Electronics Systems (938II)

Lecture 3.6

Semiconductor Memories – DRAM, SDRAM, and DDR

RAM – Reminder

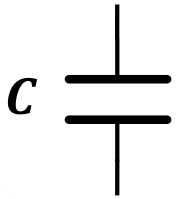
- RAM classification
  - **Static** RAM (**SRAM**)
    - The memory content is hold over the time, as long as the memory is powered
  - **Dynamic** RAM (**DRAM**)
    - Even if the memory is powered, the memory content needs to be refreshed over the time, otherwise it is lost

RAM – Reminder

- RAM classification
  - **Static** RAM (**SRAM**)
    - The memory content is hold over the time, as long as the memory is powered
  - **Dynamic** RAM (**DRAM**)
    - Even if the memory is powered, the memory content needs to be refreshed over the time, otherwise it is lost

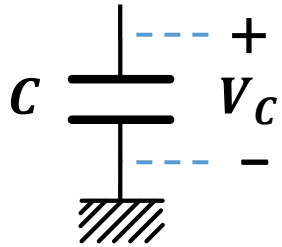
DRAM – Memory cell

- Capacitor



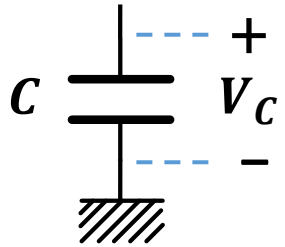
DRAM – Memory cell

- Capacitor
  - **Memory bit = Voltage on capacitor (V_C)**

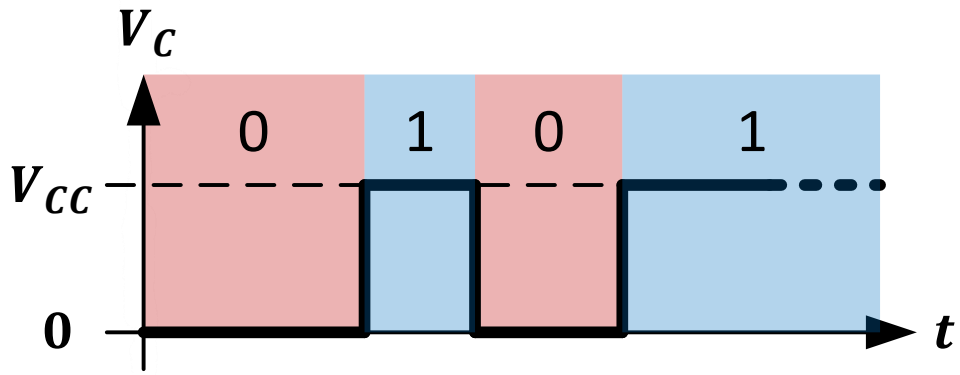


DRAM – Memory cell

- Capacitor
  - **Memory bit = Voltage on capacitor (V_C)**



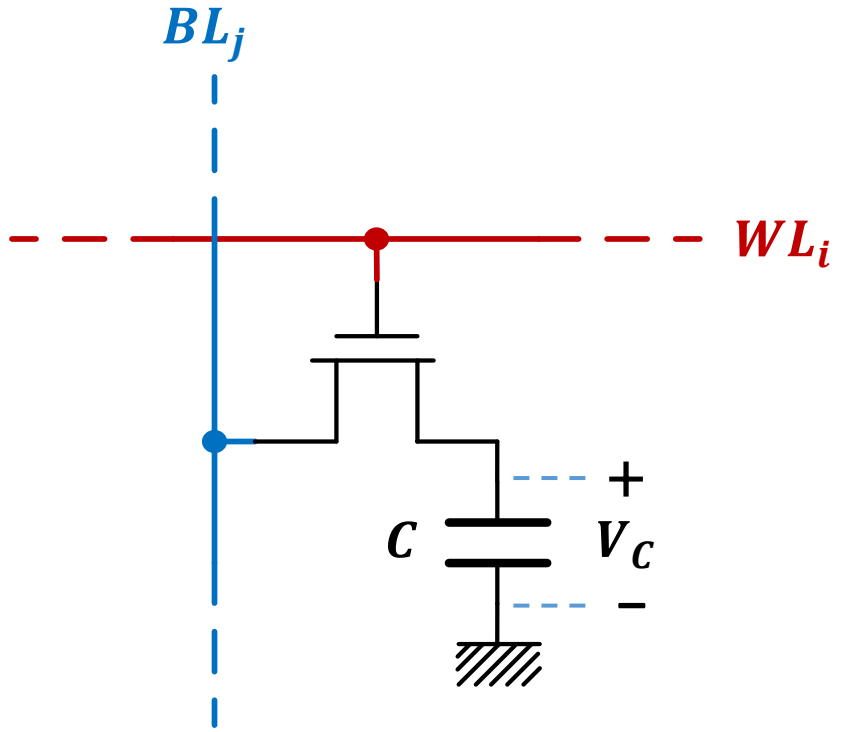
- $V_C = \begin{cases} V_{CC} & (logic\ 1) \\ 0 & (logic\ 0) \end{cases}$



DRAM – Memory cell

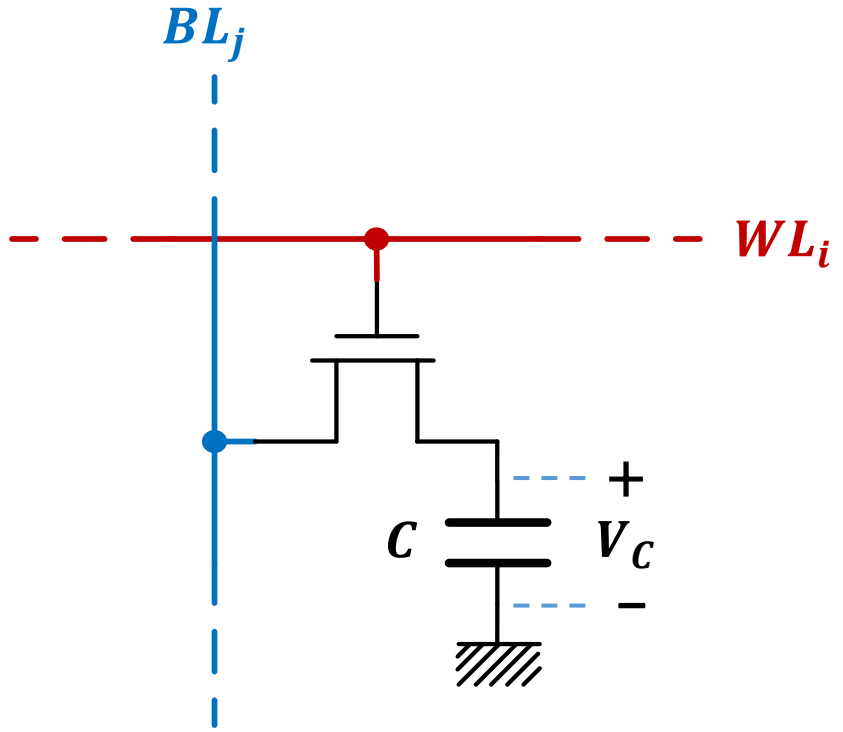
- Architecture
  - Capacitor
  - 1x access transistor

  - WL_i = Word Line
  - BL_j = Bit Line



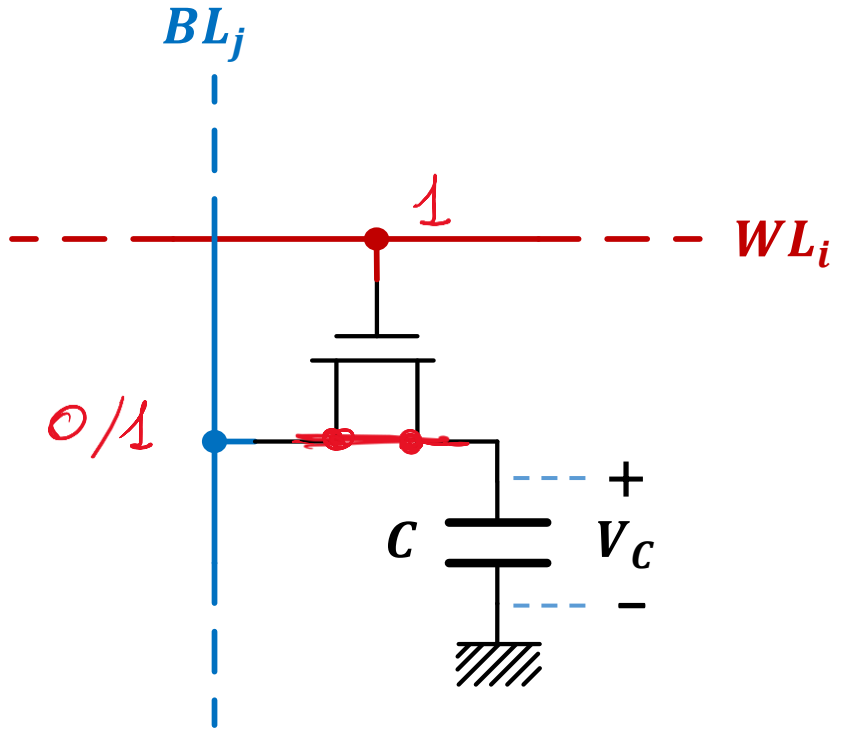
DRAM – Memory cell

- Working principle



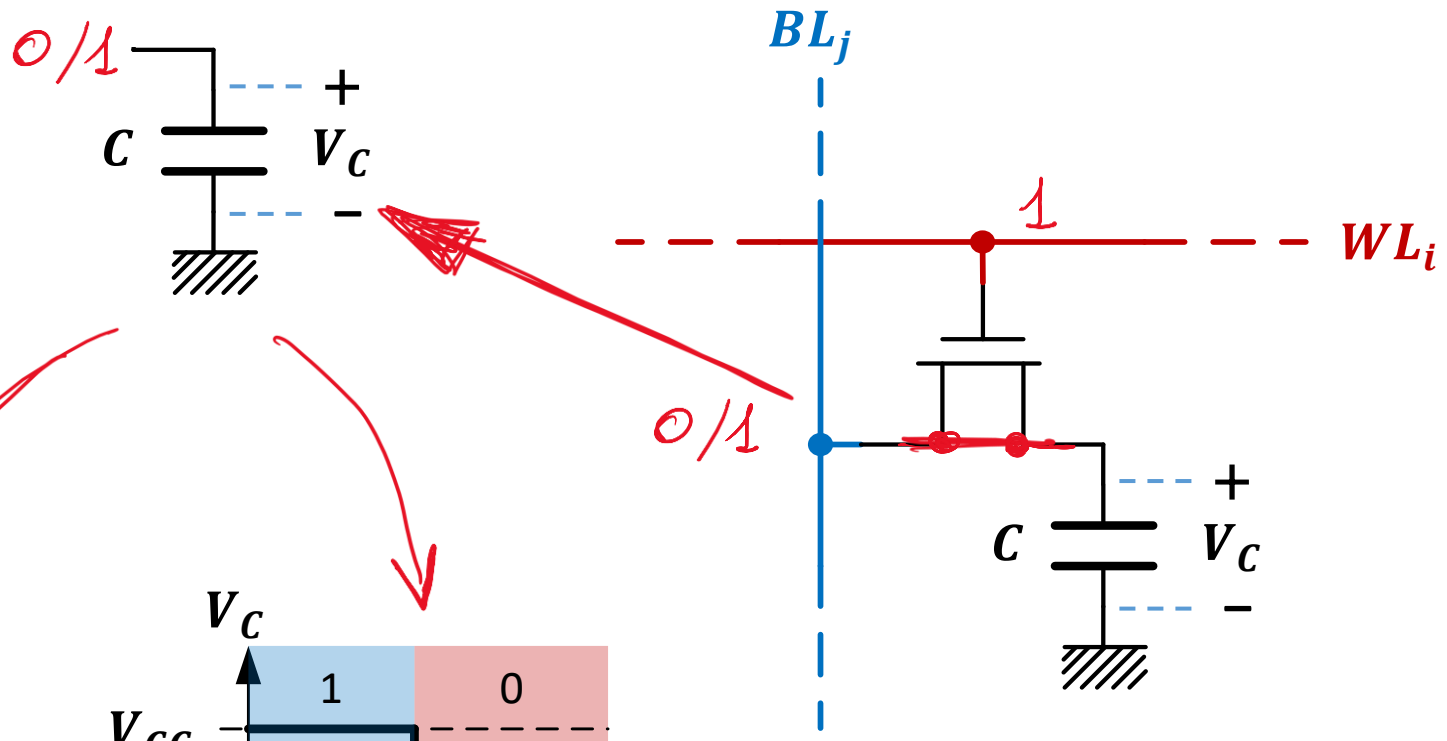
DRAM – Memory cell

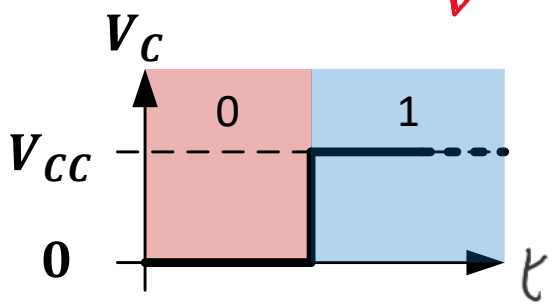
- Working principle
  - **Write**
    - $WL_i = 1$
    - $BL_i = D$
      - $0\ V$ (logic 0)
      - V_{CC} (logic 1)



DRAM – Memory cell

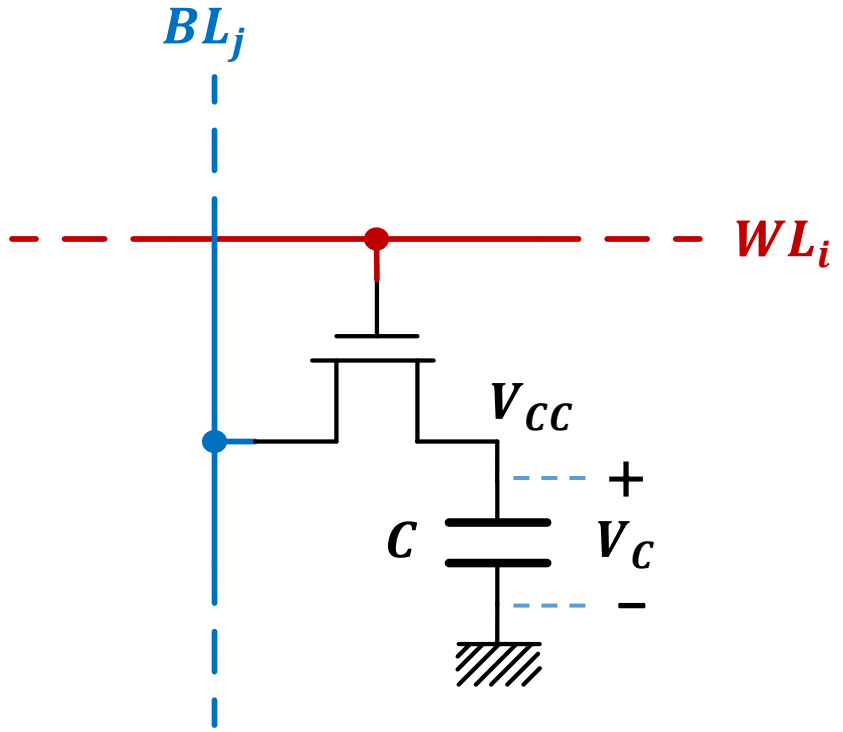
- Working principle
  - **Write**
    - $WL_i = 1$
    - $BL_i = D$
      - $0\ V$ (logic 0)
      - V_{CC} (logic 1)





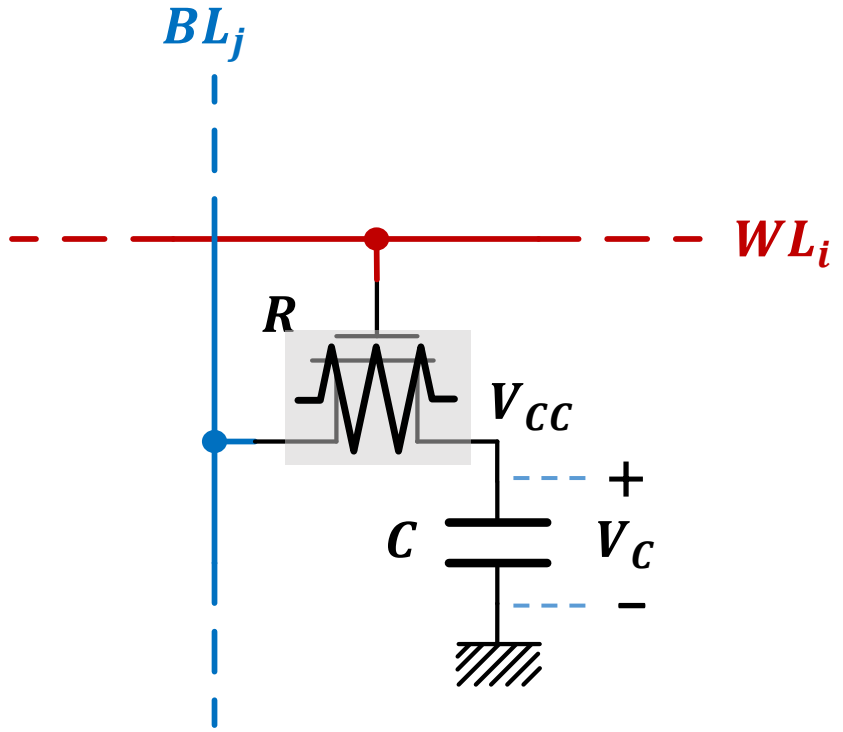
DRAM – Memory cell

- Working principle
  - **After write**
    - $WL_i = 0$
    - **Assume 1 (V_{CC}) written in memory cell / on capacitor**



DRAM – Memory cell

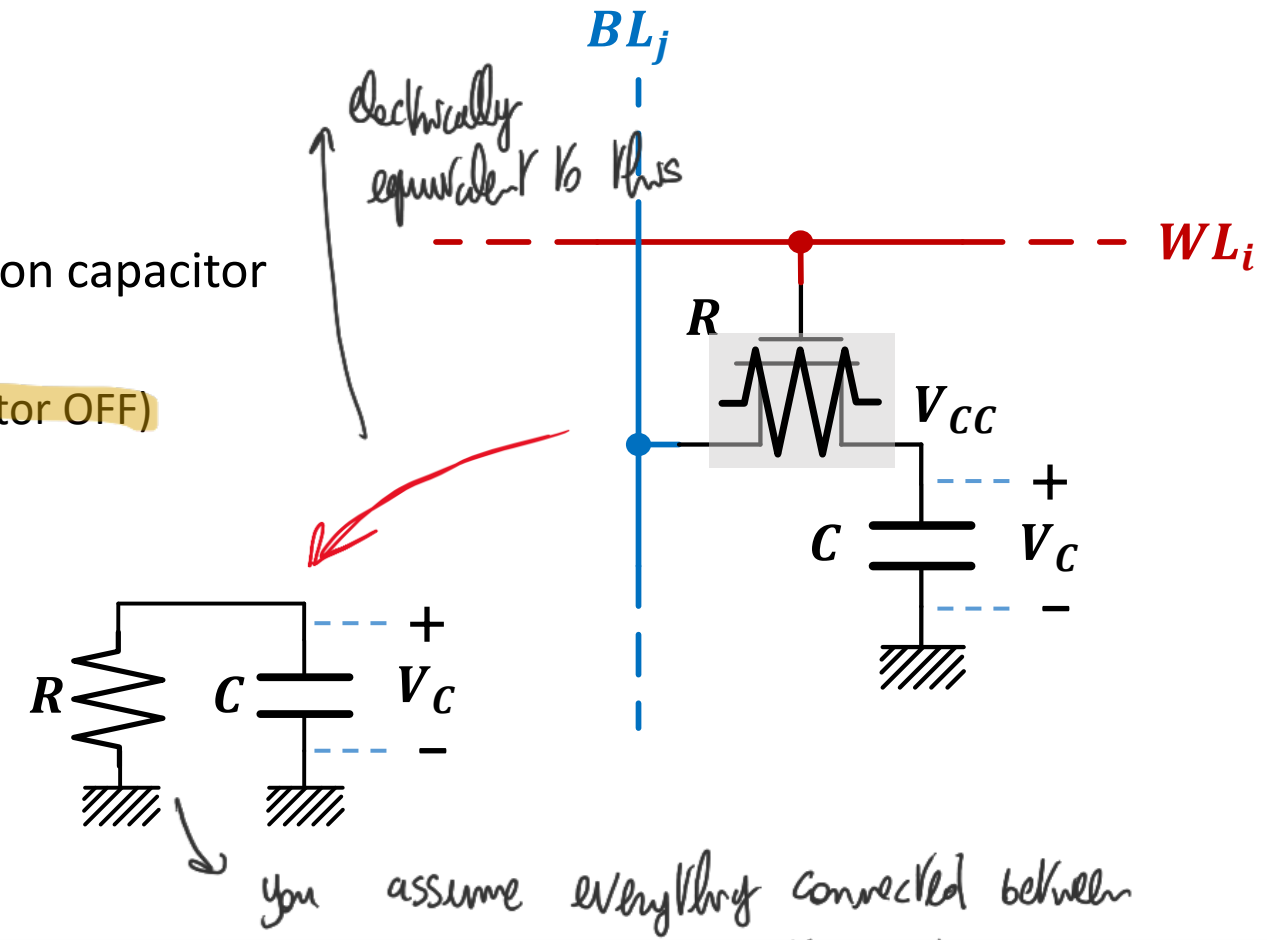
- Working principle
  - **After write**
    - $WL_i = 0$
    - Assume 1 (V_{CC}) written in memory cell / on capacitor
    - **Access transistor ≈ resistor (R)**
      - High impedance when $WL_i = 0$ (transistor OFF)



