

Le prove di Jurassic News

Rockwell - AIM 65/20

C'è stata una stagione, alla fine degli anni '70, dove la dizione "Personal Computer" non aveva significato e quello che si pensava potesse servire era semplicemente un sistema per imparare ad usare il micro processore.



Contesto storico

Siamo nell'anno 1976 e il computer personale non è ancora una idea acquisita. Bisognerà aspettare la geniale mente di Wozniak per avere a breve l'Apple 1. I microprocessori vengono prodotti ma in realtà non si sa che cosa farne. Non che le idee non ci siano, ma questo chip è troppo complicato e le sue potenzialità sono ancora in fase di studio da parte dei progettisti.

In questo scenario hanno assunto estrema importanza i sistemi di sviluppo messi a punto dalle stesse ditte produttrici dei chip con lo scopo dichiarato di favorire la com-

prendimento dell'elettronica digitale supportata dal calcolatore ai nuovi progettisti.

Non è che non esistessero prima i sistemi di sviluppo, i microprocessori e la documentazione, solo che era estremamente complicato usarli. Ad esempio l'8080 di Intel richiedeva una serie di chip di supporto indispensabili a far funzionare un circuito e la "potenza" non è che fosse così elevata. L'altro source, la Motorola, produce il 6800; ottimo chip ma costoso e anch'esso complicato.

Chi produce il 6502, derivato direttamente dal 6800, ha necessità di far conoscere le potenzialità

Una bella immagine prospettica del sistema di sviluppo della Rockwell; AIM 65 o R6500.

del nuovo chip ed ecco che nascono i vari Kim-1 dalla MOS Technology e appunto l'AIM 65 dalla Rockwell International con sede ad Anaheim in California, seconda source per la produzione del chip.

Una persona all'epoca impiegata presso Rockwell, tale Mark Reardon, sostiene che la ditta prevedeva di vendere dai 400 al 600 sistemi ma che alla fine furono più di 50.000 (fonte: old-computers.com Museum ~ Rockwell AIM 65).

Il prezzo del computer con 4 Kb di RAM si aggira attorno ai 375 dollari, mentre una versione "Light", senza tastiera e stampante, viene venduta attorno ai 175 dollari.



Primo approccio

L'AIM 65 è un computer single-board che viene venduto "nudo", cioè senza cabinet (che apparirà qualche tempo dopo) e senza alimentatore. L'utente compra in pratica una piastra elettronica densa di componenti, una tastiera alfanumerica di tipo telescrivente, da collegare alla piastra madre con un cavo piatto di pochi centimetri e una pletora di documentazione.

A questo punto ci si deve preoc-

cupare dell'alimentazione, per la verità non tanto difficoltoso come compito e soprattutto di imparare ad utilizzarlo questo sistema, con l'aritmetica esadecimale, le istruzioni mnemoniche, gli indirizzi di memoria, etc...

L'output è assicurato attraverso un display a led di un'unica riga da 20 caratteri e da una stampantina termica delle stesse caratteristiche cablata direttamente sulla piastra madre.

Rispetto alla concorrenza, l'AIM 65 gode di una maggiore integrazione e professionalità che si tra-

Il sistema visto da sopra



In questa pagina una breve rassegna di contenitori approntati per rendere il sistema più simile ad un vero calcolatore, oltre che proteggere l'elettronica da danni accidentali.



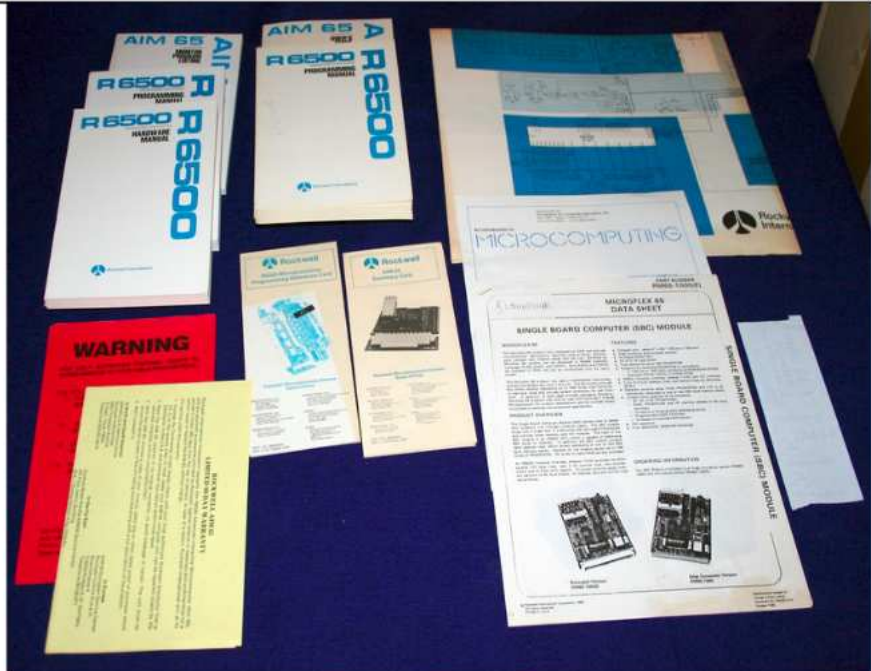
duce nell'avere una vera tastiera per l'input dei dati, al posto della scomoda tastierina esadecimale dei sistemi di sviluppo concorrenti. Oltre a questo l'ampiezza del display, non limitato ai quattro byte di indirizzo e due byte per i dati, offre maggiori possibilità di editing e di esecuzione del codice. Infine poter ottenere su carta il listing/dump della memoria è ulteriormente gradito.

La seconda cosa di cui preoccuparsi, dopo aver costruito l'alimentatore e scoperto come alimentare il sistema, è quello di trovare un panno morbido ma non troppo e soprattutto non conduttore, per appoggiarvi sopra tastiera e piastra madre. Un'altra accortezza è prevedere un cavo di collegamento con la tastiera un attimino più lungo dei miseri sei centimetri di quello originale, permettendo un posizionamento meno rigido delle due unità una dall'altra.

A questo punto avrete già letto il manuale e scoperto che i due connettori a pettine che sporgono dal retro della piastra si chiamano "Application" e "Expansion".

La piastra supporta nativamente due registratori audio compreso un segnale per l'azionamento dei motori di trascinamento della cassetta e una telescrivente in current loop da usare al posto della tastiera e della stampante nativa.

