Chapter 8

Semiconductor Memories

(based on Kang, Leblebici. *CMOS Digital Integrated Circuits*

8.1 General concepts

- Data storage capacity available on a single integrated circuit grows exponentially being doubled approximately every two years.
- **Capacity** of the dynamic read/write memory (DRAM) chip exceeds now 1 Gigabit.
- **Data transfer speed** of a standard DRAM is at the level of 200Mb/sec/pin.
- **Static and dynamic power consumption** is of the order of

Semiconductor Memories are classified according to the **type of data storage** and the **type of data access** mechanism into the following two main groups:

- Non-volatile Memory (NVM) also known as Read-Only Memory (ROM) which retains information when the power supply voltage is off. With respect to the data storage mechanism NVM are divided into the following groups:
  - Mask programmed ROM. The required contents of the memory is programmed during fabrication,
  - Programmable ROM (PROM). The required contents is written in a permanent way by burning out internal interconnections (fuses). It is a one-off procedure.
  - Erasable PROM (EPROM). Data is stored as a charge on an isolated gate capacitor (“floating gate”). Data is removed by exposing the PROM to the ultraviolet light.
  - Electrically Erasable PROM (EEPROM) also known as Flash Memory. It is also base on the concept of the floating gate. The contents can be re-programmed by applying a suitable voltages to the EEPROM pins. The Flash Memories are very important data storage devices for mobile applications.
- Read/Write (R/W) memory, also known as Random Access Memory (RAM). From the point of view of the data storage mechanism RAM are divided into two main groups:
  - Static RAM, where data is retained as long as there is power supply on.
  - Dynamic RAM, where data is stored on capacitors and requires a periodic refreshment.

Typical organization of a single chip semiconductor memory is shown in Figure 8.1.

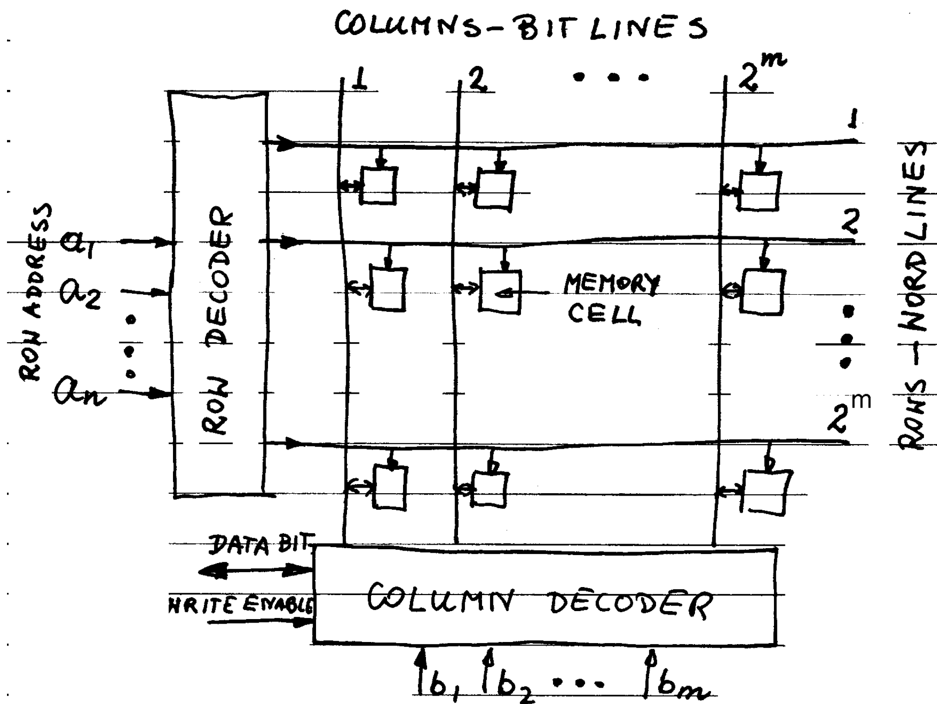


Figure 8.1: Typical memory organization

The memory consists of the following basic blocks:

- The array of 1-bit memory cells,
- The row decoder which selects a single **word line** for a given n-bit row address a[1:n],
- The column decoder which selects a single **bit line** for a given m-bit column address b[1:m], and routs a 1-bit data to or from a selected memory cell.

8.2 Mask programmed (ROM) memory circuits.

In this section we consider memory cells of Read-Only Memories programmed by application of specific masks during the fabrication process. Two basic types of the ROM cells are based on NOR and NAND gates.

8.2.1 NOR-based ROM

The building block of this ROM is a pseudo-nMOS NOR gate as in Figure 8.2.

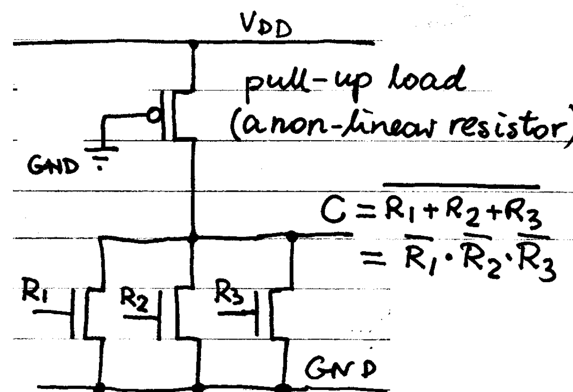


Figure 8.2: A 3-input pseudo-nMOS NOR gate.

Unlike in a standard CMOS gate, the pMOS pull-up circuitry is replaced by a single pMOS with its gate tied up to GND, hence being permanently on acting as a load resistor.
If none of the nMOS transistors is activated (all R_i being low) then the output signal C is high.
If any of the nMOS transistors is activated (R_i being high) then the output signal C is low.
To reduce the power consumption the gate of the pMOS pull-up transistor is connected to a clock signal. The power is consumed only during low period of the clock.

A **NOR-based ROM** consists of m n-input pseudo-nMOS NOR gates, one n-input NOR per column as shown in Figure 8.3.

