Electronics Systems (938II)

Lecture 1.1

Modern Electronic Systems (intro) – Design flow and HDL

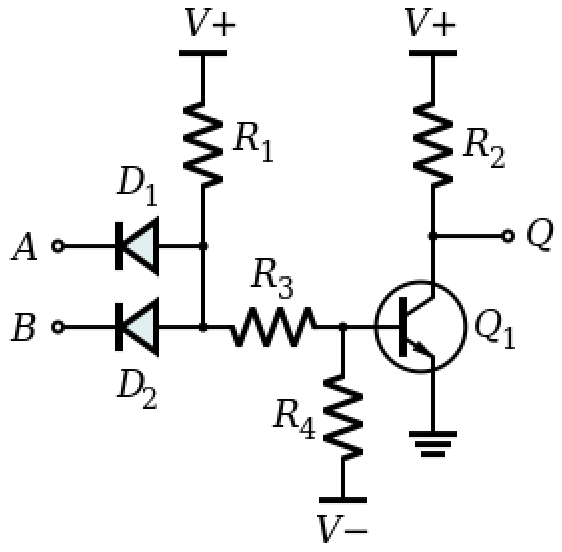
Luca Crocetti

Brief recall on CMOS technologies

- Digital electronics systems
  - Different technologies

Brief recall on CMOS technologies

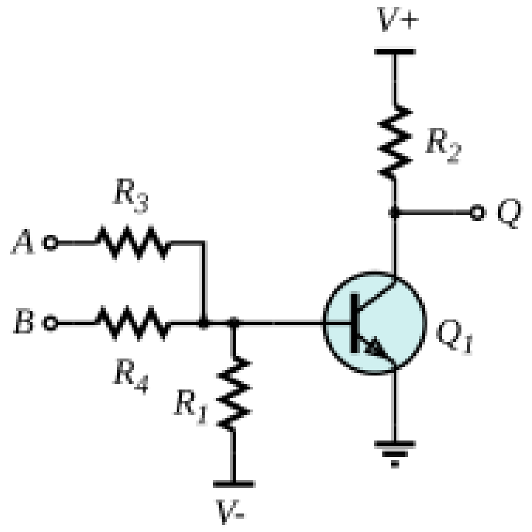
- Digital electronics systems
  - Different technological families
    - DTL = Diode-Transistor Logic



NAND gate

Brief recall on CMOS technologies

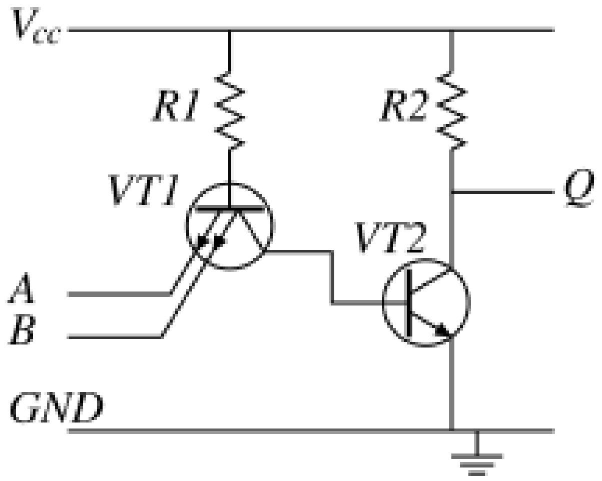
- Digital electronics systems
  - Different technological families
    - DTL
    - RTL = Resistor-Transistor Logic



NOR gate

Brief recall on CMOS technologies

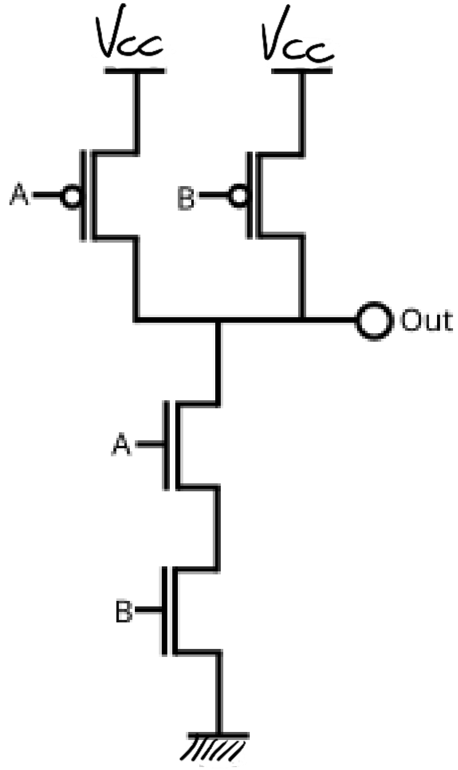
- Digital electronics systems
  - Different technological families
    - DTL
    - RTL
    - TTL = Transistor-transistor Logic



NAND gate

Brief recall on CMOS technologies

- Digital electronics systems
  - Different technological families
    - DTL
    - RTL
    - TTL
    - CMOS = Complementary MOS (logic)
      - p-MOS
      - n-MOS



NAND gate

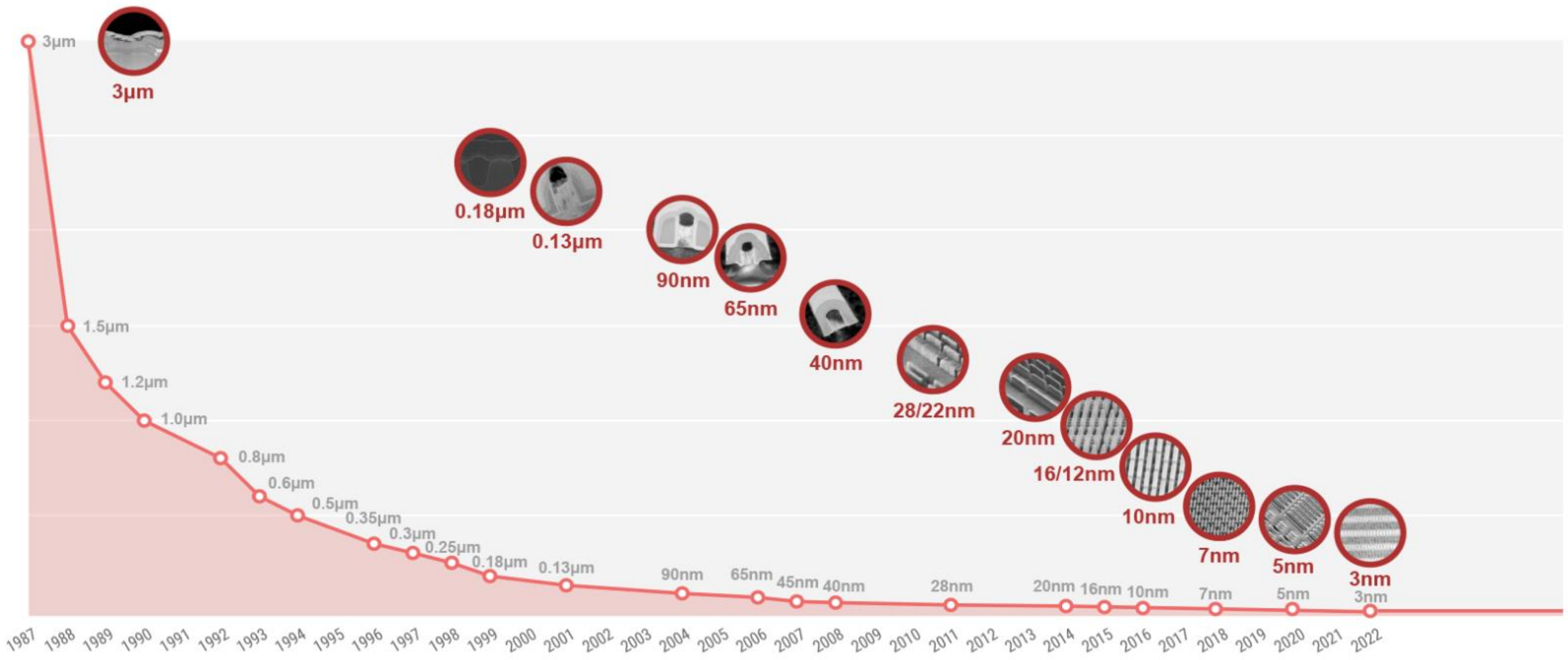
Brief recall on CMOS technologies

- Digital electronics systems
  - Different technological families
    - DTL
    - RTL
    - TTL
    - **CMOS**

