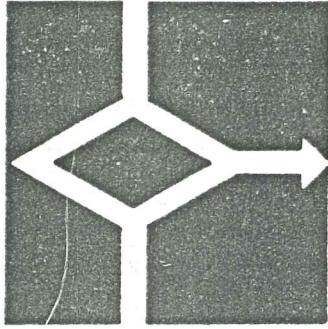


681-09
ST. EL. NF.
BIBLIOTECA
Post ARCHIVIO

1961



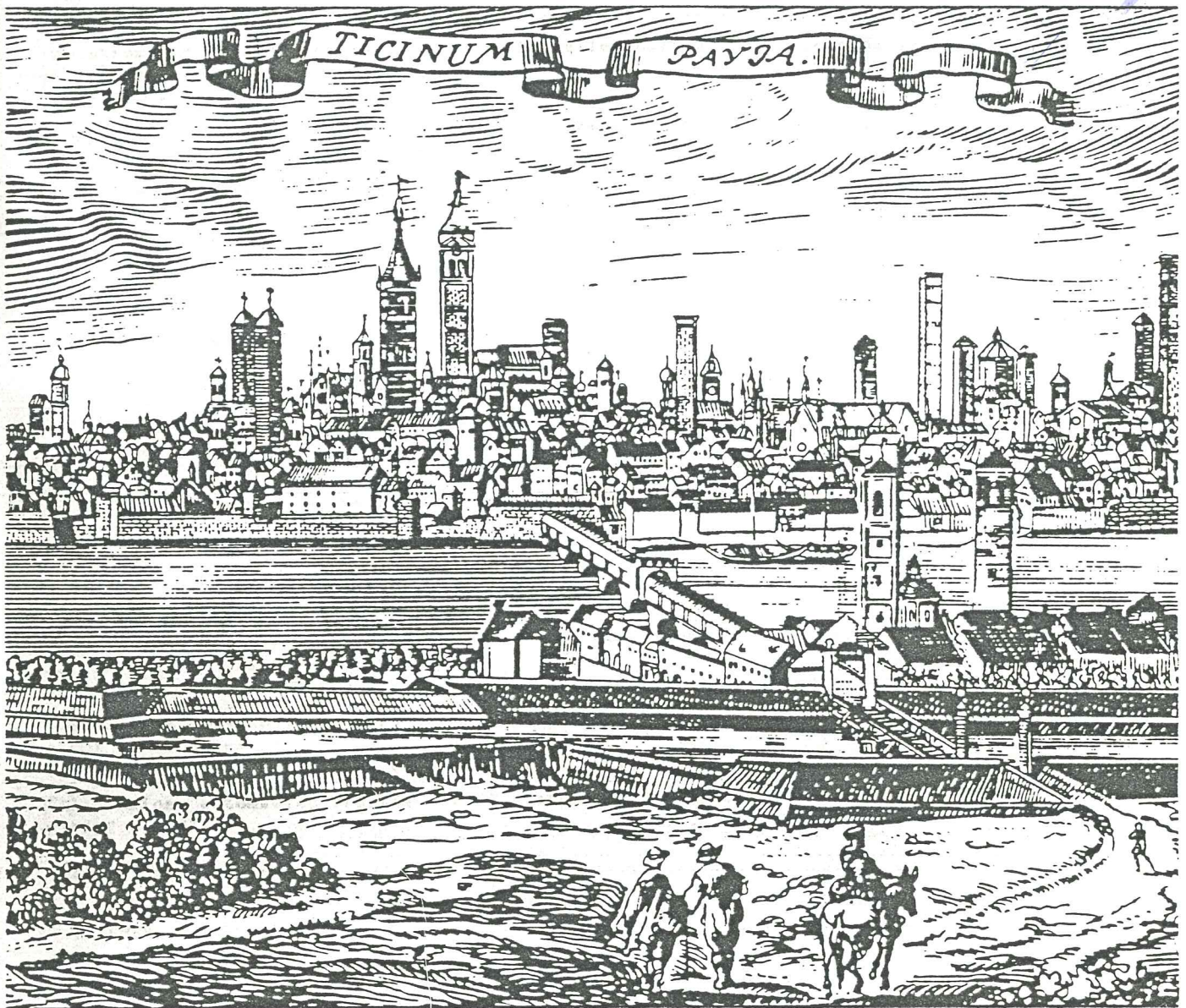
CONGRESSO ANNUALE
ANNUAL CONFERENCE

A.I.C.A. ASSOCIAZIONE
ITALIANA
PER IL CALCOLO
AUTOMATICO

1981

PAVIA 23-25 Settembre 1981

atti
volume II



MuTEAM: UN SISTEMA DISTRIBUITO MULTIMICROPROCESSORE

R. Cardini*, P. Corsini**, G. Frosini***, M. La Manna*, L. Lopriore****, L. Strigini*

* Selenia, Industrie Elettroniche Associate, S.p.A., Roma

** Istituto di Elettronica e Telecomunicazioni, Università di Pisa

*** Istituto di Informatica, Università di Ancona

**** Istituto di Elaborazione dell'Informazione, Consiglio Nazionale delle Ricerche, Pisa

The MuTEAM is a multimicroprocessor system, intended for embedded real-time applications, whose main characteristics are expandability, modularity, error confinement and robustness. In this paper, the system architecture is described in some detail, with particular respect to the hardware supporting the required characteristics of fault tolerance and distributed message-oriented operating system.

1. INTRODUZIONE

MuTEAM è un prototipo di sistema multimicroprocessore /1/, progettato nell'ambito del Progetto Finalizzato Informatica del Consiglio Nazionale delle Ricerche (Sottoprogetto P1, Obiettivo MUMICRO), la cui realizzazione, iniziata da alcuni mesi, dovrebbe concludersi entro il 1981.

Scopo del prototipo è lo sviluppo di metodologie integrate di progetto di sistemi multimicroprocessore distribuiti, per applicazioni in tempo reale. I principali obiettivi sono la modularità, l'espandibilità, il confinamento dell'errore e la robustezza. Tali obiettivi hanno richiesto la definizione di una architettura che unifica la configurazione dell'hardware con gli aspetti di tolleranza ai guasti (descritti in /2/) e di nucleo di sistema operativo (descritti in /3/).

