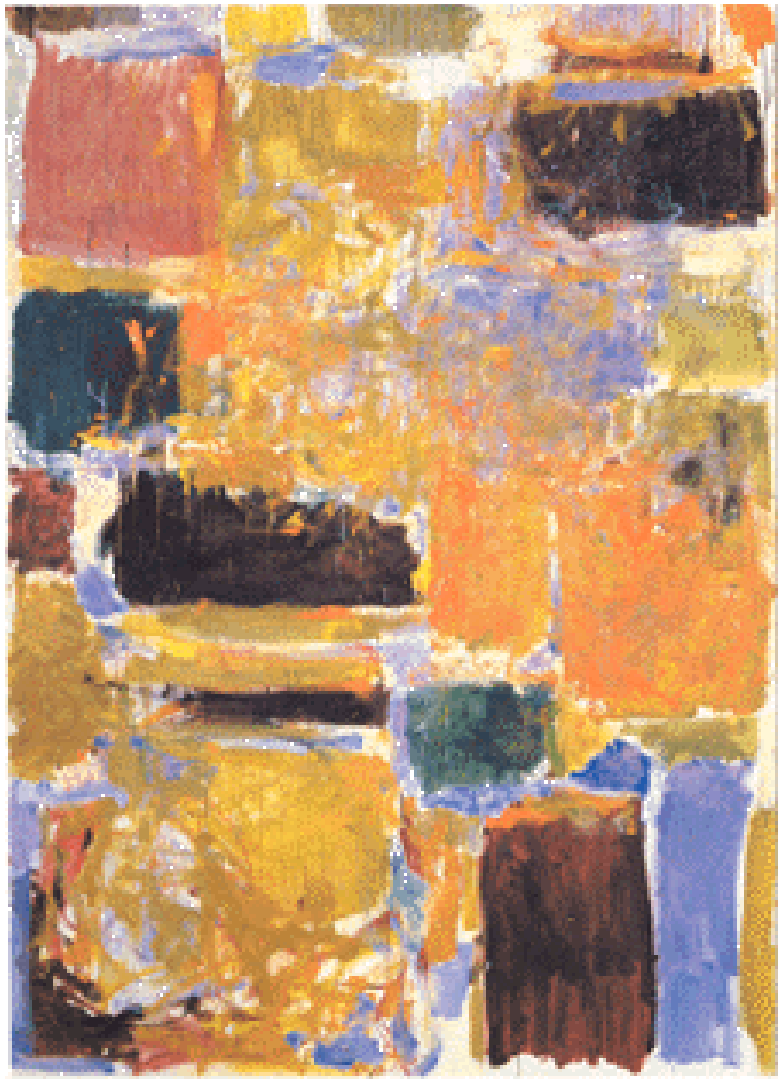
Digital Integrated Circuits
A Design Perspective

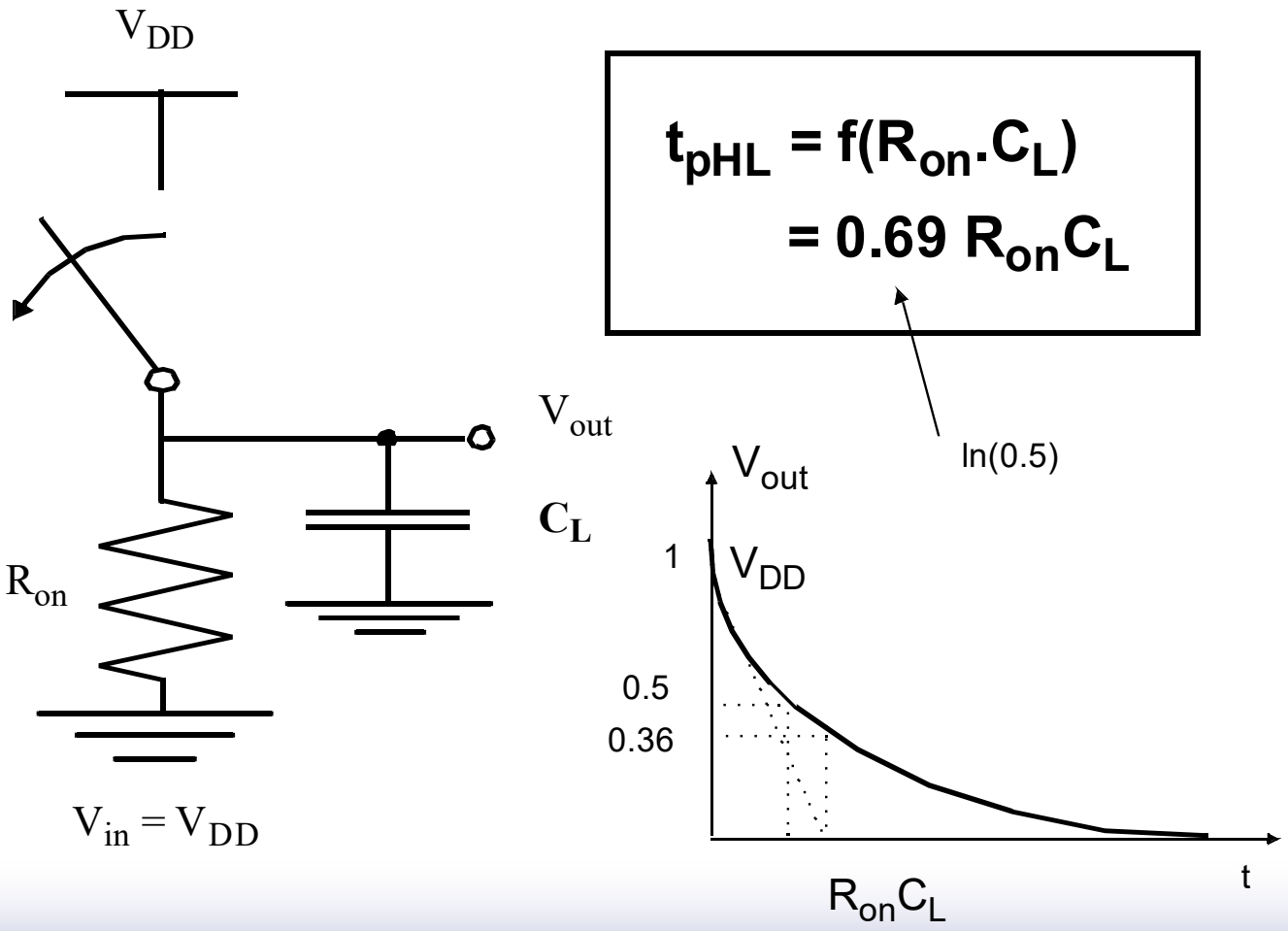
Jan M. Rabaey
Anantha Chandrakasan
Borivoje Nikolic

The Inverter

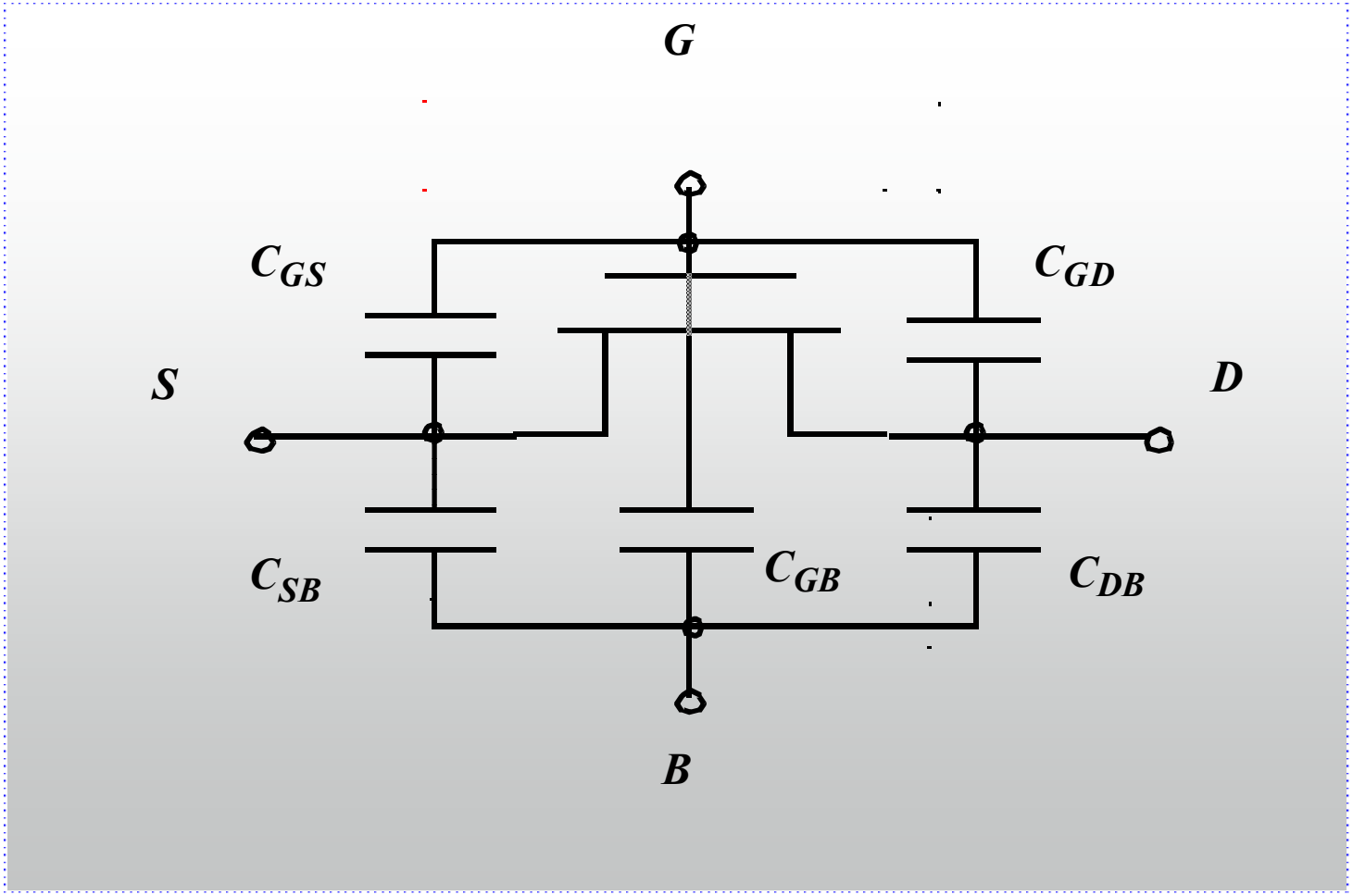


Propagation Delay

CMOS Inverter Propagation Delay

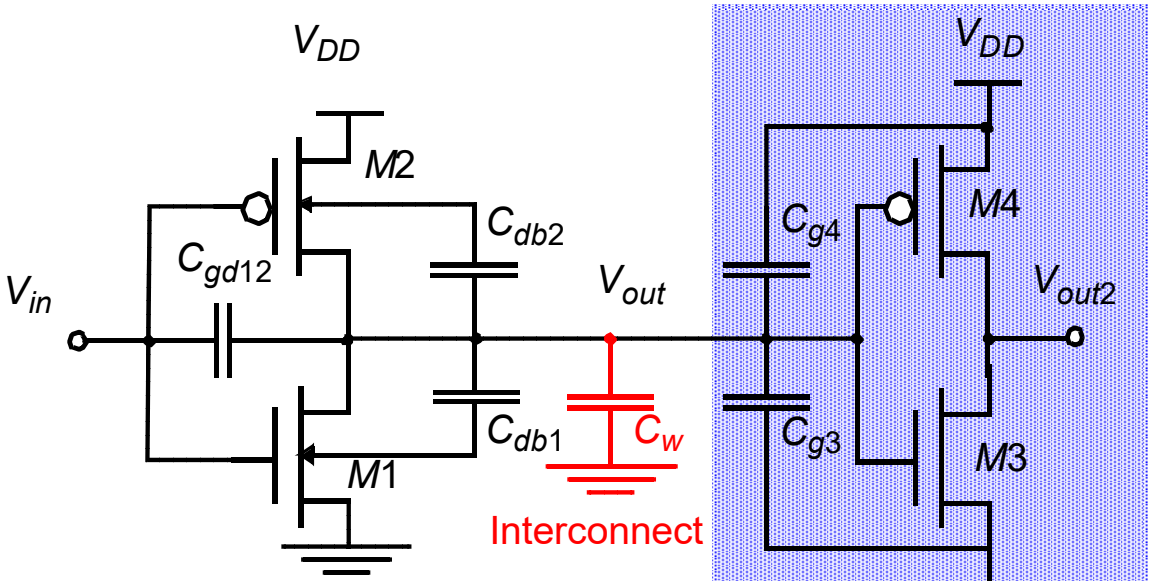


MOS transistor model for simulation

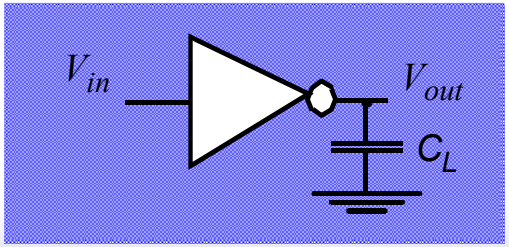


Computing the Capacitances

Consider each capacitor individually is almost impossible for manual analysis. What capacitors count in C_L?



Simplified Model



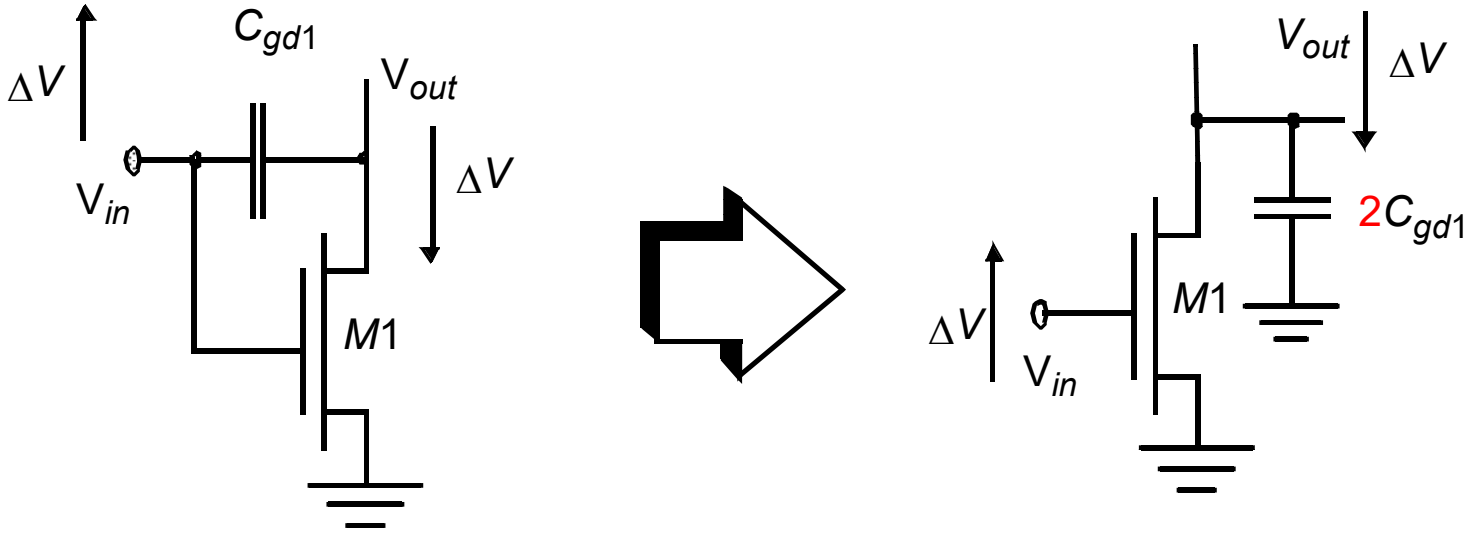
Computing the Capacitances

- NMOS and PMOS transistor are either in cutoff or saturation mode during at least the first half (50%) of the output transient.

- So, the only contributions to C_{gd} are the overlap capacitance, since channel capacitance occurs between either Gate-Body for transistors in cutoff region or Gate-Source for transistors in saturation region.

The Miller Effect

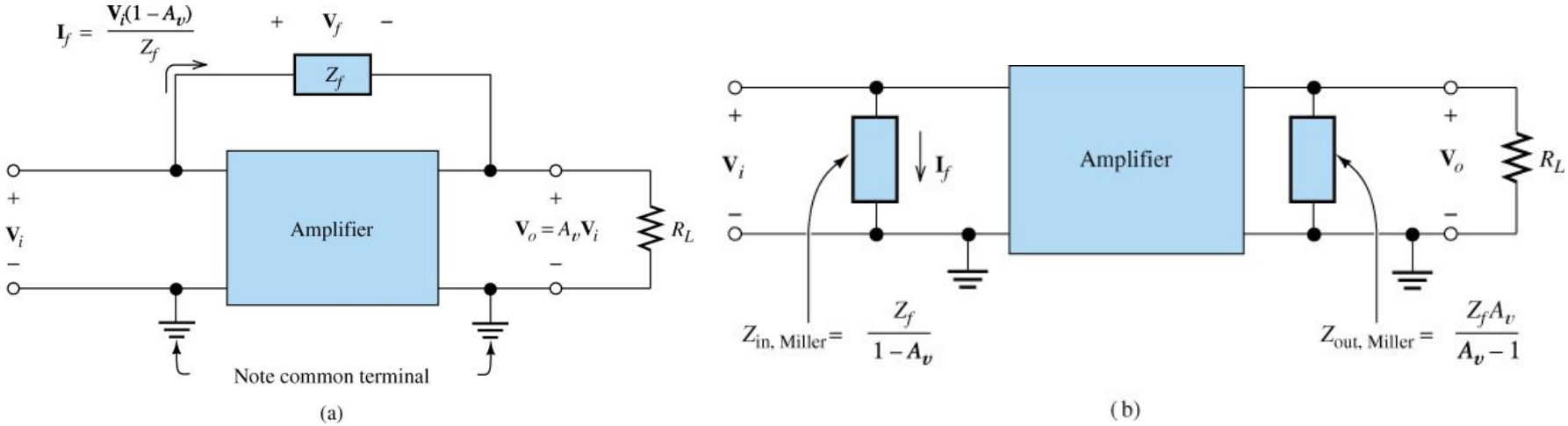
The lumped capacitor model requires the floating C_{gd1} capacitor be replaced by a capacitor to GND using Miller effect.



