

Presentata alle
riunioni su
"I minielaboratori
dell'INFN",
Frascati, Ottobre 1975 e
Legnaro, Gennaio 1976

ISTITUTO NAZIONALE DI FISICA NUCLEARE
Laboratori Nazionali di Frascati

LNF-76/57(P)
Novembre 1976

G. Bologna: Sviluppi del sistema CAMAC. -

SVILUPPI DEL SISTEMA CAMAC

(Relazione sul "2nd International Symposium on Camac, Bruxelles 1975")

G. Bologna

Laboratori Nazionali di Frascati

1. - INTRODUZIONE

Si è svolto a Bruxelles dal 14 al 16 ottobre 1975 il II^o Simposio Internazionale sul CAMAC, articolato in 6 sessioni. Le sessioni I, III, V, erano plenarie e riguardavano:

- I - Stato del CAMAC nei vari paesi.
- III - Rassegna delle applicazioni del CAMAC nell'industria, nel laboratorio e in medicina.
- V - Rassegna delle applicazioni del CAMAC per la comunicazione dati, per i pubblici servizi e per il controllo ambientale.

Le altre sessioni erano parallele ed accoglievano comunicazioni singole, come di seguito:

- II. 1. - Introduzione al CAMAC per non specialisti
- II. 2. - Nuovi sviluppi del CAMAC rispetto al primo Simposio (Lussemburgo 4-6/12/1973)
- IV. 1. - Controlli di processi industriali
- IV. 2. - Automazione del laboratorio (1^a parte)
- IV. 3. - Servizi di medicina e sanità.
- VI. 1. - Controllo ambientale, pubblici servizi, e miscellanea.
- VI. 2. - Automazione del laboratorio (2^a parte)
- VI. 3. - Comunicazione dati e controllo degli acceleratori

- Associata al Simposio vi era anche una estesa mostra di strumentazione alla quale hanno partecipato i più importanti costruttori ed anche alcuni enti scientifici, con la più recente produzione CAMAC.

Dalla panoramica generale della Sessione I^a, e dalle discussioni seguitene, così come dalla mostra, è risultato che il sistema CAMAC è proiettato ormai a svilupparsi anche al di fuori del campo della fisica nucleare, in seno al quale è nato. Il comitato ESONE (che come è noto, è il promotore del CAMAC), in stretto contatto con il comitato NIM USAEC ed anche con la CFI (Commissione Elettrotecnica Internazionale) sta gettando le basi per un adeguamento delle norme CAMAC alle necessità del controllo dei processi industriali, pur sempre rispettando la compatibilità con le specifiche attuali.

In conclusione il CAMAC ha tutti i requisiti per potersi affermare internazionalmente come unico sistema normalizzato per acquisizione digitale di dati, in tempo reale.

Non è possibile in questa sede fare una rassegna esauriente di tutti i lavori presentati al Simposio. Ci si limiterà ad illustrare i punti più interessanti emersi. A giudizio dello scrivente essi sono stati:

- a) - Sviluppo di sistemi CAMAC dotati di intelligenza autonoma distribuita, mediante l'uso di microprocessori incorporati nei vari moduli CAMAC.
- b) - Sviluppo del sistema CAMAC dotato della highway seriale; esso è venuto ad affiancare il tradizionale sistema con branch highway parallela.
- c) - Sviluppo di software CAMAC orientato. Particolarmente significativa è stata la implementazione del macro-linguaggio IML per il PDP11.

2. - USO DEI MICROPROCESSORI NEL CAMAC

Un microprocessore è funzionalmente simile all'unità centrale di elaborazione (CPU) di un minicalcolatore. Mediante l'uso della tecnologia LSI (integrazione su larga scala), un microprocessore con funzioni equivalenti a quelle di parecchie migliaia di transistor, realizzato con processo

MOS, occupa uno spazio estremamente ridotto, potendo essere contenuto in pochi packages di circuiti integrati (tipicamente da 1 a 4). Essi hanno il vantaggio di poter essere direttamente interfacciati al dataway CAMAC VI. Ovviamente, per effettuare tutte le operazioni di controllo, di ingresso-uscita e di elaborazione richieste dal programma di utilizzazione, sono necessari altri circuiti integrati ausiliari e circuiti integrati contenenti componenti di memoria. Globalmente però essi possono essere tutti alloggiati su un'unica scheda stampata CAMAC.

Il complesso di questi circuiti potrà essere chiamato microcalcolatore.

Il costo totale è molto contenuto ($\approx 10^6$ L). Per questa ragione a Bruxelles essi hanno incontrato molto favore. Tipicamente il CAMAC era in precedenza concepito come un sistema asservito ad un calcolatore esterno. In qualche caso era stata introdotta una sorta di autonomia mediante la generazione di comandi e la elaborazione di dati. A questo scopo erano stati costruiti moduli CAMAC basati su componenti SSI od MSI, di solito non competitivi dal punto di vista economico. L'avvento di microprocessori LSI ha invece aperto al CAMAC una nuova strada e già prima del Simposio di Bruxelles se ne conoscevano applicazioni. Però è stato a Bruxelles (dove sono state presentate otto applicazioni diverse) che i microprocessori nel CAMAC hanno incontrato improvvisamente un grosso favore. La più ovvia applicazione dei microprocessori è nei crate controllers. Si possono così ottenere soluzioni estremamente economiche per sistemi CAMAC autonomi, con un solo crate e senza un minicalcolatore separato (essendo il "microcalcolatore" incluso nel crate stesso).