DRAM – Memory cell

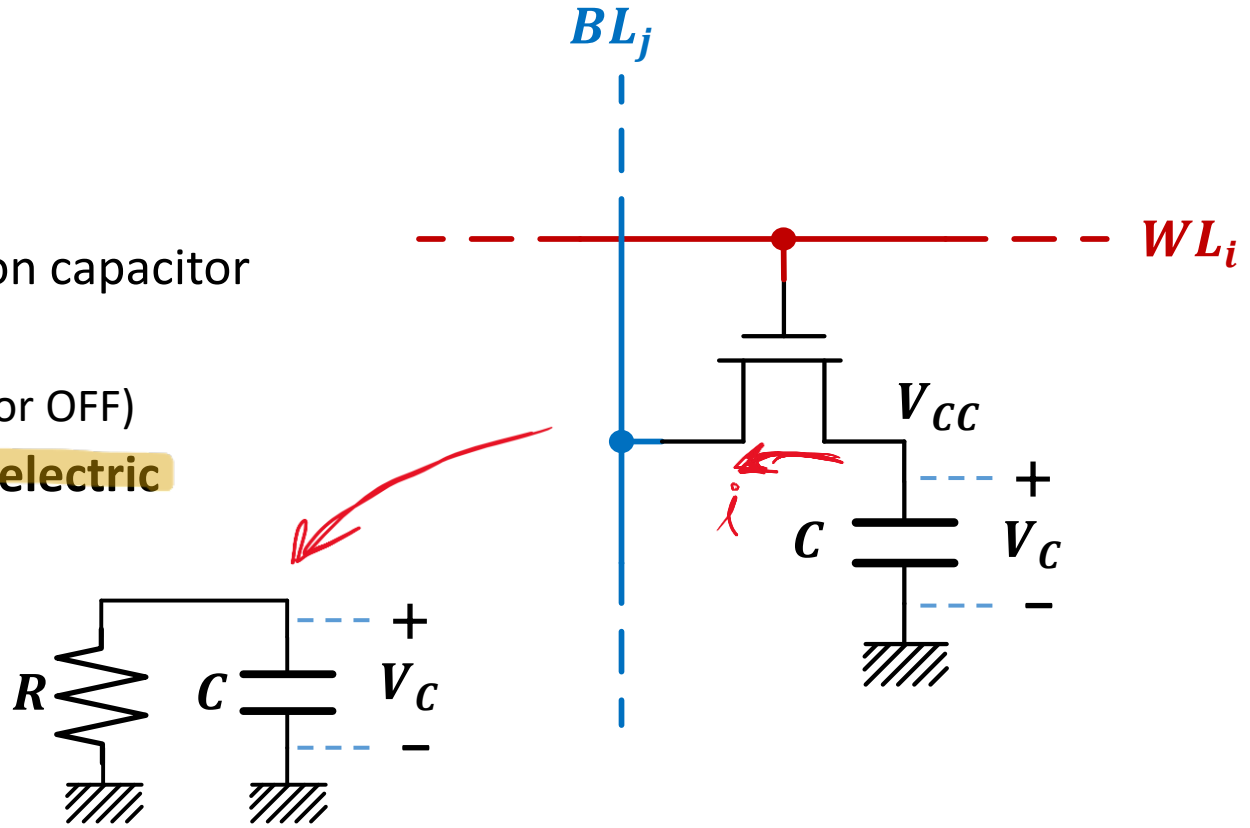
- Working principle
  - **After write**
    - $WL_i = 0$
    - Assume 1 (V_{CC}) written in memory cell / on capacitor
    - **Access transistor ≈ resistor (R)**
      - High impedance when $WL_i = 0$ (transistor OFF)

So we can assume R is very high.



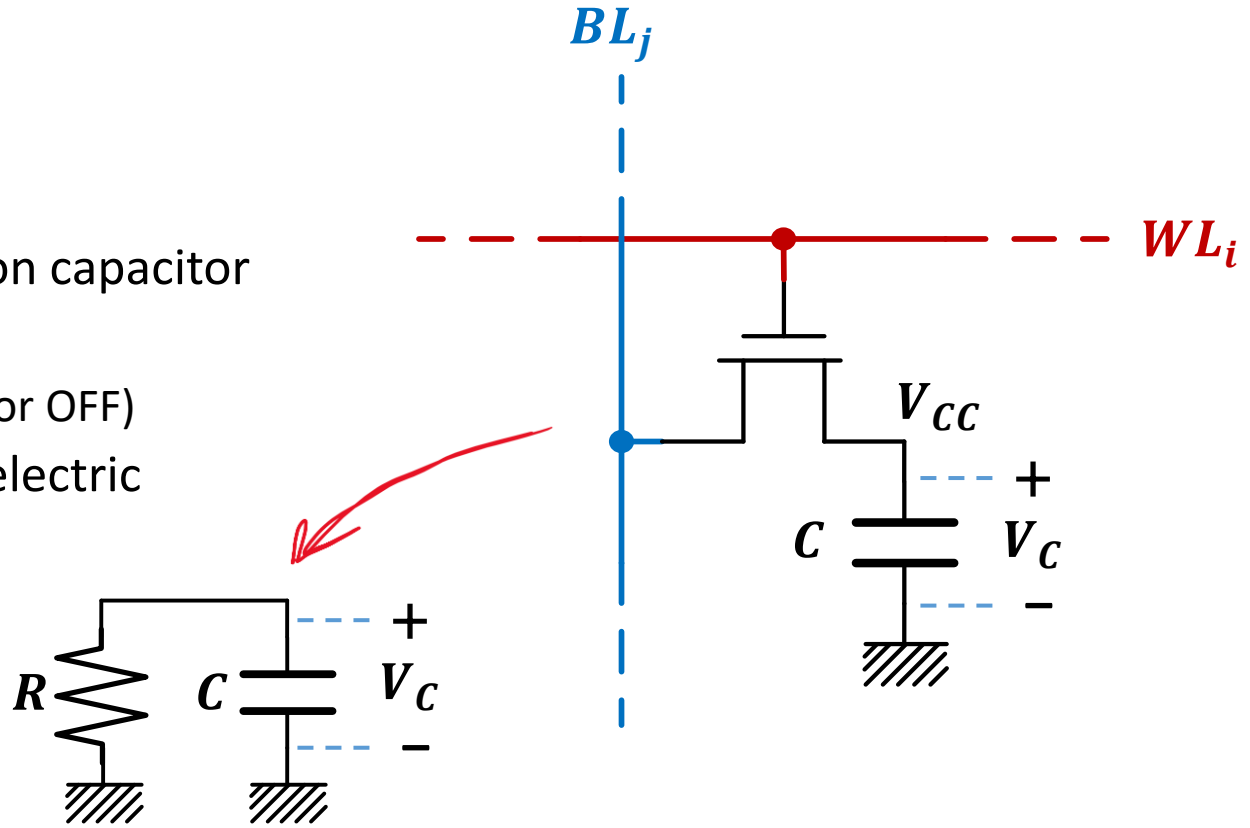
DRAM – Memory cell

- Working principle
  - **After write**
    - $WL_i = 0$
    - Assume 1 (V_{CC}) written in memory cell / on capacitor
    - Access transistor ≈ resistor (R)
      - High impedance when $WL_i = 0$ (transistor OFF)
    - **Leakage (current, i) through capacitor dielectric**
      - Not ideal!



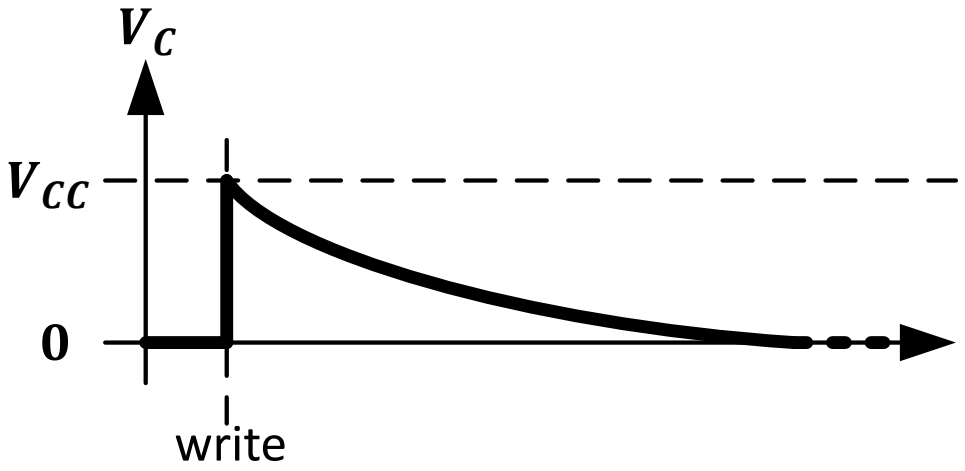
DRAM – Memory cell

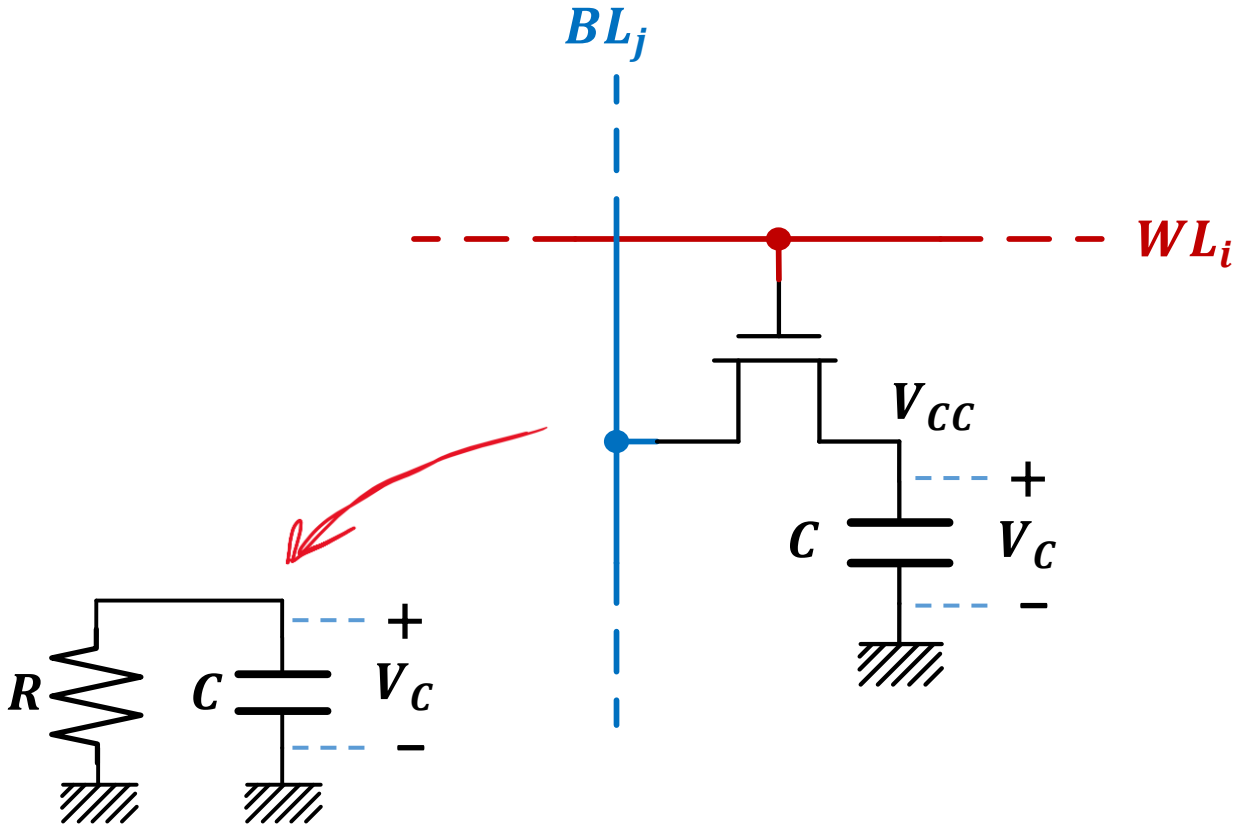
- Working principle
  - **After write**
    - $WL_i = 0$
    - Assume 1 (V_{CC}) written in memory cell / on capacitor
    - Access transistor ≈ resistor (R)
      - High impedance when $WL_i = 0$ (transistor OFF)
    - Leakage (current, i) through capacitor dielectric
      - Not ideal!
    - **C discharges!!!**



DRAM – Memory cell

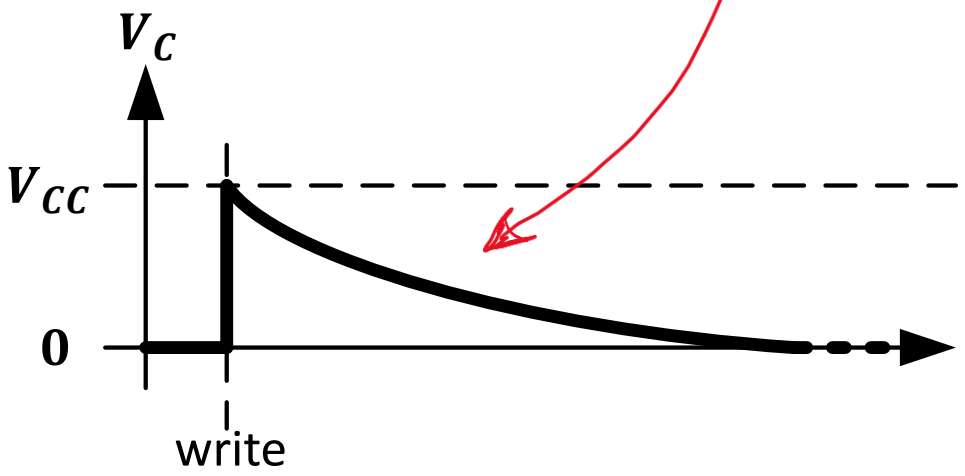
- Working principle
  - **After write**
    - **C discharges!!!**

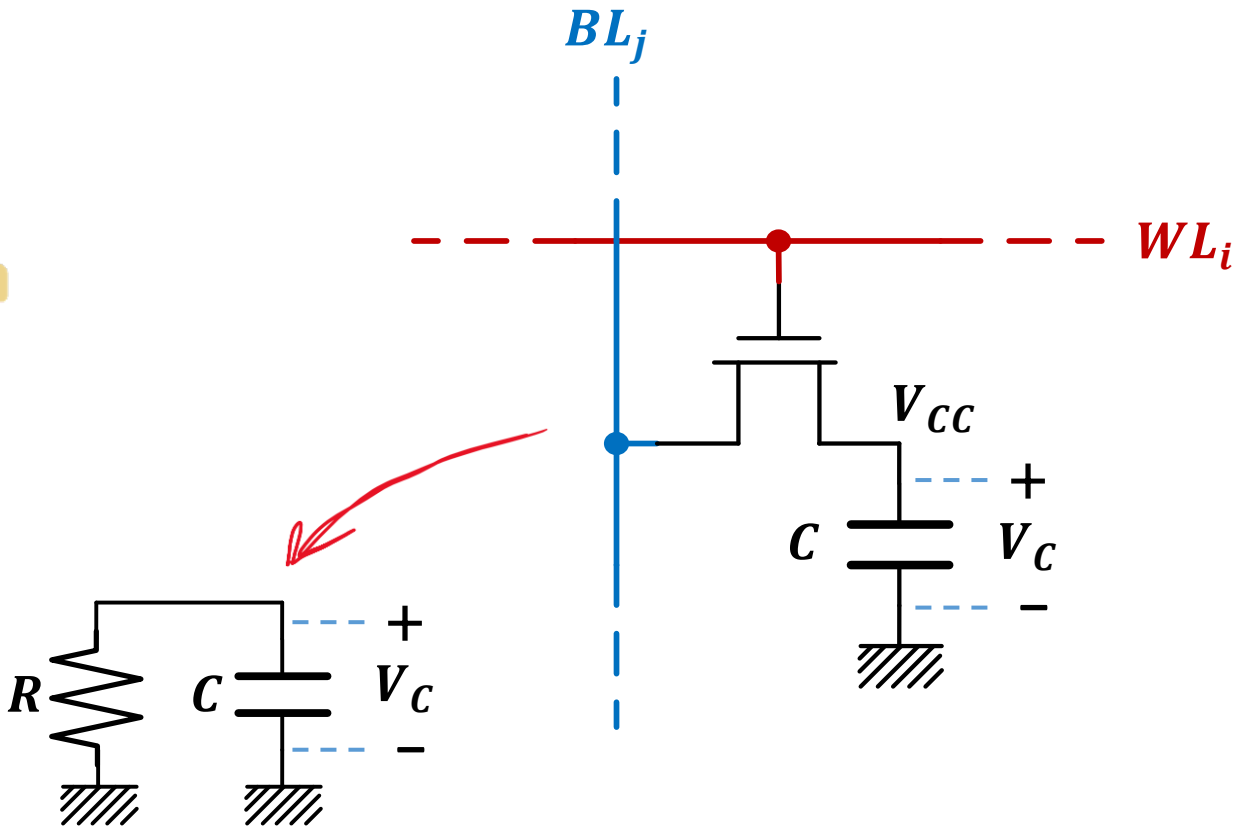




DRAM – Memory cell

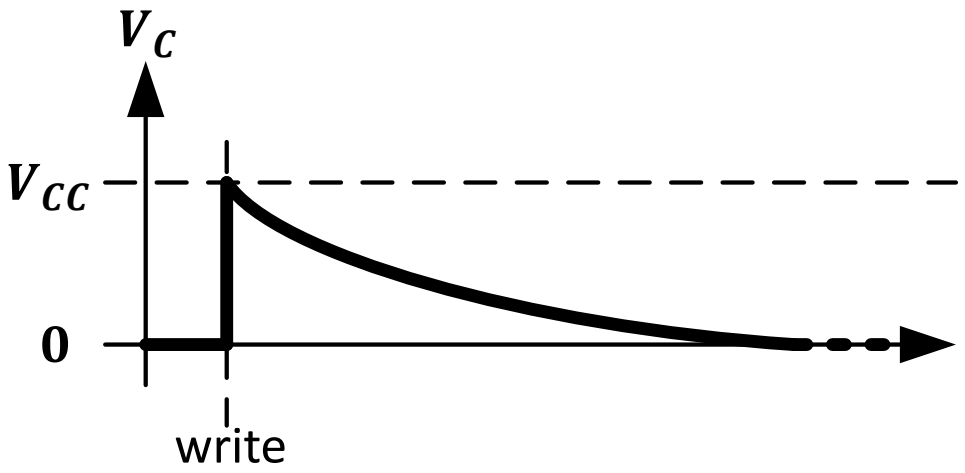
- Working principle
  - **After write**
    - **C discharges!!!**
      - Proportional to $R \cdot C = \tau$ (time constant)