Dominant technology:
- Very Large Scale Integration (VLSI)
- Complete system on unique chip
  - Sensor
  - ADC
  - Digital logic

Integration Scale and VLSI circuits

- Extreme miniaturization supported by CMOS technologies
  - From TSMC* website:



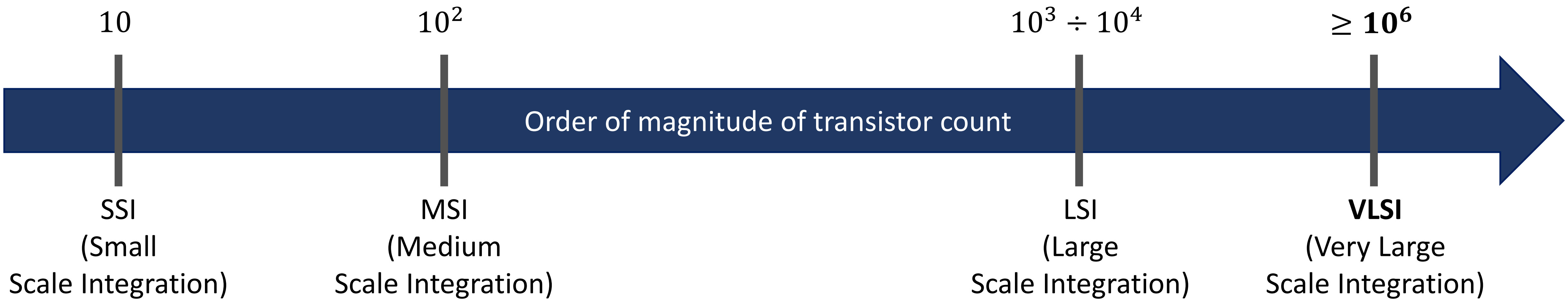
* TSMC = semiconductor manufacturer

Integration Scale and VLSI circuits

- Extreme miniaturization supported by CMOS technologies
  - Larger number of transistors for the same silicon area
    - Cost reduction: chip cost $\propto$ silicon area
  - Lower supply voltage
    - Lower power consumption
  - Higher frequencies
    - More compact logic gates $\approx$ lower propagation delays

Integration Scale and VLSI circuits

- Number of transistors in the same chip
- Over the years, we passed
  - From SSI (Small Scale Integration): transistor count $\propto 10$
  - To **VLSI** (**V**ery **L**arge **S**cale **I**ntegration): transistor count $\geq 10^6$

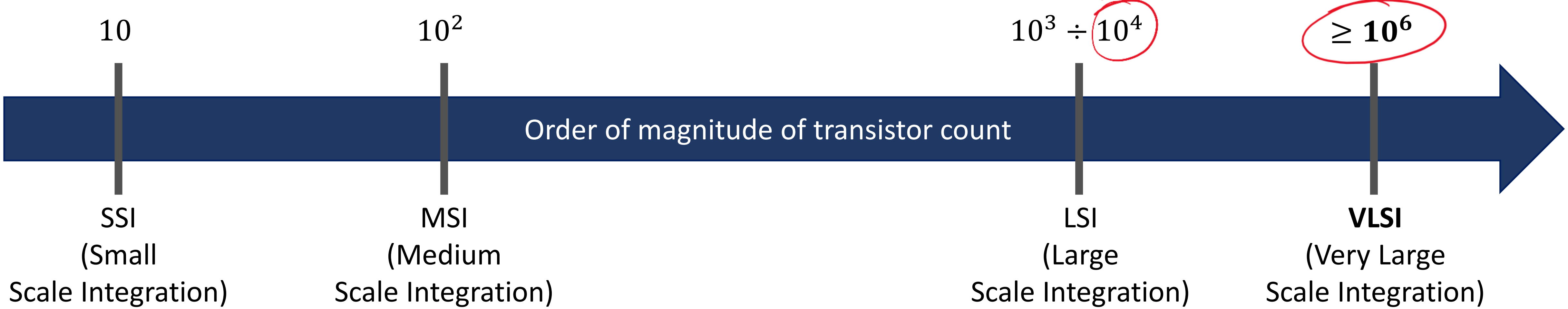


Integration Scale and VLSI circuits

- Number of transistors in the same chip
- Over the years, we passed
  - From SSI (Small Scale Integration): transistor count $\propto 10$
  - To **VLSI** (**V**ery **L**arge **S**cale **I**ntegration): transistor count $\geq 10^6$

TTL

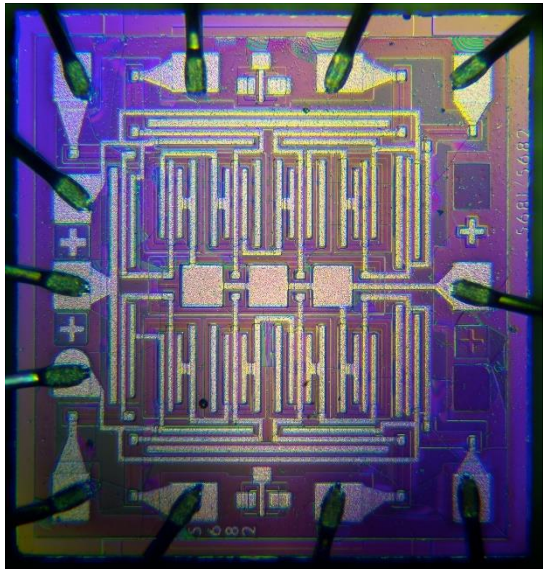
CMOS



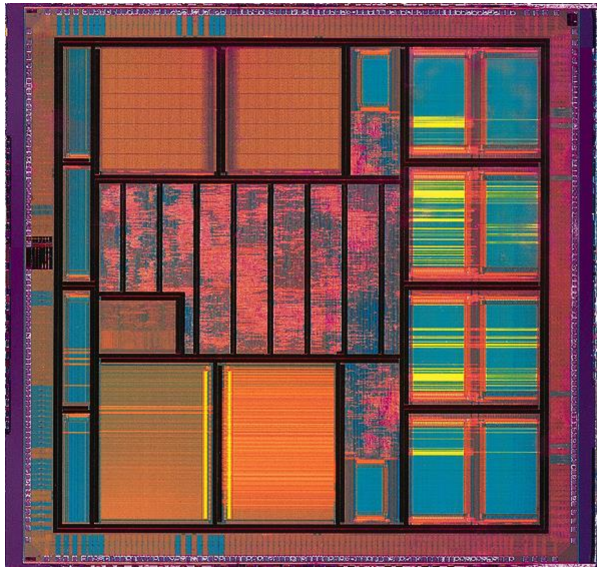
Integration Scale and VLSI circuits

- In other words, we passed …

… from this, …



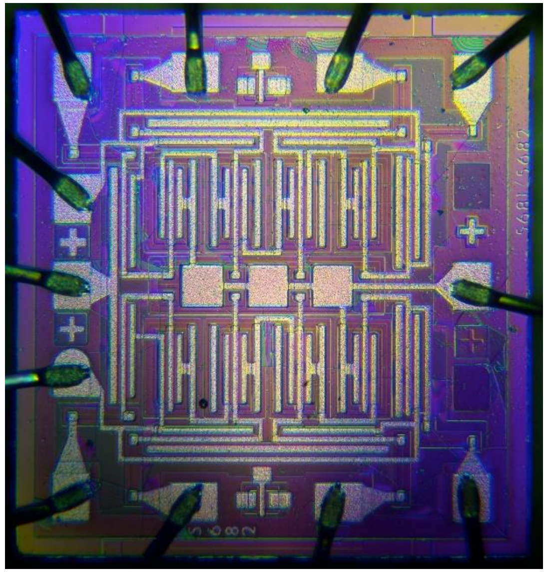
… to this!



