Computer Organization

Internal Memory

Semiconductor Memory Types

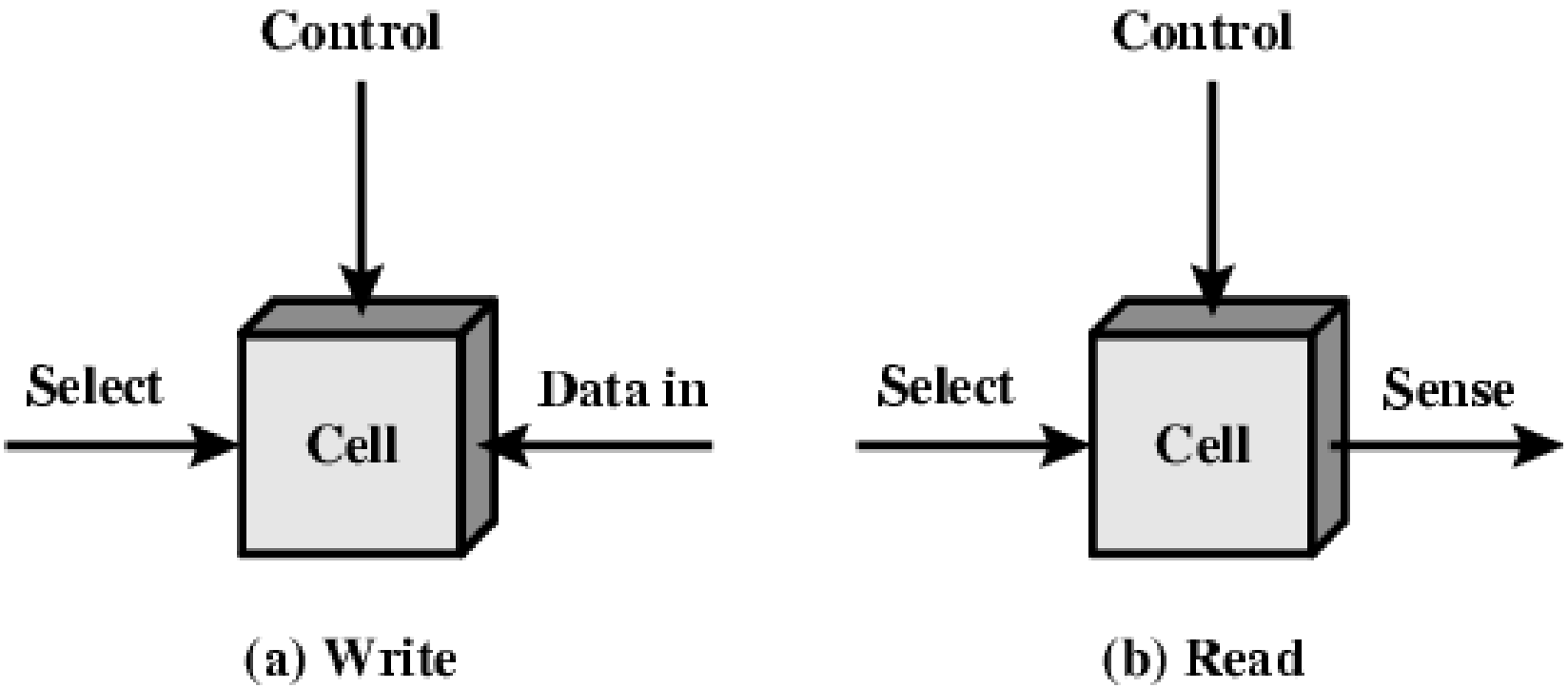
Table 5.1 Semiconductor Memory Types

Memory Type	Category	Erasure	Write Mechanism	Volatility
Random-access memory (RAM)	Read-write memory	Electrically, byte-level	Electrically	Volatile
Read-only memory (ROM)	Read-only memory	Not possible	Masks	Nonvolatile
Programmable ROM (PROM)			Electrically	
Erasable PROM (EPROM)	Read-mostly memory	UV light, chip-level		
Electrically Erasable PROM (EEPROM)		Electrically, byte-level		
Flash memory		Electrically, block-level		

Semiconductor Memory

- RAM
  - Misnamed as all semiconductor memory is random access
  - Read/Write
  - Volatile
  - Temporary storage
  - Static or dynamic

