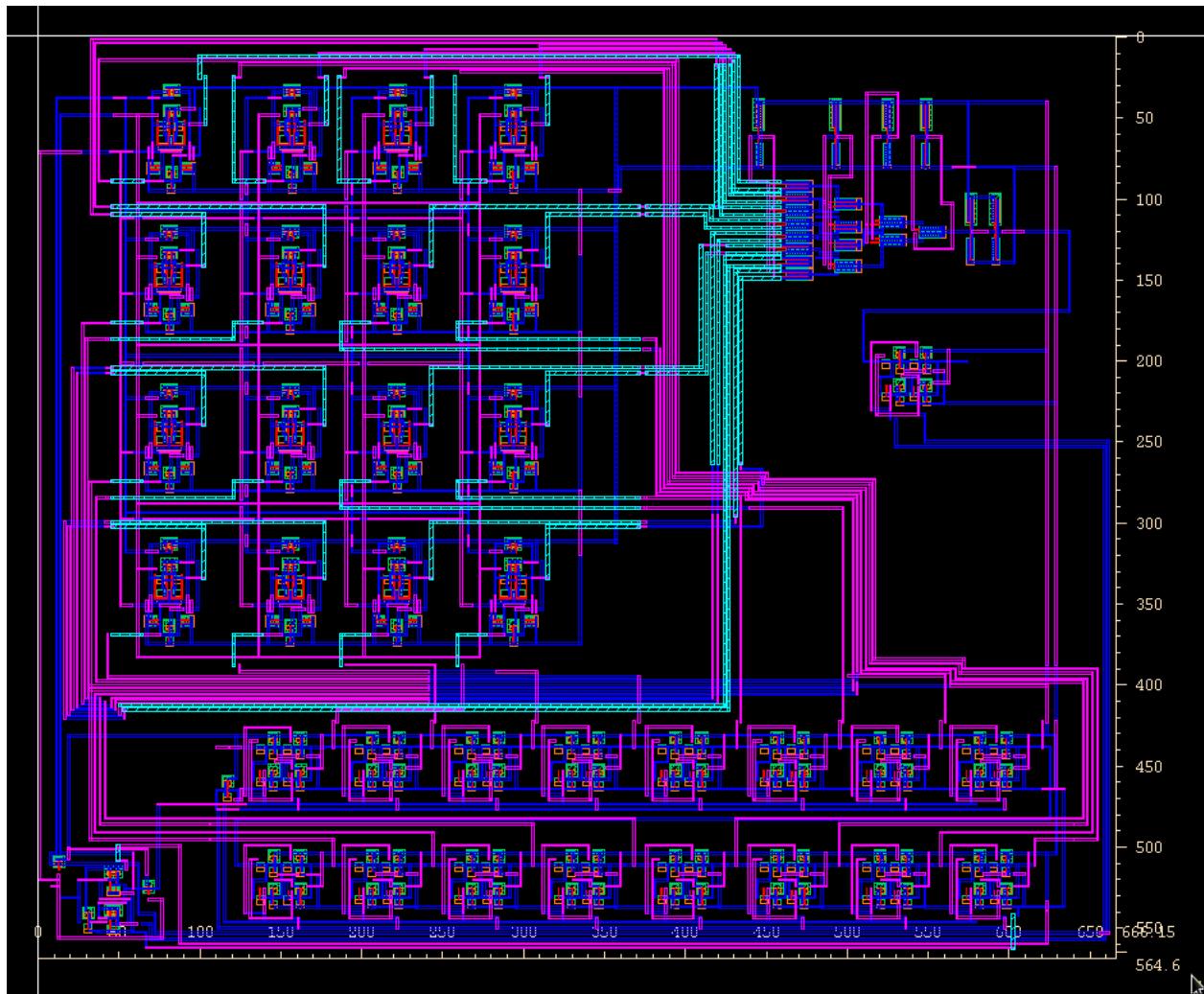


Figure 66: Complete Layout of the Configurable Logic Block

8.5 Proof of Passing DRC

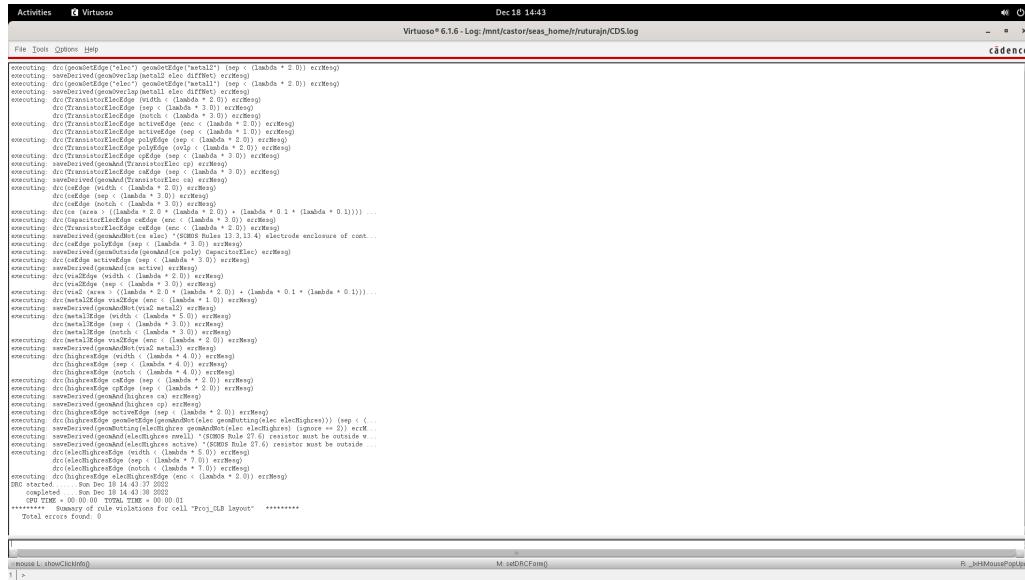


Figure 67: DRC for the Configurable Logic Block

8.6 Proof of Passing LVS

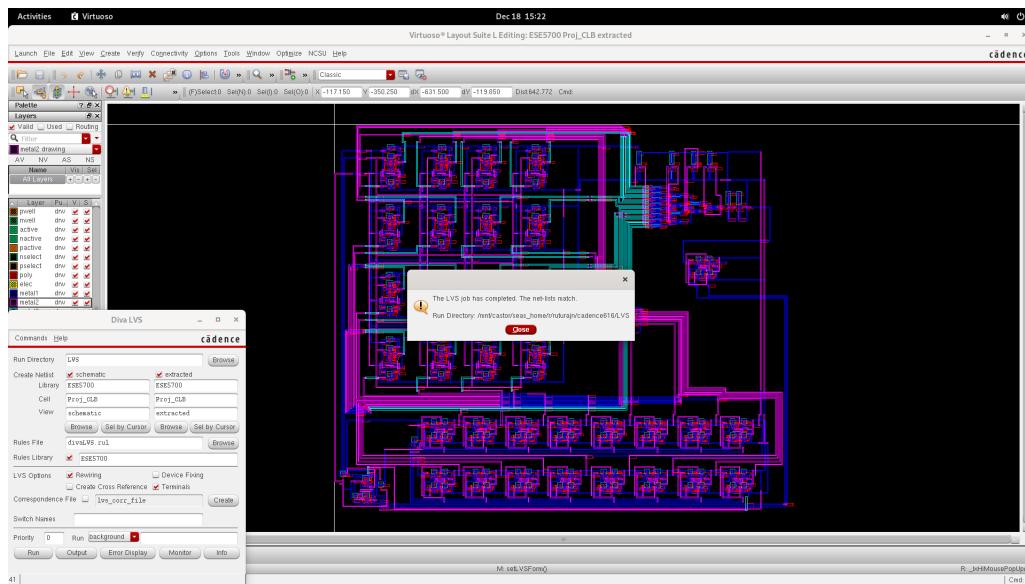


Figure 68: LVS - Configurable Logic Block

```

Activities  Virtuoso
Dec 18 15:23
/mnt/castor/seas_home/c/ruturajn/cadence616/LVS/si.out

LVS Version 6.1.6 09/01/2015 15:38 (sjfcl125) 8
Command line: /opt/pnlun/software/cadence/lolsis/tools.lx09/df11/bin/Dbit/LVS -dir /mnt/castor/seas_home/c/ruturajn/cadence616/LVS -l -o -t /mnt/castor/seas_home/c/ruturajn/cadence616/LVS/Layout /mnt/castor/seas_home/c/ruturajn/cadence616/LVS/schematic
Net mapping is enabled
Dont care points will be correspondence points
Net-list summary for /mnt/castor/seas_home/c/ruturajn/cadence616/LVS/Layout/netlist
  count      nets    terminals
  202        0       162
  162        1       162
  322        2       322
  322        3       322

Net-list summary for /mnt/castor/seas_home/c/ruturajn/cadence616/LVS/schematic/netlist
  count      nets    terminals
  202        0       162
  162        1       162
  322        2       322
  322        3       322

Terminal correspondence points
  N354  N34  GND
  N355  N34  VDD
  N351  N35  GND
  N350  N35  VDD
  N352  N46  GND
  N353  N46  VDD
  N358  N44  S1
  N359  N44  S2
  N360  N45  S3
  N361  N45  S4
  N362  N46  S5
  N363  N46  S6
  N364  N47  S7
  N365  N48  VDD

Devices in the netlist but not in the rules:
  probeout
  Device in the rules but not in the netlist:
  cap net pfet nacod pwoe
The net-lists match

  layout schematic
  instances
  un-watched      0      0
  rewired        0      0
  site errors    0      0
  printed        0      0
  active         484    484
  total          484    484

  nets
  un-watched      0      0
  merged         0      0
  printed        0      0
  total          252    252

  terminals
  un-watched      0      0
  rewired        0      0
  site errors    0      0
  different type  0      0
  total          10    10

Probe file from /mnt/castor/seas_home/c/ruturajn/cadence616/LVS/schematic
deadbad.out:
pathbad.out:
ascrgnet.out:
terabad.out:
42]

