

# CMOS Transistors and Boolean Logic Gates



# CMOS Transistors

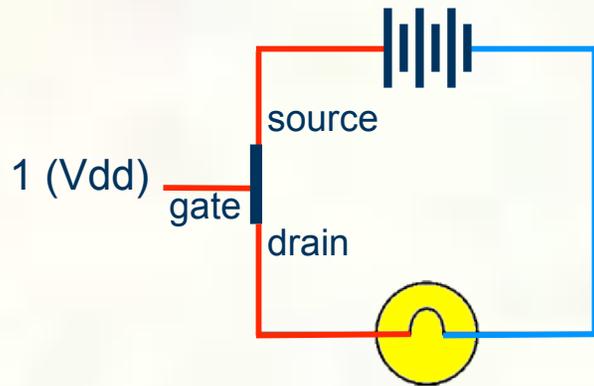
---

- Need circuits to represent 2 discrete values
  - 1,0 for binary representations
  - True, False for Boolean logic
- Let high voltage ( $V_{dd}$ ) represent 1, or true
- Let low voltage (0 volts or gnd) represent 0, or false
- If we have some switches to control whether or not these voltages can propagate through a circuit, we can build a computer with them
  - Note, the earliest digital computers were electromechanical, made out of relays, so this is hardly a new idea
- Our switches will be CMOS transistors

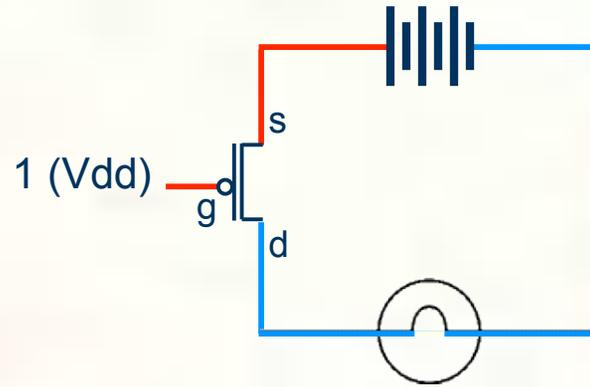
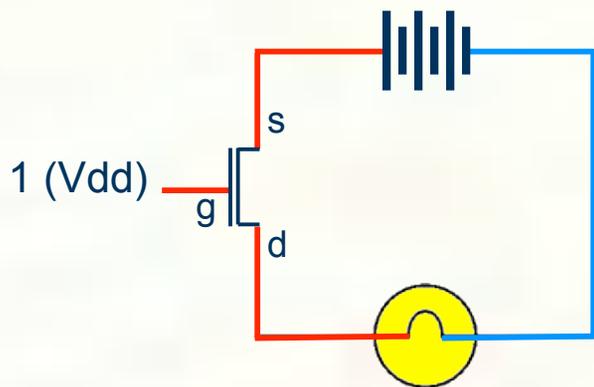
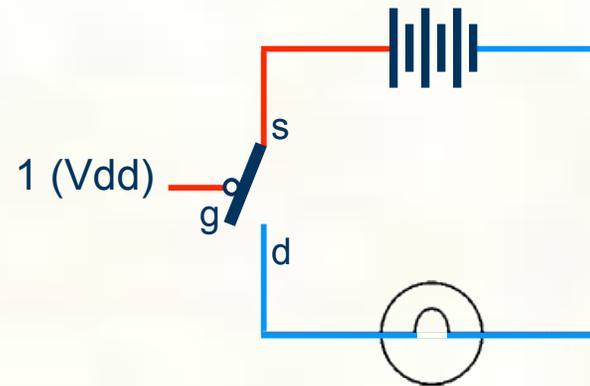


# Two kinds of transistors

## N-type



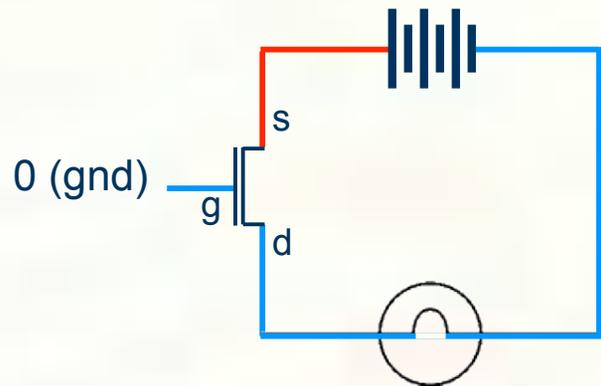
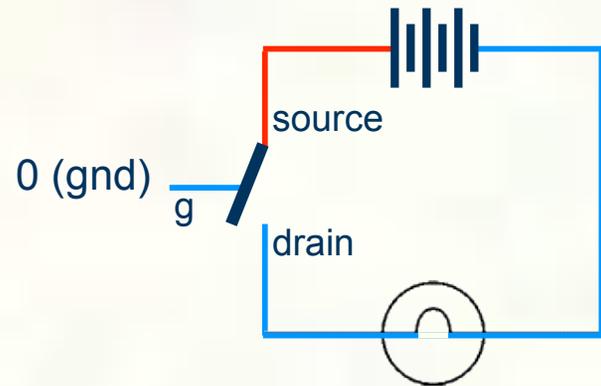
## P-type