2. CONFIGURAZIONE DEL SISTEMA

Il sistema è formato da un insieme di clusters /4,5,6,7/, connessi in modo lasco mediante linee seriali. Ciascun cluster è costituito da un insieme di Elementi di Elaborazione o nodi (16 al massimo) essenzialmente costituiti da un processore, alcuni banchi di memoria, ed alcuni dispositivi di ingresso e uscita. I nodi di uno stesso cluster sono connessi in modo stretto, mediante un Bus di Cluster, parallelo ad alta velocità, e mediante un Bus di Segnalazione, usato principalmente per la trasmissione di interruzioni tra processori. Ciascun cluster ha un proprio spazio degli indirizzi comuni, cui possono accedere tutti e soli i nodi di quel cluster. Tale spazio è realizzato mediante blocchi di memoria condivisa, un blocco per ogni nodo. Gli accessi ad un blocco di un nodo da parte del processore di un altro nodo avvengono mediante il Bus di Cluster; invece,

Questo lavoro è stato finanziato dal Consiglio Nazionale delle Ricerche nell'ambito del Progetto Finalizzato Informatica.

ogni processore può accedere al blocco contenuto nel suo stesso nodo mediante un Bus Locale. In generale, gli accessi locali sono più veloci, poiché non richiedono l'allocazione del Bus di Cluster, che deve essere conteso con gli altri nodi del cluster.

Oltre allo spazio degli indirizzi comuni, ogni cluster è dotato di uno spazio degli indirizzi privati: tale spazio è anch'esso realizzato mediante blocchi di memoria, un blocco per ogni nodo del cluster, ma ciascuno di tali blocchi è accessibile solo al processore del proprio nodo. Sia lo spazio degli indirizzi comuni che quello degli indirizzi privati sono divisi in segmenti: gli indirizzi segmentati generati dai processori sono rilocati da apposite circuiterie in uno dei due spazi.