```

Figure 69: LVS Output 1 - Configurable Logic Block

```

Activities  Virtuoso
Dec 18 15:23
/mnt/castor/seas_home/c/ruturajn/cadence616/LVS/si.out

N351  N34  DATA
N352  N34  VDD
N360  N5  LOAD
N361  N5  DOUT
N359  N15  GND
N358  N15  VDD
N357  N36  S1
N356  N36  S2
N355  N4  VDD

Devices in the netlist but not in the rules:
  probeout
  Device in the rules but not in the netlist:
  cap net pfet nacod pwoe
The net-lists match

  layout schematic
  instances
  un-watched      0      0
  rewired        0      0
  site errors    0      0
  printed        0      0
  active         484    484
  total          484    484

  nets
  un-watched      0      0
  merged         0      0
  printed        0      0
  active         252    252
  total          252    252

  terminals
  un-watched      0      0
  rewired        0      0
  site errors    0      0
  different type  0      0
  total          10    10

Probe file from /mnt/castor/seas_home/c/ruturajn/cadence616/LVS/schematic
deadbad.out:
pathbad.out:
ascrgnet.out:
terabad.out:
printednet.out:
printeddev.out:
sudit.out:
42]

```

Figure 70: LVS Output 2 - Configurable Logic Block

9 DESIGN METRICS

All our simulations are done using the **Analog Extracted Design**. This includes all the components:

- 16:1 Mux
- SRAM Cell
- Clock
- D Flip Flop
- Serial In Parallel Out Register
- SRAM Array
- Configurable Logic Block

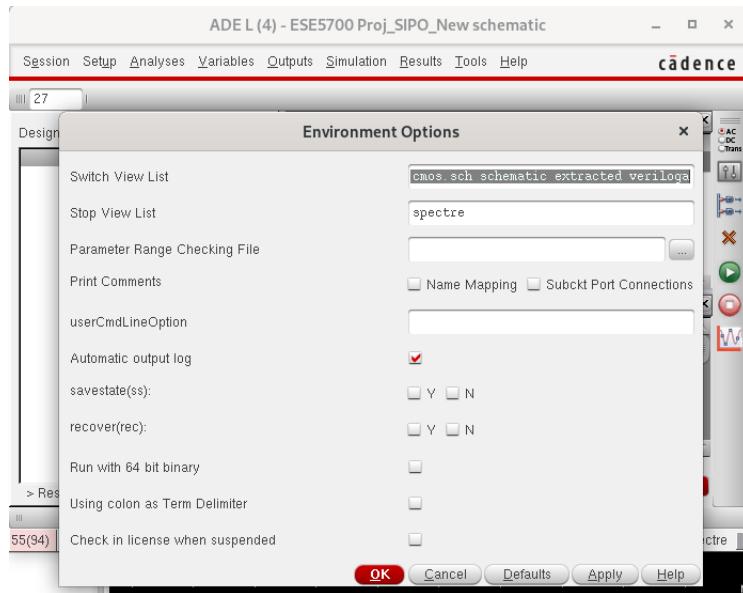


Figure 71: Use of Analog Extracted Design

Fig. 71 shows that we have used the *Analog Extracted Design* for testing functionality. The same has been done for each component of the Configurable Logic Block and the CLB itself as mentioned above.

9.1 Energy

Following are the design metrics for the Configurable Logic Block, namely the loading energy and the active energy.

