

Corso di Sistemi Automatici di Misura

SETTIMA LEZIONE:

II VXI BUS

Introduzione al VXI Bus

Oggi che i sistemi di misura automatici costituiscono una realtà consolidata, l'interesse va spostandosi sempre più verso la qualità dell'automazione

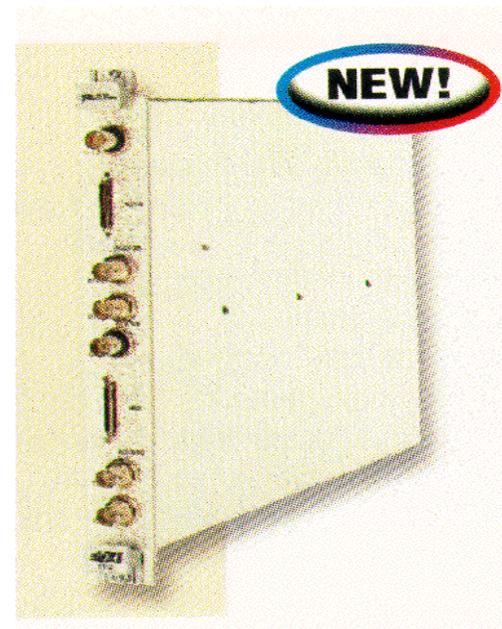
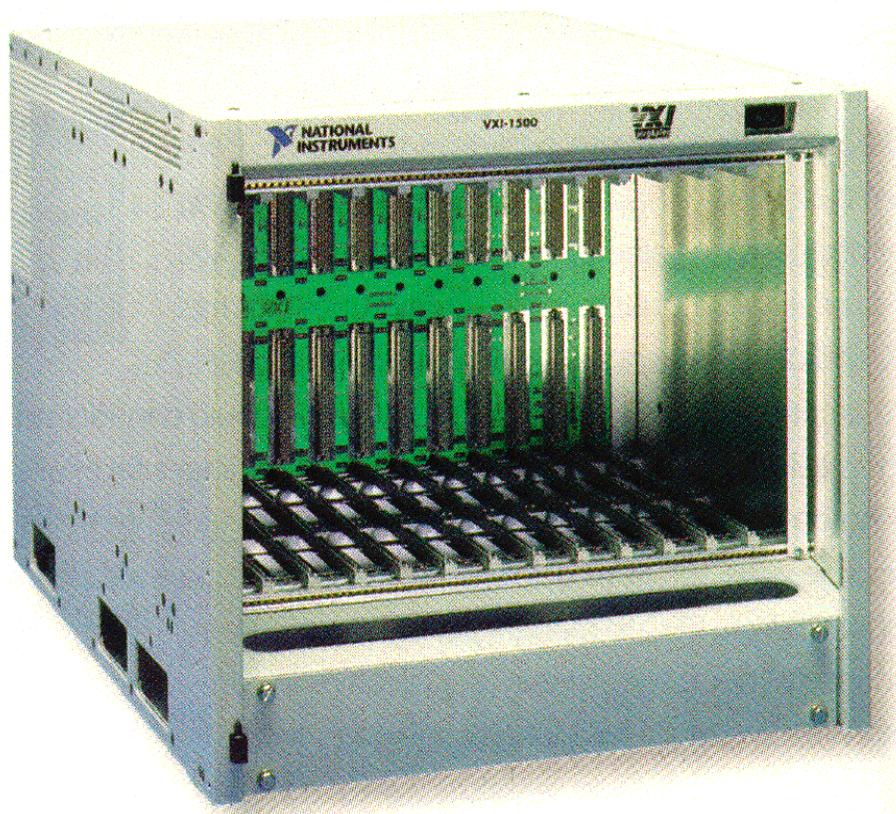
Oltre alle caratteristiche metrologiche, la “qualità” di un sistema automatico di misura è senz'altro legata al mezzo e ai protocolli di comunicazione tra dispositivi

La norma IEEE-1155 per strumenti su scheda definisce un bus di comunicazione, dei protocolli ed una classe di strumenti dedicati alla realizzazione di sistemi di misura ad elevato livello di automazione

Le risorse offerte dal bus VXI mirano a superare quelle offerte dal GPIB (IEEE-488) principalmente per ciò che riguarda le *capacità* e le *velocità* di trasferimento dati ed il *sincronismo* tra i dispositivi

Introduzione al VXI Bus

IL SISTEMA VXI



Introduzione al VXI Bus

VXI = Vme eXtension for Instrumentation

....si parte dal **VME Bus (IEEE-1014)**

- 1) Bus parallelo asincrono (è possibile collegare tra loro schede che operano a velocità diverse)
- 2) Bus NON multiplexato
- 3) Intraboard intracabinet
- 4) Gestione Multimaster (con conseguente necessità di arbitraggio)
- 5) Consente il trasferimento diretto di dati tra due moduli qualsiasi (il Master provvede al controllo del trasferimento, lo Slave esegue le operazioni richieste dal Master)
- 6) Supporta controllori a 8, 16, e 32 bit
- 7) Elevata velocità di trasferimento dati

Introduzione al VXI Bus

BUS VME

E' un architettura basata su cestello il cui pannello posteriore (backplane) è dotato di connettori attraverso i quali dei dispositivi su scheda si allacciano al bus (VME)

Le specifiche VME prevedono un massimo di 21 schede all'interno di un cestello

Nessuna specifica è data circa la possibilità di “estendere” il cestello e circa la possibilità di mettere in comunicazione più cestelli

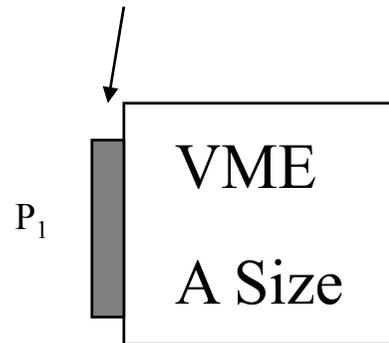
Nessuna specifica di compatibilità elettromagnetica e quelle riguardanti il raffreddamento e la dissipazione di potenza vengono definite nel VME

Si possono avere schede di due taglie differenti: SIZE A e SIZE B

Introduzione al VXI Bus

BUS VME

3 colonne
da 32 linee



VME size A

Connettore DIN P1 definito

10cm * 16cm

Bus dati a 16 bit

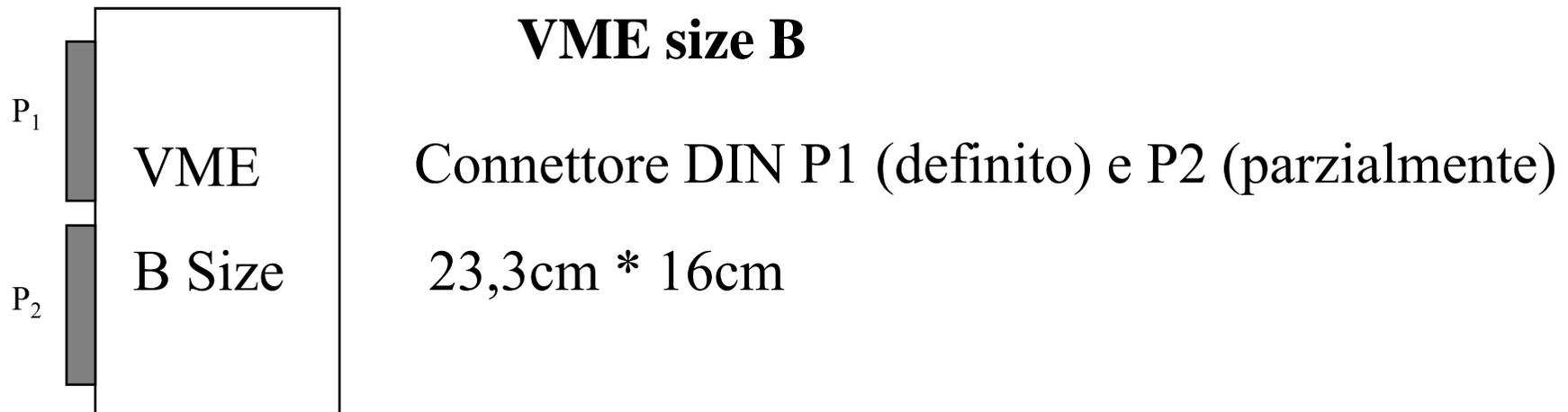
Indirizzamento fino a 16MByte di memoria (32 linee di indirizzamento)

Bus di arbitraggio

Bus utilità generali

Introduzione al VXI Bus

BUS VME



Sono assegnati solo i Pin della colonna centrale del connettore P2

Altre 16 linee dati

Indirizzamento fino a 4GByte di memoria (32+16 linee di indirizzamento)

Introduzione al VXI Bus

BUS VME

Le linee del BUS VME sono suddivise in 4 gruppi logici (BUS):

DTB (Data Transfer Bus): linee da utilizzare per il trasferimento dati tra moduli; comprendono linee di indirizzo, dati e sincronismo. Ogni ciclo di trasmissione sul DTB comporta, generalmente, il trasferimento di una word (due byte) tra due moduli comunicanti

ARBITRATION BUS: linee utilizzate dai moduli *master* che hanno necessità di operare sul DTB e dal modulo che esegue la funzione di *arbitro*

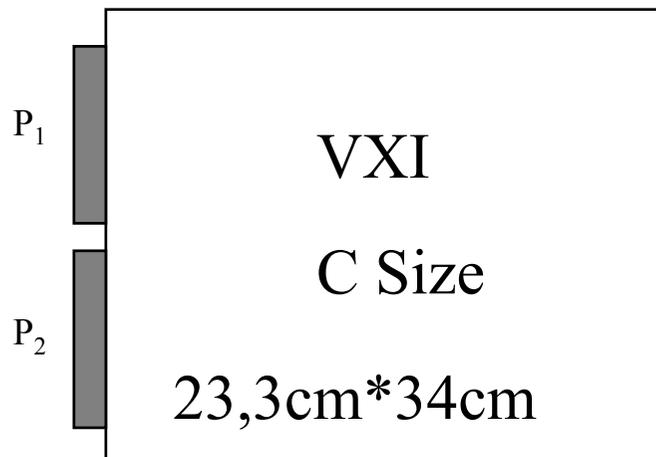
INTERRUPT BUS: linee utilizzate dai moduli (*interrupter*) che richiedono l'interruzione delle normali attività del DTB e dai moduli preposti a gestire le interruzioni (interrupt handler)

UTILITY BUS: linee di utilità generali quali, clock, reset di sistema, ecc.

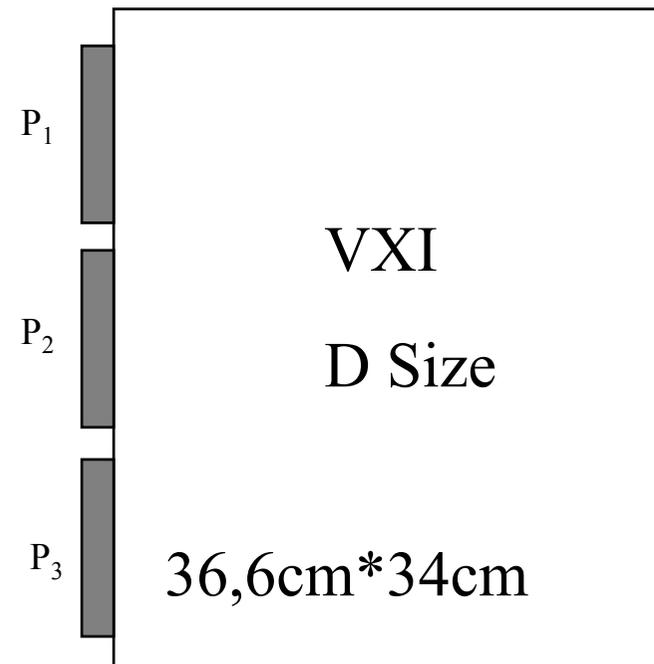
Introduzione al VXI Bus

BUS VXI

Sono state conservate le taglie **A** e **B** e in più, nell'ottica di avere strumentazione con potenzialità superiori sono state introdotte due ulteriori taglie: **C** e **D**



Connettori DIN P1 e P2
(entrambi definiti)



Connettori DIN P1, P2
e P3 (tutti definiti)

Introduzione al VXI Bus

BUS VXI

Contrariamente al VME, per il sistema VXI sono state definite diverse specifiche riguardanti:

- 1) Raffreddamento
- 2) Alimentazione
- 3) Suscettibilità ai disturbi condotti
- 4) Suscettibilità ai disturbi radiati
- 5) Schermatura
- 6) Messa a terra

Introduzione al VXI Bus

IL SISTEMA VXI

Può essere costituito da uno o più sottosistemi, per un ammontare massimo di 256 dispositivi (moduli)

Tali dispositivi sono alloggiati all'interno di un cestello (da 5 a 13 slot)

Il cestello deve necessariamente includere un modulo temporizzatore ("Slot 0") nello slot 0. Questo genera il clock di sistema ed i segnali di configurazione e di sincronizzazione. Mediante un opportuno software (*resource manager*) identifica i dispositivi del sistema, assegna gli indirizzi logici, configura la memoria e stabilisce le gerarchie