Ogni cluster è dotato di un meccanismo di protezione dei segmenti dello spazio degli indirizzi comuni, del tipo a liste di controllo degli accessi /8,9/: a ciascuno di tali segmenti è associata una lista dei processori del cluster aventi diritto ad effettuare operazioni di lettura e/o scrittura. Inoltre, il processore di un nodo può accedere in scrittura ai segmenti contenuti negli altri nodi solo se è nello stato di supervisore; nello stato utente, esso può accedere in scrittura solo ai segmenti contenuti nel suo stesso nodo. Nonostante la sua semplicità, questo meccanismo di protezione assicura il necessario grado di confinamento degli errori, sia hardware che software.

Nel paragrafo seguente, l'architettura di un nodo sarà descritta in dettaglio. Nel paragrafo 4 verrà analizzato il meccanismo di traduzione di un indirizzo generato da un processore nel corrispondente indirizzo fisico in un blocco di memoria del cluster. Nel paragrafo 5 verranno descritti i meccanismi di protezione dei segmenti dello spazio degli indirizzi comuni.

3. ARCHITETTURA DI UN NODO

Ciascun nodo è costituito essenzialmente da (Fig. 1): i) una Unità Centrale di Elaborazione capace

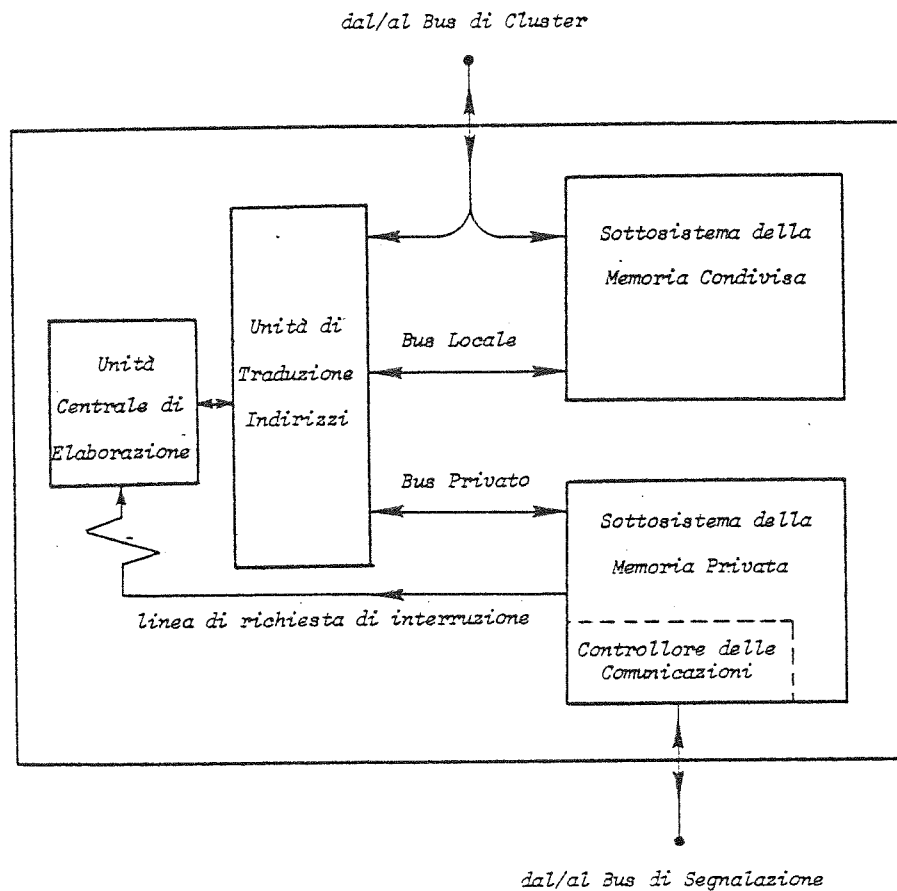


Fig. 1 Architettura di un nodo

di generare indirizzi segmentati (attualmente un microprocessore Zilog Z8000); ii) una Unità di Traduzione degli Indirizzi; iii) un Sottosistema della Memoria Condivisa; iv) un Sottosistema della Memoria Privata; v) un Controllore delle Comunicazioni.