9.1.1 Loading Energy

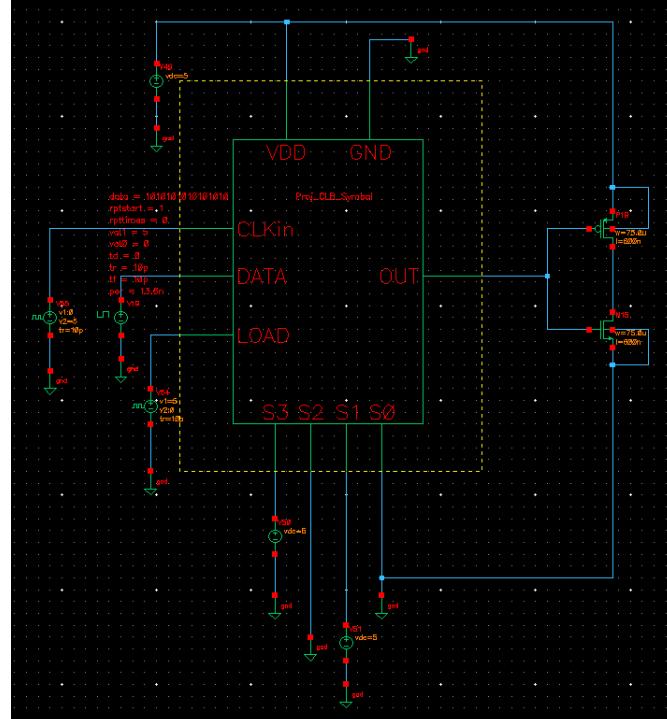


Figure 72: Loading Energy Test Schematic

For measuring the Loading Energy we probed the *VDD* pin of the Serial In Parallel Out register and the SRAM array. Then, the current obtained from the graph was integrated over a time period of 17 cycles (16 cycles for loading our data into the Serial In Parallel Out register and another to write the data into the SRAM array). Finally, the value of energy obtained from both the plots were summed up to give us the total loading energy. Also, the output is loaded with $100C_g$, as mentioned in the handout, for which we have used an inverter with the widths of the PMOS and the NMOS set to $75\mu\text{m}$.

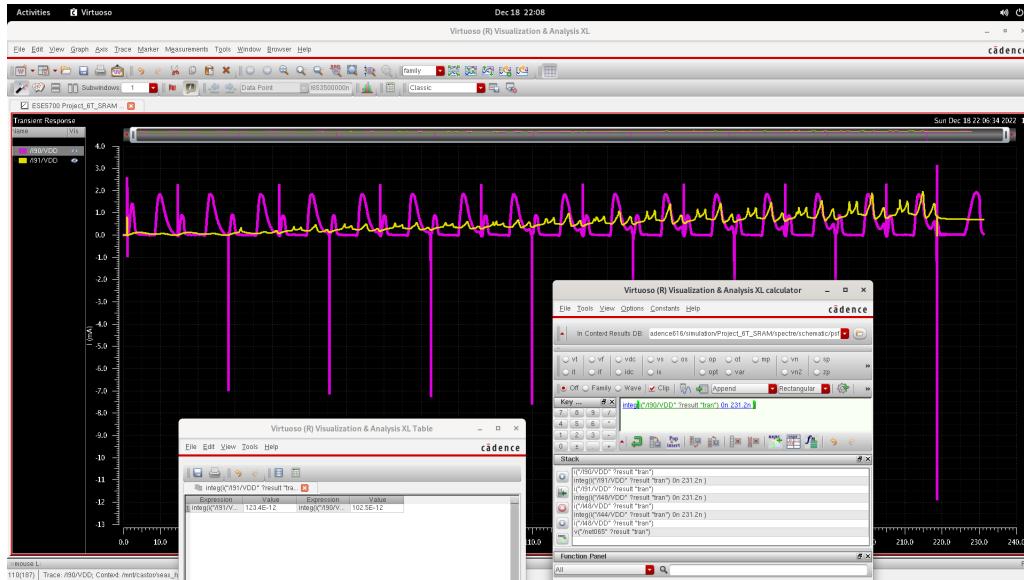


Figure 73: Loading Energy - Current Value calculation



Figure 74: Loading Energy Calculation Graph

9.1.2 Active Energy

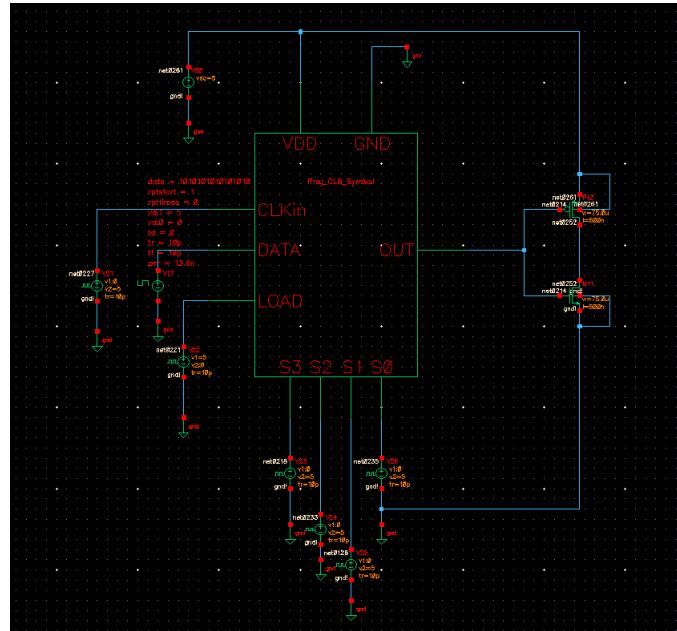


Figure 75: Test Schematic for measuring Active Energy

For measuring the Active Energy we probed the VDD pin of the complete Configurable Logic Block and all the 16 inputs of the LUT. Then, the current obtained from the graph was integrated over a time period of 33 cycles (16 cycles for loading our data into the Serial In Parallel Out register and another to write the data into the SRAM array, finally 16 more to cycle through all the select lines/address bits). Finally, the value of energy obtained from all the plots were summed up to give us the total active energy.

