

COSTRUIAMOCI UN VERO MICROELABORATORE

HOME COMPUTER AMICO 2000

Nello scorso articolo abbiamo presentato alcuni elementi base di un microelaboratore, cominciando da una facile spiegazione di elementi come CPU, RAM, ROM ecc. Abbiamo anche parlato di codice binario e di codice esadecimale e introdotto il concetto di indirizzamento. Il linguaggio e il modo di affrontare i problemi sono stati, rivolti a tutti, non ai soliti supertecnici, e così intendiamo proseguire. Iniziamo questo articolo ricordando che il microelaboratore, oltre all'adeguata alimentazione elettrica, ha bisogno, per "funzionare", anche di un programma.

a cura della A.S.E.L. - parte seconda

COSA E' UN PROGRAMMA

Partendo da certi input e volendo ottenere certi output occorre far muovere i segnali così come noi vogliamo; questo risultato è ottenibile grazie ad un insieme di istruzioni (o comandi) che possiamo combinare logicamente secondo le nostre esigenze, insieme che chiameremo programma.

Proprio la possibilità di eseguire diversi programmi costruiti dall'utilizzatore differenzia radicalmente il microelaboratore dalle altre apparecchiature digitali.

Infatti quelli che probabilmente avete realizzato sinora sono stati progetti "specifici", che vi hanno permesso di costruire un antifurto, un frequenzimetro, un orologio digitale, sempre utilizzando per ogni specifica realizzazione delle diverse combinazioni di circuiti integrati. Ogni progetto espleta solamente una determinata funzione, era uno strumento "specifico", in grado di svolgere solo quella specifica attività per la quale era stato disegnato.

Con il microelaboratore invece viene effettuato un salto di qualità rispetto a questo tipo di impostazione perché, cambiando semplicemente programma, è possibile cambiare la funzione del microelaboratore, che infatti è uno strumento flessibile e di utilizzo universale, cioè non più specifico.

Sostituendo il programma sarà possibile cambiare in pochi istanti l'utilizzo del microelaboratore, passando per esempio da un gioco fatto sul televisore di casa alla gestione del conto in banca.

Il microcomputer è in grado di eseguire un insieme di operazioni elementari (ad esempio $A + B$, $A \text{ and } B$, $A = \bar{A}$; che, opportunamente disposte, ci permetteranno di creare insieme applicazioni nei più svariati campi ed anche giochi di vario genere. Nessun limite alle applicazioni secondo la fantasia e lo spirito di inventiva.

IL MICROELABORATORE E IL SUO LINGUAGGIO

Riprendendo il paragone iniziato nello scorso articolo tra microelaboratore e uomo, possiamo dire che volendo, ad esempio, far eseguire ad un nostro amico una addizione tra due numeri gli diremo: "Somma il numero A al numero B e scrivi il risultato su un foglio". Ci siamo avvalsi di un linguaggio che anche il nostro amico parla e

Fig. 1 - Questa è la piastra base del sistema microelaboratore AMICO 2000. Si tratta di un vero e proprio microcomputer in grado di colloquiare con l'ambiente esterno e pur essendo autosufficiente può essere considerato come il pezzo fondamentale di un sistema molto più potente che verrà via via presentato nel corso di questi articoli. L'AMICO 2000/A, questa la sigla della scheda (misura 30 x 10 cm) viene fornito in scatola di montaggio o montato e collaudato dalla società costruttrice ASEI, s.r.l.