L'Unità di Traduzione degli Indirizzi trasforma gli indirizzi segmentati generati dal processor in indirizzi segmentati rispettivamente nello spazio degli indirizzi comuni o nello spazio degli indirizzi privati. Il Sottosistema della Memoria Condivisa contiene il blocco di memoria condivisa di quel nodo. Il sottosistema è dotato di due porte, controllate da un arbitro locale: una porta è utilizzata per gli accessi al blocco da parte del processore del nodo, e tramite il Bus Locale; l'altra porta è utilizzata per gli accessi al blocco da parte degli altri processori del cluster, e tramite il Bus di Cluster. Il Sottosistema della Memoria Privata contiene il blocco di memoria privata di quel nodo: tale blocco è accessibile solo al processore del nodo, tramite un Bus Privato. Fanno parte di questo sottosistema anche gli eventuali dispositivi di ingresso e uscita, privati del nodo. Il Controllore delle Comunicazioni

è utilizzato per lo scambio di richieste di interruzione con gli altri processori del cluster. Esso è connesso: i) al Bus Privato, per ricevere comandi e scambiare dati con il processore del proprio nodo; ii) al Bus di Segnalazione, per la trasmissione delle richieste di attenzione ai processori degli altri nodi del cluster; iii) al processore del proprio nodo, mediante linee di interruzione.

4. MECCANISMI DI TRADUZIONE DEGLI INDIRIZZI

A ciascun processo attivo è associato uno spazio di processo, composto da 128 segmenti di 64K bytes al massimo. L'indirizzo generato da un processo è costituito da un numero di segmento (7 bit) e da un offset (16 bit): il numero di segmento specifica uno dei 128 segmenti dello spazio di processo, e l'offset specifica un byte nel segmento. A ciascun processore possono essere associati fino a 8 processi attivi. Gli spazi di tutti i processi attivi del cluster (128 al massimo) sono allocati in parte nello spazio degli indirizzi comuni, (composto di 2^{14} segmenti), e in parte nello spazio degli indirizzi privati (anch'esso composto di 2^{14}

segmenti). Tale operazione di allocazione è effettuata in ciascun nodo dalla Unità di Traduzione degli Indirizzi (Fig. 2), composta essenzialmente da: i) 1K Registri di Traduzione di 15 bit ciascuno; ii) due Registri di Processo, di 3 bit ciascuno; e iii) una Logica di Richiesta. I Registri di Traduzione sono suddivisi logicamente in 8 gruppi di 128 registri, un gruppo per ciascuno degli 8 processi attivi associati a quel processore. I due Registri di Processo, detti di Utente e di Sistema, specificano il gruppo attivo rispettivamente quando il processore è nello stato di utente o di sistema: in questo modo, quando l'esecuzione di un processo utente viene sospesa, per eseguire un processo di sistema, il relativo cambio di gruppo è effettuato automaticamente. Inoltre, il passaggio da un processo utente attivo ad un altro può essere effettuato semplicemente scrivendo il numero del gruppo associato al nuovo processo nel Registro di Processo di Utente.

Il numero di segmento, specificato dall'indirizzo

generato dal processore, individua uno dei 128 Registri di Traduzione del gruppo attuale. Ciascuno di tali registri è composto di tre campi P/C, ND, e SEG, rispettivamente di 1, 4 e 10 bit. Il campo P/C specifica se il segmento è allocato nel blocco di memoria privata di quel nodo, o in un blocco di memoria condivisa. Il campo ND è significativo solo se il segmento è allocato in un blocco di memoria condivisa e specifica il nodo che contiene il blocco. Infine, il campo SEG specifica il nome di un segmento nel blocco individuato dai campi P/C e ND.

