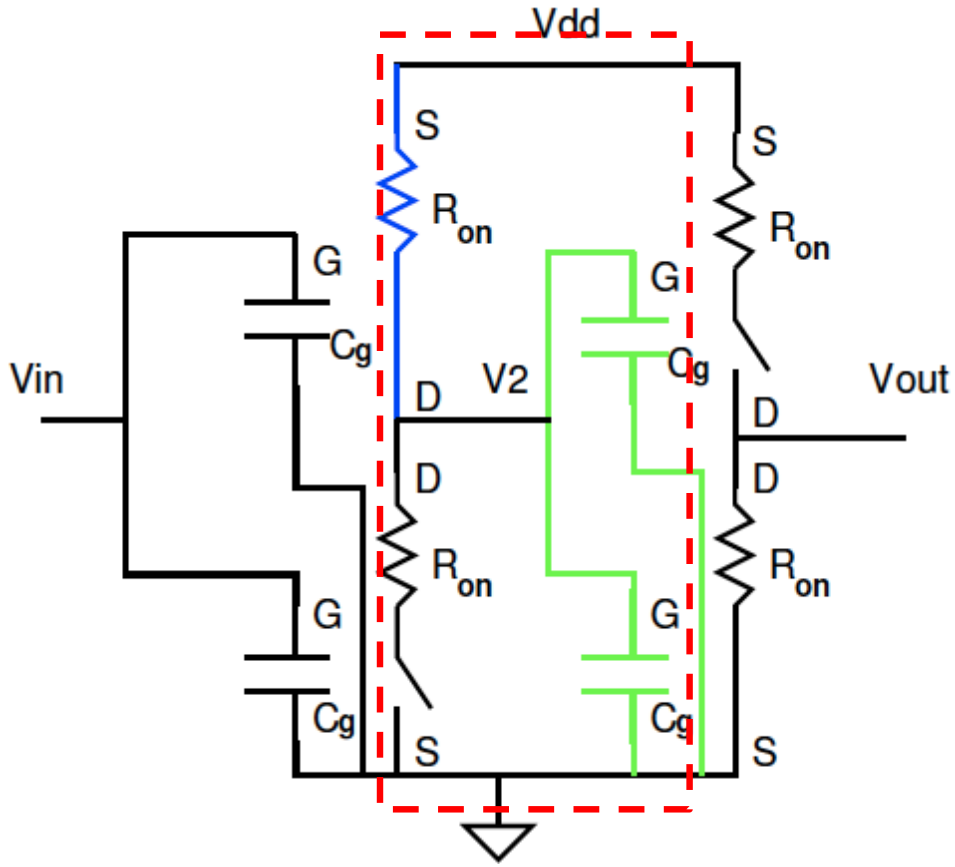
ESE370: Circuit-Level Modeling, Design, and Optimization for Digital Systems

Lec 4: September 13, 2021

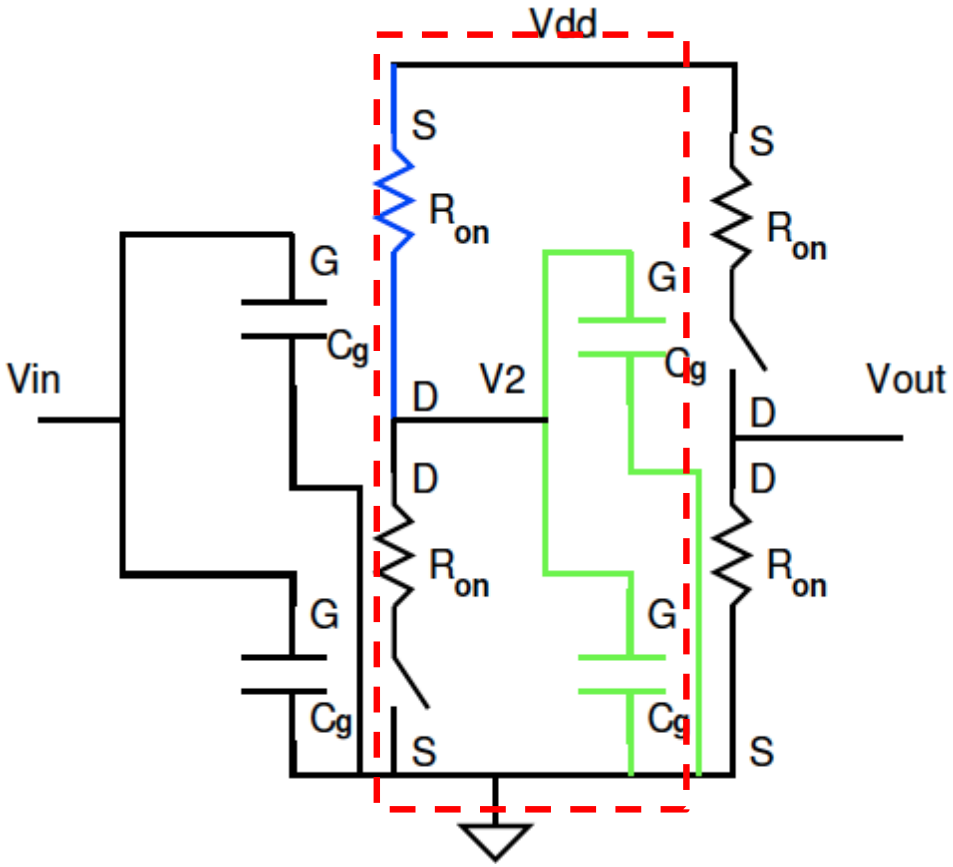
Regenerative Property

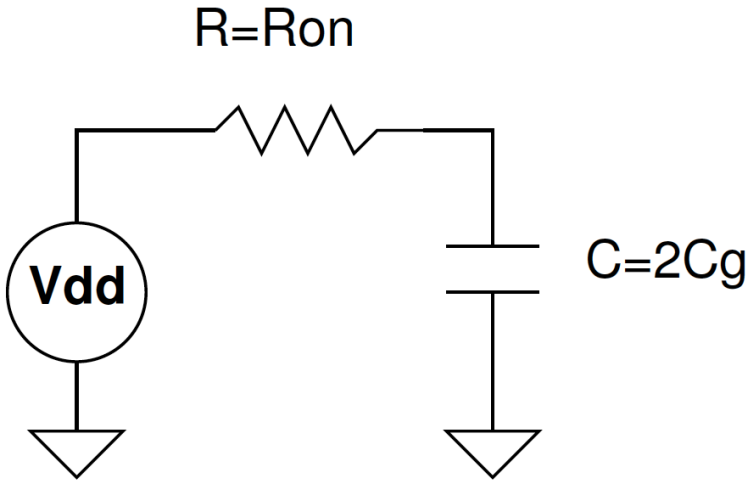
CMOS Buffer Gate - First Order



- Stage 1 equivalent circuit for the gate output
- Load on V_2
  - Capacitive, input of stage 2

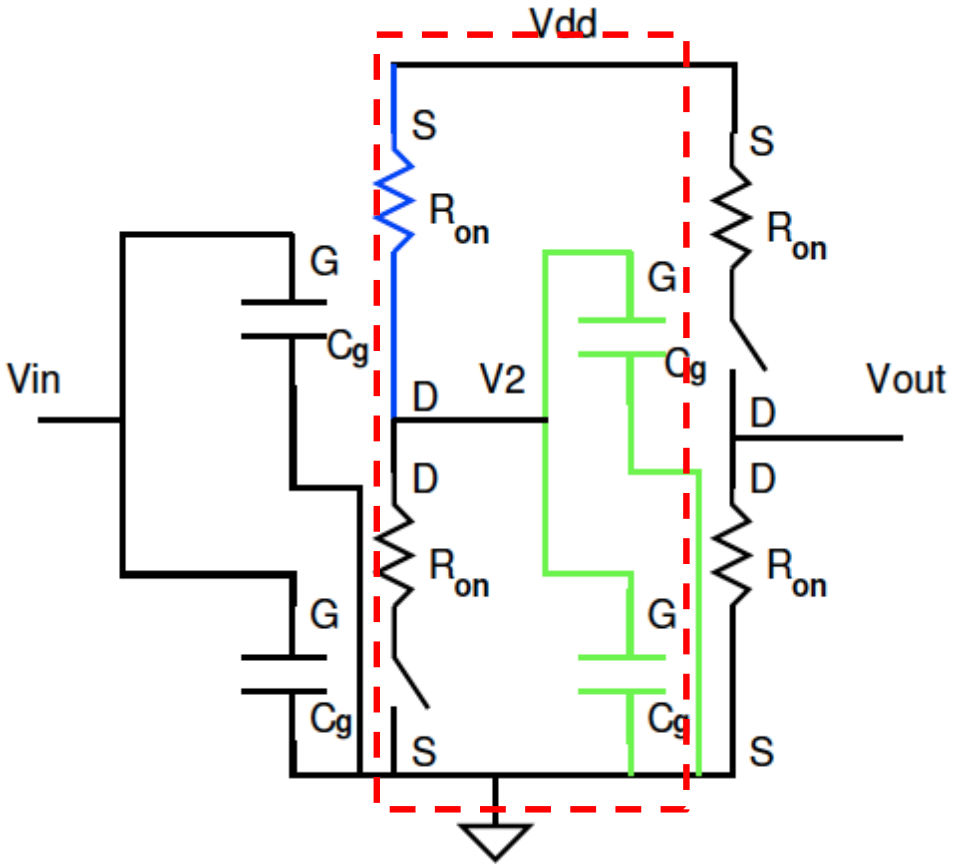
CMOS Buffer Gate - First Order

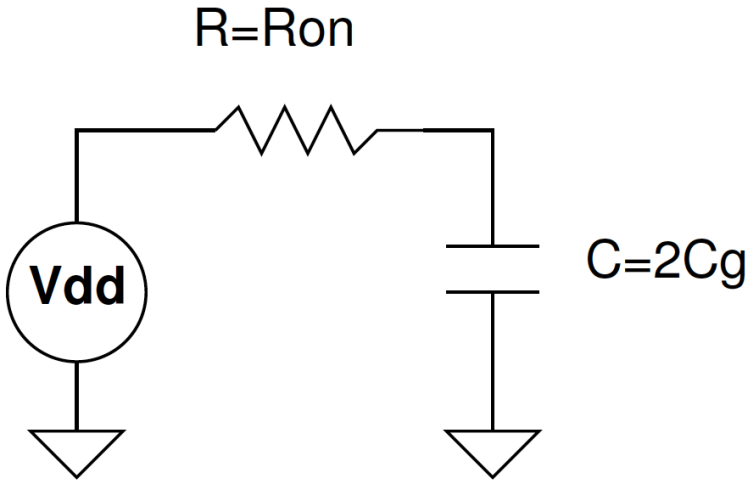




- Stage 1 equivalent circuit for the gate output
- Load on V_2
  - Capacitive, input of stage 2

CMOS Buffer Gate - First Order





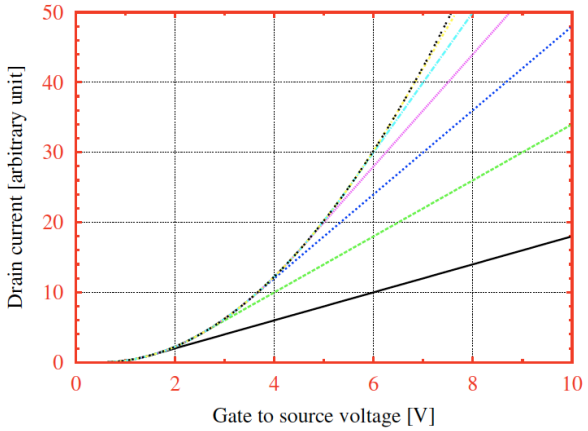
- What is time constant of V_2 when V_{in} switches from V_{DD} to 0?
  - $\tau = 2R_{on}C_g$

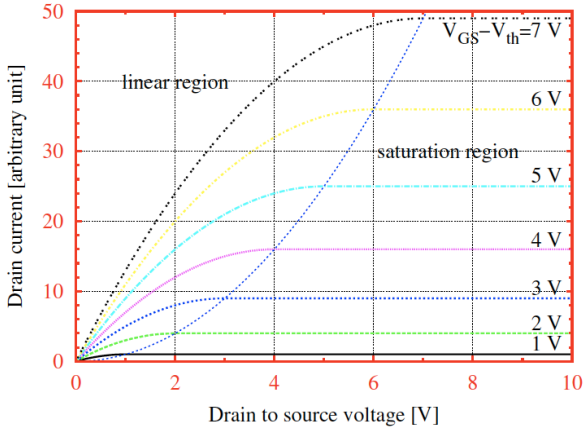
First-Order Model

- Includes settling times/delay
- Voltage settling with capacitive loads
  - At least some basis for reasoning about delay

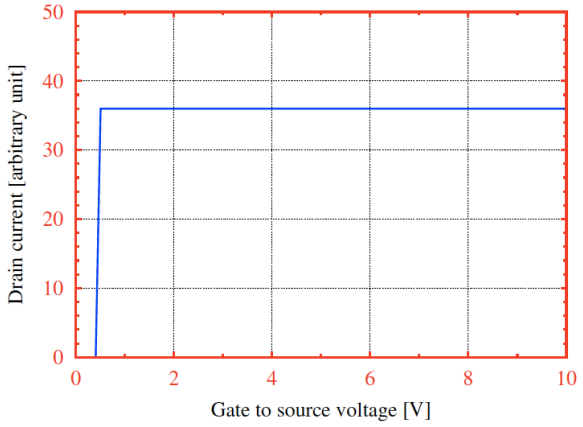
What is still missing?