La qualità della tastiera è ottima, garantendo un feedback alla digitazione non stancante, anche dopo ore di utilizzo. Qualche riserva sul display per via della luminosità non



esaltante e della dimensione dei caratteri forse troppo minuscola. La stampante, pur non essendo in tecnologia ad impatto, è comunque abbastanza rumorosa perché utilizza una tecnica di "scarica ad alta tensione" per bruciare la carta e far apparire i caratteri.

La carta termina da utilizzare per la stampante, un rotolo è contenuto nella confezione acquistata, è abbastanza reperibile perché usata anche da alcuni modelli di calcolatrici da tavolo e registratori di cassa, ma è costosa e certo non da utilizzare con leggerezza se non per un listing

L'abbondanza della documentazione fornita in dotazione.

La pagina pubblicitaria più classica dell'importatore italiano, apparsa su tutte le riviste di settore.

AIM 65... come vi pare

LOTUS/Milano
(02) 2502095
AL SOL/Milano
(02) 2155822
DETAS/Brescia
(030) 302304
MICROWAY/Veneta
(045) 218143
RO ELETTRONICA
Mestre (VE)
(049) 875578
LINT/Medea
(059) 300303
MCC/Livorno
(0586) 408112
ALESSI/Romano (LI)
(0595) 38090
ALDORE/Benedetto
Del Treno (AP)
(0735) 850902
PANTALONI/Roma
(06) 272902
THYRISTOR/Catania
(095) 444051
B&S ELETTRONICA
Gorizia
(0481) 32193
RTE ELETTRONICA
Padova
(049) 605710
ELVER ELETTRONICA
Bari (NO)
(0825) 85227

RIVENDITORI AUTORIZZATI
GOMA/Torino
(011) 7482147
C.T.S. ELECTRONICS
IVREA (0125) 40364
GARGIOLA/Genova
(010) 892397

Linguaggi: ASSEMBLER, BASIC, PL/1, FORTRAN, Esapoint di memoria, RAM statica e dinamica, di memoria ROM, EPROM, EEPROM, programmi.
A/D, D/A converter, interfaccia CRT, floppy, minifloppy, cassette, Scheda I/O dedicate, opzionale, general purpose.

Dot. Ing. Giuseppe De Mico S.p.A.
20060 Cassina De' Pecchi - V.le Vittorio Veneto, 8
Tel. (02) 952055/9520551 (10 linee)
USCJ regionali: Torino/Padova/Rovigo/Firenze/Roma

finale, piuttosto che attivare il semplice eco del video che riporta tutto quanto appare sul display anche su carta.

Assieme alla macchina arrivano quattro manuali che sono:

- AIM 65 User' s Guide
- AIM 65 Monitor Program Listing
- R6500 Microcomputer System Programming Manual
- R6500 Microcomputer System Hardware Manual

Inoltre altra documentazione e informativa la troviamo negli allegati:

- R6500 Programming Reference Card
- AIM 65 Summary Card
- Warranty Card

Dai manuali si capisce che questo prodotto viene chiamato indifferentemente AIM65 o anche "R6500 Microcomputer", intendendo per la Rockwell essere un vero e proprio sistema di sviluppo per la sua soluzione di micro calcolo ma soprattutto micro-controllo.

La lettura di questi manuali è una vera miniera di informazione che può costituire autonomamente un corso pratico di elettronica digitale, anche se una infarinatura è meglio averla già prima...

Hardware

La CPU è un Rockwell mp6502 funzionante a 1 Mhz montato su zoccolo; per le caratteristiche del 6502 questo significa che una istruzione semplice viene eseguita in due microsecondi. Assieme alla CPU la piastra ospita 4 Kbyte di RAM statica (ma ne esiste una versione "Light" che porta a bordo un solo K), e 4 Kbyte di ROM contenente il monitor di sistema. La ROM è espandibile fino a 20 Kb grazie alla presenza di cinque zoccoli in grado di ospitare ROM da 4Kb e che permettono di avere un ottimo sistema comprensivo di editor, assembler simbolico e disassembler. La versione venduta in Italia ha già 4 Kb di RAM e monitor espanso a 12 Kb con editor, assembler e disassembler. Esiste anche una ROM BASIC da 8 Kb da usare al posto della ROM assembler e altre con altri linguaggi come il PL/65 (una variante del PL/1), il PASCAL, il FORTH, etc... che richiedono però una maggiore espansione di RAM, ottenibile solo via card esterna, per essere utilizzati con qualche profitto.

Le due porte di I/O sono già predisposte per una interfaccia a cassette magnetiche e per un terminale TTY. A questo proposito sulla piastra è presente uno switch che esclude le periferiche interne (display e stampante) in favore di un terminale seriale.

Un pulsante di Reset è accessibile sulla piastra in prossimità della tastiera.

Come chip di supporto il sistema porta un 6532 per il controllo del timing della memoria, un VIA (Versatile Interface Adapter) 6522 e un PIA 6520 (Peripheral Interface Adapter).

La stampante a venti colonne è in tecnologia termica in grado di stampare un carattere formato da una matrice 5x7 con riga di stampa di 20 caratteri (come la riga del display) su un nastro di carta continuo di larghezza 2 pollici e un quarto.

Il display a led è costituito da unità a 16 segmenti, il che permette di riprodurre tutti e 64 i caratteri del set ASCII standard.

La tastiera è una QWERTY a 54 tasti e comprende il codice ASCII più tasti di controllo (Control, Shift, Del, Escape) e lascia liberi tre tasti (siglati F1, F2 e F3) per la definizione di funzioni utente.

L'alimentazione non è fornita con la macchina (ma ne viene offerto uno come opzione se si preferisce non costruirselo), consiste in due tensioni: +5Volt per l'elettronica (2 Ampere) e +24Volt per la stampante termica (2.5 Ampere) ed usabile anche come linea di alimentazione per altre unità collegate attraverso le porte di espansione.