Una apposita Logica di Richiesta provvede a generare le richieste di accesso al Bus di Cluster e al Sottosistema della Memoria Condivisa. Più esattamente, se il segmento indirizzato è allocato nel blocco di memoria condivisa di un altro nodo, la logica invia una richiesta all'arbitro del Bus di Cluster, e l'accesso al blocco avviene tramite il Bus di Cluster stesso. Se il segmento indirizzato è allocato nel blocco di memoria condivisa di quel

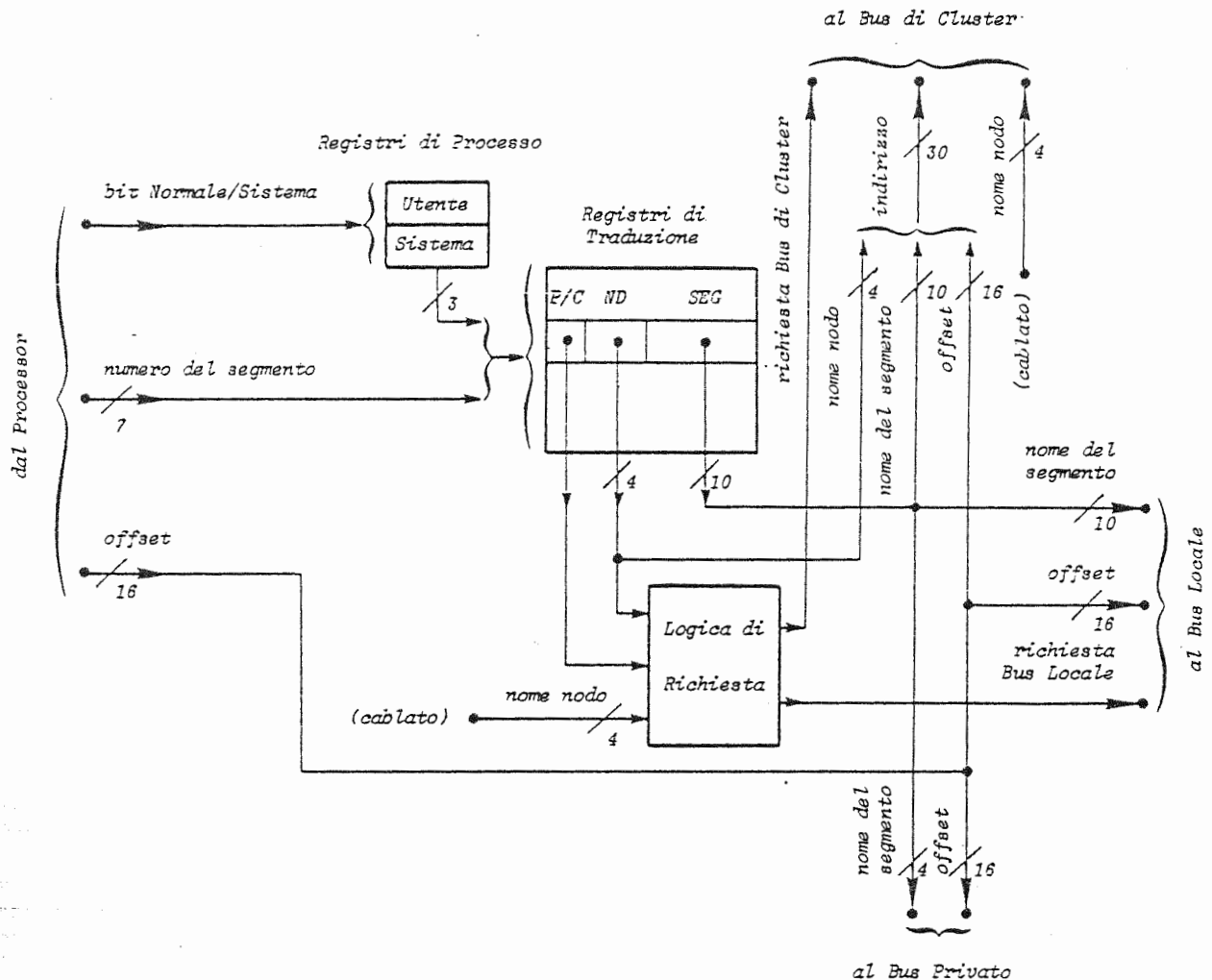


Fig. 2 Schema a blocchi della Unità di Traduzione degli Indirizzi

nodo, la logica invia una richiesta ad un arbitro locale, contenuto nel Sottosistema della Memoria Condivisa, che risolve le eventuali contese con richieste di accesso al sottosistema proveniente dagli altri nodi tramite il Bus di Cluster. Se invece il segmento è allocato nel blocco di memoria privata del nodo, viene effettuato un accesso al Sottosistema della Memoria Privata tramite il Bus Privato.

Il Sottosistema della Memoria Condivisa riceve un indirizzo nello spazio degli indirizzi comuni composto dal nome di un segmento (10 bit) ed un offset nel segmento (16 bit). Il sottosistema contiene una Unità di Rilocalizzazione (Fig. 3), che traduce tale indirizzo in un indirizzo fisico nel blocco di memoria del sottosistema stesso. L'unità è essenzialmente costituita da un insieme di 1K Registri di Rilocalizzazione, da 16 bit: il nome del segmento individua un registro, il cui contenuto specifica una quantità che moltiplicata per 16 costituisce la base del segmento nel blocco. A tale base viene sommato l'offset per ottenere l'indirizzo fisico nel blocco.

