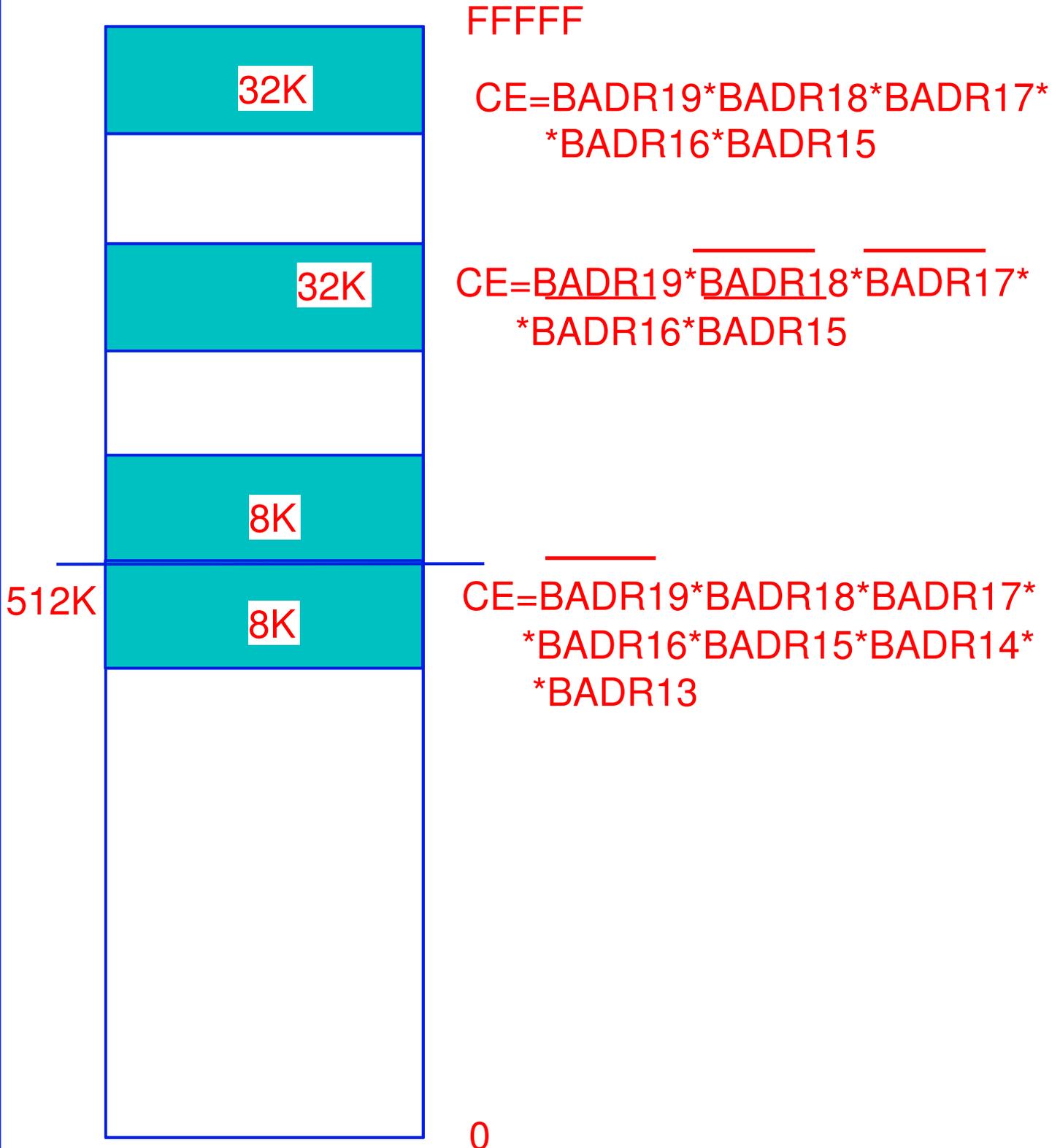


INTERFACCIA PROCESSORE MEMORIE

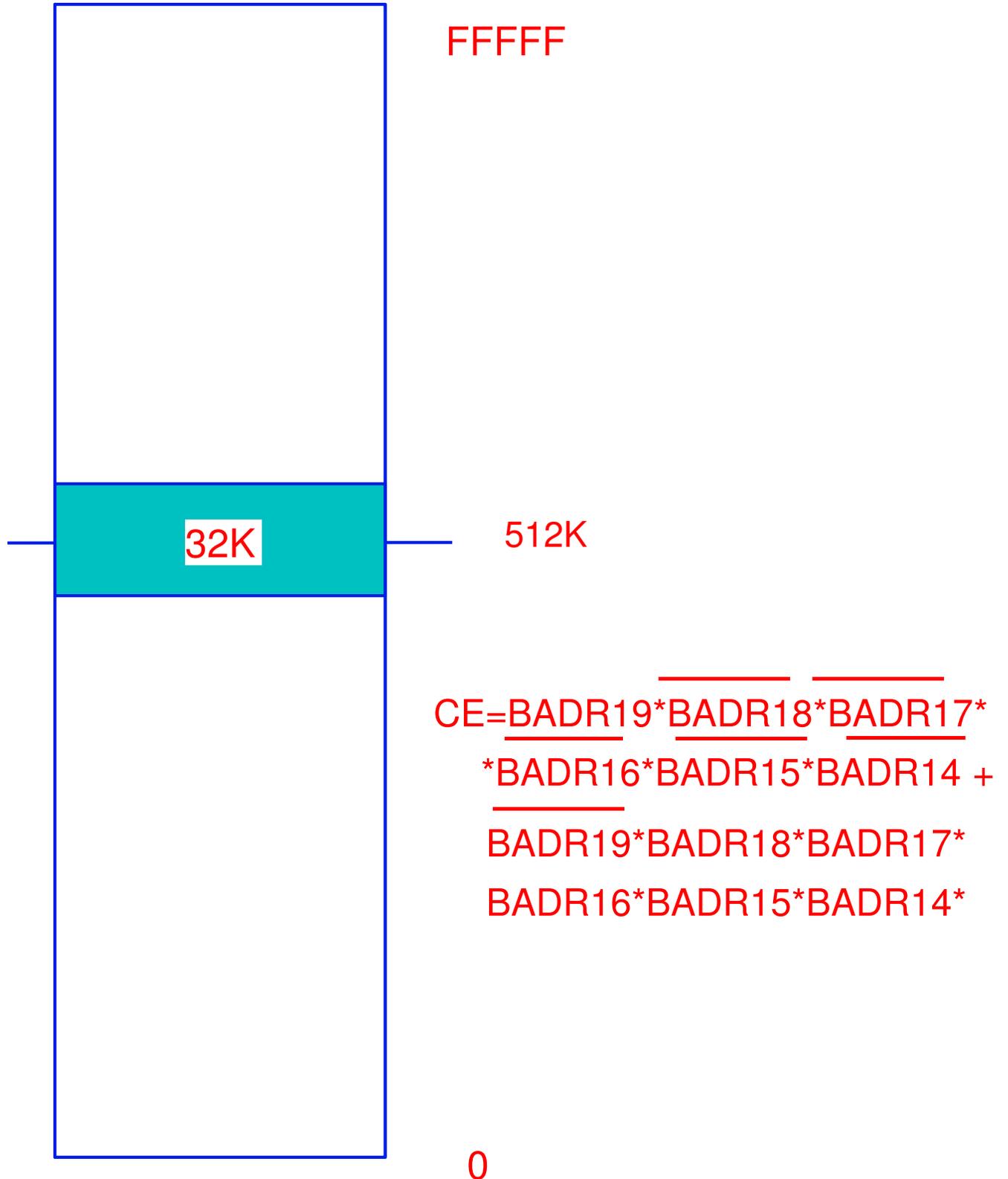
Decodifica degli indirizzi

Esempi di sistemi di memorie con 8086

SISTEMI DI MEMORIE



SISTEMI DI MEMORIE



MEMORIE CON 8086

❖ Parallelismo 16: BHE e ADDR0

- ❖ 8088: accesso a byte in un solo ciclo di bus, accesso a word in due cicli
- ❖ 8086: accesso a byte e a word allineate in un solo ciclo di bus, a word disallineate in due cicli

