EE141-Spring 2012 Digital Integrated Circuits

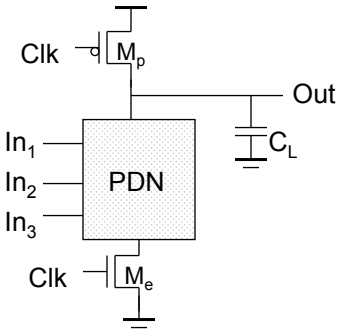
Lecture 18
Dynamic Logic

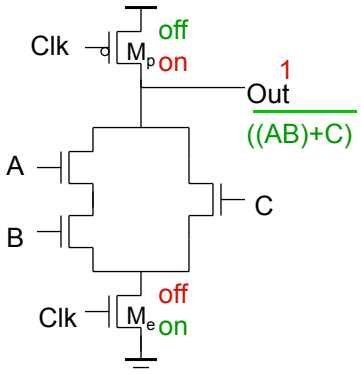
Administrativia

- Midterm 2 Graded!
- Project Phase 1 discussion
- Phase 2 Launch - Energy

Dynamic Logic

Dynamic Gate





Two phase operation

Precharge (Clk = 0)

Evaluate (Clk = 1)

tlh=zero, the result has been stored in CL. If we want to get one, we do not need to do anything. If we want to get zero, we discharge the CL.

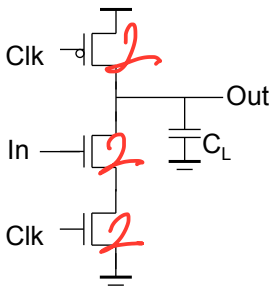
Conditions on Output

- Once the output of a dynamic gate is discharged, it cannot be charged again until the next precharge operation.
- Inputs to the gate can make at most one transition during evaluation.
- Output can be in the high impedance state during and after evaluation (PDN off), state is stored on C_L

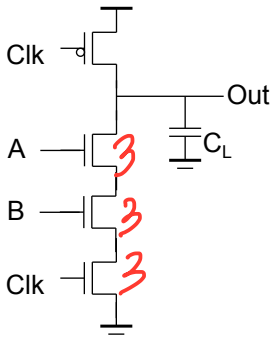
Properties of Dynamic Gates

- Logic function is implemented by the PDN only
  - number of transistors is N + 2 (versus 2N for static complementary CMOS)
- Full swing outputs (V_{OL} = GND and V_{OH} = V_{DD})
- Non-ratioed - sizing of the devices does not affect the logic levels
- Faster switching speeds
  - reduced capacitance due to lower input capacitance (C_{in})
  - no I_{sc}, so all the current provided by PDN goes into discharging C_L ???

LE of Dynamic Gates



C_{gate} = 2
LE = 2/3



C_{gate} = 3
LE = 1

EECS141 Lecture #18 7

Previously, most chips used dynamic logics. But now, we only use a few of dynamic logic. The main reason is leakage.

Properties of Dynamic Gates

- Overall power dissipation usually higher than static CMOS
  - no static current path ever exists between V_{DD} and GND (including P_{sc})
  - no glitching
  - higher transition probabilities
  - extra load on Clk
- PDN starts to work as soon as the input signals exceed V_{Tn}, so V_M, V_{IH} and V_{IL} equal to V_{Tn}
  - low noise margin (NM_L)
- Needs a precharge/evaluate clock

EECS141 Lecture #18 8

CMOS $CV_{DD}^2 \cdot P_{0\to1} f$ $P_{0\to1} = P_0 \cdot P_1$
Dynamic $CV_{DD}^2 P_0 \cdot f$

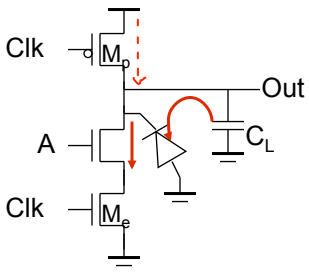
Dynamic logic not allow glitch which will cause the voltage of CL down and happen in static logic.

