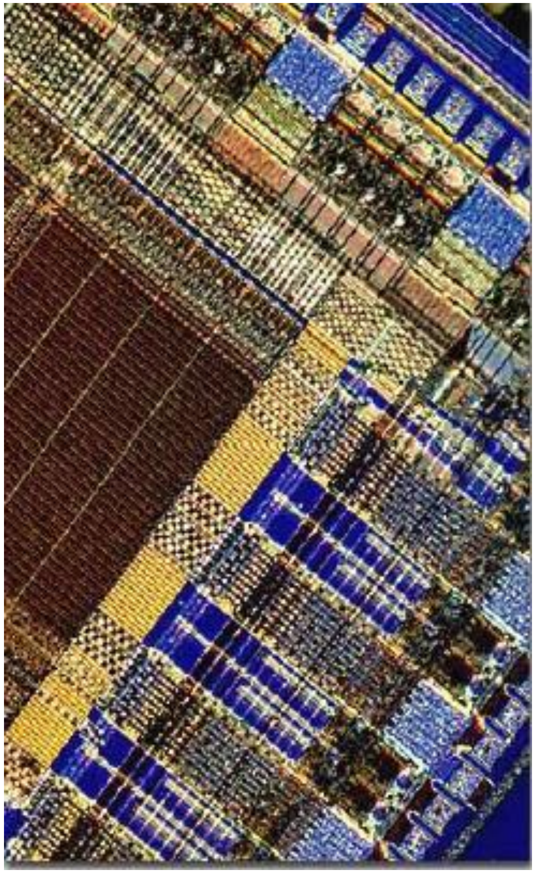
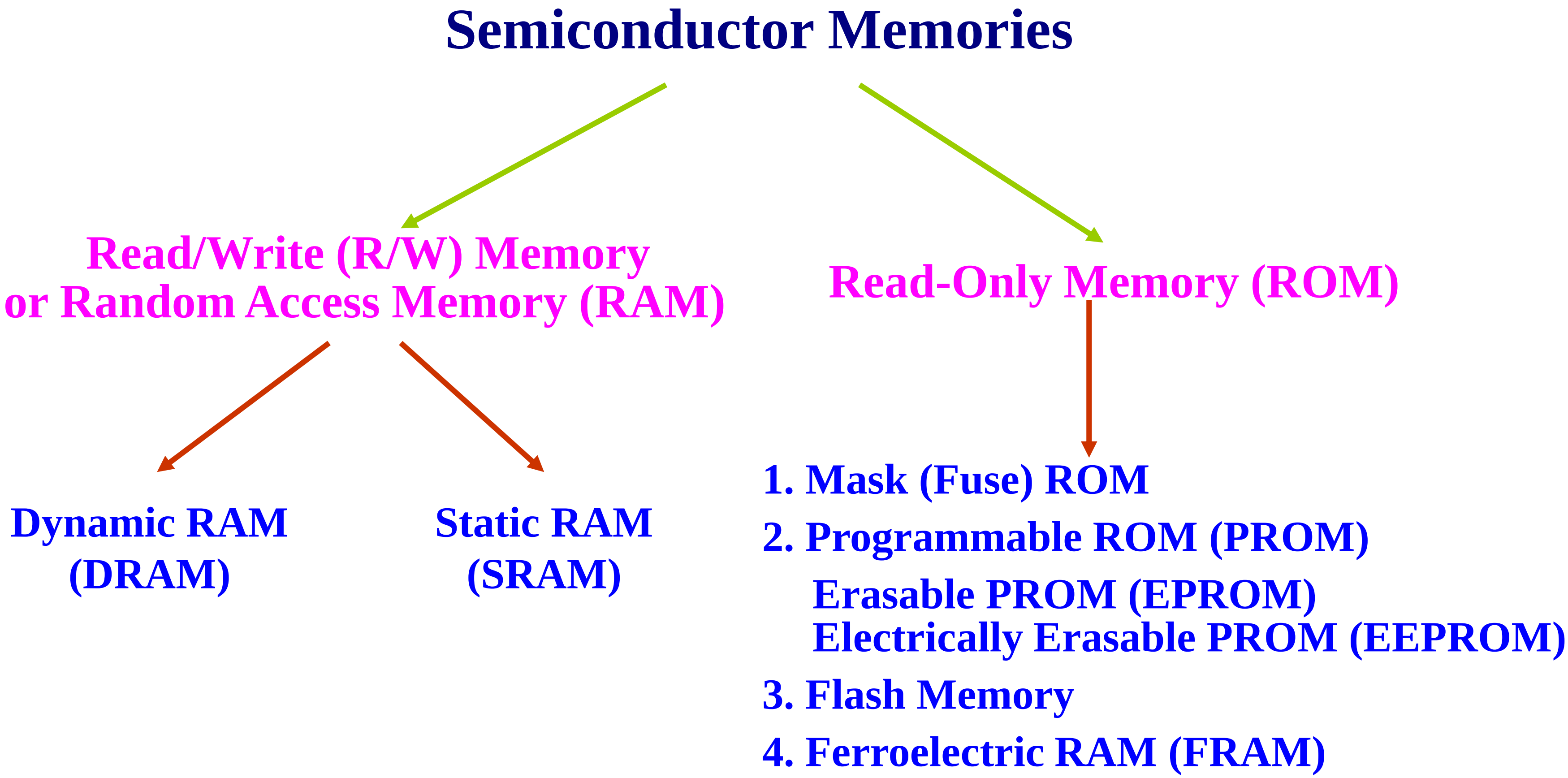
CMOS Digital Integrated Circuits



Lec 13

Semiconductor Memories

Semiconductor Memory Types



Semiconductor Memories

Read/Write (R/W) Memory or Random Access Memory (RAM)

- **Dynamic RAM (DRAM)**
- **Static RAM (SRAM)**

Read-Only Memory (ROM)

1. **Mask (Fuse) ROM**
2. **Programmable ROM (PROM)**

   Erasable PROM (EPROM)
   Electrically Erasable PROM (EEPROM)
3. **Flash Memory**
4. **Ferroelectric RAM (FRAM)**

Semiconductor Memory Types (Cont.)

- **Design Issues**
  - **Area Efficiency of Memory Array: #** of stored data bits per unit area
  - **Memory Access Time:** the time required to store and/or retrieve a particular data bit.
  - **Static and Dynamic Power Consumption**
- **RAM: the stored data is volatile**
  - ***DRAM***
    - » A capacitor to store data, and a transistor to access the capacitor
    - » **Need refresh operation**
    - » **Low cost**, and **high density** ⇒ it is used for main memory
  - ***SRAM***
    - » Consists of a latch
    - » **Don't need the refresh operation**
    - » **High speed** and **low power consumption** ⇒it is mainly used for cache memory and memory in hand-held devices

Semiconductor Memory Types (Cont.)

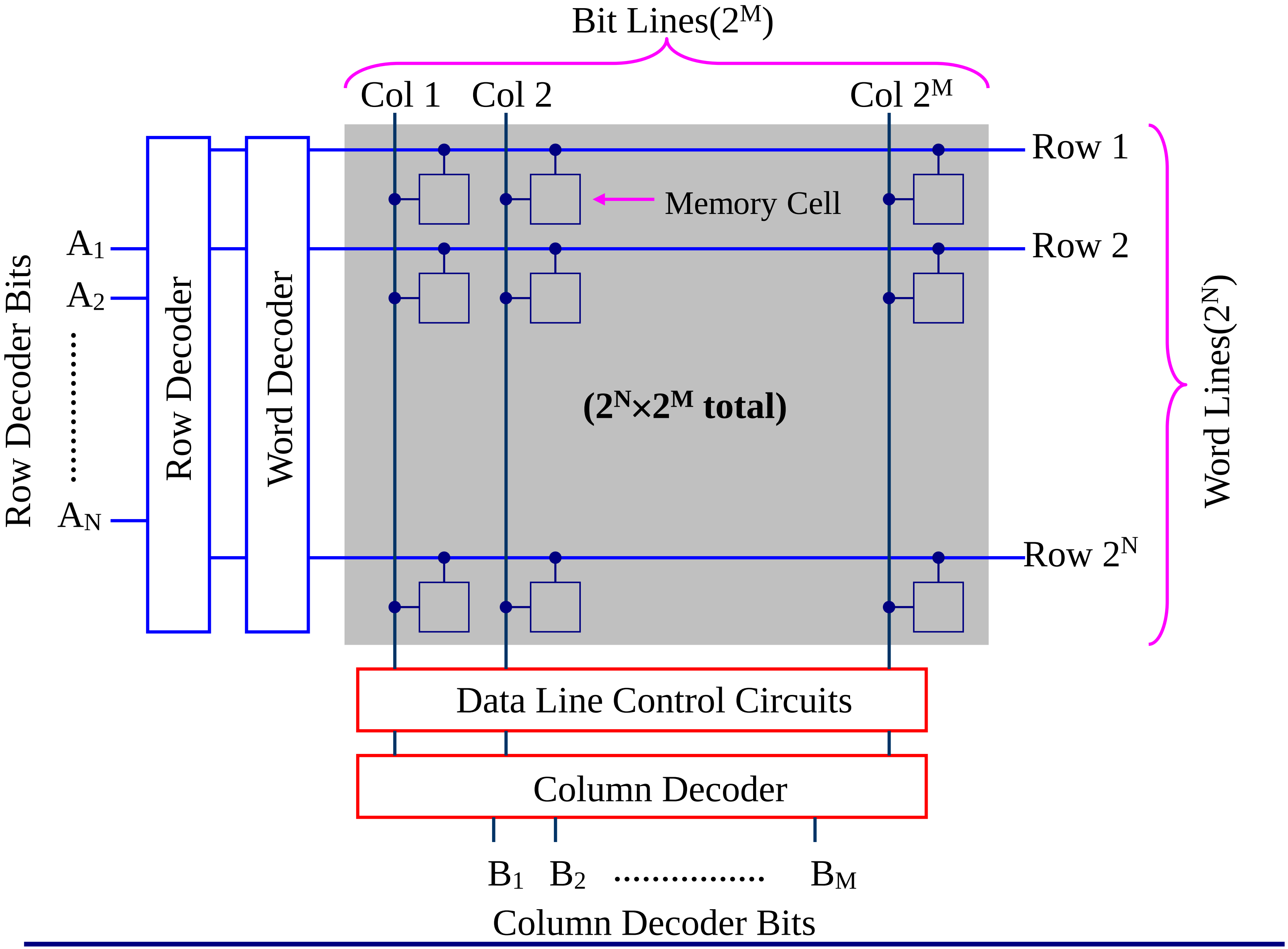
- **ROM: 1, nonvolatile memories**

  2, only can access data, cannot to modify data

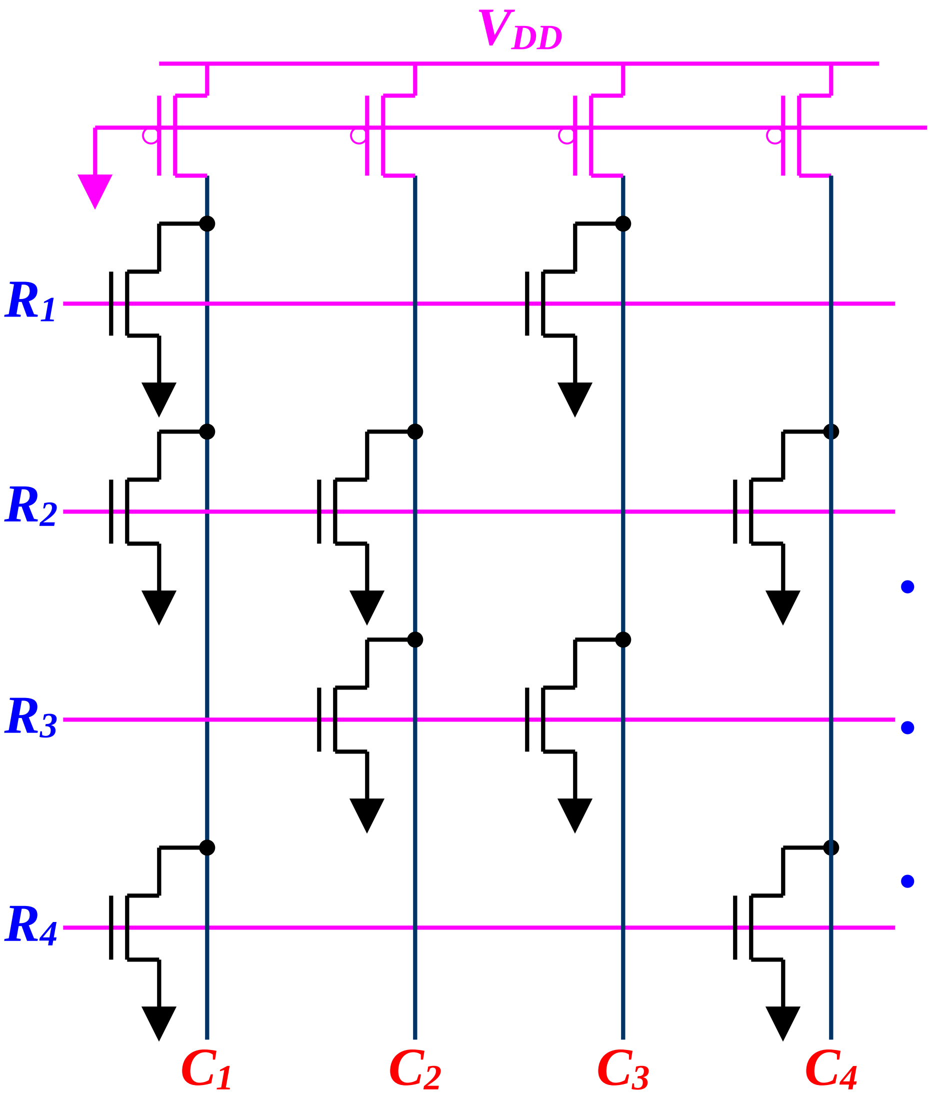
  3, lower cost: used for permanent memory in printers, fax, and game machines, and ID cards

  - ***Mask ROM*****:** data are written **during** chip fabrication by a **photo mask**
  - ***PROM*****:** data are written electrically **after** the chip is fabricated.
    - » ***Fuse ROM*****:** data **cannot** be erased and modified.
    - » ***EPROM and EEPROM*****:** data **can be rewritten**, but the number of subsequent re-write operations is limited to ***10^4-10^5***.
      - ***EPROM*** **uses ultraviolet rays** which can penetrate through the crystal glass on package to erase whole data simultaneously.
      - ***EEPROM*** **uses high electrical voltage** to erase data in 8 bit units.
  - ***Flash Memory*****:** similar to EEPROM
  - ***FRAM*****:** utilizes the **hysteresis** characteristics of a capacitor to overcome the slow written operation of EEPROMs.

Random-Access Memory Array Organization



Nonvolatile Memory
4Bit × 4Bit NOR-based ROM Array

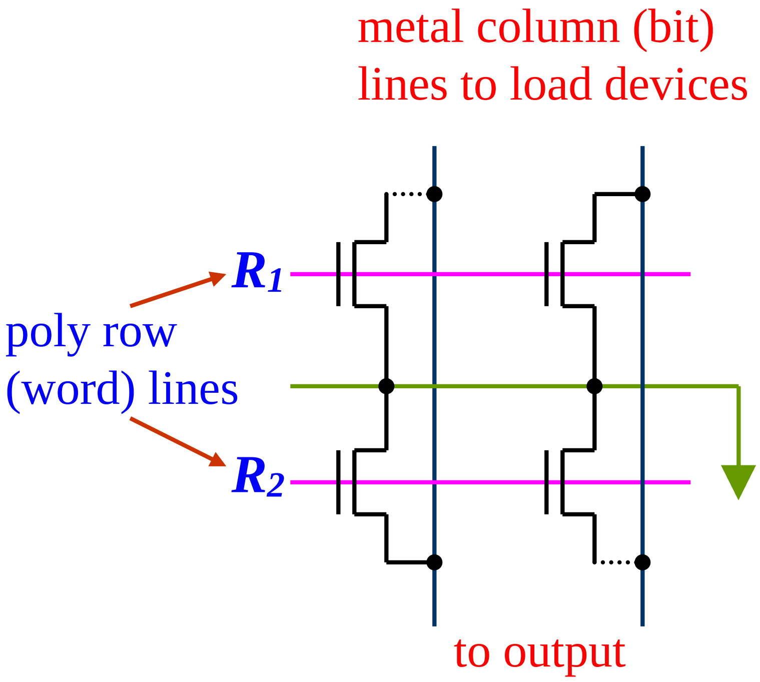


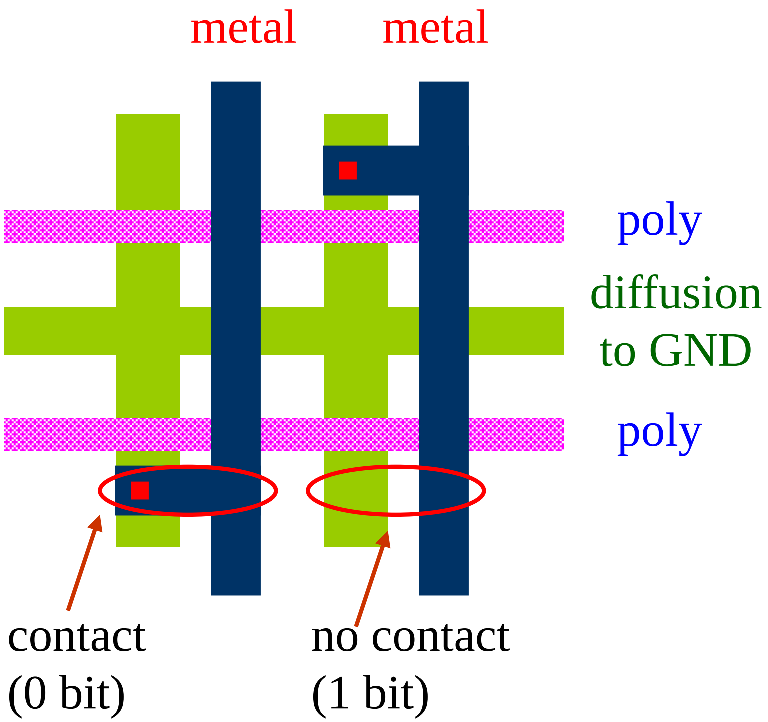
R_1	R_2	R_3	R_4	C_1	C_2	C_3	C_4
1	0	0	0	0	1	0	1
0	1	0	0	0	0	1	1
0	0	1	0	1	0	0	1
0	0	0	1	0	1	1	0

- One word line “R_i” is activated by raising its voltage to V_{DD}
- Logic “1” is stored: Absent transistor

  Logic “0” is stored: Present transistor
- To reduce static power consumption, the pMOS can be driven by a periodic pre-charge signal.

