

CHAPTER 11

SEMICONDUCTOR MEMORIES

11.1 INTRODUCTION

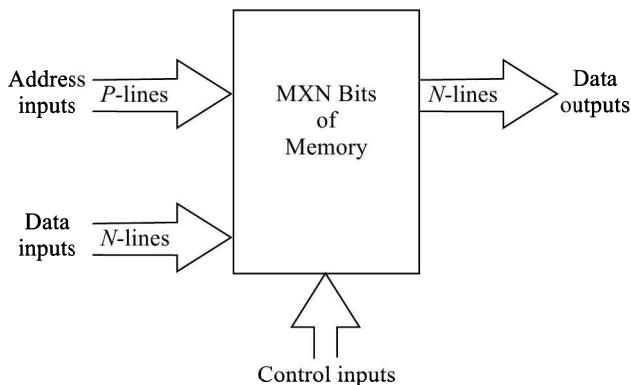
A digital processing system invariably requires a facility for storing digital information. The information usually consists of instructions (processing steps) coded in binary form, data to be processed, intermediate and final results, etc. The sub-system of a digital processing system, which provides the storage facility, is referred to as the *memory*. Till recently, the memories used were mostly of magnetic type. With unprecedented developments in semiconductor technology, it has become possible to make semiconductor memories of various types and sizes. These memories have become very popular due to their small size, low cost, high speed, high reliability, and ease of expansion of the memory size. Therefore, it is necessary for a designer of digital processors to know thoroughly the principles of operation and limitations of various semiconductor memory devices.

11.2 MEMORY ORGANISATION AND OPERATION

The basic element of a semiconductor memory is a which has been discussed in Chapter 7. The information is stored in binary form. There are a number of locations in a memory chip, each location being meant for one word of digital information. The number of locations and the number of bits comprising the word vary from memory to memory. The size of a memory chip is specified by two numbers M and N as $M \times N$ bits. The number M specifies the number of locations available in the memory and N is the number of bits at each location. In other words, this means that M words of N bits each can be stored in the memory. The commonly used values of the number of words per chip are 64, 256, 512, 1024, 2048, 4096, etc. whereas the common values for the word size are 1, 4, and 8, etc. Memories requiring higher number of words and/or larger word sizes can be formed by using these chips.

The block diagram of a memory device is shown in Fig. 11.1. Each of the M locations of the memory is defined by a unique *address* and, therefore, for accessing any one of the M locations, P inputs are required, where $2^P = M$. This set of lines is referred to as *address inputs* or *address bus*. The address is specified in the binary form. For convenience, octal and hexadecimal representations are commonly employed.

In fact, the address input is applied to a P -to- M decoder circuit, which activates one of its M outputs depending on the address and, thus, the desired memory location is selected.

Fig. 11.1 **Block Diagram of a Memory Device****Example 11.1**

Consider a memory of size 16 words. Find the binary address of each location.

Solution

Since $M = 16$, therefore, $2^P = M$ gives $P = 4$, i.e. for selecting one out of 16 words, a 4-bit address is required. The address is specified as $A_3 A_2 A_1 A_0$, where A_3 represents the most-significant bit (MSB) and A_0 represents the least-significant bit (LSB) of the address. The address of each location is given in Table 11.1.

Table 11.1 **Memory Addresses for Ex. 11.1**

Word number	Binary address			
	A_3	A_2	A_1	A_0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

The number of inputs required to store the data into or read the data from any memory location is N . One set of N lines is required for storing the data into the memory, referred to as *data inputs* and another set of N lines is required for reading the data already stored in the memory, which is referred to as *data outputs*. The concept of *bus* is used to refer to a group of conductors carrying related set of signals. Therefore, the set of lines meant for data inputs is *input data bus* and for data outputs is *output data bus*. Input and output data buses are unidirectional, i.e. the data can flow in one direction only. In most of the memory chips available, the same set of lines is used for data input as well as data output and is referred to as *bidirectional bus*. This means that the data bus is time multiplexed. It is used as input bus for some specific time and as output bus for some other time depending upon a Read/Write control input as shown in Fig. 11.2.

A number of control inputs are required to give commands to the device to perform the desired operation. For example, a command signal is required to tell the memory whether a read or a write (R/W is Fig. 11.2) operation is desired.

When R/\bar{W} is HIGH, the data bus will be used for reading the memory (output bus) whereas when R/\bar{W} is LOW, the bus will be acting in the input direction and the data on the bus will go into the memory. Other command include inputs chip enable (CE), chip select (CS), etc.

In addition to the above-mentioned functional pins, a minimum of two pins are required for power supply and ground. The internal organization of a 16×4 memory chip is illustrated in Fig. 11.3. The write and read operations are discussed below.

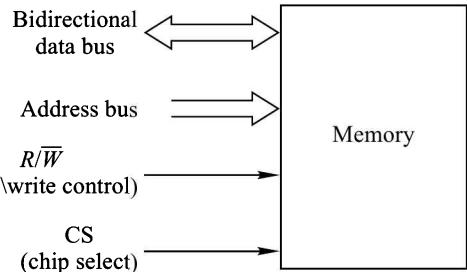


Fig. 11.2 **Block Diagram of Memory with Bidirectional Data Bus**

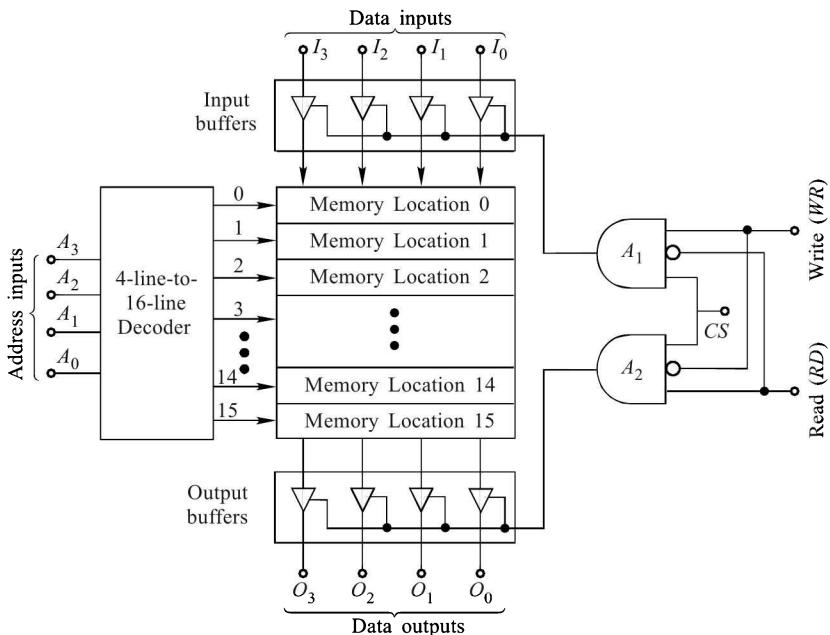


Fig. 11.3 **Internal Organization of a 16×4 Memory Chip**

11.2.1 Write Operation

To write a word into the selected memory location requires logic 1 voltage to be applied to *CS* (Chip select), and Write (*WR*) inputs and logic 0 voltage to Read (*RD*) input. This combination of inputs gives outputs of AND gates A_1 and A_2 1 and 0 respectively. A 1 at the output of A_1 enables the input buffers so that the 4-bit word applied to the data inputs will be loaded (entered) into the selected (addressed) memory location. A 0 at the output of A_2 disables (tristate) the output buffers so that the data outputs are not available. The outputs are in the Hi-Z state.

For writing a word into a particular memory location, following sequence of operations is to be performed:

1. The chip select signal is applied to the *CS* terminal.
2. The word to be stored is applied to the data-input terminals.
3. The address of the desired memory location is applied to the address-input terminals.
4. A write command signal is applied to the write-control input terminal with *RD* = 0.

In response to the above operations, the addressed memory location is cleared of any word that might have been stored in it earlier, and the information presented at the data input terminal replaces it.

Figure 11.4 illustrates the various waveforms during the write operation. The important timing characteristics of the write cycle are:

Write Cycle Time (t_{WC})

This is the minimum amount of time for which the valid address must be present for writing a word in the memory. In other words, it is the minimum time required between successive write operations.

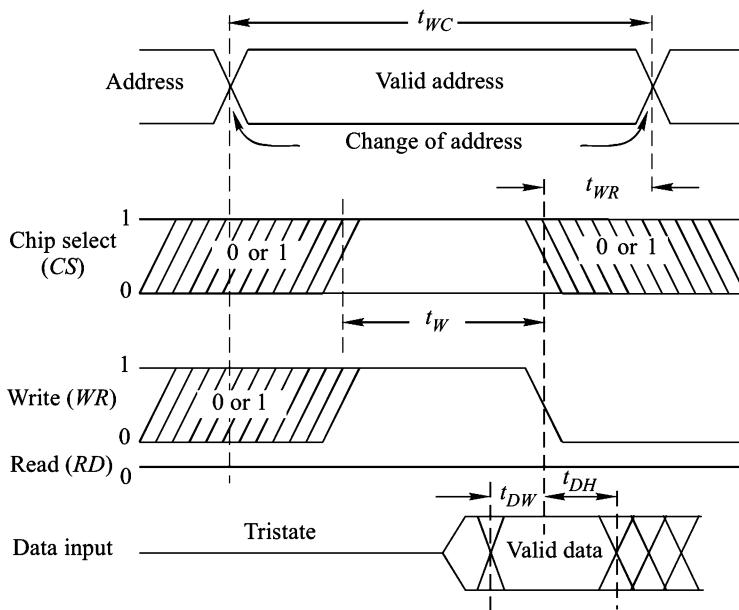


Fig. 11.4 **Write-Cycle Waveforms**

Write Pulse Time (t_w)

This is the minimum length of the write pulse.

Write Release Time (t_{WR})

This is the minimum amount of time for which the address must be valid after the write pulse ends.

Data Set Up Time (t_{DW})

This is the minimum amount of time for which the data must be valid before the write pulse ends.

Data Hold Time (t_{DH})

This is the minimum amount of time for which the data must be valid after the write pulse ends.

11.2.2 Read Operation

In order to read the contents of a selected memory location, the Read (RD), and the Chip select (CS) inputs must be at logic 1 level and WR at logic 0 level. This gives output of $A_2 = 1$ which enables the output buffers so that the contents of the selected (addressed) memory location will appear at the data outputs. $RD = 1$ tristates the input buffers so that the data inputs do not affect the memory during a read operation.

To read (or retrieve) a data word, known to be stored at a particular address, the following sequence of operations is required to be performed:

1. The chip-select signal is applied to the CS terminal.
2. The address of the desired memory location is applied to the address-input terminals.
3. A read-command signal is applied to the read control-input terminal.

In response to the above operations, the data word stored at the addressed location appears on the data-output terminals.

Figure 11.5 illustrates the various waveforms during the read operation. The important timing characteristics of the read cycle are:

Read Cycle Time (t_{RC})

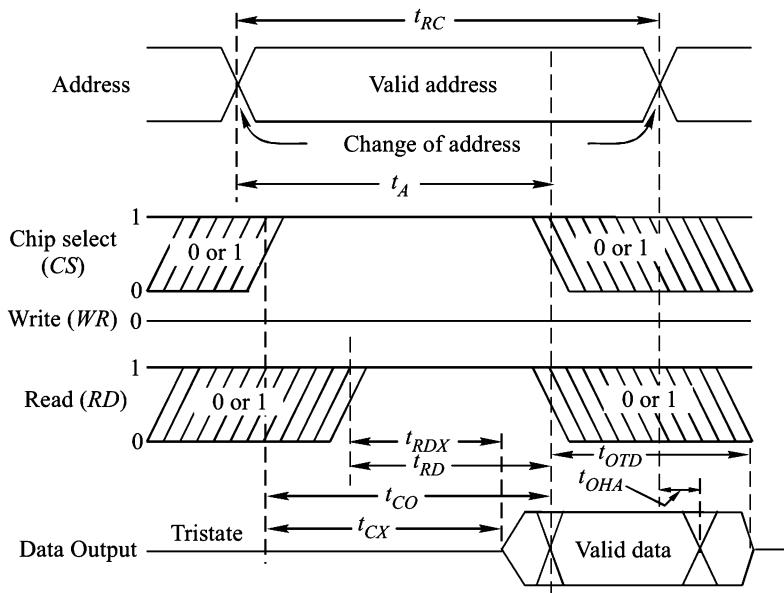
This is the minimum amount of time for which the valid address must be present for reading a word from the memory. In other words, it is the minimum time required between successive read operations.

Access Time (t_A)

This is the maximum time from the start of the valid address of the read cycle to the time when the valid data is available at the data outputs. The access time is at most equal to the read-cycle time, i.e. $t_A \leq t_{RC}$. In other words, the data outputs might be ready before the memory is actually ready for the next read operation.

Read To Output Valid Time (t_{RD})

This is the maximum time delay between the beginning of the read pulse and the availability of valid data at the data outputs.

Fig. 11.5 *Read-Cycle Waveforms*

Read To Output Active Time (t_{RDX})

This is the minimum time delay between the beginning of the read pulse and the output buffers coming to active state (from the high-impedance state).

Chip-Select To Output Valid Time (t_{CO})

This is the maximum time delay between the beginning of the chip-select pulse and availability of valid data at the data outputs.

Chip-Select To Output Active Time (t_{CX})

This is the minimum time delay between the beginning of the chip-select pulse and the output buffers coming to active state.

Output Tristate From Read (t_{OTD})

This is the maximum time delay between the end of the read pulse and the output buffers going to high-impedance state.

Data Hold Time (t_{OHA})

This is the minimum time for which the valid data is available at the data outputs after the address ends.

The write- and read-cycle timings of a typical memory chip are given in Table 11.2.

Table 11.2 *Timing Parameters of a Typical Memory Chip*

Parameter	Time (ns)
t_{WC}	200
t_w	120
t_{WR}	0
t_{DW}	120
t_{DH}	0
t_{RC}	200
t_A	200
t_{RD}	70
t_{RDX}	20
t_{CO}	70
t_{CX}	20
t_{OTD}	60
t_{OHA}	50

Example 11.2

For the memory timing of Table 11.2, find the maximum rate (words/second) at which

- (a) Data can be stored, and
- (b) Data can be read.

Solution

- (a) The maximum rate at which data can be stored is

$$\frac{1}{t_{WC}} = \frac{1}{200 \times 10^{-9}} = 5 \times 10^6 \text{ words/s}$$

- (b) The maximum rate at which data can be read is

$$\frac{1}{t_{RC}} = \frac{1}{200 \times 10^{-9}} = 5 \times 10^6 \text{ words/s}$$

11.3 EXPANDING MEMORY SIZE

In many memory applications, the required memory capacity, i.e. the number of words and/or word size, cannot be satisfied by a single available memory IC chip. Therefore, several similar chips have to be combined suitably to provide the desired number of words and/or word size.

11.3.1 Expanding Word Size

If it is required to have a memory of word size n and the word size of the available memory ICs is N ($n > N$), then a number of similar ICs can be combined together to achieve the desired word size. The number of IC

chips required is an integer, next higher to the value n/N . These chips are to be connected in the following way:

1. Connect the corresponding address lines of each chip individually, i.e. A_0 of each chip is connected together and it becomes A_0 of the overall memory. Similarly, connect other address lines together.
2. Connect the RD input of each IC together and it becomes the read input for the overall memory. Similarly, connect the WR and CS inputs.

Now, the number of data-input/output lines will be equal to the product of the number of chips used and the word size of each chip. The following example illustrates the above procedure clearly.

Example 11.3

Obtain a 16×8 memory using 16×4 memory ICs.

Solution

Since the word size required is $n = 8$ and the word size of the available IC is $N = 4$, therefore, $n/N = 2$ chips are required to obtain the desired memory.

Since each chip can store 16 4-bit words and we want to store 16 8-bit words, each chip is required to store half of each word. Figure 11.6 shows the relevant connections of the two chips. Here, we have assumed bidirectional input/output (I/O) lines which is common in available memory chips. In this 16×8 memory, the higher order four bits (D_7, D_6, D_5, D_4) of each 8-bit word are located in memory M_1 and the lower order four bits (D_3, D_2, D_1, D_0) are located in memory M_0 .

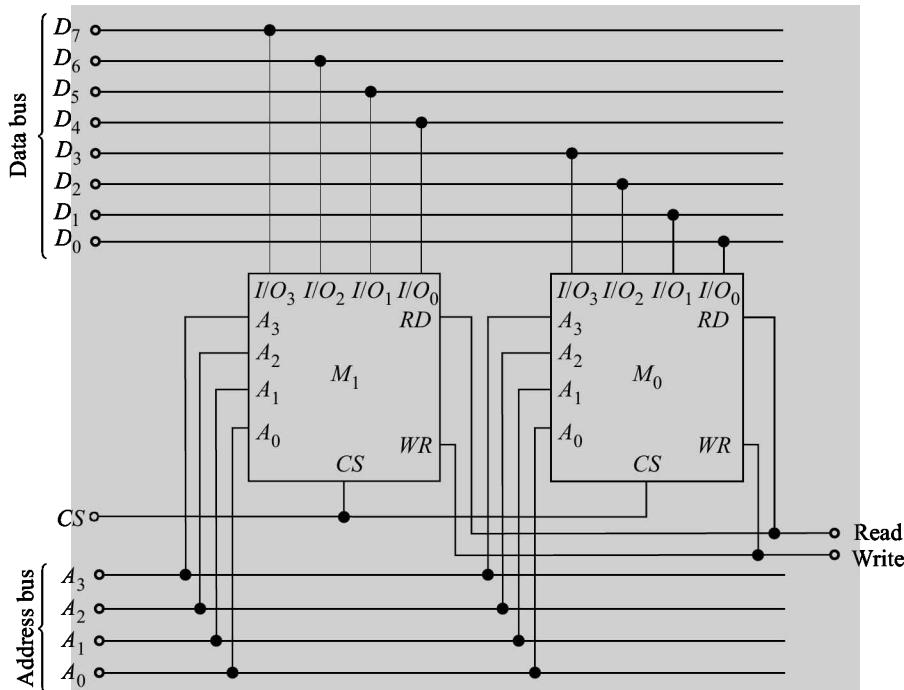


Fig. 11.6 A 16×8 Memory Obtained by Combining Two 16×4 Memory Chips

11.3.2 Expanding Word Capacity

Memory chips can be combined together to produce a memory, with the desired number of locations. To obtain a memory of capacity m words, using the memory chips with M words each, the number of chips required is an integer next higher to the value m/M . These chips are to be connected in the following way:

1. Connect the corresponding address lines of each chip individually.
2. Connect the RD input of each chip together. Similarly, connect the WR inputs.
3. Use a decoder of proper size and connect each of its outputs to one of the CS terminals of memory chips. For example, if eight chips are to be connected, a 3-line-to-8-line decoder is required to select one out of eight chips at any one time.

The following example clearly illustrates the above procedure.

Example 11.4

Obtain a 2048×8 memory using 256×8 memory chips.

Solution

The number of chips required is $\frac{2048}{256} = 8$. At any one time, only one of the 2048 locations is to be accessed, which

will be in one of the eight chips. That means only one of the eight chips must get selected at a time.

For selecting one out of 2048 locations, the number of address lines required is 11 ($2^{11} = 2048$). The lower order eight bits of the address $A_7 - A_0$ will be same for each chip, and the higher order three bits of the address $A_{10} - A_8$ must select one out of the eight chips. For this purpose, a 3-line-to-8-line decoder is required. The memory connections are shown in Fig. 11.7. Here, we have assumed a common terminal (R/\bar{W}) for read and write. For the read operation, logic 1 is to be applied to R/\bar{W} , whereas logic 0 is to be applied for the write operation. The chip-select input is assumed to be active-low. The addresses of the chips are given in Table 11.3.