Challenges of Dynamic Gates

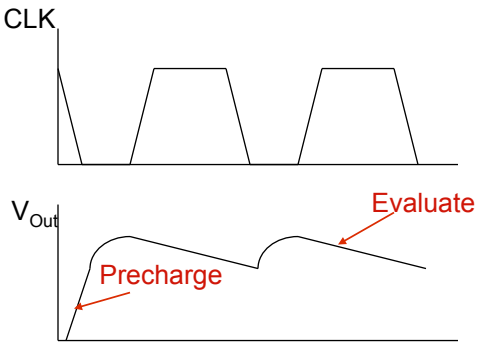
- **Noise sensitivity and small noise margins**
  - Leakage
  - Charge sharing
  - Clock feedthrough

Issues in Dynamic Design 1: Charge Leakage

Dynamic logic only for fast logic to reduce leakage.

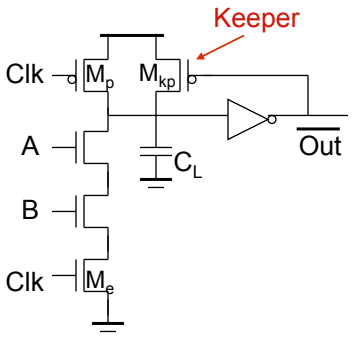


Leakage sources



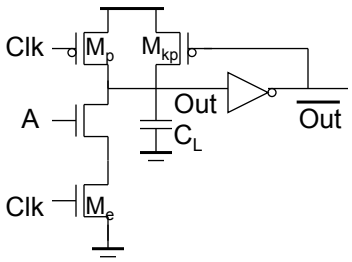
Dominant component is subthreshold current

Solution to Charge Leakage



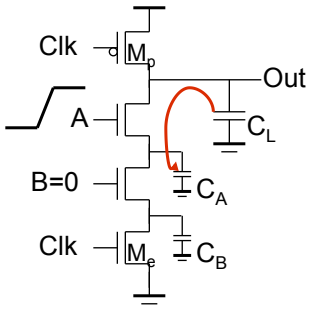
Same approach as level restorer for pass-transistor logic

Dynamic Gate VTC



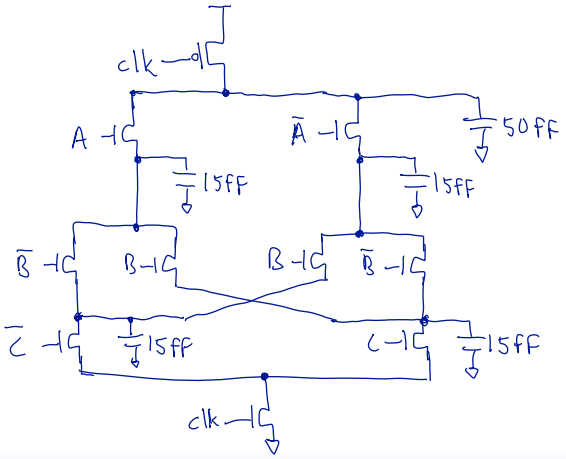
Issues in Dynamic Design 2: Charge Sharing

- Charge initially stored on C_L
  - C_A previously discharged

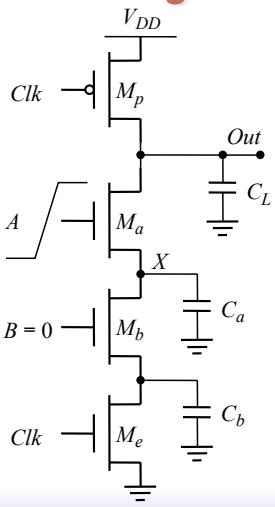


- When A rises, this charge is redistributed (shared) between C_L and C_A
- Makes Out drop below V_{DD}

Charge Sharing Example



Charge Sharing



- Two cases:
  - M_a stays on – complete charge share
  - M_a turns off – incomplete charge share

•Complete charge share:

- $Q_{Ca} = V_{Out}C_a$
  $\Delta Q_{CL} = -V_{Out}C_a$

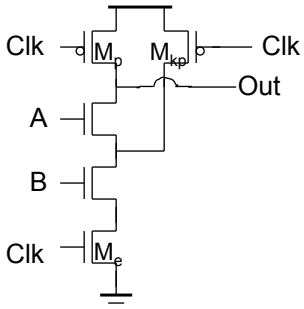
→ $\boxed{\Delta V_{Out} = -V_{DD}C_a/(C_a+C_L)}$

•Incomplete charge share:

- $Q_{Ca} = (V_{DD}-V_{TN}^*)C_a$
  $\Delta Q_{CL} = -(V_{DD}-V_{TN}^*)C_a$

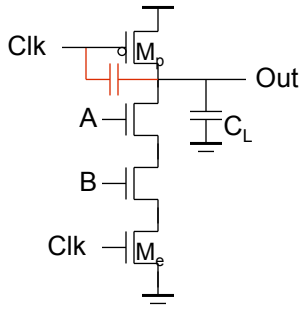
→ $\boxed{\Delta V_{Out} = -(V_{DD}-V_{TN}^*)C_a/C_L}$

Solution to Charge Sharing



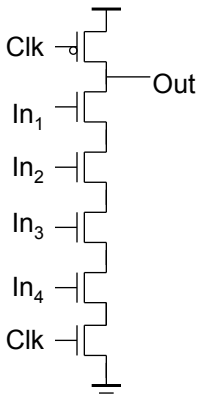
- Keeper helps a lot
  - Can still get failures if Out drops below inverter V_{sw}
- Another option: precharge internal nodes
  - Increases power and area

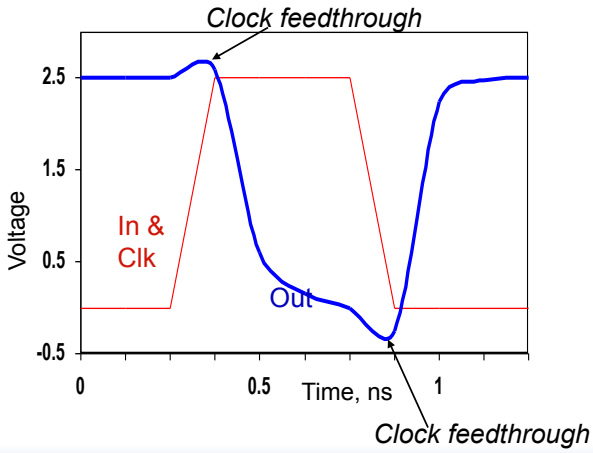
Issues in Dynamic Design 3: Clock Feedthrough



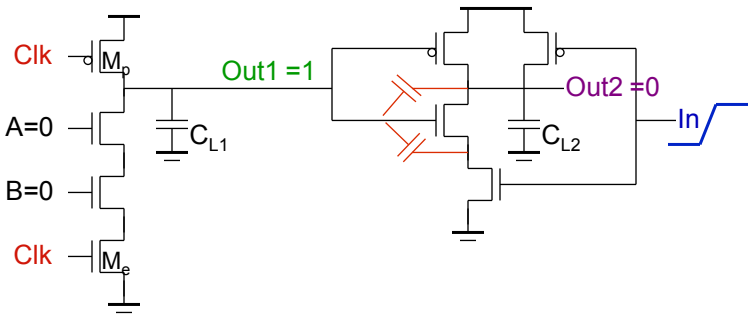
Coupling between Out and Clk input of the precharge device due to the gate to drain capacitance. So voltage of Out can rise above V_{DD}. The fast rising (and falling edges) of the clock couple to Out.