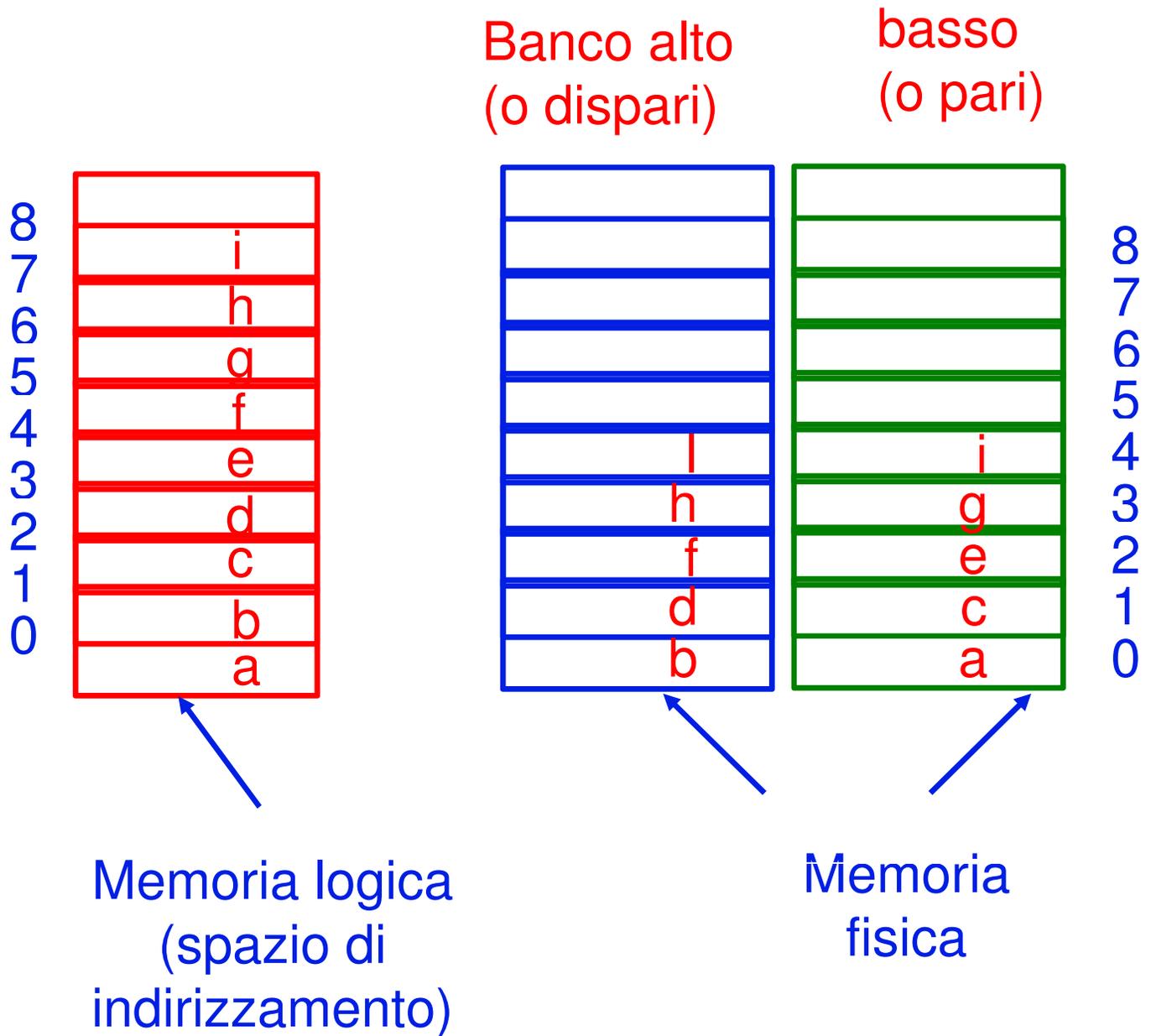
BHE*	A0	
0	0	Word
0	1	Byte alto (ind dispari)
1	0	Byte basso (pari)
1	1	-

