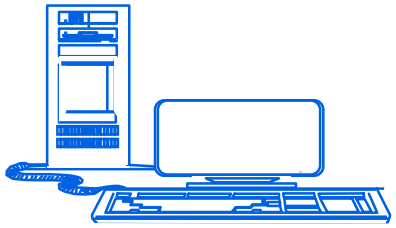
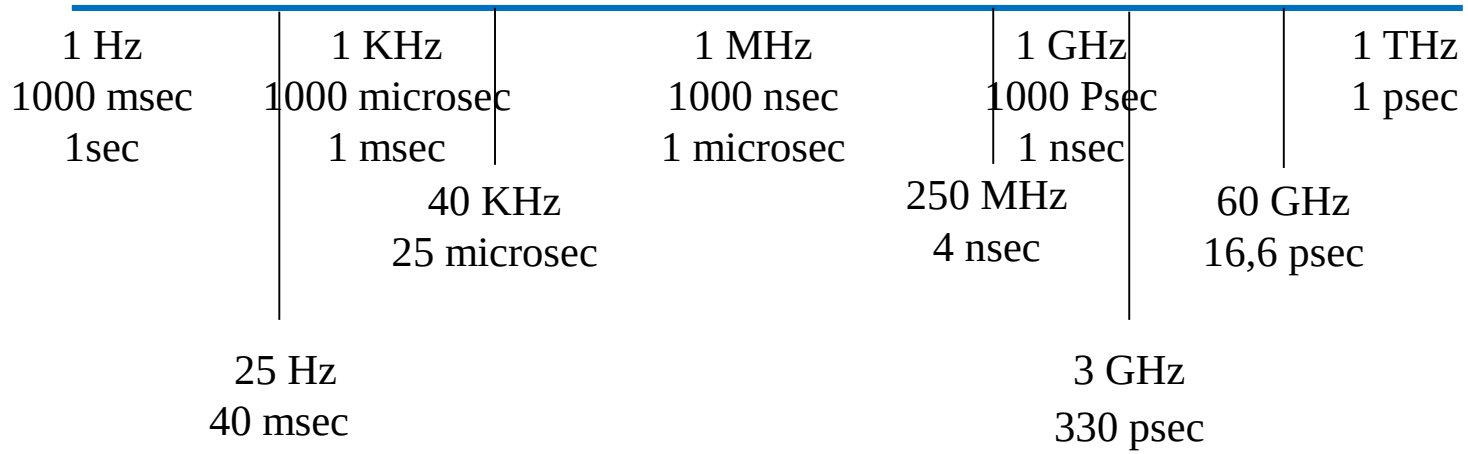
Memory Systems

Computer Architectures M

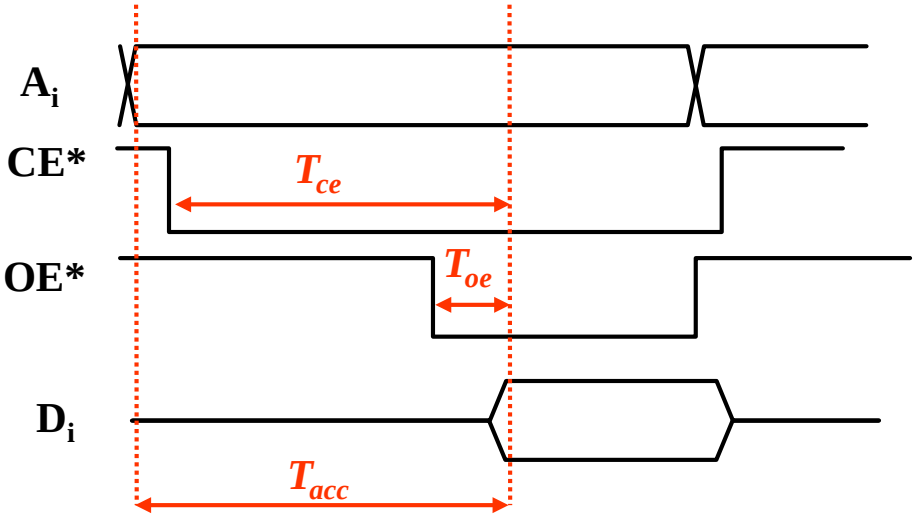


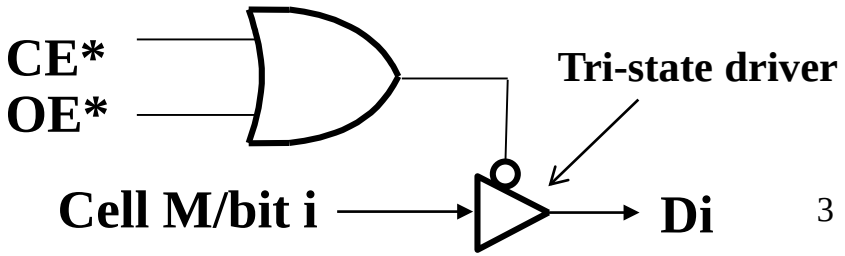
Frequencies vs. periods



EPROM memories

- Non volatile read-only memories
- Capacities: two's multiples: 32K, 64K, 128K, 256K……
- Access time: 50-80 ns





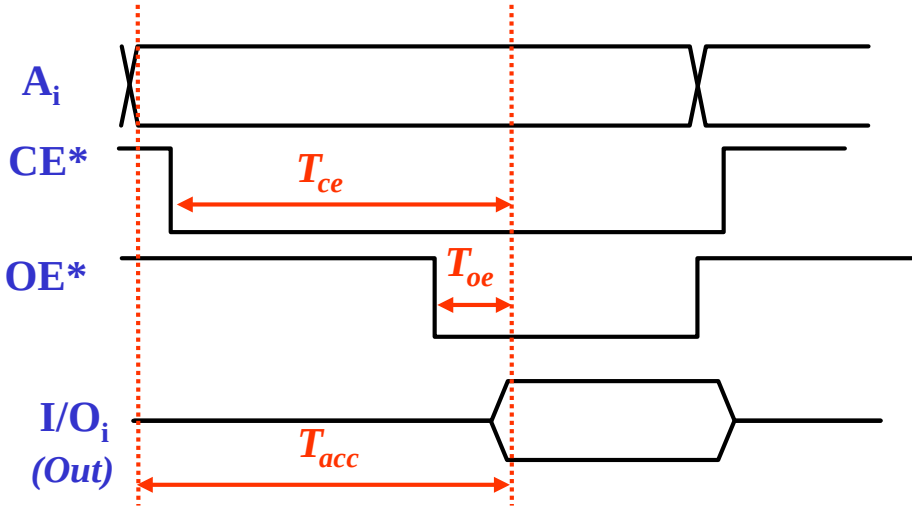
RAM memories(*SRAM*)

RAM

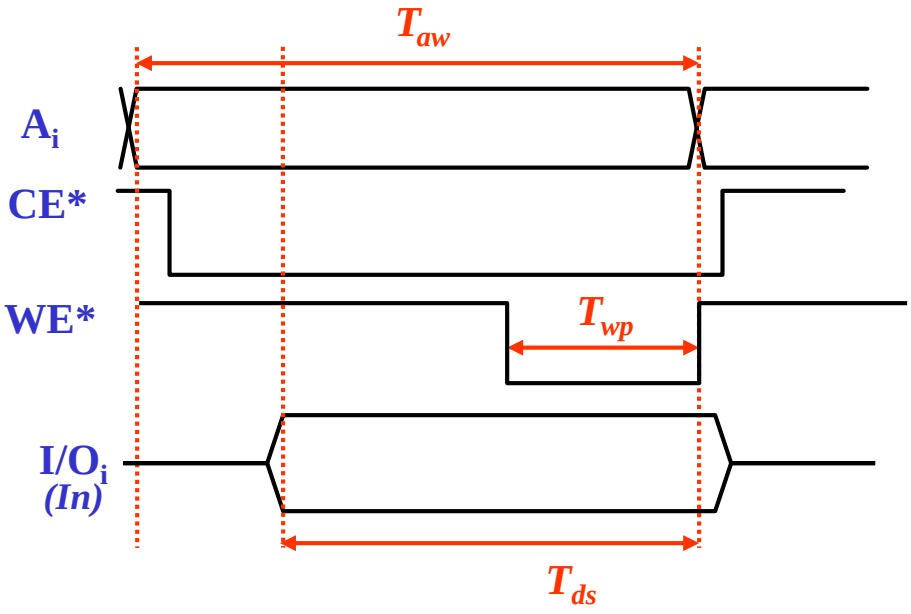
1	NC	VCC	32
2	A16	A15	31
3	A14	NC	30
4	A12	WE*	29
5	A7	A13	28
6	A6	A8	27
7	A5	A9	26
8	A4	A11	25
9	A3	OE*	24
10	A2	A10	23
11	A1	CE*	22
12	A0	I/O7	21
13	I/O0	I/O6	20
14	I/O1	I/O5	19
15	I/O2	I/O4	18
16	GND	I/O3	17