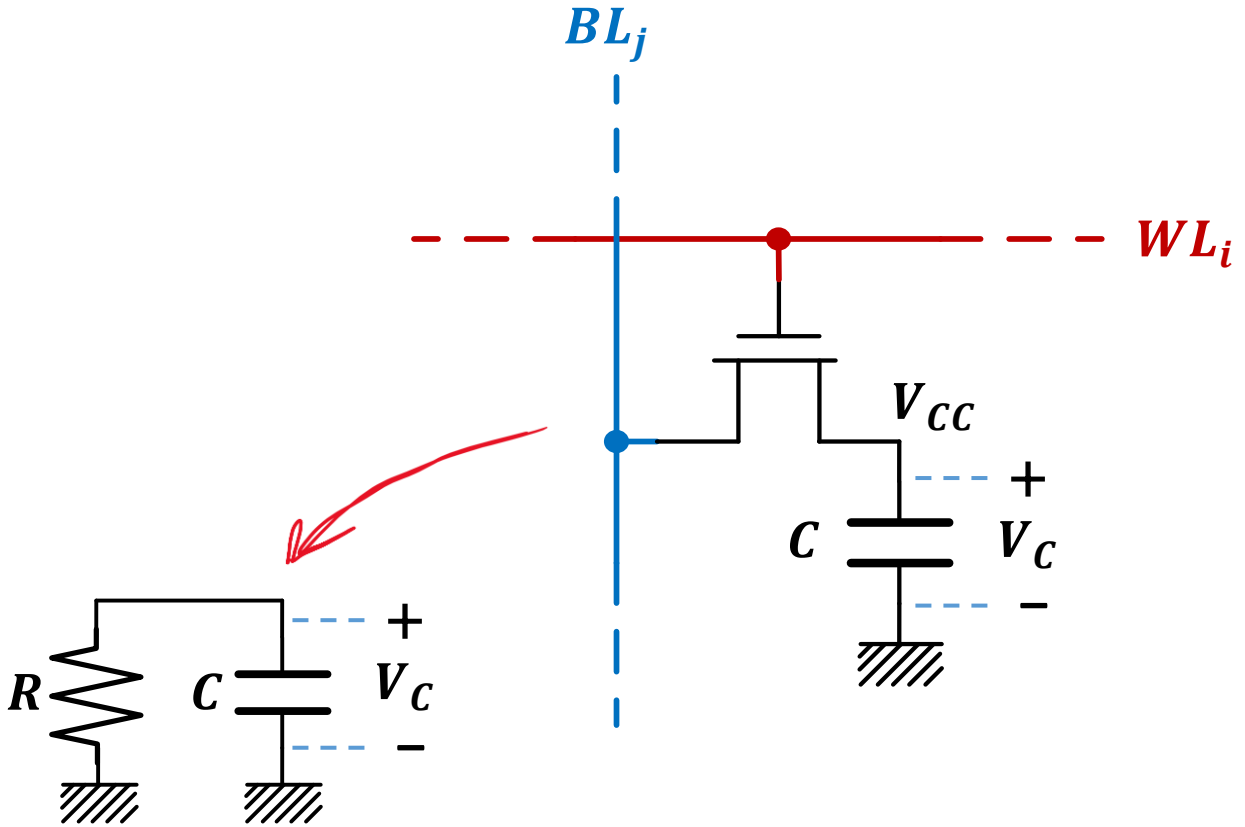




DRAM – Memory cell

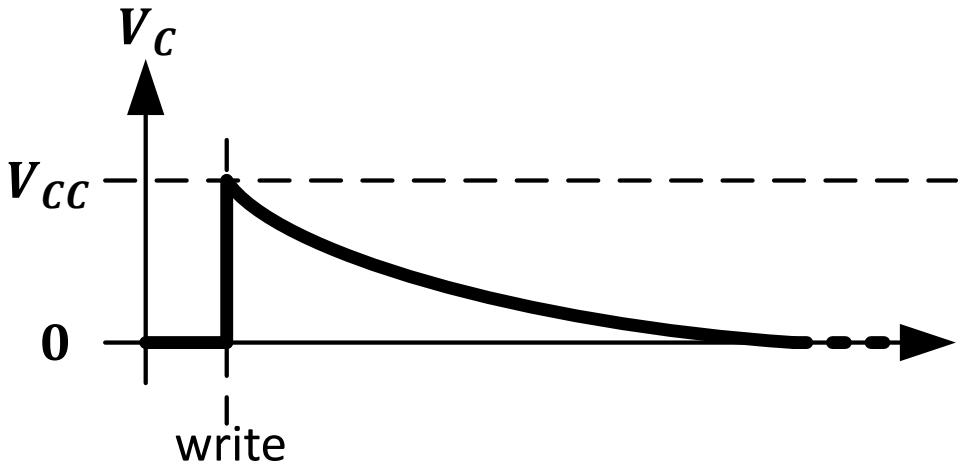
- Working principle
  - **After write**
    - **C discharges!!!**
      - Proportional to $\boldsymbol{R \cdot C = \tau}$ (time constant)
      - $\tau = 10 \div 100\ ms$

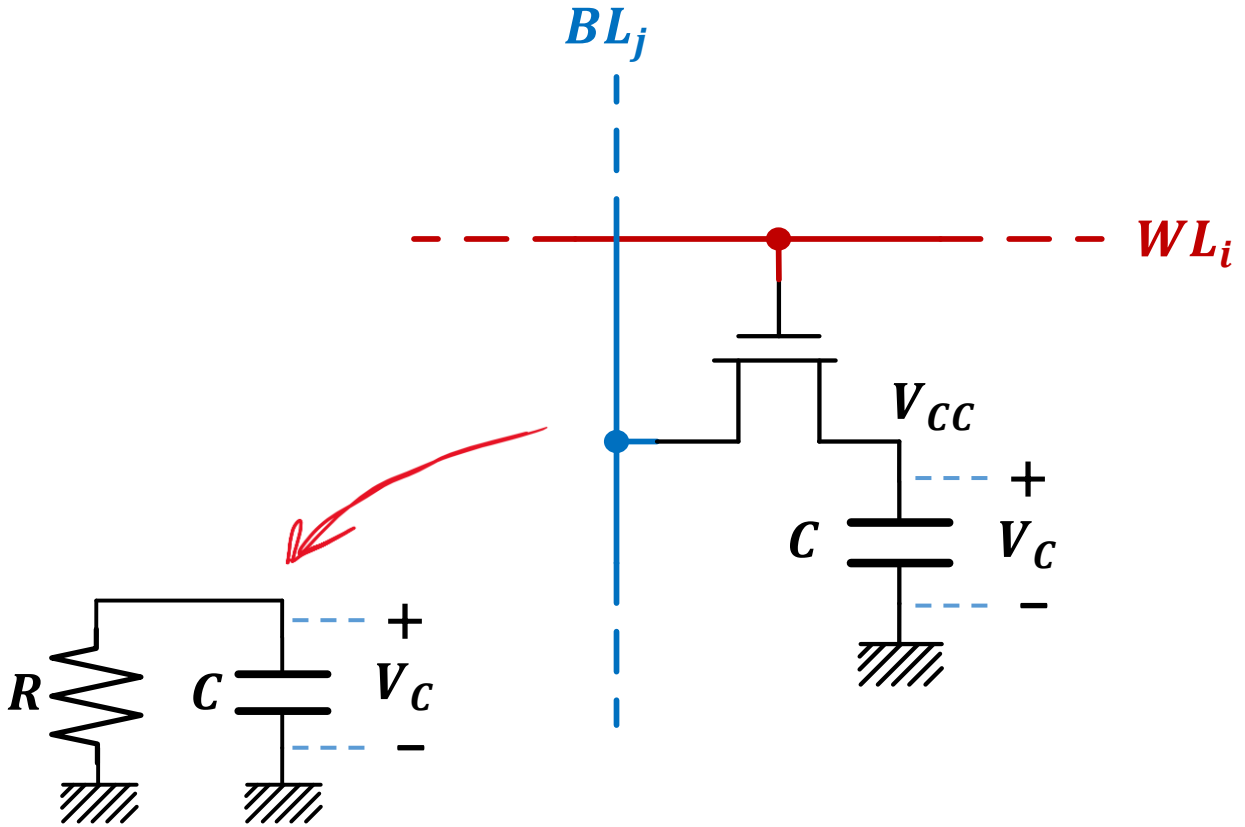




DRAM – Memory cell

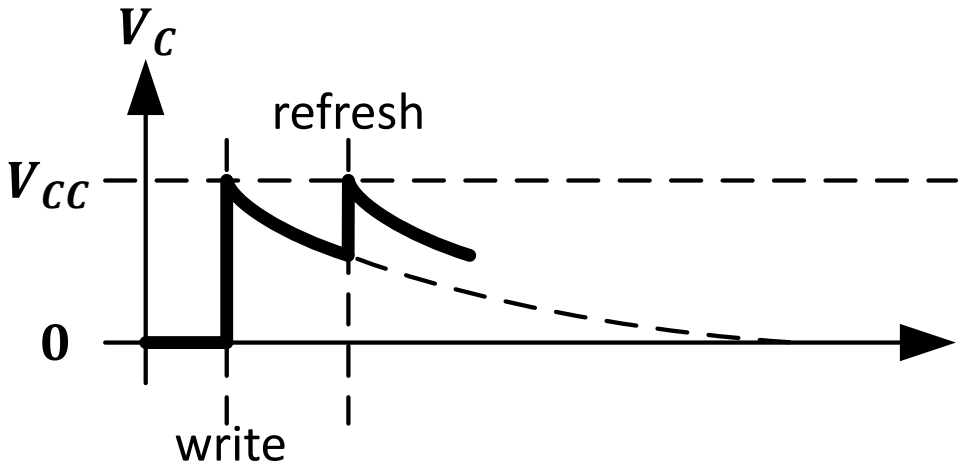
- Working principle
  - **After write**
    - **C discharges!!!**
      - Proportional to $\boldsymbol{R \cdot C = \tau}$ (time constant)
      - $\tau = 10 \div 100\ ms$
    - **Need of refresh**

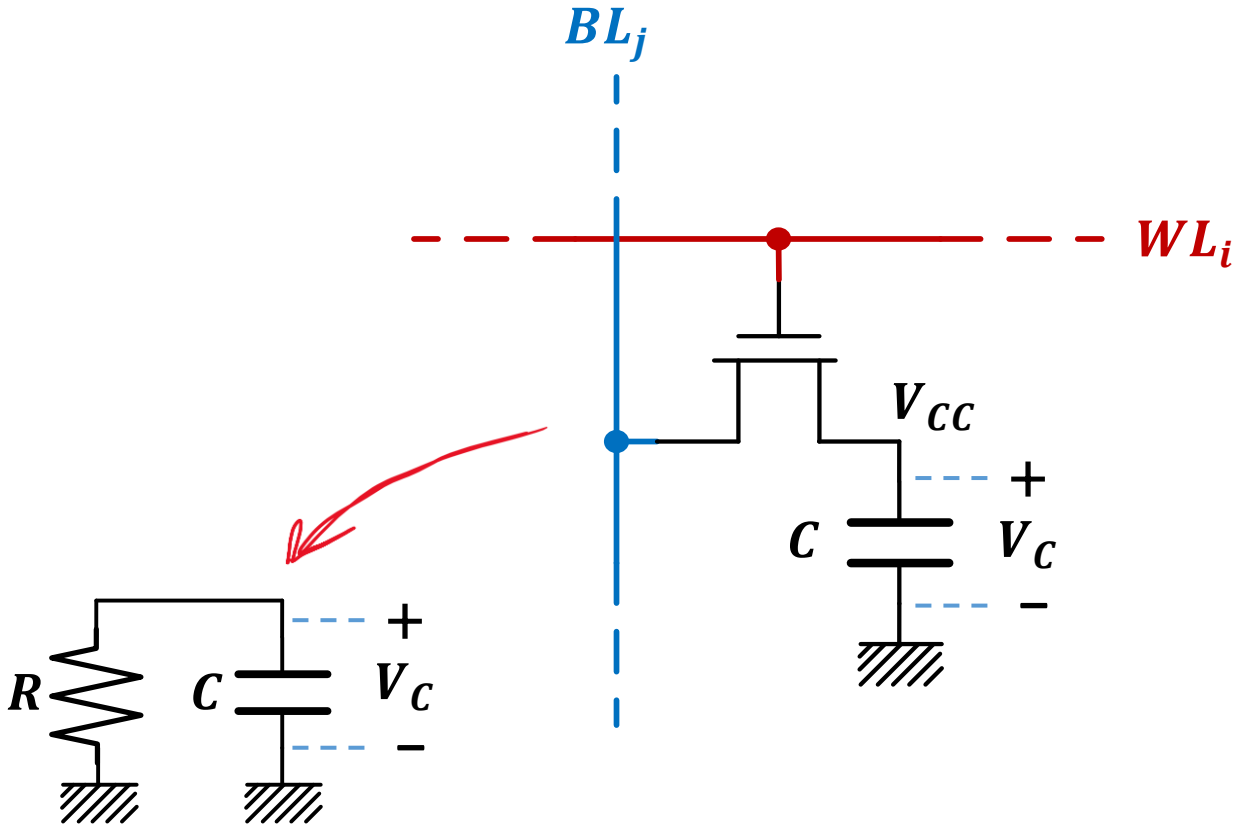




DRAM – Memory cell

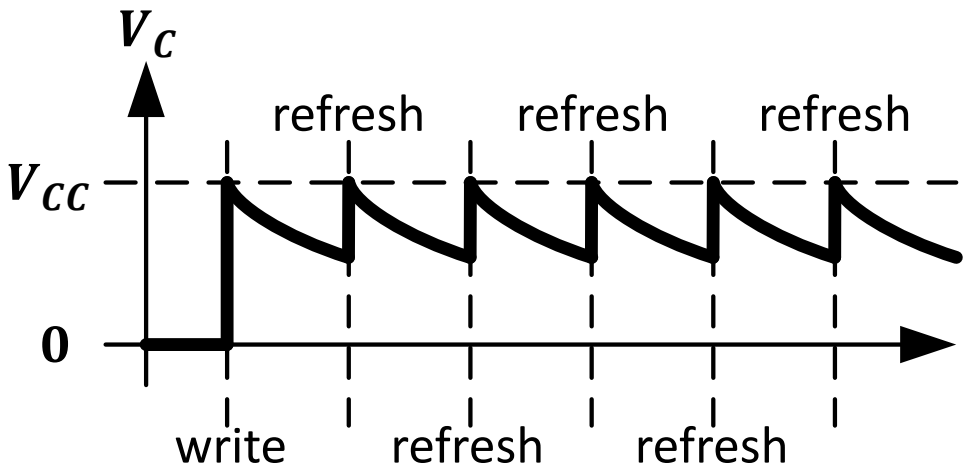
- Working principle
  - **After write**
    - **C discharges!!!**
      - Proportional to $\boldsymbol{R \cdot C = \tau}$ (time constant)
      - $\tau = 10 \div 100\ ms$
    - **Need of refresh**

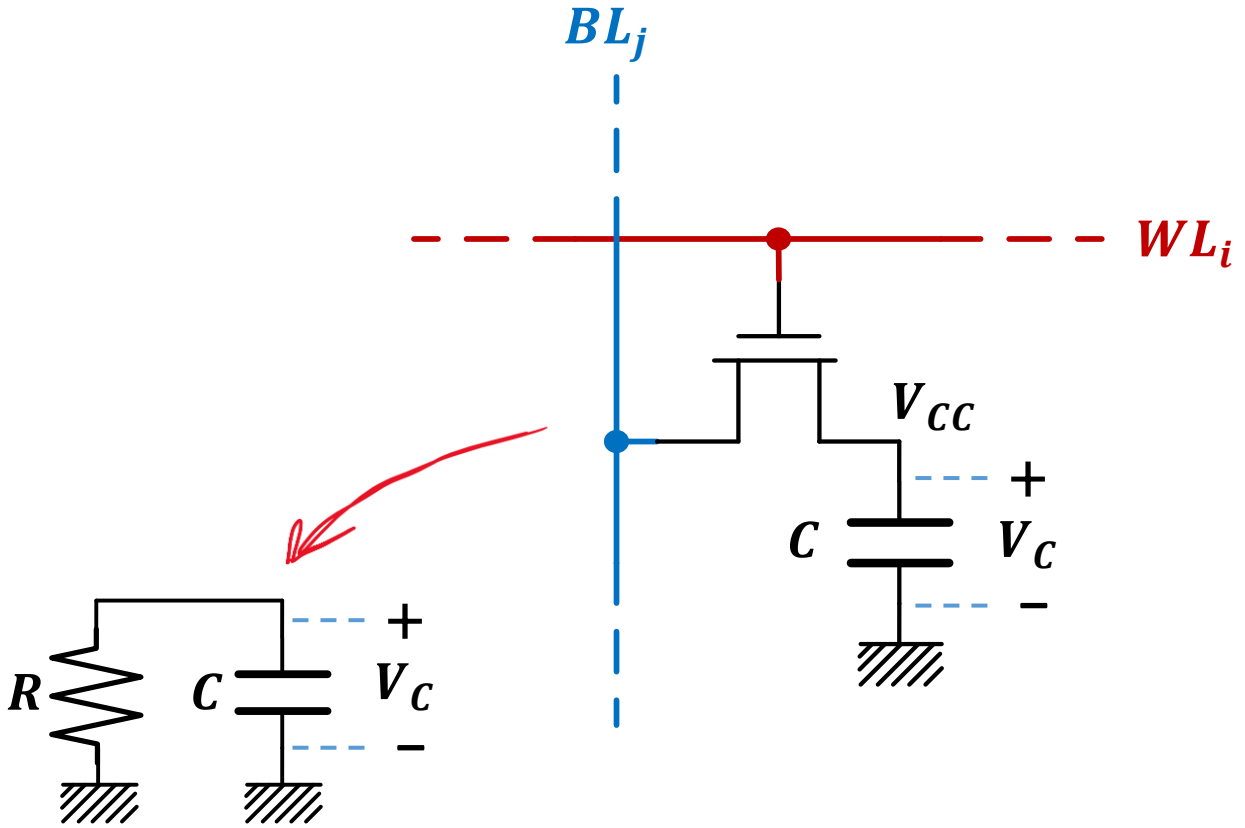




DRAM – Memory cell

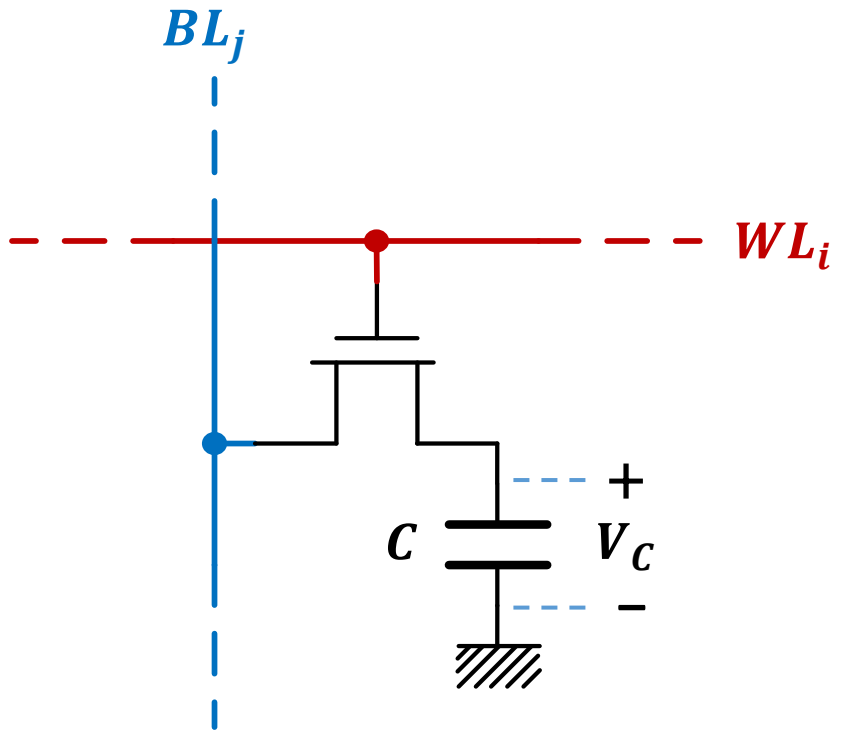
- Working principle
  - **After write**
    - **C discharges!!!**
      - Proportional to $\boldsymbol{R \cdot C = \tau}$ (time constant)
      - $\tau = 10 \div 100\ ms$
    - **Need of refresh**