❖ Esempi: mov BX,0

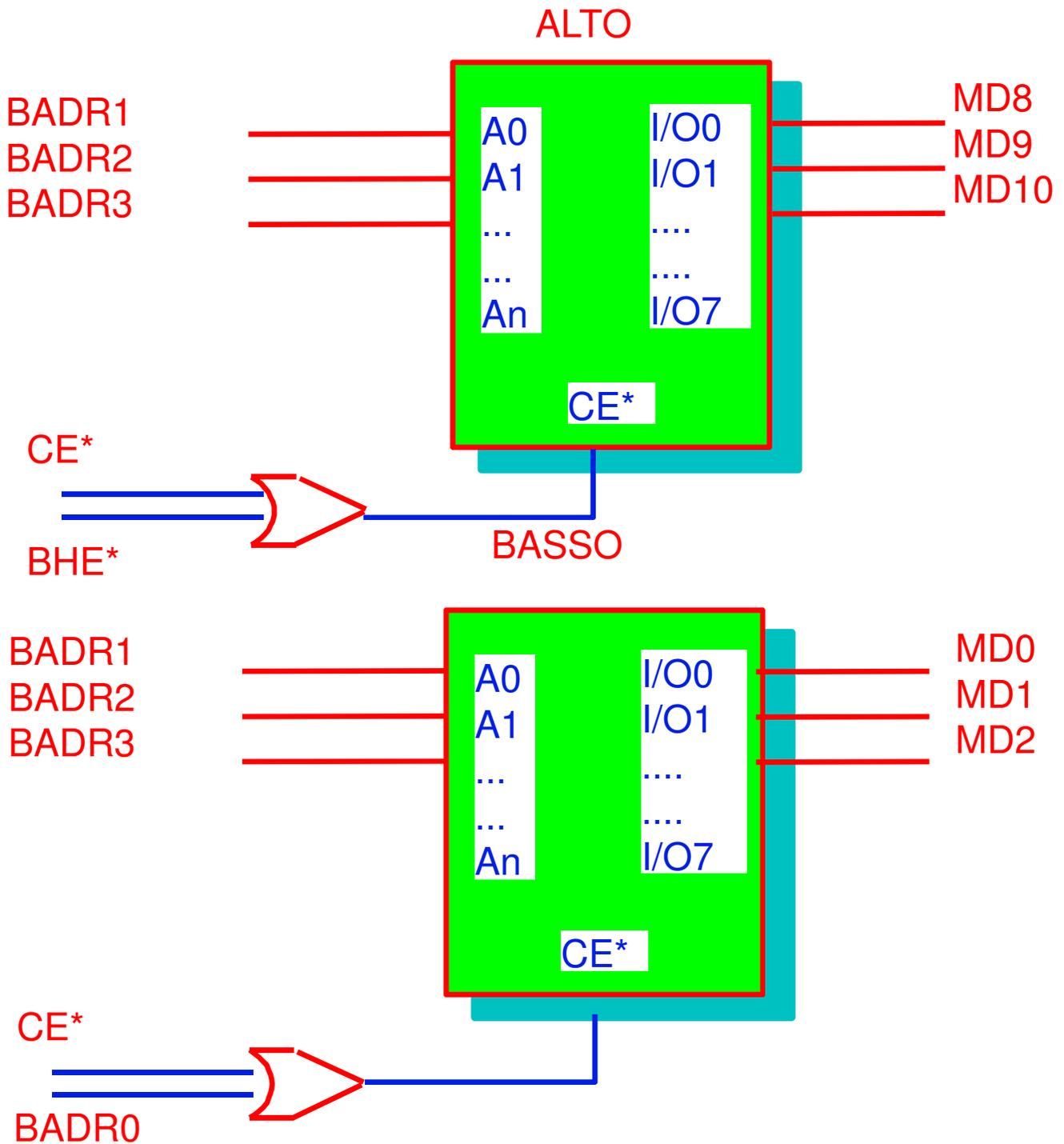
- ❖
mov AH, [BX]23
mov AL, [BX]22
mov AX, [BX]22
mov AX, [BX]23

- ❖ Lo scambio byte alto esterno, byte basso registro avviene all'interno del microprocessore (8086)

