ELECTRICAL ENGINEERING
UNIVERSITY of WASHINGTON

Lecture 2: CMOS Logic

Based on material prepared by prof. Visvesh S. Sathe

ELECTRICAL & COMPUTER ENGINEERING

Acknowledgements

All class materials (lectures, assignments, etc.) based on material prepared by Prof. Visvesh S. Sathe, and reproduced with his permission

Visvesh S. Sathe
Associate Professor
Georgia Institute of Technology
https://psylab.ece.uw.edu

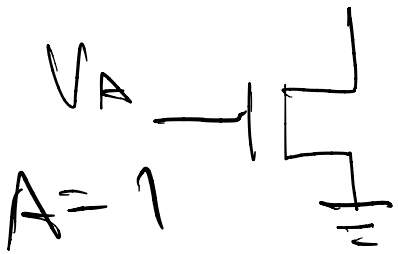
UW (2013-2022)
GaTech (2022-present)

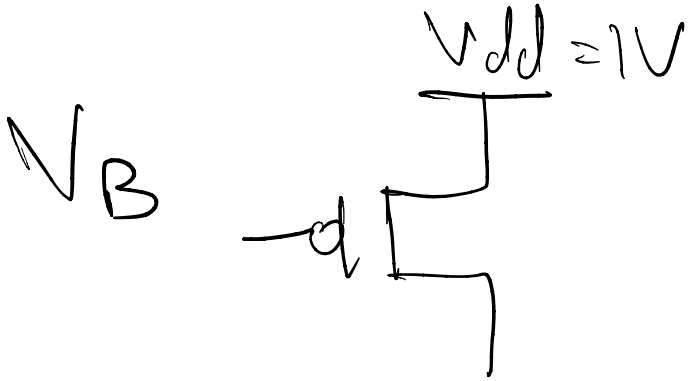
Course Announcements

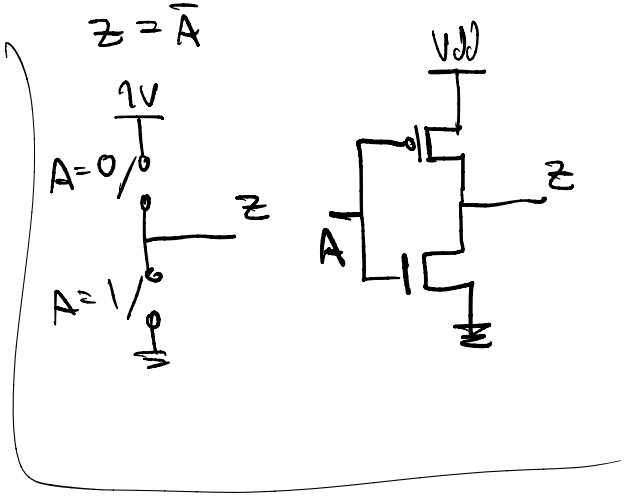
- By now, you should have
  - Make sure you can access the linux labs and have a directory for this class
  - Set up VNC connection

- Reservations for ECE 357:
  - Monday to Thursday: 2:00 pm to 5:00 pm
  - Friday: 11:30 am to 1:30 pm *(Kevin's office hours)*
  - Saturday and Sunday: 9:00 am to 1:00 pm
  - ***You're welcome to work there during any other time as well***, but there is no priority reservation.