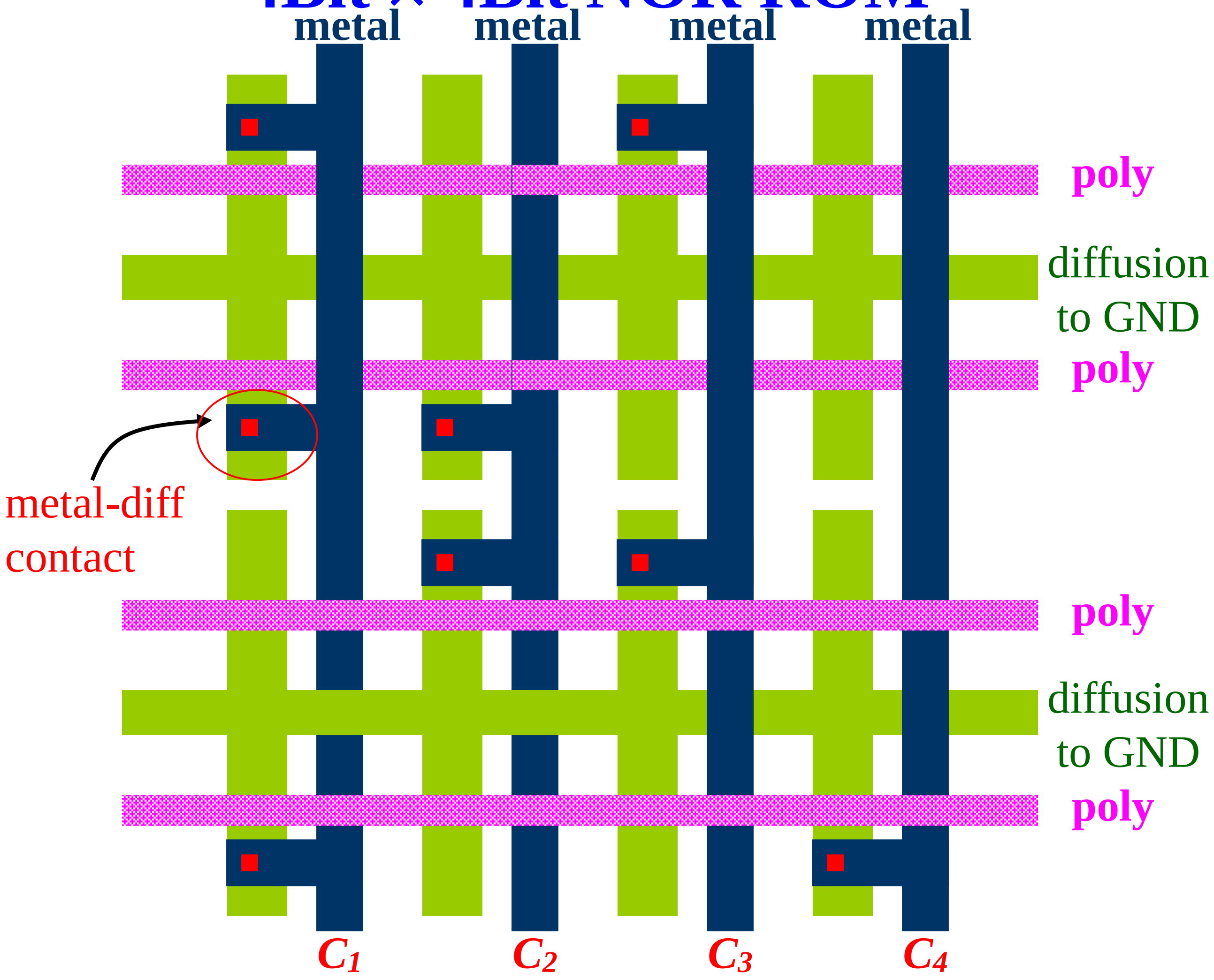
Layout of Contact-Mask Programmable NOR ROM





- **"0" bit:** drain is connected to metal line via a metal-to-diffusion contact
  "1" bit: omission the connect between drain and metal line.
- **To save silicon area:** the transistors on every two adjacent rows share a common ground line, also are routed in n-type diffusion

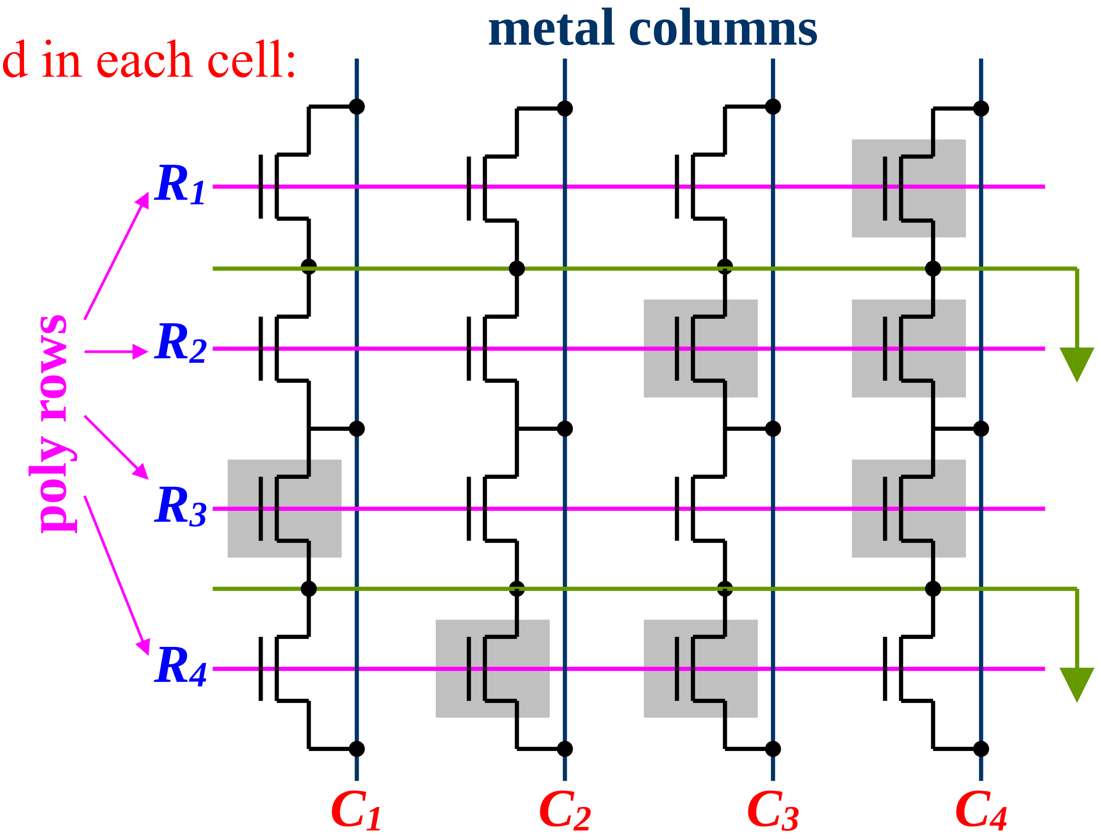
Layout of Contact-Mask Programmable 4Bit × 4Bit NOR ROM



- In reality, the metal lines are **laid out directly on top** of diffusion columns to reduce the horizontal dimension.

Implant-Mask Programmable NOR ROM Array

Logic “0” is stored in each cell:
Present transistor

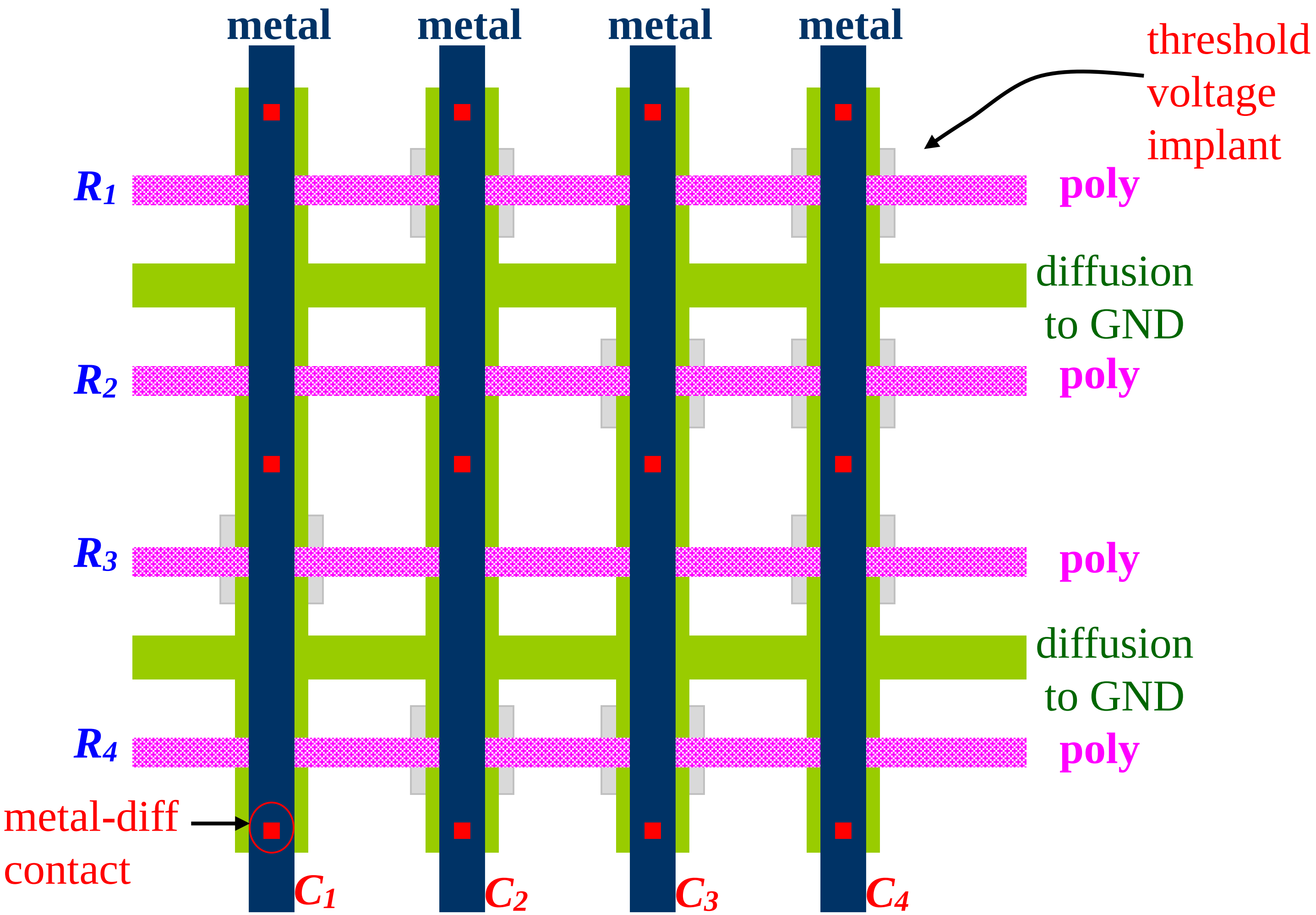


- V_{T0} is implanted to activate 1 bit:

  Let $V_{T0} > V_{DD}$ ⇒ permanently **turn off** transistor

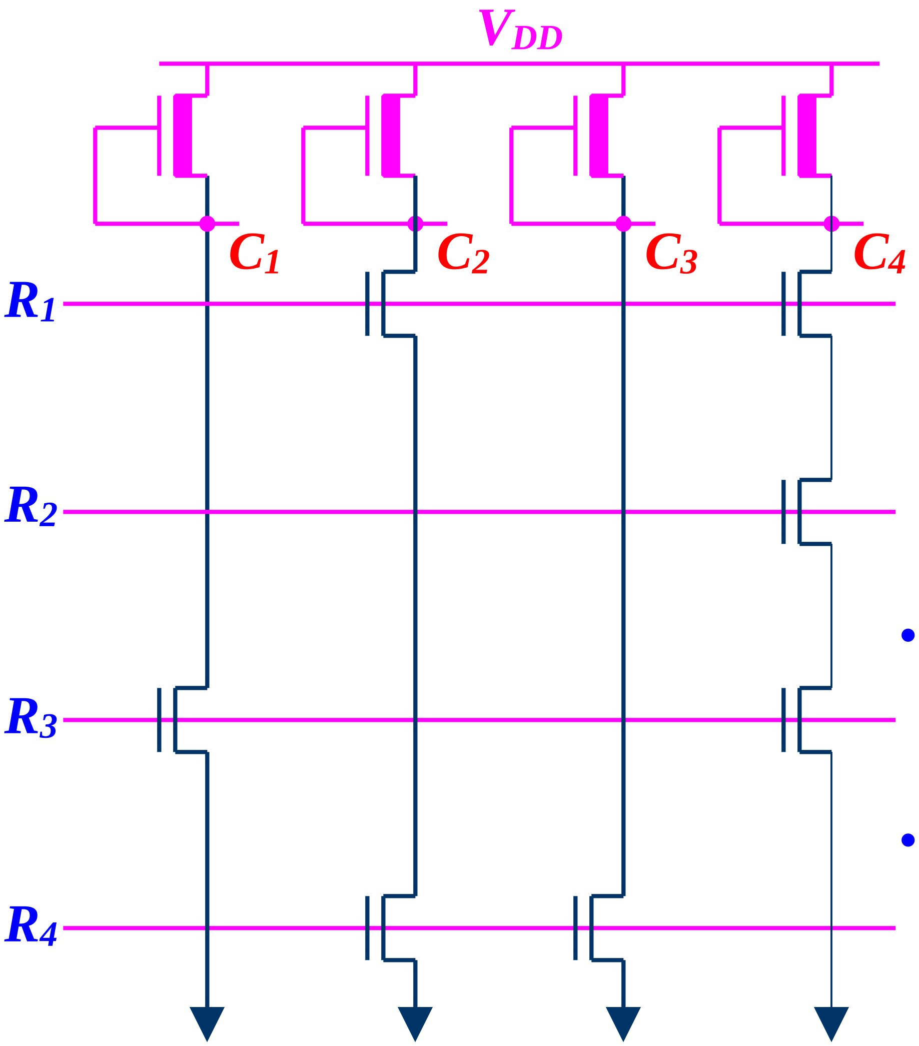
  ⇒ disconnect the contact

Layout of Implant-Mask Programmable 4Bit × 4Bit NOR ROM



- Each diffusion-to-metal contact is **shared by two adjacent transistors** ⇒ need smaller area than contact-mask ROM layout

4Bit × 4Bit NAND-based ROM Array

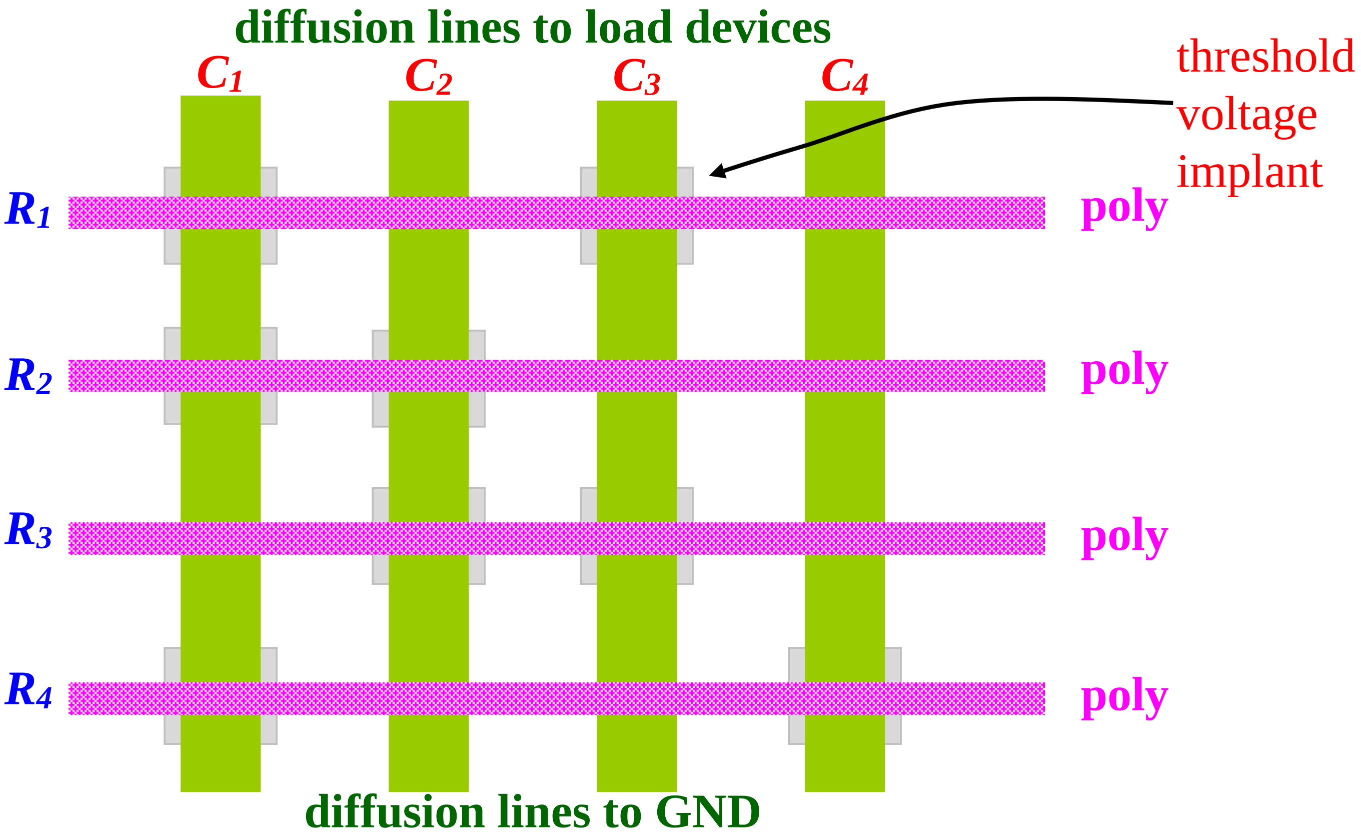


R_1	R_2	R_3	R_4	C_1	C_2	C_3	C_4
0	1	1	1	0	1	0	1
1	0	1	1	0	0	1	1
1	1	0	1	1	0	0	1
1	1	1	0	0	1	1	0

- All word lines are kept at logic “1” level, except the selected line pulled down by “0” level.
- Logic “0” is stored: Absent transistor

  Logic “1” is stored: Present transistor

Layout of Implant-Mask Programmable 4Bit × 4Bit NAND ROM

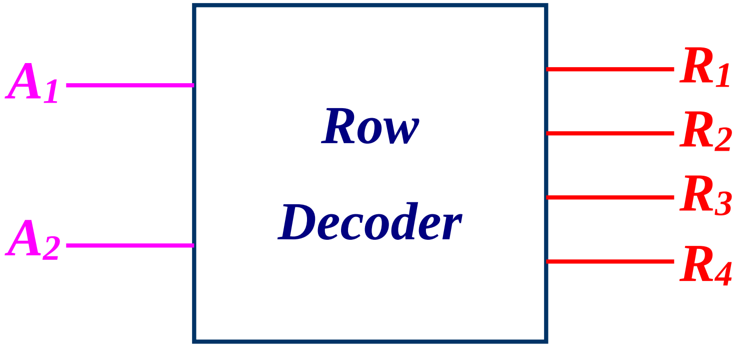


- No contact in the array ⇒ **More compact than NOR ROM array**
- Series-connected nMOS transistors exist in each column

⇒ **The access time is slower than NOR ROM**

Design of Row and Column Decoders

- Row and Column Decoders: To select **a particular memory location** in the array.

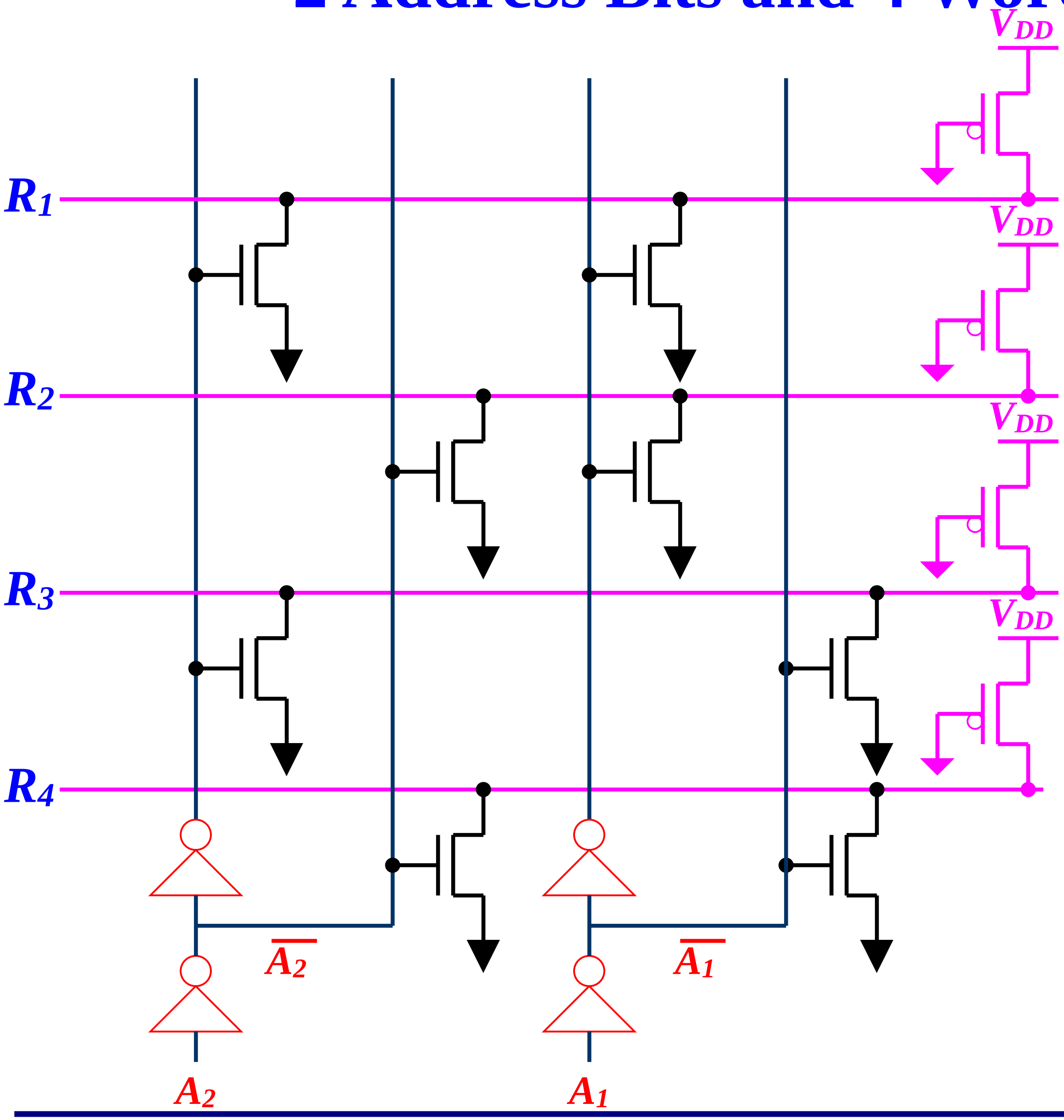


2 address bits

⇒ 4 word lines

A_1	A_2	R_1	R_2	R_3	R_4
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

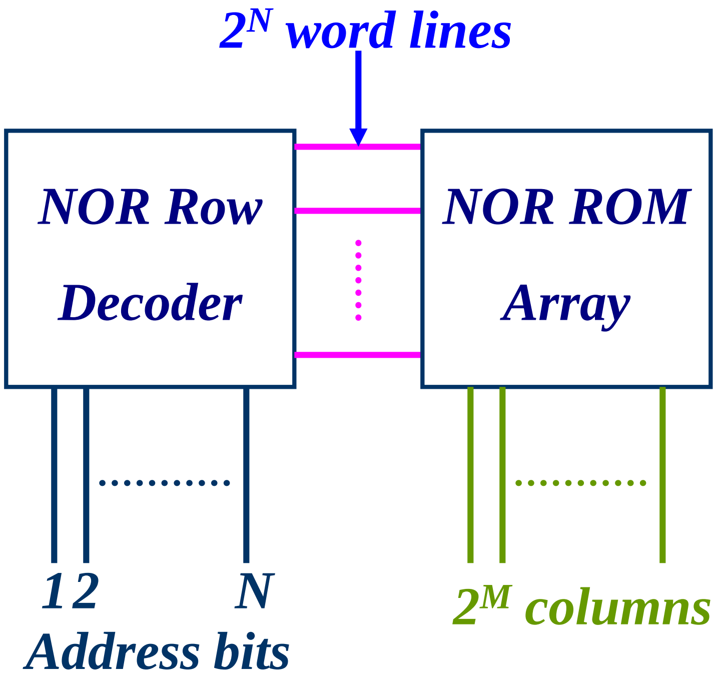
NOR-based Row Decoder Circuit
2 Address Bits and 4 Word Lines



A_1	A_2	R_1	R_2	R_3	R_4
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

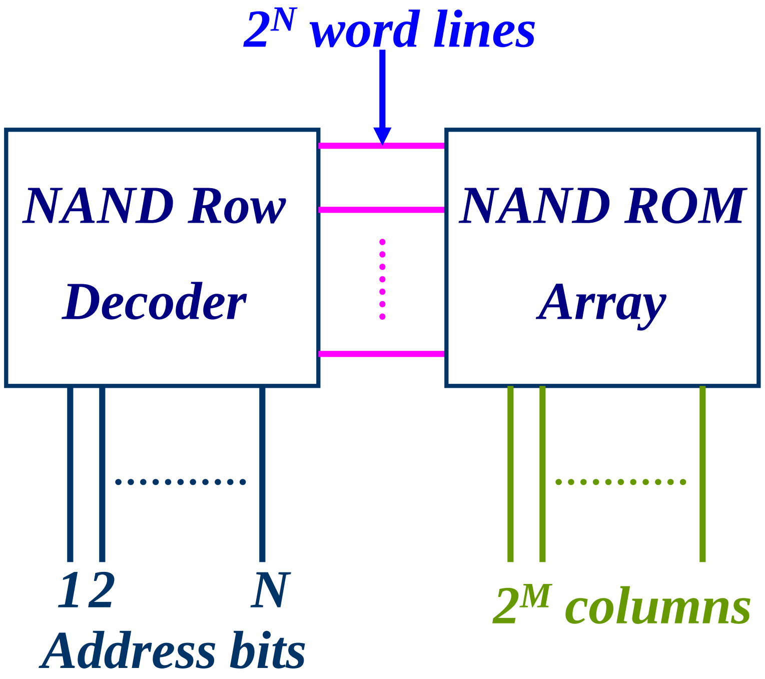
Implementation of Row Decoder and ROM

- Can be implemented as ***two adjacent*** NOR planes



Implementation of Row Decoder and ROM (Cont.)