Also, the output is loaded with $100C_g$ as mentioned in the handout, for which we have used an inverter with the widths of the PMOS and the NMOS set to $75\mu\text{m}$.

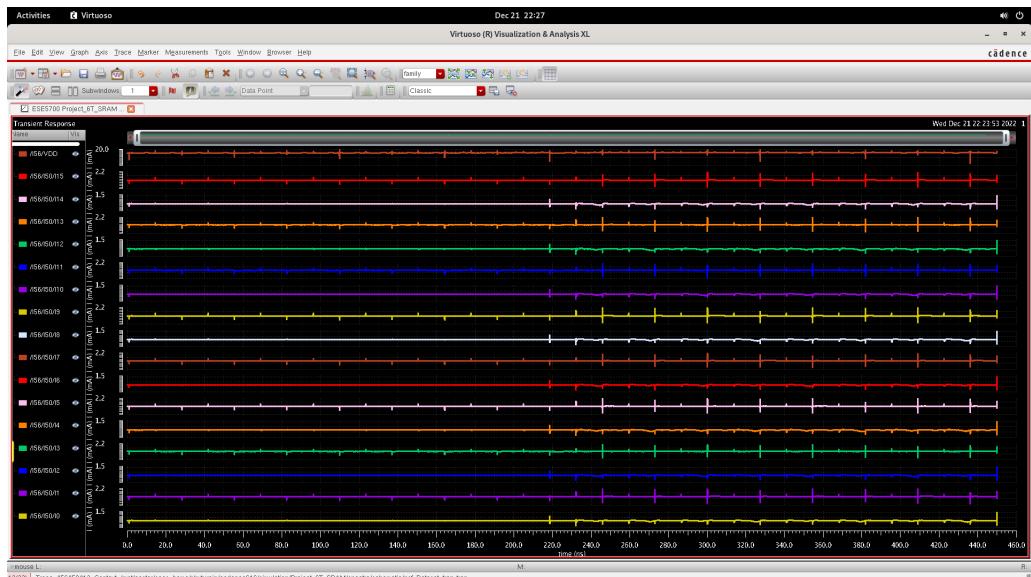
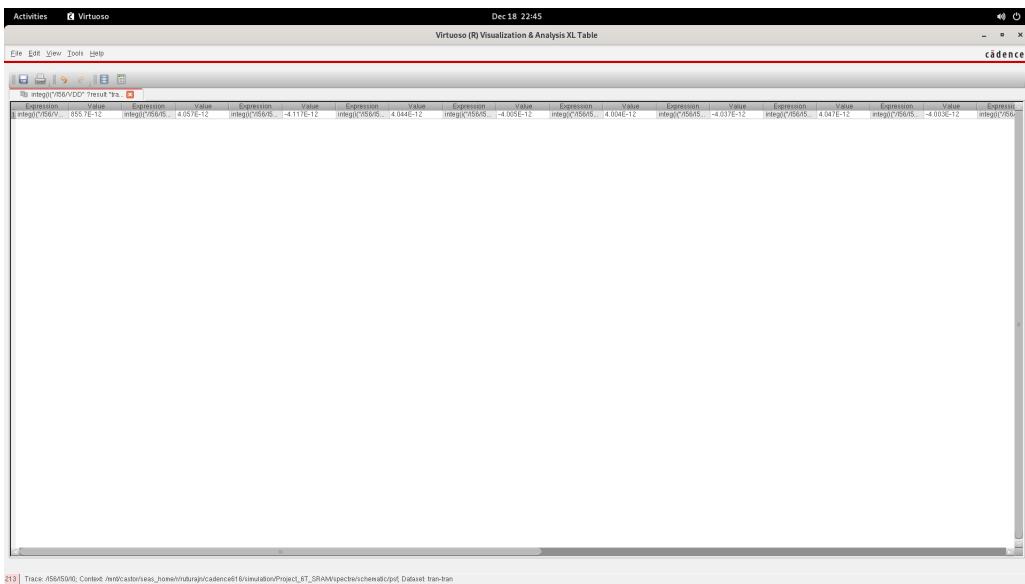