Transistors as switches: CMOS inverter







$V_A \geq V_{thn}$

$V_A = Vdd = 1V$ on

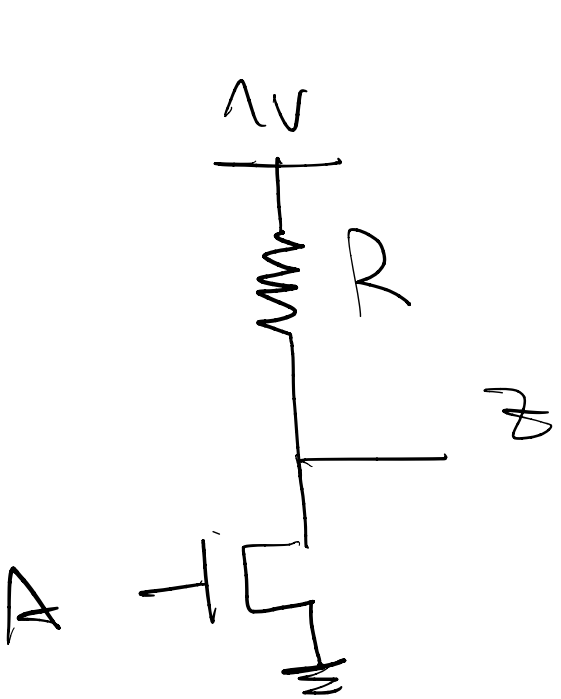
$V_A = 0$ off

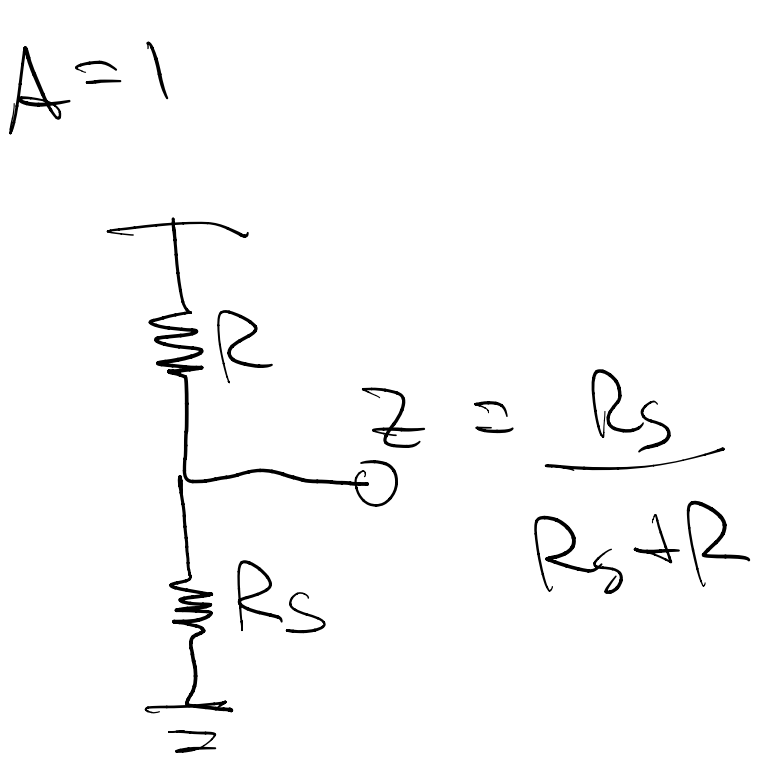
$V_B \leq Vdd - V_{thp}$ on

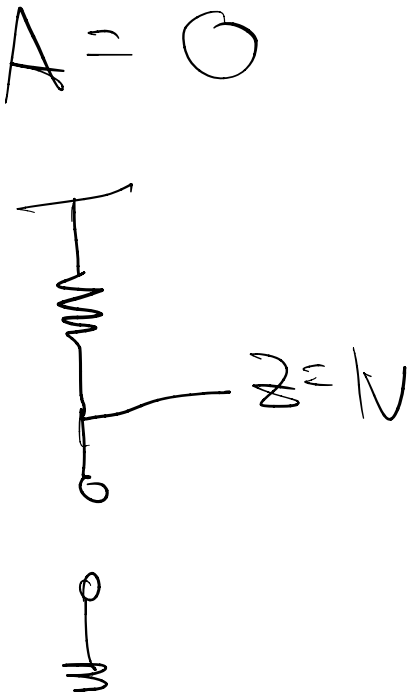
$V_B = 1V$ off

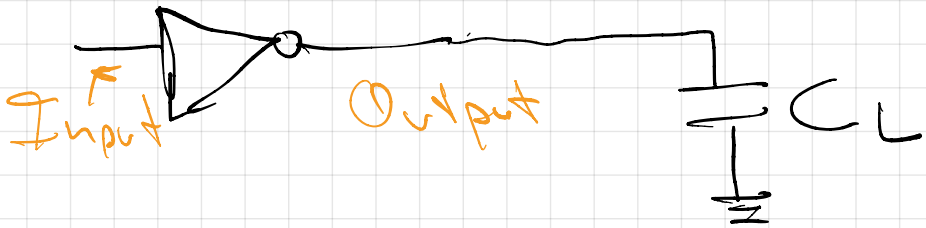
$V_B = 0V$ on

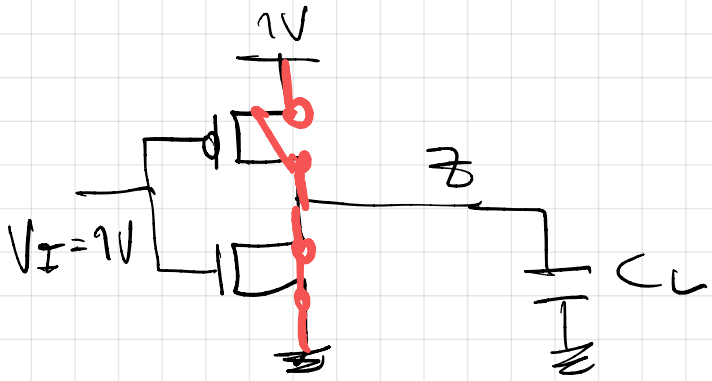
Quick aside: NMOS logic



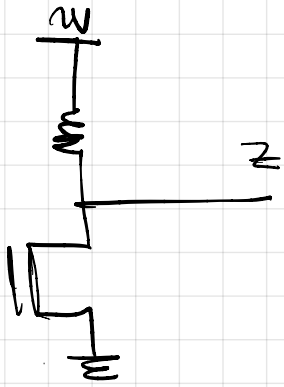






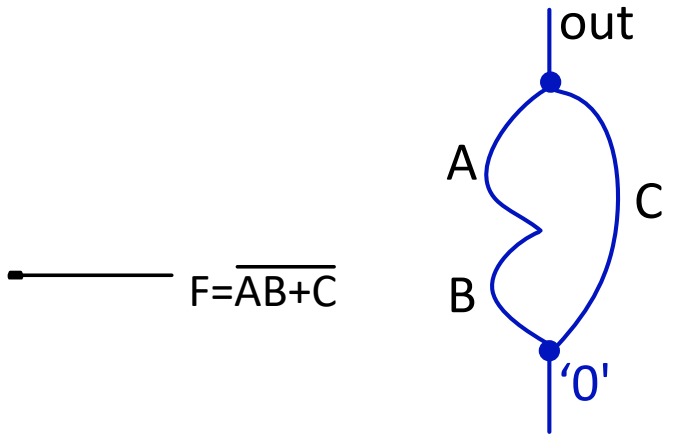


$V_Z(t=0) = 0.5V$



CMOS logic

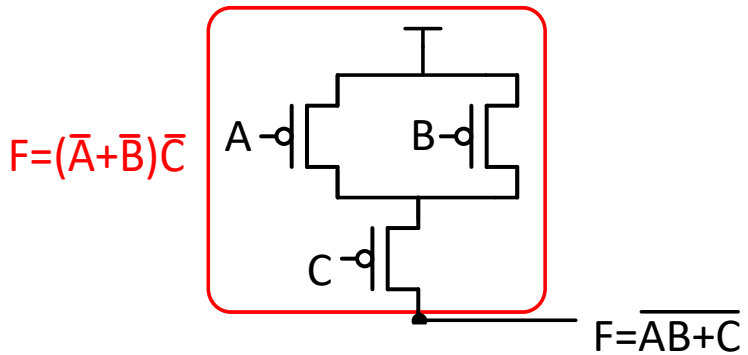
$F=(\overline{A}+\overline{B})\overline{C}$

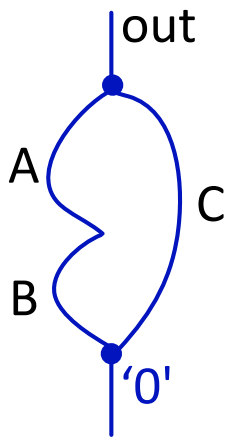


$F=\overline{AB+C}$

- Implement logic by connecting output to 1 (V_{dd}) or 0 (V_{ss})
  - Connect output Z to V_{dd} if function evaluates to '1'
  - Relay network connecting V_{dd} to Z referred to as the pull-up network (PUN)
  - Connect Z to V_{ss} if function evaluates to '0'
  - Relay network connecting V_{ss} to Z referred to as the pull-down network (PDN)
  - ➔Z will not be connected to *both* V_{dd} and V_{ss} . PUN and PDN ***complementary***
- Viewed as graphs, the networks are duals of each other