- Can also be implemented as ***two adjacent*** NAND planes

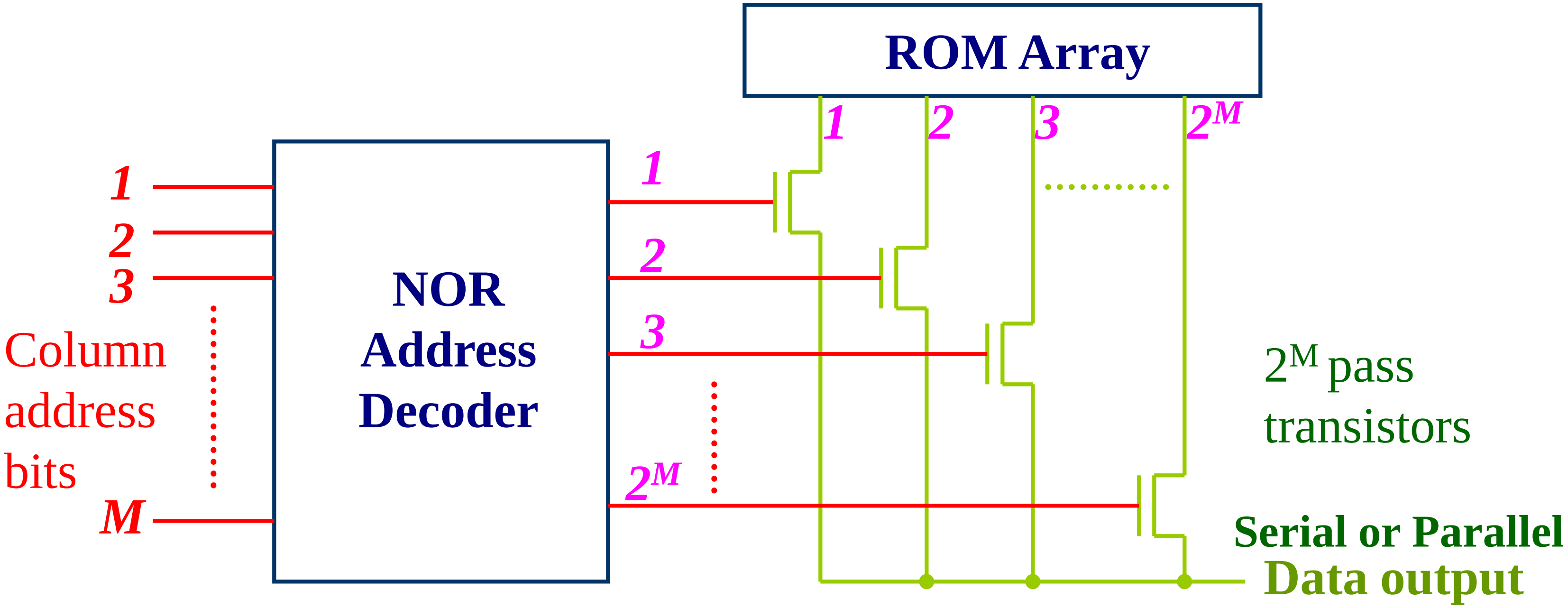


A_1	A_2	R_1	R_2	R_3	R_4
0	0	0	1	1	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

4×4 NAND ROM Array

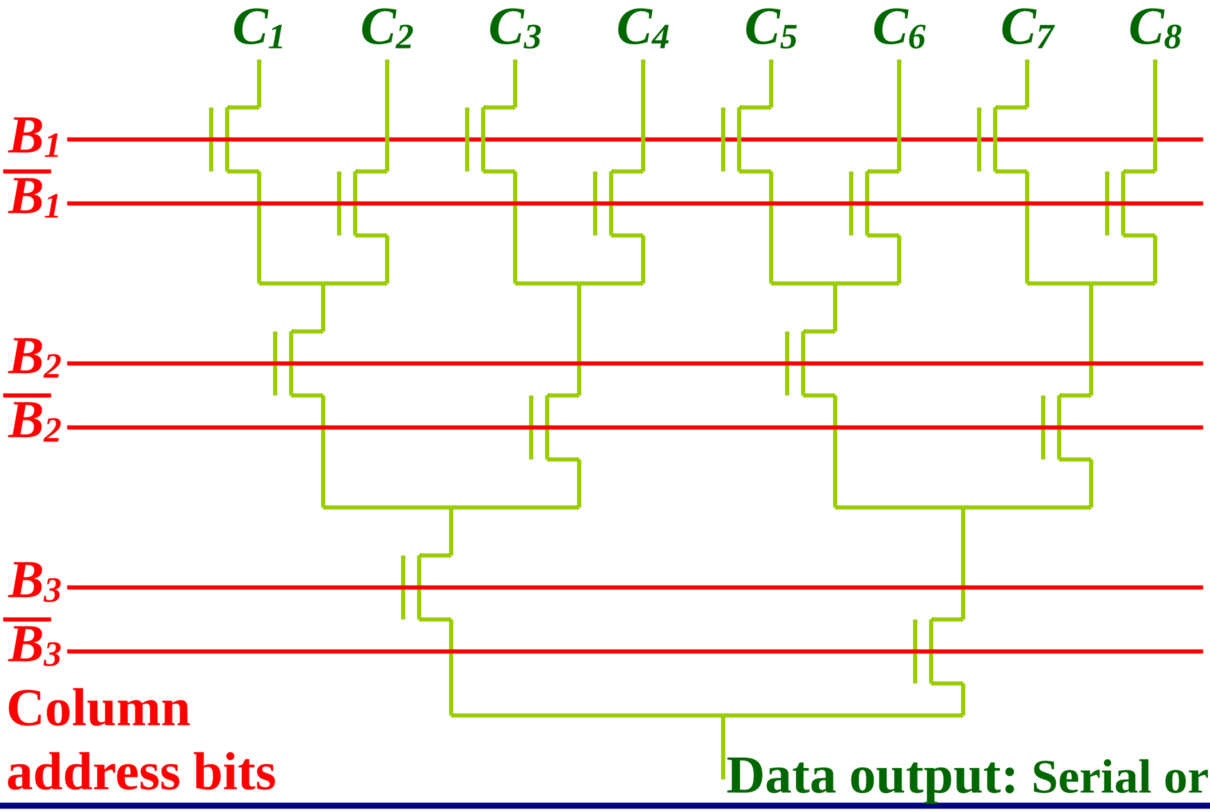
Column Decoder (1)
NOR Address Decoder and Pass Transistors

- **Column Decoder:** To select one out of 2^M bits lines of the ROM array, and to route the data of the selected bit line to the data output
- **NOR-based column address decoder and pass transistors:**
  - » Only one nMOS pass transistor is turned on at a time
  - » # of transistors required: $2^M(M+1)$ (2^M pass transistors, $M2^M$ decoder)



Column Decoder (2)
Binary Tree Decoder

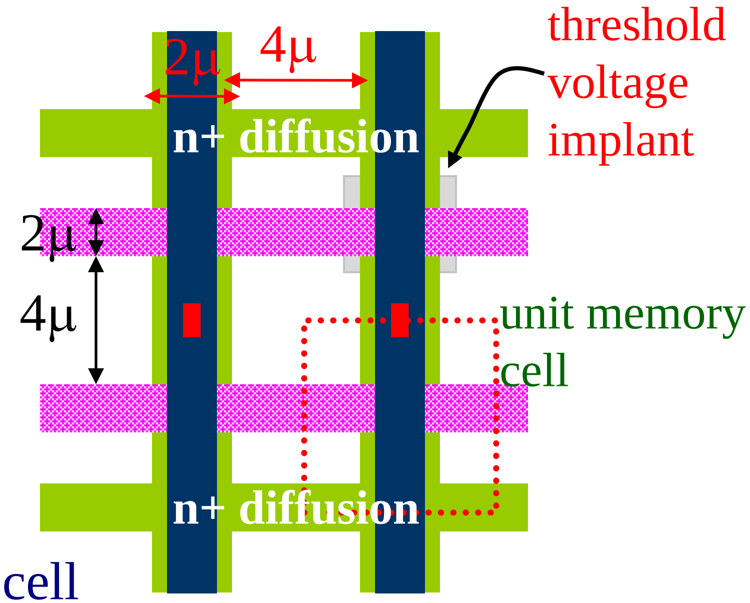
- **Binary Tree Decoder: A binary selection tree with consecutive stages**
  - » The pass transistor network is used to select one out of every two bit lines at each stages. The NOR address decoder is not needed.
  - » **Advantage:** ***Reduce the transistor count*** (2^{M+1}-$2+2M$)
  - » **Disadvantage:** Large number of series connected nMOS pass transistors ⇒ ***long data access time***



Data output: Serial or Parallel

An Example of NOR ROM Array

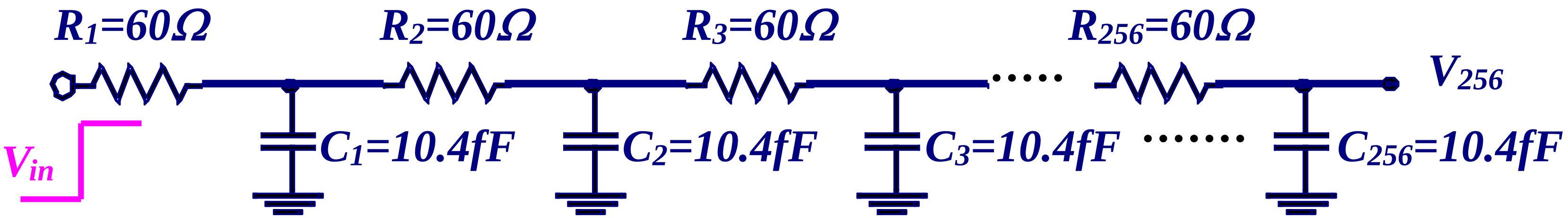
- Consider the design of a 32-kbit **NOR ROM** array and the design issues related to ***access time analysis***
  - » # of total bits: 15 (2^{15}=32,768)
  - » 7 row address bits (2^7 = 128 rows)
  - » 8 column address bits (2^8 = 256 columns)
  - » Layout: implant-mask
  - » W = 2 μm, L = 1.5 μm
  - » $\mu_n C_{ox}$ = 20 μA/V^2
  - » C_{ox} = 3.47 μF/cm^2
  - » $R_{sheet\text{-}poly}$ = 20 Ω/square



- R_{row}, and C_{row} / unit memory cell
  - » $C_{row} = C_{ox} \cdot W \cdot L = 10.4\ fF/bit$
  - » R_{row} = (# of squares) × $R_{sheet\text{-}poly}$ = $3 \times 20 = 60\ \Omega$

An Example of NOR ROM Array (Cont.)

- The poly word line can be modeled as a RC transmission line with up to 256 transistors

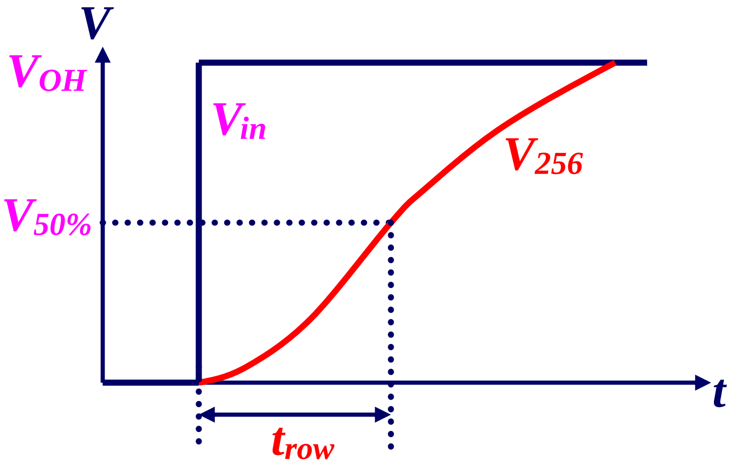


- The row access time t_{row}: delay associated with selecting and activating 1 of 128 word lines in ROM array. It can be approximated as

$$t_{row} \approx 0.38 \cdot R_T \cdot C_T = 15.53 \text{ ns}$$

$$R_T = \sum_{\text{all cols}} R_i = 15.36 \text{ k}\Omega$$

$$C_T = \sum_{\text{all cols}} C_i = 2.66 \text{ pF}$$

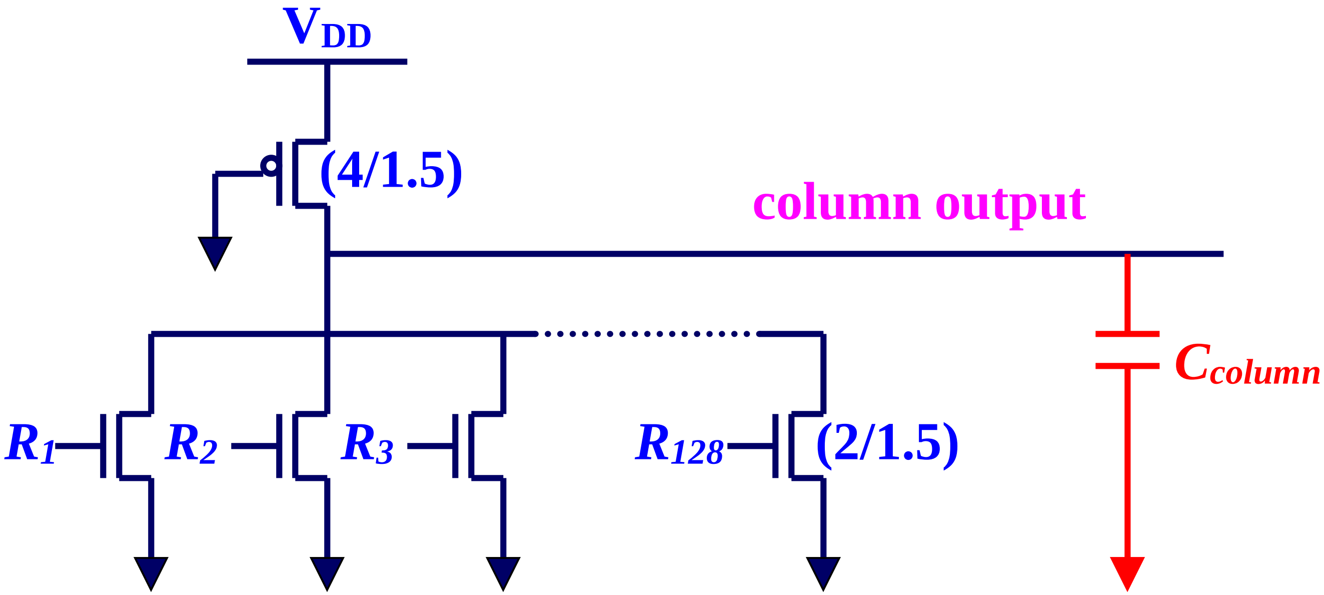


An Example of NOR ROM Array (Cont.)

- A **more accurate** RC delay value: ***Elmore time constant*** for RC ladder circuits

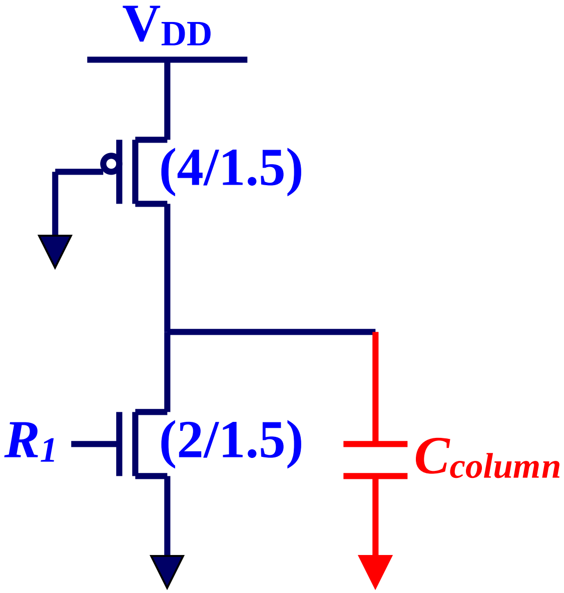
  $t_{row} = \sum_{k=1}^{256} R_{jk} C_k = 20.52$ ns

- The column access time t_{column}: worst case delay τ_{PHL} associated with discharging the precharged bit line when a row is activated.



An Example of NOR ROM Array (Cont.)

- $C_{column} = 128 \times (C_{gd,driver}+C_{db,driver}) \approx 1.5\text{pF}$
  where $C_{gd,driver}+C_{db,driver} = 0.0118$ pF/word line
- Since only one word line is activated at a time, the above circuit can be reduced to an inverter circuit



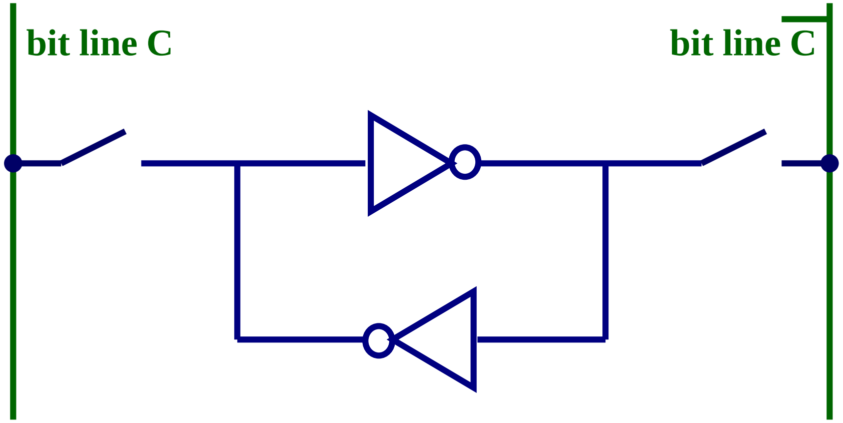
Remark: τ_{PLH} is not considered because the bit line is precharged high before each row access operation

$$t_{column} = \tau_{PHL} = \frac{C_{load}}{k_n(V_{OH}-V_{T0,n})}\left[\frac{2V_{T0,n}}{V_{OH}-V_{T0,n}} + \ln\left(\frac{4(V_{OH}-V_{T0,n})}{V_{OH}+V_{OL}} - 1\right)\right] = 18ns$$

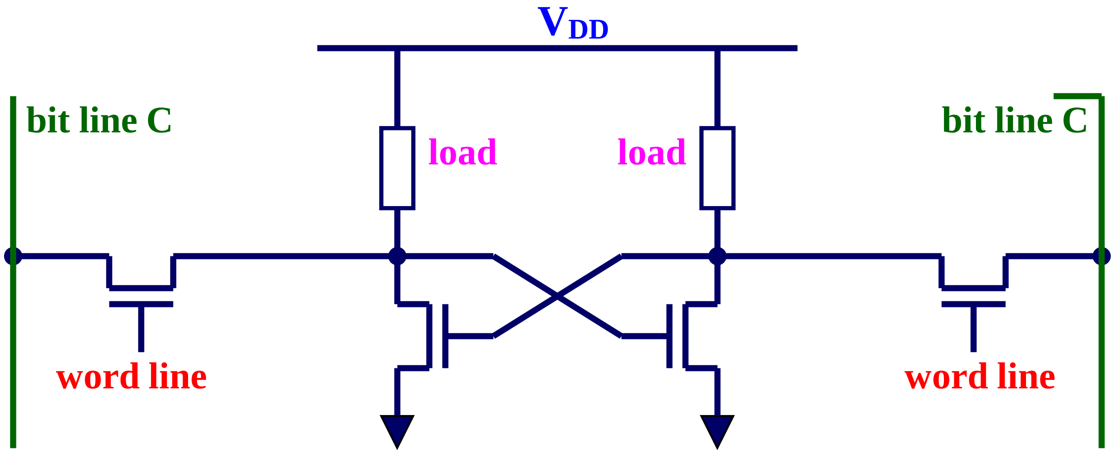
$$t_{access} = t_{row} + t_{column} = 20.52 + 18 = 38.52\ ns$$

Static Random Access Memory (SRAM)

- **SRAM:** The stored data can be retained indefinitely, without any need for a periodic refresh operation.