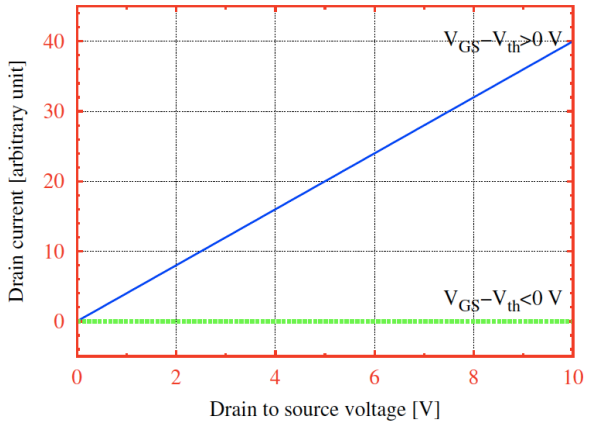
IV curve





1st Order





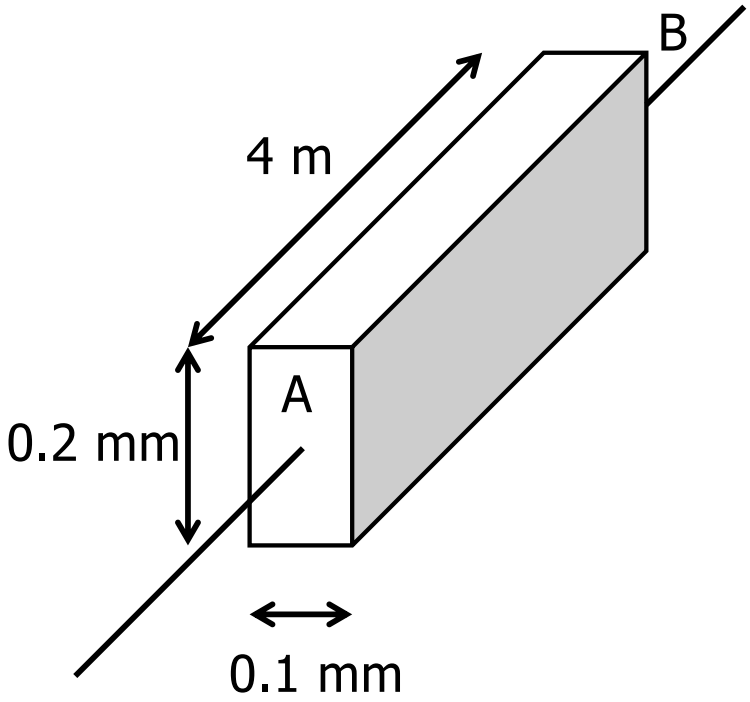
What is still missing?

- What happens at intermediate voltages?
  - When the input is not rail-to-rail (not just gnd or V_{dd} inputs)
- Details of dynamics, including…
  - Input transition is not an ideal step
  - Intermediate drive strengths change with V_{GS}
    - Drain resistance changes
- Sub-threshold operation
  - When $V_{gs}<V_{th}$

Design: Engineering Control

- V_{th}
  - Process engineer
- Drive strength (R_{on})
  - Circuit engineer
  - Control with sizing transistors
- Supply voltages (V_{dd})
  - Range set by process engineer
  - Detail use by circuit engineer

Wire Resistance

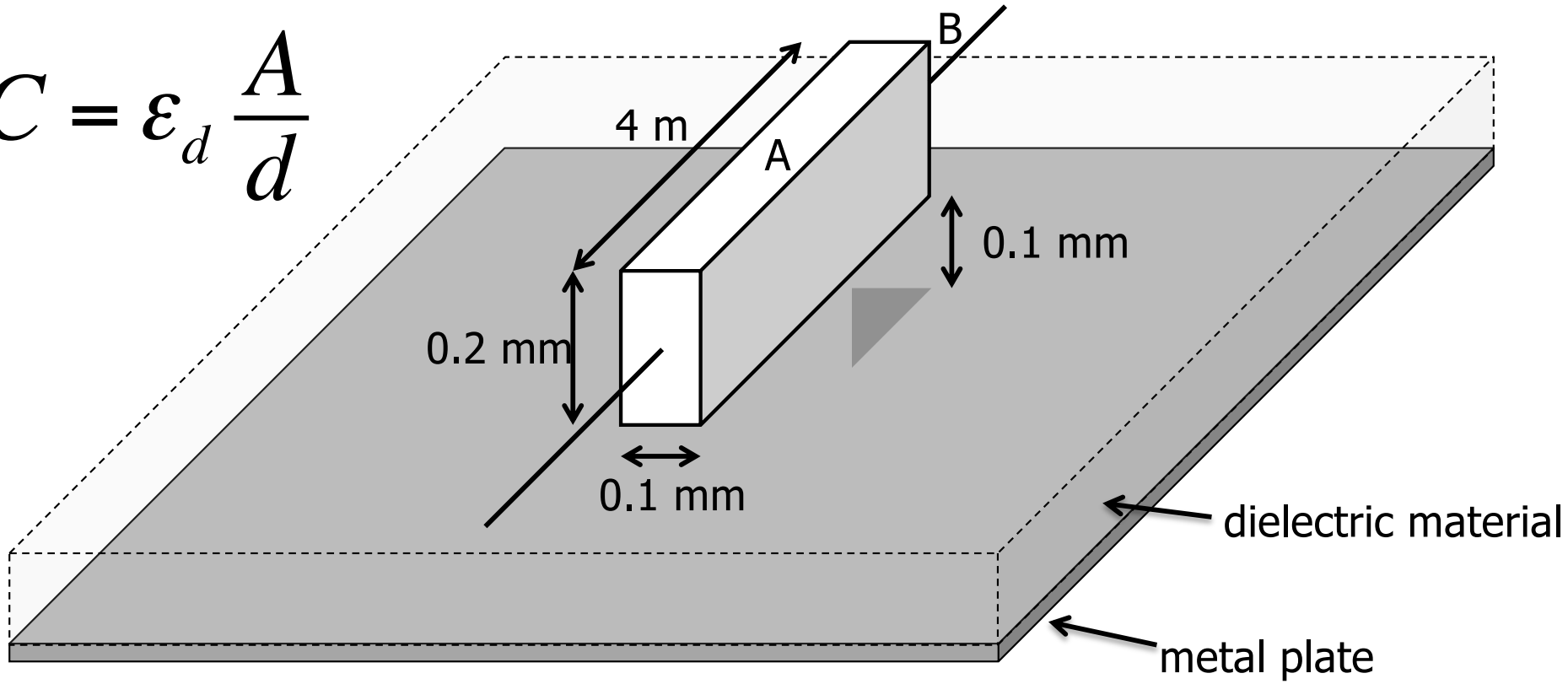


$$R = \frac{\rho L}{A}$$

- Sanity check
  - Wire twice as long = resistors in series
  - Wire twice as wide = resistors in parallel

Wire Capacitance

$$C = \varepsilon_d \frac{A}{d}$$



- Sanity check
  - Wire twice as long = capacitors in parallel
  - Wire twice as wide = capacitors in parallel

There are always Rs and Cs

- Every wire (connection) has resistance
- Every wire has capacitance
- (Every wire has inductance)
  - More later
- Dominant effects
  - $R_{big} + R_{small} \approx R_{big}$ ($R_{wire} << R_{on}$)?
  - $C_{big} \,||\, C_{small} \approx C_{big}$ ($C_{wire} << C_g$) ?
    - Today more likely ($C_{wire} >> C_g$)

Today

- We know how to make our logic functional, but how do we make sure logic is robust?
  - To enable design cascading gates into any (feed forward) graph and still tolerate voltage drops and noise, while maintaining digital abstraction

Outline

- Two signal problems → Gate Cascade failure
- Regeneration Solution → Gate Abstraction
  - Transfer Curves
  - Noise Margins
  - Non-linearity

Two Signal Problems

1. Output does not go to rail

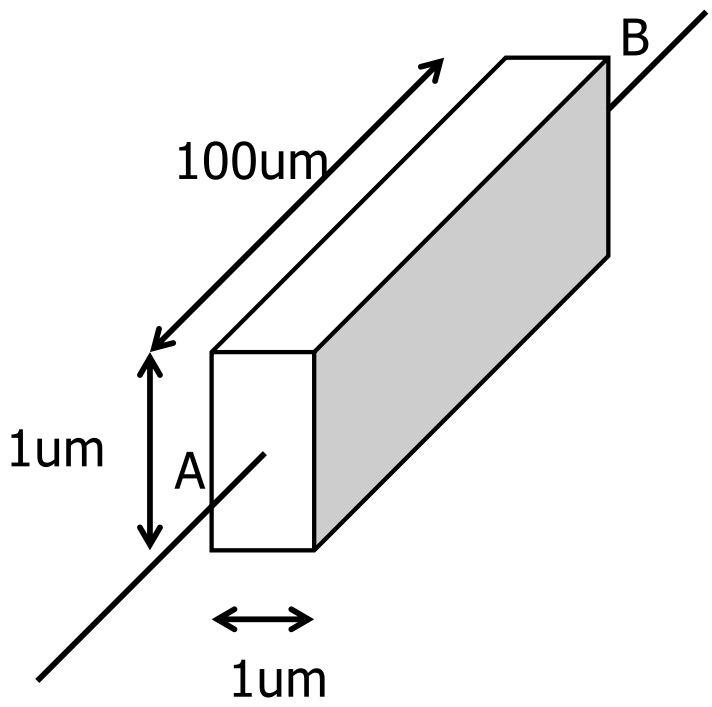
   Stops short of V_{dd} or Gnd

2. Signals may be perturbed by noise

$$V_x = V_{ideal} \pm V_{noise}$$

Wire Resistance (Pg 2 preclass)

- Resistance of 100 μm long wire?

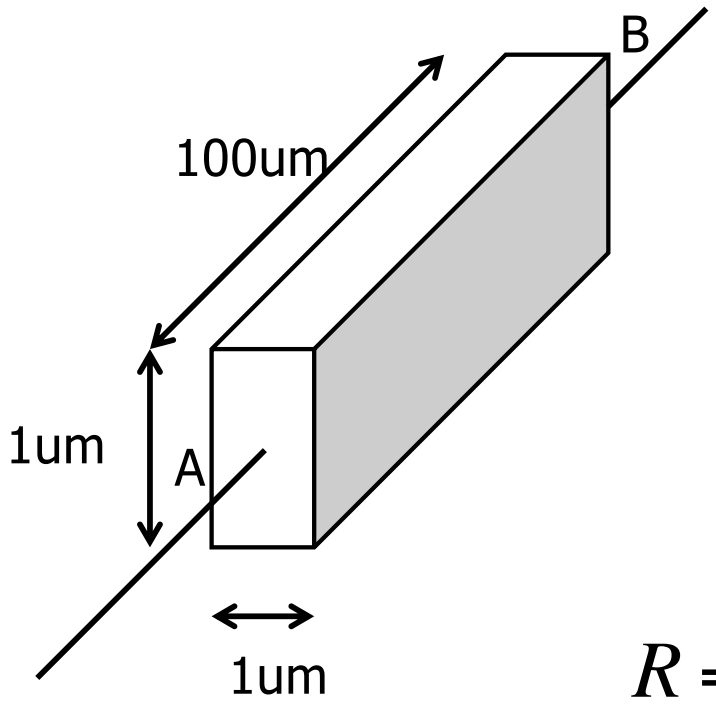


$\rho = 10^{-7}\text{Ohm}\bullet\text{m}$

$$R = \frac{\rho L}{A}$$

Wire Resistance

- 100 μm long wire?



$$R = \frac{\rho L}{A}$$

$$R = \frac{10^{-7}\,\Omega \cdot m \cdot 100\,\mu m}{1\,\mu m \cdot 1\,\mu m} = 10\Omega$$

Wire Resistance

- 1 mm long wire?
- 1 cm long wire?
- Length of integrated circuit chip side?
  - (we often call an IC chip a "die")

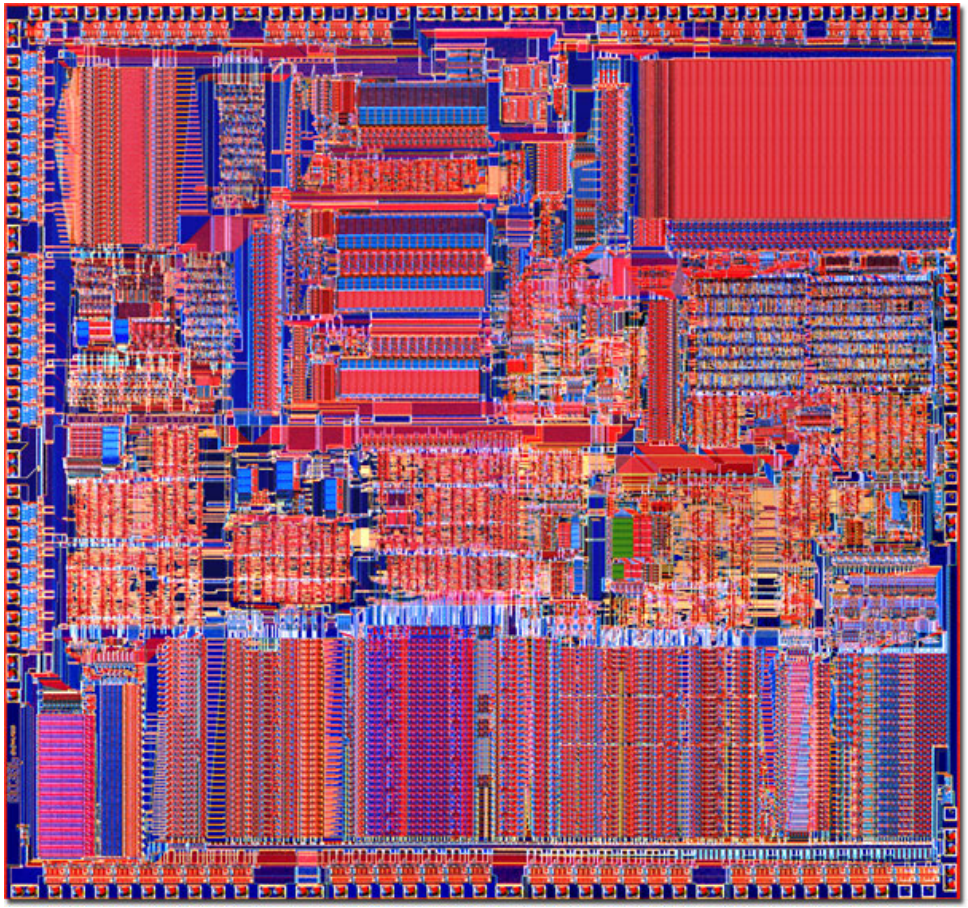
Die Sizes

Chip	#Trans.	Year	Maker	process	mm^2
GK10 Kepler	7B	2012	NVIDIA	28nm	561
22-core Xeon Broadwell-E5	7B	2016	Intel	14nm	456
GC2 IPU	23.6B	2018	Graphcore	16nm	825
Apple A12X Bionic	10B	2018	Apple	7nm	122
Tegra Xavier SoC	9B	2018	Nvidia	12nm	350
Navi 10	10B	2021	AMD	7nm	251