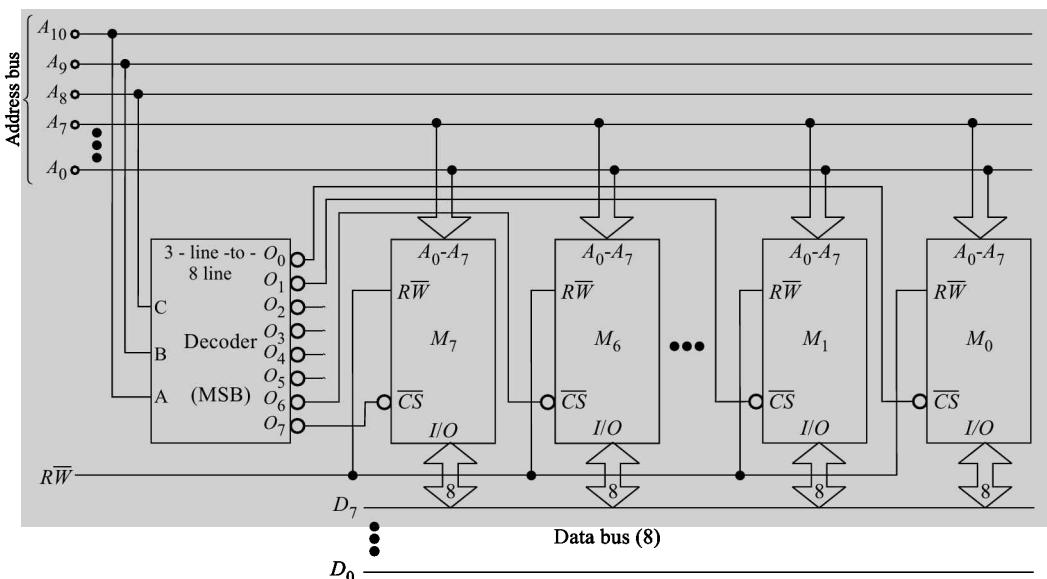


Fig. 11.7 A 2048×8 Memory Obtained by Combining Eight 256×8 Memory Chips

Table 11.3 *Addresses of the Memory Chips*

Memory chip	Addresses (hex.)
M_0	000 – 0FF
M_1	100 – 1FF
M_2	200 – 2FF
M_3	300 – 3FF
M_4	400 – 4FF
M_5	500 – 5FF
M_6	600 – 6FF
M_7	700 – 7FF

11.4 CLASSIFICATION AND CHARACTERISTICS OF MEMORIES

Various memory devices can be classified on the basis of their principle of operation, physical characteristics, mode of access, technology used for fabrication, etc.

11.4.1 Principle of Operation

Memories can be classified according to their principle of operation. The most commonly used memories are:

- Read-only memories (ROM)
- Read-and Write memories (RWM or RAM)
- Flash memories
- Content-addressable memories (CAM)
- First-in, First-out (FIFO) memories

Read-Only Memories (ROM)

Read-only-memory (ROM), as the name suggests, is meant only for reading the information from it. This does not mean that information is not written into it, because unless some information is stored into it, there cannot be anything to read from. Actually, the process of entering information into this type of memory is much more complicated than for RAM and is done outside the system where it is used. Therefore, it is called as read-only memory. It is used to store information which is fixed, such as tables for various functions, fixed data and instructions. The ROMs are also organised so that every memory location requires equal time for reading the data already stored in that location.

The various types of read-only memories are further sub-classified on the basis of the technique employed for storing information into the memory (referred to as *programming*) or their erasability properties. These are:

Read-Only Memory

It is programmed at the time of manufacturing, as the last process of fabrication, according to the information specified by the user. It is referred to as *custom programmed* or mask programmed. The data stored can not

be changed after fabrication. Since, this process is quite costly, therefore, this type of ROM is suitable only for bulk requirements, of the order of millions of chips.

Programmable Read-Only Memory (PROM)

This type of ROM is programmed by the user using a special circuit—a PROM programmer. A PROM can be programmed only once after which its contents are permanently fixed as in a ROM. This type of ROM is suitable for storage of data which is of permanent nature. The chip is available without any data stored from the vendor.

Erasable and Programmable ROM

This type of ROMs are reprogrammable i.e. it can be programmed again and again. It is referred to as *erasable and programmable* ROM.

There are two techniques used for erasing; in one technique the chip is exposed to ultraviolet radiation, and in the other technique the contents are altered electrically. The erasable programmable ROM using ultraviolet radiation for erasing is known as EPROM, whereas the device using electrical voltage for erasing is known as *electrically alterable* ROM (EAROM) or (EEPROM).

Electrically Erasable and Programmable (EEPROM)

The detailed operation of various types of ROMs is discussed in Section 11.5.

Random Access Memories (RWM or RAM)

In this type of memories, the memory locations are organised in such a way so that any memory location requires equal time for writing or reading. This type of memory is also known as read-and-write memory (RWM) or RAM. RAMs can be static or dynamic and are fabricated using bipolar or unipolar technologies. The static RAM (SRAM) generally uses bistable latch as storage element, whereas the *dynamic* RAM (DRAM) uses capacitor as storage element which requires periodic refreshing. The SRAMs are faster than the DRAMs. However, DRAMs can store much more data than SRAMs for a given physical size and cost. The SRAMs can be fabricated by using bipolar devices or MOSFETs, but the DRAMs can only be made using MOSFETs.

The SRAMs can be *asynchronous* SRAM or *synchronous* SRAM. An asynchronous SRAM is one in which the operation is not synchronised with a system clock. The operation a synchronous SRAM is synchronised with the system clock. The synchronous SRAMs normally have a *burst* feature. The burst feature allows the memory to read or write at upto four locations using a single address. The two lowest order address bits A_1 and A_0 are applied to the burst logic circuit which contains a Mod-4 counter. This produces a sequence of four internal addresses by using the two lowest order bits of the address as 00, 01, 10, and 11 on successive clock pulses. The sequence always begins with the base address, which is the external address applied.

The types of DRAMs are:

- Fast Page Mode DRAM (FPM DRAM)
- Extended Data Out DRAM (EDO DRAM)
- Burst EDO DRAM (BEDO DRAM)
- Synchronous DRAM (SDRAM)

The first three types of DRAMs are asynchronous DRAMs.

The detailed operations of bipolar SRAM, and MOSFET SRAM and DRAM are discussed in Section 11.6.

Flash Memories

Flash memories are non-volatile, large bit storage capacity, read and write memories. The storage cell in a flash memory consists of a single floating-gate MOS transistors. Its operation is explained in Section 11.7.

Content Addressable Memories (CAM)

Is a special purpose random access memory which performs association operation in addition to read/write operations. The detailed operation of the CAM is discussed in Section 11.8.

First-in, First-out Memories (FIFO)

In this type of memory, the data which is entered first is taken out first. The storage device used in this is a SRAM array with two separate ports that allows data to be written into and read from its array at independent data rates. The detailed operation of FIFO memory is discussed in Section 11.9.

11.4.2 Physical Characteristics

Memories can be classified according to their physical characteristics, such as:

1. Erasable or non-erasable, and
2. Volatile or non-volatile.

Erasable or Non-Erasable Memories

A memory in which the information stored can be erased and new information stored is called *erasable* memory. On the other hand, the information stored in the *non-erasable* memory cannot be erased, for example ROM is non-erasable.

The erase operation is performed in the following ways:

- Electrically
- By exposing the chip to ultraviolet (UV) radiation. The EEPROMs are electrically erasable whereas the EPROMs are erased by exposing the chip to UV radiation.

Volatile or Non-Volatile Memories

If the information stored in a memory is lost when electrical power is switched off, the memory is referred to as a *volatile memory*. For example, the RAM is a volatile memory. On the other hand, in a *non-volatile memory*, the information once stored remains intact until changed deliberately. All types of ROMs are non-volatile memories.

11.4.3 Mode of Access

Mode of access refers to the manner in which a memory location is accessed for reading or writing. In case of ROMs only reading is possible. There are two modes of access. These are:

- Sequential access, and
- Random access.

Sequential memories are referred to as sequentially accessed memories, whereas RAM, ROM, and CAM are random-access memories. In random-access memories any memory location requires equal time for accessing (*access time*) whereas the access time is different for different locations in the case of sequentially accessed memory.

11.4.4 Fabrication Technology

Memories can be classified on the basis of the fabrication technology used. The two broad categories of memories based on fabrication technology used are:

1. Bipolar, and
2. Unipolar (MOS).

These technologies have been discussed in detail in Chapter 4. Static RAM, ROM and PROM can be fabricated using either bipolar technology (TTL, ECL, etc.) or MOS technology, whereas dynamic RAM, EPROM, EEPROM and flash memories can be fabricated using only unipolar devices (MOSFETs).

11.5 READ-ONLY MEMORY

A read-only memory (ROM) is a semiconductor memory device used to store information which is permanent in nature. It has become an important part of many digital systems because of its low cost, high speed, system-design flexibility and data non-volatility. The read-only memory has a variety of applications in digital systems, such as implementation of combinational logic and sequential logic, character generation, look-up table, microprocessor programme storage, etc.

ROMs are well-suited for LSI manufacturing processes and are available in many forms. Two major semiconductor technologies are used for the manufacturing of ROM integrated circuits, viz. bipolar technology and MOS technology, which differ primarily in access time. In general, bipolar devices are faster and have higher drive capability, whereas MOS devices require less silicon area and consume less power. With improvements in MOS technology, it is now possible to make MOS memories with speeds comparable to those of bipolar memories.

The process of entering information into a ROM is referred to as *programming* the ROM. Depending on the programming process employed, the ROMs are categorized as:

1. *Mask programmable read-only memories*, which are referred to as ROMs. In these memories, the data pattern must be programmed as part of the fabrication process. Once programmed, the data pattern can never be changed. These are highly suited for very high volume usage due to their low cost.
2. *Programmable read-only memories*, which are referred to as PROMs. A PROM is electrically programmable, i.e. the data pattern is defined after final packaging rather than when the device is fabricated. The programming is done with an equipment referred to as *PROM programmer*. The programming techniques used will be discussed later.
3. *Erasable programmable read-only memories*, which are referred to as EPROMs. As the name suggests, in these memories, data can be written any number of times, i.e. they are reprogrammable. Reprogrammable ROMs are possible only in MOS technology. For erasing the contents of the memory, one of the following two methods are employed:
 - (a) Exposing the chip to ultraviolet radiation for about 30 minutes.
 - (b) Erasing electrically by applying voltage of proper polarity and amplitude. Electrically erasable PROM is also referred to as E²PROM or EAROM (Electrically alterable ROM).

11.5.1 ROM Organisation

A read-only memory is an array of selectively open and closed unidirectional contacts. The address decoder of Fig. 11.3 is usually divided in two parts for simplifying the decoder design. One half of the address lines are decoded by one decoder used to energize one of the row lines, whereas the other half of the address lines are decoded by another decoder used to activate column lines. This method of addressing is referred to as two-dimensional, $X-Y$, or coincident-selection, addressing. A unidirectional switch is incorporated at the junction of every row and column.

A 16-bit ROM array is shown in Fig. 11.8. To select any one of the 16 bits, a 4-bit address (A_3, A_2, A_1, A_0) is required. The lower order two bits (A_1, A_0) are decoded by the decoder D_L which selects one of the four rows, whereas the higher order two bits (A_3, A_2) are decoded by the decoder D_H which activates one of the four-column sense amplifiers.

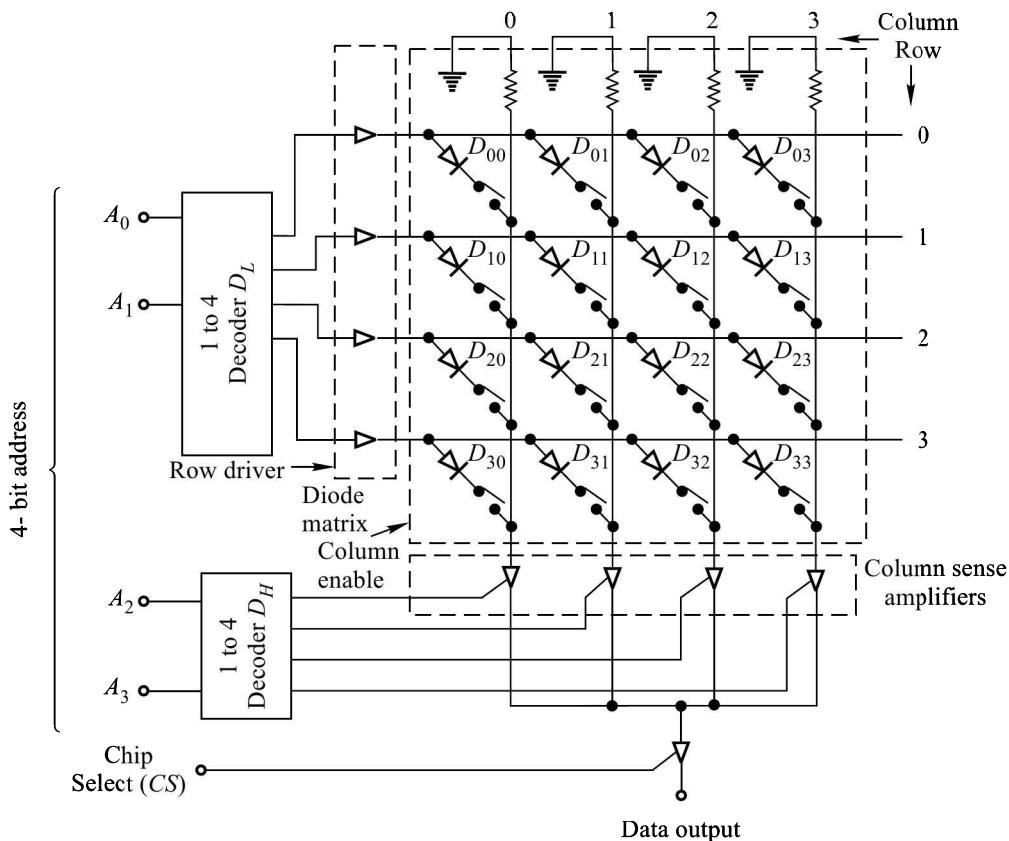


Fig. 11.8 16-bit ROM Array

The diode matrix is formed by connecting one diode, along with a switch between each row and column. For example, the diode D_{21} is connected between row 2 and column 1.

The output is enabled by applying logic 1 at the chip select (CS) input.

Programming a ROM means to selectively open and close the switches in series with the diodes. For example, if the switch of the diode D_{21} is in closed position and if the address input is 0110, row 2 is activated connecting it to column 1. Also, the sense amplifier of column 1 is enabled which gives logic 1 output if the chip is selected ($CS = 1$). This shows that a logic 1 is stored at the address 0110. On the other hand, if the switch of diode D_{21} is open, logic 0 is stored at the address 0110.

Example 11.5

In the 16-bit ROM of Fig. 11.8, the switches of diodes D_{00} , D_{03} , D_{12} , D_{13} , D_{21} , and D_{33} are programmed as closed. Find the bit stored in each location.

Solution

The bit stored in each location are given in Table 11.4.

Table 11.4 Bit Pattern of Ex. 11.5

Address				Bit stored
A_3	A_2	A_1	A_0	
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

The ROMs can also be implemented by using bipolar junction transistors or MOSFETs, instead of diodes. Each transistor is to be connected as shown in Fig. 11.9a for bipolar memories and as shown in Fig. 11.9b, for MOS memories. Each column line is to be connected to ground through a resistance for bipolar devices, whereas it is to be connected to the V_{DD} supply through a load MOSFET for MOS devices. The fuse used in the circuit of Fig. 11.9a is meant for programming, and is either retained intact or blown off depending upon whether 1 or 0 is to be stored in a selected location.

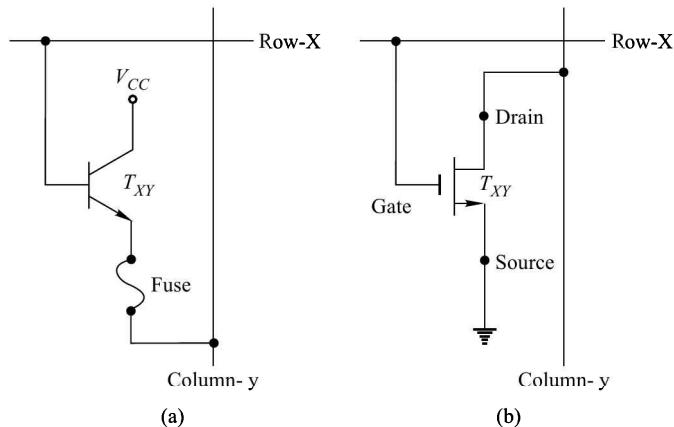


Fig. 11.9 ***Unidirectional Switch (a) Bipolar (b) MOS***

11.5.2 Programming Mechanisms

Mask Programmable ROMs

Integrated circuits are fabricated through a number of processing steps, such as photomasking, etching, diffusion, etc. One of the final steps in the manufacturing process is to deposit a layer of aluminium on the entire surface of the silicon wafer. The desired interconnecting pattern is produced by selectively etching away portions of aluminium. The row-to-column contacts can be, retained or etched away, as desired, in the final aluminium etching process, in the manufacture of mask-programmed read-only memories.

Programmable ROMs

The most commonly used programming mechanisms for the bipolar memory devices are:

1. Fusible link process, and
 2. Avalanche-induced migration (AIM) process.

In the fusible link process, a fuse is added in the emitter lead of every transistor (Fig. 11.9a). Usually, the fuse material used is either nichrome or polycrystalline silicon. At the time of the fabrication of these memory devices, all the fuse links are in place and can be selectively open circuited after packaging.

When nichrome is used as the fuse material, it is deposited as a very thin layer, which can be blown off (opening the connection between the row and column lines) by making a large current (20–50 mA) to flow through it. The fusing time varies between 5 and 200 μ s. Typically, the programming rate is 5 ms per bit. It has power dissipation of about 170 μ W per bit while operating from a 5 V supply. The nichrome fuse elements are to be left intact for storing 1s and blown off for storing 0s.

Another fuse material used is polycrystalline silicon. It is deposited as a thick layer (~ 3000Å) during the manufacturing process. The fuse is blown with a pulse train of successively wider pulses. Typically, a current of 20–30 mA is needed to blow the fuse. Temperatures of the order of 1400°C are reached during the blowing process. The silicon gets oxidized and forms an insulating material.

From Fig. 11.9a, we observe the following: When a row is selected (say X), the transistor T_{XY} is turned ON. If the fuse is intact, the column line Y is pulled towards V_{cc} (logic 1) whereas if the fuse is blown (open), the column bus is left floating (logic 0).

Another mechanism used for programming PROMs utilizes an avalanche approach. Figure 11.10 shows the arrangement used. In this, the diode D_1 is reverse-biased and the heavy flow of electrons in the reverse direction causes aluminium atoms from the emitter contact to migrate through the emitter to the base. This causes an emitter-to-base short. This process requires higher currents (200–300 mA) and voltages than the fusible link, but is faster, requiring 0.02–0.05 ms time.

The above mechanisms are irreversible, i.e. a device can be programmed only once in its life time. These mechanisms do not work with MOS memory devices, where resistance and current levels required for the fusing process are incompatible with MOS impedance levels. Commonly used MOS technologies for the fabrication of programmable memories are:

1. Floating gate avalanche injection MOS (FAMOS), and
2. MAOS.

FAMOS PROM

In this, the storage device used is silicon gate MOSFET with no electrical connection to the gate, i.e. the gate is electrically floating in an insulating layer of silicon dioxide, as shown in Fig. 11.11a. The symbol of FAMOS is shown in Fig. 11.11b.