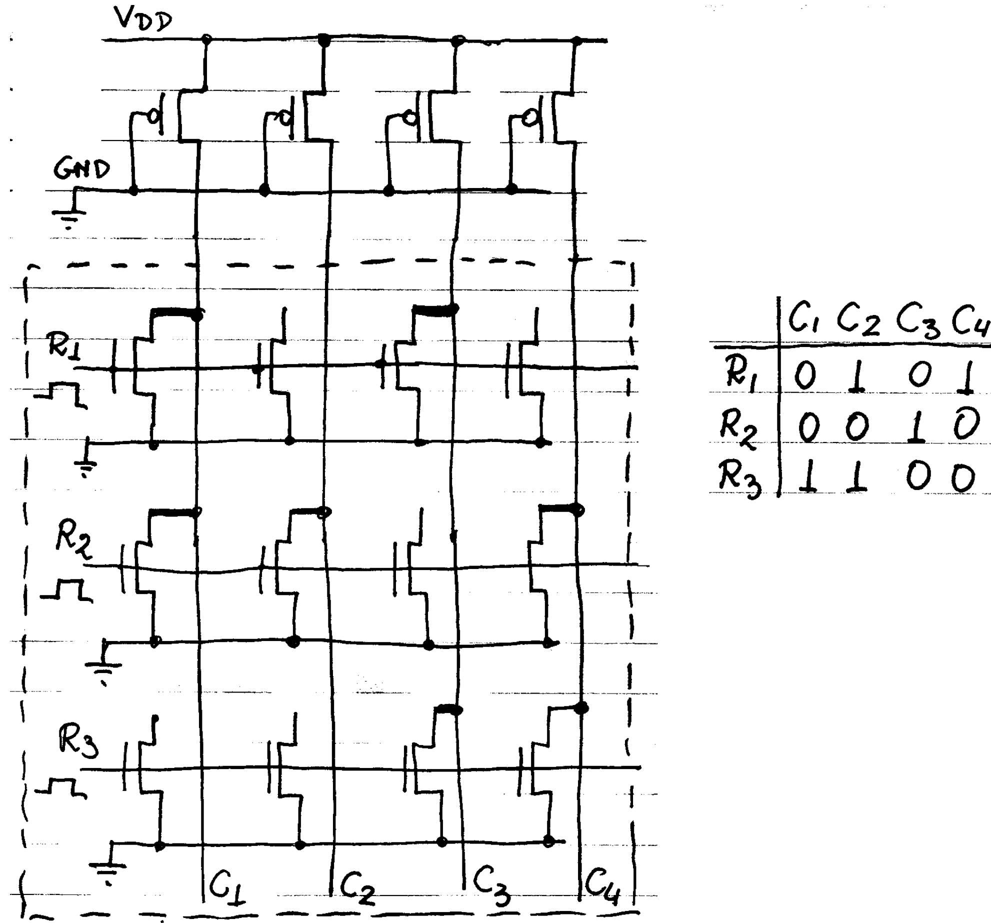


Figure 8.3: A 3-by-4 NOR-based ROM array

Each memory cell is represented by one nMOS transistor and a binary **information is stored** by **connecting or not** the drain terminal of such a transistor to the **bit line**.
For every row address only one **word line** is activated by applying a high signal to the gates of nMOS transistors in a row.
If a selected transistor in the i-th column is connected to a bit line then the **logic '0'** is stored in this memory cell. if the transistor is not connected, then the **logic '1'** is stored.

Example of the layout (stick diagram) of a 4-by-4 **NOR ROM array** is shown in Figure 8.4.

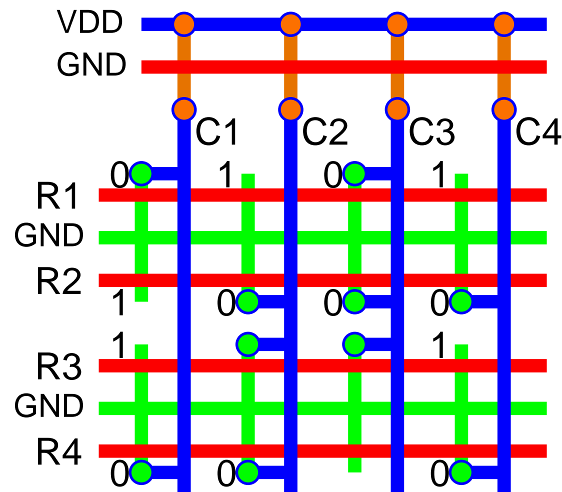


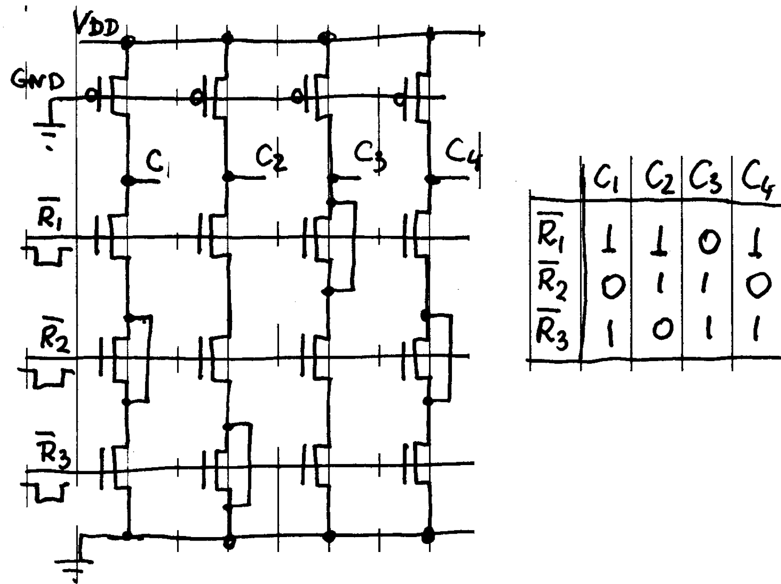
Figure 8.4: A stick diagram of a 4-by-4 NOR ROM array

In the layout the bit lines (columns) are implemented in metal 1 and the word lines (rows) connecting the gates of the nMOS "memory" transistors are implemented in polysilicon. The sources of the nMOS transistors are connected to GND through the n diffusion. To save silicon area two adjacent rows share the GND diffusion connection.

The programming is performed by adding ('0') or not ('1') a contact between the drain of a nMOS transistor and the bit line. Note that in the real layout the bit lines are laid out directly on the top of the nMOS transistors.

8.2.2 NAND-based ROM

A **NAND-based ROM** consists of m n-input pseudo-nMOS NAND gates, one n-input NAND per column as shown in Figure 8.5. In this case, we have up to n serially connected nMOS transistors in each column



	C_1	C_2	C_3	C_4
$\overline{R_1}$	1	1	0	1
$\overline{R_2}$	0	1	1	0
$\overline{R_3}$	1	0	1	1

Figure 8.5: A 3-by-4 NAND-based ROM array

For every row address only one **word line** is activated by applying a low signal to the gates of nMOS transistors in a row. When no word line is activated, all nMOS transistors are on and the line signals, C_i are all low.