source: http://en.wikipedia.org/wiki/Transistor_count

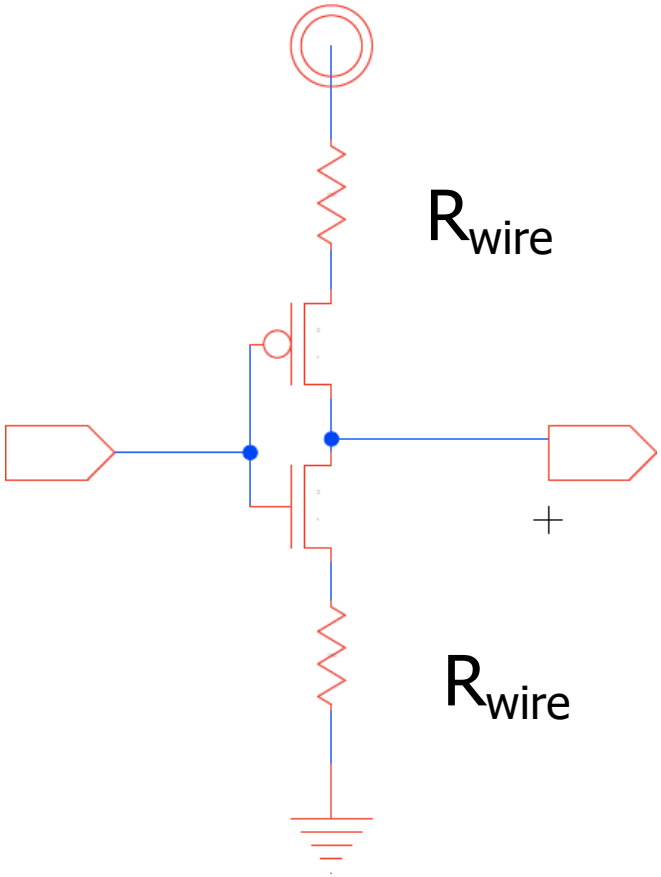
Implications

- What does the circuit really look like for an inverter in the middle of the chip?



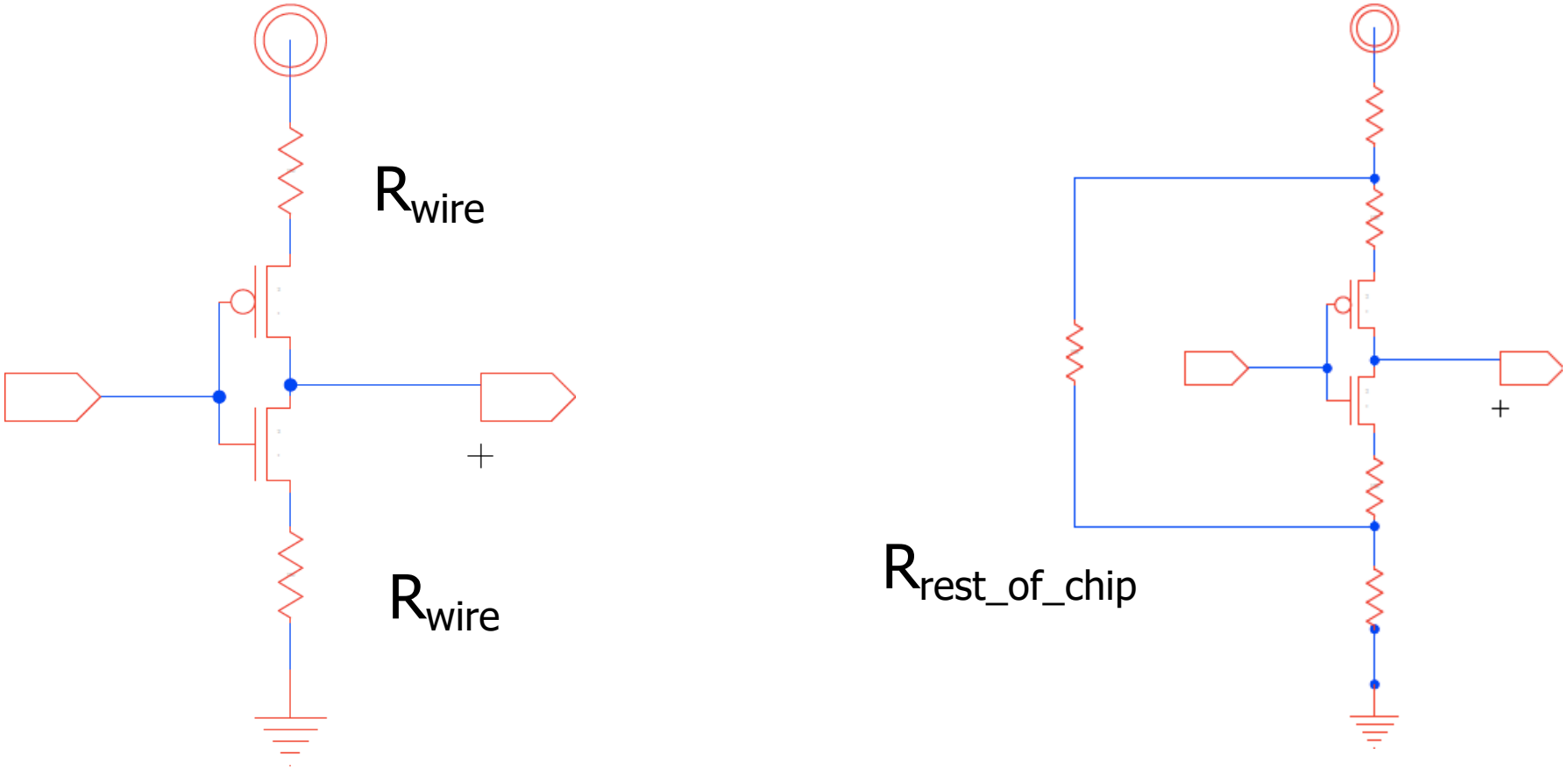
Implications

- What does the circuit really look like for an inverter in the middle of the chip?



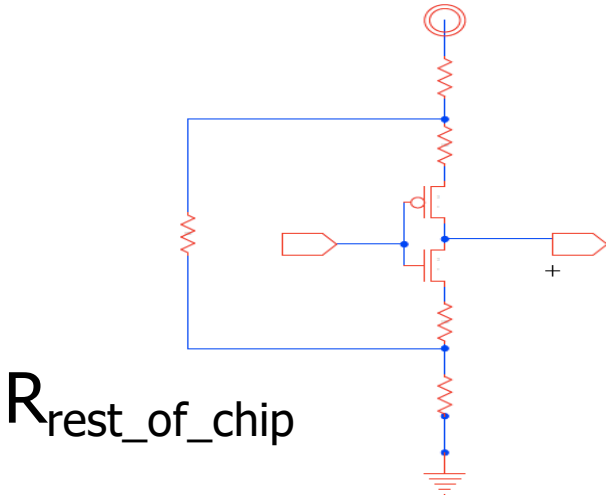
Implications

- What does the circuit really look like for an inverter in the middle of the chip?



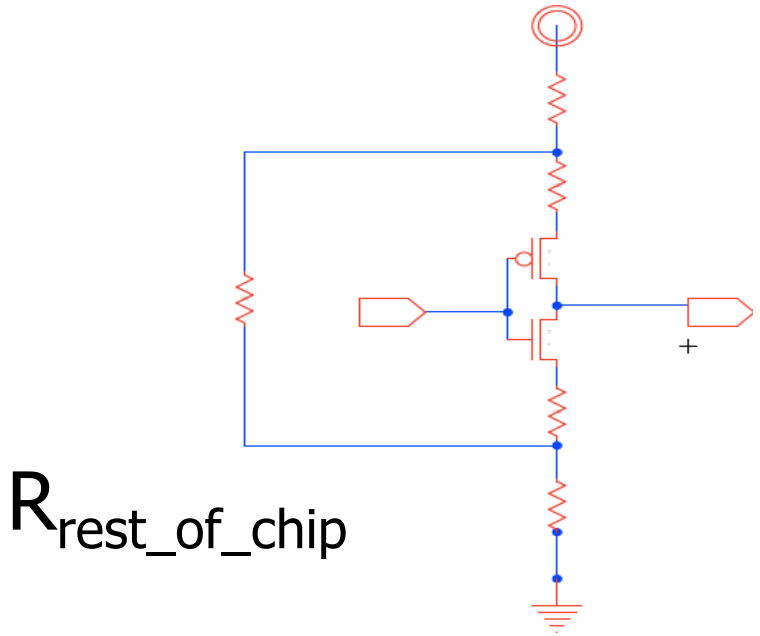
IR-Drop

- Since interconnect is resistive and gates pull current off the supply interconnect
  - The V_{dd} seen by a gate is lower than the supply Voltage by
    - $V_{drop}=I_{supply}$ x $R_{distribute}$
  - Two gates in different locations
    - See different $R_{distribute}$
    - Therefore, see different V_{drop}



Output does not go to Rail

- Due to V_{drop}, "rails" for two communicating gates may not match



Two Signal Problems

1. Output not go to rail
   - Different swing for gates
2. Signals may be perturbed by noise
   - Voltage seen at input to a gate may be lower/higher than input voltage

Noise Sources?

- Signal coupling
  - Crosstalk
- Inductive noise
- Leakage
- Ionizing particles (shot noise)

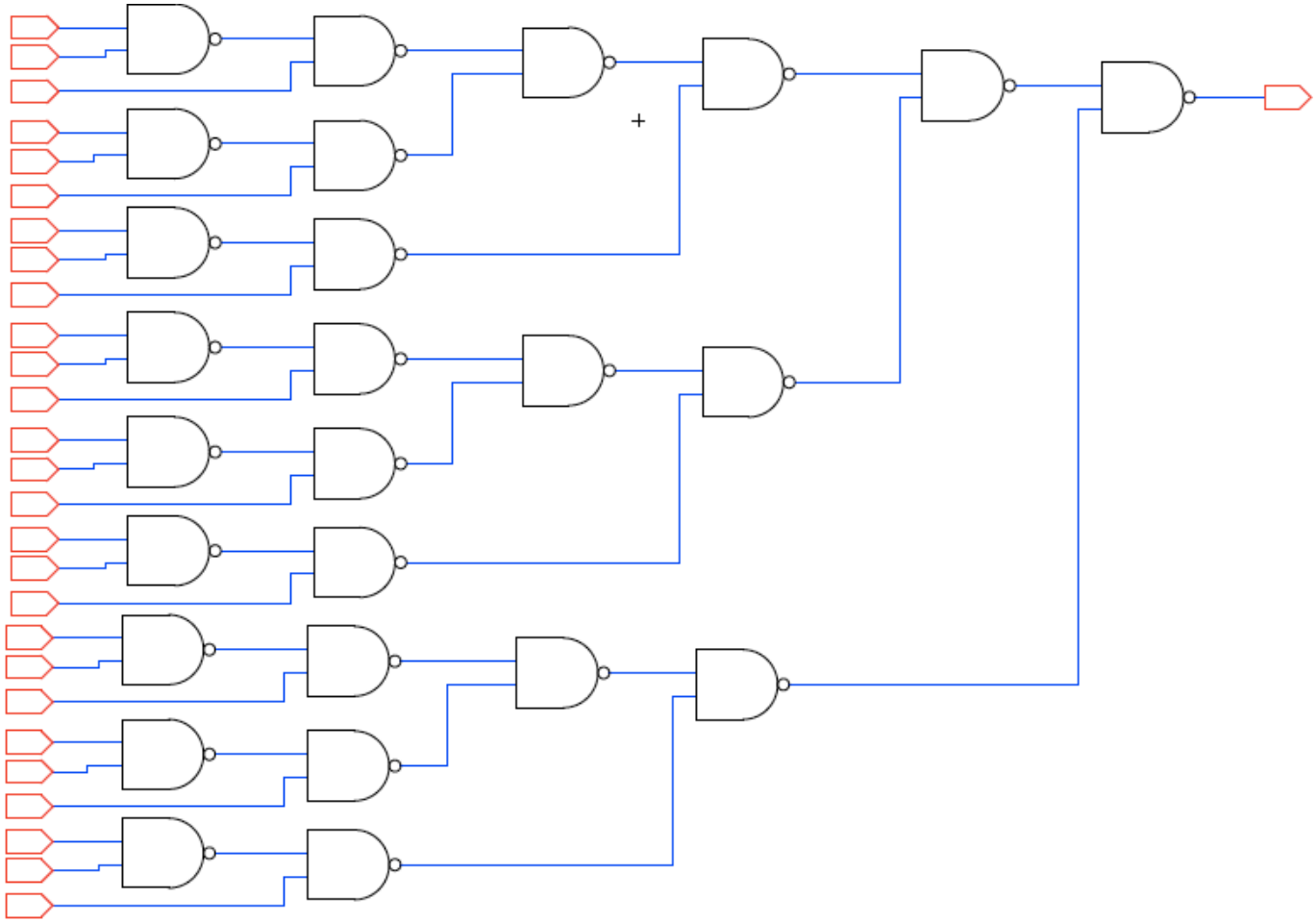
Signals **will** be degraded

1. Output not go to rail
   - Is this tolerable?
2. Signals may be perturbed by noise
   - Voltage seen at input to a gate may be lower/higher than expected input voltage

- What happens to degraded signals?