128K × 8

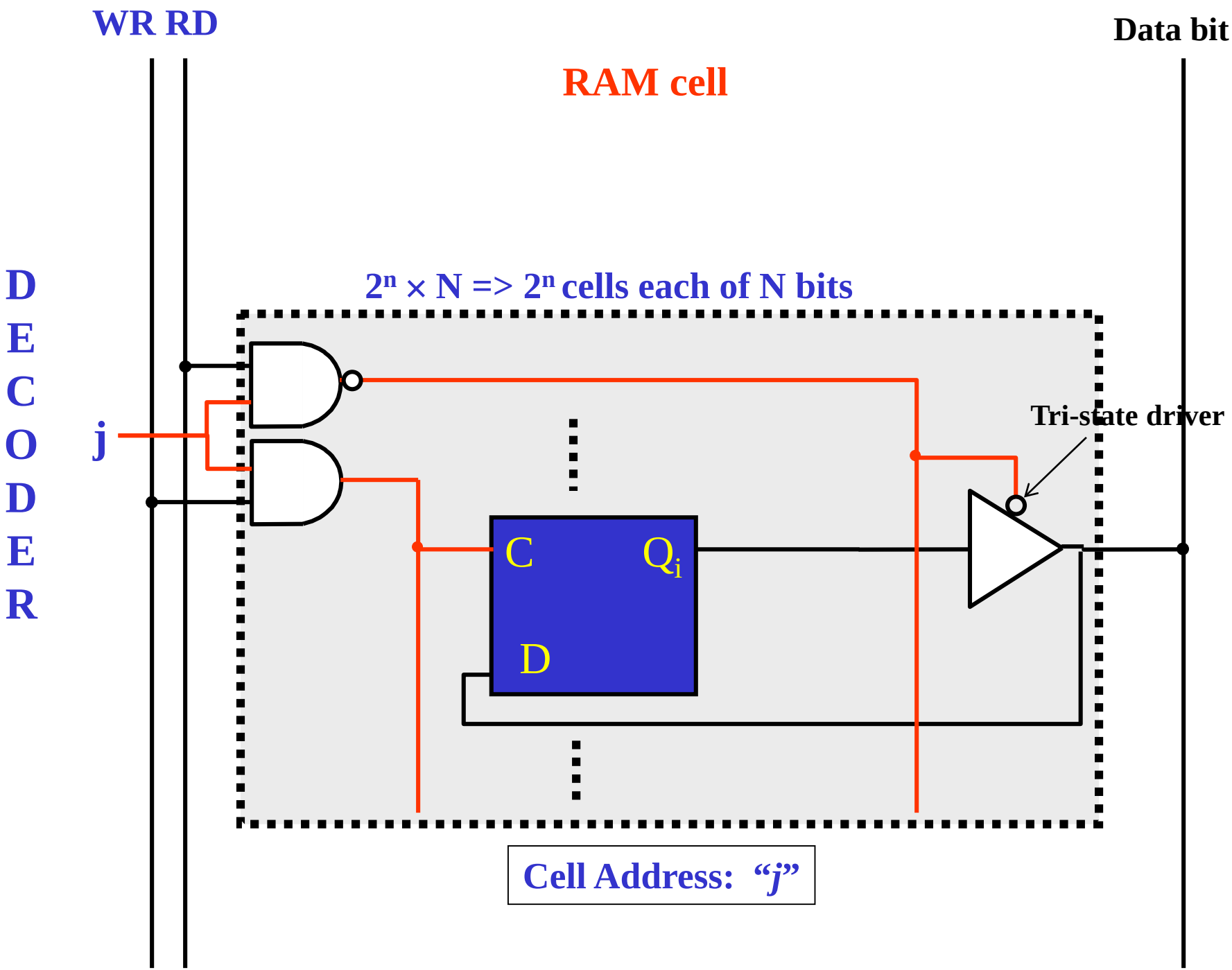
Read Cycle



Write Cycle



- Volatiles memories, readable and writable
- Capacity four's multiples: 8K, 32K, 128K, 512K……
- Access time: 5-40 ns
- *DRAM: 1 transistor per bit, higher capacity, less speed*

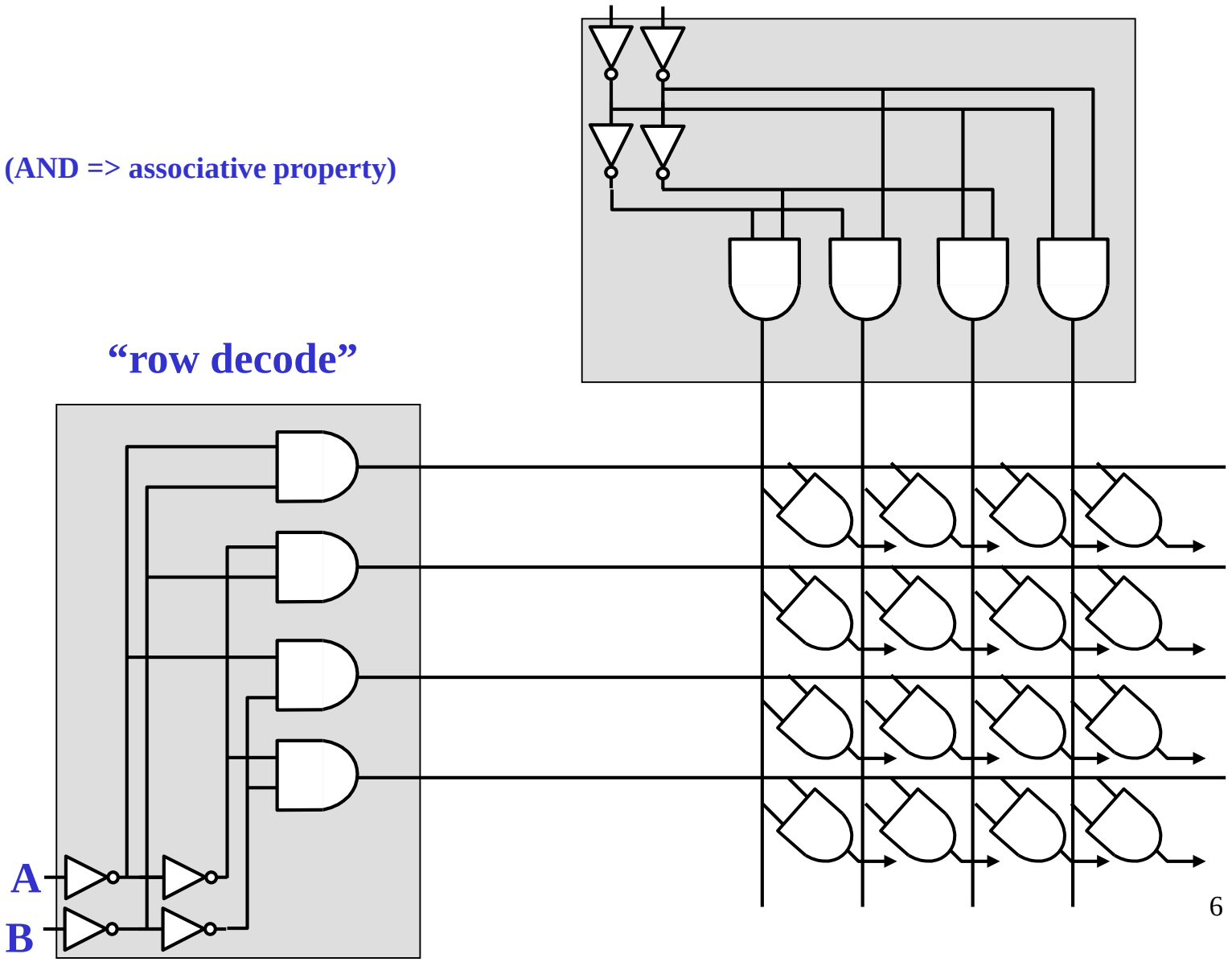

WR RD
Data bit
RAM cell
DECODER
2n × N => 2n cells each of N bits
j
Tri-state driver
C
Qi
D
Cell Address: "j"

Decoder and Decoder matrix

C D “column decode”

(AND => associative property)

“row decode”



32K x 8 Static RAM

Functional Description

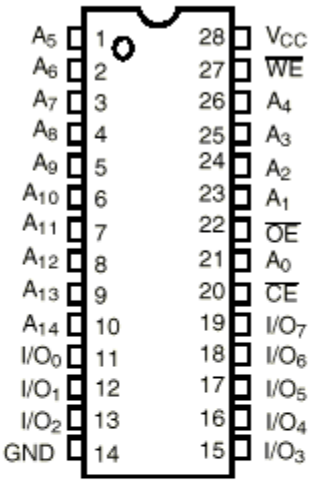
The CY7C199 is a high-performance CMOS static RAM organized as 32,768 words by 8 bits. Easy memory expansion is provided by an active LOW chip enable ($\overline{CE}$) and active LOW output enable ($\overline{OE}$) and three-state drivers. This device has an automatic power-down feature, reducing the power consumption by 81% when deselected. The CY7C199 is in the standard 300-mil-wide DIP, SOJ, and LCC packages.

An active LOW write enable signal ($\overline{WE}$) controls the writing/reading operation of the memory. When $\overline{CE}$ and $\overline{WE}$ inputs are both LOW, data on the eight data input/output pins (I/O_0 through I/O_7) is written into the memory location addressed by the address present on the address pins (A_0 through A_{14}). Reading the device is accomplished by selecting the device and enabling the outputs, $\overline{CE}$ and $\overline{OE}$ active LOW, while $\overline{WE}$ remains inactive or HIGH. Under these conditions, the contents of the location addressed by the information on address pins are present on the eight data input/output pins.

The input/output pins remain in a high-impedance state unless the chip is selected, outputs are enabled, and write enable ($\overline{WE}$) is HIGH. A die coat is used to improve alpha immunity.

