Objectives :

- To familiarize with schematic-driven layout design.
- To perform schematic-level verification, DRC and LVS.
- To perform post-layout simulation of NAND Gate.

Software Requirements :

- Cadence Virtuoso

Design Process:

- Schematic Design :

We created the schematic design of the NAND gate

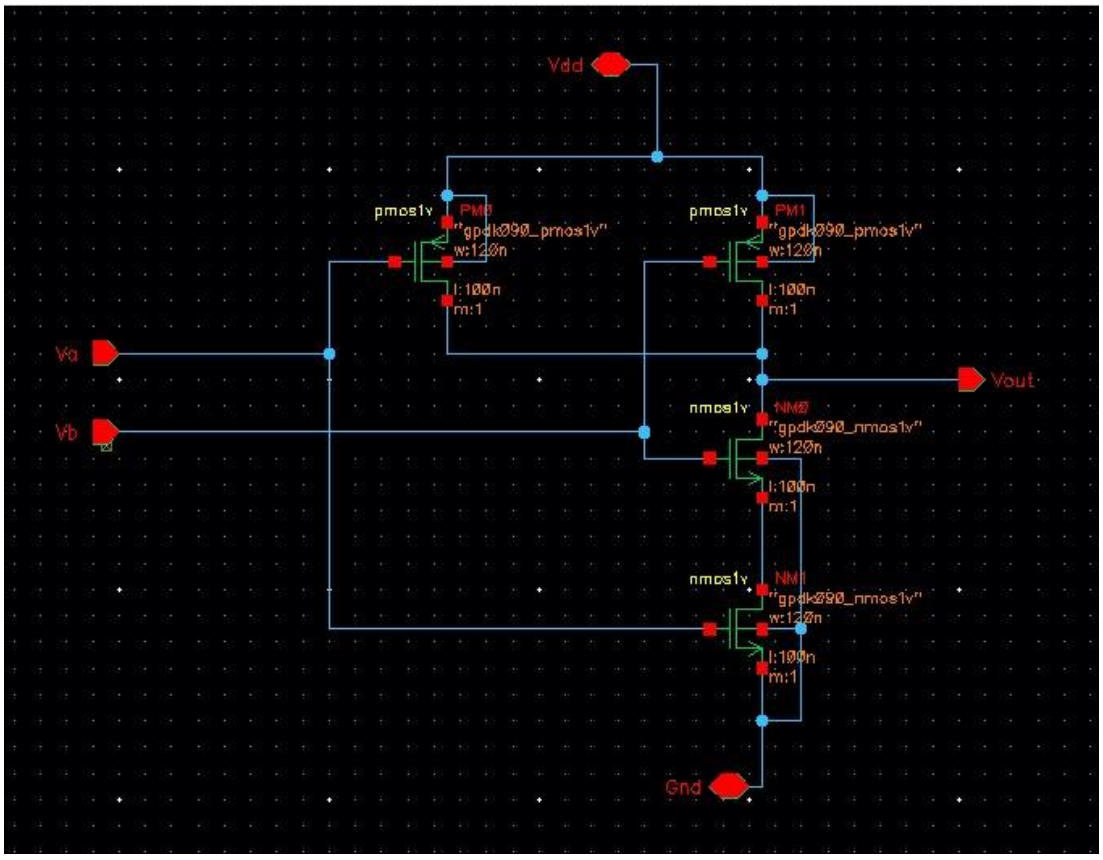

Figure 01: Schematic diagram of NAND gate

• Layout Design:

The 'Virtuoso Layout Editor XL' window automatically generates every component that is specified in the schematic that we extract. There will be the first pin and transistor. To provide room for all components, the bounding box was resized. Before starting to design the layout, we need to set the layout display configuration Minor Spacing 0.01, Major Spacing 0.1, X Snap Spacing 0.005 Y and Snap Spacing 0.005. Then use the path tool to join several levels together. By hitting the 'o' key on the keyboard, construct through may be used to link one layer to another (for example, Poly to Metal1 or Metal1 to Metal2). PMOS should link to VDD for its supply, while NMOS should connected to gnd