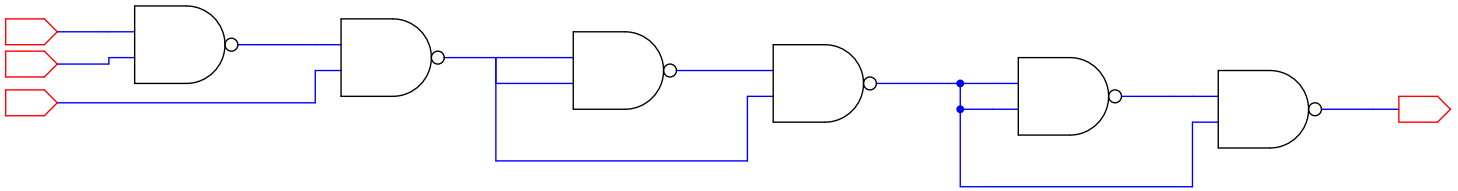
Preclass Pg 1

- What is the output when all inputs are all 1s?



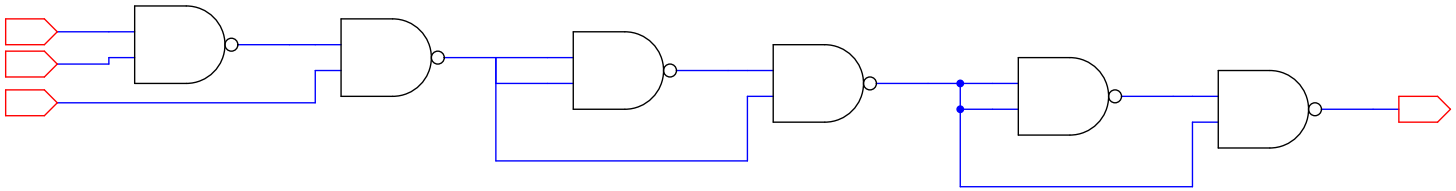
Preclass Pg 1

- What is the output when all inputs are all 1s?



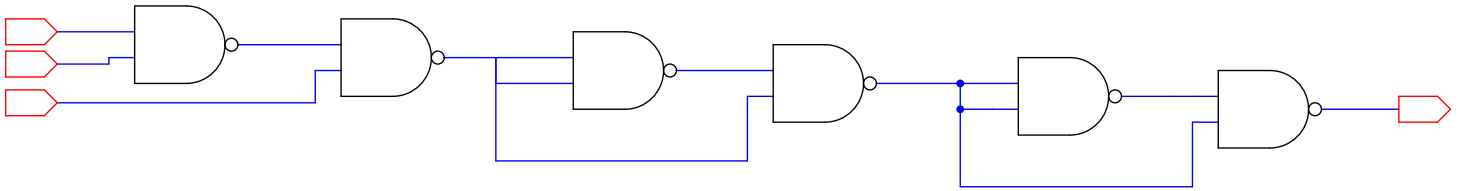
Preclass Pg 1

- What is the output when all inputs are all 1.0 and NAND(A, B) = 1-A*B?



Preclass Pg 1

- What is the output when all inputs are all 0.95 and NAND(A, B) = 1-A*B?



Degradation

- Cannot have signal degrade across cascaded gates
- Want to be able to cascade arbitrary set of gates
  - No limit on number of gates to maintain signal integrity

Gate Creed

- Gates should leave the signal "better" than they found it
  - "better" → closer to the rails

Regeneration Discipline

- Define legal inputs
  - Gate works if V_{in} "close enough" to the rail

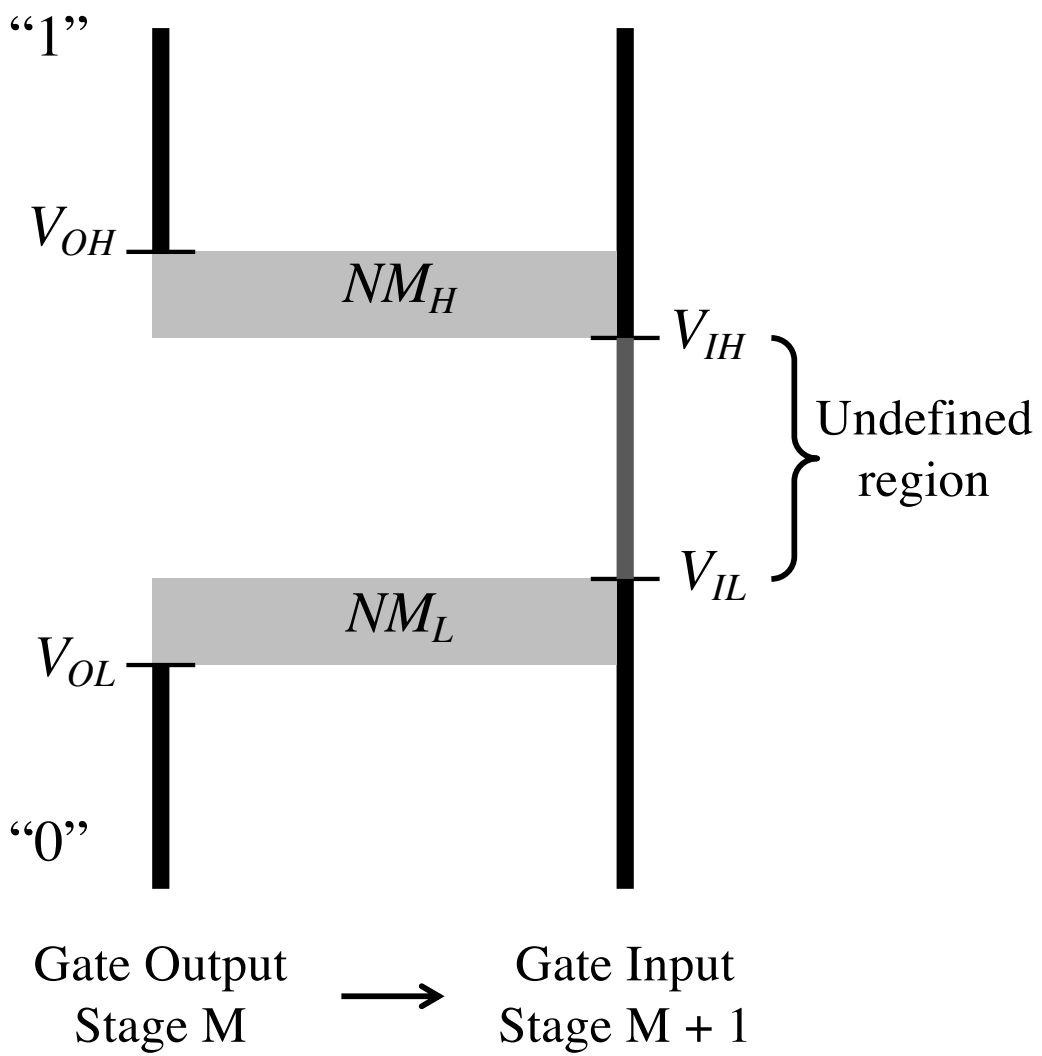
Regeneration Discipline

- Define legal inputs
  - Gate works if V_{in} "close enough" to the rail
- Regeneration
  - Gate produces V_{out} "closer to rail"
    - This tolerates some drop between one gate and next (between out and in)
    - Call this our "Noise Margin"

Regeneration/Restoration/Static Discipline

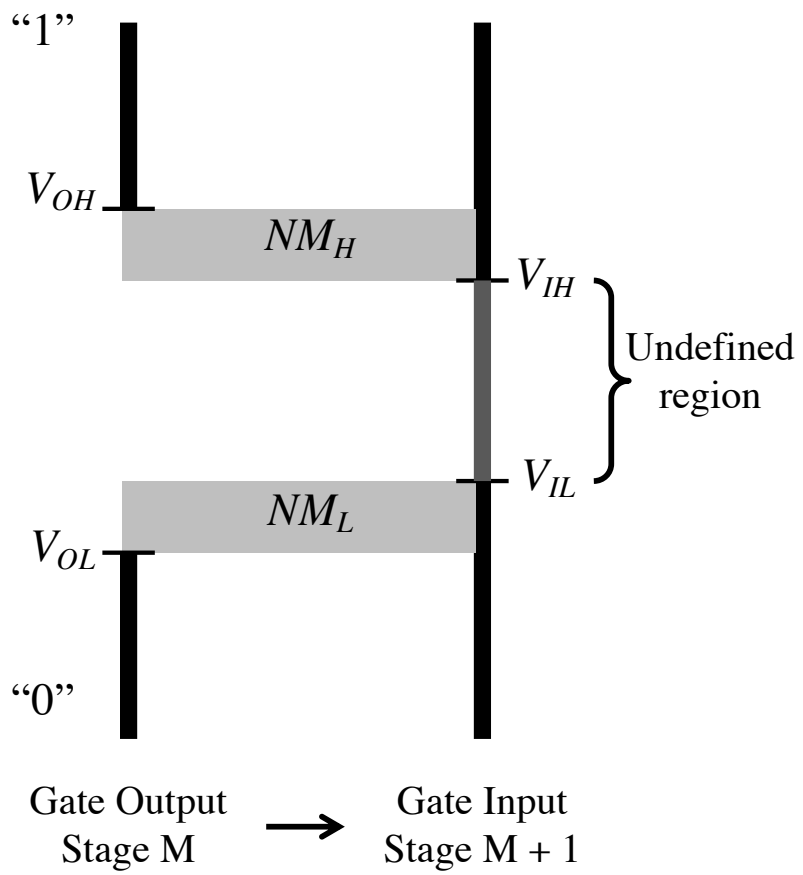
Noise Margin

- V_{OH} – output high
- V_{OL} – output low
- V_{IH} – input high
- V_{IL} – input low
- $NM_H = V_{OH}-V_{IH}$
- $NM_L = V_{IL}-V_{OL}$



Regeneration Discipline (getting precise)

- Define legal inputs
  - Gate works if V_{in} "close enough" to the rail
  - $V_{in} > V_{IH}$ or $V_{in} < V_{IL}$
- Regeneration
  - Gate produces V_{out} "closer to rail"
    - $V_{out} < V_{OL}$ or $V_{out} > V_{OH}$