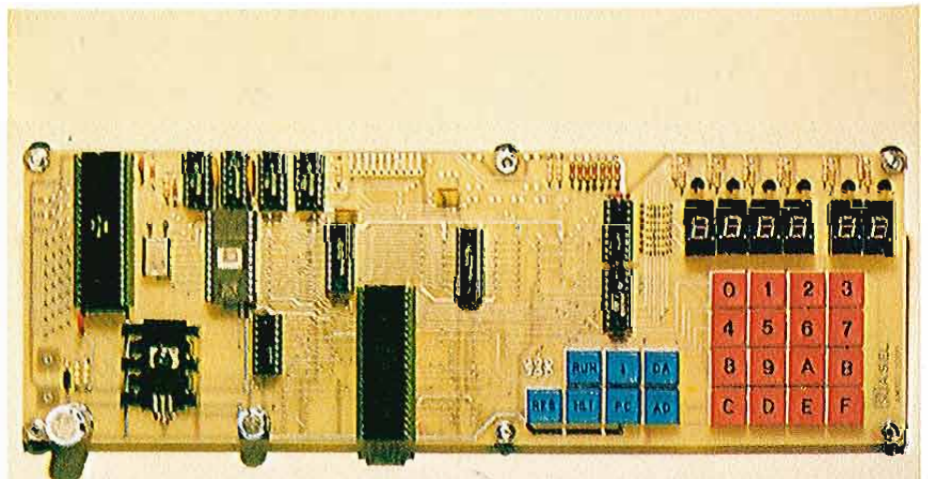


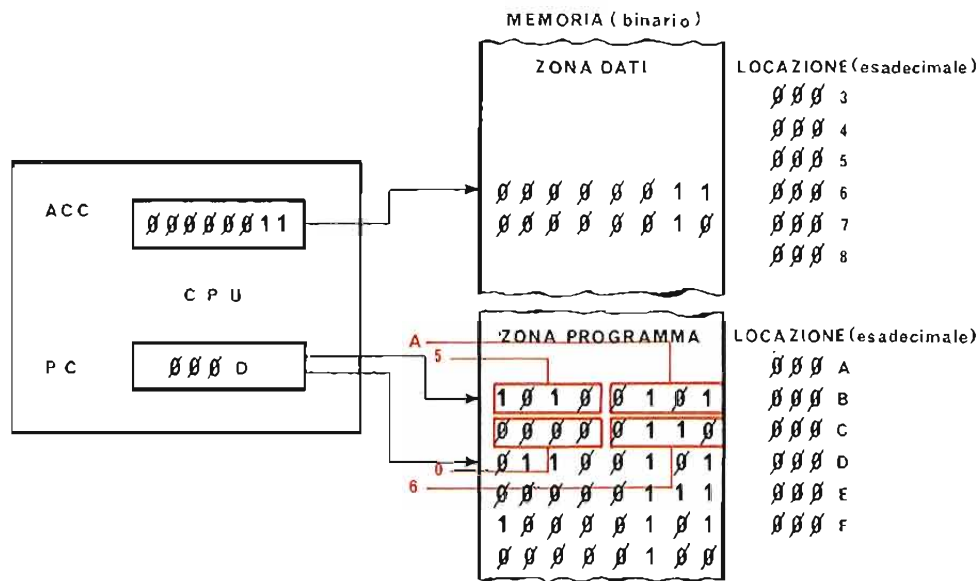
Tavola A - Rappresentazione dei "linguaggi" e del movimento dei dati all'interno del microelaboratore in base alle istruzioni contenute in un semplice programma che consente di sommare due numeri.

ISTRUZIONI	LINGUAGGIO DISCORSIVO (il nostro, non comprensibile dal microelaboratore)	LINGUAGGIO SIMBOLICO (intermedio tra il nostro e quello del micro)	LINGUAGGIO MACCHINA (è il linguaggio del microelaboratore)
1 ^a istruzione	Preleva dalla sesta posizione di memoria RAM il dato e trasporta nel registro interno (ACCUMULATORE) dell'unità di elaborazione (CPU).	LDA 06 (LDA sta per Load Accumulator e 06 indica in esadecimale la posizione di memoria nella quale si trova il dato da trasferire).	A5 06 (A5 è la traduzione di LDA in esadecimale).
2 ^a istruzione	Preleva il dato contenuto nella settima posizione di memoria RAM e sommalo al dato contenuto nel registro interno (ACCUMULATORE) della CPU.	ADC 07 (ADC sta per Add Memory to Accumulator con Carry; la funzione del Carry sarà spiegata in seguito. 07 indica, in esadecimale, la posizione di memoria che contiene il dato da sommare all'ACCUMULATORE.	65 07 (65 è la traduzione di ADC in esadecimale).
3 ^a istruzione	Trasferisci il risultato dall'ACCUMULATORE alla quarta locazione di memoria RAM.	STA 04 (STA significa Store Accumulator, istruzione di scrittura nella RAM). 04 indica in esadecimale la posizione di memoria nella quale deve essere ricopiato il contenuto dello ACCUMULATORE.	85 04 (85 è la traduzione di STA in esadecimale).

