

Bit



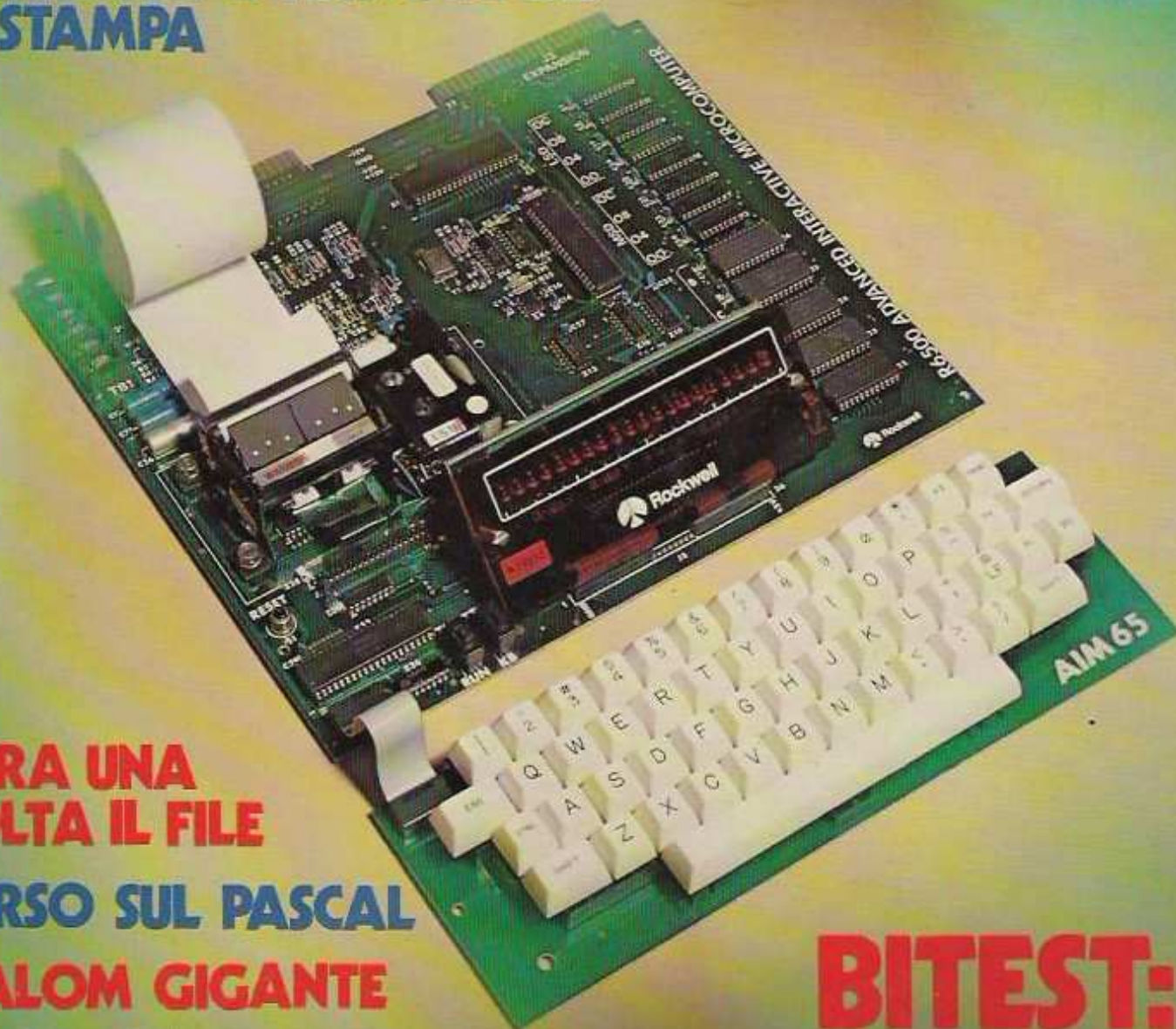
UNA PUBBLICAZIONE
DEL GRUPPO EDITORIALE JACKSON

ANNO 4 N. 16

MICROPROCESSORS - HARDWARE -
SOFTWARE
HOME & PERSONAL COMPUTER

L. 3.000

- LA CASA DEL FUTURO
- HOME COMPUTER
- STAMPANTI E TECNOLOGIE DI STAMPA



- C'ERA UNA VOLTA IL FILE
- CORSO SUL PASCAL
- SLALOM GIGANTE
- APPLE NANOCOMPUTER PET E SINCLAIR PRATICO

BITEST: AIM 65



Bitest: L'AIM 65

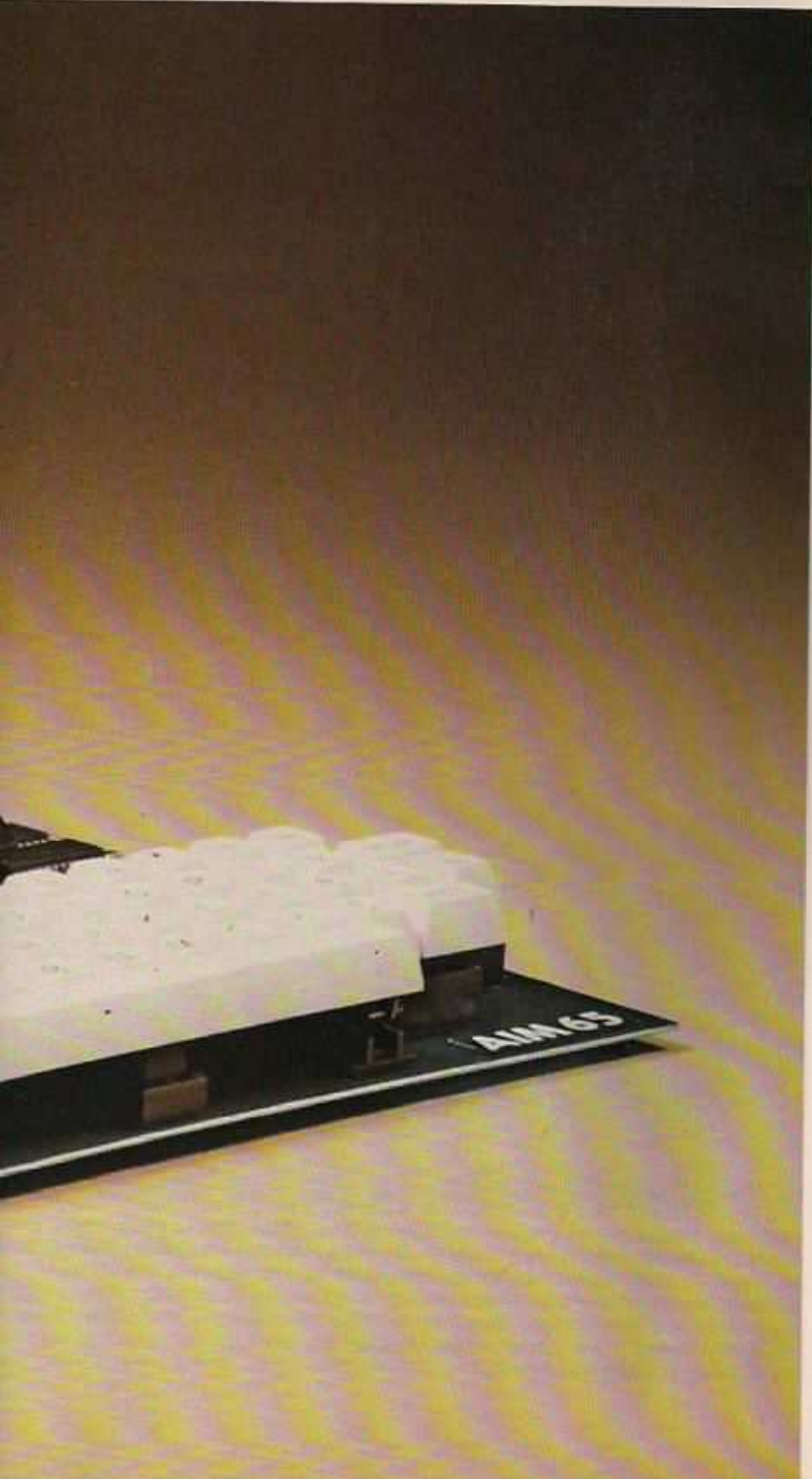
a cura della Redazione

Per "sistema di sviluppo" abitualmente s'intende un computer monoscheda basato su di un microprocessore, che non comprende che la circuiteria ed il software minimo. Oltre al microprocessore il "sistema di sviluppo" comprende una memoria da qualche kilobyte, alcuni componenti periferici e soprattutto dei connettori che permettono al progettista di effettuare tutte le manipolazioni di prova possibili. Dal punto di vista del software, un monitor, un assembler ed un semplice sistema editor sono sufficienti per poter mettere a punto le routines adatte all'applicazione prevista. Mettendo in produzione l'AIM 65, la Rockwell ha voluto realizzare e diffondere sul mercato un sistema di sviluppo che presenta, da un lato le caratteristiche di un "vero" elaboratore, dall'altro è un sistema che potrà essere adottato da numerosi utilizzatori finali.

