Lec# 3.
Layout editor, cell concepts, design hierarchies

1. Layout editor

![Diagrams of n-wells and active areas]

- **n-well**

  - **Wnw**: minimum width of an n-well mask feature
  - **Snw-nw**: minimum edge-to-edge spacing of adjacent n-wells

2. Active areas

![Diagram of field oxide and active areas]

- **Fox**: field oxide
n⁺ = n_{diff}

P⁺ = p_{diff}

4 Drawn and Effective Values

d_{po}: minimum extension of poly beyond Active

Left = L - 2L₀

W_eff = W - 0.5W

\[
\frac{W_{eff}}{Left} \quad \text{ratio}
\]
5) Active Contacts

\[ \text{d}_{ac,v} = \text{vertical size of the contact} \]
\[ \text{d}_{ac,h} = \text{horizontal size of the contact} \]

A square contact:
\[ \text{d}_{ac,v} = \text{d}_{ac,h} = \text{d}_{ac} \]

6) Metal 1

7) Vias and Higher level Metals

Layout depends on what mask process to use:
For example, bulk CMOS tech

Latch-up.
two rules to prevent. 
- include a n-well if PFET \rightarrow Vdd
- include a p-sub contact if NFET \rightarrow Gnd

2. Cell - basic building block

\[
\text{cell-based}
\]

we only care ports!!

3. Design Hierarchies (we talked about it before)

- cells used to build more complex units
- the entire chip is built based on hierarchy.

flattened, we bring the cell transistors out!!

\[
\text{layout/polygon}
\]

\[
\text{library}
\]

When flattened, we look at transistors
(no cell boundaries)
like substitution!!
cell-based concept brings influence to litho, spice

1. litho:
   - rule-based (pattern match)
   - model-based (litho-simulation, correct mask based on simulation results)

   like using spice to do design.

   ![Diagram showing Edge map and OPC conversion](image)

   Current progress:
   - litho-aware (2005, June)
   - not litho-aware

   model-based tools
   - SPLAT, SAMPLE (Berkeley Neurath's group)
   - proliith, System C (industry)

   input: layout, light λ, focus rate, etc.
   output: defocus rate, edge map.

   Need to be litho-aware cells!!

2. SPICE:

   flatten netlist runs SPICE
   may have too many transistors
How does Sp ice work?

@ differential equation solver
@ model computation

\[ (G_{d}, C_{c}, \ldots, I_{d}) = \text{MOS}(V_{ds}, V_{gs}, V_{gs}, \text{W}, \text{L}, \text{A}, \text{P}_{d}) \]

Everything looks like RC(L) circuit.

(Transistor, R, C, are not constant)

To speed up Sp ice.

@ Event-driven
@ cell-based characterization

\[ \frac{a}{b} \]

\[ \text{delay} \ T_{ac}, T_{pc} \]

with different slew rate.

Prime-Time (synopsys).

Characterize cell by Sp ice.