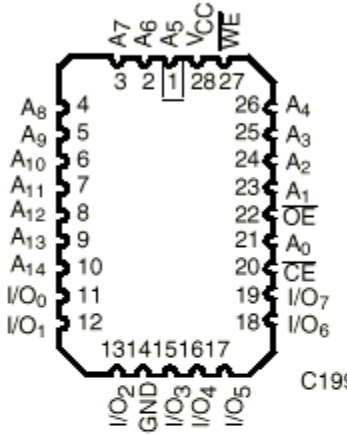
Pin Configurations

DIP / SOJ / SOIC
Top View

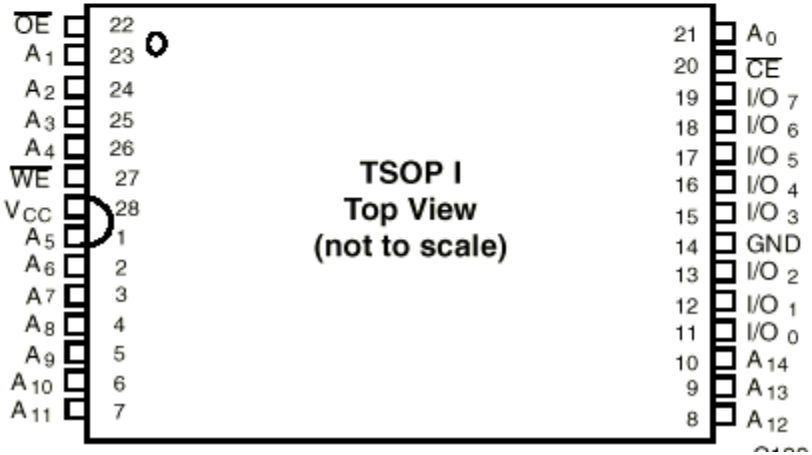


C199–2

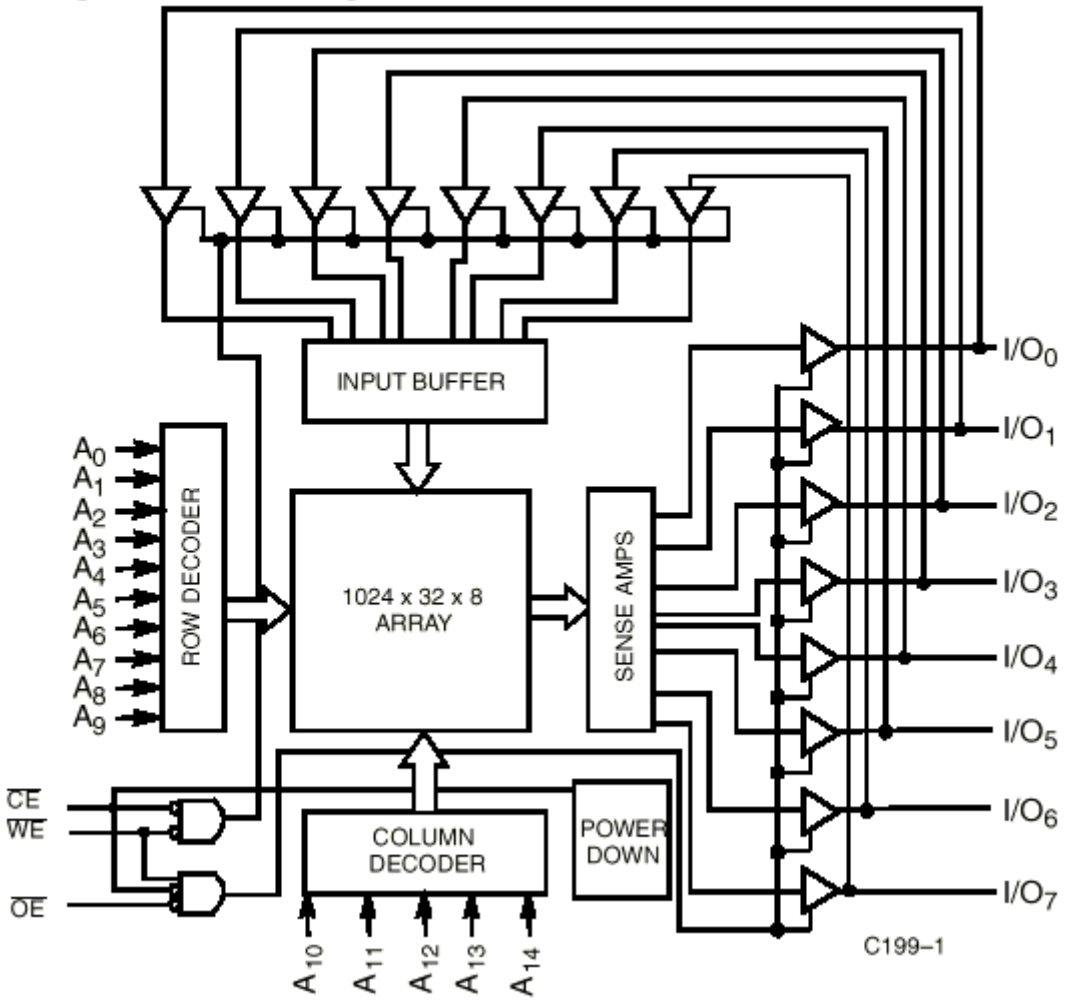
LCC
Top View



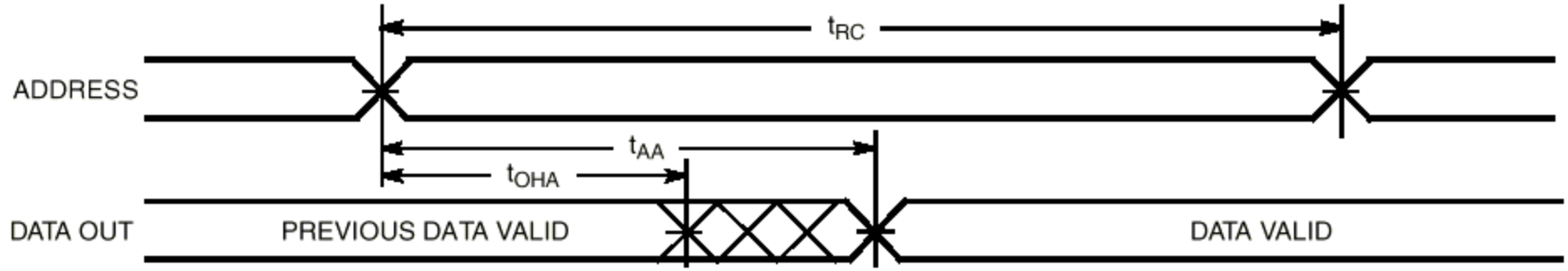
C199–3



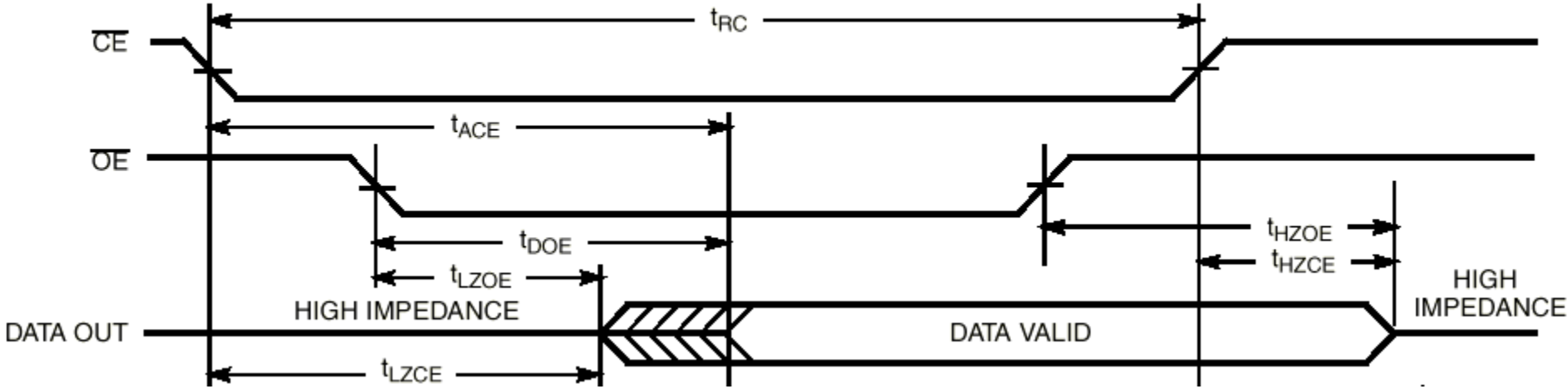
Logic Block Diagram



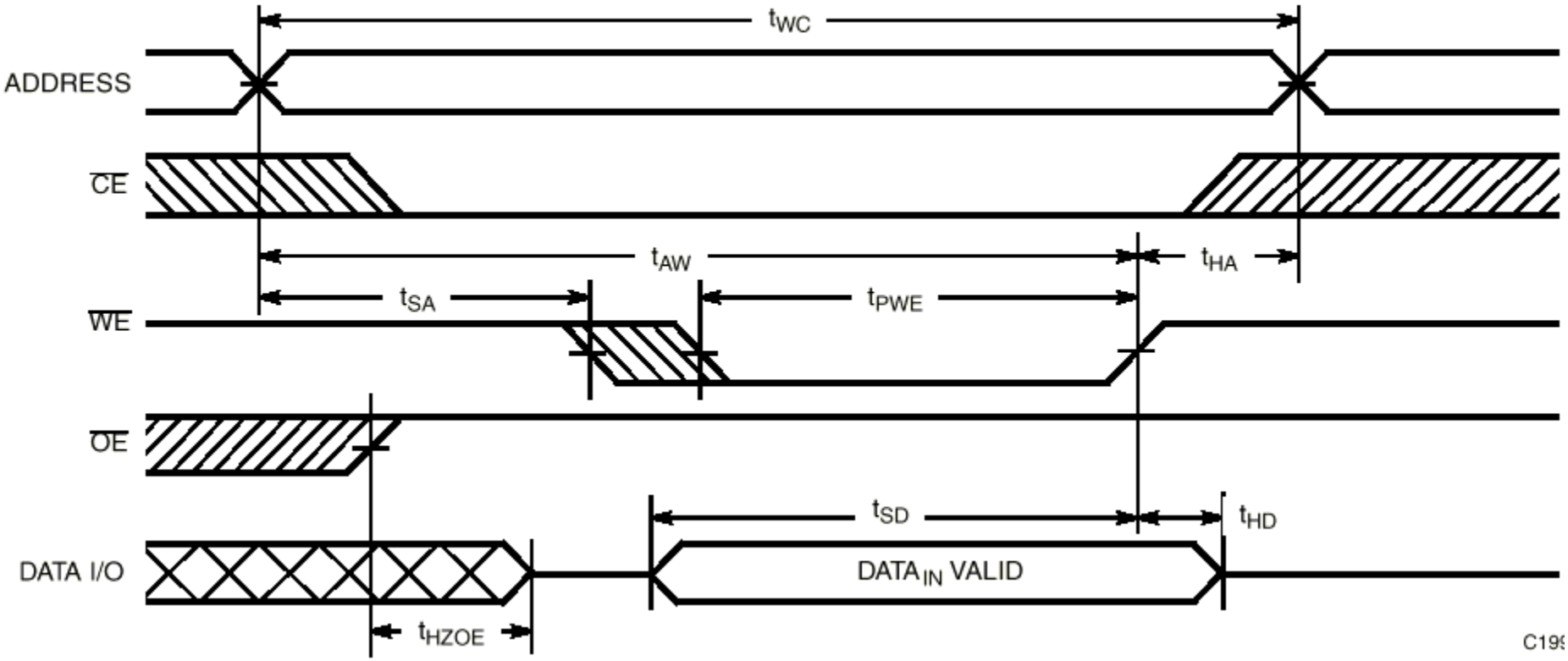
Read Cycle No. 1[12, 13]