Vista d'insieme

Ad una prima occhiata ciò che meraviglia, osservando un AIM65, tutti coloro che non ne hanno sentito parlare (questi "ignari" devono essere piuttosto infrequenti tra i nostri lettori) è l'assenza di un qualunque contenitore. L'AIM65 si presenta come un sistema formato da due schede stampate, reciprocamente connesse tramite un cavo piatto a 16 conduttori, che alle estremità è munito di due prese che hanno il passo di un circuito integrato. La prima scheda, comprende i 53 pulsanti che formano una tastiera tipo macchina da scrivere. Vi è un tasto di controllo (CTRL), un tasto DEL e tre tasti speciali (F1, F2 ed F3) che hanno delle funzioni definite dal software. Per tutti coloro che sono abituati a delle tastiere più ridotte (ad esempio quelle che sono installate sugli elaboratori SYM e KIM) la tastiera ASCII completa, sembra un poco sorprendente per un sistema di sviluppo. Peraltro, una tastiera del genere, si dimostra di facile uso, e non soffre di quella nota malattia che è l'obsolescenza, sovente micidiale (le fasi della malattia sono scandite dall'apparire dei modelli nuovi...).

Il circuito stampato principale serve come supporto generale per l'elaboratore. I componenti più voluminosi, sono la stampante termica ed il display ad elettroluminescenti che mostra sino a 20 cifre, e che da la pos-



sibilità all'elaboratore di "comunicare" con l'operatore. Oltre a queste due periferiche, il circuito stampato maggiore, che misura 260 x 300 mm, monta 17 zoccoli per circuiti integrati, sui quali sono inseriti i chip principali, come dire il microprocessore, i circuiti d'interfaccia e le memorie; vi sono poi altri 19 circuiti integrati DIL che sono direttamente saldati alle piste. Sempre in ordine ai componenti principali, vale la pena di citare il quarzo da 4 MHz che pilota la base dei tempi del microprocessore 6502.

La parte retrostante della basetta principale, è munita di due connettori a doppia ramatura che impiegano 44 contatti e che servono per interfacciare il circuito con i componenti esterni.

L'alimentazione, fatto da notare, non è inclusa nel sistema, e deve essere applicata dall'esterno ad un connettore a sei terminali.

Le tensioni richieste dal microcomputer sono: + 5V, regolati (tolleranza $\pm 5\%$), con 2A, e + 24V (tensione non regolata, che quindi può avere una variazione del $\pm 5\%$), con 2.5A di picco. La tensione delta a 24V, non serve altro che a far funzionare la stampante. Le prese che sono previste per la tensione d'ingresso di $\pm 12V$, non servono per il microcomputer, ma per apparati supplementari che potranno essere collegati alla base principale, formando i sistemi di estensione; per tale motivo, giungono direttamente ai connettori di applicazione e di espansione. L'AIM 65 reca due interruttori ed un pulsante. L'ultimo detto è il RESET che effettua un'inizializzazione di tutti i parametri controllati dal monitor. L'interruttore KB/TTY permette di sconnettere sia la tastiera che la stampante termica, allorché si prevede l'utilizzo di una stampante esterna, o di un complesso di visualizzazione. L'interruttore STEP/RUN, infine, portato su "single step" obbliga il microprocessore a bloccarsi alla fine di ciascuna istruzione, il che risulta assai utile per effettuare la ricerca degli errori di programmazione, o "debugging" che dir si voglia, e per la prova dei circuiti d'interfaccia applicati ai connettori di espansione.

La stampante termica a 20 caratteri è assai semplice da utilizzare. Può essere collegata di continuo, in simultanea, al display LED e può anche essere comandata, tramite il software, in modo tale da non stampare altro che i risultati più interessanti.

La stampante è gestita da un "VIA" (versatile Interface Adapter) del modello 6522 e può stampare 120 linee per minuto. Ciascun carattere (alfanumerico maiuscolo) è



La tastiera completa ASCII, è senza dubbio il dettaglio che sorprende di più in un sistema di sviluppo di questo genere. È peraltro un fattore essenziale per la programmazione con un linguaggio d'alto livello.

È forse proprio questa caratteristica che eleva l'AIM 65 all'altezza degli altri veri e propri "personal computer". Oltre al gruppo di caratteri ASCII, il sistema comprende un tasto CTRL, che, azionato simultaneamente al PRINT, permette di mettere in azione la stampante ed anche un tasto DEL che consente di far arretrare il cursore di una posizione per correggere gli errori, nonché di tre tasti di funzioni speciali, ovvero F1, F2 ed F3, che possono richiamare delle routines scritte dall'utilizzatore.



L'AIM 65 si presenta, in sostanza, come un grande circuito stampato a doppia ramatura che supporta anche la stampante, il display e tutte le parti dell'elaboratore. Se vogliamo, un complesso un po' insolito. Guardando la base generale, si nota, in alto a sinistra, la morsettiera munita di contatti a vite che serve per il raccordo all'alimentazione. Tale settore è l'unico mancante per completare il sistema. Il lato componente è completamente serigrafato, e la serigrafia consente, con l'appoggio del grande schema elettrico, compreso nella documentazione, di individuare con facilità tutti i componenti circuitali. Balza all'occhio che tutti i principali circuiti integrati impiegano degli zoccoli.



La stampante è una piccola unità termica "intelligente" che ha delle capacità di rappresentazione perfettamente eguali a quelle del display LED. Può stampare i 64 caratteri ASCII con una disponibilità di 20 caratteri per riga. La testina di stampa è composta da un gruppo di 10 elementi termici, ciascuno dei quali s'incarica della stampa di 2 caratteri. La testina compie quattro brevi percorsi di andata e ritorno per la stampa di ciascuna riga. La velocità di stampa massima è di 120 righe al minuto, il che significa stampare 40 caratteri al secondo, se le righe sono complete.

compreso in una matrice da 5 x 7 punti. I caratteri sono impressi da un gruppo di sottili elettrodi che bruciano la superficie della carta, che è un modello speciale termosensibile.

Il meccanismo della stampante non è munito di alcun sistema di protezione, e tutto il complesso ha una semplicità tale, che prima d'averlo impiegato, ciascuno si chiede se l' assieme metalplastico, possa davvero lavorare alla bella velocità di 40 caratteri al secondo! In vero, scorrendo le specifiche dell'AIM 65, si nota che il costruttore è stato un pochino ottimista circa le caratteristiche della stampante, specie quando afferma che è "totalmente silenziosa". Praticamente, però, questa sopravvalutazione è divenuta un po' un'abitudine, e d'altronde non si può proprio chiedere troppo ad un dispositivo di questo genere. L'essenziale è ottenere una scrittura giusta e netta, e da questo punto di vista, non vi è proprio nulla da obiettare, considerata la natura di questa "stampantina".

Quando la stampante funziona di continuo, è reciproca al display LED: ogni volta che un messaggio compare sul display, la stampante inizia a scriverlo sulla carta. Purtroppo, com'è logico, d'altronde, la stampa avviene in modo molto più lento dell'indicazione LED, e se un programma in fase di esecuzione ha bisogno di stampare vari risultati, il microprocessore passa la maggior parte del suo tempo operativo ad aspettare che la stampante abbia compiuto il suo lavoro. Al contrario, se la macchina non è posta in azione, il programma può essere sviluppato alla velocità normale, ed i risultati appaiono unicamente sul display a LED, però con un inconveniente diverso: se i risultati sono multipli, ed uno segue l'altro, non si potrà che leggere quello finale, senza il controllo delle operazioni intermedie.

Un ultimissimo piccolo difetto del display a LED è che l'angolo di lettura è piuttosto ridotto. Basta deviare di circa 20° dal migliore angolo di lettura, cioè dall'asse perpendicolare al piano del lettore, e la decifrazione diviene subito difficile.

Struttura dell'hardware

L'organizzazione hardware dell'AIM 65, è descritta dal manuale d'impiego dell'elaboratore (User's Guide) e da un grande schema, ben spazioso e comprensibile, forniti entrambi con il calcolatore. La disposizione

BITTEST AIM 65



Il display dell'AIM 65 può mostrare la serie completa dei 64 caratteri ASCII maiuscoli, e malgrado le limitazioni dell'affissione a 16 segmenti, non se la cava troppo male anche nella presentazione di caratteri del genere di ►, %, * e !, inoltre si nota che la differenza tra la "O" e lo "0" è abbastanza netta. Sostanzialmente, il display si comporta come una specie di tubo catodico che può riportare una sola riga di testo. Allorché le righe del programma o del testo comportano più di venti caratteri, il testo inizia a slittare dopo l'introduzione dei primi 20 caratteri, come dire che il primo carattere della riga, a sinistra, scompare, mentre il gruppo degli altri 19 caratteri si sposta di una posizione verso sinistra, e l'ultimo carattere appare sulla destra. Se il programma compare sul display una riga dopo l'altra, si ha appena il tempo di veder lampeggiare il display, prima che giunga la riga successiva.

dei vari elementi sul circuito stampato, è praticamente identica a quella dello schema elettrico; in tal modo è assai facile identificare i vari gruppi funzionali della macchina.