Il connettore di alimentazione presente sulla piastra, dotato di una



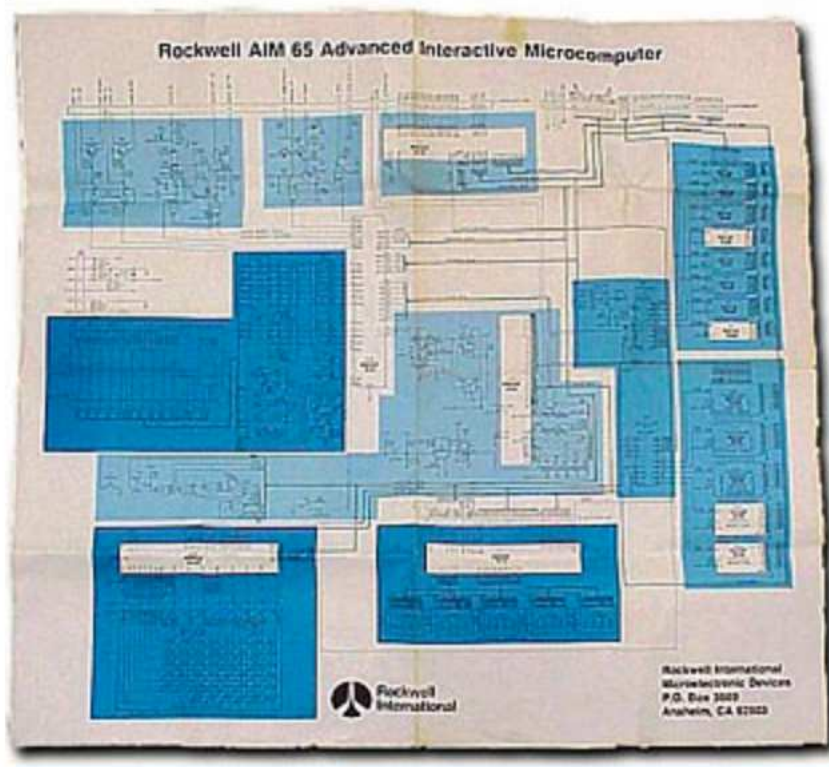
morsettiera robusta, riporta anche i connettori per le tensioni +12 e -12 Volt, da usarsi nell'eventualità si usino chip con queste necessità, ad esempio EPROM.

La struttura del progetto è sufficientemente espansa da consentire l'approntamento di esperimenti di programmazione e controllo anche abbastanza sofisticati, grazie alle due porte da 8 bit programmabili, il contatore digitale e gli slot di espansione che portano all'esterno tutti i segnali dei bus di sistema. L'esposizione dei segnali all'esterno non è protetta da chip di buffering, cosa che sarà prudente attuare a fronte dell'uso con schede di espansione

Un particolare della cop-pia tastiera-display.

(Sotto) la piadinatura del micro processore 6502

VSS	1	40	RES
RDY	2	39	ϕ_2 (OUT)
ϕ_1 (OUT)	3	38	S0
IRQ	4	37	ϕ_0 (IN)
N.C.	5	36	N.C.
NMI	6	35	N.C.
SYNC	7	34	R/W
VCC	8	33	D0
A0	9	32	D1
A1	10	31	D2
A2	11	30	D3
A3	12	29	D4
A4	13	28	D5
A5	14	27	D6
A6	15	26	D7
A7	16	25	A15
A8	17	24	A14
A9	18	23	A13
A10	19	22	A12
A11	20	21	VSS



La "mappa" hardware del sistema evidenzia i vari blocchi funzionali per la facile identificazione delle componenti.

Il display a 16 segmenti prevede una configurazione dei led come quella riportata in figura.



di nostra progettazione.

Limitata anche la possibilità nativa di espandere la RAM che richiede all'occorrenza la predisposizione (o l'acquisto) di una apposita scheda in grado di codificare gli indirizzi necessari, oltre a portare a bordo i chip di memoria statica.

I due connettori di espansione posti sul retro della piastra assomigliano sorprendentemente, anche nel nome, a quelli del KIM-1, un progetto analogo della MOS Technology. Infatti sui manuali vengono dichiarati compatibili con quelli del KIM, salvo poche e documentate differenze. Probabilmente le due società fornitrici del 6502 si sono accordate su questo punto o più semplicemente il progetto è stato elaborato da un team comune e poi differenziato nei particolari e nel target di riferimento (l'AIM65

costa sensibilmente di più di un KIM-1).

Il pilotaggio dei due registratori a cassetta che sono collegabili alla macchina prevede dal punto di vista del segnale due pin di uscita con un livello basso e un livello alto, da scegliere secondo le caratteristiche del circuito audio di uscita. Prevede inoltre un segnale per pilotare il motore del registratore che può essere quindi messo in moto da programma. Questo particolare deve essere controllato dall'utilizzatore perché la corrente di commutazione potrebbe caricare eccessivamente l'alimentatore dell'AIM e quindi essere consigliabile adottare una soluzione che prevede il pilotaggio di un relè con il segnale dell'AIM in modo da disaccoppiare la periferica.

Data la necessità di disporre di maggiore RAM, oltre che di avere a disposizione altre periferiche, quasi subito sono apparse delle schede di espansione che l'AIM65 ha in qualche caso condiviso con il KIM-1, altro sistema single board basato sul 6502 e quasi compatibile, come affermato poco sopra. Le più diffuse sono state prodotte da una certa MTU come ad esempio la scheda "visible memory" o la scheda grafica per pilotare un monitor. E' stato prodotto addirittura un sintetizzatore musicale oltre ad altre cosucce come programmatori di EPROM, box di espansione,

vari sensori e attuatori esterni utili in esperimenti e per tenere sotto controllo parametri fisici come temperatura, tensione, etc...