When a word line is activated all transistors in the row are switched off and the respective C_i signals are high. If a transistor in the selected row is short-circuited, then the respective C_i signal is low.

In other words, the **logic '0'** is stored when a transistor is replaced with a wire, whereas the **logic '1'** is stored by an nMOS transistor being present.

Example of the layout (stick diagram) of a 3-by-4 **NAND ROM array** is shown in Figure 8.6.

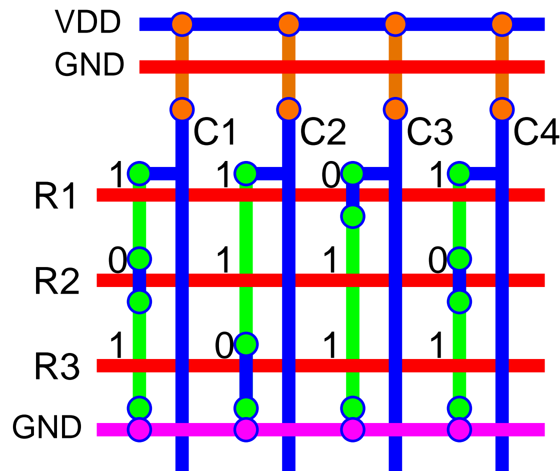


Figure 8.6: A stick diagram of a 3-by-4 NAND ROM array

In the layout, similarly to the NOR ROM, the bit lines (columns) are implemented in metal 1 and the word lines (rows) connecting the gates of the nMOS “memory” transistors are implemented in polysilicon.

Programming of logic ‘0’s is performed by replacing respective nMOS transistors with direct metal 1 connections.

In general, the layout of the NAND ROM can be smaller to that of corresponding NOR ROM, specifically if the short-circuiting is implemented by additional implant modifying the threshold od the ‘0’ transistors so that they are always ‘on’. This eliminates the need for additional contacts to perform ‘0’–programming.

The drawback of the NAND ROM however is that they are usually slower comparing with the corresponding NOR ROM, because of a significant number of serially connected nMOS transistors between the bit line and the ground.

8.2.3 Row and Column Decoders

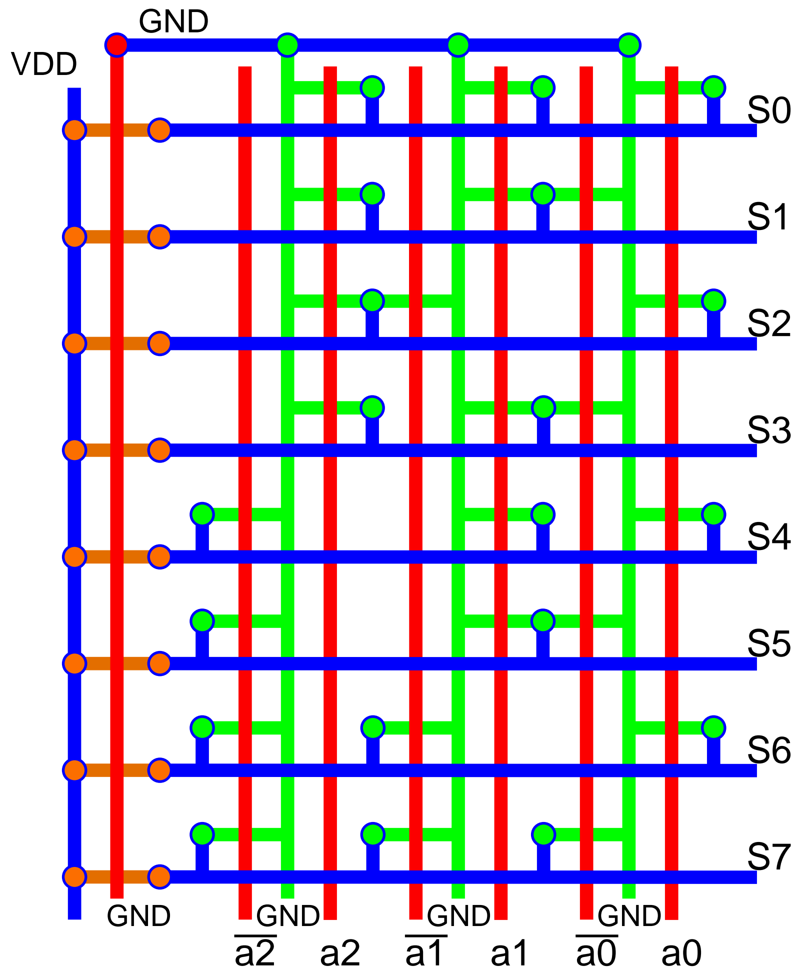


Figure 8.7: A stick diagram of a 3-to-2^3 decoder

8.3 Static Read/Write Memory (SRAM)

8.3.1 Structure of the CMOS memory cell

Static Read/Write (or Random Access) memory (SRAM) is able to read and write data into its memory cells and retain the memory contents as long as the power supply voltage is provided.
Currently SRAM are manufactured in the CMOS technology which offers very low static power dissipation, superior noise margin and switching speed.

The cells of the CMOS SRAM are based on a simple latch circuit as shown in Figure 8.8.

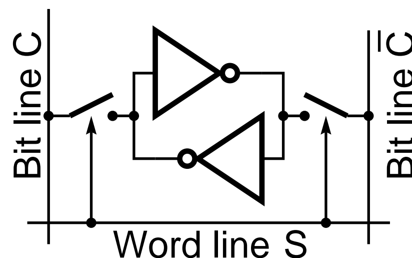


Figure 8.8: A logic diagram of a CMOS static memory cell

The two-inverter latch is able to store one bit data. In order to access the cell the word line is activated with high-level signal S, which closes access switches on both sides of the cell.

The state of the cell (and its complement) is now available on two complemented bit lines and the **read operation** can be performed.

In order to perform write operation the data and its complement is supplied through the bit line. We consider some details of the cell operation later.

The schematic of a CMOS SRAM cell is shown in Figure 8.9.