Il cuore dell'AIM 65, è il microprocessore 6502, che è uno dei CPU ad 8 bits più rapidi e dalla migliore reputazione che vi siano sul mercato. Il 6502 non comprende che due "register index" ed un solo accumulatore, oltre ai normali program counter, registro di stato e puntatore dello stack. Tale scarsità di registri non è certo per facilitare la programmazione nel linguaggio "assembler", ma tuttavia, il 6502 resta un microprocessore dall'impiego molto semplice grazie ai 13 sistemi d'indirizzo disponibili per raccogliere le istruzioni. Il gruppo d'istruzioni del

6502 si compone di 56 tipi d'istruzioni diverse.

I tre bus principali del 6502 (dati, indirizzo e controllo), sono distribuiti un po' ovunque sulla basetta dell'AIM 65. In tal modo, sono portati al connettore d'espansione J3 senza passare per alcun tipo di stadio buffer o separatore.

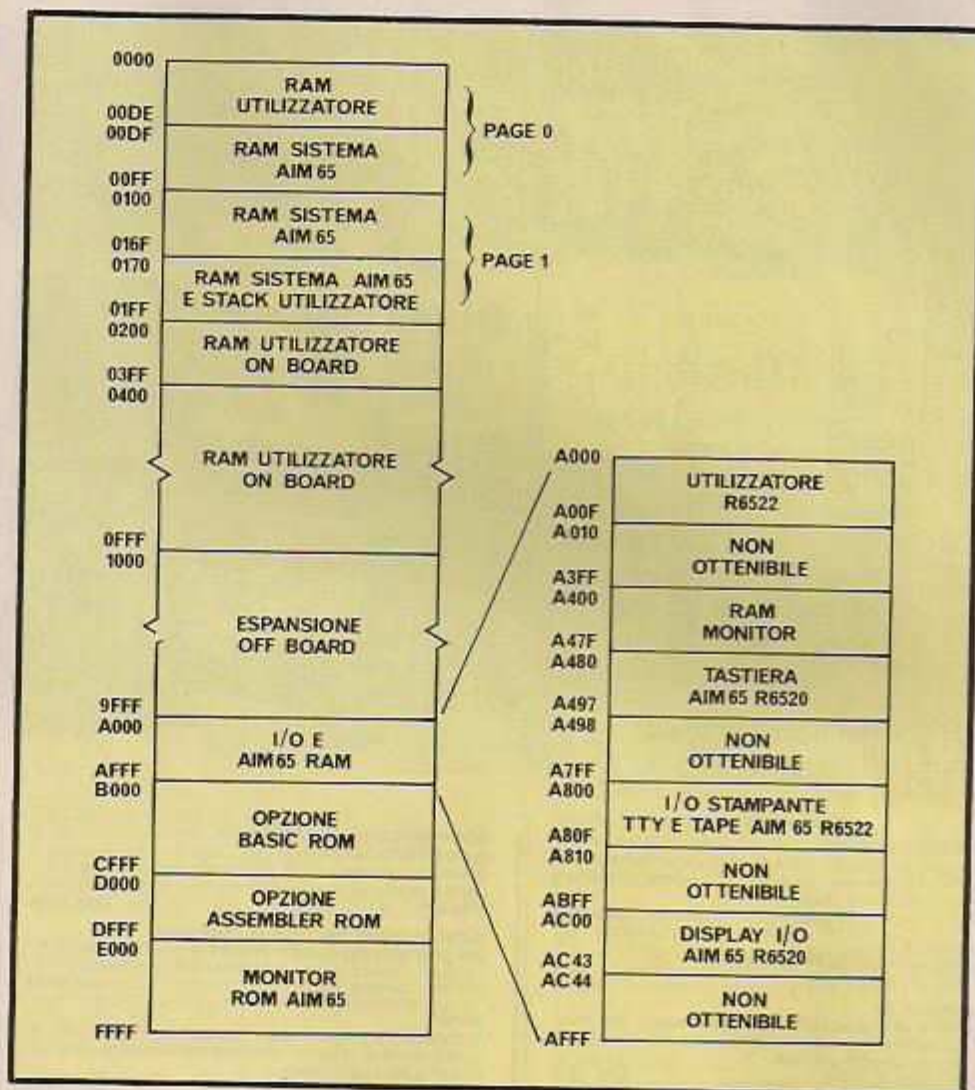
L'oscillatore della base dei tempi, impiega, come si è detto, un quarzo da 4 MHz ed un divisore per quattro, che fornisce al microprocessore un'onda quadra dall'andamento simmetrico che ha la frequenza di 1 MHz. Il 6502 s'incarica di produrre le due fasi $\Phi 1$ e $\Phi 2$ di clock, che sono necessarie per il timing del circuito.

Il circuito "chip-select" è costituito da una decodifica da 1 ad 8, che gestisce le otto



pagine da 4 Kbyte degli indirizzi più elevati, e da una doppia decodifica da 2 a 4, che assicura la selezione dei 4 Kbyte della RAM sul pannello e dei circuiti d'ingresso-uscita. Tre dei segnali d'uscita della decodifica da 1 a 8 sono portati sul connettore J3 a permettere l'estensione della memoria ROM e l'unione di un chip di I/O. Per estendere invece il settore RAM è necessario un ulteriore circuito di decodifica esterno.

La tastiera è controllata dalle linee d'ingresso/uscita del RIOT 6532 (RAM, I/O, Timer). Le otto linee dei pulsanti sono connesse alle otto linee del porto B, le otto colonne dei tasti alle otto linee del porto A. Il display a 20 caratteri alfanumerici che s'impiega sull'AIM 65, è costituito da 5 settori ciascuno munito di 4 digit a 16 segmen-



La configurazione dell'AIM 65, messo a nostra disposizione per effettuare questo "Bitest", era equipaggiata con 4 Kbyte di memoria RAM (8 chip del tipo 2114) e di 20 Kbyte di memoria ROM (5 chip del tipo 2332). La ROM è divisa come segue: 4 Kbyte per il monitor, 4 Kbyte per l'editor, 4 Kbyte per l'assembler ed 8 Kbyte per il BASIC. La RAM è disponibile per l'utilizzatore a partire dall'indirizzo 0200 (esadecimale); i primi 200 bytes sono utilizzati per i parametri del sistema.

I 5 chip DL-1416A, sono pilotati da un PIA 6520 (Parallel Interface Adapter) che è montato sulla stessa scheda, e non necessitano di alcun sistema di decodifica o multiplexing. Ciascun "digit" è indirizzato singolarmente e pilotato direttamente in ASCII. Il display stesso svolge la funzione di latch e decodifica. Sfortunatamente, i display costituiscono la parte più delicata di tutto il sistema, infatti sono molto sensibili agli sbalzi della tensione d'alimentazione

ed in più hanno una vita media lavorativa non molto elevata. Ove si renda necessaria la sostituzione, si deve notare che ciascuno di essi riporta sull'involucro un punto colorato che serve per l'identificazione, ad evitare differenze nella luminosità ed una minor durata. Peraltro, la mancanza di necessità di dover procedere al "refreshing" del display, semplifica notevolmente la gestione del sistema, ed aumenta la velocità di elaborazione. La stampante termica, che

