

CSE 321b

# Computer Organization (2)

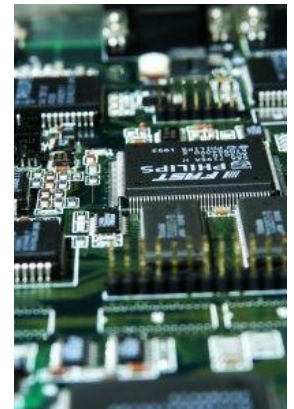
## تنظيم الحاسب (2)

---



3<sup>rd</sup> year, Computer Engineering  
Winter 2015

### **Lecture #6**



Dr. Hazem Ibrahim Shehata

Dept. of Computer & Systems Engineering

Credits to Dr. Ahmed Abdul-Monem Ahmed for the slides

# Adminstrivia

---

- Schedule:
  - Lectures: Monday 9:00am – 11:30am
  - Tutorials: Sunday 10:00am – 11:30am
  - Office hour: TBA
- Assignment #1:
  - To be released next week

Website: <http://www.bit.do/hshehata-courses-zu-cse321b>

Office hours: TBA

## **Ch 5: Internal Memory (*Cont.*)**

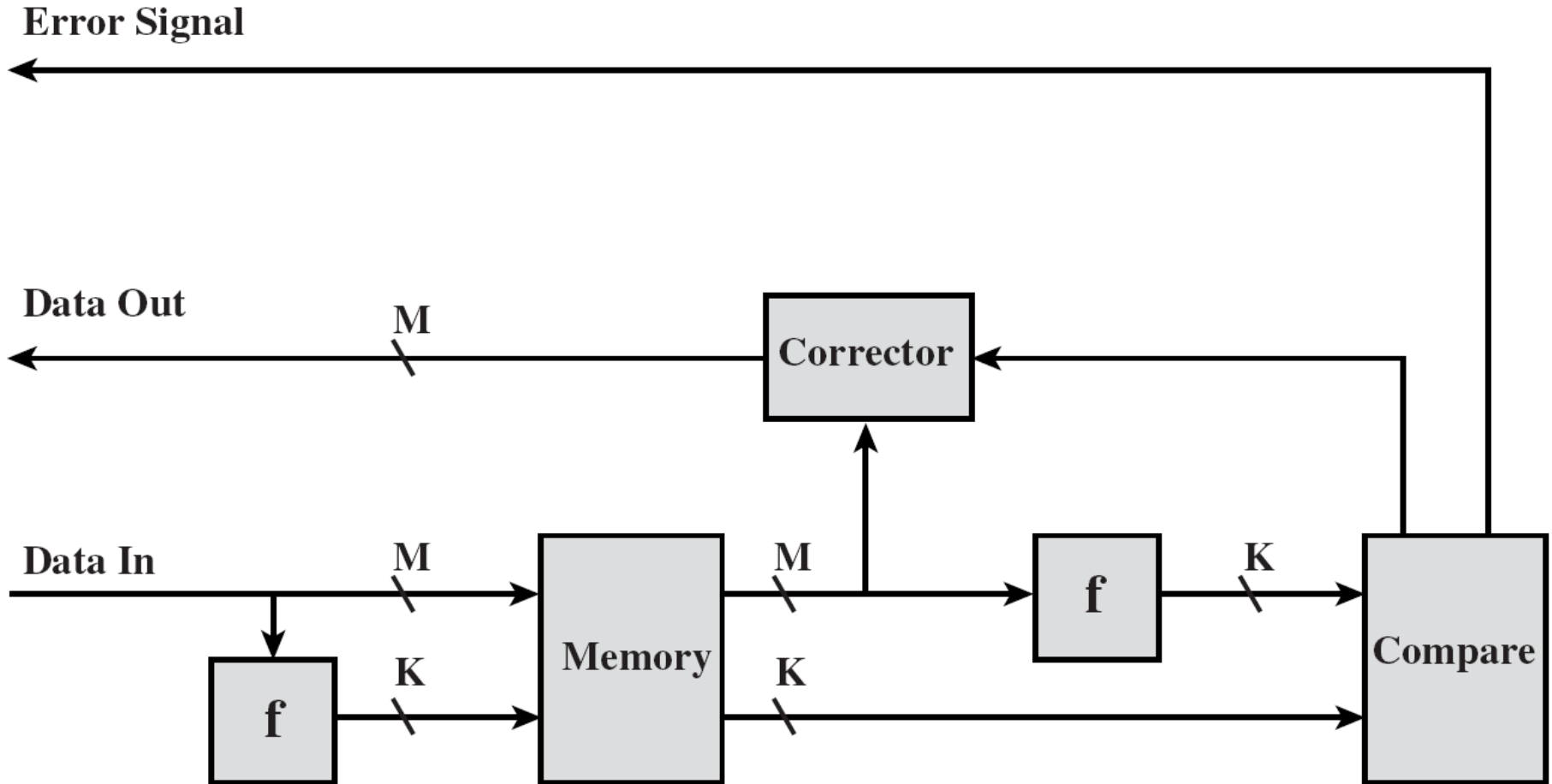
---

# Error Correction

---

- Semiconductor memory is subject to errors