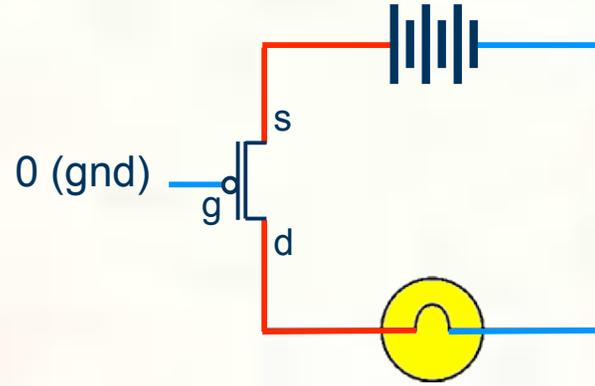
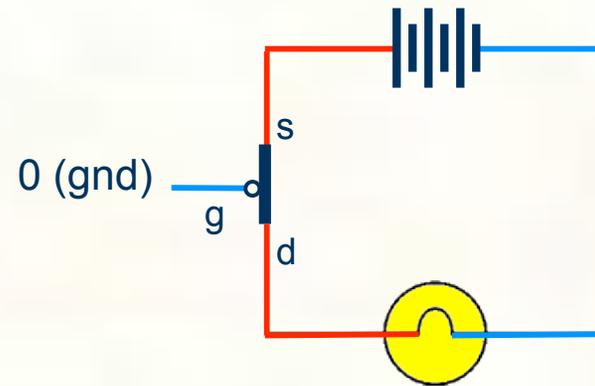


# Two kinds of transistors

## N-type



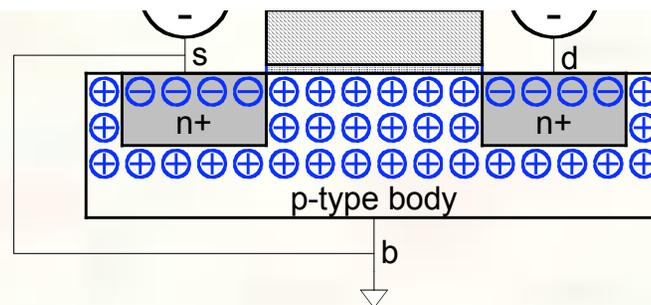
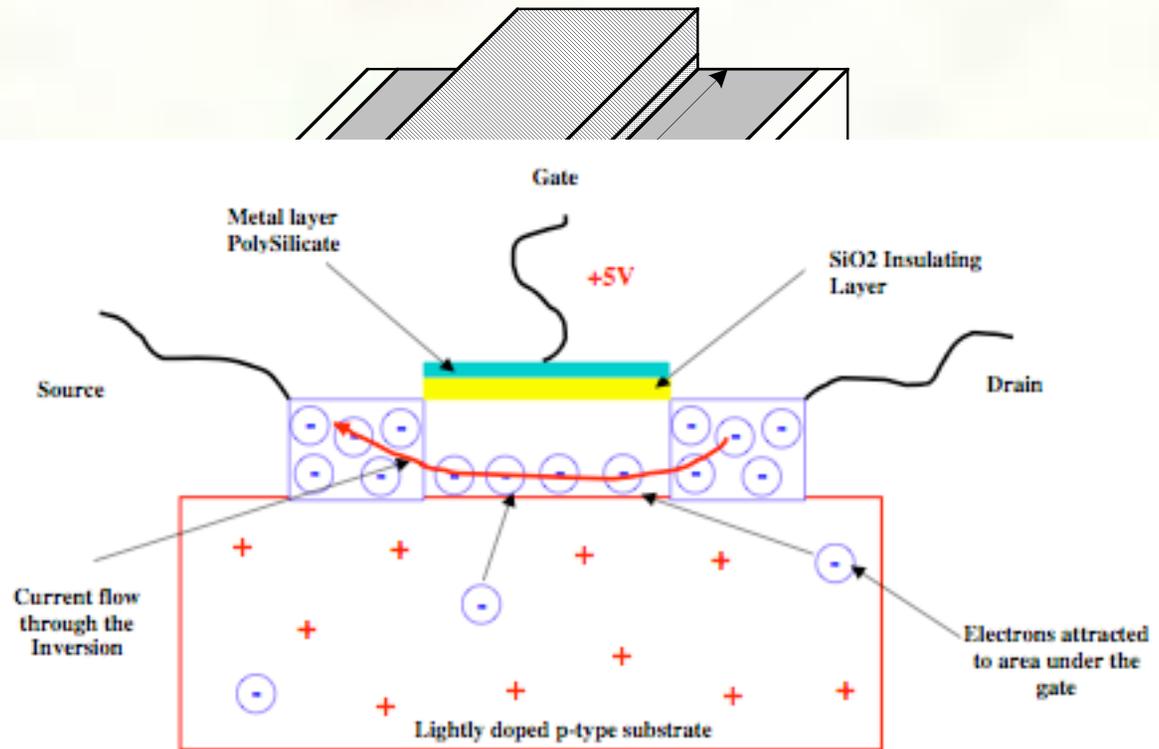
## P-type