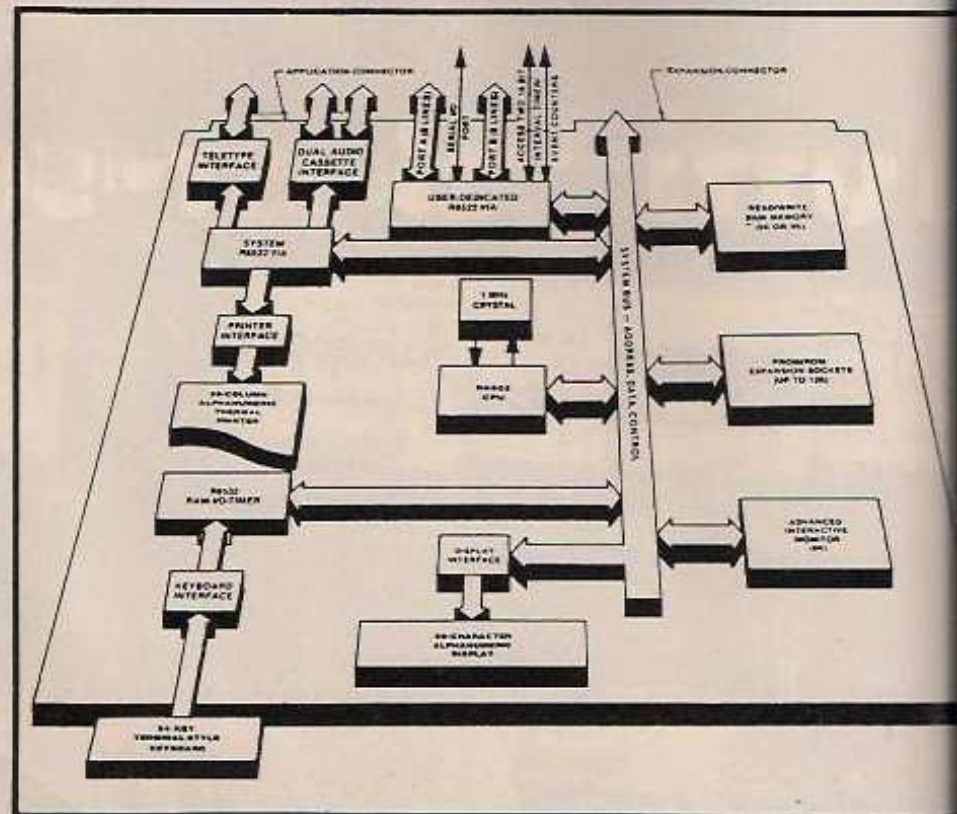
come abbiamo visto unisce ad una notevole semplicità una buona efficienza, è pilotata da una parte del VIA 6522 montato sulla base, ed è totalmente gestita dal software. I caratteri sono scritti in sette passaggi successivi ad alta velocità, secondo una matrice di 5 x 7. Analoghi tipi di stampante sono impiegati in molte calcolatrici economiche (Olivetti, Triumph, ecc.) cosicché gli eventuali pezzi di ricambio che si rivelassero necessari, sono facili da reperire.

La seconda parte del VIA interno è dedicata alla gestione dell'interfaccia per le cassette e dell'interfaccia TTY. All'AIM 65, si possono collegare due registratori a cassette. Sul connettore J1 vi sono due uscite audio di diverso livello (HI e LOW), un ingresso audio e due linee di controllo dei motori di trascinamento, che rendono l'interfaccia elementare. I due dispositivi esterni, hanno in comune le linee che riportano i segnali, e sono scelti tramite il sistema di controllo dei motori. Logicamente, i due sistemi di registrazione non devono e non possono lavorare contemporaneamente.

I registratori da impiegare possono essere del tutto convenzionali, ma non per questo di qualità scadente, anche se non serve una risposta eccezionalmente elevata. Per una corretta lettura dei dati introdotti dall'esterno, è necessario regolare opportunamente il trimmer VR1, che funge come soglia per il comparatore d'ingresso. In vero, l'operazione, anche seguendo il programma di test consigliato non è molto agevole, essendo assai critica la posizione del trimmer, ed in più è resa complicata dal fatto che oggi quasi tutti i registratori in commercio sono dotati del controllo automatico del livello di registrazione. Quando all'ingresso non è presente alcun segnale, l'automatismo interno fa sì che il guadagno degli stadi che s'incaricano della registrazione sia al massimo, ed in tal modo, si ha una incisione distorta del primo treno di segnali che giunge dall'AIM. In parte, l'impiego di buoni nastri aiuta a risolvere il problema.

Il controllo dei motori può essere fatto manualmente, a meno che non si vogliano impiegare cassette alternativamente a controllo di programma, o come input e output dell'assembler.

Aumentando gli intervalli tra i blocchi di dati che si registrano, il controllo manuale risulta abbastanza fattibile. Le connessioni da eseguire, dipendono dal tipo di registratore che si vuole utilizzare, ma se non si fa la massima attenzione si finisce per rompere i



transistori sulla piastra. In pratica, la miglior soluzione, è impiegare due piccoli relais che comandino i registratori.

Alla massima velocità, si registrano circa 100 caratteri al secondo. Ogni blocco di dati, sarà sempre preceduto dal segnale di sincronismo, che inizia con una stringa di riconoscimento che varia a seconda dei tipi di dati e termina con due byte di controllo ottenuti, sommando senza riporto, il contenuto del blocco, byte per byte su 16 bit. In tal modo si ha la possibilità di rilevare eventuali errori di trasmissione. A differenza del KIM 1 che trasforma la registrazione di ogni byte in due caratteri ASCII, l'AIM65 registra i dati in binario raddoppiando sia la capacità del nastro che la velocità di trasferimento. Opportune routines di monitor permettono però anche l'incisione di programmi nel formato KIM 1, ma sebbene le specifiche dicono il contrario, sembra che non sia possibile leggere con l'AIM dei dati registrati nel formato KIM.

Nell'AIM 65, ogni bit registrato inizia con un semiperiodo a 2400 Hz, che è seguito da tre semiperiodi a 1200 Hz se vale zero, o da tre semiperiodi a 2400 Hz se vale uno. Il metodo, in verità sembra strano, e, tra l'altro, implica che la lunghezza di un bit non sia costante ma dipenda dal suo valore, però in sostanza è molto affidabile e sicuro. Le frequenze scelte, sono adatte alla banda passante dei registratori audio, ed il semiperiodo a 2400 Hz, permette all'AIM 65 di adeguarsi alle variazioni di velocità dei motori di trascinamento. Le frequenze indicate sono infatti usate per la registrazione nel tratto della massima efficienza; in lettura, il sistema usa la durata del primo semiperiodo per determinare il valore del bit ricevuto. Volendo trasferire dati tra due AIM 65, si possono semplicemente effettuare le op-

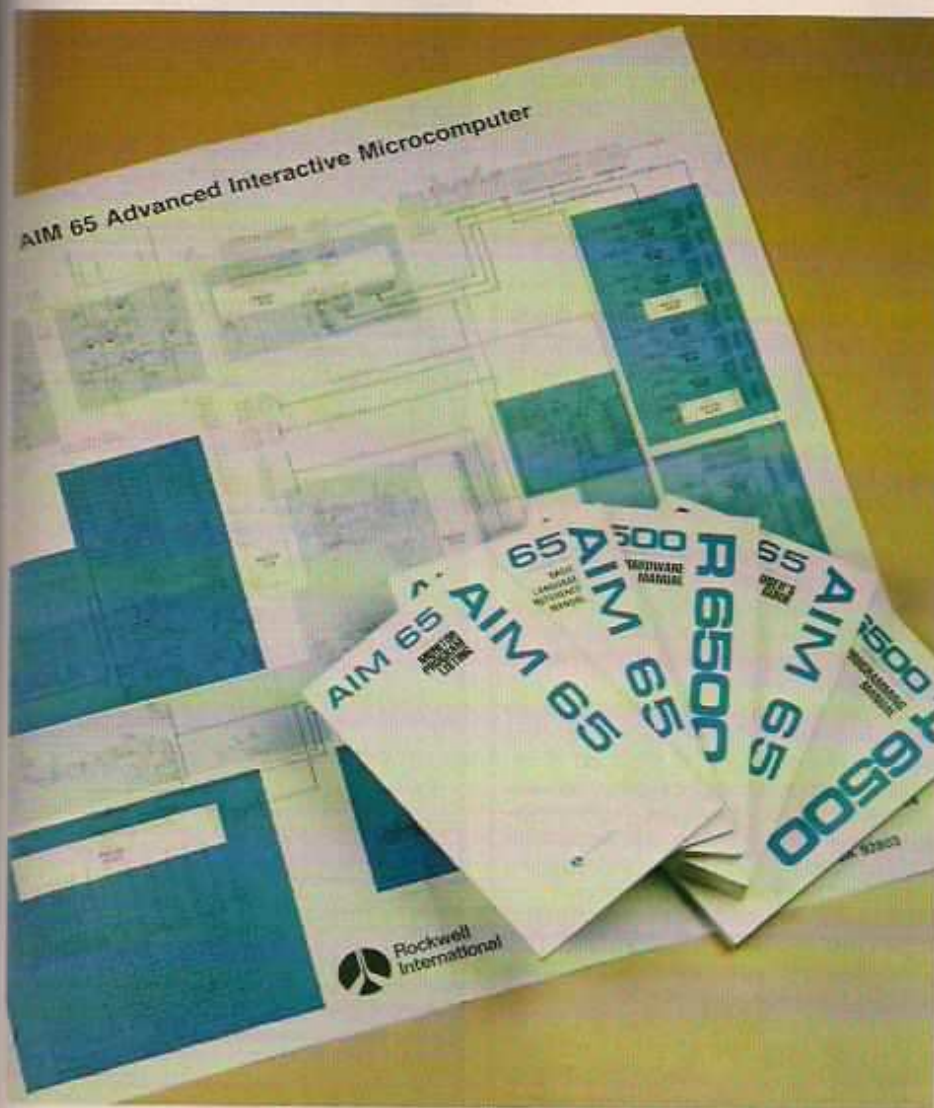
portune connessioni tra le interfacce audio senza ulteriori problemi.