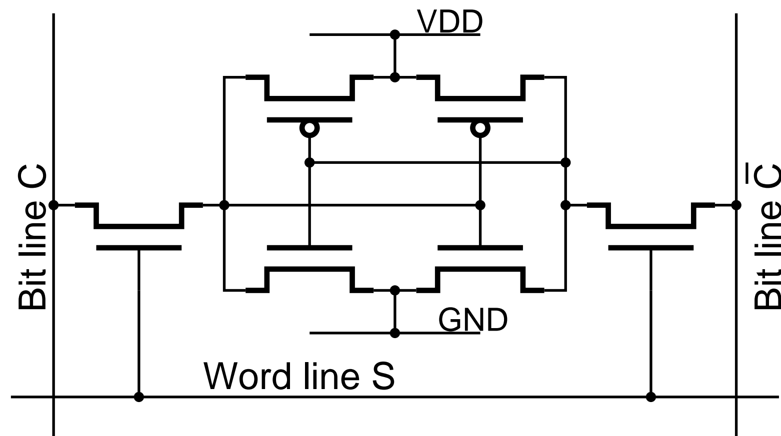


Figure 8.9: A schematic of a CMOS static memory cell

The cell consists of six transistors: four nMOS a two pMOS. Two pairs of transistors form a pair of inverters and two nMOS transistors form the access switches.
The stick diagram of a possible layout of a CMOS SRAM cell is shown in Figure 8.10.

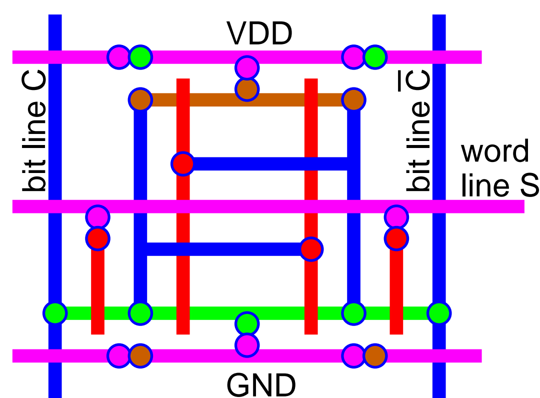


Figure 8.10: The stick diagram of a CMOS static memory cell

8.3.2 Principles of operations

In order to consider operation of the static read/write memory we have to take into account:

- relatively large parasitic column capacitances, C_C and $C_{\bar{C}}$,
- column pull-up pMOS transistors,

as shown in Figure 8.11.

Figure 8.11: A CMOS static memory cell with column pull-up transistors and parasitic column capacitances.

When **none of the word lines is selected**, that is, all S signals are '0', the pass transistors n3, n4 are turned off and the data is retained in all memory cells. The column capacitances are charged by the drain currents of the pull-up pMOS transistors, p3, p4.

The column voltages V_C and $V_{\bar{C}}$ both reach the level just below $V_{DD} - V_{Tp}$, say 3.5V for $V_{DD} = 5$V and the threshold voltage $V_{Tp} = 1$V.

For the **read or write operations** we select the cell asserting the word line signal S='1'.

For the **write operation** we apply a low voltage to one of the bit line, holding the other one high.

To **write '0'** in the cell, the column voltage V_C is forced to low ($C = 0$). This low voltage acts through a related pass transistor (n3) on the gates of the corresponding inverter (n2, p2) so that its input goes high. This sets the signal at the other inverter $Q = 0$.

Similarly, to **write '1'** in the cell, the opposite column voltage $V_{\bar{C}}$ is forced to low ($\bar{C} = 0$) which sets the signal $Q = 1$.

During the **read '1' operation**, when the stored bit is $Q = 1$, transistors n3, p1 and n4, n2 are turned on. This maintains the column voltage V_C at its steady-state high level (say 3.5V) while the opposite column voltage $V_{\bar{C}}$ is being pulled down discharging the column capacitance $C_{\bar{C}}$ through transistors n4, n2 so that $V_C > V_{\bar{C}}$.

Similarly, during the **read '0' operation** we have $V_C < V_{\bar{C}}$.

The difference between the column voltages is small, say 0.5V, and must be detected by the **sense amplifiers** from data-read circuitry.

8.3.3 SRAM Write Circuitry

The structure of the write circuitry associated with one column of the memory cells is shown in Figure 8.12.

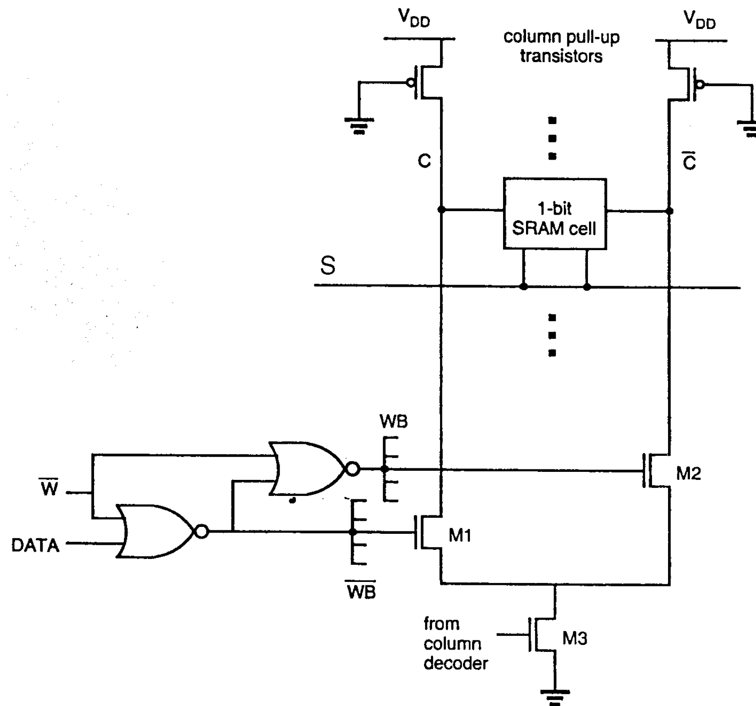


Figure 8.12: The structure of the write circuitry associated with one column of the memory cells.

The principle of the write operation is to assert voltage on one of the columns to a low level. This is achieved by connecting either C or $\bar{C}$ to the ground through the transistor M3 and either M1 or M2.

The transistor M3 is driven by the signal from the column decoder selecting the specified column.

The transistor M1 is on only in the presence of the write enable signal ($\bar{W} = 0$) when the data bit to be written is '0'.

The transistor M2 is on only in the presence of the write signal ($\bar{W} = 0$) when the data bit to be written is '1'.

8.3.4 SRAM Read Circuitry

The structure of the read circuitry is shown in Figure 8.13.