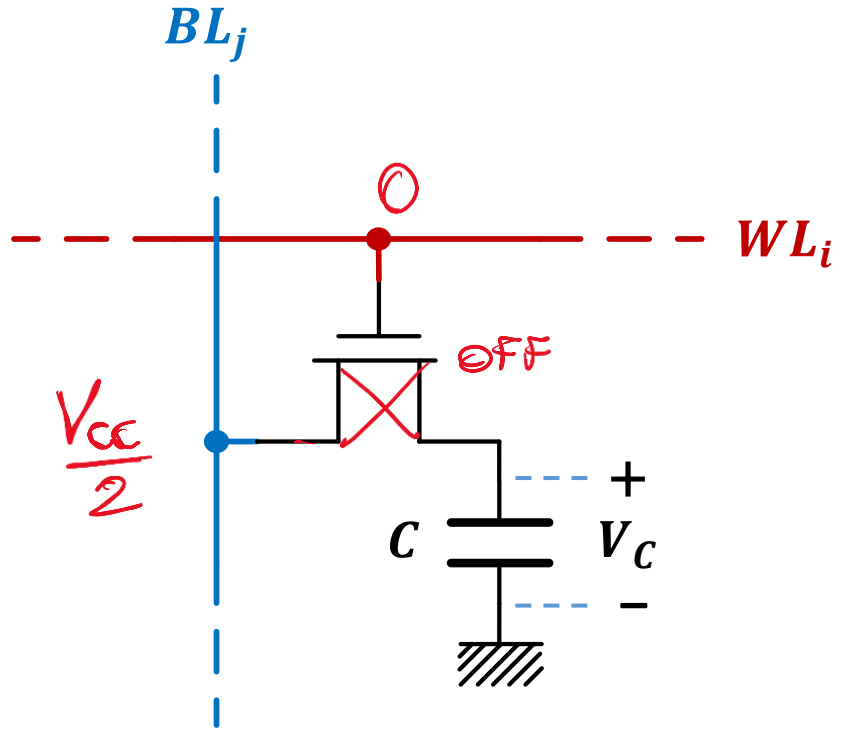
DRAM – Memory cell

- Working principle
  - Read



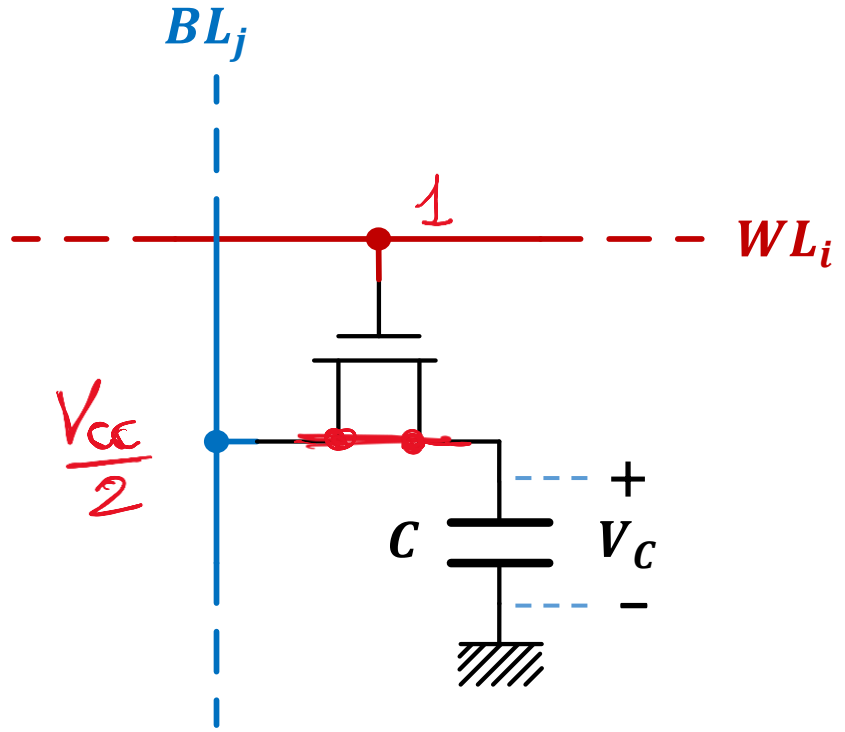
DRAM – Memory cell

- Working principle
  - **Read**
    - **Step 1**
      - $WL_i = 0$
      - $BL_j = \frac{V_{CC}}{2} = V_R$ (reading voltage)



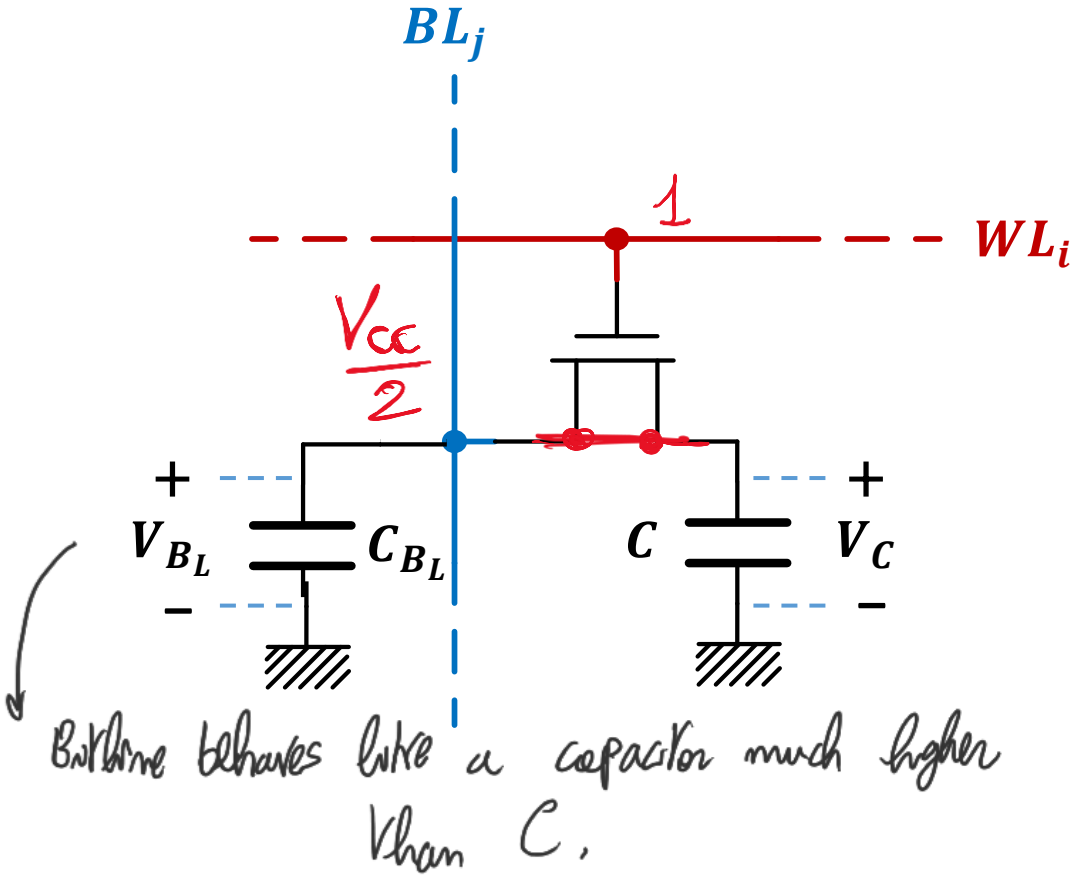
DRAM – Memory cell

- Working principle
  - **Read**
    - Step 1
      - $WL_i = 0$
      - $BL_j = \frac{V_{CC}}{2} = V_R$
    - **Step 2**
      - $WL_i = 1$



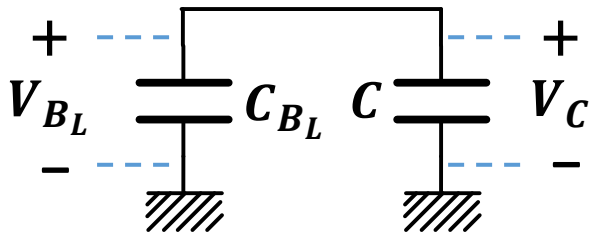
DRAM – Memory cell

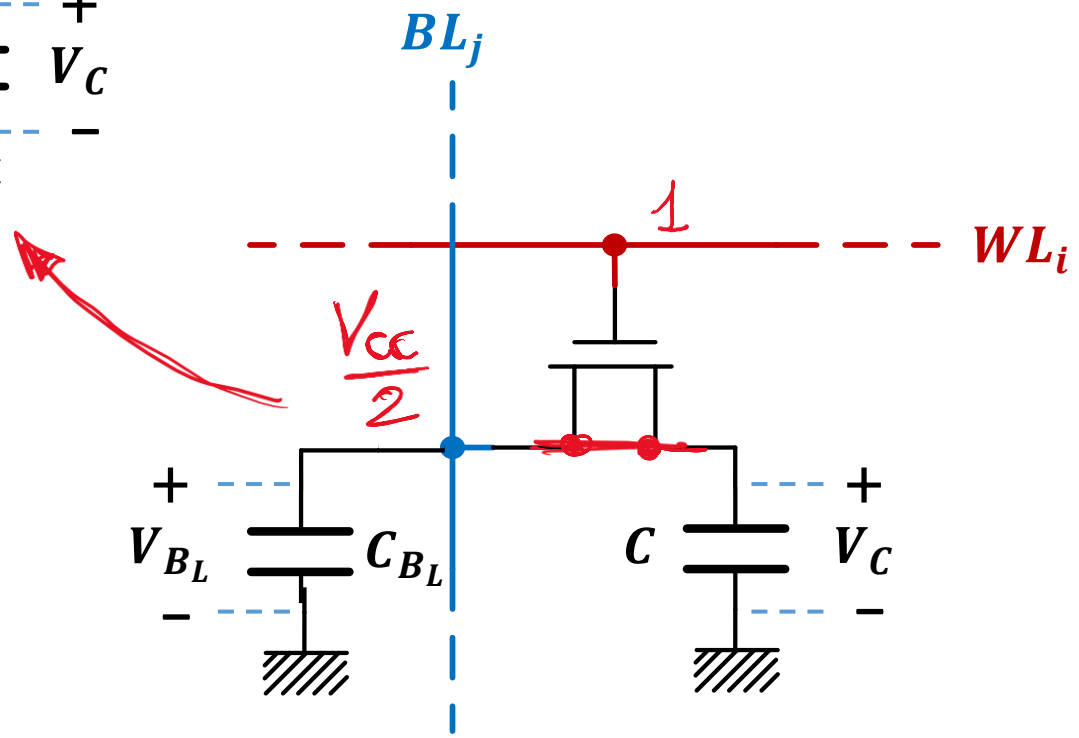
- Working principle
  - **Read**
    - Step 1
      - $WL_i = 0$
      - $BL_j = \frac{V_{CC}}{2} = V_R$
    - **Step 2**
      - $WL_i = 1$
      - $BL_j \approx C_{B_L}$ **(capacitor)**
        - $C_{B_L} \gg C$
        - V_{B_L} = voltage on $C_{B_L} = \frac{V_{CC}}{2}$



DRAM – Memory cell

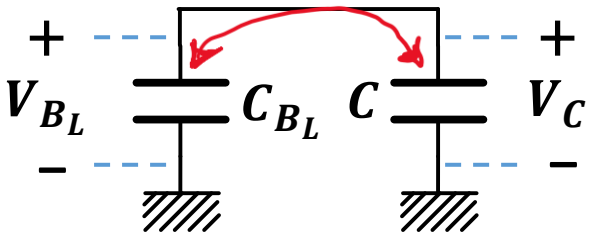
- Working principle
  - **Read**
    - Step 1
      - $WL_i = 0$
      - $BL_j = \frac{V_{CC}}{2} = V_R$
    - **Step 2**
      - $WL_i = 1$
      - $\boldsymbol{BL_j \approx C_{B_L}}$ **(capacitor)**
        - $C_{B_L} \gg C$
        - V_{B_L} = voltage on C_{B_L}

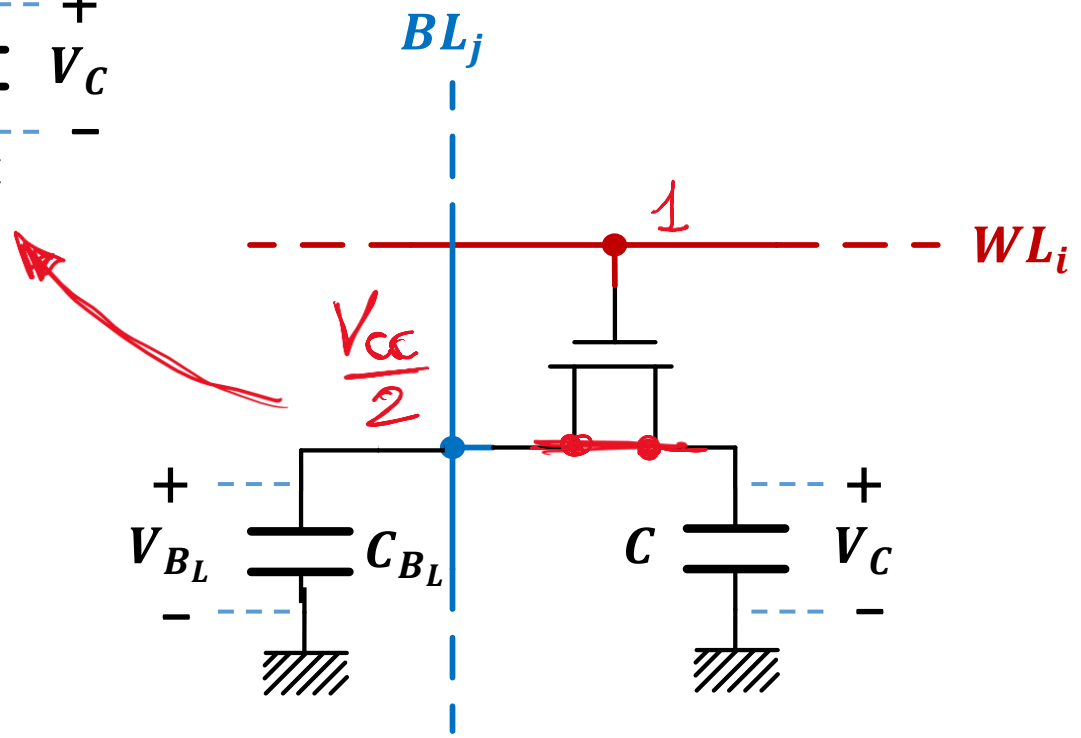




DRAM – Memory cell

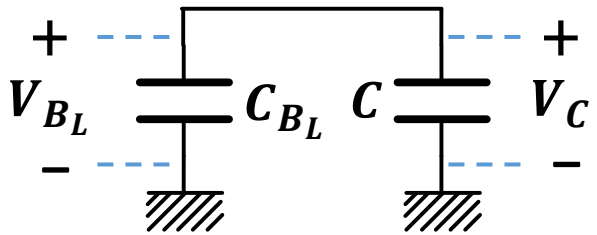
- Working principle
  - **Read**
    - Step 1
      - $WL_i = 0$
      - $BL_j = \frac{V_{CC}}{2} = V_R$
    - **Step 2**
      - $WL_i = 1$
      - $BL_j \approx C_{B_L}$ (capacitor)
        - $C_{B_L} \gg C$
        - V_{B_L} = voltage on C_{B_L}
      - **Charge redistribution between C_{B_L} and C !**





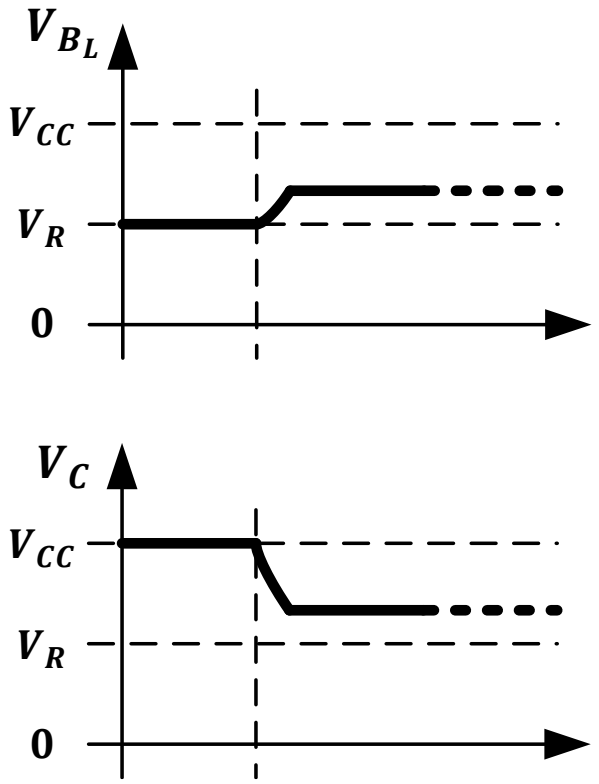
DRAM – Memory cell

- Working principle
  - **Read**
    - Charge redistribution!

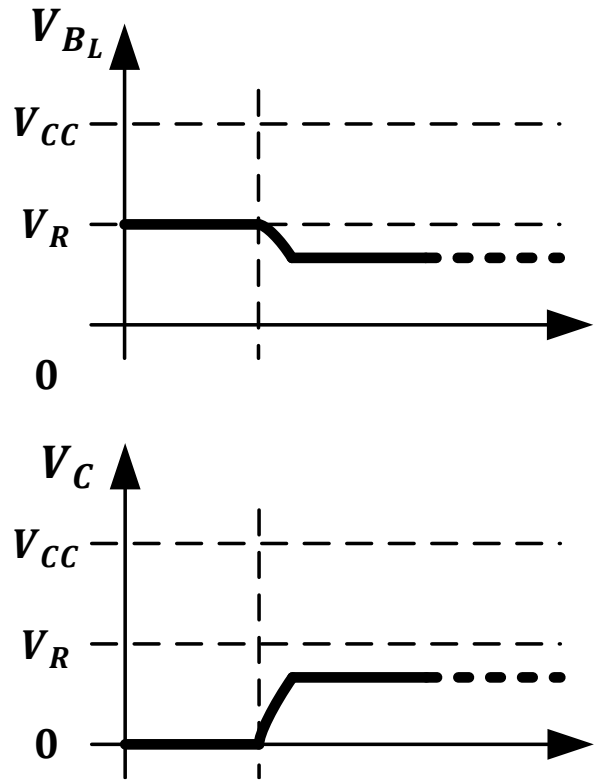


    - **Read destroys data!**
    - **Need of re-write!**

Memory bit = 1 ($V_C = V_{CC}$)

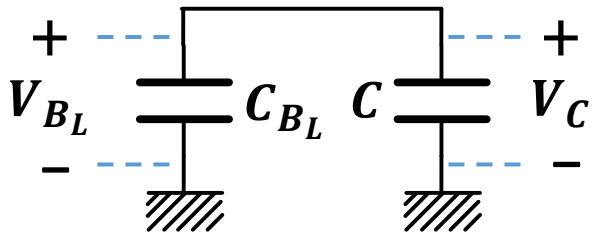


Memory bit = 0 ($V_C = 0\ V$)



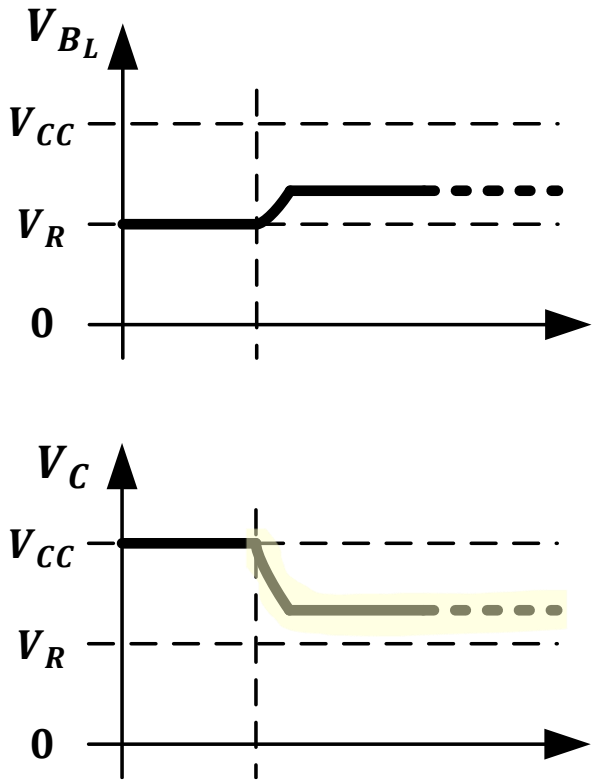
DRAM – Memory cell

- Working principle
  - **Read**
    - Charge redistribution!

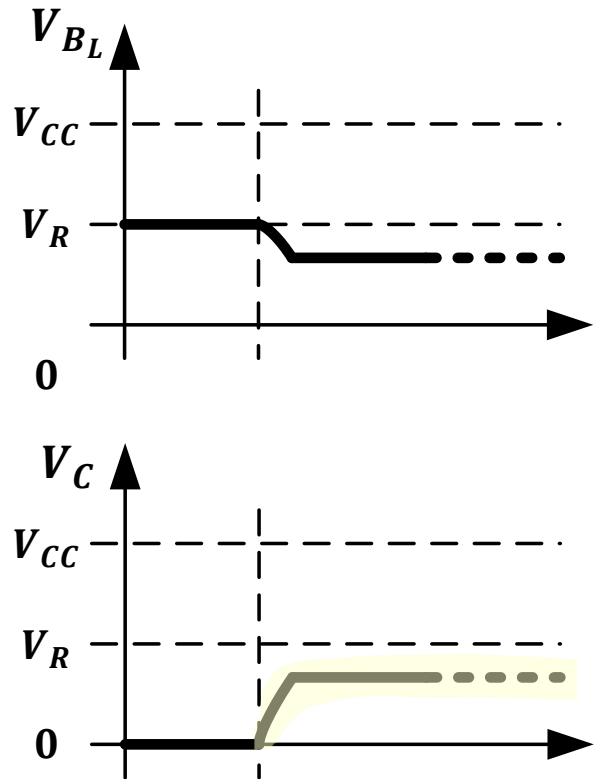


    - Read destroys data!