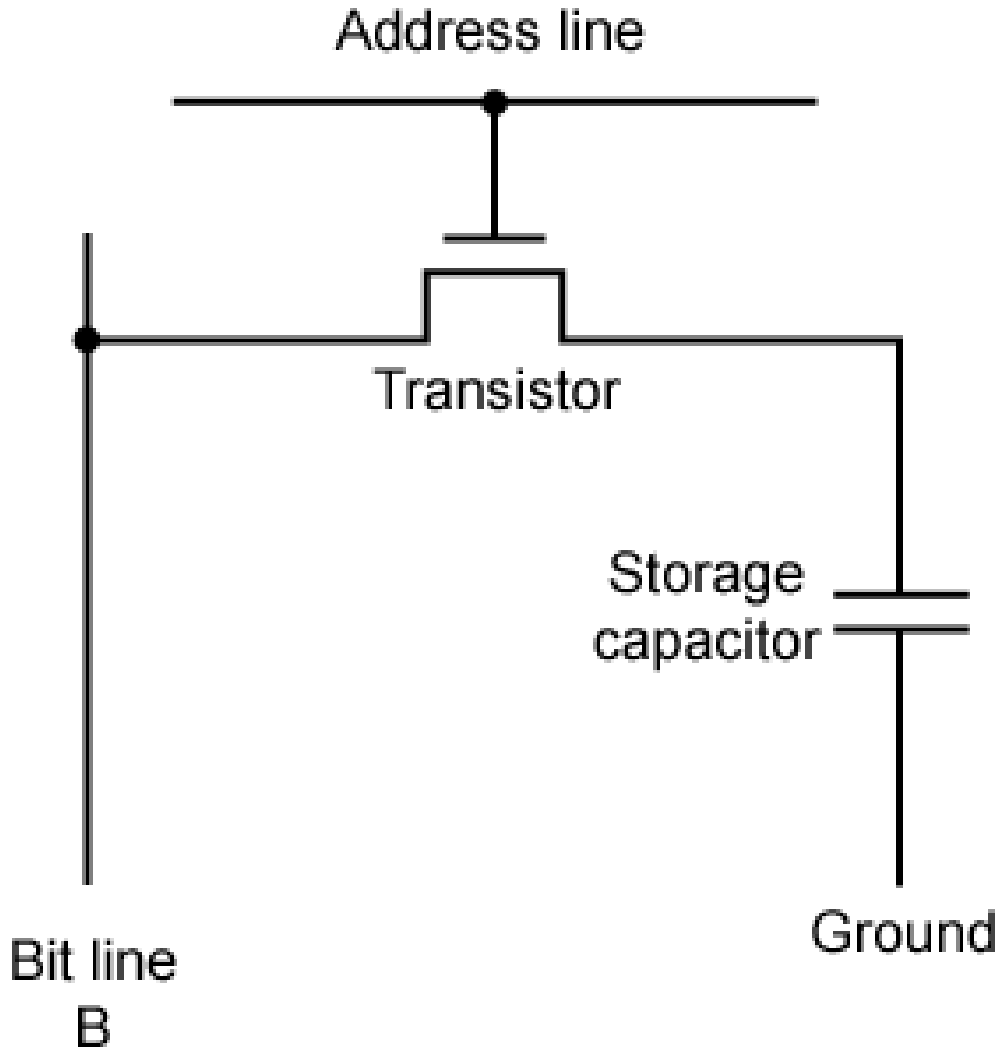
Memory Cell Operation



Dynamic RAM

- Bits stored as charge in capacitors
- Charges leak
- Need refreshing even when powered
- Simpler construction
- Smaller per bit
- Less expensive
- Need refresh circuits
- Slower
- Main memory
- Essentially analogue
  - —Level of charge determines value