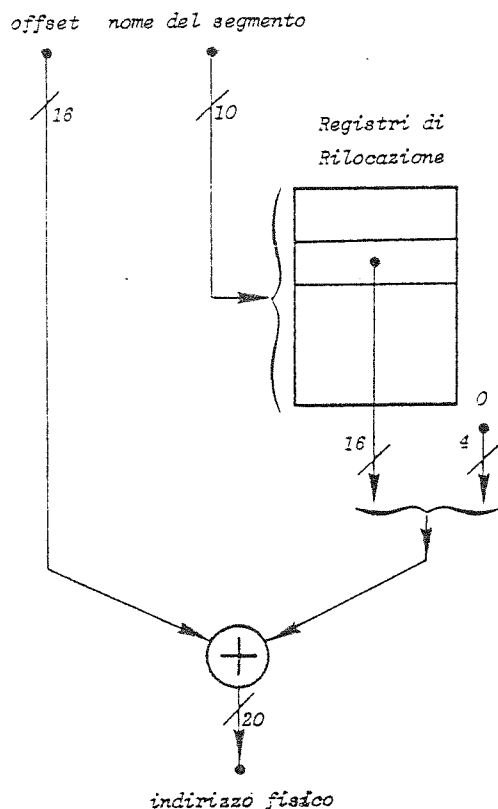


Fig. 3 Schema a blocchi della Unità di Rilocalizzazione

Una simile Unità di Rilocalizzazione è contenuta anche nel Sottosistema della Memoria Privata, e traduce un indirizzo nello spazio degli indirizzi privati in un indirizzo fisico nel blocco di memoria privata.

Il contenuto dei registri interni alla Unità di Traduzione degli Indirizzi ed alla Unità di Rilocalizzazione di un certo nodo può essere modificato dal solo processore di quel nodo, che li indirizza nel proprio spazio di I/O Speciale. Poiché le istruzioni che operano in tale spazio sono privilegiate, il contenuto di tali registri non può essere alterato da processi utente.

5. MECCANISMI DI PROTEZIONE

Il Sottosistema della Memoria Condivisa contiene anche una Unità di Protezione (Fig. 4), che controlla gli accessi al blocco di memoria condivisa. Tale unità contiene un insieme di 1K Registri di Protezione, di 48 bit ciascuno. Il nome del segmento indirizzato individua uno di questi registri, il cui contenuto è suddiviso in tre campi L, R e W, di 16 bit ciascuno. Il campo L specifica la dimensione del segmento, espressa in bytes. I campi R e W specificano i processori nel cluster aventi diritto ad effettuare rispettivamente operazioni di lettura e scrittura nel segmento. Più precisamente, se lo i -esimo bit del campo R è ad 1, lo i -esimo processore del Cluster può effettuare operazioni di lettura nel segmento, $i=0,1,\dots,15$; similmente, se il j -esimo bit del campo W è ad 1, il j -esimo processore del cluster può effettuare operazioni di scrittura nel segmento, $j=0,1,\dots,15$.