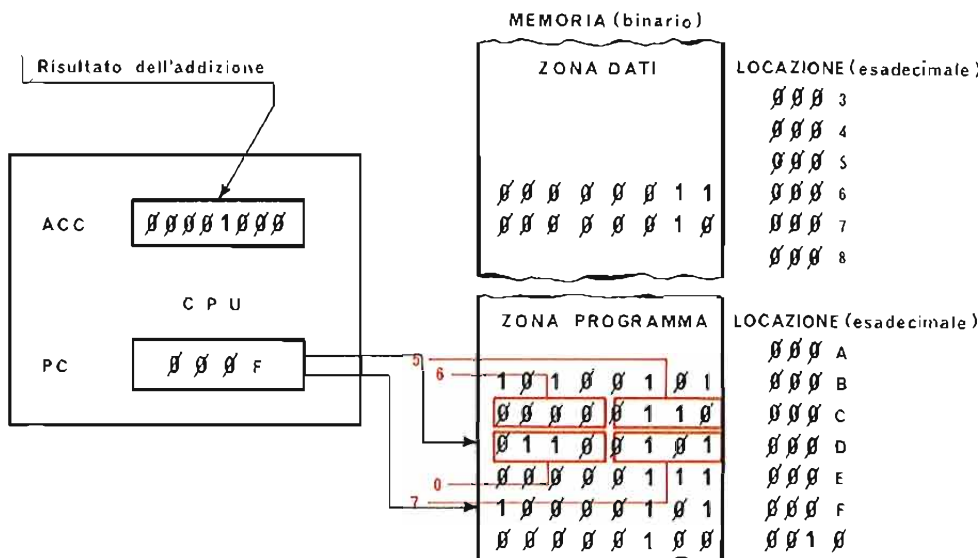
* Il significato del termine viene spiegato nel corso dell'articolo.

RAPPRESENTAZIONE DELLO STATO DEL SISTEMA

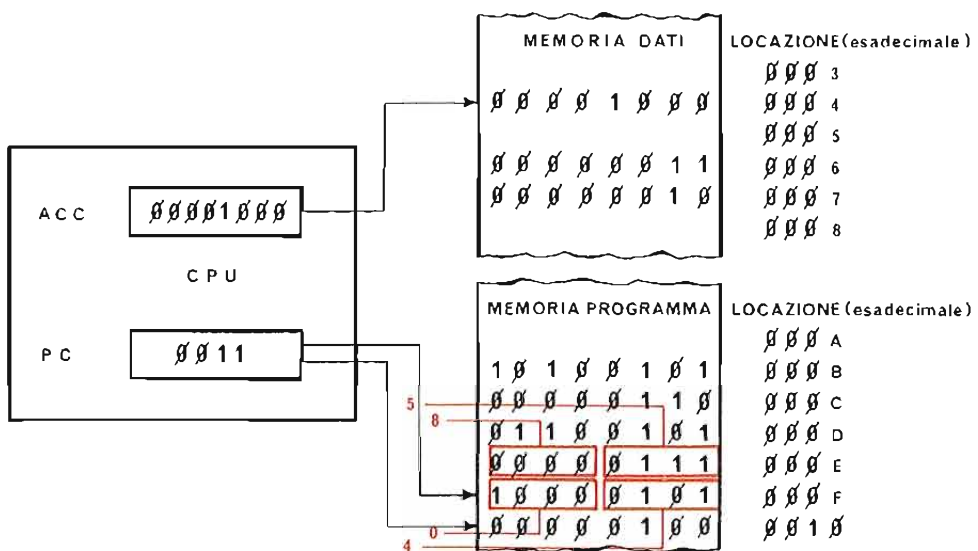
NOTE



Si tratta di una istruzione da due bytes, contenuta nella posizione di memoria 000B e 000C, posizioni indicate dal Program Counter (PC) prima della esecuzione. Eseguita l'istruzione il contenuto del PC è 000D che rappresenta l'indirizzo da cui verrà prelevata la prossima istruzione.