CMOS logic





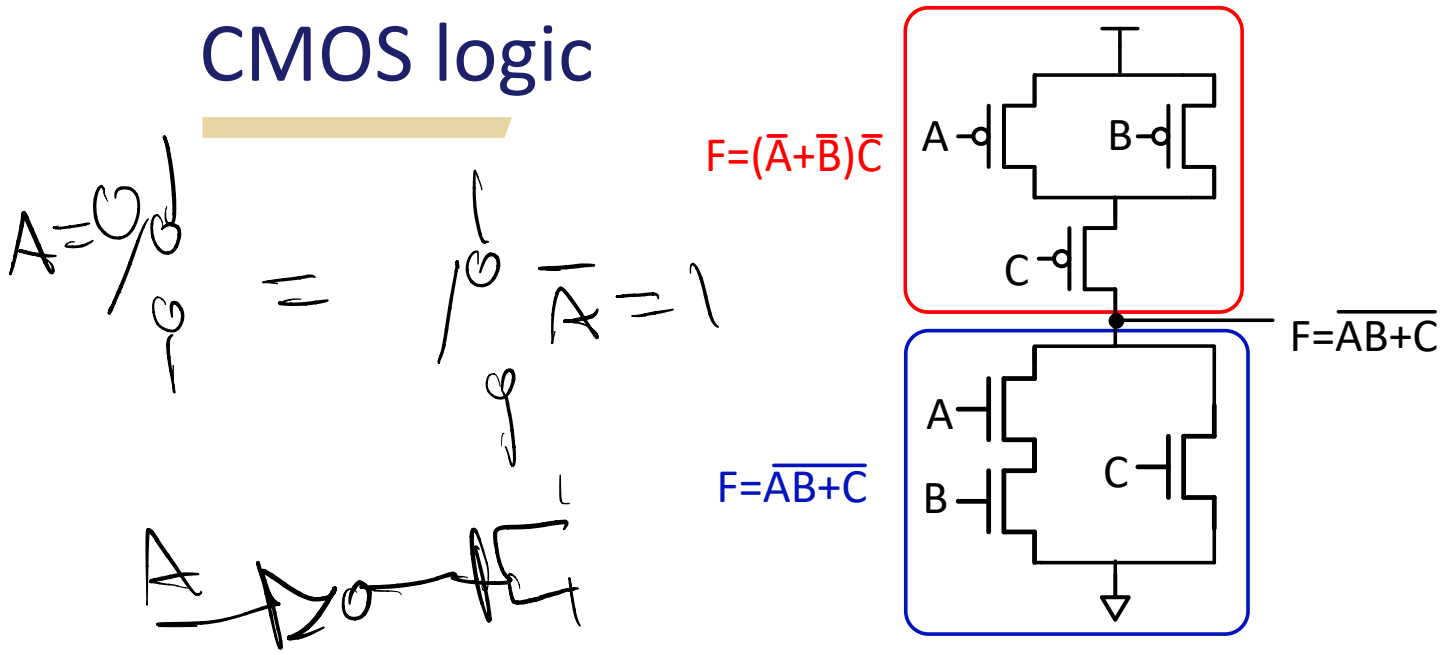
$\overline{AB+C}$

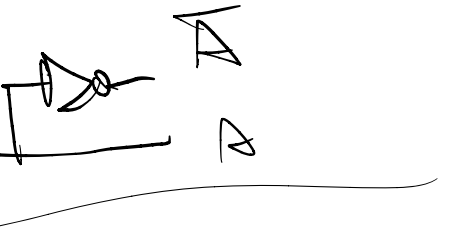
$(\bar{A}+\bar{B})\bar{C}$

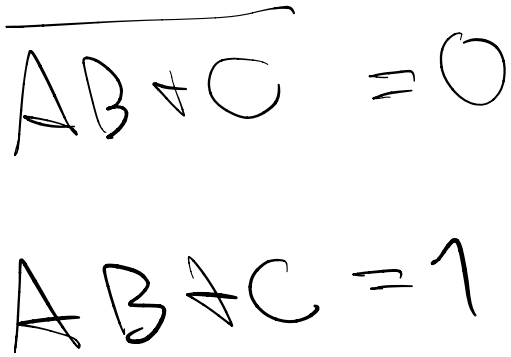
$F=\overline{AB+C}$

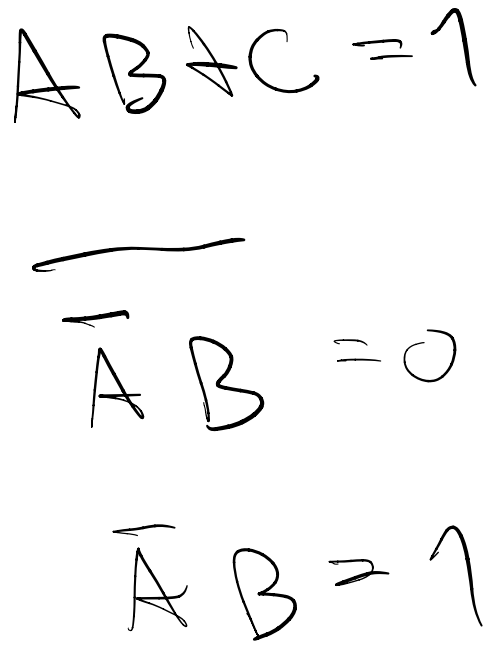
- Implement logic by connecting output to 1 (V_{dd}) or 0 (V_{ss})
  - Connect output Z to V_{dd} if function evaluates to '1'
  - Relay network connecting V_{dd} to Z referred to as the pull-up network (PUN)
  - Connect Z to V_{ss} if function evaluates to '0'
  - Relay network connecting V_{ss} to Z referred to as the pull-down network (PDN)
  - ➔Z will not be connected to *both* V_{dd} and V_{ss} . PUN and PDN ***complementary***
- Viewed as graphs, the networks are duals of each other