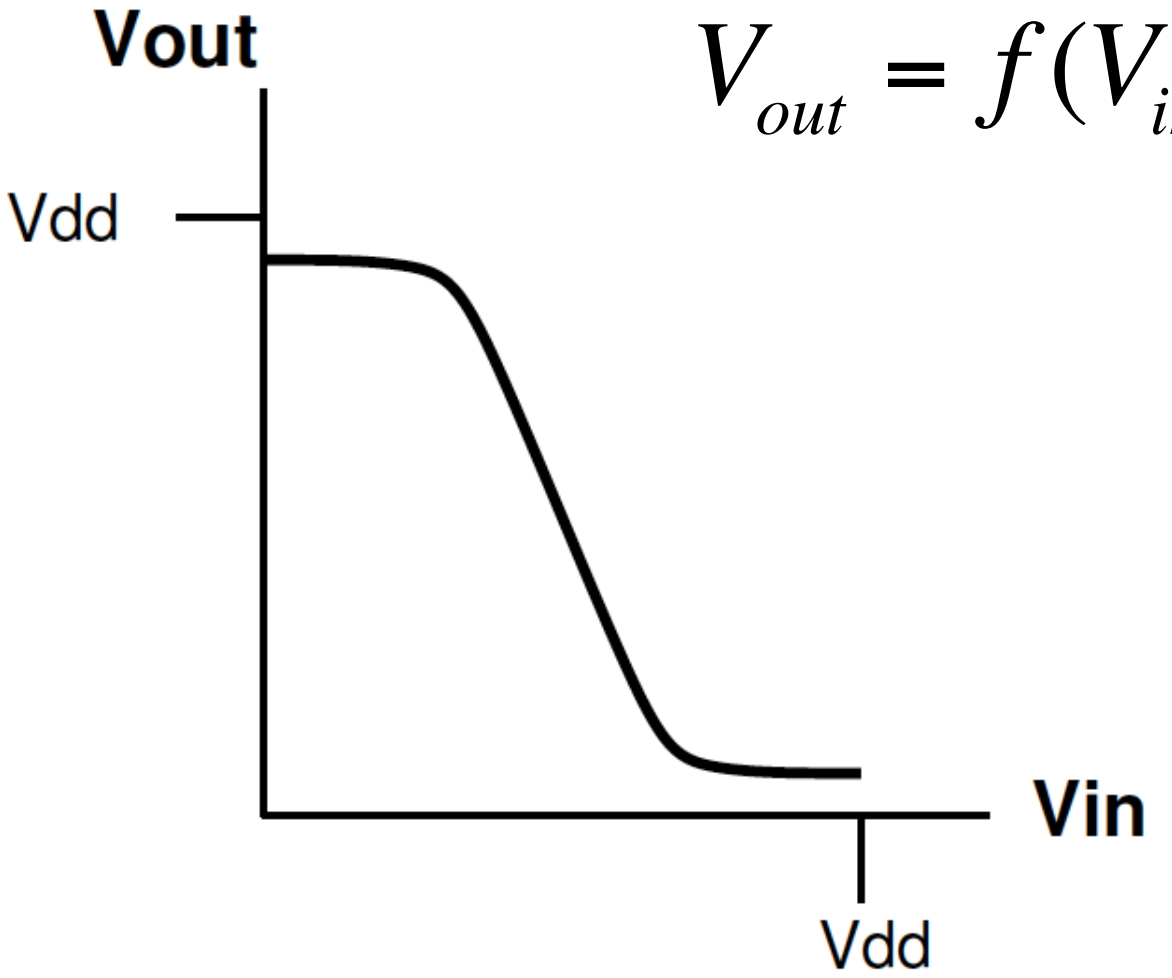


Transfer Function

- Describes what the output is given logic gate input
  - $V_{out}=f(V_{in})$

Transfer Function

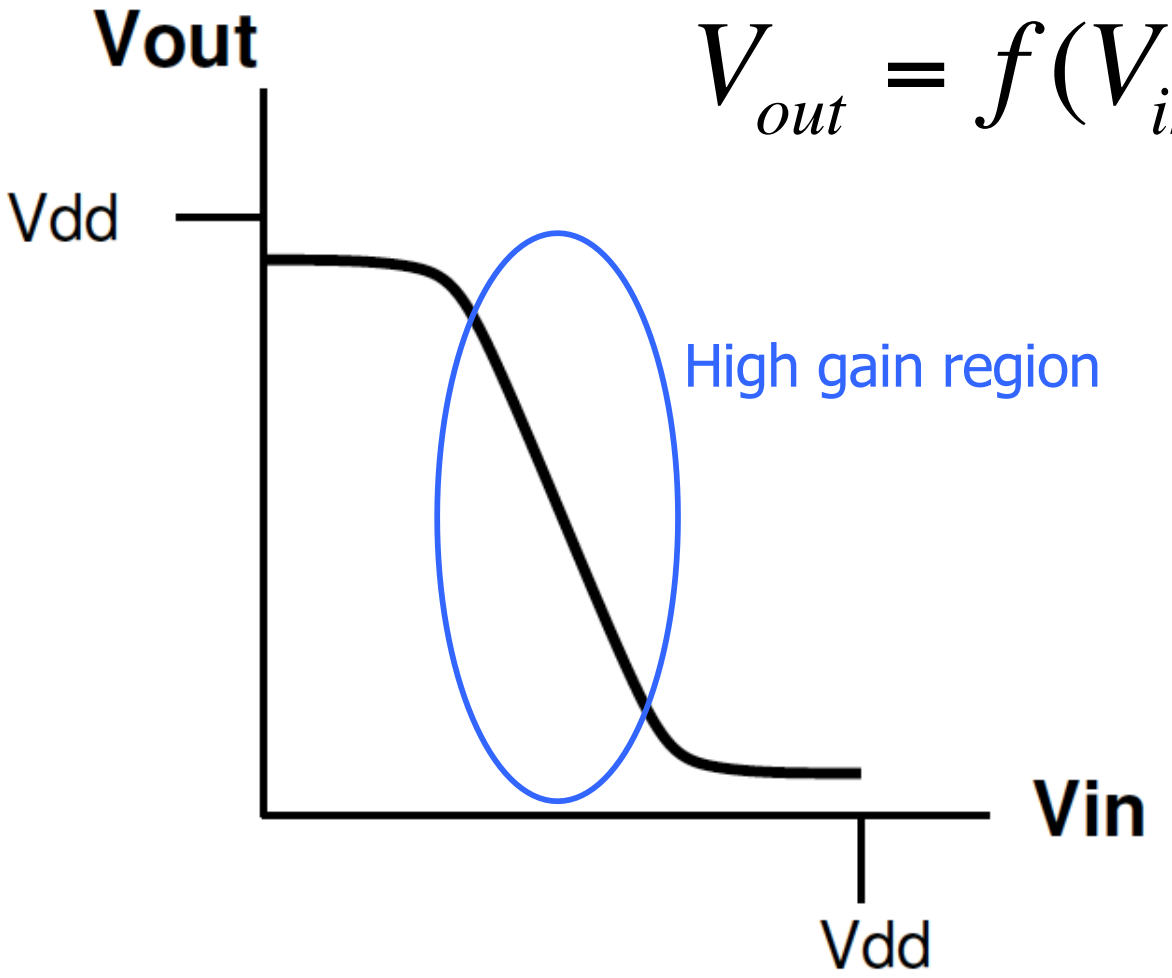
- What gate is this?



$$V_{out} = f(V_{in})$$

Transfer Function

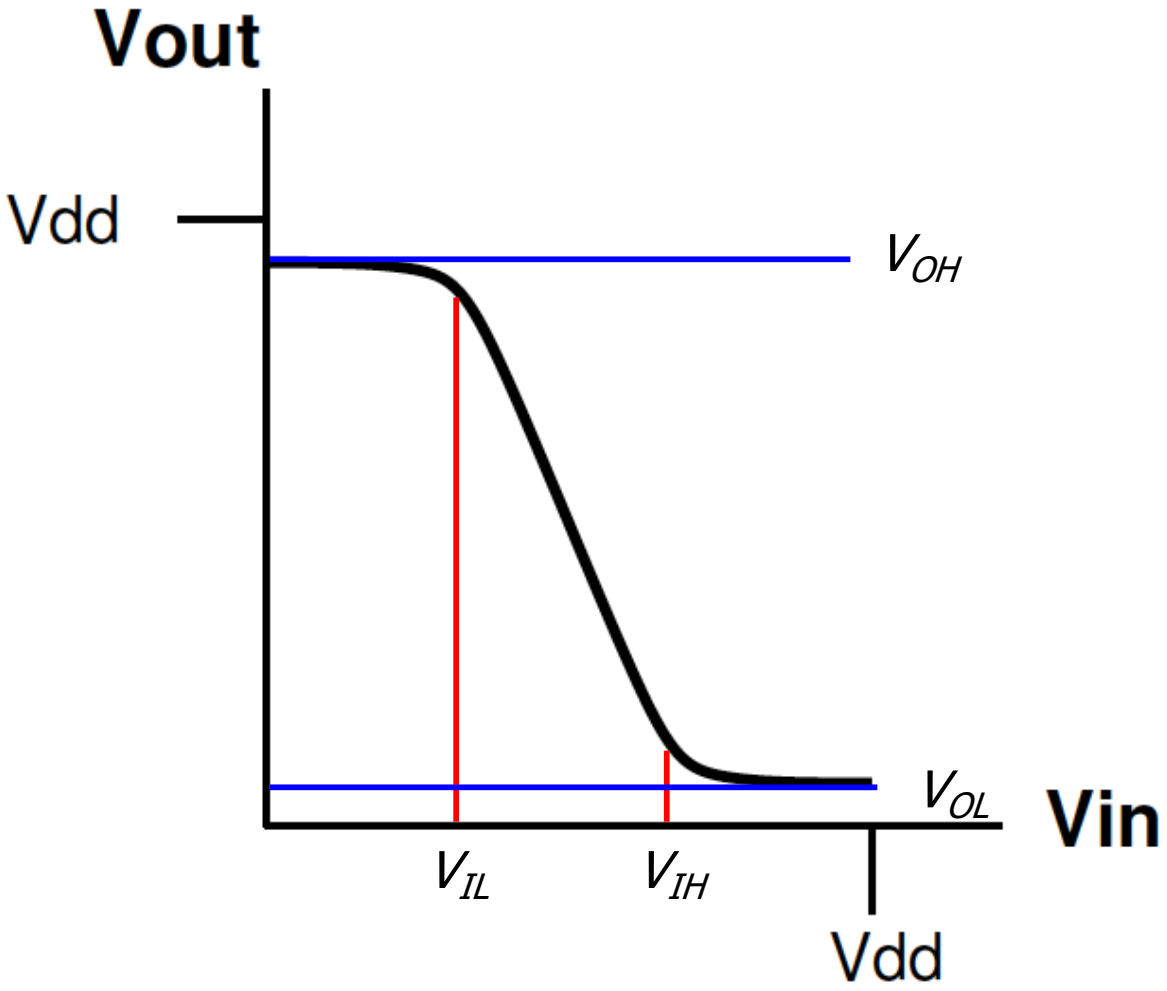
- What gate is this?