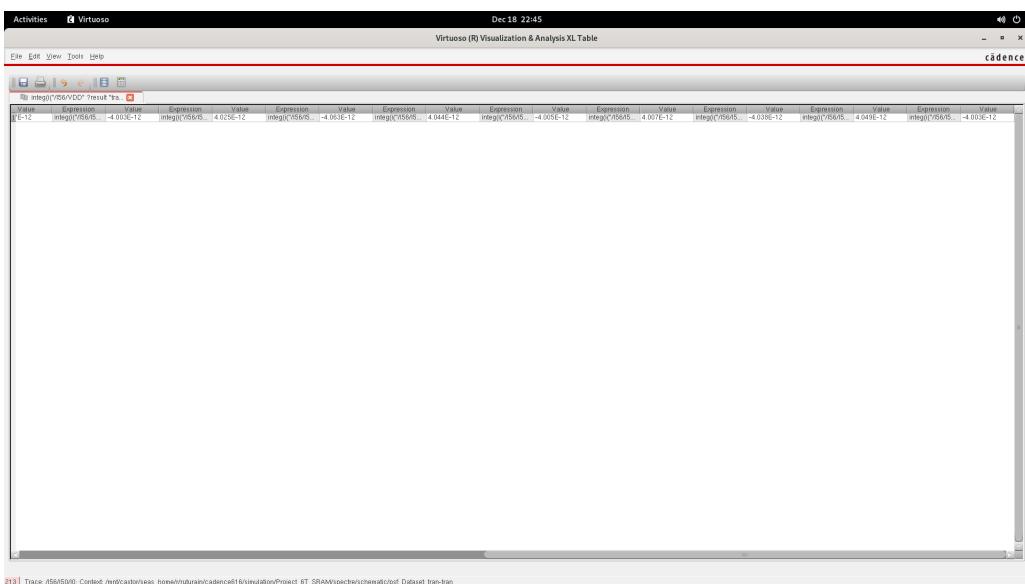
Figure 76: Current Plot



The screenshot shows a Cadence Virtuoso R Visualizer window titled "Virtuoso (R) Visualization & Analysis XL.Table". The table has columns labeled "Expression" and "Value" repeated 16 times. The first few rows of data are as follows:

Expression	Value	Expression	Value	Expression	Value	Expression	Value	Expression	Value	Expression	Value	Expression	Value	Expression	Value	Expression	Value
$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	655.7E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	4.057E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	-4.117E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	4.044E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	-4.056E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	4.037E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	-4.056E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	4.047E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	-4.036E-12
$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	655.7E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	4.057E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	-4.117E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	4.044E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	-4.056E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	4.037E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	-4.056E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	4.047E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	-4.036E-12

Figure 77: Current Integral values for some of the plots - 1



The screenshot shows a Cadence Virtuoso R Visualizer window titled "Virtuoso (R) Visualization & Analysis XL.Table". The table has columns labeled "Expression" and "Value" repeated 16 times. The first few rows of data are as follows:

Expression	Value	Expression	Value	Expression	Value												
$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	655.7E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	-4.036E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	4.025E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	-4.062E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	4.044E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	-4.056E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	4.037E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	-4.036E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	4.049E-12
$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	655.7E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	-4.036E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	4.025E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	-4.062E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	4.044E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	-4.056E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	4.037E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	-4.036E-12	$\int \text{integ}(VDD/VDD) \cdot \text{result} * t_s$	4.049E-12

Figure 78: Current Integral values for some of the plots - 2

9.1.3 Average Energy

The Loading Energy for our Design is given by,

$$\text{LoadingEnergy} = (\int I) \times V_{DD} = (123.4\text{pA} + 102.5\text{pA}) \times 5 = 1129.5\text{pJ}$$

The Active Energy for our Design is,

$$\text{ActiveEnergy} = (\int I) \times V_{DD} = (\dots + 4.003\text{pA}) \times 5 = 4601.24\text{pJ}$$

Therefore the Average energy is,

$$AverageEnergy = (0.8 \times ActiveEnergy) + (0.2 \times LoadingEnergy)$$

$$AverageEnergy = (0.8 \times 4601.24pJ) + (0.2 \times 1129.5pJ)$$

$$AverageEnergy = 3680.992pJ + 225.9pJ$$

$$AverageEnergy = 3906.892pJ$$

9.2 Maximum Operating Frequency

The max operating frequency that we obtained was **73.529 MHz**, which was based on our time period of $13.6ns$ at which our main clock is operating.

9.3 Figure of Merit

The area that we obtained for the whole CLB was $(666.15 \times 666.15)\mu m^2$. Hence, our FOM turns out to be,