Clock Feedthrough



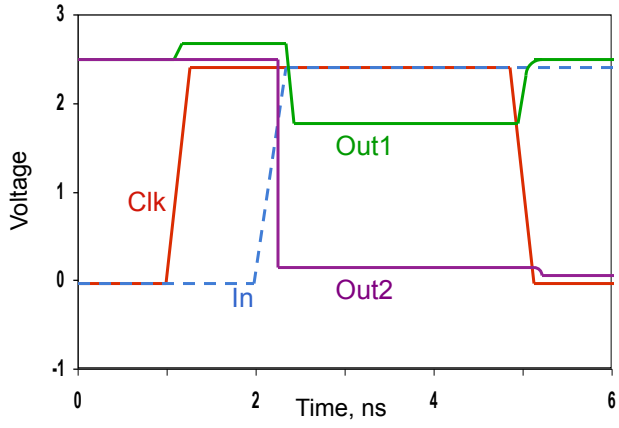


Issues in Dynamic Design 4: Backgate Coupling



Dynamic NAND Static NAND

Backgate Coupling Effect

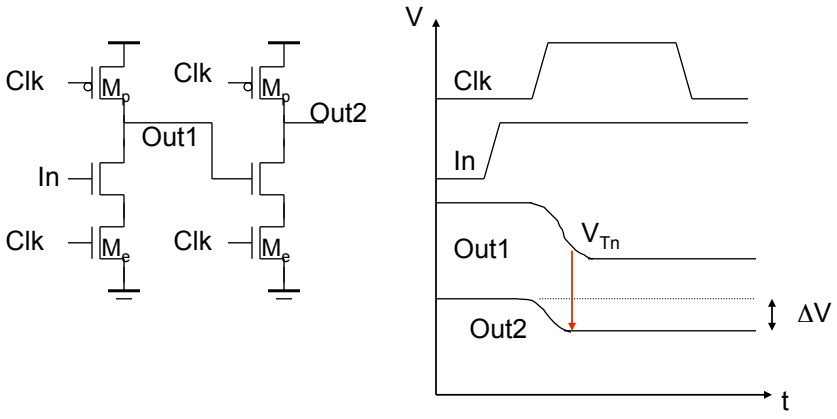


Other Effects

- Capacitive coupling
- Substrate coupling
- Minority charge injection
- Supply noise (ground bounce)

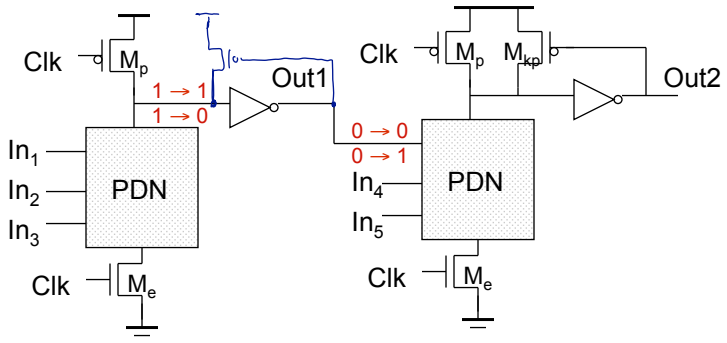
Domino Logic

Cascading Dynamic Gates

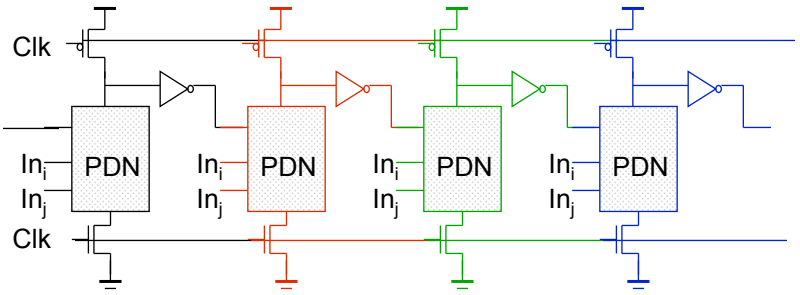


Only 0 → 1 transitions allowed at inputs!

Domino Logic



Why Named Domino?

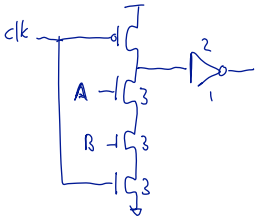


Like falling dominos!