"A capacitor experiencing identical but opposite voltage swings at both its terminals can be replaced by a capacitor to ground, whose value is two times the original value."

The Miller Effect

- Consider the situation that an impedance is connected between *input* and *output* of an amplifier



- The same current flows from (out) the top input terminal if an impedance $Z_{in,Miller}$ is connected across the input terminals
- The same current flows to (in) the top output terminal if an impedance $Z_{out,Miller}$ is connected across the output terminal
- This is know as Miller Effect
- Two important notes to apply Miller Effect:
  - ✓ There should be a common terminal for input and output
  - ✓ The gain in the Miller Effect is the gain after connecting feedback impedance Z_f

Computing the Capacitances

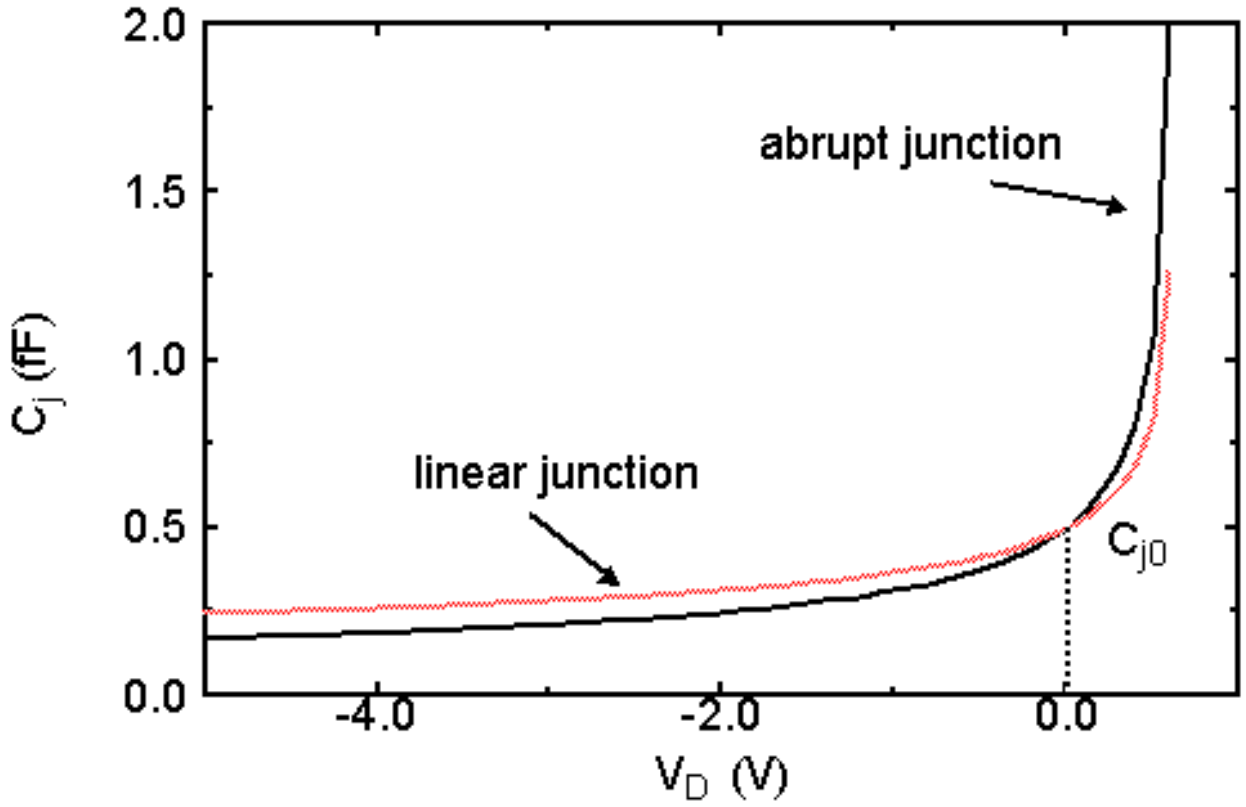
Capacitor	Expression
C_{gd1}	2 CGD0 W_n
C_{gd2}	2 CGD0 W_p
C_{db1}	K_{eqn} (AD_n CJ + PD_n CJSW)
C_{db2}	K_{eqp} (AD_p CJ + PD_p CJSW)
C_{g3}	C_{ox} W_n L_n
C_{g4}	C_{ox} W_p L_p
C_w	From Extraction
C_L	Σ

The capacitance between drain and bulk, C_{db1} and C_{db2}, are due to the reverse-biased pn-junction.

Such a capacitor is, unfortunately, quite nonlinear and depends heavily on the applied voltage.

In Chapter 3 we replaced the nonlinear capacitor by a linear one with the same change in charge for the voltage range of interest. A multiplication factor, K_{eq}, is introduced to relate the linearized capacitor to the value of the junction capacitance under zero-bias conditions (usually in the range of 0.6 to 0.9).

Junction Capacitance



$$C_j = \frac{C_{j0}}{(1 - V_D/\phi_0)^m}$$

m = 0.5: abrupt junction
m = 0.33: linear junction

Linearizing the Junction Capacitance

Replace non-linear capacitance by
large-signal equivalent linear capacitance
which displaces equal charge
over voltage swing of interest

$$C_{eq} = \frac{\Delta Q_j}{\Delta V_D} = \frac{Q_j(V_{high}) - Q_j(V_{low})}{V_{high} - V_{low}} = K_{eq} C_{j0}$$

$$K_{eq} = \frac{-\phi_0^m}{(V_{high} - V_{low})(1-m)} [(\phi_0 - V_{high})^{1-m} - (\phi_0 - V_{low})^{1-m}]$$

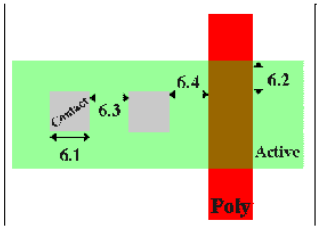


Table 6: Contact to Active

Rule	Description	Lambda
6.1	Exact contact size	2 x 2
6.2	Minimum active overlap	1.5
6.3	Minimum contact spacing	3
6.4	Minimum spacing to gate of transistor	2

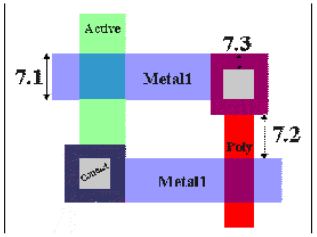


Table 7: Metal1

Rule	Description	Lambda
7.1	Minimum width	3
7.2	Minimum spacing	3
7.3	Minimum overlap of any contact	1

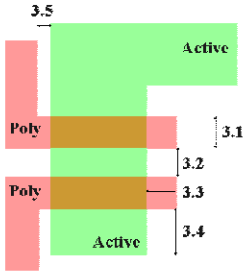
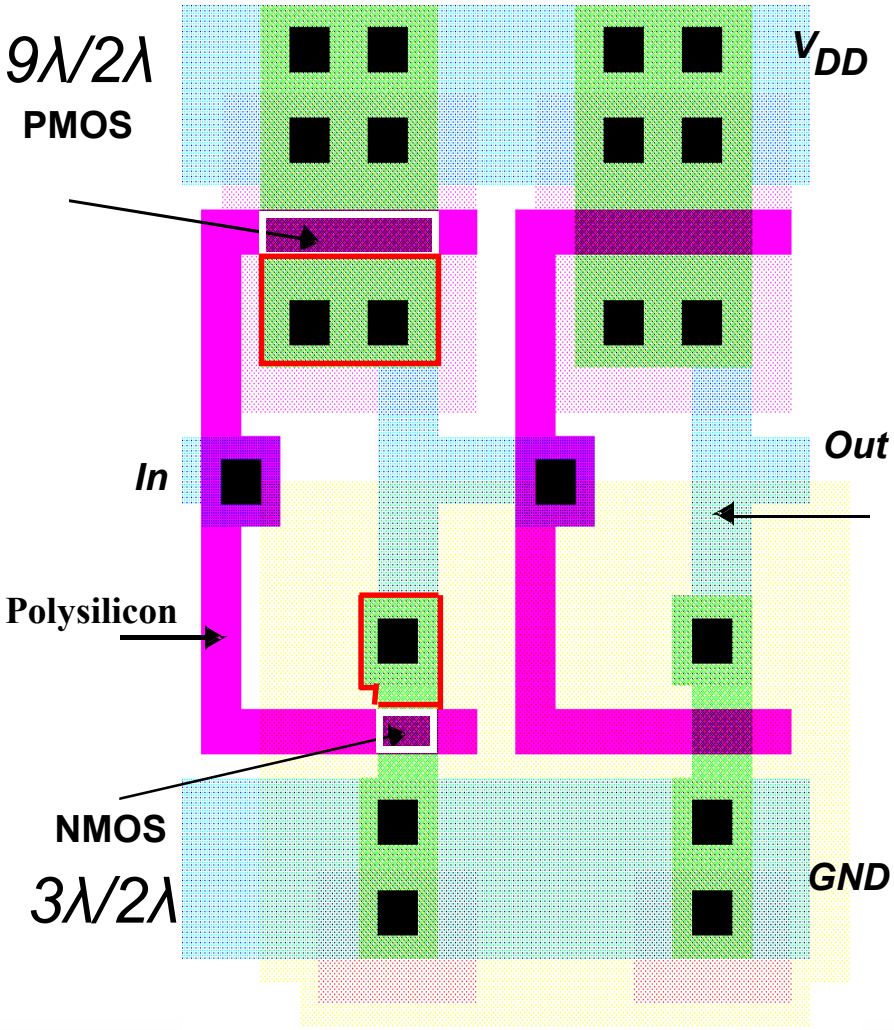
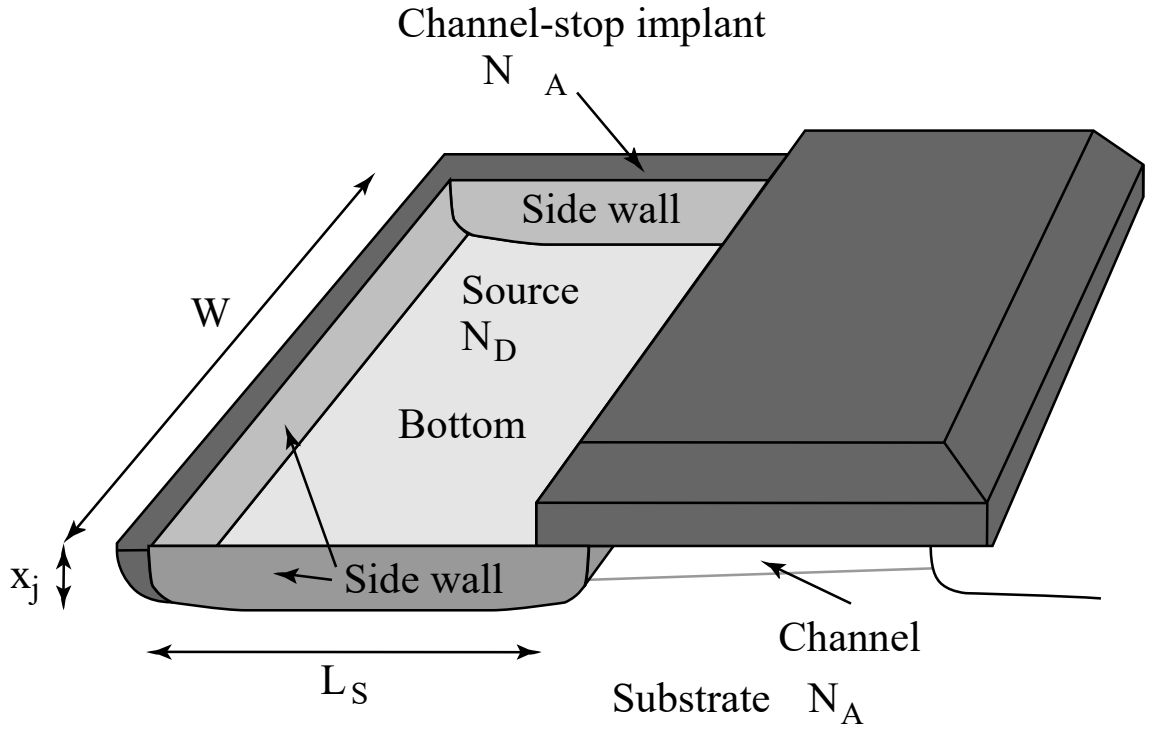


Table 3: Poly

Rule	Description	Lambda
3.1	Minimum width	2
3.2	Minimum spacing	3
3.2.a	Minimum spacing over active	3
3.3	Minimum gate extension of active	2
3.4	Minimum active extension of poly	3
3.5	Minimum field poly to active	1



Junction Capacitance Ls (from Ch. 3)



Junction capacitance per unit length

$$C_{diff} = C_{bottom} + C_{sw} = C_j \times AREA + C_{jsw} \times PERIMETER$$
$$= C_j L_S W + C_{jsw}(2L_S + W)$$

Junction capacitance per unit area

No channel side

Computation of all capacitors

	C_{ox} (fF/μm²)	C_O (fF/μm)	C_j (fF/μm²)	m_j	ϕ_b (V)	C_{jsw} (fF/μm)	m_{jsw}	ϕ_{bsw} (V)
NMOS	6	0.31	2	0.5	0.9	0.28	0.44	0.9
PMOS	6	0.27	1.9	0.48	0.9	0.22	0.32	0.9

Table 5.2 Components of C_L (for high-to-low and low-to-high transitions).

Capacitor	Expression	Value (fF) (H→L)	Value (fF) (L→H)
C_{gd1}	$2\ CGD0_n\ W_n$	0.23	0.23
C_{gd2}	$2\ CGD0_p\ W_p$	0.61	0.61
C_{db1}	$K_{eqn}\ AD_n\ CJ + K_{eqswn}\ PD_n\ CJSW$	0.66	0.90
C_{db2}	$K_{eqp}\ AD_p\ CJ + K_{eqswp}\ PD_p\ CJSW)$	1.5	1.15
C_{g3}	$(CGD0_n+CGSO_n)\ W_n + C_{ox}\ W_n\ L_n$	0.76	0.76
C_{g4}	$(CGD0_p+CGSO_p)\ W_p + C_{ox}\ W_p\ L_p$	2.28	2.28
C_w	From Extraction	0.12	0.12
C_L	Σ	6.1	6.0

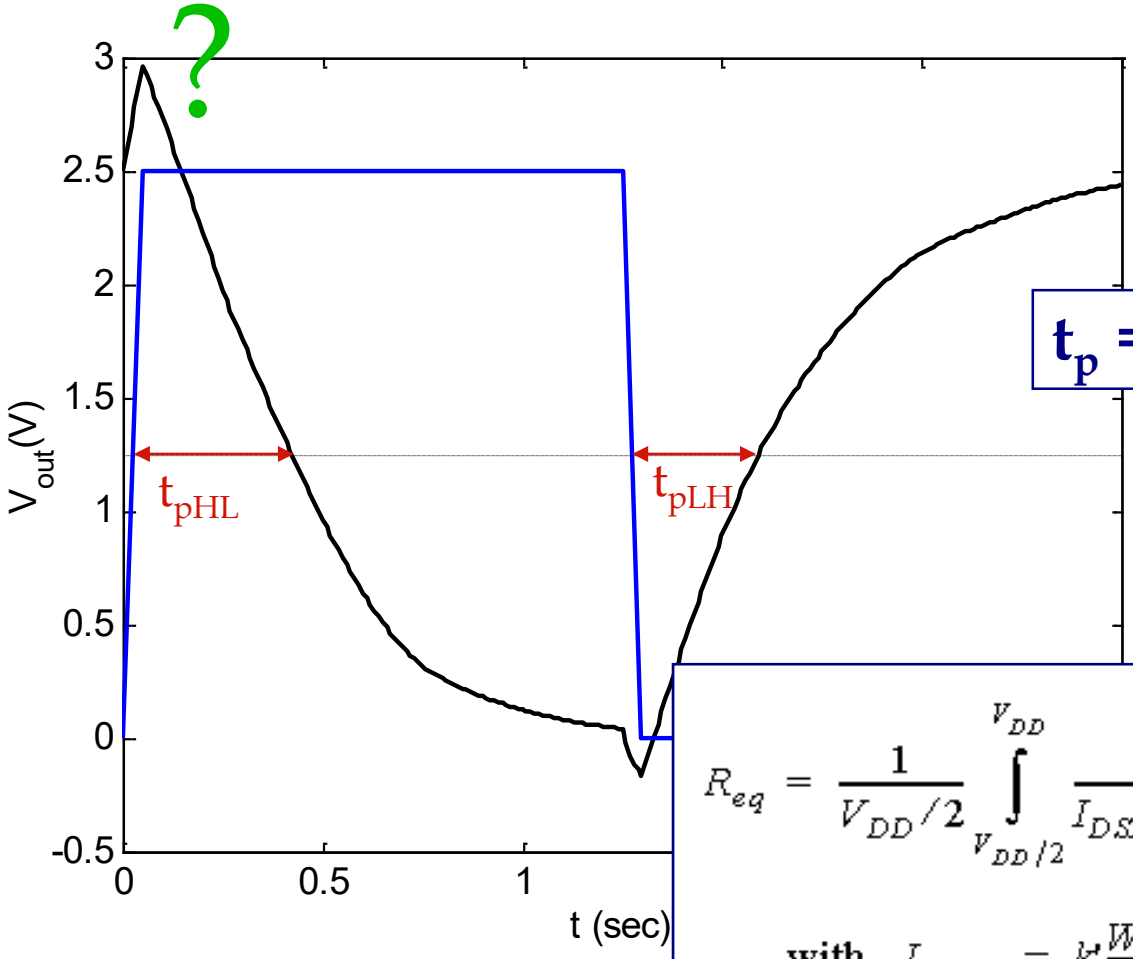
Intrinsic

extrinsic

*C_{db} will be **slightly** different in L-to-H and H-to-L, why? (the pn junction reverse bias voltage range)*

Transient Response

C_{gd} directly couples the steep input change before the circuit can even start to react to the changes at input (potential forward bias the pn junction)



$$t_p = 0.69\, C_L\, (R_{eqn}+R_{eqp})/2$$

$$R_{eq} = \frac{1}{V_{DD}/2}\int_{V_{DD}/2}^{V_{DD}} \frac{V}{I_{DSAT}(1+\lambda V)}dV \approx \frac{3}{4}\frac{V_{DD}}{I_{DSAT}}\left(1-\frac{7}{9}\lambda V_{DD}\right)$$

$$\text{with} \quad I_{DSAT} = k'\frac{W}{L}\left((V_{DD}-V_T)V_{DSAT}-\frac{V_{DSAT}^2}{2}\right)$$

Low-to-High and High-to-Low delay

- It is desired to have identical propagation delays for both rising and falling inputs.
- Equal delay requires equal equivalent on-resistance, thus equal current I_{DAST} (neglecting the channel length modulation)
- This demands almost the same requirements for a V_m at $V_{DD}/2$. Why?