Il VIA interno, gestisce anche l'interfaccia "TTY full duplex" a loop di 20 mA, che permette di utilizzare una telescrivente come terminale per l'AIM 65. A tale scopo, è sufficiente, dopo l'accensione, portare il commutatore KB/TTY su TTY. Il carattere RETURN permette all'AIM 65 di determinare la velocità di trasmissione del terminale e adeguarsi. Il display alfanumerico resta comunque attivo anche in quest'altra funzione, mentre la tastiera non è più scansata dal sistema. È presente anche un ingresso (SERIAL IN) per la connessione di un terminale per la trasmissione, sino a 9600 Baud (240 caratteri/secondo).

Il "Baud rate" che è programmabile tra 1 e 9600, può comunque essere impostato modificato manualmente.

Oltre alla connessione per l'interfaccia della cassetta, e per l'interfaccia TTY, il connettore "J1" rende disponibili per l'utilizzatore i due porti d'ingresso/uscite e le linee di controllo di un secondo VIA che è interamente riservato alle applicazioni esterne. Ciò significa che vi sono 16 linee di I/O programmabili singolarmente, due timer a 16 bit, uno shift register, le linee per l'halt, shake ed altre utili possibilità. C'è da osservare che per un sistema della potenza dell'AIM 65 non è molto ed è spesso necessario espandere le possibilità di input/output per risolvere vari problemi di controllo. Il VIA residente svolge, al di fuori di ogni dubbio, una importante funzione didattica sperimentale, che può essere ben sfruttata dall'utente.

Il connettore di espansione J3 porta all'esterno i tre bus di sistema per le eventuali espansioni. A parte due segnali (Φ 2 e R/W) che sono bufferizzati, l'insieme dei



La documentazione è uno dei punti di forza dell'AIM 65. Anche se non è stata scritta per i novizi, decisamente aiuta l'utente a comprendere tutte le finenze dell'AIM 65 e del suo firmware.

linee dell'address bus, del data bus, e del control bus giungono direttamente dal CPU. Per ottenere il corretto funzionamento di un sistema espanso è necessario bufferizzare almeno l'address bus, anche se motivi di sicurezza consiglierebbero lo stesso trattamento per tutte le linee uscenti.

L'espansione della memoria ROM e l'aggiunta di un chip I/O è agevole, essendovi già tre "chip select" disponibili. L'espansione della memoria RAM richiede l'impiego di una decodifica supplementare.

Firmware

Nella versione più semplice, l'AIM è fornito con un monitor ed un editor di testo che occupano 8 Kbyte della memoria ROM. La Rockwell propone anche un assembler più potente di quello che fa parte del monitor ed un interprete BASIC appositamente realizzato dalla Microsoft. L'elaboratore sul quale abbiamo effettuato questo "Bitest" era equipaggiato con un assembler da 4

Kbyte e dal BASIC da 8 Kbyte. In tal modo abbiamo potuto renderci conto che l'AIM 65 è un sistema di sviluppo dalla potenza e dalla facilità di utilizzazione insolite.

Cercheremo di chiarire a chi legge questo parere, e per iniziare "ab ovo" presenteremo prima di tutto il programma monitor che è destinato alla messa a punto, all'analisi ed al debugging dei programmi, così come al richiamo dei programmi, come per esempio l'assembler o l'interprete BASIC. Tramite una serie di comandi, che non impiegano che un solo carattere, il monitor consente di visualizzare, ed eventualmente modificare il contenuto dei sei registri del microprocessore e della memoria in modo semplicissimo. È d'altra parte sufficiente premere un solo tasto (K) per far sì che il display inizi a tradurre la lista delle istruzioni che sono state introdotte in forma di codice mnemonico. Ciò significa che l'operatore può impiegare i comandi mnemonici (come ad esempio "NOP") che sono abbreviazioni delle istruzioni (No OPERATION), invece dei codici delle diverse istruzioni in linguaggio macchina (EA, nel nostro esempio). In tal modo, diviene molto facile familiarizzarsi con il set d'istruzioni del 6502, e non è più necessario ricercare i codici operativi sulla tavola di programmazione.

Un'altra possibilità del monitor è quella di "seguire i passi" dell'esecuzione di un programma, istruzione dopo istruzione. Per questo tipo di funzionamento, è sufficiente posizionare l'interruttore RUN/STEP su STEP e dare il comando adeguato alla funzione che si richiede: Z, V oppure H per avere stampato, rispettivamente, l'istruzione da eseguire, lo stato dei registri o lo stato del program counter, dopo l'esecuzione di ogni istruzione. In tal modo diviene possibile ottenere la "storia" delle istruzioni eseguite, o la "storia" (nel senso di sequenza) dello stato dei registri durante l'esecuzione di un programma, il che può essere molto utile per la messa a punto.

Il monitor gestisce ugualmente i quattro break point. È possibile attivarli o cancellarli sia complessivamente che in modo individuale. Quando il programma si è fermato al break point, l'operatore può esaminare così come modificare il contenuto della memoria, anche se non sarà purtroppo sempre possibile riprendere l'esecuzione normale del programma.

Il monitor si occupa anche di caricare (L) i programmi che giungano da un sistema collegato all'ingresso (tastiera, cassetta o TTY) nella memoria, o di scaricare la memoria su di un sistema collegato all'uscita

(stampante, display, cassetta o TTY). Il controllo delle periferiche è analogamente effettuato dal monitor (messa in funzione della stampante e delle due cassette, avanzamento della carta sulla stampante, verifica della registrazione sulle cassette, ecc.). Tutte queste operazioni sono molto comode. Bisogna tuttavia notare che il comando (3) di verifica di un file scritto su nastro non ne controlla il contenuto con quello di memoria ma semplicemente, tramite 12 byte di checksum, verifica che non siano avvenuti errori fisici di scrittura sul nastro.

Citeremo infine le tre funzioni F1, F2 ed F3 che sono gestite dal monitor, e che permettono di richiamare delle routines specifiche che sono state introdotte precedentemente nella memoria RAM o che fanno parte del programma monitor.

Il text editor, permette la gestione di un testo a livello di linea e carattere. L'input del testo può essere selezionato dalla tastiera o da una qualunque delle periferiche; ed altrettanto vale per l'output. È possibile allocare il buffer del testo in una qualunque parte della memoria RAM e definire l'ampiezza. Se non si indica il limite superiore, il sistema lo pone automaticamente nell'ultima locazione di RAM disponibile. Le locazioni \$E1, \$E2 ed \$E3, \$E4 contengono in ogni momento gli indirizzi dell'ultimo e del primo byte occupato, rispettivamente. Ciò permette di sapere sempre l'effettiva occupazione di RAM per il testo.

Oltre ai normali comandi di inserimento e di eliminazione di righe, vi sono altri due potenti comandi per la ricerca (F) e per la sostituzione (C) di una stringa di testo. Se non si indica una nuova stringa, essa è aggiunta uguale all'ultima cercata ed in tal modo si può elaborare tutto il testo in maniera molto comoda e veloce.

Il testo può evidentemente essere costituito solo da caratteri ASCII con un RETURN come comando di delimitazione di linea, e con due RETURN per la fine del testo. La massima lunghezza della linea è 60 caratteri.

Delle configurazioni binarie non valide, che fossero erroneamente introdotte, possono produrre dei malfunzionamenti strani, e, non costituendo caratteri stampabili, sono di difficile individuazione. Un piccolo errore, è riscontrabile nel comando d'inserimento ed eliminazione dell'ultima riga di testo, che in certe condizioni non è eseguita correttamente, mentre per il resto il text editor è a prova di qualunque "sevizza" e si rivela molto utile per la compilazione di programmi

in assembler o in BASIC.

Una ROM da 4 Kbyte, che contiene un assembler a due passi, può essere fornita come accessorio opzionale dalla Rockwell. L'assembler serve a convertire i codici mnemonici delle istruzioni in codici operativi comprensibili da parte del 6502, ed in tal modo consente di poter specificare gli indirizzi dei dati e delle istruzioni sotto forma di simbolo.

Durante il passo 1, l'assembler determina i valori dei simboli ed il loro posto su di una tavola. Il vero codice oggetto, è così generato dal passo 2; l'assembler impiega la tavola dei valori dei simboli stabiliti durante il passo 1 per calcolare gli indirizzi, i displacement ed i dati del programma. Il controllo degli errori, è contemporaneamente effettuato durante il passo 2.

L'assembler dell'AIM 65, così dispone di nove direttive che permettono di attribuire un valore ai contatori dei simboli e di locazione, di riservare e inizializzare delle locazioni di memoria, di comandare gli ingressi/uscite dell'assembler e di controllare la presentazione del listing.