Read Cycle No. 2 [13, 14]



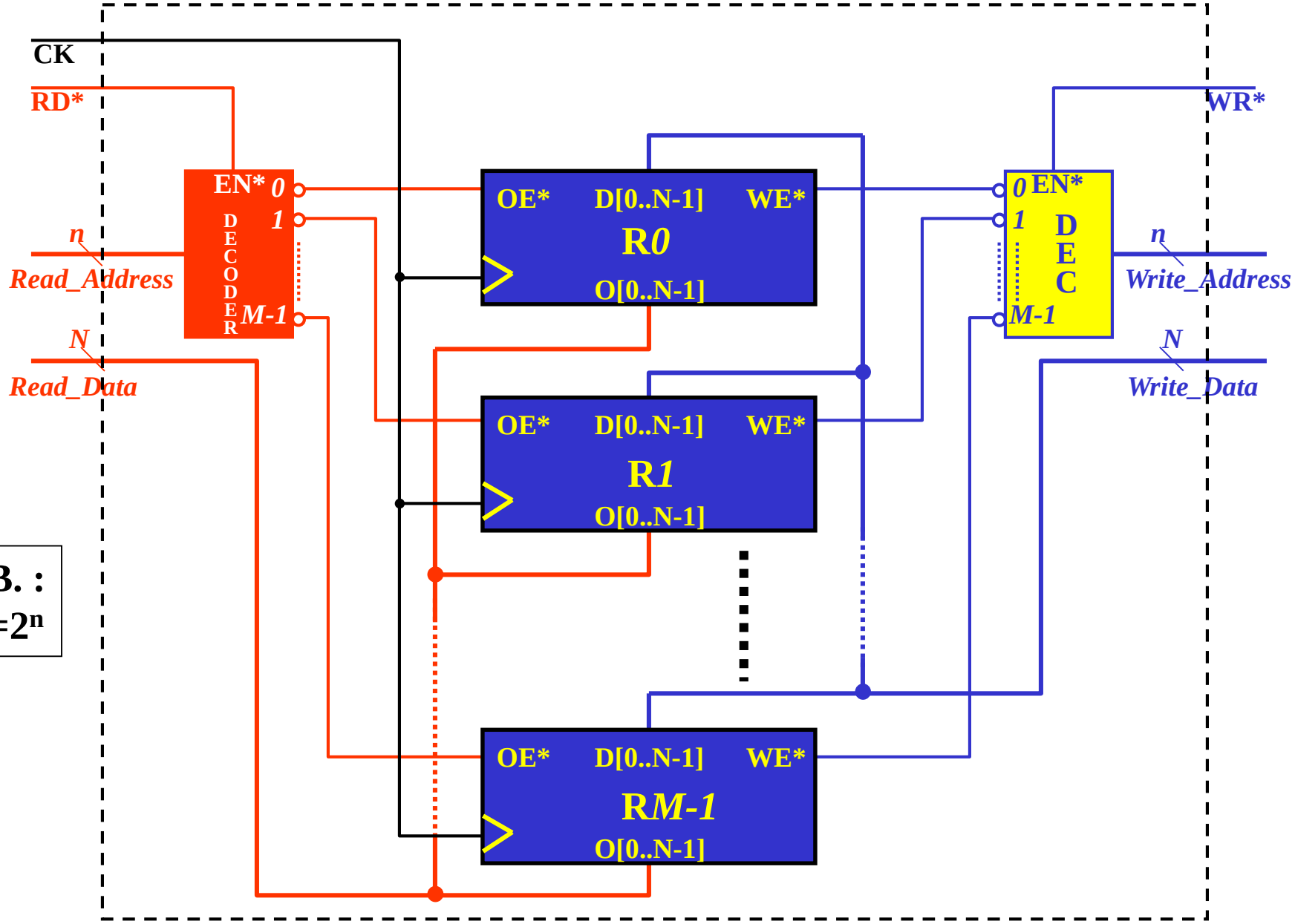
Write Cycle No. 1 (WE Controlled)[10, 15, 16]



Switching Characteristics Over the Operating Range[3,7] (continued)

Parameter	Description	7C199-20		7C199-25		7C199-35		7C199-45		Unit
		Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	
READ CYCLE										
t_{RC}	Read Cycle Time	20		25		35		45		ns
t_{AA}	Address to Data Valid		20		25		35		45	ns
t_{OHA}	Data Hold from Address Change	3		3		3		3		ns
t_{ACE}	$\overline{CE}$ LOW to Data Valid		20		25		35		45	ns
t_{DOE}	$\overline{OE}$ LOW to Data Valid		9		10		16		16	ns
t_{LZOE}	$\overline{OE}$ LOW to Low Z[8]	0		0		0		0		ns
t_{HZOE}	$\overline{OE}$ HIGH to High Z[8,9]		9		11		15		15	ns
t_{LZCE}	$\overline{CE}$ LOW to Low Z[8]	3		3		3		3		ns
t_{HZCE}	$\overline{CE}$ HIGH to High Z[8,9]		9		11		15		15	ns
t_{PU}	$\overline{CE}$ LOW to Power-Up	0		0		0		0		ns
t_{PD}	$\overline{CE}$ HIGH to Power-Down		20		20		20		25	ns
WRITE CYCLE[10,11]										
t_{WC}	Write Cycle Time	20		25		35		45		ns
t_{SCE}	$\overline{CE}$ LOW to Write End	15		18		22		22		ns
t_{AW}	Address Set-Up to Write End	15		20		30		40		ns
t_{HA}	Address Hold from Write End	0		0		0		0		ns
t_{SA}	Address Set-Up to Write Start	0		0		0		0		ns
t_{PWE}	$\overline{WE}$ Pulse Width	15		18		22		22		ns
t_{SD}	Data Set-Up to Write End	10		10		15		15		ns
t_{HD}	Data Hold from Write End	0		0		0		0		ns
t_{HZWE}	$\overline{WE}$ LOW to High Z[9]		10		11		15		15	ns
t_{LZWE}	$\overline{WE}$ HIGH to Low Z[8]	3		3		3		3		ns

Register File (1 read-port, 1 write-port)



N.B. :
$M=2^n$

A memory device is a register file

Memories with 2^K bytes

- Any memory (and in particular memories with 8-bit parallelism – the only we will address), with 2^k cells, has internally a k-variables decoder with E enable (negative true) input (this means that when E = 1 none of the outputs of the decoder is active) which selects one of the memory bytes. The memory interface has therefore the following signals

- Address bits (A_n….A_0) which select the addressed byte (if E enabled)

- Data bits (D_7….D_0)

- At least one *chip select* (CS*) (called also "*chip enable*" (CE* - asterisk means «negative true») which is the enable of the decoder. If CS* is inactive (=1) it means that no byte of this memory (of this «chip») is addressed. CS* is used to selected a memory device when multiple devices are present

- A memory bank (device) is «aligned» if its lowest address within the entire memory system is located at an address multiple of 2^k (a 4 cells device is aligned if its two LSBs address are 0 – multiple of 4 -, a 64K cells device if its 16 LSBs are 0 etc.)

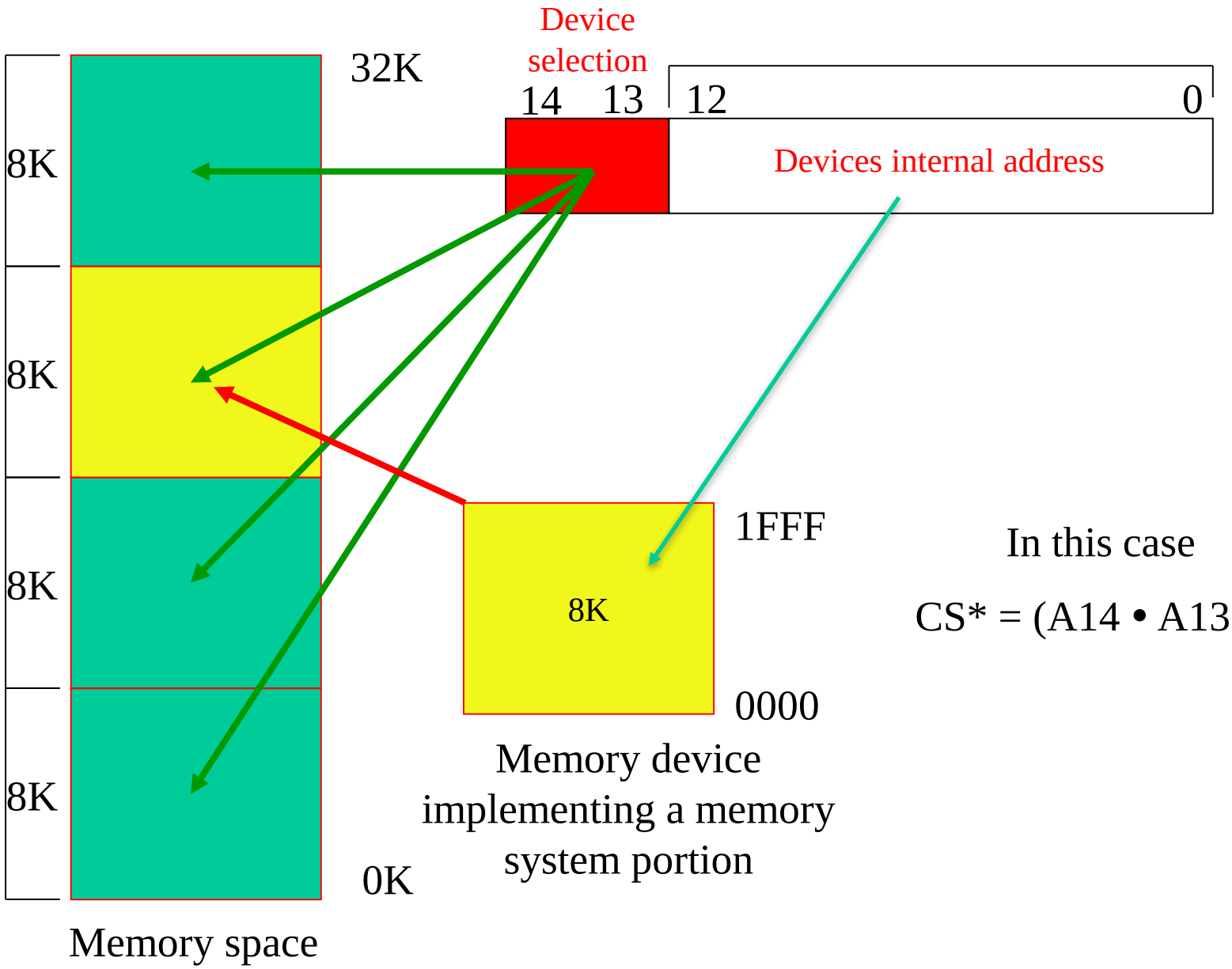
- A Read Command (RD*) which enables the data onto the bus. Normally there is also an Output Enable (OE*) signal. The addressed byte of the memory is output only if CS*=RD*=OE*=0