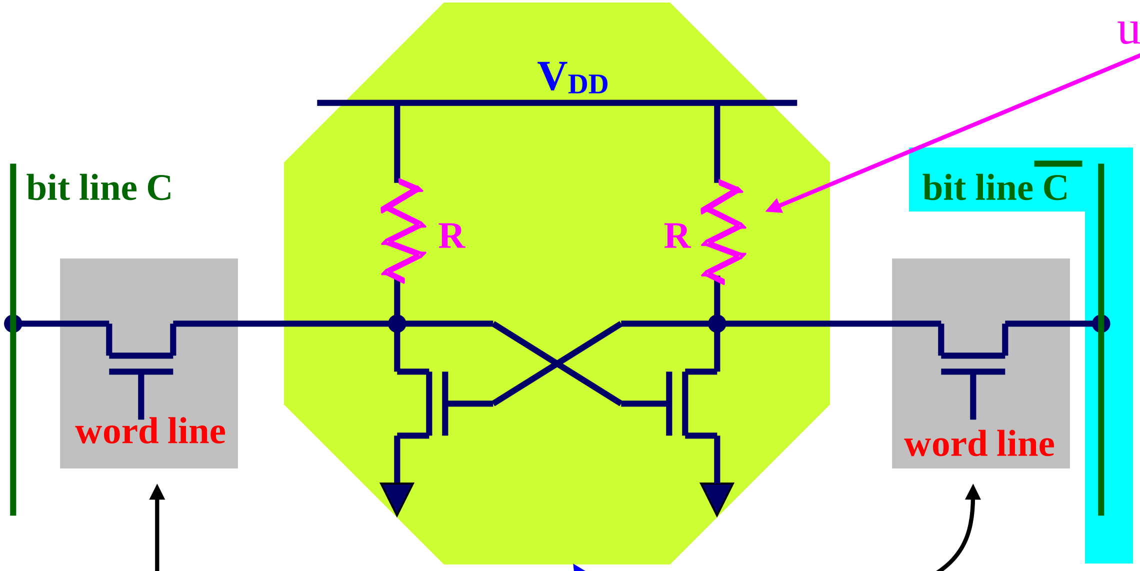


1-bit SRAM cell



- **Complementary Column** arrangement is to achieve a more reliable SRAM operation

Resistive-Load SRAM Cell



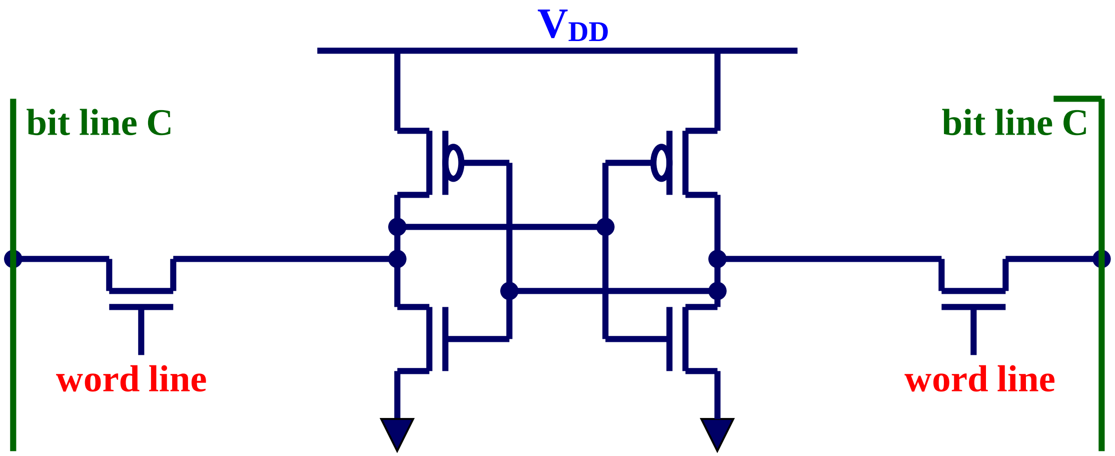
undoped polysilicon resistor

SRAM cell is accessed via two bit (column) lines C and its complement for reliable operation

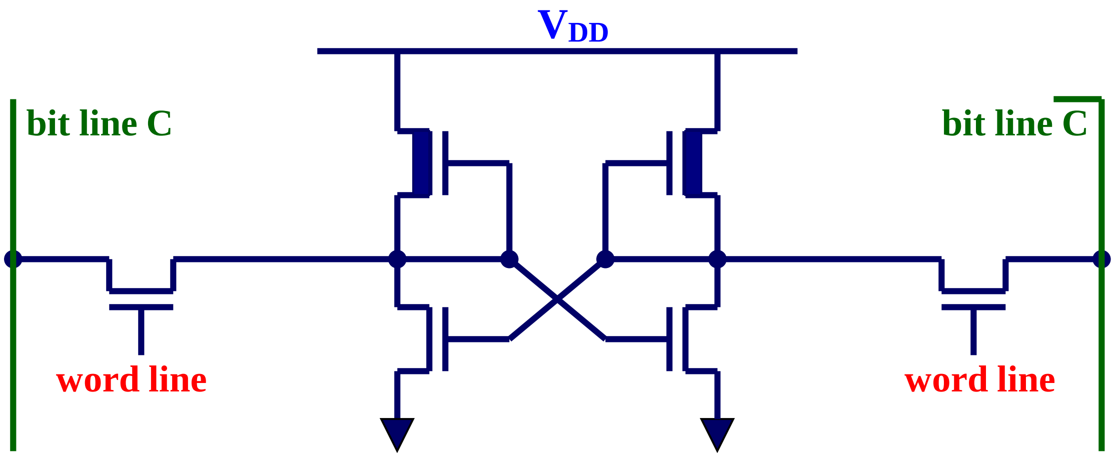
pass transistors to activated by a row select (RS) signal to enable read/write operators

Basic cross-coupled 2-inverter latch with 2 stable op points for storing one-bit

Full CMOS and Depletion-Load SRAM Cell

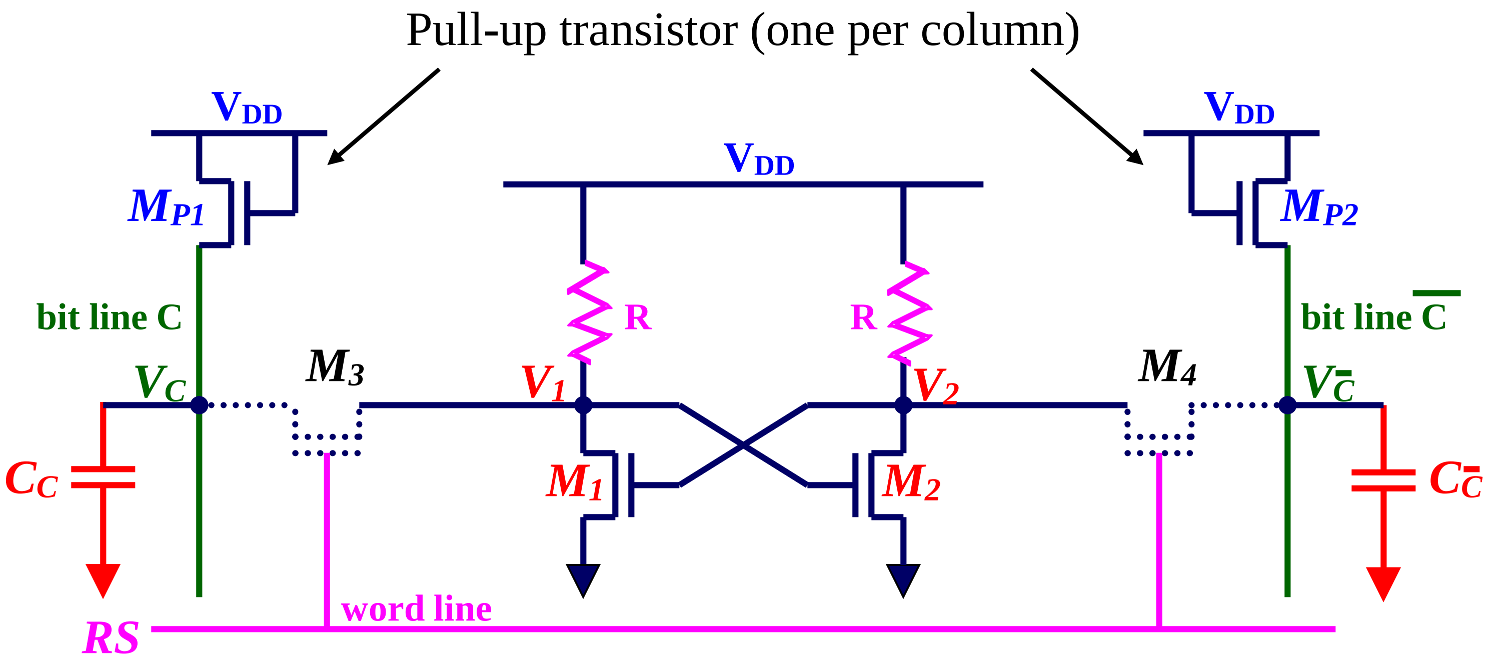


Full CMOS SRAM Cell



Depletion-Load SRAM Cell

SRAM Operation Principles



Pull-up transistor (one per column)

- ***RS=0*:** The word line is not selected. ***M₃*** and ***M₄*** are OFF

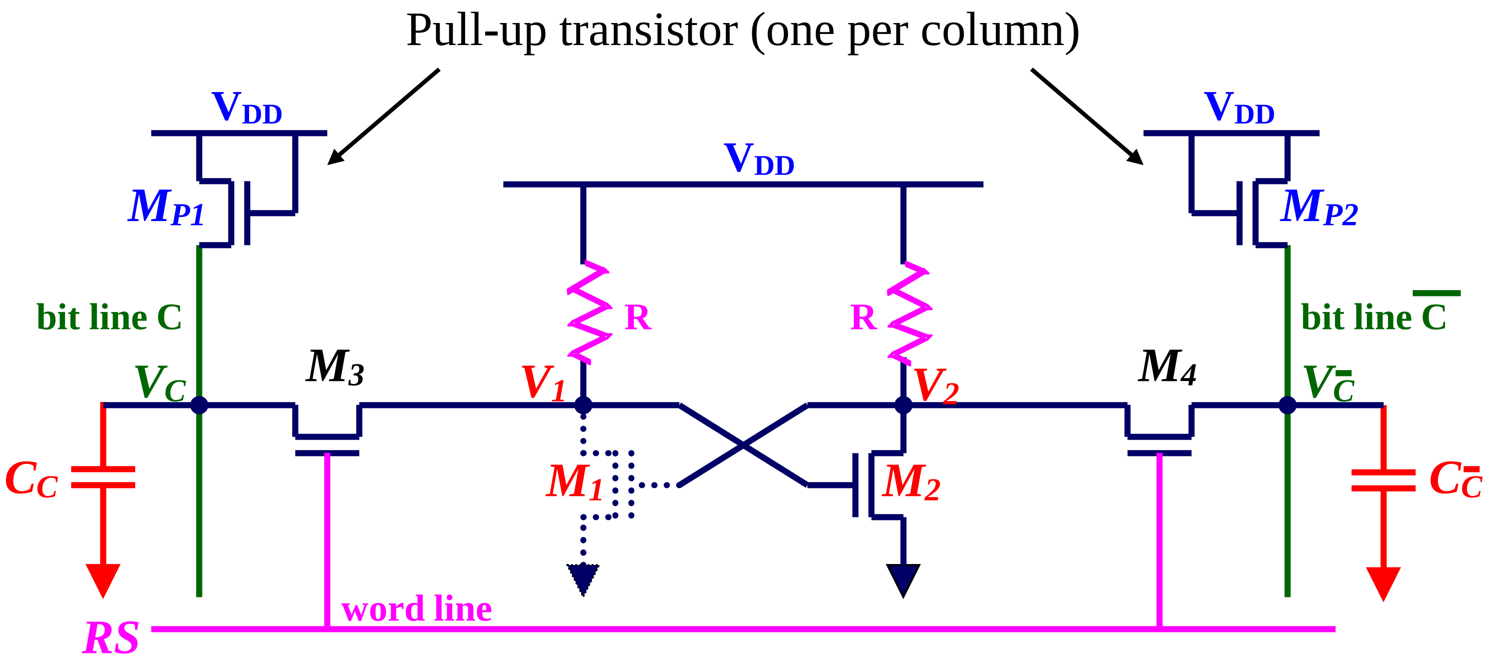
➢ One data-bit is held: The latch preserves one of its two stable states.

➢ ***If RS=0 for all rows*:** ***C_C*** and ***$C_{\bar{C}}$*** are charged up to near ***V_{DD}*** by pulling up of ***M_{P1}*** and ***M_{P2}*** (both in saturation)

$$V_{\bar{C}} = V_C = V_{DD} - \left(V_{T0} + \gamma\sqrt{|2\phi_F| + V_C} - \sqrt{|2\phi_F|}\right)$$

➢ Ex: **V_C = $V_{\bar{C}}$=3.5V** for ***V_{DD}*** = 5V, ***V_{T0}***=1V, |2ϕ_F|=0.6V, =0.4V$^{1/2}$

SRAM Operation Principles (Cont.)



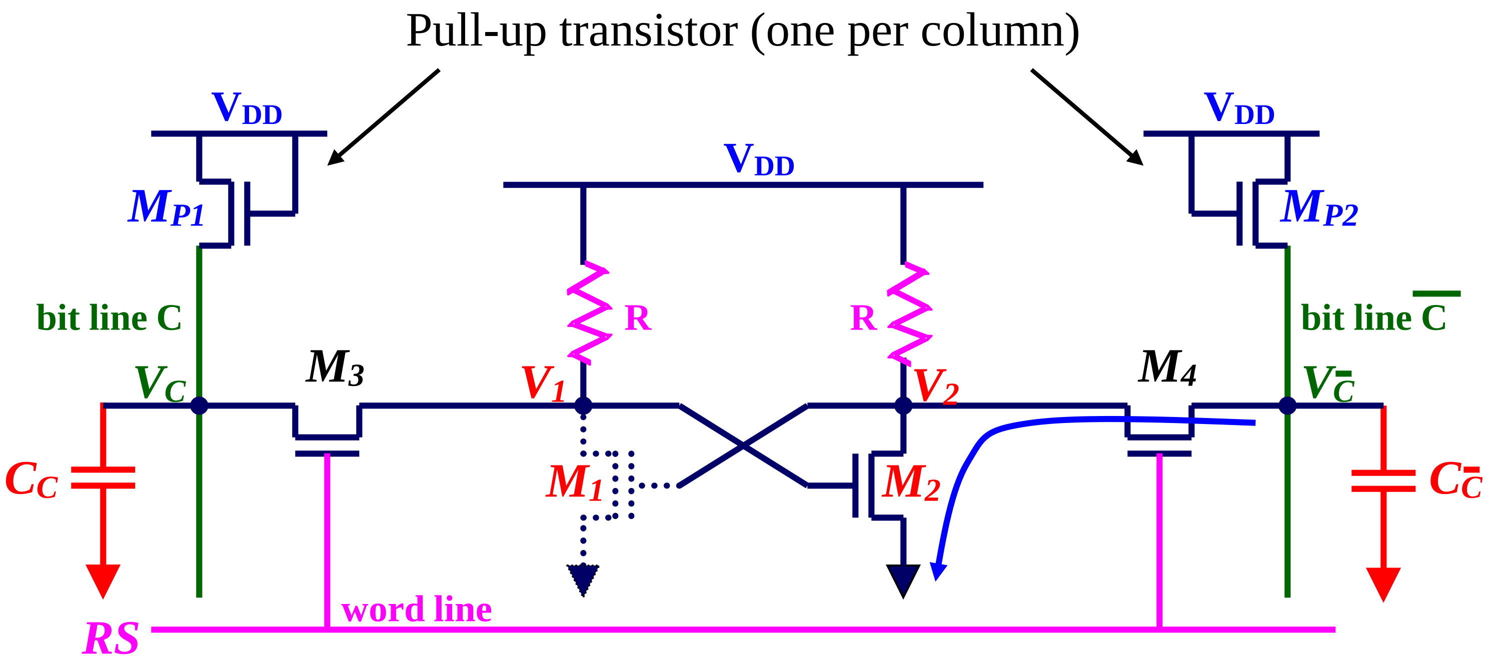
- ***RS*=1:** The word line is now selected. M_3 and M_4 are ON

Four Operations

1. **Write "1" Operation** (V_1=V_{OL}, V_2=V_{OH} at $t=0^-$):

   $V_{\overline{C}} \Rightarrow V_{OL}$ by the ***data-write circuitry***. Therefore, $V_2 \Rightarrow V_{OL}$, then M_1 turns ***off*** $V_1 \Rightarrow V_{OH}$ and M_2 turns on pulling down $V_2 \Rightarrow V_{OL}$.

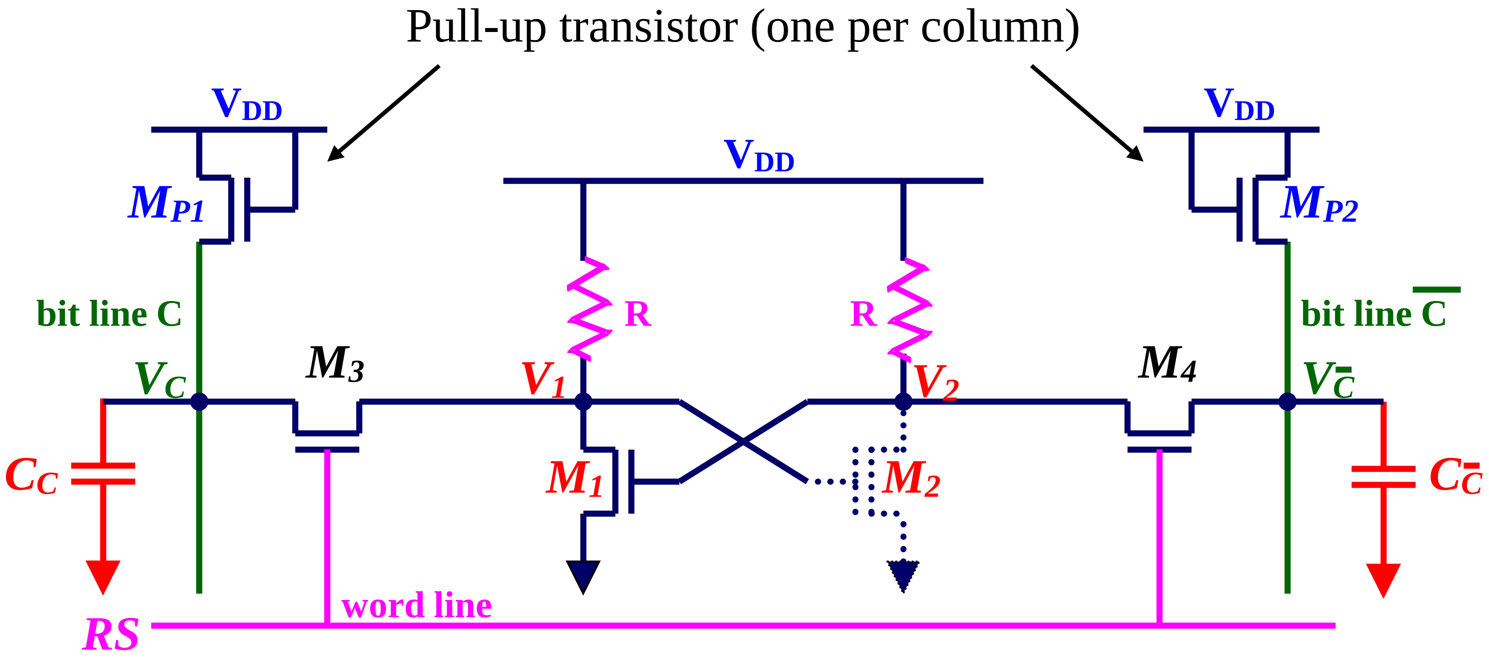
SRAM Operation Principles (Cont.)



2. **Read "1" Operation** (V_1=V_{OH}, V_2=V_{OL} at t=0^-):

V_C retains pre-charge level, while $V_{\bar{C}} \Rightarrow V_{OL}$ by M_2 **ON**. ***Data-read circuitry*** detects small voltage difference $V_C - V_{\bar{C}} > 0$, and amplifies it as a "*1*" data output.

SRAM Operation Principles (Cont.)