Per utilizzare l'assembler, il metodo più semplice consiste nell'elaborare il codice sorgente tramite l'editor di testo ed in seguito caricarlo su di una cassetta o passarlo direttamente all'assembler. Tutte le indicazioni per l'utilizzazione dell'assembler sono riportate in modo chiaro e dettagliato nell'User's Guide.

Essendo stato scritto dalla Microsoft, il BASIC dell'AIM rassomiglia molto a quello del PET: la maggior parte dei comandi sono identici e l'impiego è pressoché il medesimo. La differenza più importante, è che i sistemi d'uscita dell'AIM 65 sono più limitati di quelli del PET. Fortunatamente, i progettisti hanno pensato che la connessione di una stampante ulteriore, oltre a quella piccola termica a 20 colonne, non sia superflua, e proprio per tale ragione la lunghezza delle linee stampate è programmabile.

La lunghezza massima delle linee del programma BASIC è di 80 caratteri, ma siccome il display non può accettare altro che 60 caratteri, risulta più pratico limitarsi a questo valore massimale.

L'interprete BASIC è in grado di manipolare le matrici e le stringhe di caratteri, permette d'altra parte di utilizzare routines scritte in assembler e di conservare programmi scritti su cassetta magnetica; in breve, si tratta di un BASIC completo, o quasi, visto che manca una caratteristica importante; il poter effettuare delle entrate/uscite sul supporto magnetico e poter manipolare le

INTERVISTA

Per sapere cosa si può fare e non fare con l'AIM 65, siamo andati a intervistare il gruppo incaricato delle applicazioni elettroniche per uso interno alla Borletti a Milano. Queste persone utilizzano l'AIM 65 da molto tempo nell'ambito del controllo industriale ed hanno avuto la cortesia di rispondere alle nostre domande.

Domanda: Quali applicazioni trova presso la Borletti, l'AIM 65?

Risposta: La prima applicazione è stata quella di tester per orologi digitali prodotti da noi e basati su un chip custom. Il progetto è stato realizzato prima attraverso programma sull'AIM e poi applicato su una scheda dedicata. Abbiamo poi fatto una specie di archivio telefonico che chiama e seleziona automaticamente il nome o il numero e si mette in contatto attraverso la linea telefonica. È capace di agganciare la linea e di riconoscere il tono. È una rubrica che memorizza i numeri di telefono e, una volta dato il cognome di chi si vuol chiamare, genera i segnali elettrici necessari.

D. Questo dispositivo è già utilizzato nella vostra azienda?

R. No, non è ancora utilizzato nell'azienda; dovrebbe essere parte di un'apparecchiatura più complessa e si è trattato piuttosto di una prova. L'azienda avrebbe bisogno di una macchina per il controllo automatico delle comunicazioni per la quale serve una specie di database. La nostra realizzazione è un database che accetta come informazione i cognomi, o i numeri di telefono, abbinando le due cose e genera il numero di telefono sulla linea.

D. Ci sono delle apparecchiature basate sull'AIM 65 che funzionano già nell'azienda?

R. Sì, il tester d'orologio, ma due altri progetti verranno utilizzati prossimamente: un sistema di analisi d'immagini per il controllo degli strumenti delle vetture ed una macchina di controllo di una parte dei circuiti di "check-control" per le vetture Fiat. Il gruppo delle applicazioni militari sta lavorando su un sintetizzatore vocale basato sull'AIM, mentre un altro dipartimento sviluppa applicazioni con l'AIM 65 e le riporta su delle schede quando il dispositivo funziona.

D. Utilizzate anche l'AIM a fini didattici?

R. Sì, molti dei nostri designers hanno impiegato l'AIM 65 per familiarizzare con i microprocessori. A questo scopo l'AIM è stato montato in una valigetta 24 ore per dare la possibilità, a chi lo volesse, di istruirsi per conto proprio. La valigetta contiene l'AIM, il registratore, l'alimentatore ed il porta cassette. Può essere portata a casa facilmente consentendo di lavorare in modo completamente autonomo. Queste valigette sono

BITEST

...izzate a scopo didattico e come macchina di test da portare in giro velocemente nelle nostre diverse succursali.

Q. Quali sono i vantaggi, ed eventualmente svantaggi, dell'AIM 65 sul piano didattico?

R. Il punto più importante è che il suo rapporto prezzo/prestazione è migliore degli altri sistemi del genere. Le carenze principali riguardano il monitor che manca d'interfacce con le periferiche. Il meglio in senso assoluto sarebbe la possibilità di gestire un monitor video. A parte questo inconveniente, si può dire che la completezza dell'AIM è un vantaggio considerevole. Costerebbe di più aggiungere una stampante ed una tastiera a un piccolo sistema. Tutto sommato, l'AIM 65 è un sistema molto valido dal punto di vista didattico.

Q. Ritorniamo al tester d'orologio. Come avete fatto per arrivare ai risultati ottenuti?

R. Il nostro orologio è basato su un chip custom. Ha l'ingresso per un quarzo e per l'alimentazione, ha l'uscita per i segmenti dei digits. Il nostro problema era di vedere se l'orologio funzionava e se la composizione delle cifre sul display era corretta, cioè se una certa ora veniva rappresentata realmente come doveva essere. Come abbiamo fatto? Abbiamo semplicemente preso i segnali elettrici della nostra schedina, l'AIM ne ricomponne la cifra risultante e la comparava con quella che sarebbe stata se l'orologio avesse funzionato correttamente. In altre parole, l'AIM andava a vedere se la sequenza dei pacchetti di segnali corrispondenti ai sette segmenti era quella dell'ordine naturale dei numeri, cioè se il numero dei segmenti per 1 era seguito dall'insieme dei segmenti per 2, per 3 e così via. La procedura controllava prima se c'era alimentazione e se alcuni componenti erano difettosi. Poi, l'AIM faceva avanzare l'orologio più velocemente del normale per non dover aspettare il tempo reale che rendeva la prova lunghissima e, ogni volta che l'orologio avanzava di una cifra, l'AIM ricomponne questa cifra e controllava che fosse corretta. In caso di errore, scriveva sulla stampante di che tipo era e dove era accaduto, cioè scriveva l'ora alla quale l'orologio si era sbagliato o identificava eventualmente il pezzo difettoso. In pratica, l'AIM faceva il test del chip montato sulla piastrina.

Q. Avete incontrato delle difficoltà nello sviluppo di questo progetto?

R. C'erano delle difficoltà soggettive nel senso che era la prima volta che facevo un programma un po' impegnativo. Quindi, ho dovuto andare incontro a difficoltà tecniche mie proprie e mi sono trovato di fronte a certe condizioni in cui avevo difficoltà nel trovare una soluzione.

Tabella I - Elenco dei comandi del BASIC 8K dell'AIM 65.

Commenti	Ingresso/Uscita
CLEAR	DATA
CONT	GET
FRE	INPUT
LIST	READ
LOAD	PRINT
NEW	SPC
PEEK	TAB
POKE	
RUN	
SAVE	
Istruzioni di programma	Funzioni stringhe
DEF FN	ASC
DIM	CHR\$
END	LEFT\$
FOR	LEN
GOSUB	MID\$
GOTO	RIGHT\$
IF ... GOTO	STR\$
IF ... THEN	VAL
LET	
NEXT	
ON ... GOSUB	
ON ... GOTO	
REM	
RESTORE	
RETURN	
STOP	
USR	
WAIT	
	Funzioni aritmetiche
	ABS
	ATN
	COS
	EXP
	INT
	LOG
	RND
	SIN
	SGN
	SQR
	TAN

Tabella II - Tempi di esecuzione di alcune istruzioni

Ciclo FOR...NEXT	10 FOR I=1 TO 1000 20 NEXT	11 sec	Divisione frazionaria	10 A=3.14159, B=1.4142 20 FOR I=1 TO 1000 30 C=A/B, NEXT	5.6 sec
Addizione istantanea	10 A=2, B=3 20 FOR I=1 TO 1000 30 C=A+B, NEXT	1.4 sec	Radice quadrata	10 A=3.14159 20 FOR I=1 TO 100 30 B=SQR(A), NEXT	5.2 sec
Addizione frazionaria	10 A=3.14159, B=1.4142 20 FOR I=1 TO 1000 30 C=A+B, NEXT	2.4 sec	Logaritmo	10 A=3.14159 20 FOR I=1 TO 100 30 B=LOG(A), NEXT	2.4 sec
Moltiplicazione istantanea	10 A=2, B=3 20 FOR I=1 TO 1000 30 C=A*B, NEXT	4.2 sec	Costante di stringhe	10 AS=" " FOR I=1 TO 255 20 AS=AS+" ", NEXT	1.6 sec
Moltiplicazione frazionaria	10 A=3.14159, B=1.4142 20 FOR I=1 TO 1000 30 C=A*B, NEXT	5.2 sec	Rimpiazzamento di matrice	10 DIM A(20,12) 20 FOR I=1 TO 20 30 FOR J=1 TO 12 40 A(I,J)=3.14159, NEXT J, I	3.8 sec

"file" di dati di tutti i tipi. Un altro piccolo difetto è che le possibilità di editing sono ridotte al minimo e che l'elaborazione dei programmi tramite il text editor è un poco faticosa.