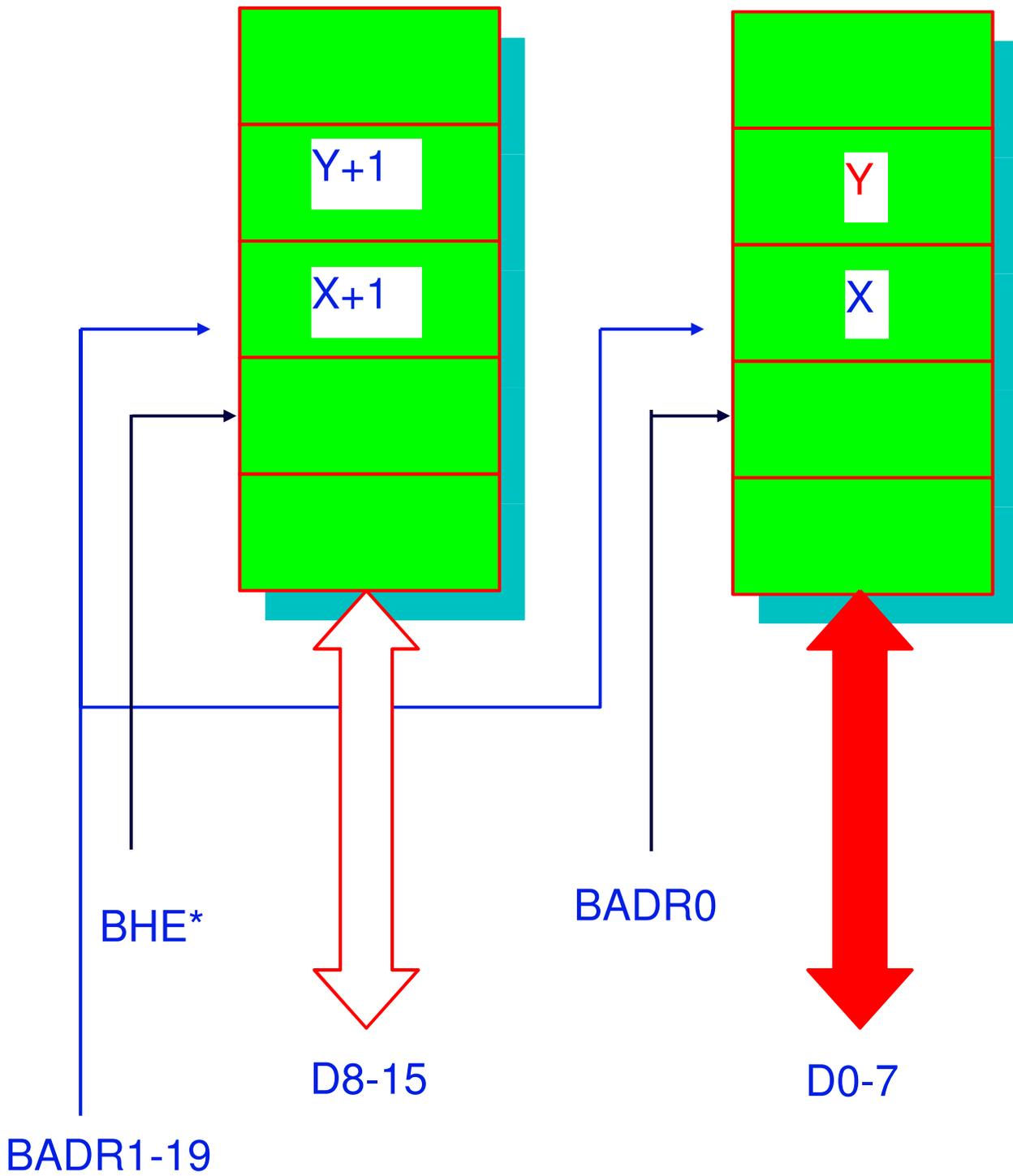
MEMORIE CON 8086



MEMORIE E 8086

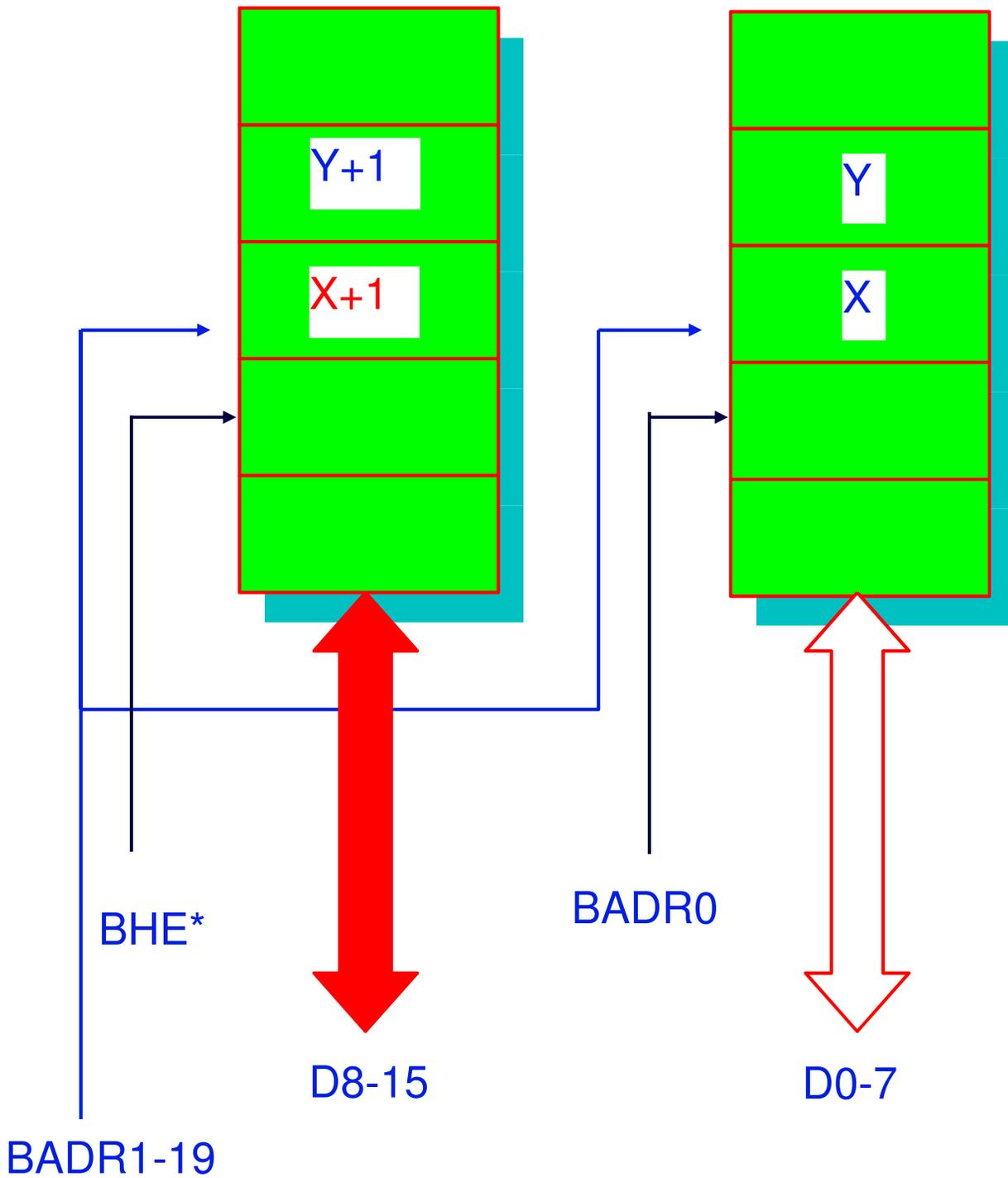