- If the memory is a RAM (read/write memory) another signal is present: write command WR* or WE* which allows to store bus data into the selected memory byte. Obiously this can occur only if CS*=RD*=OE*=0

- Needless to say RD* and WR* are mutually exclusive

- In a system there are normally multiple memory devices in order to implement memory systems bigger that a single device. Multiple devices are therefore needed, each one implementing a portion of the entire memory

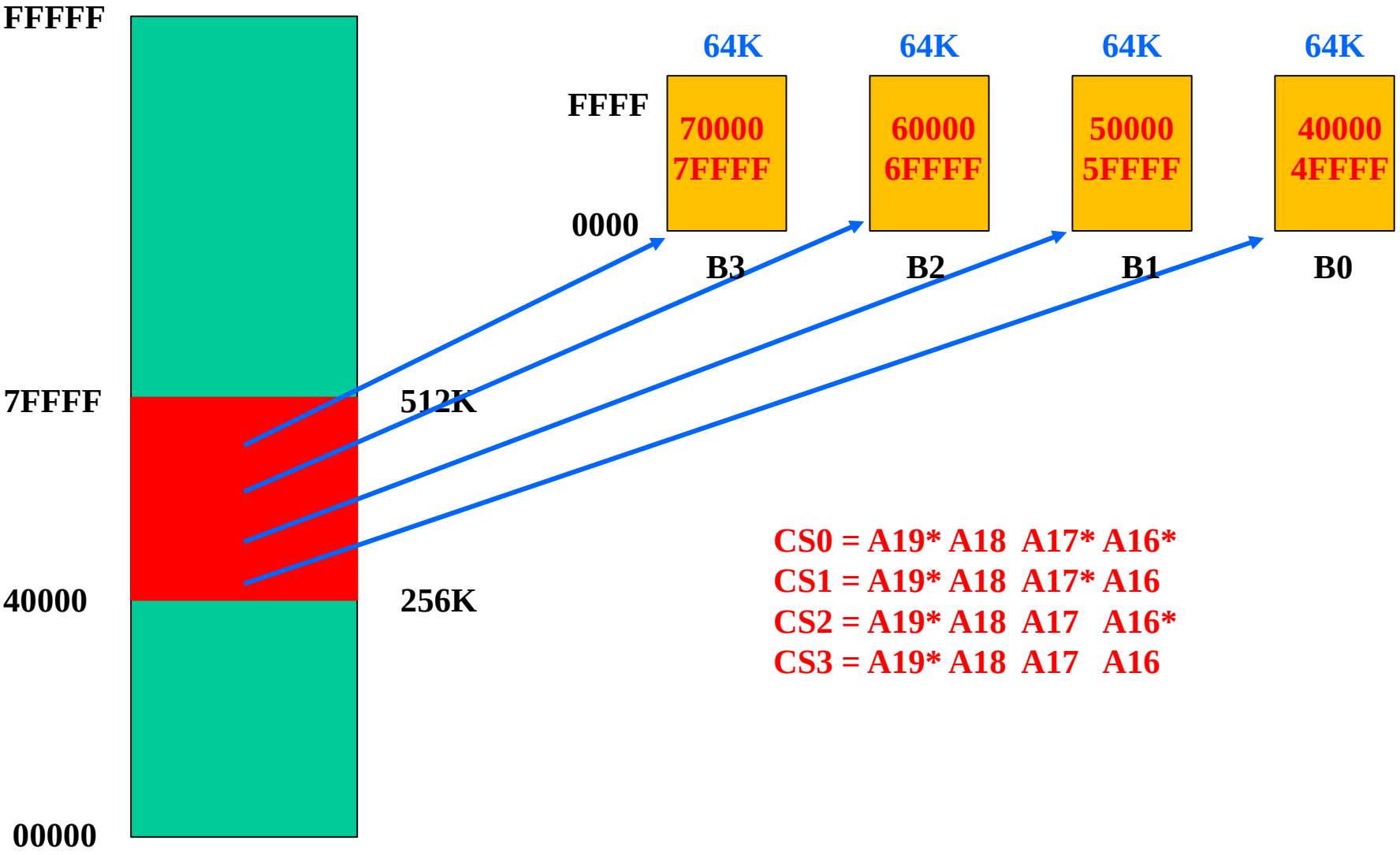
An example: a 32 Kbytes memory system made with 8K devices
32 Kbytes correspond to a 15 bit address



In this case

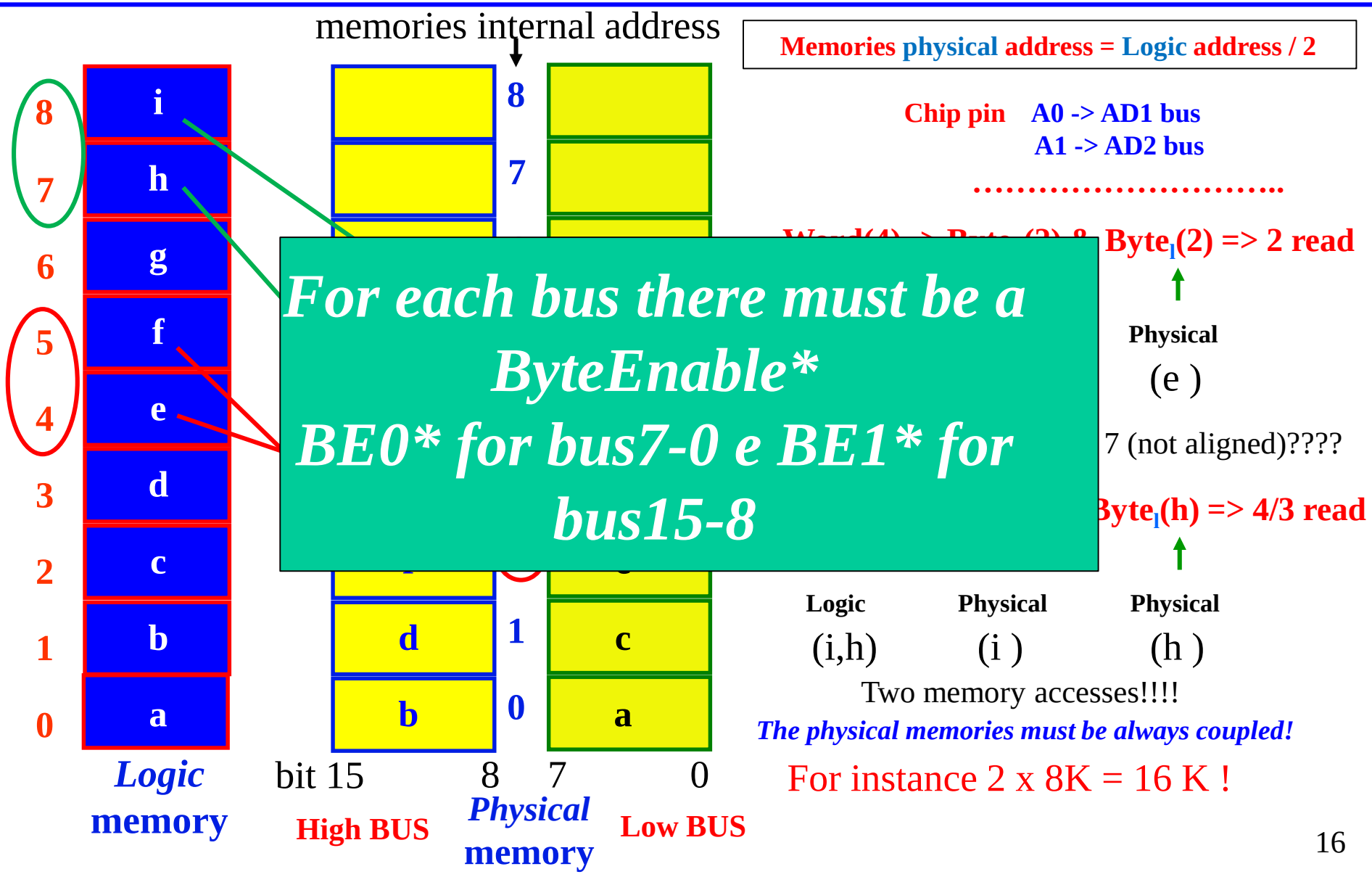
$CS^* = (A14 \bullet A13^*)^*$

1MB address space (20 bit address 0-19) - 8 bit parallelism memory
64K bit ROM devices
256K memory bank – Addresses 40000-7FFFF