# How they work as switches

N-type



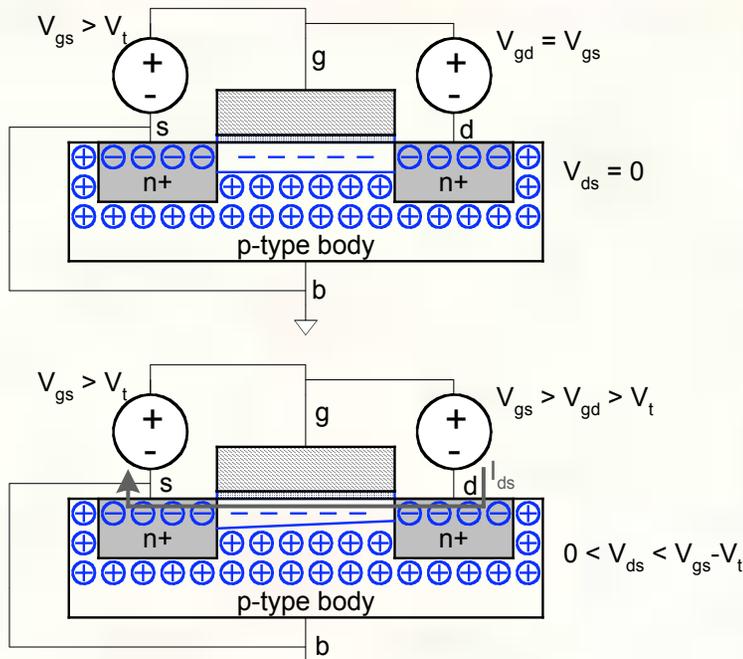


# How they work as switches

When  $V_{gs} > V_{th}$ , the threshold voltage

- excess electrons attracted into channel
- current flows and switch is closed
- drain voltage cannot be more than source voltage =  $V_g - V_{th}$
- this is at most  $V_{dd} - V_{th}$
- $V_{dd} - V_{th}$  is still considered a 1, but a *weak* 1
- if source voltage is 0, then drain voltage is too, so 0 still strong

N-type



CMOS transistor pictures from  
UT ECE VLSI course slides



# CMOS circuit rules

---

- Never create a path from  $V_{dd}$  to gnd
- Don't pass weak values
  - N-type transistors pass weak 1's ( $V_{dd} - V_{th}$ )
  - N-type transistors pass strong 0's (gnd)
  - Use N-type transistors only to pass 0's (n to negative)
  - Conversely for P-type transistors
    - Pass weak 0's ( $V_{th}$ ), strong 1's ( $V_{dd}$ )
    - Use P-type transistors only to pass 1's (p to positive)
- Never leave a wire undriven
  - Make sure there's always a path to  $V_{dd}$  or gnd

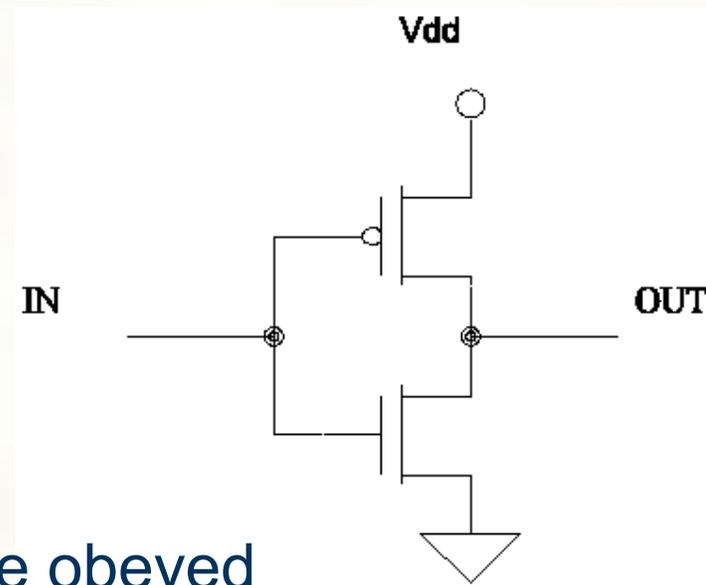


# Example CMOS gate - inverter

Truth table

In	Out
0	1
1	0

Circuit



Note how all 3 design rules are obeyed  
Circuit amplifies weak input 1 or 0