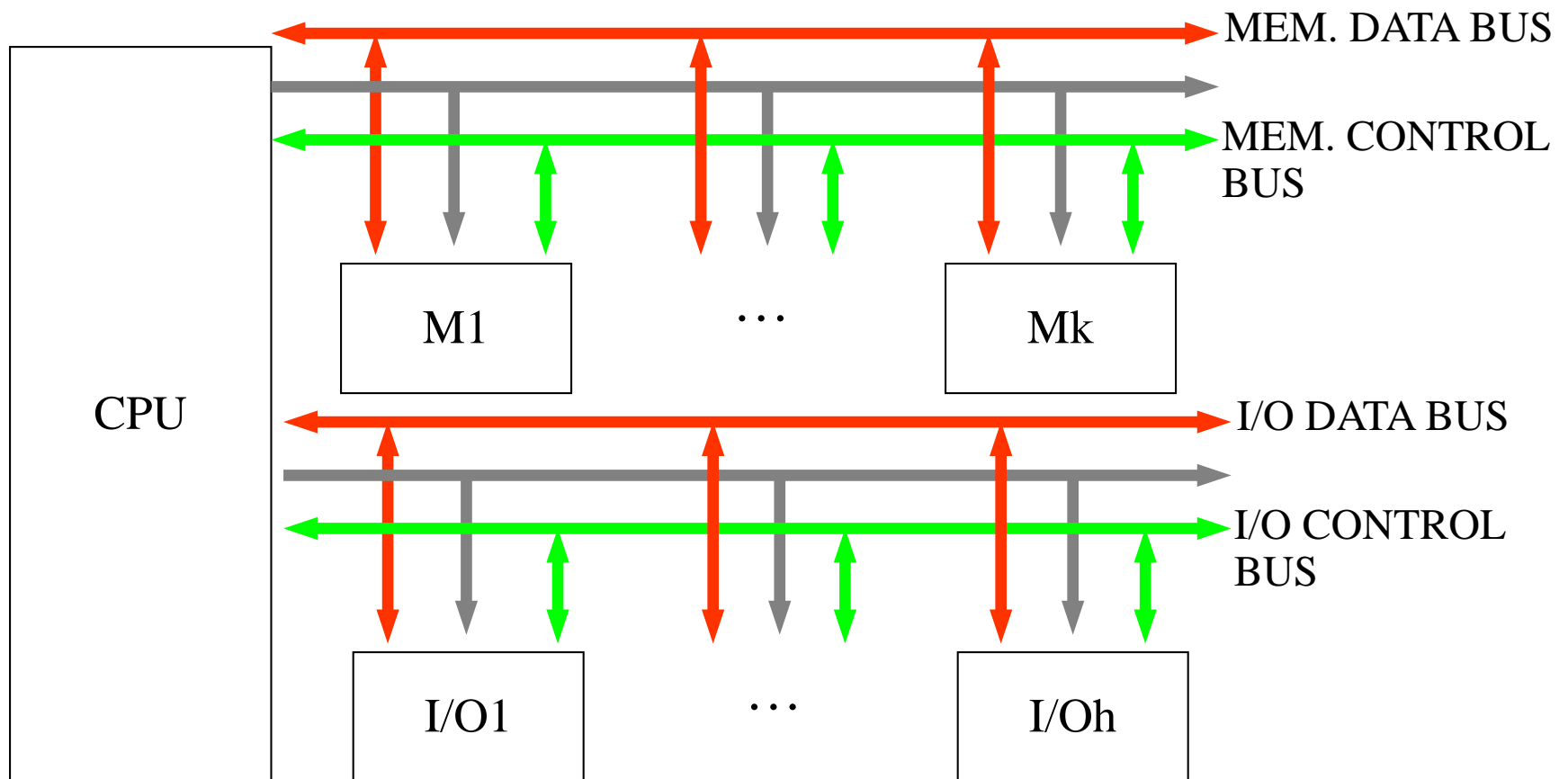
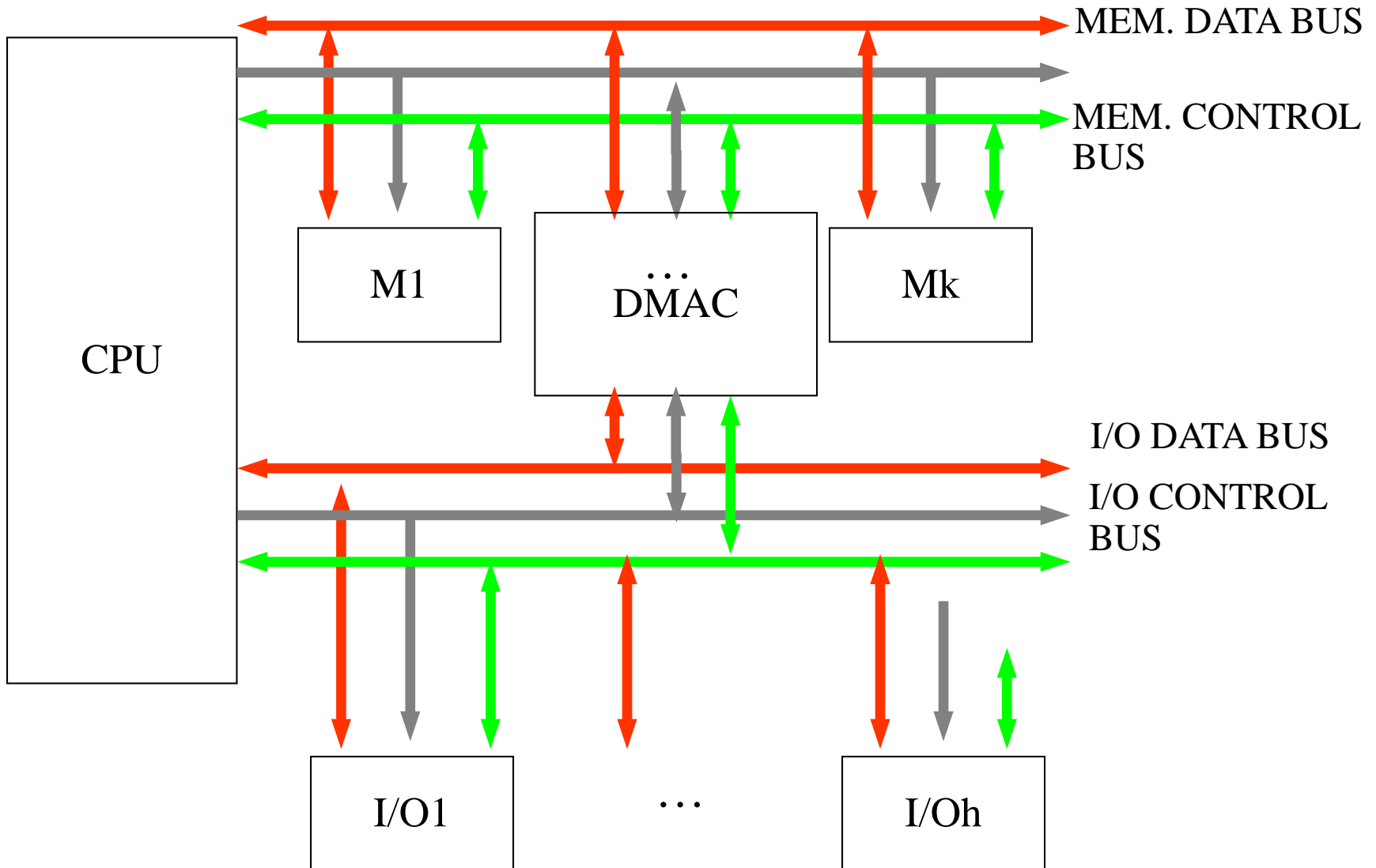


Comunicazione del processore
con le memorie e con i dispositivi
di I/O

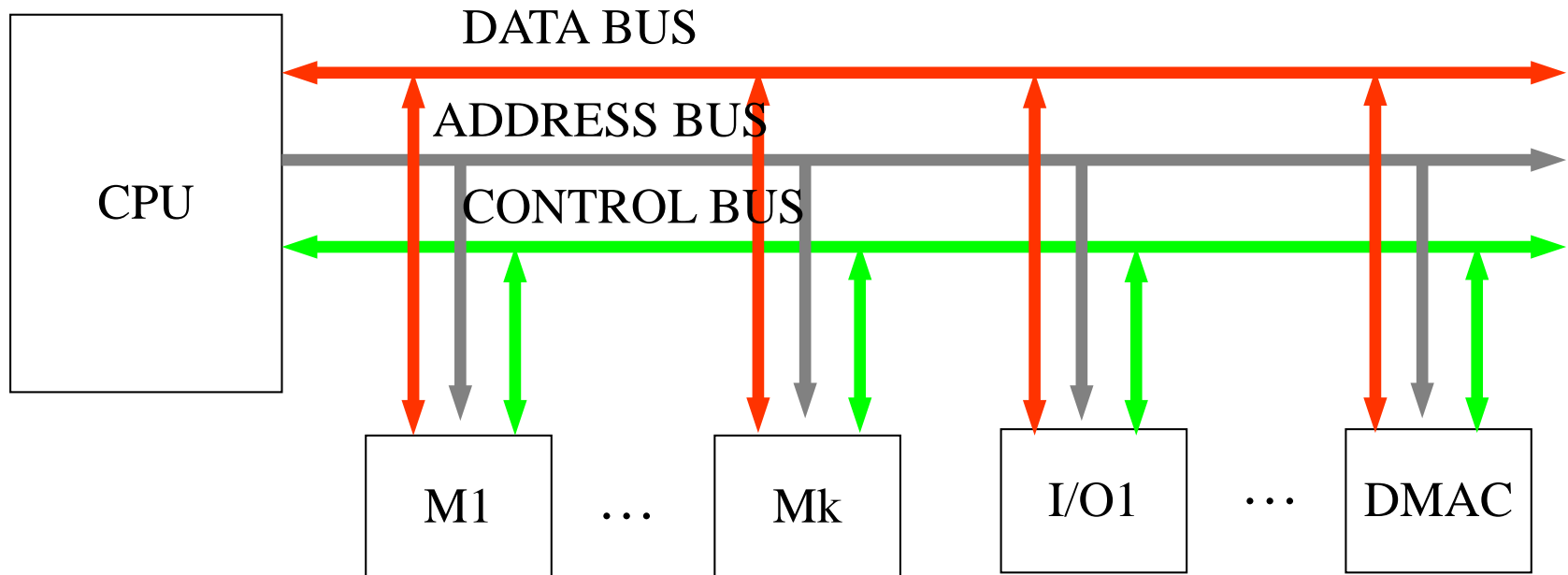
Possibile organizzazione di un calcolatore a due bus