CS0 = A19* A18 A17* A16*
CS1 = A19* A18 A17* A16
CS2 = A19* A18 A17 A16*
CS3 = A19* A18 A17 A16

Memory systems with parallelism > 8
An example: a 16 bit bus



Memory systems with parallelism > 8
An example: a 16 bit bus

mov al, even_byte_address ; transfer one byte at even address => one bus cycle 8 bit =>BE0*

mov al, odd_byte_address ; transfer one byte at odd address => one bus cycle 8 bit => BE1*

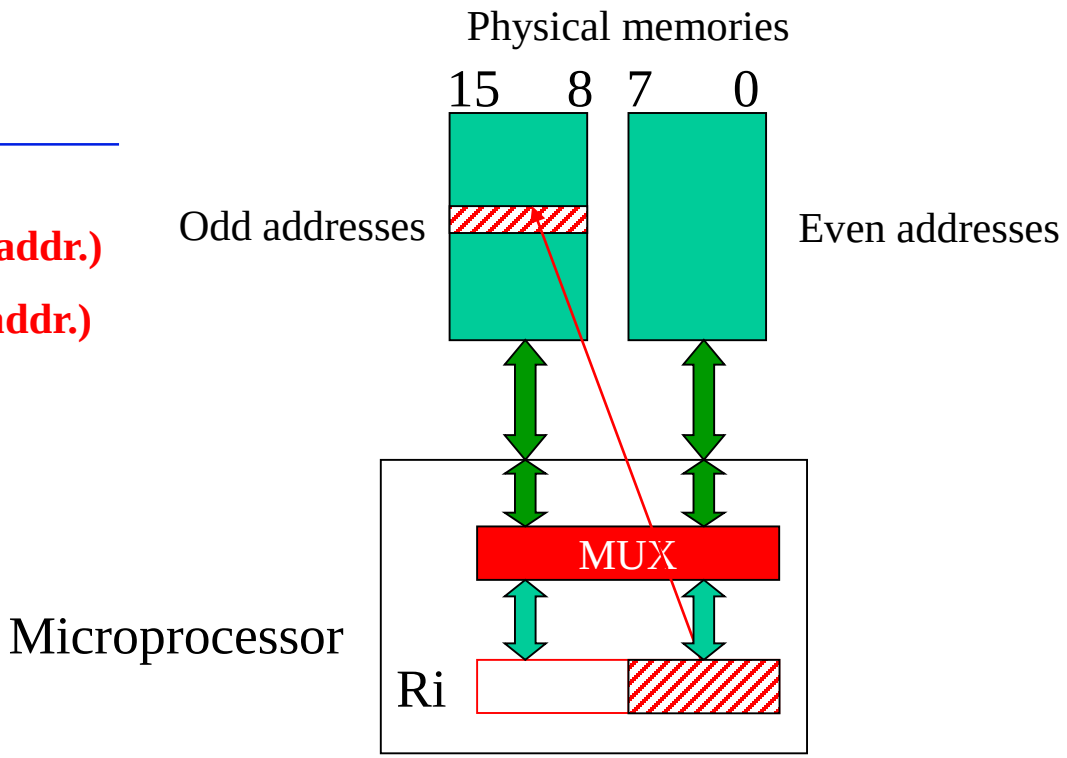
mov ax, even_word_address ; transfer one word at even address => one bus cycle 16 bit => BE0* and BE!*

mov ax, odd_word_address ; transfer one word at odd address => two 8 bit cycles => BE1* and then BE0*

MEMORIES in a 16 BIT SYSTEM (I.E. 8086)

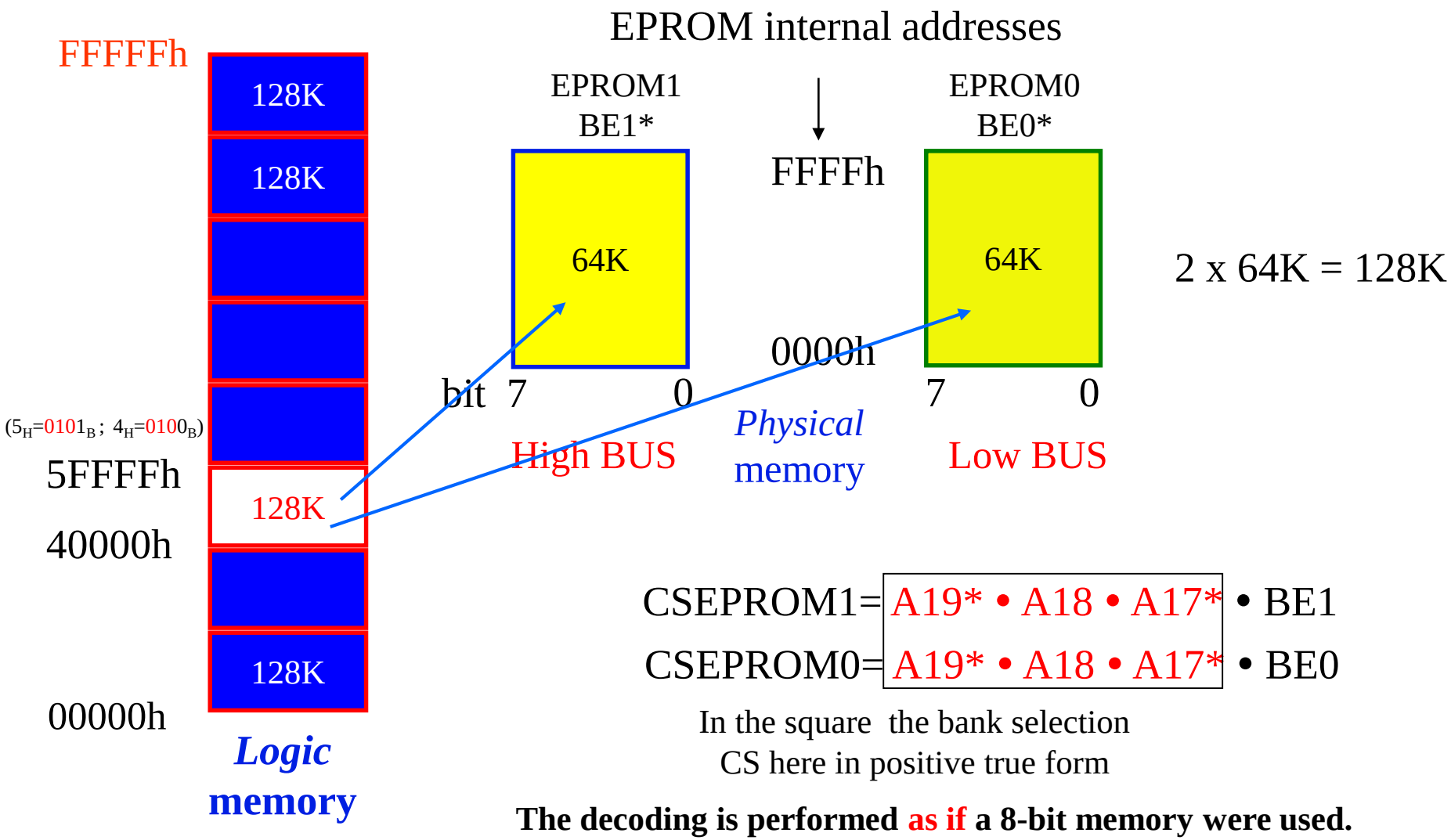
The routing of byte high *external* to byte low *internal* is implemented within the microprocessor chip

BE1*	BE0*	
0	0	Word
0	1	High Byte (odd addr.)
1	0	Low Byte(even addr.)
1	1	No transfer



***A0* of the processor *not* generated (instead *BE0** and *BE1**)**
A1* of the processor connected to chip pin *A0
A2* of the processor connected to chip pin *A1
etc. etc.

Memory systems with parallelism > 8
A 16 bit bus in a 1MB memory (19 to 0 addresses)



