- Static complementary logic gate structures.

Gate design

Why designing gates for logic functions is non-trivial:

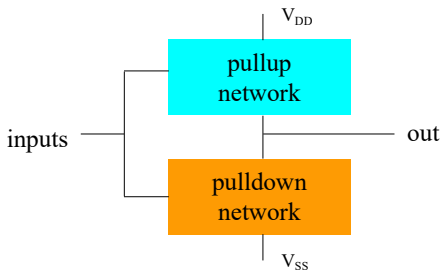
- may not have logic gates in the libray for all logic expressions;
- a logic expression may map into gates that consume a lot of area, delay, or power.

Static complementary gates

- Complementary: have complementary pullup (p-type) and pulldown (n-type) networks.
- Static: do not rely on stored charge.
- Simple, effective, reliable.

Static complementary gate structure

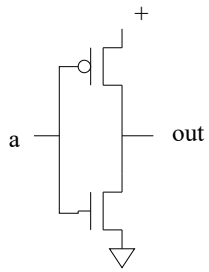
Pullup and pulldown networks:



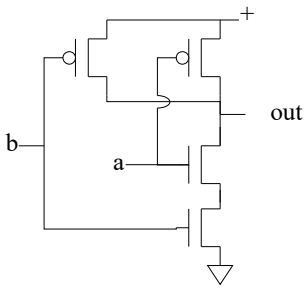
Pullup/pulldown network design

- Pullup and pulldown networks are duals.
- To design one gate, first design one network, then compute dual to get other network.
- Example: design network which pulls down when output should be 0, then find dual to get pullup network.

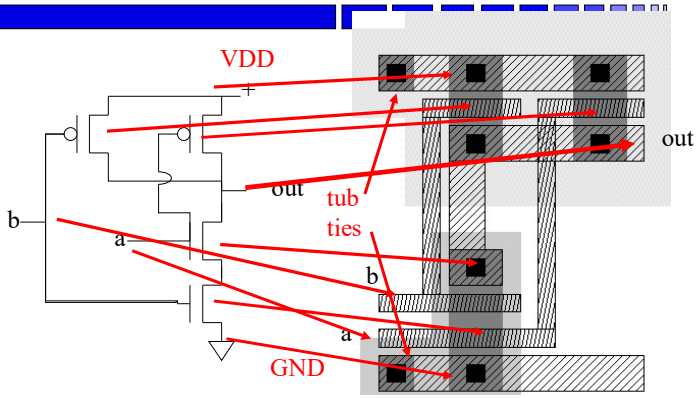
Inverter



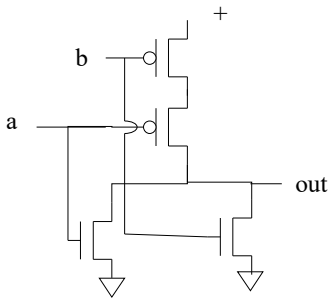
NAND gate



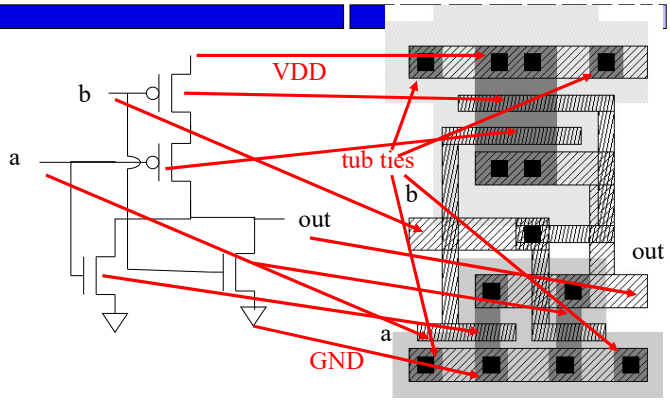
NAND layout



NOR gate



NOR layout



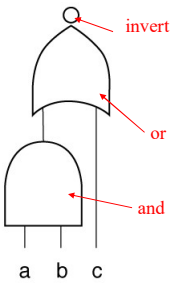
AOI/OAI gates

- AOI = and/or/invert; OAI = or/and/invert.
- Implement larger functions.
- Pullup networks (PUN) and pulldown networks (PDN) are compact: smaller area, higher speed than NAND/NOR network equivalents.
- AOI312: AND 3 inputs, AND 1 input (dummy), AND 2 inputs; OR together these terms; then invert.

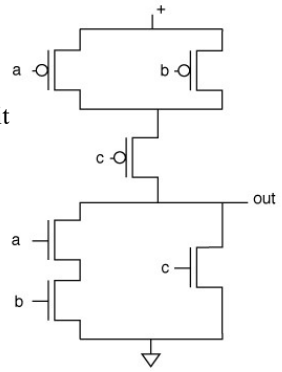
AOI example

AOI-21: out = [ab+c]'

symbol

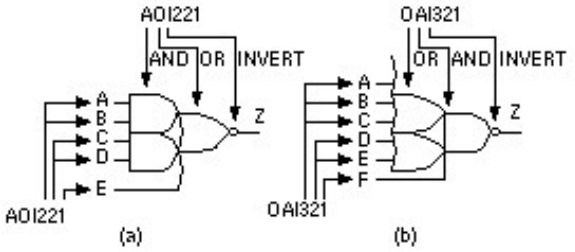


circuit



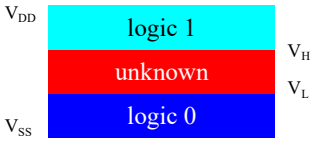
OAI-321 ??

Example: AOI221, OAI321



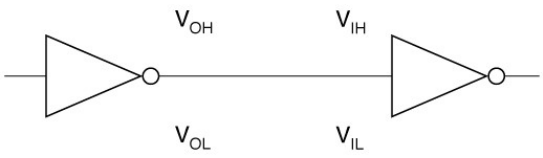
Logic levels

- Solid logic 0/1 defined by V_{SS}/V_{DD}.
- Inner bounds of logic values V_L/V_H are not directly determined by circuit properties, as in some other logic families.



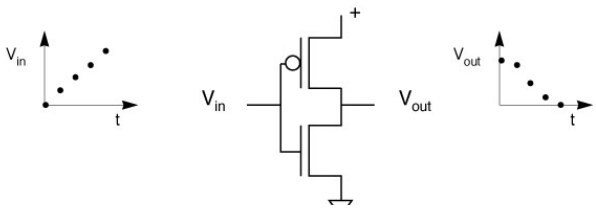
Logic level matching

- Levels at output of one gate must be sufficient to drive next gate.

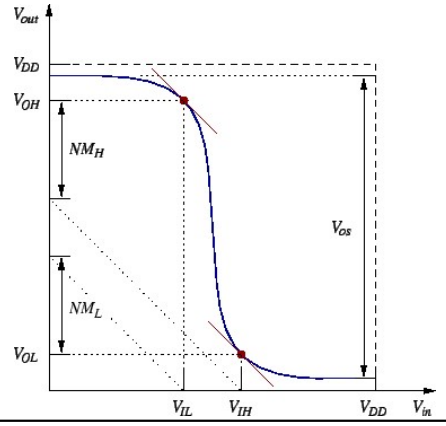


Transfer characteristics

- Transfer curve shows static input/output relationship—hold input voltage, measure output voltage.



Inverter transfer curve



Noise Margin

$$NM_L = \frac{V_{IL} - V_{OL}}{V_{DD}}$$

$$NM_H = \frac{V_{OH} - V_{IH}}{V_{DD}}$$