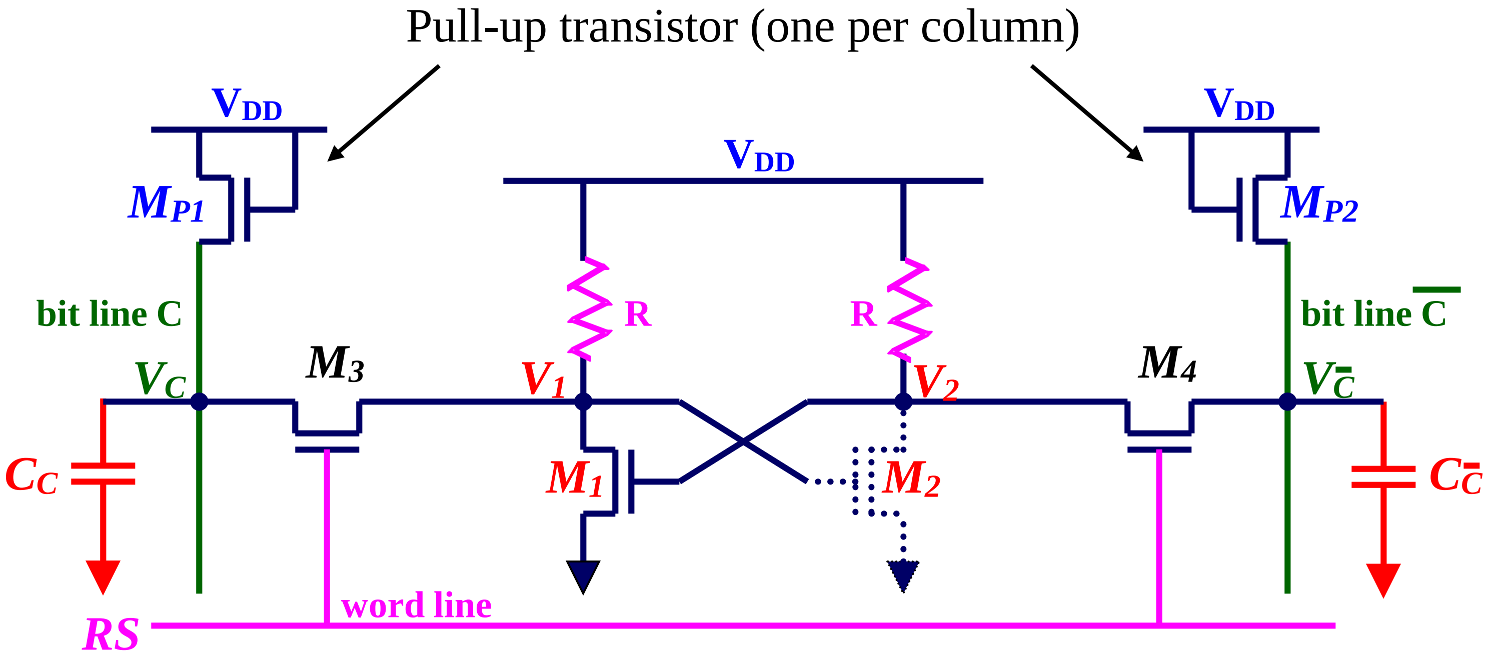


3. **Write "0" Operation** (V_1=V_{OH}, V_2=V_{OL} at t=0^-):

   $V_C \Rightarrow V_{OL}$ by the ***data-write circuitry***.

   Since $V_1 \Rightarrow V_{OL}$, M_2 turns off, therefore $V_2 \Rightarrow V_{OH}$.

SRAM Operation Principles (Cont.)



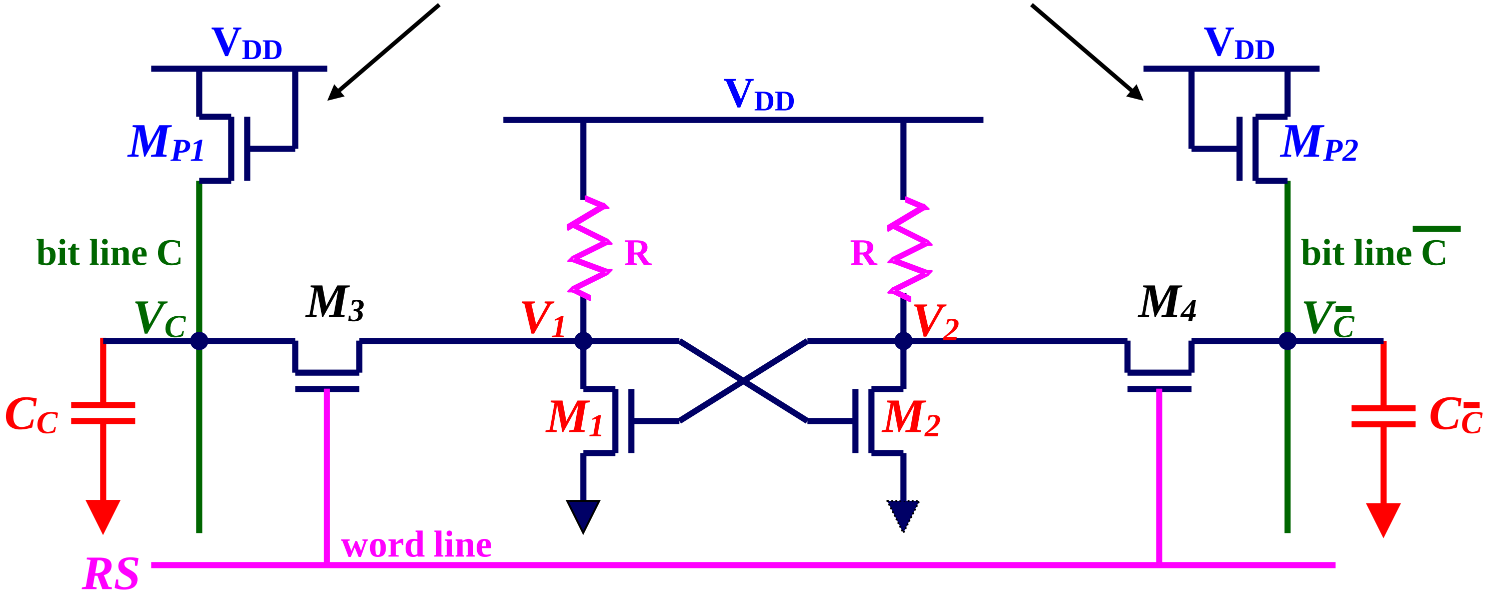
4. **Read "0" Operation** (V_1=V_{OL}, V_2=V_{OH} at $t=0^-$):

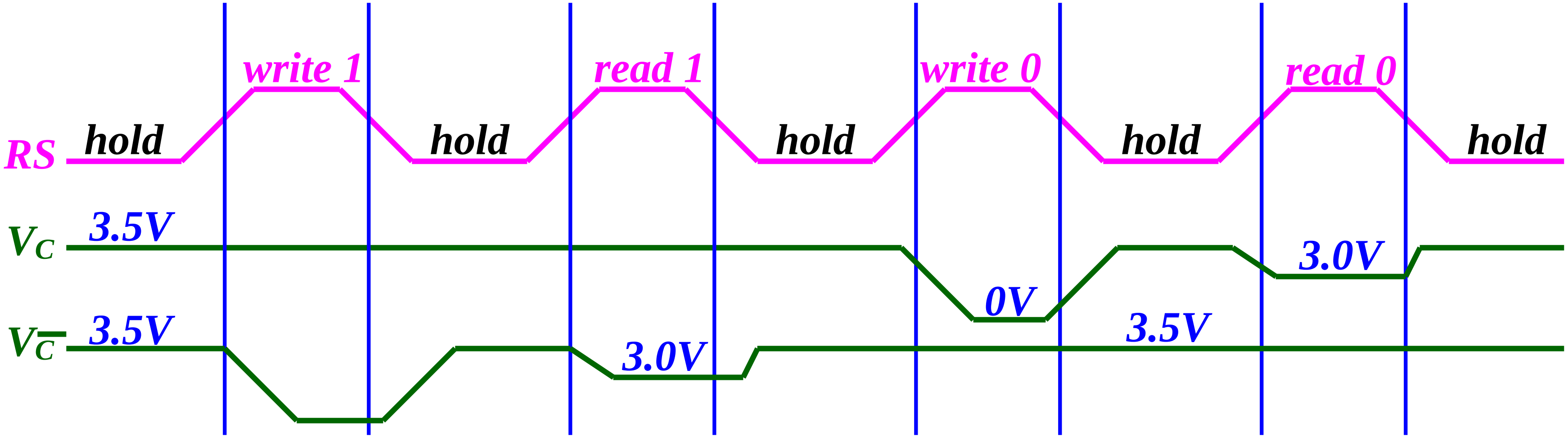
$V_{\overline{C}}$ retains pre-charge level, while $V_C \Rightarrow V_{OL}$ by ***M₁ ON***.

Data-read circuitry detects small voltage difference $V_C - V_{\overline{C}} < 0$, and amplifies it as a "*0*" data output.

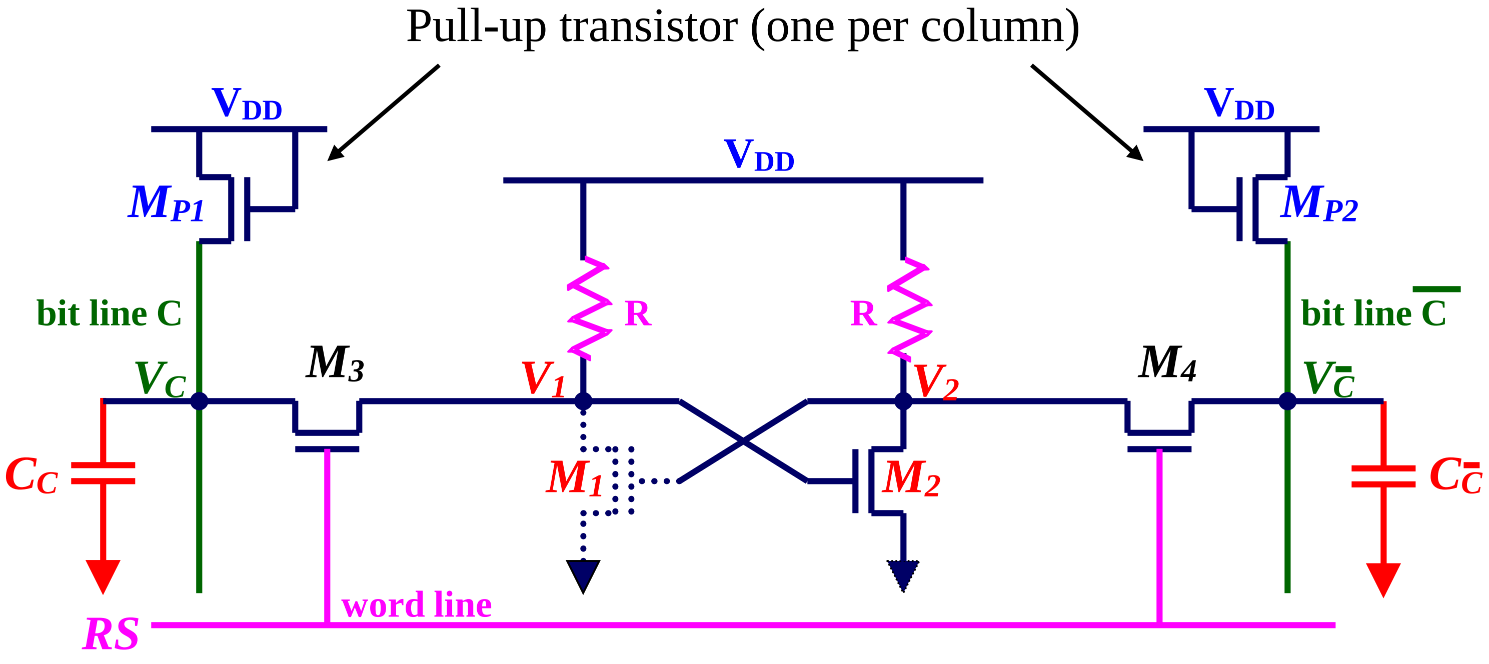
SRAM Operation Principles (Cont.)

Pull-up transistor (one per column)





Static or "Standby" Power Consumption

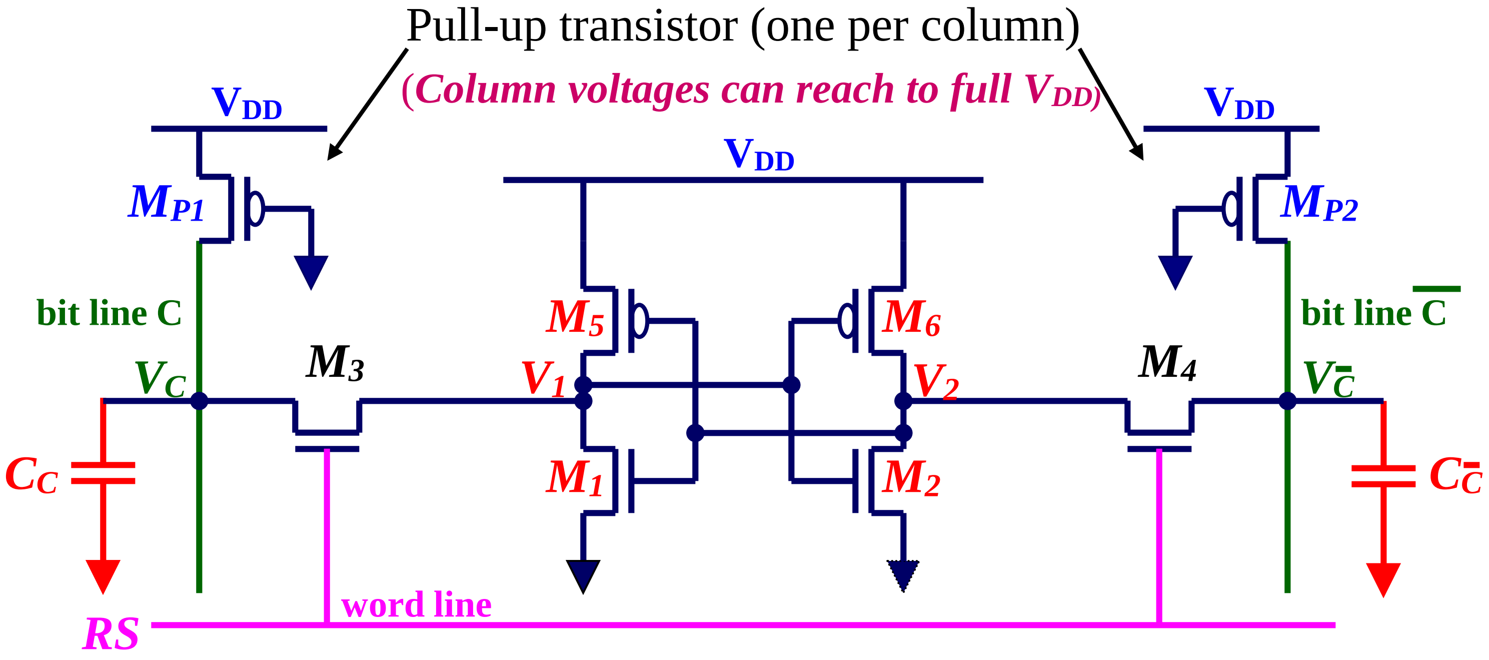


- ***Assume:*** 1 bit is stored in the cell ⇒ ***M_1 OFF, M_2 ON*** ⇒ ***V_1=V_{OH}, V_2=V_{OL}***. *I.E. One load resistor is always conducting non-zero current.*

$$P_{standby} = (V_{DD}-V_{OL})^2/R$$

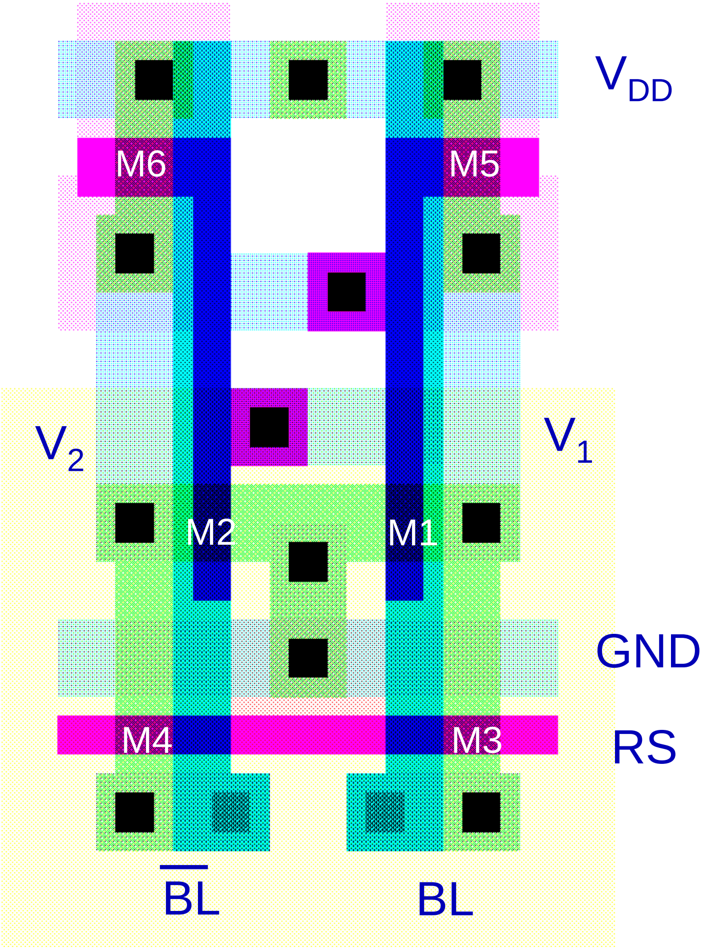
with R = 100MΩ (undoped poly), $P_{standby}$ ≈ 0.25 μW per cell for V_{DD} =5V

Circuit of CMOS SRAM Cell



- **Advantages**
  - Very **low standby power** consumption
  - **Large noise margins** than ***R***-load **SRAMS**
  - **Operate at lower supply voltages** than ***R***-load **SRAMS**
- **Disadvantages**
  - **Larger die area**: To accommodate the n-well for pMOS transistors and polysilicon contacts. The area has been reduced by using multi-layer polysilicon and multi-layer metal processes
  - **CMOS more complex process**

6T-SRAM — Layout



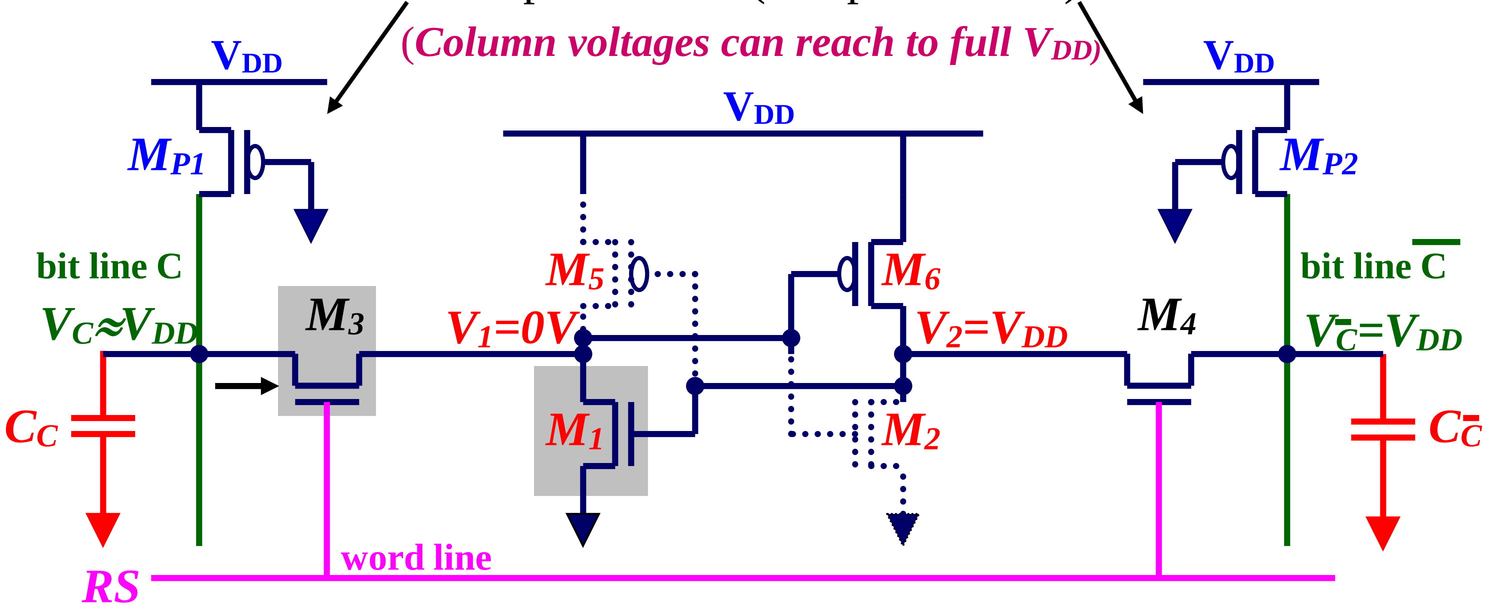
Source: Digital Integrated Circuits 2nd

CMOS SRAM Cell Design strategy

- Two basic requirements which dictate ***W/L*** ratios
  1. Data-read operation should **not destroy data** in the cell
  2. **Allow modification** of stored data during data-write operation