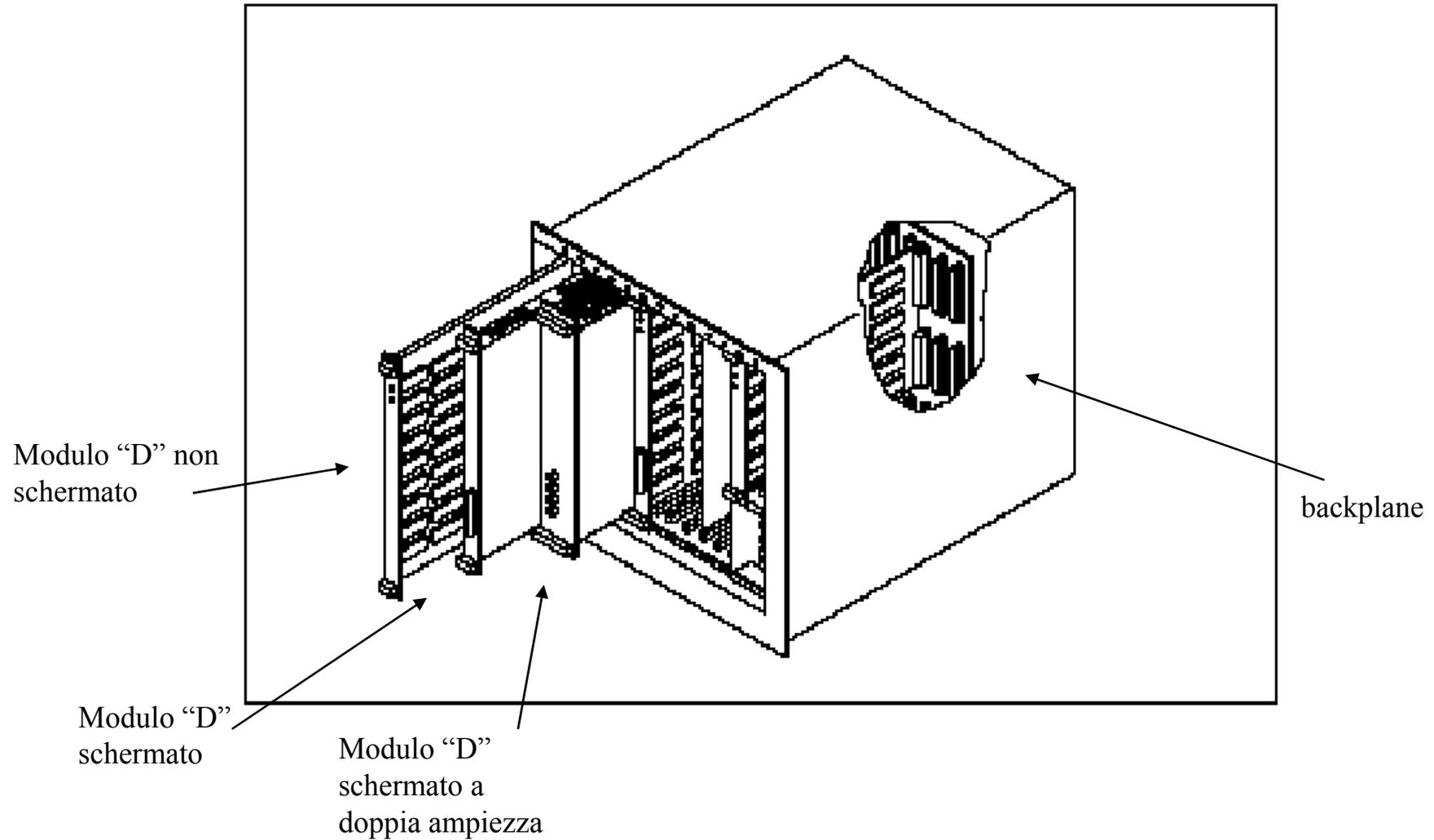
Ciascun modulo può occupare più di uno slot

Ad ogni dispositivo è associato un indirizzo logico (da 1 a 255)

Due o più sottosistemi comunicano tra di loro, per formare un sistema VXI per mezzo di un'interfaccia verso un bus intermedio (IEEE-488, RS-232, MXIbus)

Introduzione al VXI Bus

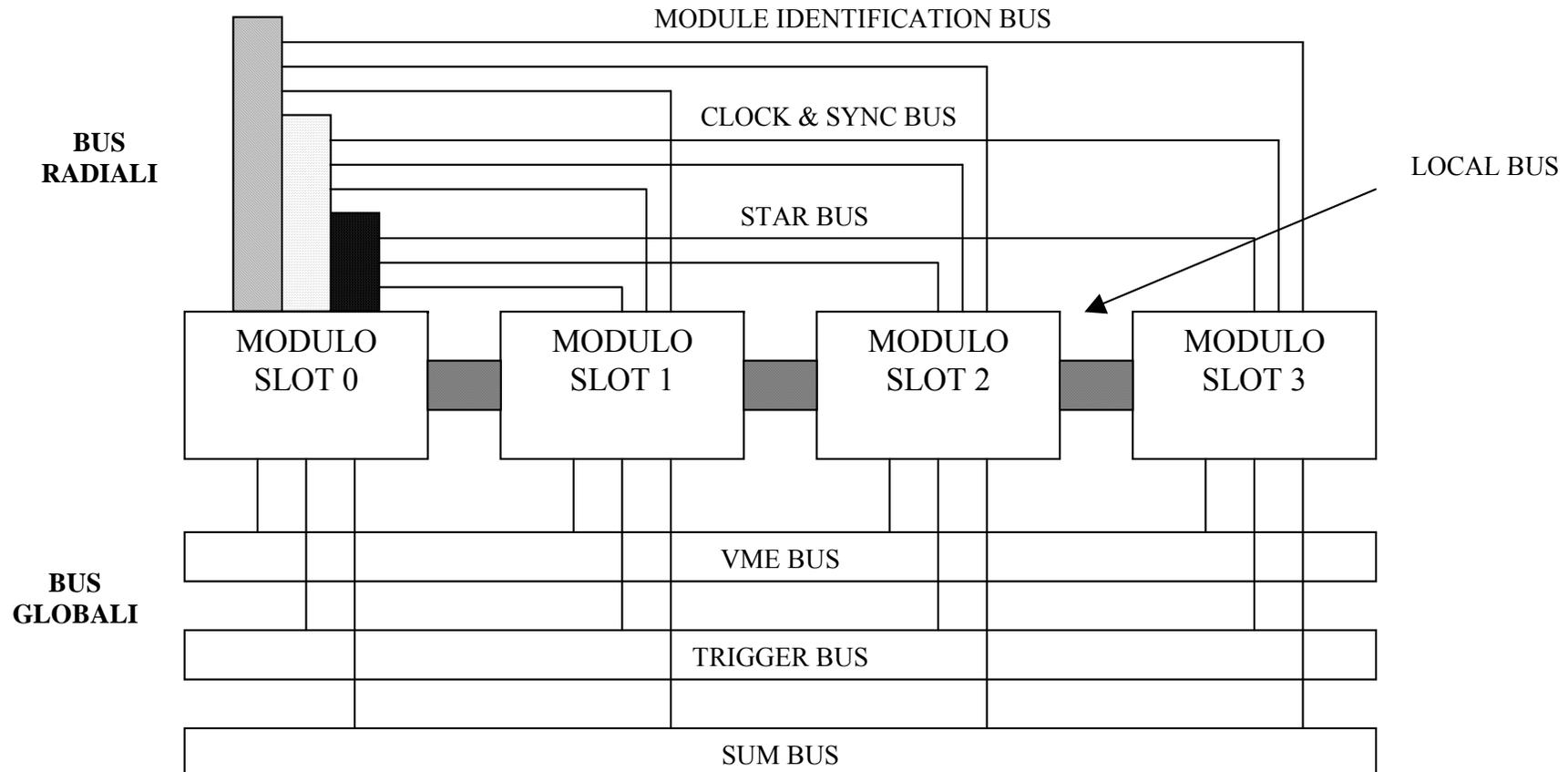
IL SISTEMA VXI



Introduzione al VXI Bus

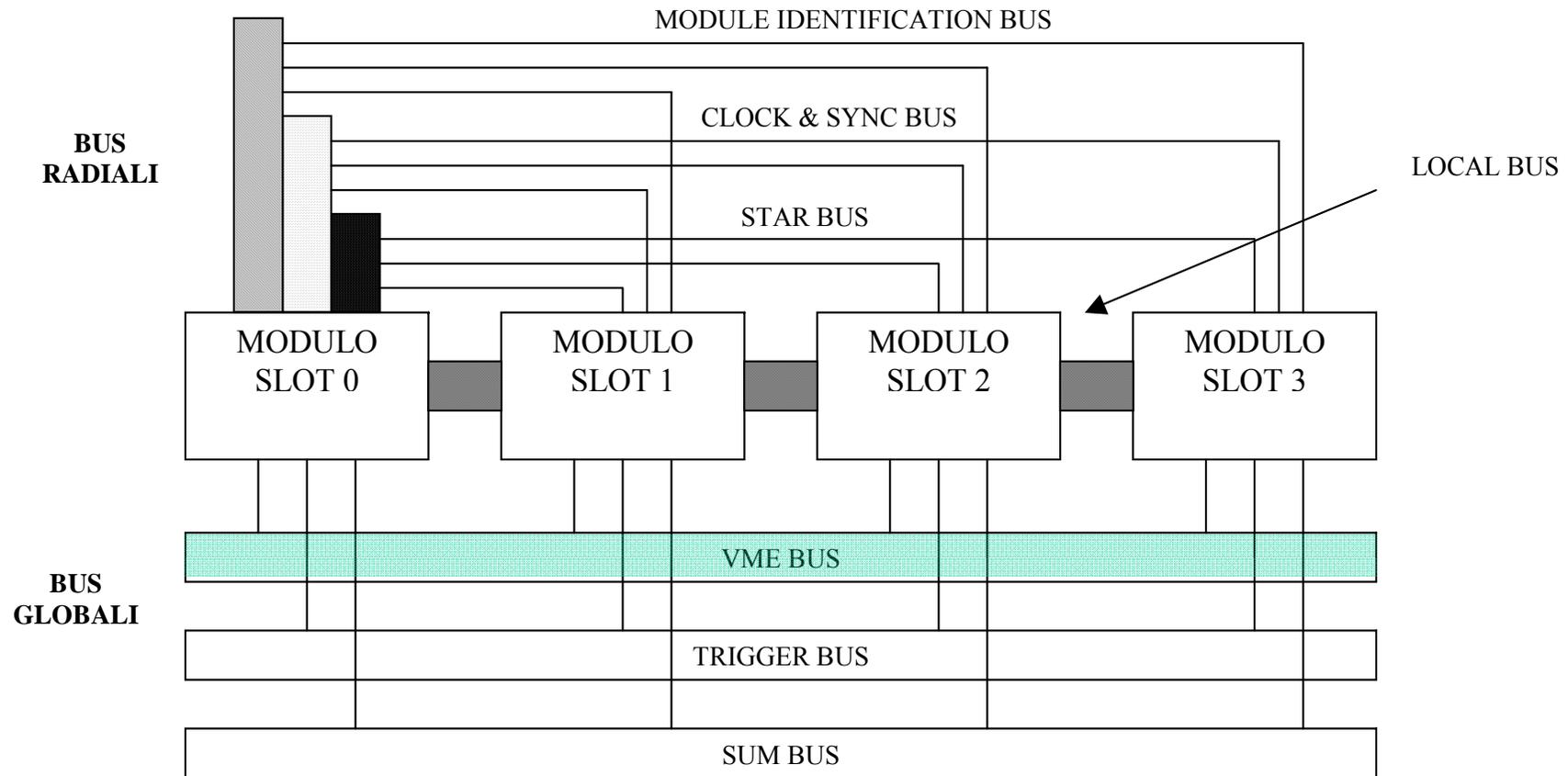
BUS VXI

Il sistema VXI è organizzato secondo diversi tipi di bus a cui sono delegate specifiche funzioni.



Introduzione al VXI Bus

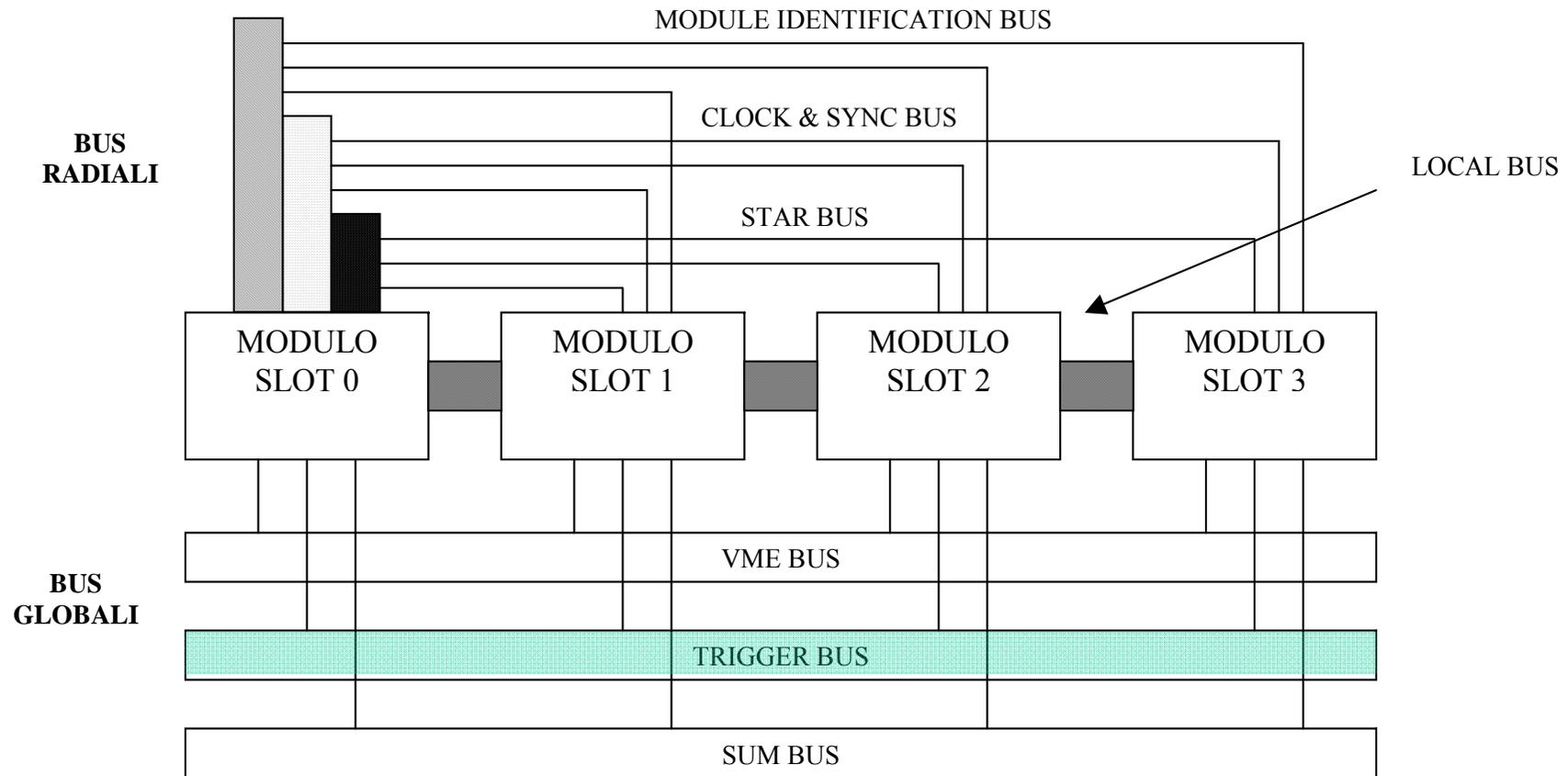
BUS VXI



VME bus (DTB, Arbitration bus, Interrupt Bus, Utility Bus)