CMOS logic



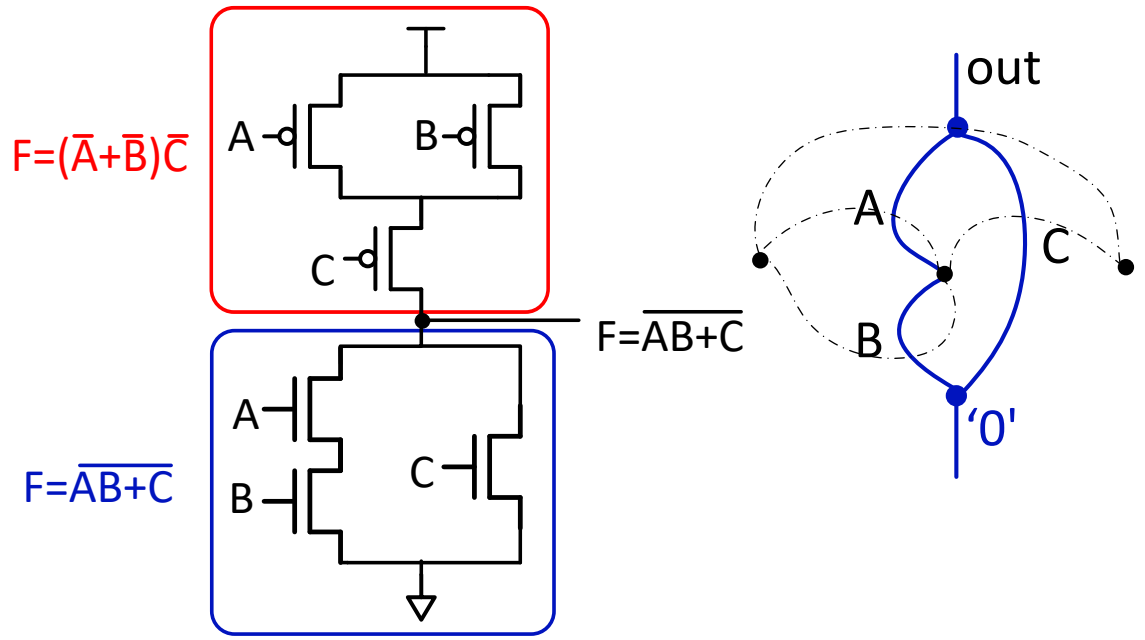






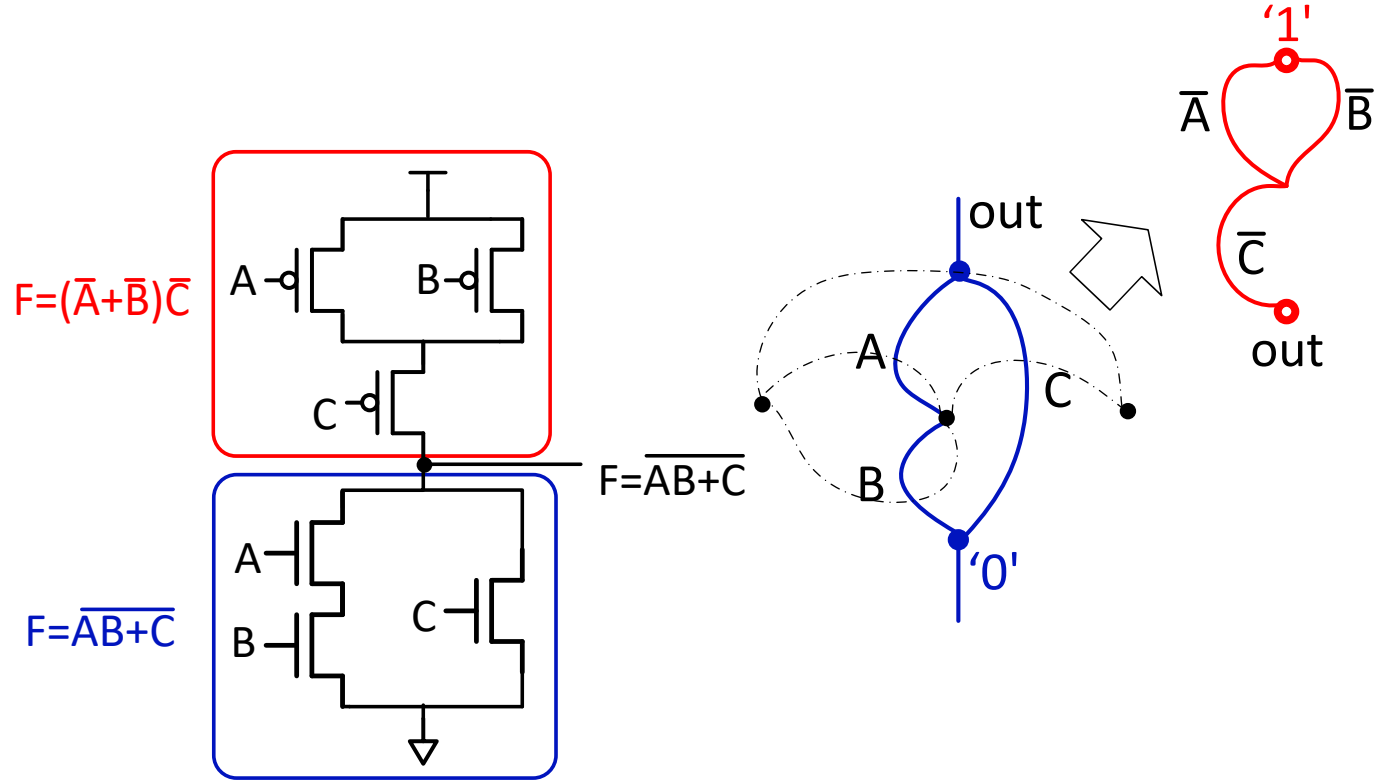
- Implement logic by connecting output to 1 (V_{dd}) or 0 (V_{ss})
  - Connect output Z to V_{dd} if function evaluates to '1'
  - Relay network connecting V_{dd} to Z referred to as the pull-up network (PUN)
  - Connect Z to V_{ss} if function evaluates to '0'
  - Relay network connecting V_{ss} to Z referred to as the pull-down network (PDN)
  - ➔Z will not be connected to *both* V_{dd} and V_{ss} . PUN and PDN ***complementary***
- Viewed as graphs, the networks are duals of each other

