UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

La gerarchia di memoria

Luigi Palopoli

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Organizzazione gerarchica della memoria

- Un elaboratore senza memoria non funziona ...
- La memoria negli elaboratori non è tutta uguale
- Per decenni il sogno di ogni programmatore e' stato quello di avere *tanta* memoria con accessi ultrarapidi.
- Esistono compromessi tra costo, prestazioni e dimensione della memoria

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Basics ...

- La memoria serve a contenere dati, bisogna poter leggere e scrivere in memoria ...
- La memoria indirizzata direttamente (principale o cache) è
  - di tipo volatile, cioè il suo contenuto viene perso se si spegne l'elaboratore
  - limitata dallo spazio di indirizzamento del processore
- La memoria indirizzata in modo indiretto
  - di tipo permanente: mantiene il suo contenuto anche senza alimentazione
  - ha uno spazio di indirizzamento "software" non limitato dal processore

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Basics ...

- Le informazioni nella memoria principale (indirizzamento diretto) è accessibile al processore in qualsiasi momento
- Le informazioni nella memoria periferica (indirizzamento indiretto) devono prima essere trasferite nella memoria principale
- Il trasferimento dell'informazione tra memoria principale e memoria periferica è mediato dal software (tipicamente il S.O.)

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Terminologia

- **Tempo di accesso**: tempo richiesto per ***una*** operazione si lettura/scrittura nella memoria
- **Tempo di ciclo**: tempo che intercorre tra l'inizio di due operazioni (es. due read) tra locazioni diverse; in genere leggermente superiore al tempo di accesso
- **Accesso Casuale**:
  - non vi è alcuna relazione o ordine nei dati memorizzati
  - tipico delle memorie a semiconduttori

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

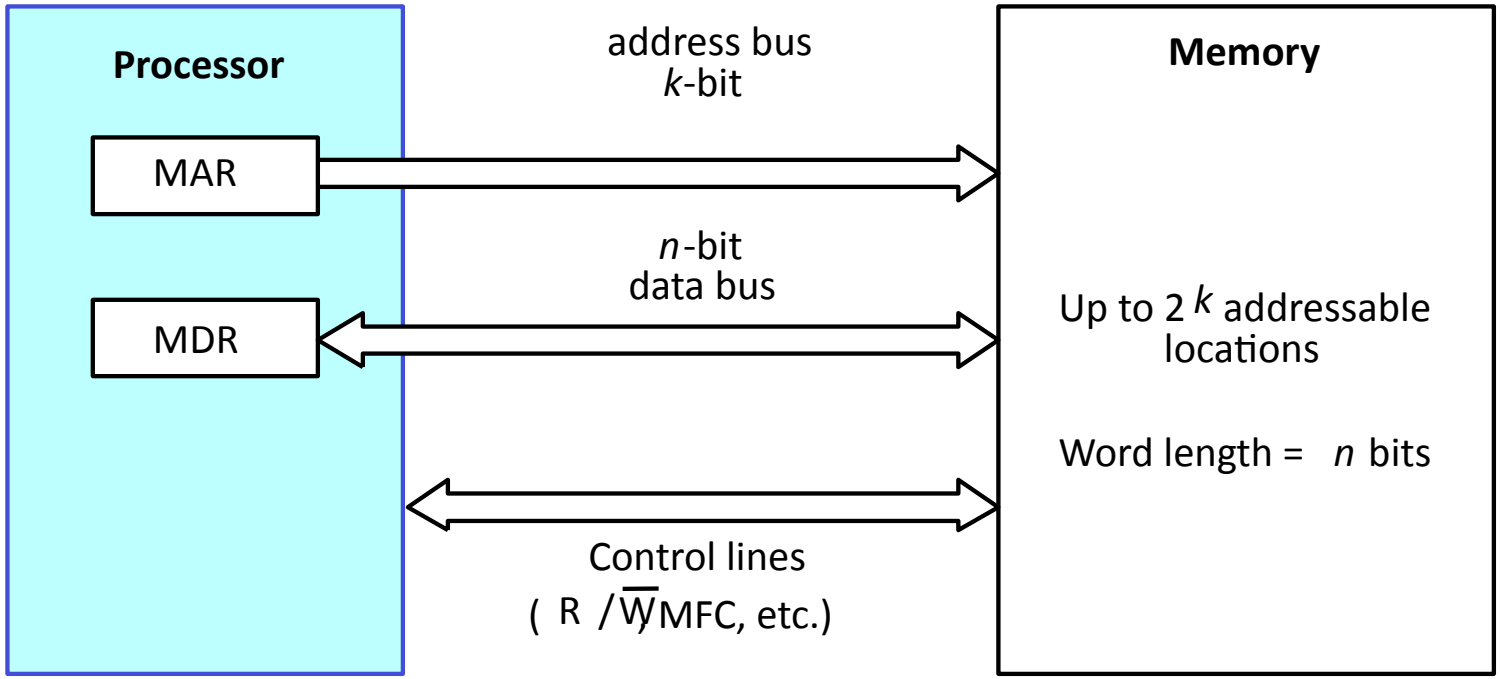
Terminologia

- **Accesso Sequenziale:**
  - l'accesso alla memoria è ordinato o semi-ordinato
  - il tempo di accesso dipende dalla posizione
  - tipico dei dischi e dei nastri
- **RAM**: Random Access Memory
  - memoria scrivibile/leggibile a semiconduttori
  - tempo di accesso indipendente dalla posizione dell'informazione
- **ROM**: Read Only Memory
  - memoria a semiconduttori in sola lettura
  - accesso casuale o sequenziale

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Memoria principale

- Connessione "logica" con il processore



MAR: Memory Address Register
MDR: Memory Data Register
MFC: Memory Function Completed

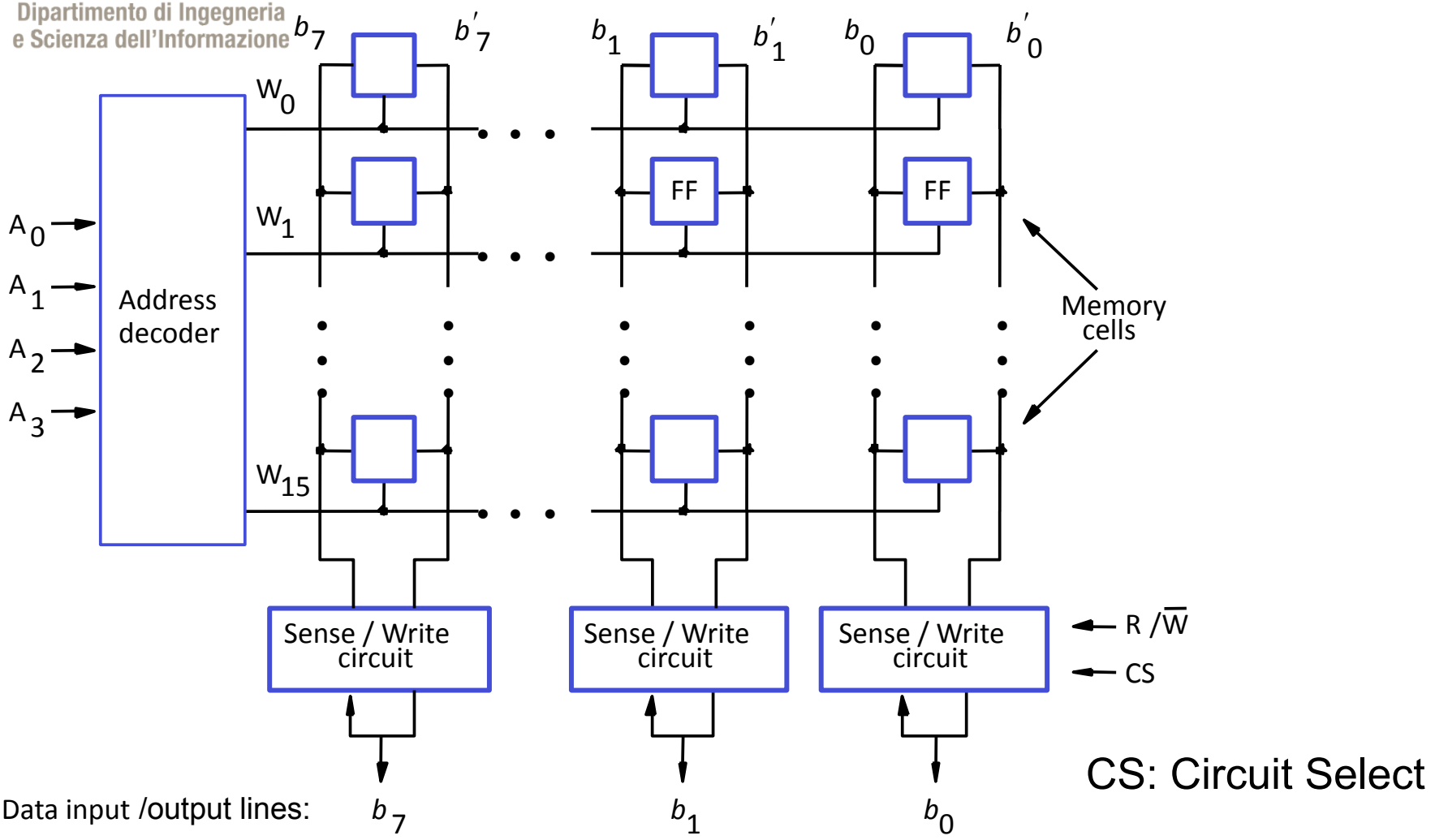
UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Memorie RAM a semiconduttori

- Memorizzano singoli bit, normalmente organizzati in byte e/o word
- Data una capacità N (es. 512 Kbit) la memoria può essere organizzata in diversi modi a seconda del parallelismo P (es. 8, 1 o 4)
  - 64K X 8
  - 512K X 1
  - 128K X 4
- L'organizzazione influenza in numero di pin di I/O dell'integrato (banco) che realizza la memoria

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Organizzazione dei bit in un banco di memoria 16 X 8