Una apposita circuiteria genera un segnale di violazione di protezione se un processore del cluster tenta un accesso illegittimo ad un segmento, o se l'indirizzo contiene un offset maggiore della lunghezza del segmento. Il segnale di violazione viene generato anche se viene effettuato un tentativo di scrittura da parte del processore di un altro nodo, e questo processore non si trova nello stato supervisore. In conseguenza ad una violazione: i) viene impedito l'accesso al banco di memoria; ii) viene registrata in un apposito Registro di Stato la causa della violazione, e cioè il nome del processore che ha tentato l'accesso ed il nome del segmento indirizzato; iii) viene inviata una richiesta di interruzione al processore del nodo. I registri della Unità di Protezione sono anch'essi indirizzati nello spazio di I/O Speciale del processore del nodo: il driver della interruzione potrà pertanto leggere il nome del processore che ha causato la violazione nel Registro di Stato. Se tale processore appartiene ad un nodo diverso, il driver invierà a tale processore un messaggio di notifica.

Il Sottosistema della Memoria Privata contiene anch'esso una Unità di Protezione, essenzialmente costituita da 1K registri di 16 bit ciascuno. Ciascun registro specifica la dimensione di un segmento, espressa in bytes. Una apposita circuiteria genera un segnale di violazione di protezione se l'indirizzo specifica un offset maggiore della lunghezza del segmento interessato.