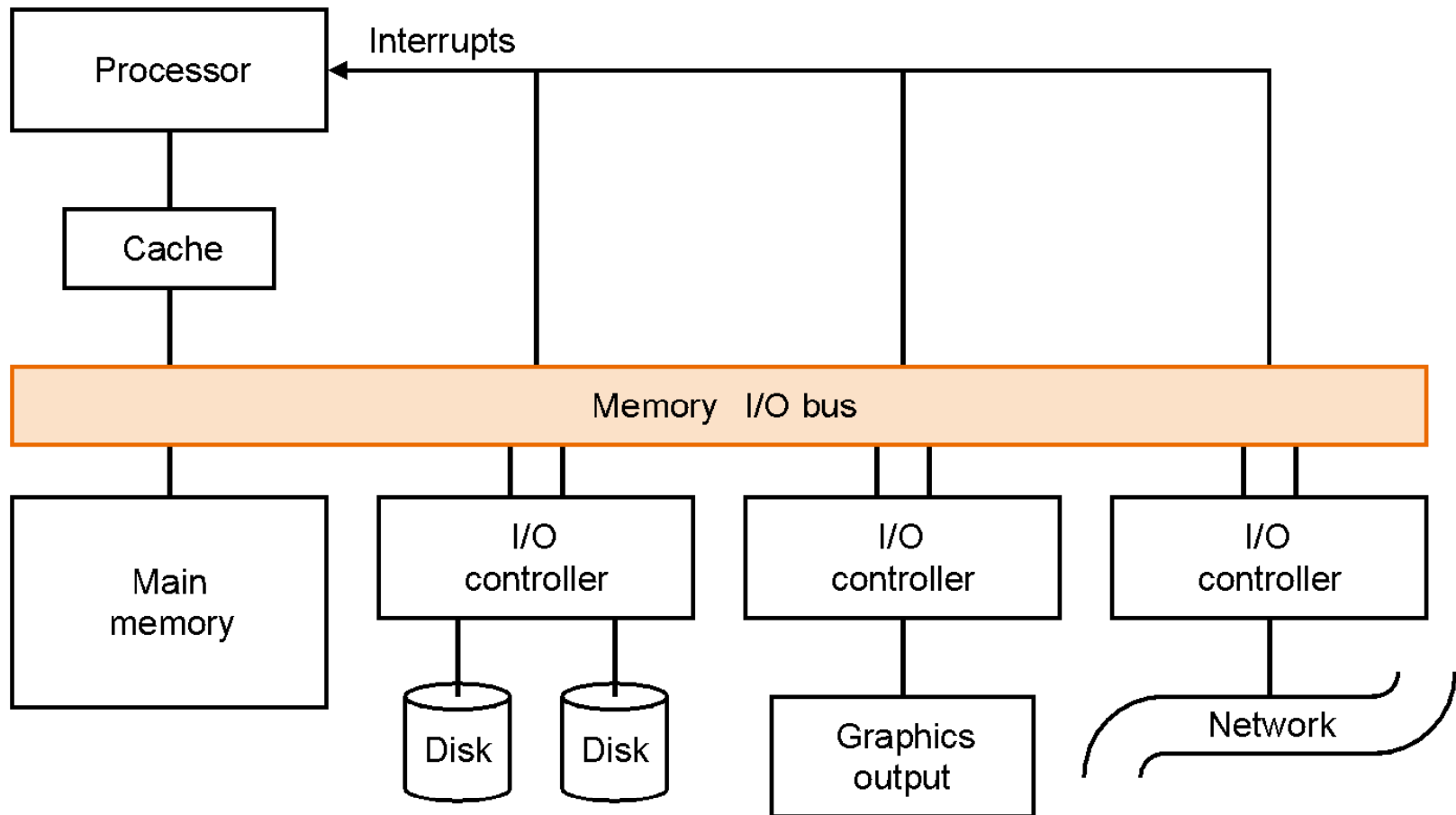
Variante con DMAC



Possibile organizzazione di un calcolatore ad un bus



Possibile organizzazione di un calcolatore ad un bus



Bus

- Rappresenta il canale di comunicazione tra le varie componenti del calcolatore
 - Mezzo di trasmissione condiviso, al quale sono collegati più componenti
 - Un calcolatore contiene svariati bus
- Potenziale collo di bottiglia essendo le sue prestazioni limitate da:
 - Lunghezza
 - Numero di dispositivi connessi
- Bus composto da:
 - **Linee dati (e indirizzi)**
 - Informazioni: dati, indirizzi (anche comandi complessi)
 - Ampiezza: numero di linee dati
 - Possibile condividere le linee per dati e indirizzi (*multiplexing*)
 - **Linee di controllo**
 - Per controllare l'accesso e l'uso delle linee dati ed indirizzi
 - Richieste ed ack, tipo di informazione sulle linee dati

Transazioni sul bus

- Transazione sul bus
 - Invio dell'indirizzo e del comando da parte dell'unità master
 - Invio o ricezione dei dati da parte dell'unità slave
- Operazione di *input* (o transazione di *scrittura*):
trasferimento dati *dal* dispositivo di I/O *alla* memoria
 - Linee di controllo: indicano che in memoria occorre eseguire una scrittura
 - Linee di dati: contengono l'indirizzo di memoria in cui scrivere il dato
- Operazione di *output* (o transazione di *lettura*):
trasferimento dati *dalla* memoria *al* dispositivo di I/O
 - Linee di controllo: indicano che in memoria occorre eseguire una lettura
 - Linee di dati: contengono l'indirizzo di memoria in cui leggere il dato

Generalità sulle organizzazioni/architetture (1/3)

- Ci sono differenti tipi di architetture, dipendenti dagli obiettivi che si sono dati i costruttori e della tipologia dei bus che si volevano utilizzare.
- Inoltre i bus possono essere proprietari o a specifica pubblica. Normalmente i bus processori/memoria sono proprietari perché essendo sincroni devono lavorare alle velocità proprie del processore e quindi i segnali di controllo, così come le linee dell'address bus e del data bus, dovendosi connettere ai pin del processore dipendono fortemente da come è realizzato il processore stesso. Il bus processore/memoria oltre al processore, come dice la parola stessa, è connesso alla memoria, pertanto molto spesso le memorie sono realizzate con le stesse tecnologie dei processori e comunque tenendo conto delle specificità dei processori che le utilizzano. Naturalmente, sia che costruite dallo stesso costruttore dei processori che da terzi, il ragionamento può essere capovolto e quindi è il processore che potrebbe essere realizzato tenendo conto delle specificità delle memorie presenti sul mercato o realizzabili secondo la tecnologia del momento.

Generalità sulle organizzazioni/architetture (2/3)

Invece i bus a specifica pubblica (denominati anche bus standard sull'Hennessy Patterson) sono stati introdotti per abbattere i costi di realizzazione dei sistemi di elaborazione.

Infatti il poter utilizzare componenti e dispositivi realizzati indipendentemente dalla tecnologia dei costruttori dei processori, consente di poter utilizzare lo stesso tipo di dispositivo in soluzioni utilizzando processori di tipo differente, realizzando così economie di scala. Infatti in tal caso i costruttori di dispositivi devono solo condividere il modo di interfacciarsi, sia a livello logico che fisico, mentre tutta la logica di funzionamento può essere realizzato utilizzando soluzioni e tecnologie completamente indipendenti da quelle utilizzate dai costruttori dei processori.

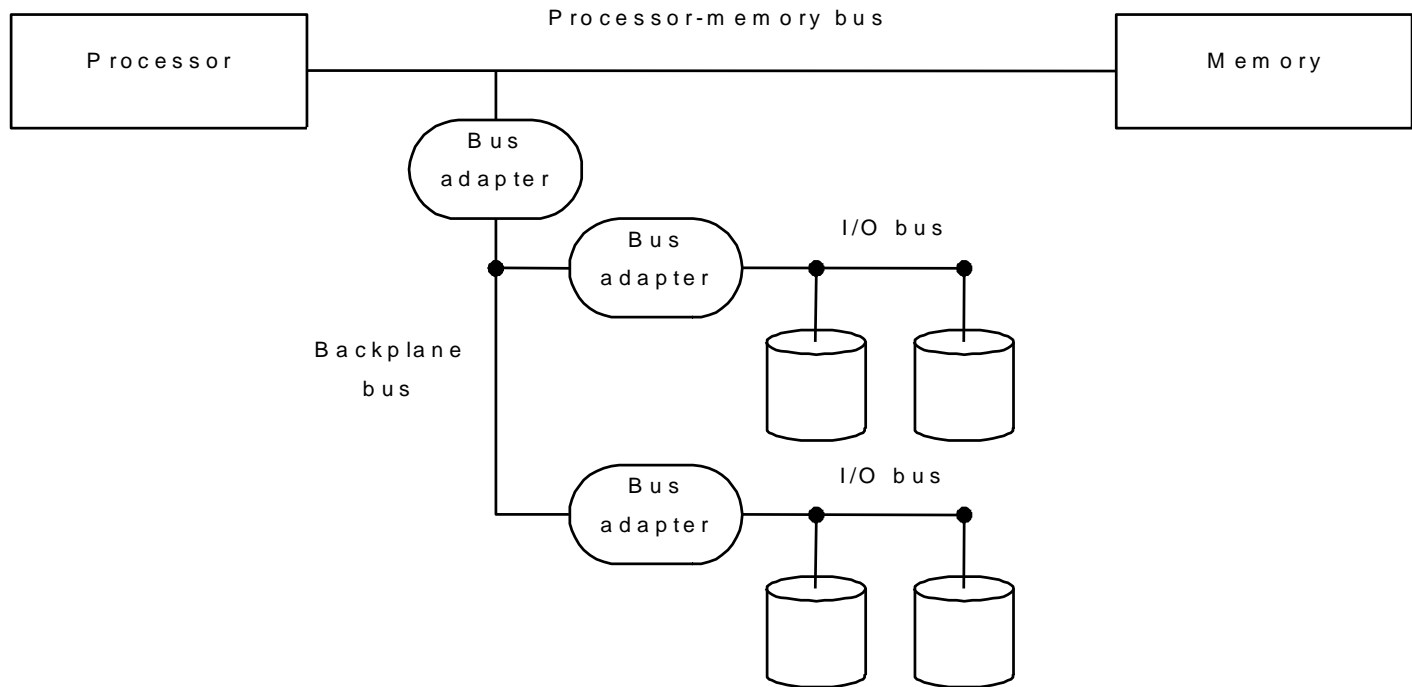
Generalità sulle organizzazioni/architetture (3/3)

- Di seguito, quindi, faremo vedere una serie di soluzioni possibili per l'interconnessione interna dei sistemi di elaborazione, ciò verrà fatto con l'obiettivo di permettere al lettore di capire i problemi di base che nel tempo hanno dovuto risolvere i progettisti. Tale presentazione verrà fatta anche facendo riferimento a come sono state proposte nel tempo alcune soluzioni da alcuni costruttori di elaboratori utilizzando processori della famiglia X86. Naturalmente la presentazione non vuole essere esaustiva, in quanto l'obiettivo di questo libro non è quello di presentare tutte le possibili soluzioni, ma quello di presentare le metodologie di progettazione dei sistemi di elaborazione tenendo conto dei costi e delle prestazioni ed in particolare di come evitare i colli di bottiglia prestazionali.
- Alcune soluzioni potranno avere solo un bus (bus generico di sistema)
- Altre un bus processore memoria con bus di I/O connessi al primo con degli adattatori
- Altre con bus generico di sistema (mother board, in questo caso), un bus processore-memoria e più bus di I/O

Tipologie di bus

- Bus processore-memoria
 - Lunghezza ridotta, alta velocità
 - In generale proprietario
 - Progettato per massimizzare la banda di trasferimento processore-memoria
- Bus di I/O
 - Tipicamente di lunghezza maggiore e più lenti
 - Una gran varietà di dispositivi di I/O connessi
 - Standard, ad es. Firewire (IEEE 1394), USB, SCSI
- Bus backplane
 - Struttura di interconnessione all'interno dello chassis
 - Usati spesso come struttura intermedia tra i bus di I/O ed il bus processore-memoria

Esempio di organizzazione



- Bus backplane connesso al bus processore-memoria
- Bus di I/O connessi al bus backplane

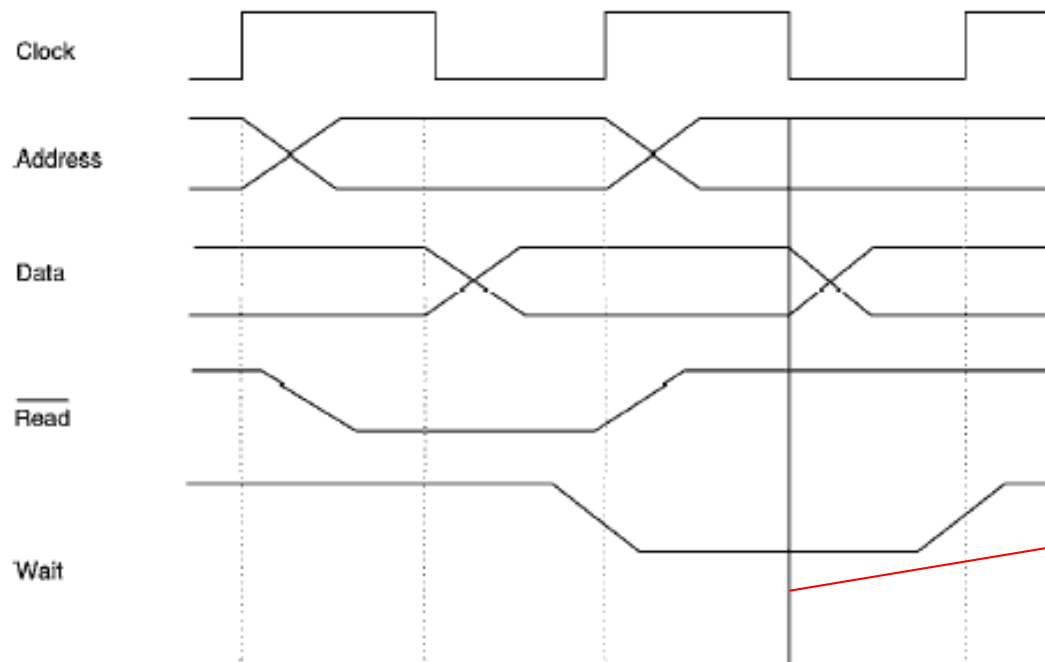
Schemi di comunicazione su un bus

- La comunicazione sul bus deve essere regolata attraverso un protocollo di comunicazione
- Esistono due schemi principali di comunicazione (temporizzazione) su di un bus
 - Bus **sincroni**: protocollo sincrono
 - Bus **asincroni**: protocollo asincrono

