PART-B_LIST_MICROWIND_DSCH_TOOLS

06 November 2020 11:38

1) Revised Performance Order of PART-B Assignments prescribed by SPPU

2) Hands-On Learning of TOOL : MICROWIND 3.1 (LAYOUT EDITOR)

- MICROWIND VS MultiSim , PROTEUS
- Invoking the Tool from HARD-DISK
- Tool Workspace
- Target Technology
- Tool GUI
- Correct Starting Point for Drawing LAYOUT
- Enlisting STEPS to DRAW Layout of CMOS INVERTER
- Demonstrating STEPS Enlisted above
- Observing Various Simulations (Vin-Vo vs t , Vo vs Isupply , Vo vs Vin)
- Noting DYNAMIC Power Dissipation, P(dynamic) of Circuit
- Mathematical Model for P(dynamic)
- Seeing Effect of Changing fclk , C(load) & Vdd on P(dynamic)

(I) List of PART-B Assignments:-

- Draw Layout
- Simulate
- Obseive the o/ps
- Analyze & Conclude

(1) Combinational circuits

1 (a) CMOS Inverter
2 (b) CMOS NAND Gate
3 (c) CMOS NOR Gate
4 (d) CMOS AND — Gate
(e) CMOS OR Gate

(7) (f) CMOS Half-Adder → 1 EXOR = $\bar{A}.B + A.\bar{B}$ ⟹ 22 mosfets (2 6 6 6 2)
→ 1 AND = A.B ⟹ 6 mosfets
(28)

1 TG ↓ (2) 2:1 MUX — (1) → XOR, (2) → AND } < 10

(5) → (6) → (7)

(2) 2:1 MUX using TG Logic

(8) (3) 1-bit SRAM Memory (6T cell) → NMOS S/W
→ TG S/W

(2) MicroWind vs Multisim, Proteus

Similarity → All 3 are Electronics CAD Tools

Difference → Multisim , Proteus (Circuit schematic Editors)

- Workspace Represents a PCB
- All Dimensions are in cms/mm

- All Dimensions are in cms/mm

* MicroWind is a - VLSI CAD Tool
  - BackEnd Tool
  - Layout Editor
  (Making ICs)
- Workspace of μWind Represents a Si-Wafer
- All dimensions are in nm

5 lambda
0.250 μm

$5\lambda = 0.250\,\mu m$

$= 250\,nm$

$\therefore \lambda = 50\,nm$

* Target Technology = $2\lambda = 100\,nm$

Foundry
Node
Process
Min. Feature size
Channel length (L)

$\lambda = 50\,nm$

250 nm = 5λ

* Calculating Foundry value from scale gives us "2λ" ⇒ Fabricated channel length (L)

* Rule-file ⇒ Available channel length (L_{eff})

$L_{eff} = L - 2(\Delta L)$

ΔL : Width of Depletion Regions @ S–B & D–B

* CSP for ED ⇒ Block Diagram (on paper)
* CSP for AB ⇒ TT/FT/Bool. exp'n/State-Diagram

⊛ CSP for Making Any Layout in μWind is

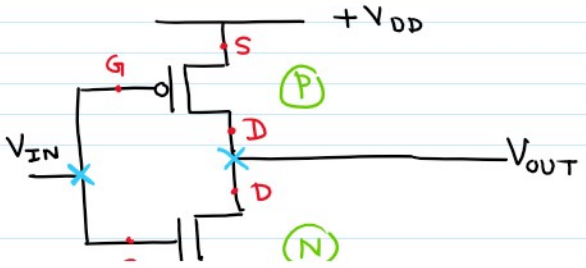
⊛ Transistor-Level schematic (circuit) of the system (on paper)

→ MOSFETs

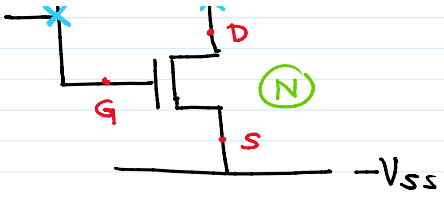
⊛ STEPS for Drawing Layout of CMOS Inverter:-