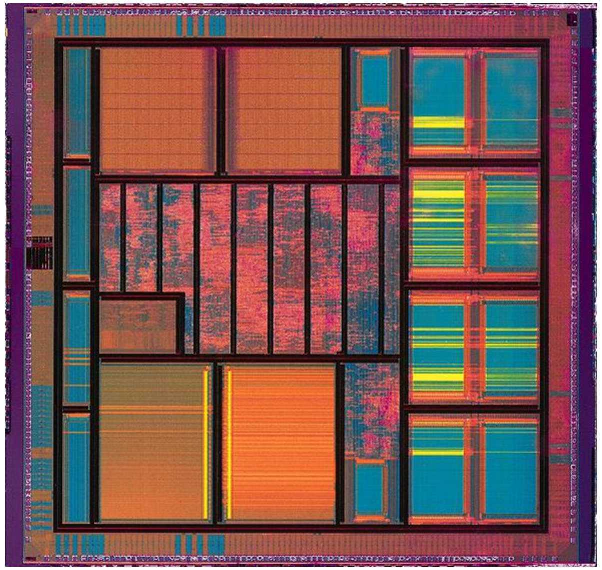
Integration Scale and VLSI circuits

- In other words, we passed …

… from this, …



… to this!

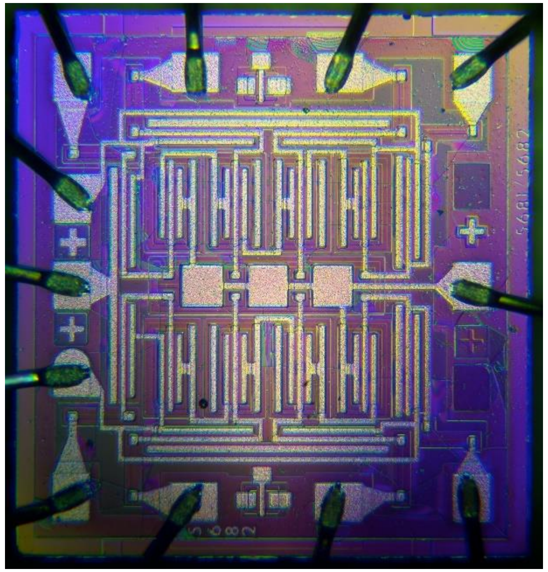


Layout of a modern Intel processor

Integration Scale and VLSI circuits

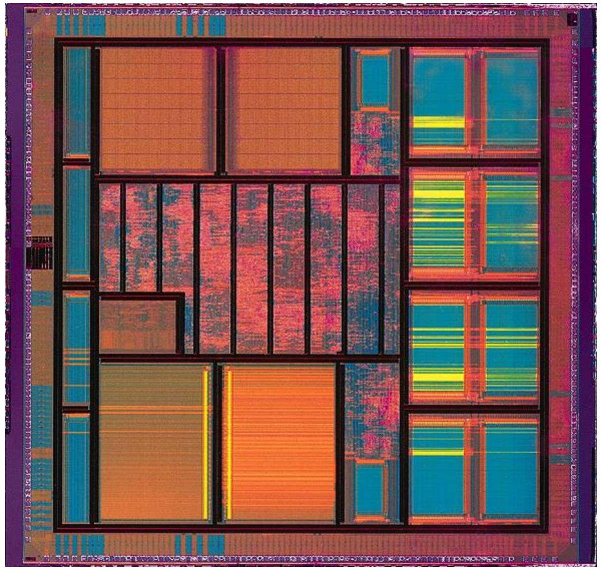
- In other words, we passed …

… from this, …



Hand design

… to this!



Automated design!!!

VLSI circuits

- Most modern digital electronic systems (or the digital part of an analog and digital system) are …
  - … VLSI circuits
  - … based on CMOS technologies
  - … designed using automated tools
    - EDA = Electronic Design Automation

VLSI circuits

- Most modern digital electronic systems (or the digital part of an analog and digital system) are …
  - … VLSI circuits
  - … based on CMOS technologies
  - … designed using **automated tools**
    - **EDA** = Electronic Design Automation

HDL for design of VLSI circuits

- Automated design of modern digital circuits relies on the usage of **HDL** languages
  - HDL = Hardware Description Language
  - Languages (programming code) that describe a (digital) circuit