Bus sincrono

- Le linee di controllo del bus includono un segnale di sincronizzazione (**clock**)
- Il protocollo di comunicazione è scandito dai cicli di clock
- Ogni ciclo del bus per lettura/scrittura richiede più cicli di clock
- Vantaggi
 - Molto veloce
 - Non richiede molta logica, perché tutti gli eventi sono sincroni con il clock
- Svantaggi
 - Ogni dispositivo deve essere sincronizzato con il clock
 - Non può avere lunghezza elevata (problemi di clock skew)
- I bus processore-memoria sono spesso sincroni
 - Hanno lunghezza ridotta
 - Hanno pochi elementi connessi

Bus sincrono: transazione di lettura



I dati sono pronti
per essere letti
dal processore

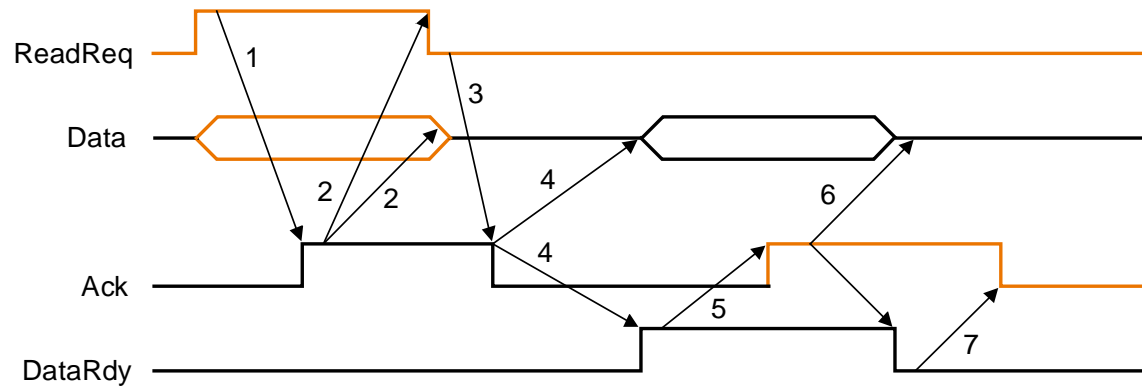
- Read: segnale di controllo che indica la richiesta di lettura (o scrittura)
- Wait: indica al processore di non aspettare
- Sono necessari più cicli di clock per leggere un dato dalla memoria

Bus asincrono

- Non è dotato di clock
- La comunicazione tra le due parti avviene tramite un ***protocollo di handshaking***
- Vantaggi:
 - Può avere lunghezza elevata e connettere molti dispositivi
 - Il tempo impiegato dalle singole operazioni sul bus è legato esclusivamente alla velocità delle parti coinvolte
- Svantaggi:
 - Più lento dei bus sincroni
- Spesso i bus di I/O sono asincroni

Bus asincrono: ciclo di lettura

evitare di usare quello di Patterson Hennessy



Esempio: lettura di una parola dalla memoria ed invio ad un dispositivo di I/O

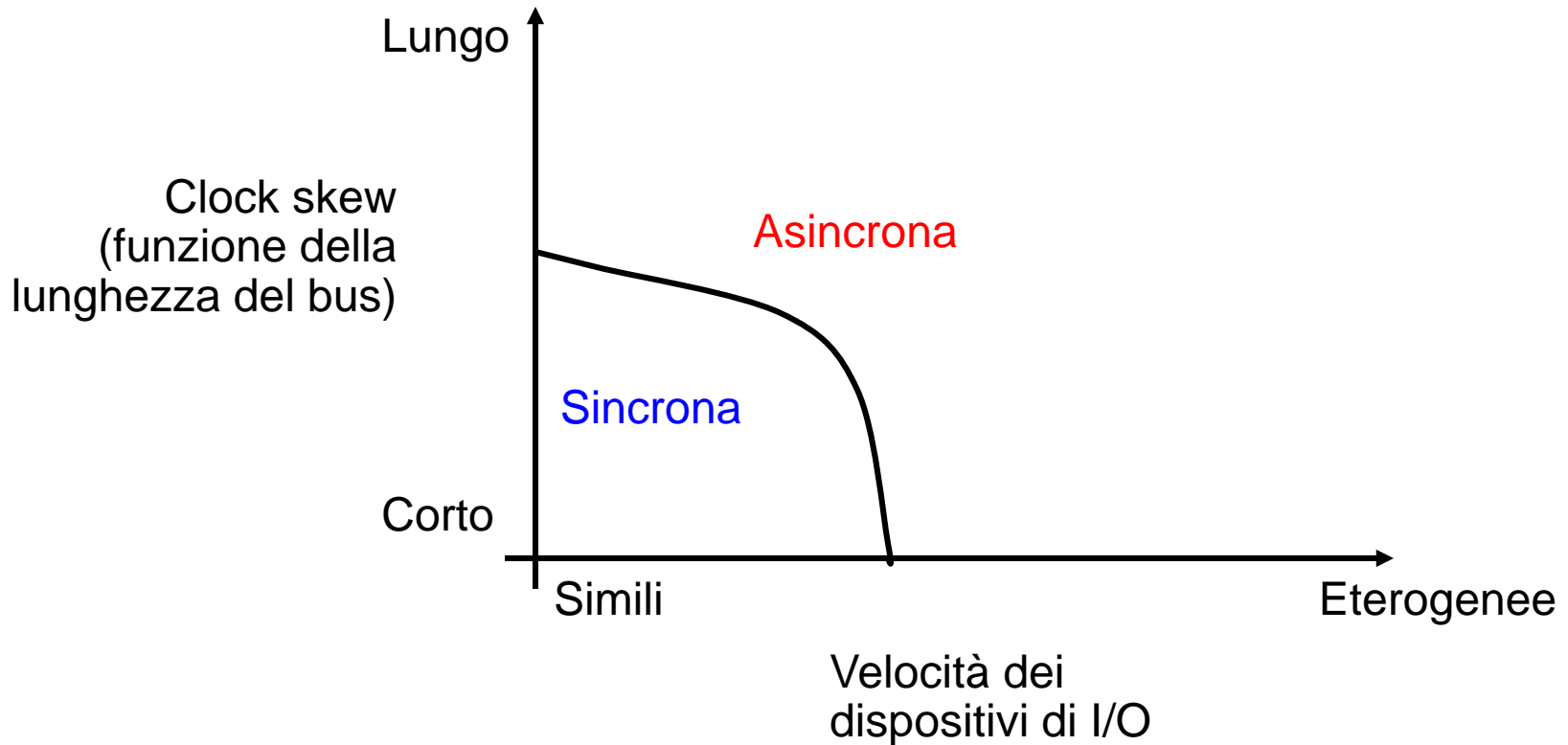
- 1) Quando la memoria vede ReadReq, legge l'indirizzo dal bus Data e asserisce Ack
- 2) Il dispositivo di I/O vede Ack asserito, nega ReadReq e rilascia Data
- 3) La memoria vede che ReadReq è negato e nega Ack
- 4) Quando la memoria ha il dato pronto, lo mette su Data ed asserisce DataRdy
- 5) Il dispositivo di I/O vede DataRdy asserito, legge il dato ed asserisce Ack
- 6) La memoria vede Ack asserito, nega DataRdy e rilascia Data
- 7) Il dispositivo di I/O nega Ack

Bus asincroni: protocollo di handshaking

- Lo schema asincrono visto è incentrato sulla seguente procedura:
 - ReadReq viene asserito
 - Ack viene asserito in risposta a ReadReq
 - ReadReq viene non asserito in risposta ad Ack
 - Ack viene non asserito in risposta a ReadReq

 - DataRdy viene asserito
 - Ack viene asserito in risposta a DataRdy
 - DataRdy viene non asserito in risposta ad Ack
 - Ack viene non asserito in risposta a DataRdy
- La procedura ha il nome di handshake

Temporizzazione sincrona o asincrona?



Accesso al bus

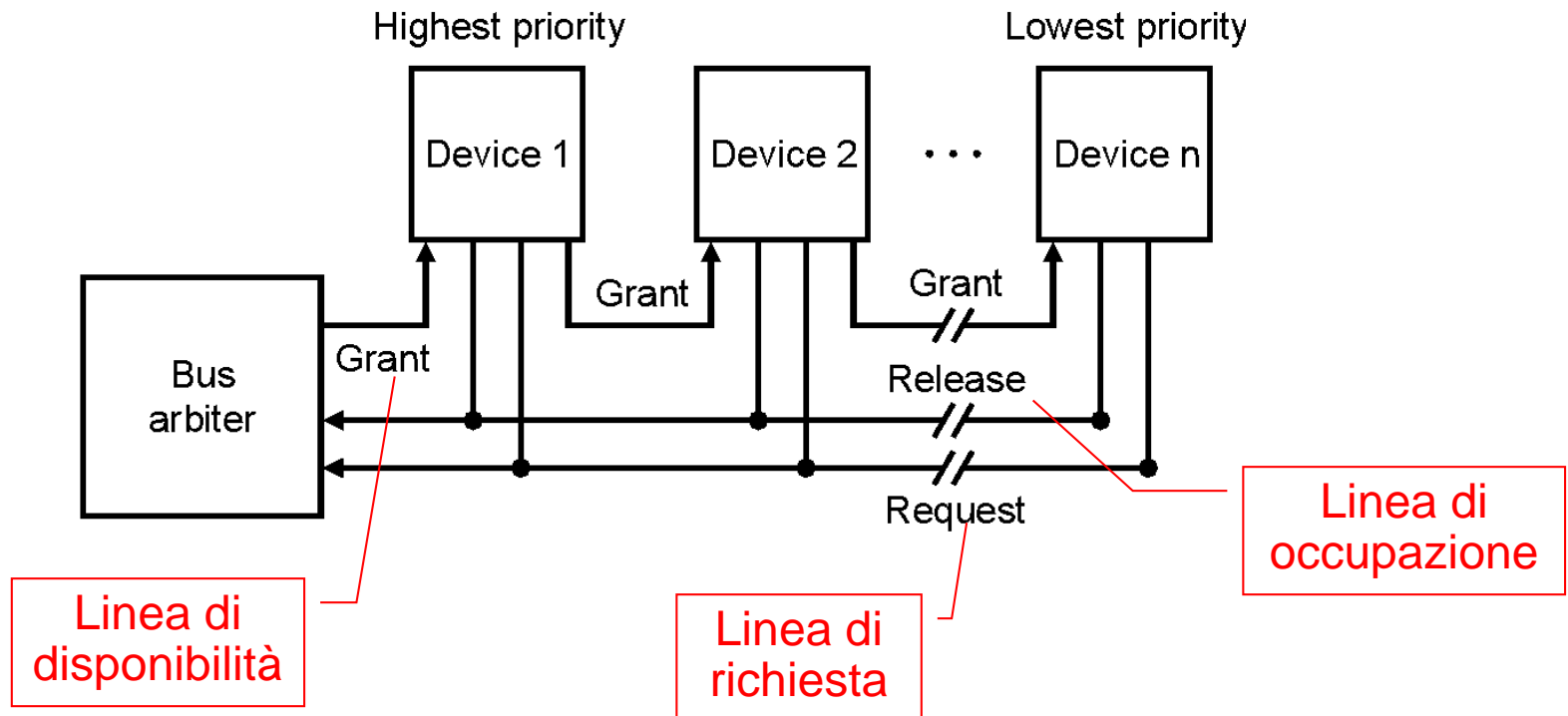
- Problema: come ottenere l'accesso al bus (mezzo di comunicazione condiviso)
- Accesso regolato tramite ruoli: master e slave
 - **Unità master**: può iniziare attivamente una transazione di lettura o scrittura
 - Il processore è sempre un master, la memoria uno slave
 - Un bus può avere molteplici master
- Architettura più semplice: un solo bus master (un processore), che media tutte le comunicazioni
 - Svantaggio: il processore deve prendere parte ad ogni transazione sul bus
- Alternativa: avere più master e seguire un protocollo per coordinare le richieste dei master
 - Occorre un meccanismo di **arbitraggio del bus**

Arbitraggio del bus

- Permette di decidere quale dispositivo sarà il prossimo bus master autorizzato all'utilizzo del bus
 - Consente di risolvere possibili contese per l'accesso
- Obiettivi:
 - Assegnare il bus ai dispositivi con priorità più alta
 - Garantire che non si verifichino situazioni di attesa indefinita o di paralisi del sistema
- Schemi di arbitraggio *centralizzati*:
 - Un controllore decide a chi assegnare il bus
 - Daisy chain e livelli multipli di priorità
- *Schemi di arbitraggio distribuiti* (decentralizzati):
 - Nessun controllore centralizzato: i dispositivi seguono un algoritmo per il controllo d'accesso e cooperano nella condivisione del bus
 - Round-robin e rilevamento della collisione