Dynamic RAM Structure



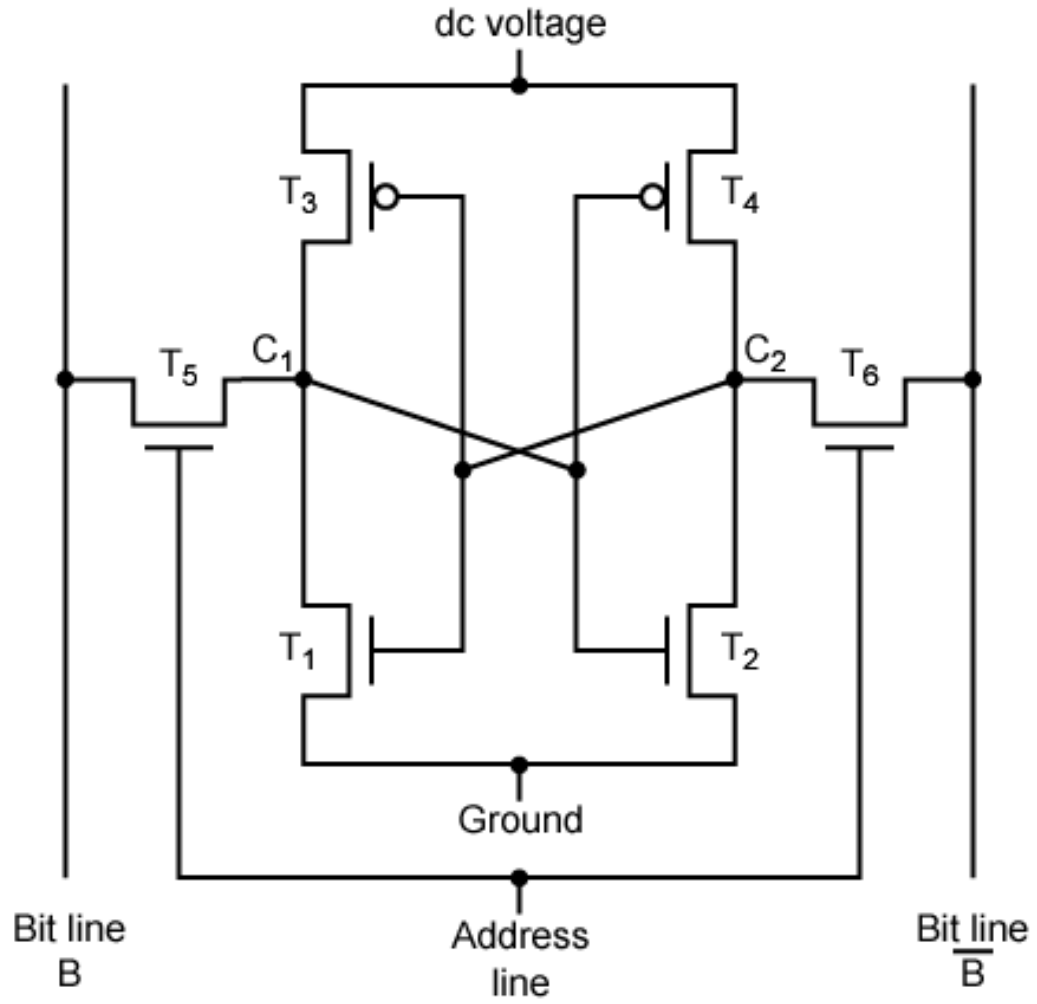
DRAM Operation

- Address line active when bit read or written
  - Transistor switch closed (current flows)
- Write
  - Voltage to bit line
    - High for 1 low for 0
  - Then signal address line
    - Transfers charge to capacitor
- Read
  - Address line selected
    - transistor turns on
  - Charge from capacitor fed via bit line to sense amplifier
    - Compares with reference value to determine 0 or 1
  - Capacitor charge must be restored

Static RAM

- Bits stored as on/off switches
- No charges to leak
- No refreshing needed when powered
- More complex construction
- Larger per bit
- More expensive
- Does not need refresh circuits
- Faster
- Cache
- Digital
  - —Uses flip-flops

Static RAM Structure



Static RAM Operation

- Transistor arrangement gives stable logic state
- State 1
  - C_1 high, C_2 low
  - T_1 T_4 off, T_2 T_3 on
- State 0
  - C_2 high, C_1 low
  - T_2 T_3 off, T_1 T_4 on
- Address line transistors T_5 T_6 is switch
- Write – apply value to B & compliment to B
- Read – value is on line B

SRAM v DRAM

- Both volatile
  - Power needed to preserve data
- Dynamic cell
  - Simpler to build, smaller
  - More dense
    - Smaller cells → more cells per unit area
  - Less expensive
  - Needs refresh
  - Used in larger memory units (RAM)
- Static
  - Faster
  - Used in Cache

Read Only Memory (ROM)

- Permanent storage
  - Nonvolatile
- Applications
  - Microprogramming (see later)
  - Library subroutines
  - Systems programs (BIOS)
  - Function tables

Types of ROM

- Written during manufacture
  - Very expensive for small runs
- Programmable (once)
  - PROM
  - Needs special equipment to program
- Read "mostly"
  - Erasable Programmable (EPROM)
    - Erased by UV
  - Electrically Erasable (EEPROM)
    - Takes much longer to write than read
  - Flash memory
    - Erase whole memory electrically