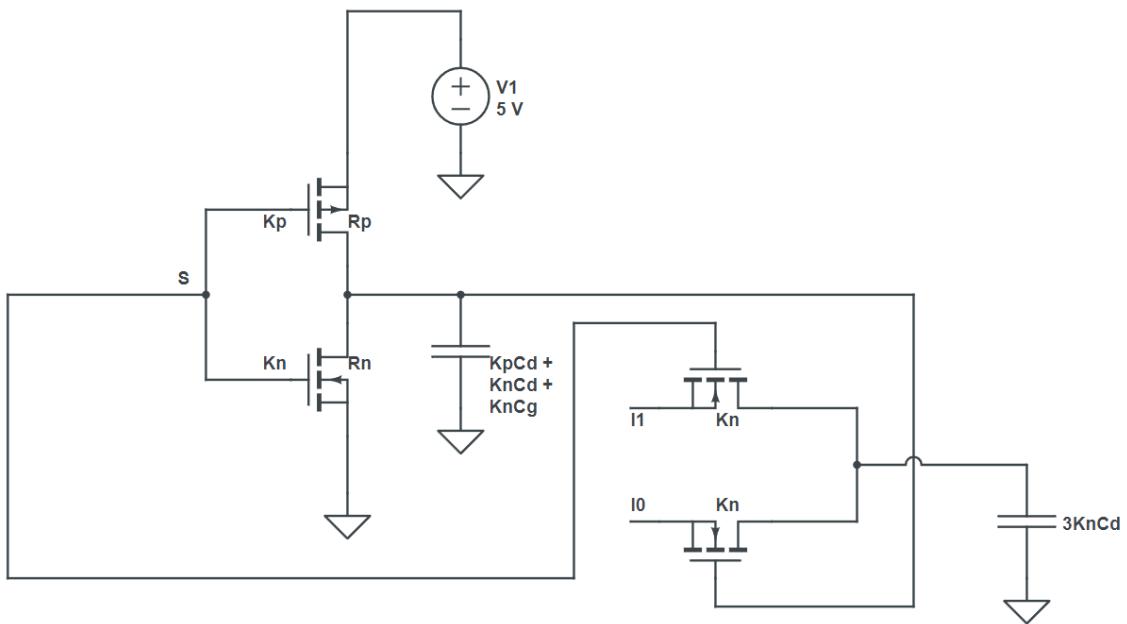
$$FOM = \text{area} \times \text{averageEnergy} \times \frac{1}{\text{maxFrequency}}$$
$$FOM = (666.15 \times 666.15)\mu m^2 \times 3906.892pJ \times \frac{1}{73.529MHz}$$
$$FOM = 2.3578 \times 10^{-23}m^2Js$$

9.4 Summary Table

Max. Operating Frequency	Area	Average Energy
73.529MHz	$(666.15 \times 666.15)\mu m^2$	3906.892pJ

APPENDIX : REPRESENTATION OF ELMORE DELAY FOR THE LUT

Stage 1 - Input Drive for 16:1 Mux



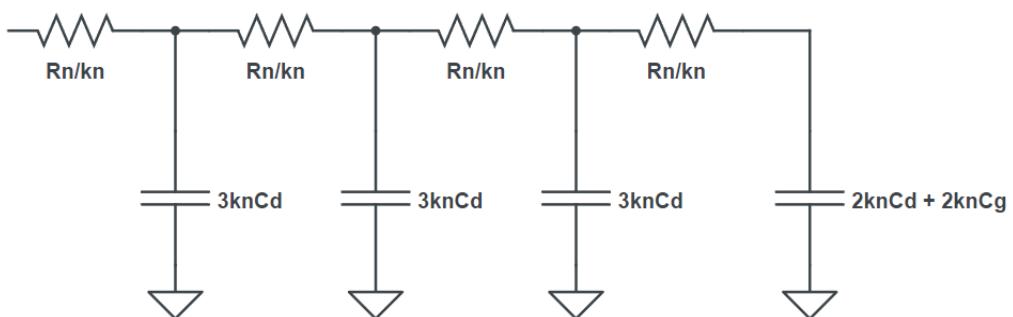
The worst case for the Elmore delay is when the PMOS in the inverter is ON, and '1' passes through 'I0'.

$$\tau_p = 0.69 \left(\frac{R_p}{k_p} (k_p C_d + k_n C_d + k_n C_g) + \frac{R_n}{k_n} (3k_n C_d) \right)$$

$$\tau_p = 0.69 \left(R_p C_d + R_p \frac{k_n}{k_p} C_d + R_p C_g + 3R_n C_d \right)$$

$$\tau_p = 0.69 \left(R_p \left(\left(C_d + \frac{k_n}{k_p} C_d \right) + C_g \right) + 3R_n C_d \right)$$

Stage 2 - 16:1 Mux (Device Under Test)



The RC Delay circuit for the 16:1 Mux is shown above. At any point in time only 1 of the multiplexers in each stage will be switched ON, and hence they will contribute to the delay.

$$\tau_p = 0.69 \left(\frac{R_n}{k_n} (3k_n C_d) + \frac{2R_n}{k_n} (3k_n C_d) + \frac{3R_n}{k_n} (3k_n C_d) + \frac{4R_n}{k_n} (2k_n C_d + k_n C_g + k_p C_g) \right)$$