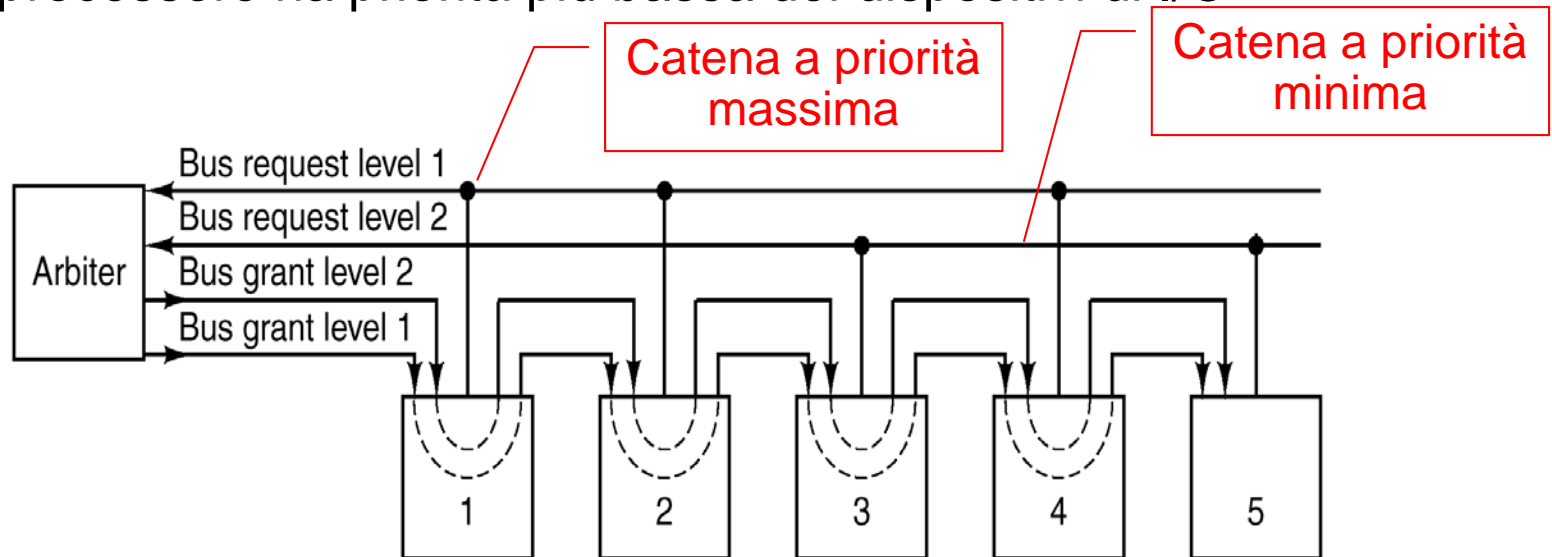
Daisy chain

- Ad ogni dispositivo è assegnata una priorità
- Sceglie il dispositivo che richiede l'accesso al bus e possiede priorità maggiore (più vicino all'arbitro)
- Problema: non garantisce la *fairness*
 - Favorisce alcuni dispositivi rispetto ad altri



Livelli multipli di priorità

- Anche detto parallelo centralizzato
- Diverse linee di richiesta associate a diversi *livelli di priorità*
 - In caso di conflitto favorite le catene a priorità più alta
 - All'interno di ciascuna catena vale la posizione (daisy chain)
 - In genere, se c'è un solo bus con anche la memoria, il processore ha priorità più bassa dei dispositivi di I/O



Schemi di arbitraggio distribuiti

- **Round-robin**
 - Disciplina circolare
 - Scambio ciclico di un segnale di disponibilità tra le unità utilizzatrici del bus
- **Rilevamento delle collisioni**
 - Esiste un'unica linea su cui è segnalato lo stato del bus (libero/occupato)
 - Più unità contemporaneamente possono occupare il bus: situazione di collisione
 - Occorre rilevare la collisione ed annullare la trasmissione
 - La trasmissione sarà ripetuta dopo un intervallo di tempo (il cui valore è generato in modo casuale)
 - Simile a rete Ethernet

Bus interni ed esterni

- I bus in un calcolatore si possono anche distinguere in bus interni ed esterni
- Bus *interni* (o locali)
 - Confinati all'interno di un singolo chip (tra processore e cache) o tra processore e memoria
 - Elevata velocità per massimizzare la banda passante
 - Tecnologia proprietaria
- Bus *esterni*
 - Collegano dispositivi diversi
 - Maggiore lunghezza
 - Velocità inferiore

Banda passante di un bus

- Un bus trasmette sequenze di dati: la rapidità con cui si passa da un dato al successivo è detta *ciclo di bus*
- Più alta è la frequenza, maggiori sono le prestazioni del bus (*bandwidth* o *banda passante*)
- Per ricavare la massima banda passante teorica:
$$\text{max banda} = \text{frequenza} * \text{numero di linee [MB/sec]}$$
- Le fasi di inattività e di scambio comandi riducono la banda passante reale
- I limiti fisici all'aumento della frequenza sono:
 - Alte frequenze creano disturbi
 - Ritardi del segnale
 - Bus skew (segnali su linee diverse che viaggiano a velocità diverse)

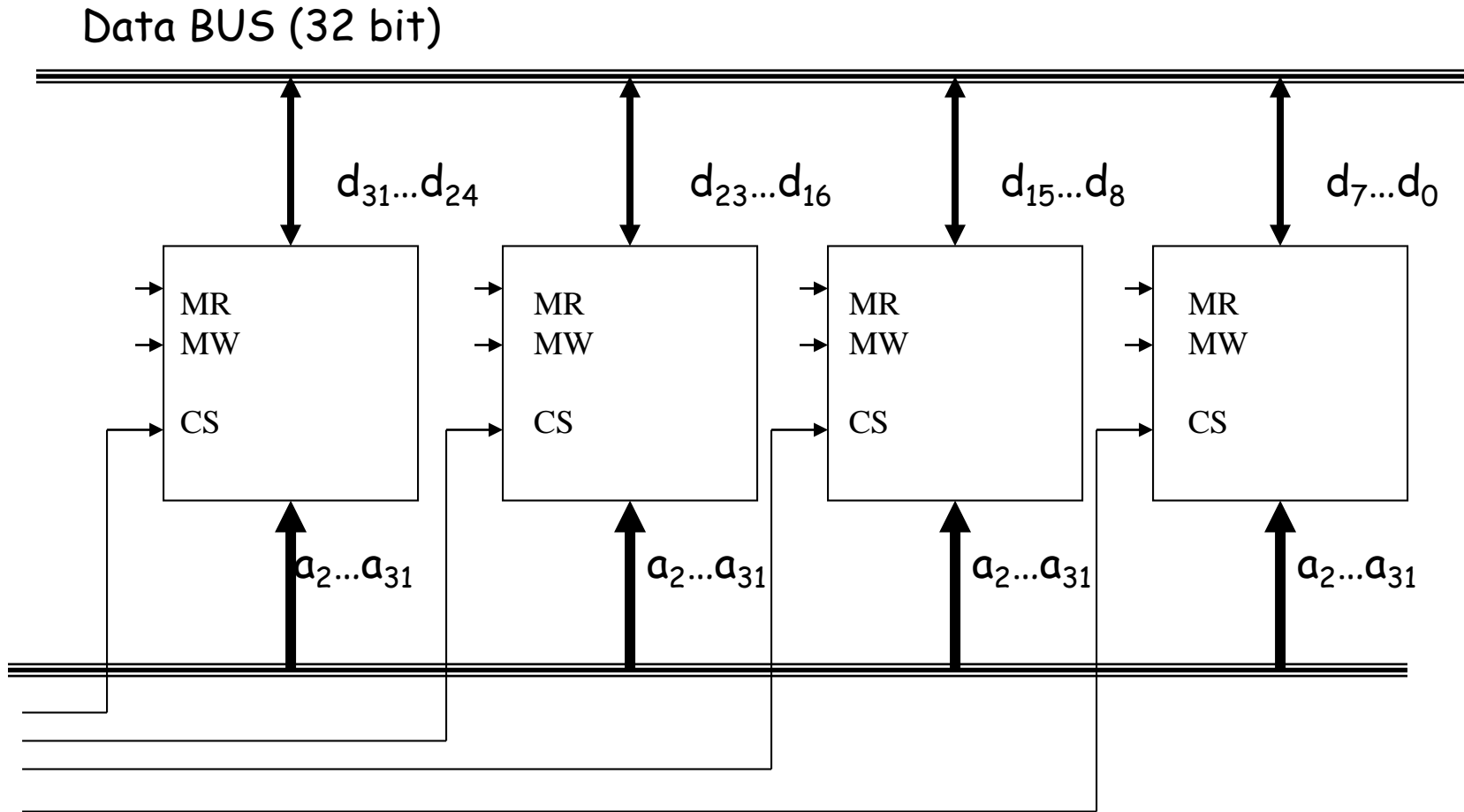
Tecniche per aumentare la banda passante

- Parallelismo delle linee dati
 - Aumento del numero di linee
- Linee dati ed indirizzi separate
 - Aumento del numero di linee
- Trasferimento di dati a blocchi
 - Riduzione del tempo di risposta
- Protocollo **split transaction**
 - La transazione sul bus è divisa in due parti: transazione di richiesta e transazione di risposta
 - Al termine della transazione di richiesta viene rilasciato il bus; per la transazione di risposta occorre nuovamente competere per l'accesso al bus
 - Vantaggio: si evitano tempi di non utilizzo del bus, sfruttando meglio la banda del bus
 - Svantaggio: tempi di transazione più lunghi
 - Usato nei sistemi multiprocessore che condividono il bus di memoria

Tecniche per aumentare la banda passante

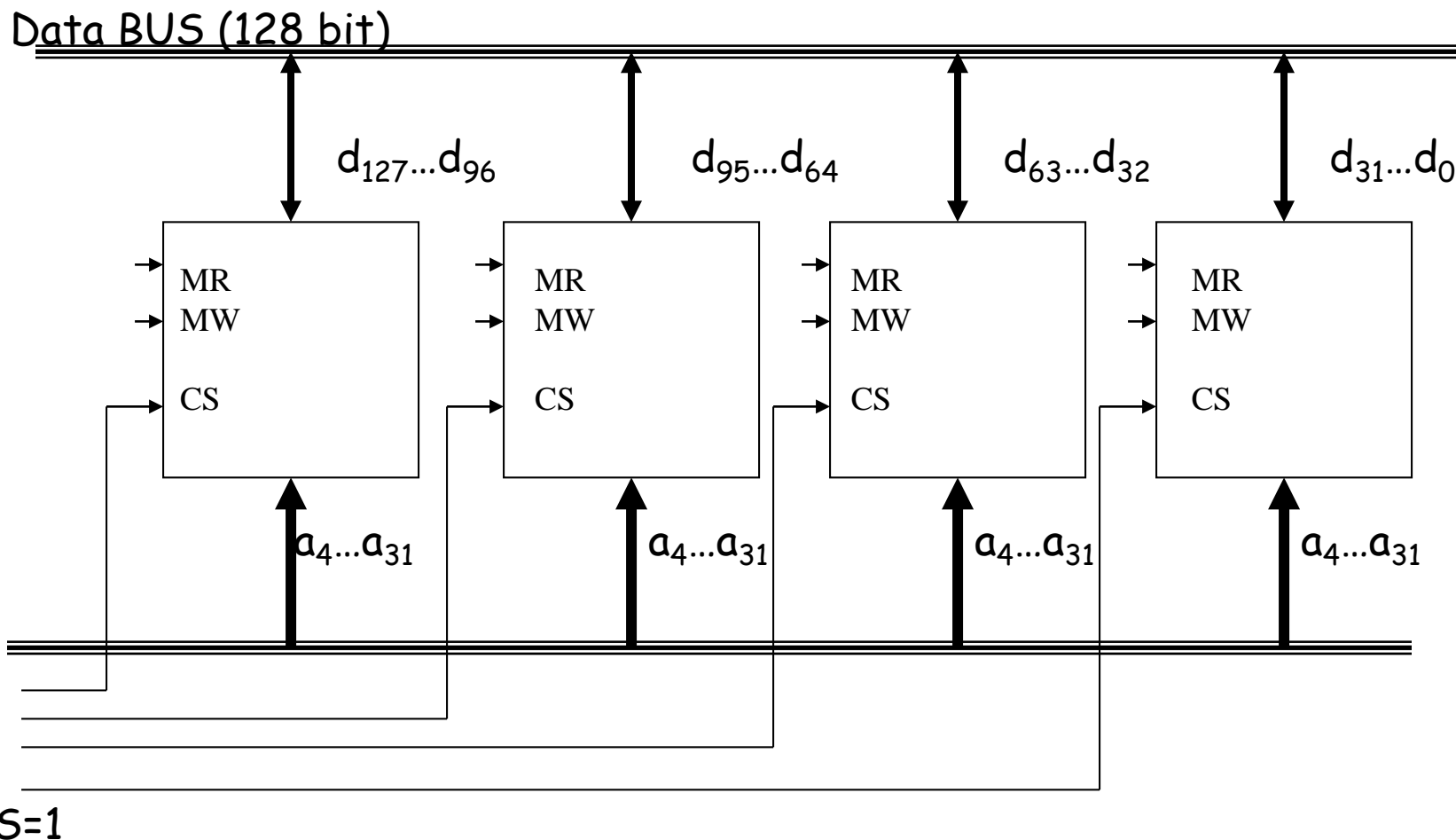
- Parallelismo delle linee dati
 - Aumento del numero di linee (passare da 32 a 64 o 128 bit)
- Linee dati ed indirizzi separate
 - Aumento del numero di linee (come nel PD32)
- Trasferimento di dati a blocchi
 - Riduzione del tempo di risposta (trasferimento di più parole contemporaneamente, identificate senza dover trasferire più indirizzi, come nel DMAC)