In modo analogo il microprocessore può essere associato ad un branch driver, nel qual caso si può ancora fare a meno di un calcolatore separato. Oppure quest'ultimo può esistere, ma con funzioni di supervisore in sistemi di notevole complessità.

Al limite la distribuzione dell'intelligenza può essere spinta al livello dei singoli moduli, utilizzando il microprocessore come pre-elaboratore al fine di inviare al dataway dati selezionati e diminuire quindi il traffico verso i crate controllers.

Questi microprocessori, costruiti con processo MOS, hanno anche degli inconvenienti: essi sono relativamente lenti (clock rate: 2 Mhz tipico); ed hanno la parola relativamente corta (tipicamente 8 bits). Tuttavia la tecnica di costruzione è in rapida evoluzione. Quelli presentati a Bruxelles erano molto migliori di quelli di prima generazione (1973). Già sono annunciati ulteriori miglioramenti. In quei casi limite in cui occorra una velocità maggiore non è possibile usare componenti MOS. Ad esempio il microprocessore rapido GESPRO (clock rate 10 Mhz) è stato costruito da J. M. Meyer (CRN Strasbourg) con circuiti integrati bipolari della serie INTEL 3000, aventi costo ed occupazione di spazio maggiore del MOS.

Anche al CERN (M. Collins, G. Guillaume, R. Rausch) è stato costruito un microprocessore rapido chiamato AFC (Autonomons Function controller).

3. - IL MICROPROCESSORE INTEL 8080.

Prima di passare in rassegna le varie realizzazioni è opportuno ricordare brevemente le principali caratteristiche di un tipico microprocessore LSI, l'INTEL 8080, che a Bruxelles è stato presentato in quattro applicazioni diverse. Il microprocessore INTEL 8080 è un circuito integrato n-MOS posto in un unico contenitore a 40 piedini.

Esso contiene una unità centrale avente le seguenti caratteristiche:

- indirizzamento della memoria su 16 bits (64 K parole max indirizzabili)
- modalità di indirizzamento: diretto; registro; registro indiretto; immediato,
- lunghezza della parola: 8 bits,
- 1 unità aritmetico - logica (ALU) per l'effettuazione di operazioni logiche e in aritmetica binaria e decimale,
- 9 registri accessibili a programma di cui 1 accumulatore da 8 bits per le operazioni aritmetico-logiche, 3 coppie di registri da 8 bits per uso generale, e 2 registri da 16 bits per controllo memoria (contatore di programma e puntatore di catasta) . -

- Durata del ciclo base . 0,5 μ sec. Durata delle istruzioni: min. 2,0 μ sec; max 8,5 μ sec.
- 74 istruzioni così classificabili: operazioni su registri e di riferimento alla memoria; salto condizionato ed incondizionato e chiamata di subroutine; salvataggio e ripristino dello stato; operazioni in doppia lunghezza sui registri; operazioni logiche ; aritmetica binaria e decimale; ingresso uscita; mascheramento delle interruzioni.
- 8 livelli di interruzione con salto automatico a subroutine
- 256 accessi possibili per l'ingresso-uscita
- possibilità di accesso diretto alla memoria.

L'architettura generale del sistema è mostrata in fig. 1. -

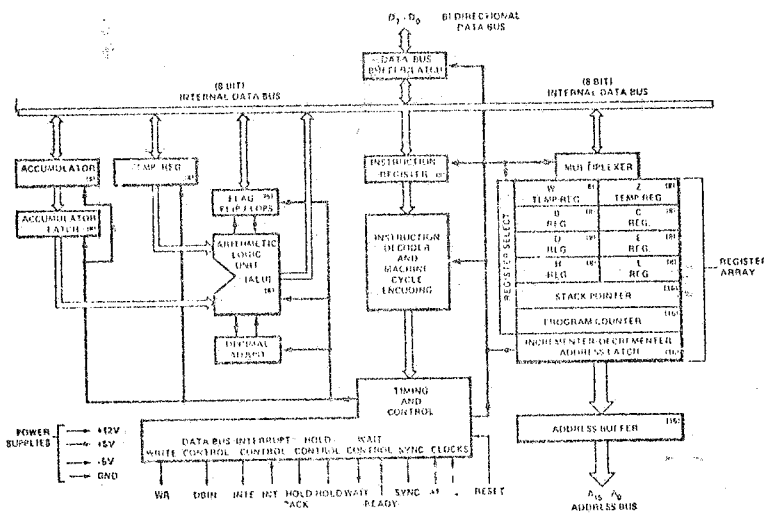


Fig. 1 - Struttura del microprocessore INTEL 8080.

I circuiti di temporizzazione del microprocessore richiedono il collegamento di un altro circuito integrato per la generazione di due cronimpulsi. Altri circuiti integrati devono essere connessi per realizzare la memoria (a semiconduttore). Questa è essenzialmente di due tipi ad accesso casuale (RAM) e di sola lettura (ROM). La RAM serve per memorizzare dati e programmi in via di sviluppo e quindi soggetti a variazioni, mentre la ROM serve per programmi codificati in via definitiva (tipicamente: la parte del sistema operativo residente in memoria). Per questo scopo si è molto affermata un tipo di ROM (fino a 8k bits) programmabile anche dallo stesso utilizzatore mediante semplici circuiti elettronici eventualmente collocati in un modulo CAMAC. Questo tipo di ROM si può cancellare mediante esposizione alla radiazione ultravioletta e quindi riprogrammare (REPROM).