Introduzione al VXI Bus

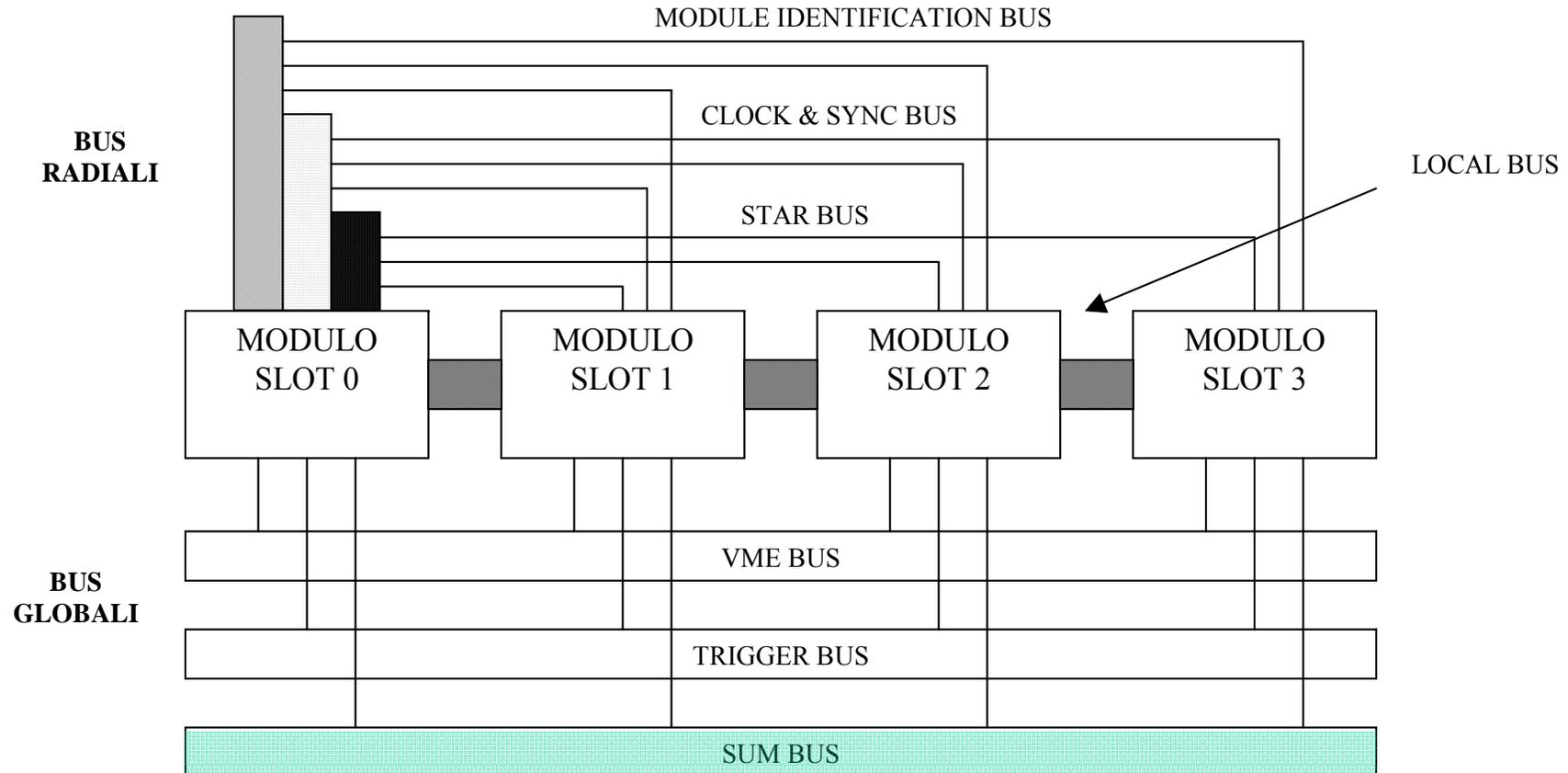
BUS VXI



Trigger bus: di tipo globale, composto da linee di trigger in logica TTL ed ECL

Introduzione al VXI Bus

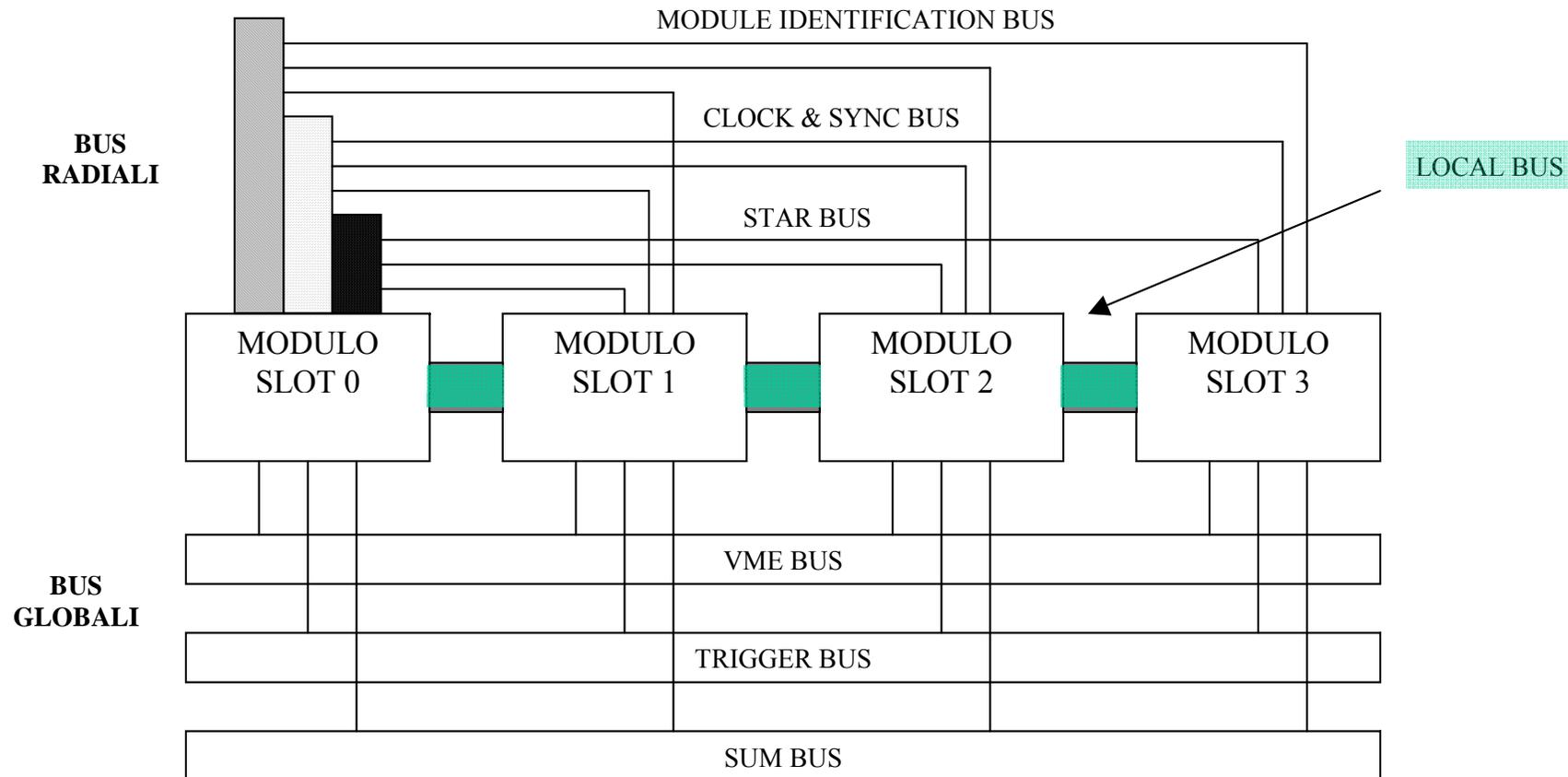
BUS VXI



Analog Sum bus: di tipo globale, è un nodo di somma analogica che percorre tutto il backplane e termina su un carico di 50Ω . Utilizzato per generare forme d'onda complesse attraverso la somma delle uscite elementari di più moduli sorgenti

Introduzione al VXI Bus

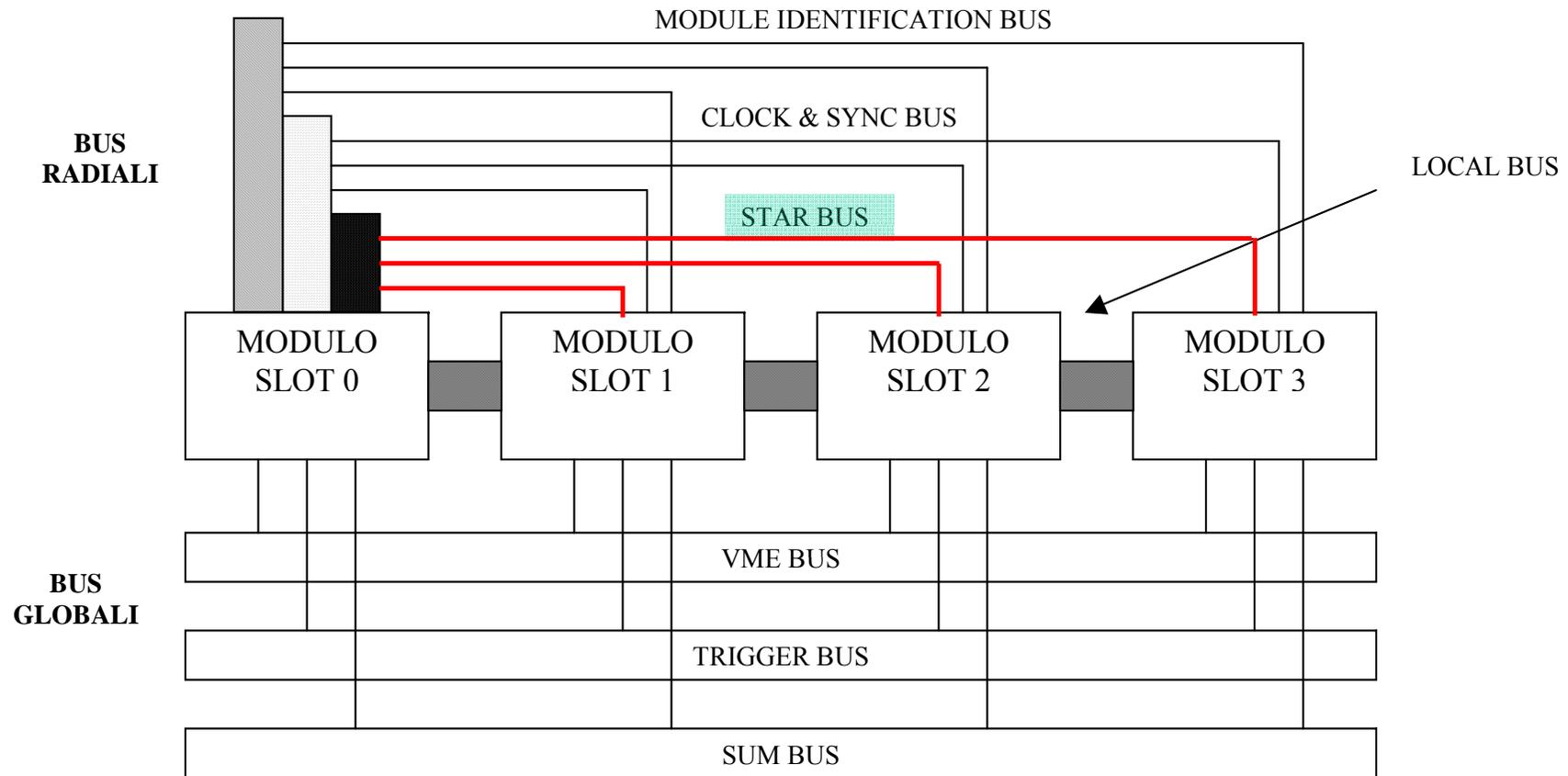
BUS VXI



Local bus: di tipo locale, consente di realizzare una struttura daisy-chain per le comunicazioni private tra moduli adiacenti

Introduzione al VXI Bus

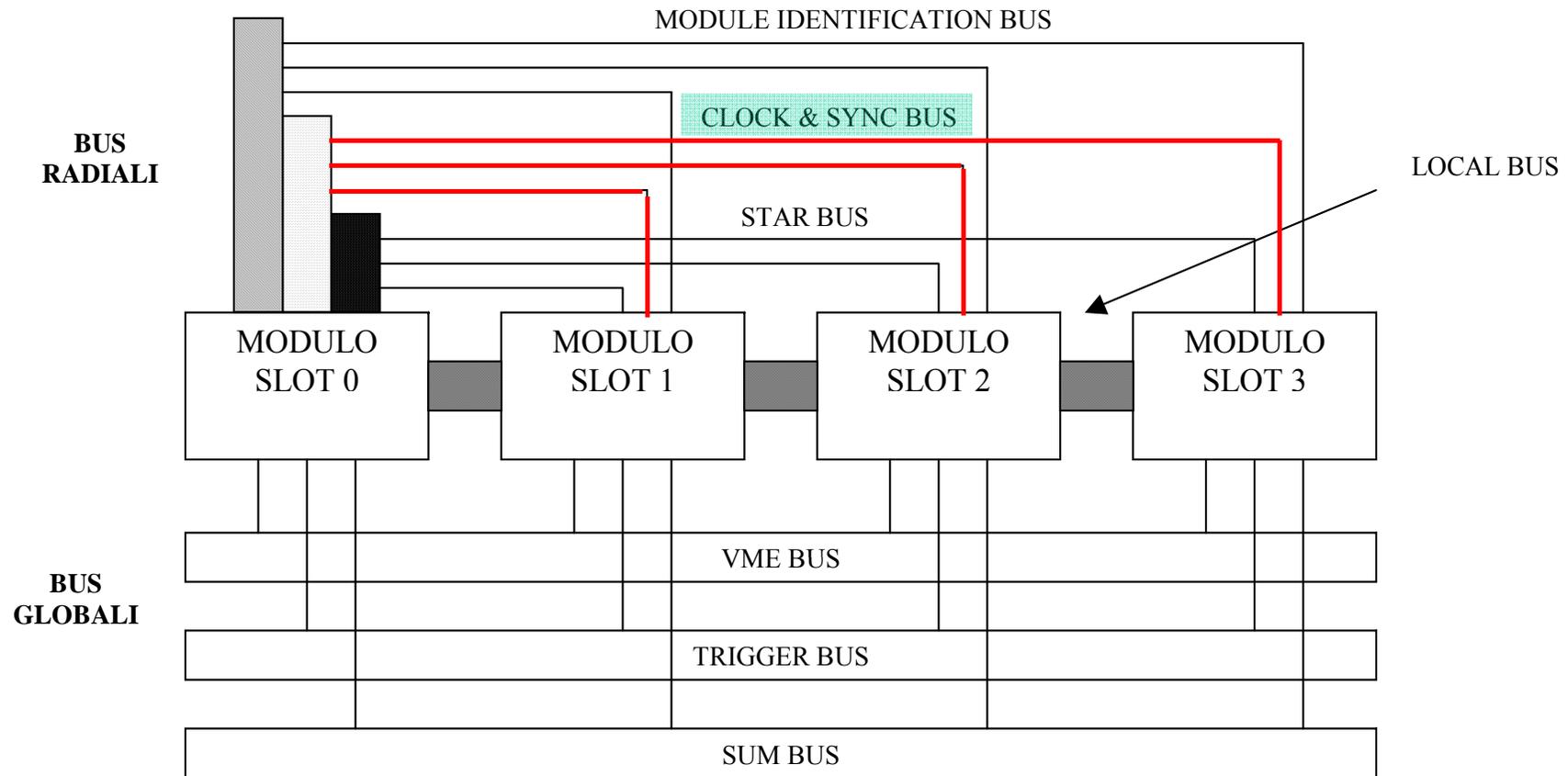
BUS VXI



Star bus: di tipo radiale, è disponibile solo per i moduli di taglia D (si trova sul connettore P3). E' costituito da una coppia di linee per ogni slot e consente di avere diverse tensioni di alimentazione.