Parallelismo sul bus dati (p.e. da 32 a 128) (1/2)



Parallelismo sul bus dati (p.e. da 32 a 128) (2/2)

Ogni singola memoria da 32 è organizzata con 4 moduli da 8 bit
se si ipotizza allineamento dei dati $\rightarrow CS = 1$



Parallelismo sul bus dati (p.e. da 32 a 128) (2/2)

- Domanda:
come si “riorganizza” l’architettura del processore?
- Risposta:
organizzazione del processore con “prefetch”

Prendere appunti «architetture avanzate»

Opzioni di progettazione di un bus

Opzione	Prestazioni elevate	Costo basso
<i>Parallelismo bus</i>	Linee indirizzi e dati separate	Linee indirizzi e dati multiplexate
<i>Parallelismo dati</i>	Ampio (es. 64 bit)	Limitato (es. 8-16 bit)
<i>Dimensione del trasferimento</i>	Più parole per trasferimento riduce l'overhead	Trasferimento di singole parole più semplice
<i>Bus master</i>	Multiplo (necessario arbitraggio)	Singolo (nessun arbitraggio)
<i>Split transaction</i>	Sì (pacchetti request/reply separati forniscono più banda)	No (una connessione continua è più economica ed ha latenza minore)
<i>Temporizzazione</i>	Sincrona	Asincrona

Alcuni standard per bus

	IDE/Ultra ATA	SCSI	PCI	PCI-X
<i>Ampiezza dati</i>	16 bit	8 o 16 bit	32 o 64 bit	32 o 64 bit
<i>Frequenza clock</i>	Fino a 100 MhZ	10 MhZ (Fast) 20 MhZ (Ultra) 40 MhZ (Ultra2) 80 MhZ (Ultra3) 160 MhZ (Ultra4)	33 o 66 MHz	66, 100, 133 MhZ
<i>Numero di bus master</i>	1	multipli	multipli	multipli
<i>Bandwidth (picco)</i>	200 MB/sec	320 MB/sec	528 MB/sec	1064 MB/sec
<i>Temporizzazione</i>	asincrono	asincrono	sincrono	sincrono

PCI (Peripheral Component Interconnect) e PCI-X usati per connettere la memoria principale ai dispositivi periferici; IDE/Ultra ATA e SCSI (Small Component System Interface) per dispositivi di storage

Bus paralleli e seriali

- Bus *paralleli*
 - Più bit alla volta: i bit vengono inviati contemporaneamente su più linee
- Bus *seriali*
 - Un bit alla volta: i bit vengono inviati in tempi diversi su un'unica linea
 - Un bus seriale può avere un clock con frequenza superiore rispetto ad un bus parallelo
- Necessità di avere a disposizione una velocità di trasferimento dei dati sempre più elevata: maggiore attenzione verso bus seriali e collegamenti punto-punto

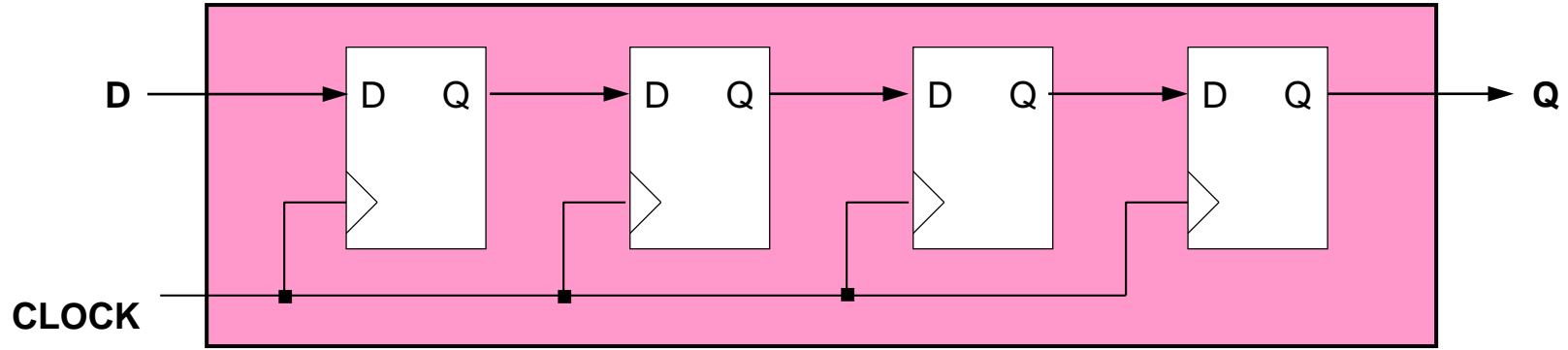
Esempi di bus ad alte prestazioni

- PCI Express
 - Evoluzione *seriale* del bus PCI (che è un bus parallelo)
 - Bus bidirezionale (full-duplex) usato principalmente da Intel
 - Costituito da un serie di canali, che possono essere aggregati per aumentare la banda
 - Banda aggregata fino a 7,5 GB/sec
 - Sostituisce il collegamento AGP con schede grafiche
- HyperTransport
 - Usato principalmente da AMD e Transmeta
 - Collegamento punto-punto unidirezionale ad alta velocità e bassa latenza
 - Ogni link è costituito da due canali (per le due direzioni di trasmissione) che operano in maniera indipendente e concorrente
 - Fino a 32 bit per link
 - Banda aggregata fino a 22,4 GB/sec

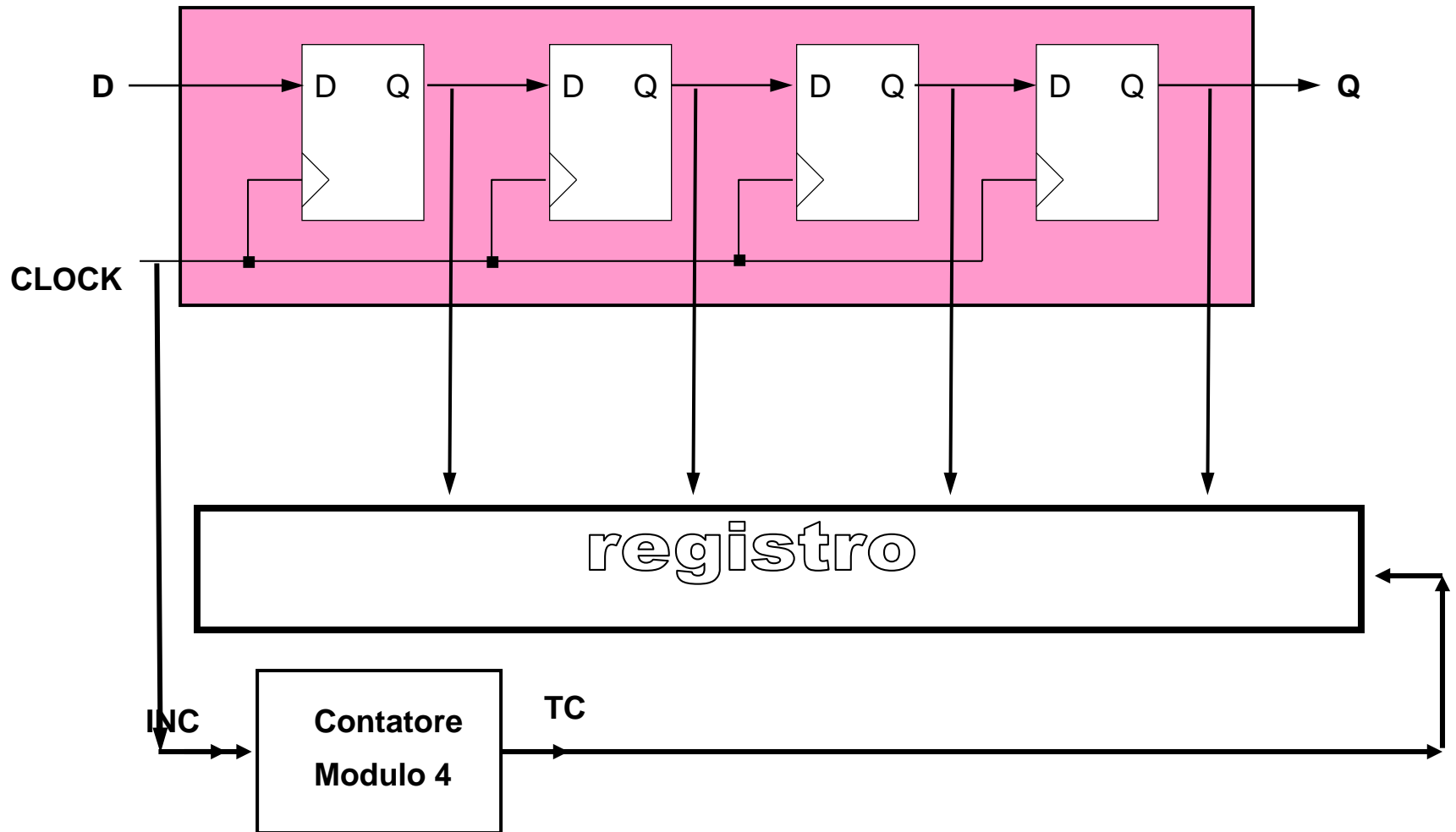
USB e Firewire

- Sono due bus di I/O seriali:
 - Permettono di collegare con un unico bus molte periferiche (fino a 63 per FireWire e 127 per USB 2.0)
 - Supportano entrambi l'inserimento a caldo
- Firewire (IEEE 1394)
 - Bus ad alta velocità progettato per dispositivi di I/O ad alta capacità (dispositivi di archiviazione e acquisizione video)
 - Fino a 50 MB/sec
 - Ideato dalla Apple
- USB (Universal Serial Bus)
 - Bus *economico* per la gestione di dispositivi di I/O a medio/bassa velocità
 - Fino a 60 MB/sec per USB 2.0
 - Flessibilità, semplicità:
 - Un unico bus per molte periferiche
 - Non sono necessari dispositivi di controllo e porte dedicate
 - Facilmente espandibile

Da seriale a parallelo (1/2)

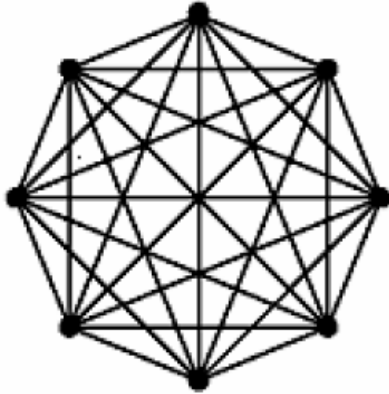


Da seriale a parallelo (2/2)

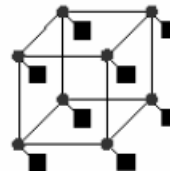
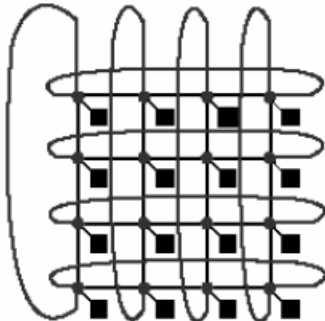


Da bus a collegamento punto-punto (link) (1/2)

(non nel programma)