Requirements for equal delay

$$I_{DSAT} = k' \frac{W}{L}\left((V_{DD} - V_T)V_{DSAT} - \frac{V_{DSAT}^2}{2}\right)$$

$$I_{Dn} = k_n'(\frac{W}{L})_n V_{DSATn}\left((V_{DD} - V_{Tn}) - \frac{V_{DSATn}}{2}\right)$$

$$I_{Dp} = k_p'(\frac{W}{L})_p V_{DSATp}\left((V_{DD} - V_{Tp}) - \frac{V_{DSATp}}{2}\right)$$

Assume $V_{DD} >> V_{Tp} + V_{DSATp}/2$

then $$\frac{k_p' V_{DSATp}(W/L)_p}{k_n' V_{DSATn}(W/L)_n} = \frac{k_p V_{DSATp}}{k_n V_{DSATn}} = 1$$

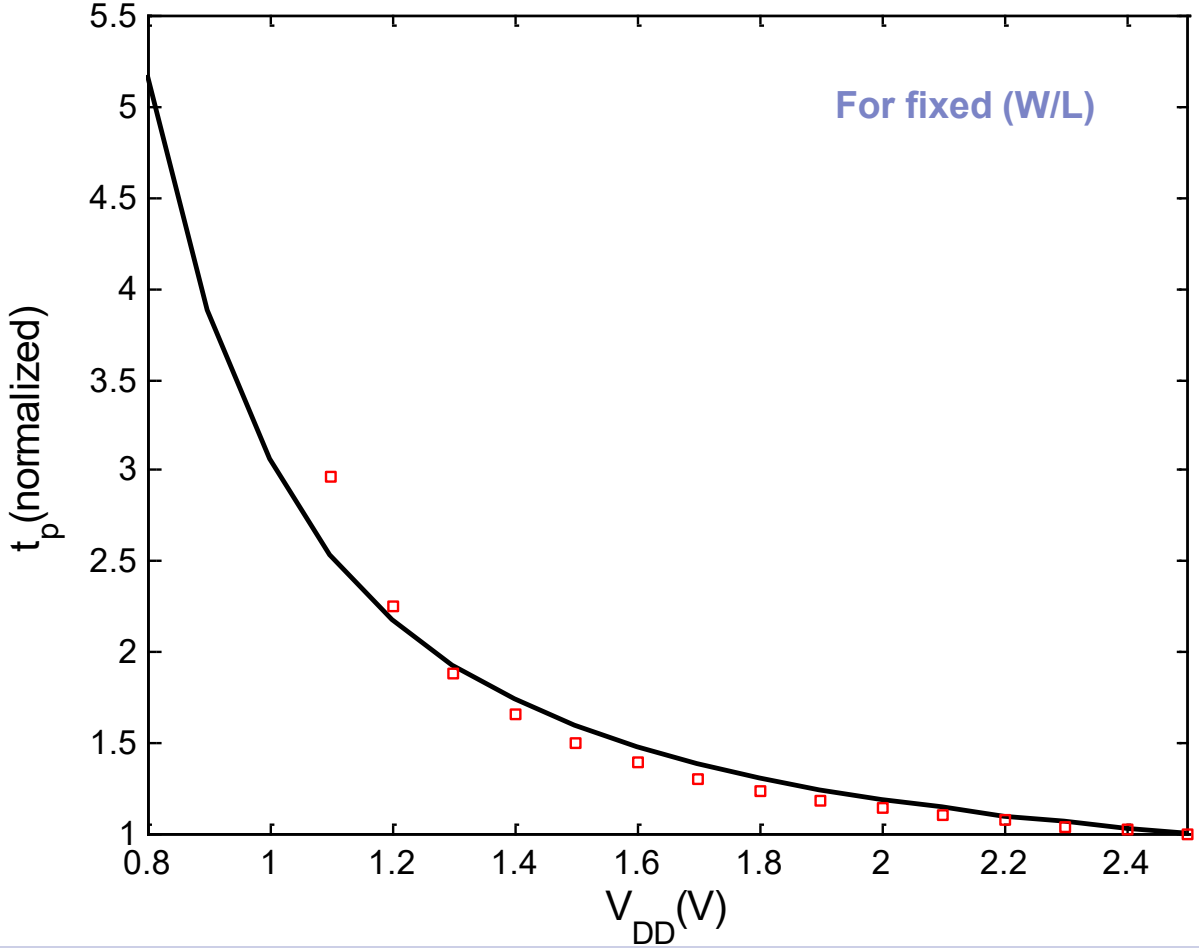
This is exactly the formerly defined parameter r (last lecture)

Design for delay performance

- Keep capacitances small
  - careful layout, e.g. to keep drain diffusion as small as possible
- Increase transistor sizes
  - watch out for self-loading! When intrinsic capacitance starts to dominate the extrinsic ones
- Increase VDD (????)

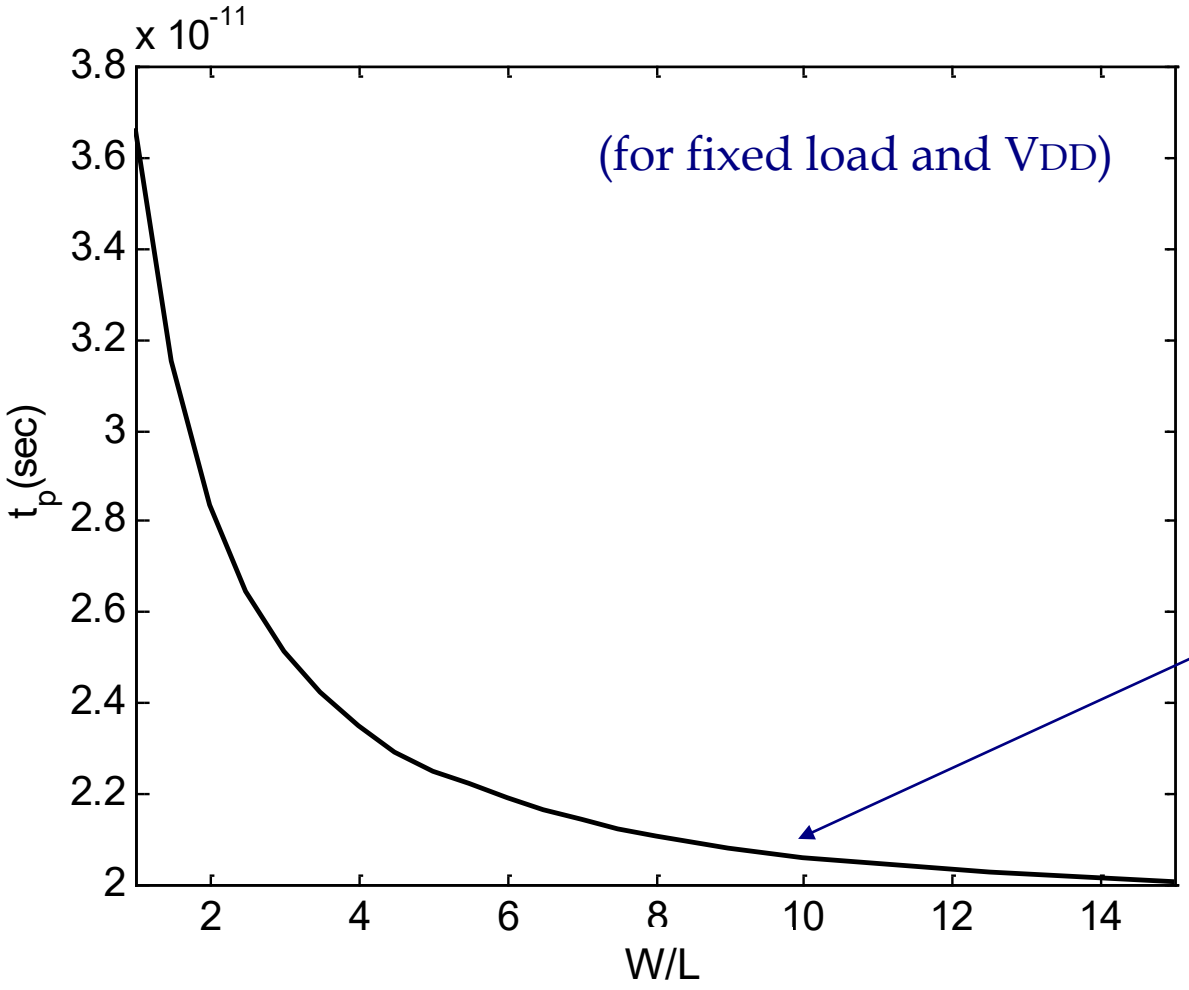
Delay as a function of V_{DD}

$$t_{pHL} = 0.69\frac{3}{4}\frac{C_L V_{DD}}{I_{DSATn}} = 0.52\frac{C_L V_{DD}}{(W/L)_n k'_n V_{DSATn}(V_{DD} - V_{Tn} - V_{DSATn}/2)}$$



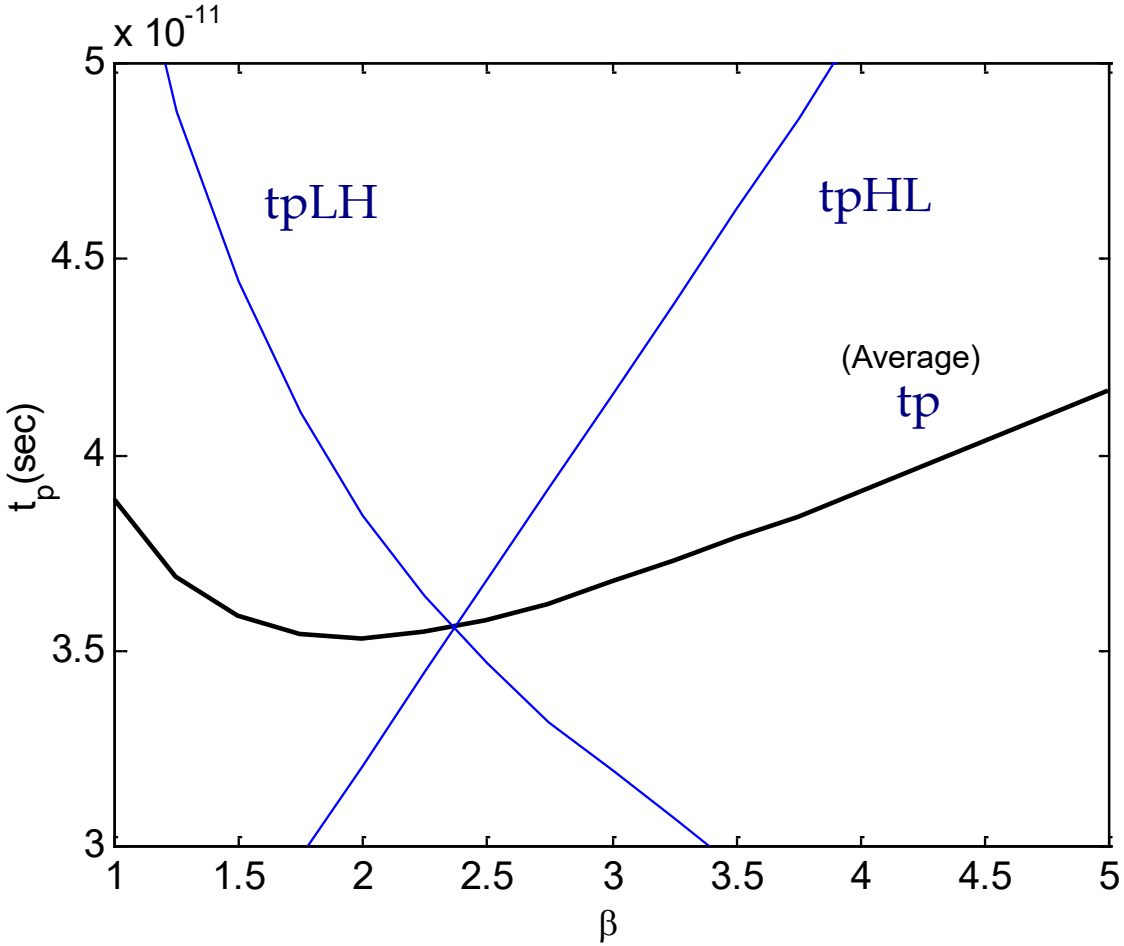
Recall a range of low voltage is able to give even better voltage transfer characteristic

Device Sizing



Self-loading effect:
Intrinsic capacitances dominate

NMOS/PMOS ratio



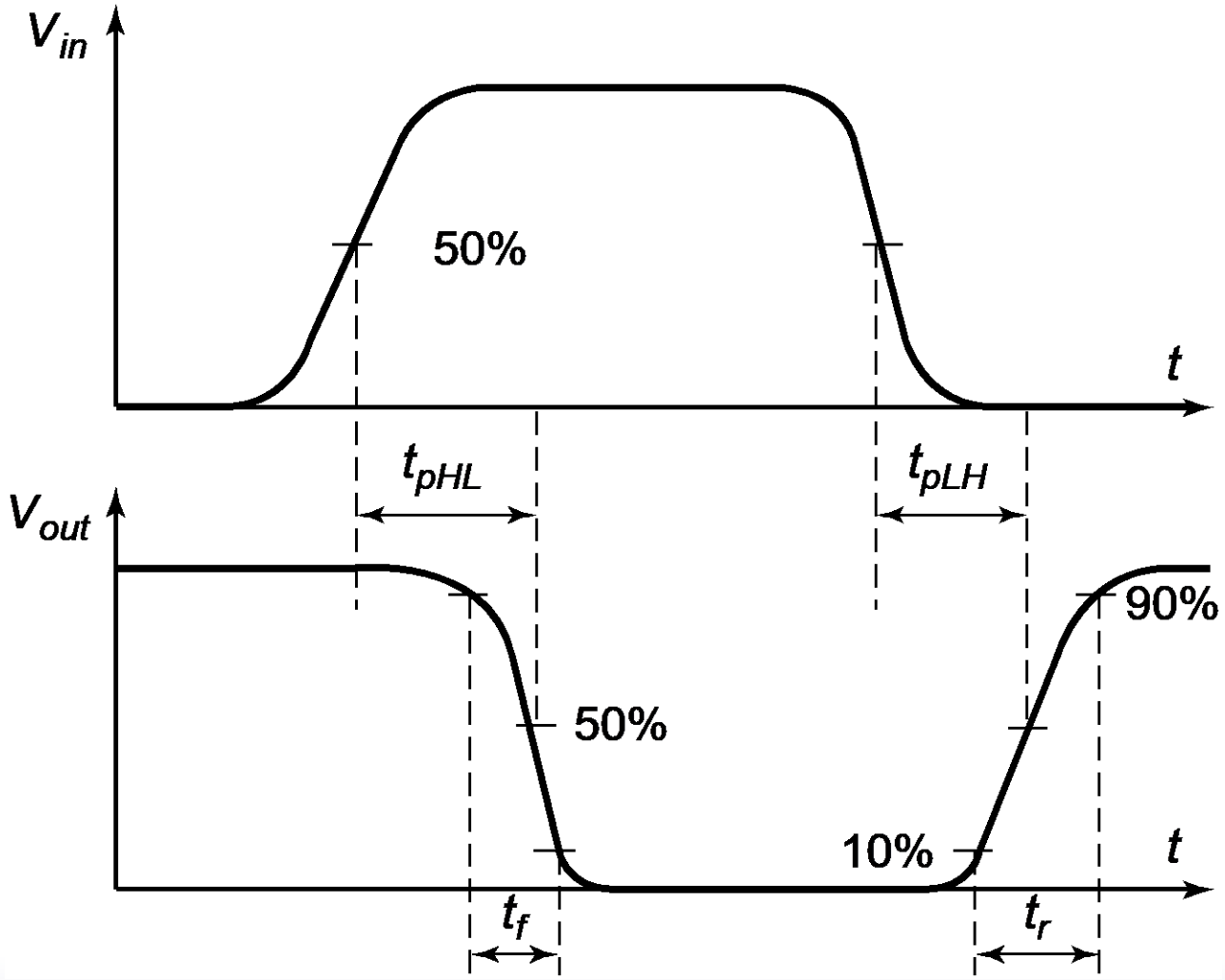
$\beta = W_p/W_n$

$Lp = L_n$

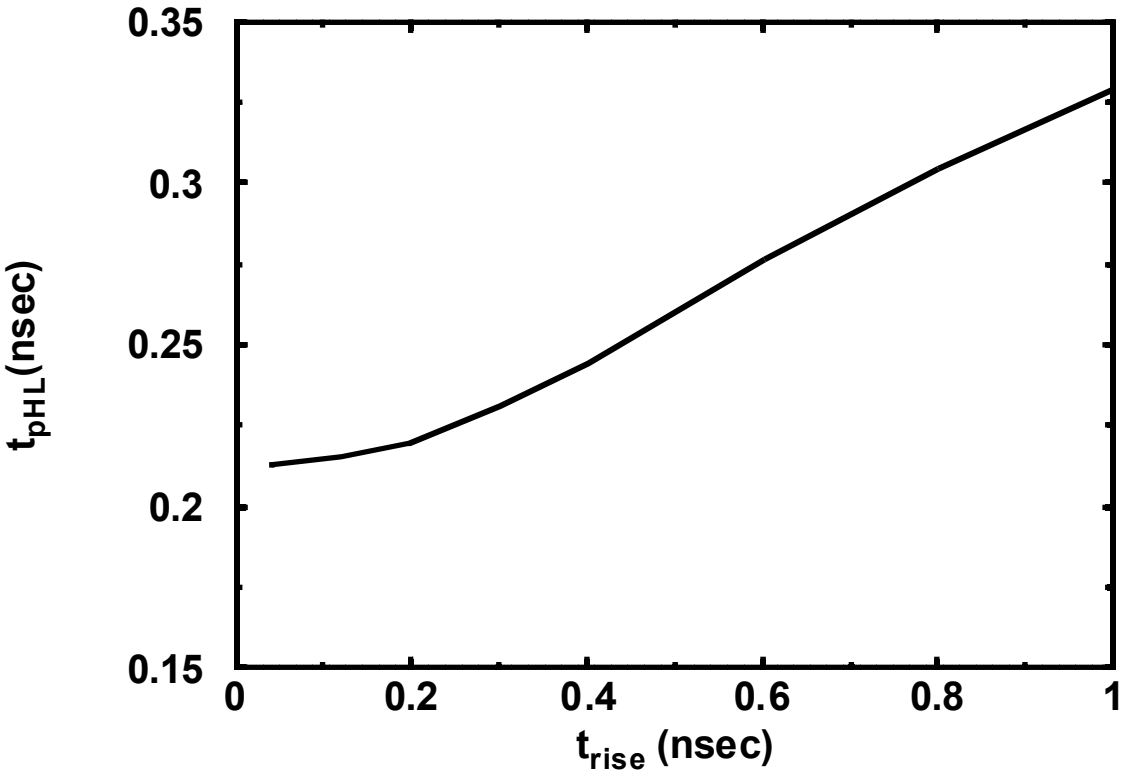
Widening PMOS improves the L-H delay by increasing the charge current, but it also degrades the H-L by giving a larger parasitic capacitance.