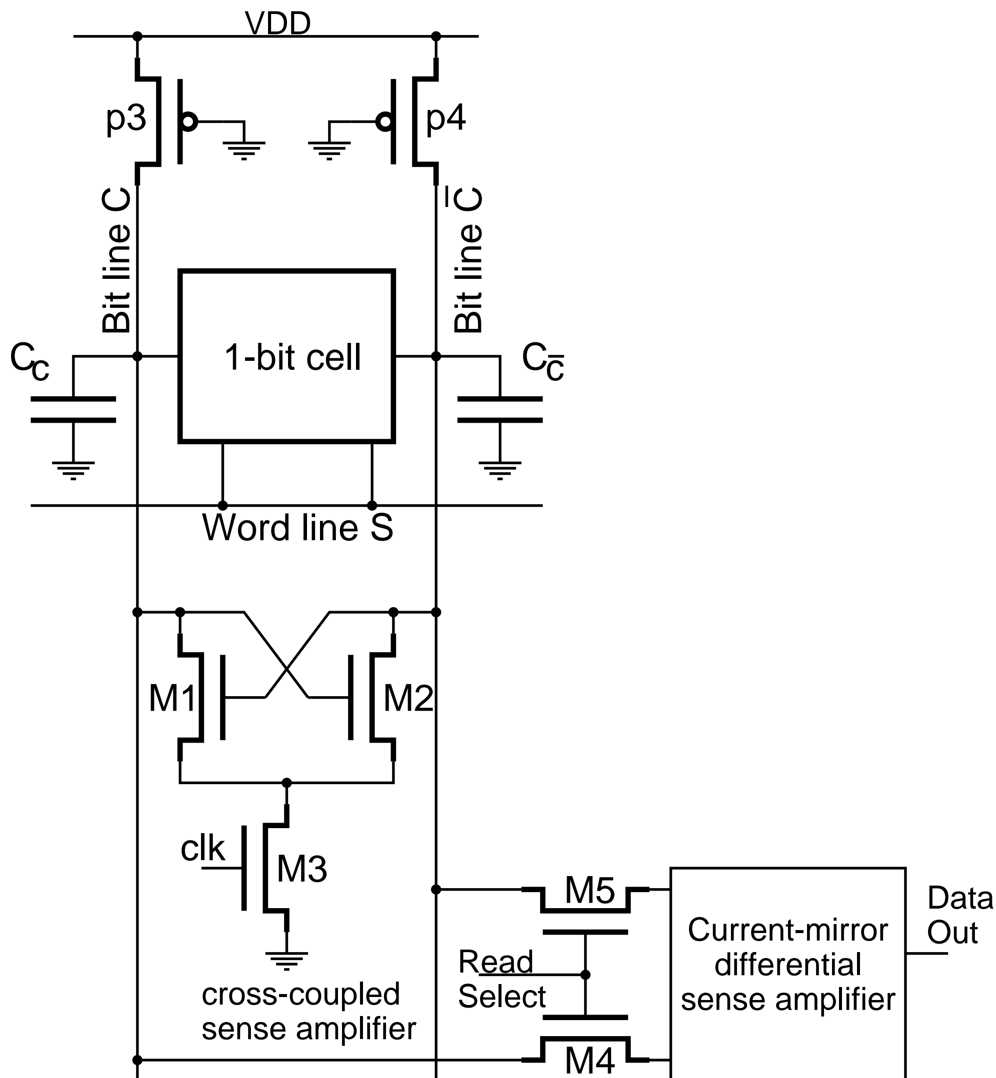


Figure 8.13: The structure of the write circuitry associated with one column of the memory cells.

During the read operation the voltage level on one of the bit lines drops slightly after the pass transistors in the memory cell are turned on.

The read circuitry must properly sense this small voltage difference and form a proper output bit:

$$\begin{array}{lll} `0' & \text{if} & V_C < V_{\bar{C}} \\ `1' & \text{if} & V_C > V_{\bar{C}} \end{array}$$

The read circuitry consists of two level sense amplifiers:

- one simple cross-coupled sense amplifier per column of memory cells,
- one current-mirror differential sense amplifier per the memory chip.

The **cross-coupled sense amplifier** works as a latch. Assume that the voltage on the bit line C start to drop slightly when the memory access pass transistors are activated by the word line signal S, and that the clk signal is high so that the transistor M3 is turned on. Now, higher voltage on the gate of M1 transistor than on the gate of M2 starts the latching operation which pulls the V_C voltage further down switching the transistor M2 off. As a result the parasitic capacitance, C_C is discharged through M1 and M3. In this way a small difference between column voltages is amplified.

The amplified (discriminated) column voltages are passed through transistors M4 and M5 to the main sense amplifier.

The schematic of a typical differential current-mirror sense amplifier is shown in Figure 8.14.

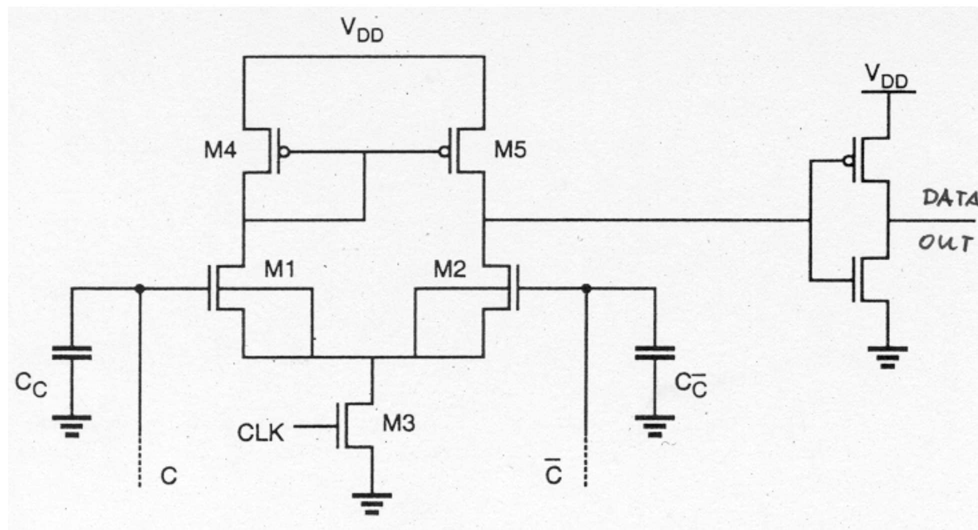


Figure 8.14: A CMOS differential current-mirror sense amplifier.

In this circuit, the gates of the two nMOS transistors M1 and M2 are connected to the bit lines. Their substrate terminals are tied to their respective source terminals in order to remove the substrate-bias effect. Notice that each bit line is represented by a large parasitic capacitance. The nMOS transistor M3 is a long-channel device which acts as a current source for both branches, and is controlled by a clock signal. The output inverter is not a part of the differential amplifier, but it is used to drive the output node.

Before the beginning of a "read" operation, the two bit lines (columns) are pulled up for equalization, as discussed earlier. The CLK signal is low during this phase, so that the nMOS transistor M3 remains off. Since both M1 and M2 conduct, the common source node is pulled up, and the output node of the amplifier also goes high. Therefore, the output of the inverter is at a logic-low level initially.