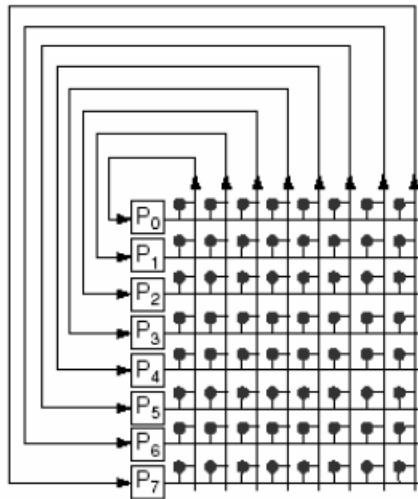


- Struttura ideale
 - *non scalabile*

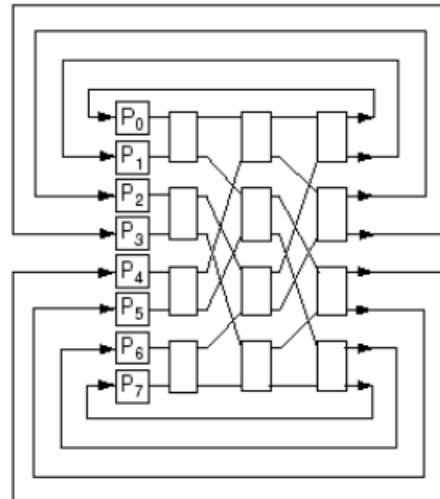


- Strutture reali (dirette)
 - *Ring*
 - *Mesh*
 - *Ipercube*

Da bus a collegamento punto-punto (link) (2/2)



a. Crossbar

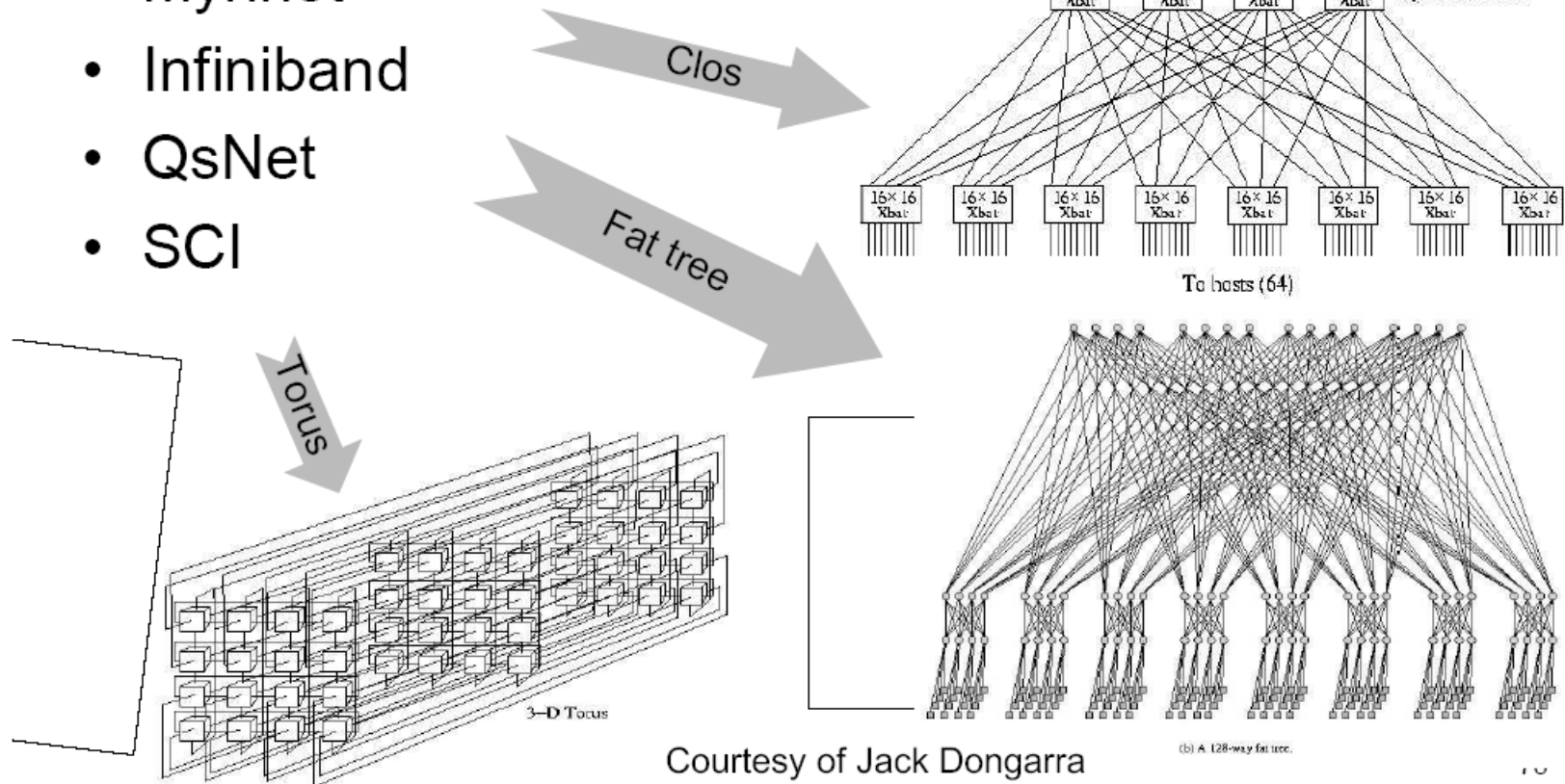


b. Omega network

- Strutture reali (indirette)
 - *Crossbar*
 - *Omega*

Commodity Interconnects

- Gig Ethernet
- Myrinet
- Infiniband
- QsNet
- SCI



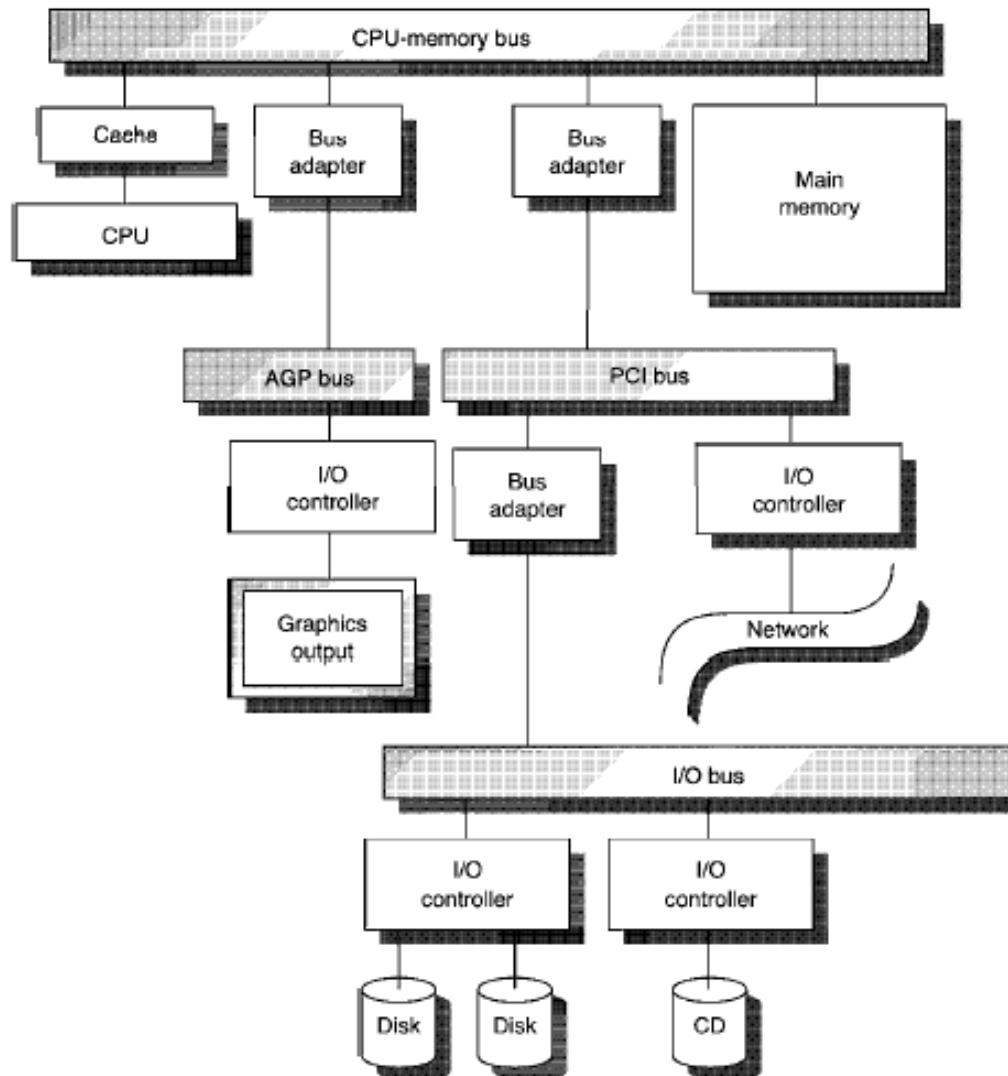
Esempi di bus ad alte prestazioni

- PCI Express
 - Evoluzione *seriale* del bus PCI (che è un bus parallelo)
 - Bus bidirezionale (full-duplex) usato principalmente da Intel
 - Costituito da un serie di canali, che possono essere aggregati per aumentare la banda
 - Banda aggregata fino a 7,5 GB/sec
- HyperTransport
 - Usato principalmente da AMD e Transmeta
 - Collegamento punto-punto unidirezionale ad alta velocità e bassa latenza
 - Ogni link è costituito da due canali (per le due direzioni di trasmissione) che operano in maniera indipendente e concorrente
 - Fino a 32 bit per link
 - Banda aggregata fino a 22,4 GB/sec

USB e Firewire

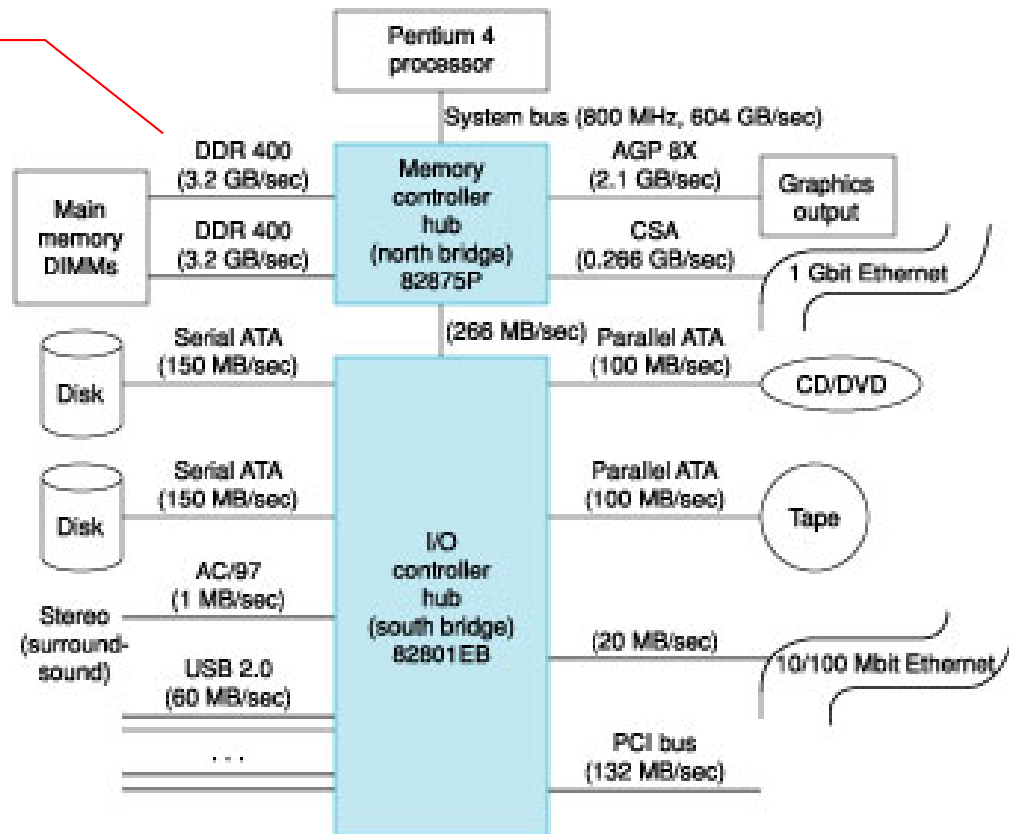
- Sono due bus di I/O seriali:
 - Permettono di collegare con un unico bus molte periferiche (fino a 63 per FireWire e 127 per USB 2.0)
 - Supportano entrambi *l'inserimento a caldo*
- Firewire (IEEE 1394)
 - Bus ad alta velocità progettato per dispositivi di I/O ad alta capacità (dispositivi di archiviazione e acquisizione video)
 - Fino a 50 MB/sec
 - Ideato dalla Apple
- USB (Universal Serial Bus)
 - Bus *economico* per la gestione di dispositivi di I/O a medio/bassa velocità
 - Fino a 60 MB/sec per USB 2.0
 - Flessibilità, semplicità:
 - Un unico bus per molte periferiche
 - Non sono necessari dispositivi di controllo e porte dedicate
 - Facilmente espandibile