Memory bit = 1 ($V_C = V_{CC}$)

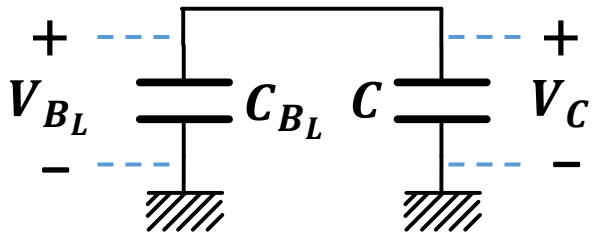


Memory bit = 0 ($V_C = 0\ V$)



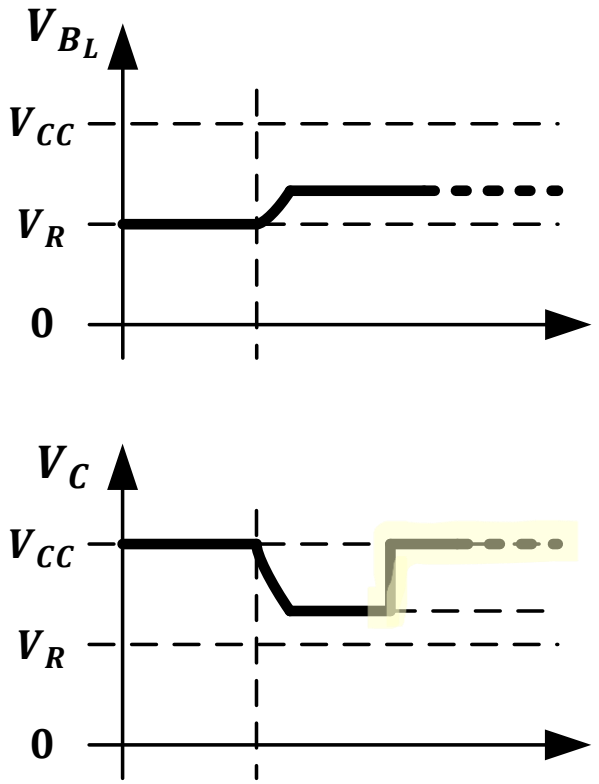
DRAM – Memory cell

- Working principle
  - **Read**
    - Charge redistribution!

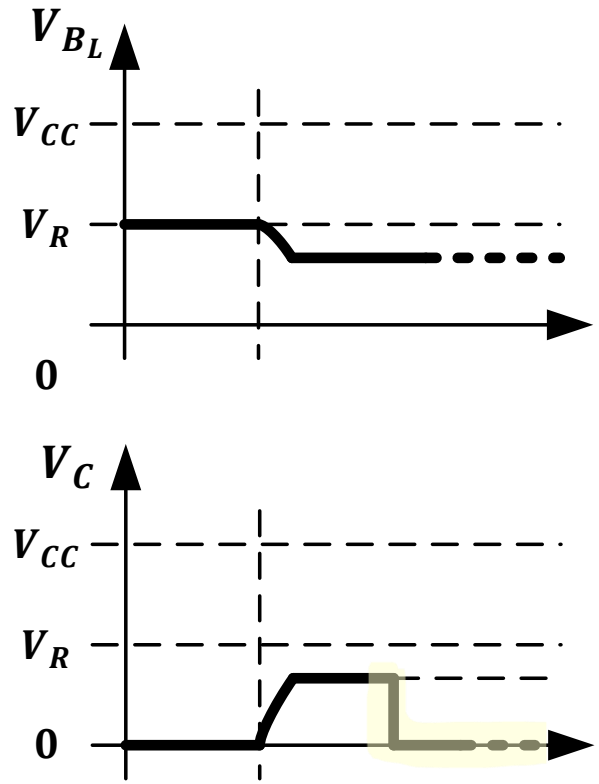


    - Read destroys data!
    - Need of re-write!

Memory bit = 1 ($V_C = V_{CC}$)



Memory bit = 0 ($V_C = 0\ V$)

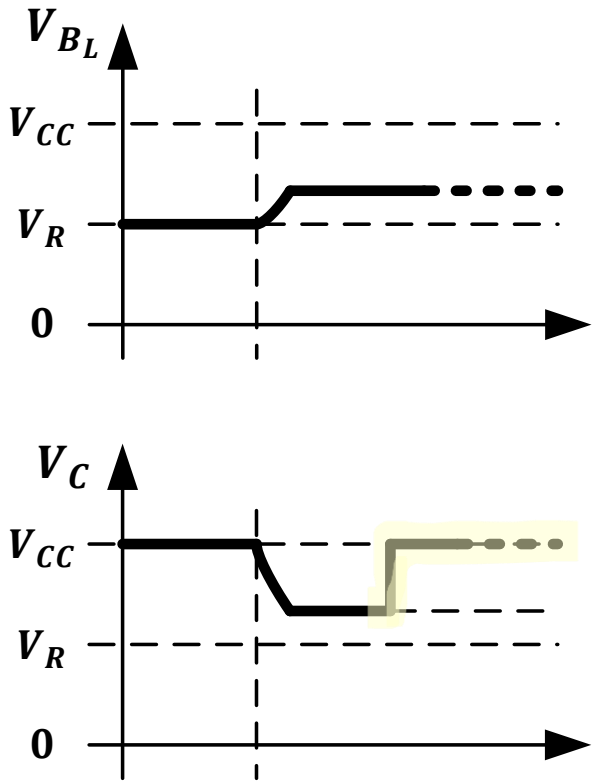


DRAM – Memory cell

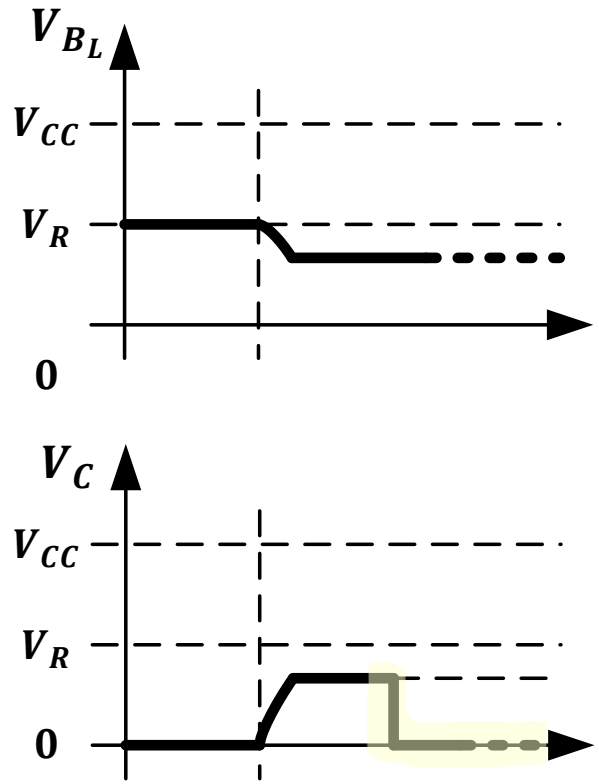
- Working principle
  - **Read**
    - How to re-write?
      - If $V_{B_L} > V_R \rightarrow 1$
      - If $V_{B_L} < V_R \rightarrow 0$

If voltage is higher means I had a logic 1.

Memory bit = 1 ($V_C = V_{CC}$)



Memory bit = 0 ($V_C = 0\ V$)

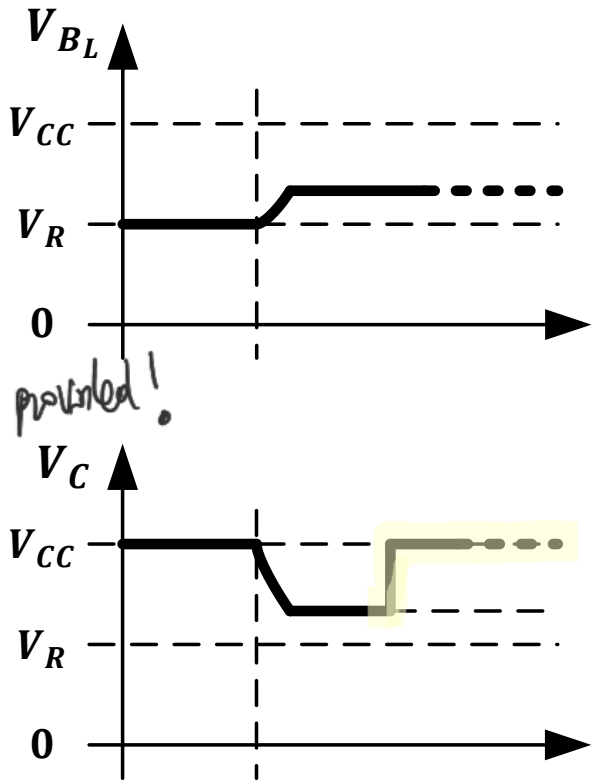


DRAM – Memory cell

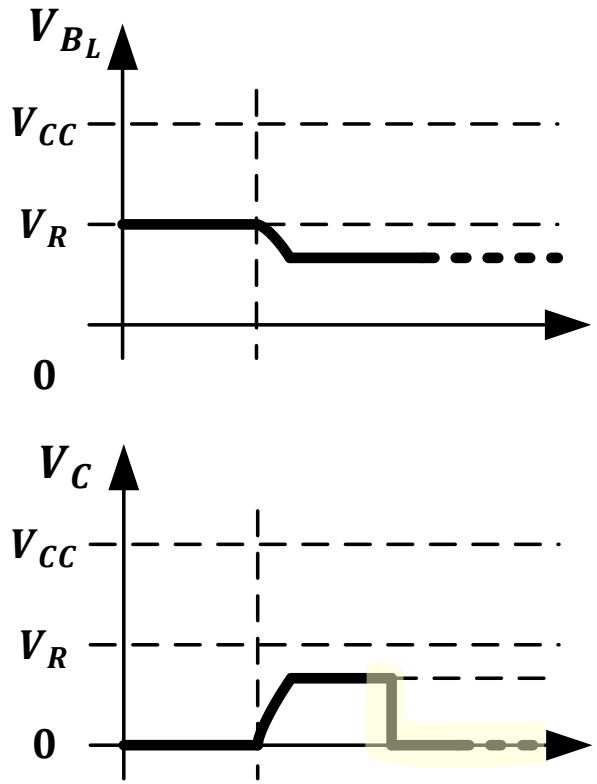
- Working principle
  - **Read**
    - How to re-write?
      - If $V_{B_L} > V_R \rightarrow 1$
      - If $V_{B_L} < V_R \rightarrow 0$
    - $|V_{B_L} - V_R|$
      - Very small value
      - Must be **amplified**!

Amplify the difference provided!

Memory bit = 1 ($V_C = V_{CC}$)



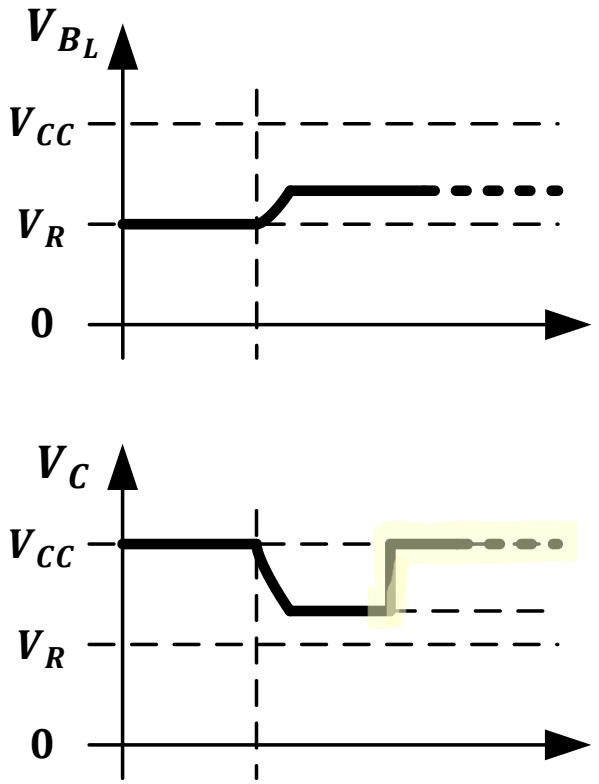
Memory bit = 0 ($V_C = 0\ V$)



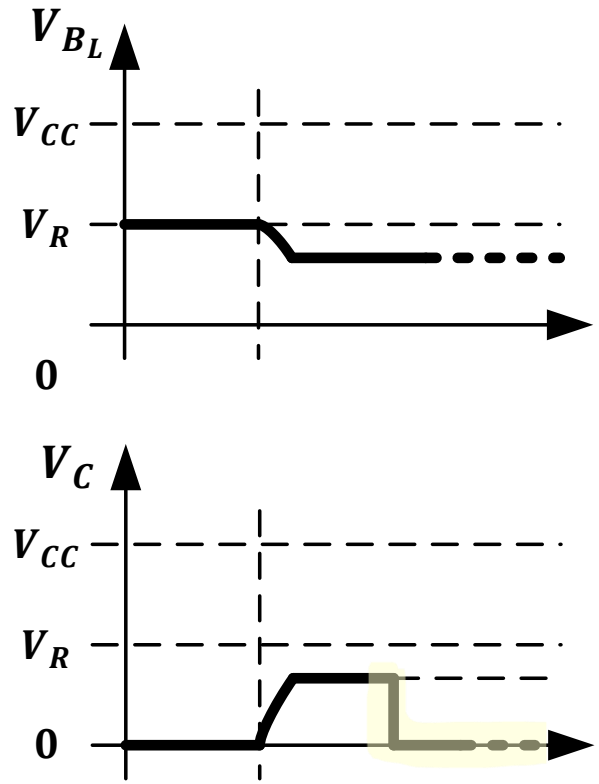
DRAM – Memory cell

- Working principle
  - **Read**
    - How to re-write?
      - If $V_{B_L} > V_R \rightarrow 1$
      - If $V_{B_L} < V_R \rightarrow 0$
    - $|V_{B_L} - V_R|$
      - Very small value
      - Must be amplified!
    - Sense Amplifier (**SA**)

Memory bit = 1 ($V_C = V_{CC}$)

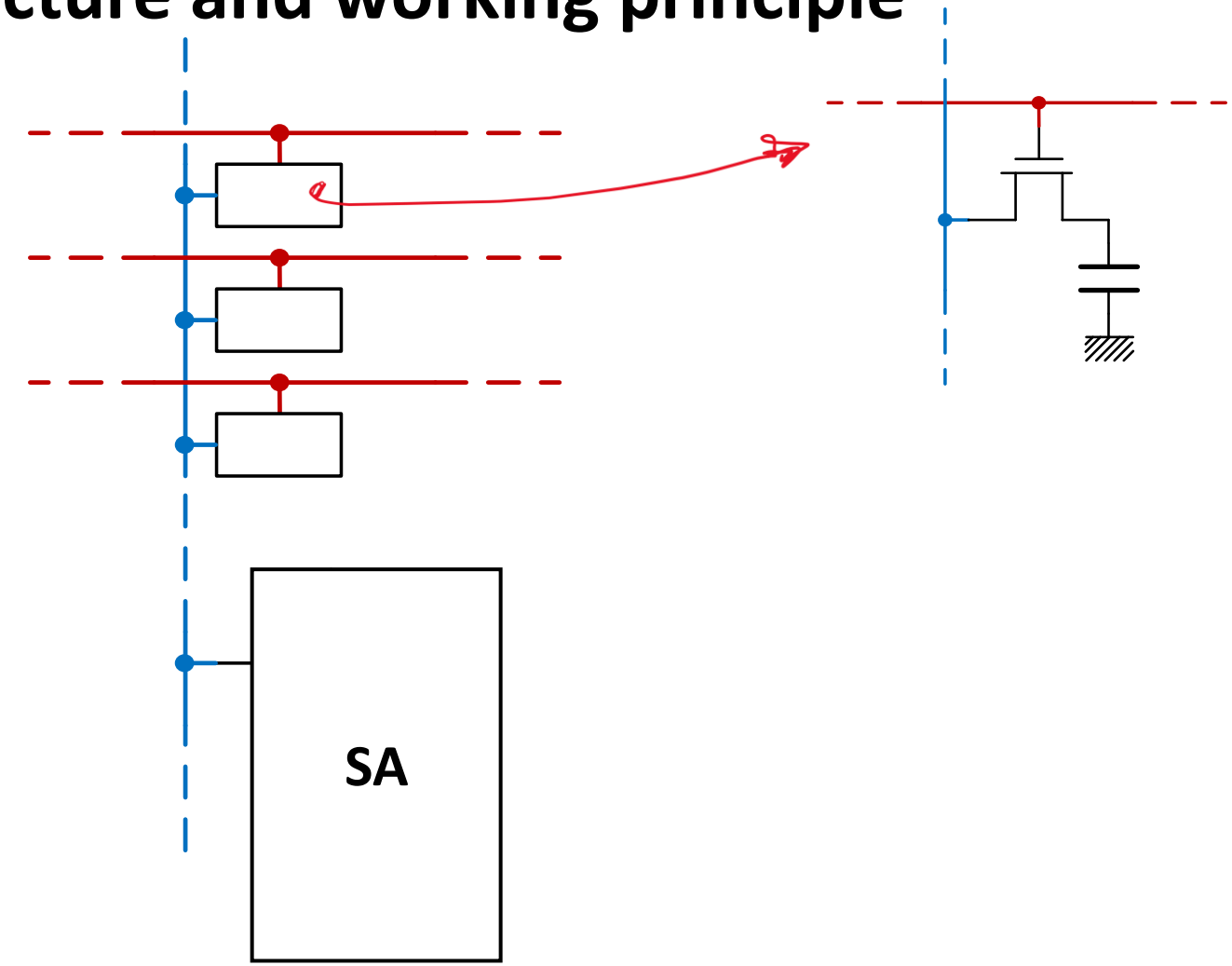


Memory bit = 0 ($V_C = 0\ V$)



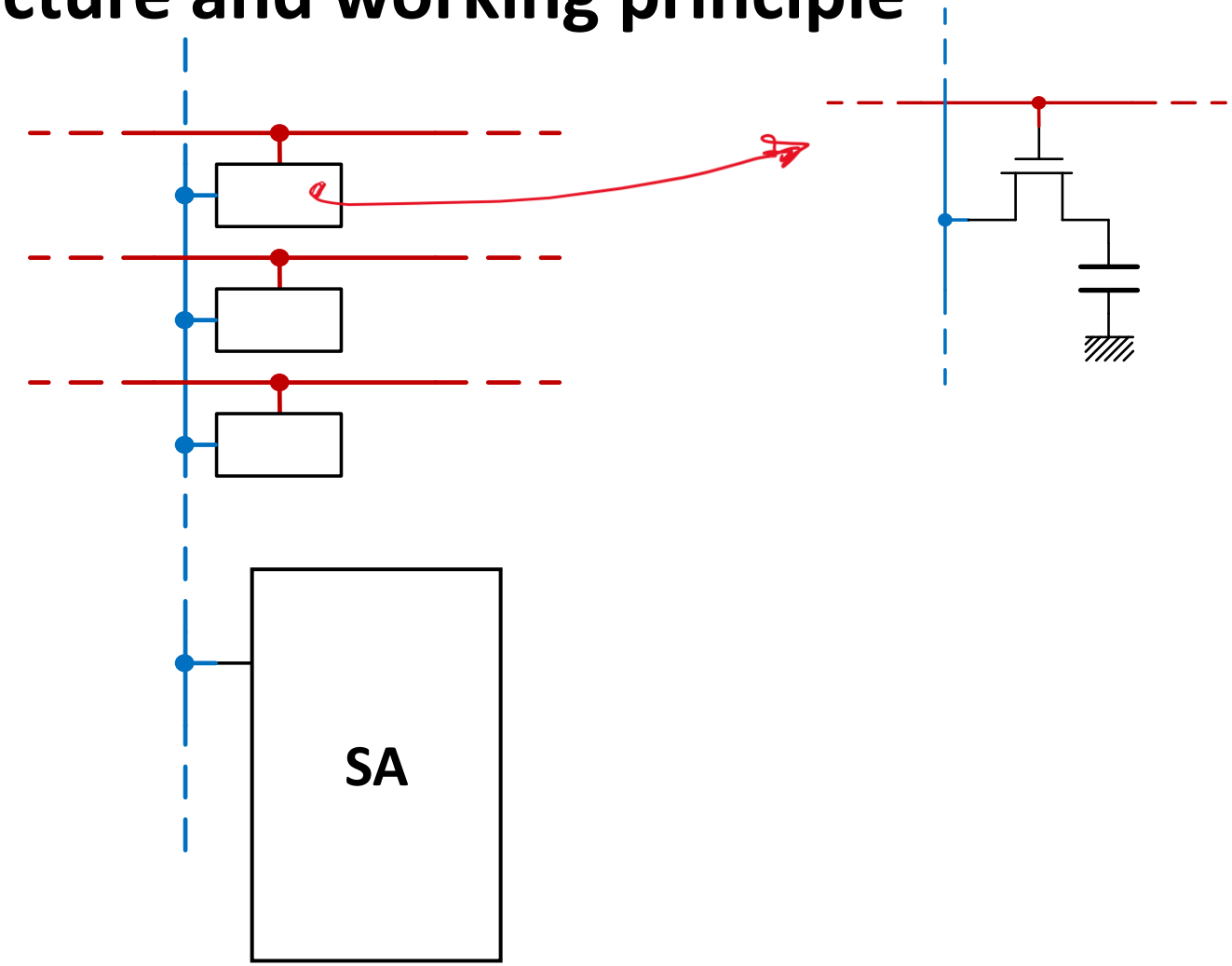
DRAM – Architecture and working principle

