Ch-5

## 5.1 Semiconductor Main Memory →

The basic element of a semiconductor memory is the memory cell

All semiconductor memory cell having following properties:

(1) They exhibit two stable states such as 1 & 0
(2) They are capable of being written into, to set the state.
(3) They are capable of being read to sense the state

Memory cell has 3 functional terminals capable of carrying an electrical signal.

(1) The select terminal selects a memory cell for read or write operation.
(2) The control terminal indicates the read or write operation.
(3) Through 3-rd terminal read or write operation takes place.

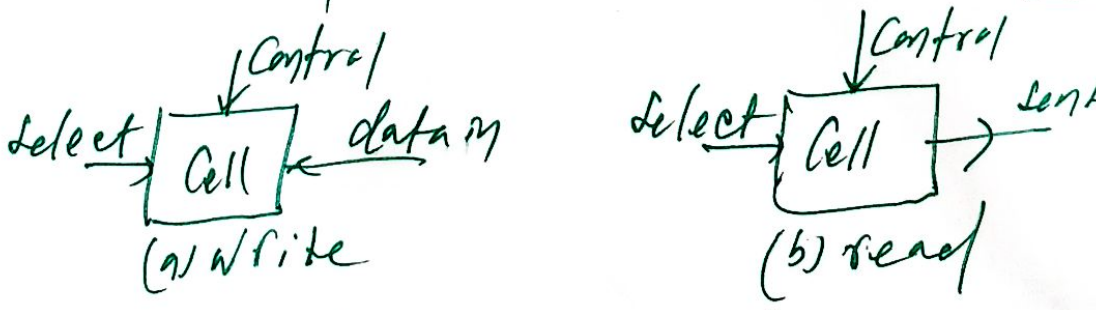


Fig 1 Memory cell operation.

The main memory are the RAM & ROM

① RAM (Random access memory)

* The most common main memory is RAM.
* It can be read & to write new data into the memory easily & rapidly.
* Both reading & writing accomplished through the use of electrical signals.
* It is volatile in nature. It must be provided with constant power supply.
* RAM acts as a temporary data storage.

RAM is two types:

① DRAM (Dynamic RAM)
② SRAM (static RAM)

① DRAM (Dynamic RAM)

RAM technology is divided into two techniques ① dynamic ② static.

* A Dynamic RAM is made with cells that store data as charge on capacitors. The presence or absence of charge in a capacitor is interpreted as a binary 1 or 0. As capacitors have a natural tendency to discharge, DRAM requires periodic charge, refreshing to maintain data storage.

## (2) Static RAM →

* SRAM is a digital device that uses the same logic elements used in processor.

In SRAM, binary values are stored using traditional Flip-flop logic-gate configuration.

A SRAM will hold it's data as long as power is supplied to it.

## 5.2 Error Correction →

A semiconductor memory system is subjected to errors. They can be categorized as:

① Hard error (Hard Failure)
② Soft error.

### ① Hard error →

A Hard failure is a permanent physical defect so that the memory cells affected cannot reliably store data. Hard error can be caused due to harsh environmental abuse, manufacturing defects & wear.

### ② Soft error →

A soft error is a random & non-destructive events, that alters the content of one or more memory cells without damaging the memory. It is caused due to power supply problem or alpha particles.

Both hard & soft errors are clearly un-desirable, & most modern main memory system include logic for both detecting & correcting errors.

Fig 5.7 shows the error-correcting code function.

When data are to be written into memory, a calculation (f) is performed on the data to produce a code (check bit). Both the code & data are stored.

If an M-bit word of data is to be stored & the code is of length k, then the actual size of the stored word is M+k bits.

When the previously stored word is read out, the code is used to detect & correct errors.

When the word is fetched, a new code is generated & compared to the stored code. The comparison yields one of three results:

- No errors are detected, the fetched data bits are sent out.
- An error is detected, & it is possible to correct the error. The data bits plus error-correcting bits are fed into a corrector, which produces a corrected set of M bits to be sent out.
- An error is detected, but it is not possible to correct it. This condition is reported.

Fig 5.7 →

## Humming Error correcting code →

The simplest of the error-correcting code is the Humming code. It is a code to detect the error & correcting. It is a single error correctim code.

$$2^k - 1 \geq M + k$$

It follows the no of check bits.

M → no of data bits

k → no of check bits.

### ① Placement of the check bits in the code →

The check bits are located in the following positions that are ascending power of 2 (1, 2, 4, 8, ...)