Esempio tipico per desktop



Esempio: Pentium 4

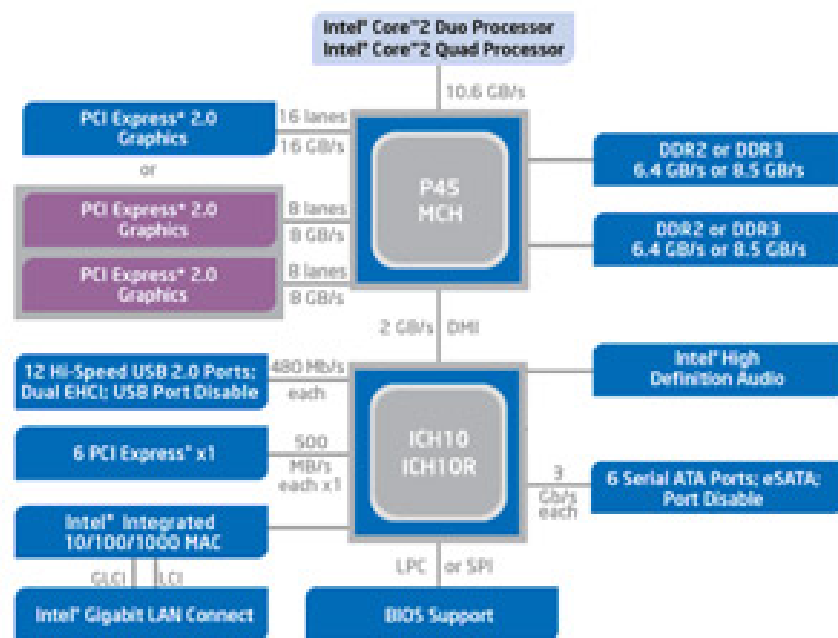
DDR (double-data rate): invio dei dati sia sul fronte di salita che sul fronte di discesa del clock



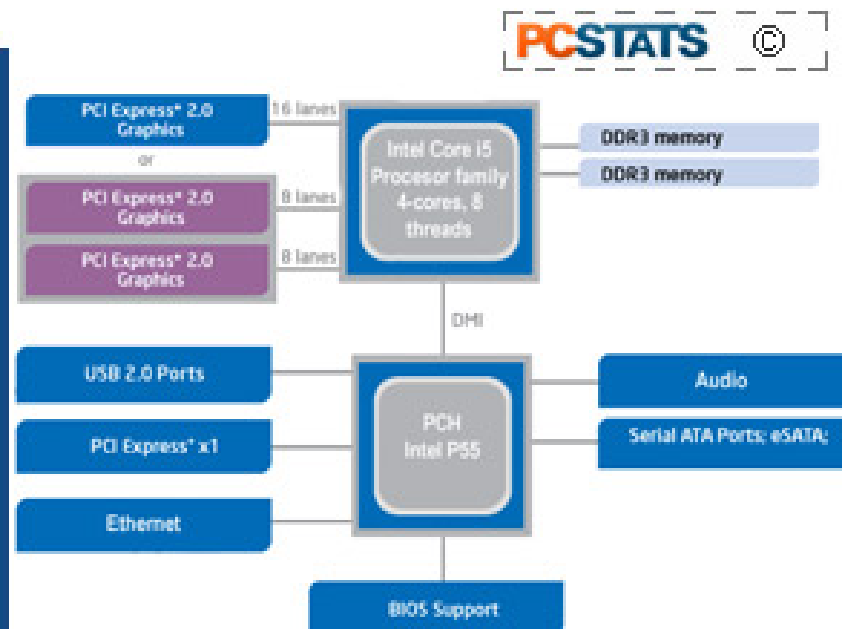
Il tasso di trasferimento tra north bridge e south bridge è 266 MB/sec: per questo AGP (Accelerated Graphics Port) e la Gigabit Ethernet sono connesse al north bridge anziché al south bridge

AGP non è un vero e proprio bus ma un collegamento punto-punto

Altri processori Intel con e senza north bridge

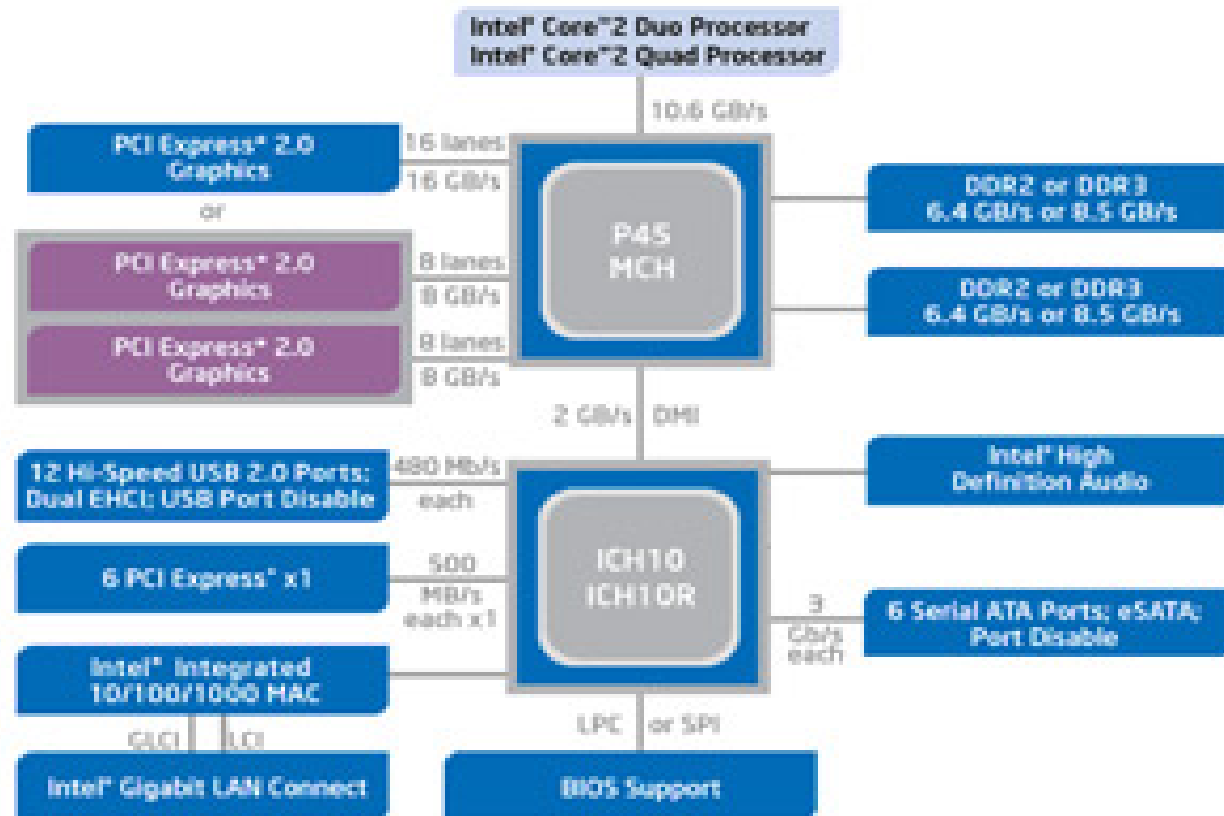


3-Chip Architecture
(old)

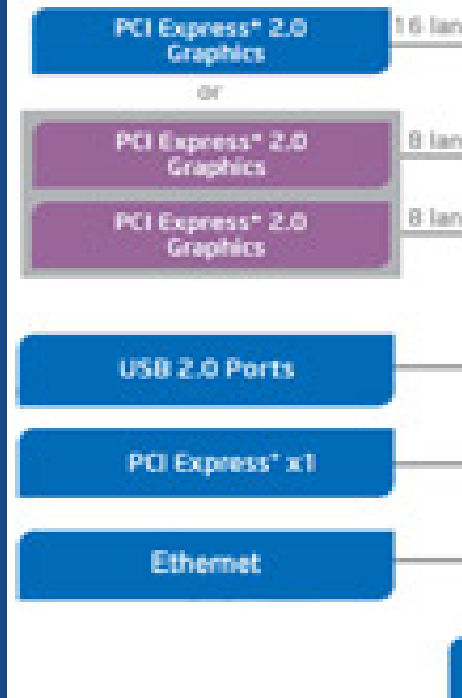


2-Chip Architecture
(new)

«old» 3 chip architecture

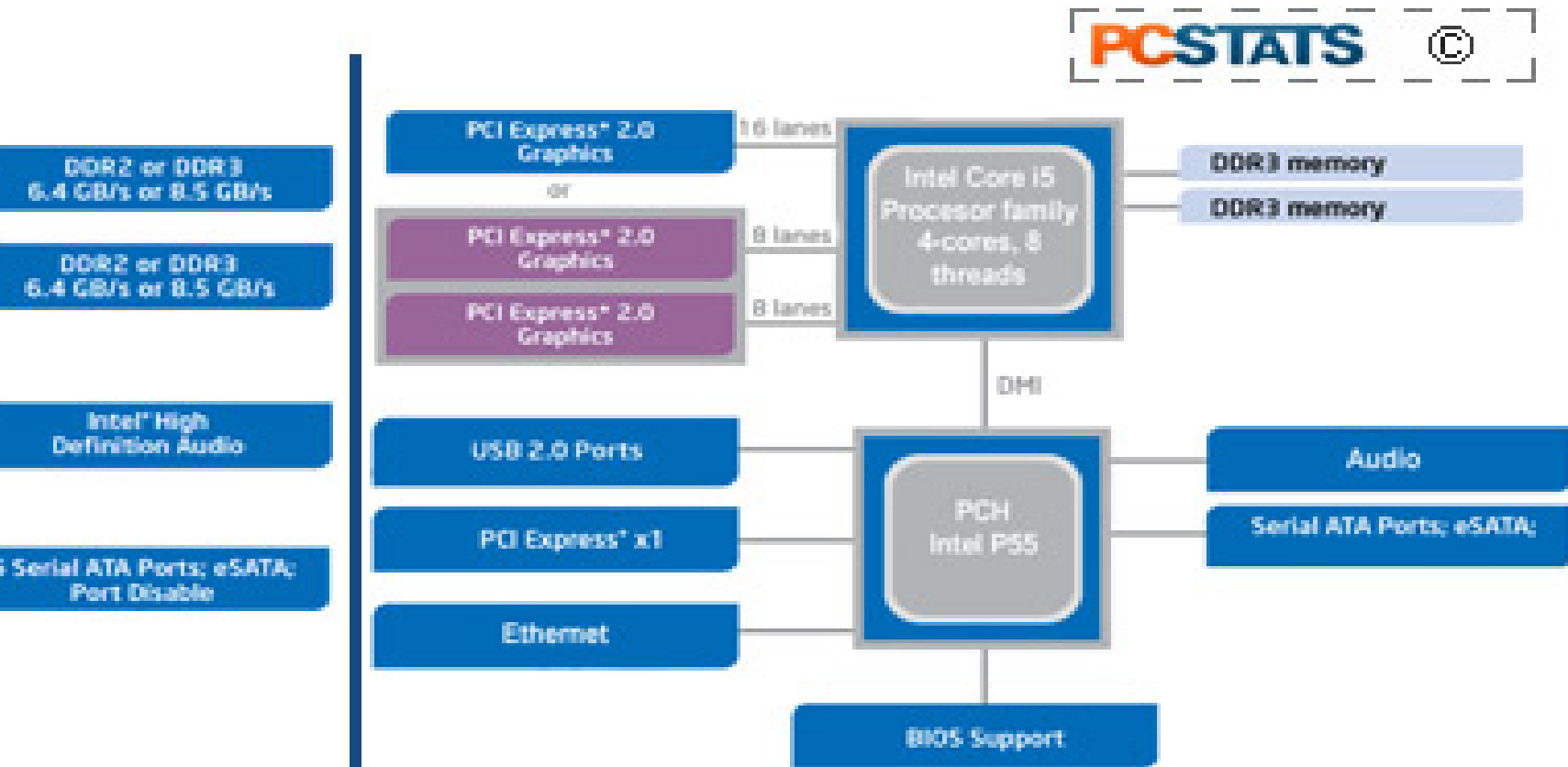


3-Chip Architecture
(old)