Considering average is more meaningful!!

Delay Definitions



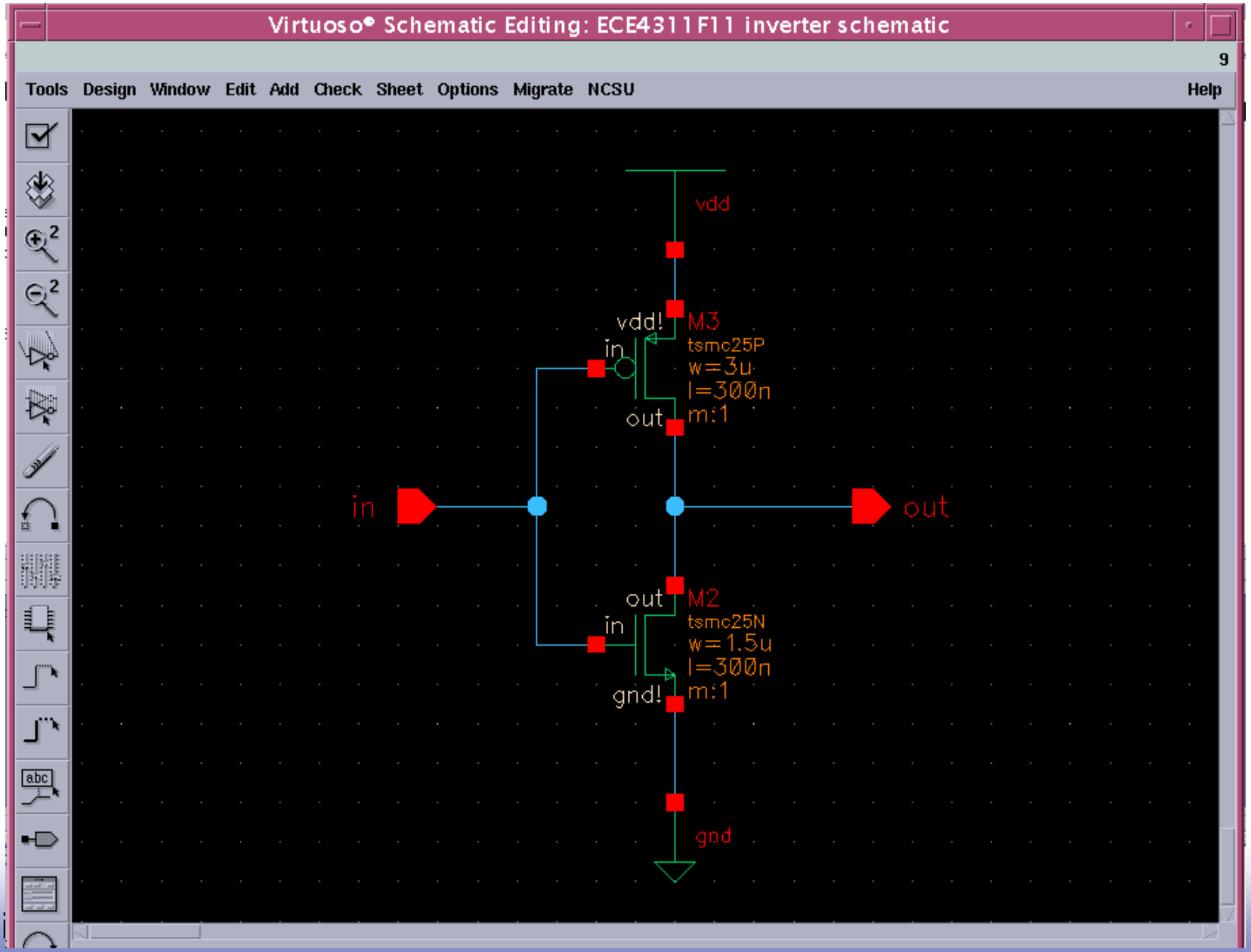
Impact of Rise Time on Delay



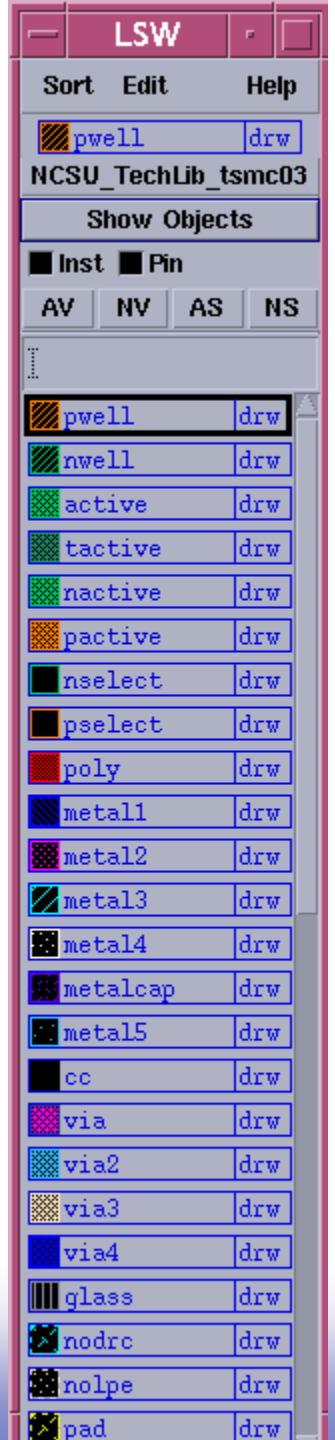
$$t_{pHL} = \sqrt{t_{pHL(step)}^2 + (t_r/2)^2}$$

Custom design process: An inverter design example

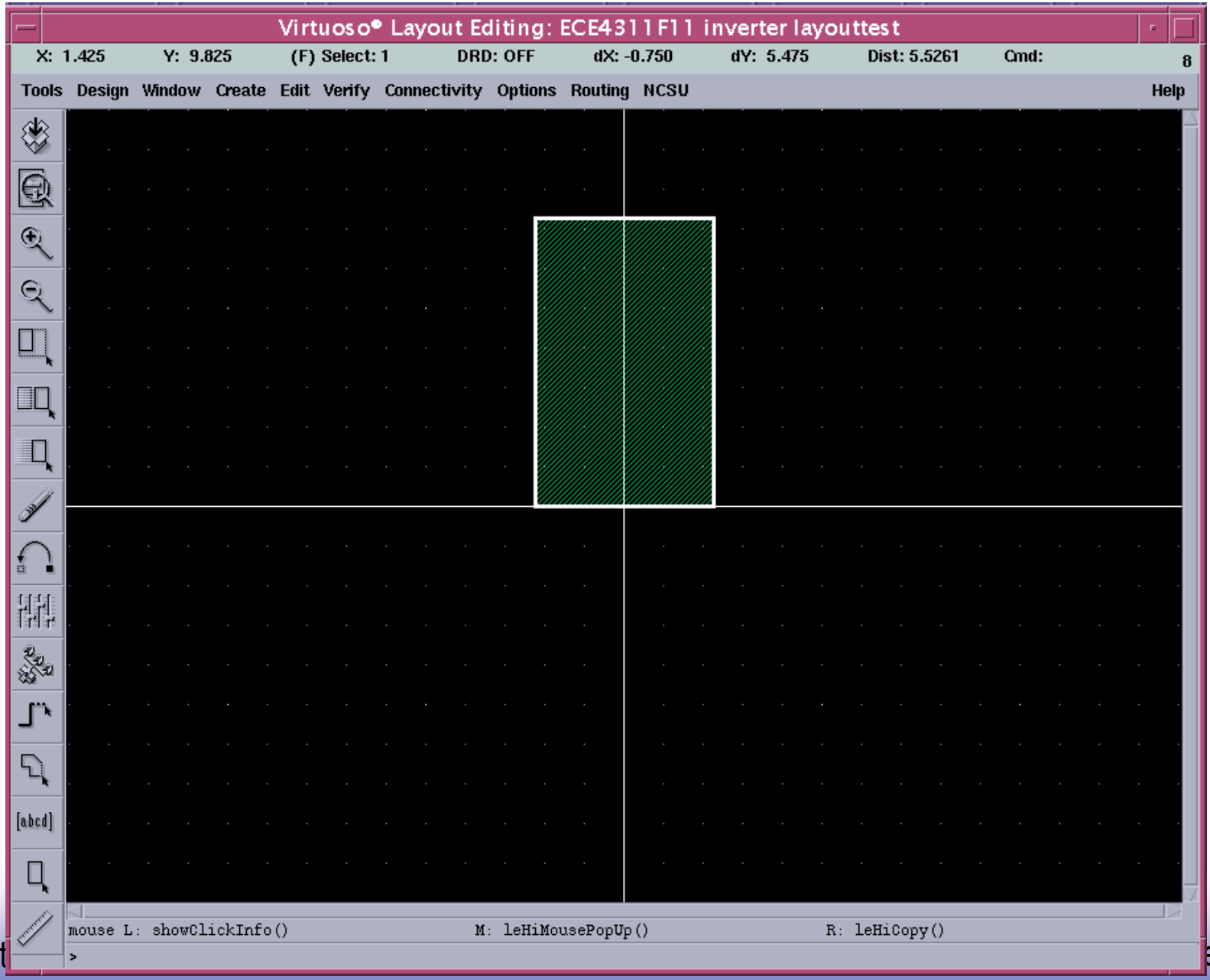
1. Schematic design



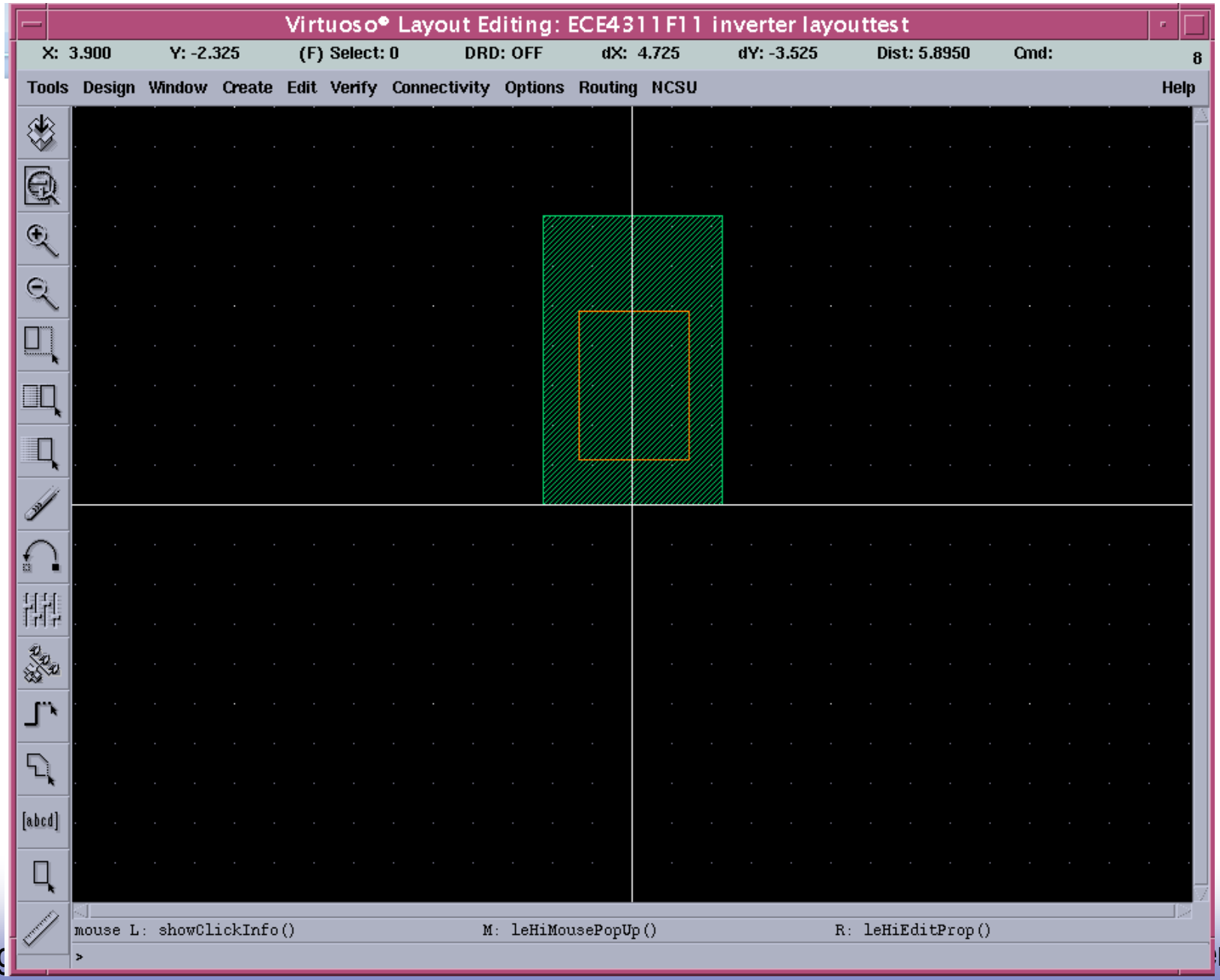
2. Layout design



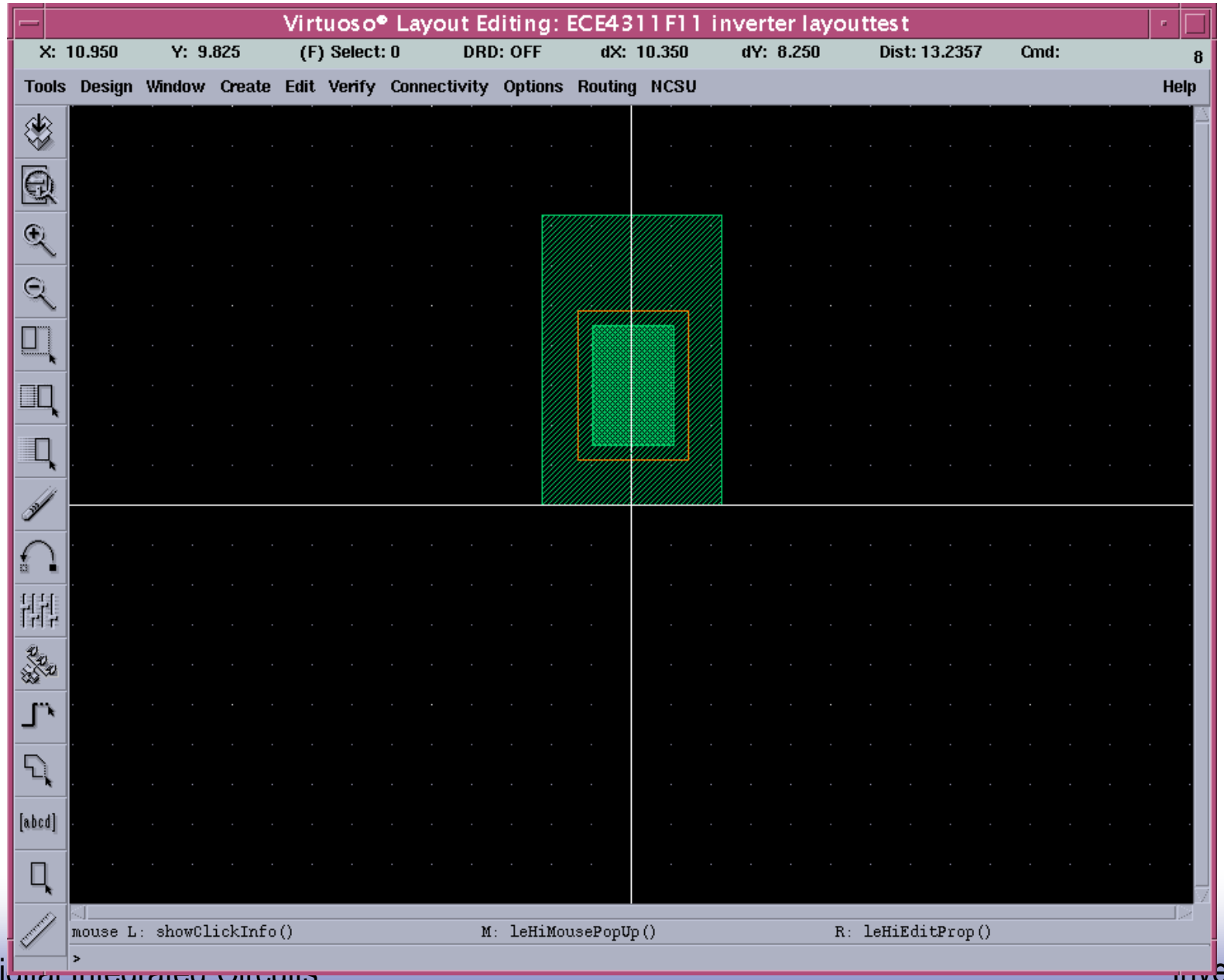
Step 1: define Nwell (for PMOS)