Organization in detail

- A 16Mbit chip can be organised as 1M of 16 bit words

- A 16Mbit chip can be organised as

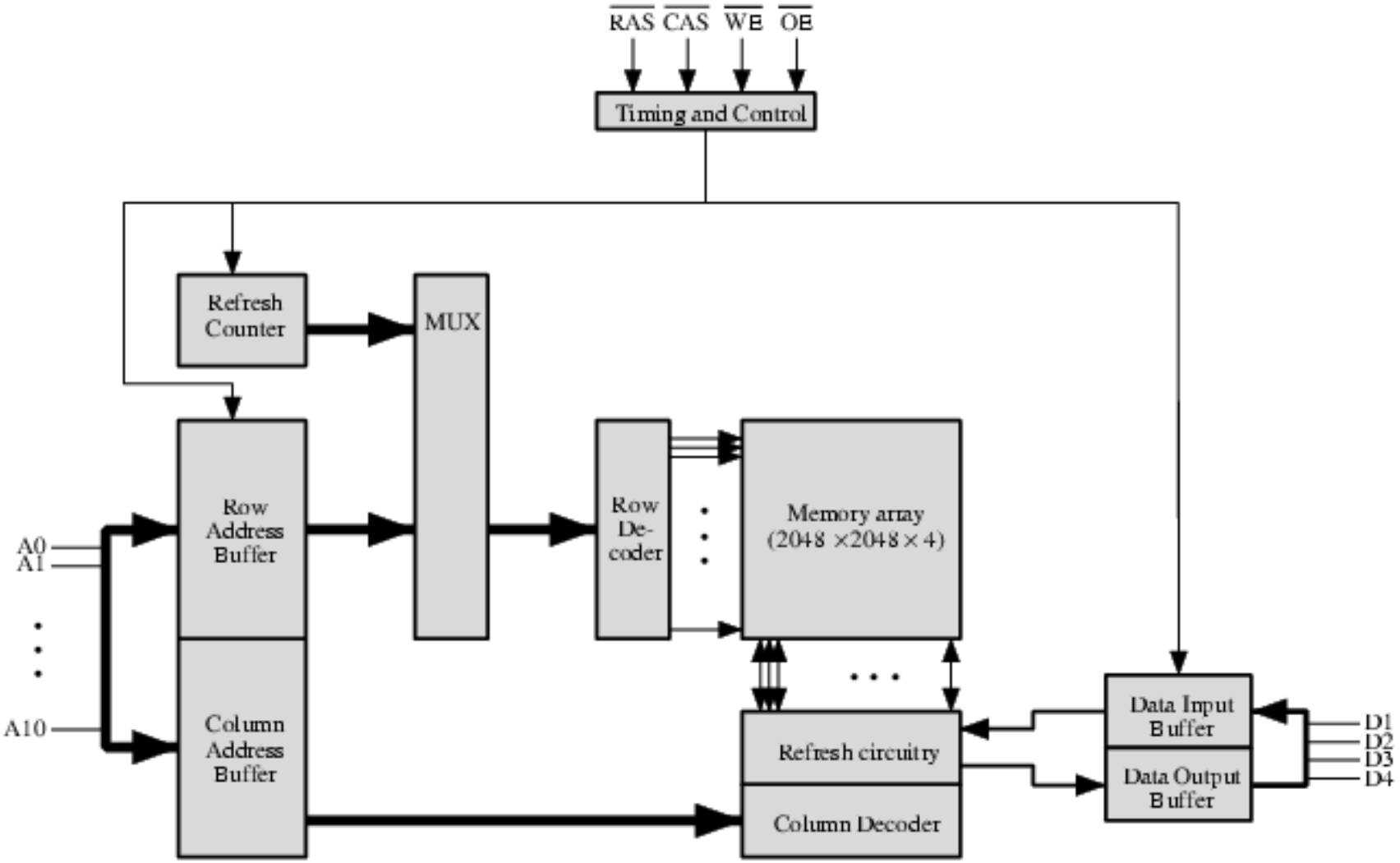
  2048 x 2048 x 4bit array

  - Reduces number of address pins
    - Multiplex row address and column address
    - 11 pins to address (2^{11}=2048)
    - Adding one more pin doubles range of values so x4 capacity

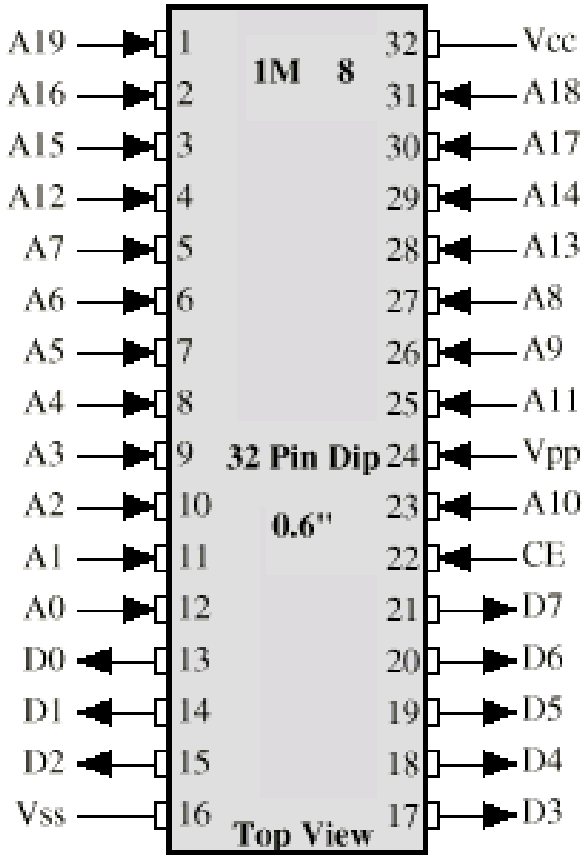
Refreshing

- Refresh circuit included on chip
- Disable chip
- Count through rows
- Read & Write back
- Takes time
- Slows down apparent performance