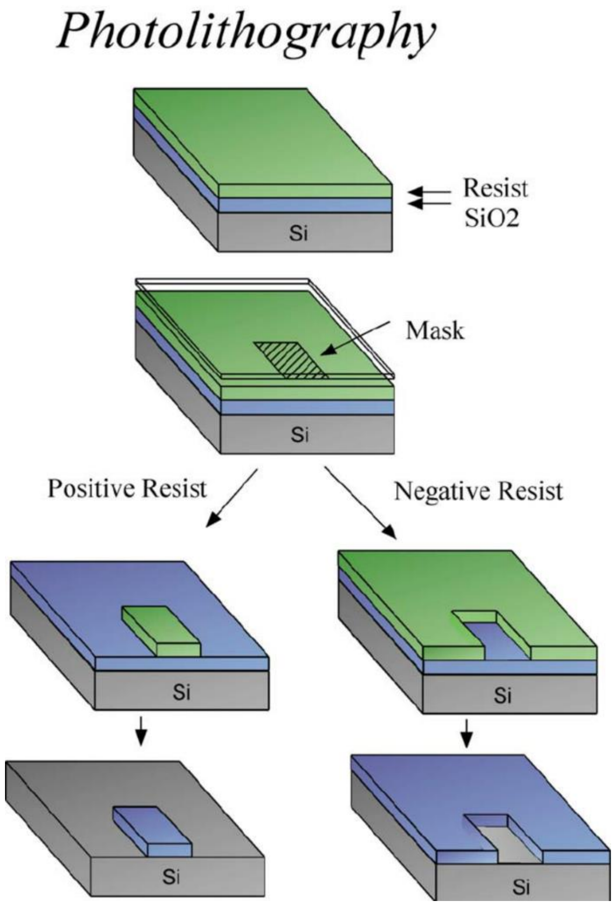
HDL code

Corresponding circuit

HDL for design of VLSI circuits

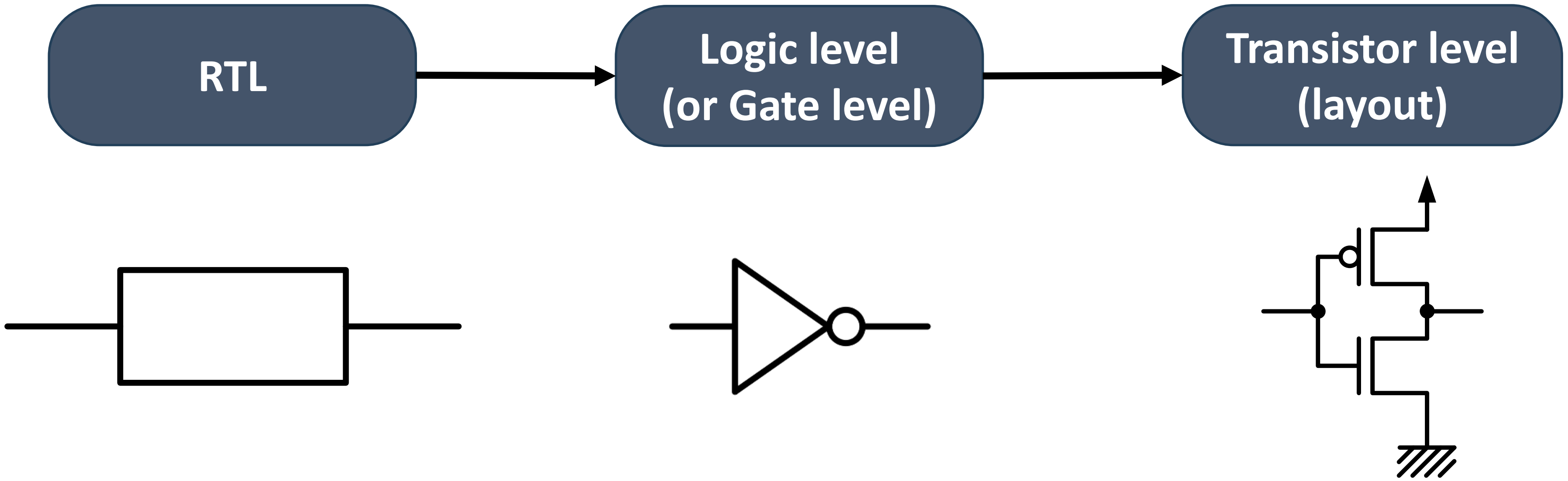
- Design flow
  - HDL code → Layout of the circuit

- Circuits manufacturing
  - Photolithography
  - The circuit layout define the mask(s) used in the photolithographic process



HDL for design of VLSI circuits

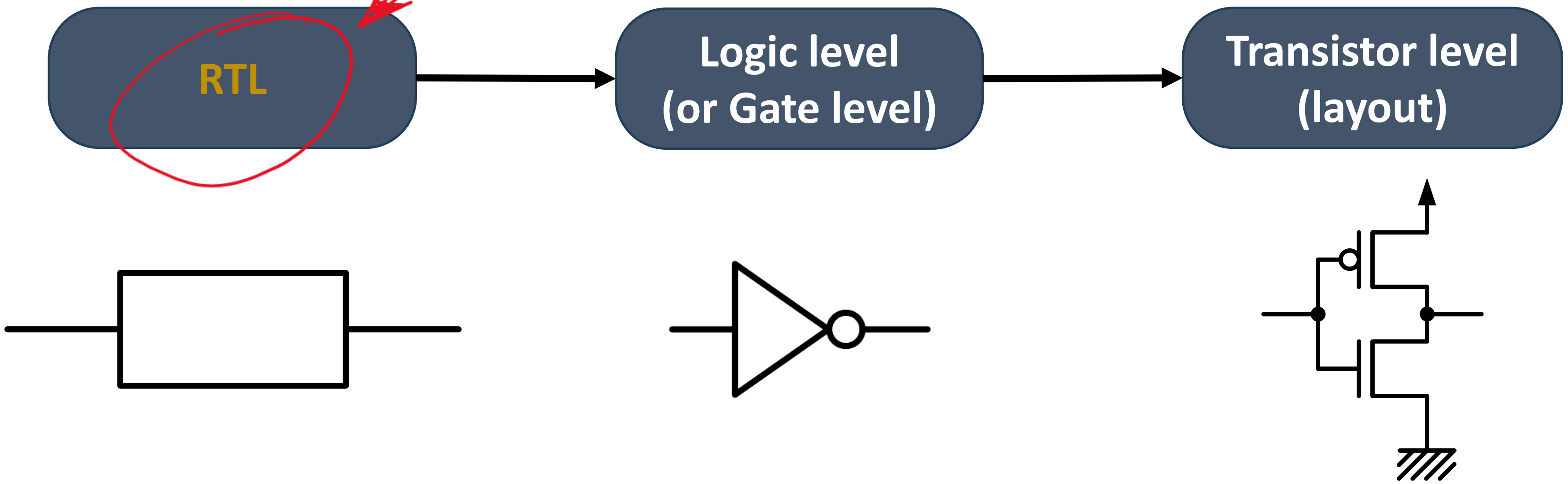
- Design flow
  - HDL code → Layout of the circuit
  - 3 representation levels



HDL for design of VLSI circuits

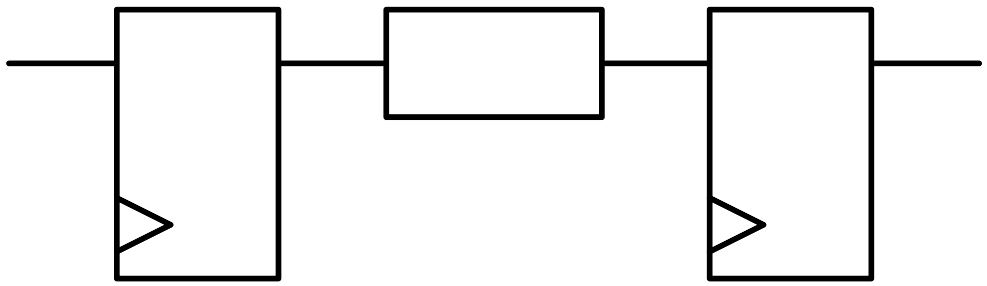
- Design flow
  - HDL code → Layout of the circuit
  - 3 representation levels

RTL = **R**egister **T**ransfer **L**evel



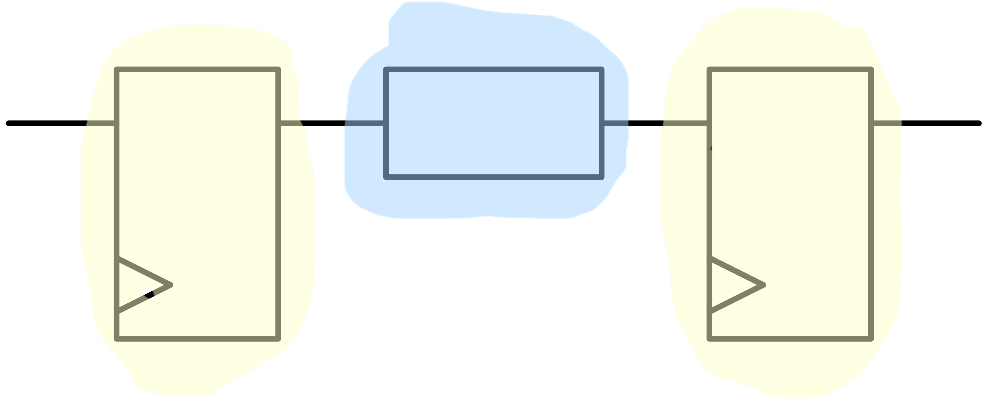
HDL for design of VLSI circuits

- RTL
  - Abstract description/representation of the circuit in terms of
    - **Registers** (we are going to see registers later)
    - **Logical operation(s)** on the signals from registers