Il BASIC dell'AIM 65 è uno dei più rapidi, sul mercato attuale. L'AIM 65, munito delle due ROM BASIC, è uno degli elaboratori più completi (con unità d'ingresso e d'uscita),

reperibili a prezzo comparativamente limitato, che può essere programmato con un linguaggio evoluto.

Vi è la possibilità di rimpiazzare le due ROM del BASIC con un compilatore PL/65 che occupa anch'esso 8 Kbyte. Il PL/65 è una "fusione" di PL/1 ed ALGOL, particolarmente adatto ad applicazioni di controllo di processo in tempo reale. Queste caratteri-

stiche facilitano la programmazione modulare; la struttura di controllo dei loop condizionali e iterativi permettono di utilizzarlo per l'elaborazione di programmi ad elevati livelli strutturali. È possibile inserire in un programma scritto in PL/65 dei gruppi di istruzioni in linguaggio macchina quando l'utilizzo di quest'ultimo risulta più semplice oppure consente un'esecuzione più veloce. Il compilatore PL/65 non trasforma i programmi direttamente in linguaggio macchina, ma in codice sorgente per l'assembler AIM 65. Questo permette di ottimizzare i programmi e di correggerne gli errori a livello assembler prima di dare la forma definitiva in linguaggio macchina. La Rockwell propone infine il linguaggio Forth che occupa anch'esso 8 Kbyte. Questo linguaggio è destinato alle applicazioni di comando in tempo reale. La tendenza è quella di rimpiazzare l'assembler in futuro con il Forth nelle applicazioni industriali. Il suo successo è dovuto al fatto che esso offre la semplicità di salto e loop proprie dei linguaggi evoluti pur conservando l'efficacia dell'assembler e del linguaggio macchina. In applicazioni critiche per i tempi di esecuzione, parti del programma possono essere scritte in assembler. Il Forth consente tempi di esecuzione ridotti fino ad un decimo rispetto a quelli dello stesso programma scritto in un altro linguaggio evoluto.

Manuali

Contrariamente a ciò che usano fare alcuni costruttori più negligenti, la Rockwell ha munito il computer di una documentazione molto vasta, e certamente i proprietari dell'AIM 65 approveranno questa impostazione. Senza dubbio, essi troveranno molto pratici i cinque manuali formato "tascabile", i tre riassunti tracciati su fogli pieghevoli ed il grande schema elettrico. L'User's Guide è l'unico manuale esistente in versione italiana. Contiene tutte le informazioni che servono per una precisa conoscenza ed un corretto impiego dell'AIM 65 e delle sue possibilità d'interfacciamento. Il manuale è molto ampio (oltre 380 pagine) ed è ricco di schermi elettrici ed esempi di programmi. Sia per apprendere l'utilizzazione dell'assembler, o del text editor, oppure per avviarsi alla programmazione del 6502, il novizio potrà basarsi completamente sul manuale, che non lo lascerà mai senza risposte. Il principiante, una volta che abbia

Tabella III - Caratteristiche dell'AIM 65

Caratteristiche del monitor

Visualizzazione e modifica dei registri e della memoria
Manipolazione dei 4 break point
Tre funzioni "trace": istruzioni, registri, program counter
Controllo di load e dump su 2 unità a cassetta
Tre funzioni definite dall'utente
Text editor: possibilità di allocazione del buffer di testo in qualunque parte della memoria RAM, inserimento ed eliminazione righe, ricerca e sostituzione stringhe

Caratteristiche dell'assemblatore

Assemblatore simbolico a due passi
Possibilità di assemblaggio da memoria a memoria o da cassetta a cassetta
Accetta costanti decimali, esadecimali, ottali o binarie
Quattro operatori: +, -, < (selezione del byte maggiore), > (selezione del byte minore)
Nove direttive per: riservare e inizializzare locazioni memoria, controllo degli ingressi/uscite dell'assemblatore, presentazione del listing

Caratteristiche del BASIC

Modi operativi: immediato, da programma
Trattamento stringhe
Codifica dei caratteri in ASCII
Rappresentazione dei numeri reali (5 byte), interi (2 byte)
Cifre significative: 9
Rappresentazioni variabili: 2 caratteri significativi
Lunghezza linee programma sorgente: 60 caratteri
Matrici di qualunque dimensione
Editor di linea: possibilità di cancellazione carattere
Livello massimo di nesting delle subroutine: 26
PEEK/POKE
Istruzione di colloquio con il linguaggio macchina
I/O esterno completamente programmabile

Caratteristiche hardware

Microprocessore 6502 a 1 MHz
Memoria RAM: 4 Kbyte massimo on-board
Memoria ROM: monitor: 8 Kbyte
assemblatore (opzionale): 4 Kbyte
BASIC (opzionale): 8 Kbyte
Due porte di I/O parallela a 8 bit
Connettore d'espansione con tre bus di sistema e decodifica per espansione ROM
I/O
Porta per unità a cassetta magnetica

Tastiera

54 tasti
Caratteri e simboli in ASCII
Tasto DEL
Tre tasti per funzioni speciali

Stampante

Stampante termica parallela a 10 elementi
Gestita da software
Larghezza stampa: 20 colonne
Matrice di stampa: 5 x 7 punti
64 caratteri alfanumerici ASCII
Velocità: 120 linee per minuto

Visualizzatore

20 caratteri di 16 segmenti
Possibilità di visualizzazione dei 64 caratteri ASCII

D. Quanto tempo avete impiegato per sviluppare questo progetto?

R. Diciamo che il tempo effettivo di programmazione è stato un mese, lavorando una persona sola. Ci sono state delle difficoltà perché i segnali erano difficili da analizzare e bisognava pulirli.

D. Per questo tipo di applicazione, trovate che l'AIM presenti qualche carenza?

R. La cosa che manca di più è la memoria, quindi abbiamo dovuto aggiungere delle memorie esterne. D'altra parte abbiamo trovato che l'editor presenta una carenza di segnalazione di errori.

D. Come funziona la rubrica telefonica?

R. L'archivio telefonico è praticamente diviso in due programmi separati: uno di acquisizione ed uno di risposta. Il programma è molto colloquiale, è basato su domande e risposte. Il sistema funziona nel modo seguente: all'inizio domanda se si vuol chiamare o inserire dei nomi. In base alla risposta, si pone sulla chiamata o sull'inserimento dei nomi. Per cominciare l'inserimento, chiede se quello che si inserisce è un nome o un numero. A questo punto si possono inserire i dati che l'AIM converte in codice ASCII per poterli mettere in memoria. Il computer va a testare fin dove è stato scritto l'ultimo numero o l'ultimo nome, dopo di che inserisce il dato seguente in coda a questo. Quando l'inserimento di dati è finito, l'AIM controlla se il numero o nome che viene inserito è realmente quello in memoria. Se tutto funziona, ritorna a chiedere se si vuol continuare a inserire, oppure se si vuol chiamare. In base alla risposta, orienta la ricerca sui nomi o sui numeri. Quando ha trovato, lo comunica sul display, se non va bene, si preme un tasto per la ricerca ulteriore, se va bene, compone il numero. Se la linea è occupata, l'AIM può rifare continuamente il numero e andare avanti così fin che trova la linea libera o fino a quando gli si chiede di arrestarsi.

D. Come interfaccia hardware, cosa utilizzavate?

R. Semplicemente un relais collegato a una porta d'ingresso/uscita dell'AIM. Era l'hardware aggiuntivo minimo, tutto era fatto dal software.

D. Che spazio memoria occupava questo sistema?

R. Un totale di 4 Kbyte per il programma e per i dati (nomi e numeri). Ogni Kbyte poteva contenere 16 nomi. Il program-

ma occupava circa $1\frac{1}{2}$ Kbyte.

D. In che cosa consiste il sistema d'analisi d'immagine?

R. Una telecamera con una stringa di fotodiodi inquadra uno strumentino da tavola. Un convertitore analogico/digitale passa i dati a 8 bits all'AIM. Il computer preleva questi dati, studia l'immagine e vede se corrisponde a quello che dovrebbe essere. Per questo, l'AIM stimola attraverso una VIA lo strumento per posizionare la lancetta in una certa posizione. Se la lancetta si mette al posto giusto, la procedura passa allo strumento successivo. Se la posizione non è giusta, l'AIM ha la possibilità di comandare un motorino, del tipo da modellismo, che tara, cioè mette a punto, lo strumento meccanicamente fino a che la lancetta si trova al punto giusto.