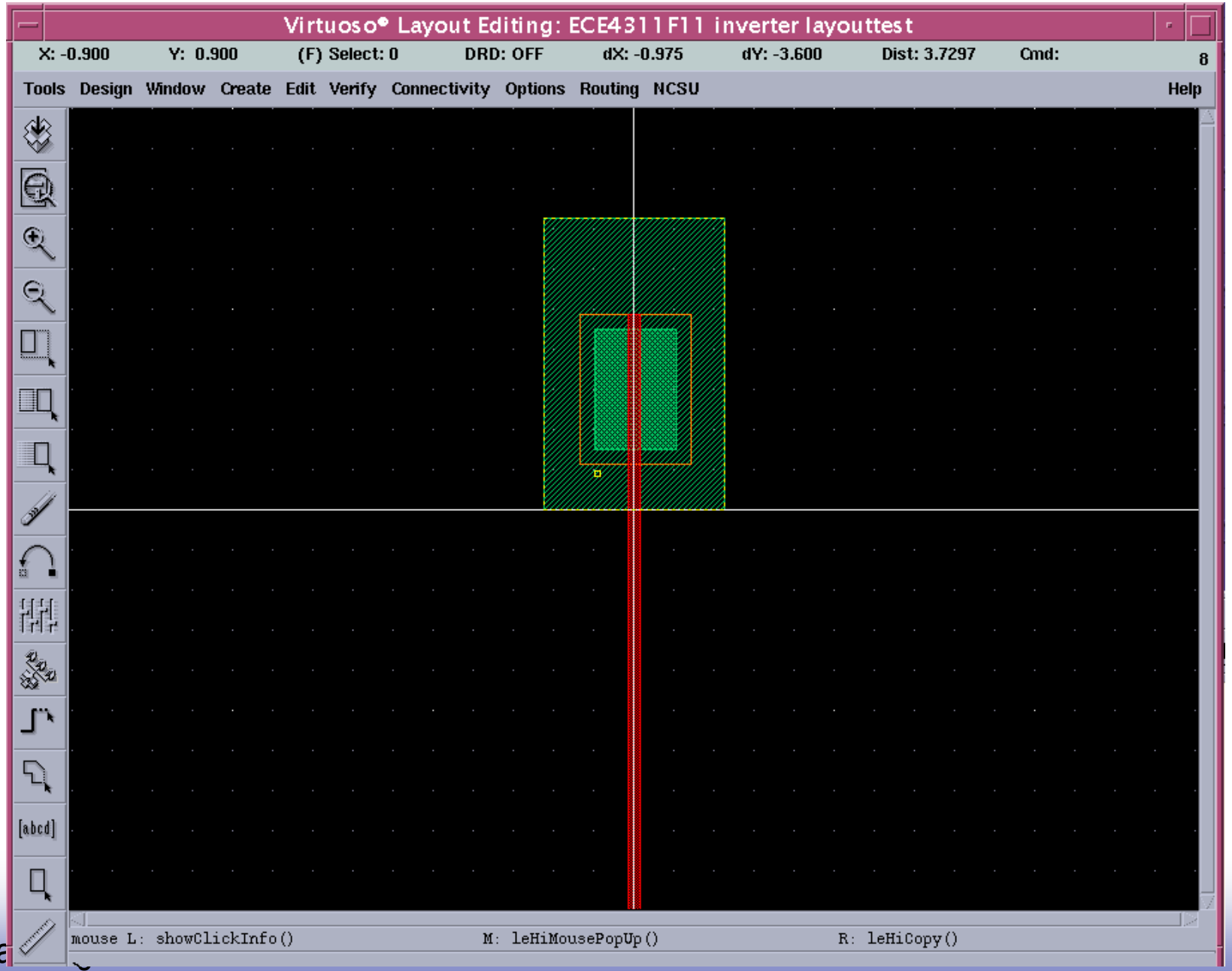
Step 2: define pselect (for PMOS location)



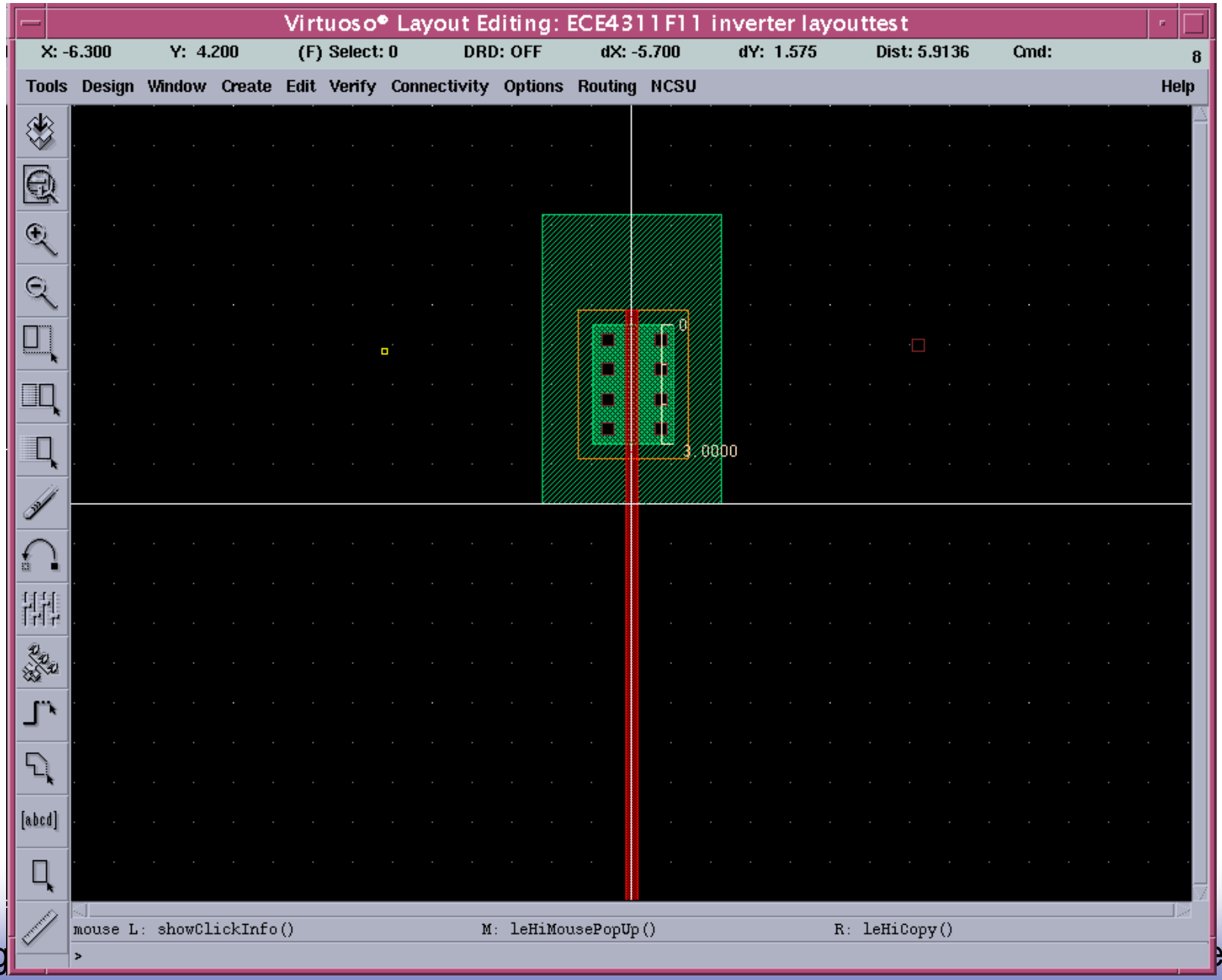
Step 3: define active region (for PMOS)



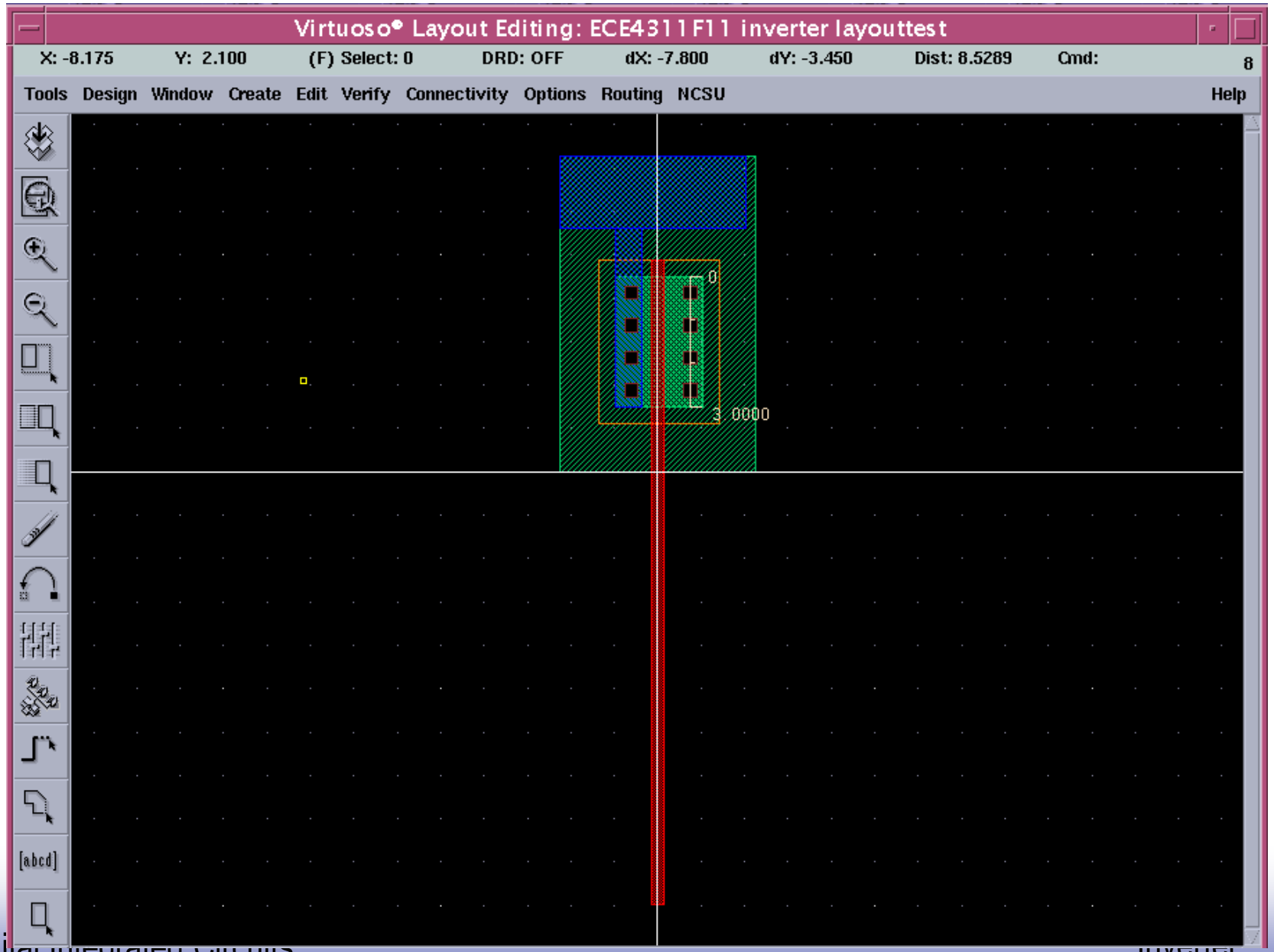
Step 4: define poly (gate for PMOS)



Step 5: define contacts (for PMOS)