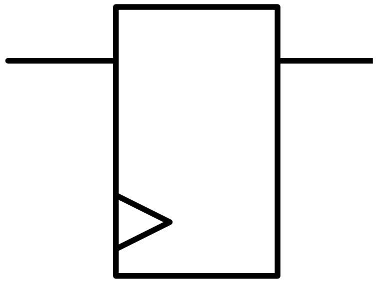
HDL for design of VLSI circuits

- RTL
  - Abstract description/representation of the circuit in terms of
    - **Registers** (we are going to see registers later)
    - **Logical operation(s)** on the signals from registers



HDL for design of VLSI circuits

- RTL
  - Representation of circuit components/elements: corresponding **functional symbol**

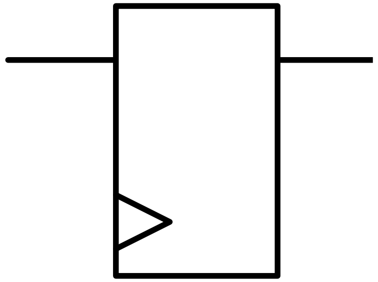


Functional symbol of register

Functional symbol of generic function

HDL for design of VLSI circuits

- RTL
  - Representation of circuit components/elements: corresponding **functional symbol**



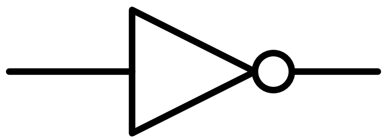
Functional symbol of register

Functional symbol of generic function

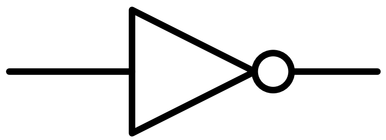
- For functions/operations
  - The symbol may change depending on the operation it performs
  - For high-complexity functions, a generic box with an appropriate number of input(s) and output(s) is generally used
  - <u>For single-gate functions (e.g., NOT, AND, NAND, OR, NOR, …), the functional symbol correspond to the gate-level symbol</u>

HDL for design of VLSI circuits

- RTL
  - Representation of circuit components/elements: corresponding **functional symbol**
    - Examples of single-gate logical functions: **NOT**



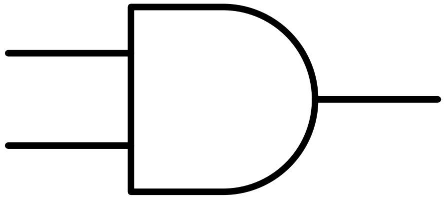
(functional symbol – RTL)



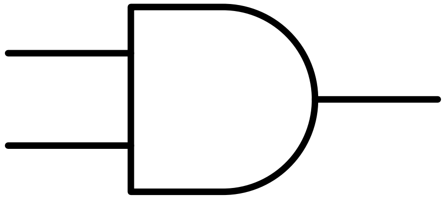
(gate-level)

HDL for design of VLSI circuits

- RTL
  - Representation of circuit components/elements: corresponding **functional symbol**
    - Examples of single-gate logical functions: **AND**



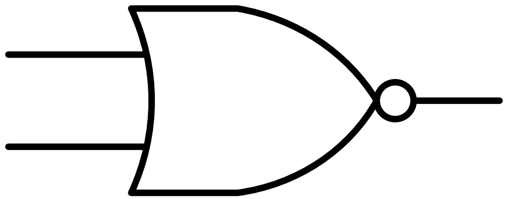
(functional symbol – RTL)



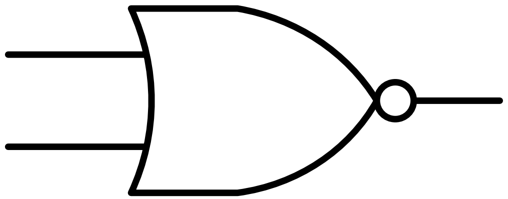
(gate-level)

HDL for design of VLSI circuits

- RTL
  - Representation of circuit components/elements: corresponding **functional symbol**
    - Examples of single-gate logical functions: **NOR**



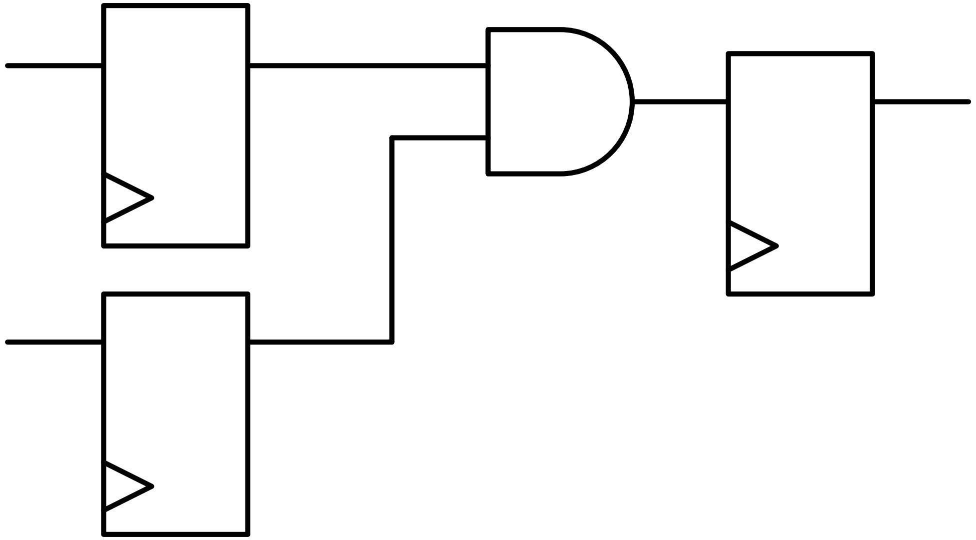
(functional symbol – RTL)

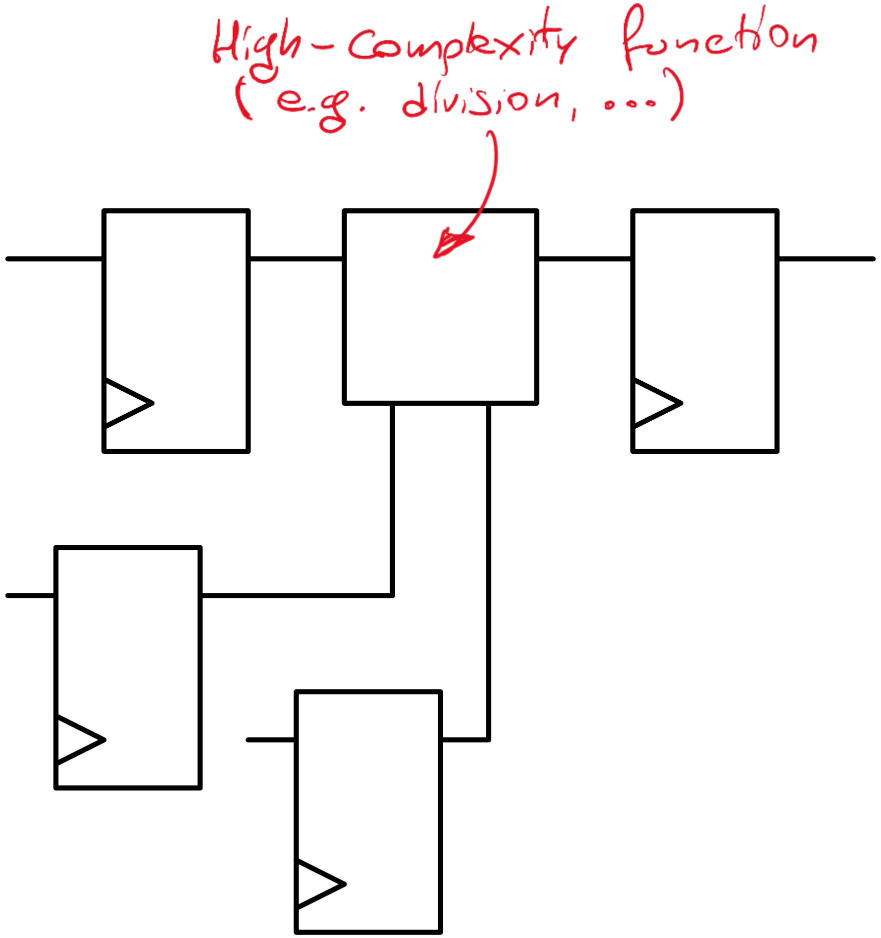


(gate-level)