Fra gli altri circuiti integrati disponibili da collegare al microprocessore menzioniamo un ricevitore trasmettitore sincrono ed asincrono universale (USART) a 28 piedini mediante il quale è possibile comunicare serialmente con dispositivi esterni (telescriventi, video- display, modem, etc.) ad una velocità variabile fra 110 e 9600 baud (= bit/sec.). Esiste anche un circuito integrato a 40 piedini, mediante il quale si può programmare l'ingresso /uscita di dati in modo parallelo.

Esso è sufficientemente di uso generale, per poter interfacciare le ordinarie periferiche (lettori-perforatori di nastro, stampanti veloci, unità a dischi magnetici, etc.) senza bisogno di ulteriori circuiti logici.

Infine è molto importante agli effetti della acquisizione dati in tempo reale la flessibilità del sistema di interruzione, per il quale la INTEL offre due circuiti integrati a 24 piedini realizzanti un registro con logica associata ed una unità di controllo e codifica di priorità ad 8 livelli. Il servizio delle interruzioni multiple viene molto facilitato con l'uso congiunto della cascata accessibile sulla base "dell'ultimo entrato-primo uscito" (LIFO-Last In-First out). Questo concetto era già molto usato nei minicalcolatori PDP 11. La INTEL ha anche annunciato il lancio sul mercato, entro breve tempo, di un circuito integrato a 40 piedini per il controllo programmabile dell'accesso diretto in memoria, mediante il quale si renderà possibile un trasferimento dal CAMAC alla memoria che non farà intervenire il microprocessore. Si potranno avere così velocità di trasferimento molto alte (prossime a 10^6 parole/sec.) come nei ben più costosi sistemi attuali. Altrettanto importante dal punto di vista dell'utilizzatore è il software di sistema esistente. L'INTEL offre i seguenti componenti di base: Monitor del sistema, Assemblatore editore di testo, e biblioteca di programmi matematici e di conversione.

Il monitor del sistema può risiedere stabilmente in una ROM o REEPROM ed occupa 2k parole. Le sue funzioni di controllo sono:

- 1) - Caricamento della RAM da astiera o da lettore di nastri di carta;
- 2) - invio in esecuzione del programma caricato;
- 3) - modifica del contenuto della RAM
- 4) - perforazione e stampa del contenuto della memoria.

L'assemblatore lavora in tre passi ed occupa 8k parole (compresa la tavola dei simboli). E' bene che risieda in una ROM, per evitare di doverlo continuamente ricaricare. Esso traduce il programma sorgente (scritto in codice mnemonico) in codice binario rilocabile e produce un nastro perforato ricaricabile per mezzo del monitor. L'editore di testo viene caricato da nastro di carta e serve per sviluppare in linea i vari programmi sorgente.

Poiché le possibilità di sviluppo in linea del software sono abbastanza limitate con sistemi di dimensioni così modeste, la INTEL fornisce anche un compilatore PL/M, un assemblatore incrociato ed un simulatore, scritti in FORTRAN IV e destinati a lavorare su un calcolatore di grandi dimensioni. Il PL/M ha le caratteristiche di un linguaggio ad alto livello (come il FORTRAN) senza sacrificare l'efficienza del linguaggio assemblatore, ma con notevole risparmio di tempo di programmazione. Il simulatore fornisce una simulazione software del microprocessore 8080, così come comandi per il controllo dell'esecuzione, al fine di aiutare nello sviluppo programmi fuori linea.

Come si può notare nelle sue dimensioni ridottissime, un microcalcolatore come quello de-

scritto ha prestazioni simili a quelle di un minicalcolatore, eccettuato per i più lunghi tempi di esecuzione.

La logica del microprocessore è molto simile a quella CAMAC (tranne che la parola dello INTEL 8080 ha 8 bits, mentre la parola CAMAC è lunga 16 o 24 bits, per cui normalmente sono necessari rispettivamente 2 o 3 trasferimenti per accedere ai dati CAMAC).

L'interfaccia fra il CAMAC ed il microprocessore è dunque molto semplice e gli indirizzi e sottoindirizzi CAMAC sono accessibili come locazioni di memoria (questo approccio deriva ancora dall'architettura del PDP11).

4. - APPLICAZIONI DELL'INTEL 8080

A Bruxelles sono stati presentati quattro crate controllers autonomi, incorporanti un microprocessore INTEL 8080, tutti abbastanza simili fra loro. Di questi, uno era un prodotto commerciale già noto e cioè il MK-X della Standard Engineering corporation, U.S.A., del quale ricordiamo solo che può essere utilizzato anche come crate controller ausiliario in associazione con un crate controller tipo A (CCA) o un crate controller seriale tipo L1. In questo caso esso occupa tre stazioni CAMAC normali ed è collegato al crate controller principale mediante linee a parte. Se quest'ultimo è un CCA, vi si devono apportare alcune modifiche rispetto alle specifiche ESONE, per consentire il collegamento citato e per generare alcuni segnali di colloquio che permettano un arbitraggio di priorità fra i due controllers (quello ausiliario deve poter cedere il controllo ogni qual volta il principale, avente più alta priorità lo richieda). L'obiettivo di questo approccio è di destinare al crate controller ausiliario la gestione decentrata di un certo numero di segnali LAM provenienti dai moduli CAMAC, consentendo così al driver generale del sistema una gestione alleggerita delle richieste a livello più alto di priorità.