CSEPROM1= A19* • A18 • A17* • BE1

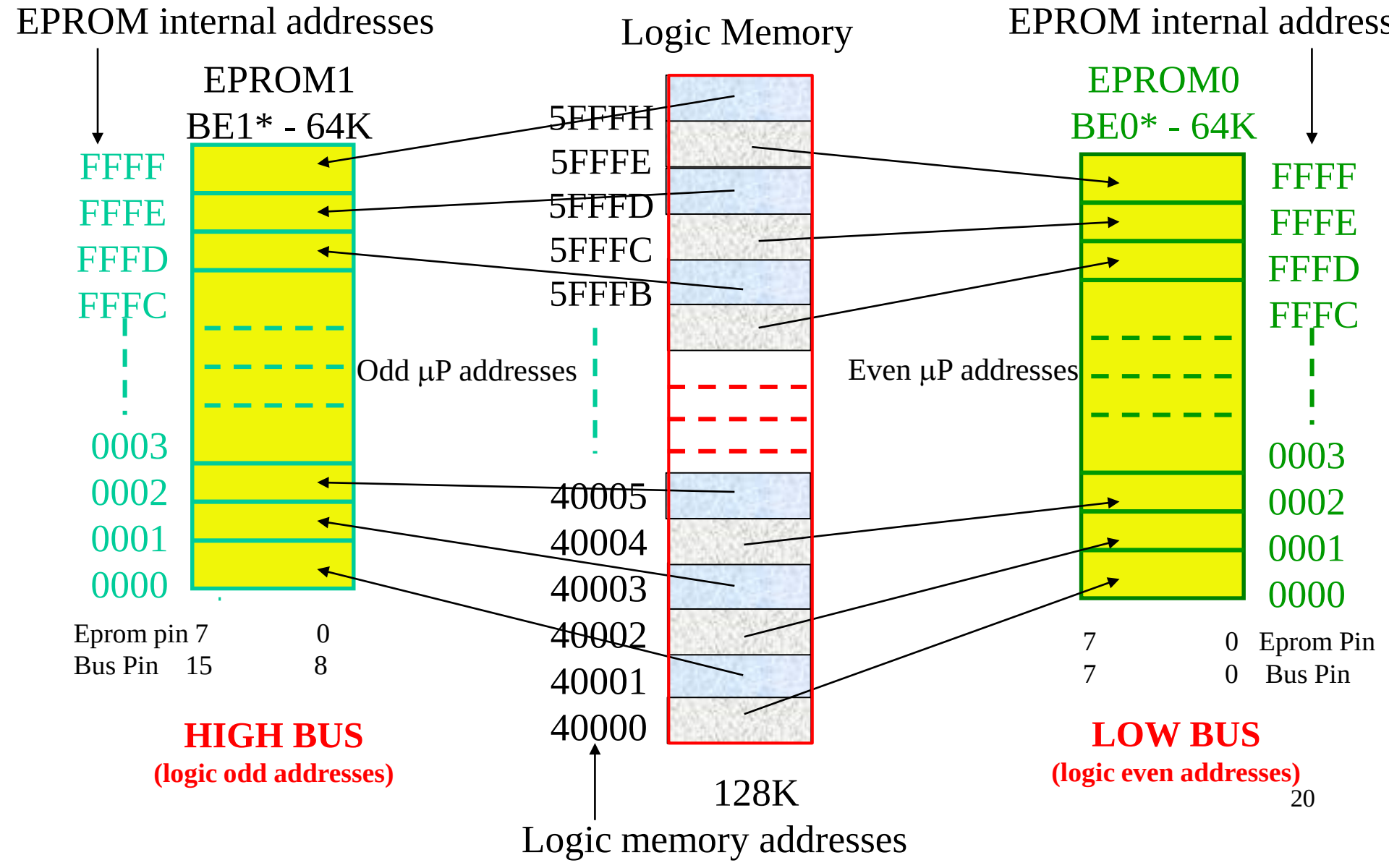
CSEPROM0= A19* • A18 • A17* • BE0

In the square the bank selection
CS here in positive true form

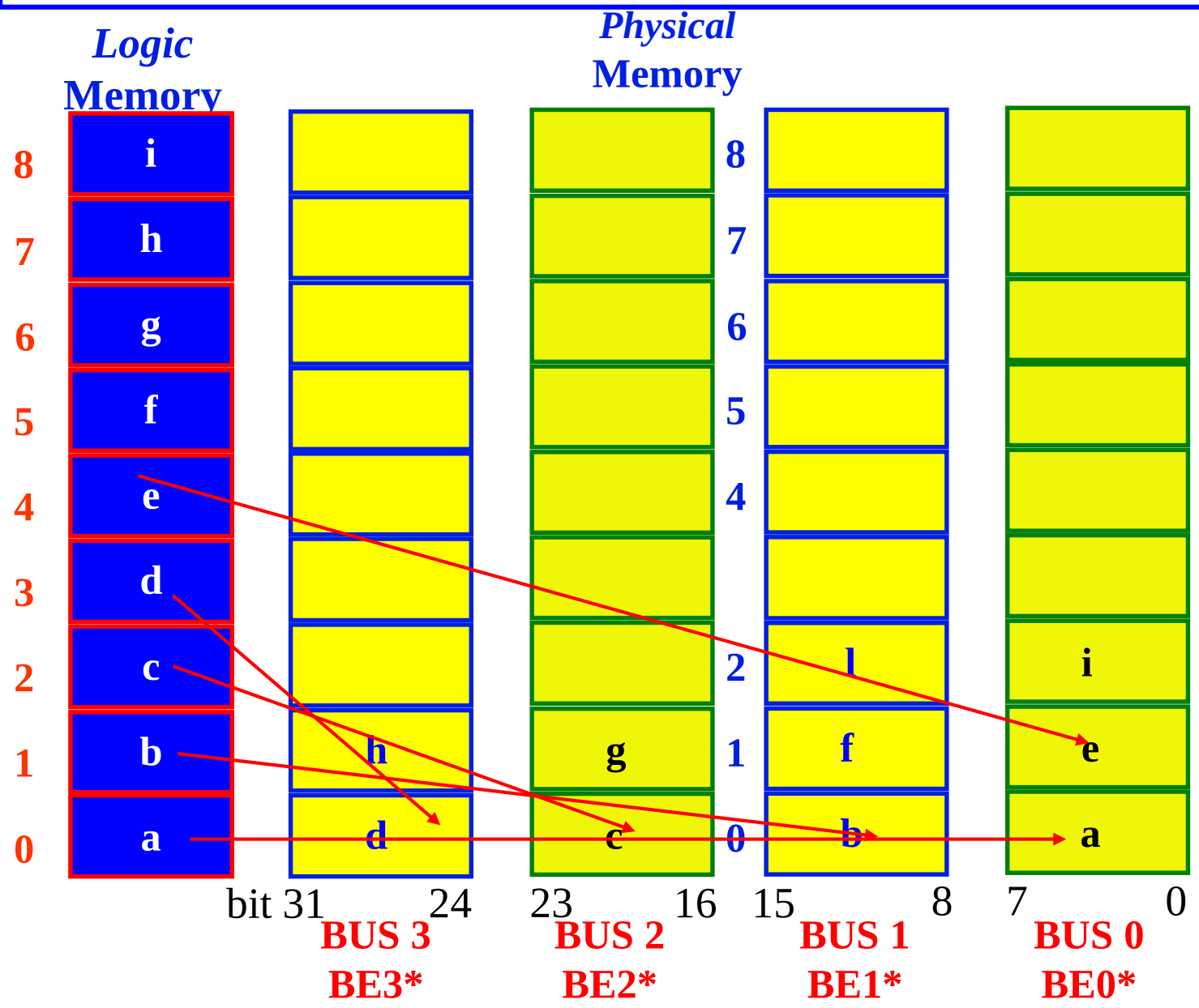
The decoding is performed as if a 8-bit memory were used.
Devices (a couple!) (each one half size of the bank) are used and selected by BE0 e BE1

Memory systems with parallelism > 8

A 16 bit bus in a 1MB memory (19 to 0 addresses)



Memory systems with parallelism > 8
A 32 bit bus



Always 4 coupled devices

Internal memories physical address is the processor address/4

32 BIT bus memories

BE3*	BE2*	BE1*	BE0*	
0	0	0	0	Word 32 bit
1	1	0	0	Half word low
0	0	1	1	Half word high
1	1	1	0	byte 0-7
1	1	0	1	byte 15-8

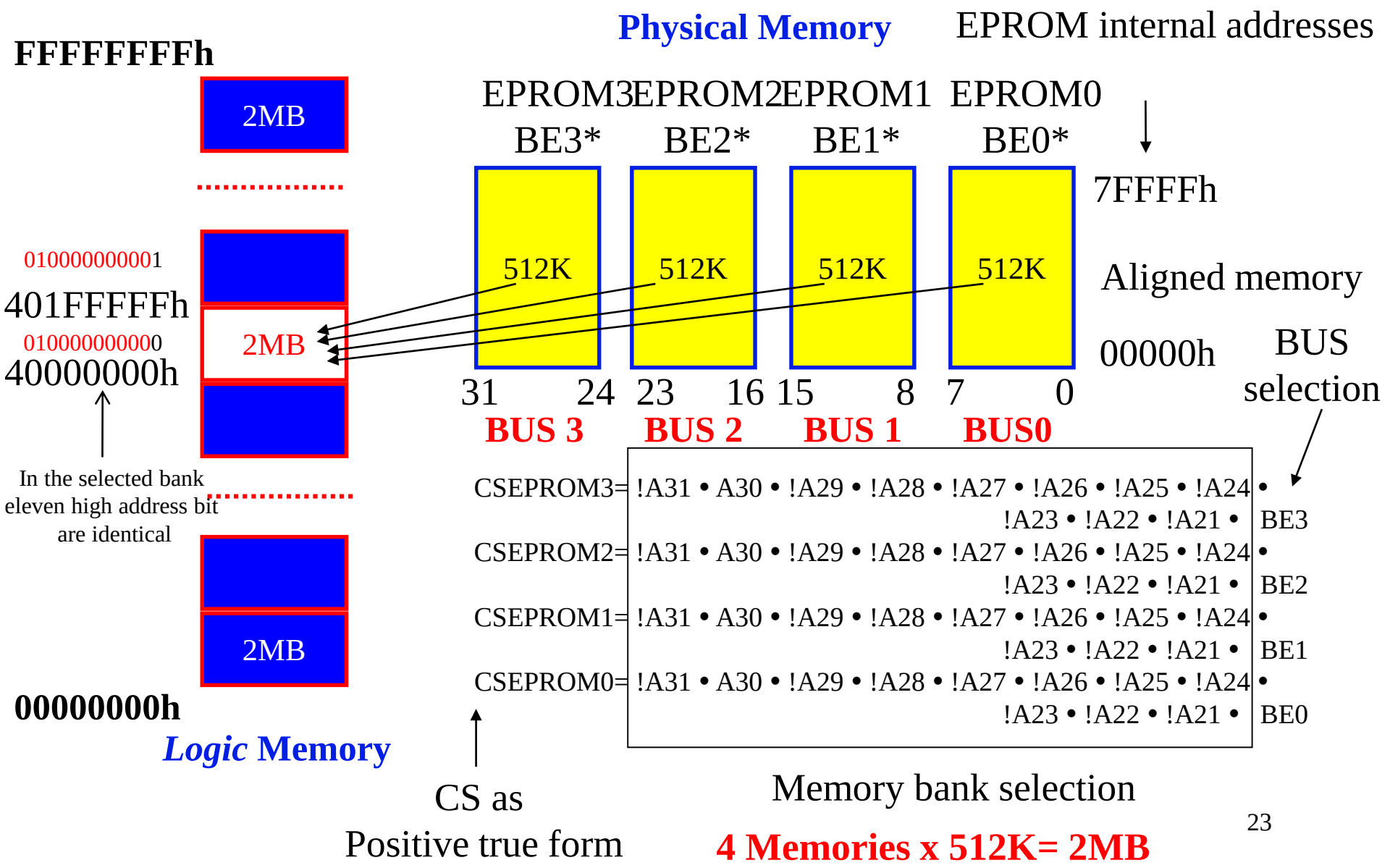
etc.