Introduzione al VXI Bus

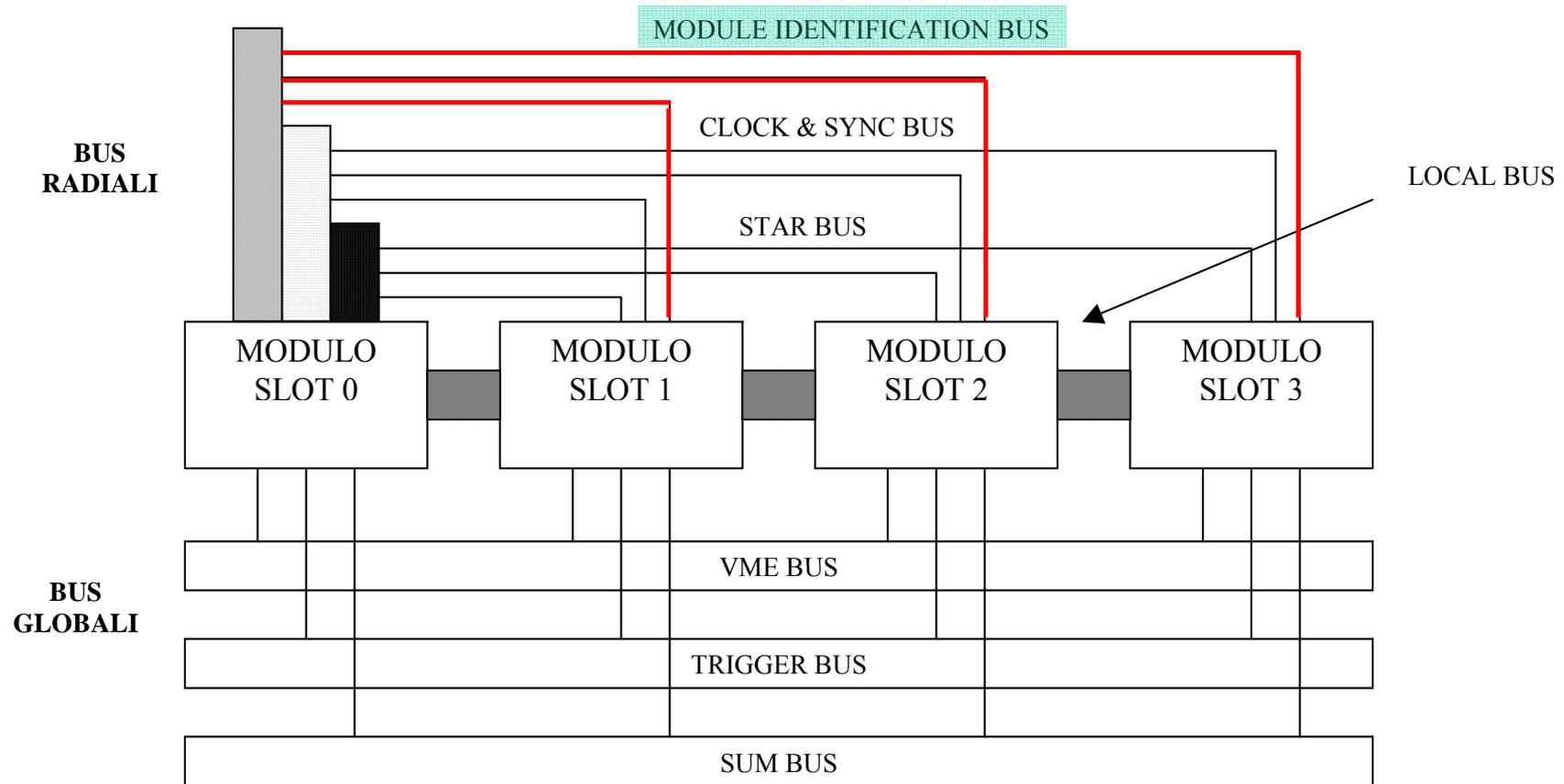
BUS VXI



Clock bus: di tipo radiale, fornisce due linee di clock (10MHz e 100MHz) ed una di sincronismo.

Introduzione al VXI Bus

BUS VXI



Module Identification bus: di tipo radiale, è costituito da 13 linee che consentono al controllore (in slot 0) la localizzazione logica e fisica di un dispositivo appartenente al sistema

Introduzione al VXI Bus

IL SISTEMA VXI

Ogni sistema VXI è caratterizzato da una struttura gerarchica ad albero:

COMMANDER: può controllare tutti i dispositivi appartenenti al proprio sotto-albero, detti SERVANT

Solo un COMMANDER può avviare in modo asincrono la comunicazione con un suo SERVANT

Un SERVANT può richiedere servizio attraverso un interrupt al suo COMMANDER

Introduzione al VXI Bus

I DISPOSITIVI VXI

Lo standard IEEE-1155 classifica i dispositivi in base al numero ed alla funzione dei registri di cui dispongono:

- 1) Register-based
- 2) Message-based
- 3) A memoria
- 4) Estesi

Tutti i dispositivi VXI possiedono un set di registri standard “*di configurazione*” che contengono le informazioni necessarie alla configurazione e al controllo del dispositivo all’interno del sistema VXI

Introduzione al VXI Bus

I DISPOSITIVI REGISTER-BASED

Sono dotati di un unico set di registri standard: i *registri di configurazione*

Sono dotati di alcuni registri non standard di dispositivo (device dependent)

La comunicazione con i dispositivi Register-based, si sviluppa accedendo ai *registri di dispositivo* mediante cicli di handshake VME in lettura e/o scrittura

Il contenuto e la locazione di questi registri è device dependent, di conseguenza non è previsto un protocollo di interfaccia standard, ma spetta all'utente, nella realizzazione del software applicativo, tenere conto dell'influenza della parte strumentale

La necessità di sviluppare per ogni dispositivo dei programmi *ad hoc* per la comunicazione ed il controllo comporta il vantaggio di poter sfruttare la semplicità dell'handshake VME per ottenere elevate velocità di trasferimento dati

Introduzione al VXI Bus

I DISPOSITIVI MESSAGE-BASED

Sono dotati di un secondo set di registri standard: i *registri di comunicazione* mediante i quali si stabilisce un canale di comunicazione standard tra dispositivi dello stesso tipo

Il contenuto e la locazione di tali registri sono specificati dalle norme e su di essi è basata una famiglia di protocolli di comunicazione standard: WSP (Word Serial Protocols)

I WSP sfruttano in particolare due dei 6 registri di comunicazione: *Response Register* e il *Data Low Register*

Introduzione al VXI Bus

I DISPOSITIVI MESSAGE-BASED

I bit del *Response Register* indicano al Commander se il registro dati (*Data Low Register*) è vuoto in scrittura, ovvero pieno per la lettura:

1) è possibile scrivere nel *Data Low Register* solo quando il bit “Write Ready” del *Response Register* è ad 1. Viene posto a 0 nel momento in cui vi si accede e mantenuto tale finchè il dato non viene accettato dal Servant

2) è possibile leggere dal *Data Low Register* solo quando il bit “Read Ready” del *Response Register* è ad 1. Quando il dato viene letto, il bit passa a 0 e vi rimane finchè il Servant non pone un nuovo dato nel registro

Modalità di handshake: Normal Transfer Mode

Introduzione al VXI Bus

I DISPOSITIVI MESSAGE-BASED VXI-BUS INSTRUMENT

Sono dei dispositivi message-based per i quali si distinguono due parti fondamentali:

- 1) Interfaccia verso il BUS
- 2) Parte strumentale

Oltre agli *interface commands*, sono previsti dei comandi non standard device dependent

In tali dispositivi, oltre ai bit “write ready” e “read ready” che “informano” sullo stato dell’interfaccia, nel *Response Register* sono presenti altri due bit (DIR e DOR) che sincronizzano la comunicazione con la parte strumentale attraverso il protocollo BTP (Byte Transfer Protocol)

Introduzione al VXI Bus

I DISPOSITIVI A MEMORIA

Funzionano come un register-based e sono specializzati per il trasferimento veloce di grandi quantità di dati

I DISPOSITIVI ESTESI

Aggiungono un *Subclass Register* che consente di definire nuove sottoclassi di dispositivi VXIbus per applicazioni future

Al momento non vi è nessuna definizione di sottoclasse standard

Introduzione al VXI Bus

RISORSE DEL SISTEMA VXI

Un sistema VXI prevede una serie di risorse comuni, alcune allocate allo start-up, altre disponibili anche dopo lo start-up come proprietà di slot0

Il *Resource Manager* è un modulo software con indirizzo logico 0 che, sfruttando alcune proprietà di slot 0 effettua la configurazione del sistema eseguendo le seguenti funzioni:

- 1) Identifica tutti i dispositivi VXIbus nel sistema (attraverso il servizio MODID)
- 2) Gestisce i risultati del self-test e la sequenza diagnostica
- 3) Configura le mappe degli indirizzi A24 e A32
- 4) Configura la gerarchia Commander/Servant del sistema
- 5) Alloca le linee IRQx VMEBus
- 6) Da' il via al "Normal Operation" ("pronto operativo") del sistema

Il *Resource Manager* deve essere un Commander VXI in testa alla gerarchia (altrimenti non potrebbe accedere ai registri di configurazione e comunicazione di ogni altro dispositivo)

Introduzione al VXI Bus

CONFIGURAZIONE DEL SISTEMA VXI

Una tipica configurazione di un sistema VXI è composta da:

- 1) Controllore
- 2) Interfaccia tra Controllore e VXIBus
- 3) Uno o più sottosistemi VXI contenenti moduli di strumentazione

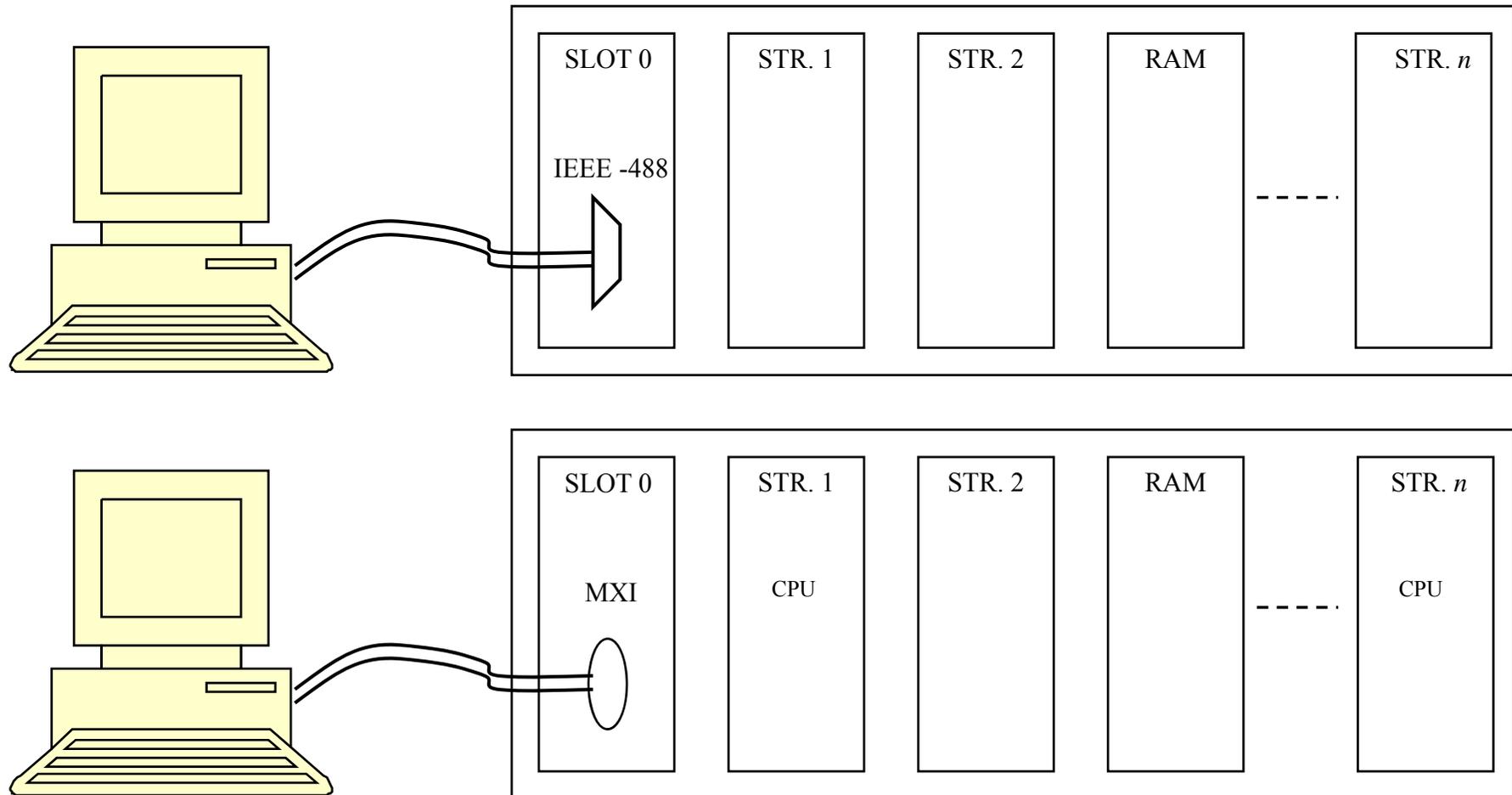
Tutte le possibili configurazioni possono essere raggruppate in due classi dipendentemente da come il Controllore si collega al bus VXI