D. Avete sviluppato quest'applicazione per eliminare un posto di lavoro?

R. No, non è per questo, è per fare che tutti gli strumenti siano aggiustati nello stesso identico modo. Con una stringa da 128 fotodiodi, si arriva a una precisione dell'ordine del 1/10 di millimetro, risultato che non si può ottenere a occhio. Poiché la telecamera è sempre allo stesso posto rispetto allo strumento eliminiamo anche l'errore di parallasse.

D. Avete altri progetti con l'AIM 65?

R. Desideriamo collegare la macchina che fa il test degli orologi con un sistema di gestione della produzione, vale a dire sapere quanti pezzi vengono prodotti, quanti sono scartati, generare delle etichette d'errore che poi vengono lette automaticamente con un bar code, quindi praticamente, togliere al lavoro della catena di montaggio tutte quelle operazioni ripetitive necessarie per la contabilità del lavoro ma che tecnicamente non hanno nessun contenuto.

D. Qui nel laboratorio avete molti altri computers. Complessivamente, siete soddisfatti dell'AIM 65?

R. Se compariamo l'AIM con sistemi equivalenti, il rapporto prestazioni/costo è uno dei migliori. Praticamente, uno compra la scheda e può partire soltanto con quella in un modo abbastanza completo. È un computer autonomo. Con altre schede che costano circa la metà, non si può fare niente, mentre con l'AIM 65 abbiamo tutto per cominciare. Forse è stato il punto vincente della Rockwell di proporre un sistema di sviluppo così completo e a buon prezzo.

studiato ben a fondo il manuale, potrà iniziare lo sviluppo di programmi già abbastanza interessanti, ad esempio come quello dell'orologio funzionante sulle ventiquattrore, che impiega più di 250 istruzioni e di cui il listing è riportato alla fine del manuale in tre diverse forme (disassemblato, sorgente ed assemblato).

L'"Hardware Manual" comprende, oltre ad una dettagliata descrizione delle caratteristiche funzionali ed elettriche dei chips che costituiscono il sistema, alcuni capitoli che riguardano le tecniche di progetto di sistemi a microprocessore basati sul 6502. Una volta superato il primo impatto con il sistema, questi risultano veramente utili.

Il "Programming Manual", invece, costituisce un corso completo di programmazione nell'assembler del 6502. Il metodo, non certo originale, ma molto efficace, è quello di unire all'esame delle diverse istruzioni quello della struttura interna della CPU.

Il listing del programma di gestione del sistema in linguaggio assembler è presentato sotto forma di piccolo libro detto appunto "Monitor Program Listing". Le istruzioni sono accompagnate da commenti chiari e precisi. Il programma è molto esteso (oltre 3.000 istruzioni) ed anche se non è proprio di facile lettura, costituisce un ottimo esempio di programmazione del 6502, e per un lettore che abbia una certa esperienza rappresenta una fonte inesauribile d'informazioni.

Il "Basic Language Reference Manual", presenta i concetti principali relativi al BASIC e poi studia in una maniera approfondita le istruzioni, che sono raggruppate a seconda delle loro funzioni. Questo manuale si rivela pratico e semplice da impiegare. Tuttavia, non si tratta di un libro per chi non sa proprio nulla, ed anzi, prima di prenderlo in esame, è bene avere già una piccola conoscenza di base, nel campo della programmazione.

L'AIM 65 è fornito con tre dépliant, che non appena l'operatore si sarà un poco familiarizzato con il sistema, saranno utilizzati più di frequente che non i manuali. La "R6500 Microprocessor Programming Reference Card", contiene tutti gli elementi che si devono conoscere per programmare il 6502, ed in particolare il set d'istruzioni. La "AIM 65 Summary Card" rammenta tutti i comandi del monitor, del text editor e dell'assembler. Nella mappa, si osservano i riassunti delle subroutine del monitor e gli elementi principali della programmazione del VIA 6522. La "AIM 65 8K BASIC Reference Card", presenta in modo succinto in assie-

me le istruzioni BASIC.

La dotazione dei manuali è quindi ampia ed esauriente, non solo nei confronti dell'AIM 65 nella sua configurazione di base, ma soprattutto se si punta alla sua espansione ed all'interfacciamento con il mondo esterno. L'unico appunto che si deve muovere, è purtroppo la presenza di numerosi errori di stampa che mettono a dura prova la pazienza del lettore, e che la versione italiana dell'"User's Guide" non è riuscita ad eliminare completamente.

Conclusione

Dal punto di vista dell'hardware, l'AIM 65 è uno dei rari sistemi che mettono a disposizione dell'utente una tastiera ASCII, un dispositivo di visualizzazione, una stampante e due porte d'ingresso/uscita programmabili. Al livello di software, la dotazione di un monitor "muscoloso", di un editor di testo e di un assembler, fanno dell'AIM un sistema ideale per lo sviluppo dei programmi in assembler considerando anche che la disponibilità di linguaggi ad alto livello come il BASIC, il FORTH, il PL65, così come il PASCAL rappresenta una caratteristica preziosa per elaboratori di questo genere. Malgrado il suo aspetto un poco "precario", dovuto principalmente alla mancanza di contenitore, l'AIM 65 è quindi un elaboratore, in verità, eccezionalmente completo. Si può dire che il sistema abbia due tipi principali di applicazioni: i controlli industriali e le applicazioni didattiche. Grazie al suo firmware sofisticato, l'AIM 65 costituisce un ausilio di sviluppo molto pratico per lo studio di piccoli sistemi specializzati che si basino sulla famiglia 6500. In più, la semplicità d'interfaccia con il mondo esterno, facilita l'impiego dell'AIM 65 come elemento centrale di piccoli sistemi automatizzati di controllo industriale. Gli utilizzatori professionisti, lo possono considerare come un ausilio di valutazione a basso costo, ma certo, se lo vogliono raccordare ad un gruppo di controllo di dischi floppy e ad una stampante che possa scrivere delle righe più lunghe, possono ottenere un sistema dalle applicazioni più complesse. L'AIM è fornito con una documentazione dalla qualità eccezionale, notevolmente ben organizzata, e considerando che è equipaggiato con tutte le unità d'ingresso/uscita necessarie per ciascuna utilizzazione individuale, si può dire che sia particolarmente adatto per il primo approccio

Costruttore:

Rockwell International Corp.
3310 Miraloma Avenue
P.O. Box 3669
Anaheim, California 92803
USA

Distributore per l'Italia:

Dott. Ing. Giuseppe De Mico
Viale Vittorio Veneto, 8
20060 Cassina De' Pecchi

Listini prezzi AIM 65

AIM 65 (1 Kbyte RAM)	L. 594.000 + IVA (*)
AIM 65 (4 Kbyte RAM)	L. 660.000 + IVA (*)
Assembler	L. 132.000 + IVA (*)
BASIC	L. 155.000 + IVA (*)
PL/65	L. 222.000 + IVA (*)
FORTH	L. 222.000 + IVA (*)
Alimentatore separato	L. 80.000 + IVA
Back panel	L. 105.000 + IVA
Bus driver	L. 95.000 + IVA
Contenitore	L. 95.000 + IVA
Alimentatore per contenitore	L. 85.000 + IVA
8 Kbyte RAM statica + VIA 6522	L. 360.000 + IVA
EPROM programmer con software	L. 190.000 + IVA
Extender BASIC (cassetta)	L. 50.000 + IVA
Rilatore BASIC	L. 150.000 + IVA
Interfaccia video	L. 345.000 + IVA
I/O optoisolate	L. 235.000 + IVA
A/D 12 bit, 35 microsec.	L. 335.000 + IVA
A/D, D/A 8 bit	L. 250.000 + IVA
Minifloppy controller con software	L. 430.000 + IVA

(*): Questi prezzi sono legati alla parità del dollaro

con un sistema a microprocessore da parte dei novizi, ed anche per l'insegnamento della programmazione. Il costo del sistema è relativamente basso, e il complesso può soddisfare le pretese di coloro che in sostanza vogliono avere a disposizione un elaboratore completo. Tra l'altro, la stampante a 20 colonne, risulta molto pratica per apprendere la programmazione in assembler. Non è comunque escluso l'impiego del cacciavite; almeno, per collegare l'alimentazione che manca, ed in via generale, l'u-

tente si deve abituare ad una certa interdipendenza tra software ed hardware, se vuole giungere al massimo delle prestazioni ottenibili. Ciò è tutt'altro che un male, a ben vedere, considerando che l'impiego dell'AIM 65, sul piano logico, deve essere compreso a fondo, ma teoria a parte, coloro che si porranno il problema di estenderlo e migliorarlo, potranno poi far parte del ristretto gruppo di persone che si possono vantare di conoscere proprio a fondo la natura di un computer.