  - Rate: 1 error/hour to 1 error/century in a 1GB memory!
  - Types: hard and soft.
- Hard Failure
  - Permanent physical defect.
  - Mem. cells can't store data: stuck at 0 or 1, or switching.
  - Caused by harsh environments, manufacturing defects, or wear.
- Soft Error
  - Random, non-destructive event that alters contents of one or more memory cells.
  - No permanent damage to memory.
  - Caused by power supply problems or alpha particles.
- Detected/corrected using Hamming error correcting code.

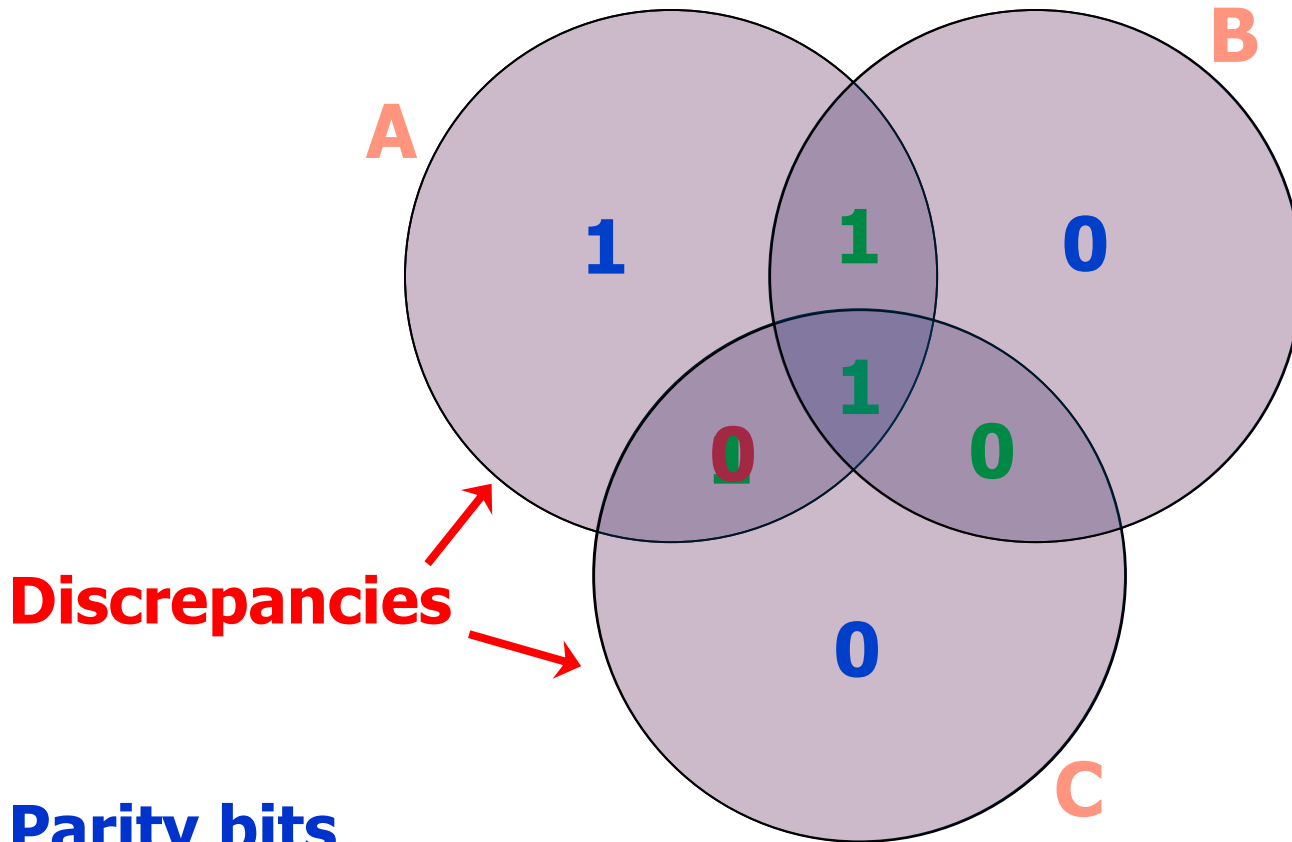
# Error-Correcting Code Function



# Hamming Error-Correcting Code

---

Data bits: 1110



## Parity bits

Chosen so that total number of 1s in each circle is even.