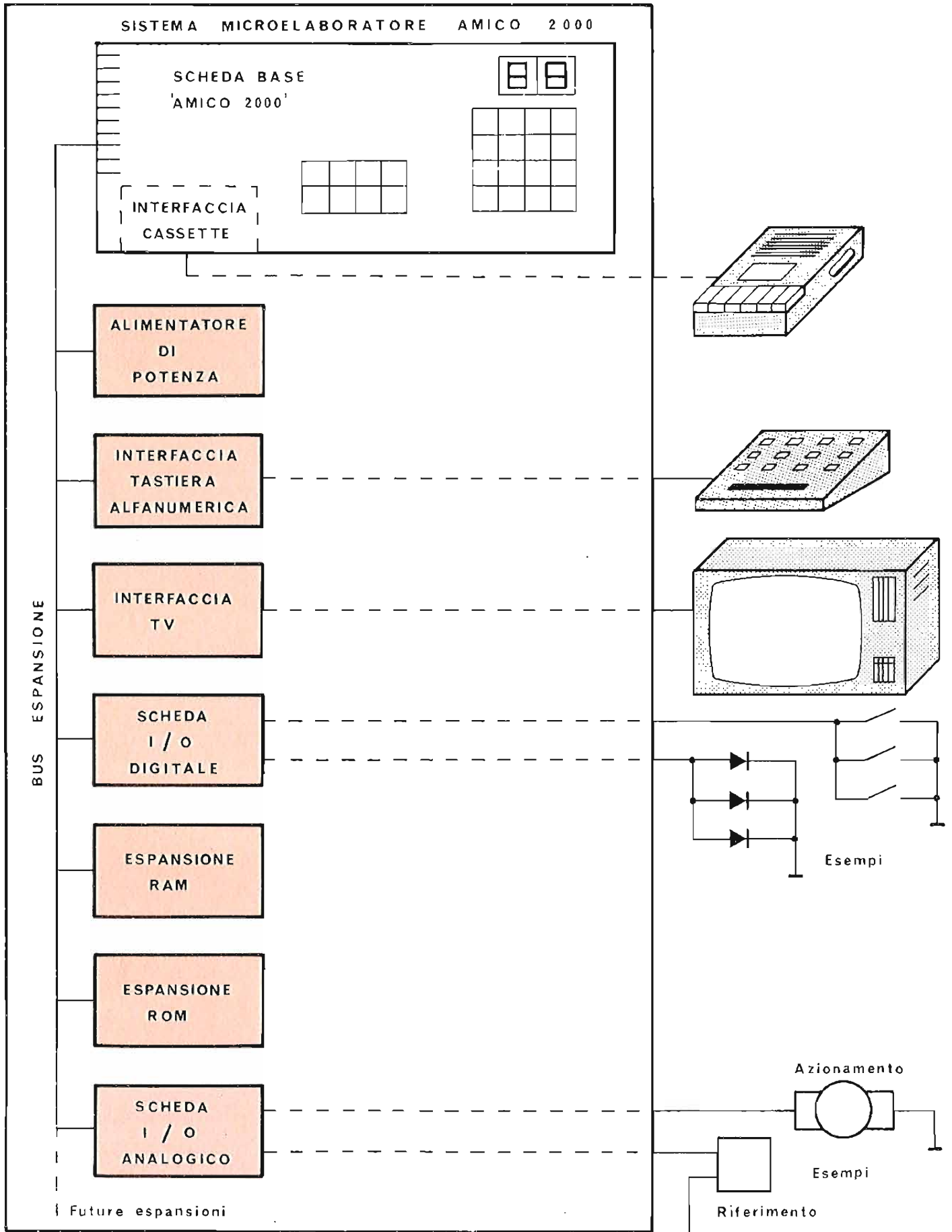


Anche questa è una istruzione da due bytes. Il dato contenuto nella settima posizione di memoria viene sommato al contenuto dell'ACCUMULATORE che, a istruzione eseguita, contiene il risultato dell'addizione.



Come si vede dalla figura a lato il contenuto dell'ACCUMULATORE è stato ricopiato nella locazione di memoria 0004. Il contenuto dell'ACCUMULATORE rimane invariato.