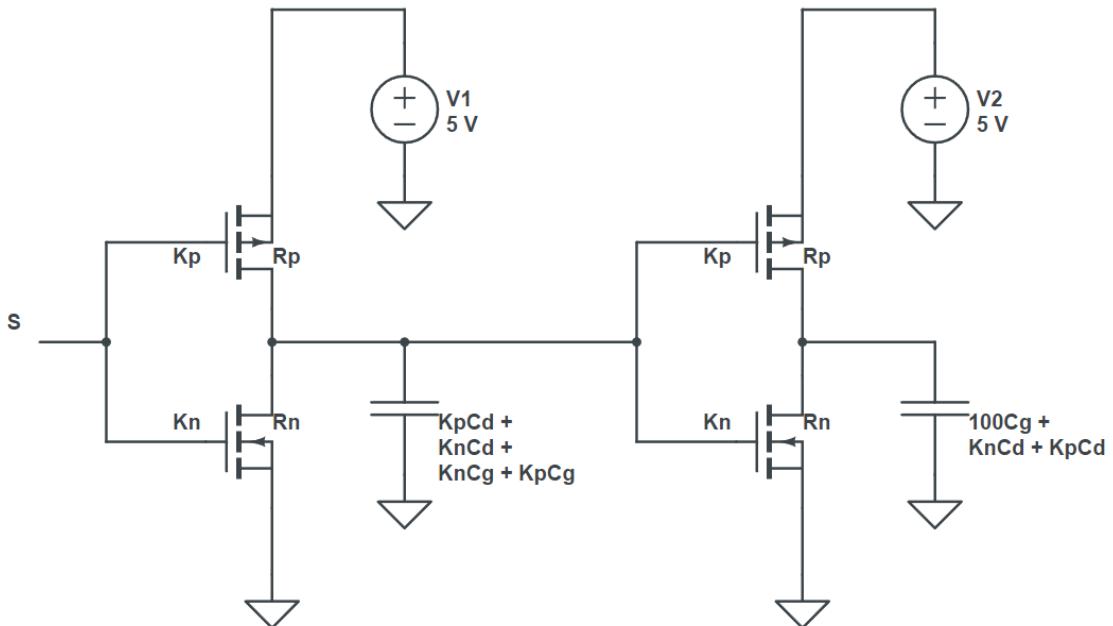
$$\tau_p = 0.69 \left(26R_n C_d + 4R_n C_g + 4R_n \frac{k_p}{k_n} C_g \right)$$

The delay for the 4 inverters that are placed for driving the select lines of our 16:1 mux also needs to be added. The last stage consisting of 8 multiplexers becomes the worst case, since the select line needs to drive 8 multiplexers. Also, the path that goes via \bar{S} , will be the one that is considered since the inverter is also a part of it. Hence, that delay will need to be included in our calculations.

The resulting equation then comes out to be,

$$\tau_p = 0.69 \left(\left(26R_n C_d + 4R_n C_g + 4R_n \frac{k_p}{k_n} C_g \right) + 4 \left(\frac{R_n}{k_n} (9C_g + k_p C_g + k_p C_d + k_n C_d) \right) \right)$$

Stage 3 – Restoration with Buffer



Finally, the delay for the restoration stage is calculated. Here we assume the output is '1' and therefore the 'NMOS' for the first inverter will be switched 'ON' and the 'PMOS' for the second inverter will be 'switched ON'.

$$\tau_p = 0.69 \left(\frac{R_n}{k_n} (k_n C_g + k_p C_g + k_n C_d + k_p C_d) + \frac{R_p}{k_p} (100C_g + k_n C_d + k_p C_d) \right)$$

Complete Delay

Putting together the delay for the 1st, 2nd and 3rd stage we obtain the complete delay equation,

$$\begin{aligned}\tau_p = & 0.69 \left(R_p \left(\left(C_d + \frac{k_n}{k_p} C_d \right) + C_g \right) + 3R_n C_d \right) \\ & + 0.69 \left(\left(26R_n C_d + 4R_n C_g + 4R_n \frac{k_p}{k_n} C_g \right) \right. \\ & \left. + 4 \left(\frac{R_n}{k_n} (9C_g + k_p C_g + k_p C_d + k_n C_d) \right) \right) \\ & + 0.69 \left(\frac{R_n}{k_n} (k_n C_g + k_p C_g + k_n C_d + k_p C_d) + \frac{R_p}{k_p} (100C_g + k_n C_d + k_p C_d) \right) \\ \tau_p = & 0.69 \left\{ R_n \left[C_d \left(34 + 5 \frac{k_p}{k_n} \right) + C_g \left(4 + \frac{36}{k_n} + 9 \frac{k_p}{k_n} \right) \right] + R_p \left[C_d \left(2 + 2 \frac{k_n}{k_p} \right) + C_g \left(\frac{100}{k_p} + 1 \right) \right] \right\}\end{aligned}$$

It can be seen from the equation above that if we assume $k_p = k_n = k$, the expression for τ_p reduces to,

$$\begin{aligned}\tau_p = & 0.69 \left\{ R_n \left[C_d (34 + 5(1)) + C_g \left(4 + \frac{36}{k} + 9(1) \right) \right] + R_p \left[C_d (2 + 2) + C_g \left(\frac{100}{k} + 1 \right) \right] \right\} \\ \tau_p = & 0.69 \left\{ R_n \left[39C_d + C_g \left(13 + \frac{36}{k} \right) \right] + R_p \left[4C_d + C_g \left(\frac{100}{k} + 1 \right) \right] \right\}\end{aligned}$$

This shows that k is inversely proportional to the delay and therefore as k increases the value for delay decreases.