Properties of Domino Logic

- Only non-inverting logic can be implemented
- Very high speed
  - static inverter can be skewed, only L-H transition critical
  - Input capacitance reduced – smaller logical effort

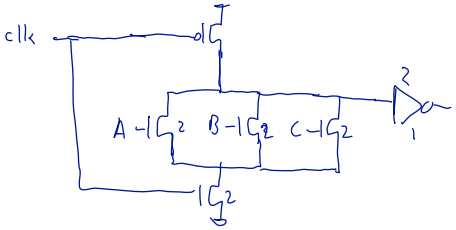
Domino Logic LE



$LE_{dyn} =$

$LE_{inv} = 1$

$\pi LE =$



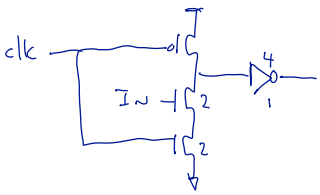
$LE_{dyn} =$

$LE_{inv} = 1$

$\pi LE =$

$LE_{static} = \frac{7}{3}$

Domino Logic LE (skewed static gate)

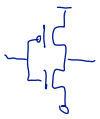


$LE_{dyn} =$

$LE_{inv} =$

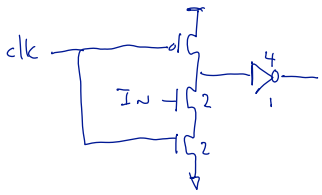
$\pi LE =$

reference inverter:



Buffer "Average" LE

Domino buffer:



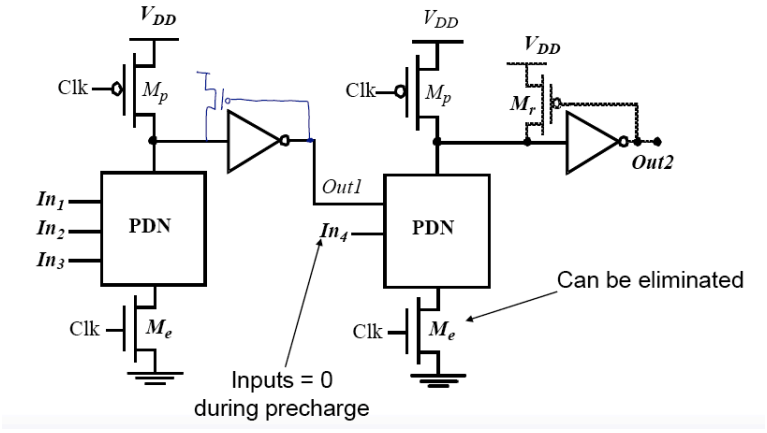
$LE_{dinv} = \frac{2}{3}$

$LE_{sinv} = \frac{5}{6}$

$\Pi LE = \frac{10}{18}$

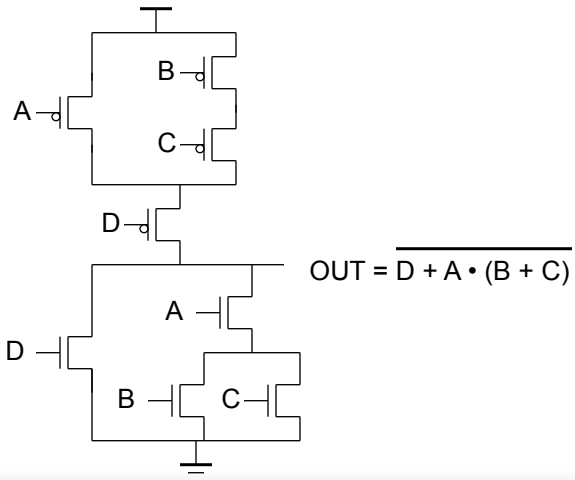
"Average" $LE = \sqrt{10/18} \approx \frac{3}{4}$