$$V_{out} = f(V_{in})$$

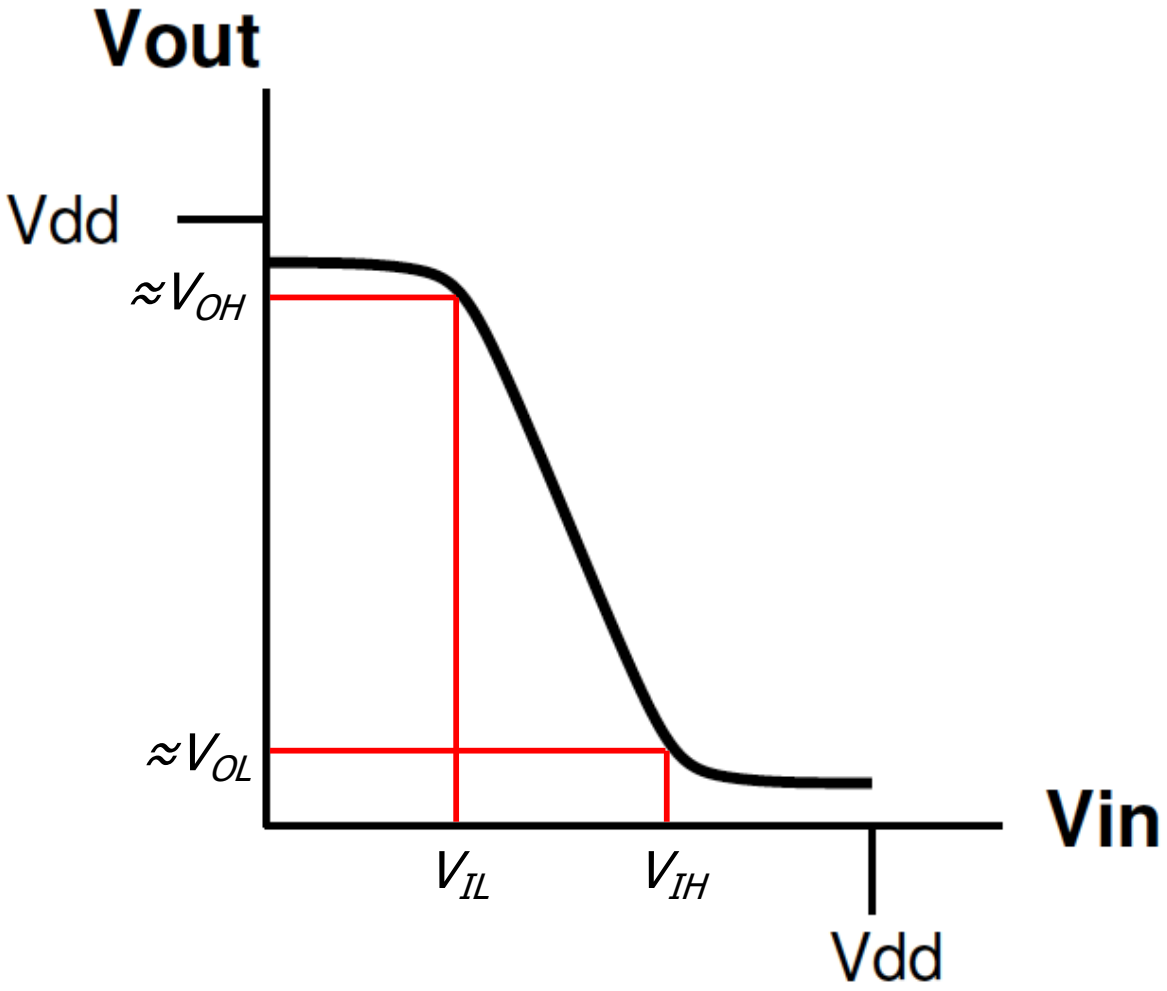
Regenerating Transfer Function

- Use gain (i.e. slope) to define noise margins



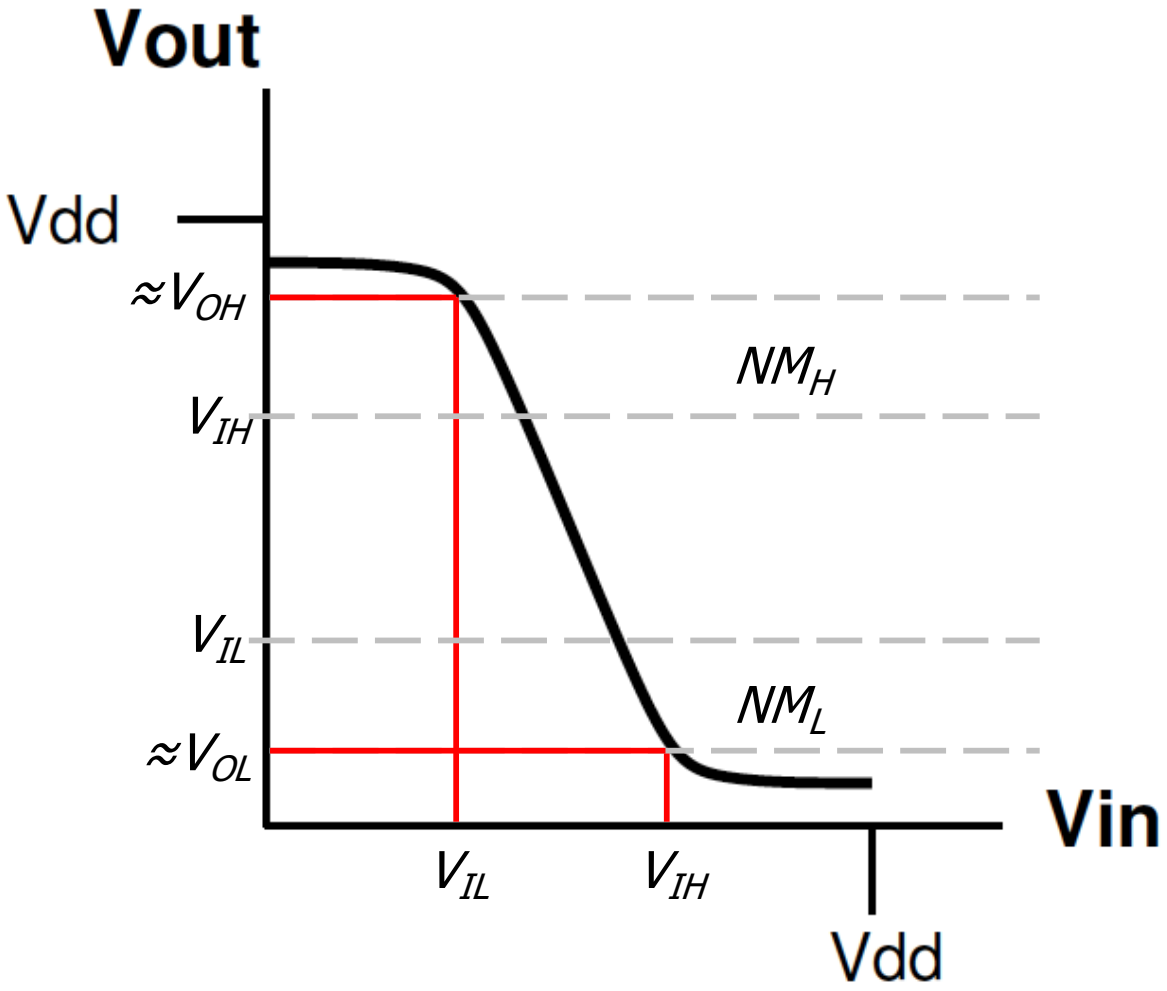
Regenerating Transfer Function

- Use gain (i.e. slope) to define noise margins



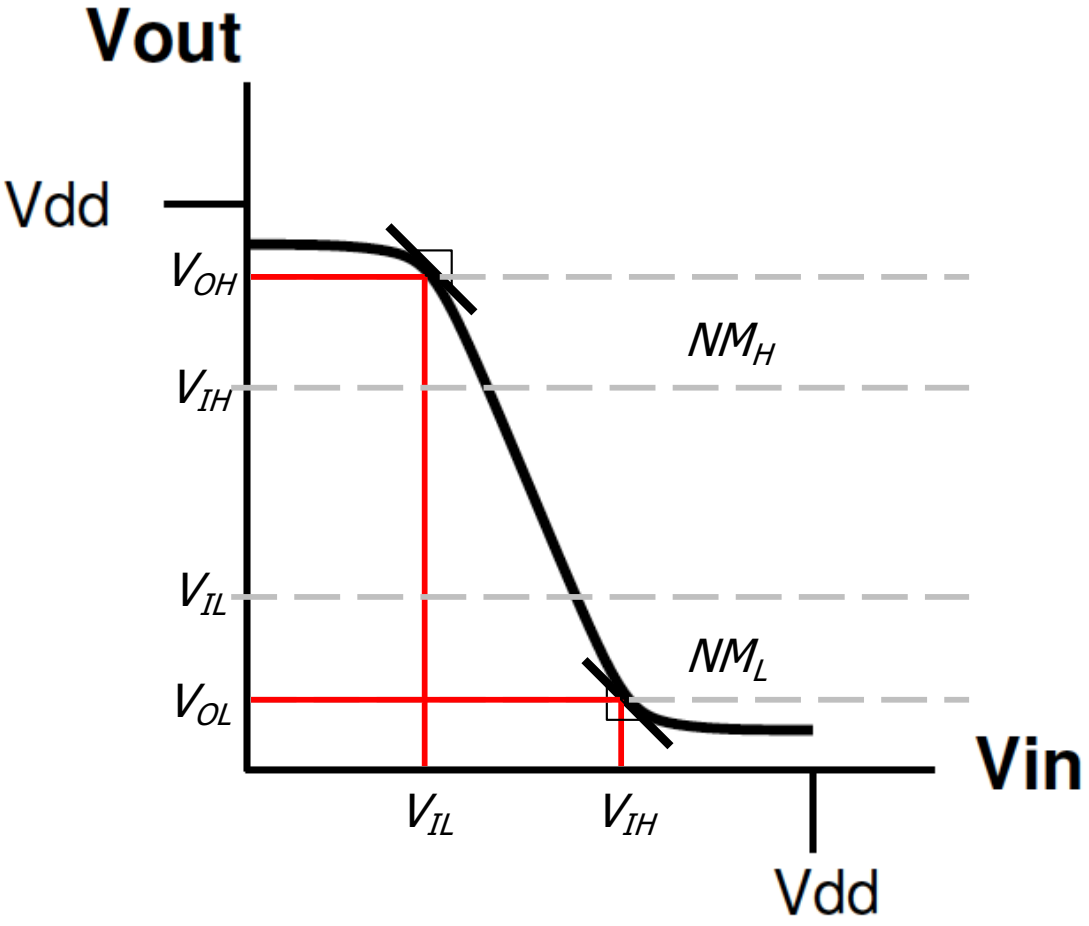
Regenerating Transfer Function

- Use gain (i.e. slope) to define noise margins



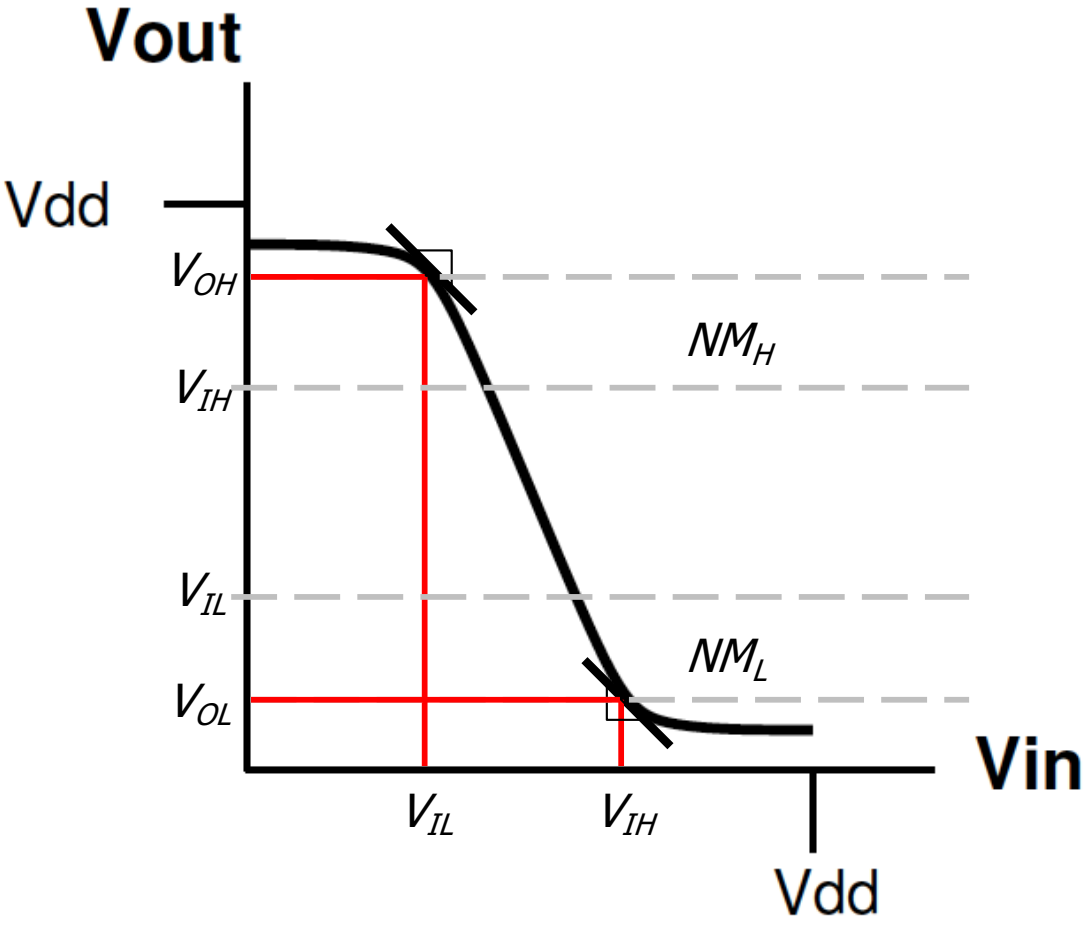
Decomposing

- What is gain?
  - |ΔVout/ΔVin|



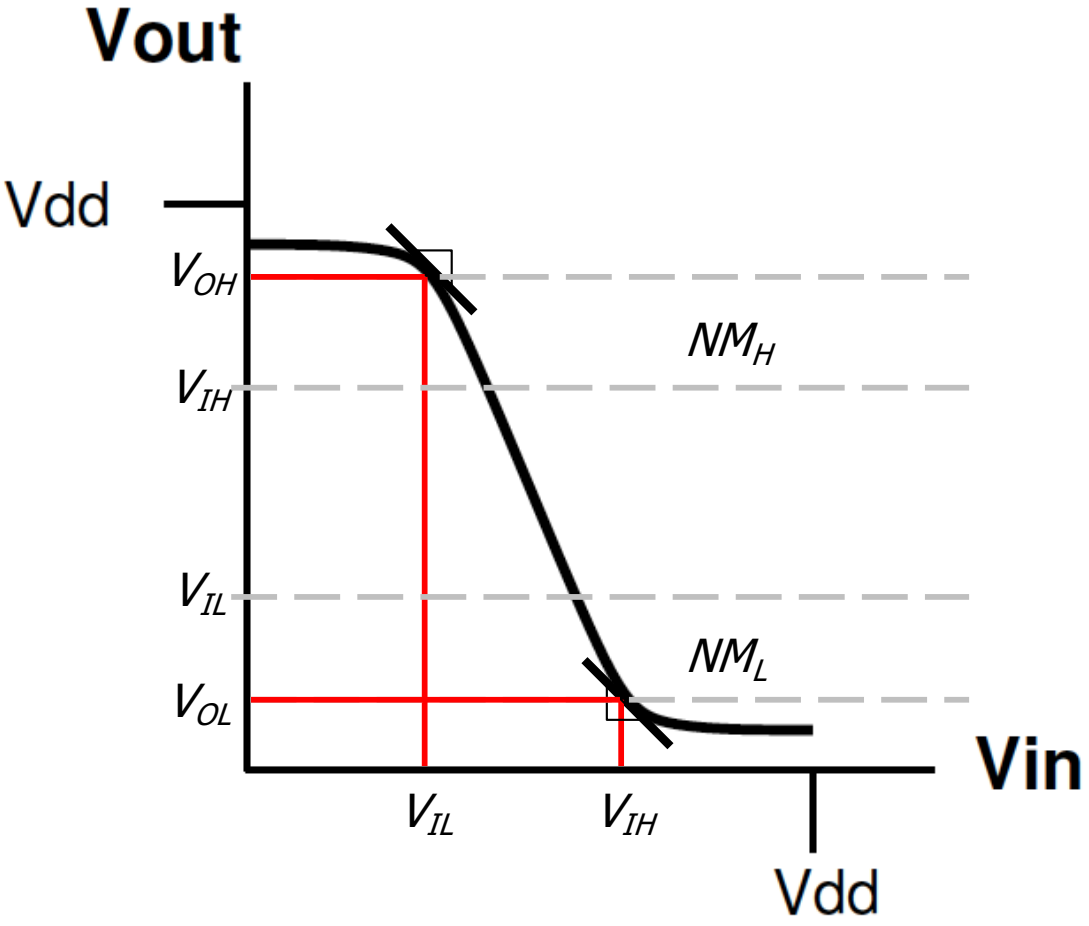
Decomposing

- What is gain?
  - |ΔVout/ΔVin|
- Where is there high gain?
  - |ΔVout/ΔVin| > 1



Decomposing

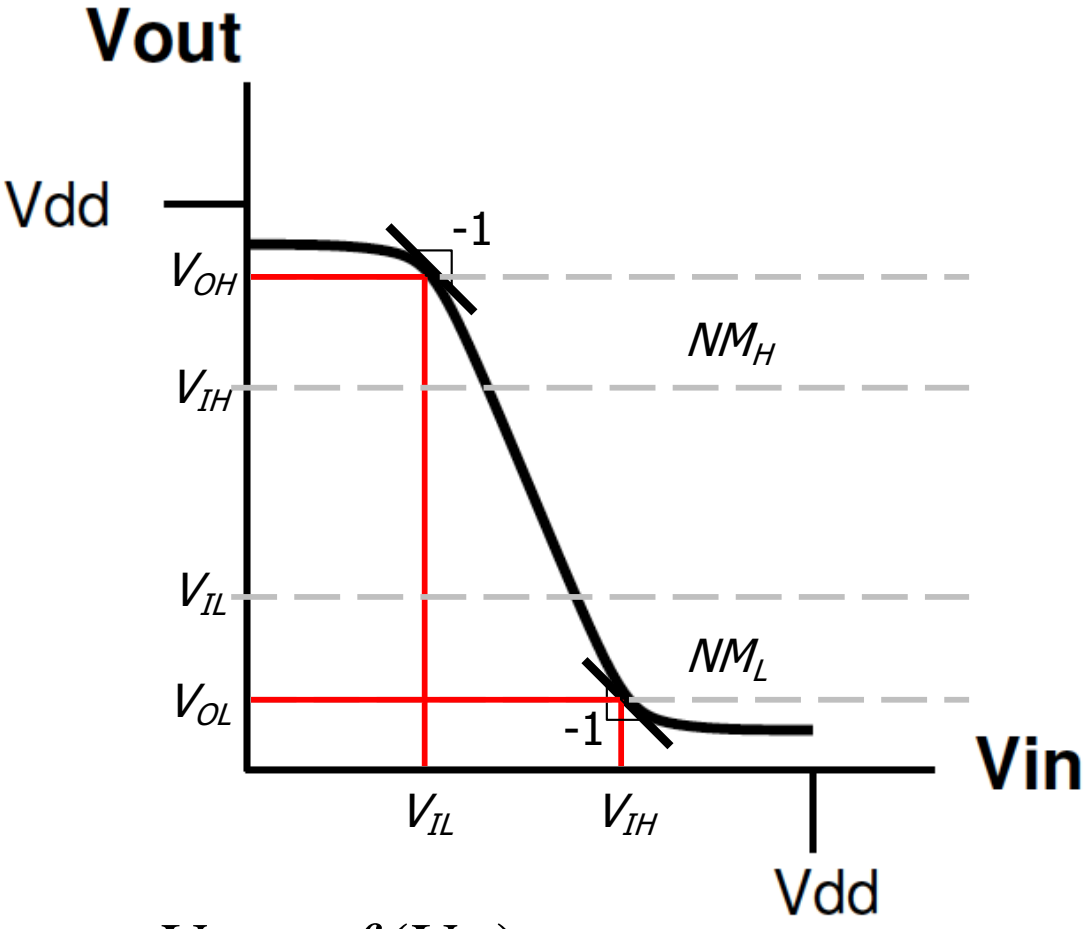
- What is gain?
  - |ΔVout/ΔVin|
- Where is there high gain?
  - |ΔVout/ΔVin| > 1
- Where is there low gain?
  - |ΔVout/ΔVin| < 1



Decomposing

- What is gain?
  - |ΔVout/ΔVin|
- Where is there high gain?
  - |ΔVout/ΔVin| > 1
- Where is there low gain?
  - |ΔVout/ΔVin| < 1
- Dividing point?