CMOS logic



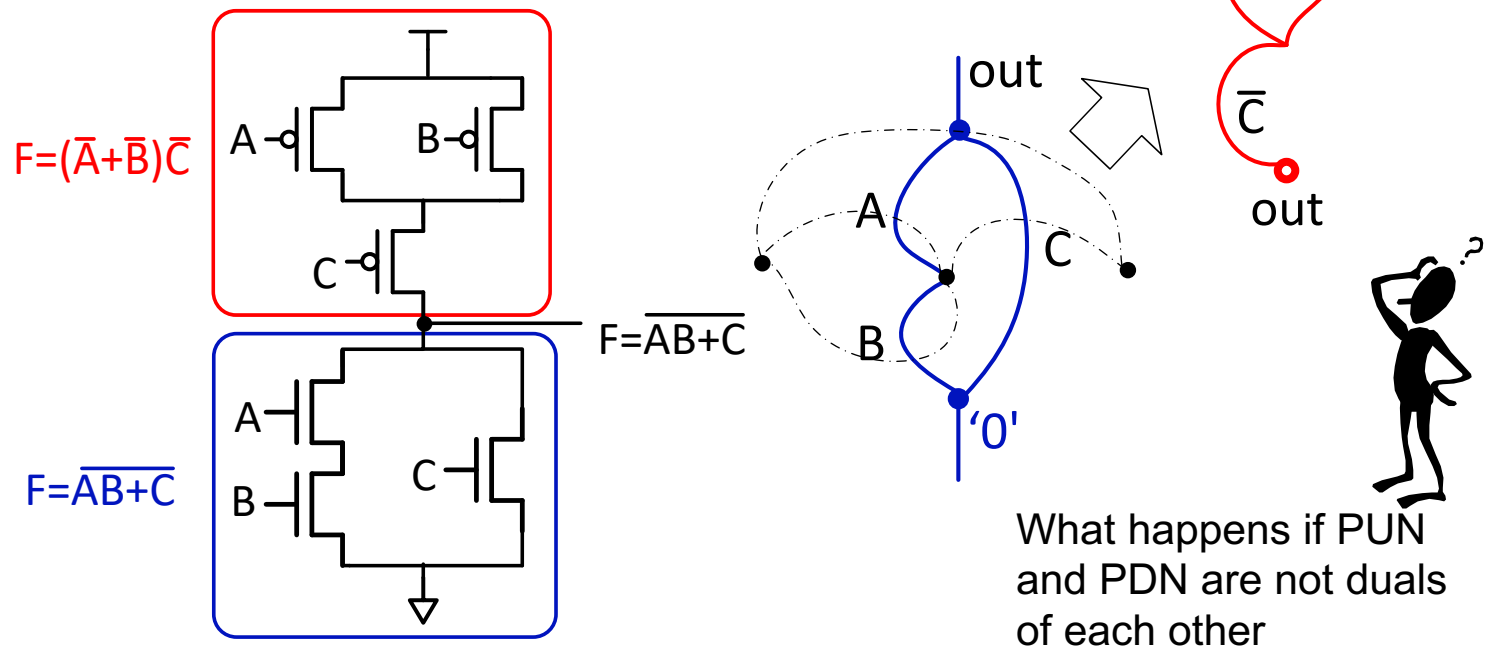
- Implement logic by connecting output to 1 (V_{dd}) or 0 (V_{ss})
  - Connect output Z to V_{dd} if function evaluates to '1'
  - Relay network connecting V_{dd} to Z referred to as the pull-up network (PUN)
  - Connect Z to V_{ss} if function evaluates to '0'
  - Relay network connecting V_{ss} to Z referred to as the pull-down network (PDN)
  - ➔Z will not be connected to *both* V_{dd} and V_{ss} . PUN and PDN ***complementary***
- Viewed as graphs, the networks are duals of each other

CMOS logic



- Implement logic by connecting output to 1 (V_{dd}) or 0 (V_{ss})
  - Connect output Z to V_{dd} if function evaluates to '1'
  - Relay network connecting V_{dd} to Z referred to as the pull-up network (PUN)
  - Connect Z to V_{ss} if function evaluates to '0'
  - Relay network connecting V_{ss} to Z referred to as the pull-down network (PDN)
  - ➔Z will not be connected to *both* V_{dd} and V_{ss} . PUN and PDN ***complementary***
- Viewed as graphs, the networks are duals of each other

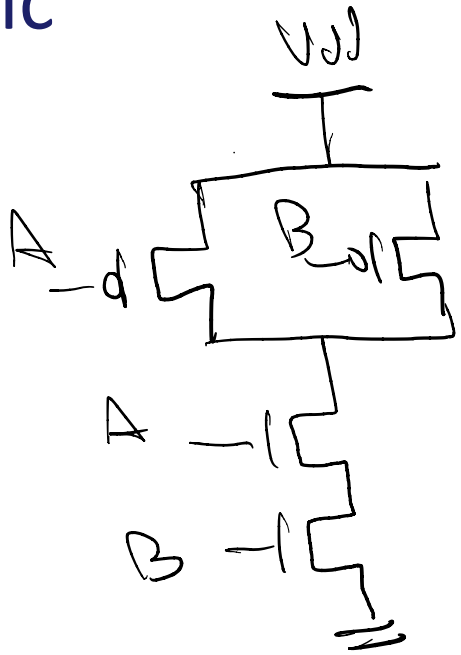
CMOS logic

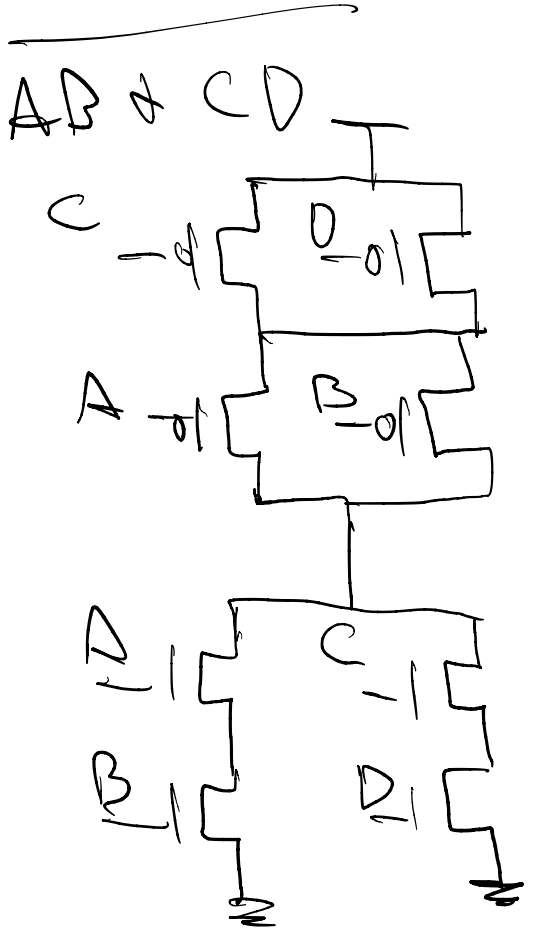


- Implement logic by connecting output to 1 (V_{dd}) or 0 (V_{ss})
  - Connect output Z to V_{dd} if function evaluates to '1'
  - Relay network connecting V_{dd} to Z referred to as the pull-up network (PUN)
  - Connect Z to V_{ss} if function evaluates to '0'
  - Relay network connecting V_{ss} to Z referred to as the pull-down network (PDN)
  - ➔Z will not be connected to *both* V_{dd} and V_{ss} . PUN and PDN ***complementary***
- Viewed as graphs, the networks are duals of each other