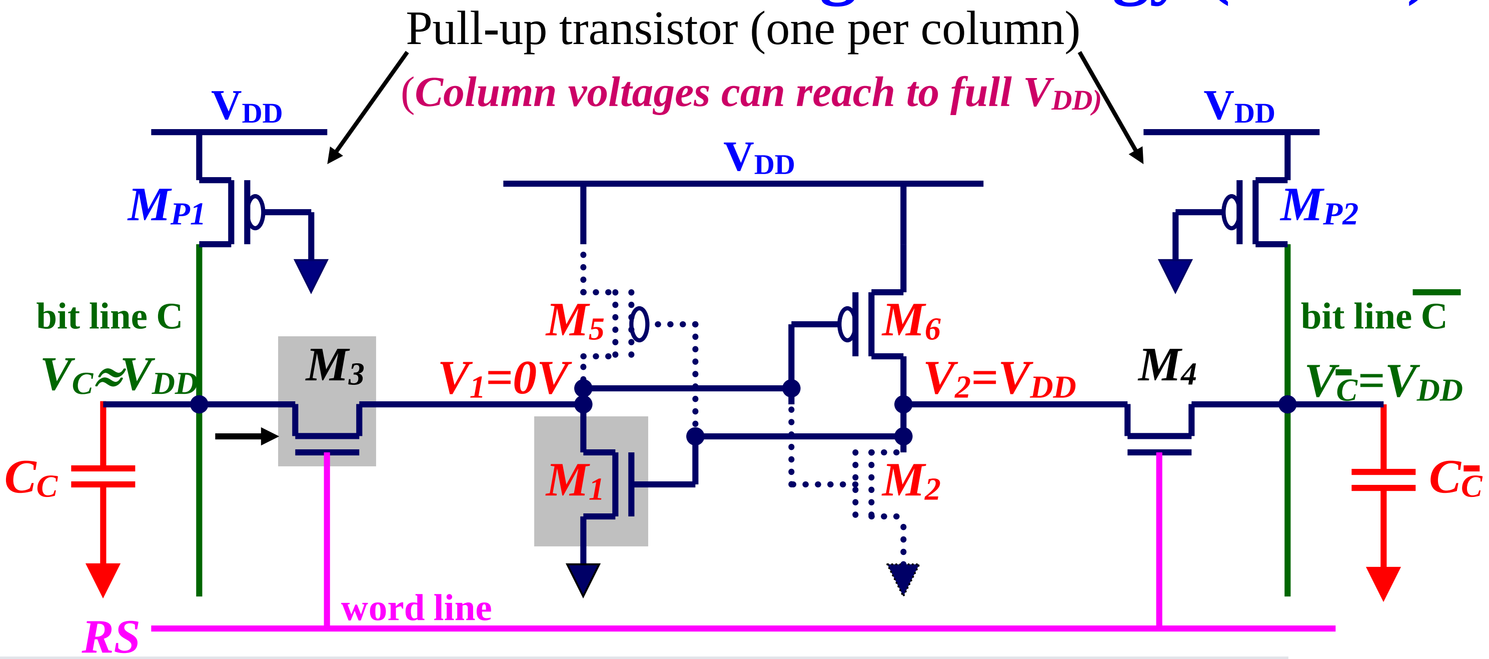
Pull-up transistor (one per column)

(Column voltages can reach to full V_{DD})



- **Read "0" operation**
  - » at $t=0^-$: $V_1 = 0V$, $V_2 = V_{DD}$; M_3, M_4 OFF; M_2, M_5 OFF; M_1, M_6 Linear
  - » at $t=0$: $RS = V_{DD}$, M_3 Saturation, M_4 Linear; M_2, M_5 OFF; M_1, M_6 Linear
    - **Slow discharge of large C_C**: Require $V_1 < V_{T,2}$ ⇒**Limits** M_3 W/L wrt M_1 W/L

CMOS SRAM Cell Design Strategy (Cont.)



- **Design Constraint:** $V_{1,max} < V_{T,2} = V_{T,n}$ to keep M_2 **OFF**
  - » *M_3 saturation, M_1 linear* $\Rightarrow$

$$k_{n,3}(V_{DD}-V_1-V_{T,n})^2/2 = k_{n,1}(2(V_{DD}-V_{T,n})V_1-V_1^2)/2$$

  - » Therefore,

$$\frac{k_{n,3}}{k_{n,1}} = \frac{\left(\frac{W}{L}\right)_3}{\left(\frac{W}{L}\right)_1} < \frac{2(V_{DD}-1.5V_{T,n})V_{T,n}}{(V_{DD}-2V_{T,n})^2}$$

Symmetry:

Same for $k_{n,4}/k_{n,2}$

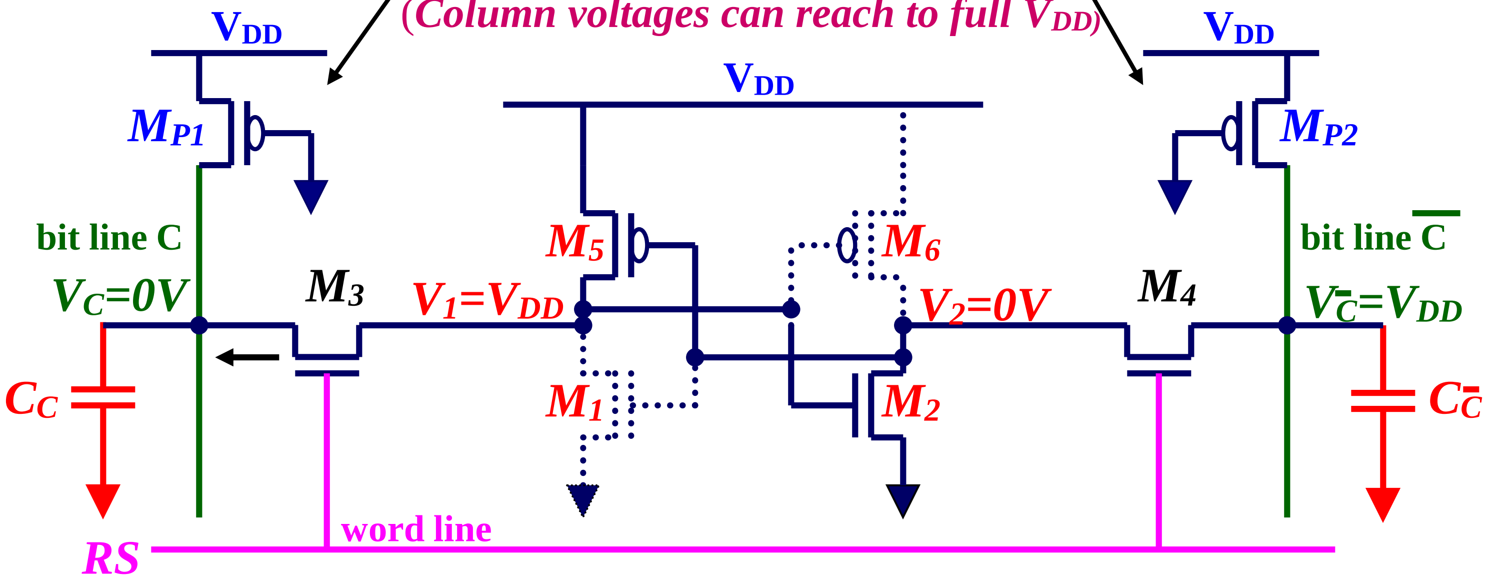
(M_1 **OFF** for Read "1")

CMOS SRAM Cell Design Strategy (Cont.)

- Write "0" operation with "1" stored in cell:

Pull-up transistor (one per column)

(*Column voltages can reach to full V_{DD}*)



- V_C is set "0" *by data-write circuit*

("1" stored)

- at $t=0^-$: $V_1=V_{DD}$, $V_2=0V$; M_3, M_4 OFF; M_2, M_5 Linear; M_1, M_6 OFF
- at $t=0$: $V_C=0V$, $V_{\overline{C}}=V_{DD}$; M_3, M_4 saturation; M_2, M_5 Linear; M_1, M_6 OFF
  - » Write "0"⇒V_1: $V_{DD}\rightarrow 0(<V_{2T,n})$ and $V_2: 0\rightarrow V_{DD}(M_2\rightarrow OFF)$

CMOS SRAM Cell Design Strategy (Cont.)

- **Design constraint:** $V_{1,max} < V_{T,2} = V_{T,n}$ to keep M_2 **OFF**
  - » When $V_1 = V_{T,n}$: M_3 Linear and M_5 saturation ⇒

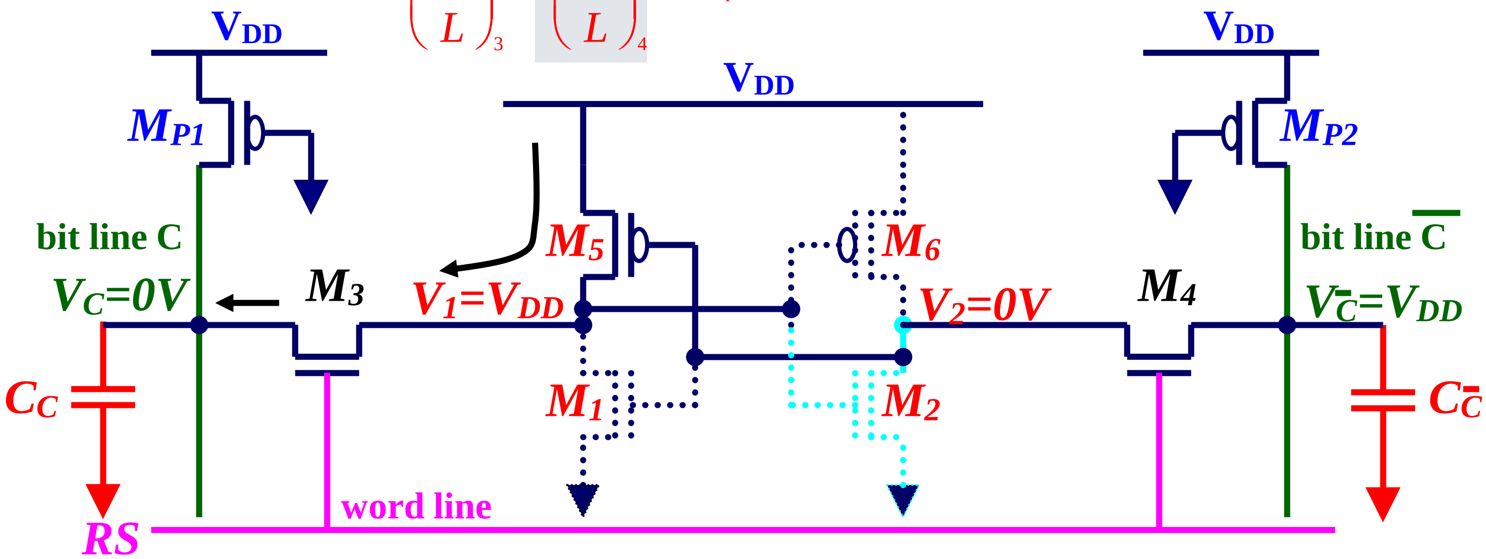
$$k_{p,5}(0-V_{DD}-V_{T,p})^2/2 = k_{n,3}(2(V_{DD}-V_{T,n})V_{T,n}-V_{T,n}^2)/2$$

  - » $V_1 < V_{T,n}$, i.e. M_2(M_1) **forced OFF**

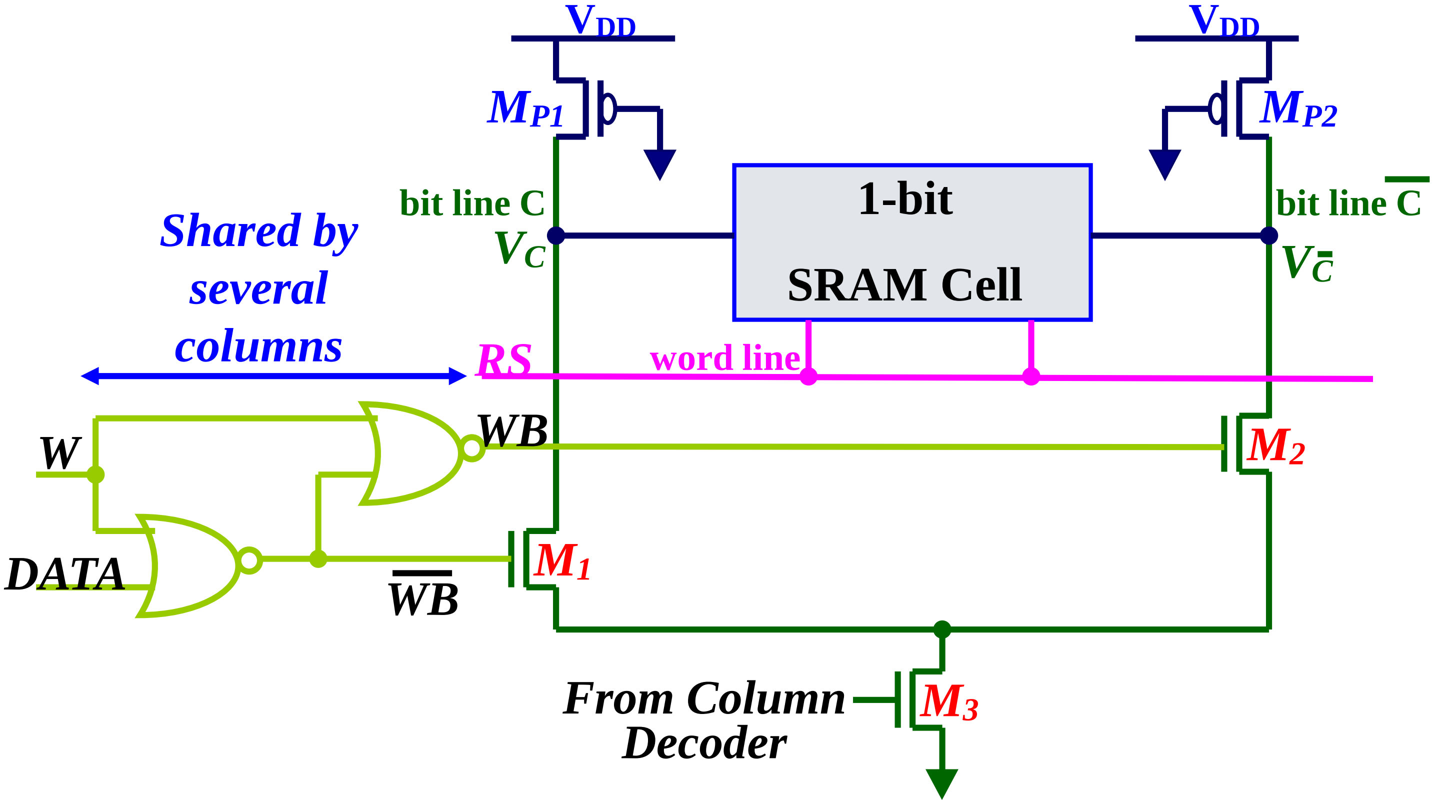
$$\frac{k_{p,5}}{k_{n,3}} = \frac{k_{p,6}}{k_{n,4}} < \frac{2(V_{DD}-1.5V_{T,n})V_{T,n}}{(V_{DD}+V_{T,p})^2} \Rightarrow$$

By symmetry

$$\frac{\left(\frac{W}{L}\right)_5}{\left(\frac{W}{L}\right)_3} = \frac{\left(\frac{W}{L}\right)_6}{\left(\frac{W}{L}\right)_4} < \frac{\mu_n}{\mu_p}\frac{2(V_{DD}-1.5V_{T,n})V_{T,n}}{(V_{DD}+V_{T,p})^2} \Rightarrow$$



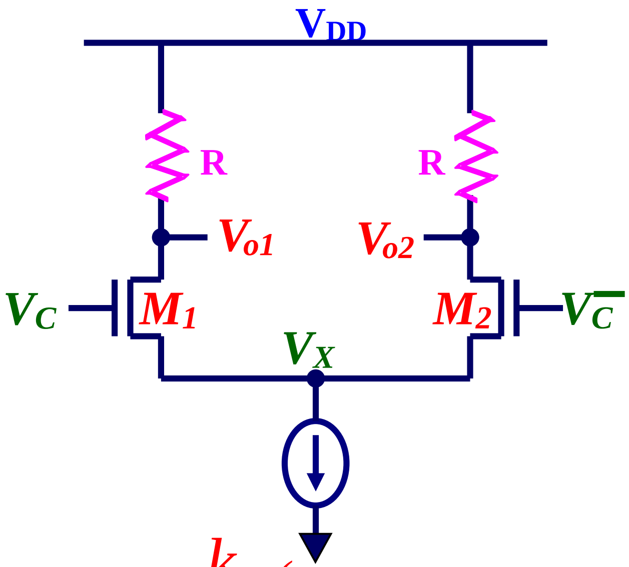
SRAM Write Circuit



W	DATA	$\overline{WB}$	WB	Operation (M3 on)
0	1	0	1	M_1 off, M_2 on $\Rightarrow V_{\overline{C}} \rightarrow$ low
0	0	1	0	M_1 on, M_2 off $\Rightarrow V_C \rightarrow$ low
1	X	0	0	M_1 off, M_2 off $\Rightarrow V_C$, $V_{\overline{C}}$ no change

SRAM Read Circuit

Source coupled differential amplifier



$$I_{D1} = \frac{k_n}{2}\left(V_C - V_X - V_{T1,n}\right)^2$$

$$I_{D2} = \frac{k_n}{2}\left(V_{\overline{C}} - V_X - V_{T2,n}\right)^2$$

$$A_{sense} = \frac{\partial\left(V_{o1} - V_{o2}\right)}{\partial\left(V_C - V_{\overline{C}}\right)} = -g_m R\nearrow$$

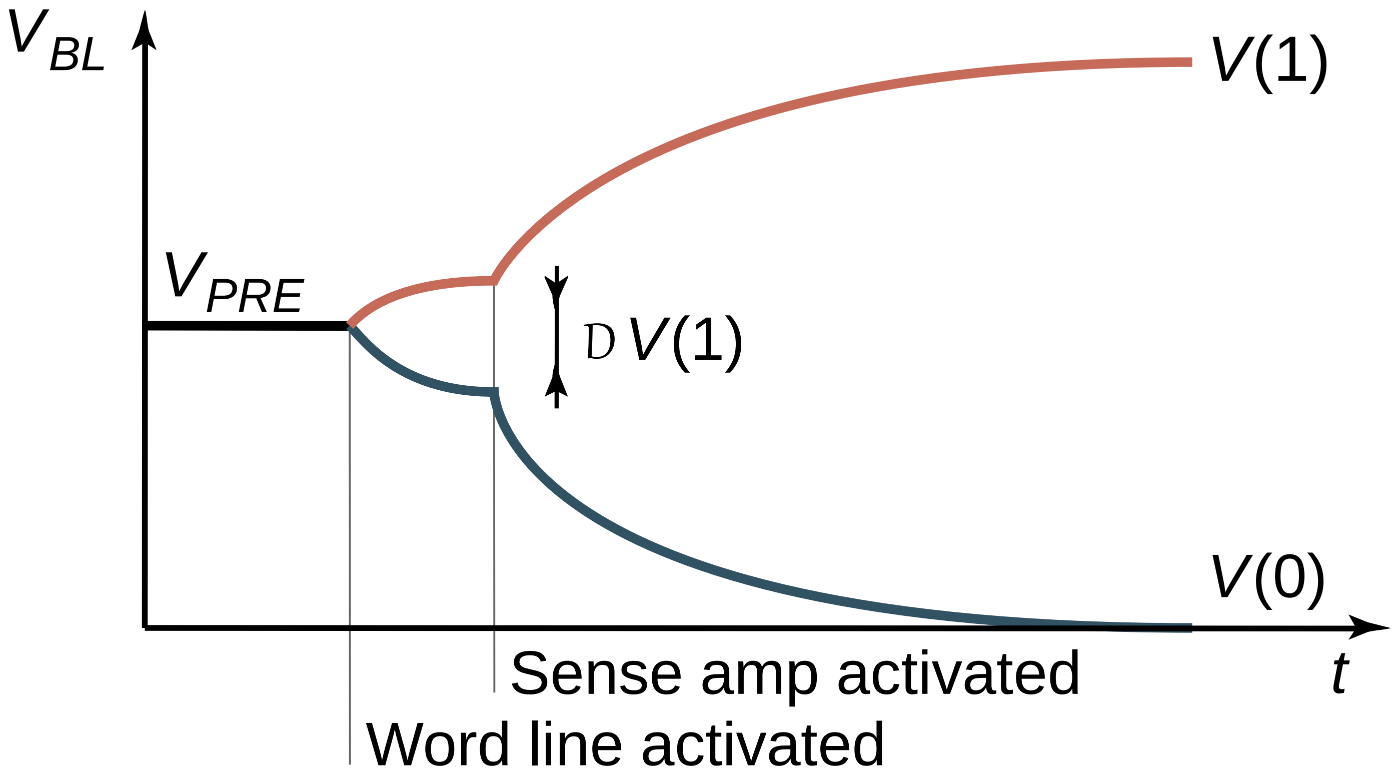
$$g_m = \frac{\partial I_D}{\partial V_{GS}} = \sqrt{2k_n I_D}$$