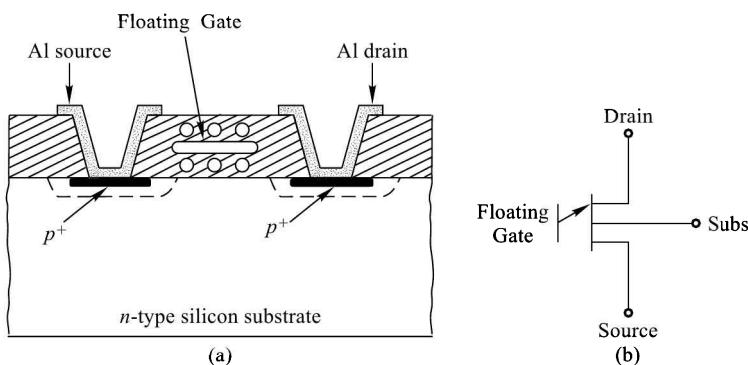


Fig. 11.11 *p-channel FAMOS Device: (a) Cross-Sectional View, and (b) Symbol*

The operation of this device depends on the charge transport to the floating gate by avalanche injection of electrons from either the source or drain, caused by the application of high voltage (25–30 V) across the transistor. The amount of charge transferred to the floating gate is a function of the amplitude and duration of the applied voltage. The presence or absence of charge can be sensed by measuring the conductance between the source and the drain.

When the applied voltage is removed, the charge remains trapped in the gate, since no discharge path is available for the accumulated electrons because the gate is surrounded by a very low conductivity dielectric. The transistor now behaves as if an external voltage were permanently connected to the gate terminal.

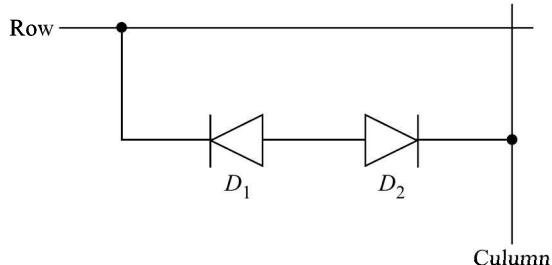


Fig. 11.10 *A Shorted Junction Cell*

The charge accumulated on the gate can be removed by illuminating the FAMOS device with ultraviolet light. This results in the flow of a photo current from the floating gate back to the silicon substrate, thereby discharging the gate to its initial condition.

These devices are packaged with a transparent quartz lid for exposing the device to ultraviolet radiation, for the purpose of erasing. Since erasing is possible in these devices, these are reprogrammable and are known as *erasable programmable read-only memory* (EPROM) devices. In case the device is packaged in inexpensive package without quartz lid, it works as a *one-time programmable* (OTP) ROM which is same as a PROM.

The use of second metal gate (*erase gate*) permits reprogramming without the use of ultraviolet radiation. Figure 11.12 shows a cross-sectional view of a *p*-channel device with an erase gate. The programming is done by applying a negative voltage to both the source (V_{SX}) and the drain (V_{DX}), for inducing avalanche breakdown at both junctions. Simultaneously, a positive voltage applied to the second gate G_2 (V_{G2S}) increases the rate at which electrons accumulate on the floating gate. As electrons accumulate on the floating gate, the channel appears between the source and the drain and the device turns ON.

For the purpose of erasing, again a negative voltage is applied to both the source and drain junctions, for inducing avalanche breakdown similar to programming mode, but a negative voltage is applied to the gate G_2 . This results in the accumulation of holes on the floating gate, which neutralizes the existing charge. Thus, the erasing is done electrically rather than with ultraviolet radiation.

These devices are referred to as *electrically alterable ROMs* (EAROMs) or *electrically erasable and programmable ROMs* (EEPROMs or E²PROMs), and are reprogrammable.

MAOS PROM

Another PROM uses the gate dielectric, such as alumina (Al_2O_3) and silicon nitride itself, for charge storage and provide a reprogramming feature. A cross-sectional view of such a device using alumina dielectric is shown in Fig. 11.13. This device is referred to as MAOS memory element.

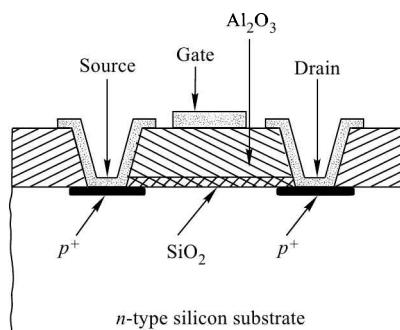


Fig. 11.13 *Cross-Sectional View of a MAOS Device*

For a *p*-channel device, a positive gate voltage of about 50 V amplitude is required for 10–20 μs , for programming. Erasing requires a voltage of opposite polarity on the gate.

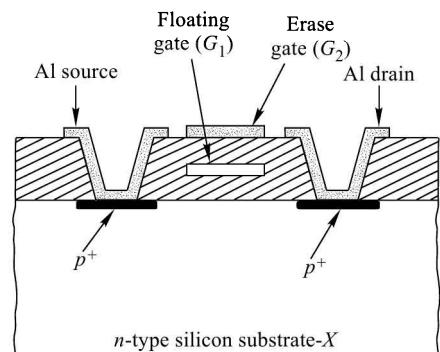


Fig. 11.12 *Cross-Sectional View of a FAMOS Device with an Erase Gate*

11.5.3 ROM ICs

Various types of programmable/erasable ROM ICs are commercially available. The programmable ROM devices (PROM) are available in bipolar technology in which erasing is not possible. In CMOS technology, PROM devices are available which are actually EPROM devices with the provision of only one time programming. These EPROM devices are without quartz lid and therefore, erasure is not possible in these devices. These EPROMs are known as one time programmable (OTP EPROM). The other type of EPROM with quartz lid are used for multiple programming. For reprogramming, the already stored data is erased by exposing the chip to ultraviolet light. The electrically erasable and programmable read-only memory (EEPROM) devices are available as *parallel EEPROM* and *serial EEPROM*. Some of the available ICs are given in Table 11.5. A number of low-voltage CMOS (LVC MOS) devices are also available which require lower than 5 V power supply.

Some of these devices are briefly described here. However, their complete technical and operational details can be obtained from the websites of the manufacturers/vendors.

Table 11.5 *Available ROM ICs*

IC No.	Organization No. of bits	Output*	Power supply voltage	Technology	Types of ROM
74S188	32 × 8	O.C	5 V	Schottky TTL	PROM
74S288	32 × 8	TS	5 V	Schottky TTL	PROM
74S571	512 × 4	TS	5 V	Schottky TTL	PROM
27C010	128 K × 8	TS	5 V	CMOS EPROM	OTP EPROM
27C020	256 K × 8	TS	5 V	CMOS EPROM	OTP EPROM
27C040	512 K × 8	TS	5 V	CMOS EPROM	OTP EPROM
27C080	1024 K × 8	TS	5 V	CMOS EPROM	OTP EPROM
27C1024	64 K × 16	TS	5 V	CMOS EPROM	OTP EPROM
27C64	8 K × 8	TS	5 V	CMOS EPROM	EPROM/OTP EPROM
27C128	16 K × 8	TS	5 V	CMOS EPROM	EPROM/OTP EPROM
27C256	32 K × 8	TS	5 V	CMOS EPROM	EPROM/OTP EPROM
27C512	64 K × 8	TS	5 V	CMOS EPROM	EPROM/OTP EPROM
28C64	8 K × 8	TS	5 V	NV CMOS	Parallel EEPROM
28C256	32 K × 8	TS	5 V	NV CMOS	Parallel EEPROM
28C010	128 K × 8	TS	5 V	NV CMOS	Parallel EEPROM
28C040	512 K × 8	TS	5 V	NV CMOS	Parallel EEPROM
24C01	128 × 8	O.D	5 V	NV CMOS	Serial EEPROM
24C02	256 × 8	O.D	5 V	NV CMOS	Serial EEPROM
24C04	512 × 8	O.D	5 V	NV CMOS	Serial EEPROM
24C08	1024 × 8	O.D	5 V	NV CMOS	Serial EEPROM
24C16	2048 × 8	O.D	5 V	NV CMOS	Serial EEPROM
24C32	4096 × 8	O.D	5 V	NV CMOS	Serial EEPROM
24C64	8 K × 8	O.D	5 V	NV CMOS	Serial EEPROM
24C128	16 K × 8	O.D	5 V	NV CMOS	Serial EEPROM
24C256	32 K × 8	O.D	5 V	NV CMOS	Serial EEPROM

*O.C-open collector, O.D-open drain, TS-Tristate

74S288 TTL PROM

The 74S288 is a 256 bit Schottky TTL PROM organised as 32×8 bits. Its logic diagram is shown in Fig. 11.14. It is available in 16-pin DIP and has one enable (\overline{G}) input terminal which controls the output state. When the device is enabled (\overline{G} LOW), the outputs ($O_0 - O_7$) represent the contents of the selected word by the address input. When disabled (\overline{G} HIGH), the outputs go to the OFF (high-impedance) state. This IC is available with LOWs in all locations. A HIGH may be programmed into any selected location by blowing off the titanium-tungsten fuse which requires 10.5 V for programming.

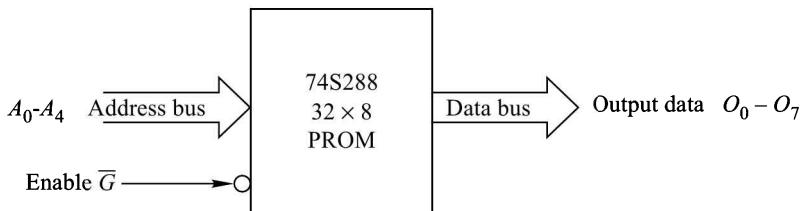


Fig. 11.14 Logic Diagram of 74S288 32 x 8 Schottky TTL PROM

27C010 OTP EPROM

Figure 11.15 shows the logic diagram of 27C010-1 Megabit OTP EPROM. It is a one time programmable CMOS read-only memory (OTP EPROM) organised as $128 \text{ K} \times 8$ bits. It has three control inputs: Chip Enable (\overline{C}), Output Enable (\overline{OE}), and Program Store (\overline{G}).

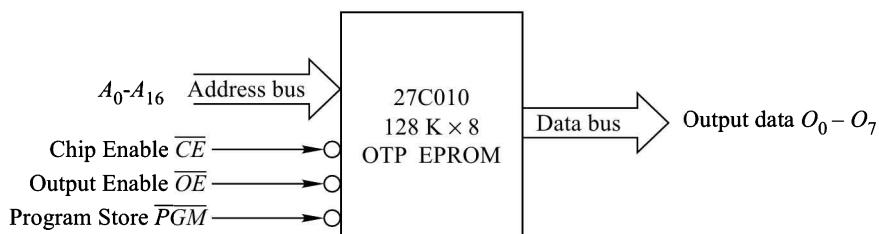


Fig. 11.15 Logic Diagram of 27C010 128 K x 8 OTP EPROM

The chip enable input (\overline{C}) is used for enabling the chip for read operation and the output enable input (\overline{OE}) for gating the data to the output pins. For programming, a LOW level pulse is applied at \overline{G} input with \overline{C} LOW.

27C64A UV EPROM/OTP EPROM

Figure 11.16 shows the logic diagram of the 27C64A 8 K \times 8 EPROM. It is available in two ranges UV EPROM (reprogrammable version) and OTP EPROM (one time programmable version). The reprogrammable version is having a transparent lid for erasing the chip when exposed to ultraviolet light, whereas the device for one time programmable (OTP) version is not having the transparent lid. There are three control inputs whose

operation is similar to that explained for the 27C010 device. This device is available with all the bits in '1' state. Data is stored by selectively programming '0's into the desired bit locations.

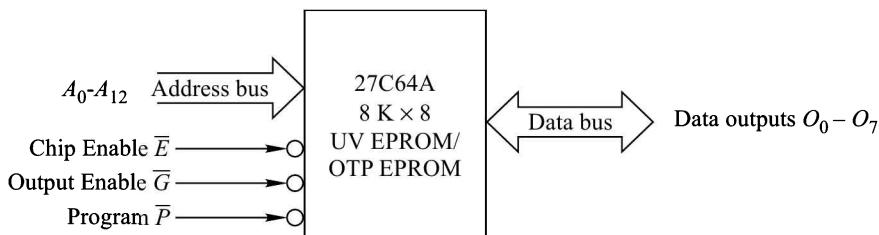


Fig. 11.16 *Logic Diagram of 27C64A UV EPROM/OTP EPROM*

28HC256 Parallel EEPROM

Figure 11.17 shows the logic diagram of 28HC256 high-speed parallel EEPROM organized as 32K \times 8 bits.

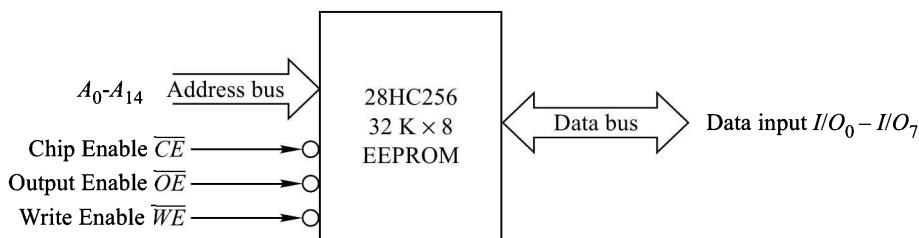


Fig. 11.17 *Logic Diagram of 28HC256 32 K \times 8 EEPROM*

It is accessed like a static RAM for the read or write cycle. This device contains a 64-byte page register to allow writing of upto 64 bytes simultaneously. During a writing cycle, the address and 1–64 bytes of data are internally latched in the device and the data and address buses are freed for other operations. The end of a write cycle is detected by $\overline{\text{DATA}}$ polling of I/O_7 . Once the end of a write cycle has been detected a new access for a read or write can begin.

Device Operation

Read When \overline{C} and \overline{W} inputs are LOW and \overline{OE} input is HIGH, the data stored in the addressed memory location are available on the outputs $I/O_0 - I/O_7$. The outputs are in the high-impedance state when either the chip is not selected ($\overline{C} = 1$) or output is disabled ($\overline{OE} = 1$).

Write A single byte or a page containing 64 bytes can be written in a single internal programming period. For write operation, \overline{C} and \overline{W} are required to be LOW and \overline{OE} HIGH. The address is latched on the falling edge of \overline{C} or \overline{W} whichever occurs last. The data is latched by the first rising edge of \overline{C} or \overline{W} . In the page write operation, the first byte written can be followed by 1 to 63 additional bytes. All the bytes during a

page write operation must reside on the same page. A page is defined by $A_6 - A_{14}$ address inputs. The A_0 to A_5 inputs are used to specify which bytes within the page are to be written.

During a byte or page write cycle, the complement of the last byte written will appear on the I/O_6 pin. This is used for DATA polling. Once the write cycle has been completed, valid data in true form is available on all outputs. In addition to DATA polling, there is another method of determining the end of a write cycle. During the write operation, successive attempts to read data from the device will result in I/O_6 toggling between '1' and '0'. Once the write cycle has completed, I/O_6 will stop toggling and valid data will be read.

It is possible to erase the entire contents of the device by using a 6-byte software code by the controlling device such as a microprocessor.

Serial EEPROM

Due to the unprecedented developments and applications of programmable ICs, a large number of consumer, telecommunication, medical, industrial and PC related products, etc. have become available which are small in size but may require a large non-volatile memory capable of read and write operations. The electrically erasable and programmable read-only memory (EEPROM) is the only type of memory which is non-volatile read and write memory. The parallel EEPROM discussed above requires a large number of pins for the input/output and addressing resulting in a large package size and high power consumption. Therefore, a need was felt to develop memories for large storage capacity which are non-volatile SRAM requiring smaller number of pins, smaller package size, and lower power consumption, etc. To meet all these requirements, *serial EEPROM* devices were developed. Table 11.5 gives some of the available serial EEPROMs.

24C01/24C02/24C04/24C08/24C16 Serial EEPROMs

All these serial memories are electrically erasable and programmable read-only memory (EEPROM) available in 8-pin package and organised as given in Table 11.5. Figure 11.18 shows their logic diagram. In all these devices, there are three address pins A_2 , A_1 , and A_0 which are used for device and page addressing as discussed below. The addressing mechanism is different in different devices. The serial data (SDA) pin is bi-directional for serial input/output data transfer. It is open-drain (O.D) driven and may be wire-ORed with any number of other open-drain or open-collector devices. The serial clock input (SCL) is used to clock data into EEPROM (write operation) at the positive edge and clock data out (read operation) at the negative edge.

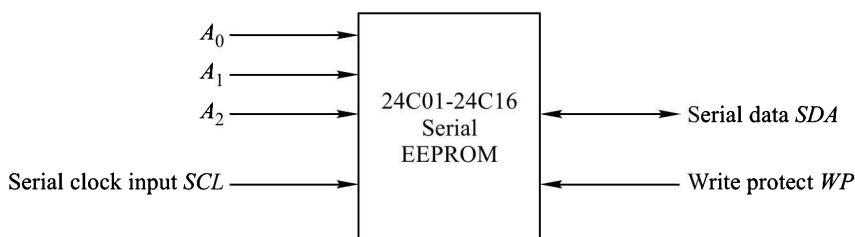


Fig. 11.18 *Logic Diagram of 24C01–24C16 Serial EEPROMs*

Device Addressing

All these serial EEPROM devices require an 8-bit device address word. The four most significant bits D_7 , D_6 , D_5 , D_4 are fixed as 1010 for all the devices and the next three bits D_3 , D_2 , D_1 correspond to A_2 , A_1 and A_0 .

respectively. These are used for device/page addressing. The least-significant bit is the read/write operation select bit. It is HIGH for read and LOW for write operation. Figure 11.19 gives the arrangement of bits of the addressing word. The read and write operations are controlled by the addressing device such as a microcontroller.

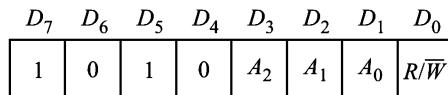


Fig. 11.19 *Serial EEPROMs Addressing Word*

The operation of A_2 , A_1 , A_0 pins is given in Table 11.6.

Table 11.6 *Serial EEPROM Device Addressing Operation*

Device	Status of pins		
	A_2	A_1	A_0
24C01/24C02	Hard wired and upto eight 24C01/24C02 devices may be addressed		
24C04	Hard wired and upto four 24C04 devices may be addressed	Memory page address bit P_0 To be left unconnected	
24C08	Hard wired for upto two 24C08 devices	Memory page address bits P_1 To be left unconnected	P_0
24C16	P_2	P_1 To be left unconnected	P_0

11.6 READ AND WRITE MEMORY

Many digital systems require memories in which it should be possible to write into, or read from, any memory location with the same speed. In such memories, the data stored at any location can be changed during the operation of the system. This type of memory is known as a read/write memory and is usually referred to as RAM (random-access memory).

The read-write memories (RWM)/random-access memories (RAM) are fabricated using bipolar devices or unipolar (MOS) devices. There are two types of RAMs. These are *static* RAM (SRAM) and *dynamic* RAM (DRAM). Bipolar RAMs are static, whereas the MOS RAMs can be static or dynamic. The basic storage cell of a static RAM is a bistable circuit, i.e., a latch, which simply consists of two cross-coupled inverters as shown in Fig. 7.3. A RAM is an array of these storage cells requiring as many FLIP-FLOPs as the bit storage capacity of the RAM, which is usually a large number. The storage cell of a DRAM is simply a capacitor, therefore, only MOS devices can be used for dynamic random-access memories. Since capacitors leak charge, therefore, the voltage stored in it gets reduced with time which requires periodic refreshing. In general, bipolar RAMs are faster than the MOS RAMs. With improvements in the MOS technology, it has become possible to make MOS RAMs with speeds (access time) comparable to those of bipolar RAMs.

11.6.1 Bipolar RAM Cell

The basic bipolar RAM storage cell is shown in Fig. 11.20. In this FLIP-FLOP, one transistor is ON and the other is OFF. When the OFF transistor is forced into the ON state by an external trigger pulse, the transistor that was initially ON turns OFF. Thus, it has two stable states which are used to store information in the form of logic 0 and 1. Special features are incorporated in this cell for addressing the cell, writing into it, and reading from it.