Designing with Domino Logic



CMOS Layout

Complex CMOS Gate

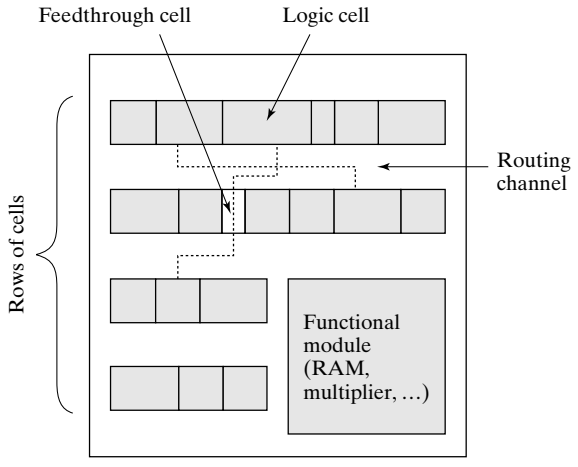


$$OUT = \overline{D + A \cdot (B + C)}$$

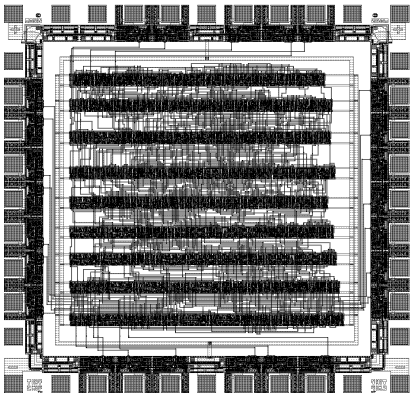
Cell Design

- Standard Cells
  - General purpose logic
  - Used to synthesize RTL/HDL
  - Same height, varying width
- Datapath Cells
  - For regular, structured designs (arithmetic)
  - Includes some wiring in the cell

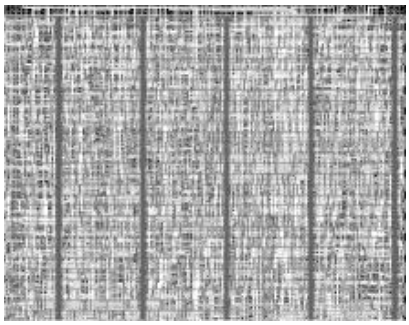
Standard Cell Methodology



Standard Cells – Then and Now

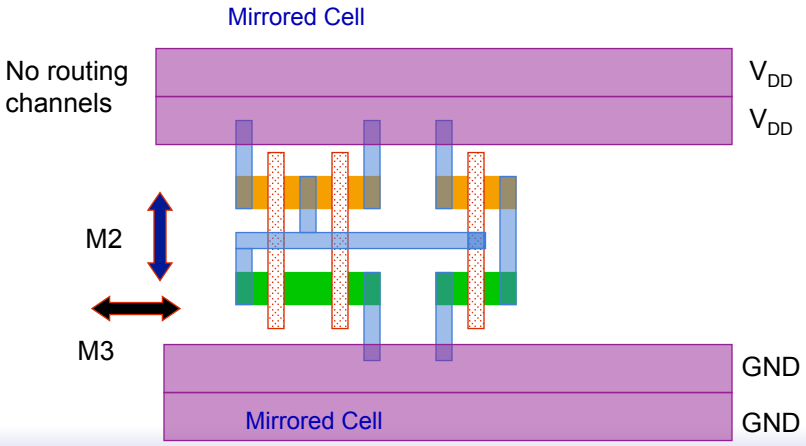


(a)

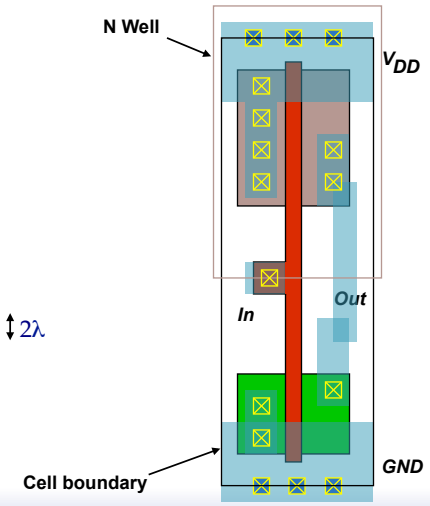


(b)