Un secondo prodotto era il risultato della collaborazione fra il KFA Juelich Germania (relazione di R. Conway ed H. Halling) e la ditta Borer Electronics, Svizzera (Relazione di H. Lieben doerfer). Il modulo in questione era il Borer MACAMAC controller. Esso occupa solo due unità CAMAC, ma ha struttura simile al precedente (anch'esso può funzionare come controller ausiliario).

Un terzo crate controller, il CMC 8080, è stato presentato da E. Schoeberl (SGAE, Austria). Lo schema a blocchi del sistema è presentato in fig. 2. Il CMC 8080 di per se stesso occupa 2 unità CAMAC ed include 2K parole (8 bits) RAM ed 2K parole REEPROM. Il sistema è espandibile fino a 60k con moduli CAMAC aggiuntivi collegati ad esso con un sistema di linee a parte. Le 4K parole di memoria di indirizzo più alto (60-64K) non sono utilizzabili perchè i relativi indirizzi sono usati per i vari indirizzi e sottoindirizzi CAMAC (cfr. fig. 2). Come esempio diciamo che il tempo di un'operazione di scrittura CAMAC (parola di 24 bits) è di 35 μ sec. (piuttosto lungo in verità). Il CMC 8080 comprende l'interfaccia seriale (USART) mentre l'interfaccia per l'accesso parallelo diretto dalla memoria di altro sistema (DMA) è realizzato su un modulo CAMAC ad esso collegato con linee a parte.

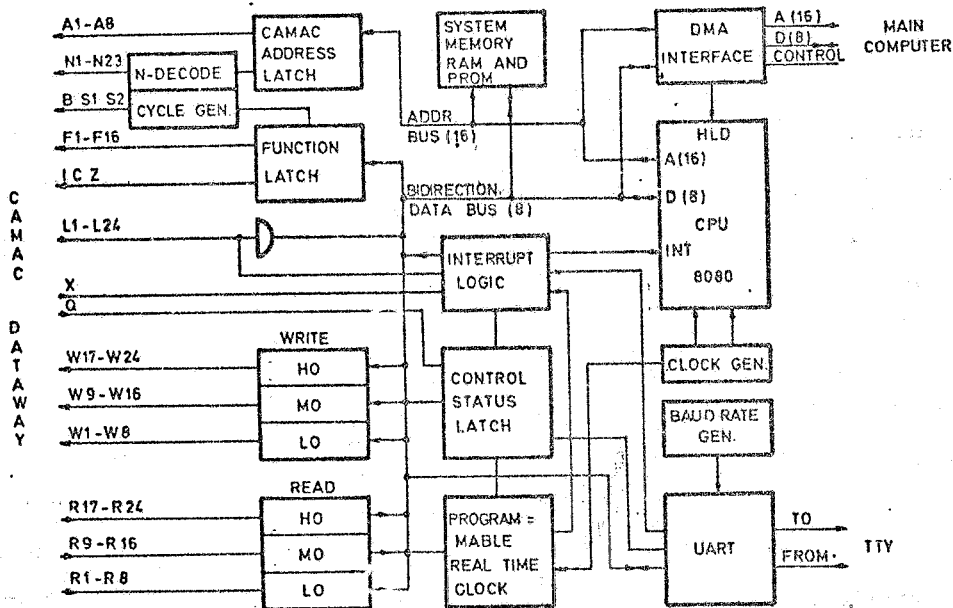


Fig. 2 - Schema a blocchi del CMC 8080.

La logica del sistema di interruzione sfrutta la potenzialità dell'INTEL 8080. Al codificatore di priorità sono collegabili 6 segnali LAM distinti e l'OR di tutti gli altri LAM. All'ultimo ingresso (quello a più alta priorità) è collegato l'OR del segnale X (comando CAMAC rifiutato) e della segnalazione proveniente dalla USART. Diverse possono essere le applicazioni di questo sistema. La più semplice è quella di sistema autonomo e richiede solo il CMC 8080 collegato serialmente (mediante la USART) ad una periferica esterna (telescrivente etc.). E' possibile realizzare anche sistemi più complessi costituiti dal CMC 8080 e da un'altro calcolatore mediante una interconnessione seriale (ad esempio, con dei modems); od in parallelo, per mezzo del canale di DMA (cfr. 2), quando siano necessarie alte velocità di trasferimento da e verso la RAM del CMC 8080.

Più interessante del CMC 8080 si presenta il quarto crate controller autonomo, presentato da J. Kaiser, CEN Saclay, Francia). Esso si chiama J. CAM-10 ed è frutto di una collaborazione fra il CEN di Saclay e la ditta Schlumberger di Parigi, che lo produce commercialmente. Esso si occupa unità CAMAC ed è stato progettato per consentire la massima velocità di scambio fra la memoria ed i moduli CAMAC.

Oltre il microprocessore 8080, esso comprende 4K parole REPROM, 2k parole RAM (8 bits) ed 1K parole RAM da 24 bits per il trasferimento di dati secondo il formato CAMAC. Inoltre esso comprende i circuiti di collegamento al dataway CAMAC, il codificatore di priorità delle interruzioni e l'interfaccia seriale per la telescrivente. Lo schema a blocchi è presentato in fig. 3.