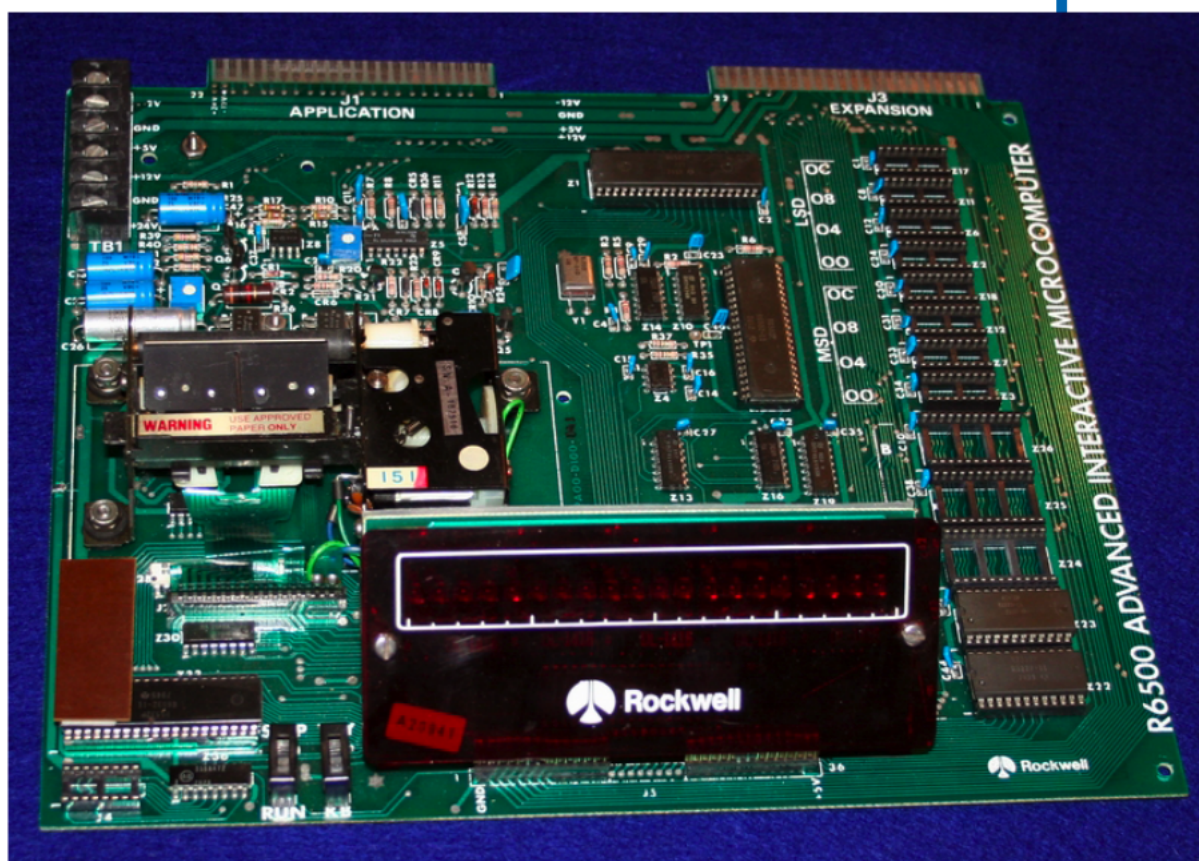
Non è un caso che la piastra AIM65 sia stata usata spesso come supporto ad esperimenti scientifici che prevedevano la misurazione e l'elaborazione dei dati; soprattutto si vede apprezzata la possibilità di pilotare i registratori con comando di accensione e spegnimento dei rispettivi motori di trascinamento cassette. Pare niente ma immaginate di dover immagazzinare il risultato di misure periodiche fatte da qualche strumento interfacciato con il vostro AIM65; potreste, ogni tanto, accendere uno dei registratori e registrarvi i dati letti e magari nei tempi morti accendere l'altro per leggere ed elaborare altri dati...

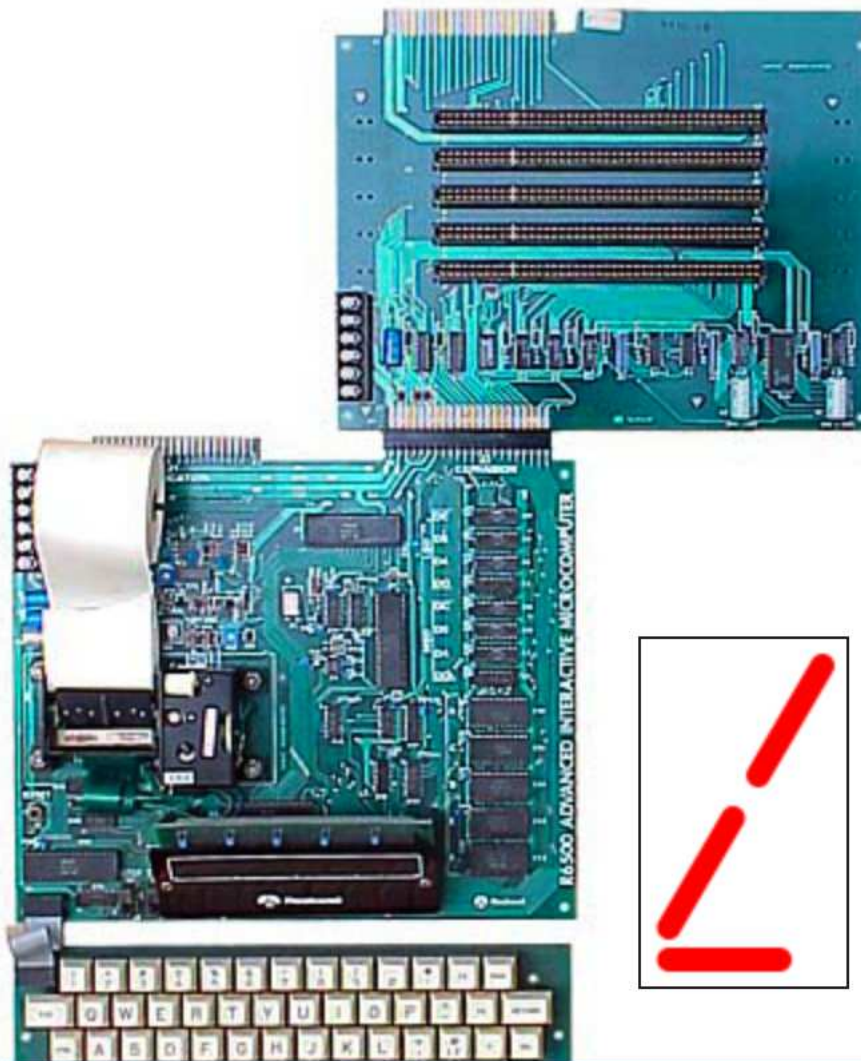
Espansioni

Ne sono state progettate e commercializzate parecchie, principalmente dalla stessa Rockwell, ma anche molte di origine più obbistica.

Quasi indispensabile l'estensione del bus in grado di accogliere più schede in una configurazione meccanica. Si tratta del "AIM 65 Expansion Motherboard", costruito ufficialmente dalla Rockwell è in grado di ospitare tutte le schede costruite dallo stesso produttore.

Una curiosità l'espansione che ospita una memoria a bolle da 64 Kbytes, una tipologia di storage non volatile che sembrava potesse avere un futuro più radioso di quello che gli è toccato nella realtà.





quindi l'AIM65 da sistema di sviluppo in sistema di elaborazione personale. La scheda ha a bordo 64 Kb di RAM e una PROM da 256 byte in grado di "boot-are" dal disco il relativo CP/M.