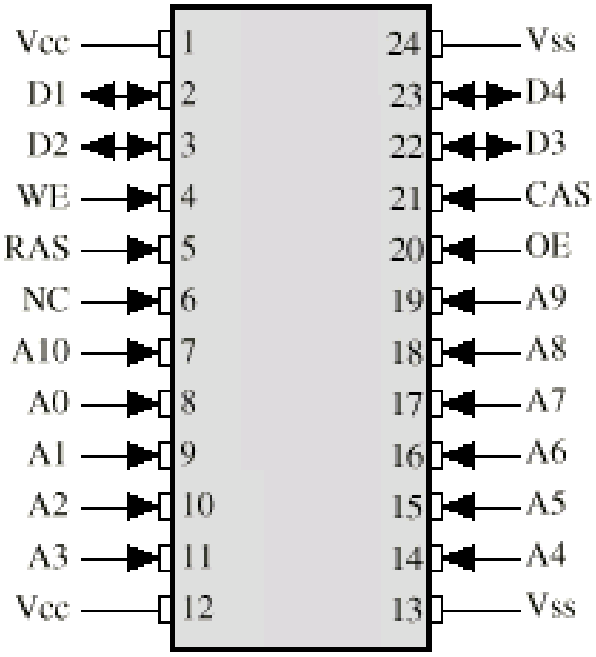
Typical 16 Mb DRAM (4M x 4)



Packaging



(a) 8 Mbit EPROM



(b) 16 Mbit DRAM

1M x 8

4M x 4

256kByte Module Organisation

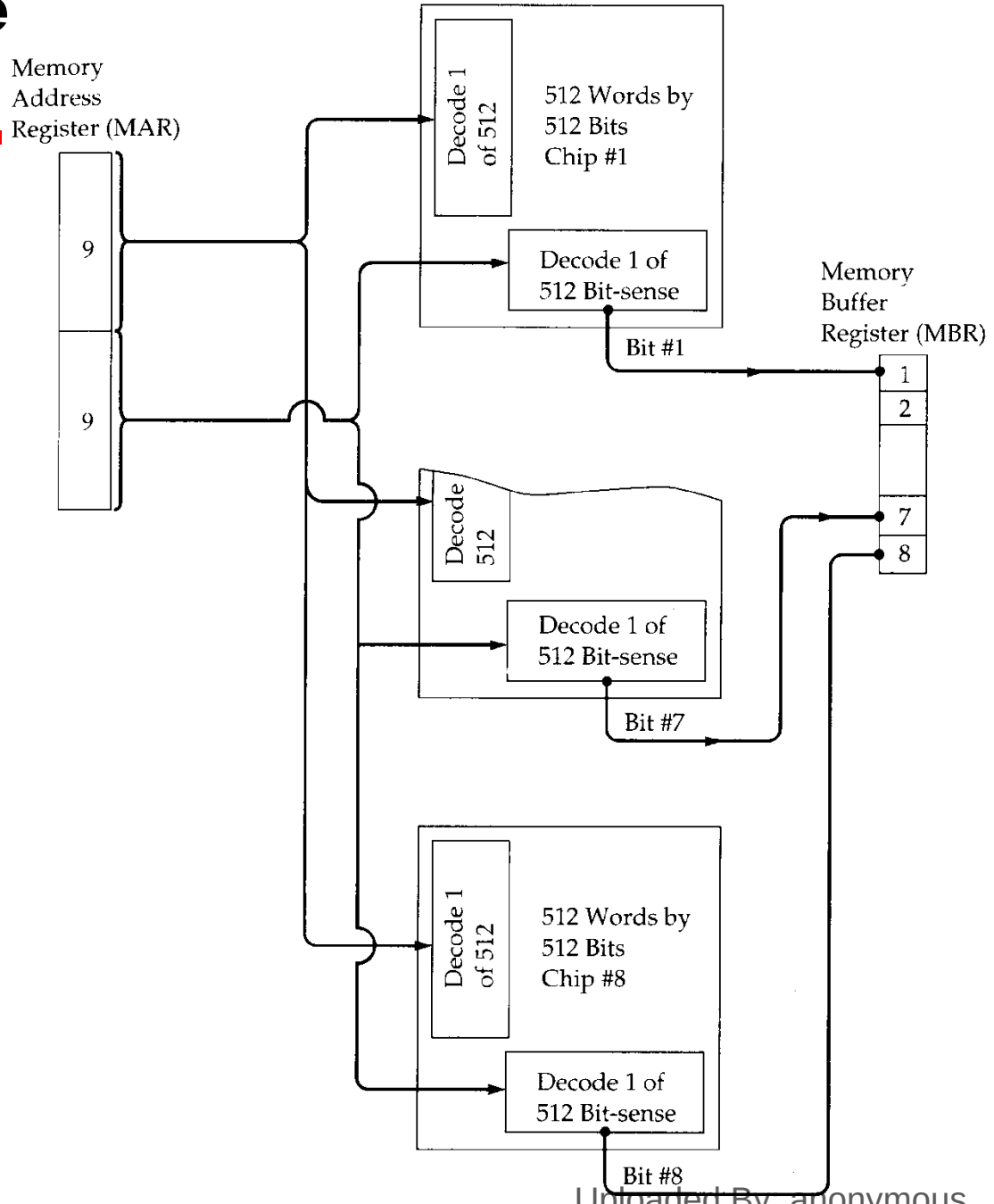
- Word size= 8 bit
- Memory organized as 8 chips
- Each chip with 1 bit word
- To read word from the memory, take 1 bit from each chip
- Size of each chip =