CS: Circuit Select

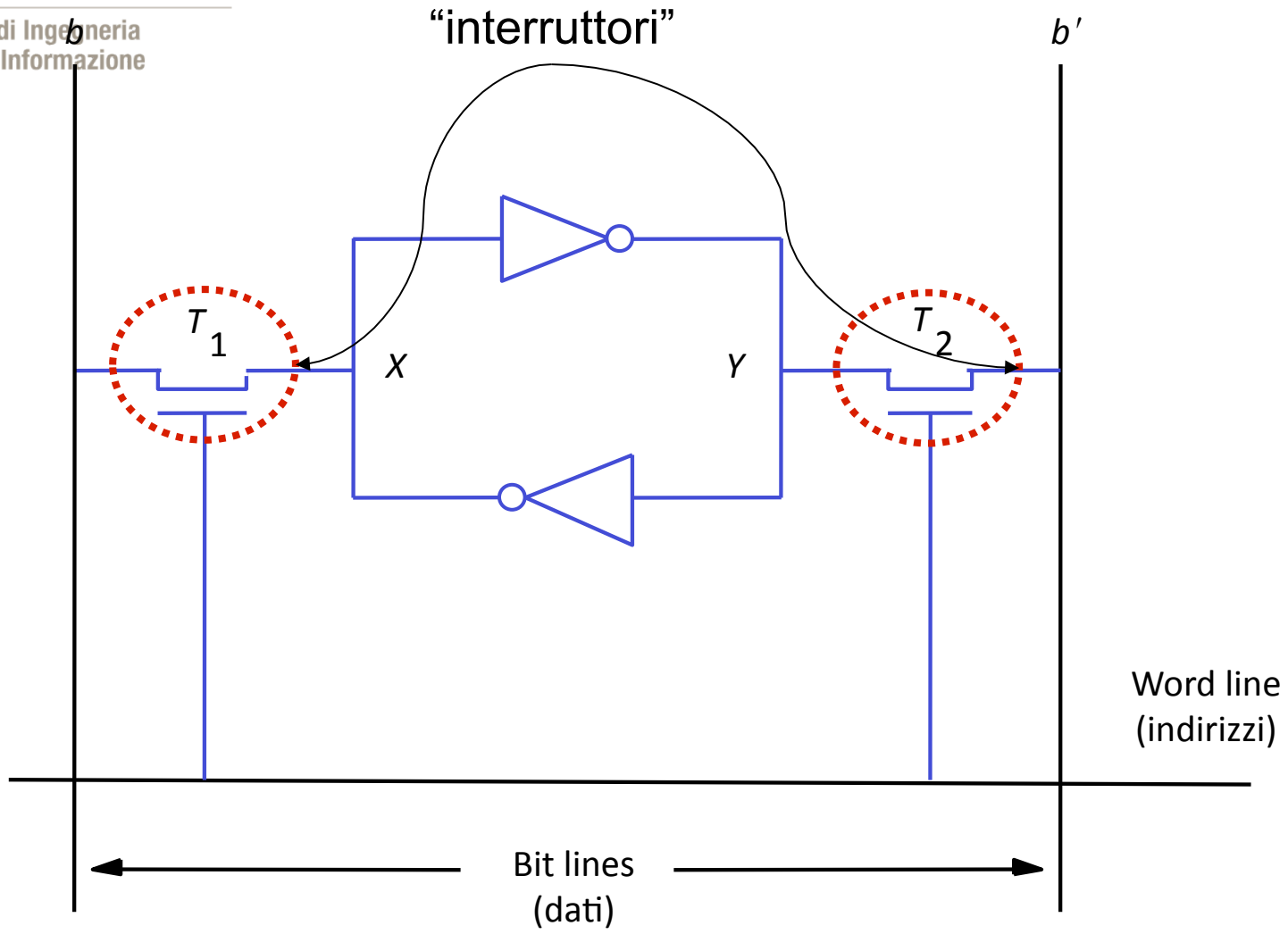
UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Memorie Statiche (SRAM)

- Sono memorie in cui i bit possono essere tenuti indefinitamente (posto che non manchi l'alimentazione)
- Estremamente veloci (tempo di accesso di pochi ns)
- Consumano poca corrente (e quindi non scaldano)
- Costano care perchè hanno molti componenti per ciascuna cella di memorizzazione

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

SRAM: cella di memoria



UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

SRAM: lettura e scrittura

- b' = NOT(b): i circuiti di terminazione della linea di bit (sense/write circuit) interfacciano il mondo esterno che non accede mai direttamente alle celle
- La presenza contemporanea di b e NOT(b) consente un minor tasso di errori
- **Scrittura:** la linea di word è alta e chiude T_1 e T_2, il valore presente su b e b', che funzionano da linee di pilotaggio, viene memorizzato nel latch a doppio NOT
- **Lettura:** la linea di word è alta e chiude T_1 e T_2, le linee b e b' sono tenute in stato di alta impedenza: il valore nei punti X e Y viene "copiato" su b e b'
- Se la linea di word è bassa T_1 e T_2 sono interruttori aperti: il consumo è praticamente nullo

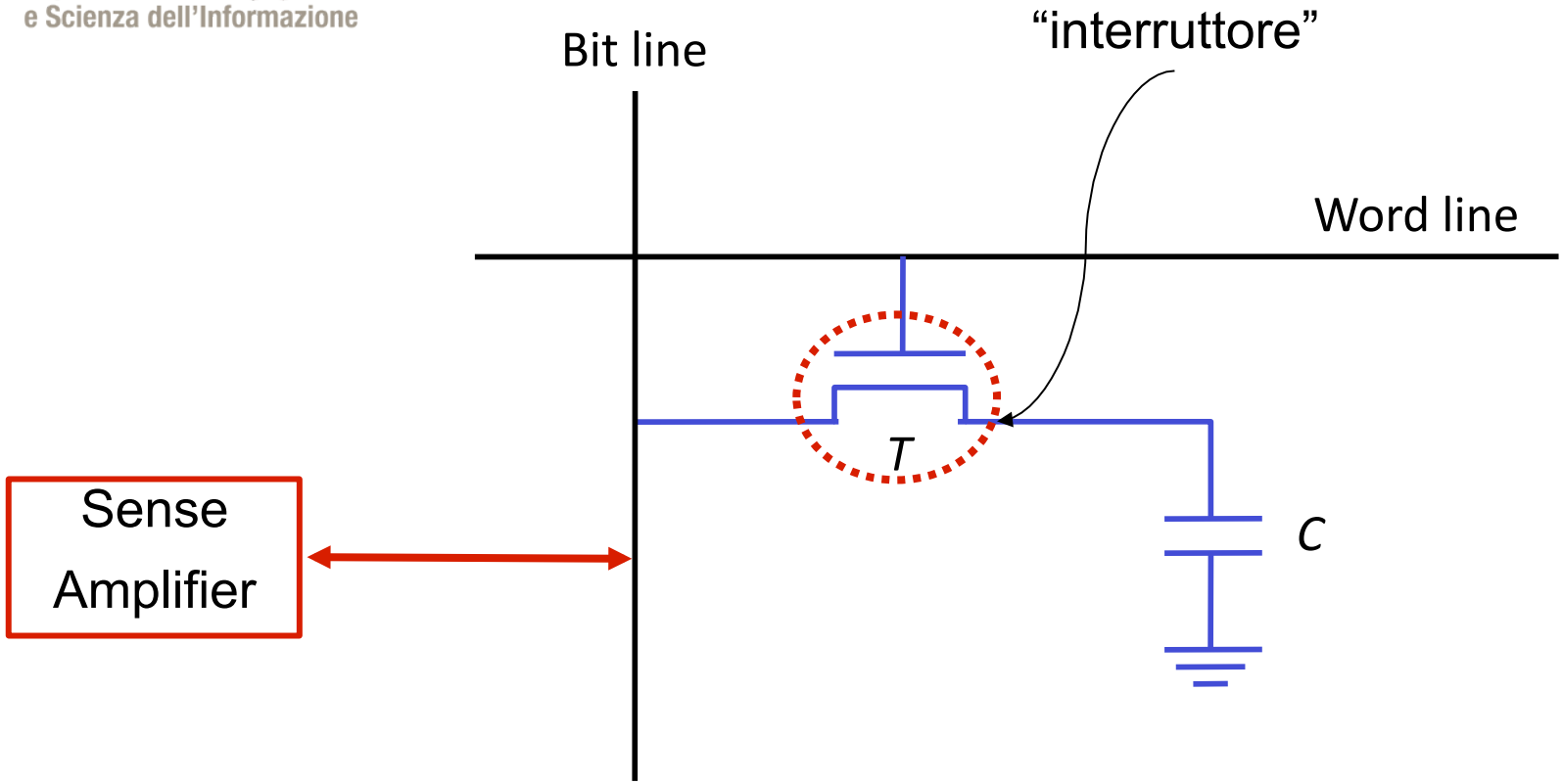
UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

RAM dinamiche (DRAM)

- Sono le memorie più diffuse nei PC e simili
- Economiche e a densità elevatissima (in pratica 1 solo componente per ogni cella)
  - la memoria viene ottenuta sotto forma di carica di un condensatore
- Hanno bisogno di un rinfresco continuo del proprio contenuto che altrimenti "svanisce" a causa delle correnti parassite
- Consumi elevati a causa del rinfresco continuo

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

DRAM: cella di memoria



UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

DRAM: lettura e scrittura

- **Scrittura:** la linea di word è alta e chiude T, il valore presente su b viene copiato su C (carica il transistor)
- **Lettura:** la linea di word è alta e chiude T, **un apposito circuito (sense amplifier)** misura la tensione su C
  - se è sopra una soglia data pilota la linea b alla tensione nominale di alimentazione, ricaricando il condensatore C,
  - se è sotto la soglia data mette a terra la linea b scaricando completamente il condensatore

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

DRAM: tempi di rinfresco

- Nel momento in cui T viene aperto il condensatore C comincia a scaricarsi (o caricarsi, anche se più lentamente) a causa delle resistenze parassite dei semiconduttori
- E` necessario rinfrescare la memoria prima che i dati "spariscano"

  basta fare un ciclo di lettura
- In genere il chip di memoria contiene un circuito per il rinfresco (lettura periodica di tutta la memoria); l'utente non si deve preoccupare del problema