Uso

Dopo aver connesso l'alimentatore ci si deve assicurare che lo switch KB/TTY sia in posizione Kb (tastiera) e che lo switch (RUN/STEP) sia in posizione RUN. A questo punto si può dare tensione e verificare che la procedura di boot sia eseguita correttamente.

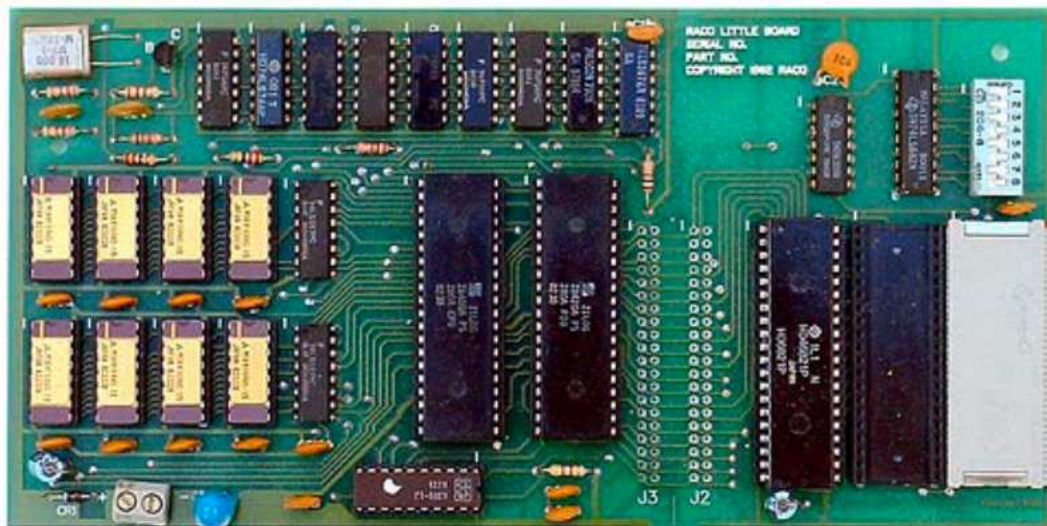
Dopo un lampeggiamento del display se tutto è OK, il sistema stampa e visualizza sul display il messaggio "ROCKWELL AIM 65". A questo punto con il tasto <ESC> si attiva il monitor di sistema che ha come prompt il carattere '<' (non proprio questo, ma nell'intenzione...), che si trasformerà in '>' dopo l'inserimento del primo carattere in modo che il singolo carattere iniziale della riga rimanga isolato ad individuare un comando dato al monitor.

Piastra base più bus di espansione (un insieme decisamente ingombrante...).

Nell'inserito il carattere prompt "<".

(Sotto) la scheda Z80 per usare il CP/M e relativi programmi.

Addirittura si è prodotta da una certa RACO, che starebbe per Rockwell Anaheim Computer Organization, una scheda con Z80 e relativo CP/M (foto in fondo alla pagina). Evidentemente una scorciatoia per dotare il sistema di un ambiente applicativo compatibile con molto software e trasformare

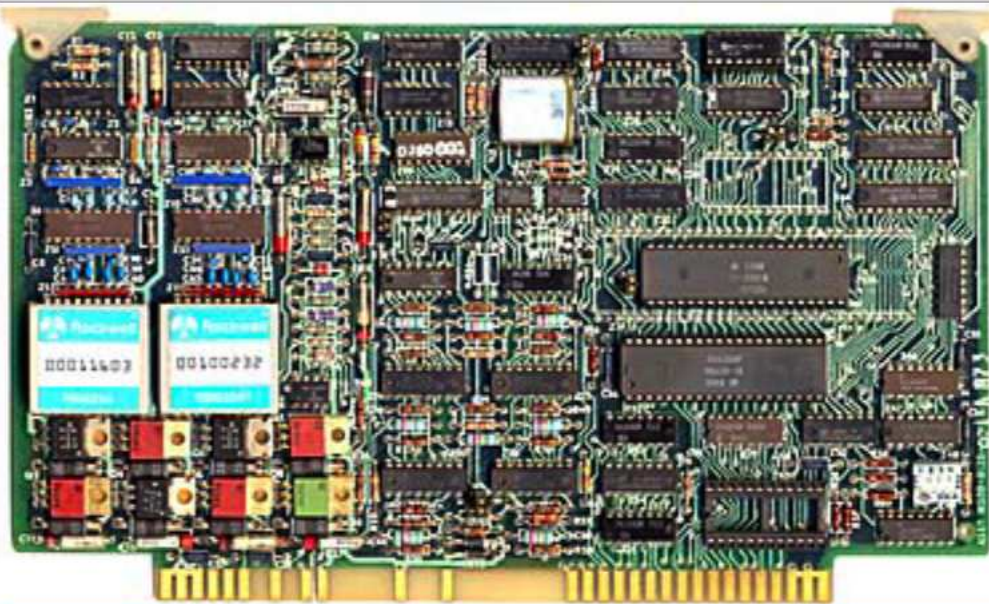


A questo punto sono attivi i comandi del ricco monitor di sistema che permettono un discreto controllo delle periferiche e dell'esplorazione della memo-

ria ed esecuzione dei programmi residenti in ROM.

Scopo principale di chi acquista l'AIM65 è probabilmente quello della programmazione in linguaggio macchina. Ecco dunque che l'AIM65 nella configurazione "standard" mette a disposizione un editor e un assembler che permette di creare il sorgente usando le istruzioni mnemoniche del processore piuttosto che i codici esadecimale direttamente inseriti in ROM. Da un certo punto di vista la programmazione esadecimale diretta è più agile per chi appronta piccole routine, ma indubbiamente è più ostica e non può godere di quelle features che sono tipiche di un linguaggio più vicino all'utente. Disporre di un sorgente permette infatti di intervenire con correzioni, cancellazioni o inserimento di istruzioni senza ricorrere ai salti mortali (o alla proliferazione delle istruzioni NOP) tipici di chi si deve "arrangiare" con il codice macchina. Pensiamo ad esempio alla comodità delle label e dei salti relativi.