Le frecce grigie indicano la situazione ad istruzione eseguita



Schema a blocchi del sistema di microelaboratore AMICO 2000. La 1ª scheda base AMICO 2000/A verrà fornita montata e in scatola di montaggio con il prossimo articolo di questa serie. Il sistema verrà arricchito in seguito con tutte le schede qui rappresentate in blocchi

capisce, e che associa ad un vocabolo come "somma" una operazione univocamente identificabile ed eseguibile su dei dati che pure vengono forniti in modo definito.

Anche il microelaboratore parla il suo linguaggio (linguaggio macchina) ed è in grado di fare esattamente ciò che vogliamo, se sappiamo parlargli nel suo linguaggio. Come avevamo accennato la volta scorsa il microelaboratore lavora in termini binari e pertanto il suo linguaggio originale è binario. D'altro canto abbiamo visto che sarebbe troppo lungo e fonte di molti errori utilizzare insiemi di 0 e di 1 per parlare con il microelaboratore, essendo il "linguaggio" basato sul codice binario troppo diverso da quello che comunemente utilizziamo per risolvere i nostri problemi. Come fare allora per comunicare in modo più semplice con il nostro microelaboratore indicandogli le operazioni che deve svolgere?

In un primo momento viene analizzato il problema e sviluppato un programma nel nostro linguaggio, cioè in termini colloquiali, come ad esempio "Somma il numero A al numero B e scrivi il risultato". Successivamente questo viene tradotto in un linguaggio intermedio fra noi e la macchina, cioè un linguaggio che noi possiamo capire essendo esso "simbolico" o "mnemonico" in quanto ogni istruzione è formata da abbreviazioni di parole che associamo immediatamente ad una operazione che ci è ben chiara.

A questo punto è facile scrivere a fianco la traduzione esadecimale di questo linguaggio intermedio, cioè quella comprensibile direttamente dalla macchina. Noi forniremo pertanto i programmi in esadecimale, e la macchina provvederà alla ulteriore traduzione diretta in binario.

Il passaggio più complesso è pertanto quello da linguaggio intermedio (mnemonico) a linguaggio macchina in esadecimale, passaggio che viene effettuato o manualmente tramite una tabella, (vedi tabella 1) oppure, in maniera più evoluta tramite un apposito programma chiamato Assembler o traduttore, che sarà illustrato in seguito.

UN ESEMPIO PRATICO

Per meglio comprendere tutto ciò e per cominciare ad avvicinarci al nostro AMICO 2000 facciamo un esempio applicativo piuttosto elementare, ma molto efficace ai nostri fini.

Supponiamo di voler eseguire la seguente operazione: $3 + 5 = 8$ e supponiamo anche di aver già introdotto il numero "3" 00000011 in binario e 03 in esadecimale) nella sesta locazio-

occhio alle sorprese!!!

L'UNICO AL MONDO!!

ZG/6999-00

2000 RAWA x-Band

L. 225.000

rivelatore Multanova che "vede" anche dietro le curve.

RAWA 2000 X-Band è un rivelatore radar ad elevata sensibilità, in grado di rivelare tempestivamente la presenza degli apparecchi radar per il controllo della velocità attualmente conosciuti in Europa.

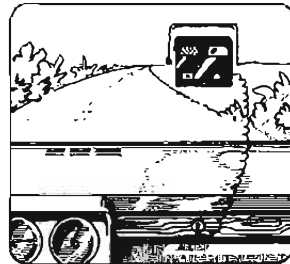
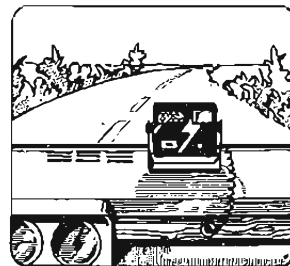
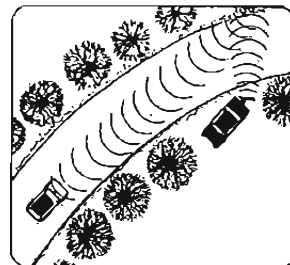
Queste apparecchiature di misurazione, grazie agli ultimi progressi tecnici, hanno piccole dimensioni e possono essere sistemate sia su veicoli che in installazioni fisse.