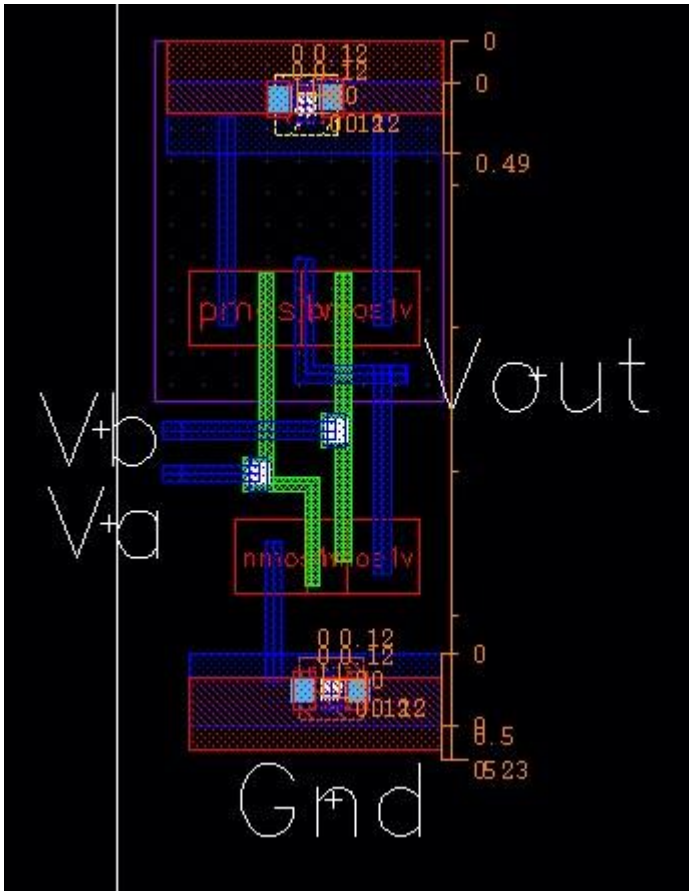

Figure 02: Layout design of NAND gate

DRC Check:

DRC Using Assura :

- Assura technology lib was selected by executing Assura > Technology.
- Assura > Run DRC was executed.
- All errors were solved.

Figure 03: Check DRC error

LVS Using Assura:

- Assura > Run LVS. Was executed.
- All mismatches were solved.

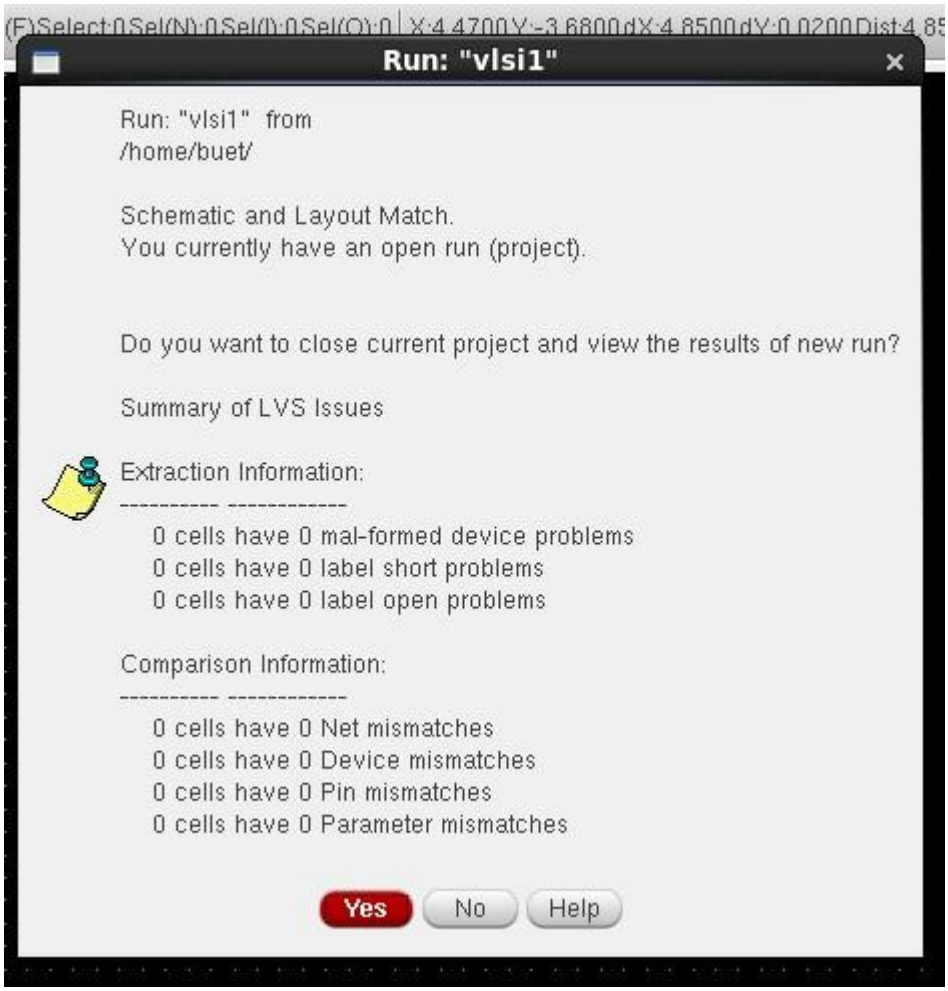

Figure 04: Checked LVS error

Parasitic Extraction Using RCX:

- Assura > Open Run was executed.
- Final error free LVS run name was selected.
- Assura > Run RCX was executed.
- Extracted View in the output field under Setup tab was selected.
- RC as Extraction Type was selected.
- Name of the reference node was given as ‘gnd!’
- Finally we have the extracted view.