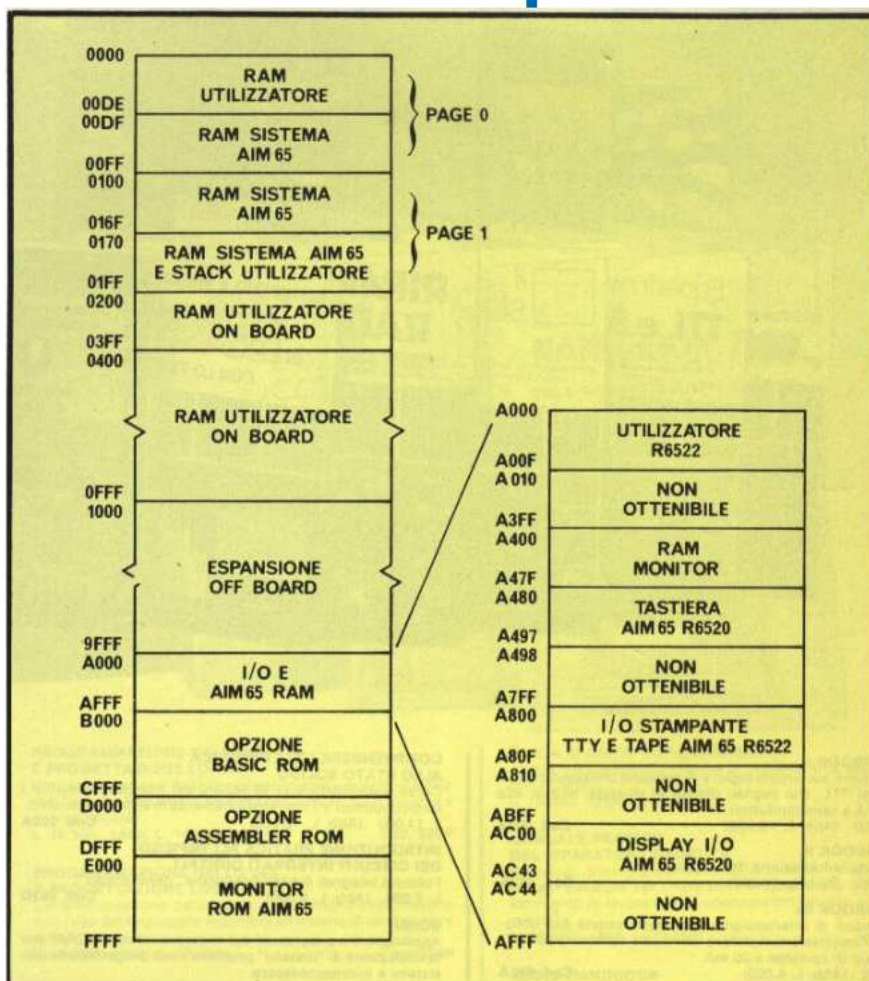
L'assembler fornito in 4 Kb di ROM è del tipo "a due passate"; nel primo step il sorgente viene scandito per creare la cosiddetta "symbols table", la tabella dei simboli, nella seconda passata si produce il codice macchina pronto

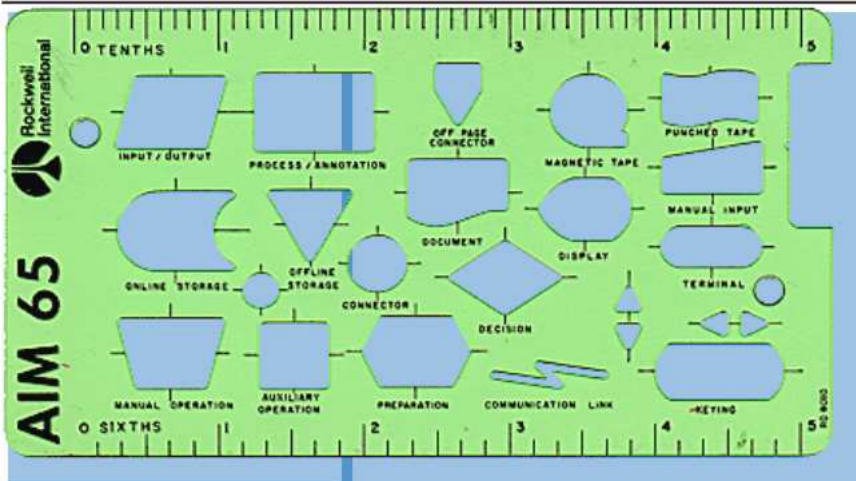


per l'esecuzione.

Con appena 4 Kb di RAM non è che si possano fare i salti mortali, ma si sa che la programmazione in assembly è adatta a piccoli progetti e non certo a migliaia di righe di codice. Disponibili all'utente non sono nemmeno tutti e quattro i Kb di memoria, dal momento che le due

(Sopra) la scheda di memoria a bolle e (sotto) la mappa di memoria, informazione indispensabile per chi programma con l'assembler.





La Rockwell ha pensato a tutto: addirittura il normografo con i simboli da flowchart.

prime pagine (la zero e la 1, cioè gli indirizzi da 0x0000 a 0x01FF) sono riservate. Il microprocessore 6502 sappiamo che non è particolarmente ricco di registri interni ma ha dei modi di indirizzamento in pagina zero (i primi 256 byte di memoria) che di fatto permettono di usarla come dei veri e propri registri interni.

Troviamo interessante la possibilità di pilotare due registratori a cassetta direttamente da programma. Questo può aiutare a mettere a punto un sistema per l'acquisizione automatica di segnali, magari a fronte del verificarsi di una qualche situazione che genera un interrupt, segnali che vengono poi

scaricati su cassetta per la successiva elaborazione. Questo tipo di utilizzo della piastra della Rockwell è stata riportata in numerosi articoli apparsi su riviste specializzate attorno agli anni '80.

Da non sottovalutare nemmeno la disponibilità di linguaggi di alto livello come il principe di tutti: il BASIC. Il BASIC a corredo (opzionale) dell'AIM65 è una versione Microsoft standard, secondo la tradizione della ditta di Bill Gates, con la possibilità aggiuntiva di pilotare la stampantina interna e programmarsene delle entry per la gestione delle periferiche eventualmente usate come espansione.

Per quanto riguarda l'utilizzo di altri interpreti/compiler riteniamo che non siano stati in testa ai desideri di acquisto da parte degli appassionati. Ogni software aggiuntivo, anche per il fatto che viene rilasciato su ROM, ha un certo costo: dalle 100 alle 150 mila lire in Italia, il che non è proprio pochissimo per una macchina che originariamente si aggira attorno alle 600.000 lire dall'importatore ufficiale.

Pieghevole con le istruzioni e i comandi dell'interprete BASIC.