Once a memory cell is selected for the "read" operation, the voltage on one of the complementary bit lines will start to drop slightly. At the same time as the row selection signal, the CLK signal driving M3 is also turned on. If the stored data on the selected SRAM cell forces the bit line C to decrease slightly, transistor M1 turns off, and the output voltage of the differential amplifier drops immediately. Consequently, the output voltage of the inverter goes high. Otherwise, if the stored data on the selected memory cell forces the bit line $\overline{C}$ to drop slightly, M2 turns off. Thus, the voltage level at the output node of the differential amplifier remains high in this case, and the inverter also preserves its logic-low output level.

8.3.5 Dual-Port SRAM (Kang, Leblebici)

In some cases, the memory array may have to be accessed simultaneously by multiple processors, or by one processor and another peripheral device. This could result in a timing conflict called "contention," which can be resolved only by having one of the processors wait until the SRAM is free. The added wait state, however, significantly reduces the advantages of the high-speed processor. The dual-port RAM architecture is implemented in systems in which a main memory array must serve multiple high-speed processors and peripheral devices with minimum delay.

The ideal dual-port SRAM allows simultaneous access to the same location in the memory array, by using two independent sets of bit lines and associated access switches for each memory cell. The circuit structure of a typical CMOS dual-port SRAM cell is shown in Fig. 10.35. Here, "word line 1" is used to access one set of complementary bit lines (bit line 1), while "word line 2" allows access to the other set of bit lines (bit line 2). The capability of simultaneous access eliminates wait states for the processors during "data read" operations. However, contention may still occur if both external processors accessing the same memory location simultaneously attempt to write data onto the accessed cell, or if one of the processors attempts to read data while the other processor

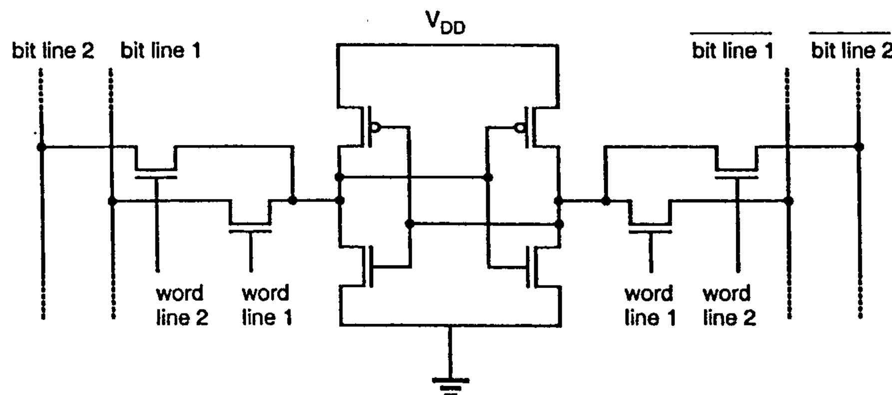


Figure 10.35. **Circuit diagram of the CMOS dual-port SRAM cell.**

writes data onto the same cell. In most cases, overlapping operations to the same memory location can be eliminated by a contention arbitration logic. It can either allow contention to be ignored and both operations to proceed, or it can arbitrate and delay one port until the operation on the other port is completed.

8.4 Dynamic Read-Write Memory (DRAM)

In the static CMOS read-write memory data is stored in six-transistor cells. Such a memory is fast and consumed small amount of static power. The only problem is that a SRAM cell occupies a significant amount of silicon space. This problem is addressed in the dynamic read-write memory (DRAM).

In a **dynamic RAM** binary data is stored as **charge in a capacitor**. The memory cell consists of a storage capacitor and an access transistor as shown in Figure 8.15.

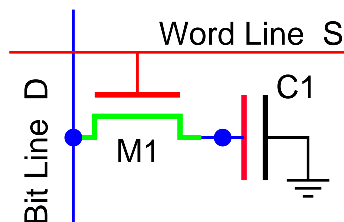


Figure 8.15: A one-transistor DRAM memory cell.

Data stored as charge in a capacitor can be retained only for a **limited time** due to the leakage current which eventually removes or modifies the charge.

Therefore, all dynamic memory cells require a **periodic refreshing** of the stored data before unwanted stored charge modifications occur.

Typical storage capacitance has a value of 20 to 50 fF.

Assuming that the voltage on the fully charged storage capacitor is $V = 2.5$V, and that the leakage current is $I = 40$pA, then the time to discharge the capacitor $C = 20$fF to the half of the initial voltage can be estimated as

$$t = \frac{1}{2}\frac{C \cdot V}{I} = \frac{20 \cdot 10^{-15} \cdot 2.5}{40 \cdot 10^{-12}} = 0.625\text{ms}$$

Hence ever memory cell must be refreshed approximately every half millisecond.

Despite of the need for additional refreshing circuitry SRAM has two fundamental features which have determined is enormous popularity:

- The DRAM cell occupies much smaller silicon area than the SRAM cell. The size of a DRAM cell is in the order of $8F^2$, where F is the smallest feature size in a given technology. For $F = 0.2\mu$m the size is $0.32\mu\text{m}^2$
- No static power is dissipated for storing charge in a capacitance.

The **storage capacitance** C_S, which is connected between the drain of the access transistor (the storage node) and the ground, is formed as a **trench** or **stacked**. capacitor.
The stacked capacitor is created between a second polysilicon layer and a metal plate covering the whole array area. The plate is effectively connected to the ground terminal.

A schematic of four adjacent SRAM cells is shown in Figure 8.16. A simplified layout of four adjacent SRAM cells related to the schematic is shown in Figure 8.17.
A typical layout consists of

- polysilicon word lines WL,
- metal 1 bit lines BL,
- n-diffusion forming an access nMOS with polysilicon gate,
- a bit-line contact BC between n-diffusion and metal1,
- a storage node contact SC between n-diffusion and second polysilicon layer forming one side of the storage capacitor,
- the metal 2 plate forming the common second side of all storage capacitors.