2-Chip Architecture

«new»: 2 chip architecture



2-Chip Architecture (new)

un po' di storia

Intel P55 Express Chipset

- Up until recently desktop computer motherboards have been built around a three chip solution that hasn't really changed in the past decade. It consisted of a CPU that communicated directly with the Memory [Controller Hub](#) (MCH) also known as the 'Northbridge,' and this in turn talks to an I/O Controller Hub (ICH), commonly called the 'Southbridge.'
- The *CPU-Northbridge-Southbridge* architecture happily survived many years of upgrading intact, but as memory started using more channels, CPUs grew extra cores and videocards grew more powerful, the interconnects that let these parts send information to one another became increasingly swamped with information and prone to bottlenecks. It was time to redesign the system architecture.
- What was once a three chip system has been simplified to just two parts with Intel's 'Lynnfield' [P55](#) platform: *CPU-Southbridge*.
- In the past couple years Northbridge chipsets for the Intel platform have included the 945P, 965P, 975X, P35, P45 and so on. Common to each generation was a discrete silicon chip, and of course, each time some aspect of the technology advanced an new generation of chipset and motherboard emerged. This is partly because it's good business to make existing technologies obsolete, and because a Northbridge chipset acts as bridge between the CPU, system memory, graphics card and the rest of the PC. Without it the PC cannot function, and because of their heavy reliance on integration, core logic chipsets cannot be upgraded like processors.
- The third member of the family (the first being the CPU) is called the Southbridge. It's job is to bridge the CPU-Northbridge with the rest of the computer; input/output devices like the hard drive, PCI/PCI Express expansion slots, sound card, network controller and even mouse/keyboard. On the Intel platform these have included the ICH7, ICH8, ICH9 and ICH10 Southbridge chipsets.
- Here's a block diagram to illustrate the relationship between these three critical chips in a simplified computer system, the way it has been and the way it has become with the Intel P55 platform. The main difference between the two is that tasks previously handed by the Northbridge chipset have been rolled into the Core i5 processor.

un po' di storia

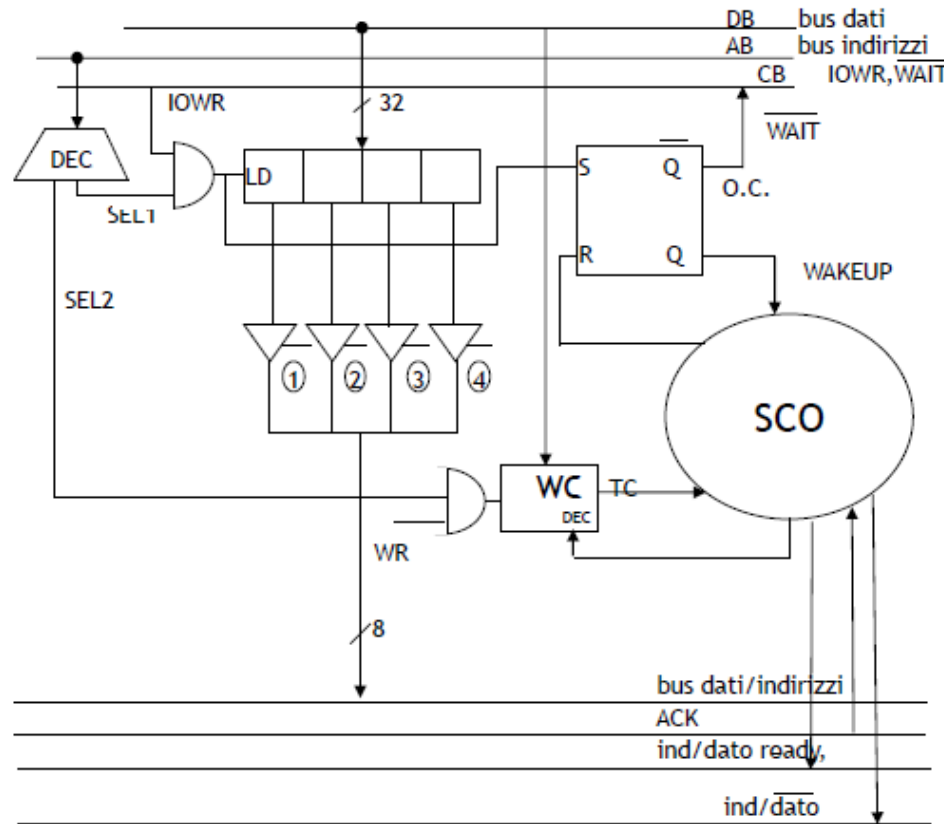
The Gigabyte Smart 6

- Gigabyte has integrated a suite of software and hardware technologies called the Smart6.
- For quite some time, Gigabyte has placed two Basic Input Output System (BIOS) chips on its motherboards. The secondary BIOS chip works as a physical backup should the primary BIOS chip fail. With Smart DualBIOS, Gigabyte has doubled the [memory size](#) from 8MB up to 16MB, and the extra memory means that these BIOS chips can now do some new tricks. Smart DualBios can save critical data such as passwords into memory, or even work as a scheduler for important dates and events.
- Smart QuickBoost is a built-in system tuning feature can adjust [CPU speed](#) on the fly, for dynamic processor overclocking inside of windows.
- Smart QuickBoost has three levels of overclocking that will adjust processor speeds up to 400MHz (3x133MHz) above the CPU's stock speed. This feature works well when paired with Smart QuickBoot, which has two modes. The first bypasses BIOS hardware checks that can take a few seconds during start-up, while the second enables a hybrid S3 (suspend) and S4 (hibernate) state, which allows the system to store its last used state, which allows the system to power on and resume exactly where it left off last time.
- [Gigabyte's](#) Smart Recovery utility allows for system backup by creating snapshots of the hard drive periodically, so it's possible to restore deleted or altered files and settings from several different dates in the past. Smart Recorder is a system logging tool will monitor when your computer has been turned on or off, as well as keeping a log of when large amounts of data are transferred between hard drives. Finally, Scheduling using Gigabyte's Smart Timelock feature can lock down the system during set hours on certain days of the week, for both system security purposes and parental control.

Progetto di un adattatore di bus

Gli adattatori di bus devono consentire la trasmissione dei dati da un bus ad un altro, Come visto negli esempi precedenti ogni bus usa un proprio protocollo di comunicazione (sincrono od asincrono), le funzioni di un adattatore sono quindi simili a quelle di un interprete umano che per consentire a due persone con linguaggio differente di poter comunicare deve conoscere almeno entrambe le lingue. Quindi gli adattatori devono essere progettati in modo da poter prelevare informazioni utilizzando un protocollo da un bus e di inviarle con un altro protocollo sull'altro bus. Di seguito faremo vedere come progettare un adattatore che deve prelevare informazioni con un protocollo sincrono e trasmetterle con un protocollo asincrono. Per semplicità di presentazione si utilizzeranno protocolli già introdotti precedentemente, ma la progettazione che si presenta può essere adattata a qualunque protocollo. Nel caso specifico si progetterà un adattatore in grado di trasmettere informazioni alla velocità più lenta tra i due bus (normalmente quello asincrono). Nell'ipotesi, invece, che il protocollo sincrono, come avviene normalmente, fosse il bus con maggiore banda passante e non si volesse rallentare il produttore delle informazioni, sarebbe necessario modificare l'adattatore che dovrebbe essere in grado di acquisire informazioni alla velocità di produzione del produttore, filtrandogli la lentezza del consumatore finale. In questo caso si potrebbe utilizzare una memoria tampone in cui prima si memorizzano le informazioni e poi le si ritrasmettono al consumatore. Questa è una modalità tipica dei sistemi di comunicazione tra i calcolatori, utilizzata dai **router**, e denominata **store and forward**.

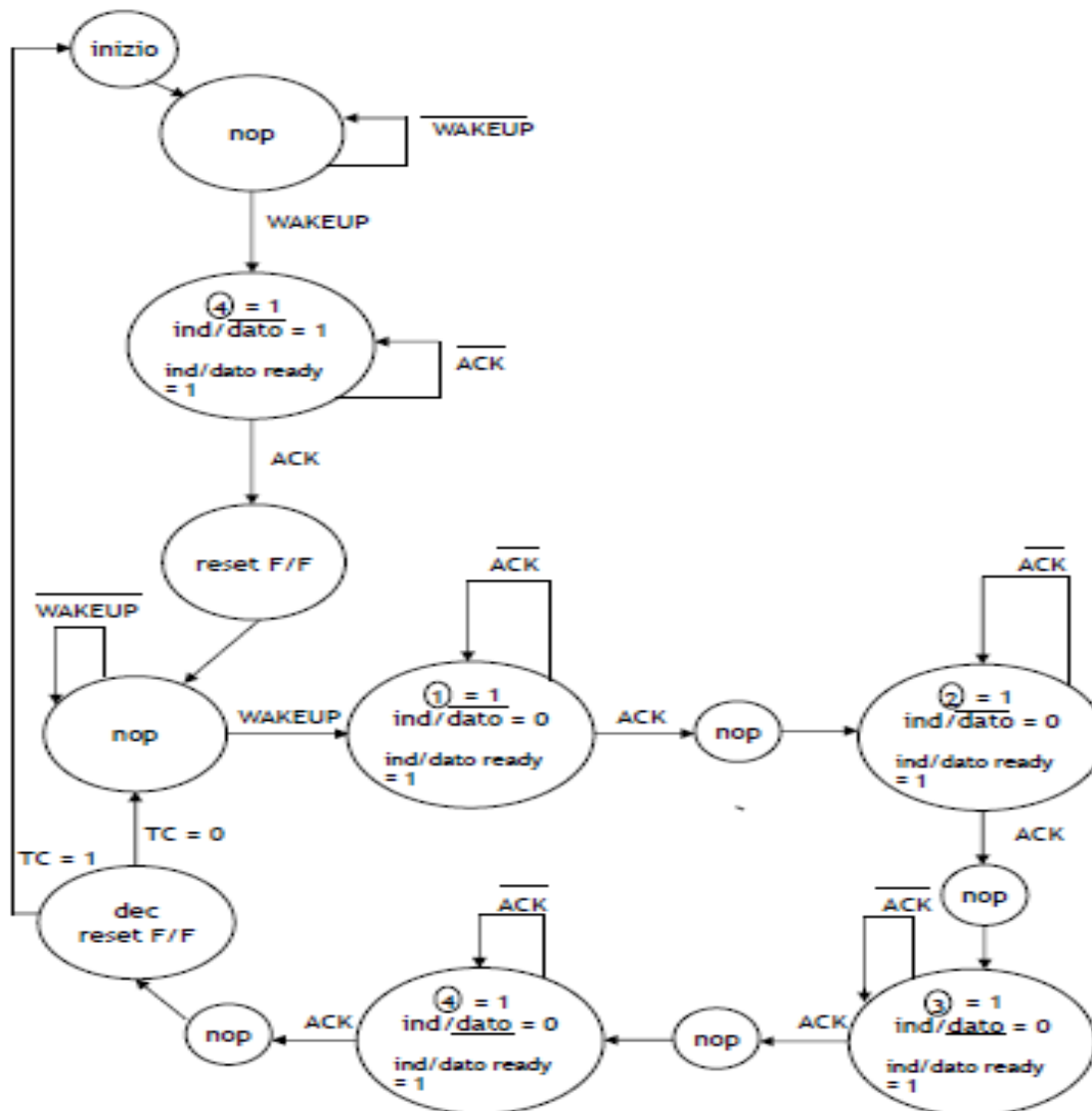
Progetto di un adattatore di bus (tra un bus sincrono ed un bus asincrono)



-
- Nei primi 32 bit del messaggio da spedire vengono indicate il numero di “parole” contenute nello stesso, e questa quantità viene salvata nel WC. Nei secondi 32 bit viene indicato un numero identificativo di una periferica connessa al bus asincrono. Di questi bit vengono selezionati gli 8 meno significativi (abilitando il segnale “4”) e spediti al bus. Da questo momento sino all'asserirsi di TC inizia il trasferimento controllato da un F/F che funge da semaforo, in quanto viene settato ad 1 non appena viene scritto un dato sul registro d'interfaccia, e ciò genera i segnali di WAIT e WAKEUP i quali rispettivamente mantengono in attesa il processore e “svegliano” lo SCO dell'adattatore. Appena viene consumato il dato lo SCO resetta il F/F, decrementa il WC e si mette in attesa di un nuovo dato dalla CPU. I 4 segnali di controllo “1”, “2”, “3” e “4” e i relativi buffer three state fungono da multiplexer e selezionano un blocco di 8 bit dal registro di interfaccia (32 bit) per poter inviare il dato sul bus asincrono (8 bit). Sono necessari 2 segnali di selezione SEL1 e SEL2 in quanto ne occorre uno per il registro da 32 bit ed uno per il contatore.

Macchina a stati finiti

Macchina a stati finiti SCO



Progetto dello z64 con «north bridge»

Progetto dello z64 con north bridge incluso nel chip