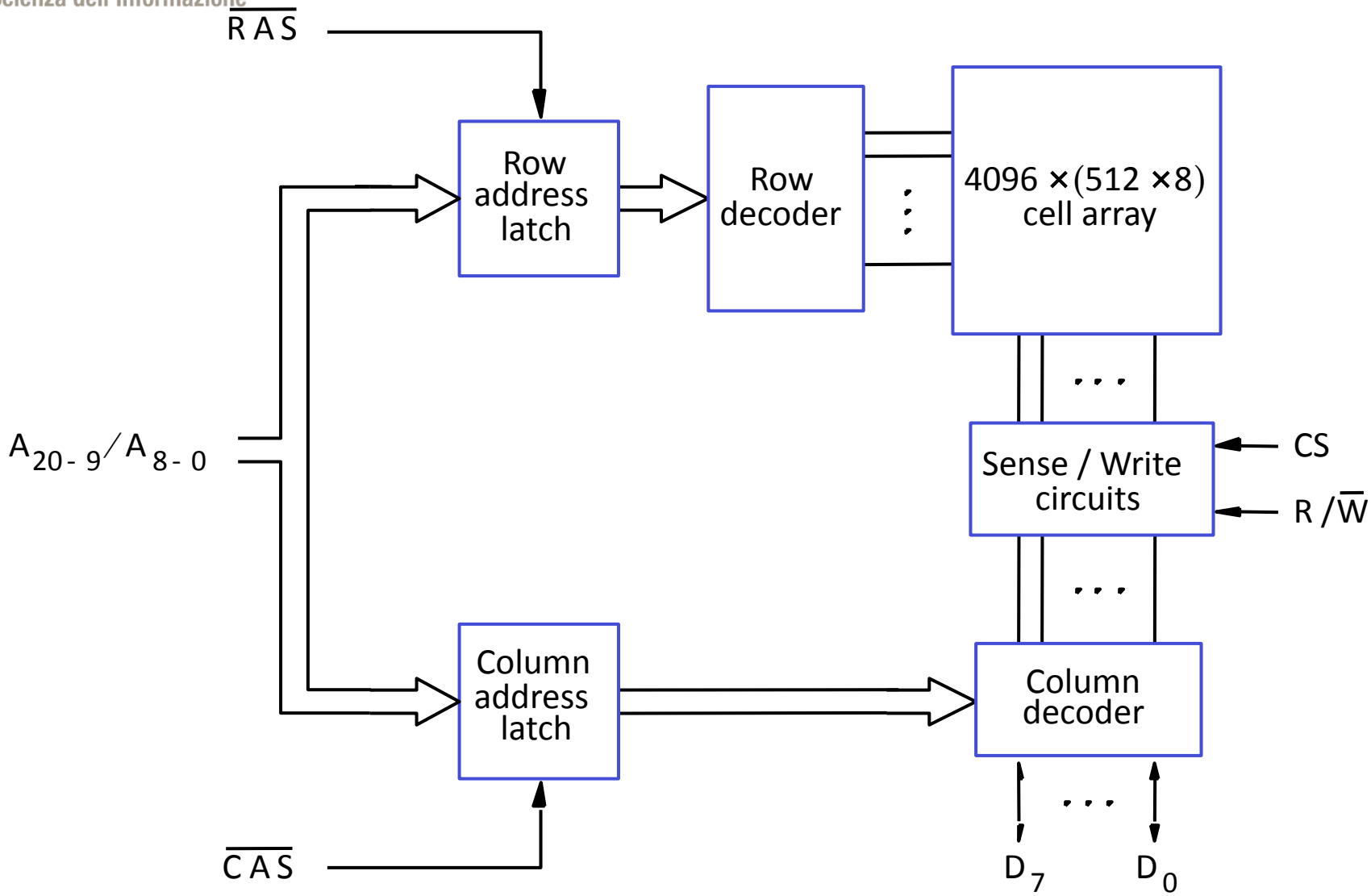
UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

DRAM: multiplazione degli indirizzi

- Dato l'elevata integrazione delle DRAM il numero di pin di I/O è un problema
- E` usuale multiplare nel tempo l'indirizzo delle righe e delle colonne negli stessi fili
- Normalmente le memorie non sono indirizzabili al bit, per cui righe e colonne si riferiscono a byte e non a bit
- Es. una memoria 2M X 8 (21 bit di indirizzo) può essere organizzata in 4096 righe (12bit di indirizzo) per 512 colonne (9bit di indirizzo) di 8 bit ciascuno

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Organizzazione di una DRAM 2M X 8



UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

DRAM: modo di accesso veloce

- Spesso i trasferimenti da/per la memoria avvengono a blocchi (o pagine)
- Nello schema appena visto, vengono selezionati prima 4096 bytes e poi tra questi viene scelto quello richiesto
- E` possibile migliorare le prestazioni semplicemente evitando di "riselezionare" la riga ad ogni accesso se le posizioni sono consecutive
- Questo viene chiamato "fast page mode" (FPM) e l'incremento di prestazioni può essere significativo

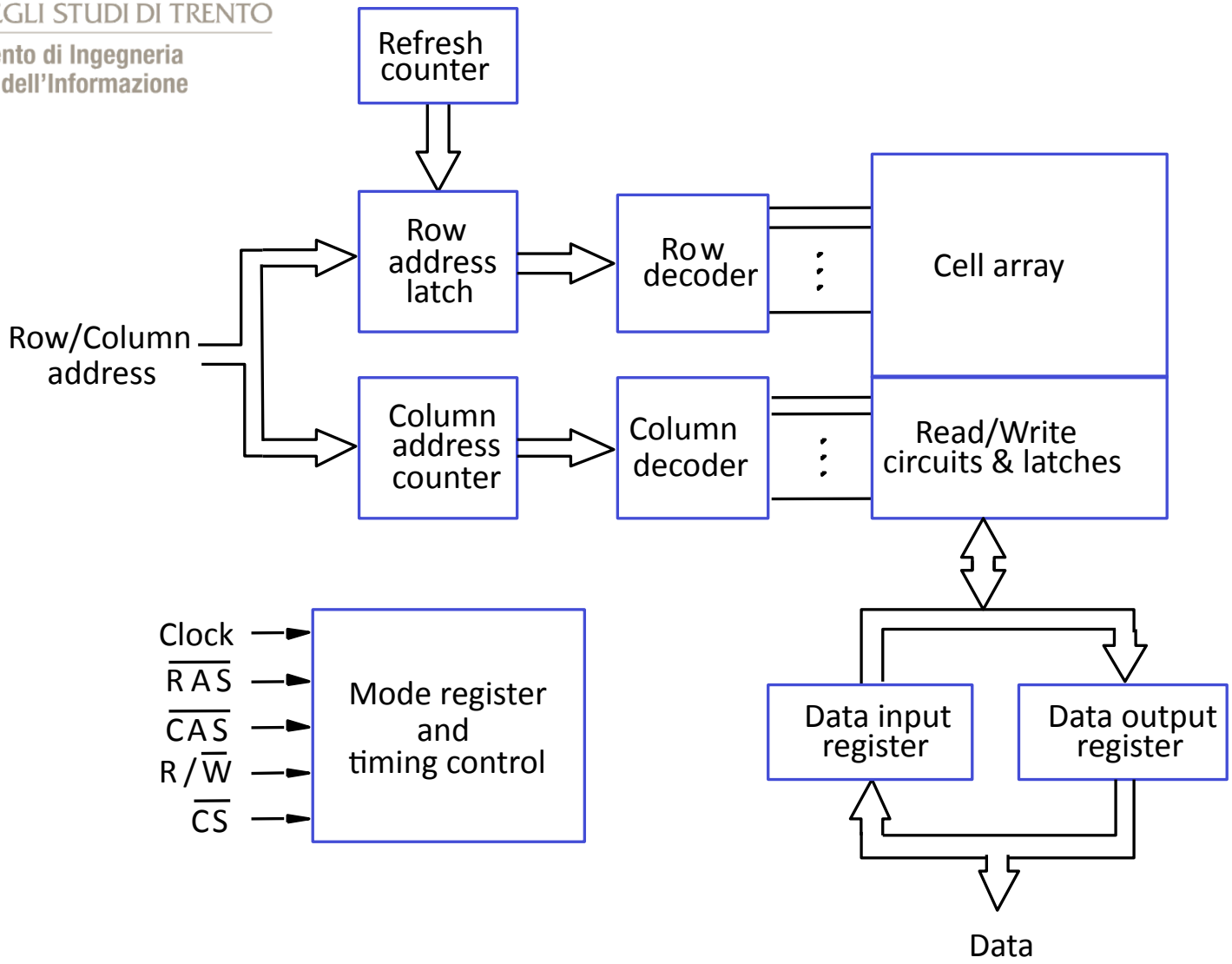
UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

DRAM sincrone (SDRAM)

- Le DRAM visto prima sono dette "asincrone" perché non esiste una precisa temporizzazione di accesso, ma la dinamica viene governata dai segnali RAS e CAS
- Il processore deve tenere conto di questa potenziale "asincronicità"
  - in caso di rinfresco in corso può essere fastidiosa
- Aggiungendo dei buffer (latch) di memorizzazione degli ingressi e delle uscite si può ottenere un funzionamento sincrono, disaccoppiando lettura e scrittura dal rinfresco e si può ottenere automaticamente una accesso FPM pilotato dal clock

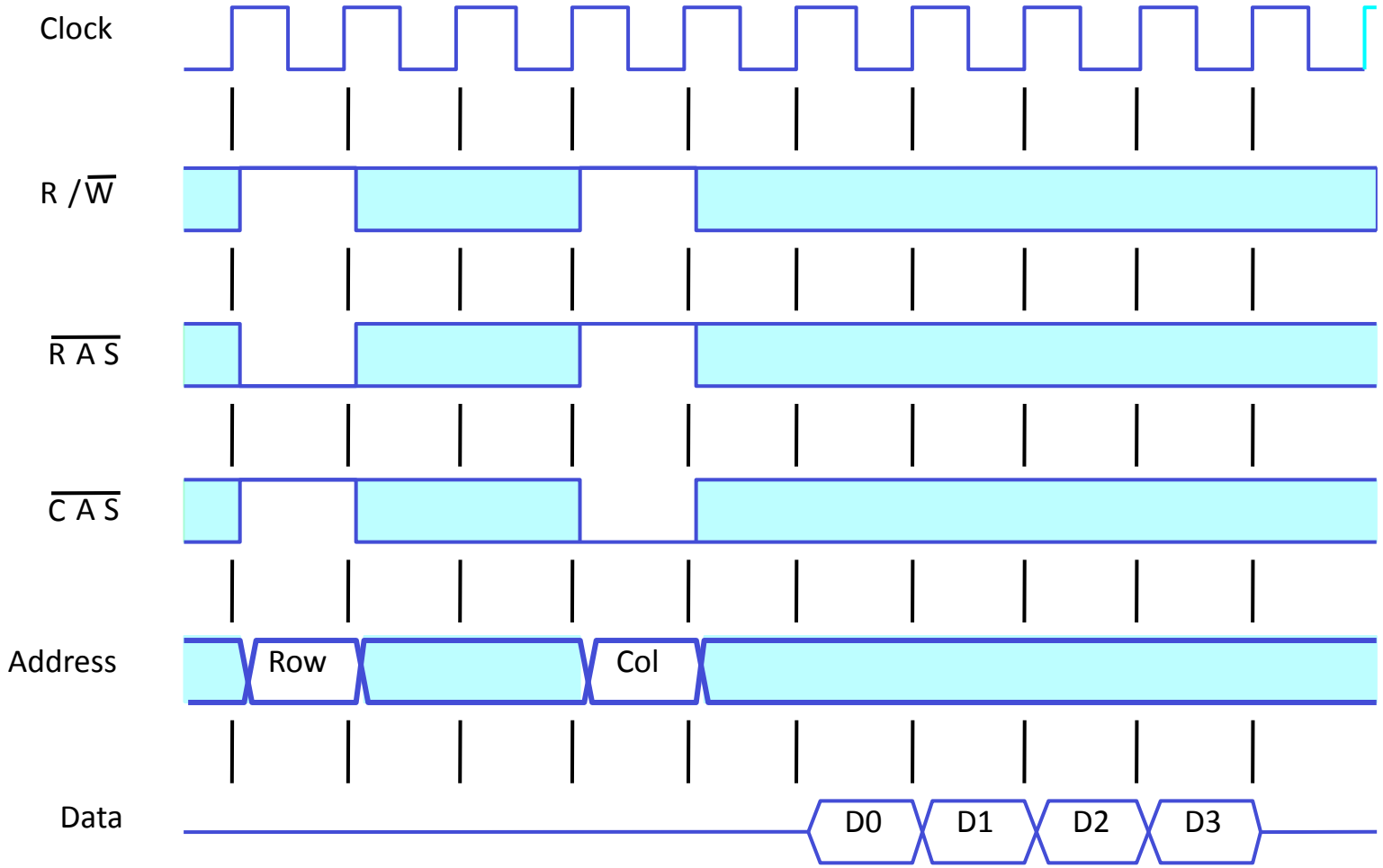
UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Organizzazione base di una SDRAM



UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

SDRAM: esempio di accesso in FPM



UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Velocità e prestazione

- **Latenza:** tempo di accesso ad una singola parola
  - è la misura "principe" delle prestazioni di una memoria
  - da una indicazione di quanto il processore dovrebbe poter aspettare un dato dalla memoria nel caso peggiore
- **Velocità o "banda":** velocità di trasferimento massima in FPM
  - molto importante per le operazioni in FPM che sono legate all'uso di memorie cache interne ai processori
  - è anche importante per le operazioni in DMA, posto che il dispositivo periferico sia veloce

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Double-Data-Rate SDRAM (DDR-SDRAM)

- DRAM statica che consente il trasferimento dei dati in FPM sia sul fronte positivo che sul fronte negativo del clock
- Latenza uguale a una SDRAM normale
- Banda doppia
- Sono ottenute organizzando la memoria in due banchi separati
  - **uno contiene le posizioni pari: si accede sul fronte positivo**
  - **l'altro quelle dispari: si accede sul fronte negativo**
- Locazioni contigue sono in banchi separati e quindi si può fare l'accesso in modo interlacciato

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Gerarchia di memoria

- Abbiamo visto i vari blocchi di memoria con diverse caratteristiche di velocita' e capacita'
- Ma ritorniamo a come ottenere velocita' e capacita' insieme
- Il problema e' noto da molto tempo….

Idealmente si desidererebbe una memoria indefinitamente grande, take che ogni particolare .. parola risulti immediatamente disponible….

Burks, Goldstine, Von Neumann, 1946

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Gestione "piatta"

- Immaginiamo di essere un impiegato che lavora al comune
- Per effettuare il mio lavoro ho bisogno di avere accesso ad un archivio dove sono presenti le varie pratiche
- Ogni volta che mi serve una pratica vado a prenderla, ci opero su e poi la rimetto a posto

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Gestione "piatta"

- Per accedere alla pratica scrivo su un bigliettino lo scaffale dove la pratica puo' essere trovata, lo affido a un attendente e aspetto che me la porti
- Osservazioni:
  - la mia capacita' di memorizzazzione e' molto grande
  - la gran parte del mio tempo (direi il 90%) lo spreco aspettando che l'attendente vada a prendere le pratiche
  - Sicuramente non e' una gestione efficiente del mio tempo
- Posso essere piu' veloce?

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Gestione "veloce"

- In alternativa posso tenere le pratiche sul mio tavolo e operare solo su quelle
- Osservazioni
  - Sicuramente non perdo tempo (non ho da aspettare attendenti che vadano in su e in giu')
  - Tuttavia il numero massimo di pratiche che possono gestite e' molto basso
- Posso operare su piu' dati?

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Capra e Cavoli

- Per riuscire ad avere ad un tempo velocita' e ampiezza dell'archivio posso fare due osservazioni:

1. Il numero di pratiche su cui posso concretamente lavorare in ogni giornata e' limitato
2. Se uso una pratica, quesi sicuramente dovro' ritornare su di essa in tempi brevi….tanto vale tenersela sul tavolo

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Un approccio gerarchico

- L'idea e' che ho un certo numero di posizioni sulla mia scrivania
  - Man mano che mi serve una pratica la mando a prendere
  - Se ho la scrivania piena faccio portare a posto quelle pratiche che non mi servono piu' per fare spazio
- In questo modo posso contare su un'amplia capacita' di memorizzazione, *ma* la maggior parte delle volte accedo ai dati molto velocemente

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Torniamo ai calcolatori

- Fuor di metafora, possiamo fare calcolatori dando ai programmi l'illusione di avere uno spazio di memoria molto grande ma con grande velocità
- Questo e' possibile in forza di due principi
  - Principio di localita' spaziale
  - Principio di localita' temporale

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Localita' temporale

- Quando si fa uso di una locazione, si riutilizzera' presto con elevata probabiita'.
- Esempio.

```
Ciclo: sll $t1, $s2, 2
       add $t1, $t1, $s6
       lw  $t0, 0($t1)
       bne $t0, $s5, Esci
       addi$s2, $s2, 1
       j    Ciclo
Esci: ….
```

Queste istruzioni vengono ricaricate ogni volta che si esegue il ciclo

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Localita' spaziale

- Quando si fa riferimento a una locazione, nei passi successivi si fara' riferimento a locazioni vicine

```
Ciclo: sll $t1, $s2, 2
       add $t1, $t1, $s6
       lw  $t0, 0($t1)
       bne $t0, $s5, Esci
       addi$s2, $s2, 1
       j    Ciclo
Esci: ….
```

ISTRUZIONI: la modalita' di esecuzione normale e' il prelievo di istruzioni successive

DATI: Quando si scorre un array si va per word successive

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Qualche dato

Facciamo riferimento a qualche cifra (relativa al 2008)

Tecnologia di Memoria	Tempo di accesso tipico	$ per GB (2008)
SRAM	0.5-2.5ns	$2000 - $5000
DRAM	50 – 70 ns	$20 - $75
Dischi magnetici	5 000 000 – 20 000 000 ns	$0.2 - $2

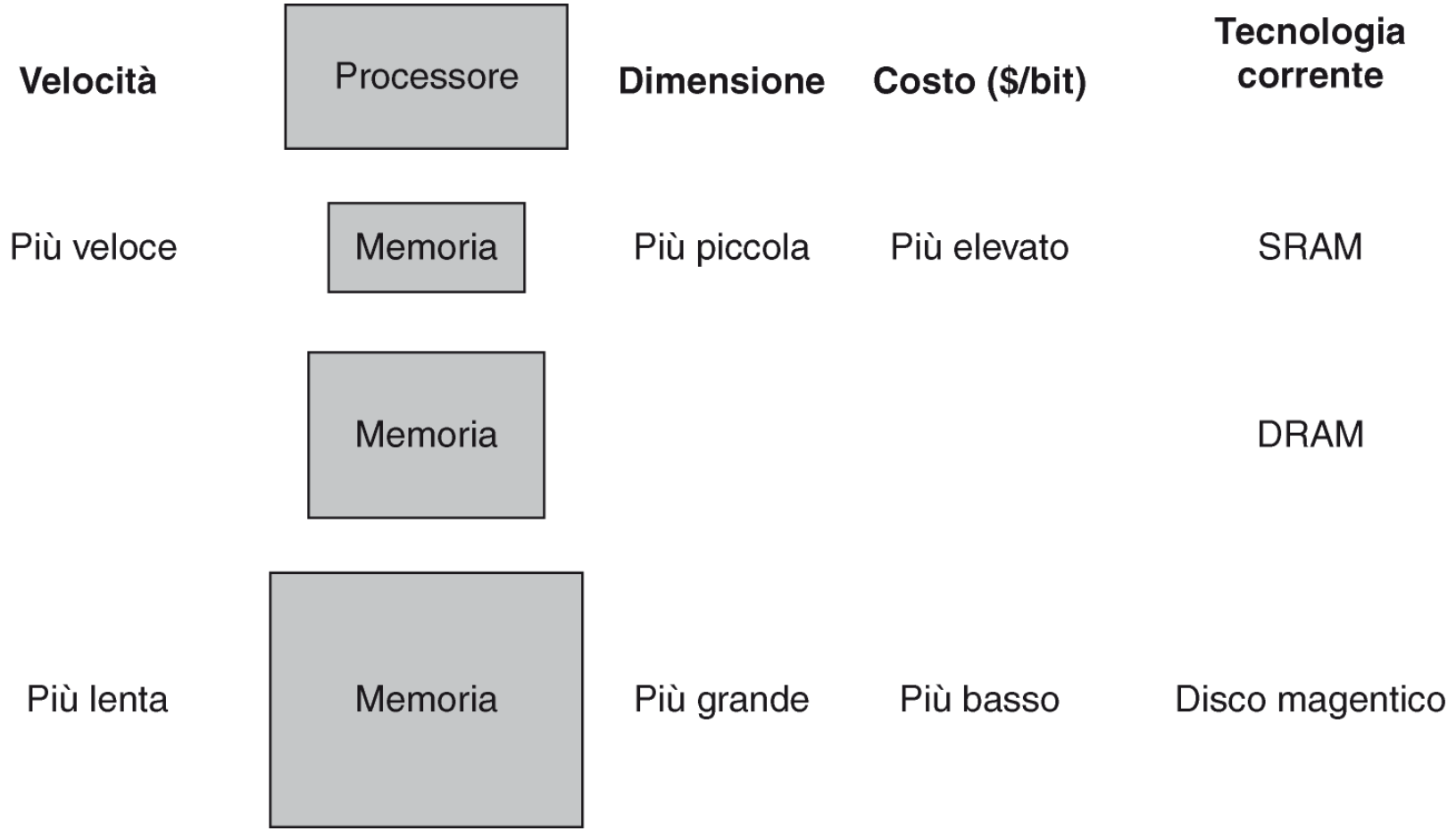
Queste cifre suggeriscono l'idea della gerarchia di memoria

- Memorie piccole e veloci vicino al processore
- Memorie grandi e lente lontane dal processore