By checking the parity bits, discrepancies are found → error can be easily found and corrected.

# Error-Correcting Codes

---

- A **codeword** consists of  $N$  bits split into  $M$  data bits and  $K$  check (redundant) bits
  - $N = M + K$ .
- **Hamming distance**: Number of bit-positions in which two codewords differ.
  - Ex.: 1**1001**001, 1**0100**001  $\rightarrow$  Hamming distance = 3  $\rightarrow$  3 bit errors are needed to convert one into the other.
- Note: in a code, not all  $2^N$  codewords are legal.
- **Hamming distance of the whole code**: minimum Hamming distance between 2 legal codewords.
- A distance  $d$  code can:
  - **Detect**:  $d - 1$  errors.
  - **Correct**:  $(d - 1)/2$  errors if  $d$  is odd, or  $(d/2) - 1$  errors if  $d$  is even.

# Error Detection/Correction

---

- **Detection:** parity bit.
  - Distance = 2  $\rightarrow$  can detect up to 1 bit error.
  - Ex.: data=1011010  $\rightarrow$  codeword=10110100  $\rightarrow$  any codeword with a distance = 1 (such as: 10100100) is considered illegal  $\rightarrow$  single-bit errors are detectable.
- **Correction:** Consider a code with 4 valid codewords:  
0000000000, 0000011111, 1111100000, 1111111111
  - Distance = 5  $\rightarrow$  can correct up to 2 bit errors.
  - If 0000000111 arrives,  $\rightarrow$  0000011111
  - If 0000000000 becomes 0000000111 due to 3 errors  $\rightarrow$  cannot be corrected properly.