CMOS logic

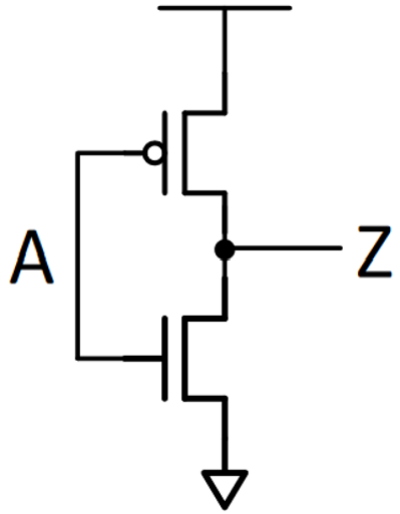
$Z = \overline{AB}$





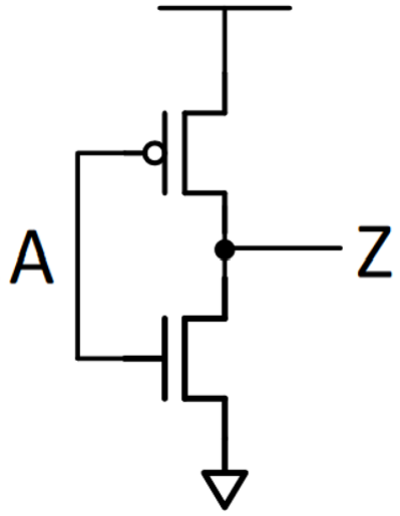
- Using A,B,C,D as inputs, implement using CMOS Logic
  - Z = ~A
  - Z=~(A&B)
  - Z=~(A&B + C&D)

CMOS logic



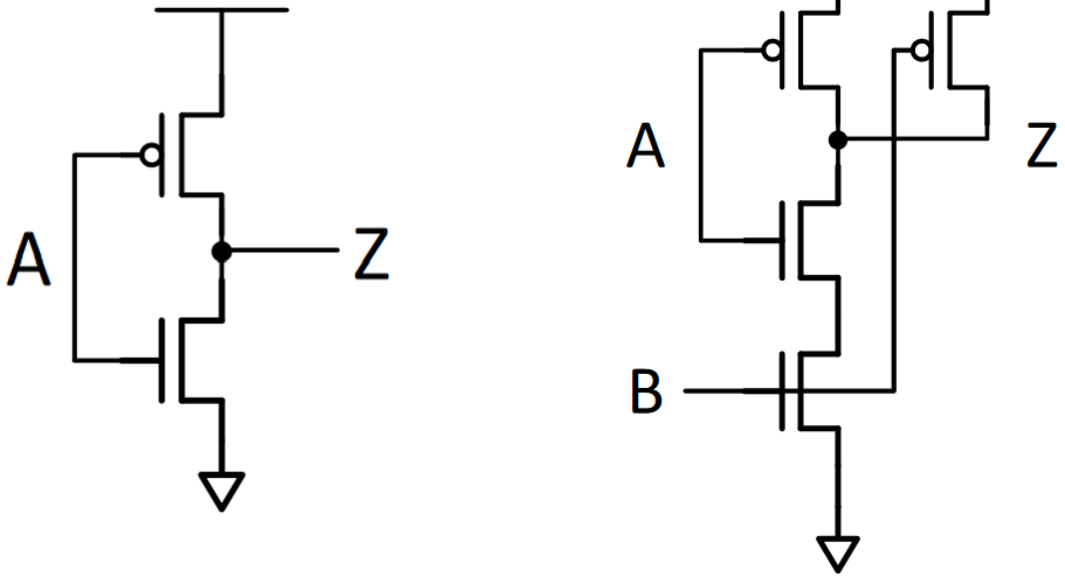
- Using A,B,C,D as inputs, implement using CMOS Logic
  - Z = ~A
  - Z=~(A&B)
  - Z=~(A&B + C&D)

CMOS logic



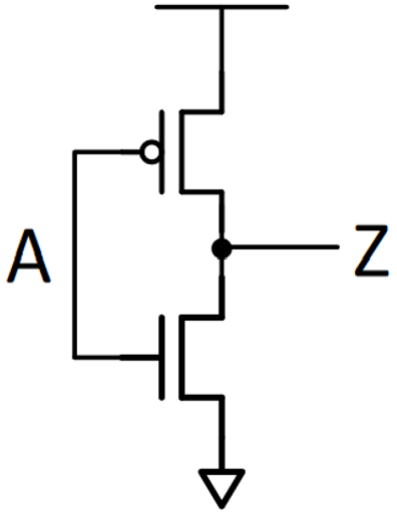
- Using A,B,C,D as inputs, implement using CMOS Logic
  - Z = ~A
  - Z=~(A&B) === ~A|~B
  - Z=~(A&B + C&D)