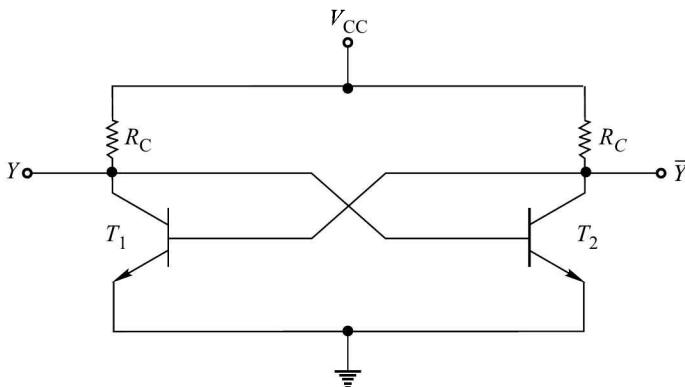


Fig. 11.20 Basic Bipolar RAM Storage Cell

Figure 11.21 illustrates a bipolar RAM cell with the required facilities incorporated. The FLIP-FLOP consists of the transistors T_1 and T_2 . These transistors are provided with additional emitters to incorporate the facility of addressing them. The remaining circuitry provides a mechanism for writing and reading data. Signals A_x and A_y are used to address the cell which are the outputs of the row-select and column-select address decoders, respectively. The cell is accessed for reading or writing when $A_x = A_y = 1$.

Let A_x , A_y , and W/\bar{R} inputs be at logic 0. The outputs of gates G_1 and G_2 will be 1, which make transistors T_3 and T_4 ON. Therefore, the diodes D_1 and D_2 are non-conducting. If, say, the state of the FLIP-FLOP is such that T_1 is ON and T_2 is OFF, then the emitter current will flow in E_x and E_y . A bias voltage of 0.5 V is applied through the resistor R_3 to emitter E_D . Therefore, E_D is more positive than E_x and E_y and hence E_D does not conduct. The transistors T_5 and T_6 are also OFF, and the data output terminal is at logic 1 and will remain in this state, independent of the state of the FLIP-FLOP.

Now, let the cell be addressed by raising both A_x and A_y to logic 1. Then, the currents through E_x and E_y will be diverted to E_D . A component of this current will flow into the base of T_5 and the data output terminal will assume the same logic level as present at the collector of T_1 . Thus, with the W/\bar{R} at logic 0, the operation of addressing the cell provides a reading of the cell.

Now, let $A_x = A_y = 1$ and W/\bar{R} input be at logic 1. If the data input is at logic 1, the output of G_1 will be 1 and the output of G_2 will be 0. Therefore, T_3 is ON and T_4 is OFF. The voltage at the collector of T_4 rises, D_2 conducts and raises the voltage at \bar{E}_D . Hence, irrespective of the original state of the FLIP-FLOP, T_2 cannot conduct. Hence, the logic level at the collector of T_2 will become the logic level of the data input. If the cell is not addressed, E_D and \bar{E}_D will not be carrying current and the FLOP-FLOP will not respond to the writing operation. When a new data is written, the earlier stored data, if any, gets replaced with the new data.

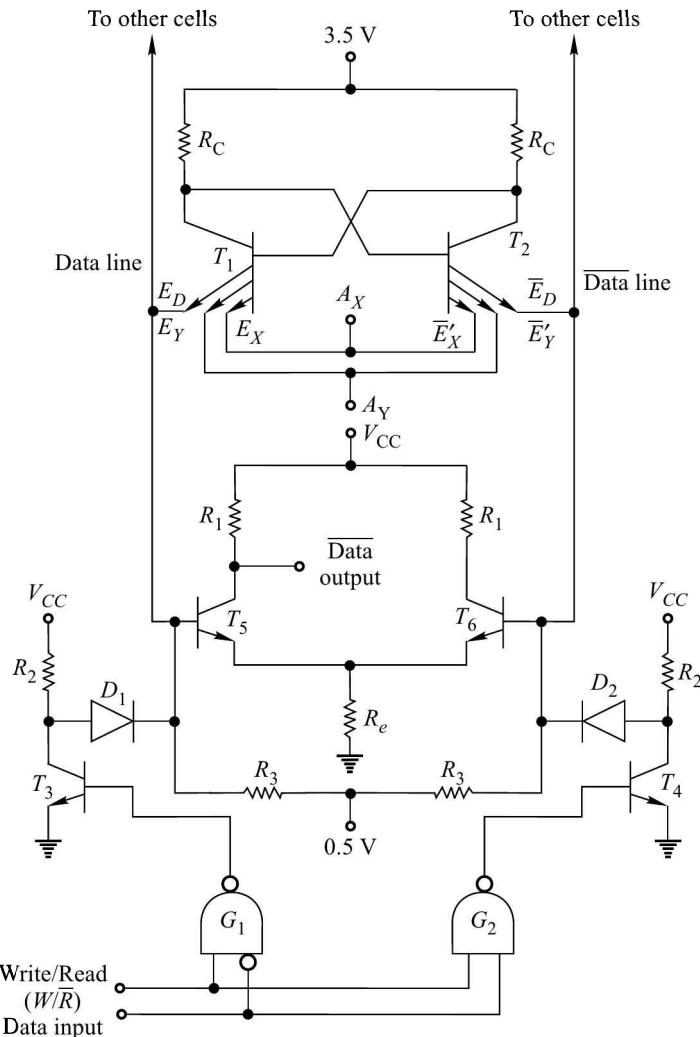


Fig. 11.21 Bipolar RAM Cell

11.6.2 MOSFET Static RAM (SRAM) Cell

A CMOS SRAM memory cell is shown in Fig. 11.22. Each bit in an SRAM is stored on four transistors, two NMOS and two PMOS, that form two cross-coupled inverters. Two additional transistors T_5 and T_6 serve to control the access to a storage cell for read and write operations. Access to the cell is enabled by the word line (WL) which controls the two transistors T_5 and T_6 which, in turn control whether the cell should be connected to the bit lines BL and \bar{BL} . These bit lines are used to transfer data for both read and write operations.

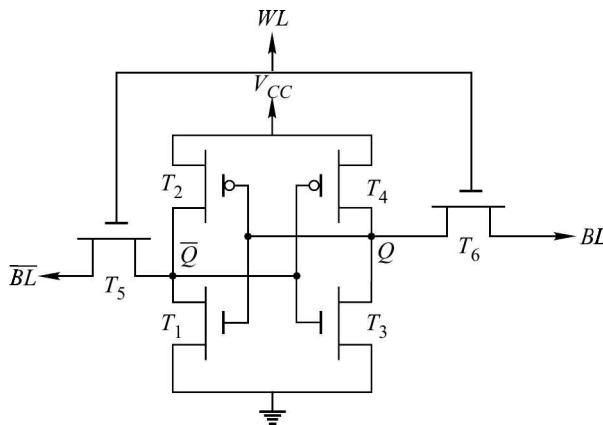


Fig. 11.22 A Six-Transistor CMOS SRAM Cell

Read Operation

Assume that the content of the memory is $Q = 1$. The read cycle is started by precharging both the bit lines BL and \bar{BL} to logic 1, then asserting the word line $WL = 1$ enables both the access transistors T_5 and T_6 . The values stored in Q and \bar{Q} are now transferred to the bit lines by leaving BL at its precharged value and discharging \bar{BL} through the transistors T_1 and T_5 to logic 0. On the BL side, the transistors T_4 and T_6 pull the bit line to V_{CC} , i.e., logic 1. If the content of memory is $Q = 0$, the opposite will happen and \bar{BL} would be pulled towards 1 and BL towards 0.

Write Operation

For writing into the cell, the bit to be stored is applied at BL and its inverse at \bar{BL} . When the word line WL is asserted, the value to be stored is latched. The new bit replaces the earlier bit stored.

11.6.3 Asynchronous SRAM

Figure 11.23 shows a functional block diagram of an $8\text{ K} \times 8$ CMOS static RAM. It has a single decoder circuit and three control inputs: chip enable (\bar{CE}), output enable (\bar{OE}) and write enable (\bar{WE}). All the three control inputs are active-low. Its truth table is given in Table 11.7.

Table 11.7 Truth Table of Asynchronous SRAM of Fig. 11.23

Mode	\bar{WE}	\bar{CE}	\bar{OE}	I/O operation
Not selected (Power-down)	X	H	X	High-Z
Output disabled	H	L	H	High-Z
Read	H	L	L	D_{out}
Write	L	L	X	D_{in}

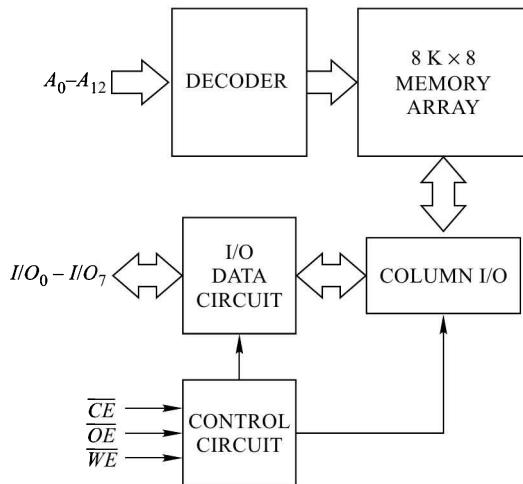


Fig. 11.23 Functional Block Diagram of an Asynchronous SRAM

Figure 11.24 shows the functional block diagram of another asynchronous SRAM which has two-dimensional decoding mechanism. It uses two decoders.

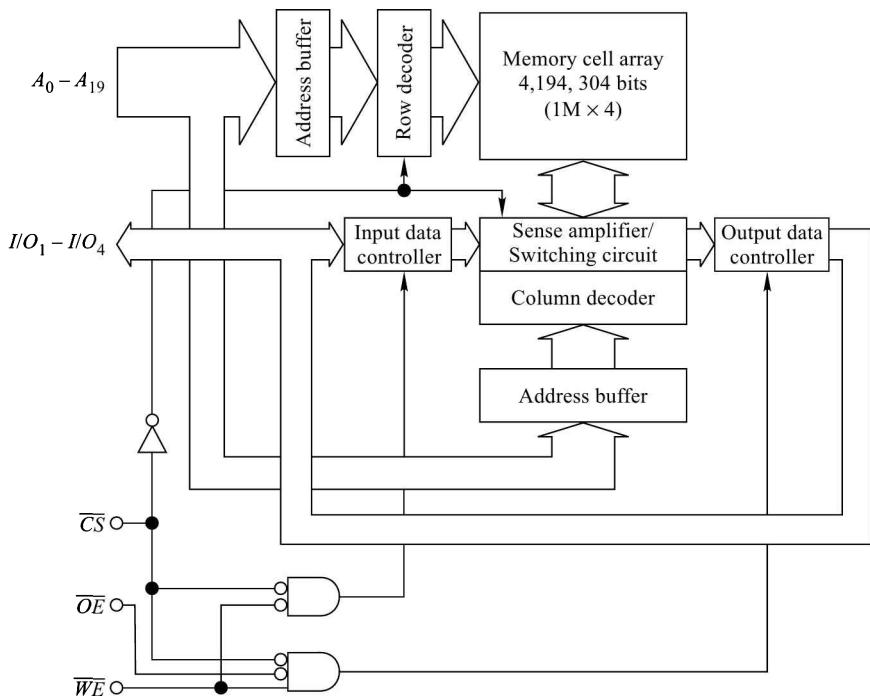


Fig. 11.24 Functional Block Diagram of an Asynchronous SRAM with Two Dimensional Decoding

The first half of the address bus A_0 – A_9 are decoded by the row decoder and the other half A_{10} – A_{19} are decoded by the column decoder. Using two decoders save hardware requirement for decoders in comparison to using one decoder. The truth table of this SRAM is same as that given in Table 11.7.

11.6.4 Synchronous SRAM

A synchronous SRAM uses the system clock to synchronise its operation with the controlling device, such as a microprocessor, for faster operation. Its memory cell, memory array, address decoder, and read (\overline{OE})/write (\overline{WE}) enable inputs are similar to an asynchronous SRAM, but the various registers used operate in synchronism with the system clock. The address, read/write and chip enable (or select), and the input data are all latched into their respective registers on an active edge of the clock pulse. Figure 11.25 shows the block diagram of a synchronous SRAM with burst feature. Here, the address and the data bus are shown as single line with a slash and the number of lines mentioned by its side. This is a most commonly used convention in digital systems.

There are two basic types of synchronous SRAMs depending upon the way the output data is obtained from the SRAM. These are:

- Flow-through SRAM
- Pipe-lined SRAM

The only difference between these two types of SRAMs is the presence or absence of the Data output register (shown as shaded in Fig. 11.25). The *flow-through* synchronous SRAM does not have a Data output register, therefore, the output data flow asynchronously to the data I/O lines through the output buffers. On the other hand, the data output register is present in the pipelined synchronous SRAM so that the output data are synchronously placed on the data I/O lines.

The *Burst feature* is generally incorporated in a synchronous SRAM. It allows the memory read from or write at upto four locations using a single address. A 2-bit binary counter alongwith the two EX-OR gates is used to achieve this feature in the synchronous SRAM chip. When the external address, 16-bit in this case, $A_{15}A_{14}\dots A_2A_1A_0$ is latched into the address register, the lowest significant two bits A_1 and A_0 are applied to the EX-OR gates, as shown in the figure. The outputs of the EX-OR gates are A'_1 and A'_0 . These $A'_1A'_0$ bits are used to replace A_1A_0 bits when the address (16-bit) gets applied to the address decoder.

Let the A_1A_0 bits of the external 16-bit address be 00. The counter outputs Q_1Q_0 will be 00, 01, 10, 11 on the successive clock pulses, which makes $A'_1A'_0$ to be 00, 01, 10, 11 on every successive clock pulse. The address used for accessing the memory array is $A_{15}A_{14}\dots A_2A'_1A'_0$. Thus, four memory locations are accessed for read/write using a single external address. This increases the speed of the memory.

There are various other features incorporated in synchronous SRAM to make them suitable for different applications. Various types of such SRAMs are:

- Late Write Synchronous SRAM
- No Wait State Bus Synchronous SRAM
- DDR Synchronous SRAM
- Quad Synchronous SRAM

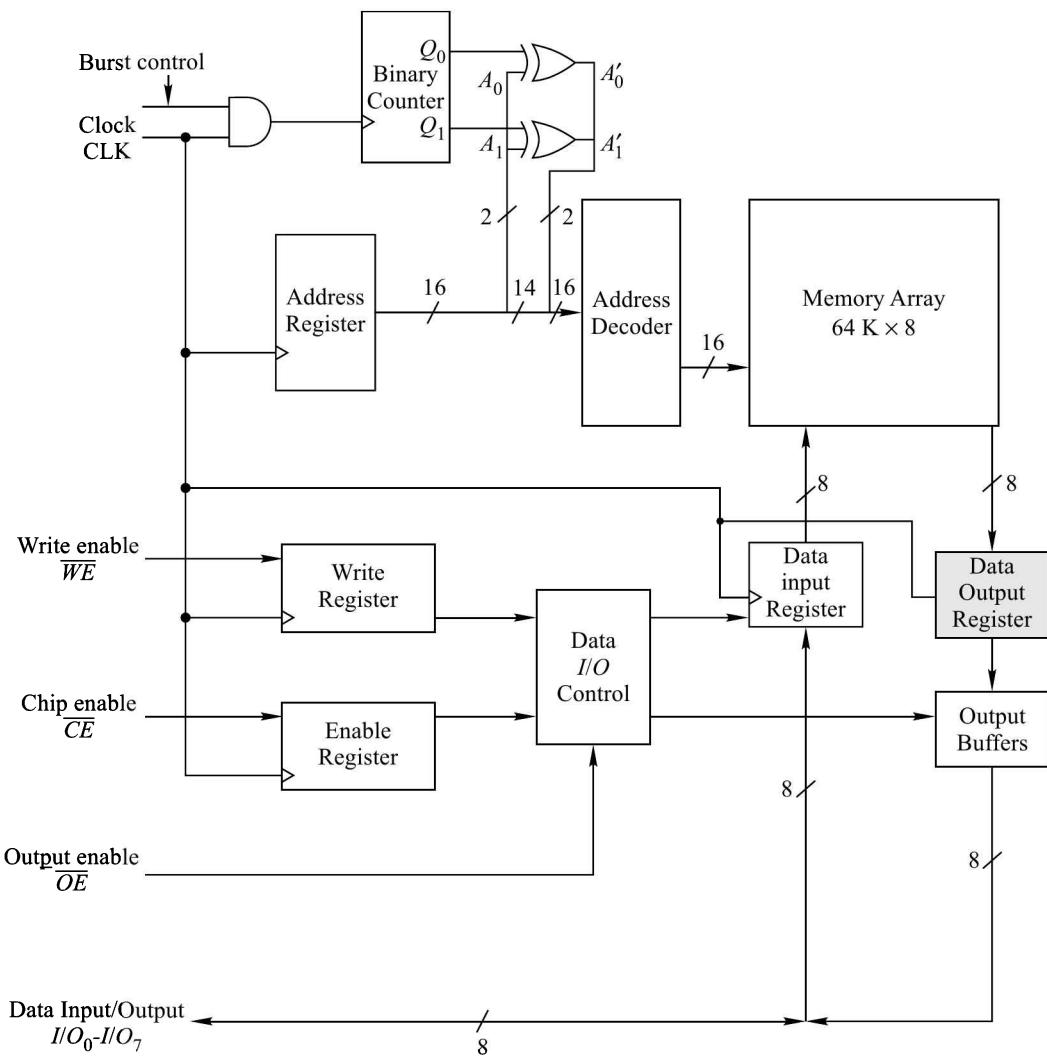


Fig. 11.25 **Block Diagram of a Synchronous SRAM with Burst Feature**

Late Write Synchronous SRAM

In a synchronous SRAM, when the operation repeats from read to write to read to write ..., wait cycles are there which decreases its speed of operation. To reduce the number of wait cycles, Late Write synchronous SRAM has been developed which has a ‘Write Address Register’ which is not present in a standard synchronous SRAM. This allows an operation to write data in a cycle next to the one in which an address input to a synchronous memory is applied. The write address register is used to save an address to the write address register once even if an address for read operation is input in a cycle in which written data is to be input. The

operation can be completed by using the address of the write address register after the read operation has been completed.

No Wait State Bus Synchronous SRAM

Similar to the Late write synchronous SRAM, zero wait state synchronous SRAM is available in, which there is no wait state. It has a ‘Write Address Register’, like the Late Write SRAM. Its flow-through version has a ‘Write Address Register’ of one address and the pipeline version has a ‘Write Address Register’ of two addresses. The operation of this ‘Write Address Register’ can completely eliminate wait cycles when an operation is performed from read → write → read, and so on. This device has ZBT as the trademark of IDT, and ZEROSB is the trademark of NEC.

Double Data Rate (DDR) Synchronous SRAM

A DDR enables data to be read or written twice every clock. This type of SRAM realises a much higher data transfer rate than conventional synchronous SRAM. Data *I/O* at 500 MHz is obtained with an external input clock of 250 MHz. Two versions of DDR synchronous SRAM are

- Common *I/O* version
- Separate *I/O* version

In the case of common *I/O* version, there is common bus for input/output, whereas in separate *I/O* version, there are separate buses for input and output.

Quad Synchronous SRAM

Quad synchronous SRAM enables data to be read or written four times to that of conventional synchronous SRAM. It has separate read and write ports with concurrent read and write operation, and DDR interface for read and write operation.