# Single Bit Error Correction

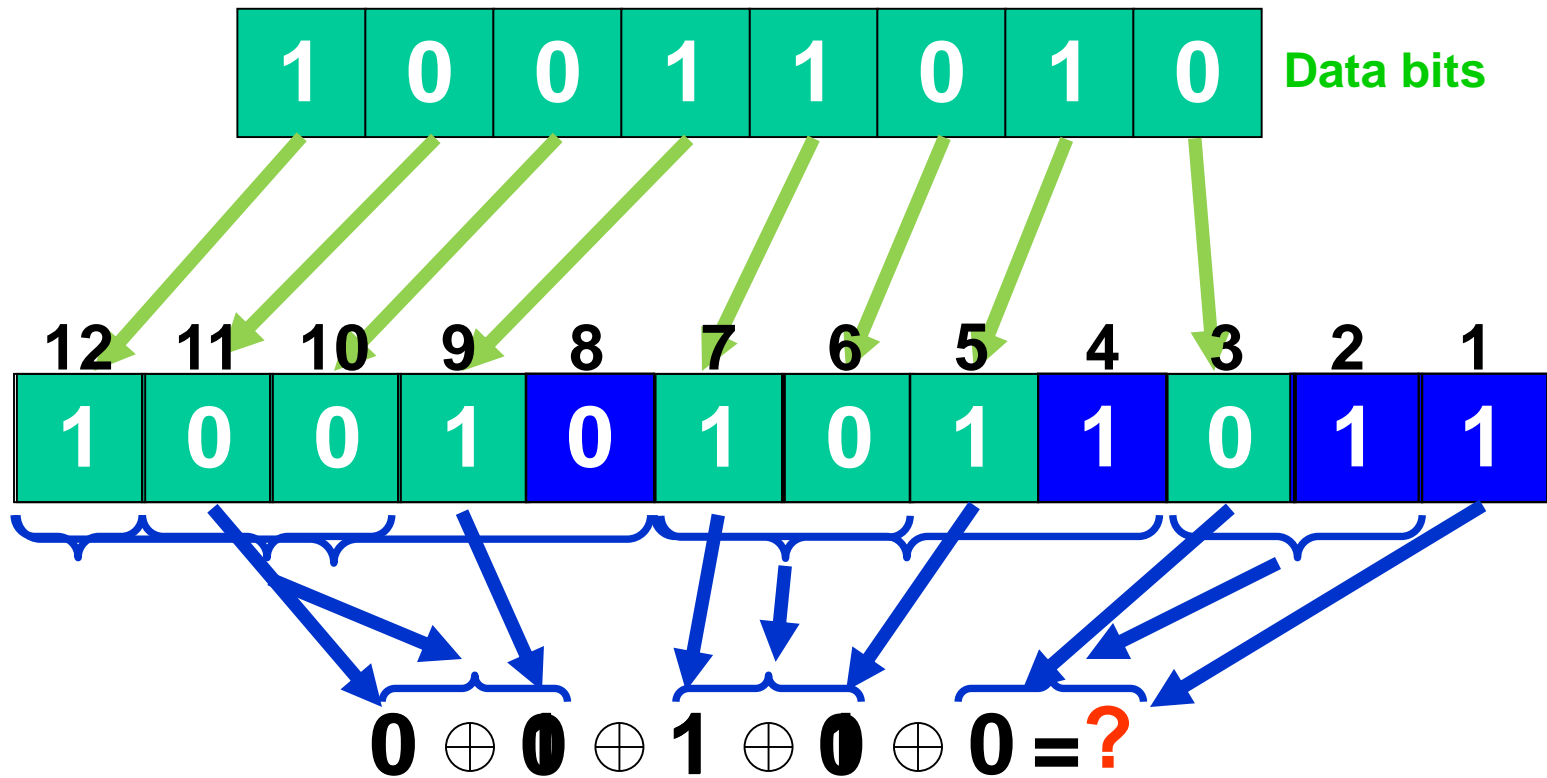
---

Design a code to correct all single bit errors.

- $M$  = data bits,  $K$  = check bits
- $N = M + K$
- **Rule:** Choose  $K$  s.t.  $M + K + 1 \leq 2^K$
- **Justification:**
  - Each of the  $2^M$  legal words has  $N$  illegal codewords at distance 1.