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

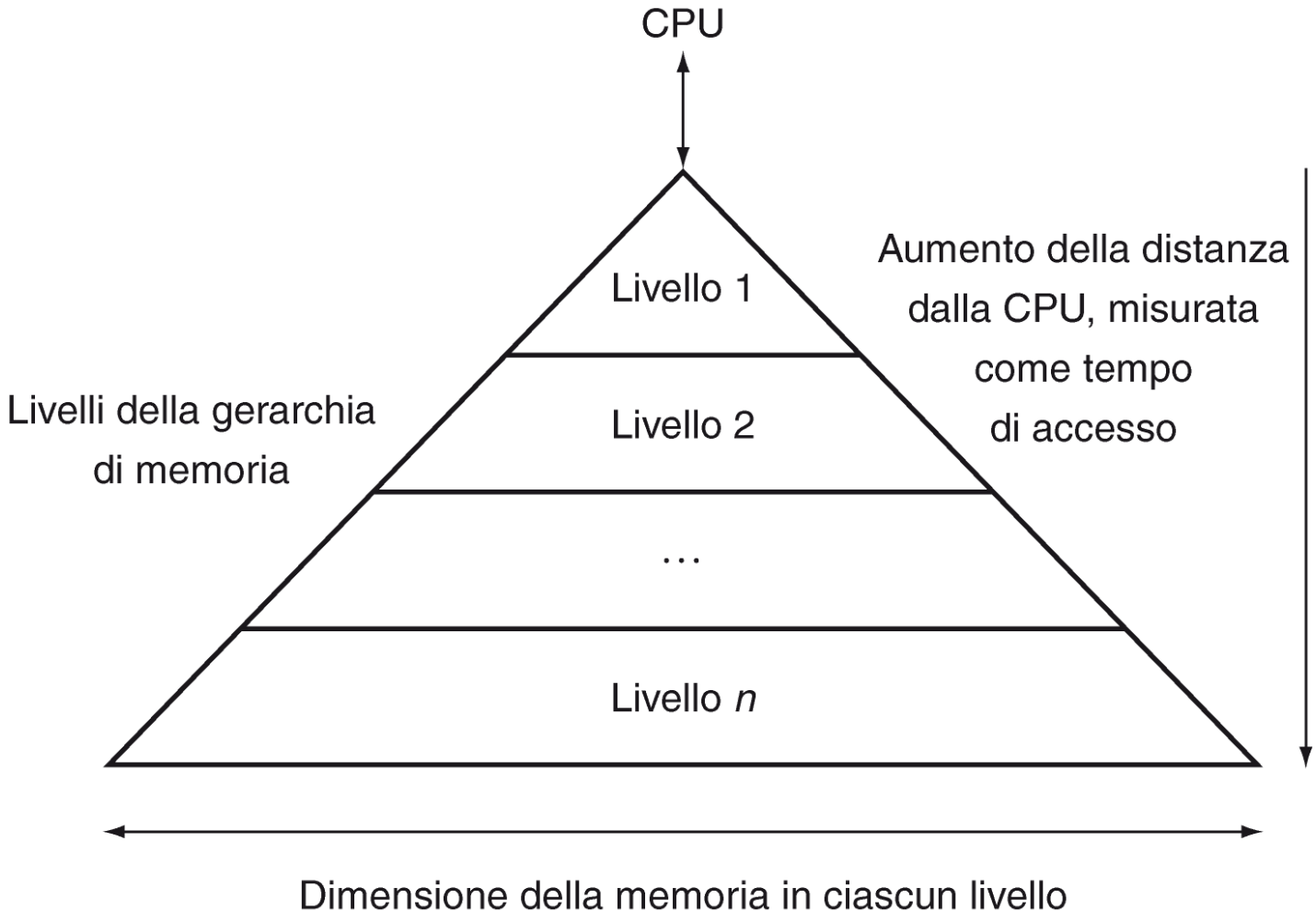
Gerarchia di memoria

- Struttura base



UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

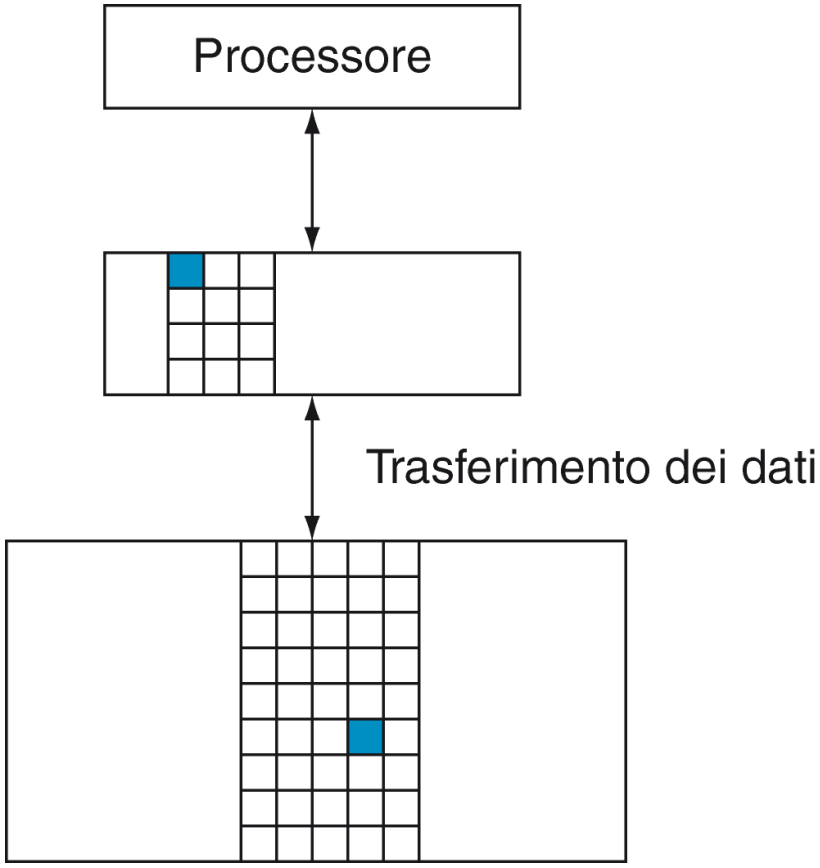
Struttura della gerarchia



UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Esempio

- Due livelli



UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Terminologia

- **Blocco o Linea:** unita' minima di informazione che puo' essere presente o assente nel livello superiore
  - Un faldone nell'esempio di un archivio
- **Hit rate**: Frequenza di successo = frazione degli accessi in cui trovo il dato nel lovello superiore
  - Quante volte trovo il faldone che mi serve nella scrivania
- **Miss Rate:** 1 – hitrate: frazione degli accessi in cui non trovo il dato nel livello superiore
  - Quante volte devo andare a cercare un faldone in archivio
- **Tempo di Hit:** Tempo che mi occorre per trovare il dato quando lo trovo nel livello superiore
  - Quanto mi ci vuole a leggere un documento nel faldone
- **Penalita' di miss:** quanto tempo mi ci vuole per accedere al dato se non lo trovo nel livelo superiore
  - Tempo per spostare il faldone + tempo di accesso al documento nel faldone

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Considerazioni

- La penalita' di miss e' molto maggiore del tempo di hit (e anche del trasferimento in memoria di un singolo dato)
  - Da cui il vantaggio
- quindi e' cruciale che non avvenga troppo spesso
  - in questo ci aiuta il principio di localita' che il programmatore deve sfruttare al meglio

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

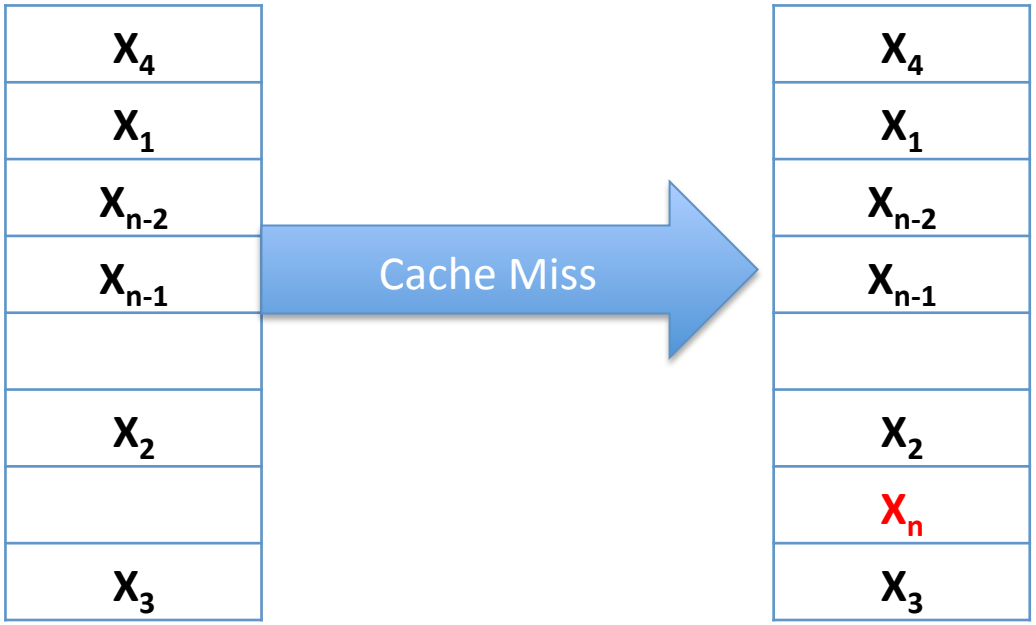
Cache

- **Cache**: posto sicuro [nascosto] dove riporre le cose
- Nascosta perche' il programmatore non la vede direttamente
  - l'uso gli e' interamente trasparente
- L'uso della cache fu sperimentato per la prima volta negli anni 70 e da allora e' divenuto assolutamente prevalente in tutti i calcolator

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Un esempio semplice

- Partiamo da un semplice esempio in cui la i blocchi di cache siano costituiti da una sola word.
- Supponiamo che a un certo punto il processore richieda la parola Xn che non e' in cache



UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Domande

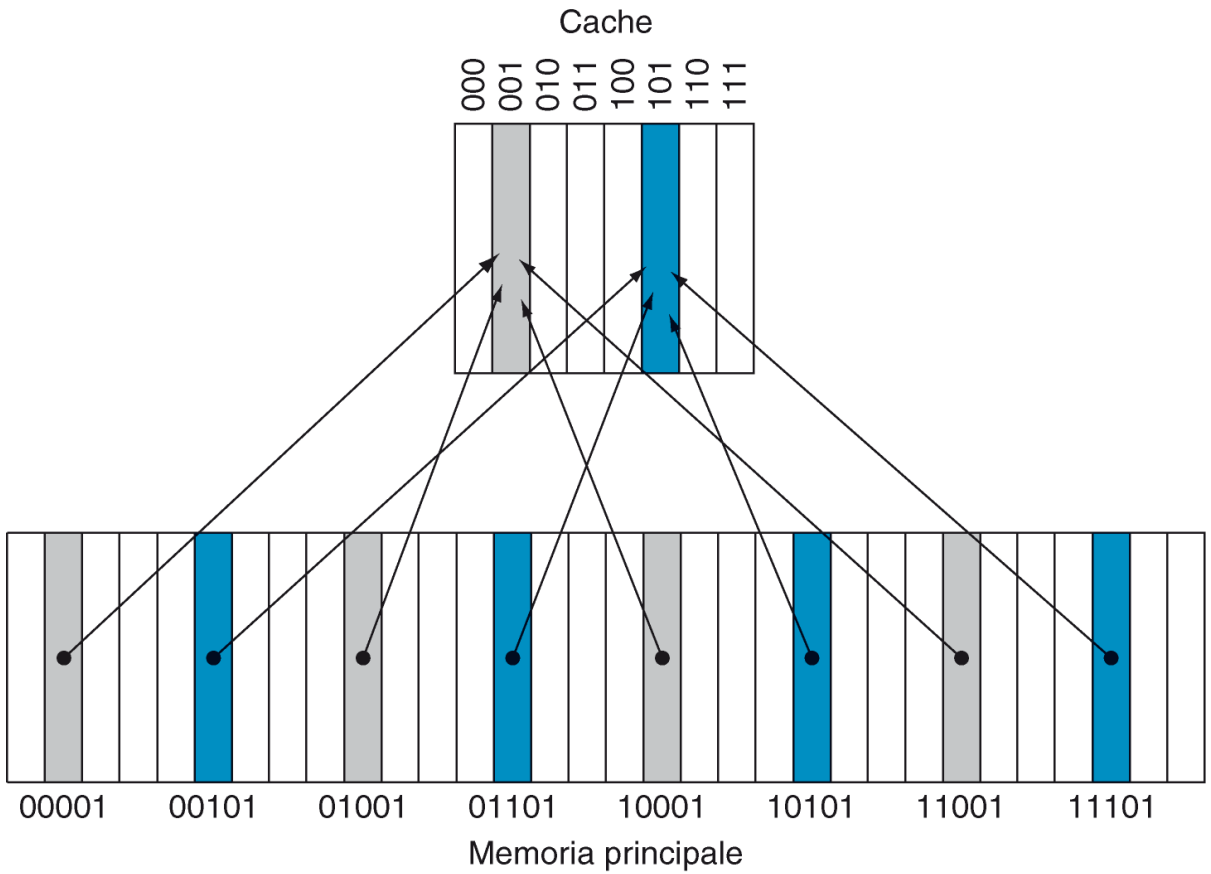
- Come facciamo a capire se un dato richiesto e' nella cache?
- Dove andiamo a cercare per sapere se c'e?