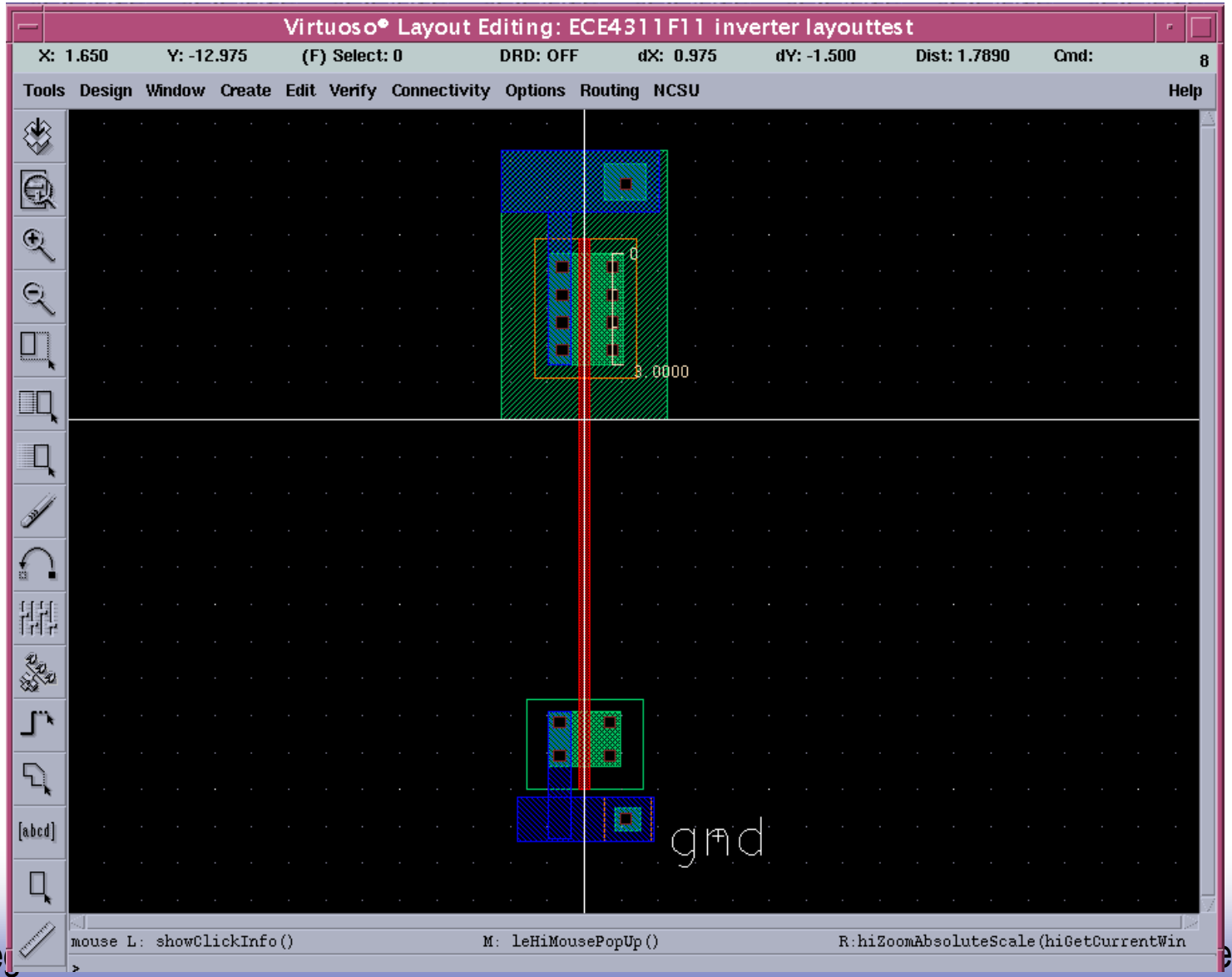


Step 6: define Vdd and connect source (of PMOS) to it

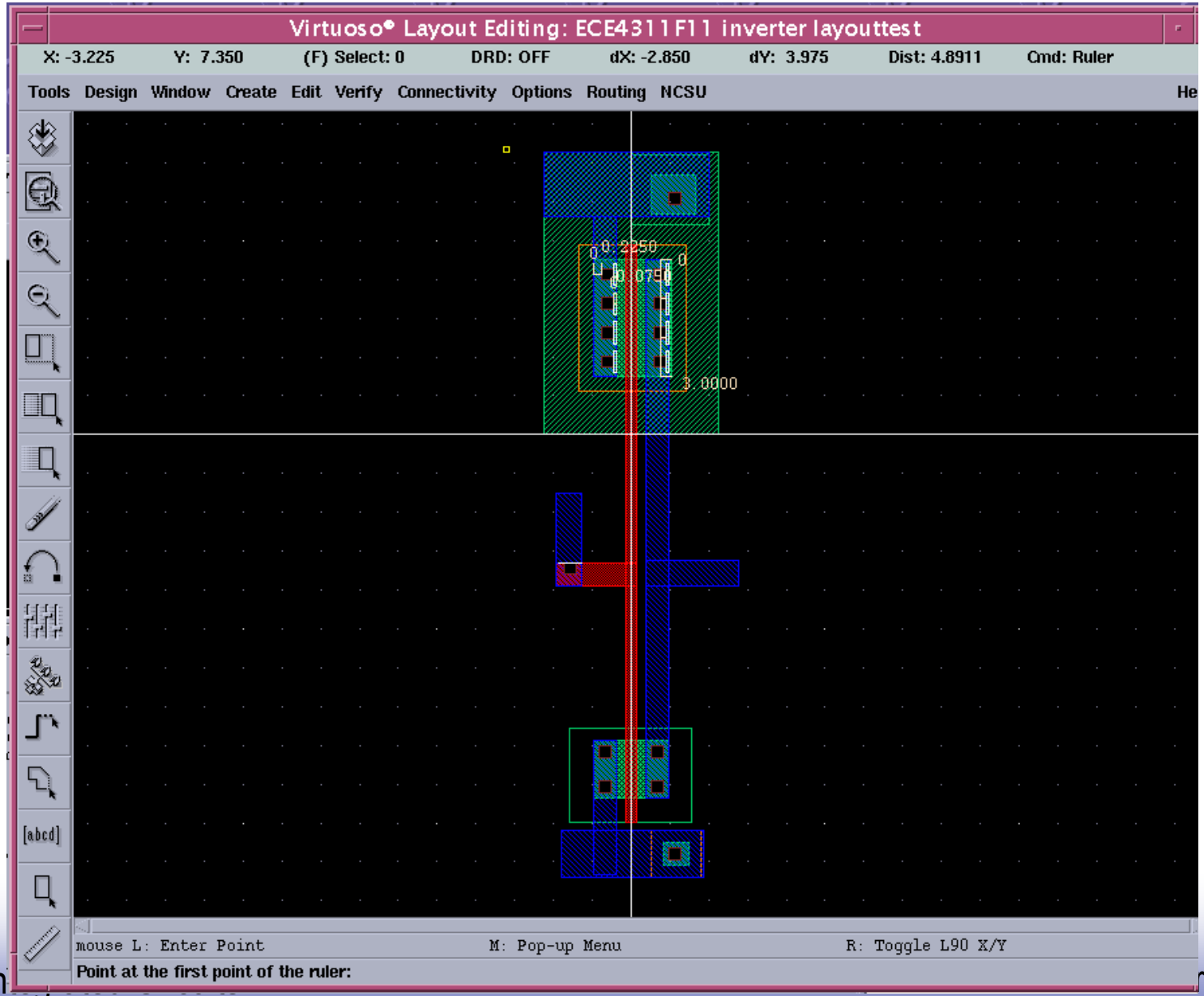


Step 7: make at least one Nwell contact

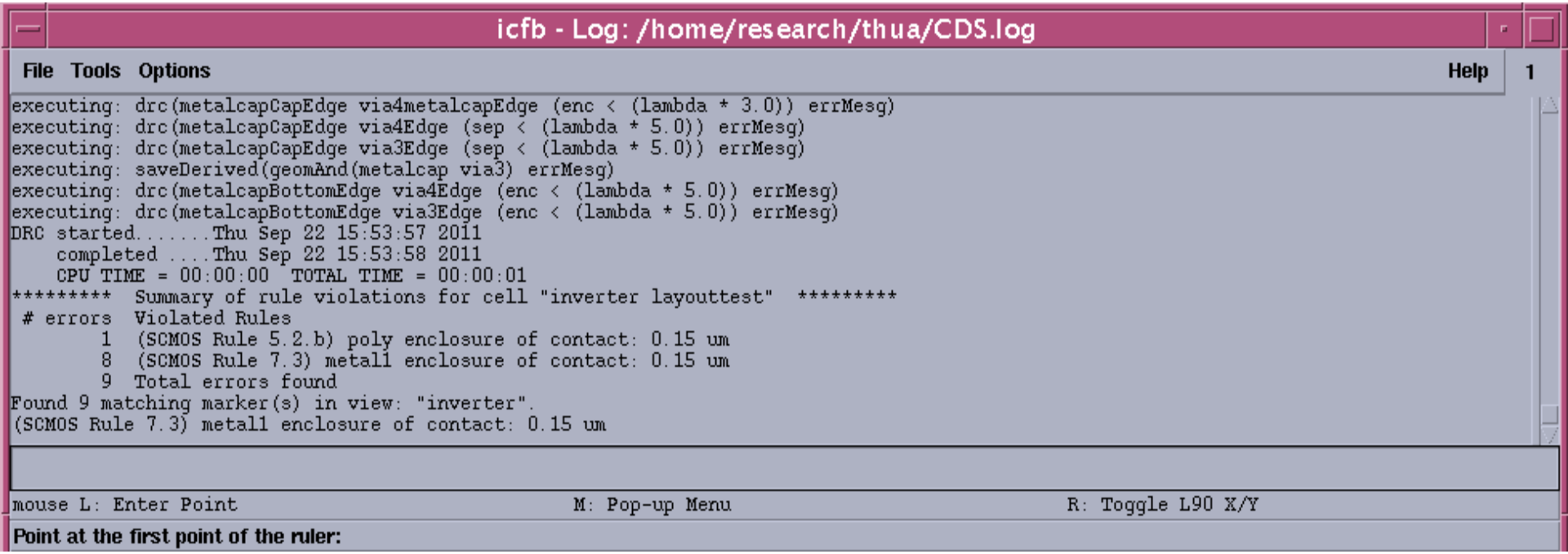
Step 8: create NMOS (repeat similar steps before except you do not need make Nwell)



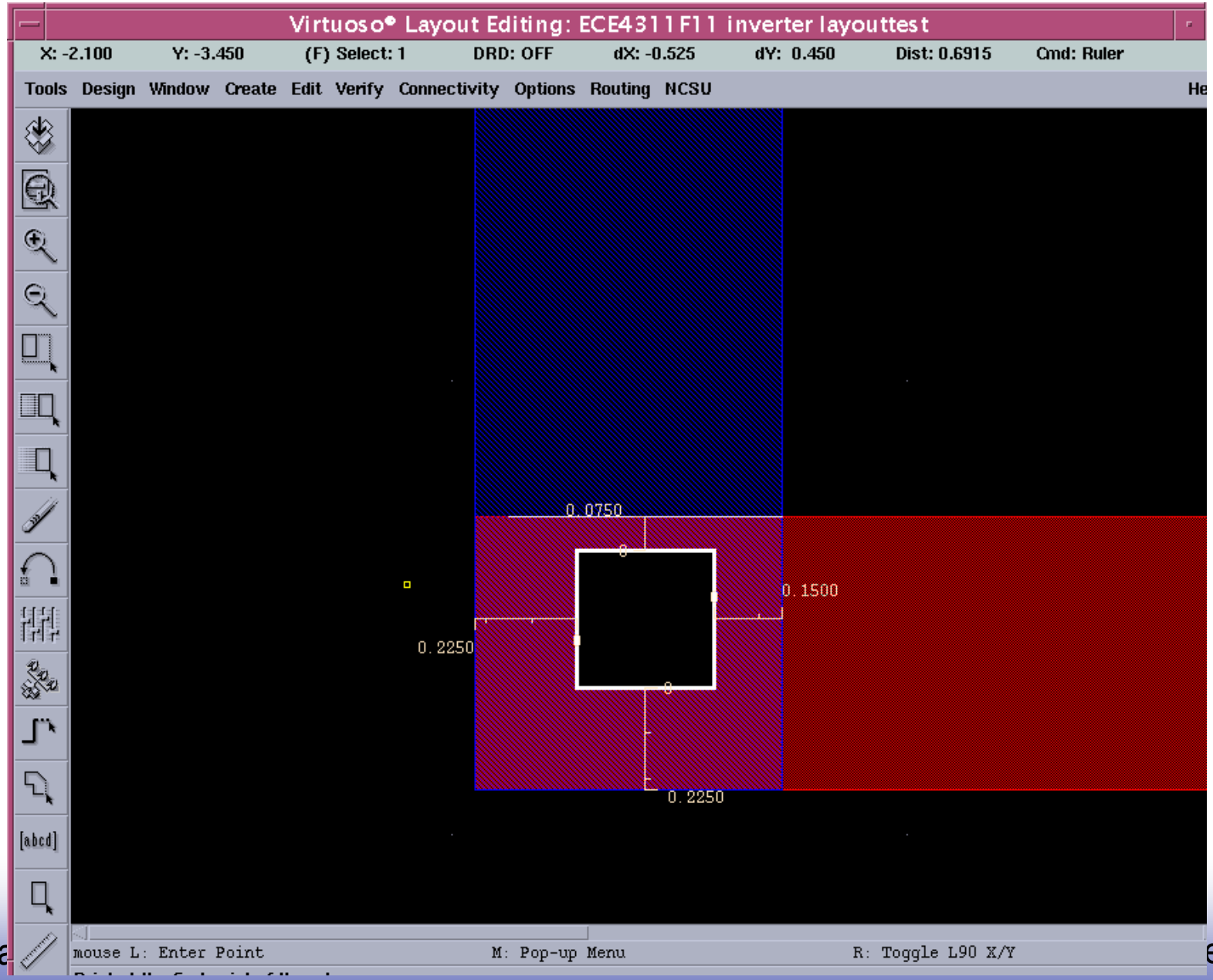
step9: make input and output connections



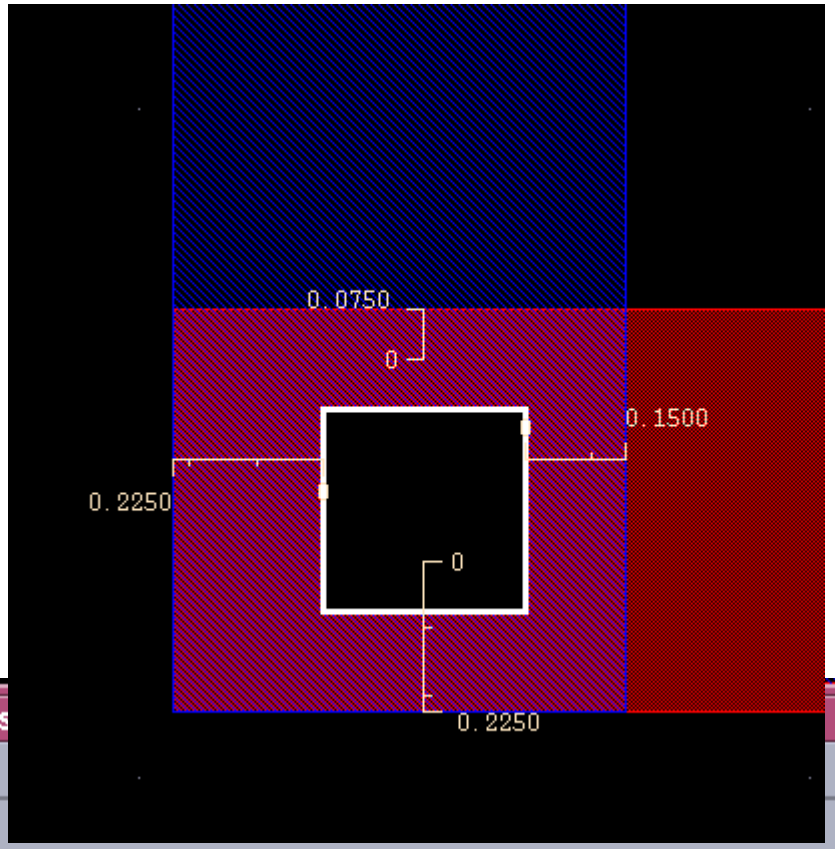
2. DRC (Design Rule Check)



Correct error if there is any



After correction



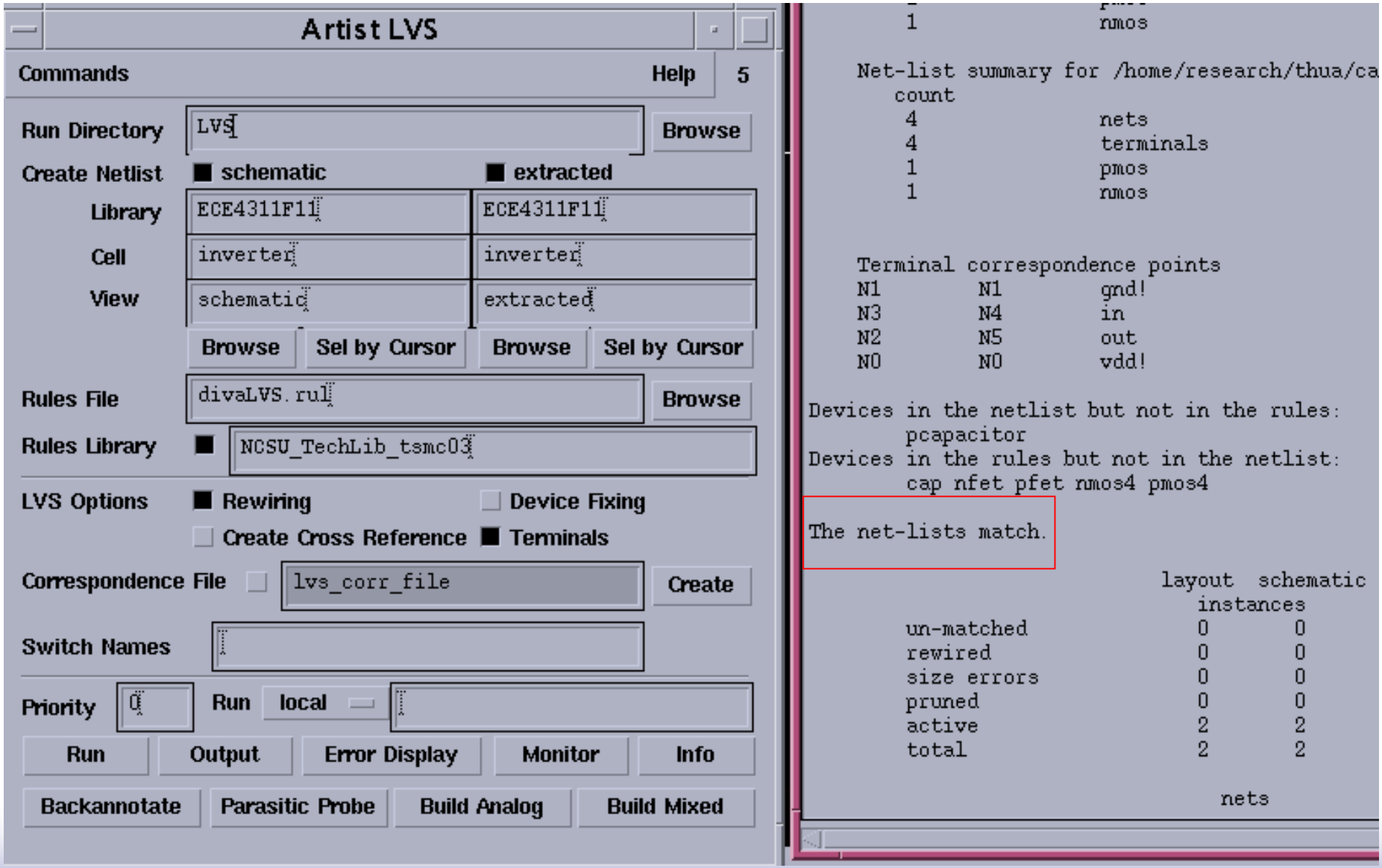
icfb - Log: /home/res

File Tools Options Help

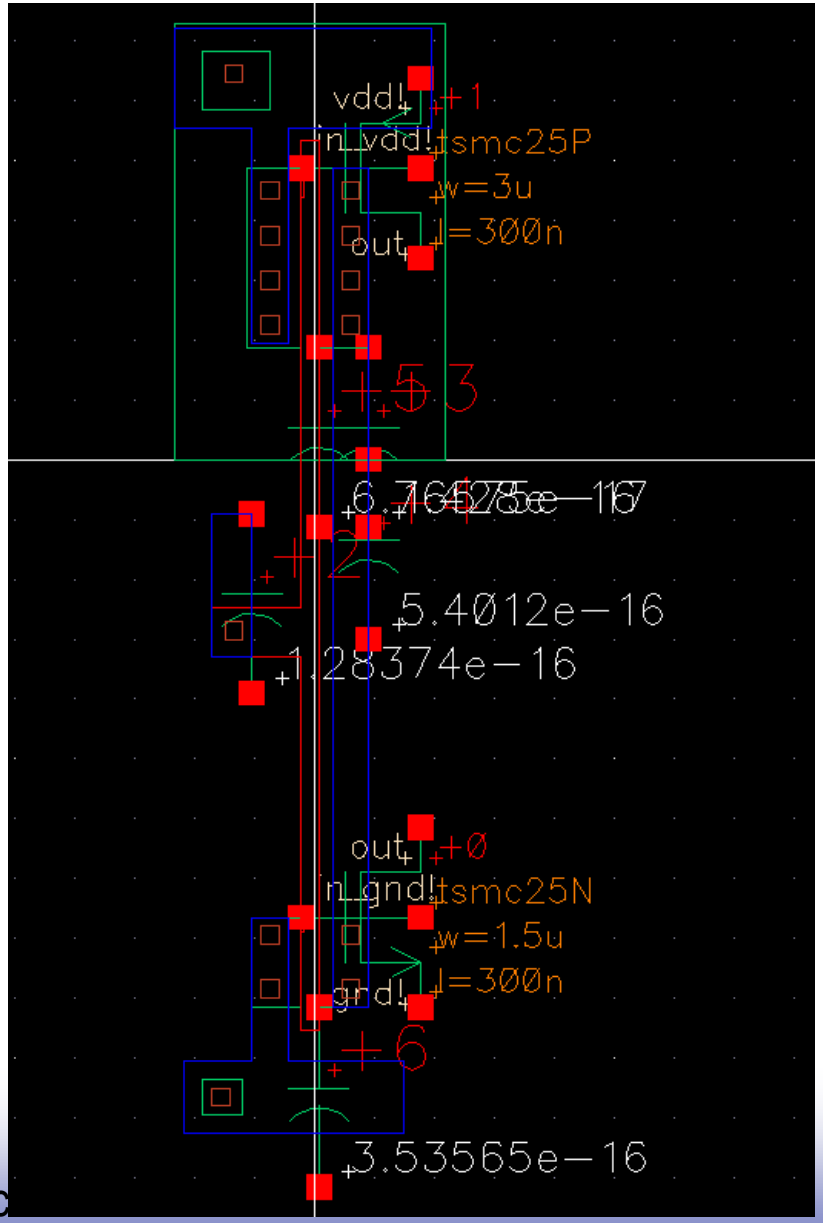
```
executing: drc(metalcapCapEdge (width < (lambda * 50.0)) errMesg)
           drc(metalcapCapEdge (sep < (lambda * 2.0)) errMesg)
           drc(metalcapCapEdge (notch < (lambda * 2.0)) errMesg)
executing: saveDerived(geomAndNot(metalcap metal4) errMesg)
executing: drc(metalcapBottomEdge metalcapCapEdge (enc < (lambda * 5.0)) errMesg)
executing: drc(metalcapCapEdge via4metalcapEdge (enc < (lambda * 3.0)) errMesg)
executing: drc(metalcapCapEdge via4Edge (sep < (lambda * 5.0)) errMesg)
executing: drc(metalcapCapEdge via3Edge (sep < (lambda * 5.0)) errMesg)
executing: saveDerived(geomAnd(metalcap via3) errMesg)
executing: drc(metalcapBottomEdge via4Edge (enc < (lambda * 5.0)) errMesg)
executing: drc(metalcapBottomEdge via3Edge (enc < (lambda * 5.0)) errMesg)
DRC started.......Thu Sep 22 16:03:54 2011
    completed ....Thu Sep 22 16:03:55 2011
    CPU TIME = 00:00:00  TOTAL TIME = 00:00:01
*********   Summary of rule violations for cell "inverter layouttest"   *********
   Total errors found: 0
```

mouse L: showClickInfo() M: leHiMousePopUp() R: setDRCForm()