  - Thus, each of the  $2^M$  legal words requires  $(N + 1)$  bit patterns dedicated to it.
  - $(N + 1) 2^M \leq 2^N \rightarrow M + K + 1 \leq 2^K$

# Hamming Code

$$2^K - 1 \geq M + K \rightarrow 2^K \geq 9 + K \rightarrow K = 4$$



Bit position 8

Bit position 8: 0

## Hamming Code (2)

---

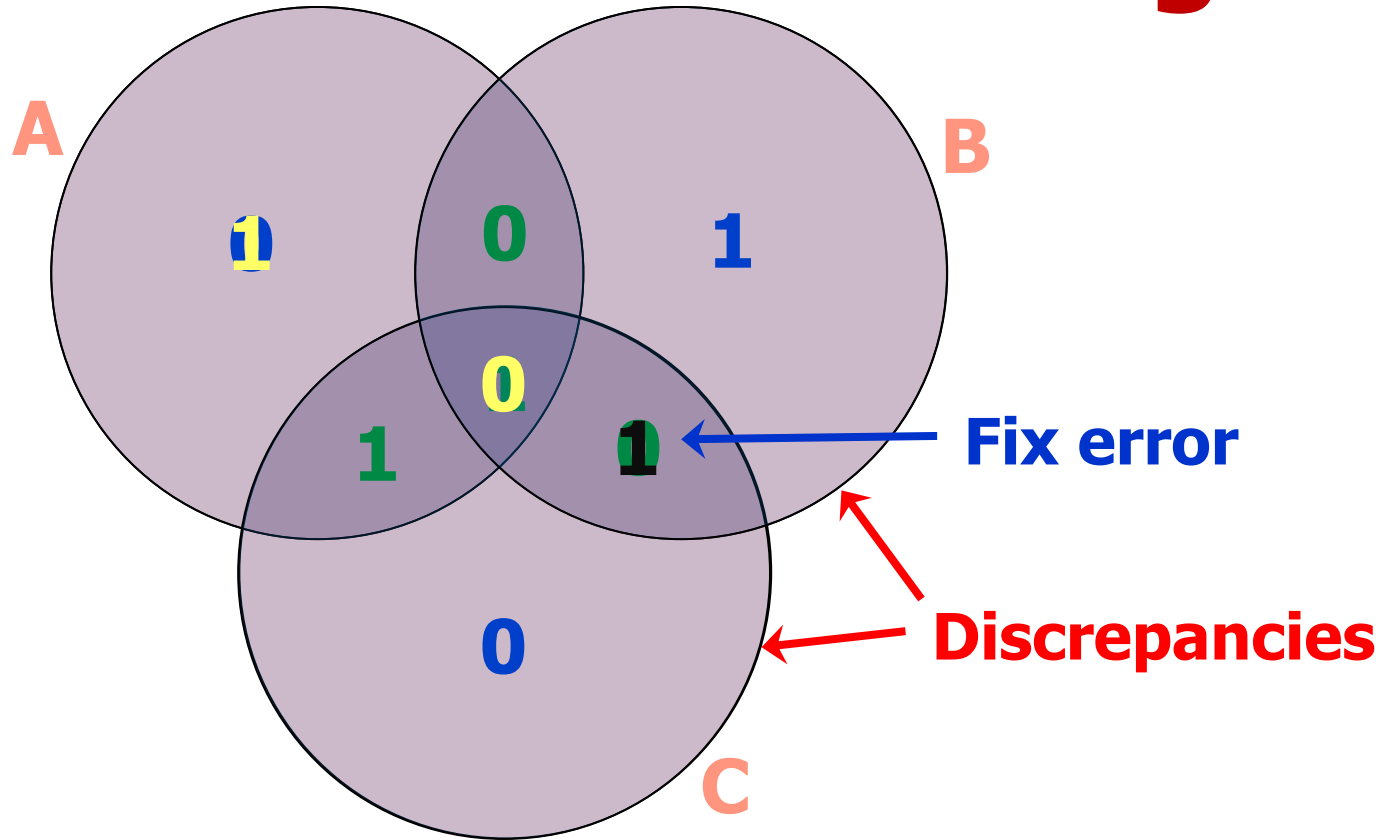
12	11	10	9	8	7	6	5	4	3	2	1
1	0	0	0	0	1	0	1	1	0	1	1