- *Cache direct mapped:* a ogni indirizzo in della memoria corrisponde una precisa locazione della cache
- Possibilita': Indirizzo locazione dove un indirizzo e' mappato = indirizzo blocco *modulo* numero di blocchi in cache
  - Se il numero di elementi nella cache e' potenza di due, ci e' sufficiente prendere i bit meno significativi dell'indirizzo in numero pari al logaritmo in base due della dimensione della cache

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Esempio

- Se la nostra cache dispone di 8 parole devo prendere i tre bit meno significativi



UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Problema

- Siccome molte parole posso essere mappate su un blocco di cache, come facciamo a capire se, in un dato momento, vi si trova l'indirizzo che serve a noi?

- ***Si ricorre a un campo, detto tag, che contiene una informazione sufficiente a risalire al blocco correntemente mappato in memoria***
- *Ad esempio possiamo utilizzare i bit piu' significativi di una parola, attualmente non usati per trovare la locazione in cache dove l'indirizzo e' mappato.*

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Validita'

- Usiamo i bit piu' significativi (due nell'esempio fatto) per capire se nella linea di cache memorizziamo l'indirizzo richiesto
- Inoltre abbiamo un bit di validita' che ci dice se quello che memorizziamo in una linea ad un certo momento sia o meno valido

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Esempio

Indice	V	Tag	Dati
000	N		
001	N		
010	N		
011	N		
100	N		
101	N		
110	N		
111	N		

a. Lo stato iniziale della cache dopo l'accensione del calcolatore

Accesso a 10110: Miss

Indice	V	Tag	Dati
000	N		
001	N		
010	N		
011	N		
100	N		
101	N		
110	S	10_{due}	Memoria (10110_{due})
111	N		

b. Dopo avere gestito una miss all'indirizzo (10110_{due})

Accesso a 11010: Miss

Accesso a 11010: hit

Indice	V	Tag	Dati
000	N		
001	N		
010	S	11_{due}	Memoria (11010_{due})
011	N		
100	N		
101	N		
110	S	10_{due}	Memoria (10110_{due})
111	N		

c. Dopo avere gestito una miss all'indirizzo 11010_{due}

Indice	V	Tag	Dati
000	S	10_{due}	Memoria (10000_{due})
001	N		
010	S	11_{due}	Memoria (11010_{due})
011	N		
100	N		
101	N		
110	S	10_{due}	Memoria (10110_{due})
111	N		

d. Dopo avere gestito una miss all'indirizzo 10000_{due}

Indice	V	Tag	Dati
000	S	10_{due}	Memoria (10000_{due})
001	N		
010	S	11_{due}	Memoria (11010_{due})
011	S	00_{due}	Memoria (00011_{due})
100	N		
101	N		
110	S	10_{due}	Memoria (10110_{due})
111	N		

e. Dopo avere gestito una miss all'indirizzo 00011_{due}

Accesso a 10010: miss

Indice	V	Tag	Dati
000	S	10_{due}	Memoria (10000_{due})
001	N		
010	S	10_{due}	Memoria (10010_{due})
011	S	00_{due}	Memoria (00011_{due})
100	N		
101	N		
110	S	10_{due}	Memoria (10110_{due})
111	N		

f. Dopo avere gestito una miss all'indirizzo 10010_{due}

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Esempio MIPS 32 bit

- Esempio
  - Indirizzo su 32 byte
  - cache mappatura diretta
  - dimensioni della cache 2^n blocchi, di cui n bit usati per l'indice
  - dimensione del blocco di *cache* 2^m parole ossia 2^{m+2} byte

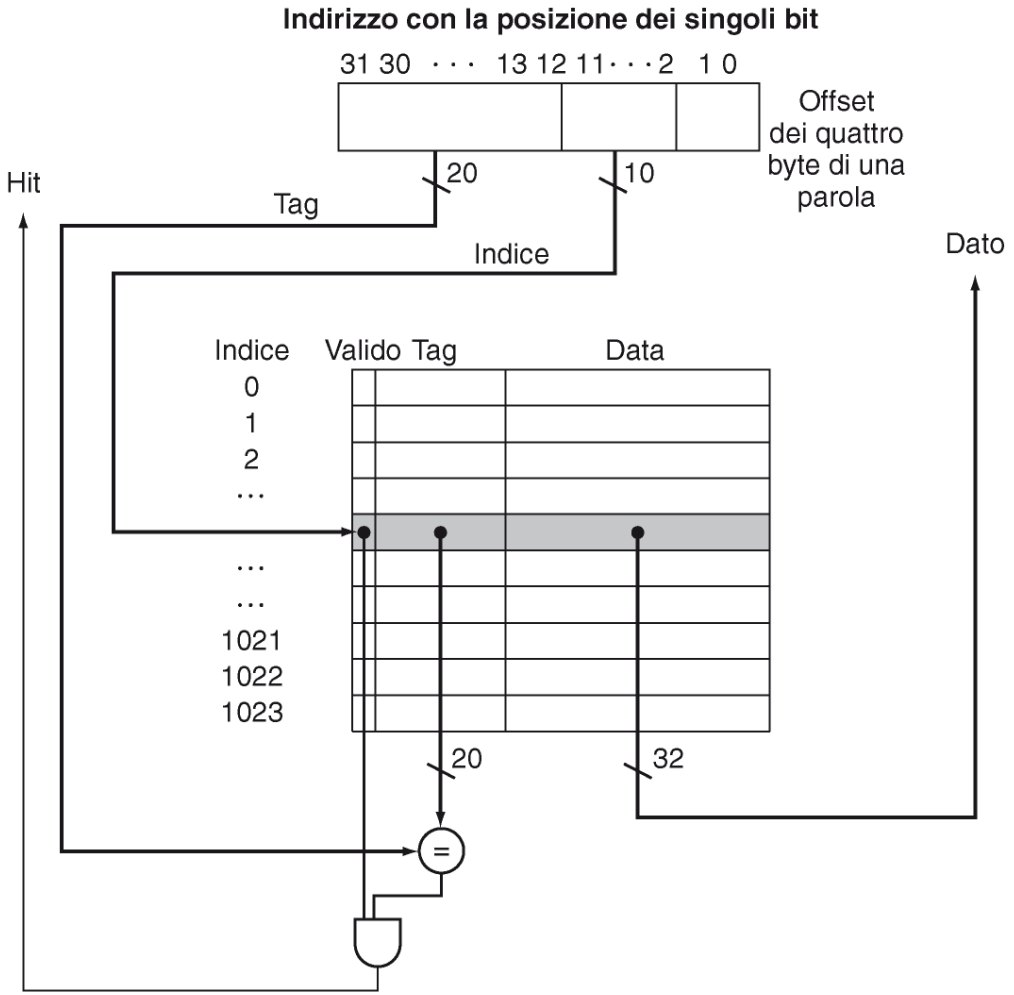
In questo caso la dimensione del tag e' data da

32-(n+m+2)

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Schema di risoluzione

- Un indirizzo viene risolto in cache con il seguente schema:



UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Esempio

- Si consideri una cache con 64 blocchi di 16 byte ciascuno. A quale numero di blocco corrisponde l'indirizzo 1200 espresso in byte?

- Blocco identificato da
  (indirizzo blocco) modulo (numero blocchi in cache)
- Dove

$$\text{Indirizzo Blocco} = \frac{\text{Indirizzo del Dato in byte}}{\text{Byte per blocco}}$$