ARITHMETIC FUNCTIONS

Statement	Syntax/Function	Example
ABS	ABS(expression)	$Y = ABS(A * B)$
ATN	ATN(expression)	PRINT ATN (X)
COS	COS(expression)	$A = COS(2.3)$
EXP	EXP(expression)	$B = EXP(C)$
INT	INT(expression)	$C = INT(D * 3)$
LOG	LOG(expression)	$D = LOG(Y * 2)$
RND	RND(parameter)	$E = RND(1)$
SIN	SIN(expression)	$B = SIN(X + Y)$
SQR	SQR(expression)	$C = SQR(D)$
TAN	TAN(expression)	$D = TAN(D * 14)$

BASIC INSTRUCTION FORMAT

Where: 00 = Line address; i.e., start of new statement line
A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z = Statement line number, in binary
XX, YY, ZZ = Token (delimiter) code and data in ASCII and BASIC encoded symbols

TABLE OF ERROR CODES

Code	Error
BS	Bad syntax
CH	Character overflow
DD	Double dimension
FC	Illegal function call
FD	Illegal direct
LS	String too long
NP	Next without FOR
OD	Out of data
OM	Out of memory
OV	Overflow
RG	RETURN without GOSUB
RS	String error
ST	String terminator
TM	Type mismatch
UF	Undefined function
US	Unqualified statement
VB	Division by zero

MESSAGES

NEED PROMPT - A non-decimal character was entered in response to an INPUT function. Re-enter the entire number.
EXTRA IGNORED - An extra parameter was entered in response to an INPUT function.
BREAK - Break command (F1) or STOP command was executed.

ZERO PAGE PARAMETERS

Parameter	New Address	Old Address
USR Routine address (L, H)	04, 05	04, 05
Line width	12	16
Input Buffer	14, 00	20, 00
Pointer to program start (L, H)	23, 24	115, 116
Pointer to variable start (L, H)	75, 76	117, 118
Pointer to array start (L, H)	117, 118	119, 120
Pointer to top of user memory (L, H)	119, 120	121, 122
Pointer to top of user memory (L, H)	122, 123	123, 124
Pointer to top of user memory (L, H)	123, 124	125, 126
Stacking point accumulator	125, 126	126, 127
Stacking point accumulator	126, 127	127, 128
Stacking point accumulator	127, 128	128, 129
Stacking point accumulator	128, 129	129, 130

Rockwell AIM 65 8K BASIC Reference Card

Commands	Input/Output
CLEAR	DATA
CONT	INPUT
END	PRINT
LIST	READ
LOAD	SPC
NEW	TAB
PEEK	String Functions
POKE	ASC
RUN	CHR
SAVE	GET
	LEN
	LEFT
	MID
	RIGHT
	STR
	VAL
	Arithmetic Functions
	ABS
	ATN
	COS
	EXP
	INT
	LOG
	RND
	SIN
	SQR
	TAN

Rockwell Microelectronic Devices Sales Offices

WESTERN REGION U.S.A.	CENTRAL REGION U.S.A.
3000 Wilshire Blvd. P.O. Box 399 Beverly Hills, CA 90207 Phone (310) 206-2000	10000 Wilshire Blvd. Suite 1000 Beverly Hills, CA 90207 Phone (310) 206-2000
EASTERN REGION U.S.A.	EUROPE
10000 Wilshire Blvd. Suite 1000 Beverly Hills, CA 90207 Phone (310) 206-2000	Rockwell International (Europe) Ltd. 10000 Wilshire Blvd. Suite 1000 Beverly Hills, CA 90207 Phone (310) 206-2000
HONG KONG REGION U.S.A.	
10000 Wilshire Blvd. Suite 1000 Beverly Hills, CA 90207 Phone (310) 206-2000	
DOC. NO. 29650N05 MAY 1979	

Software

Dopo aver trattato dell'Assembley, linguaggio assolutamente d'elezione per un sistema di

sviluppo di questo genere, viene la naturale curiosità di esplorare le possibilità di programmazione offerte dagli altri idiomi messi a disposizione dalla Rockwell. In particolare il BASIC e il PL/65.

Per quanto riguarda il BASIC, si tratta di un classico "alla Microsoft"

Un linguaggio interessante, perché "esoterico" rispetto al classico BASIC, è il PL/65, creato dalla stessa Rockwell al fine di mettere a disposizione un agile idioma per la programmazione di interfacce di controllo cui l'AIM65 costituisce il core.

Si tratta di un mix fra il PL/I e l'ALGOL, due linguaggi implementati da alcuni anni sulle macchine mainframe e mini dipartimentali. Non è il caso di dilungarci troppo nella descrizione dei due linguaggi "genitori" del PL/65, peraltro ben documentati su Internet, tuttavia un piccolo assaggio è utile per capire le origini dell'implementazione fatta dalla Rockwell.

Ecco un esempio di procedura scritta in PL/I:

```
FINDSTRINGS: PROCEDURE
OPTIONS (MAIN);
```

```
DECLARE
PAT VARYING CHARACTER(100),
LINEBUF VARYING
CHARACTER(100),
(LINENO, NDFILE, IX)
FIXED BINARY;
```

```
NDFILE = 0;
ON ENDFILE(SYSIN) NDFILE=1;
```



```
GET EDIT(PAT) (A);
LINENO = 1;
DO WHILE (NDFILE=0);
GET EDIT(LINEBUF) (A);
IF LENGTH(LINEBUF) > 0
THEN DO;
IX = INDEX(LINEBUF, PAT);
IF IX > 0 THEN DO;
PUT SKIP EDIT
(LINENO, LINEBUF) (F(2), A);
END;
END;
LINENO = LINENO + 1;
END;
END FINDSTRINGS;
```

La funzione *FINDSTRINGS* riceve in input (file *sysin*) una serie di righe di testo e le stampa in output precedute dal numero di riga.

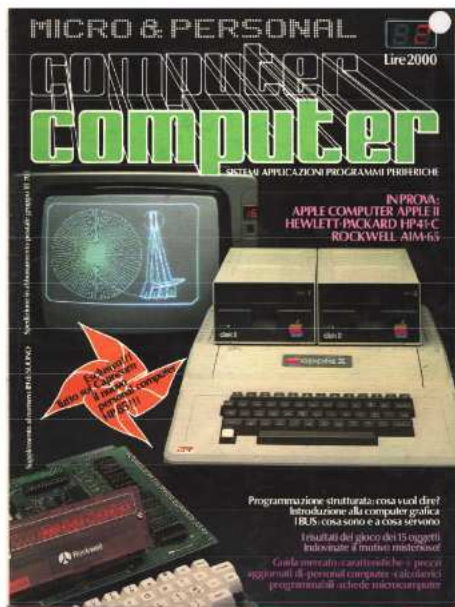
Nel PL/I è molto interessante ed appare forse per la prima volta l'idea di trattare l'input come flusso di byte registrati su file.