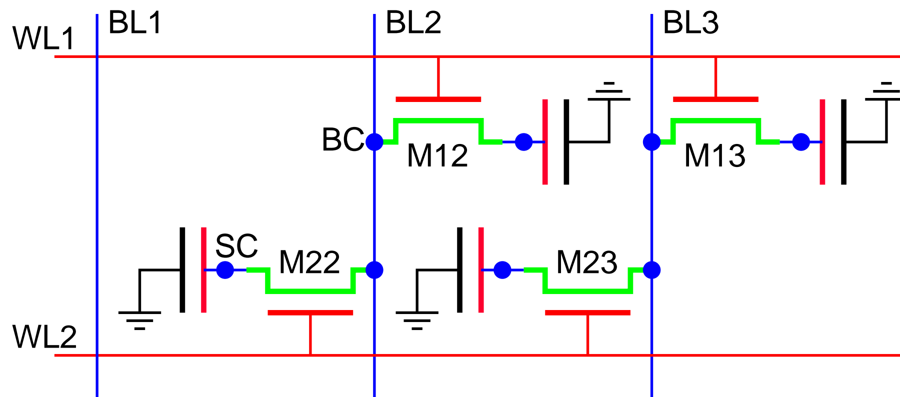


Figure 8.16: A schematic of four adjacent SRAM cells.

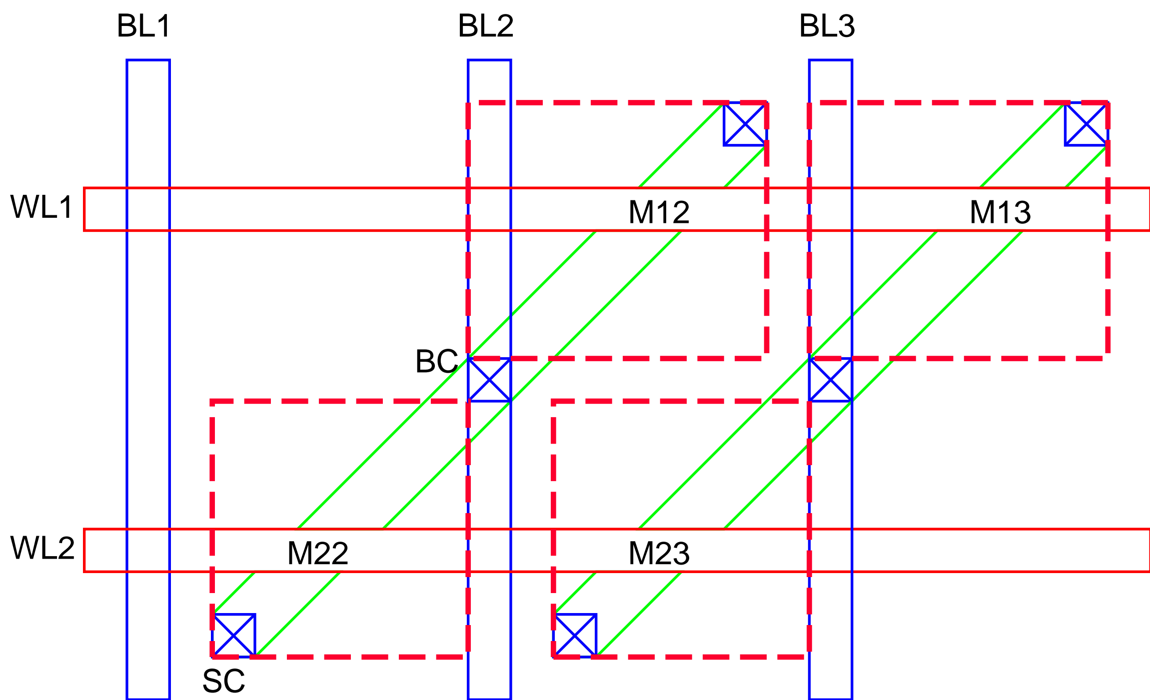


Figure 8.17: A simplified layout of four adjacent SRAM cells.

Note that the bit-line contacts BC are shared between two diagonally adjacent memory cells.

To consider **read/write operations** we have to take into account a significant parasitic capacitance C_C associated with each column, as shown in Figure 8.18.

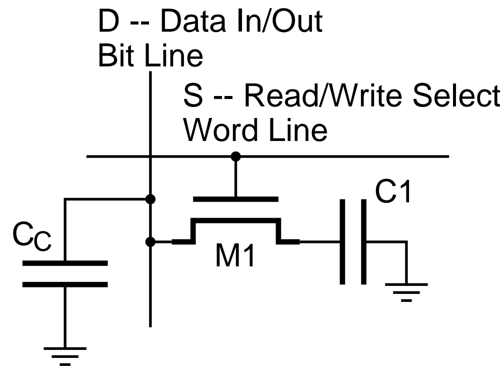


Figure 8.18: A single SRAM cells with a column capacitance shown.

Typically, before any operation is performed each column capacitance is precharged high.

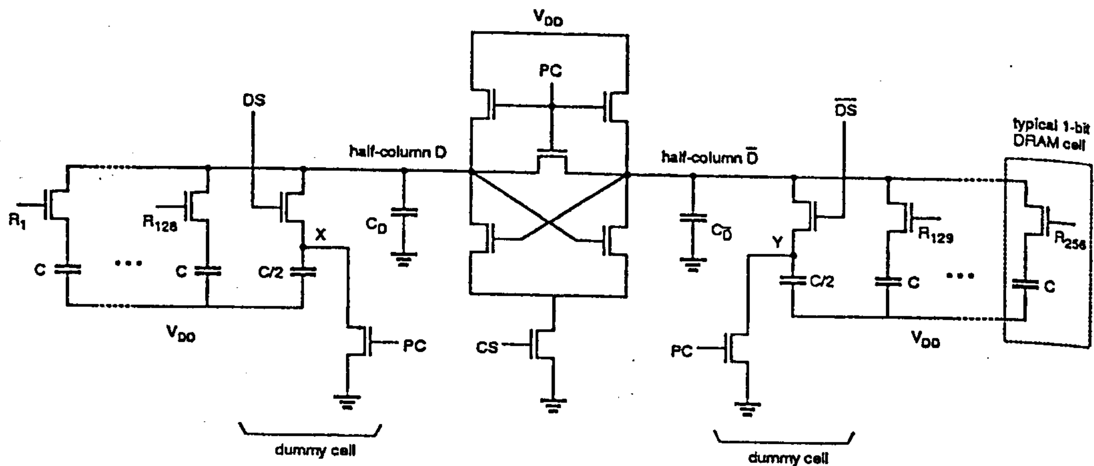
The cell is selected for a read/write operation by asserting its word line high ($S = 1$). This connects the storage capacitance to the bit line.

The **write operation** is performed by applying either high or low voltage to the bit line thus charging (write '1') or discharging (write '0') the storage capacitance through the access transistor.

During **read operation** there is a flow of charges between the storage capacitance C_1 and the column capacitance, C_C. As a result the column voltage either increases (read '1') or decreases (read '0') slightly. This difference can then be amplified by the sense amplifier.