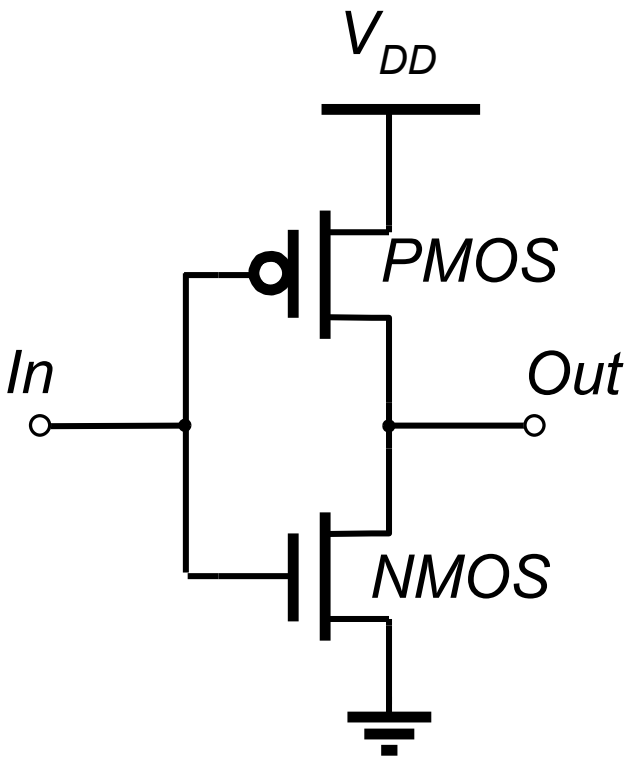
3. LVS (Layout versus Schematic)

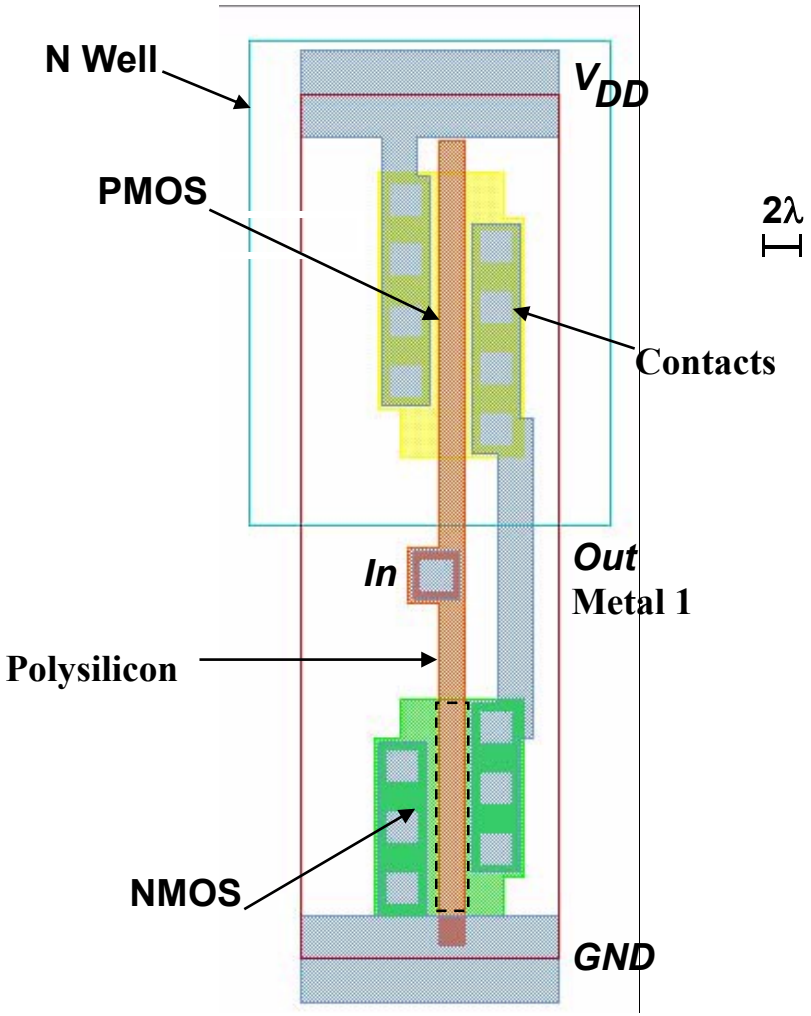


4. Extract Layout parasitics and post-layout simulation



CMOS Inverter



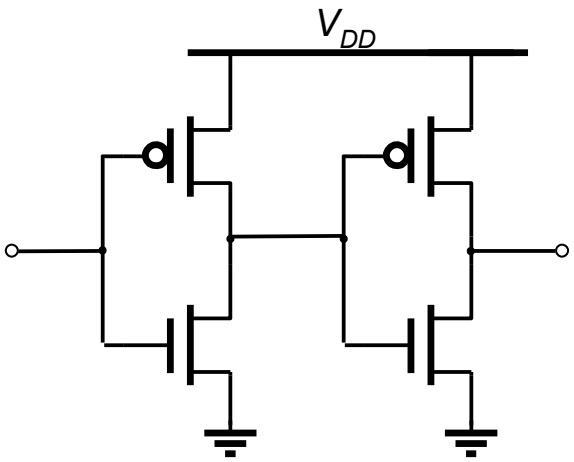


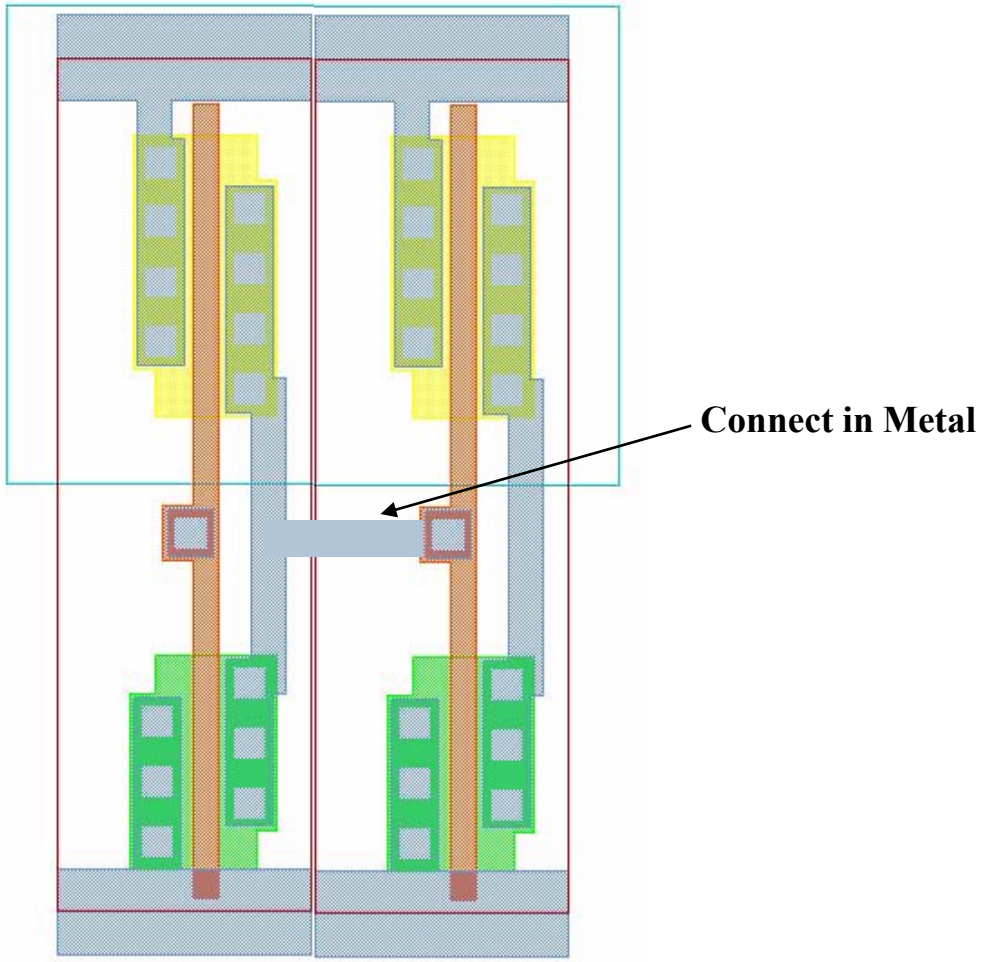
Two Inverters

Layout preference:

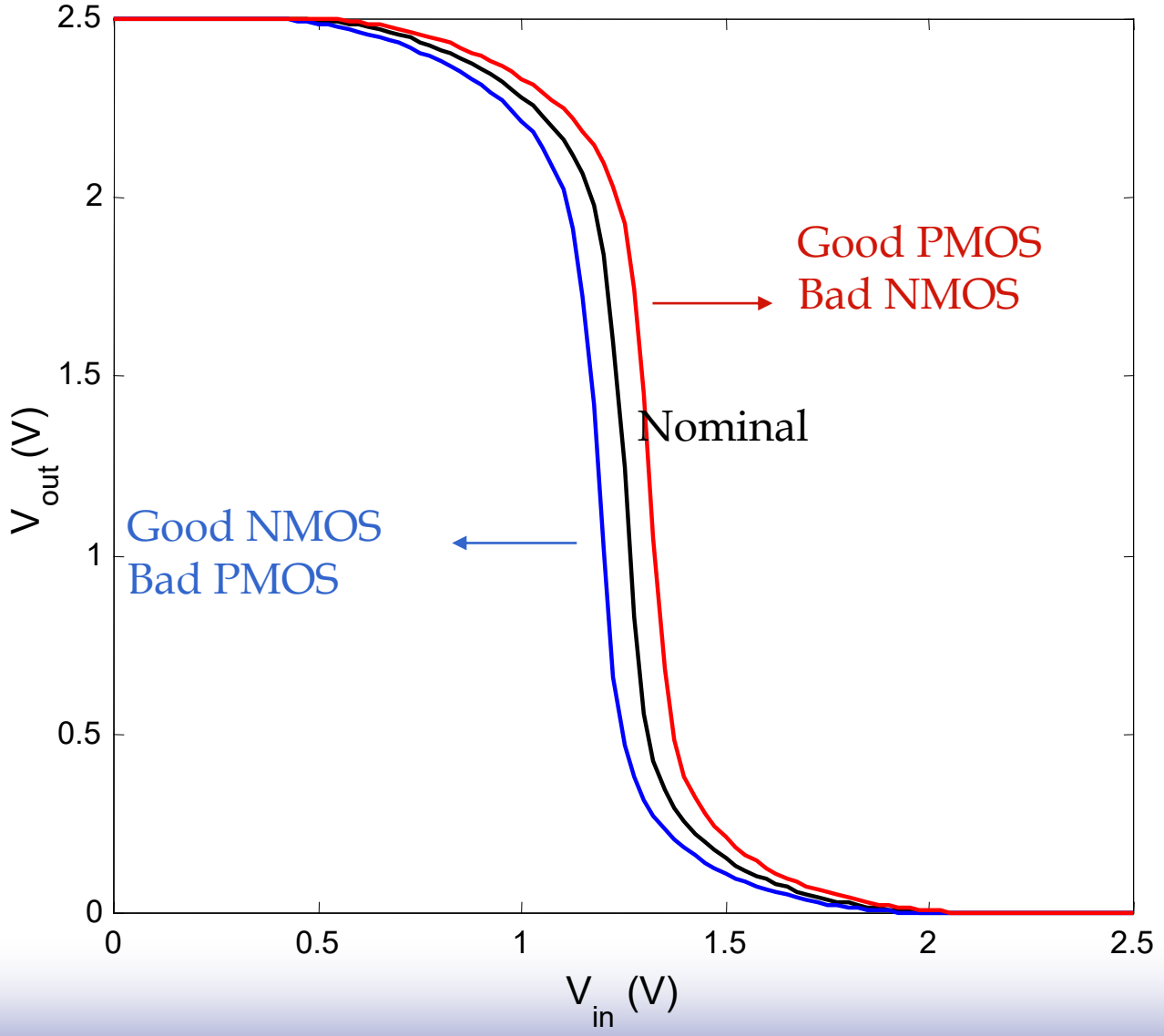
Share power and ground

Abut cells





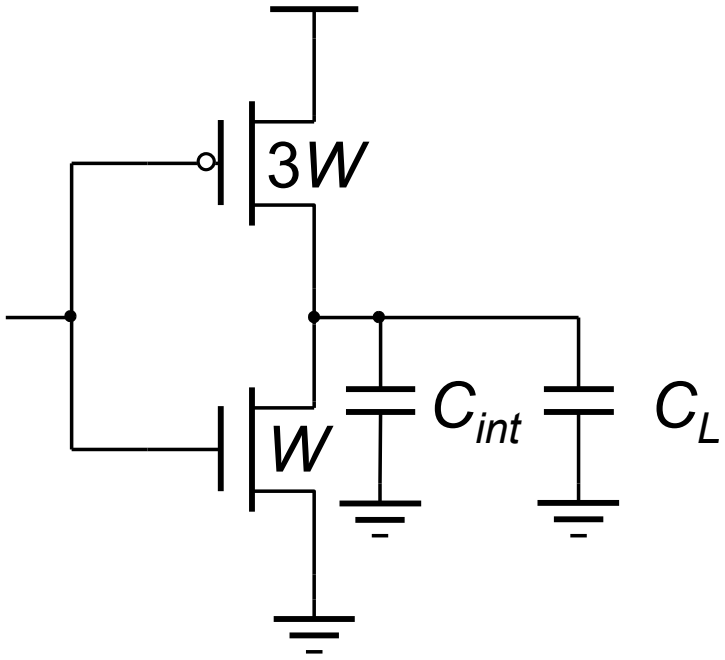
Impact of Process Variations (DFM)

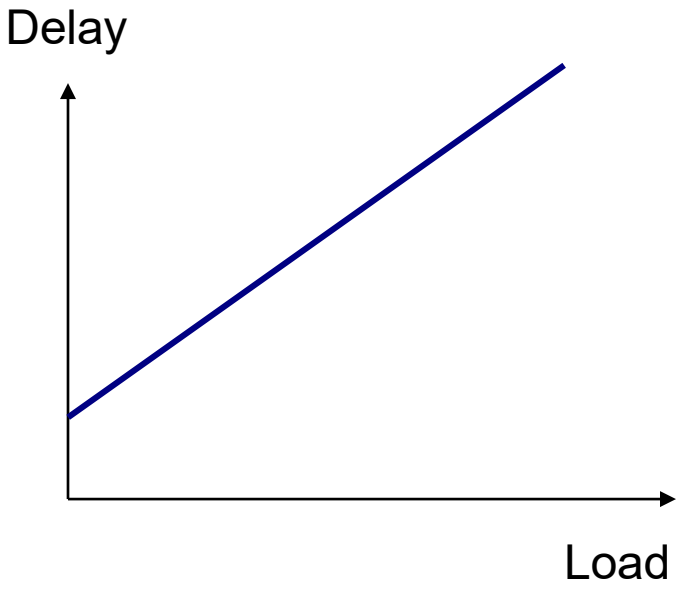


Inverter Sizing for delay

Inverter with Load

W means the size is increased by a factor of *W* with respect to the minimum size





Delay = $kR_W(C_{int} + C_L) = kR_WC_{int} + kR_WC_L$
= Delay (Internal) + Delay (Load)
= $kR_W C_{int}(1+ C_L / C_{int})$

Delay as function of size

Delay = $kR_W C_{int}(1+ C_L /C_{int})$
= Delay (Internal) + Delay (Load)

$R_W = R_{unit} / W$; $C_{int} = W C_{unit}$

$$t_p = t_{p0}(1 + C_L /(WC_{unit}))$$

$t_{p0} = 0.69 R_{unit} C_{unit}$

- *Intrinsic delay is fixed and independent of size W*

- Making *W* large yields better performance gain, eliminating the impact of external load and reducing the delay to intrinsic only. But smaller gain at penalty of silicon area if *W* is too large!

Delay Formula

$$Delay \sim R_W (C_{int} + C_L)$$

$$t_p = kR_W C_{\text{int}} (1 + C_L / C_{int}) = t_{p0} (1 + f / \gamma)$$

$C_{int} = \gamma C_{gin}$ with $\gamma \approx 1$ for modern technology

(see page199 in book or Slid 14 for an example)

C_{gin} : input gate capacitance

$C_L = f C_{gin}$ - effective fanout

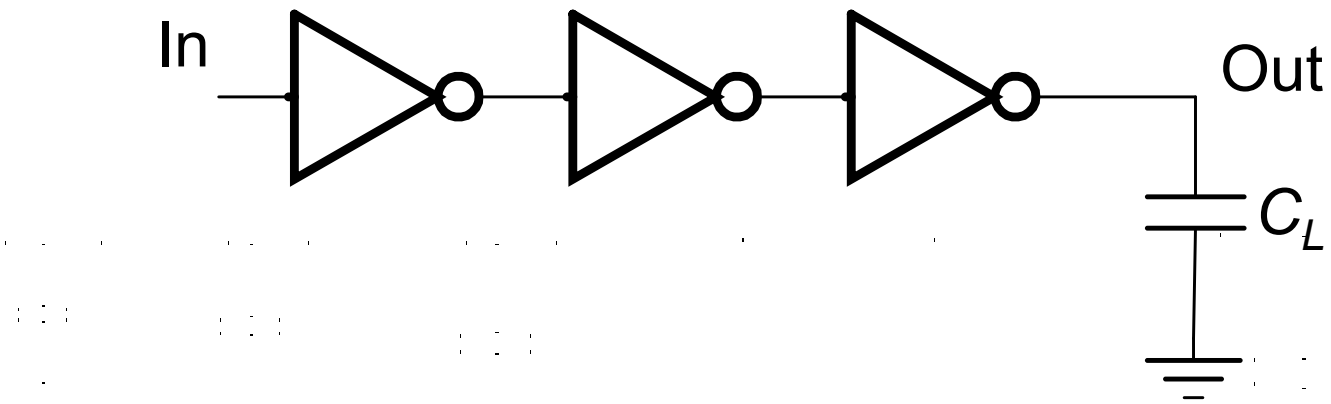
This formula maps the intrinsic capacitor and load capacitor as functions of ***a common capacitor***, which is the gate capacitance of the minimum-size inverter

Single inverter versus inverter chain

- Gate sizing for an isolated gate is not really meaningful. Realistic chips always have a long chain of gates.