Standard Cell Layout Methodology



Standard Cells



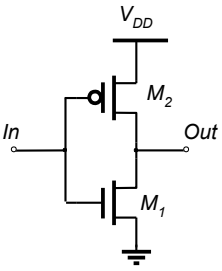
Cell height 12 metal tracks
Metal track is approx. 3λ + 3λ
Pitch =
repetitive distance between objects

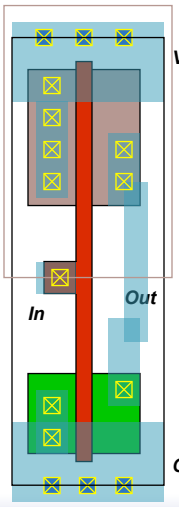
Cell height is "12 Mn pitch"

Rails ~10λ

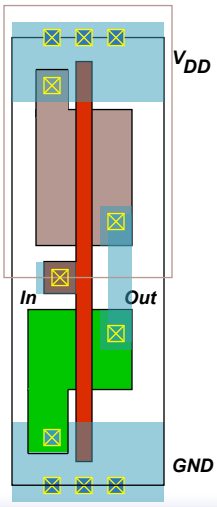
Standard Cells

With minimal diffusion routing

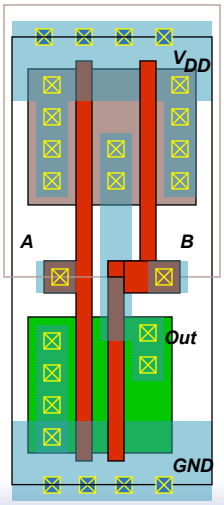




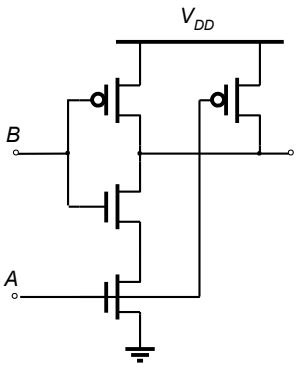
With silicided diffusion



Standard Cells



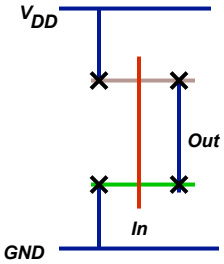
2-input NAND gate



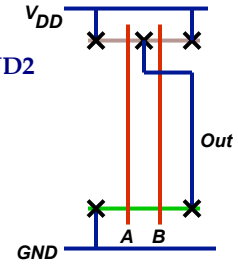
Stick Diagrams

Contains no dimensions
Represents relative positions of transistors

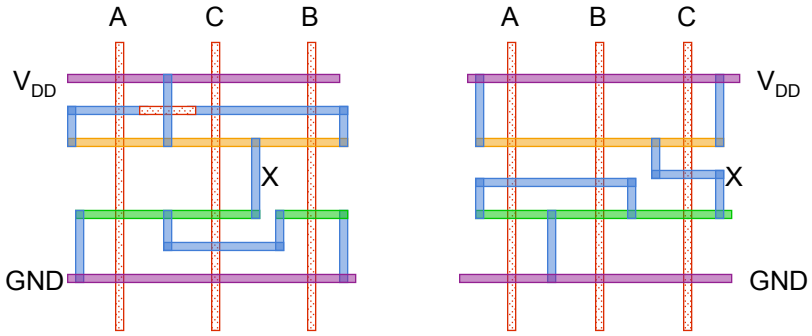
Inverter