MEMORIE E 8086



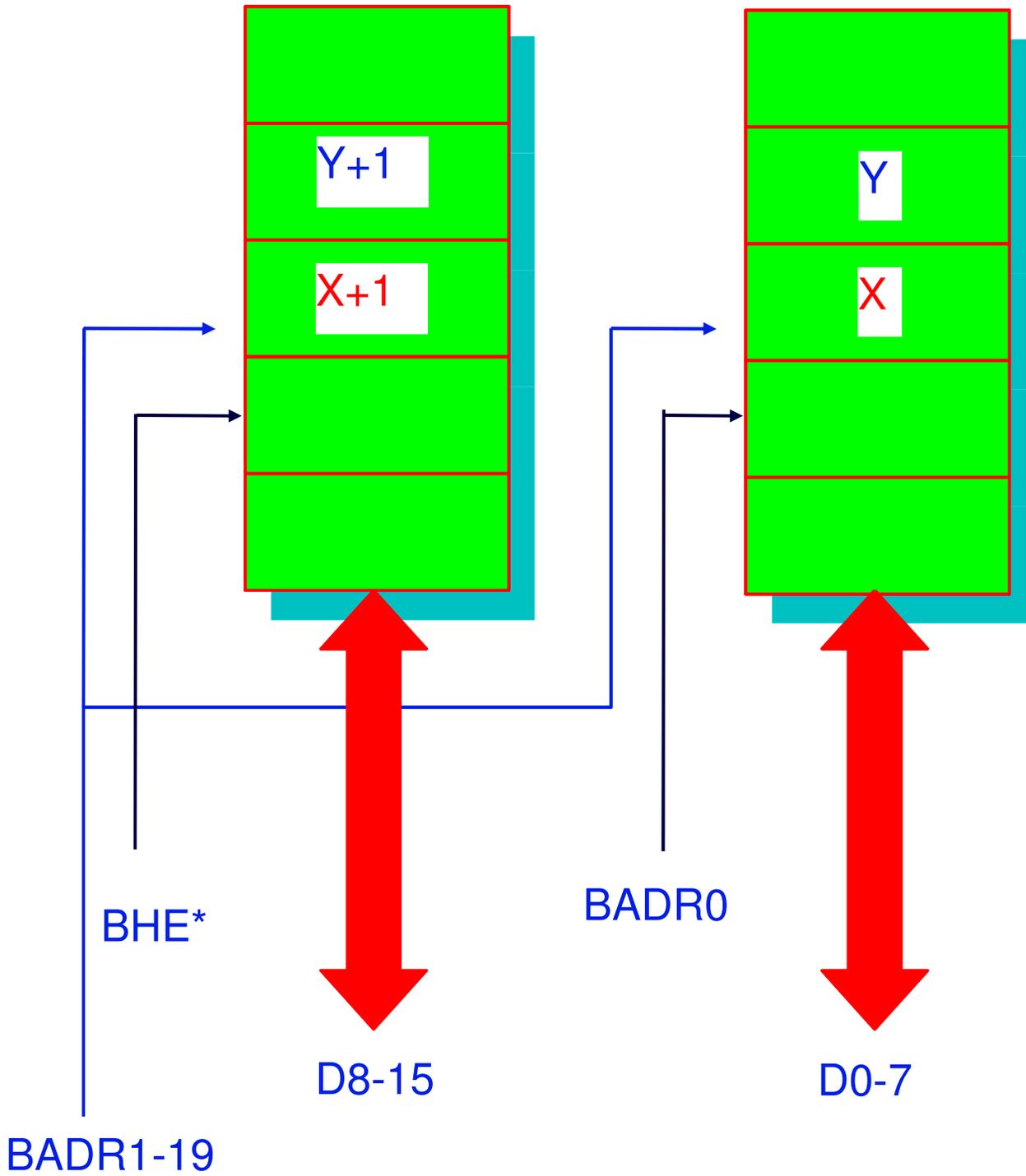
❖ Trasferimento di byte a indirizzo pari