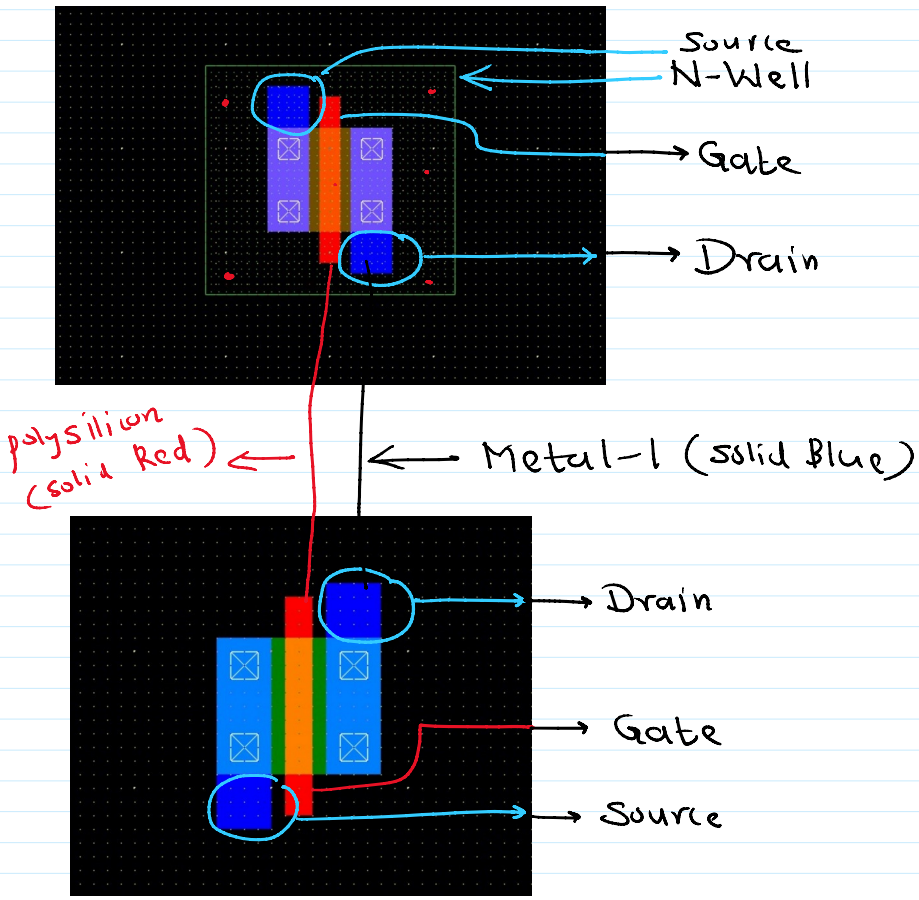
1. Draw the TLS of CMOS Inverter on paper.
2. Terminalise the circuit Diagram (Indicate D, G, S) (Pallete of Layers)
3. Place all MOSFETs into Workspace (using MOS Generator in POL)
4. Align MOSFETs suitably (Hori. / vert. / Rotate) [stretch/move, Rotate] to enable easy connectivity betn their Terminals
5. Connect Terminals of MOSFETs to each other as per point-2
6. Apply $+V_{DD}$ & $-V_{SS}$ to the system
7. Polarise all N-wells in your circuit → Connect N-well to $+V_{DD}$ } - Ensures that N-well is not floating
   - There is No Electrical Rule Error
8. Apply all I/p signals as per point-4
9. Define all o/p Nodes as per point-2
10. Perform DRC (Design Rule Check)
11. Simulate, Observe o/ps, Analyse, Conclude.

⊛ STEP-1 :- TLS of CMOS Inverter.



STEP-2 :- Terminalised





✳ $P_{dyn} = C_L \cdot (V_{DD})^2 \cdot f_{clk}$ → Double, Halve } See effect on P_{dyn}

C_L → Load capacitance; $(V_{DD})^2$ → Bias vtg; f_{clk} → clock freq

Repeat the same for C_L

$\propto C_L$ → Add virtual capacitance (From POL)

$\propto (V_{DD})^2 = f(\text{foundry})$

$\propto f_{clk}$ ✓

✳ A new ; but standard value of $+V_{DD}$ can be used only by changing the foundry

File → Select foundry (Menu)

File → Select foundry
(Menu)

✱✱ Which layers to be used connecting Terminals ?

D ⟷ D
S ⟷ S
D ⟷ S
} METAL-1

G ⟷ G → Polysilicon

✱ O/p is always available on Metal-1 Layer

✱ I/p is always given on Gate Terminal ; hence on Polysilicon Layer

---

III Using DSCH [Open Tool by clicking DSCH3.exe inside Tool's folder]

STEPS:-

1. Draw Either Transistor/Gate-Level Schematic using SYMBOL LIBRARY
2. Save it in a Pre-defined folder with a self-explanatory name (*.sch)
3. Simulate the schematic & verify functionality.
4. Generate "Verilog File" from FILE → Make Verilog File
5. Change Default File Name (example.sch) to self-explanatory.
6. Click ok
7. Open MICROWIND Tool
8. Go To (Menu) COMPILE → Compile Verilog File
9. Browse to your destination folder of Verilog File
10. Select File Type to " All Files (*.*)
11. Select your " Verilog code " contained *.txt file
12. Click " COMPILE "
13. Click " Back to Editor "
14. Simulate