The layout of the data bits & check bits in data word:

| Bit Position | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Data bits | $D_8$ | $D_7$ | $D_6$ | $D_5$ | | $D_4$ | $D_3$ | $D_2$ | | $D_1$ | | |
| check bits | | | | | $C_8$ | | | | $C_4$ | | $C_2$ | $C_1$ |

(2) Check bits are calculated as follows:

$$C_1 = D_1 \oplus D_2 \oplus D_4 \oplus 5_5 \oplus D_7$$

$$C_2 = D_1 \oplus D_3 \oplus D_4 \oplus D_6 \oplus D_7$$

$$C_4 = D_2 \oplus D_3 \oplus D_4 \oplus D_8$$

$$C_8 = D_5 \oplus D_6 \oplus D_7 \oplus D_8$$

**Q** Let 8-bit data 00111001 stored in the memory. Using the Hamming algorithm, determine what check bits would be stored in the memory with the data word.

**Solution**

$M = 8$

but $k = 4$

$$2^k - 1 \geq 8 + 4$$

This equation satisfied for $k = 4$

Hence, Total code bits $= 8 + 4 = 12$

| Bit position | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit designation | $D_8$ | $D_7$ | $D_6$ | $D_5$ | $C_8$ | $D_4$ | $D_3$ | $D_2$ | $C_4$ | $D_1$ | $C_2$ | $C_1$ |
| Data bit | 0 | 0 | 1 | 1 | | 1 | 0 | 0 | | 1 | | |
| check bit | | | | | 0 | | | | 1 | | 1 | 1 |

Hamming code = 001101001111

$C_1 = D_1 \oplus D_2 \oplus D_4 \oplus D_5 \oplus D_7 = 1 \oplus 0 \oplus 1 \oplus 1 \oplus 0 = 1$

$C_2 = D_1 \oplus D_3 \oplus D_4 \oplus D_6 \oplus D_7 = 1 \oplus 0 \oplus 1 \oplus 1 \oplus 0 = 1$

$C_4 = D_2 \oplus D_3 \oplus D_4 \oplus D_8 = 0 \oplus 0 \oplus 1 \oplus 0 = 1$

$C_8 = D_5 \oplus D_6 \oplus D_7 \oplus D_8 = 1 \oplus 1 \oplus 0 \oplus 0 = 0$

**Q** For the 8-bit word 0011 1001, the check-bit stored with it would be 0111. Suppose, when the word is read from the memory, the check-bits are calculated as 0001. What is the data word that was read from memory.

**Solution**

$C_8\ C_4\ C_2\ C_1$

0 1 1 1

0 0 0 1

0 1 1 0

The result is 0110, this indicates the position 6 is in error. The position 6 means $D_3$ is error. Hence the data word 0011 1101 that was read from memory.

| Bit position | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit designation | $D_8$ | $D_7$ | $D_6$ | $D_5$ | $C_8$ | $D_4$ | $D_3$ | $D_2$ | $C_4$ | $D_1$ | $C_2$ | $C_1$ |
| Word stored as | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| Word fetched as | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| check bit | | | | | 0 | | | | 0 | | 0 | 1 |

Q: Let 4-bit data 1001 stored in the Memory. Using Hamming algorithm, determine what check bit would be stored in the memory with data word.

Soln → Now $M = 4$

Let, $k = 3$

$2^k - 1 \geq M + k$

$2^3 - 1 \geq 4 + 3$

Total code bits $= 4 + 3 = 7$

| Bit position | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|
| Bit Designation | $D_4$ | $D_3$ | $D_2$ | $C_4$ | $D_1$ | $C_2$ | $C_1$ |
| Data bit | 1 | 0 | 0 | | 1 | | |
| check bit | | | | 1 | | 0 | 0 |

$C_1 = D_1 \oplus D_2 \oplus D_4 = 1 \oplus 0 \oplus 1 = 0$

$C_2 = D_1 \oplus D_3 \oplus D_4 = 1 \oplus 0 \oplus 1 = 0$

$C_4 = D_2 \oplus D_3 \oplus D_4 = 0 \oplus 0 \oplus 1 = 1$

Hamming code $= 1001100$

Word stored in the memory as 1001100

# Data bits & check bits (7)

| Data bits | check bits |
|---|---|
| 8 | 4 |
| 16 | 5 |
| 32 | 6 |
| 64 | 7 |
| 128 | 8 |
| 256 | 9 |

based on

$$2^k - 1 = M + k$$

k → no of check bits.

M → no of data bit