- Draft
  - Sense Amplifier (SA)



DRAM – Architecture and working principle

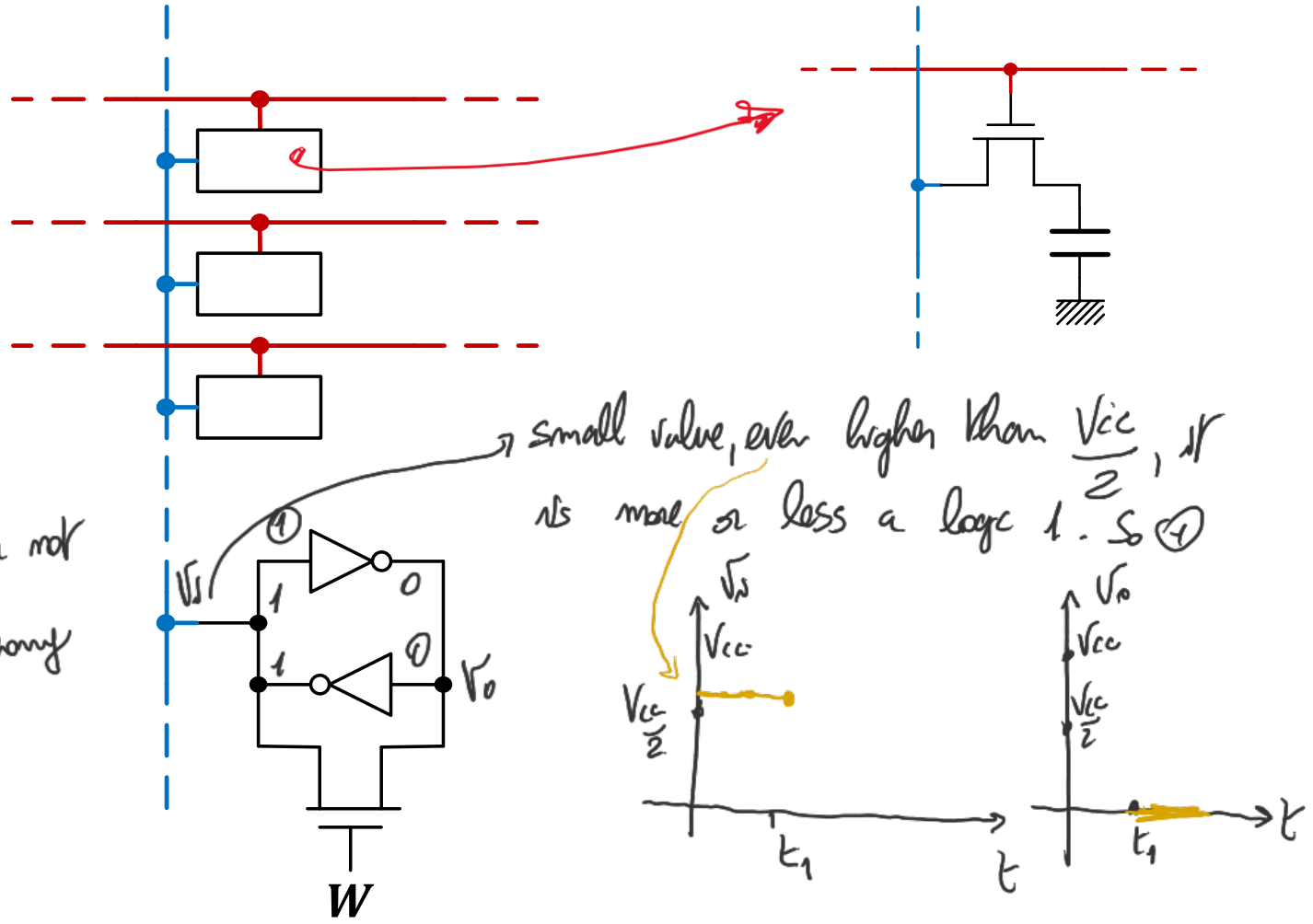
- Draft
  - Sense Amplifier (SA)
    - How?
    - **Bistable** !
      - We have already seen in SRAM



DRAM – Architecture and working principle

- Draft
  - Sense Amplifier (SA)
    - How?
    - **Bistable** !
      - We have already seen in SRAM

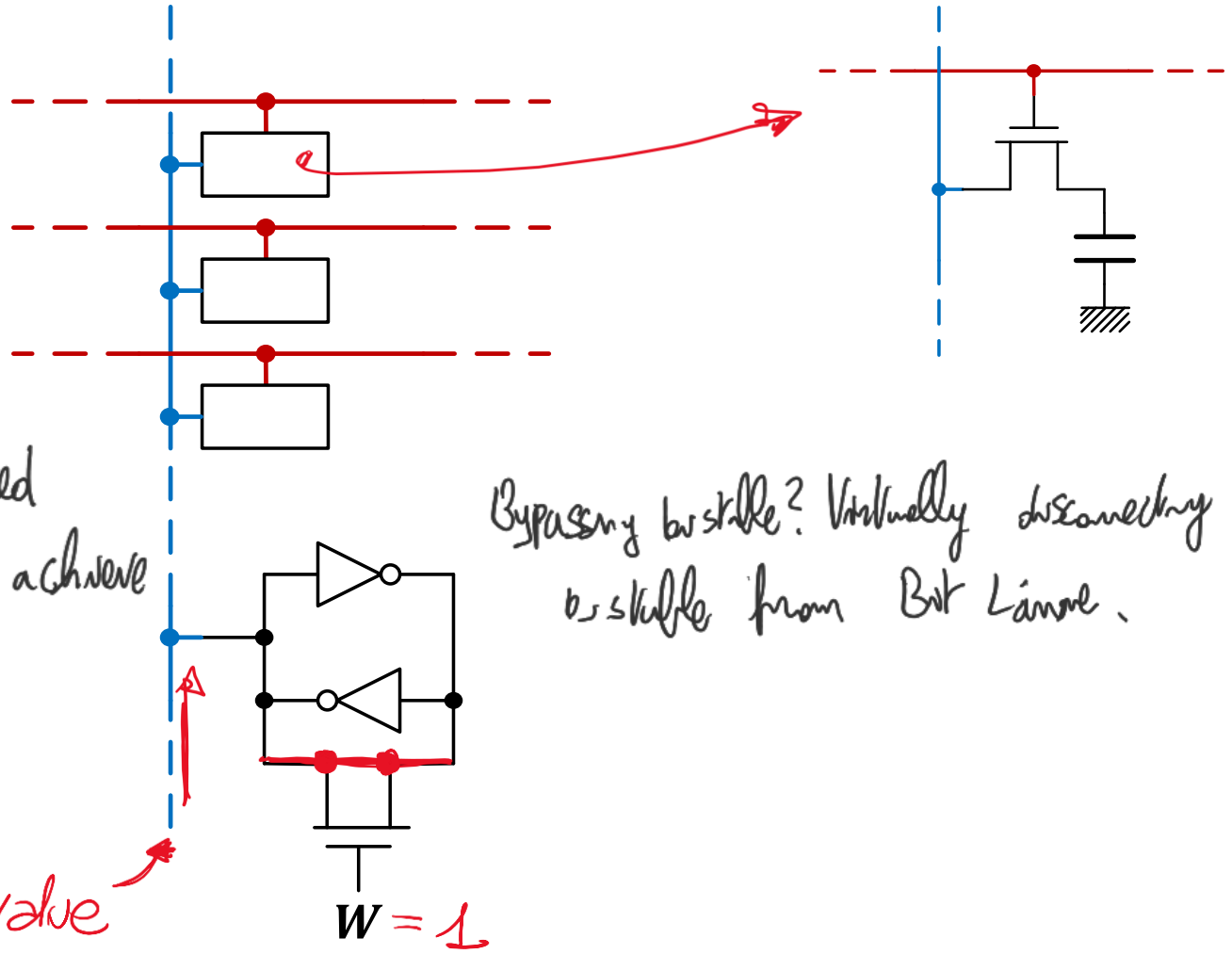
You will have a strong logic 0 even with a not strong 1. A strong logic 0 regenerates a strong logic 1.



DRAM – Architecture and working principle

- Draft
  - Sense Amplifier (SA)
    - **Write** ($W = 1$)
      - Bistable is bypassed (OFF)
      - Write value on Bit Line

When we want to write, we need to force disconnection to achieve writing.

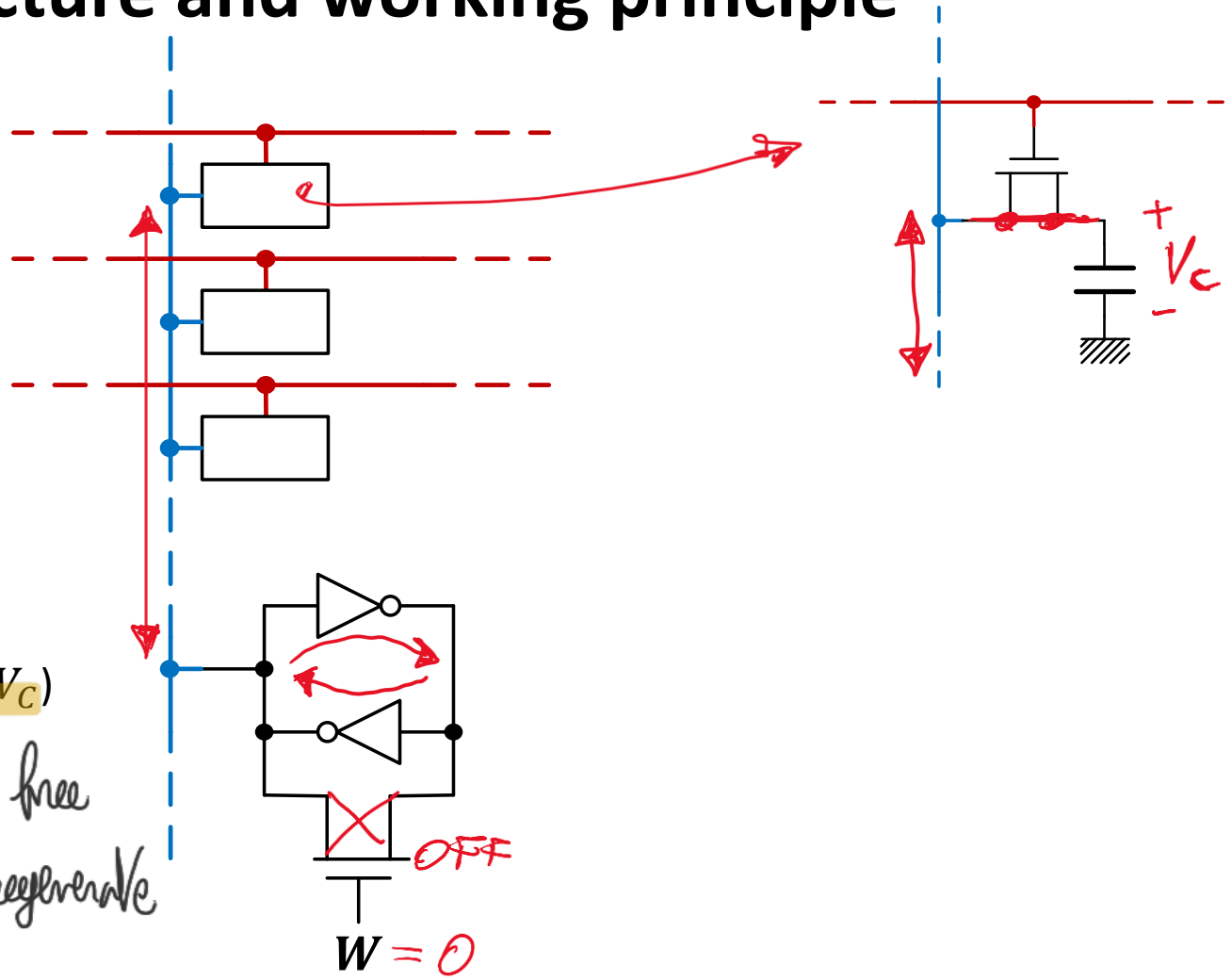


Bypassing bistable? Virtually disconnecting bistable from Bit Line.

DRAM – Architecture and working principle

- Draft
  - Sense Amplifier (SA)
    - Write ($W = 1$)
      - Bistable is bypassed (OFF)
      - Write value on Bit Line
    - **Read** ($W = 0$)
      - Bistable is ON
      - Bistable regenerates memory bit (V_C)

↳ here the bistable is free to evolve and regenerate



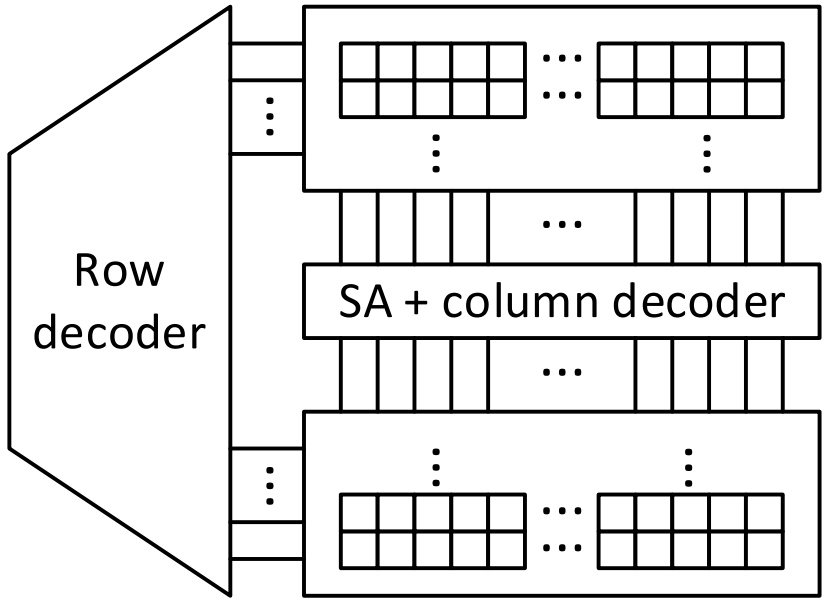
DRAM – Architecture and working principle

- Draft
  - Bit Lines are "big" capacitors (C_{B_L})
    - SAs placed in the middle (of each BL) for balancing capacitive load (C_{B_L})
  - In addition
    - SA block merged with the column-decoder for resource saving and architecture simplification

→ To have capacitive seen by the input of the SA be the same in both directions.

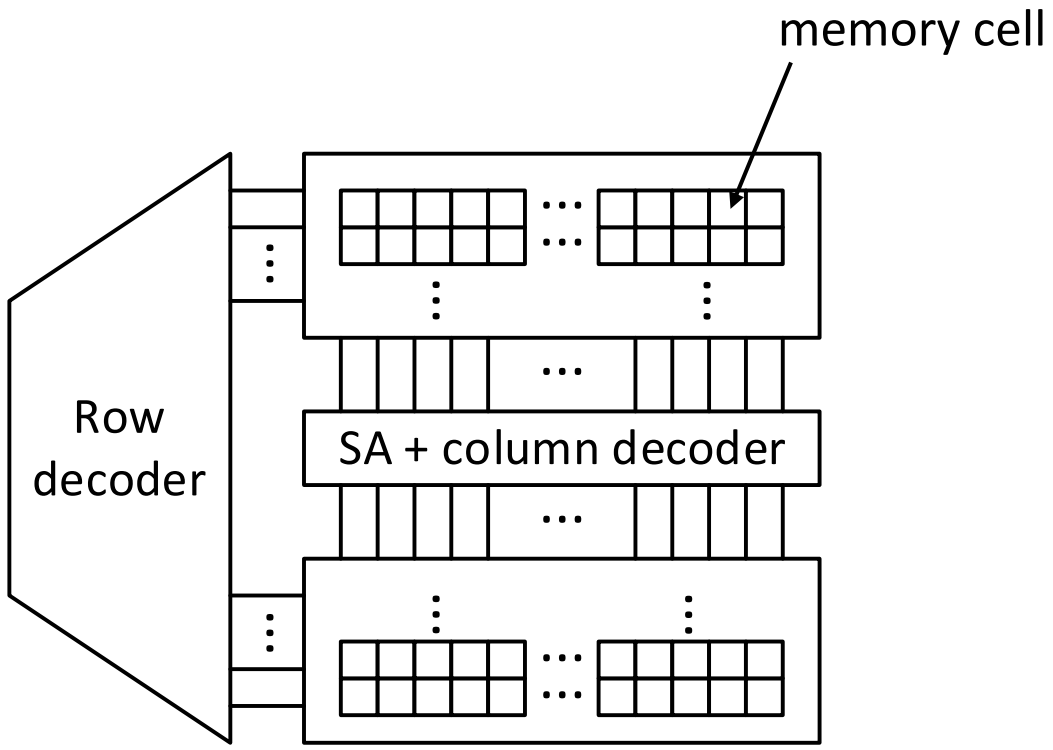
DRAM – Architecture and working principle

- Draft



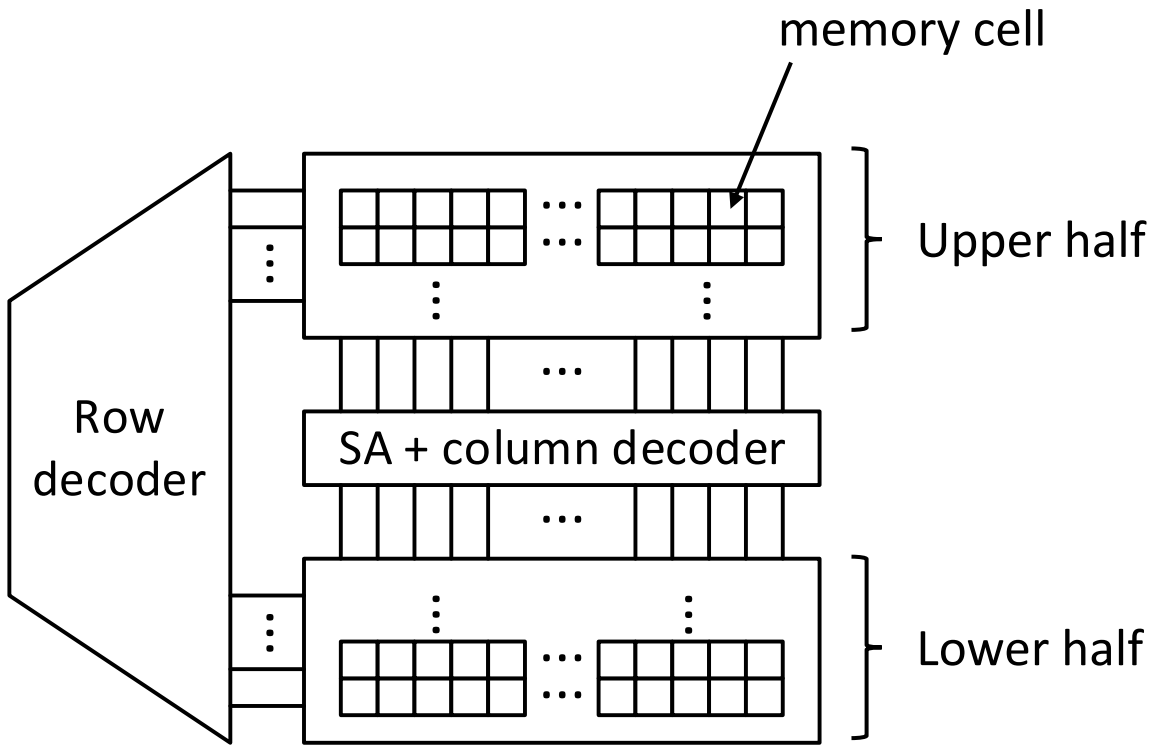
DRAM – Architecture and working principle