Per il conducente di autoveicoli non è più quindi possibile "vederle" in tempo utile.

Il rivelatore radar RAWA 2000 X-Band capta anche le più deboli onde elettromagnetiche emesse dai misuratori di velocità e riflesse dagli ostacoli che incontrano (altri veicoli, edifici, alberi, cespugli, ondulazioni del terreno), le amplifica e le invia immediatamente al segnalatore ottico od acustico di cui è provvisto, non è disturbato da altre fonti di segnali.

CARATTERISTICHE TECNICHE

- Diode luminoso che segnala il funzionamento
- Avvisatore ottico tramite lampeggiatore rosso
- Avvisatore acustico con cicalino intermittente
- Temperatura di funzionamento: $0 \div +60$ °C
- Facile montaggio sul cruscotto o sul parabrezza senza disturbare la visuale
- Contenitore nero opaco antiriflesso e supporto antivibrazione con base autoadesiva
- Cavo di alimentazione (+12 Vc.c./150 mA) con spinotto da inserire nell'accendisigari elettrico del cruscotto.



ne di memoria e il numero "5" (00000101 in binario e 05 in esadecimale) nella settima locazione di memoria.

Vogliamo ora che il microelaboratore esegua la somma e che depositi il risultato nella quarta posizione (o locazione) di memoria. Questo insieme di operazioni può essere rappresentato come nella tavola A.

La tavola A è da studiare con attenzione: in essa vengono spiegati alcuni concetti fondamentali che, se ben compresi, permetteranno di seguire con molta facilità il resto dell'articolo. In sostanza vengono esaminate tre delle più importanti istruzioni del microprocessore 6500 che è il "cervello" dell'AMICO 2000. E così in seguito saranno analizzate via via tutte le istruzioni in modo da permettervi alla fine del corso di programmare autonomamente il vostro microcomputer.

Il semplice insieme di istruzioni considerato nella tavola A permette quindi di sommare due numeri e di porre il risultato in memoria; già queste poche istruzioni possono costituire un "programma".

Un programma può risiedere sia in memoria ROM che in memoria RAM: il primo caso è classico per il programma chiamato MONITOR o SISTEMA OPERATIVO, che permette di eseguire alcune attività basilari come, ad esempio, quelle di input ed output dei dati. I programmi che effettueremo noi a scopo didattico o per diletto, verranno invece solitamente immessi in RAM, memoria che può essere scritta e letta dall'utilizzatore e pertanto modificata a piacere.

Per inciso notiamo che i programmi contenuti nella memoria RAM vengono mantenuti finché permane l'alimentazione.

Togliendo l'alimentazione i programmi contenuti in RAM vanno a carte quarantotto!! Ciò non avviene per quelli contenuti in ROM.

Le operazioni di esecuzione del programma si svolgono in pratica nel modo seguente (Vedere la figura nella Tavola A).

La CPU legge la prima istruzione del programma, la interpreta e la esegue; a questo punto la CPU legge l'istruzione successiva, la interpreta ed esegue, e procede nello stesso modo in maniera sequenziale fino alla completa esecuzione del programma, cioè fino a che non troverà una istruzione di fine programma.

La CPU è in grado di capire ed eseguire un certo numero di ordini diversi, che costituiscono l'insieme delle possibili istruzioni, o set di istruzioni.

In particolare, per il microprocessore che è utilizzato dal nostro microelaboratore AMICO 2000, il set è composto da 56 istruzioni, ma esistono anche dei microprocessori con oltre 150 istruzioni.

II. PROGRAM COUNTER

Evidentemente la CPU non può cominciare ad eseguire istruzioni partendo da un punto qualsiasi della memoria e seguendo un ordine casuale, ma deve iniziare l'esecuzione partendo da un punto ben preciso, che rappresenta l'inizio del programma (nel caso particolare del nostro esempio la locazione di memoria 000B in esadecimale).

In concreto nella CPU si trova un importantissimo registro, chiamato Program Counter (contatore di programma) che contiene l'indirizzo della locazione di memoria dalla quale verrà prelevata la prossima istruzione del programma che la CPU deve eseguire. Il linguaggio più tecnico il Program Counter (abbreviato PC) "punta" (ovvero indica alla CPU) la prossima istruzione da eseguire. Per questo motivo si parla anche di Puntatore (in inglese "Pointer").