11.6.5 MOSFET Dynamic RAM (DRAM) Cell

The earlier DRAMs were made using 3-transistor cell, which were later replaced by 1-transistor cell. Figure 11.26 shows a 1-transistor DRAM cell. In this cell, the data bit is stored in a small capacitor rather than in a latch used for SRAM cell. Also, in this cell, only one transistor and a capacitor are required per bit, compared to six transistors in SRAM (Fig. 11.22). This allows DRAM to have very high density in comparison to SRAM. The main disadvantage in a DRAM cell is that since the charge is stored in a capacitor, which can not hold it over an extended period of time. Therefore, the stored bit can not remain unless the charge is replenished or refreshed periodically. This requires additional circuitry.

Figure 11.27 shows a DRAM cell alongwith the simplified circuitry for read, write, and refresh operations.

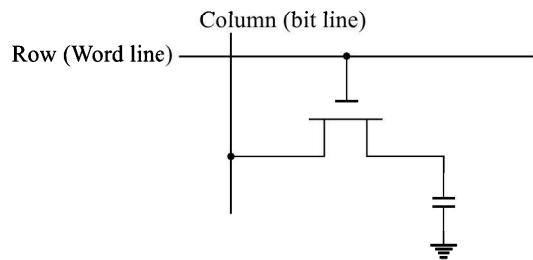


Fig. 11.26 A 1-Transistor DRAM Cell

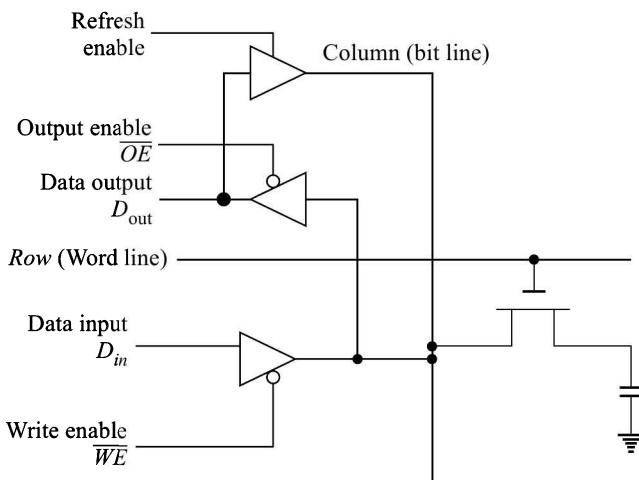


Fig. 11.27 A DRAM Cell with Read, Write, and Refresh Circuitry

Read Operation

For reading or writing operation, the word line (*Row*) is to be selected which switches ON the transistor. The output enable \overline{OE} LOW will enable the output buffer, making its output same as the bit line which is at the same logic level as the voltage on the capacitor. Thus, the output is at logic 1 corresponding to the capacitor charged and logic 0 corresponding to discharged capacitor.

Write Operation

With the *Row* line selected, the write enable \overline{WE} LOW allows writing into the cell. If the D_{in} bit is 1, the capacitor gets charged to logic 1 through the ON transistor, whereas, if D_{in} bit is 0, the capacitor gets discharged through the ON transistor to the logic 0. When the \overline{WE} is made HIGH, the charge on the capacitor remains trapped on the capacitor (1 or 0).

11.6.6 Asynchronous DRAM

DRAM Organisation

Figure 11.28 shows the functional block diagram of a $512\text{ K} \times 8$ DRAM. Its address bus width is 19. To reduce the number of pins in the chip for addressing and complexity of a decoder of 19-to-524288 size, the address is divided in two halves and two decoders are used. The lower order ten bits ($A_0 - A_9$) are applied to the row decoder and the higher order nine bits are applied to the column decoder. Also, the address is multiplexed. The number of address inputs available is ten, $A_0 - A_9$. The address is applied to the DRAM address bus in two parts using a multiplexer arrangement. The first ten address bits ($A_0 - A_9$) are entered as row address and latter nine address bits ($A_0 - A_8$) are entered as column address. The row address is latched by the Row Address Strobe (RAS) and the column addressed is latched by the Column Address Strobe (CAS).

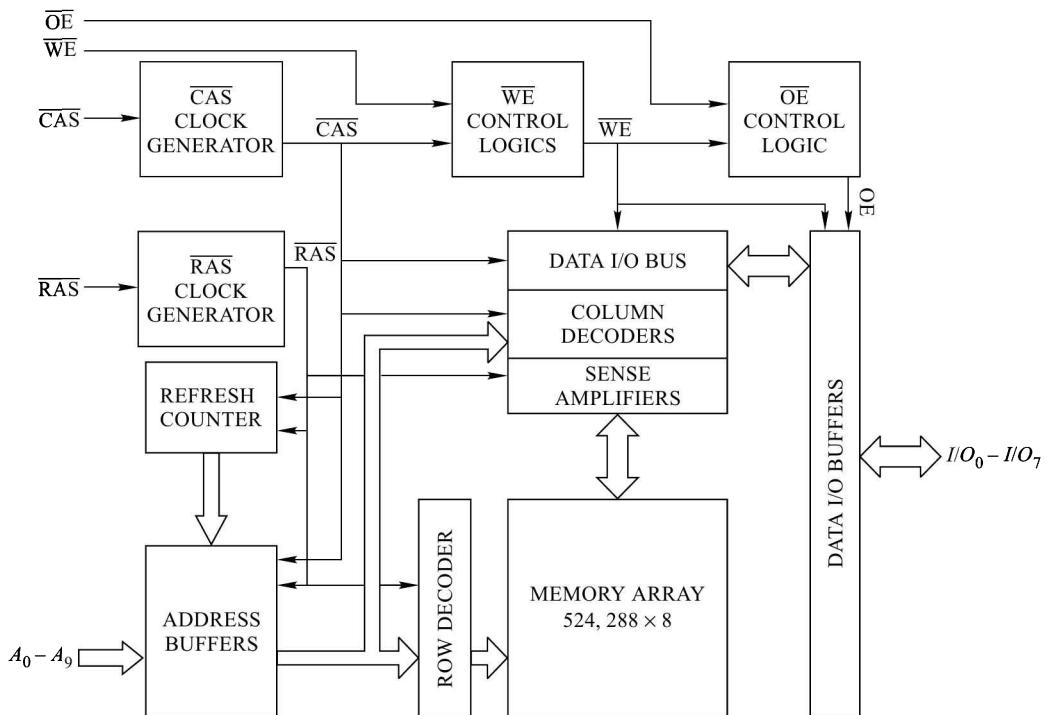


Fig. 11.28 Functional Block Diagram of a 512 K x 8 DRAM

A read cycle is initiated by falling edge of \overline{CAS} or \overline{OE} , whichever occurs last, while holding \overline{WE} HIGH. Similarly, a write cycle is initiated by falling edge of \overline{CAS} or \overline{WE} , whichever occurs last.

To retain data, all rows in the memory array are refreshed consecutively each refresh period. The refresh period is normally of 16 ms. Since there are 10 rows in 512 K x 8 DRAM, therefore, 1024 refresh cycles are required in each 16 ms period. All the 1024 row addresses ($A_0 - A_9$) with \overline{RAS} LOW are clocked atleast once every 16 ms.

Fast Page Mode DRAM (FPM DRAM)

In a DRAM, when a row address is applied, all the columns in a given row can be accessed by applying successive column addresses at random with the row address applied only once. A page in memory is referred to all of the column addresses contained within one row address. In a fast page mode operation, the \overline{CAS} must remain LOW until the valid data from a given address are accepted (latched) by the external system, such as a microprocessor. When \overline{CAS} goes HIGH, the data outputs are disabled. The next column address must not occur until the data from the current column address are transferred to the external device.

Extended Data Out DRAM (EDO DRAM)

In an EDO DRAM, the data outputs are not disabled when the \overline{CAS} signal goes from LOW to HIGH. Therefore, the valid data are available from the current address until the next \overline{CAS} signal goes LOW.

Therefore, the next column address can be accessed before the external system accepts the valid data from the current column address. Because of this type of operation, the access time of EDO DRAM is reduced in comparison to FPM DRAM.

Burst EDO DRAM (BEDO DRAM)

BEDO DRAM is a modified EDO which was developed by adding an address counter to provide for burst memory operation. Similar to synchronous burst SRAM, it can process four memory addresses in one burst. The burst operation takes place in synchronism with the controlling device, such as a microprocessor. This device is faster than EDO but it never became popular because of the development of synchronous DRAM at about the same time which is much superior to BEDO RAM.

11.6.7 Synchronous DRAM (SDRAM)

The operation of the synchronous DRAM is synchronised with the system clock. The SDRAMs are extensively used in computers and other microprocessor based systems requiring large capacity memory. Similar to synchronous SRAM, synchronous DRAMs are available with double data rate (DDR) transfer. The DDR SDRAMs are widely used in various digital systems and computers.

11.6.8 RAM ICs

A large variety of SRAM and DRAM ICs are available from various manufacturers. CMOS technology is the most preferred technology for all the latest SRAM and DRAM ICs. Some of the available SRAM ICs are given in Table 11.8 and DRAM ICs are given in Table 11.9. There complete technical and operational details can be obtained from the websites of the manufacturers/vendors.

Table 11.8 Available CMOS SRAM ICs

IC No.	Asynchronous		Speed ns	Characteristics
	Organisation No. of bits	Power supply voltage		
61C64AL	8 K × 8	5 V	10	High speed TTL compatible interface level
61C3216AL	32 K × 16	5 V	12	High speed TTL compatible interface level
444001	4 M × 1	5 V	10, 11, 12	High speed
444004	1 M × 4	5 V	8, 10, 12	High speed
62C256AL	32 K × 8	5 V	25, 45	LOW power TTL compatible inputs/outputs
6C1008	128 K × 8	2.7–5.5 V	55	LOW power
6C2008A	256 K × 8	2.7–5.5 V	55	LOW power

IC No.	Synchronous		Speed MHz	Type
	Organisation No. of bits	Power supply voltage		
61LP6432A	64 K × 32	3.3 V	133	Pipeline Burst
61LPS12832A	128 K × 32	3.3 V	250	Pipeline Burst
61LF6432A	64 K × 32	3.3 V	90	Flow through Burst
61LF6436A	64 K × 36	3.3 V	90	Flow through Burst
61NLP6432A	64 K × 32	3.3 V	250, 200	Pipeline ‘No Wait’ state
61NLP6436A	64 K × 36	3.3 V	250, 200	Pipeline ‘No Wait’ state
61NLP128 × 18 A	128 K × 18	3.3 V	250, 200	Pipeline ‘No Wait’ state
61DDB 41M36	1 M × 36	1.8 V	250	DDR II common I/O
61DDB42M18	2 M × 18	1.8 V	250	DDR II common I/O
61QDB41M36	1 M × 36	1.8 V	250	Quad Separate input, output
61QDB42M18	2 M × 18	1.8 V	250	Quad Separate input, output

Table 11.9 Available CMOS DRAM ICs

IC No.	Asynchronous		Read/Write cycle time (min.) ns	Type
	Organisation No. of bits	Power supply voltage		
41C85125	512 K × 8	5 V	60	FPM TTL compatible
41LV85125	512 K × 8	3.3 V	60	FPM TTL compatible
41LV16257B	256 K × 16	3.3 V	60	FPM TTL compatible
41C85120	512 K × 8	5 V	60	EDO TTL compatible
41LV85120	512 K × 8	3.3 V	60	EDO TTL compatible
41LV16256B	256 K × 16	3.3 V	60	EDO TTL compatible

IC No.	Synchronous		Speed MHz	Type
	Organisation No. of bits	Power supply voltage		
42S16100C1	512 K × 16	3.3 V	200	SDRAM
43R16800B	8 M × 16	2.5 V	200	DDR SDRAM
43DR32800A	8 M × 32	1.8 V	800	DDR II SDRAM
43DR32801A	8 M × 32	1.8 V	800	DDR II SDRAM

11.7 FLASH MEMORY

Flash memory is non-volatile memory that can be electrically erased and reprogrammed. It is a specific type of EEPROM that is erased and programmed, in-circuit, in large blocks in contrast to EEPROM which is erased and reprogrammed at the byte level. Since the flash memory can be written to in block size rather than byte, it is easier to update it. On the other hand, it is not useful as RAM because RAM needs to be addressable

at the byte and not the block level. However, it is also sometimes referred to as ‘non-volatile RAM’. This type of memory has been named as ‘flash memory’ because a large block of memory could be erased at one time, i.e., in a single action or ‘flash’.

11.7.1 Flash Memory Cell

The flash memory cell consists of one transistor with a floating gate similar to an EPROM cell. However, there is a difference in the geometry and technology between flash devices and EPROM devices. The gate oxide between the silicon and the floating gate is thinner for flash technology and the source and drain diffusion are also different. Other flash cell concepts are based upon EEPROM technology. Although the flash cells are larger than the conventional one-transistor EPROM cell, but are far smaller than the conventional two-transistor EPPOM cell. The flash memory chip size is thus considerably less than the EEPROM, due to which flash memory’s density is higher and its cost per bit is lower than EEPROM.

11.7.2 Flash Memory Architecture

There are two common flash memory architectures: NOR and NAND. The NOR architecture is the most popular flash memory architecture. It is commonly used in EPROM and EEPROM designs also. Although the NAND structure is considerably more compact, its main drawback is that when a cell is read, the sense amplifier sees a weaker signal, than that on a NOR configuration, since several transistors are in series in NAND structure. The transistors are in parallel in NOR structure (See Figs. 4.29 and 4.30).

11.7.3 Flash Memory ICs

The flash memory ICs are available in a variety of configurations and architectures to address various different requirements. The devices are available in different sector sizes ranging from 64–256 bytes, making it suitable for code and data storage. The flash memories are available from 256 K bits to 128 M bits in various sizes and organisations and operate with single voltage supply of 5V, 3 V, 2.7 V, and 1.8 V, etc. Similar to EEPROM devices, flash memories are also available in parallel as well as serial types.

Some of the available parallel flash memory ICs are given in Table 11.10.

Table 11.10 *Available CMOS Parallel Flash Memory ICs*

IC No.	Organisation No. of bits	Power supply voltage	No. of sectors	Sector size bytes
29C256	32 K × 8	5 V	512	64
29LV256	32 K × 8	3 V	512	64
29C512	64 K × 8	5 V	512	128
29LV512	64 K × 8	3 V	512	128
29C010A	128 K × 8	5 V	1024	128
29LV010A	128 K × 8	3 V	1024	128
29C1024	64 K × 16	5 V	512	128 words
29LV1024	64 K × 16	3 V	512	128 words

The devices listed in Table 11.10 are having large memory arrays broken up into small individually reprogrammable sectors. For example, 29C010A is divided into 1024 sectors of 128 bytes each. Each sector's contents may be altered independently and no previous erase is required. It takes about 10–20 ms for altering the data of a sector. In case of large-sectored devices or whole chip flash devices, the write time required may extend to several minutes. Therefore, when only a small portion of the total memory is required to be altered, the small sector approach saves considerable time.

The programming method requires only nanoamp of high voltage (15 V–20 V) programming current, allowing the use of an on chip charge pump to generate the necessary programming voltages.

A number of serial flash memory devices are also available with varying sizes. For example, AT25DF021 is a 2-M bit 2.3 or 2.7 V, 70 MHz serial flash memory. It can be byte or page programmable and block (4 KB, 32 KB, and 64 KB) and chip erasable.

The detailed technical and operational specifications of various parallel and serial flash memories, can be obtained from the manufacturers' catalogues or their websites.

11.8 CONTENT ADDRESSABLE MEMORY

The content addressable memory (CAM) is a special purpose random access memory device that can be accessed by searching for data content. For this purpose, it is addressed by associating the input data, referred to as *key*, simultaneously with all the stored words and produces output signals to indicate the match conditions between the key and the stored words. This operation is referred to as *association* or *interrogation* and this type of memory is also known as *associative memory*.

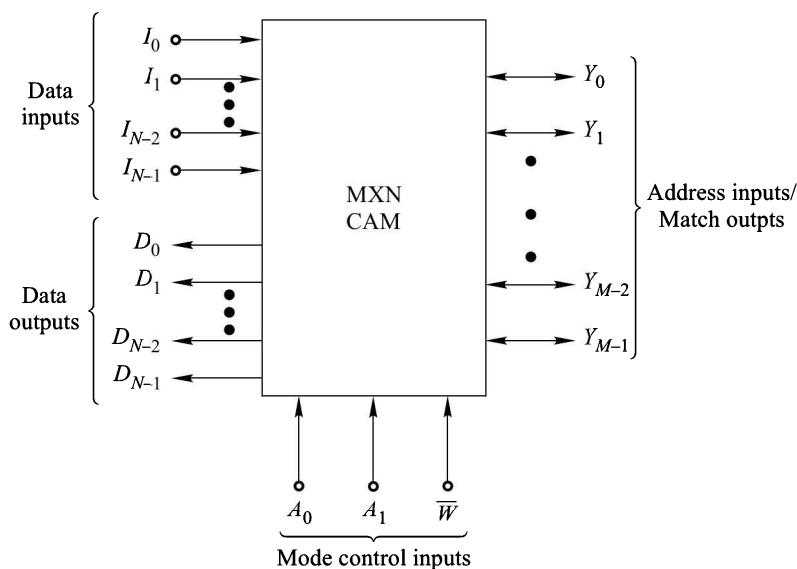
After identifying the locations whose contents match the key, read or write operations can be performed to these locations. The key to be used may either consist of the entire data word or only some specific bits of the data word, i.e. the other bits can be masked.

A CAM differs from the conventional memory organization in that the addressing of a location in the latter has no relation to the memory content. A CAM has the ability to search out or interrogate stored data on the basis of its contents and, therefore, can be a powerful asset in many applications. For example, consider a list containing the names of persons, their ages, professions, and nationalities stored in a CAM. If one is interested in finding out engineers in the list, the CAM is able to check every memory location simultaneously by using the coded form for engineer as the key. On the other hand, if it is required to find the engineers of Indian nationality, the key will consist of the combination of the codes corresponding to engineer and Indian nationality. All the memory locations with engineers of Indian nationality will be identified and the remaining data (name and age) can then be retrieved by using the read operation. To do the same search process with a conventional memory, each memory word is to be read out and compared with the key. This search is a serial process and hence time consuming. Thus, CAMs are better suited for information retrieval than the conventional memories.

CAMs are manufactured using MOS, CMOS, or bipolar technologies. The most popular CAMs use ECL circuitry because of its high speed operation.

11.8.1 Operation of CAM

A CAM can perform three basic operations: read, write and associate. Figure 11.29 shows a block diagram of a CAM. Its storage capacity is $M \times N$ bits and is organized as M words of N bits each. It has N data input and N data output lines (one line for each bit of a word). The data input lines I_0 through I_{N-1} are used to input

Fig. 11.29 **Block Diagram of a CAM**

data to be written into the memory and for key word in case of associate operation. Data are read out of the CAM at the data output lines D_0 through D_{N-1} .

The Y lines (Y_0 through Y_{M-1}) are bidirectional. During a read or write operation, these lines are used to select the storage location. There is one address input line for each word in the CAM. For example, Y_0 is the address line for memory location 0, Y_1 , for memory location 1 and so on. Notice that linear selection addressing is used in CAMs rather than coincident selection addressing.

The Y lines serve as match output lines one for each memory location, when an association operation is performed. For example, if the keyword matches with the word stored in memory locations 5 and 8, lines Y_5 and Y_8 will become HIGH to indicate the match condition.

The mode control inputs are used to select the required operation. The read and write operations are performed in a manner similar to that used for RAM. However, during the write operation, the input data also appear at the data outputs. The reading of the data is non-destructive. The internal architecture of a typical 8×2 ECL CAM is shown in Fig. 11.30. It has eight 2-bit storage locations M_0 through M_7 , each consisting of two parts, one for the higher order bit (M_{01} through M_{71}) and the other, for the lower order bit (M_{00} through M_{70}). The detailed circuitry of one of the locations M_0 is shown in the figure and all other locations have similar circuitry.