  512 x 512 x 1 bit
- 18 address lines needed
- Total size of memory =

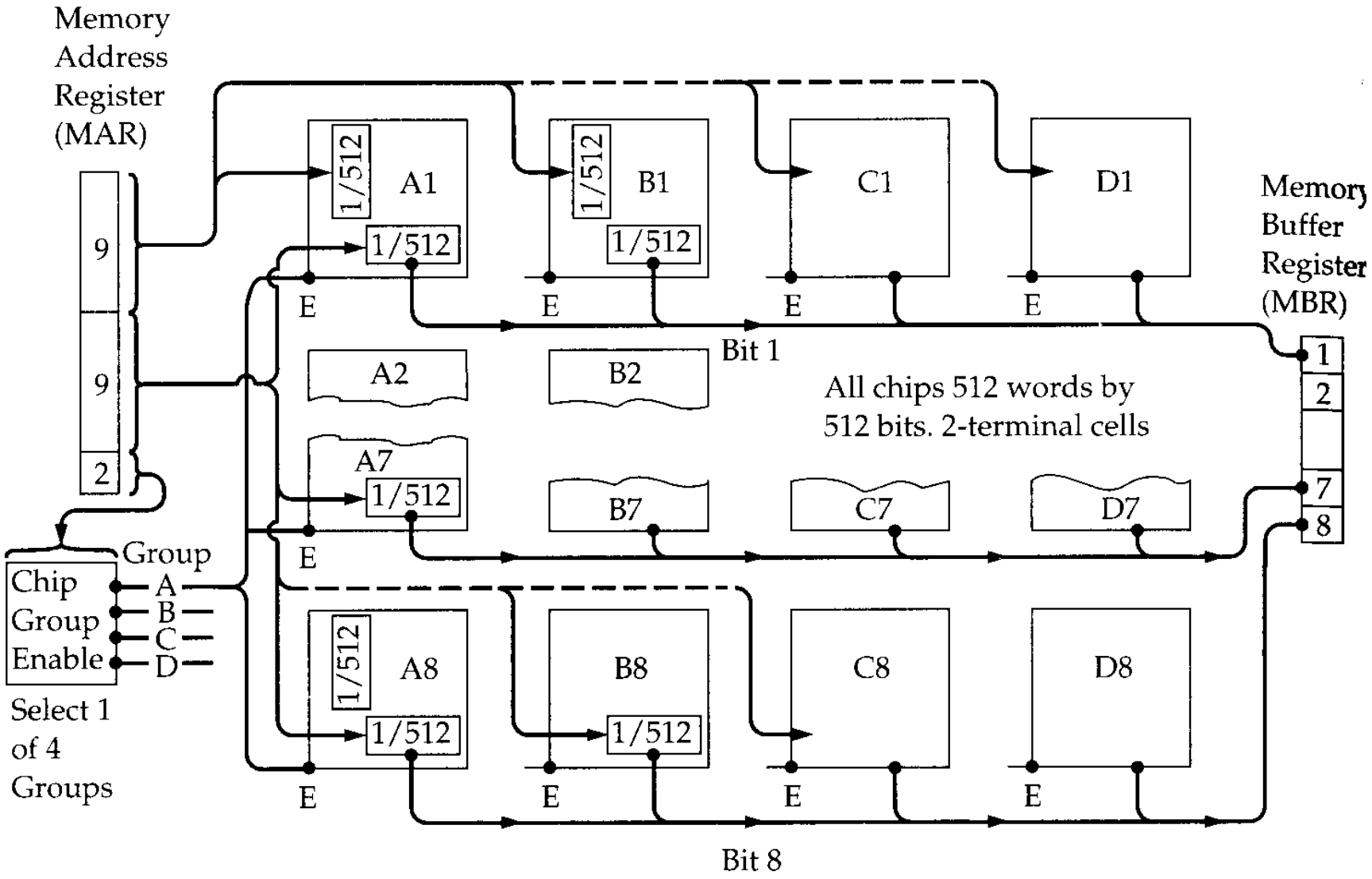
  512 x 512 x 8 chips

  = $2^9 \times 2^9 \times 2^3 = 2^{21}$ bit

  = 2^{18} byte or 256 kbyte



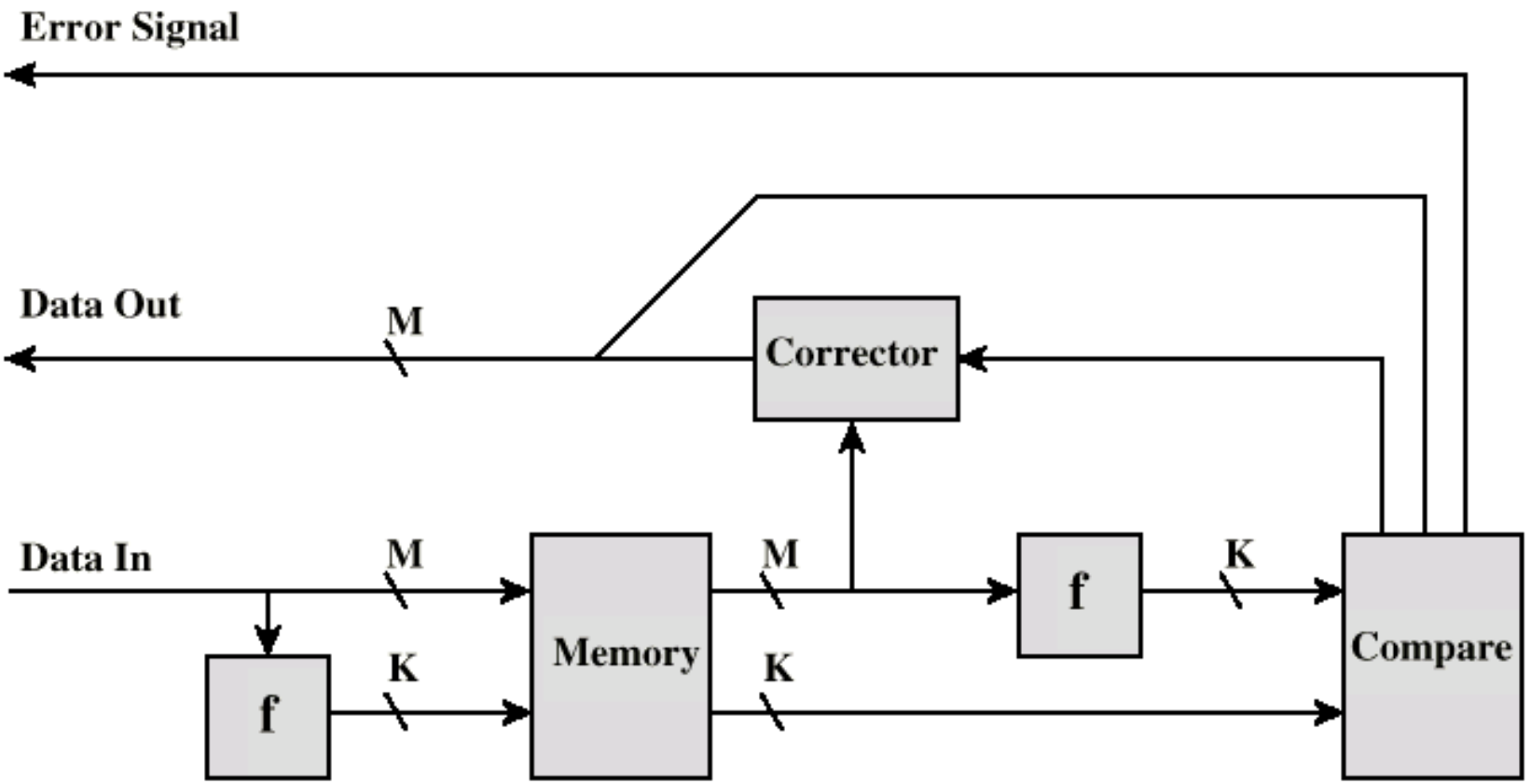
1MByte Module Organisation



Error Correction

- Two types of errors Hard failure and soft error
- Hard Failure
  - Permanent (physical) defect
- Soft Error
  - Random, non-destructive
  - No permanent damage to memory
- Detected using Hamming error correcting code