MEMORIE E 8086



- Trasferimento di byte a indirizzo dispari

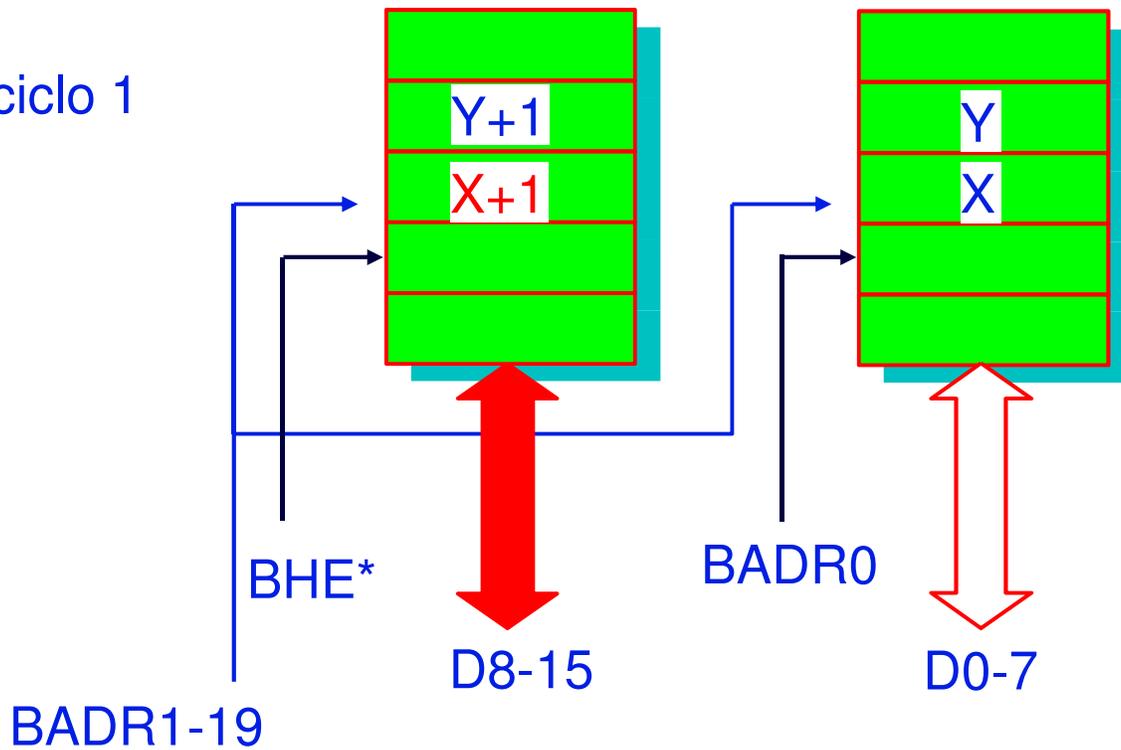
MEMORIE E 8086



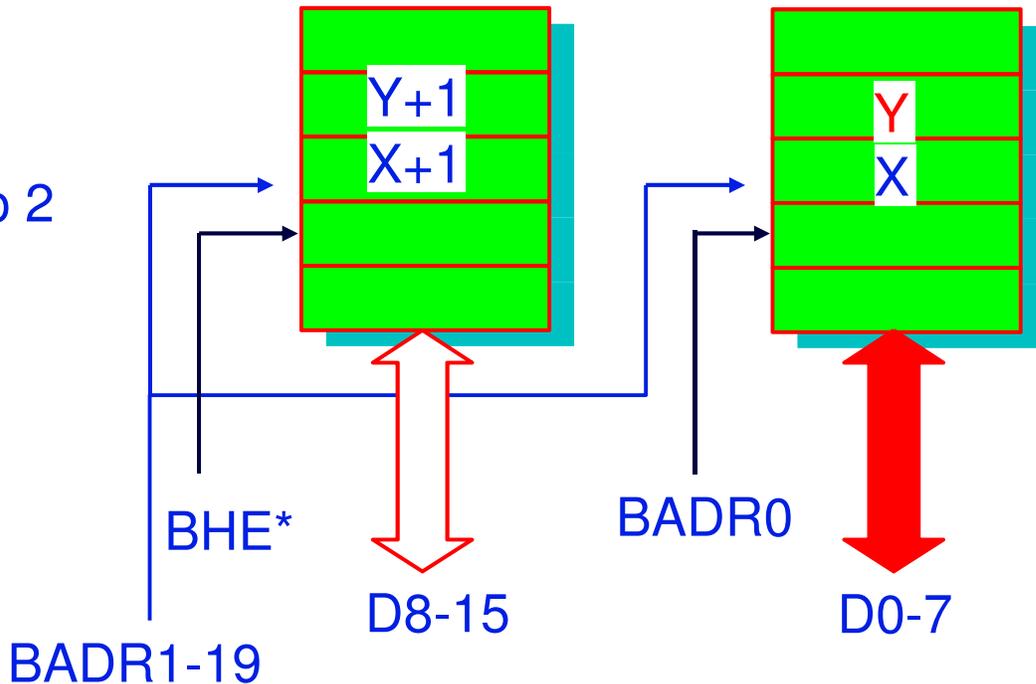
❖ Trasferimento di word a indirizzo pari

MEMORIE E 8086

❖ ciclo 1



❖ ciclo 2



□ Trasferimento di word a indirizzo dispari

DECODIFICA SEMPLIFICATA

- Non sempre vengono utilizzati tutti i bit dell'indirizzo per decodificare la cella di memoria.
- E' possibile usare in alternativa una **decodifica semplificata**, il cui scopo è discriminare tra tutte le celle di memoria **presenti (e non tra tutte le celle logicamente indirizzabili dal microprocessore)**.
- Ex.: 8088 con soli 32K EPROM e 32KRAM
- 64Kbytes -> 16 bit di indirizzo

E gli altri bit di indirizzo ?