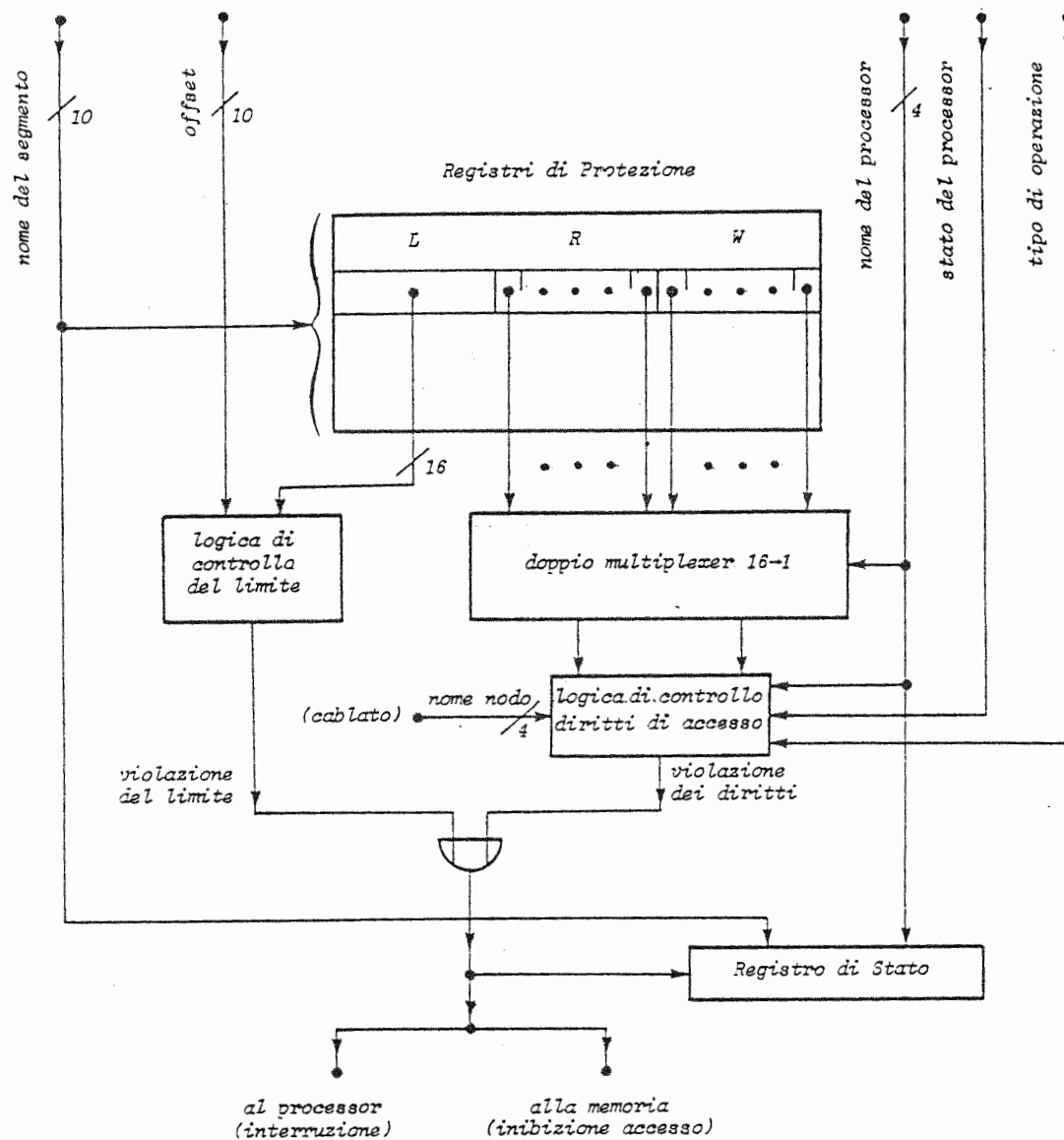


Fig. 4 Schema a blocchi della Unità di Protezione

RINGRAZIAMENTI

Molte persone hanno partecipato alla progettazione e partecipano alla realizzazione di MuTEAM, tra le quali: G. Cioffi e P. Velardi, Istituto di Automatica, Università di Roma; A. Orlandi, Oto Melara, S.p.A., La Spezia; A. Ciuffoletti, Selenia, S.p.A., Roma; e gli autori dei lavori /2/ e /3/.

RIFERIMENTI

/1/ "MuTEAM: Presentazione Preliminare", Rapporto MUMICRO, Progetto Finalizzato Informatica, Consiglio Nazionale delle Ricerche, 1980.
 /2/ P. Ciompi, F. Grandoni, L. Simoncini, "Procedure di Diagnosi Automatica nel Sistema MuTEAM", Atti Congresso Annuale AICA 1981, AICA, Pavia, 1981.

/3/ F. Baiardi, A. Fantechi, A. Tomasi, M. Vanne-schi, "Meccanismi di Fault-tolerance nella Macchi-na Virtuale Intermedia del Sistema MuTEAM", Atti Congresso Annuale AICA 1981, AICA, Pavia, 1981.

/4/ R.J. Swan, S.H. Fuller, D.P. Siewiorek, "Cm* - A Modular, Multimicroprocessor", Proc. 1977 AFIPS National Computer Conference, Vol. 46, pp. 637-644.

/5/ J. Archer Harris, D.R. Smith, "Hierarchical Multiprocessor Organizations", Proc. 4th Annual Symposium on Computer Architecture, IEEE/ACM, 1977, pp. 41-48.

/6/ M. Ajmone Marsan, G. Conte, D. Del Corso, F. Gregoretti, "Architecture, Communication Proce-dures and Performance Evaluation of the μ^* Multi-microprocessor System", Proc. 1st International Conference on Distributed Computing Systems,

Huntsville, Alabama, 1979.

/7/ G. Mazaré, "MCS - A Symmetric Multi-micro-processor System", Proc. 2nd EUROMICRO Symposium on Microprocessing and Microprogramming, Venice, 1976, pp. 135-140.

/8/ P.J. Denning, "Fault Tolerant Operating Systems", Computing Surveys, Vol. 8, No. 4, pp. 359-389.

/9/ J.H. Saltzer, M.D. Schroeder, "The Protection of Information in Computer Systems", Proc. IEEE, Vol. 63, No. 9, pp. 1278-1308.