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Esempio

- Quindi l'indirizzo del blocco e'
  1200/16=75

- Blocco contente il dato e'
  75 modulo 64 = 11

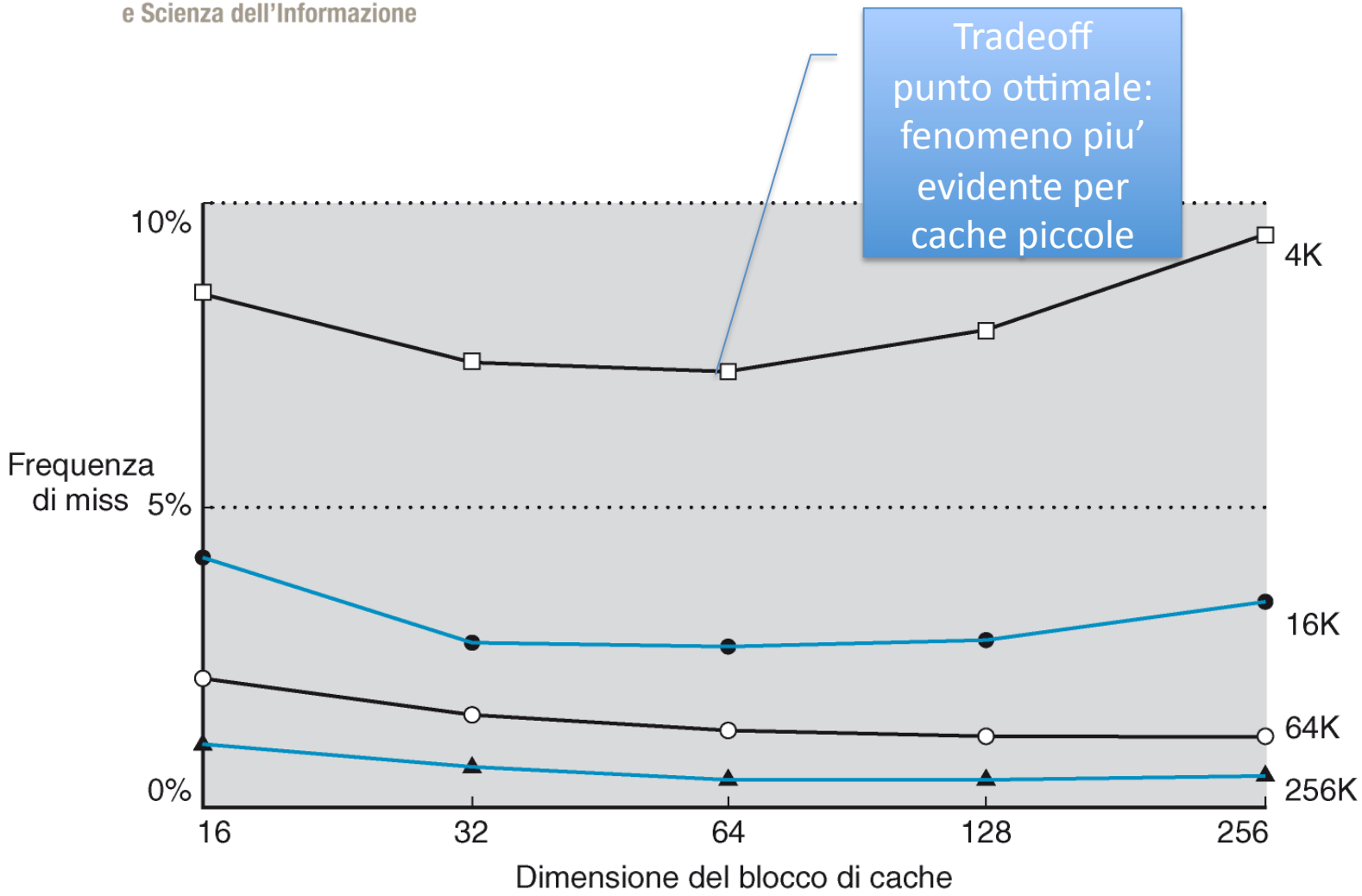
UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Tradeoff

- Dimensioni di linea di cache molto grandi esaltano la localita' spaziale e da questo punto di vista diminuiscono le probabilita' di miss
- Tuttavia avere pochi blocchi diminuisce l'efficacia nello sfruttamento della localita' temporale
- Quindi abbiamo un tradeoff
- Inoltre avere dei *miss* con blocchi grandi porta a un costo di gestione alto (bisogna spostare molti byte)

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Frequenza delle miss



UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Gestione delle miss

- La presenza di una cache non modifica molto il funzionamento del processore pipeline fino a che abbiamo delle hit
  - Il processore non si accorge della presenza della cache
  - In caso di miss, bisogna generare uno stallo nella pipeline e gestire il trasferimento da memoria principale alla cache (ad opera della circuiteria di controllo)

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Gestione delle Miss

- Ad esempio, per una miss sulla memoria istruzioni, bisognera':
  1. Inviare il valore PC – 4 alla memoria (PC viene incrementata all'inizio, quindi la miss e' su PC-4)
  2. Comandare alla memoria di eseguire una lettura e attendere il completamento
  3. Scrivere il blocco che proviene dalla memoria della cache aggiornando il tag
  4. Far ripartire l'istruzione dal fetch, che stavolta trovera' l'istruzione in cache

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Scritture

- Gli accessi in lettura alla memoria dati avvengono con la stessa logica
- Gli accessi in scrittura sono un po' piu' delicati perche' possono generare problemi di consistenza
- Una politica e' la cosidetta **write-through**
  - Ogni scrittura viene direttamente effettuata in memoria principale (sia che si abbia una hit che una miss)
  - In questo modo non ho problemi di consistenza, ma le scritture sono molto costose
  - Posso impiegare un buffer di scrittura (una coda in cui metto tutte le scritture che sono in attesa di essere completate)

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

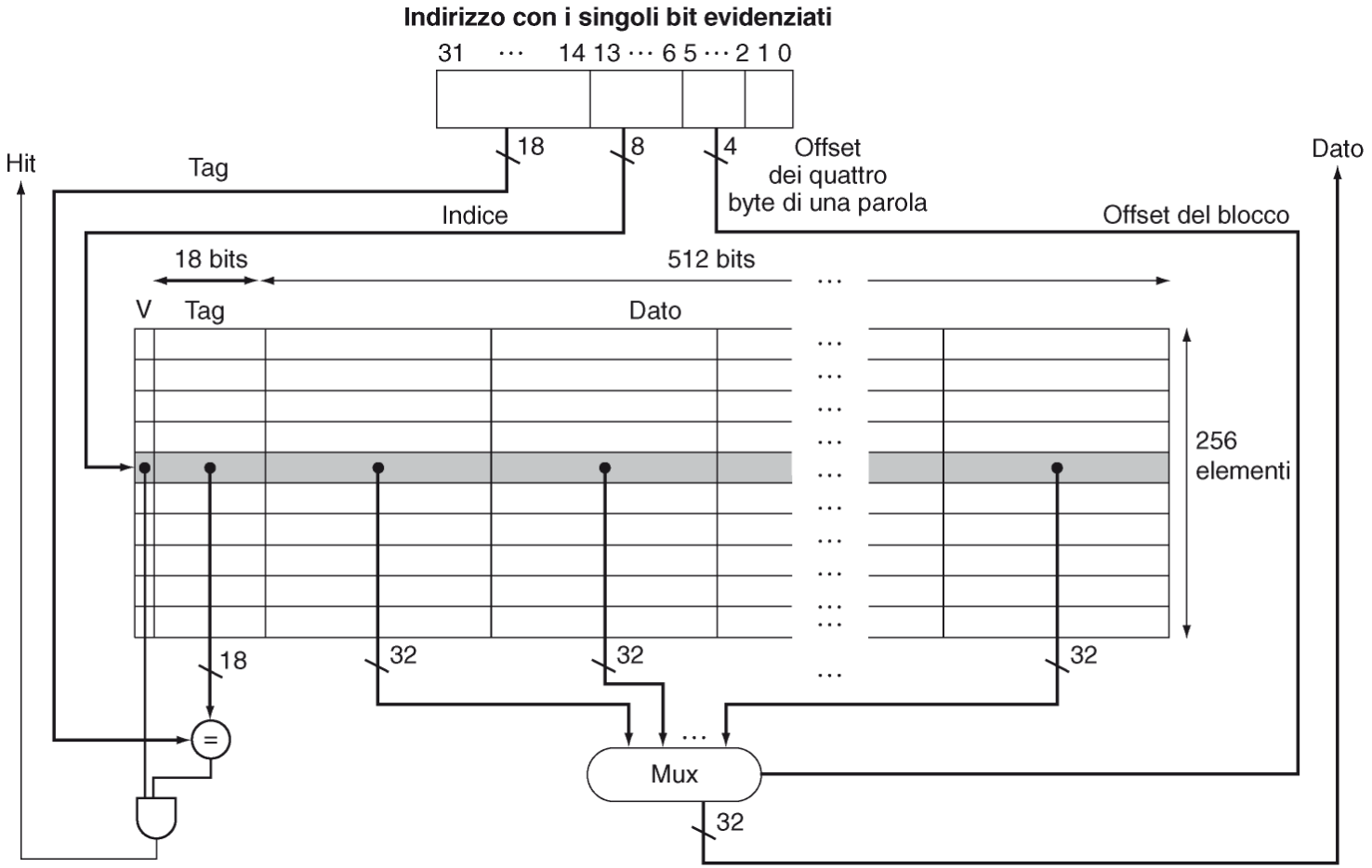
Scritture

- Un'altra possibile politica e' la **write-back**
  - Se il blocco e' in cache le scritture avvengono localmente in cache e l'update viene fatto solo quando il blocco viene rimpiazzato (o quando una locazione nel blocco viene acceduta da un altro processore)
  - Questo schema e' conveniente quando il processore genera molte scritture e la memoria non ce la fa a stargli dietro.

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Esempio
FastMath (basato su MIPS)

- Cache di 16K
- 16 parole per blocco
- Possibilita' di operare in write through o in write back



UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Esempio
FastMath (basato su MIPS)

Tipiche performance misurate su benchmark SPEC 2000

Frequenza di miss per le istruzioni	Frequenza di miss per i dati	Frequenza di miss totale
0,4%	11,4%	3,2%

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Cache set associative

- Le cache a mappatura diretta sono piuttosto semplici da realizzare
- Tuttavia hanno un problema: se ho spesso bisogno di locazioni di memoria che si mappano sullo stesso blocco, ho cache miss in continuazione
- All'estremo opposto ho una cache completamente associativa
  - Posso mappare qualsiasi blocco in qualsiasi blocco di cache

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Cache full associative

- Il problema per le cache full associative e' che devo cercare ovunque il dato (il tag e' tutto l'indirizzo del blocco)
- Per effettuare la ricerca in maniera efficiente, devo farla su tutti i blocchi in parallelo
- Per questo motivo ho bisogno di n comparatori (uno per ogni blocco di cache che operino in parallelo)
- Il costo HW e' cosi' alto che si puo' are solo per piccolo cache

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Cache set associative

- Le cache set associative sono un via di mezzo tra le due che abbiamo visto
- In sostanza ogni blocco puo' essere mappato su n blocchi diversi (vie)
- Quindi combiniamo due idee
  - associamo ciascun blocco a una linea (una degli n blocchi su cui possiamo mappare il blocco)
  - All'interno della linea effettuiamo una ricerca parallela come se avessimo una cache completamente associativa

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Mappatura del blocco

- In una cache a mappatura diretta il blocco viene mappato nel blocco dato da:

  (Numero Blocco) modulo (numero *blocchi* in cache)