Note that the read operation destroys the charge stored on the storage capacitance C_1 ("**destructive readout**"). Therefore the data must be restored (refreshed) each time the read operation is performed.

An example of the 256-cells-per-column DRAM read circuitry is shown in Fig. 10.45, along with typical control signal waveforms. A cross-coupled dynamic latch circuit is used to detect the small voltage differences and to restore the signal levels. The storage array is split in half so that equal capacitances are connected to each side of the cross-coupled latch. This means that half of the cells connected to one bit line (column) are arranged on one side of the latch, and the other half of the cells connected to the same column are arranged on the other side. As shown in Fig. 10.45, each half-column in the array also has a dummy cell which contains a capacitance half of the storage capacitance value. The capacitors C_D and $C_{\bar{D}}$ in Fig. 10.45 represent the relatively large parasitic column capacitances associated with the half-columns.



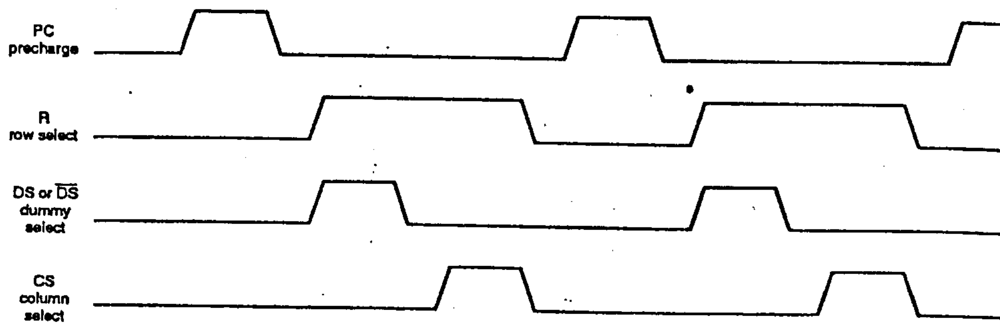


Figure 10.45. (a) Data read-restore circuit example for 256 1-T DRAM cells per column. (b) Typical control signal waveforms for two consecutive data read operations.

The "read-refresh"-operation occurs in three stages. First, the precharge devices are turned on during the active phase of PC. Both column capacitances C_D and $C_{\bar{D}}$ are charged-up to the same logic-high level, whereas the dummy nodes X and Y are pulled to logic-low level. The devices involved in the precharge operation are highlighted in Fig. 10.46. Note that during this phase, all other signals are inactive.

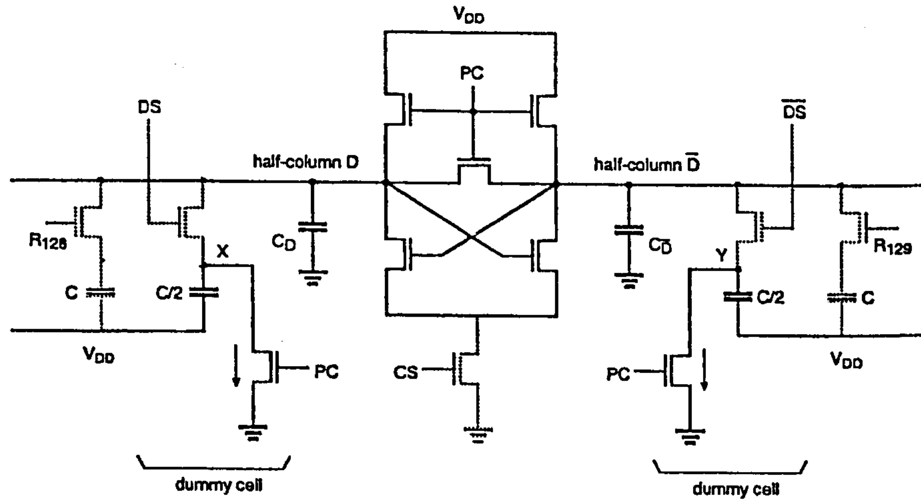


Figure 10.46. The half-columns are being charged-up during the precharge phase.

Next, one of the 256 word lines is raised to logic "1" during the row selection phase. At the same time, the dummy cell on the other side is also selected by raising either DS or $\overline{DS}$. This situation is depicted in Fig. 10.47, where only the selected

Figure 10.47. Two complementary column voltages are determined through charge sharing, on the one side between the selected storage cell and the half-column capacitance, on the other side between the dummy cell and the other half-column capacitance.

DRAM cell (left) and the corresponding dummy cell (right) are highlighted. If a logic "1" is stored in the selected cell, the voltage on the half-column D will rise slightly, while the voltage on half-column $\overline{D}$ drops, because the dummy cell is being charged up. If a logic "0" is stored in the selected cell, however, the voltage on the half-column D will also drop, and the drop in V_D will be larger than the drop in $V_{\overline{D}}$. Consequently, there will be a detectable difference between a stored "0" and a stored "1."

The final stage of the "read-refresh" operation is performed during the active phase of CS, the column-select signal. As soon as the cross-coupled latch is activated, the slight voltage difference between the two half-columns is amplified, and the latch forces the two half-columns into opposite states (Fig. 10.48). Thus, the stored data on the selected DRAM cell is refreshed while it is being read by the "read-refresh" circuitry.

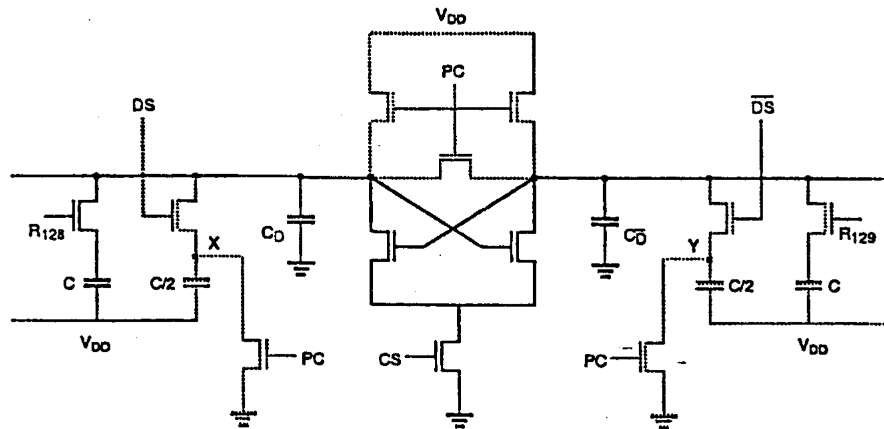


Figure 10.48. The cross-coupled latch circuit is used for detection of the voltage difference between the half-columns and for restoring the voltage level on the accessed cell.