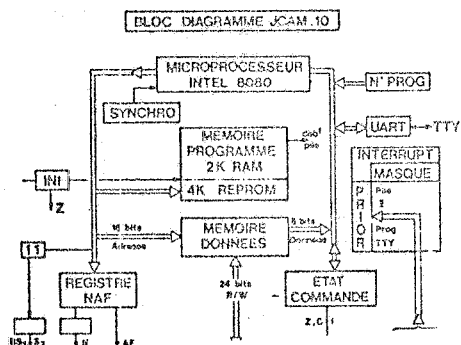


Fig. 3 - Schema a blocchi del JCAM10.

Il sistema è espandibile, con moduli di memoria ausiliaria, fino a 48K parole (8bits). I 16K indirizzi più alti sono utilizzati per trasmettere nei primi 14 bits la funzione CAMAC F, l'indirizzo del modulo N ed il sottoindirizzo A, essendo i due bits più significativi forzati veri. In questo modo si rinuncia a 16K di memoria ma si guadagna tempo, risparmiando un'istruzione rispetto ad inviare separatamente F ed N,A. Ma il risparmio più grosso di tempo lo si ottiene organizzando una parte della memoria in doppio formato. Quest'area agisce come 1K da 24 bits, essendo connessa direttamente al dataway del CAMAC, oppure come 3K da 8 bits,

connessa alle linee dati del microprocessore. Il cambiamento di formato è fatto semplicemente, a seconda delle necessità, cambiando l'indirizzo della memoria. Questa struttura con doppio indirizzamento, insieme al comando NAF in una sola parola, consente il trasferimento fra moduli CAMAC e memoria con due sole istruzioni di indirizzamento memoria: la prima istruzione indirizza la locazione (inesistente) NAF e la seconda istruzione indirizza una locazione formata da 24 bits; quest'ultima istruzione fa partire anche il ciclo CAMAC. In questo modo il tempo necessario per una operazione di lettura/scrittura CAMAC si riduce a 14 μ sec.

Il sistema di interruzione è così organizzato (fr. fig. 3), in ordine decrescente di priorità: 1 livello di allarme per trabocco dell'area riservata alla catasta; 1 livello di allarme per comando CAMAC non accettato (linea X falsa); 2 livelli ciascun dei quali può ricevere solo un segnale LAM, serviti automaticamente alla fine dell'istruzione in corso, con salto a subroutine (tempo di latenza massima di 10 μ sec.); 1 livello per 8 segnali LAM ordinati, identificabili con lettura della configurazione e rotazione dell'accumulatore (tempo impiegato: da 36 a 142 μ sec.); 1 livello comune ricevente gli altri 13 segnali LAM, riconoscibili individualmente mediante l'operazione CAMAC F(8); 1 livello per interruzione da pannello; 1 livello per la tastiera.

Al sistema si possono collegare i seguenti moduli CAMAC ausiliari: a) Console di presentazione dei principali segnali interni e di arresto ad indirizzo prefissato; b) modulo di estensione memoria RAM (4K/24bits a doppio formato); c) modulo di gestione dell'accesso diretto in memoria, per il trasferimento di blocchi di dati (il tempo minimo, per accesso prioritario è di 1.5 μ sec.) nelle tre modalità CAMAC di arresto, ripetizione e scrutazione d'indirizzo, e per la istogrammazione (incremento diretto del contenuto di memoria); d) modulo di interfaccia (1 unità CAMAC) per disco flessibile da 230K parole (8 bits), che comprende a sua volta un microprocessore 8080 ed un programma in REEPROM per la gestione degli archivi (questo è un buon esempio di come sia possibile in modo economicamente competitivo distribuire localmente la capacità elaborativa e così alleggerire il lavoro del controllo centrale); e) sistema di estensione per controllo fino ad 8 crates, mediante l'emissione di un comando di selezione di indirizzo. Il sistema si compone di un modulo emettitore e di un modulo

di controllo secondario; f) modulo lettore rapido per nastro di carta, con lettore ottico sul pannello (6 unità CAMAC), con velocità di lettura di 150 caratteri/sec; g) moduli di programmazioni della memoria REPR0M.

Per quanto riguarda il software è stata intrapresa la realizzazione di un sistema locale, operante a disco flessibile. Il sistema comprende:

- Un monitor in tempo reale;
- un supervisore per l'ingresso/uscita;
- un editore;
- un assemblatore;
- un editore;
- un programma per la messa a punto;
- un compilatore in tempo reale BASIC, specializzato per il CAMAC, secondo le specifiche del gruppo software del comitato ESONE.

5. - APPLICAZIONI DEL MICROPROCESSORE MOTOROLA M 6800

Se si eccettua un crate controller autonomo presentato alla mostra della ditta Dornier (Germania occ.) e del quale per brevità non parliamo, le altre due applicazioni dei microprocessori facevano uso del Motorola M 6800, che sul mercato internazionale è ritenuto un valido competitor dell'INTEL 8080. La prima applicazione è stata presentata da D. Kollbach et al. dell'Hahn-Meitner Institut Berlin ed era il modulo di elaborazione generale CAMOPS (CAMac MODular microProcessor System), che, inserito in una qualunque stazione di un Crate CAMAC, sostituisce in molti casi complicati circuiti speciali. Nella particolare applicazione in campo medico presentata esso serviva, unitamente a moduli di memoria separati, al controllo interattivo di un televisore a colori per la presentazione modulata in intensità di dati scintigrafici.