REPLICA dei banchi
nello spazio di indirizzamento

Esempi

FFFF

32K

Decodifica completa:

$$CE = \text{BADR19} * \text{BADR18} * \text{BADR17} * \\ * \text{BADR16} * \text{BADR15}$$

Decodifica semplificata:

$$CE = \text{BADR19}$$

Decodifica completa:

$$CE = \underline{\text{BADR19}} * \underline{\text{BADR18}} * \underline{\text{BADR17}} \\ * \text{BADR16} * \text{BADR15} * \text{BADR14} * \\ * \text{BADR13}$$

Decodifica semplificata:

$$CE = \text{BADR19} * \text{BADR13}$$

8K

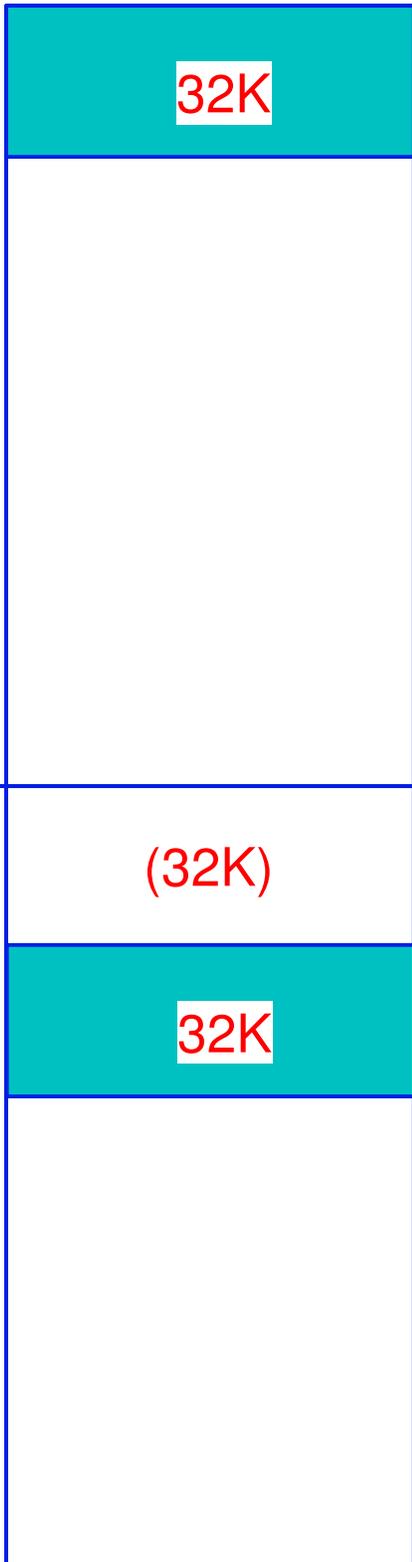
Decodifica semplificata:

$$CE = \underline{\text{BADR19}} * \underline{\text{BADR13}}$$

8K

0

Esempi



FFFFFFh

Decodifica semplificata:
CE=BADR15

Decodifica semplificata:

$\overline{\text{CE}}=\text{BADR15}$

70000h

0h

MEMORY MAPPED I/O

- I dispositivi di I/O possono essere selezionati tramite indirizzi di memoria (riducendo contemporaneamente il relativo spazio)
- Non viene usato il segnale IO/M* nella decodifica o nella generazione dei segnali di lettura e scrittura (che risultano unici per memorie e dispositivi di I/O)
- **VANTAGGI**: si ha disposizione una maggiore ricchezza di istruzioni (tutte quelle della memoria)
- **SVANTAGGI**: si riduce lo spazio della memoria, si rende tendenzialmente più complessa l'interpretazione dei programmi e si possono correre dei rischi con i compilatori ottimizzanti

MEMORIE E I/O SEPARATI

- Memorie e I/O possono avere CE* coincidenti ma un solo dispositivo è letto o scritto se si differenziano i segnali di lettura e scrittura: MEMRD, MEMWR, IORD, IOWR
- La selezione contemporanea aumenta i consumi: una selezione differenziata è comunque auspicabile per sistemi a basso consumo