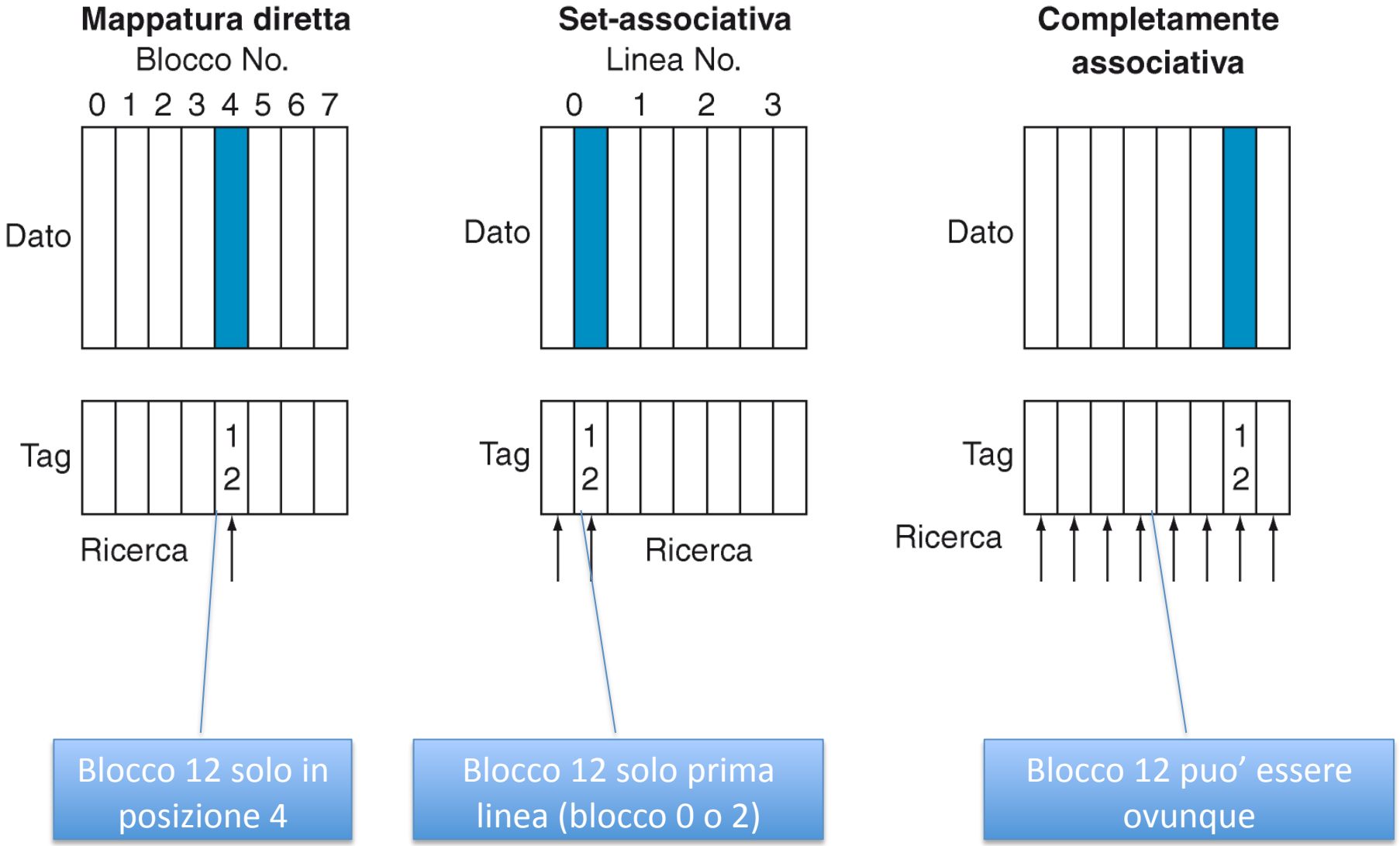
- In una cache set associative la linea che contiene il blocco viene individuata da:

  (Numero Blocco) modulo (numero *linee* in cache)

- Per trovare il blocco All'interno della linea dobbiamo confrontare il tag con tutti i tag dei blocchi della linea

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Posizione del blocco



UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Varie configurazioni

Cache set-associativa a una via
(a mappatura diretta)

Blocco	Tag	Dato
0		
1		
2		
3		
4		
5		
6		
7		

Cache set-associativa a due vie

Linea	Tag	Dato	Tag	Dato
0				
1				
2				
3				

Cache set-associativa a quattro vie

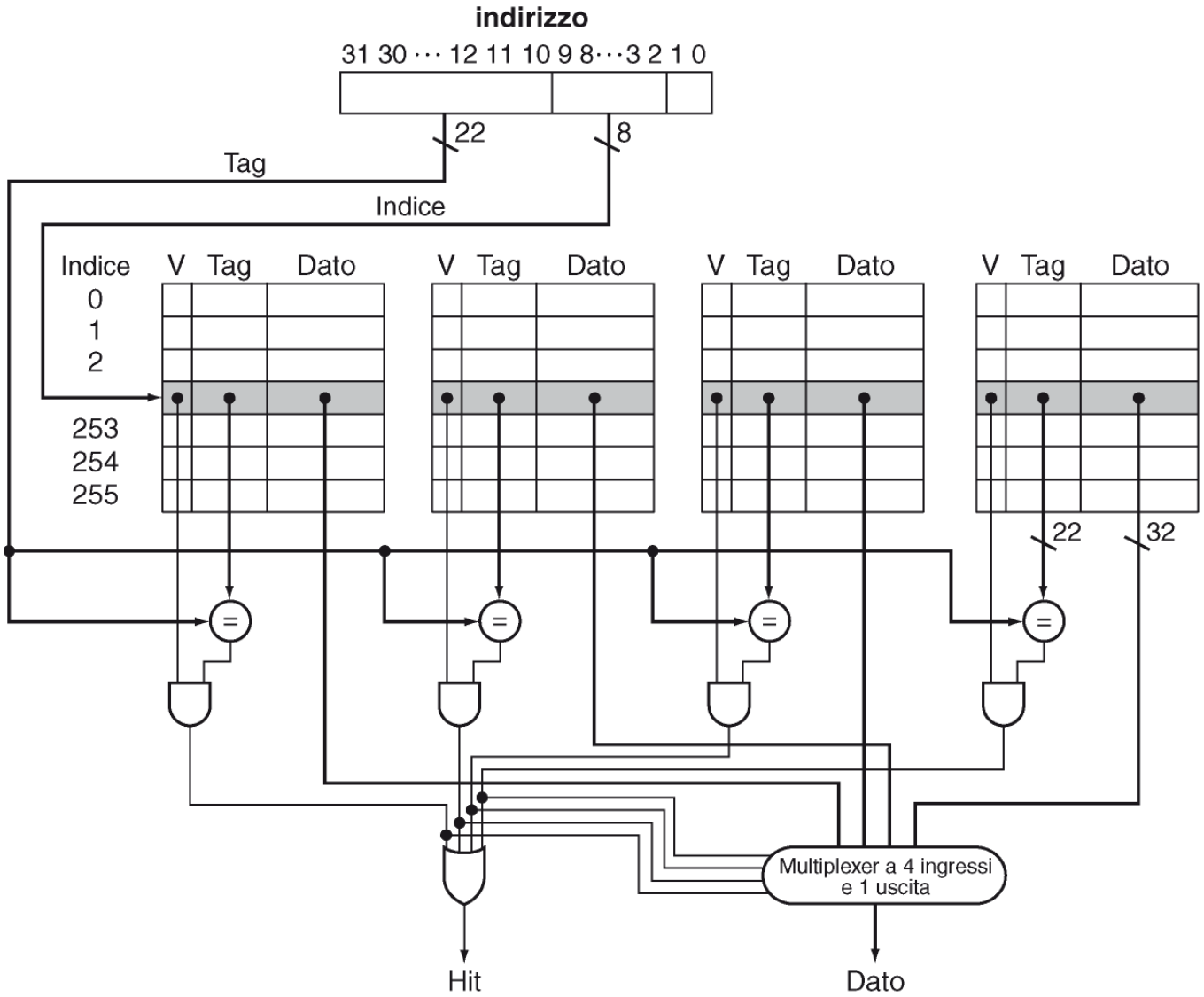
Linea	Tag	Dato	Tag	Dato	Tag	Dato	Tag	Dato
0								
1								

Cache set-associativa a otto vie (completamente associativa)

Tag	Dato	Tag	Dato	Tag	Dato	Tag	Dato	Tag	Dato	Tag	Dato	Tag	Dato	Tag	Dato

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Schema per cache a 4 vie



UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

Vantaggi dell'associativita'

Associatività	Frequenza di miss
1	10,3%
2	8,6%
4	8,3%
8	8,1%

Chiaramente aumentando l'associativita' abbiamo vantaggi (frequenza di miss) e svantaggi (complessita'). La scelta viene fatto tenendo presente questo tradeoff

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

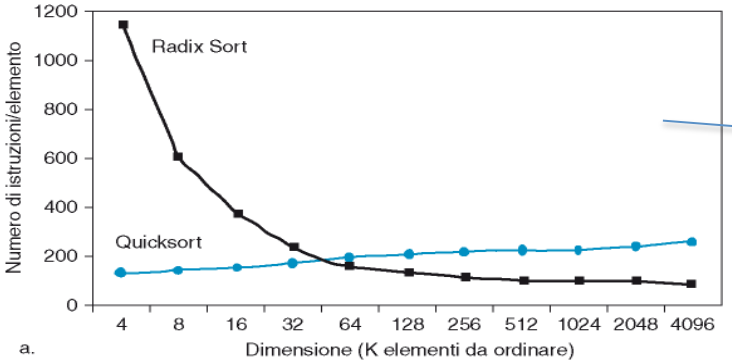
Un probema in piu'

- Nelle cache a mappatura diretta quando ho una cache miss sicuramente so chi sostituire (l'unico blocco in cui posso mapparmi)
- Nelle cache associative ho piu' scelte. Se la linea e' piena chi sostituisco
- Varie politiche
  - FIFO
  - Least Recently Used

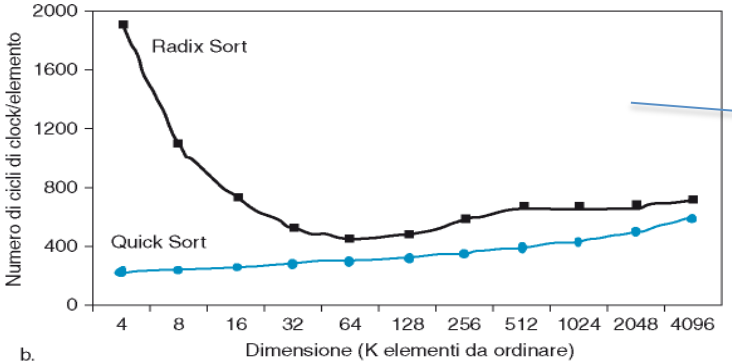
Richiede una serie di bit in piu' per contare l'ultimo accesso

UNIVERSITÀ DEGLI STUDI DI TRENTO
Dipartimento di Ingegneria
e Scienza dell'Informazione

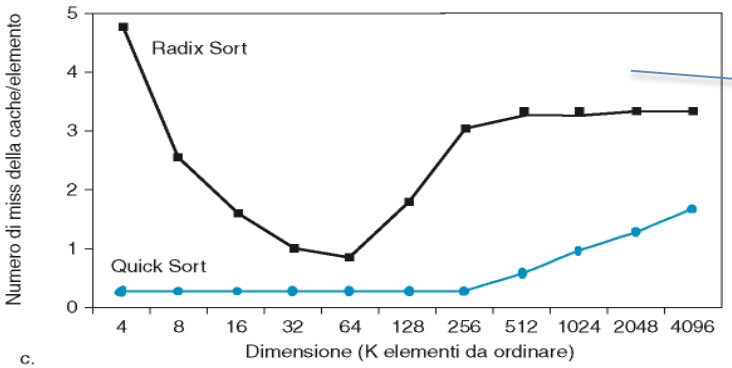
Ok, ma a che ci serve questa roba?



Numero di istruzioni per elemento da ordinare



Tempo di esecuzione



Cache miss