La seconda applicazione dell'M6800, presentata alla mostra, era simile a quelle del § 4 e riguardava il crate controller autonomo CAPRO-1, prodotto dalla ditta INCAA (Olanda)

6. IL MICROCALCOLATORE LSI 11

Fra i dispositivi MOS vogliamo infine ricordare il microcalcolatore LSI 11 della ditta DEC, che nella versione scatola si chiama anche PDP11/03. Esso è dunque un completo calcolatore della famiglia PDP11, e quindi non rientra a rigori nella categoria dei semplici microprocessori che abbiamo considerato. Ma è anche il più piccolo ed il più economico dei calcolatori della famiglia, e si avvicina come costo all'insieme di un microprocessore più gli altri circuiti integrati ausiliari presi in considerazione nei §§ precedenti. I vantaggi evidenti dell'LSI11 sono: esso ha una lunghezza di parola di 16 bits; emula l'insieme di istruzioni del PDP11/40, (comprese le istruzioni per le operazioni in virgola mobile, non implementate negli altri microprocessori); può utilizzare lo stesso software che l'utilizzatore ha sviluppato per il PDP11/40. Solo i tempi di esecuzione sono più lenti (ad esempio un trasferimento fra registri impiega 3,5 sec. anziché 0,9 (μ sec.)). Alla mostra di Bruxelles la ditta HYTEC ELECTRONICS Maidenhead, Inghilterra ha annunciato la produzione di un controller per un singolo crate CAMAC

che si collega ad un modulo esterno di controllo, contenete l' LS11. E' la soluzione già adottata in al tri casi con i calcolatori più grandi della famiglia PDP11 ma avente questa volta un prezzo sostanzial mente minore.

7. - SVILUPPO DEL CAMAC SERIALE.

L'aspirazione del comitato ESONE di imporre lo standard CAMAC anche nel campo industria le ha sottolineato la necessità di uno schema più semplice ed economico della Branch highway paral lela per l'interconnessione fra crates in sistemi più lenti e distribuiti su grandi distanze.

Sono nate così le specifiche CAMAC per la trasmissione seriale ⁽¹⁾.

La branch highway è sostituita in questo caso dalla "serial highway" formata da un anello unidirezionale che ha origine in un "driver seriale", passa successivamente in tutti i crates CAMAC attraverso un "crate controller seriale" (SCC), ed infine ritorna al driver (fr. fig. 4).

Il numero massimo di crate controller è 62. La trasmissione può essere seriale al livello bit od a livello (8 bits). Nella modalità bit-seriale vengono trasmessi gruppi di 10 od 11 bit. Il primo è un

bit di partenza. L'ultimo o gli ultimi due sono bits di fer mata. La trasmissione è sincronizzata da un orologio, la cui frequenza massima è 5 Mhz. Il protocollo seguito nella trasmissione è il seguente: il driver del sistema invia sulla highway seriale un messaggio di comando di 5 bytes in caso di lettura o controllo, o 9 byte, in caso di scrittura, accom pagnato da sufficienti bytes di spaziatura per consentire l'in serimento di una risposta. Quando un controller riceve un mes saggio a lui indirizzato, egli opera un controllo di parità longitu dinale e trasversale e, se non c'è errore, esegue il comando accettato, e risponde con un messaggio di replica di 3 bytes in lettura o controllo, o 7 bytes in scrittura. La risposta del crate indirizzato è ripetuto in sincronismo con l'orologio

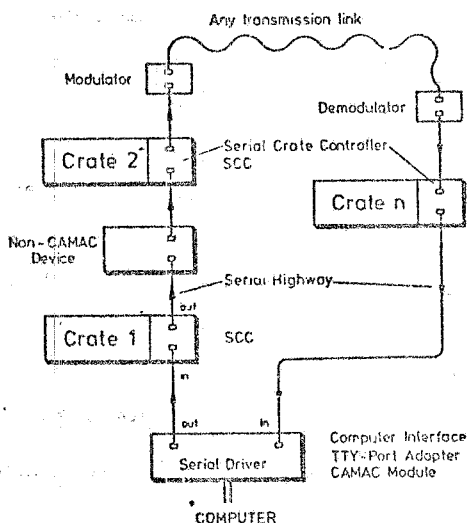


Fig. 4 - Interconnessione CAMAC seriale

da ogni controller a valle, fino a che il messaggio non arriva di nuovo al driver. Un controller può in serire un messaggio di richiesta, se un modulo da lui controllato ha inviato un segnale LAM.

E' quasi superfluo sottolineare che tutti i moduli CAMAC esistenti sul mercato sono ancora utilizzabili, perchè il dataway non è cambiato. Invece il controller CCA deve essere sostituito nel caso attuale dall'SCC. Nelle specifiche per il CAMAC seriale il comitato ESONE ha anche fissato le caratteristiche del controller seriale SCC-L1. Per quanto riguarda il driver seriale si possono avere diverse soluzioni. Quella indicata in fig. 4 può essere chiamata soluzione "diretta".

1) - CAMAC Serial system Organisation, A description (ESONE: SH 01/02/03)

in quanto il calcolatore è interfacciato direttamente al driver (E' ovvio che se il calcolatore è realizzato con un microprocessore, il calcolatore può essere alloggiato insieme al driver seriale). La soluzione indiretta consiste invece nel realizzare il driver seriale come un modulo CAMAC, pilotato da una installazione già esistente. Questa soluzione è meno efficiente della precedente, ma consente una maggiore flessibilità con la combinazione del sistema seriale e parallelo, ed una maggiore semplicità realizzativa.

Alla mostra di Bruxelles la ditta maggiormente attiva nel CAMAC seriale è stata la Kinetic System (USA), che ha presentato sei modelli diversi fra controllers seriali, drivers seriali, e moduli ausiliari.