- So, a more relevant and realistic problem is to determine the optimal sizing for a chain of gates.

Inverter Chain

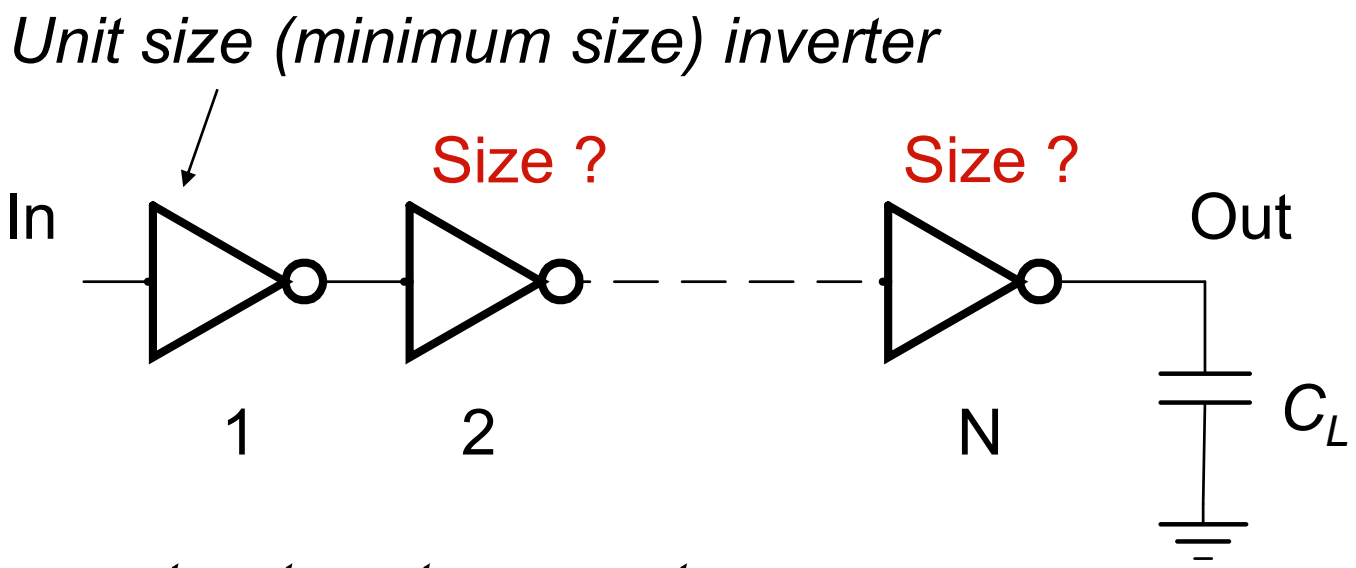


If C_L is given:

- How many stages are needed to minimize the delay?

- How to size the inverters?

Apply to Inverter Chain (fixed N stages)



$$t_p = t_{p1} + t_{p2} + \ldots + t_{pN}$$

$$t_{pj} \sim R_{unit} C_{unit} \left(1 + \frac{C_{gin,j+1}}{\gamma\, C_{gin,j}} \right)$$

$$t_p = \sum_{j=1}^{N} t_{p,j} = t_{p0} \sum_{i=1}^{N} \left(1 + \frac{C_{gin,j+1}}{\gamma\, C_{gin,j}} \right), \quad C_{gin,N+1} = C_L$$

Optimal Tapering for Given *N*

- Delay equation has *N* - 1 unknowns, $C_{\text{gin},2}$ – $C_{gin,N}$
- Minimize the delay, find *N* - 1 partial derivatives equated to 0
- Result: $C_{gin,j+1}/C_{gin,j} = C_{gin,j}/C_{gin,j-1}$
- Size of each stage is the geometric mean of two neighbors

$$C_{gin,j} = \sqrt{C_{gin,j-1}C_{gin,j+1}}$$

- *each stage has the same effective fanout*
- *each stage has the same delay*

Optimum Delay for fixed N stages

When each stage is sized by *f* and has same effective fanout *f*:

$$f^N = F = C_L / C_{gin,1}$$ ← effective fanout of the overall circuit

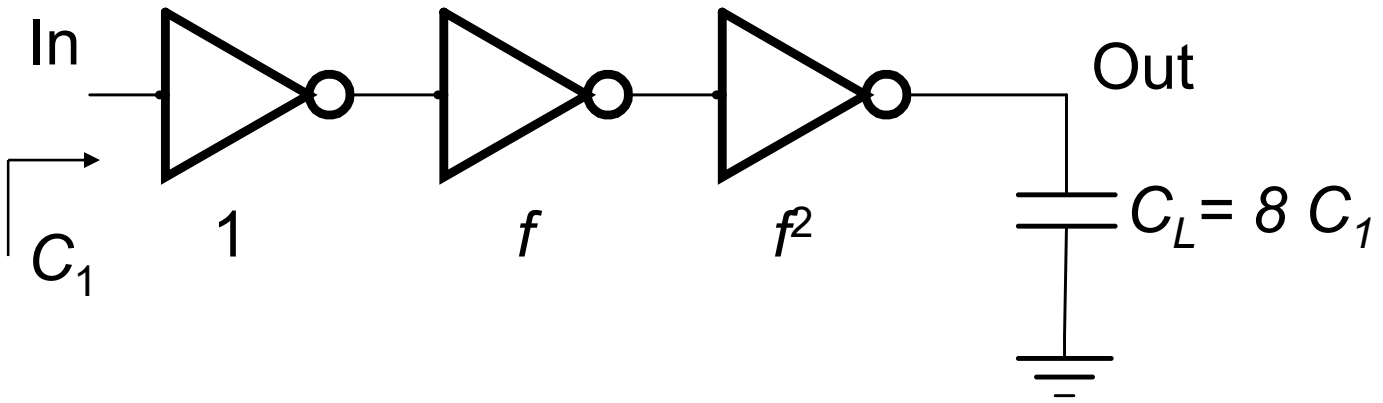
Effective fanout of each stage:

$$f = \sqrt[N]{F}$$

Minimum path delay

$$t_p = \sum_{j=1}^{N} t_{p,j} = t_{p0} \sum_{i=1}^{N} \left(1 + \frac{C_{gin,j+1}}{\gamma\, C_{gin,j}} \right) = N t_{p0} (1 + \sqrt[N]{F} / \gamma)$$

Example



C_L/C_1 has to be evenly distributed across $N = 3$ stages:

$$f = \sqrt[3]{8} = 2$$

Optimum Number of Stages

For a given load C_L and given input capacitance C_{in}
Find optimal sizing f

$$C_L = F \cdot C_{in} = f^N C_{in} \quad \text{with} \quad N = \frac{\ln F}{\ln f}$$

$$t_p = N t_{p0}\left(F^{1/N}/\gamma + 1\right) = \frac{t_{p0} \ln F}{\gamma}\left(\frac{f}{\ln f} + \frac{\gamma}{\ln f}\right)$$

$$\frac{\partial t_p}{\partial f} = \frac{t_{p0} \ln F}{\gamma} \cdot \frac{\ln f - 1 - \gamma/f}{\ln^2 f} = 0$$

For $\gamma = 0$, $f = e$, $N = \ln F$

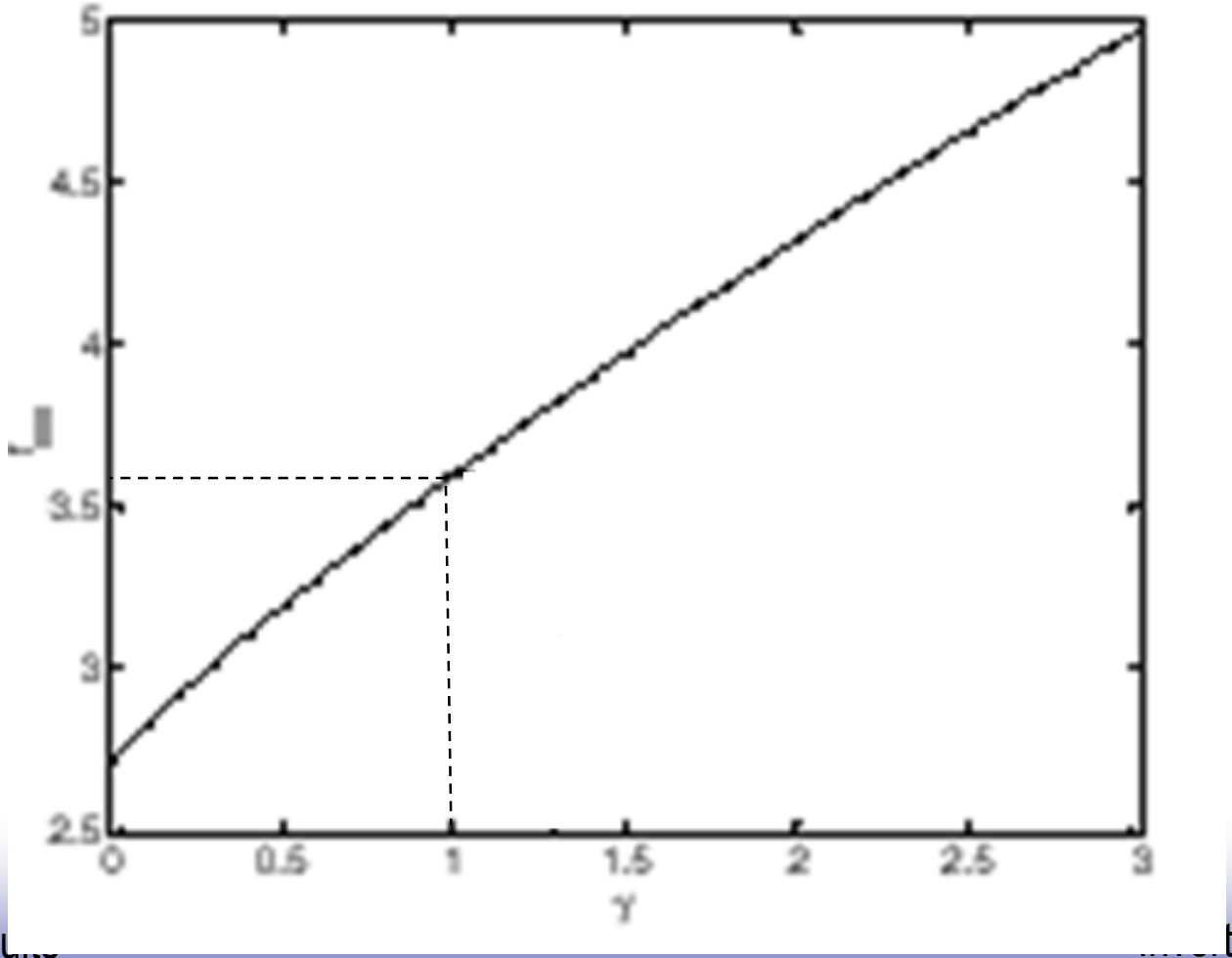
$$f = \exp\left(1 + \gamma/f\right)$$

Optimum Effective Fanout f

Optimum *f* for given process defined by γ

$$f = \exp(1 + \gamma / f)$$

f_{opt} = 3.6 for γ=1



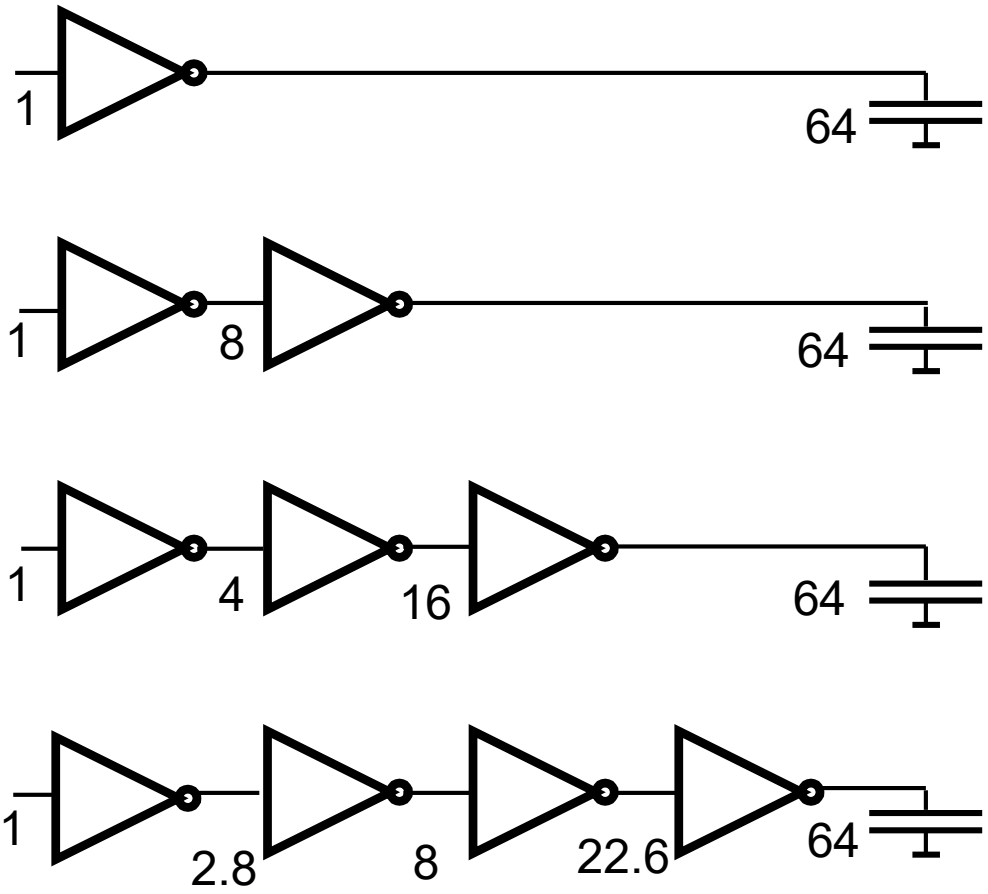
Impact of introducing buffers

$$t_p = Nt_{p0}\left(1 + \sqrt[N]{F}/\gamma\right)$$

$f_{opt} = 4$

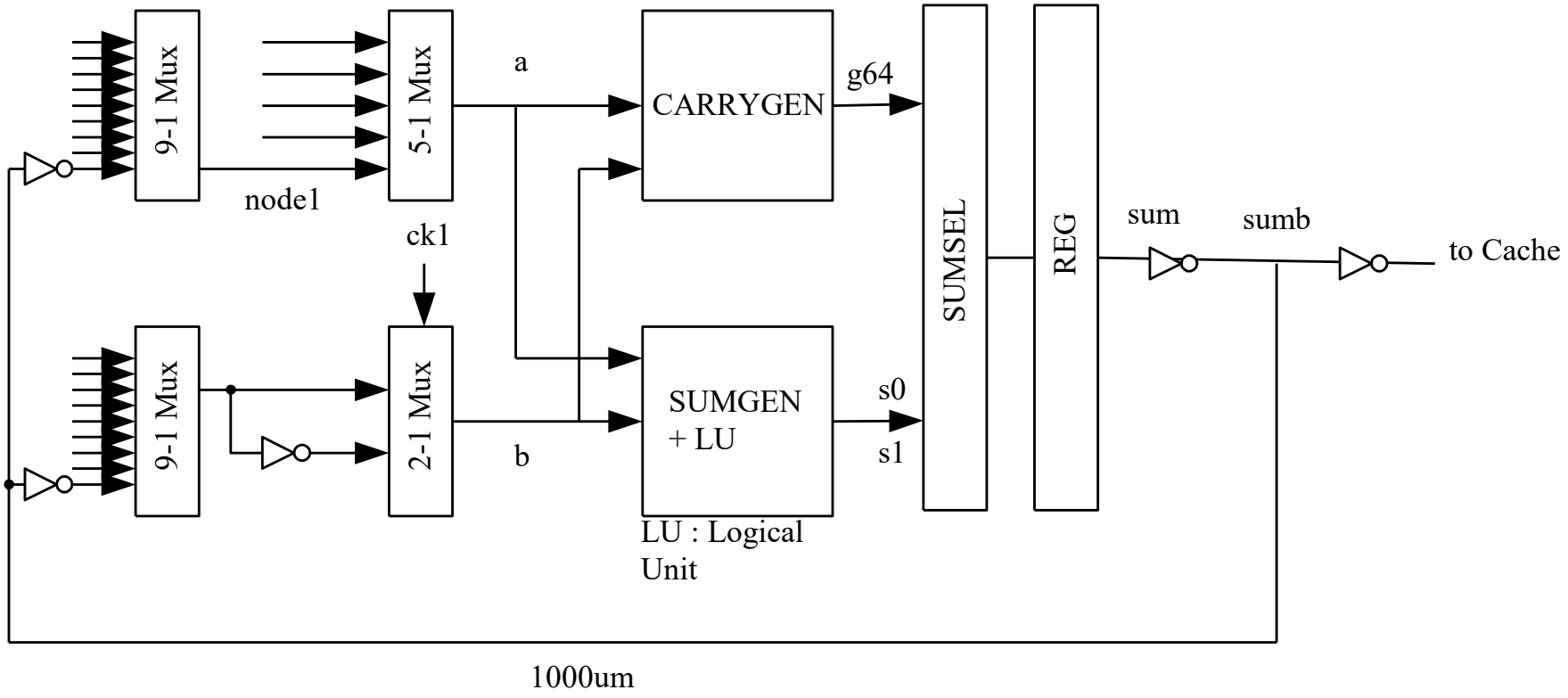
F	Unbuffered	Two Stage	Inverter Chain
10	11	8.3	8.3
100	101	22	16.5
1000	1001	65	24.8
10,000	10,001	202	33.1

Buffer Design



N	f	t_p
1	64	65
2	8	18
3	4	15
4	2.8	15.3

Intel Itanium Microprocessor



Itanium has 6 integer execution units like this