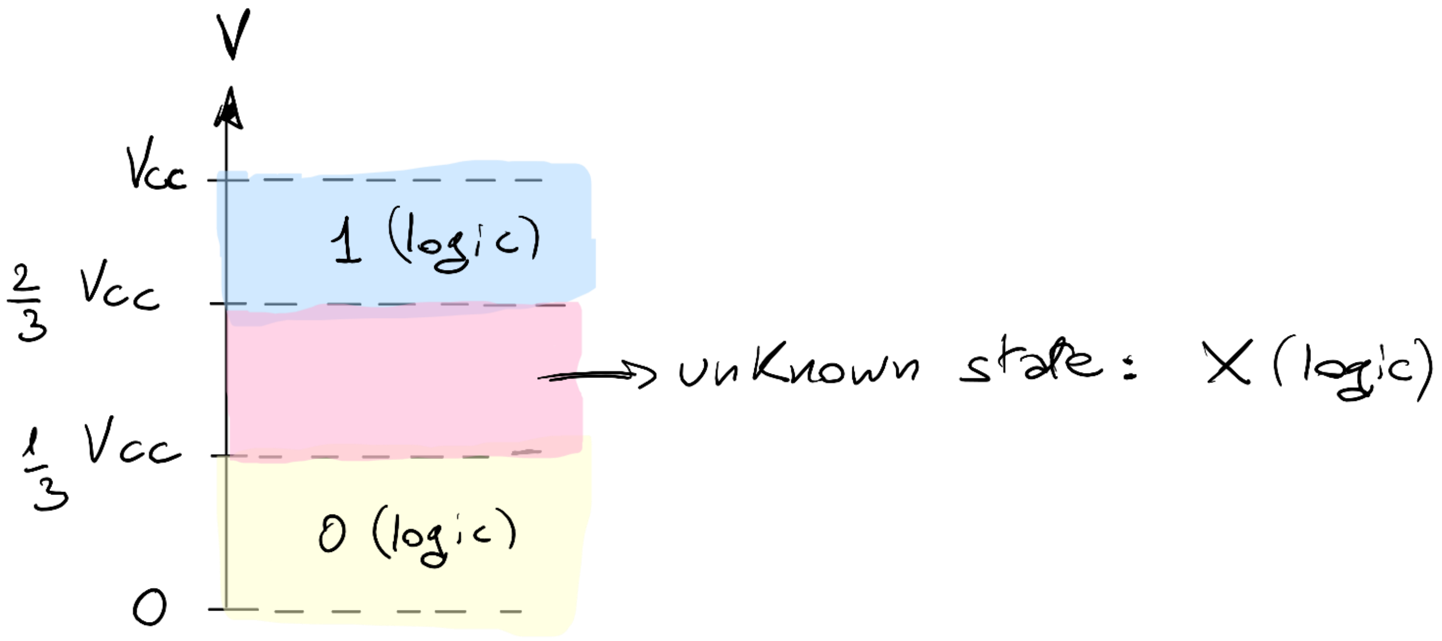
HDL for design of VLSI circuits

- RTL
  - Examples





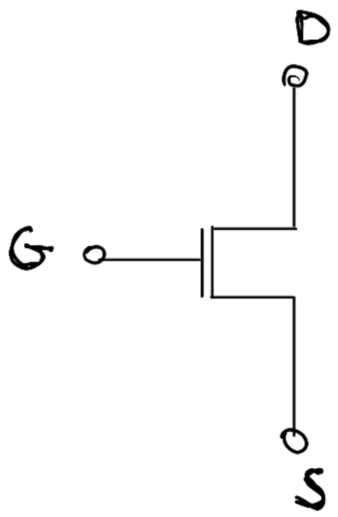
Brief recall on CMOS technologies



- Over the years, V_{CC} scaled from $5\ V$ down to $0.9\ V$

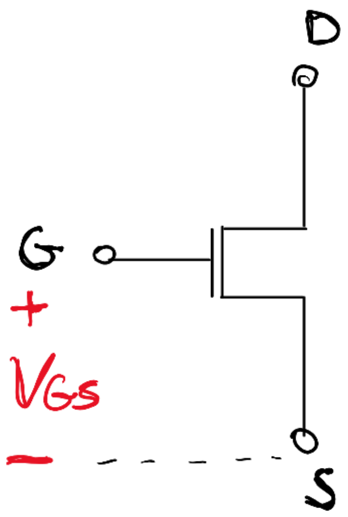
Brief recall on CMOS technologies

- n-MOS transistor ≈ controlled switch



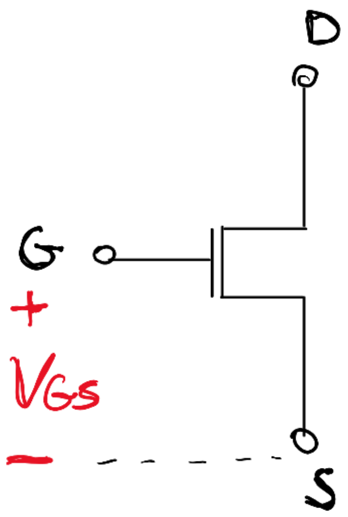
Brief recall on CMOS technologies

- n-MOS transistor ≈ controlled switch

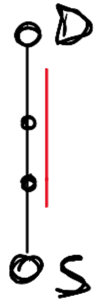


Brief recall on CMOS technologies

- n-MOS transistor ≈ controlled switch



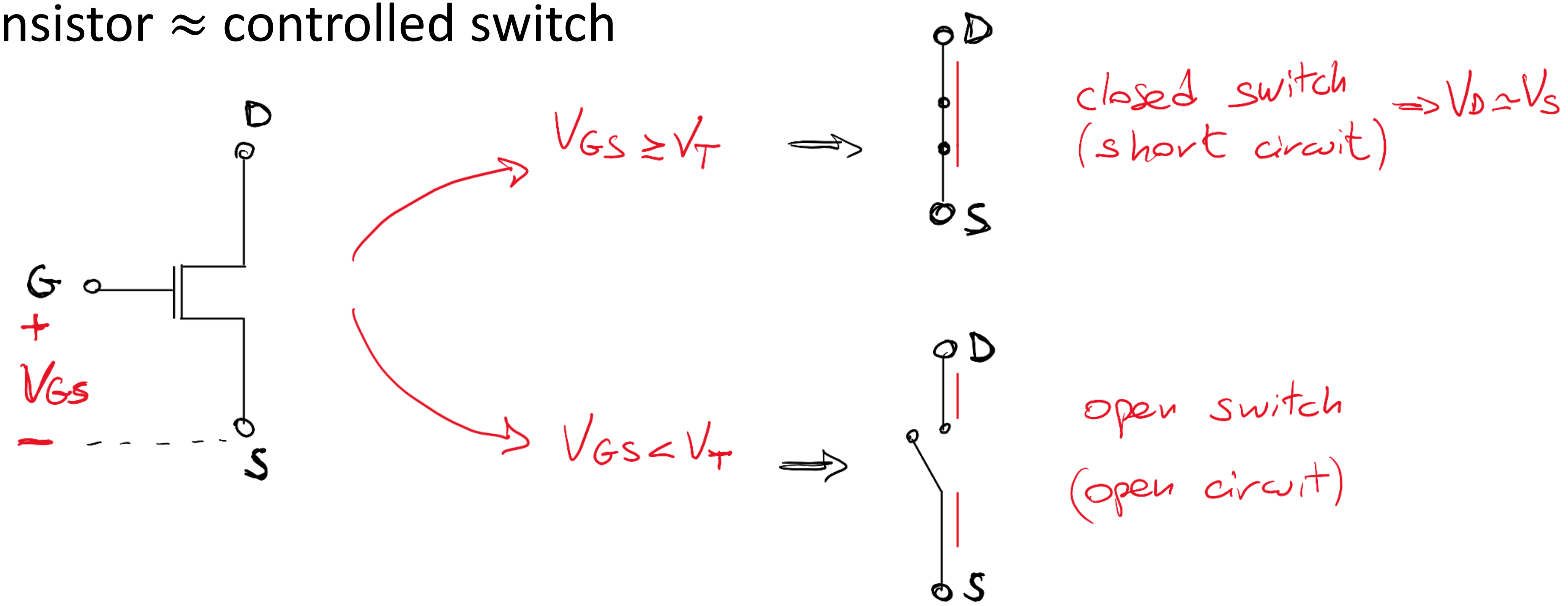
$V_{GS} \geq V_T \Longrightarrow$



closed switch