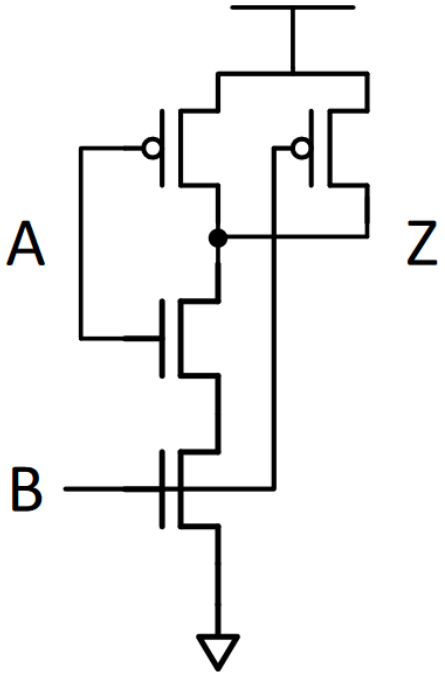
CMOS logic

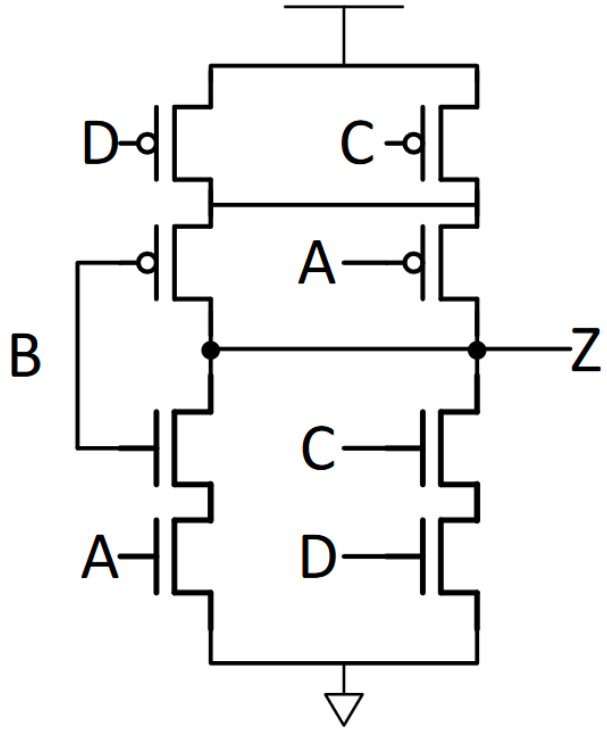


- Using A,B,C,D as inputs, implement using CMOS Logic
  - Z = ~A
  - Z=~(A&B) === ~A|~B
  - Z=~(A&B + C&D)

CMOS logic





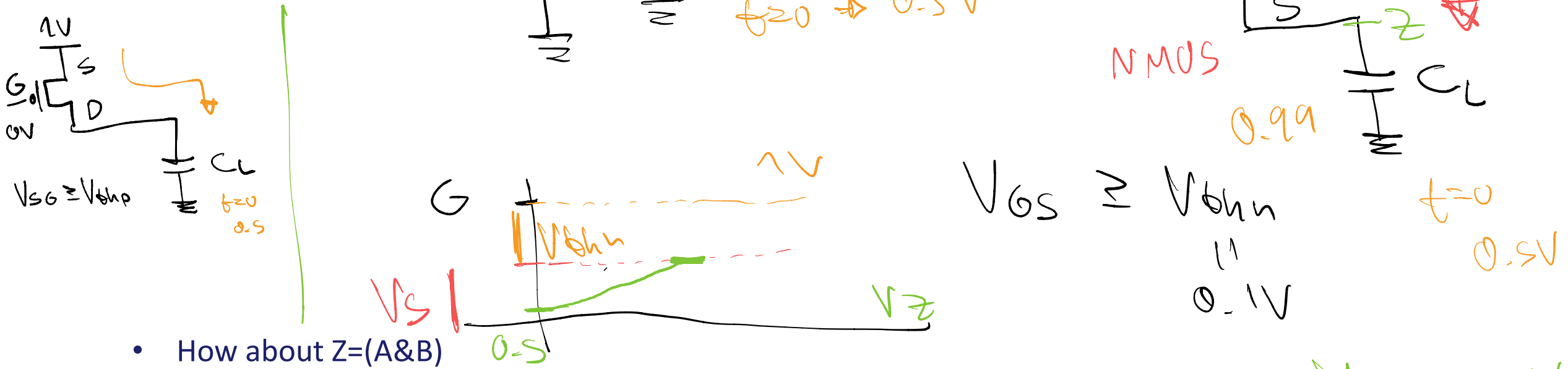


- Using A,B,C,D as inputs, implement using CMOS Logic
  - Z = ~A
  - Z=~(A&B) === ~A|~B
  - Z=~(A&B + C&D)

A short note on the Threshold Voltage

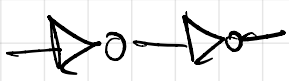
- Story thus far:
  - NMOS conducts If $V_{gate} > (V_{drain}$ or $V_{source})$
  - PMOS conducts if $V_{gate} < (V_{drain}$ or $V_{source})$
- In Reality : V_{th} , the threshold voltage plays a role
  - An “overhead” cost that must be paid to enable device to conduct
  - NMOS conducts If $V_{gate}-V_{th} > (V_{drain}$ or $V_{source})$
  - PMOS conducts if $V_{gate} + V_{th} < (V_{drain}$ or $V_{source})$

CMOS logic



- How about Z=(A&B)
  - Nmos “passes a 1” poorly, PMOS “passes a 0” poorly (More on this shortly)
    - NMOS gate must exceed source/drain by a threshold to conduct
    - PMOS gate must be lower than source/drain by a threshold to conduct
    - Recall discussion on current flow in relay networks
  - How do I get an AND gate then?

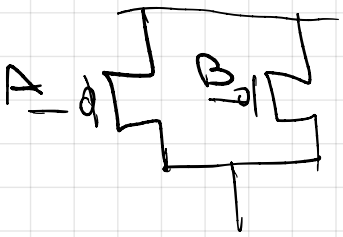
$Z = \overline{\overline{A}} = A$

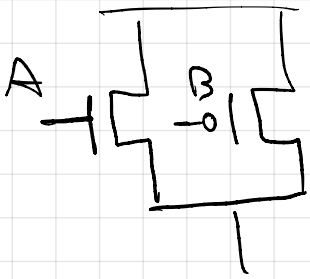


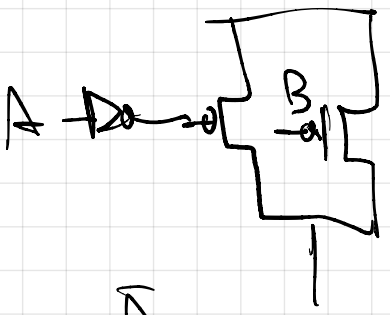
$Z = \overline{AB}$?