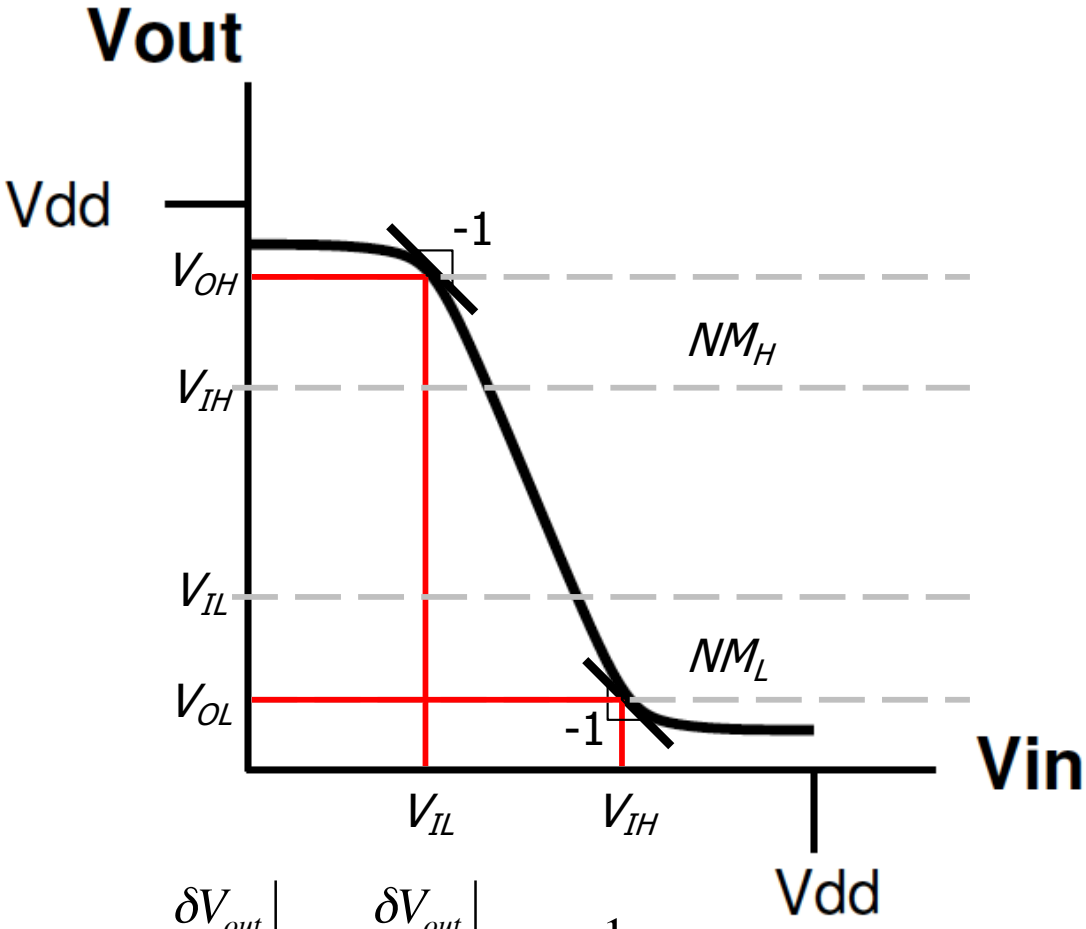
$$\left.\frac{\delta V_{out}}{\delta V_{in}}\right|_{V_{IL}} = \left.\frac{\delta V_{out}}{\delta V_{in}}\right|_{V_{IH}} = -1$$



$$V_{OH} = f(V_{IL})$$
$$V_{OL} = f(V_{IH})$$

Decomposing

- An input closer to rail than V_{IL}, V_{IH} doesn't make much difference on V_{out}
  - i.e transfer function is flat for input close to rails
- Defining V_{IL} lower (or V_{IH} higher) would reduce NMs and increase our undefined region



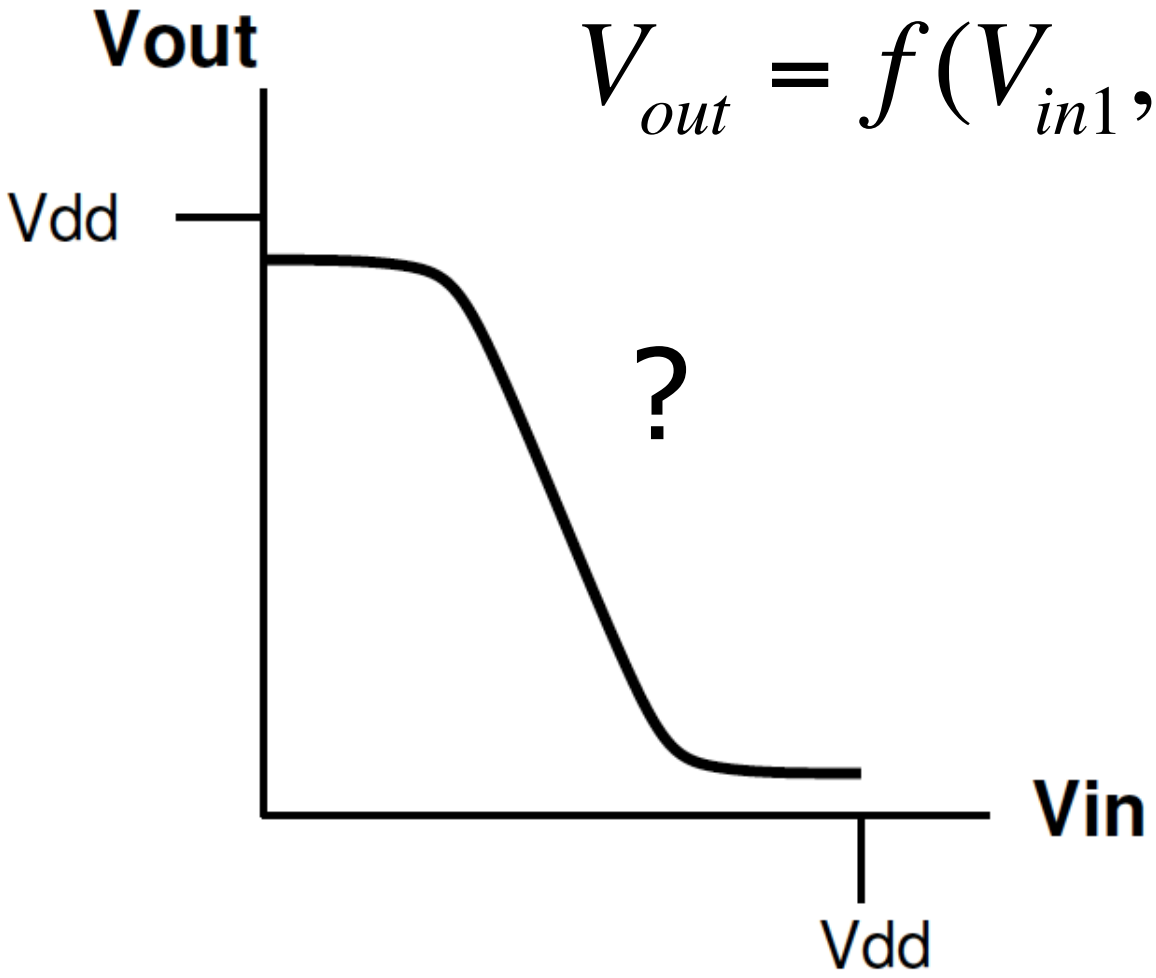
$$\left.\frac{\delta V_{out}}{\delta V_{in}}\right|_{V_{IL}} = \left.\frac{\delta V_{out}}{\delta V_{in}}\right|_{V_{IH}} = -1$$

$$V_{OH} = f(V_{IL})$$

$$V_{OL} = f(V_{IH})$$

Transfer Function for Multiple Inputs

$$V_{out} = f(V_{in1}, V_{in2})$$



Controlling Input

- Consider a nand2 gate
  - If want A to control the output
  - What value should B be?
- We call B the **non-controlling** input since it does not determine the output

Controlling Input

- Consider a nand2 gate
  - If want A to control the output
  - What value should B be?
- We call B the **non-controlling** input since it does not determine the output

A	B	NAND
0	0	1
0	1	1
1	0	1
1	1	0

Controlling Input

- Consider a nand2 gate
  - If want A to control the output
  - What value should B be?
- We call B the **non-controlling** input since it does not determine the output
- What should the non-controlling input value be for a nor2 gate?