(short circuit) $\Rightarrow V_D \simeq V_S$

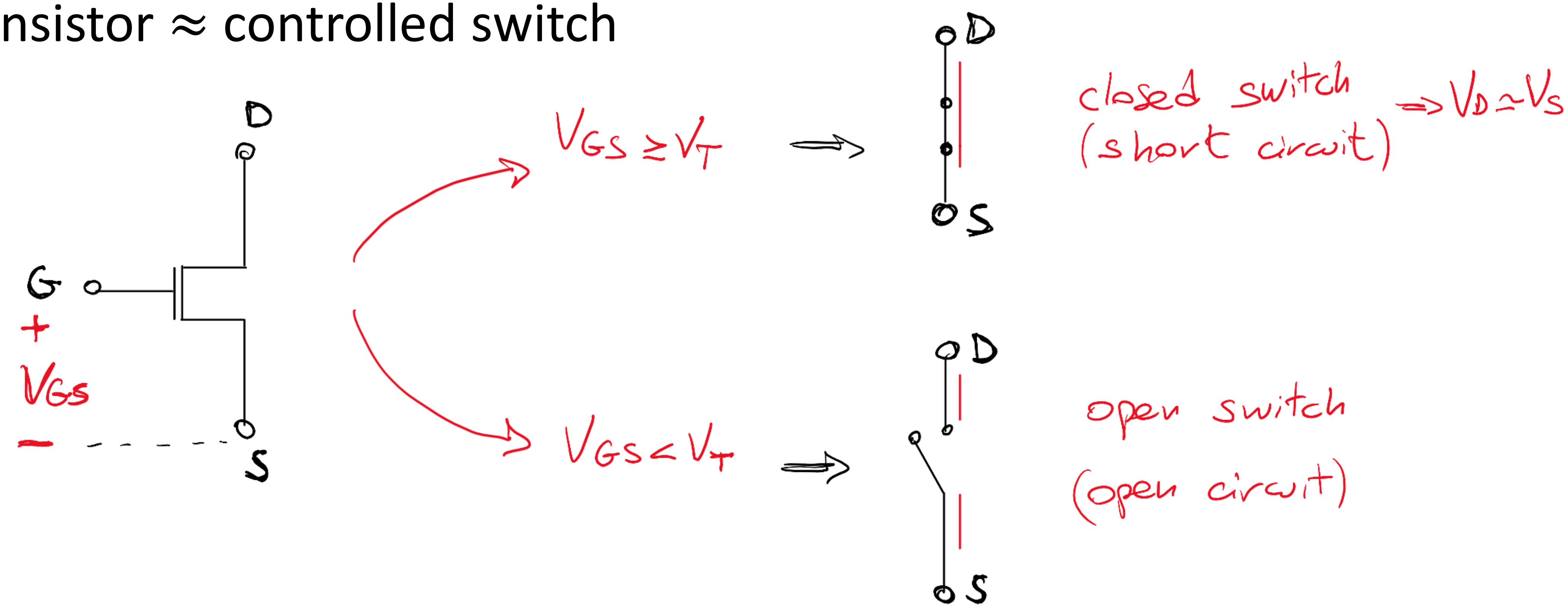
Brief recall on CMOS technologies

- n-MOS transistor ≈ controlled switch



Brief recall on CMOS technologies

- n-MOS transistor ≈ controlled switch



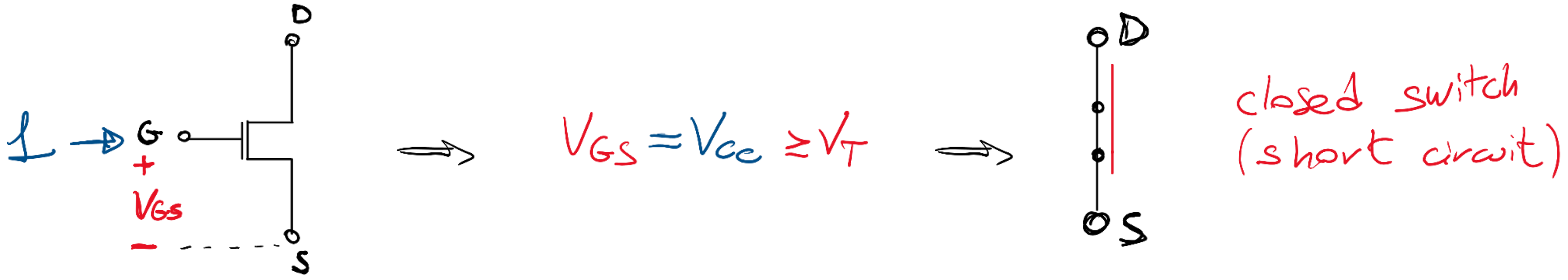
V_T = Threshold voltage

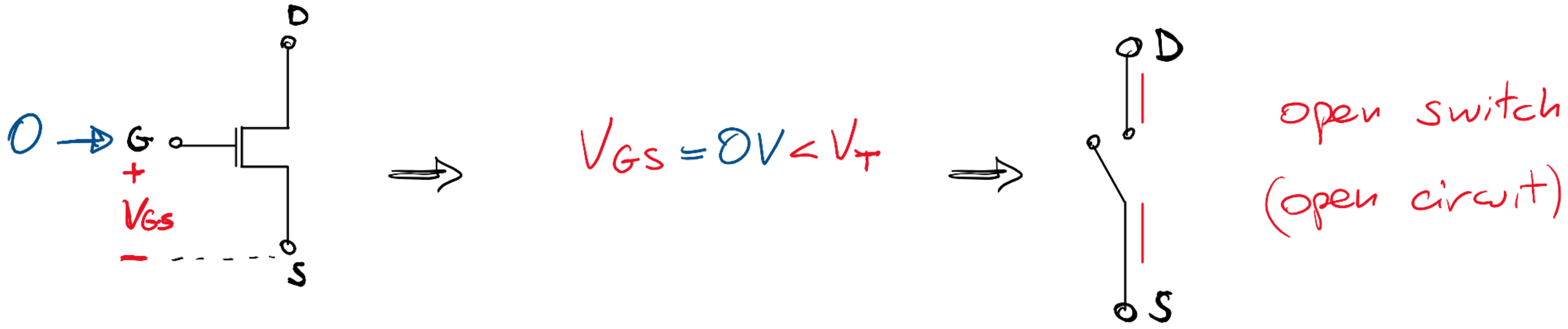
Brief recall on CMOS technologies

- In CMOS circuits
  - V_{CC} is the supply voltage → V_{CC} is the maximum voltage applied to all parts of the circuit
    - At least during the operation of the circuit
    - Some exceptions (e.g., programming, …): will be clearer later
  - **Always**: $\boldsymbol{V_{CC} \geq V_T}$
  - **Always**: $\boldsymbol{0V < V_T}$