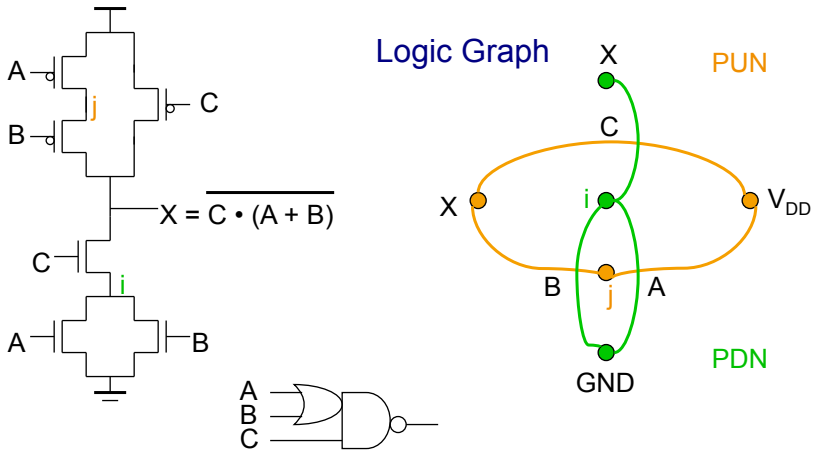
NAND2



Two Versions of C • (A + B)



Logic Graphs



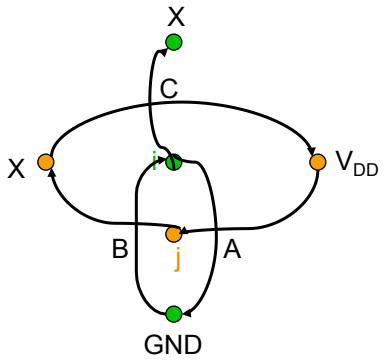
Consistent Euler Path

A B C

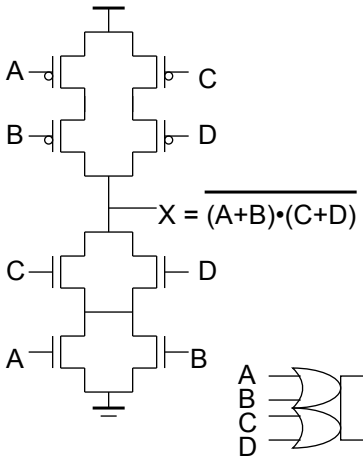
Has PDN and PUN

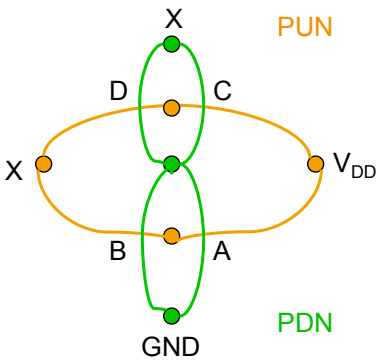
B C A

Has PUN, but no PDN

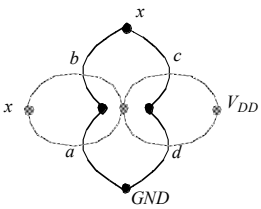


OAI22 Logic Graph

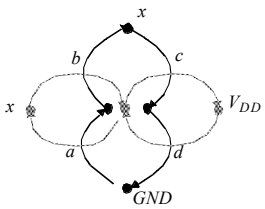




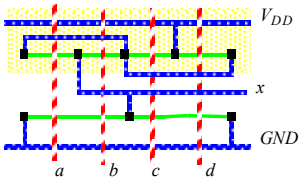
Example: x = ab+cd



(a) Logic graphs for $\overline{(ab+cd)}$



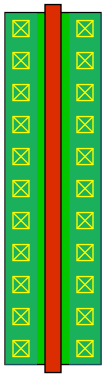
(b) Euler Paths {*a b c d*}



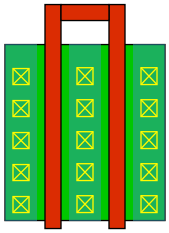
(c) stick diagram for ordering {a *b c d*}

Multi-Fingered Transistors

One finger



Two fingers (folded)



Less diffusion capacitance