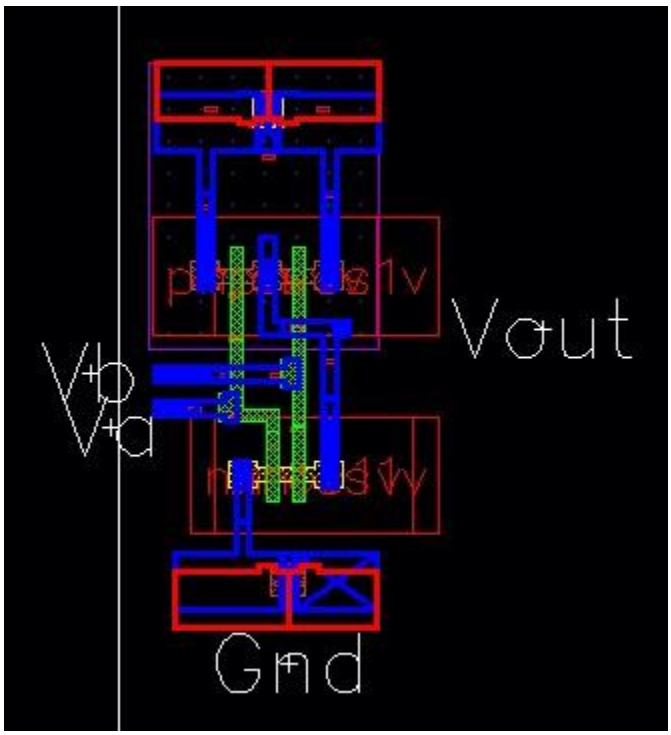

Figure 05 : Extracted View

- ADE L from the av_extracted view was launched.
- After setting everything else, Outputs > To be plotted > Select on design was executed.
- Final result was simulated and observed which is shown in the following two figures :

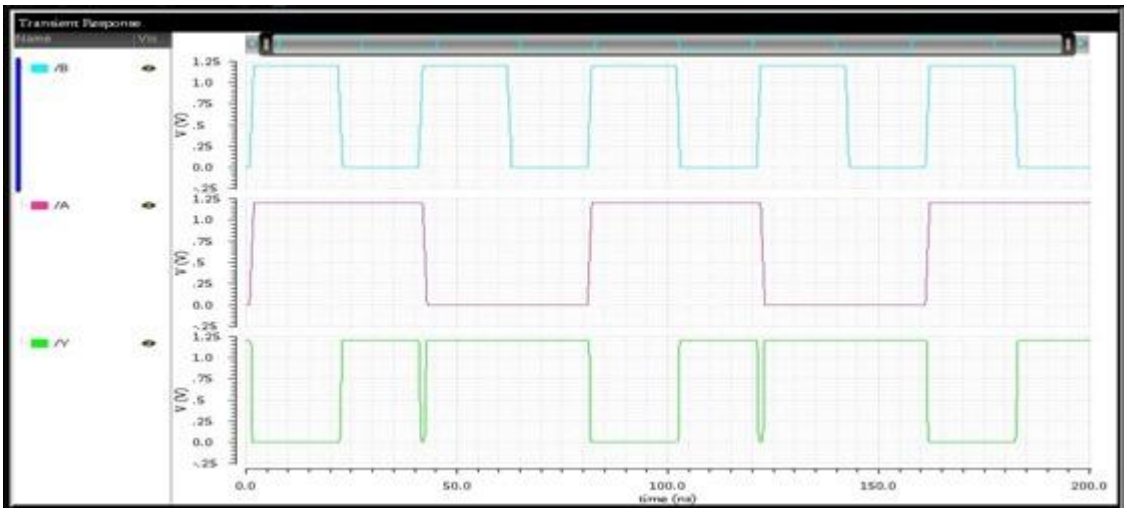

Figure 06: Simulation of NAND gate

Home Task :

1.Difference between stacked and unstacked transistors in post layout simulation is given
below:

Table 1: Stacked vs Unstacked Transistors

Stacked Transistors	Unstacked Transistor
Occupies less area by sharing source/drain terminals	Occupies less area by sharing source/drain terminals
Simplifies routing with fewer connections needed	Increases routing complexity with more connections required
Reduced parasitic capacitance and resistance	Increased parasitic capacitance and resistance
Faster switching times and improved performance	Slower switching times and improved performance
Lower power dissipation due to minimized parasitics	Higher power dissipation due to minimized parasitics
Simplified layout design, easier to manage connections	Complex layout design, harder to manage connections

Why Stacked Transistor?

Transistors stacked with shared source/drain terminals take up less space when it comes to contacts and metal routing, which minimizes parasitic effects and improves circuit performance. The strategy is required to guarantee the best possible area-performance trade off for extensive digital designs. Superior routing and higher density can be achieved without compromising the transistors' proper connectivity. In conclusion, stacking transistors improves performance, increases routing simplicity, and results in a far more efficient layout all of which are essential for small and high-performance circuit designs.

Discussion :

1. The stacked transistors were used for the 2-input NAND gate in order to minimize the area
needed for routing and connections. In this sense, space was conserved inside the bounding
box.
2. In doing so, the parasitic resistance and capacitance were further decreased, which enhanced
switching efficiency and decreased propagation delays inside the circuit.
3. A few of the simulated validations that were carried out to check if the layout was operating
properly under various input situations and within the specification bounds were DRC, LVS,
and RCX.