Each memory location consists of two D -type FLIP-FLOPs, one for each bit, and some logic gates to control the function of the CAM for read, write, and associate operations. The outputs Y_0 through Y_7 and the data outputs D_1 and D_0 are produced using wired-OR connections as shown in Fig. 11.30. The operation of this CAM is summarized in Table 11.11.

The word length and/or word size can be expanded by suitably connecting the available CAM chips in a manner similar to the one used for RAMs and ROMs expansion.

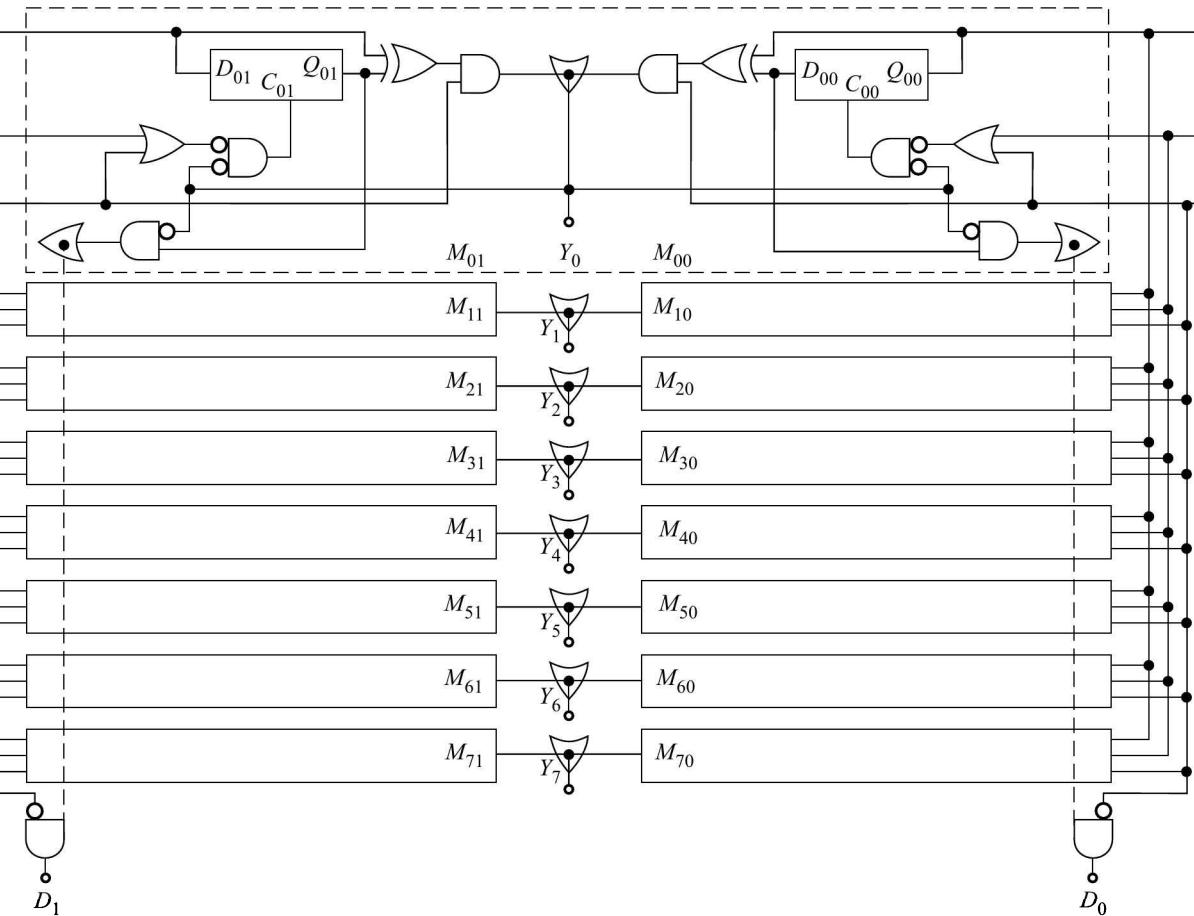


Fig. 11.30 Internal Architecture of an ECL 8×2 CAM

Table 11.11 Operation of the 8×2 CAM

Operation	Control inputs			Data inputs		Data outputs		Y_n	Value
	A_1	A_0	\bar{W}	I_1	I_0	D_1	D_0		
Associate	1	1	\times	1/0	1/0	0	0	Output	1 = Mismatch 0 = Match
Associate (higher bit masked)	0	1	1	\times	1/0	D_1	0	Output	1 = Lower bit mismatch 0 = Lower bit match
Associate (lower bit masked)	1	0	1	1/0	\times	0	D_0	Output	1 = Higher bit mismatch 0 = Higher bit match
Read	0	0	1	\times	\times	D_1	D_0	Input	0 = Selected address
Write	0	0	0	1/0	1/0	I_1	I_0	Input	0 = Selected address
	0	1	0	1/0	1/0	I_1	0	Output	1 = Lower bit mismatch 0 = Lower bit match
Associate and write at match addresses	1	0	0	1/0	1/0	0	I_0	Output	1 = Higher bit mismatch 0 = Higher bit match

Example 11.6

Consider the CAM of Fig. 11.30. Assume that it contains data given in Table 11.12

Table 11.12 Contents of CAM

Memory location	Content
0	00
1	01
2	00
3	11
4	10
5	01
6	11
7	01

What is the word at data outputs D_1 and D_0 for each of the following conditions? Also find if there is any change in the memory contents.

(a) $A_1 A_0 = 00, \overline{W} = 1$

$$Y_7 Y_6 Y_5 Y_4 Y_3 Y_2 Y_1 Y_0 = 11011111$$

(b) $A_1 A_0 = 00, \overline{W} = 1$

$$Y_7 Y_6 Y_5 Y_4 Y_3 Y_2 Y_1 Y_0 = 11001111$$

(c) $A_1 A_0 = 00, \overline{W} = 0$

$$Y_7 Y_6 Y_5 Y_4 Y_3 Y_2 Y_1 Y_0 = 11110111$$

$$I_1 I_0 = 00$$

(d) $A_1 A_0 = 00, \overline{W} = 0$

$$Y_7 Y_6 Y_5 Y_4 Y_3 Y_2 Y_1 Y_0 = 11010111$$

$$I_1 I_0 = 10$$

Solution

(a) Since $Y_5 = 0$, the memory location 5 is selected for read out, i.e.

$$D_1 D_0 = 01$$

The memory contents do not change.

(b) Since $Y_5 = Y_4 = 0$, the memory locations 4 and 5 are selected for read out. The output is obtained by ORing the contents of these locations, i.e.

$$D_1 D_0 = 11$$

The memory contents do not change.

(c) Since $Y_3 = 0$, write operation is performed in memory location 3. The input data is stored in this location and also appears at the output.

$$D_1 D_0 = 00$$

Contents of memory location 3 = 00.

(d) In this case, the memory locations 3 and 5 are selected for writing since $Y_3 = Y_5 = 0$. The contents of these locations will become 10 and also $D_1 D_0 = 10$.

Example 11.7

Assume that the CAM of Fig. 11.30 contains the data given in Table 11.12.

For each of the following conditions, find the Y outputs. Also find if there is any change in the memory contents.

(a) $A_1 A_0 = 11, I_1 I_0 = 01$

(b) $A_1 A_0 = 01, \overline{W} = 1, I_1 I_0 = 01$

(c) $A_1 A_0 = 10, \overline{W} = 1, I_1 I_0 = 01$

(d) $A_1 A_0 = 01, \overline{W} = 0, I_1 I_0 = 01$

(e) $A_1 A_0 = 10, \overline{W} = 0, I_1 I_0 = 01$

Solution

(a) The association operation is performed with keyword 01. The memory locations 1, 5, and 7 match the keyword giving out logic 0 at the corresponding Y outputs. Therefore,

$$Y_7 Y_6 Y_5 Y_4 Y_3 Y_2 Y_1 Y_0 = 01011101$$

- (b) In this case, the association operation is performed between the lower bit of the key with the lower bits of the stored words. Therefore,

$$Y_7 Y_6 Y_5 Y_4 Y_3 Y_2 Y_1 Y_0 = 00010101$$

- (c) In this case, the association operation is performed between the higher bit of the key with the higher bits of the stored words. Therefore,

$$Y_7 Y_6 Y_5 Y_4 Y_3 Y_2 Y_1 Y_0 = 01011000$$

- (d) The association operation is performed between the lower bit of the key and the lower bits of the stored words. The Y outputs are

$$Y_7 Y_6 Y_5 Y_4 Y_3 Y_2 Y_1 Y_0 = 00010101$$

The higher bit I_1 is stored in the higher bit positions of the memory locations 7, 6, 5, 3, and 1. Therefore, the contents of the memory locations will be as given in Table 11.13.

Table 11.13

Memory location	Contents
0	00
1	01
2	00
3	01
4	10
5	01
6	01
7	01

- (e) In this case, the association operation is performed between the higher bit of the key and the higher bits of the stored words. The Y outputs are

$$Y_7 Y_6 Y_5 Y_4 Y_3 Y_2 Y_1 Y_0 = 01011000$$

The lower bit I_0 is stored in the lower bit positions of the memory locations 7, 5, 2, 1, and 0. The new memory contents are given in Table 11.14.

Table 11.14

Memory location	Contents
0	01
1	01
2	01
3	11
4	10
5	01
6	11
7	01

11.9 FIRST-IN, FIRST-OUT MEMORY (FIFO)

A FIFO memory is a storage device in which data is read out from its memory array (SRAM) in the same order in which they were written into the memory. The first word written into the memory block is the first word that is read out of the memory block. Because of this type of operation in which the first word written in is the first word read out, it is referred to as the *first-in, first-out* memory or FIFO memory. When all the locations in the memory are filled, no more data can be written into it. Similarly, when the memory is empty, nothing can be read from it. The reading and writing operations are internally organised and the appropriate locations for writing into and reading out are pointed to by *Write Pointer* and *Read Pointer* respectively, and no address is required for accessing the memory for reading or writing.

The writing and reading operations are performed at independent data rates. This type of memory is used to interface slow input/output (*I/O*) devices to the fast operating computers. The FIFO memory is useful as a *data-rate buffer*.

A FIFO memory is basically a SRAM array with two ports and control circuitry. One port is for writing into and the other for reading from the memory array.

Each port has separate access, data, and control signal for accessing a common SRAM array. The SRAM with two ports is known as a *dual-port* SRAM. The dual-port SRAMs can be

- Asynchronous dual-port SRAM
- Synchronous dual-port SRAM
- Sequential access SRAM

In asynchronous dual-port SRAM, the operation of both the ports is asynchronous, whereas it is synchronous with two different clocks in the case of synchronous dual-port SRAM. There are two different options available in the case of synchronous dual-port SRAMs; similar to normal synchronous SRAMs. These are pipe-lined and flow-through versions.

A sequential access two-port SRAM has one of the ports random (asynchronous) access and the other port is sequential access. It is used to interface the asynchronous and synchronous components of a digital system. Normally, the controlling processor is connected to the asynchronous port and the slave processor on the synchronous side. This type of FIFO memory is used for peripheral controllers, networking equipment such as bridges, routers, etc.

The following types of FIFO memories are available:

- Asynchronous FIFO memory
- Synchronous FIFO memory
- Bi-directional FIFO memory

11.9.1 Asynchronous FIFO Memory

Figure 11.31 shows the functional block diagram of an asynchronous FIFO memory. It has a RAM array of size $M \times N$, where M is the number of words, also known as *depth*, and N is the number of bits in the word (width). The width is normally 9-bit or 18-bit. The 9-bit/18-bit wide data array allows for control and parity bits. This feature is especially useful in data communications applications where it is necessary to use a parity bit for transmission/reception error checking. There are two ports, one for writing data into the array (DATA INPUTS D_0-D_8) and the other for reading out the data (DATA OUTPUTS Q_0-Q_8). Here, a 9-bit wide

data array has been assumed. The READ and WRITE POINTERS point to the locations that are available at any point of time for reading or writing. There are two flags \overline{FF} (full flag) and \overline{EF} (empty flag) provided to prevent data overflow and underflow. The \overline{FF} LOW indicates that the memory is full which prevents further writing of data. Similarly, the \overline{EF} LOW indicates that the memory is empty and further reading is prevented. The expansion of data width and depth can be obtained by making use of multiple devices for which EXPANSION LOGIC BLOCK with pins \overline{XI} and \overline{XO} are available. The FIFO can be reset by activating \overline{RS} (reset) input. During reset, both the internal read and write pointers are set to the first location.

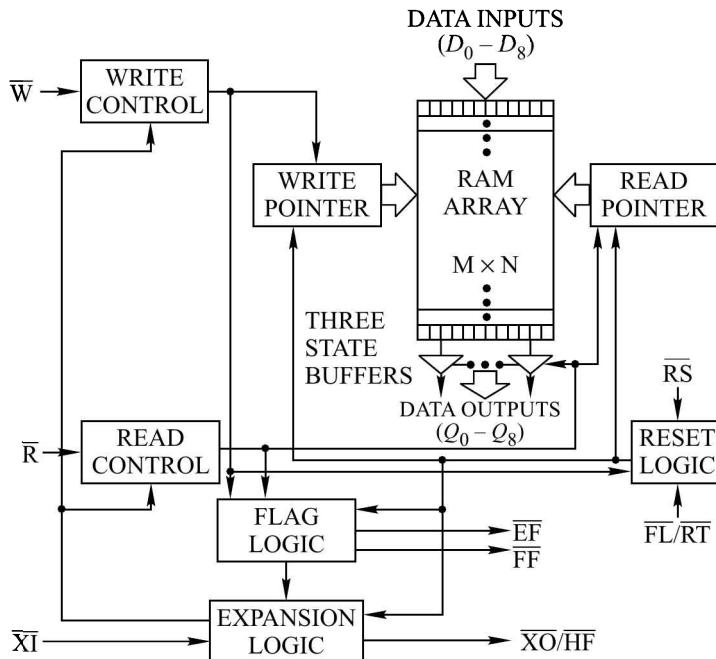


Fig. 11.31 Functional Block Diagram of an Asynchronous FIFO Memory

The $\overline{XO/HF}$ output is a dual-purpose output. In the single device mode, when \overline{XI} input is grounded, this output acts as an indication of a half-full memory. After half of the memory is filled and at the falling edge of the next write (\overline{W} low) operation, the \overline{HF} (half full flag) will be set LOW and will remain LOW until the difference between the write pointer and read pointer is less than or equal to one half of the total memory of the device. It is then reset (HIGH) by using rising edge of the read operation. In the depth expansion mode, the input \overline{XI} is connected to \overline{XO} of the previous device.

FIFO memory has a retransmit feature (\overline{RT}) that allows for the reset of the read pointer to its initial position when \overline{RT} is LOW. This allows retransmission from the beginning of data. This feature is available in only single device mode and not in the depth expansion mode. In the depth expansion mode, the $\overline{FL/RT}$ is grounded to indicate that it is the first loaded device. Figure 11.32 shows the block diagram of the asynchronous FIFO memory.

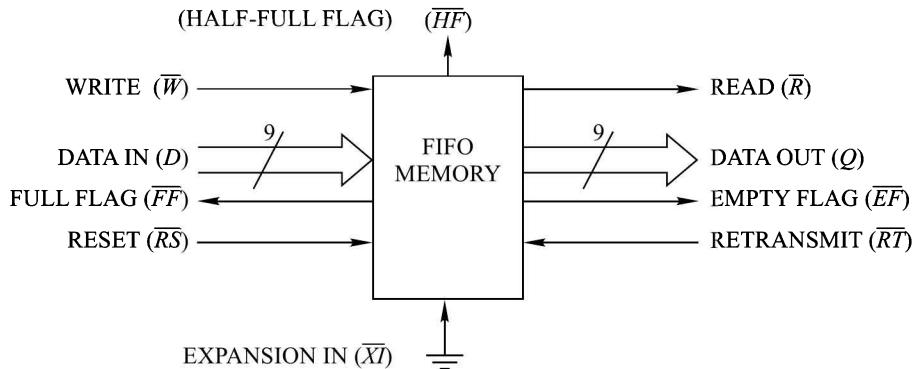


Fig. 11.32 Block Diagram of Asynchronous FIFO Memory

Width Expansion

The width of FIFO system can be increased by simply connecting the corresponding input control signals of multiple devices. The flags can be detected from any one of the device. Figure 11.33 shows the method of width expansion of FIFO system. Here, there are two FIFO devices with 9-bit width. The FIFO-1 can be used for upper 9 bits and FIFO-2 for the lower 9 bits of 18-bit data input. Similarly, the corresponding 18-bit data output can be interfaced to the processor on the other side.

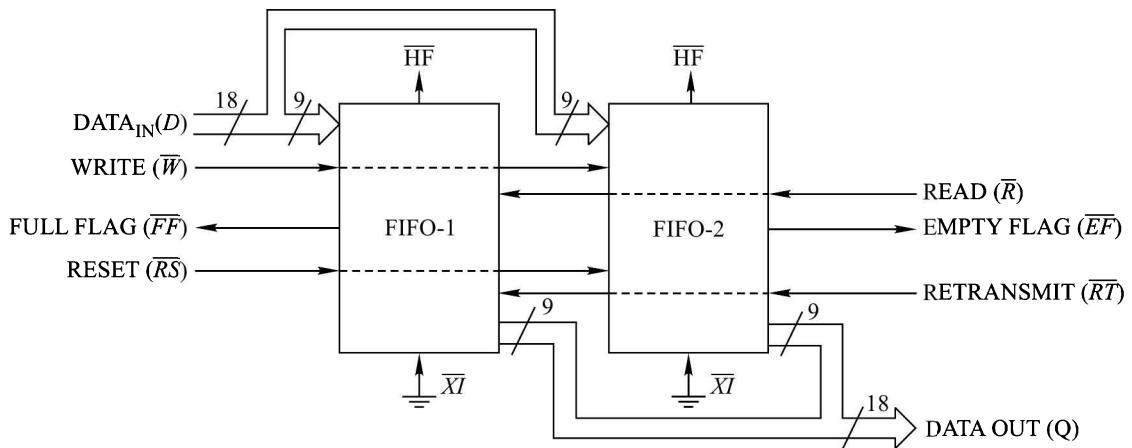


Fig. 11.33 Width Expansion of Asynchronous FIFO Memories

Depth Expansion

For the depth expansion of FIFO memories the following conditions are required to be met:

- \bar{FL} of the first device must be grounded and for all other devices, it is to be connected to logic level HIGH

- The \overline{XO} output of each device is to be connected to the \overline{XI} input of the next device.
- External logic is required to generate composite \overline{FF} and \overline{EF} signals.

Figure 11.34 shows the depth expansion method of the FIFO memories. Here, three FIFO memory devices are connected giving an overall depth of $3 \times M$ words. The FIFO-1 is the first device which will be loaded.