Vogliamo anche ricordare la possibilità, già segnalata al § 4, che il crate controller MIK-X della SEC, ed anche il MACAMAC controller della BORER, se funzionanti come controller ausiliari possono comunicare con un SCC-L1, come indicato schematicamente in fig. 5.

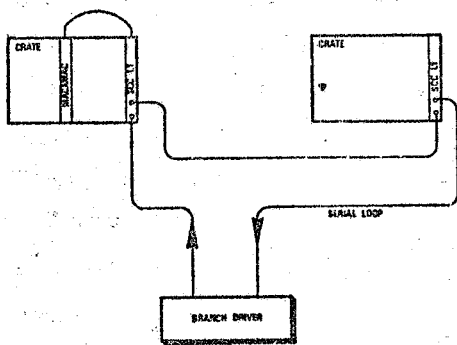


Fig. 5 - Sistema seriale con crate semiautonomo.

circa un Km. connettendo fra loro i vari crate controllers con coppie di fili aventi tipicamente impedenza caratteristica di 100

Tuttavia le specifiche del sistema seriale lasciano completamente impregiudicato il mezzo di collegamento fra i controllers, essendo esse definite solo in termini di caratteristiche di segnali e di formato dei messaggi agli ingressi e uscite dei controllers. Dunque, qualunque sezione della highway seriale può includere altri canali di comunicazione, allo scopo di perseguire le finalità specifiche dell'applicazione (ad es. per operare su distanze di parecchi Km.)

Un esempio tipico, già noto prima del Simposio, è il collegamento a microonde (~5 M bit/sec.) realizzato fra il Daresbury Nuclear Physics Laboratory (Gran Bretagna) e centri di calcolo distaccati. Si possono anche usare collegamenti a laser, collegamenti con sistemi telemetrici etc.

A Bruxelles è stato presentato da W. Branke (Dornier GmbH Germania occ.) una rete estensiva di stazioni automatiche per il controllo ambientale nell'Assia

La connessione consente un accesso alle linee Ned L. I controllers lavorano insieme, avendo il controllo seriale la priorità maggiore: un segnale di blocco consente di neutralizzare il controller ausiliario durante l'arrivo dei comandi sulla highway seriale. Il controller ausiliario sarà invece operativo nei periodi in cui non circolano comandi, e potrà quindi svolgere la sua funzione autonoma di elaborazione. Impieghi del genere sono stati illustrati nella già citata presentazione di Conway e Halbing (cfr. § 4).

Lo schema di figura 4 è realizzabile su distanze fino a

La rete si compone di 26 stazioni aventi distribuzione regionale. In ogni stazione esiste un minicalcolatore che controlla un singolo crate CAMAC. I dati vengono pre-elaborati e inviati ad un modulo CAMAC di interfaccia fra il dataway parallelo ed un modem. Essi vengono quindi trasmessi con una velocità di $\sim 10^3$ bit/sec. sulla rete telefonica regionale, fino al modem della stazione successiva per arrivare al fine al driver seriale del sistema, e quindi al sistema centrale di raccolta e calcolo. Questa non è una soluzione standard, perchè il modulo di interfaccia-trasmissione non è un crate controller seriale. Tuttavia viene rispettato il protocollo CAMAC seriale.

8. - SVILUPPI DEL SOFTWARE.

a) - Il linguaggio IML

Dall'epoca del 1° Simposio CAMAC a Lussemburgo nel 1973 si è venuto sempre più affermando l'IML (Intermediate level language), che, come è noto, è un linguaggio proposto dal software working group del comitato ESONE, per esprimere le operazioni descritte nelle specifiche hardware CAMAC.

La definizione finale di questo linguaggio viene data nel documento ESONE/IML/01⁽²⁾ che è per il software l'equivalente delle specifiche EUR 4100, EUR 4600 ed ESONE/SH/01 per l'hardware.

Poichè nelle specifiche hardware vengono descritte essenzialmente solo operazioni di trasferimento fra un elaboratore digitale e dispositivi CAMAC, l'IML è un linguaggio usato per gestire solo operazioni di I/O nel modo più flessibile possibile. Pertanto esso non può avere una esistenza autonoma, ma deve essere accolto da un linguaggio ospitante, di cui userà la particolare sintassi. Di conseguenza la definizione dell'IML nel documento citato è puramente semantica.

Tuttavia nello stesso documento, come esempio di applicazione, viene proposta anche una sintassi IML-MI particolarmente adatta a sistemi ospitanti di tipo macro-traduttore. Ad esempio un qualunque programma assembler, con possibilità di macro-definizioni, può essere usato per espandere gli statements IML.

L'IML è il linguaggio a livello più basso che opera la standardizzazione software in base alle specifiche hardware del CAMAC.

Gli statements dell'IML sono di due tipi: statements di azione e dichiarazioni. Solo i primi sono eseguibili al tempo di esecuzione e fanno uso di nomi simbolici che devono essere stati preventivamente dichiarati mediante l'altro tipo di statement. Riportiamo come esempio lo statement dichiarativo LOCD che fornisce la descrizione locale (cioè avente significato solo all'interno dell'unità di programma in cui è contenuto) dell'indirizzo di un dispositivo CAMAC:

LOCD REG, H, 1, 2, 19, 14

In questo statement REG è il nome di una costante CAMAC di tipo indirizzo (un registro hard -

2) - The definition of IML (october 1974) (ESONE/IML/01)

ware); H specifica che si sta definendo in indirizzo hardware; mentre i numeri specificano rispettivamente la diramazione, il crate, la stazione ed il sottoindirizzo assegnato al registro.