Il Controllore può essere interno (embedded) o di tipo esterno

Introduzione al VXI Bus

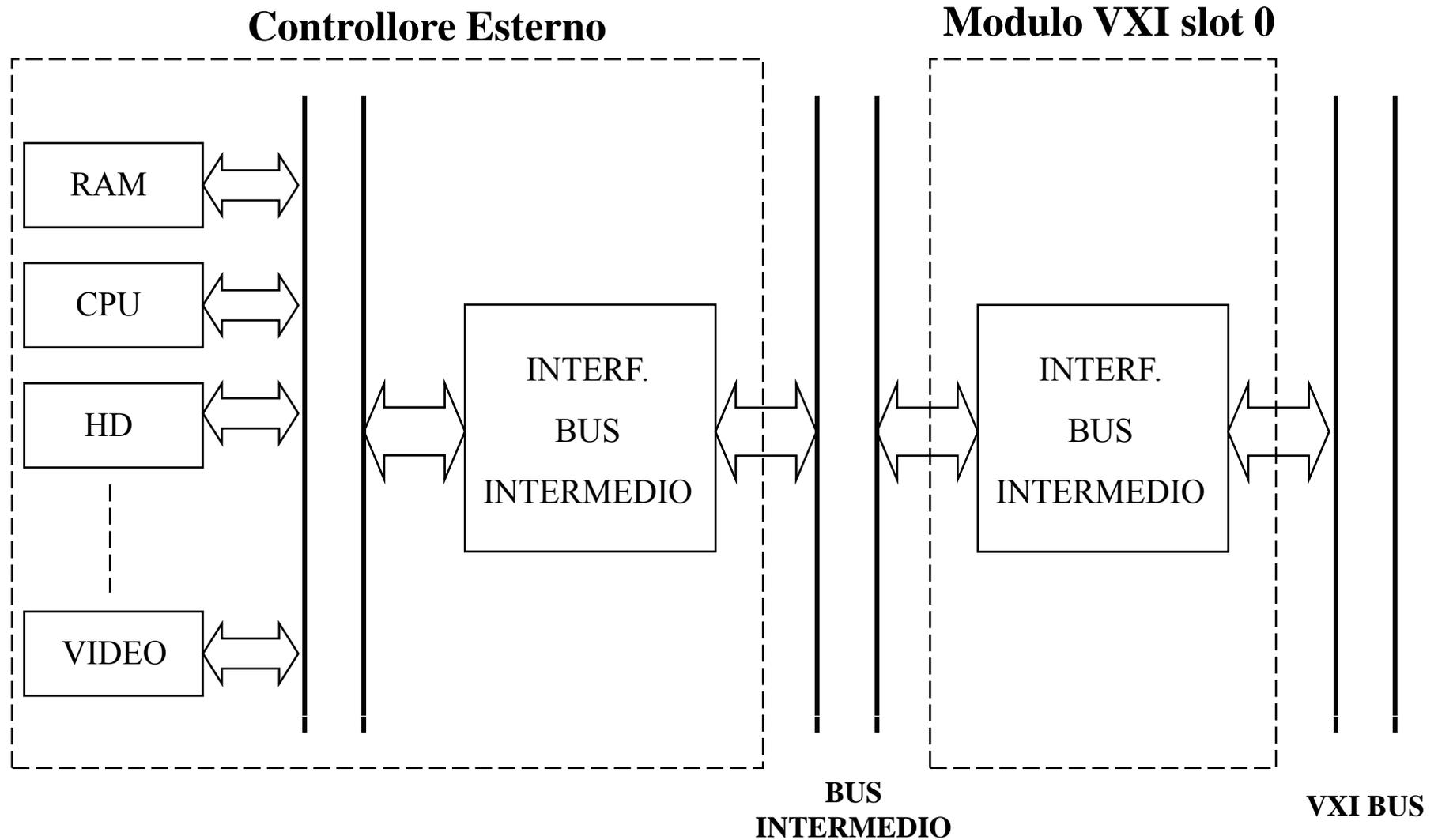
CONFIGURAZIONE DEL SISTEMA VXI

Controllore Esterno



Introduzione al VXI Bus

CONFIGURAZIONE DEL SISTEMA VXI



Introduzione al VXI Bus

CONFIGURAZIONE DEL SISTEMA VXI

Controllore Esterno

Il controllore (collocato all'esterno del cestello) comunica con il bus VXI attraverso un opportuno sistema di interfaccia composto da

1) **Scheda di interfaccia** tra il bus di I/O del PC (controllore) ed un bus intermedio

2) **Il bus intermedio**

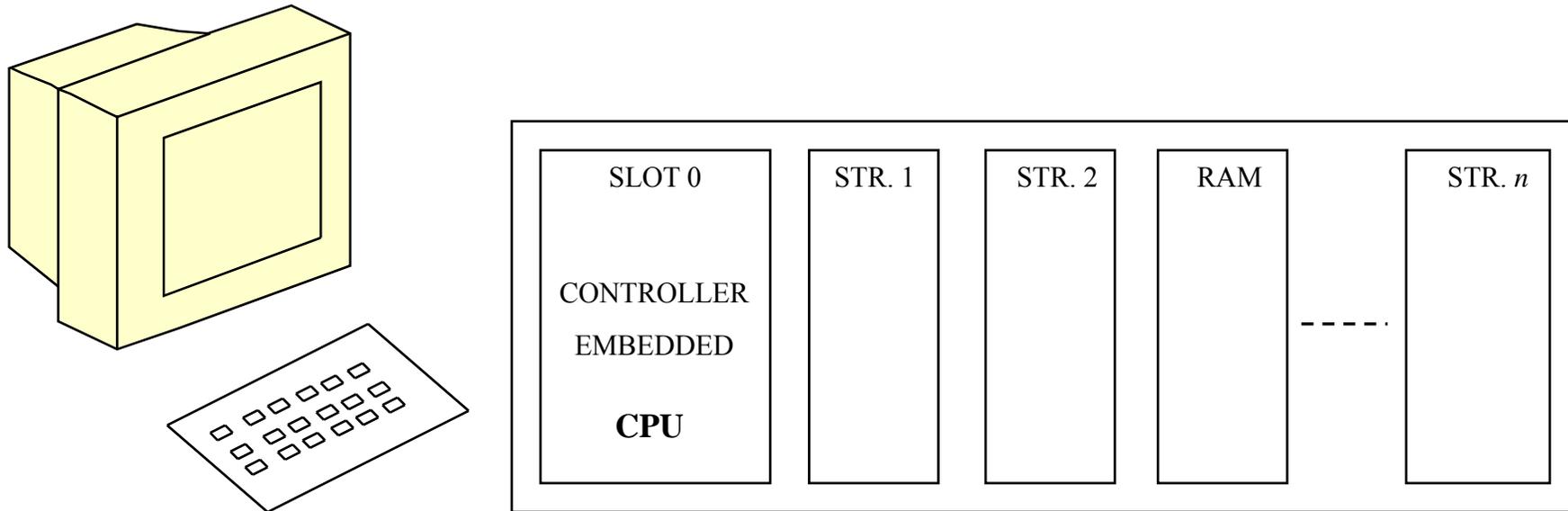
I bus intermedi maggiormente utilizzati sono: RS-232, GPIB e l'MXI (della NI) che consente delle prestazioni superiori in termini di velocità

3) Un **modulo VXI con funzioni di Slot 0**, dotato di interfaccia verso il bus intermedio ed alloggiato nel cestello VXI

Introduzione al VXI Bus

CONFIGURAZIONE DEL SISTEMA VXI

Controllore Interno



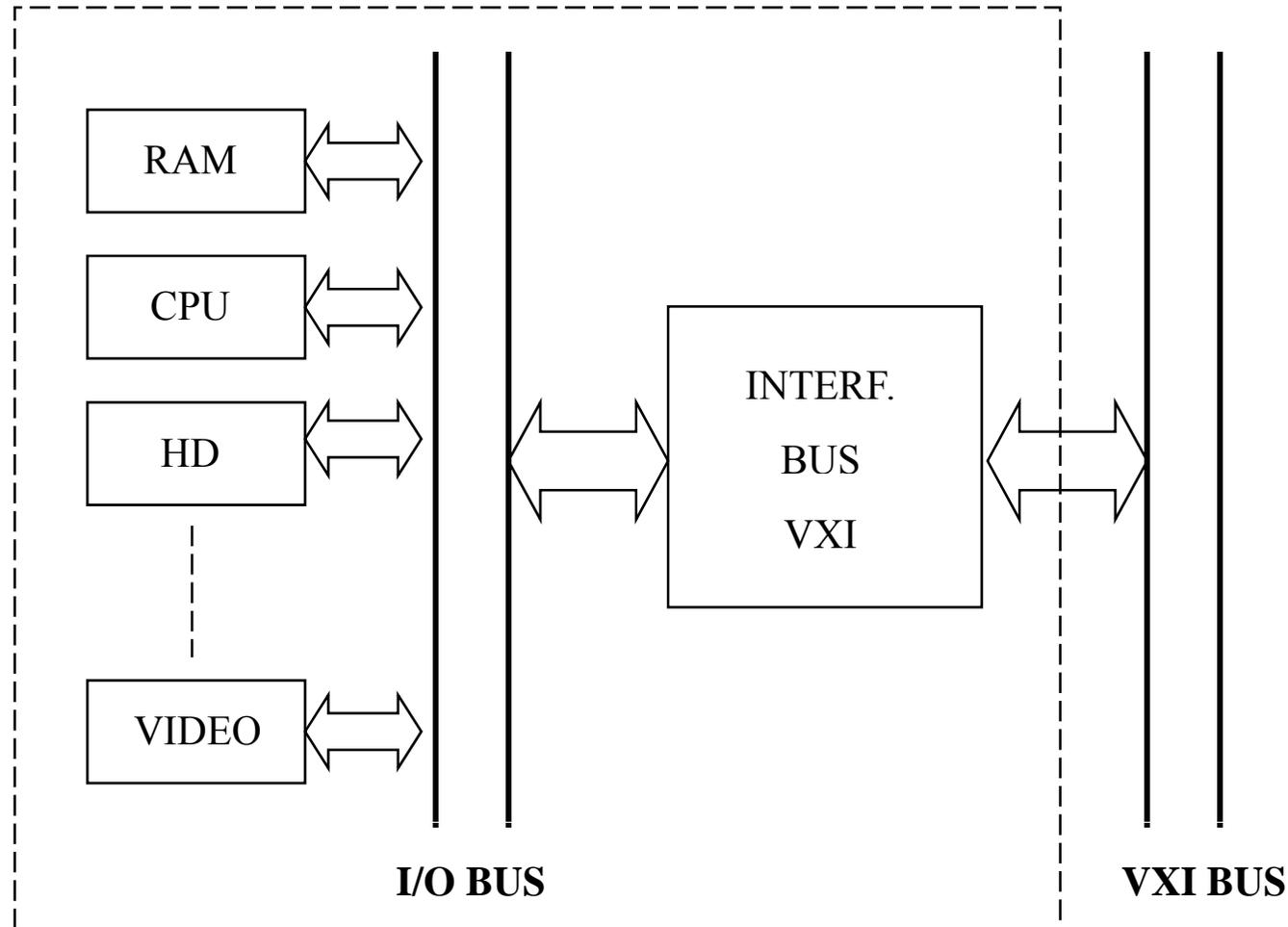
La CPU è interna al cestello

Monitor, Tastiera, Mouse, ecc. sono necessari se si vuole programmare il sistema

Introduzione al VXI Bus

CONFIGURAZIONE DEL SISTEMA VXI

Controllore Interno



Introduzione al VXI Bus

CONFIGURAZIONE DEL SISTEMA VXI

Controllore Interno

Il PC-controllore è alloggiato direttamente nel cestello e sul proprio I/O bus ospita una scheda di interfaccia diretta verso il bus VXI

La scheda di interfaccia ha capacità di master VME e consente il controllo diretto del bus VXI senza interfaccia intermedia

In questa configurazione il controllore può arrivare a velocità di trasferimento dati verso i dispositivi VXI di 5-10 MByte/s

Il VXI Bus

PROGETTAZIONE DELLA PROCEDURA DI MISURA IN UN SISTEMA VXI

La realizzazione di un sistema automatico di misura richiede necessariamente lo sviluppo di programmi per:

- 1) La gestione dei dispositivi di misura ed attuazione (eseguono l'algoritmo di misura e gestiscono la comunicazione tra i vari componenti del sistema)
- 2) L'elaborazione dei dati misurati (forniscono i risultati finali e parziali di ogni fase della procedura di misura)

Il VXI Bus

PROGETTAZIONE DELLA PROCEDURA DI MISURA IN UN SISTEMA VXI

Per potere adeguatamente definire l'algoritmo di misura occorre conoscere anche le caratteristiche di interfaccia dei dispositivi impiegati



Uso appropriato (che sfrutti appieno le potenzialità) del bus e dei dispositivi VXI

Il VXI Bus

PROGETTAZIONE DELLA PROCEDURA DI MISURA IN UN SISTEMA VXI

E' importante sapere se il dispositivo ha capacità di MASTER!!

Infatti, il trasferimento dati tra due moduli SLAVE avviene in modo indiretto, coinvolgendo il controllore:

- 1) Raddoppia il tempo di impiego del bus (dispositivo 1- controllore, controllore - dispositivo 2)
- 2) Raddoppia il tempo di impiego della CPU del controllore, che invece potrebbe essere destinata ad altri compiti

E' importante sapere se il dispositivo ha capacità di INTERRUZIONE (interrupt)!!

Infatti, in tal modo è possibile alleggerire il software di controllo dalla esecuzione periodica di onerose procedure di interrogazione (“polling”)

Il VXI Bus

PROGETTAZIONE DELLA PROCEDURA DI MISURA IN UN SISTEMA VXI

E' raro disporre di manuali che offrano informazioni dettagliate già sulle caratteristiche metrologiche del dispositivo; ancor più lacunosa risulta, in genere, la parte dedicata all'interfaccia ed alle specifiche VXI.

E' quindi necessario integrare, le scarse informazioni ricavate dai manuali, dal contenuto dei registri dei dispositivi.

Registri standard:

- **Registri di configurazione**
- **Registri di comunicazione**

Il VXI Bus

I REGISTRI DI CONFIGURAZIONE

Sono registri standard (presenti in tutti i tipi di dispositivi)

1. ID/Logical Address Register
 2. Device Type
 3. Status/Control Register
 4. Offset Register
- } OBBLIGATORI
- } OPZIONALE

Sono allocati in modo sequenziale a partire dall'indirizzo base ed occupano ognuno 16 bit.