- Assume error in bit 9.
- Recompute the check bits.
- Bit 1 = 0 (**error**).
- Bit 2 = 1.
- Bit 4 = 1.
- Bit 8 = 1 (**error**).
- Error is in bit position = 1 + 8 = 9 → flip it (correction).

# Hamming SEC-DED Code

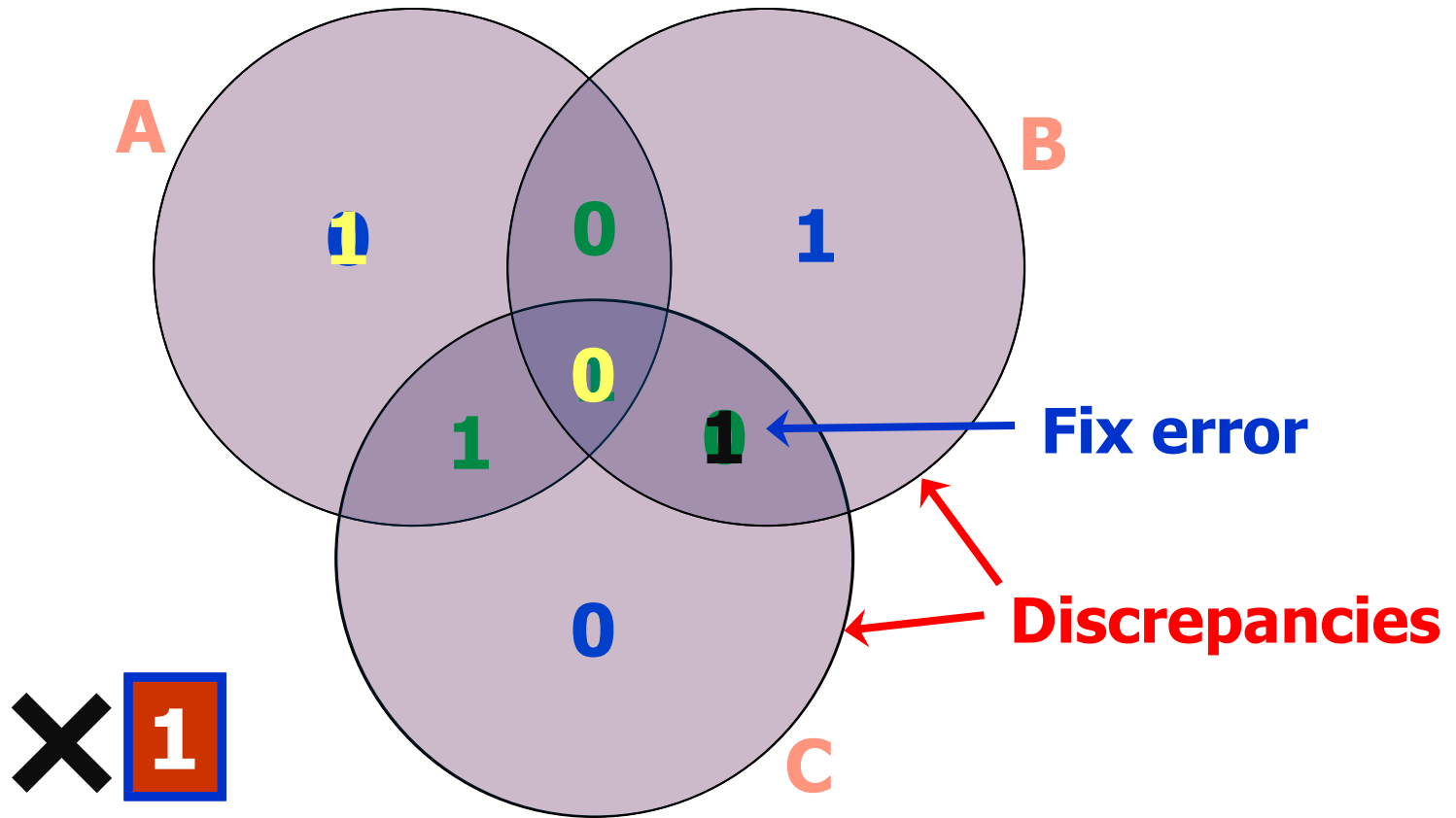
---

**Wrong!!!**



# Hamming SEC-DED Code (2)

---



# **Increase in Word Length with Error Correction**

	Single-Error Correction		Single-Error Correction/ Double-Error Detection	
Data Bits	Check Bits	% Increase	Check Bits	% Increase
8	4	50	5	62.5
16	5	31.25	6	37.5
32	6	18.75	7	21.875
64	7	10.94	8	12.5
128	8	6.25	9	7.03
256	9	3.52	10	3.91

# Advanced DRAM Organization

---

- Interface to MM is a system bottleneck.
- DRAM chip is the main building block of MM.
- Basic DRAM architecture same since 1970s!
- Enhancements to basic DRAM architecture
  - Synchronous DRAM (SDRAM)
    - DDR-SDRAM, DDR2-SDRAM, DDR3-SDRAM
  - Rambus DRAM (RDRAM)
  - Cache DRAM (CDRAM)

# Synchronous DRAM (SDRAM)

---



- Unlike traditional DRAM (which is asynchronous), SDRAM exchanges data with CPU synchronized to an external clock (system bus).
- No wait states!
  - SDRAM moves data in/out under control of system clock.
  - CPU issues command and address.
  - Latched by the SDRAM.
  - SDRAM responds after a number of clock cycles.
  - Meanwhile, CPU can do other tasks → no waits.