- Draft



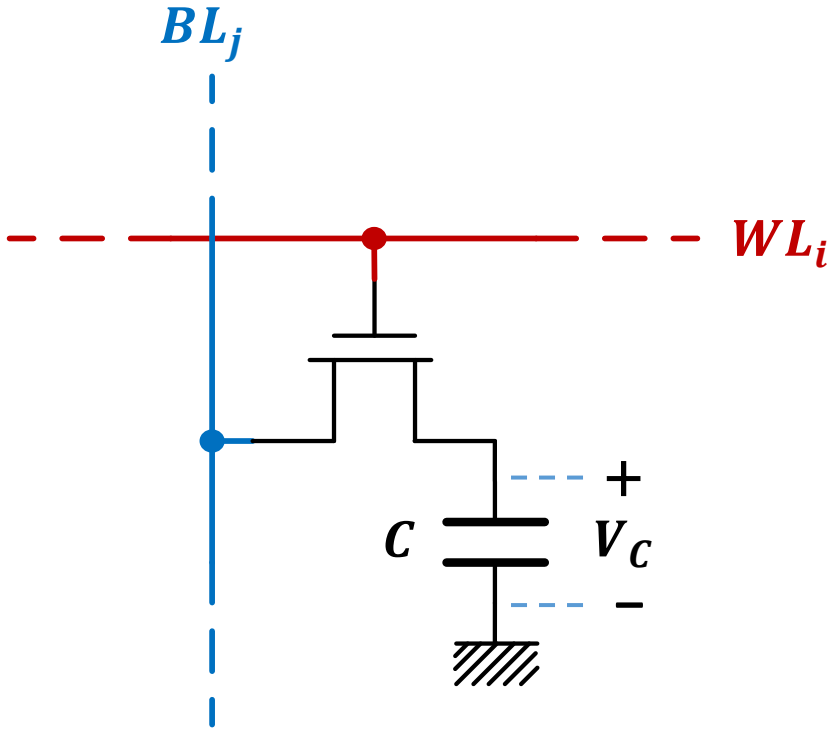
DRAM – Architecture and working principle

- Draft



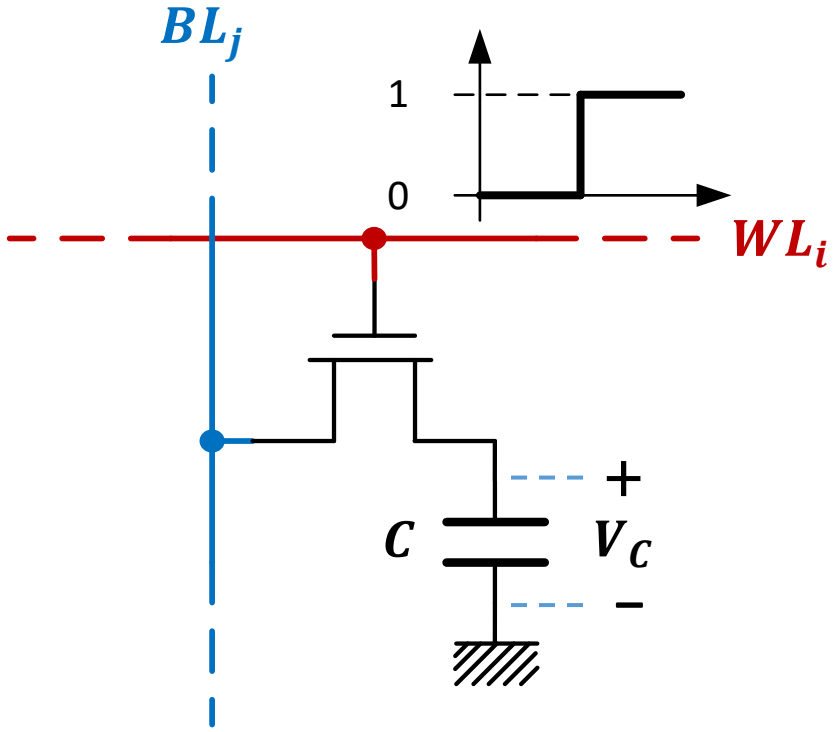
DRAM – Architecture and working principle

- Draft
  - Other problem(s)
    - When reading (or writing)
      - WL_i moves from 0 to 1



DRAM – Architecture and working principle

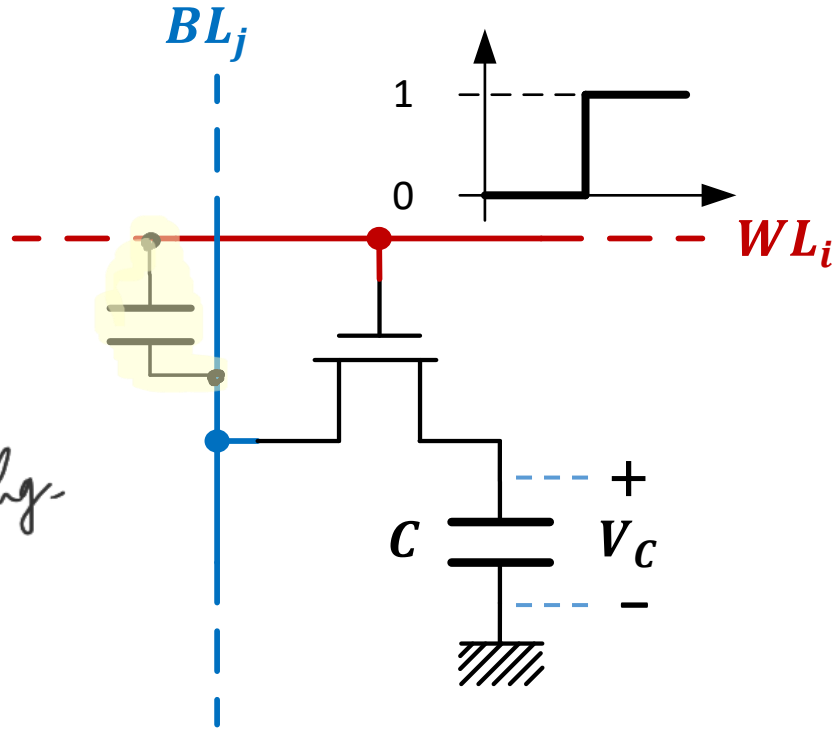
- Draft
  - Other problem(s)
    - When reading (or writing)
      - WL_i moves from 0 to 1



DRAM – Architecture and working principle

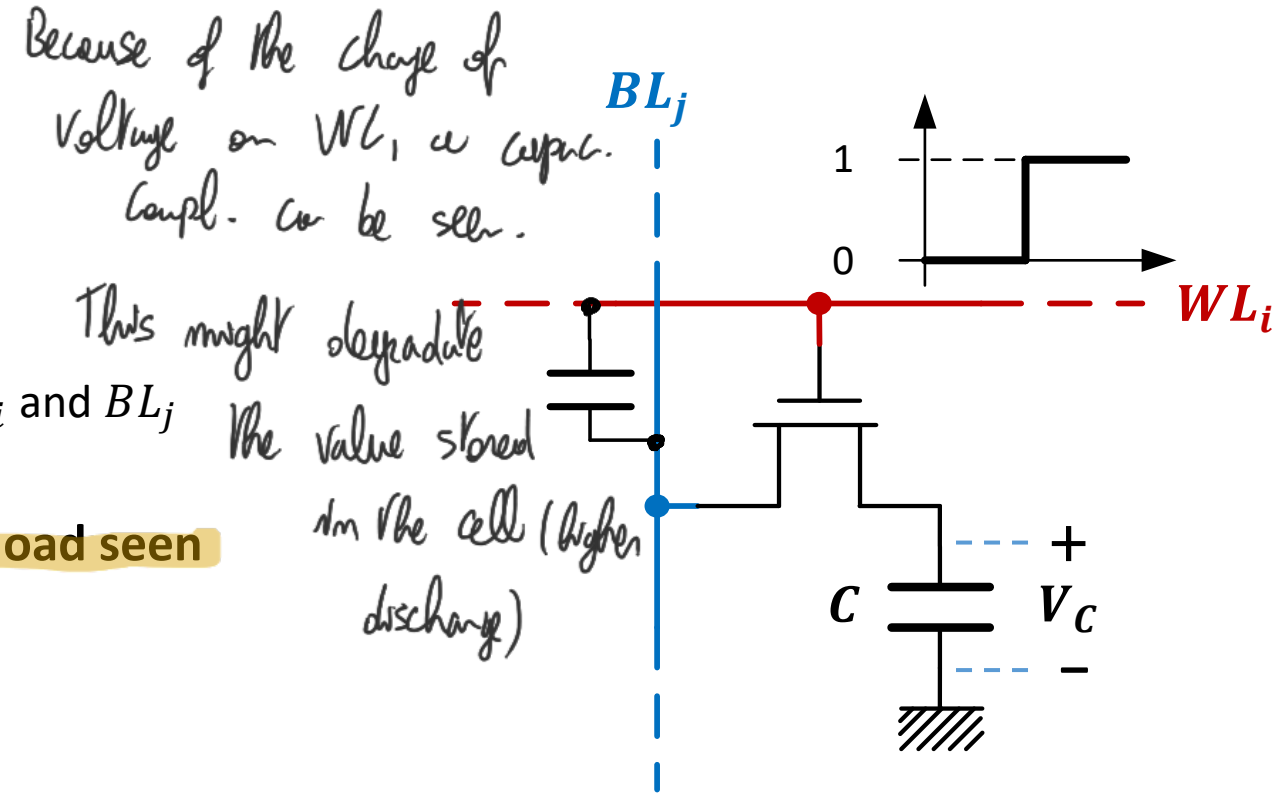
- Draft
  - Other problem(s)
    - When reading (or writing)
      - WL_i moves from 0 to 1
      - Capacitive coupling between WL_i and BL_j

This behaves like capacitive coupling.



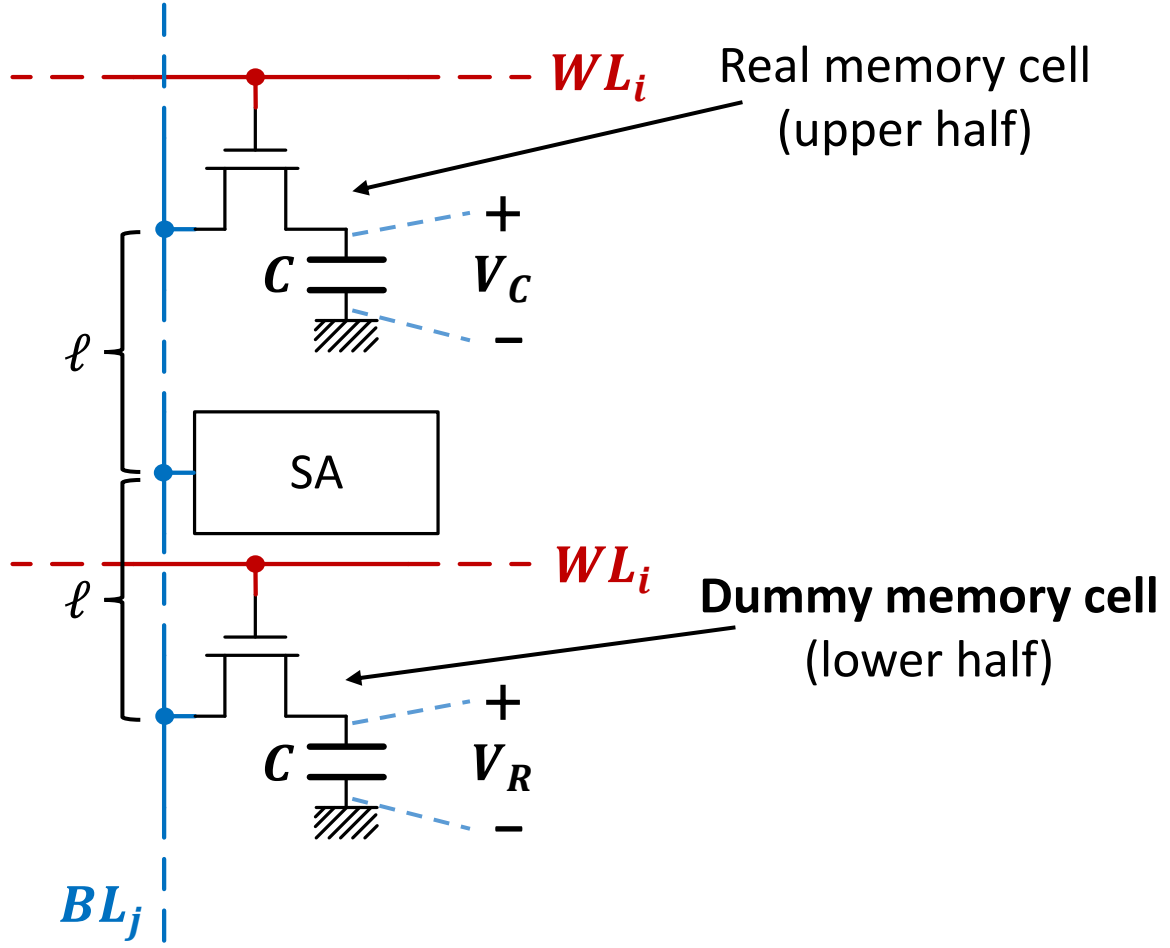
DRAM – Architecture and working principle

- Draft
  - Other problem(s)
    - When reading (or writing)
      - WL_i moves from 0 to 1
      - Capacitive coupling between WL_i and BL_j
    - **To counteract this, the capacitive load seen by the SA must be balanced !**



DRAM – Architecture and working principle

- Draft
  - Balancing capacitive load seen by SA
    - Same distance ℓ in opposite directions
    - Same WL_i
    - Dummy memory cell
      - Opposite half
      - Preloaded at V_R



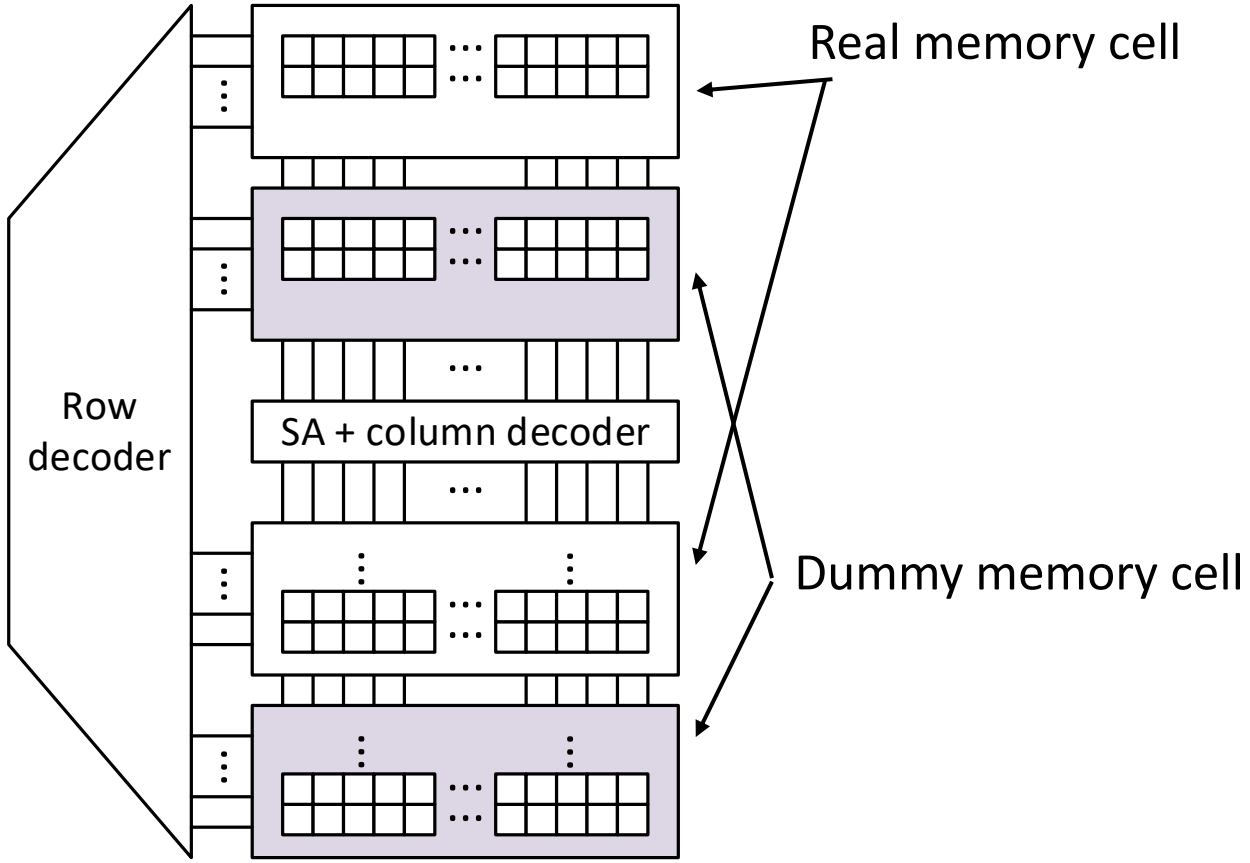
If you are doing nothing, capacity load seen by the SA up and down is the same. Now assume that you connect one memory cell, this means that capacity load on the top is more than the one we see down.

When you do any kind of operation (activating the transistor of the cell) this is a connection.

To balance this unbalance we need a dummy cell. It is connected to the SAME WORDLINE

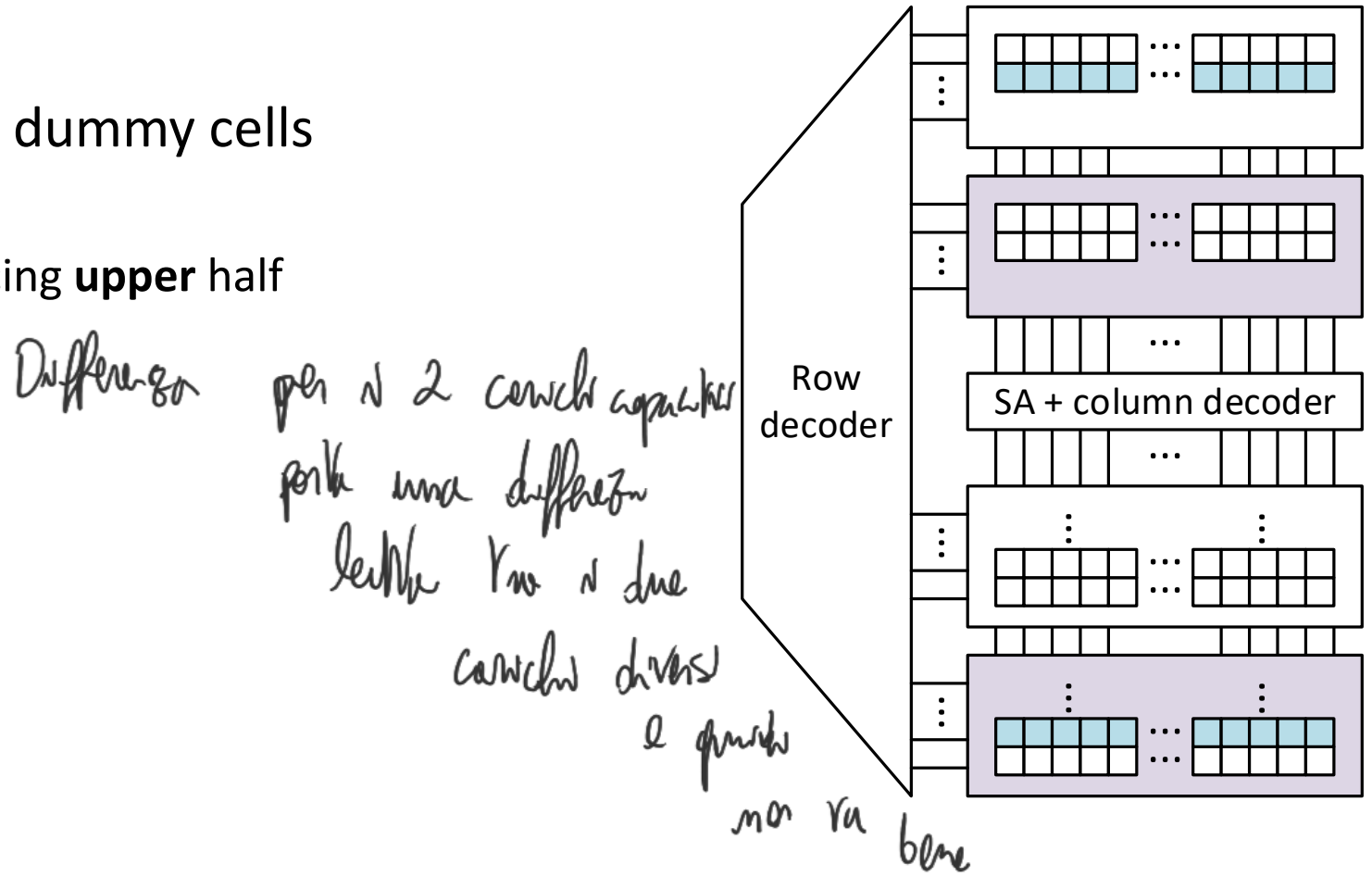
DRAM – Architecture and working principle