Increase R→

Use active load

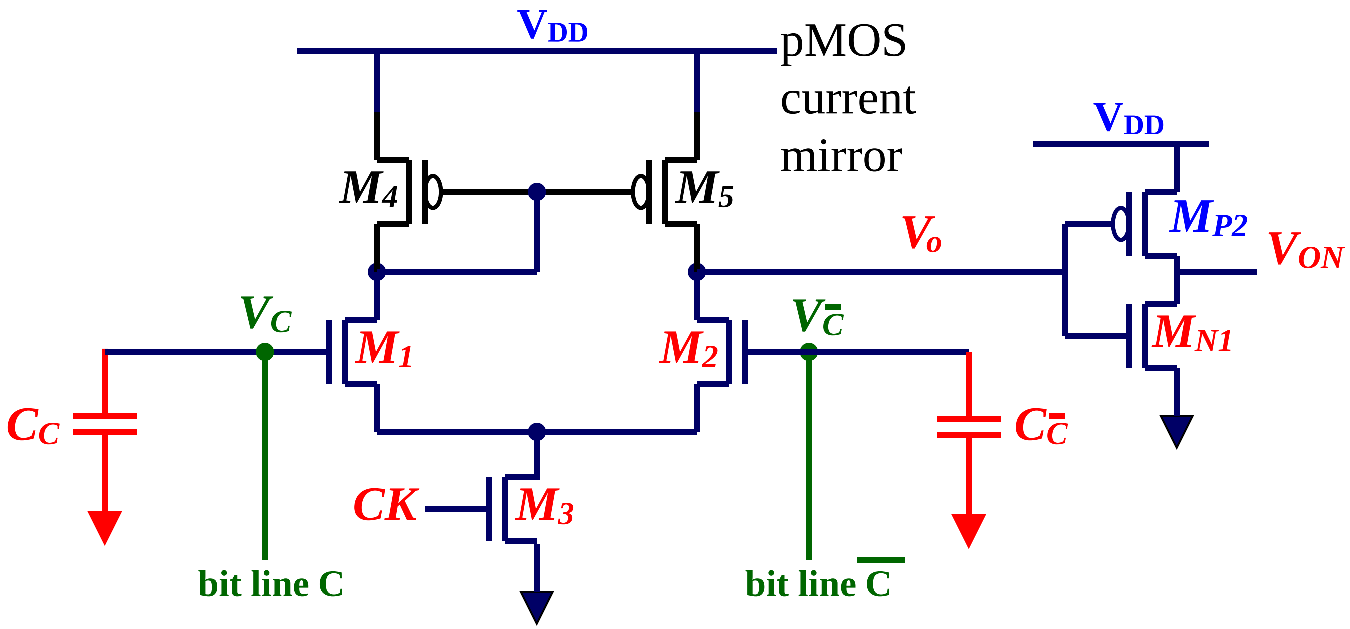
Use cascade

Sense Amp Operation



Source: Digital Integrated Circuits 2nd

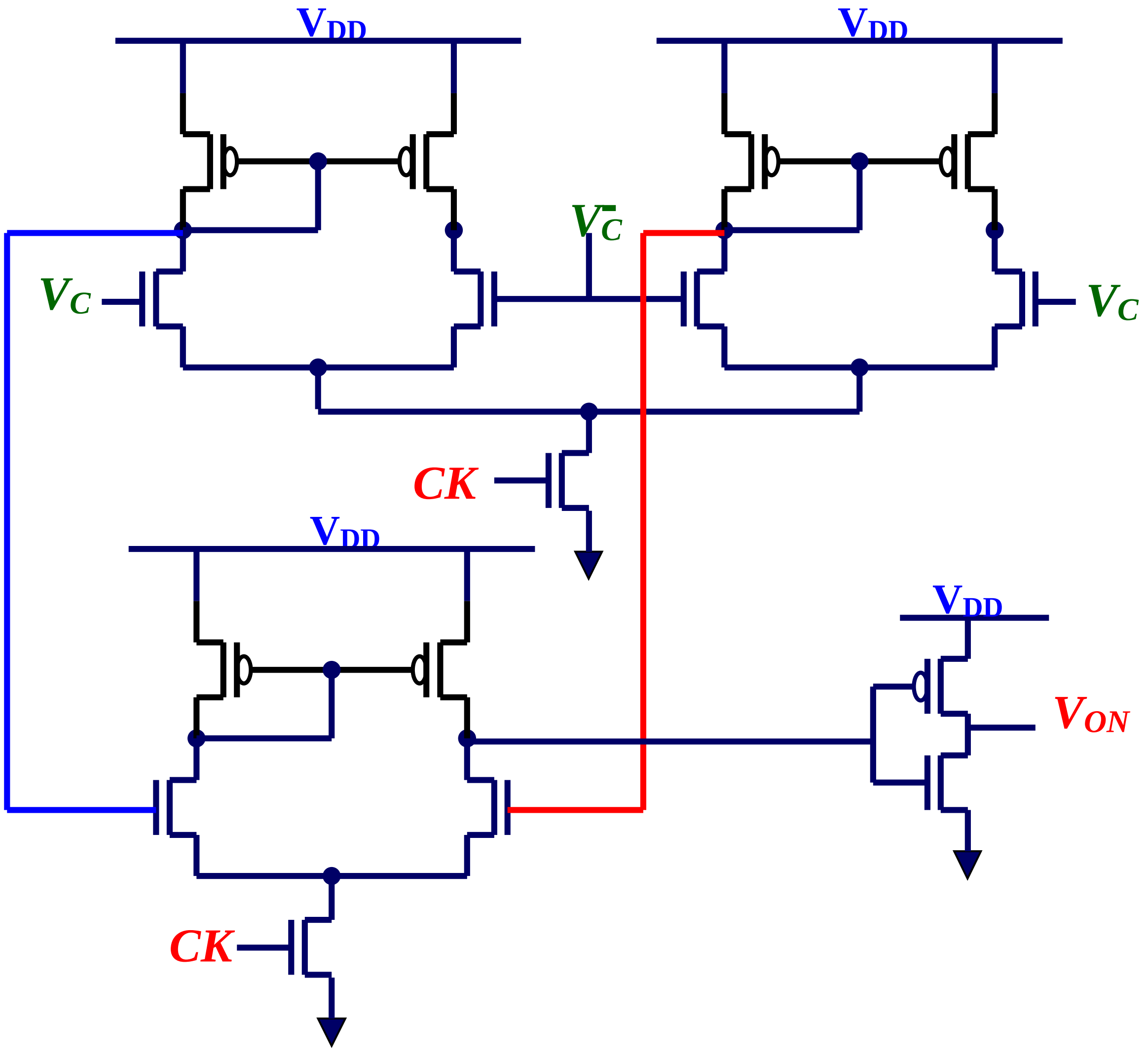
Fast Sense Amplifier



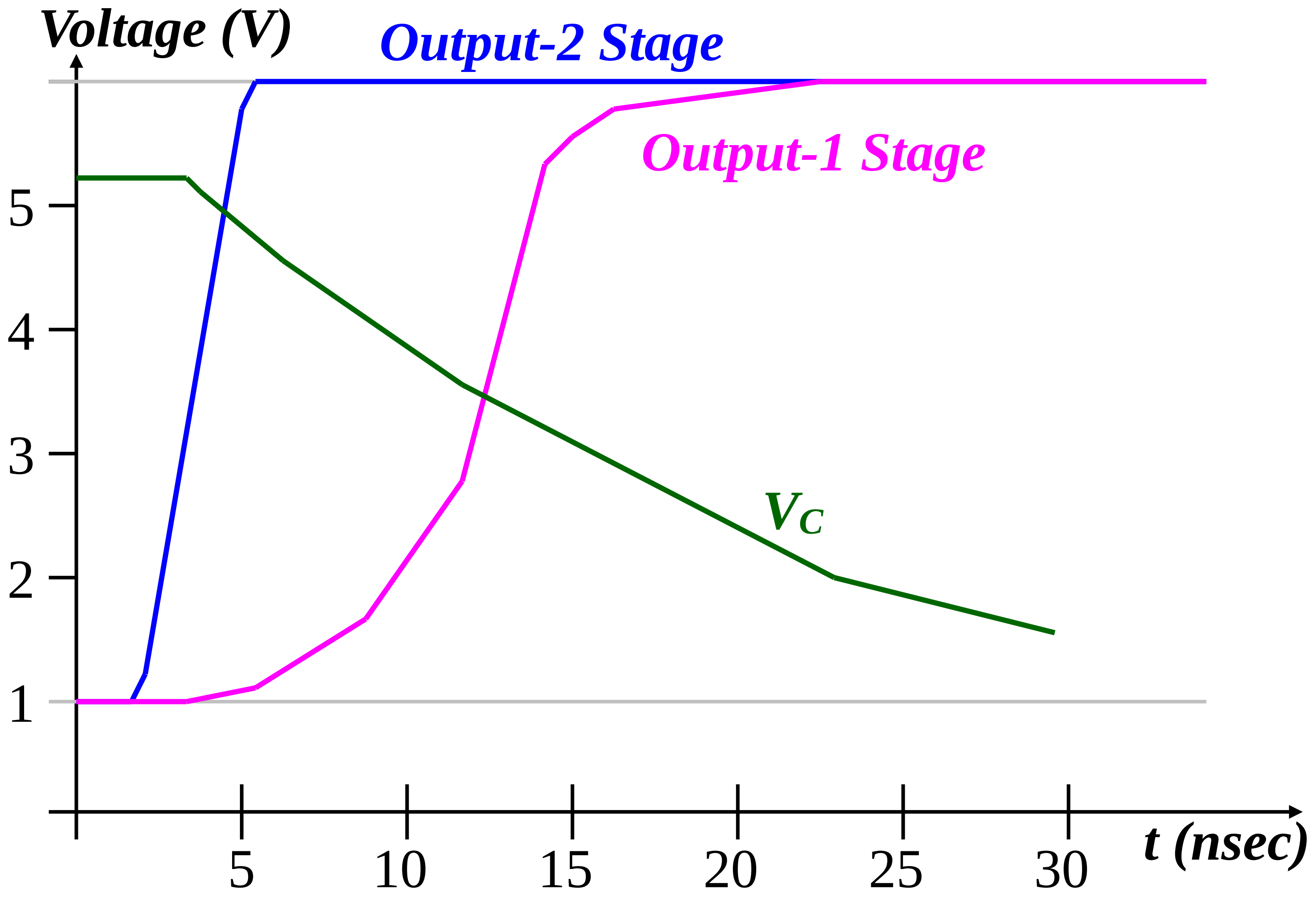
- $V_C < V_{\overline{C}}$: $M_1 \Rightarrow$OFF, V_o decreases, $V_{ON} \Rightarrow$High
- $V_C > V_{\overline{C}}$: $M_2 \Rightarrow$OFF, V_o remains high, V_{ON} =Low

$$A_{sense} = -g_{m2}(r_{o2} || r_{o5})$$

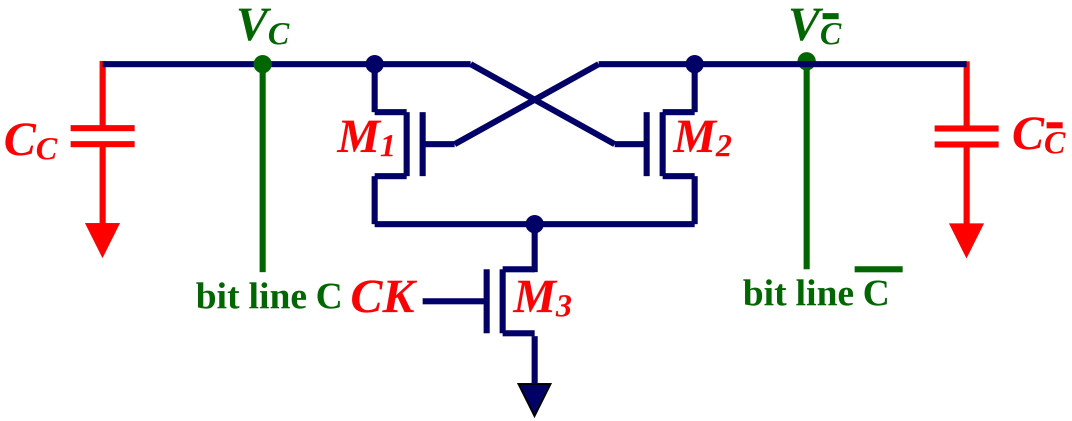
Two-Stage differential Current-Mirror Amplifier Sense Circuit



Typical Dynamic Response for One and Two Stage Sense Amplifier Circuits



Cross-Coupled nMOS Sense Amplifier



- **Assume:** M_3 **OFF**, V_C and $V_{\bar{C}}$ are initially precharged to V_{DD}
- **Access:** V_C drops slightly less than $V_{\bar{C}}$
- $M_3 \Rightarrow$ **ON** and $V_C < V_{\bar{C}}$ **:** M_1 **ON** first, pulling V_C lower

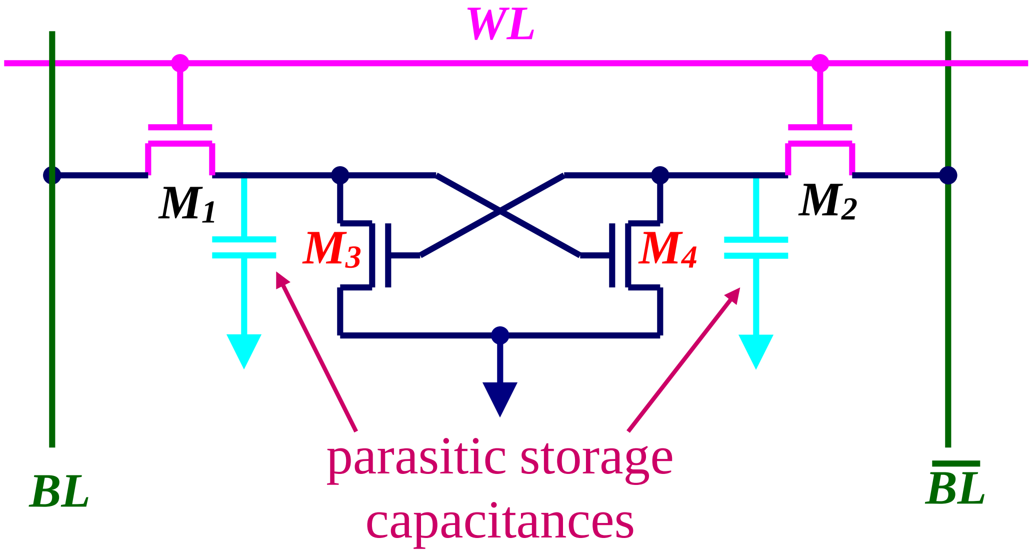
  M_2 turns **OFF**, C_C discharge via M_1 and M_3

  Enhances differential voltage V_C - $V_{\bar{C}}$

  Does not generate output logic level

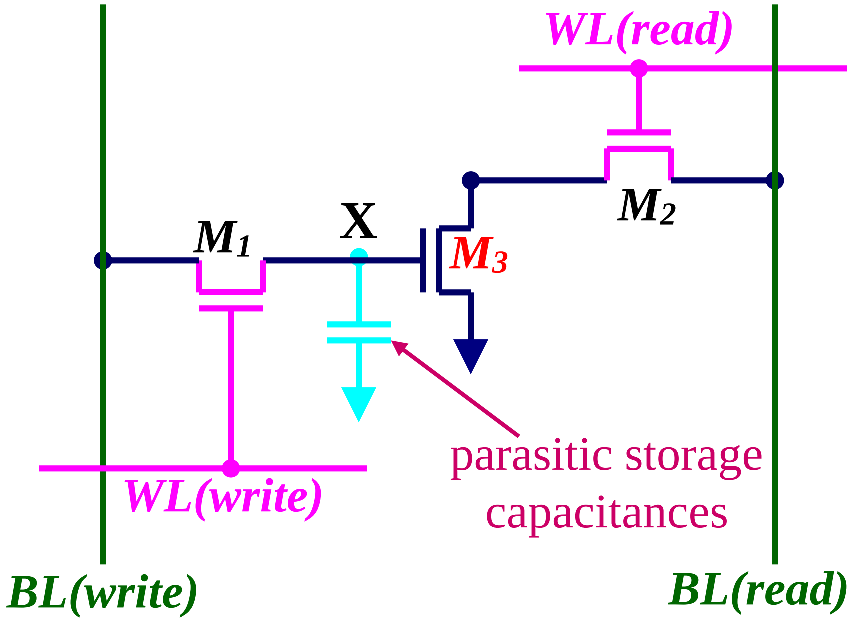
Dynamic Read-Write Memory (DRAM) Circuits

- **SRAM:** 4~6 transistors per bit
  4~5 lines connecting as charge on capacitor
- **DRAM:** Data bit is stored as charge on capacitor
  Reduced die area
  Require periodic refresh



Four-Transistor DRAM Cell

DRAM Circuits (Cont.)



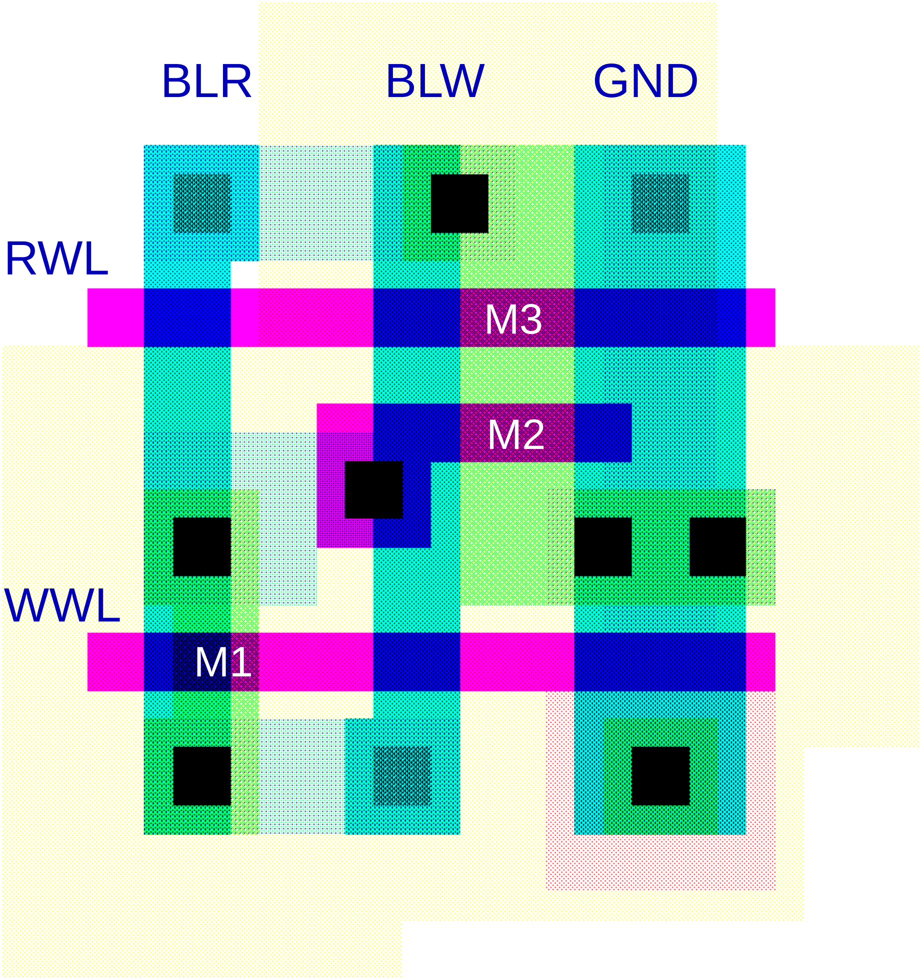
Three-Transistor DRAM Cell

No constraints on device ratios

Reads are non-destructive

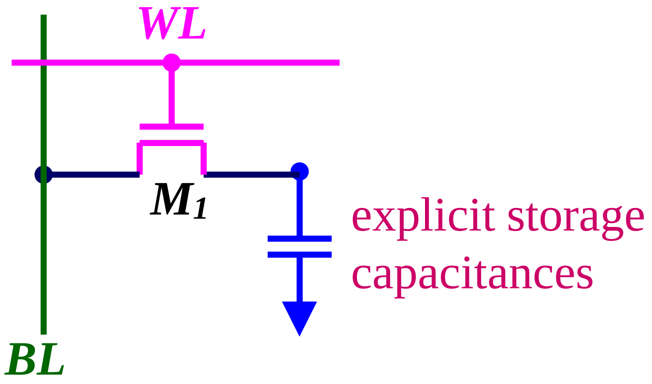
Value stored at node X when writing a “1” = V_{WWL}-V_{Tn}