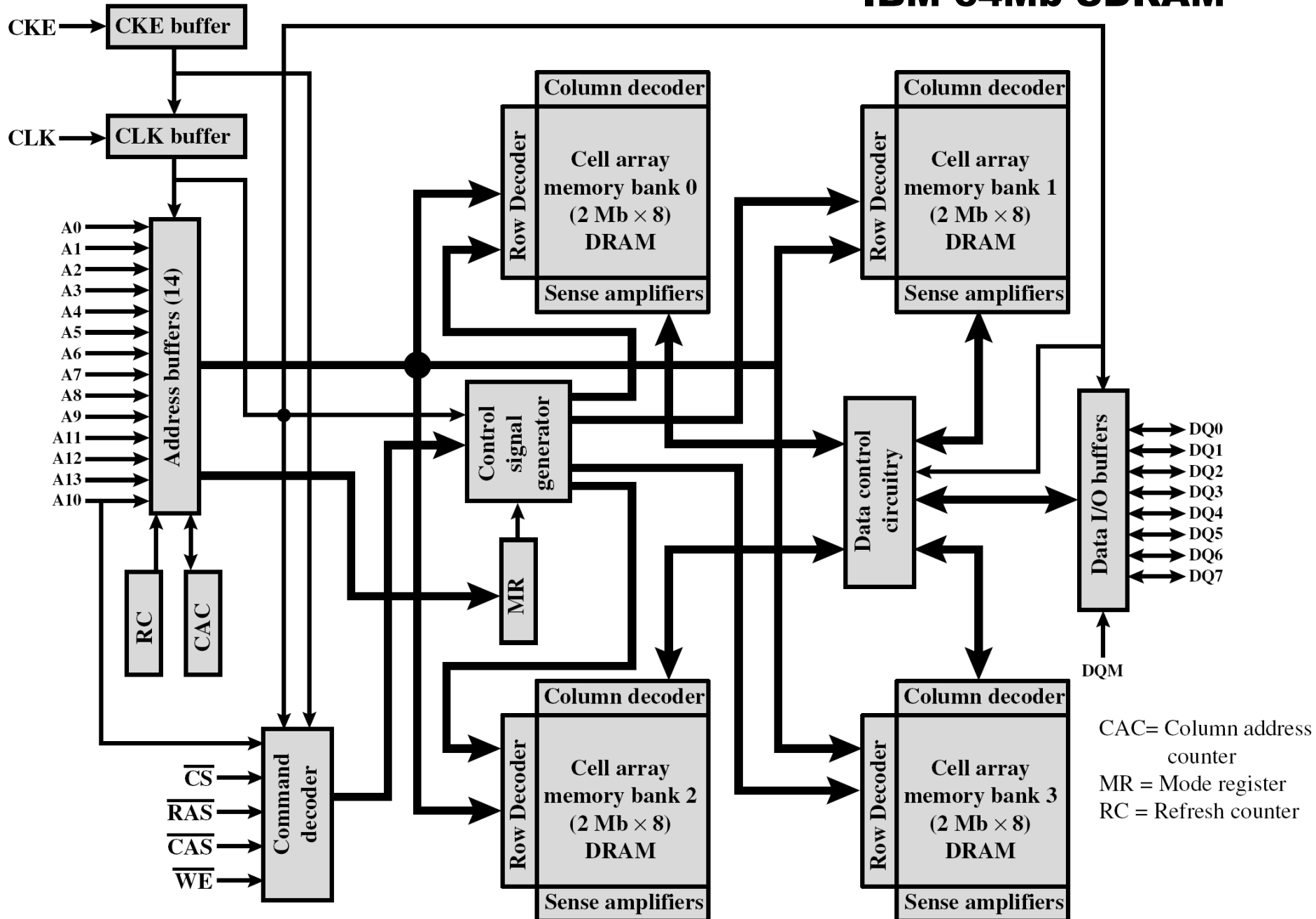


## **Synchronous DRAM (SDRAM) (2)**

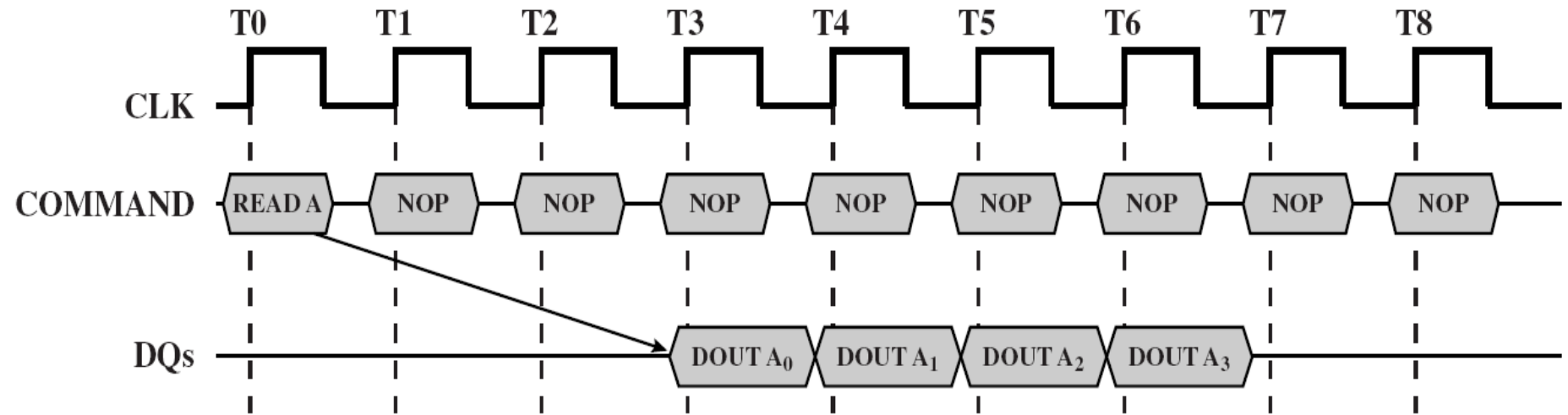
---

- **Burst mode**: a series of data bits can be clocked out rapidly after the first bit has been accessed.
- Eliminates row and column address setup time.
- Useful when the required bits are in sequence and in the same row of the array as the initial access.
- Multiple-bank internal architecture improves opportunities for parallelism.
- Mode Register (MR)
  - Specifies burst length.
  - Allows programmer to adjust latency between read request and data transfer.

# IBM 64Mb SDRAM



# SDRAM Operation



- Burst type, length and latency are set in mode reg.
  - Type: interleaved or sequential
  - In the example: length=4 and latency=3
- Burst read command is initiated.
  - At the rising edge of the clock: CS & CAS → low, and RAS & WE → high.
- Address inputs determine starting column for burst.

# Enhanced versions of SDRAM

---

- Double Data Rate (DDR-SDRAM)
  - produces two words of data every memory cycle
  - sends data to CPU twice per clock cycle (at leading & trailing edges of the clock)
  - 2x data rate of SDRAM (with same cell speed)
- DDR2-SDRAM
  - produces four words of data every memory cycle
  - 2x bus speed & data rate of DDR (with same cell speed)
- DDR3-SDRAM
  - produces eight words of data every memory cycle
  - 2x bus speed & data rate of DDR2 (with same cell speed)