Il primo e il terzo hanno un significato differente a seconda che siano utilizzati in lettura o in scrittura

Il VXI Bus

I REGISTRI DI CONFIGURAZIONE

ID/Logical Address Register

In lettura:

ID Register, fornisce informazioni riguardanti la configurazione del dispositivo:

- 1) Message-based, Register-based, ecc.
- 2) Modalità di indirizzamento dei registri operativi (A16, A16/A24, A16/A32)
- 3) Identificazione del produttore secondo un codice preciso

In scrittura:

Logical Address Register, definito dal *Resman* nell'attribuzione dinamica dell'indirizzo

Il VXI Bus

I REGISTRI DI CONFIGURAZIONE

Device Type Register

E' a sola lettura ed è diviso in due campi:

- 1) *Required Memory*: definisce la quantità aggiuntiva di memoria richiesta dai dispositivi
- 2) *Model Code*: contiene un identificatore della scheda definito dal produttore

Offset Register

Accessibile sia in lettura che scrittura

E' obbligatorio solo per dispositivi A24 o A32 e, serve a definire l'indirizzo base dei registri operativi A24 o A32 del dispositivo

Il VXI Bus

I REGISTRI DI CONFIGURAZIONE

Status/Control Register

In lettura:

Status Register, fornisce informazioni riguardanti lo stato del dispositivo:

- 1) *MODID*: se ad 1 indica che il dispositivo non è stato selezionato con 1 linea MODID .
- 2) *Ready*: utilizzato insieme al bit successivo (*Passed*), entrambi ad 1 indicano che il dispositivo è pronto per eseguire l'intero set di comandi operativi
- 3) *Passed*: se ad 1 indica che il dispositivo ha completato con successo il self-test iniziale
- 4) *Device dependent*

Il VXI Bus

I REGISTRI DI CONFIGURAZIONE

Status/Control Register

In scrittura:

Control Register, in esso vengono definite le azioni che il dispositivo deve eseguire:

- 1) *Sysfail Inhibit*: disabilita il driver del dispositivo sulla linea SYSFAIL.
- 2) *Reset*: forza il dispositivo in uno stato di reset
- 3) *A24/A32 Enable*: abilita l'accesso ai registri A24 o A32
- 4) *Device dependent*

Il VXI Bus

I REGISTRI DI COMUNICAZIONE

Sono registri standard e sono presenti per i dispositivi Message-based

- Protocol/Signal Register
- Response/Data Extended Register
- Data High Register
- Data Low Register
- Registri riservati per futuri sviluppi

Sono costituiti da 16 bit

1,2 e 4 sono obbligatori

Il primo e il secondo hanno un significato differente a seconda che siano utilizzati in lettura o in scrittura

Il VXI Bus

I REGISTRI DI COMUNICAZIONE

Protocol/Signal Register

In lettura:

Protocol Register, indica la capacità di interfaccia, i protocolli supportati, ed eventuali capacità aggiuntive di comunicazione di un dispositivo:

- 1) *CMDR*: se ad 1 il dispositivo ha solo capacità di SERVANT.
- 2) *Master*: uno 0 indica che il dispositivo ha capacità di MASTER
- 3) *Interrupter*: se ad 1 il dispositivo ha capacità di interruzione
- 4) *FHS*: se a 0 il dispositivo supporta la modalità Fast Handshake
- 5) *Shared Memory*: se a 0 il dispositivo supporta il protocollo di comunicazione opzionale "Shared Memory"
- 6) *Device dependent*

In scrittura:

Signal Register, utilizzato dal Commander per ricevere segnalazioni dai Servant aventi capacità di Master

Il VXI Bus

I REGISTRI DI COMUNICAZIONE

Response/Data Extended Register

In lettura:

Response Register, informa sullo stato dei registri di comunicazione del dispositivo e delle loro funzioni associate:

- 1) ***DOR (Data Output Ready)***: se ad 1 il dispositivo è pronto per fornire dati
- 2) ***DIR (Data Input Ready)***: se ad 1 il dispositivo è pronto per ricevere dati
- 3) ***FHS Active***: se a 0 indica che la modalità di Fast Handshake è abilitata
- 4) ***Locked***: se a 0 il Commander ha inibito l'accesso allo strumento tramite pannello frontale
- 5) ***Read Ready***: se a 1 sono disponibili nel Data Low Register e Data High Register dati provenienti dallo strumento; viene riportato a 0 in seguito ad una lettura del Data Low Register da parte del Commander
- 6) ***Write Ready***: se a 1 il dispositivo è pronto a ricevere nuovi dati; viene riportato a 0 in seguito ad una scrittura del Data Low Register da parte del Commander
- 7) ***Device dependent***

Il VXI Bus

I REGISTRI DI COMUNICAZIONE

Response/Data Extended Register

In scrittura:

Data Extended Register, è opzionale e rappresenta la Word più significativa del dato o comando

Data High Register (opzionale)

In lettura:

Rappresenta la Word più significativa del dato o comando

In scrittura:

Rappresenta la seconda Word più significativa del dato o comando

Il VXI Bus

I REGISTRI DI COMUNICAZIONE

Data Low Register

In lettura:

Rappresenta la Word meno significativa del dato o comando

In scrittura:

Rappresenta la Word meno significativa del dato o comando.

Una scrittura provoca l'esecuzione da parte del dispositivo di qualche azione dipendente dal contenuto dei registri Data High, Data Low e Data Extended Registers

Il VXI Bus

IL GESTORE DELLE RISORSE VXI: IL RESMAN

Il **RESMAN** è un modulo software che esegue tutte le funzioni che lo standard VXI definisce per il Resource Manager se il controllore è stato configurato con indirizzo logico 0

Le operazioni principali che il Resman esegue sono in sequenza:

- 1) *Identificazione dei dispositivi VXI*
- 2) *Gestione del Self-Test*
- 3) *Configurazione della mappa degli indirizzi*
- 4) *Gerarchie Commander/Servant*
- 5) *Allocazione delle linee di interruzione*
- 6) *Inizio Normali Operazioni*

Il VXI Bus

IL GESTORE DELLE RISORSE VXI: IL RESMAN

Identificazione dei dispositivi VXI

Dopo un tempo prefissato (5 secondi), il Resman, iniziando dal controllore scandisce tutti gli indirizzi logici da 0 a 254 per i dispositivi configurati staticamente

Per ogni dispositivo identificato, il Resman:

- 1) Interroga l' ID Register ed il Device Type Register
- 2) Associa lo slot leggendo lo Status Register asserendo la linea MODID di ciascuno slot
- 3) Scrive l'indirizzo logico nel Logical Address Register

Il VXI Bus

IL GESTORE DELLE RISORSE VXI: IL RESMAN

Gestione del Self-Test

Se qualche dispositivo VXI non ha passato il proprio self-test (non ha asserito il bit *passed* presente nello Status Register) entro il tempo specificato, il Resman pone il dispositivo nello stato *safe* e non viene assegnato come Servant a nessun Commander

Configurazione della mappa degli indirizzi

Il Resman determina lo spazio di indirizzamento di ogni dispositivo accedendo al registro *ID*

Il VXI Bus

IL GESTORE DELLE RISORSE VXI: IL RESMAN

Gerarchie Commander/Servant

Il Resman individua tutti i dispositivi con capacità di Commander (testando il bit CMDR del *protocol register* di ciascun dispositivo message-based) e individuando anche i corrispondenti dispositivi Servant li assegna al Commander

Allocazione delle linee di interruzione

Il Resman alloca le linee di interruzione (IRQ) del bus VME tra i vari *interrupter* e *interrupt handler*

Inizio Normali Operazioni

Il Resman invia un comando di “Begin Normal Operation” a tutti i Commander

Il VXI Bus

PROGETTAZIONE DI UN SISTEMA VXI

Scelta del controllore

L'architettura di un sistema VXI è fortemente condizionata dalla scelta del tipo di controllore in termini di dimensioni, espandibilità, costi, prestazioni

Controllore INTERNO

- 1) Maggiore compattezza
- 2) Riduzione dello spazio disponibile nel cestello (occupa anche 3 slot)
- 3) Riduzione della potenza elettrica disponibile nel cestello
- 4) Calcolatore dedicato
- 5) Maggiore throughput (velocità di comunicazione maggiori)

Il VXI Bus

PROGETTAZIONE DI UN SISTEMA VXI

Scelta del controllore

Controllore **ESTERNO**

- 1) Minore “impegno” all’interno del cestello
- 2) Sul bus intermedio possono essere allacciati ulteriori dispositivi aumentando la “dimensione” del sistema VXI
- 3) Soluzione più economica
- 4) Calcolatore non dedicato
- 5) Minore throughput (velocità di comunicazione minori) a causa del passaggio attraverso il bus intermedio

Il VXI Bus

PROGETTAZIONE DI UN SISTEMA VXI

Scelta del controllore

Controllore **ESTERNO**

Per quel che riguarda il bus intermedio, le soluzioni più diffuse sono due

GPIB (IEEE-488)



- 1) Si possono integrare stazioni VXI e IEEE-488
- 2) Velocità di trasferimento effettive di 500Kbyte/s

MXI



Riporta all'esterno del cestello un sottoinsieme del VXIbus:

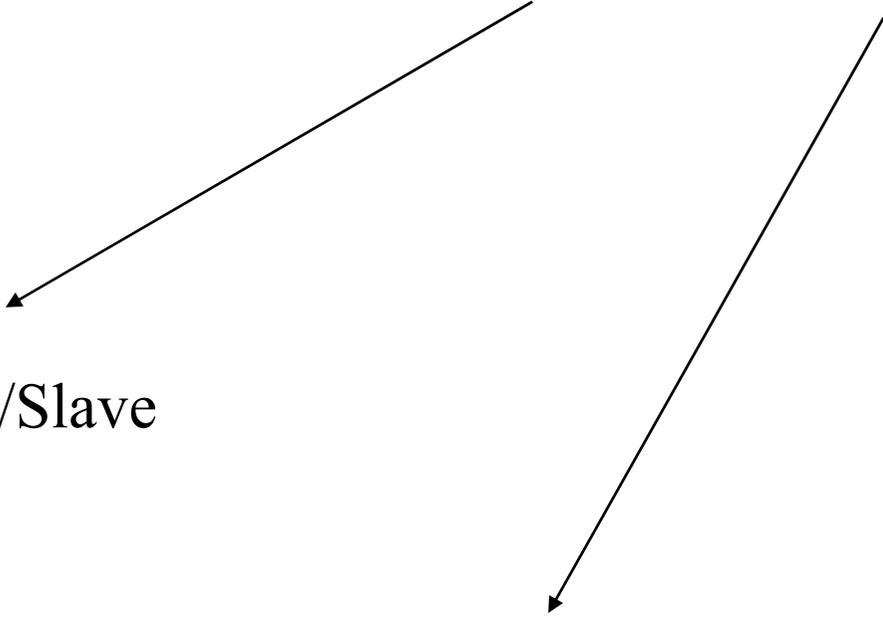
- 1) trasferimento fino a 8 dispositivi MXI, con velocità di 10MByte/s
- 2) arbitraggio del bus
- 3) gestione delle interruzioni ed anomalie

Il VXI Bus

PROGETTAZIONE DI UN SISTEMA VXI

Le gerarchie

Si può stabilire una gerarchia sia rispetto al bus VME che al bus VXI



Di basso livello: Master/Slave

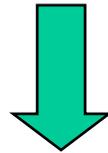
Di più alto livello: Commander/Servant

Il VXI Bus

PROGETTAZIONE DI UN SISTEMA VXI

Gerarchie del bus VME

- 1) La comunicazione sul bus VME si stabilisce tra un modulo *Master* e un modulo *Slave*, ma solo il *Master* ha la possibilità di avviarla e di deciderne la direzione
- 2) Un dispositivo che è solo *Slave* può segnalare la necessità di comunicare solo se dotato di capacità di interruzione e solo il dispositivo dotato di modulo *interrupt handler* è in grado di “sentirlo”



Si stabilisce una prima gerarchia *master - slave*

Il VXI Bus

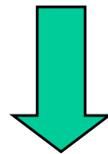
PROGETTAZIONE DI UN SISTEMA VXI

Gerarchie del bus VME

Il bus VME è *multimaster* ed è regolato da un modulo arbitro che attraverso l'Arbitration bus assegna di volta in volta, ai *master* che lo richiedono, l'accesso al DTB (Data Transfer Bus)

L'accesso al bus può essere gestito:

- 1) in *daisy chain* su di una singola linea BGnIN, BGnOUT (la priorità è dell'unità più vicina)
- 2) attraverso l'assegnazione di livelli di priorità alle linee (indipendenti) di richiesta del bus (BRn con n=0,...3)



Si stabilisce una ulteriore gerarchia tra i vari *master*

Il VXI Bus

PROGETTAZIONE DI UN SISTEMA VXI

Gerarchie del bus VXI

La comunicazione tra dispositivi è basata sulla relazione gerarchica tra *Commander* e *Servant* di tipo ad albero

Ogni *Commander* controlla solo i dispositivi appartenenti al proprio sotto-albero (*Servant*), quindi ogni *Servant* ha un solo *Commander*

In testa all'albero c'è il *Top Level Commander*

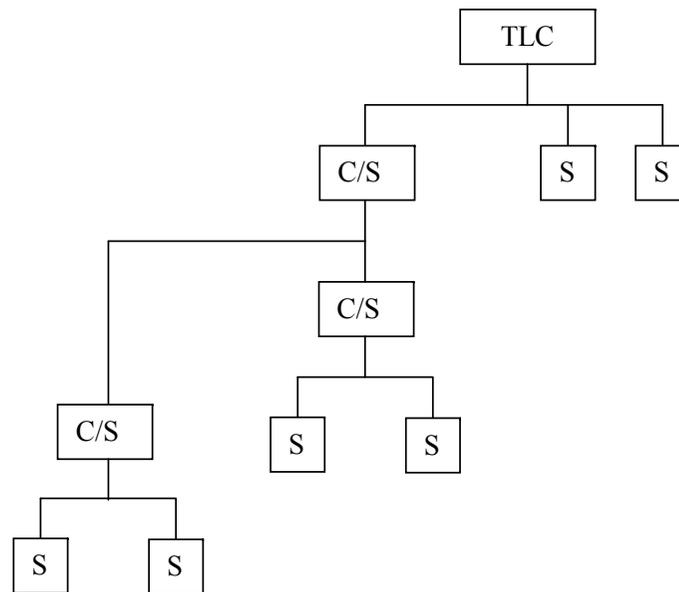
Il sistema VXI prevede la possibilità di più alberi paralleli ciascuno con il proprio *Top Level Commander*

In fase di configurazione, il Resman invia il comando WSP “Begin Normal Operation” a tutti i *Top Level Commander*. Da questi il comando si propaga in cascata attraverso i *Commander* a tutti i *Servant*

Il VXI Bus

PROGETTAZIONE DI UN SISTEMA VXI

Gerarchie del bus VXI



La funzione di *Commander* può essere esercitata solo da un dispositivo VXI:

- 1) Message-based (perché deve inviare il comando WSP “Begin Normal Operation”)
- 2) Master del bus VME per avere accesso ai registri di *configurazione* e di *comunicazione* (nel caso di Message-based) dei suoi Servant

Il VXI Bus

PROGETTAZIONE DI UN SISTEMA VXI

Scelta dei dispositivi di Misura

Nel caso di strumentazione destinata ad essere impiegata in stazioni di misura automatiche, oltre alle caratteristiche metrologiche statiche e dinamiche, anche le caratteristiche del sistema di interfaccia contribuiscono a definirne la qualità e le prestazioni

Tra gli indici che definiscono le prestazioni ci saranno:

- 1) La capacità operativa (throughput)
- 2) La riconfigurabilità
- 3) La modularità
- 4) La facilità di programmazione

Il VXI Bus

PROGETTAZIONE DI UN SISTEMA VXI

Scelta dei dispositivi di Misura

Tali indici dipendono quasi esclusivamente da:

Modalità di comunicazione

Determinano la velocità di trasmissione di dati e comandi

Capacità di sincronizzazione ed interruzione dei dispositivi che compongono il sistema

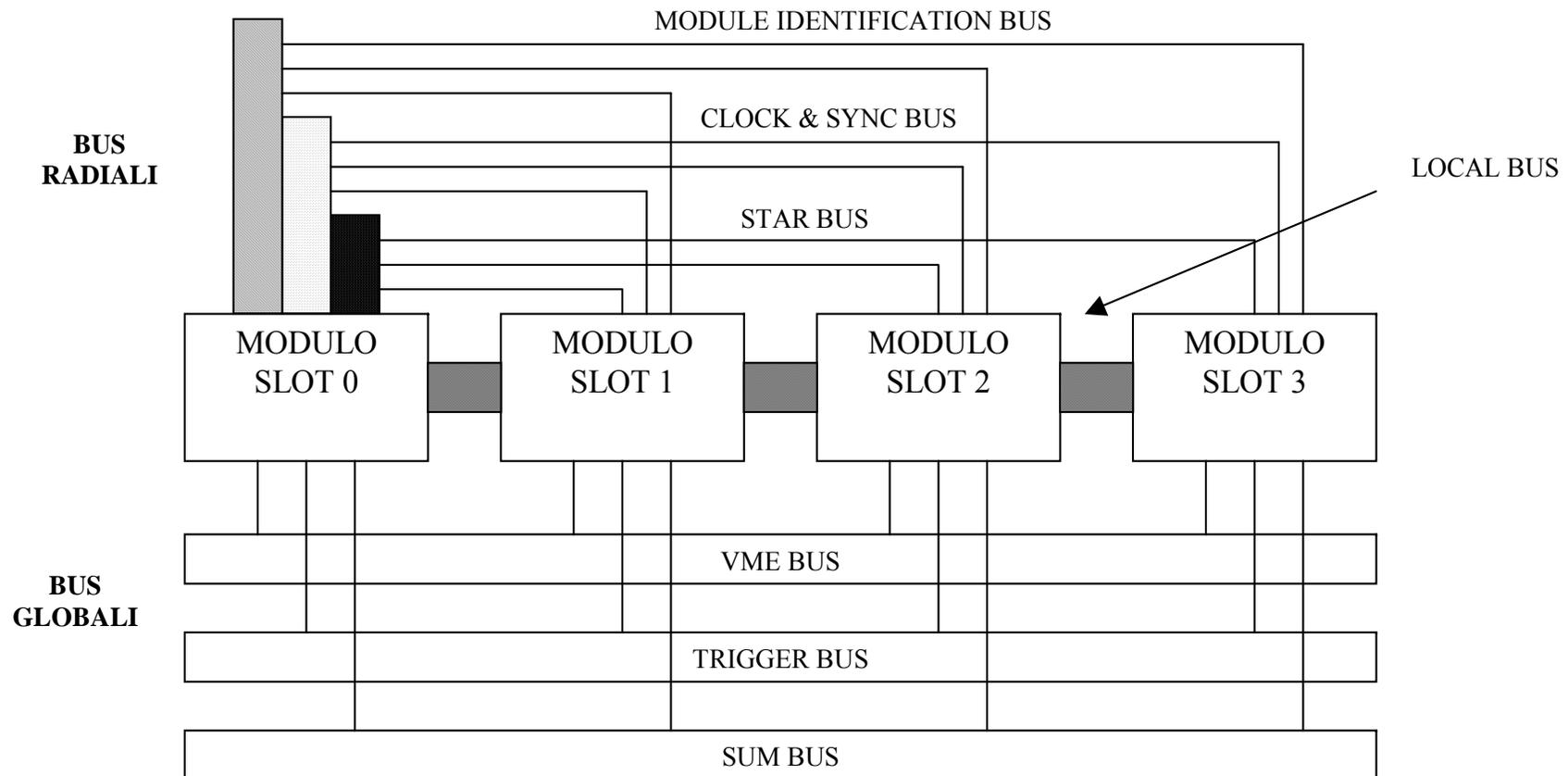
Consentono la sincronizzazione dei processi di misura e l'ottimizzazione delle risorse

*Corso di Sistemi Automatici di
Misura*

OTTAVA LEZIONE:

**I PROTOCOLLI DI COMUNICAZIONE
DEL BUS VXI**

I PROTOCOLLI DI COMUNICAZIONE DEL BUS VXI



I PROTOCOLLI DI COMUNICAZIONE DEL BUS VXI

In un sistema VXI la comunicazione tra dispositivi può svolgersi su tre diversi mezzi di trasmissione:

1) *VME Bus*: è il bus dati del sistema VXI

2) *Local Bus*: è alternativo al VME Bus ma è utilizzabile unicamente da strumenti posizionati in slot contigui.

Consente di “liberare” il bus VME rendendolo disponibile ad altri dispositivi, quindi permette di incrementare il *throughput* del sistema.

Non sono definiti dei protocolli standard

3) *Trigger Bus*: oltre a segnali di sincronizzazione può essere impiegato anche per la trasmissione dati

I PROTOCOLLI DI COMUNICAZIONE DEL BUS VXI

VME BUS

La comunicazione sul bus VME tra un *master* (che è anche *commander* del sistema VXI) ed uno slave (suo *servant*) è gestita attraverso un doppio livello di protocolli:

1. **Protocollo di handshake (primo livello)**
2. **Protocollo di interfaccia (secondo livello)**

I PROTOCOLLI DI COMUNICAZIONE DEL BUS VXI

VME BUS

Protocollo di Handshake

Il protocollo di handshake si basa su alcune linee del bus VME e permette ad un master di scrivere e leggere dati a 16 bit (*word*) in una qualsiasi locazione dello spazio di memoria VME

Ha il compito di garantire la corretta sincronizzazione tra il *master* e lo *slave*, assicurando che ogni *word* trasmessa sia correttamente ricevuta.

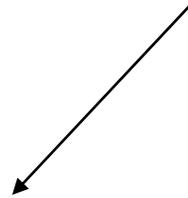
Tale protocollo è comune a tutti i dispositivi VXI ed è quello definito dallo standard VME, viene quindi chiamato **protocollo VME**

I PROTOCOLLI DI COMUNICAZIONE DEL BUS VXI

VME BUS

Protocollo di interfaccia

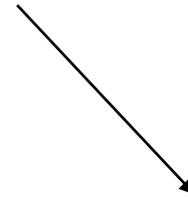
Protocollo di interfaccia si basa sul contenuto di alcuni registri dei dispositivi (ha il compito di garantire che lo stato dei registri dei dispositivi sia tale da permettere la trasmissione di una nuova *word* a 16 bit).



Dispositivi *message-based*

Sono dotati di registri di comunicazione interni standard.

Su questi è basato il protocollo di comunicazione standard WSP (Word Serial Protocol)



Dispositivi *register-based*

Non sono dotati di registri di comunicazione. Il protocollo di comunicazione deve essere sviluppato *ad hoc* dal costruttore (driver) o dall'utente

I PROTOCOLLI DI COMUNICAZIONE DEL BUS VXI

VME BUS

Protocollo VME

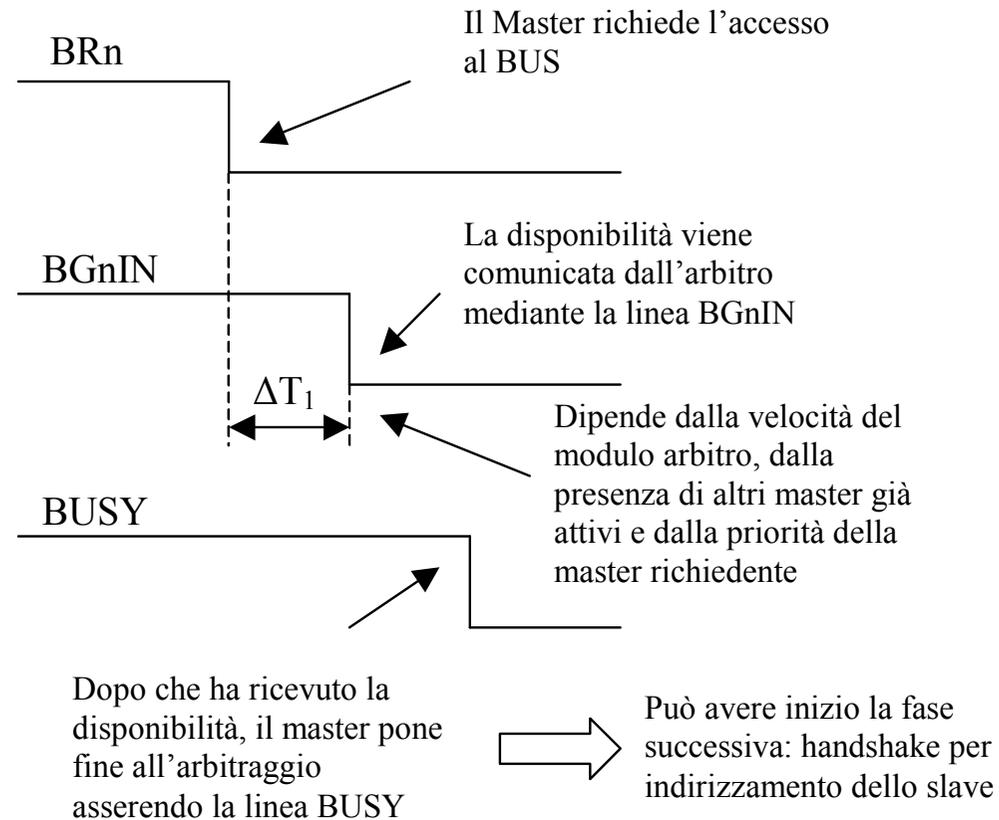
La sequenza delle operazioni è la seguente:

- 1) Arbitraggio nella quale il *master* attraverso una delle linee BR_n richiede all'arbitro l'uso del bus
- 2) Indirizzamento dello *slave*
- 3) Scrittura o lettura

I PROTOCOLLI DI COMUNICAZIONE DEL BUS VXI

VME BUS

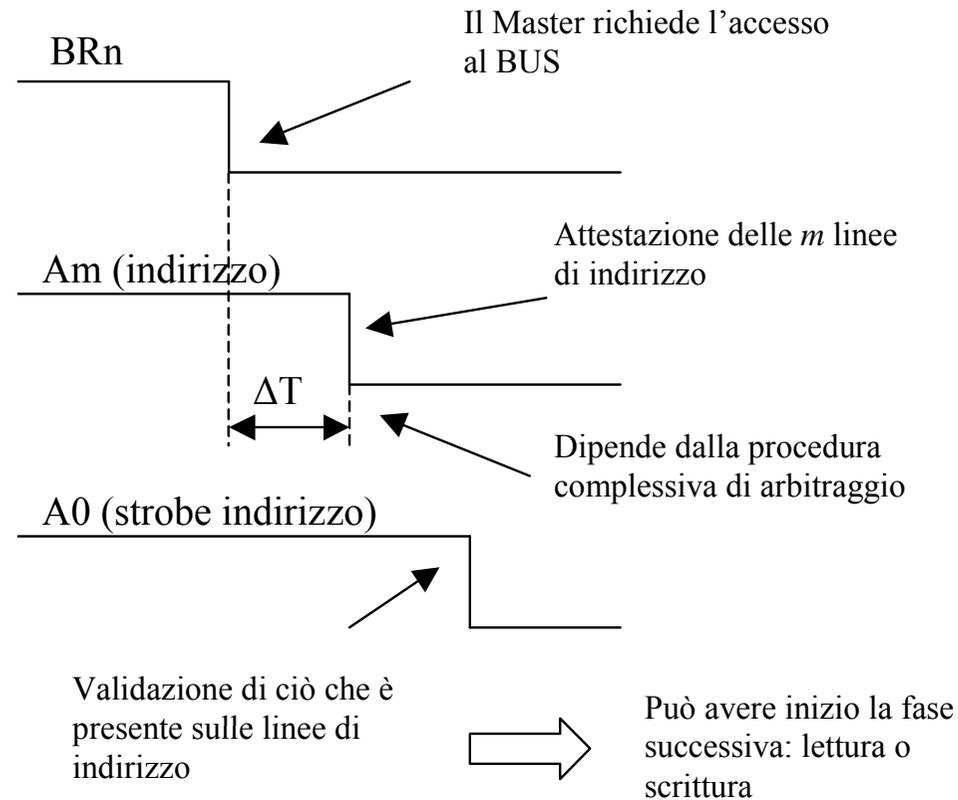
Protocollo VME (arbitraggio)



I PROTOCOLLI DI COMUNICAZIONE DEL BUS VXI

VME BUS

Protocollo VME (indirizzamento dello slave)



I PROTOCOLLI DI COMUNICAZIONE DEL BUS VXI

IN LETTURA

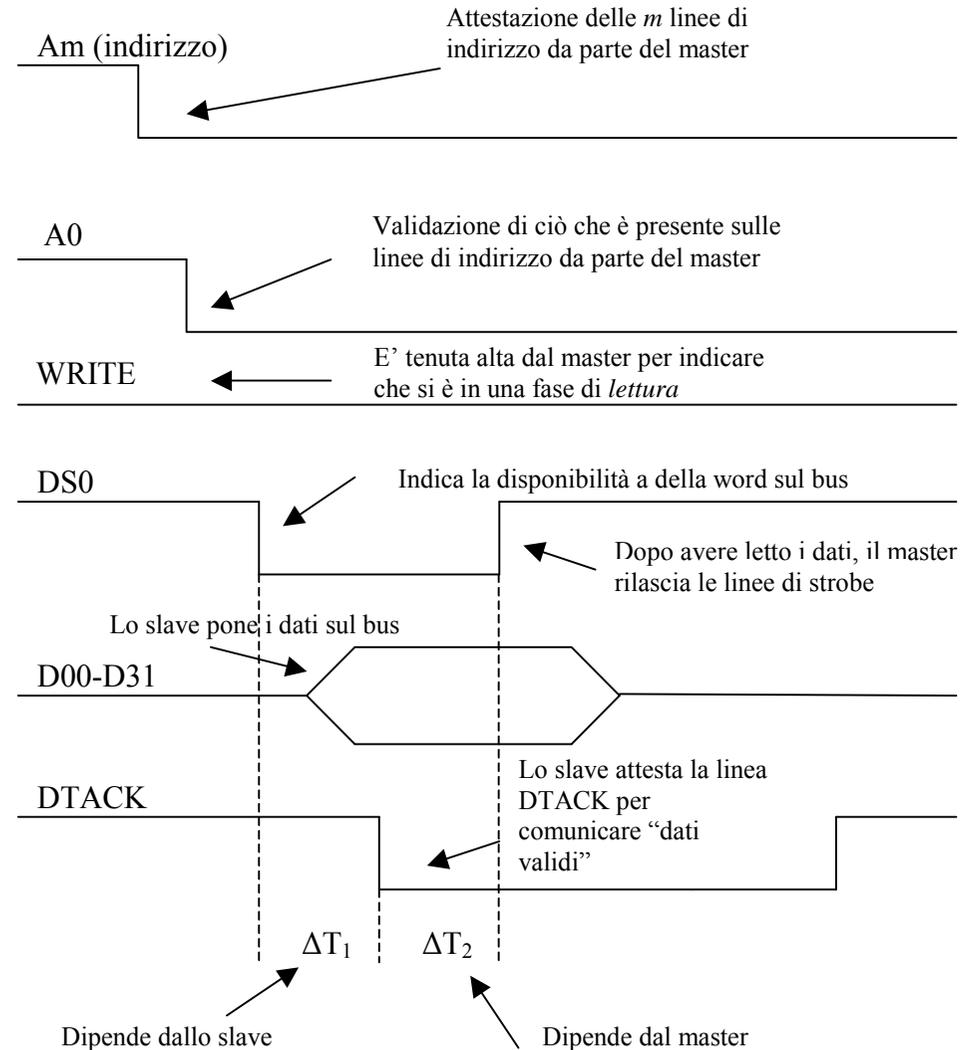
IL MASTER:

- 1) Scrive l'indirizzo sulle linee Am
- 2) Attesta la linea A0
- 3) Con DS0 e DSn indica la disponibilità della word
- 4) Attende che lo SLAVE attesti la DTACK
- 5) Legge la word scritta dallo SLAVE
- 6) Rilascia le linee DSn
- 7) Eventualmente rilascia anche la linea Busy

Lo SLAVE:

- 1) Dopo l'attestazione di A0 riconosce il proprio indirizzo
- 2) Verifica lo stato della linea WRITE
- 3) All'attestazione delle linee DSn scarica sul bus dati il contenuto del registro corrispondente all'indirizzo
- 4) Attesta la linea DTACK
- 5) Rilascia la linea DTACK alla fine

VME BUS



I PROTOCOLLI DI COMUNICAZIONE DEL BUS VXI

IN SCRITTURA

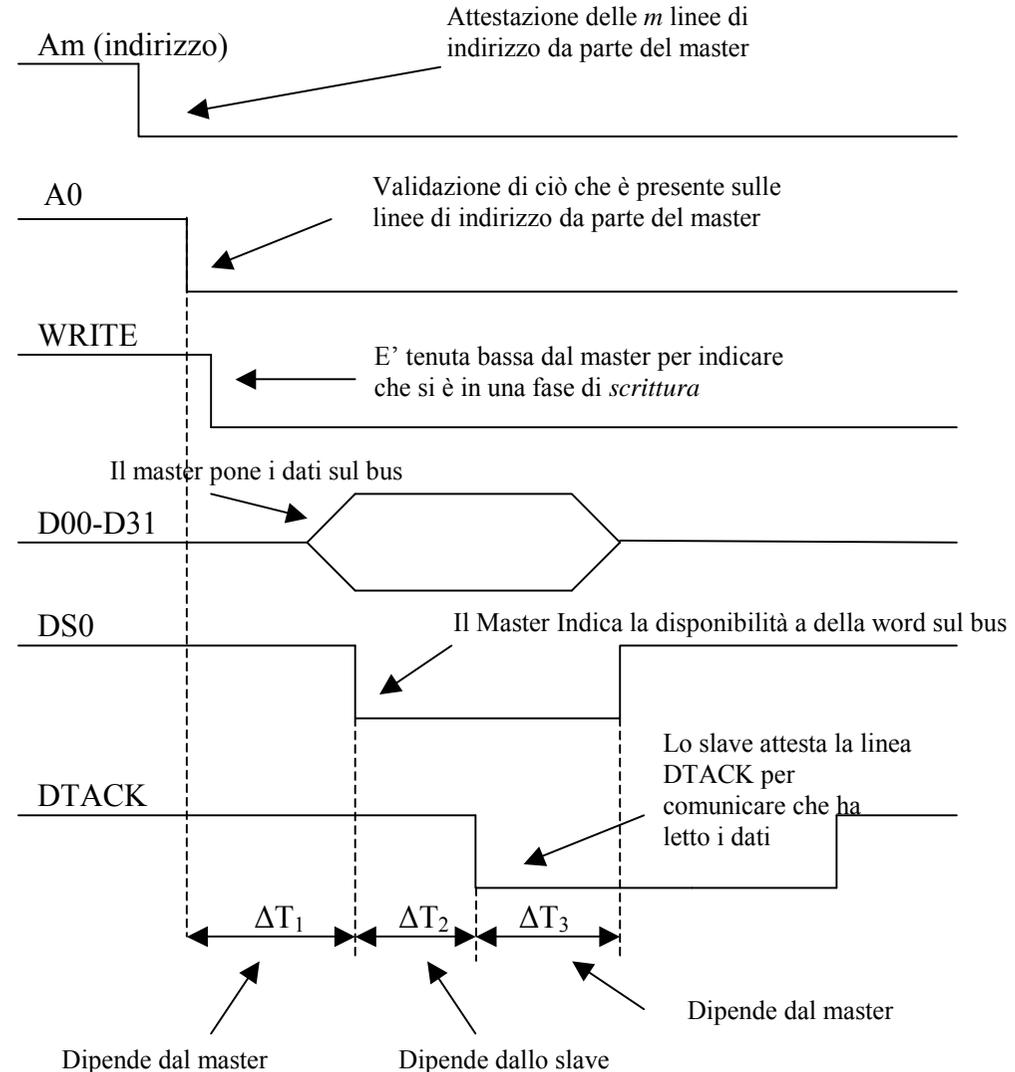
VME BUS

IL MASTER:

- 1) Scrive l'indirizzo sulle linee Am
- 2) Attesta la linea A0
- 3) Attesta la linea WRITE
- 4) Scrive la word sul bus dati
- 5) Con DS0 e DS_n indica la disponibilità della word sul bus
- 6) Attende che lo SLAVE attesti la DTACK
- 7) Rilascia le linee DS_n
- 8) Eventualmente rilascia anche la linea Busy

Lo SLAVE:

- 1) Dopo l'attestazione di A0 riconosce il proprio indirizzo
- 2) Verifica lo stato della linea WRITE
- 3) All'attestazione delle linee DS_n carica la word nel registro corrispondente all'indirizzo
- 4) Attesta la linea DTACK
- 5) Rilascia la linea DTACK alla fine della lettura



I PROTOCOLLI DI COMUNICAZIONE DEL BUS VXI

In un sistema VXI la comunicazione tra dispositivi può svolgersi su tre diversi mezzi di trasmissione:

- 1) *VME Bus*
- 2) *Local Bus*
- 3) *Trigger Bus*

I PROTOCOLLI DI COMUNICAZIONE DEL BUS VXI

IL LOCAL BUS

Protocolli di trasmissione non standard

Rispetto delle prescrizioni VXI in materia di temporizzazione e logica di gestione

Comunicazione tra dispositivi adiacenti

Si può, ad esempio, stabilire che il Local Bus fornisca un bus di dati ad 8 bit, con due linee di sincronizzazione per memorizzare word di dati a 16 bit con un formato byte seriale ad 8bit.

Compatibilità con la comunicazione sul BUS VME

Durante una trasmissione sul bus dati le eventuali capacità di master sul bus VME del dispositivo trasmittente vengono disabilitate dal controllore.

Incremento del throughput del sistema

I moduli che utilizzano il Local Bus liberano il bus VME rendendolo disponibile ad altri dispositivi che abbiano contemporaneamente necessità di comunicare

I PROTOCOLLI DI COMUNICAZIONE DEL BUS VXI

II TRIGGER BUS

Protocolli di trasmissione standard

Lo standard IEEE 1155 prevede 4 protocolli di handshake specifici:

- a) Sync
- b) Semi-sync
- c) Async
- d) Data Trasmission

La Sincronizzazione dei processi di misura

Si possono avere tre soluzioni:

1) Comando digitale (trigger software):

Per i *message-based* è previsto un comando standard di trigger ma comunque tale comando non può essere inviato contemporaneamente a più dispositivi (al contrario del comando GET dell'IEEE 488)

2) Segnale elettrico (trigger analogico):

Attraverso il Trigger Bus si hanno fino a 8 linee utilizzabili per collegare tra di loro i circuiti di trigger degli strumenti

3) Segnale elettrico (trigger analogico):

Attraverso lo Star Bus (solo per i moduli size “D”) vengono inviati segnali di trigger dal controllore a più dispositivi

Gestione concorrente dei processi di misura

La possibilità di avere dispositivi in grado di segnalare al controllore la necessità di comunicare senza che quest'ultimo li interroghi appositamente consente una gestione concorrente dei processi di misura

Infatti è possibile avere uno o più strumenti gestiti autonomamente dai microprocessori dei dispositivi di misura, lasciando il controllore libero di impegnarsi in altri compiti

Per raggiungere tali obiettivi è necessario che gli strumenti siano in grado di generare una richiesta asincrona (interrupt), al verificarsi di eventi prefissati (ad es. disponibilità del risultato)

Il controllore, dopo averne individuato la provenienza “serve” tali richieste seconda una prefissata politica di priorità

Gestione concorrente dei processi di misura

Nel GPIB è presente solo una linea di interruzione (SRQ) e a fronte di un interrupt il controllore esegue la individuazione del dispositivo attraverso una procedura software (polling)

Il VXI prevede un bus (Priority Interrupt Bus) e due meccanismi hardware per l'individuazione della provenienza dell'interrupt

Il Priority Bus si compone di:

7 linee di richiesta, IRQ1-IRQ7 (ogni linea è associata ad una priorità)

1 linea di “interrupt acknowledge” IACK

1 linea gestita in daisy-chain: IACKIN/IACKOUT

Gestione concorrente dei processi di misura

LA GESTIONE DELLE INTERRUZIONI

In un sistema VXI i sottosistemi di interruzione possono essere classificati in due gruppi:

- 1) *single handler systems*: tutte le richieste sono ricevute da un unico gestore e tutte le routine di servizio eseguite da un solo processore
- 2) *distributed handler systems*: le richieste vengono gestite da due o più “interrupt handler”

Oltre al *Priority bus* vengono impegnati anche il *Data Transfer Bus* e l'*Arbitration Bus*

Gestione concorrente dei processi di misura

LA GESTIONE DELLE INTERRUZIONI

Quando un *interrupter* abbassa la linea IRQx alla quale è connesso, l'*interrupt handler* se ne accorge e:

- 1) seleziona, tra le richieste in arrivo contemporaneamente quella a priorità più alta
- 2) ordina al proprio *requester* di richiedere l'accesso al DTB attraverso la linea BRx
- 3) Ottenuta la disponibilità dell'arbitro, l'*interrupt handler* pone sulle ultime tre linee di indirizzo il codice della linea che intende servire, invia l'acknowledge abbassando la IACK e abbassa l'address strobe (AS)

Gestione concorrente dei processi di misura

LA GESTIONE DELLE INTERRUZIONI

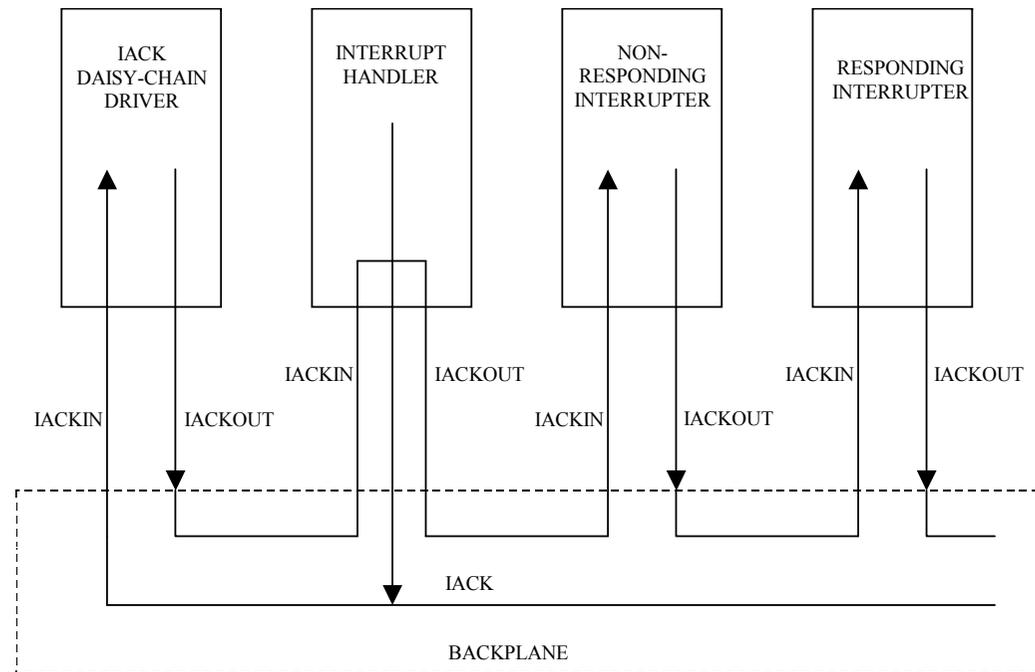
- 4) Completa il ciclo di *interrupt acknowledge* leggendo lo Status/ID Register dell' *interrupter* (quest'ultimo pone il contenuto dello Status Register sul DTB nel momento in cui riceve la IACKIN)
- 5) In base alle informazioni ricavate dallo Status Register, esegue la corrispondente procedura di servizio dell'interruzione

Gestione concorrente dei processi di misura

LA GESTIONE DELLE INTERRUZIONI

La linea IACKIN/IACKOUT assicura che un solo interrupter per volta risponda al ciclo di “interrupt acknowledge”

Infatti: quando l'*interrupt handler* asserisce l'IACK e quindi anche la IACKIN del modulo slot0, quest'ultimo (attraverso la IACKOUT) lo propaga all'*interrupter* più vicino e così via



Gestione concorrente dei processi di misura

LA GESTIONE DELLE INTERRUZIONI

Quando questo *interrupter* rileva il fronte sulla IACKIN sonda le linee A01-A03 per confrontarlo con il valore della linea IRQx che sta pilotando

Se questi valori coincidono l'*interrupter*:

- 1) non trasferisce il segnale alla sua IACKOUT evitando così che altri *interrupter* possano riceverlo
- 2) pone il suo Status/ID Register sulle linee dati e asserisce il DTACK

Appena l'*interrupt handler* sente il DTACK, legge lo Status/ID ed attiva la routine di servizio