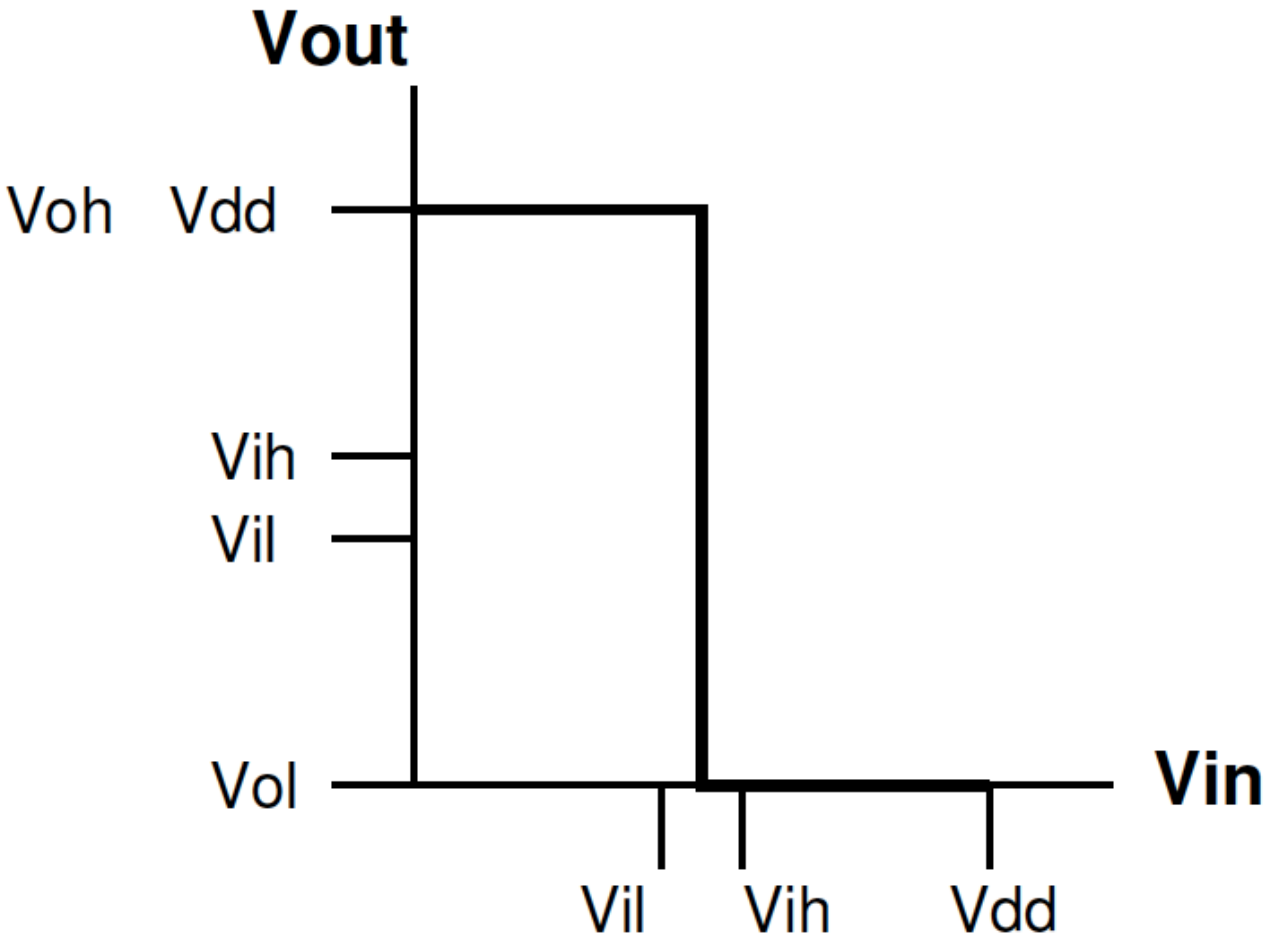
A	B	NOR
0	0	1
0	1	0
1	0	0
1	1	0

A	B	NAND
0	0	1
0	1	1
1	0	1
1	1	0

Controlling Input for Worst Case

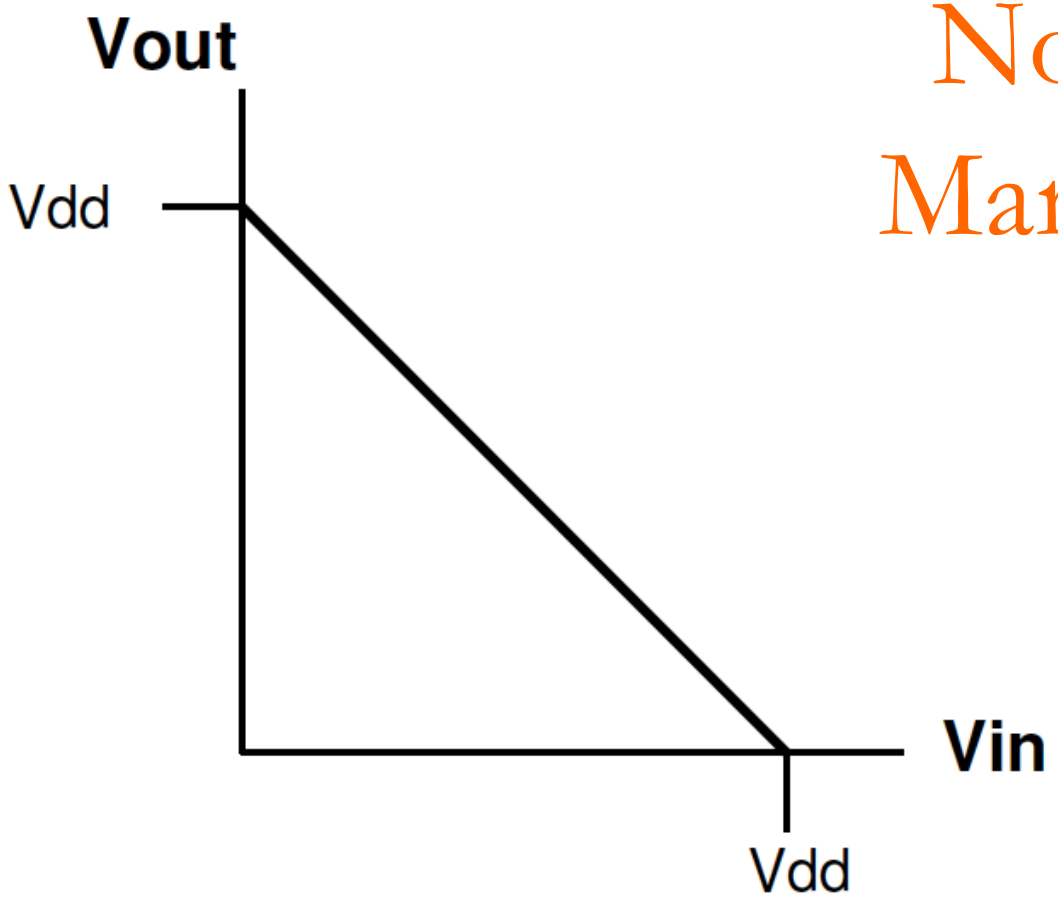
- Consider a nor2/nand2 gate
  - If want A to control the output
  - What value should B be?

Ideal Transfer Function for Inverter



Linear Transfer Function?

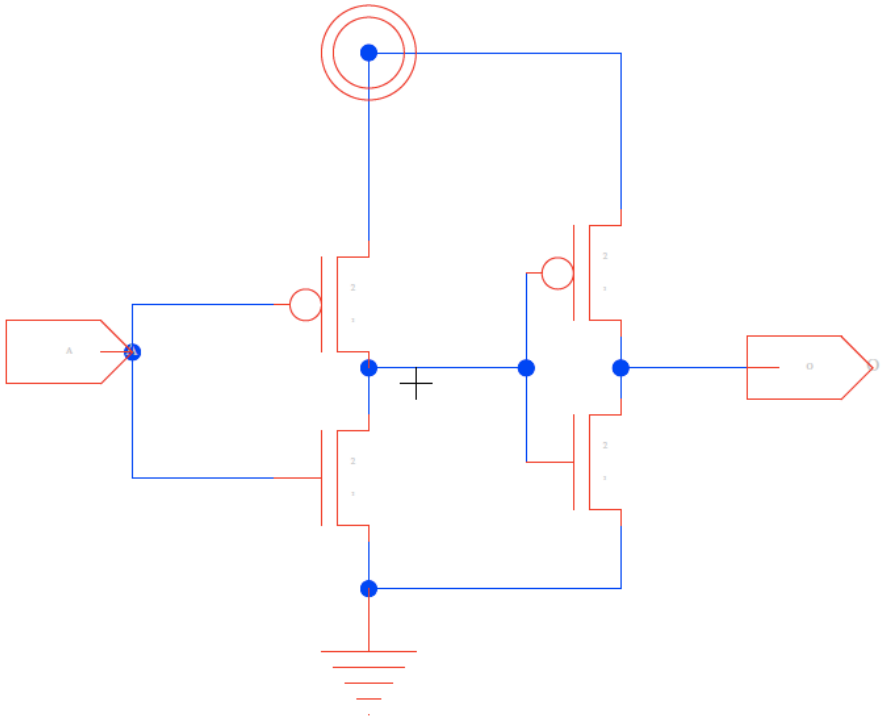
- $V_{out}=V_{dd}-V_{in}$

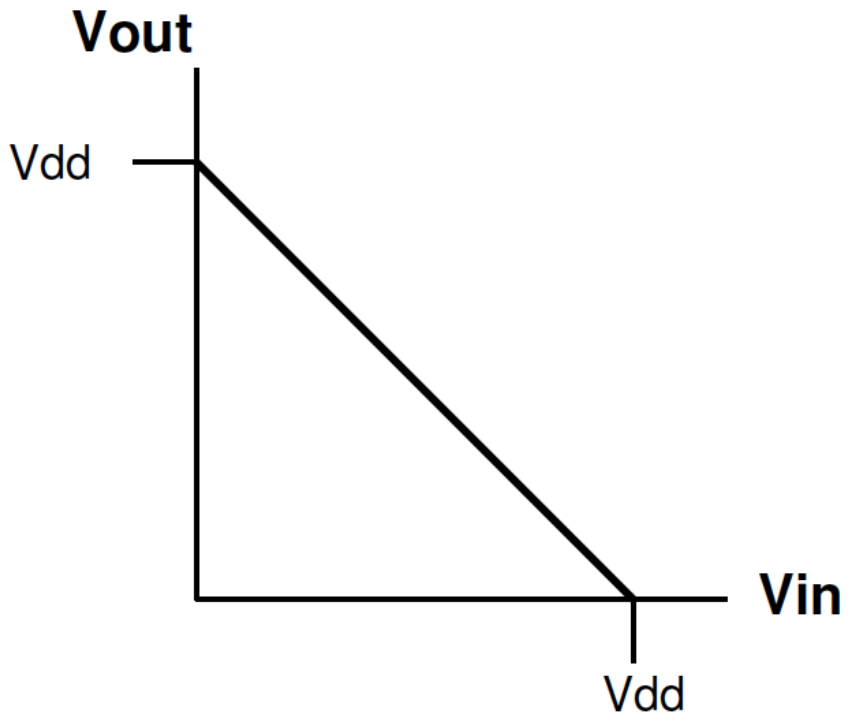


Noise Margin?

Linear Transfer Function?

- Consider two in a row (buffer)
  - $V_{out1}=V_{dd}-V_{in1}$
- What is transfer function to buffer output V_{out2}?
  - $V_{out2}=V_{dd}-V_{out1} = V_{dd}-(V_{dd}-V_{in1})=V_{in1}$



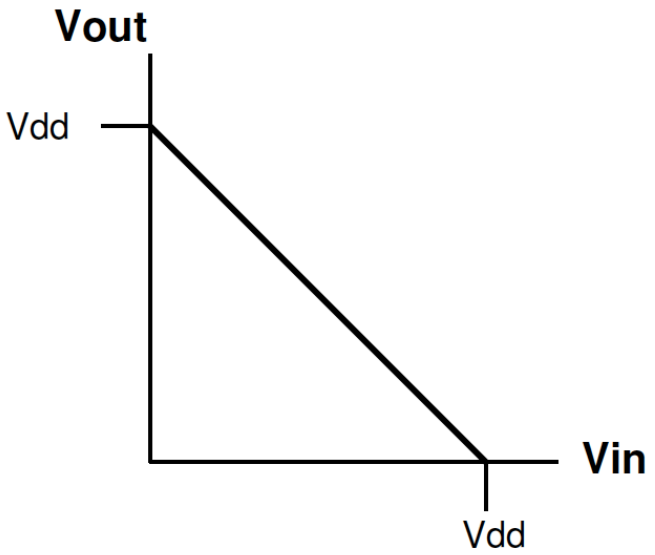


Linear Transfer Function?

- For buffer: $V_{out2}=V_{in1}$
- Consider a chain of buffers
- What happens if V_{in1} drops Δ volts between each buffer?

$$A_{i+1} = A_i - \Delta$$

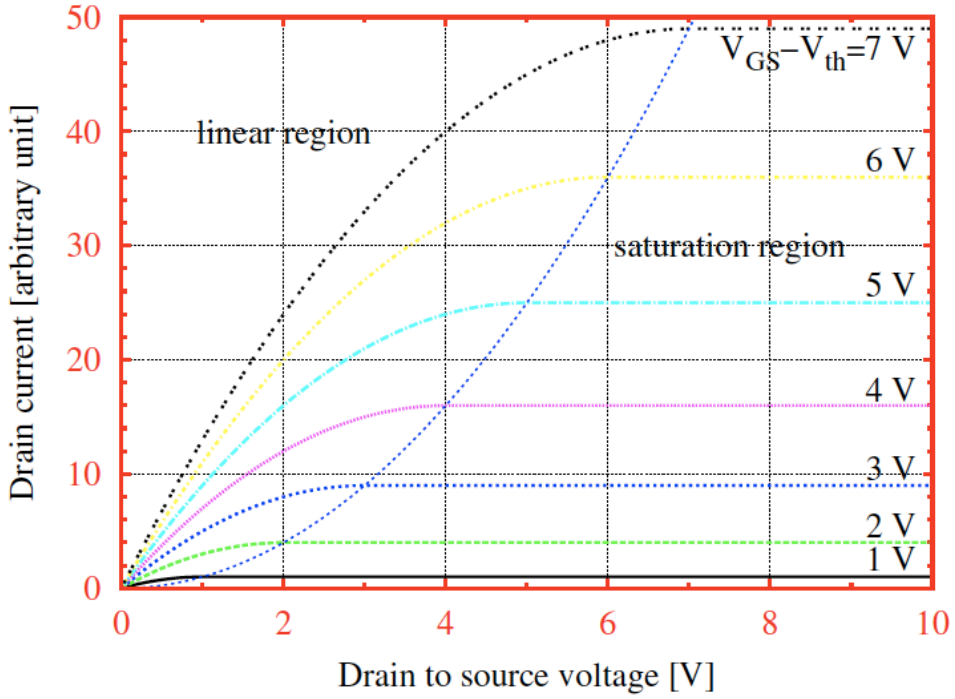
Conclude: Linear transfer functions do not provide restoration.

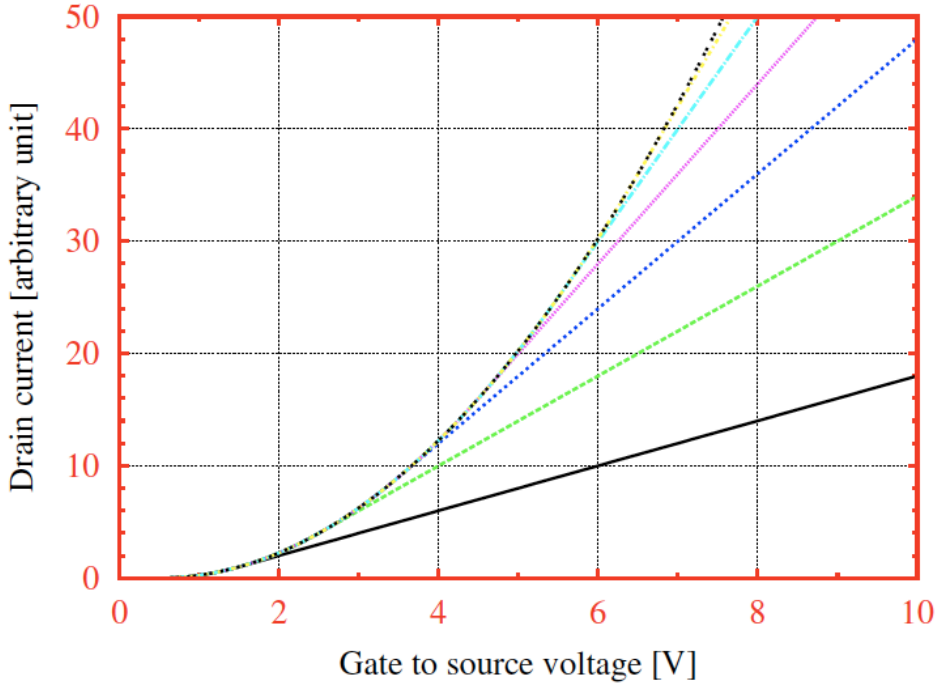


Non-linearity

- **Need** non-linearity in transfer function
- Could not have built restoring gates with R, L, C circuit
  - R, L, C are all linear elements

Transistor Non-Linearity





All Gates

- If we hope to assemble design from collection of gates,
  - Voltage levels must be consistent and supported across all gates
  - Must adhere to a V_{IL}, V_{IH}, V_{OL}, V_{OH} that is valid across entire gate set of digital circuit

$$V_{ol} = \underset{g \in G}{MAX}\left(g.V_{ol}\right) \qquad V_{il} = \underset{g \in G}{MIN}\left(g.V_{il}\right)$$

$$V_{oh} = \underset{g \in G}{MIN}\left(g.V_{oh}\right) \qquad V_{ih} = \underset{g \in G}{MAX}\left(g.V_{ih}\right)$$

Big Idea

- Need robust logic
  - Can design into any (feed forward) graph with logic gates and tolerate loss and noise, while maintaining digital abstraction
- Regeneration and noise margins
  - Every gate makes signal "better"
  - Design level of noise tolerance