Error Correcting Code Function



Advanced DRAM Organization

- Basic DRAM same since first RAM chips
- Enhanced DRAM
  - —Contains small SRAM as well
  - —SRAM holds last line read (c.f. Cache!)
- Cache DRAM
  - —Larger SRAM component
  - —Use as cache or serial buffer

- SDRAM, RDRAM AND DDRAM

Synchronous DRAM (SDRAM)

- Access is synchronized with an external clock
- Address is presented to RAM
- RAM finds data (CPU waits in conventional DRAM)
- Since SDRAM moves data in time with system clock, CPU knows that data will be ready after a set number of clock cycle
- CPU does not have to wait, it can do something else
- DDR-SDRAM sends data twice per clock cycle (leading & trailing edge)

RAMBUS DRAM (RDRAM)

- Adopted by Intel for Pentium & Itanium
- Main competitor to SDRAM
- Vertical package – all pins on one side
- Data exchange over 28 wires < cm long
- Bus addresses up to 320 RDRAM chips at 1.6Gbps
- Asynchronous block protocol
  - —480ns access time
  - —Then 1.6 Gbps

DDR SDRAM

- SDRAM can only send data once per clock
- Double-data-rate SDRAM can send data twice per clock cycle
  - Rising edge and falling edge

Hamming Error Correction Example

Suppose an 8-bit word stored in memory is 1100 0010. Using the Hamming Algorithm, determine what check bits would be stored in memory with data word ?.

step ①: Determine the number of check bits needed:

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

k = # of check bits
M = length of word

Let's $k = 3$

$2^3 - 1 \not\geq 8 + 3$

Let's $k = 4$

$2^4 - 1 \geq 8 + 4$ ✓ $\Rightarrow$ so $k = 4$. (minimum k).

step ②

Code word = M + k
= 8 + 4
= 12 bits

Bit Positions	12	11	10	9	8	7	6	5	4	3	2	1
					2^3				2^2		2^1	2^0
Binary Rep.	1100	1011	1010	1001	1000	0111	0110	0101	0100	0011	0010	0001
	D_8	D_7	D_6	D_5	C_8	D_4	D_3	D_2	C_4	D_1	C_2	C_1

Hamming Error Correction Example

Step ⑤ : Check bits Calculation :

C_1 = XOR all Data bits XXX1

$= D_1 \oplus D_2 \oplus D_4 \oplus D_5 \oplus D_7$

C_2 = XOR all Data bits XX1X

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

C_4 = XOR all Data bits X1XX

$= D_2 \oplus D_3 \oplus D_8 \oplus D_8$

C_8 = XOR all Data bits 1XXX $= D_5 \oplus D_6 \oplus D_7 \oplus D_8$

$D_8\ D_7\ D_6\ D_5\ D_4\ D_3\ D_2\ D_1$

1 1 0 0 0 0 1 0

A	B	F
0	0	0
0	1	1
1	0	1
1	1	6

$C_1 = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 1 = 0$

$C_2 = 0 \oplus 0 \oplus 0 \oplus 0 \oplus 1 = 1$

$C_4 = 1 \oplus 0 \oplus 0 \oplus 1 = 0$

$C_8 = 0 \oplus 0 \oplus 1 \oplus 1 = 0$

$C_8\ C_4\ C_2\ C_1$

0 0 1 0

⇒ Code Word Stored in memory: $D_8\ D_7\ D_6\ D_5\ C_8\ D_4\ D_3\ D_2\ C_4\ D_1\ C_2\ C_1$

1 1 0 0 0 0 0 1 0 0 1 0

C 1 2 H

Hamming Error Correction Example

* Suppose now that data bit 3 sustains an error and changed from 0 to 1 When check bits re-calculated, we have.

$C_1 = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 1 = 0$
$C_2 = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 1 = 0$
$C_4 = 1 \oplus 1 \oplus 0 \oplus 1 = 1$
$C_8 = 0 \oplus 0 \oplus 1 \oplus 1 = 0$

} new check bits

⇒ When the new check bits compared with stored (old) check bits, the syndrome word is formed:

	C_8	C_4	C_2	C_1
⊕	0	0	1	0
	0	1	0	0
	0	1	1	0

⇒ result is 0110 indicating that bit at position 6 (i.e. D_3) is in error.