Per quanto riguarda l'ALGOL vediamo qui un piccolo esempio:

```
procedure Absmax(a)
Size: (n, m) Result: (y)
Subscripts: (i, k);

value n, m;
array a;
integer n, m, i, k;
real y;
```

Oltre alla documentazione "nativa", l'R6500 ha "goduto" di numerose monografie che direttamente o indirettamente ne hanno esaminato e presentato le caratteristiche.



Le riviste di informatica in Italia hanno cominciato ad uscire proprio quando l'AIM65 era sul mercato. Qui la copertina del numero 2 della rivista "m&p microcomputer" che contiene la recensione assieme a quella dell'Apple II. La data è il 1981.

```
begin
  integer p, q;
  y := 0;
  i := k := 1;
  for p:=1 step 1
  until n do
    for q:=1 step 1
    until m do
      if abs(a[p, q]) >
y then
      begin
        y := abs(a[p,
q]);
        i := p;
        k := q;
      end
    end
  end
end Absmax
```

La funzione Absmax(a) riceve in input una matrice nxm di numeri interi, cerca

il massimo assoluto e lo restituisce in output nel parametro Result, assieme alle coordinate (i,j) dove l'elemento risiede.

Vediamo come l'ALGOL assomigli molto al Pascal, che infatti da esso deriva.

Il PL/65 mixa l'uno e l'altro per ottenere da un lato la strutturazione dell'ALGOL e dall'altra il paradigma di basso livello per l'I/O.

Esempio bubblesort scritto in PL/65:

```
PAGE '**SORT**';
;
DECLARE F, I, TMP;
ENTRY $200;
;
N=N-2;
"SET TERMINAL VALUE FOR
LOOP";
F = 1; "SET FLAG";
WHILE F = 1 DO;
  F = 0;
  FOR I = 0 TO N BEGIN;
    IF B[I] > B[I+1] THEN
      BEGIN;
        F = 1;
        TMP = B[I];
        B[I]=B[I+1];
        B[I+1]=TMP;
      END;
    END;
  END;
N: DATA 10;
B: DATA 23,55,36,28,54,39,9
9,86,21,67;
;
EXIT;
```

Si vede come il linguaggio sia stato adattato per un sistema di sviluppo. Ad esempio lo statement:

```
ENTRY $200;
```

dichiara l'indirizzo iniziale in memoria del codice oggetto.

Interessante anche la dichiarazione e caricamento immediato delle variabili (N e B), in coda al codice. In questo caso N contiene il numero di elementi da ordinare e B è il vettore unidimensionale di interi da ordinare.

Il PL/65 è talmente orientato al processore 6502 che addirittura le variabili non possono chiamarsi A, X, Y, P e S, che sono i nomi dei registri interni al processore. Alcune istruzioni agiscono a livello di bit e richiamano direttamente le corrispondenti implementate nel micro codice del processore: SHIFT, ROTATE, CLEAR, SET, CODE.

Per assegnare un valore intero short (un byte) ad una variabile si possono usare le istruzioni di assegnazione diretta o indiretta ed inoltre si possono forzare le locazioni in pagina zero per la variabile stessa. La pagina zero è molto importante per il 6502, il quale può accedervi con istruzioni di due byte, più veloci rispetto all'accesso alle altre pagine di memoria.

La cosa funziona così: se si premette il simbolo "@" al nome di variabile, significa che vogliamo che sia immagazzinata in pagina zero. Se il carattere prefissato è invece "&", allora la variabile sarà immagazzinata nel classico stack

PL/65
USER'S
MANUAL

PL/65

predisposto dal compilatore in una pagina di memoria "alta".

Ad esempio:

@T = 5

&Z = 6.

Conclusioni

Il sistema di sviluppo R6500 della Rockwell è tutto ciò che l'appassionato hobbista e il progettista di circuiti di controllo e automazione, possano desiderare. Completo, discretamente espandibile e abbastanza economico per le caratteristiche che lo contraddistinguono, esso si pone al vertice dei sistemi di sviluppo per microprocessore che hanno avuto una impensabile diffusione prima del 1980.

Con la sua tastiera alfanumerica completa, il display e la stampante in grado di mostrare più dei classici e angusti display da 6/8 nexie dei sistemi concorrenti, detta decisamente un nuovo standard in laboratorio. Non è azzardato ipotizzare che la fortuna del microprocessore 6502 derivi per buona parte dalla diffusione dei sistemi di sviluppo e dell'AIM65 in particolare.

[Sm]

Bibliografia

Wikipedia:

<http://en.wikipedia.org/wiki/AIM-65>

Wapedia:

http://wapedia.mobi/en/AIM_65

<http://wapedia.mobi/en/PL/I>

<http://wapedia.mobi/en/ALGOL>

Recensioni

m&p computer n. 2

Bit n. 16, aprile 1981

http://www.atarimagazines.com/compute/issue1/3229_1_AIM_65_REVIEW.php

<http://www.dhub.org/object/164577>

<http://www.atarimagazines.com/computeii/issue1/page40.php>

<http://www.regnirps.com/Apple6502stuff/r65c02.htm>

Collezioni

Old Computer Museum: http://www.oldcomputermuseum.com/aim_65.html

Power House Museum:

<http://www.powerhousemuseum.com/collection/database/?irn=164576>

The Freeman PC Museum Collection:

[http://www.thepcmuseum.net/details.php?RECORD_KEY\(museum\)=id&id\(museum\)=784&PHPSESSID=2eec6b0ffaa7e0671f0ae923ceaa9b5c](http://www.thepcmuseum.net/details.php?RECORD_KEY(museum)=id&id(museum)=784&PHPSESSID=2eec6b0ffaa7e0671f0ae923ceaa9b5c)

Vintage computers:

<http://www.vintage-computer.com/aim65.shtml>

Old computer net:

<http://oldcomputers.net/AIM-65.html>

Old-Computers:

<http://www.old-computers.com/museum/computer.asp?st=1&c=58>

Manuali:

AIM65 programming chart

PL/65 user's manual

AIM 65 BASIC language reference Manual

AIM 65 Schematic poster

AIM 65 FORTH manual

AIM 65 User's Manual

AIM 65 Monitor Guide

MCS6500 Instruction set Summary

R6500 Hardware Manual

Riviste dedicate:

Interactive issue 01-08; Rockwell International