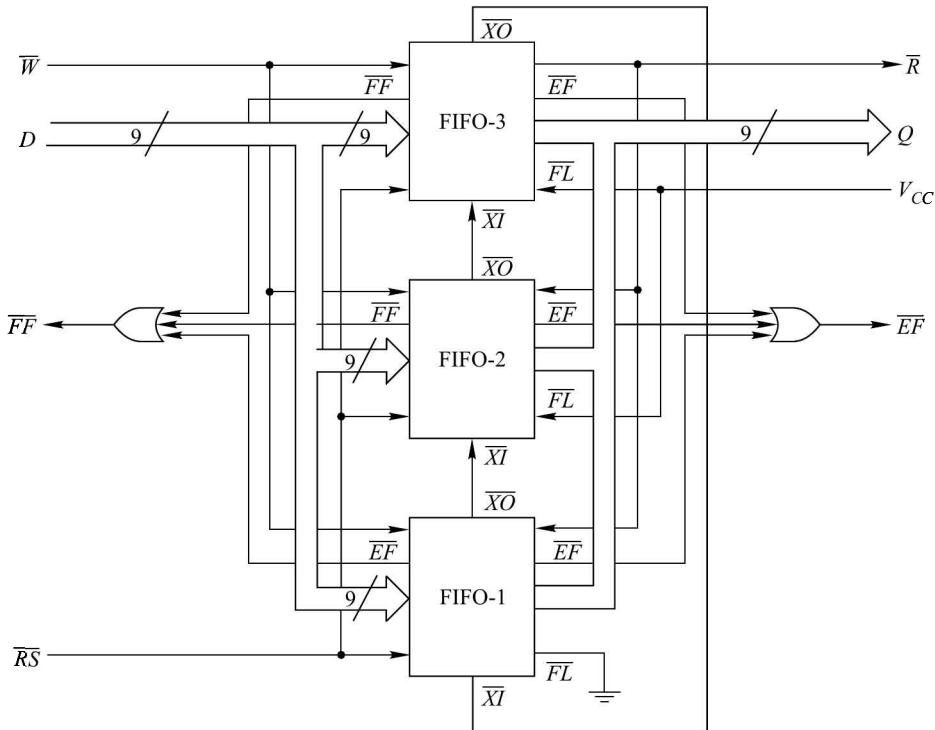


Fig. 11.34 **Depth Expansion of Asynchronous FIFO Memories**

Compound Expansion

The two expansion techniques described above can be applied together in a straight forward manner to achieve large FIFO arrays.

Bi-directional Operation

In a bi-directional FIFO system, data buffering between two systems, in which each system is capable of read and write operations, is allowed. It can be designed by pairing two FIFO memory devices as shown in Fig. 11.35.

Parallel-To-Serial FIFO

The parallel-to-serial FIFO memories have parallel input port and serial output port. These are available in various sizes. Their width and depth can be increased by using multiple devices. For the serial output, it is

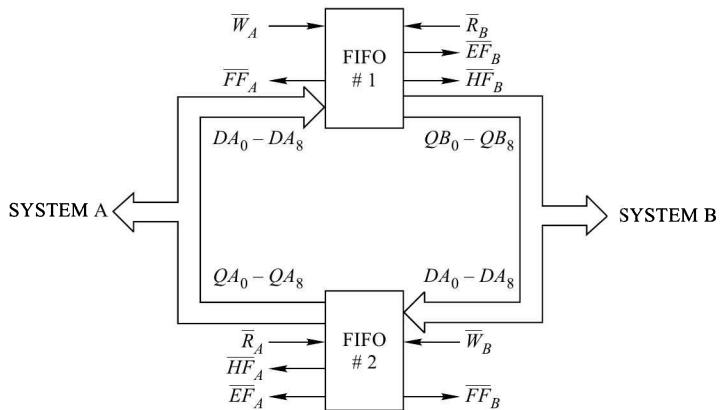


Fig. 11.35 Bi-Directional Asynchronous FIFO Memory

possible to get the least-significant bit (LSB) or the most-significant bit (MSB) first by programming. It can be used to buffer wide word widths which make these FIFOs ideal for laser printers, FAX machines, local area networks (LANs), Video storage, and disk/tape controller applications. These are asynchronous FIFO memory devices.

11.9.2 Synchronous FIFO Memory

Figure 11.36 shows the functional diagram of a synchronous FIFO memory. The synchronous FIFO memories are similar to the asynchronous FIFO memories as far as their SRAM array is concerned, but their read

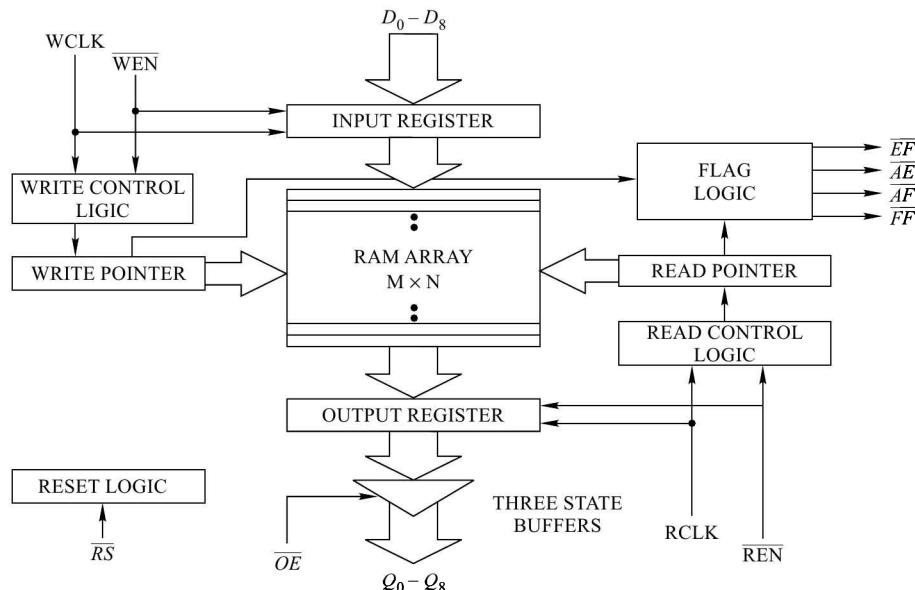


Fig. 11.36 Functional Block Diagram of a Synchronous FIFO Memory

and write controls are synchronous. There are two asynchronous (independent) clocks, one for the read operation ($RCLK$) and another one for the write operation ($WCLK$). Both the read and write operations can be performed using a single clock also by connecting $WCLK$ and $RCLK$ together. The \overline{WEN} and \overline{REN} inputs are the write enable and read enable inputs respectively. The synchronous FIFO devices have four flags. In addition to full flag (\overline{FF}) and empty flag (\overline{EF}), there are two programmable flags. These are almost-full flag (\overline{AF}) and almost-empty flag (\overline{AE}). These flags are set to Full – 7 and Empty +7 values by default, and can be programmed to any other values. For example, the default setting of \overline{AF} flag for a FIFO memory of depth 1024 words is 1017. Figure 11.37 shows the block diagram of a synchronous FIFO memory.

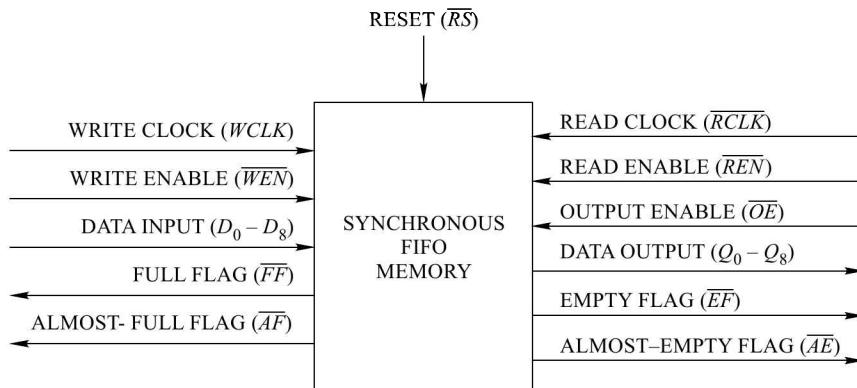


Fig. 11.37 Block Diagram of Synchronous FIFO Memory

Width Expansion

The word width of synchronous FIFO can be increased simply by connecting the corresponding input control signals of multiple devices. Composite \overline{FF} and \overline{EF} are created by using external logic. The \overline{AE} and \overline{AF} can be detected from any one device. Figure 11.38 shows the width expansion method.

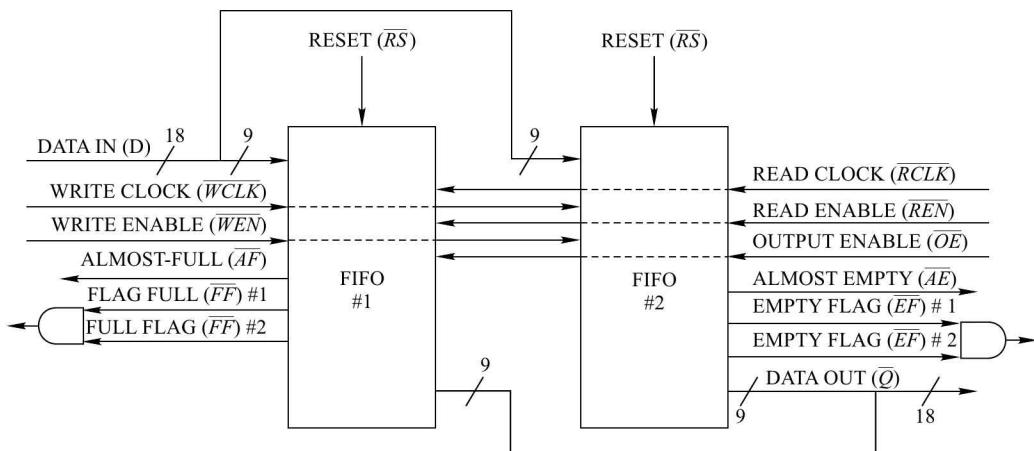


Fig. 11.38 Width Expansion of Synchronous FIFO Memories

Depth Expansion

The depth expansion of synchronous FIFO memories is possible by using expansion logic to direct the flow of data. It can have alternate data access from one device to the next in a sequential manner.

11.9.3 Bi-Directional FIFO Memory (BiFIFO)

The BiFIFO memory is a synchronous FIFO device in which there are two independent clocked FIFOs buffering data in opposite directions. There are two clocks, one for each port which may be asynchronous or coincident. Figure 11.39 shows a simplified block diagram of a synchronous Bi-directional FIFO memory.

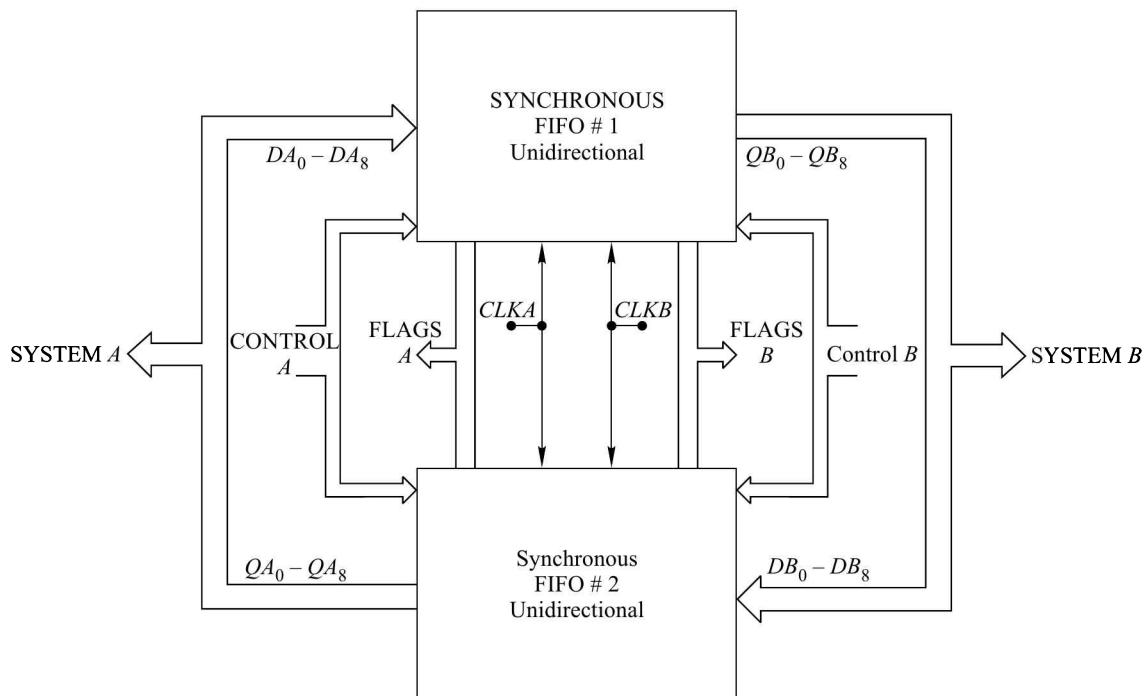


Fig. 11.39 *Block Diagram of a BiFIFO*

11.9.4 Dual-Port SRAM and FIFO Memory ICs

Multi-port RAM ICs are available in a variety of configurations, architectures, and options. The first-in, first-out (FIFO) memories are basically dual-port SRAMs with various types of input/output interfaces to suit different applications. Some of the available dual-port SRAM and FIFO memory ICs are given in Tables 11.15 and 11.16 respectively. Their detailed technical and operational specifications can be obtained from the manufacturers' catalogues or their websites.

Table 11.15 Available CMOS Dual-Port SRAM ICs

IC No.	Organisation No. of bits	Power supply voltage	Speed	Output compatibility	Type
7005	8 K × 8	5 V	15 ns	TTL Compatible	Asynchronous
70P244	4 K × 16	1.8 V	40 ns	LVTTL Compatible	Asynchronous
70P254	8 K × 16	1.8 V	40 ns	LVTTL Compatible	Asynchronous
70P264	16 K × 16	1.8 V	40 ns	LVTTL Compatible	Asynchronous
709079	32 K × 8	5 V	12 ns	TTL Compatible	Synchronous
709089	64 K × 8	5 V	12 ns	TTL Compatible	Synchronous
70824	4 K × 16	5 V	20 ns	TTL Compatible	Sequential access*

*one port is sequential and the other port is a asynchronous (random).

11.10 CHARGE COUPLED DEVICE MEMORY

The charge coupled device (CCD), a new concept for storage of digital information, was announced in early 1970 by Bell Telephone Laboratories of U.S.A. It is an array of MOS capacitors operating as a dynamic shift register. CCDs are simple, versatile, and low cost devices and can be used wherever a serially accessed memory is required.

The operation of CCDs involve the following steps:

1. Conversion of digital input signal into charge,
2. Transfer of charge through various stages in sequential manner, and
3. Conversion of charge at the output into digital signal.

During each charge transfer step, a small amount of charge is lost. Also, due to thermal effects, undesirable charge may be generated which is known as *dark current*. To overcome these defects, the charge is recirculated around the shift register for refreshing.

11.10.1 Basic Concept of CCD

Consider a *p*-type silicon substrate covered with a thin oxide layer and closely spaced metallic electrodes, as shown in Fig. 11.40. Each metallic electrode (gate) and the substrate form a MOS capacitor which can store charge. If a positive voltage is applied at a gate electrode, a depletion region is formed in the substrate immediately under the metallic electrode. This happens due to the repulsion of free holes in the substrate because of the positive voltage at the gate electrode. These holes are driven downward away from the oxide layer and consequently immobile negative ions are exposed and a depletion region comes into existence. In Fig. 11.40, a positive voltage V_1 is applied at the G_1 gate and the other two gates are held at the same potential as the substrate. The depletion region is indicated below the gate G_1 . This plot also represents the potential-energy barrier (well) for electrons, which are the minority charge carriers. Now, if a packet of negative charge is injected into the depletion region, these charges can move freely within the well, but cannot penetrate the potential-energy walls of the well. This means that as long as the voltage V_1 is present, the negative charge will be held (trapped) there.

Table 11.16 Available CMOS FIFO Memory ICs

IC No.	Organisation No. of bits	Power supply voltage	Speed	Architecture	Output compatibility	Type
72V01	512 × 9	3.3 V	15 ns	Unidirectional	3.3 V LVTTL	Asynchronous
72V02	1 K × 9	3.3 V	15 ns	Unidirectional	3.3 V LVTTL	Asynchronous
72V05	8 K × 9	3.3 V	15 ns	Unidirectional	3.3 V LVTTL	Asynchronous
72V06	16 K × 9	3.3 V	15 ns	Unidirectional	3.3 V LVTTL	Asynchronous
7207	32 K × 9	5 V	30 ns	Unidirectional	TTL	Asynchronous
7208	64 K × 9	5 V	30 ns	Unidirectional	TTL	Asynchronous
72V81	512 × 9 × 2	3.3 V	15 ns	Dual FIFO	3.3 V LVTTL	Asynchronous
72V85	8 K × 9 × 2	3.3 V	15 ns	Dual FIFO	3.3 V LVTTL	Asynchronous
72105	256 × 16	5 V	25 ns	Parallel/Serial FIFO	TTL	Asynchronous
72115	512 × 16	5 V	25 ns	Parallel/Serial FIFO	TTL	Asynchronous
72125	1 K × 16	5 V	25 ns	Parallel/Serial FIFO	TTL	Asynchronous
72V201	256 × 9	3.3 V	10 ns	Unidirectional	3.3 V LVTTL	Synchronous
72V211	512 × 9	3.3 V	10 ns	Unidirectional	3.3 V LVTTL	Synchronous
72V251	8 K × 9	3.3 V	10 ns	Unidirectional	3.3 V LVTTL	Synchronous
72421	64 × 9	5 V	10 ns	Unidirectional	TTL	Synchronous
72241	4 K × 9	5 V	10 ns	Unidirectional	TTL	Synchronous
72251	8 K × 9	5 V	10 ns	Unidirectional	TTL	Synchronous
72V801	256 × 9	3.3 V	10 ns	Dual FIFO	3.3 V LVTTL	Synchronous
72V841	4 K × 9	3.3 V	10 ns	Dual FIFO	3.3 V LVTTL	Synchronous
72V851	8 K × 9	3.3 V	10 ns	Dual FIFO	3.3 V LVTTL	Synchronous
72V3682	16 K × 36 × 2	3.3 V	10 ns	Bi-directional	3.3 LVTTL	Synchronous
72V3692	32 K × 36 × 2	3.3 V	10 ns	Bi-directional	3.3 LVTTL	Synchronous
72V36102	64 K × 36 × 2	3.3 V	10 ns	Bi-directional	3.3 LVTTL	Synchronous
72605	256 × 18 × 2	5 V	20 ns	Bi-directional	TTL	Synchronous
72615	512 × 18 × 2	5 V	20 ns	Bi-directional	TTL	Synchronous

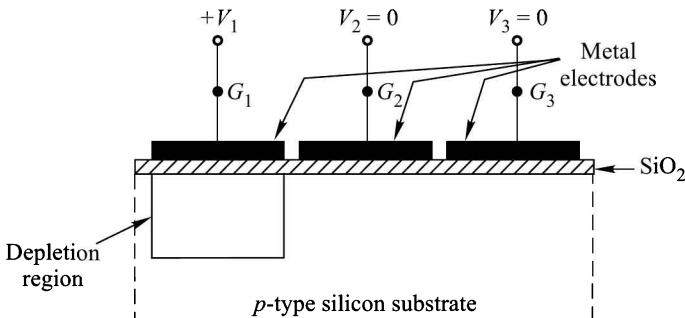


Fig. 11.40 Basic CCD Structure

In a conventional capacitor, the charges are held on conducting plates, whereas in a MOS device the charges are held on a conductor and in the depletion region under the conductor. The stored charge can be moved from left to right down the channel by applying voltages at the gates in a proper sequence. We assume that a logic 1 is stored when a negative charge is held in the depletion region and a logic 0 is stored when the depletion region is empty. This type of operation makes it possible to make long shift registers using these devices.

11.10.2 Operation of CCD

A portion of the structure of a 4-phase charge-coupled device (CCD) shift register, along with the charge transfer plots, is shown in Fig. 11.41a. The four clock waveforms ϕ_1 , ϕ_2 , ϕ_3 , and ϕ_4 used to drive the circuit are shown in Fig. 11.41b. An array of four adjacent electrodes driven by the 4-phase clock constitutes a single dynamic FLIP-FLOP. The 4-phase arrangement is required to give this dynamic FLIP-FLOP the operating features of a master-slave FLIP-FLOP, which allows shift register operation and assures that the data move in only one direction. The whole device is a long array of MOS devices in which all the ϕ_1 electrodes are connected together. A similar arrangement exists for the ϕ_2 , ϕ_3 , and ϕ_4 electrodes.

During the time interval t_1 only ϕ_1 is at a positive voltage, so that depletion regions are formed only under ϕ_1 . The charge indicated in the depletion region under ϕ_1 is injected either from an outside source or from the preceding ϕ_4 gate.