Brief recall on CMOS technologies

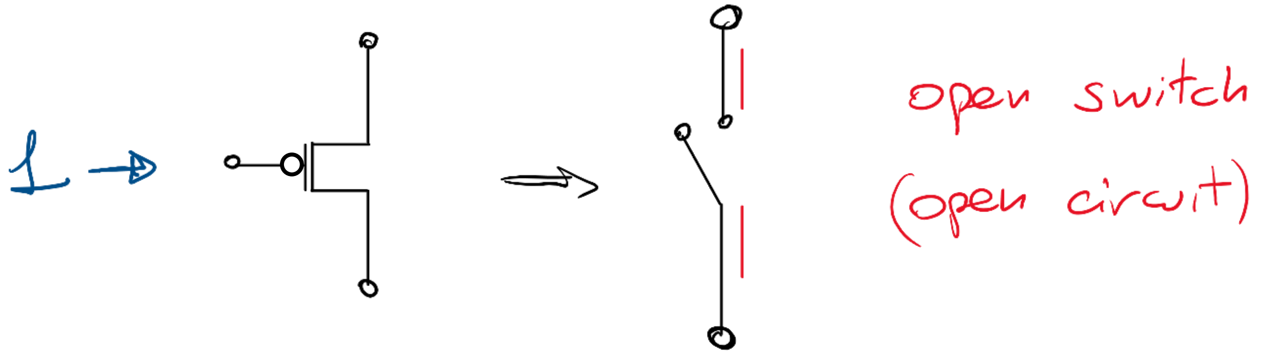
- n-MOS transistor ≈ controlled switch

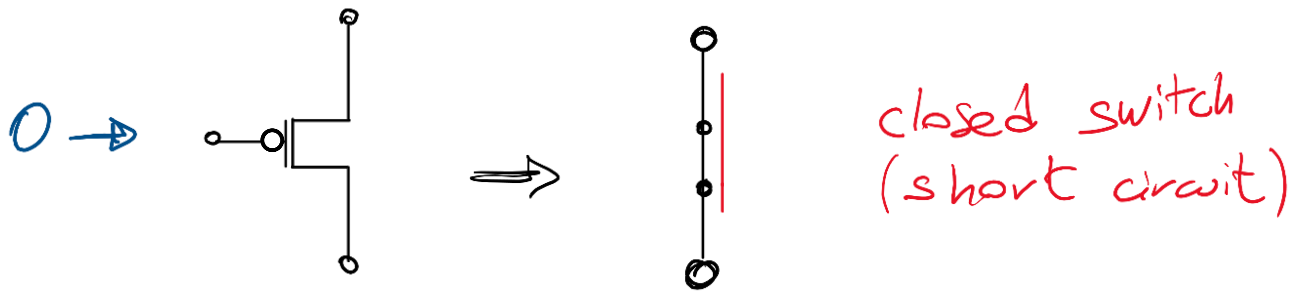




Brief recall on CMOS technologies

- p-MOS transistor ≈ controlled switch (**dual of n-MOS**)



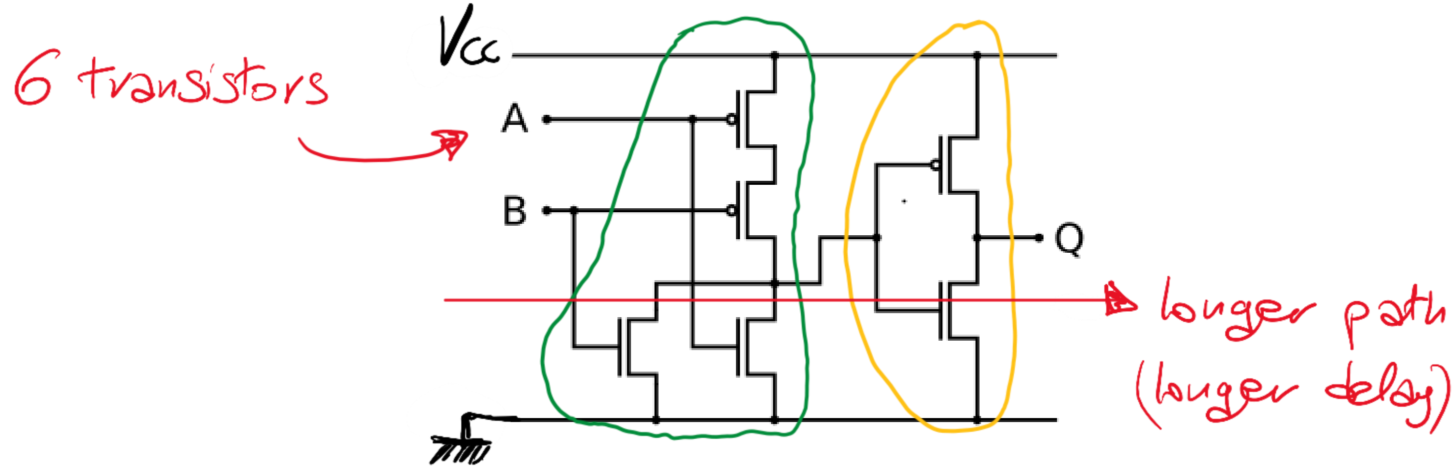


Brief recall on CMOS technologies

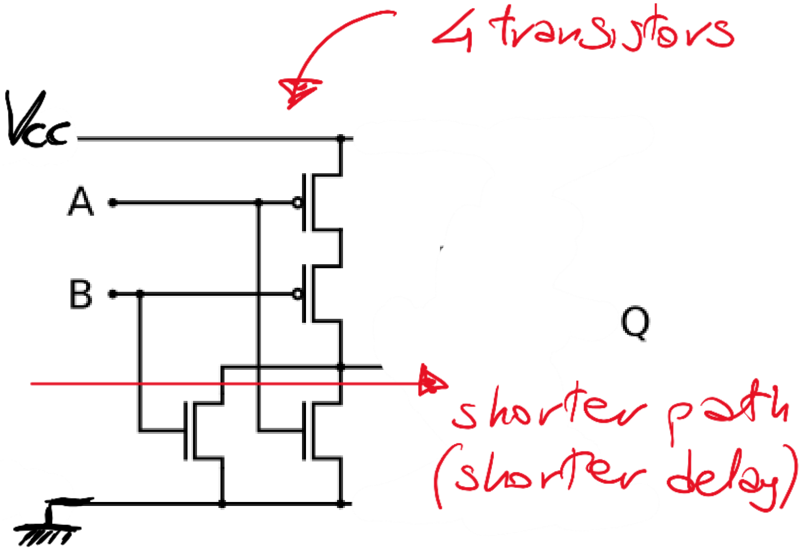
- Active-low signals
  - Definition
    - When 1, the (active-low) signal is disabled
    - When 0, the (active-low) signal is enabled

Brief recall on CMOS technologies

- Active-low signals
  - Why?
    - Active-low signals can
      - Reduce the gate count (so the costs)
      - Improve the circuit "speed" (reduce the delay)



NOR gate = NOR + NOT



NOR gate

Thank you for your attention

Luca Crocetti
(luca.crocetti@unipi.it)