- Draft
  - Including dummy cells



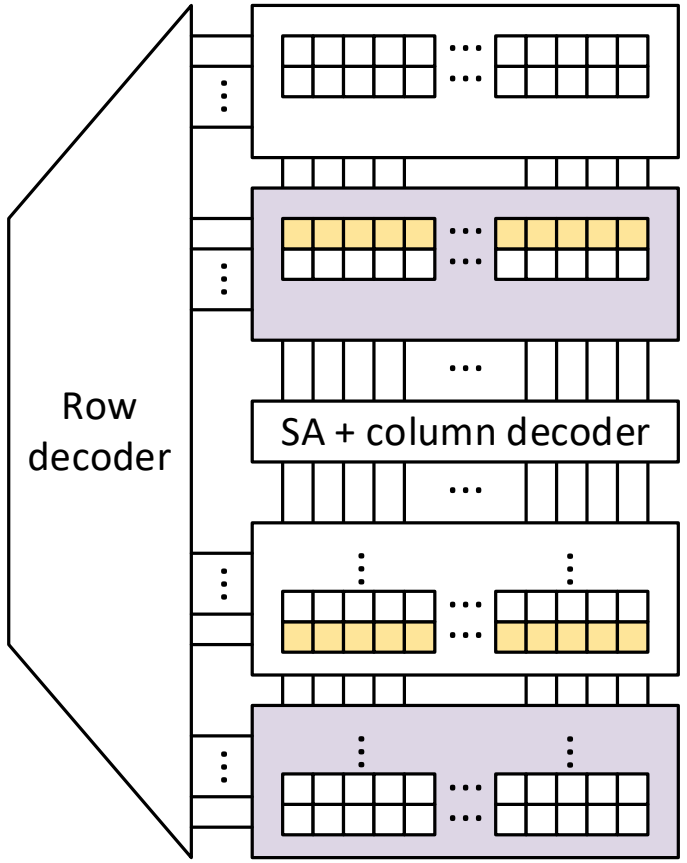
DRAM – Architecture and working principle

- Draft
  - Including dummy cells
  - Example
    - Balancing **upper** half



DRAM – Architecture and working principle

- Draft
  - Including dummy cells
  - Example
    - Balancing **lower** half



DRAM – Architecture and working principle

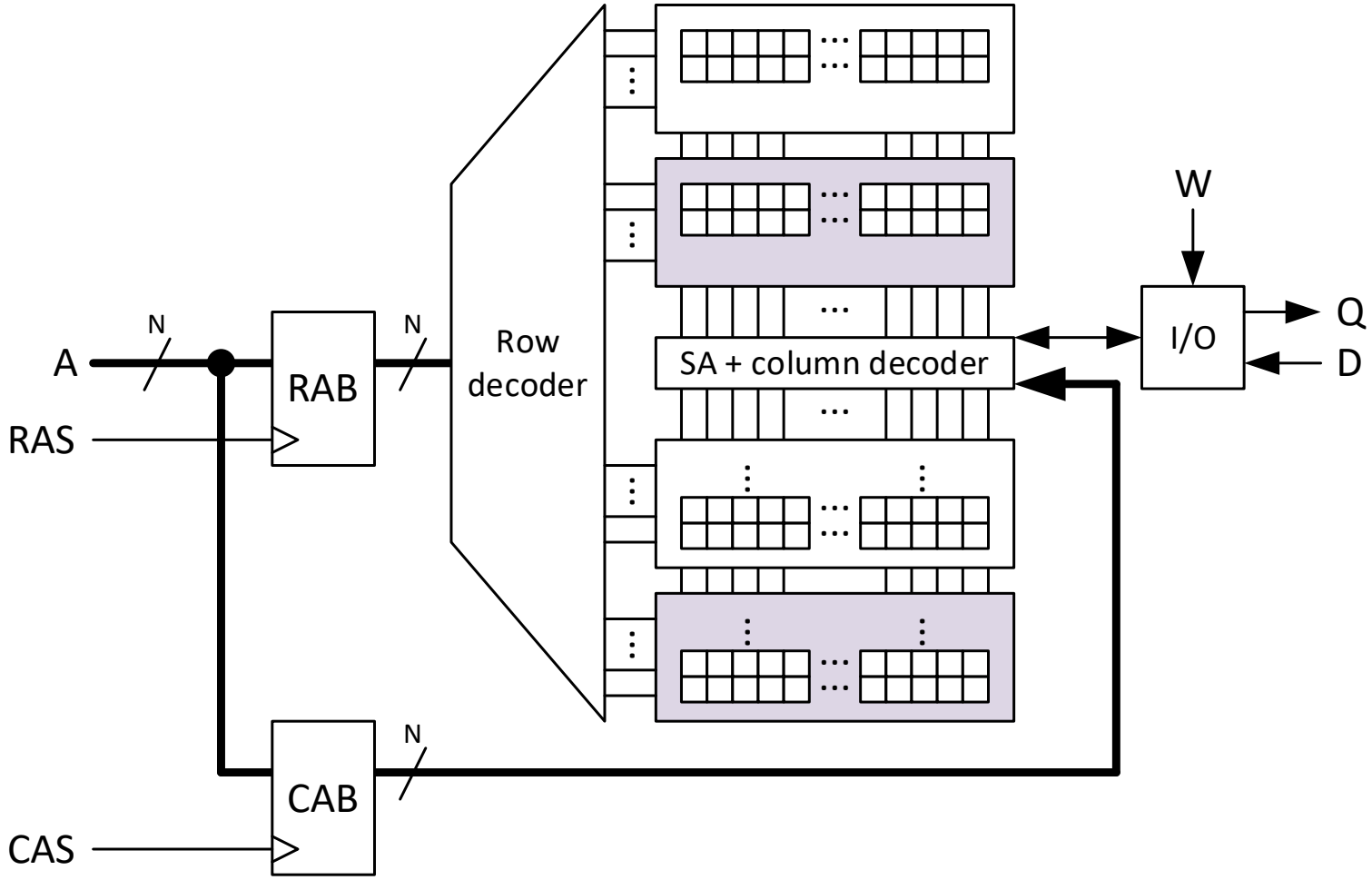
- Final
  - To complete the architecture, they should be integrated
    - Address
    - Write command/signal
    - Input data (D) and output data (Q) ports

DRAM – Architecture and working principle

- Final
  - To complete the architecture, they should be integrated
    - Address
    - Write command/signal
    - Input data (D) and output data (Q) ports
  - To reduce costs/save resources
    - Single I/O port
    - Single address port shared by row and column decoders
      - Buffer to store row/column address
        - **RAB** = **R**ow **A**ddress **B**uffer ← triggered by signal **RAS** = **R**ow **A**ddress **S**trobe
        - **CAB** = **C**olumn **A**ddress **B**uffer ← triggered by signal **CAS** = **C**olumn **A**ddress **S**trobe

DRAM – Architecture and working principle

- Final
  - A = address port
  - Q = output (data) port
  - D = input (data) port
  - W = write command



DRAM – Architecture and working principle

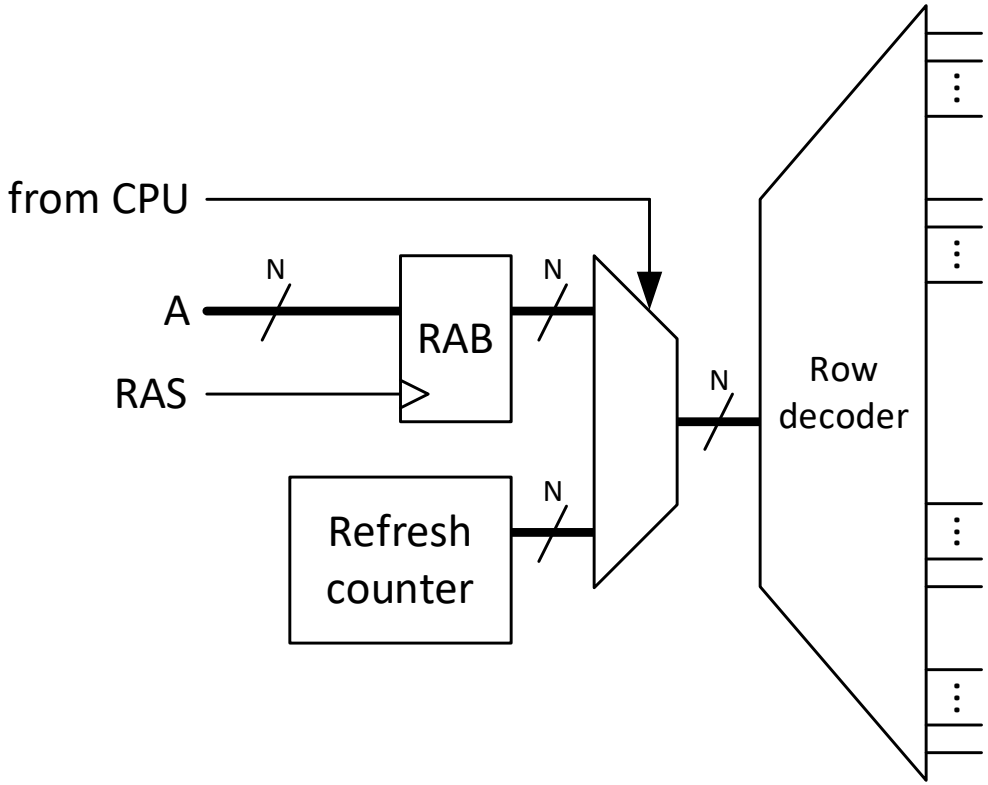
- Memory bit stored inside the cells lasts at most $10 - 100\ ms$
  - **REFRESH** operation is required

We need to precharge BL lines to $\frac{V_{CC}}{2}$

    - Only row must be addressed to refresh the memory
      - All cells in the same row (Word Line) can be refreshed at the same time
    - T_{REF} = maximum time before refresh
    - Different approaches
      - Wait for T_{REF} and then refresh all memory rows in sequence
      - Wait for $\frac{T_{REF}}{row\ number}$ and refresh one row at a time

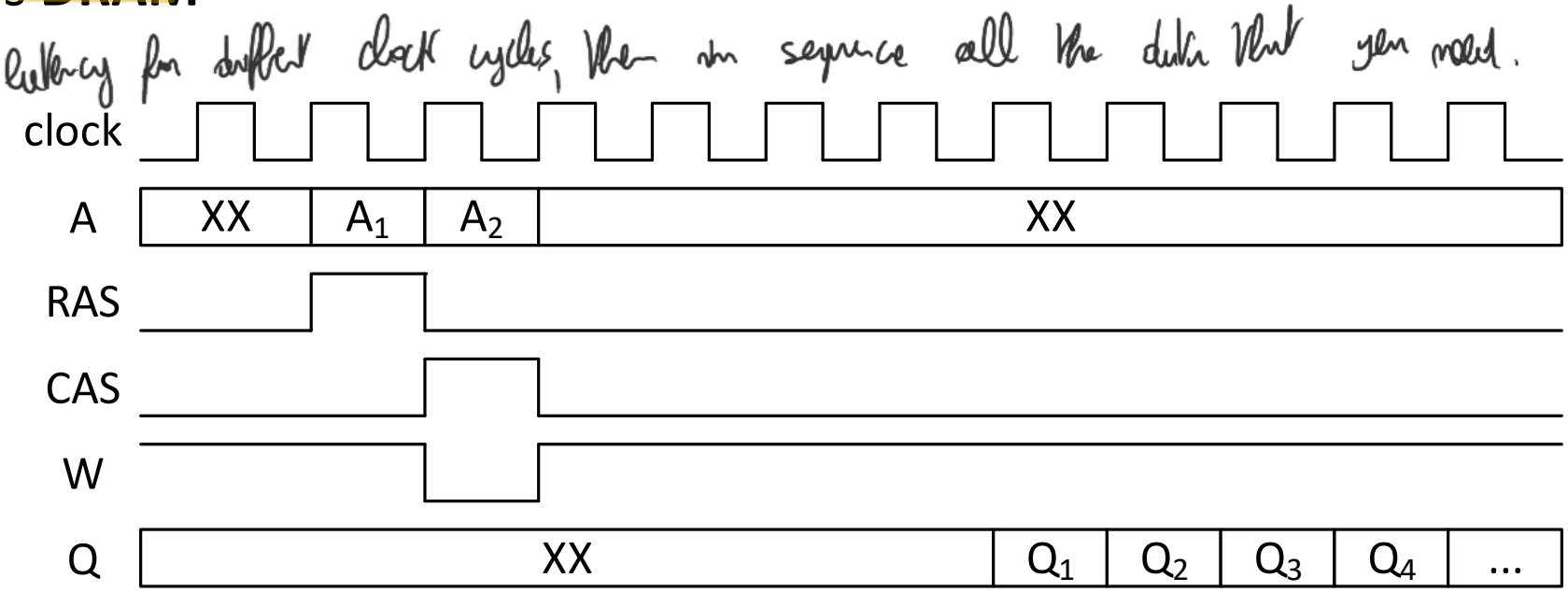
DRAM – Architecture and working principle

- Memory bit stored inside the cells lasts at most $10 - 100\ ms$
  - **REFRESH** operation is required
    - DRAM "knows" **where** (which row) to perform the refresh
      - Counter
    - Processor/CPU "knows" **when** to perform the refresh
      - Timer



SDRAM

- Modern DRAMs are synchronous
  - Control and data operations referenced to a common clock signal
  - **SDRAM** = **S**ynchronous **DRAM**
  - **Burst mode**
    - Initial latency followed by a flood of data



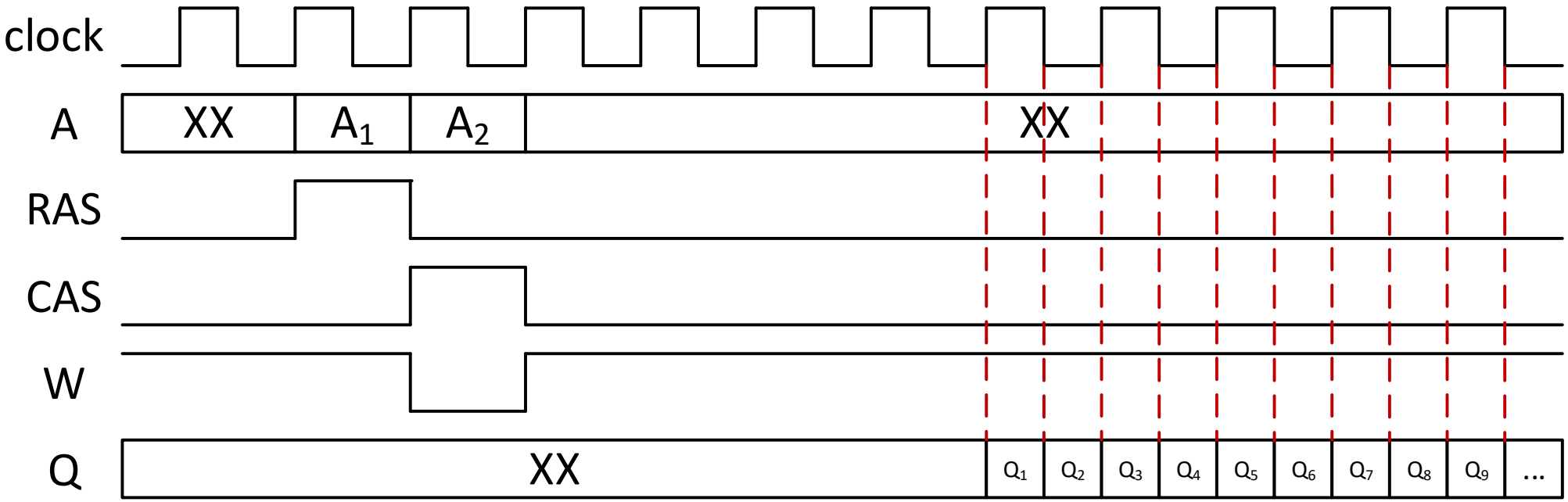
DDR

- **DDR** = **D**ouble **D**ata **R**ate
  - SDRAM in which control and data operations are performed on both edges of clock signal
    - Rising edge
    - Falling edge

↳ Both edges used to perform operations.

  - Data transfer rate is doubled!
    - From here the name
  - Standardized throughout the years

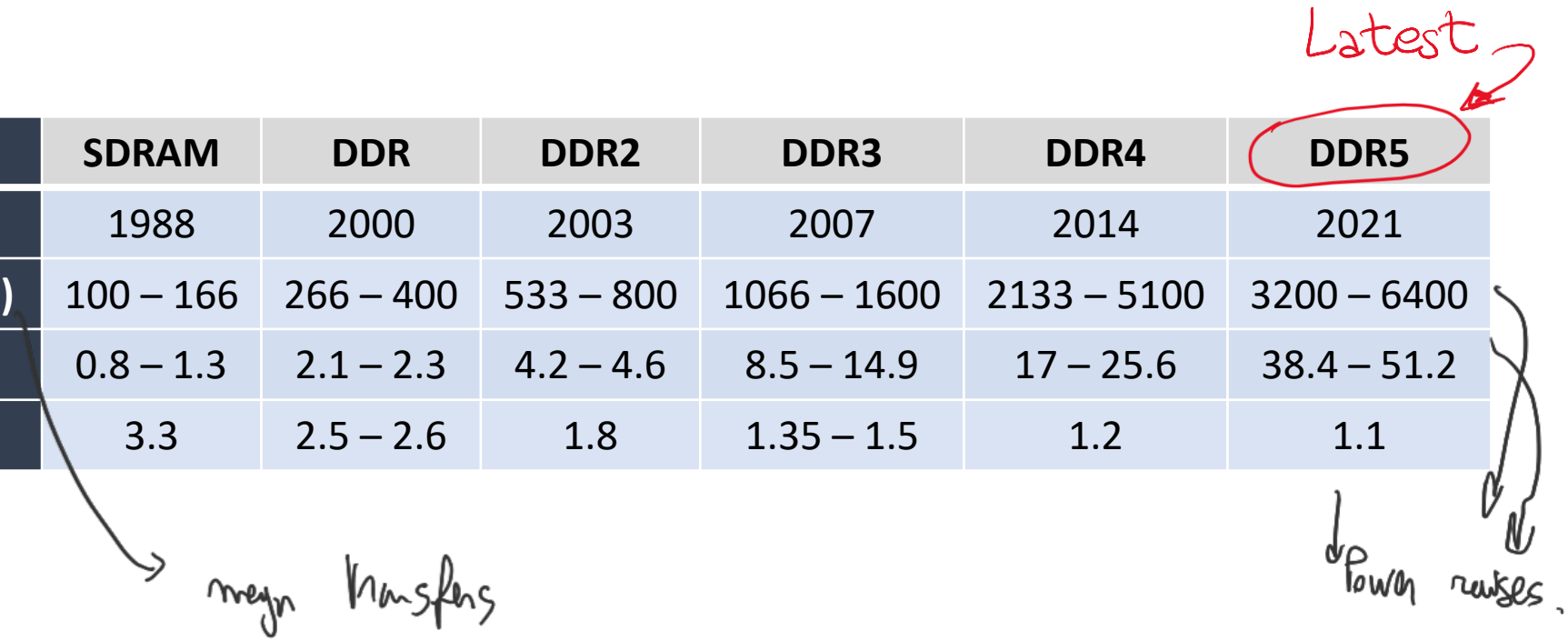
DDR



DDR

- Standards/generations

Generation	SDRAM	DDR	DDR2	DDR3	DDR4	DDR5
Year	1988	2000	2003	2007	2014	2021
Transfer rate (MT/s)	100 – 166	266 – 400	533 – 800	1066 – 1600	2133 – 5100	3200 – 6400
Data rate (GB/s)	0.8 – 1.3	2.1 – 2.3	4.2 – 4.6	8.5 – 14.9	17 – 25.6	38.4 – 51.2
Voltage (V_{CC})	3.3	2.5 – 2.6	1.8	1.35 – 1.5	1.2	1.1



Thank you for your attention

Luca Crocetti
(luca.crocetti@unipi.it)