Gli statements di azione sono di tre tipi: single action, uni-device block transfer; multi-device block transfer: riportiamo un esempio di azione singola:

SA F00 REG, MEM

in cui SA indica l'azione singola; F00 indica l'operazione di lettura CAMAC F(0); REG è il registro più sopra dichiarato, mentre MEM indica la locazione di memoria (che pure avrà dovuta essere preventivamente dichiarata) nella quale si va a leggere il registro.

A Bruxelles M. Kubutz e R. Kind (Hahn-Meitner Instiut Berlin) hanno presentato una interessante realizzazione dell'IML sotto forma di macro-definizioni per il calcolatore PDP11 con il branch driver DEC CA - 11A ed il controller per singolo crate Borer type 1533 a.

Le macro-definizioni, facenti parte di una libreria, sono state scritte in linguaggio macro assembler e comprendono anche l'ingresso-uscita dati, evitando così di dover scrivere un programma di gestione (I/O handler) dell'ingresso/uscita. In media ad esempio, uno statement di azione singola (lettura o scrittura) consiste di 5 o 6 istruzioni assembler. Per l'assemblaggio è richiesta una memoria di almeno 16K parole sotto sistema operativo RSX-11M. Durante l'assemblaggio viene operato un accurato controllo per rivelare possibili; errori sintattici e semantici. E' stata usata la direttiva di errore con stampa di messaggio negli assemblaggi condizionali.

La programmazione è modulare, per consentire minimi cambiamenti con altri controllers CAMAC.

La gestione delle chiamate LAM è effettuata con una unità di programma separata (sotto sistemi operativi RSX-11M/D) al quale viene ceduto il controllo al momento dell'invio della LAM.

b) - Estensione del linguaggio BASIC al tempo reale.

L'interprete BASIC per il PDP11 sotto sistema operativo RT 11 è stato esteso al tempo reale da G. Bianchi (CAN, Saclay), M. Kubtz (Hahn-Meitner Institut), e A. Lewis (AERE, Harwell, Great Britain). Viene gestita l'interfaccia CAMAC CA11-F della DEC. Sono state seguite le raccomandazioni del gruppo di lavoro ESONE per il software. Sono così state aggiunti al linguaggio statements dichiarativi per definire i nomi e gli attributi delle variabili di processo CAMAC. Es:

100 DEC C1:PROCESS CAMAC (1, 2, 19, 14)(F1, F16)(B8).

In questa DEC identifica uno statement dichiarativo; C1 è il nome della variabile; Process indica la dichiarazione di una variabile di processo; CAMAC indica che C1 è anche una entità CAMAC; i numeri hanno lo stesso significato che in a); (F1, F16) indicano che deve essere effettuata una operazione di lettura o scrittura; e (B16) indica che si ha dato binario da 16 bits.

Una operazione di ingresso/uscita si può ora eseguire al seguente modo:

300 LET Y = C1,

giacchè il sistema conosce gli attributi di C1 dalla dichiarazione già fatta.

Vengono anche realizzati trasferimenti in blocco (mediante vettori di memoria) e operazioni di controllo (ad esempio ENABLE C1 esegue F(26) all'indirizzo CAMAC C1 più sopra dichiarato).

Le chiamate LAM sono dichiarate, ad es:

```
120 DEC I4: INTERRUPT CAMAC C1 GL 22,
```

in cui I4 identifica l'interrupt; C1 è la variabile di processo già dichiarata più sopra; GL22 indica la LAM ordinata n. 22.

L'avvenimento asincrono di una interruzione può essere gestita con:

```
400 ON I4 GOSUB 900,
```

il cui significato è evidente.

c) - Il compilatore CASIC - 11B.

E' stata presentata da H0 - Xich-Tue e J. Servant(Schlumberger, France) una nuova versione del CASIC che, malgrado utilizzi la sintassi BASIC, non è un interprete, bensì un compilatore conservativo: cioè il programma è compilato una sola volta: viene generato un programma oggetto che è almeno 10 volte più veloce di un interprete che traduce ogni istruzione prima di eseguirla; inoltre è conservativo, in quanto consente la modifica del programma sorgente direttamente da telescrivente. Questa versione del Casic ammette come interfaccia CAMAC il branch driver ICP11 o il controller per crate singolo JCC 11 della ditta Schumberger. Esso è stato scritto per il calcolatore PDP11 con un minimo di 16K parole. Oltre le istruzioni proprie del BASIC; sono state previste le istruzioni per gestire il CAMAC in tempo reale. Le variabili CAMAC devono essere preventivamente dichiarate mediante l'istruzione DCLCAM e localizzate mediante l'istruzione STATION. Esempi di statements CAMAC, di chiaro significato, sono:

```
10 DCLCAM R1, R2, R3
```

```
20 STATION R1 = (1, 1, 3, 0)
```

```
30 STATION R2 = (1, 1, 6, 2)
```

```
40 STATION R3 = (1, 1, 8, 15)
```

```
50 R1 (16) =A          (scrittura di R1 da A)
```

```
60 B = R2(0)          (lettura di R2 in B)
```

```
70 OPER R3(8)          (Testi della LAM in R3)
```

```
80 IF Q(R3) =0 STOP    (test di Q in R3)
```

```
90 ON LAM(R3) DO 200    (attivazione del programma di servizio della chiamata LAM)
```