# Rambus DRAM (RDRAM) (1)

---



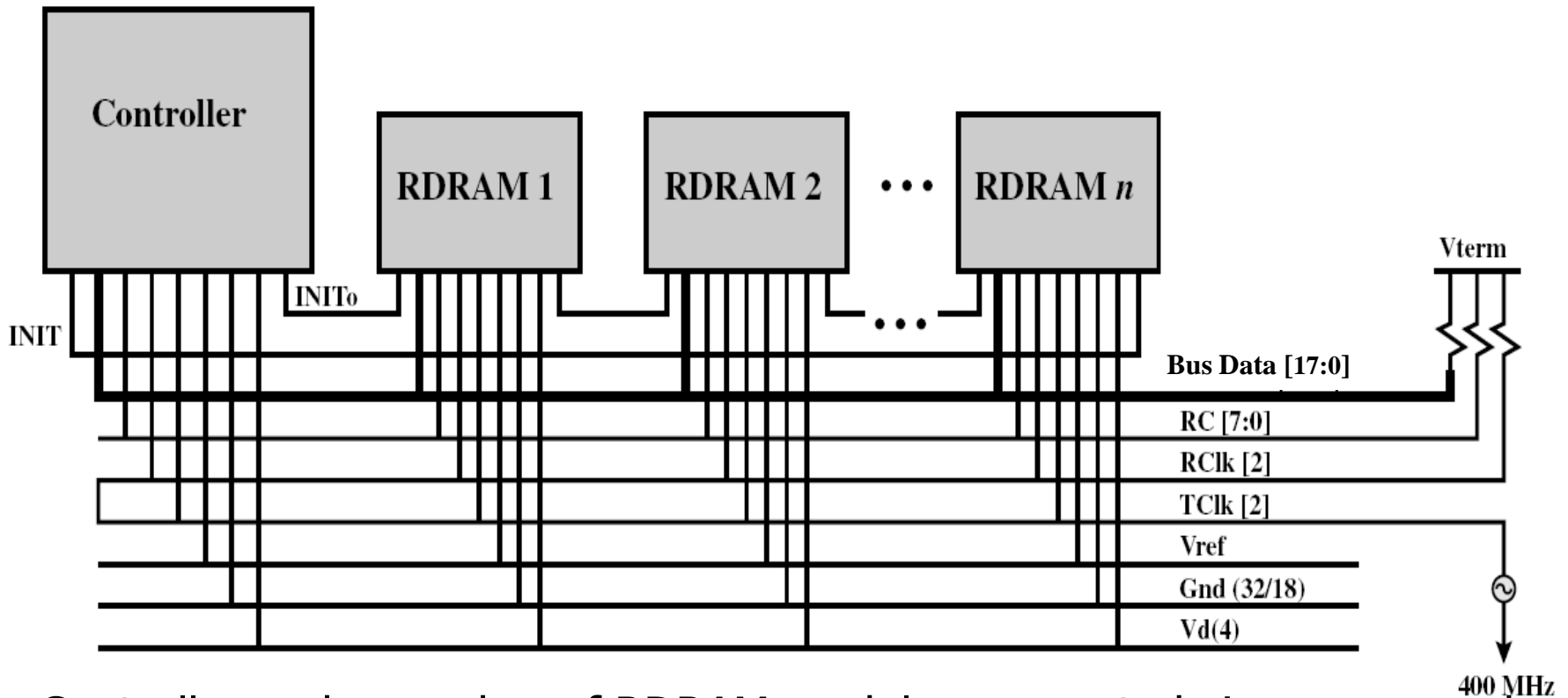
- Adopted by Intel for Pentium & Itanium in late 90s.
- Was main competitor to SDRAM.
- Chips are vertical packages – all pins on one side.
- Data exchange over 28 wires < 12 cm long.
- Bus can address up to 32 RDRAM chips and is rated at 1.6 GBps.

## **Rambus DRAM (RDRAM) (2)**

---

- Asynchronous block-oriented protocol
  - Initial 480 ns access time.
  - Then 1.6 GB/s.
- What makes this speed possible is the bus itself, which defines impedances, clocking, and signals very precisely.
- RDRAM gets a memory request over the high-speed bus (unlike conventional DRAMs controlled by RAS, CAS, R/W, and CE).
- This request contains address, operation, number of bytes.

# RDRAM Diagram



- Controller and a number of RDRAM modules connected via a common bus.
- Bus: 18 data lines, cycling at twice the clock rate → 800 Mbps per line.
- Address and control signals: 8 lines (RC)
- Clock starts at the far end from the controller, propagates to the controller end, then loops back.
- Module sends data to the controller synchronously to the clock to master.
- Controller sends data to a module synchronously with the clock in the opposite direction.

# Cache DRAM (CDRAM)

---

- CDRAM integrates a small SRAM cache (16 kb) onto a generic DRAM chip.
- CDRAM can be used in two ways:
  1. As a **true cache** with 64-bit lines.
  2. As a **buffer** to support serial access of a data block.
    - e. g., to refresh a bit-mapped screen, the CDRAM can **prefetch** the data from the DRAM into the SRAM buffer. Subsequent accesses to the chip results in accesses to the SRAM only.



# Reading Material

---

- Stallings, Chapter 5:
  - Pages 170 – 180