A=1 / = / $\bar{A}$=0

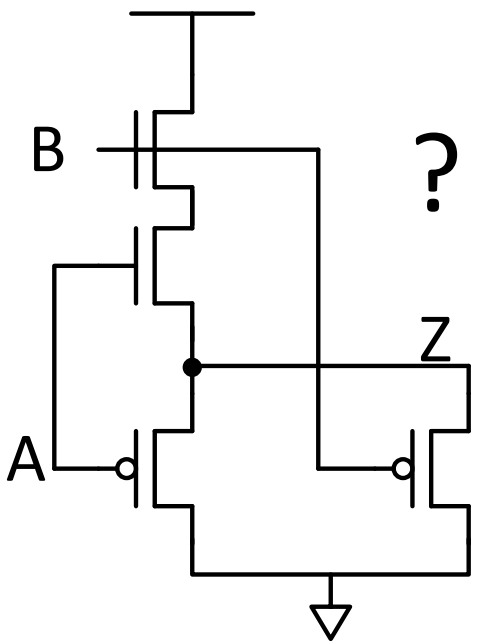
$Z = \overline{AB}$





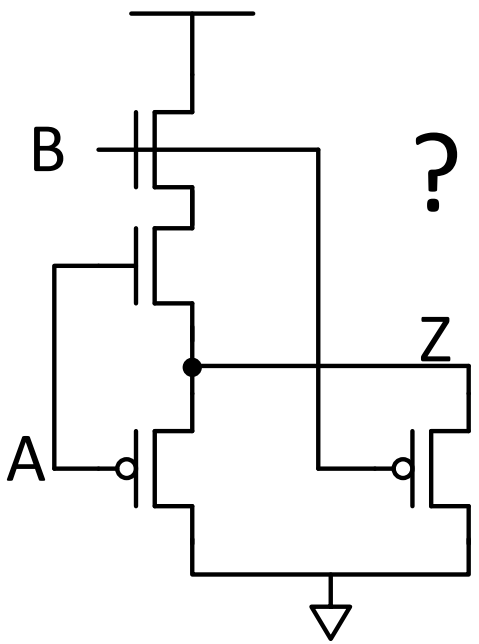


CMOS logic



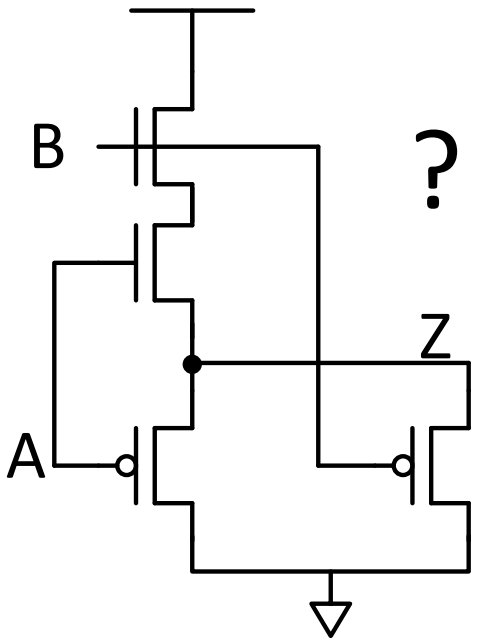
- How about Z=(A&B)
  - Nmos “passes a 1” poorly, PMOS “passes a 0” poorly (More on this shortly)
    - NMOS gate must exceed source/drain by a threshold to conduct
    - PMOS gate must be lower than source/drain by a threshold to conduct
    - Recall discussion on current flow in relay networks
  - How do I get an AND gate then?

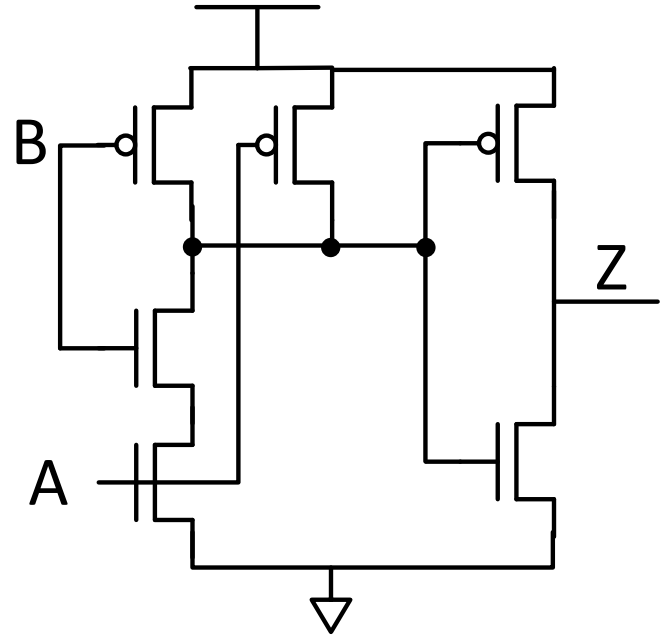
CMOS logic



- How about Z=(A&B)
  - Nmos “passes a 1” poorly, PMOS “passes a 0” poorly (More on this shortly)
    - NMOS gate must exceed source/drain by a threshold to conduct
    - PMOS gate must be lower than source/drain by a threshold to conduct
    - Recall discussion on current flow in relay networks
  - How do I get an AND gate then? Z= ~ (~(A&B)))

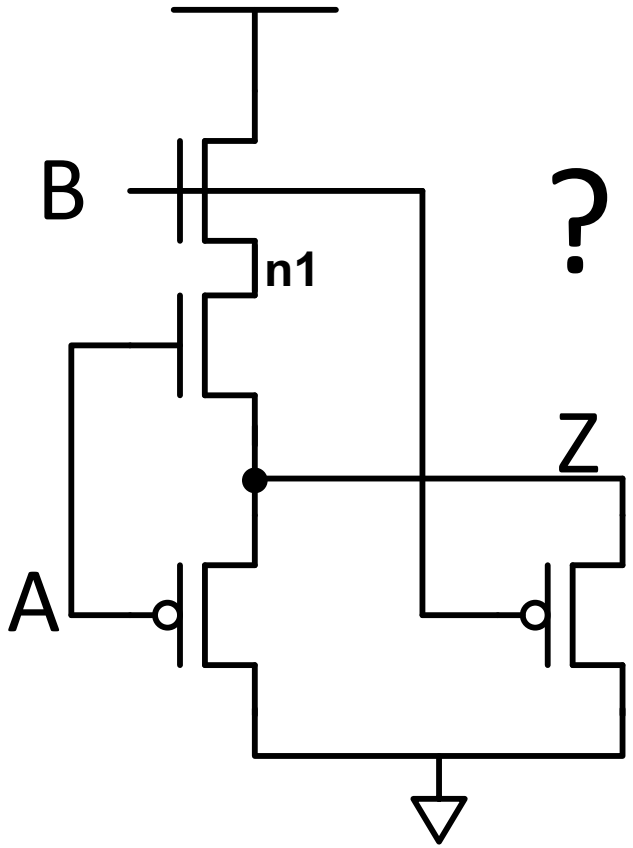
CMOS logic





- How about Z=(A&B)
  - Nmos “passes a 1” poorly, PMOS “passes a 0” poorly (More on this shortly)
    - NMOS gate must exceed source/drain by a threshold to conduct
    - PMOS gate must be lower than source/drain by a threshold to conduct
    - Recall discussion on current flow in relay networks
  - How do I get an AND gate then? Z= ~ (~(A&B)))

Quick Aside: Voltage swing range for Z



- What is the maximum voltage Z can reach (Ignoring leakage)
- What is the minimum voltage Z can reach (Ignoring leakage)