During the interval t_2 , the depletion regions under ϕ_1 gates persist while new depletion regions are formed under ϕ_3 gates because the clock ϕ_3 becomes positive. In the interval t_3 , the clock ϕ_2 also becomes positive while the clocks ϕ_1 and ϕ_3 are held positive. Therefore, depletion regions are generated extending from the ϕ_1 gates to the ϕ_3 gates. As a result, the charge is now able to spread throughout the extended region. During the interval t_4 , the clock ϕ_1 becomes 0 thereby eliminating the depletion regions under the ϕ_1 gates. Similarly, during the interval t_5 , the depletion regions under ϕ_2 gates vanish and the charge (or no charge) originally under the ϕ_1 gates is pushed laterally to regions under the ϕ_3 gates.

Continuing the above logic, we observe that in the succeeding intervals t_6 through t_8 , and finally back to t_1 , the charge under the gate ϕ_3 will be moved to the region under the next ϕ_1 gate. Hence, altogether, after eight intervals, charge (or no charge) under a ϕ_1 gate will shift to the next ϕ_1 gate.

Special arrangement must be made to inject charge into the first depletion region as required, and to detect the presence (logic 1) or absence (logic 0) of charge at the last depletion region. The injection and detection of charge must be done in synchronism with the clock waveform.

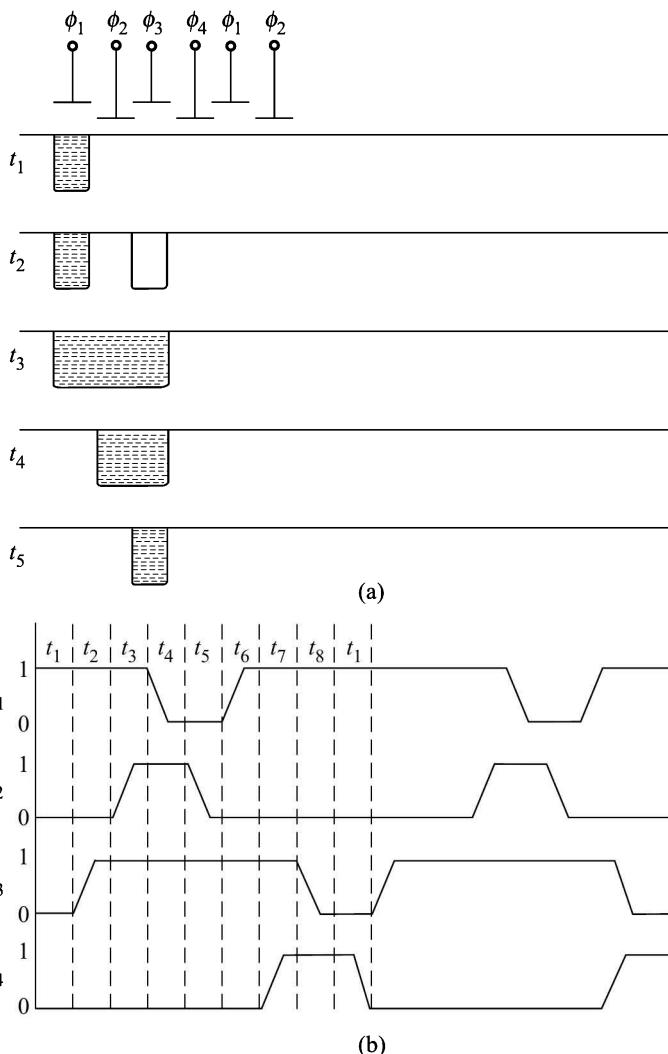


Fig. 11.41 (a) A Portion of Four-Phase CCD Structure Along with the Charge Transfer Plots and (b) Four-Phase Clock Waveform

When the charge transfer takes place down the shift register, there is some loss of charge. Therefore, it is necessary to incorporate provision for refreshing the charge at periodic intervals along the length of the CCD structure. In fact, the charge transfer efficiency is so high (~99.999%) that 100 or more FLIP-FLOPs can be cascaded before refreshing becomes necessary.

11.10.3 A Practical CCD Memory Device

The basic organisation of the intel 2416 CCD memory is shown in Fig. 11.42. It is a $16\ 384 \times 1$ bit serial memory, organised as 64 independent recirculating shift registers of 256 bits each. Any one of the 64 registers

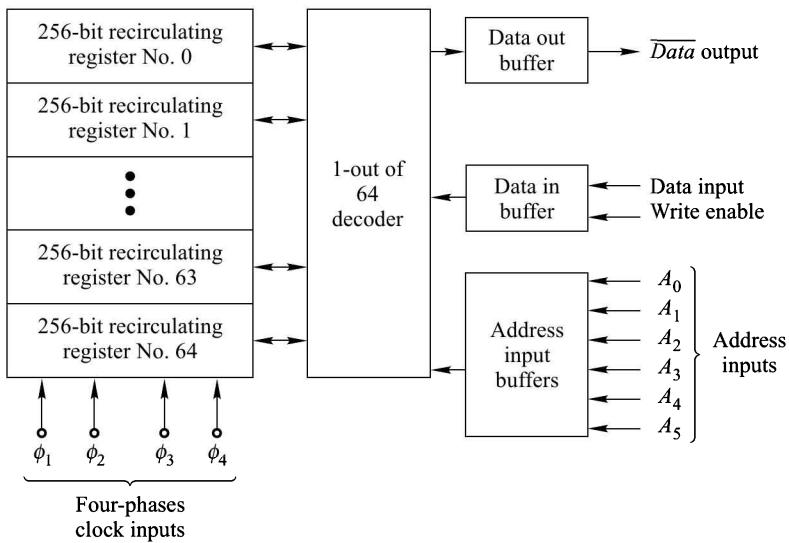


Fig. 11.42 **Basic Organisation of Intel 2416 CCD Memory**

can be accessed by applying the appropriate 6-bit code at the address inputs. The data in the shift registers is simultaneously shifted by using the 4-phase clock signals ϕ_1 through ϕ_4 . After a shift cycle, each of the 64 registers can be selected for an input/output operation by applying the appropriate 6-bit address code.

When addressed, one bit is written into or read from the memory. If the address input is fixed, then as shifting progresses, the bit positions in the addressed register will be presented serially for reading or writing. The output is open-drain which allows wired-OR connection. During the interval between shifts, we can have access to a bit from each of the registers by changing the address. The 64 bits, which can be accessed by changing address, are available on a random access basis. The 256 bits in a single register are available only in the serial mode.

In serial mode operation, access to a desired bit may require no shift or it may require upto a maximum of 256 shifts. On an average, $256/2 = 128$ shifts are required for accessing a bit. This access time in serial operation is known as *latency* or *latency time*.

The Intel 2464 is a 65536-bit CCD memory organised in a manner similar to the 2416 chip, but with 256 independent circulating registers of 256 bits each. It has an active-low chip enable signal \overline{CE} .

The bit capacity and word lengths can be expanded by using appropriate circuitry (Probs. 11.19 and 11.20).

SUMMARY

Semiconductor memories are invariably an essential part of any digital system. All memory devices store binary logic levels (1 and 0) in an array structure. Memory ICs are available with various sizes of array in terms of the number of words and the number of bits in the word. The number of words and the word size can be increased, if necessary, by using multiple chips. Different types of memory devices have been dealt in detail and some of the commercially available ICs have been given for each type. Various types of ROM devices, such as ROM, PROM, OTP EPROM, EEPROM, parallel and serial EEPROM have been discussed alongwith their erasing and programming techniques. All types of ROMs are non-volatile.

Both the types of RAM devices, i.e., SRAM and DRAM have been discussed starting from their basic cell. Asynchronous and synchronous operations of SRAM and DRAM have been covered in detail. One of the major applications of SRAM is in *Cache memories* because of their very high speed. In general, the SRAM devices are very much faster in comparison to DRAM devices. All types of RAMs are volatile.

Flash memory, a specific type of EEPROM, is a non-volatile memory which allows in-circuit writing. It is used in many digital systems, such as cellular phones, cameras, LAN switches, embedded controllers, memory cards, and USB flash drives etc. The flash memory ICs are available in parallel as well as serial interface.

The first-in, first-out FIFO memory devices are very useful for interfacing various types of *I/O* devices to the computers as data-rate buffers. They are available in various sizes. Their depth and width can be increased by using multiple devices. Bi-directional BiFIFO and parallel-to-serial FIFO devices have also been discussed.

GLOSSARY

Access time Time required for reading or writing a memory location.

Address The binary code of a memory location.

Address bus A parallel array of conductors used for accessing a memory location.

Address decoder A n -line-to- 2^n lines decoder used to select a specific memory chip or memory location.

Asynchronous memory A memory device in which read and write are unclocked.

BEDO DRAM Burst extended data output dynamic random-access memory.

Bidirectional bus A bus (group of electrical lines) capable of transmitting data in both the directions.

Bi-directional FIFO memory A FIFO memory capable of buffering data between two systems for read and write.

BIOS Basic input/output system – A set of programs in ROM that interfaces the *I/O* devices in a digital system.

Burst A memory feature which allows read from or write at upto four locations using a single address.

Cache memory A high-speed memory (normally SRAM) that stores the most recently used instructions or data from the slower main memory.

CAM (Content addressable memory) A special purpose RAM device which can be accessed by its contents. It is also known as ‘associative memory’.

CCD (Charge coupled device) It is serial-access semiconductor memory device with very high bit packing density.

Chip A piece of silicon or other semiconductor material on which an IC is fabricated.

Chip enable An input control signal which when activated enables the chip.

Chip select An input control signal that allows the chip to be selected.

Control bus A bus used for handling control signals.

Cycle time The minimum time between successive read or write cycles in a memory.

Data bus A bus used for carrying data.

Data-rate buffer A buffer that allows two digital systems with different data rates (speeds) to communicate.

DDR Double data rate.

DRAM Dynamic random-access memory.

Dual-port SRAM A static random-access memory with two ports.

Dynamic memory A memory in which data needs to be refreshed periodically. Data is stored on MOS capacitors.

EAROM (Electrically alterable read-only memory) A read-only memory in which the stored words can be erased electrically. Also known as E²PROM (Electrically erasable and programmable read-only memory).

EDO DRAM Extended data output dynamic random-access memory.

EEPROM Electrically erasable and programmable read-only memory.

EPROM (Erasable programmable read-only memory) A read-only memory which can be erased by exposure to ultraviolet light, and then reloaded with new information.

Erasable memory A memory contents of which can be erased.

First-in, First-out (FIFO) memory A static random-access memory array with two ports in which data words are read out in the same order in which they were written in.

Flash memory A specific type of EEPROM, a non-volatile memory, which allows in-circuit writing.

FPM DRAM Fast page mode dynamic random-access memory.

Fusible link A link of nichrome, or some other materials, used in PROMs and other programmable devices which can be either burnt or kept intact while storing 0s and 1s in these devices.

Non-erasable memory A memory contents of which can not be erased, such as a ROM.

Non-volatile memory A memory that does not loose its contents when power is turned off, such as ROMs.

Non-volatile RAM A flash-memory is also referred to as a non-volatile random-access memory.

OTP One time programmable.

Parallel-to-serial FIFO memory A FIFO memory with parallel input port and serial output port.

Programming (of Memory) To store the desired data in a programmable memory which is of read-only type.

PROM (Programmable read-only memory) A read-only memory that can be programmed only once by the user by selectively opening the fusible links.

PROM programmer An equipment that is used to program a programmable ROM.

RAM (Random-access memory) A read-and-write semiconductor memory in which any memory location can be accessed for reading or writing at random.

Refresh The act of restoring the data in a dynamic memory.

ROM (Read-only memory) A semiconductor memory in which data can only be read.

SDRAM Synchronous DRAM

Semiconductor memory A memory fabricated using semiconductor material.

Serial-access memory A memory in which the access time of a stored bit or word depends on its location in the memory.

Serial EEPROM An EEPROM device with serial I/O.

Serial flash memory A flash memory with serial I/O.

Static RAM (SRAM) A random-access memory in which read out and write into are not clocked.

Volatile memory A memory that loses its contents when power is turned off.

REVIEW QUESTIONS

- 11.1 Information in a memory chip is stored in _____ form.
- 11.2 The maximum number of bytes which can be stored in a memory of size 1024×8 is _____.
- 11.3 The number of address lines required in a memory of $128 K \times 8$ is _____.
- 11.4 While specifying the memory size, the letter *K* stands for _____.
- 11.5 An EPROM is a _____ access memory.
- 11.6 A shift register is a _____ memory.
- 11.7 The contents of location *0A00H* of RAM is 01001100, its contents after a read operation will be _____.
- 11.8 The contents of location *BAC0H* of EPROM is 11100101, its contents after a read operation will be _____.
- 11.9 The number of IC chips of memory size 1024×4 required to have $16 K \times 8$ memory will be _____.
- 11.10 The contents of location *FEE0H* of a RAM is *BEH*, its contents after a write operation is performed with *ECH* data at the data bus will be _____.
- 11.11 The access-time of a RAM is 10 ns, the minimum time which must elapse between two read operations will be _____.
- 11.12 An EAROM is _____ erasable.
- 11.13 An EPROM is erased by _____.
- 11.14 A dynamic RAM is fabricated using _____ technology.
- 11.15 CAM stands for _____.
- 11.16 A serial EEPROM has _____ input/output.
- 11.17 An SRAM with two ports is known as a _____ SRAM.
- 11.18 A FIFO memory has _____ ports.
- 11.19 A serial flash memory has _____ input/output
- 11.20 A FIFO memory can be used as a _____ buffer.
- 11.21 A _____ FIFO memory can be used for two way communication between two digital systems operating at different speeds.
- 11.22 A serial EEPROM IC has _____ number of pins than a parallel EEPROM IC.
- 11.23 The burst feature in a synchronous SRAM _____ its speed.
- 11.24 The data rate of a synchronous SRAM operating at 200 MHz is _____ MHz.

PROBLEMS

- 11.1 For a memory with *M* words storage, find the number of pins required for addressing and the address range in binary format for each of the following cases:

- (a) $M = 4$
- (b) $M = 16$
- (c) $M = 64$
- (d) $M = 256$
- (e) $M = 1024 = 1 \text{ K}$
- (f) $M = 2048 = 2 \text{ K}$
- (g) $M = 64 \text{ K}$
- (h) $M = 1 \text{ M}$

11.2 Express the address range for each of the cases of Prob. 11.1 in

- (a) Hexadecimal format.
- (b) Octal format.

11.3 The access time and cycle time for a set of memories are given in Table 11.17. Determine the maximum rate at which data can be accessed in each case.

Table 11.17

Memory	Access time ns	Cycle time ns
A	1500	1500
B	300	580
C	450	450
D	200	200
E	60	60
F	800	800

11.4 The block diagram of a $1 \text{ K} \times 4$ bit static RAM is shown in Fig. 11.43. Find the number of RAM chips required, if any, to obtain.

- (a) 4096×4 bit RAM
- (b) 1024×8 bit RAM
- (c) $16 \text{ K} \times 8$ bit RAM.

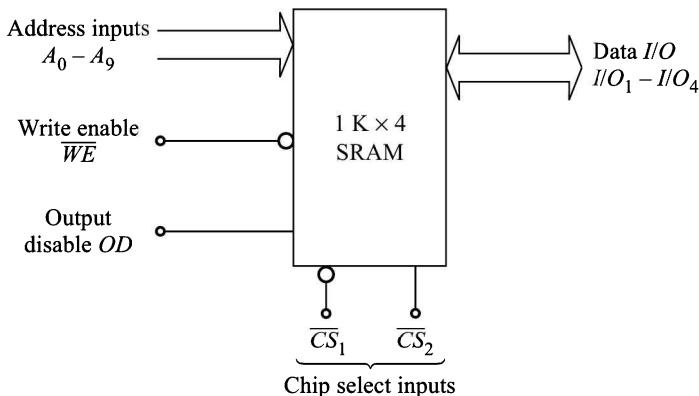


Fig. 11.43 Block Diagram of a $1 \text{ K} \times 4$ SRAM

11.5 Implement the RAMs of Prob. 11.4.

11.6 The block diagram of Intel 2716 2K \times 8 EPROM is shown in Fig. 11.44. Find the number of 2716 and other ICs required to obtain.

- (a) 4 K bytes of ROM
- (b) 2 K \times 16 ROM
- (c) 4 K \times 16 ROM

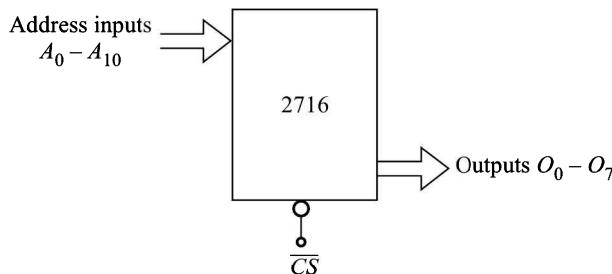


Fig. 11.44 **Block Diagram of 2716 EPROM**

11.7 Implement the ROMs of Prob. 11.6.

11.8 Explain the following:

- (a) Linear selection addressing.
- (b) Coincident selection addressing.

11.9 Figure 11.45 shows the block diagram of the asynchronous SRAM 65C256. Design an SRAM of size 64 K \times 16 bits.

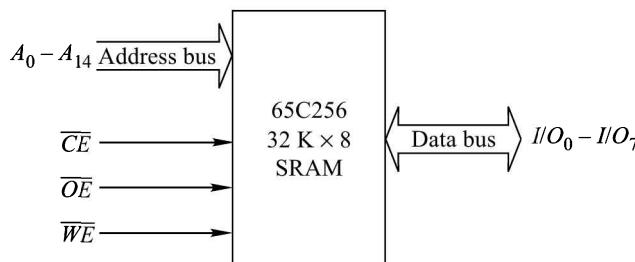
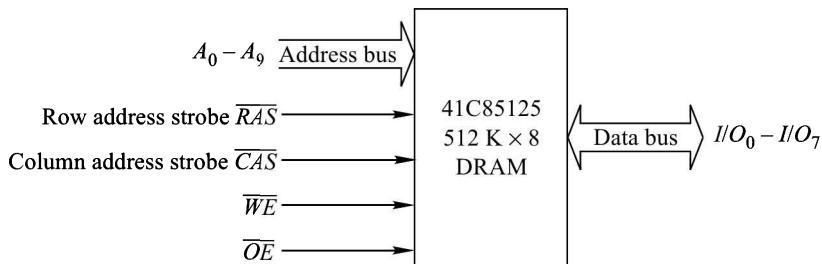


Fig. 11.45 **Block Diagram of 65C256 SRAM**

11.10 Explain the following:

- (a) Late-write SRAM
- (b) ZBT or ZEROSB SRAM
- (c) Flow-through SRAM
- (d) Pipe-lined SRAM

11.11 Figure 11.46 shows the block diagram of 41C85125 DRAM. It is a FPM DRAM with 1024 row addresses. Design a DRAM of size 512 K \times 16.

Fig. 11.46 **Block Diagram of 41C85125 DRAM**

- 11.12** Verify the operations given in Table 11.11.
- 11.13** Design a 16×2 CAM using two 8×2 CAM chips.
- 11.14** Design an 8×8 CAM using four 8×2 CAM chips.
- 11.15** Design a 16×8 CAM using 8×2 CAM chips.
- 11.16** It is desired to find the maximum valued number stored in a CAM of size 16×8 . Suggest a suitable method. Compare this method with the method used if a RAM is used instead of a CAM.
- 11.17** Repeat Prob. 11.16 to Find the minimum valued number.
- 11.18** It is desired to design a memory system for storing information which is not already stored in it. This type of memory is known as *learning memory*. Will you prefer to use RAM or CAM for this purpose, why?
- 11.19** Suggest a suitable arrangement for expanding the bit capacity of CCDs.
- 11.20** Suggest a suitable arrangement for expanding the word length of CCDs.