N.B. *Processor* A0 and A1 *are not generated*
(instead BE0*, BE1*, BE2*, BE3* are generated)

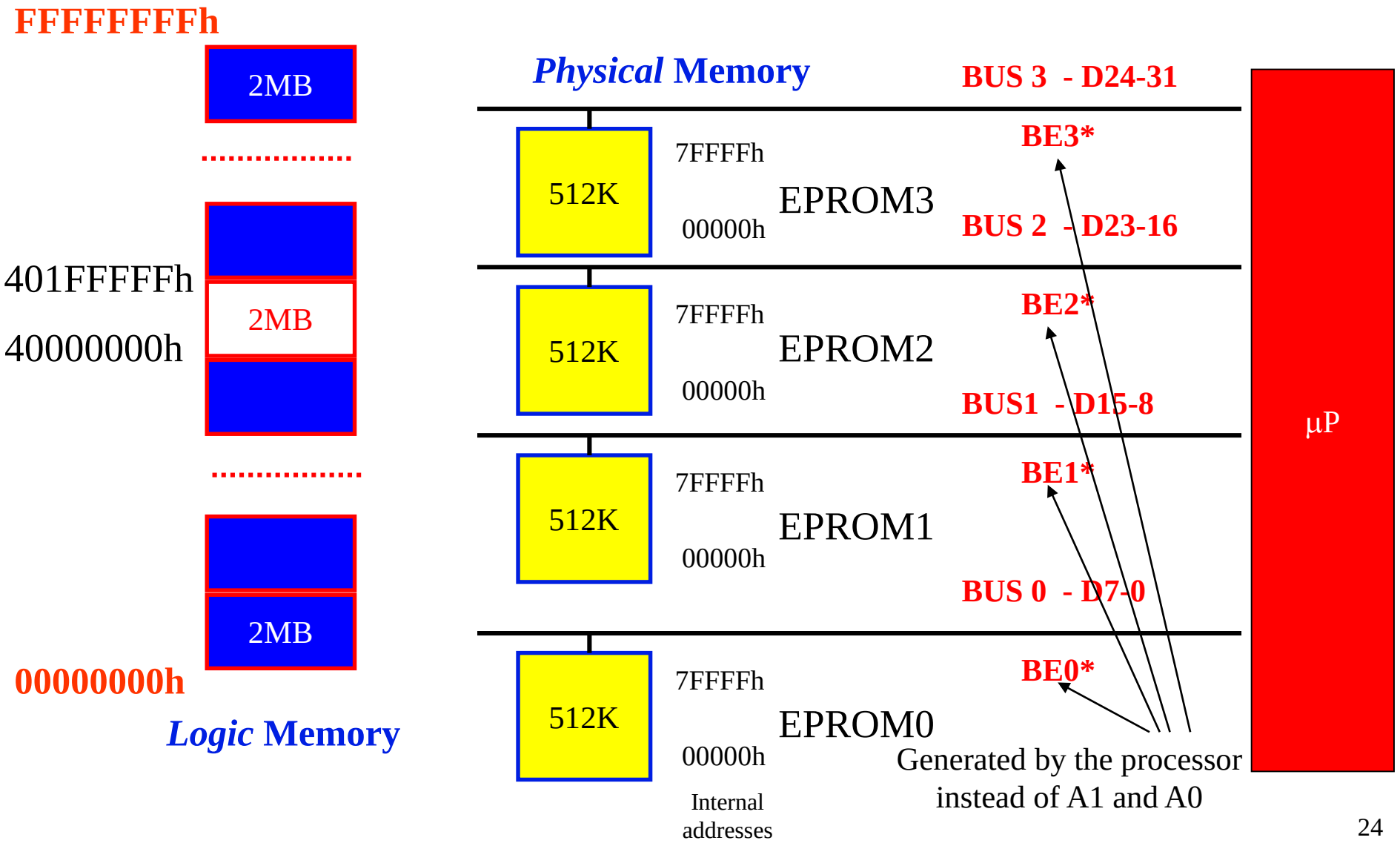
Processor A2connected to memories A0
Processor A3connected to memories A1
etc. etc.

32 BIT bus memories

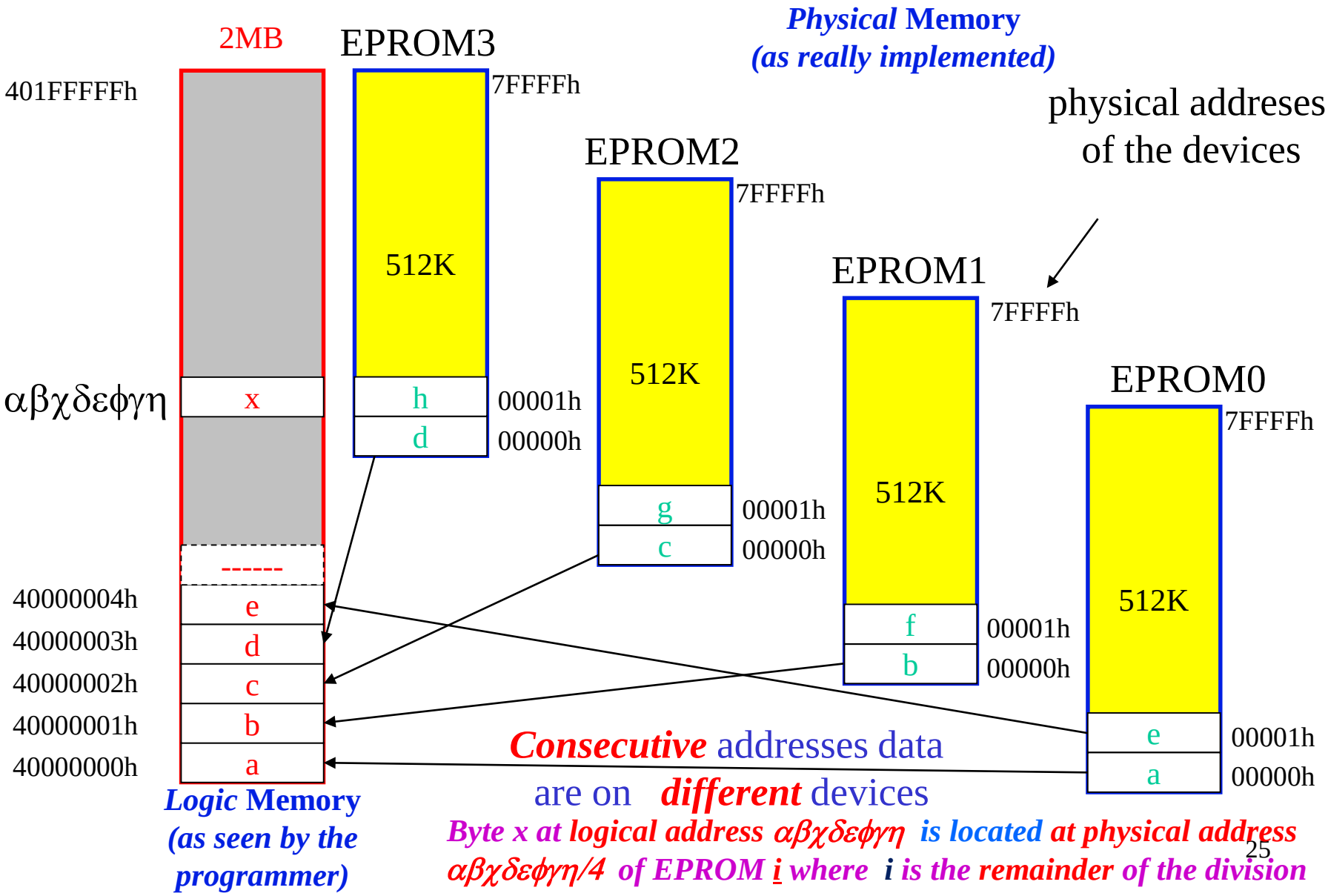
(NB the symbol ! before a logic variable is equivalent to asterisk and means negation)



32 BIT bus memories



32 BIT bus memories



Exercise

A 256K memory RAM (32 bit parallelism) bank starting at address 84000000H in a 4GB memory space using the minimum number of devices.

What if the parallelism were 64 bit?

Exercise

A 256K memory RAM (32 bit parallelism) starting at address 84000000 (aligned) in a 4GB memory space . Address range 84000000-8403FFFF. Devices: 8X32 K RAM devices (64K RAM do not exist!!!!)
Therefore two banks of 128K : the first implements the range 84000000 to 8401FFFF and the other the range 84020000 to 8403FFFF. The 32K memory chips use internally the addresses A14-A0 which are connected to bus addresses BA16-BA2 . Notice that bus address BA17 separates the two banks

First bank

CSRAM00=(BA31•BA30!•BA29!•BA28!•BA27!•BA26 •….•BA18!•BA17!)•BE0
CSRAM01=(BA31•BA30!•BA29!•BA28!•BA27!•BA26 •….•BA18!•BA17!)•BE1
CSRAM02=(BA31•BA30!•BA29!•BA28!•BA27!•BA26 •….•BA18!•BA17!)•BE2
CSRAM03=(BA31•BA30!•BA29!•BA28!•BA27!•BA26 •….•BA18!•BA17!)•BE3

Second bank

CSRAM10=(BA31•BA30!•BA29!•BA28!•BA27!•BA26 •….•BA18!•BA17) •BE0
CSRAM11=(BA31•BA30!•BA29!•BA28!•BA27!•BA26 •….•BA18!•BA17) •BE1
CSRAM12=(BA31•BA30!•BA29!•BA28!•BA27!•BA26 •….•BA18!•BA17) •BE2
CSRAM13=(BA31•BA30!•BA29!•BA28!•BA27!•BA26 •….•BA18!•BA17) •BE3

If the parallelism of the memory were 64 bit the 8 CS (CSRAM00..CSRAM07) would **NOT** be affected by BA17: ***BE7-BE0*** instead would play its role. The 8 devices would implement the entire memory bank and memory chips addresses A14-A0 would be connected to ***bus addresses BA17-BA3*** !! *It is the parallelism which changes not the address range!!!*