3T-DRAM — Layout



Source: Digital Integrated Circuits 2nd

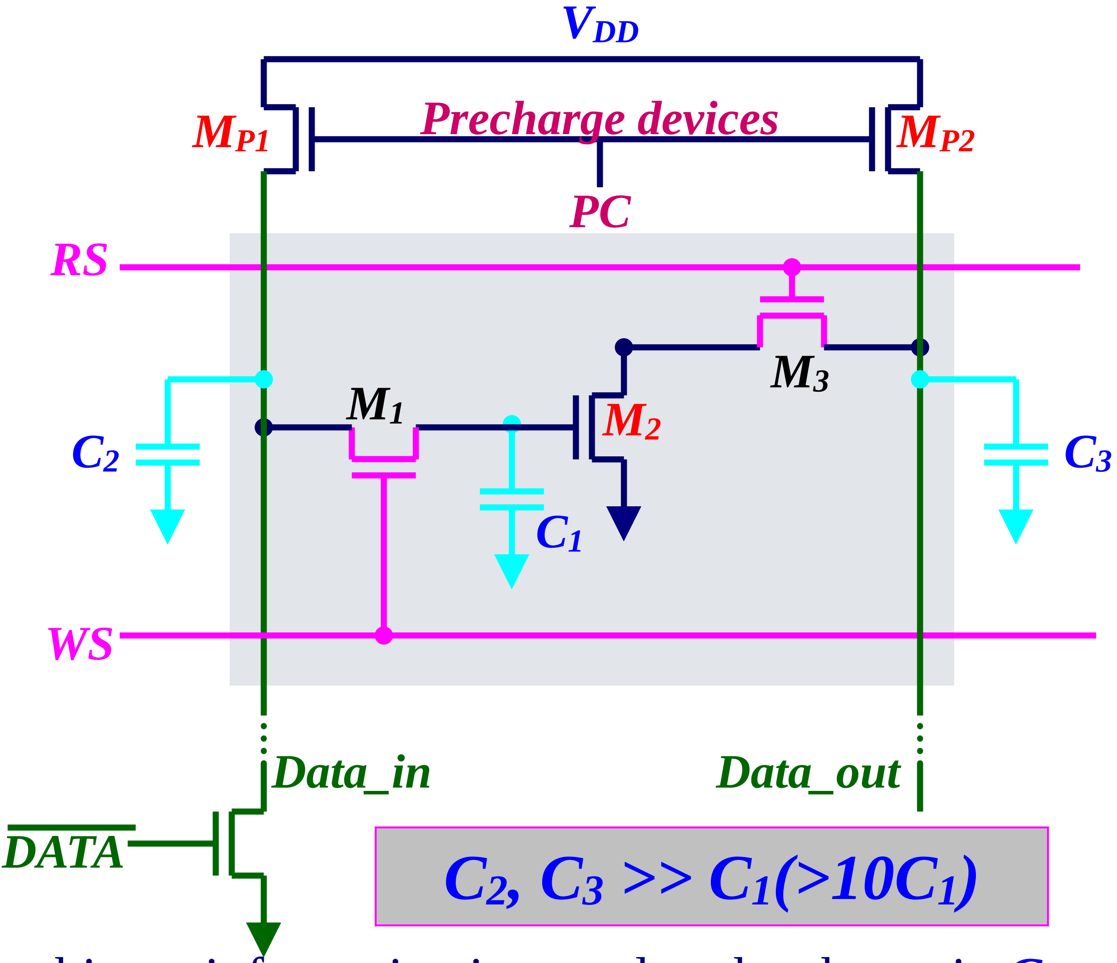
One-Transistor DRAM Cell



One-Transistor DRAM Cell

- **Industry standard** for high density dram arrays
- **Smallest** component count and silicon area per bit
- Separate or "**explicit**" capacitor (dual poly) per cell

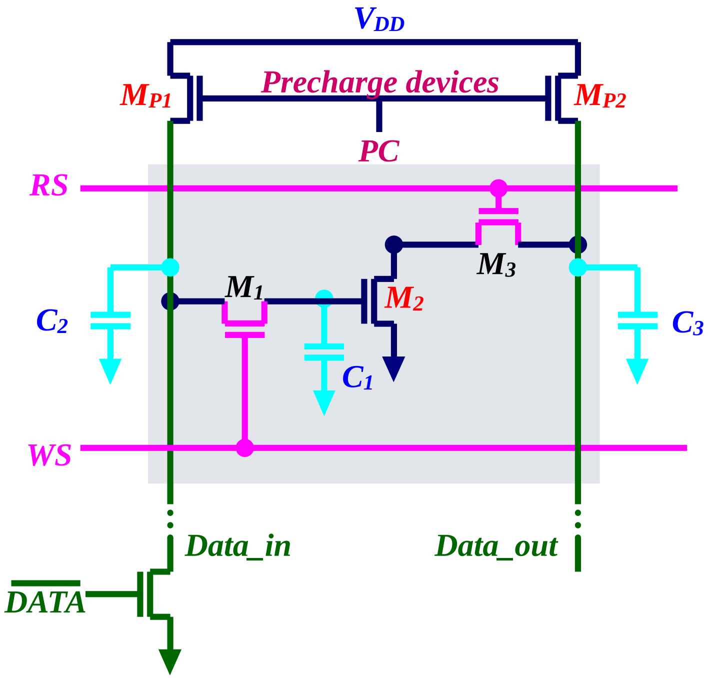
Operation of Three-Transistor DRAM Cell

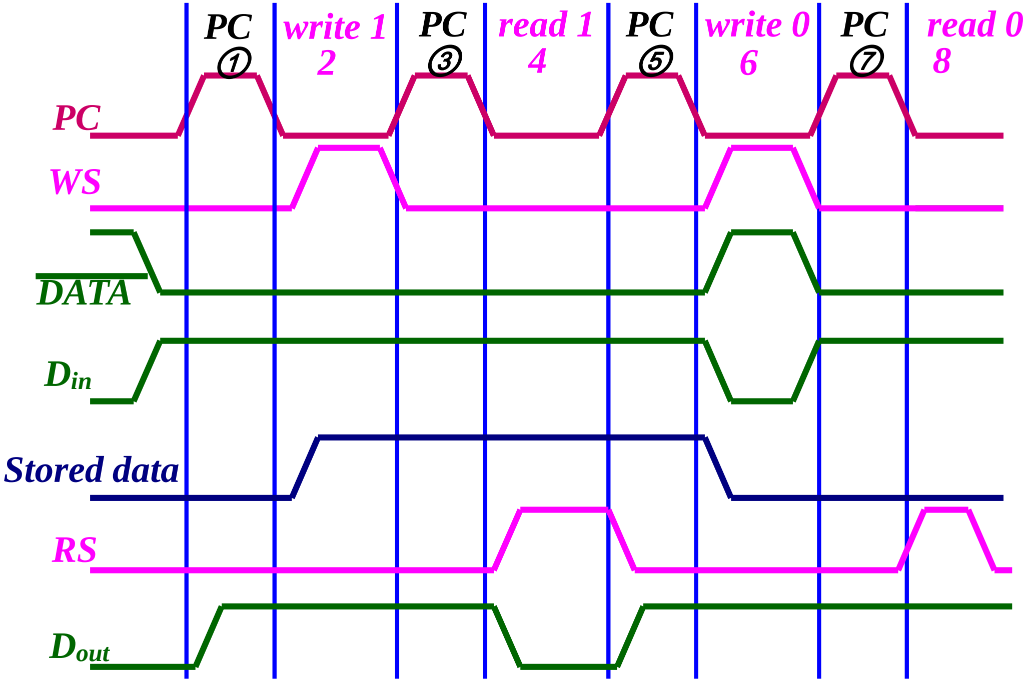


$C_2, C_3 >> C_1(>10C_1)$

- The binary information is stored as the charge in C_1
- ***Storage transistor*** M_2 is on or off depending on the charge in C_1
- **Pass transistors** M_1 **and** M_3**:** access switches
- Two separate bit lines for “data read” and “data write”

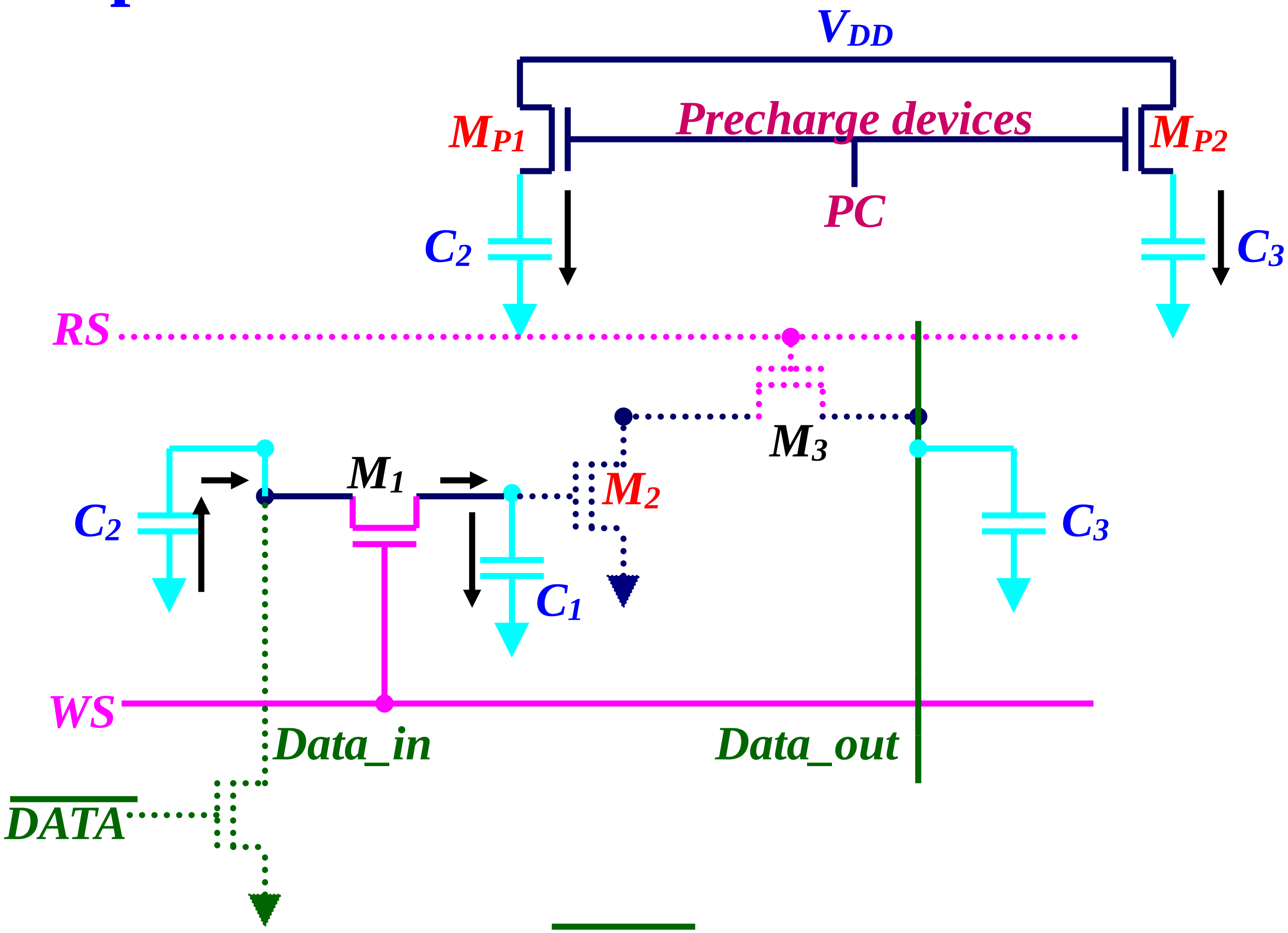
Operation of Three-Transistor DRAM Cell (Cont.)





- The operation is based on a **two-phase non-overlapping clock scheme**
  - The precharge events are driven by ϕ_1, and the “read” and “write” operations are driven by ϕ_2.
  - Every “read” and “write” operation is preceded by a precharge cycle, which is initiated with ***PC*** going **high**.

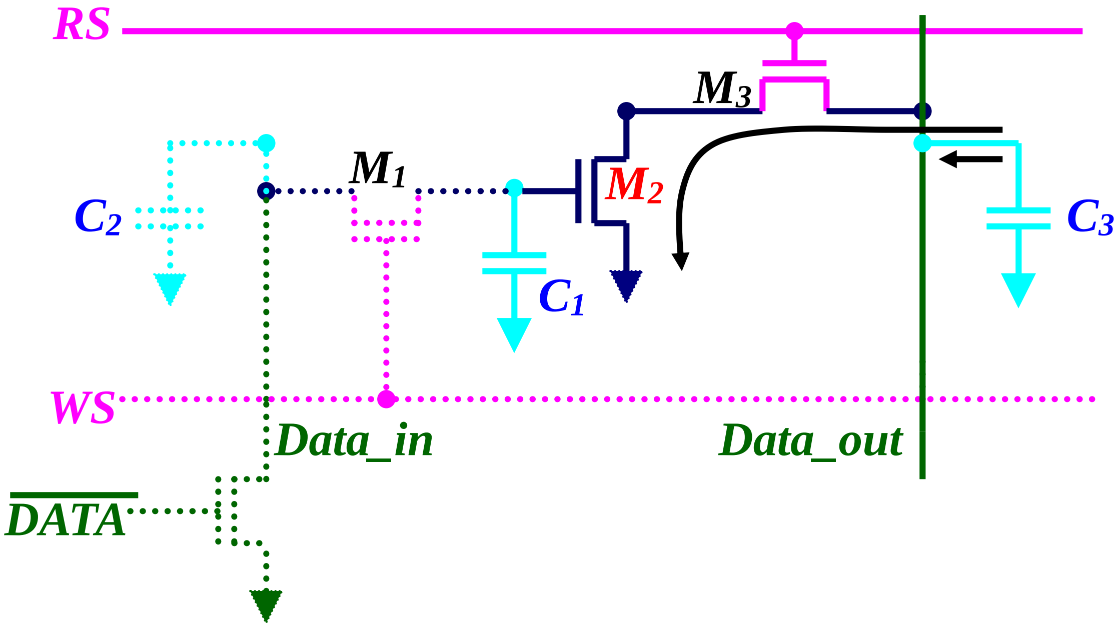
Operation of Three-Transistor DRAM Cell (Cont.)



Pre-charge Cycle

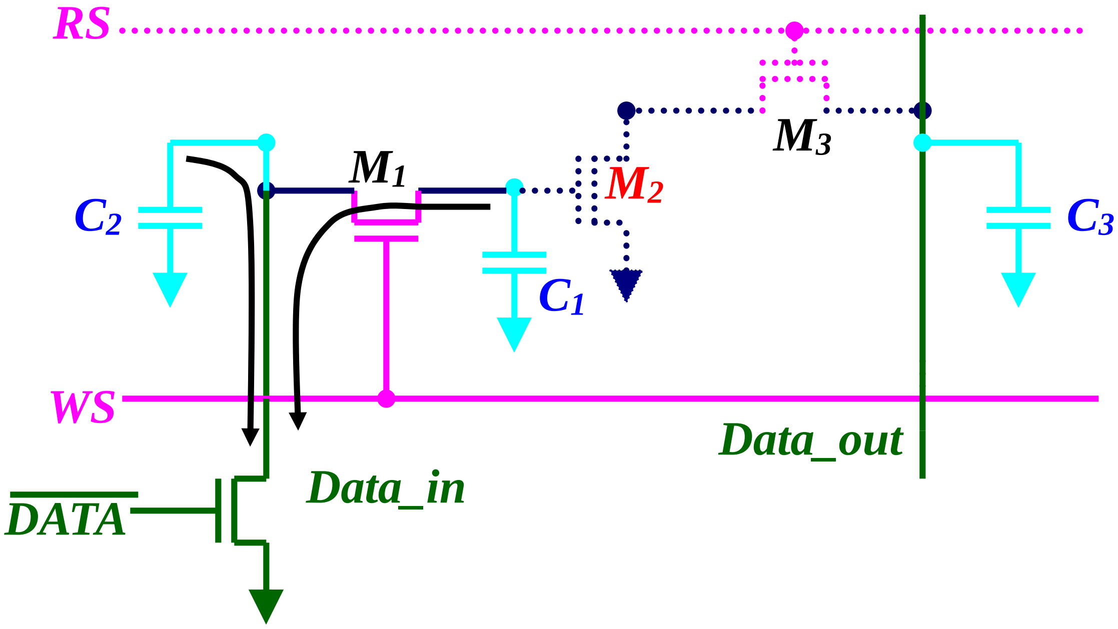
- **Write "1" OP**: $\overline{DATA} = 0$, $WS = 1$; $RS = 0$
  - » C_2, C_1 Share charge due to M_1 **ON**
  - » Since $C_2 >> C_1$, the storage node C_1 attains approximately the same logic level.

Operation of Three-Transistor DRAM Cell (Cont.)



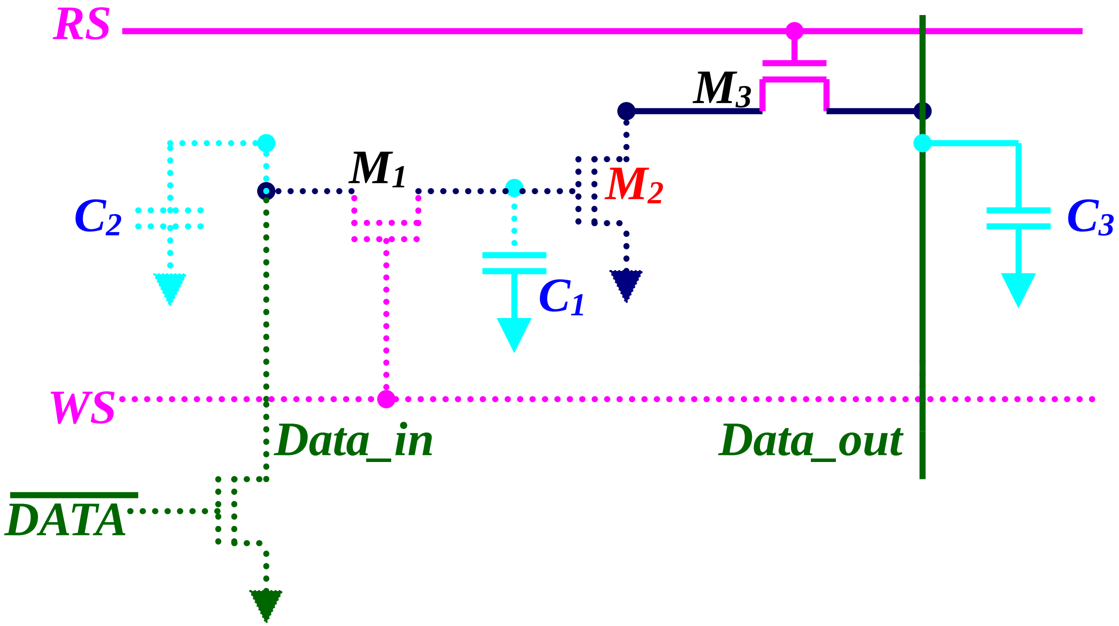
- **Read "1" OP**: ***$\overline{DATA}$ = 0***, ***WS = 0***; ***RS = 1***
  - » ***M_2, M_3 ON*** $\Rightarrow$ ***C_3, C_1*** discharges through ***M_2*** and ***M_3***, and the falling column voltage is interpreted bt the "data read" circuitry as a stored logic "1".

Operation of Three-Transistor DRAM Cell (Cont.)



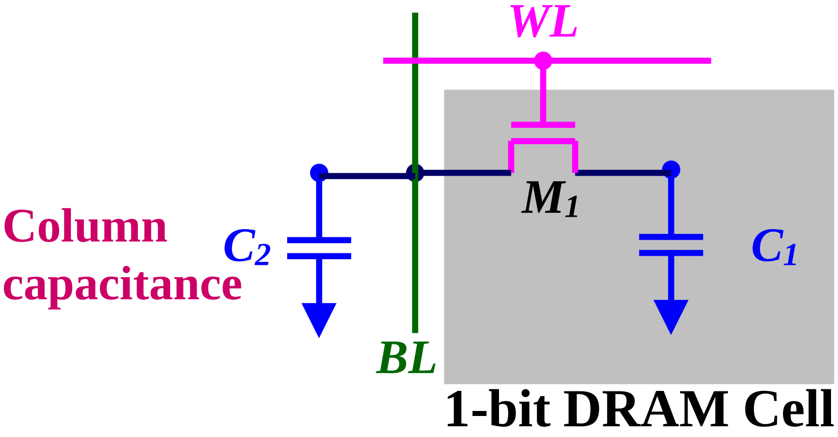
- **Write "0" OP**: $\overline{DATA}$ = 1, *WS* = 1; *RS* = 0
  - » M_2, M_3 *ON* ⇒ C_2 and C_1 discharge to 0 through M_1 and ***data_in nMOS***.

Operation of Three-Transistor DRAM Cell (Cont.)



- **Read “0” OP**: $\overline{\mathit{DATA}}$ = *1*, *WS = 0*; *RS = 1*
  - » C_3 does not discharge due to M_2 **OFF**, and the logic-high level on the ***Data_out*** column is interpreted by the data read circuitry as a stored “0” bit.

Operation of One-Transistor DRAM Cell



$C_2 >> C_1$

- **Write "1" OP:** $BL = 1$, $WL = 1$ (M_1 ON)⇒C_1 charges to "1"
- **Write "0" OP:** $BL = 0$, $WL = 1$ (M_1 ON)⇒C_1 discharges to "0"
- **Read OP:** destroys stored charge on C_1 ⇒ destructive refresh is needed after every data read operation

Appendix

- Derivation of $$\frac{k_{n,3}}{k_{n,1}} = \frac{\left(\frac{W}{L}\right)_3}{\left(\frac{W}{L}\right)_1} < \frac{2\left(V_{DD}-1.5V_{T,n}\right)V_{T,n}}{\left(V_{DD}-2V_{T,n}\right)^2}$$

$$k_{n,3}(V_{DD}-V_1-V_{T,n})^2/2 = k_{n,1}(2(V_{DD}-V_{T,n})V_1-V_1^2)/2$$

- Therefore,

$$\frac{k_{n,3}}{k_{n,1}} = \frac{\left(\frac{W}{L}\right)_3}{\left(\frac{W}{L}\right)_1} = -1 + \frac{\left(V_{DD}-V_{T,n}\right)^2}{\left(V_{DD}-V_1-V_{T,n}\right)^2} < -1 + \frac{\left(V_{DD}-V_{T,n}\right)^2}{\left(V_{DD}-2V_{T,n}\right)^2} = \frac{2\left(V_{DD}-1.5V_{T,n}\right)}{\left(V_{DD}-2V_{T,n}\right)^2}$$