Dopo aver eseguito una istruzione del programma il Program Counter (PC) si posiziona automaticamente sull'indirizzo della istruzione da eseguire subito dopo e non richiede pertanto alcun intervento da parte nostra.

In particolare riferendoci all'esempio riportato, il Program Counter (posto in partenza a 000B dall'utente) sarà 00B all'inizio del programma, 00D dopo aver eseguito la prima istruzione, 00F dopo aver eseguito la seconda e 0011 dopo aver effettuato la terza. Il fatto che il Program Counter incrementi di 2 è dovuto al tipo di istruzione che la macchina sta eseguendo: in pratica la maggior parte di istruzioni comporta un incremento di 2, ovvero esse occupano due locazioni di memoria consecutive (due byte). Esistono però anche istruzioni che occupano uno o tre byte; in questo caso il PC si incrementerà automaticamente di 1 e 3 rispettivamente, puntando sulla istruzione successiva.

Il contenuto del Program Counter può infine essere anche modificato dall'utilizzatore grazie ad alcune istruzioni che permettono di saltare da un punto all'altro del programma senza dover seguire una rigida sequenza di comandi.

LE ISTRUZIONI ESAMINATE

Le tre istruzioni si qui incontrate possono essere così esemplificate:

LDA M = M - A

ADC M = A + M - A

M = locazione di memoria qualsiasi

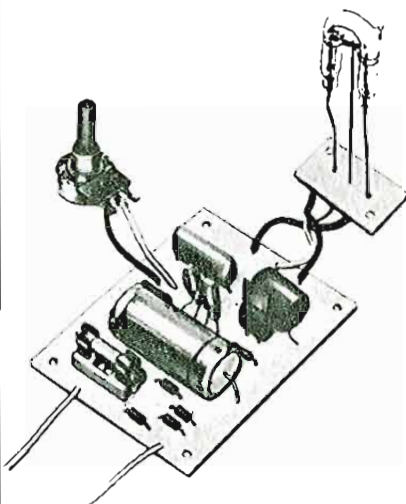
A = Accumulatore

+ = simbolo di somma

- = Simbolo di trasferimento che indica il movimento del dato da a ...

Kutiuskit

STROBOFLASH KS-270



Efficiente lampeggiatore stroboscopico a scarica nel gas Xenon, con possibilità di regolazione della frequenza. Utile per il controllo di organi rotanti e vibranti oppure per scopi di intrattenimento anche in combinazioni con luci psichedeliche.

Frequenza di lampeggiamento:

2÷25 Hz

Alimentazione: 220 Vc.a.

SM/8270-07

in vendita presso le sedi GBC

Le due istruzioni LDA e STA consentono di muovere un dato (byte) tra memoria e Accumulatore. Queste operazioni sono fondamentali e ciò le rende tra le più utilizzate nei programmi.

Sul prossimo numero: L'Amico 2000 in scatola di montaggio.

Forse per alcuni questa puntata del nostro corso pratico sul microelaboratore può essere sembrata un tantino ostica e invero non esattamente molto "pratica". Calma e sangue freddo. L'importante è tener duro fino al prossimo numero dove ci tufferemo letteralmente dal vivo del montaggio con la costruzione della piastra base di tutto il nostro sistema AMICO 2000 con la quale potremo subito cominciare a programmare e prendere confidenza con quello strano individuo chiamato microprocessore. Per il vostro bene (no, non vogliamo essere paternalisti, lo diciamo sul serio) vi consigliamo di leggere più volte questo articolo per cercare di comprenderlo fino in fondo. Le poche parti teoriche che saranno trattate durante questa serie di articoli sono assolutamente indispensabili per poter godere in futuro a pieno del vostro meraviglioso "giocattolo". In ogni caso, come sempre, vi preghiamo di scriverci se vi fosse venuto qualche dubbio o se tutto non vi è stato particolarmente chiaro: faremo del nostro meglio per mettervi nelle condizioni di comprendere tutto pubblicando le risposte in una apposita rubrica a fine articolo. Anche qui, come nello scorso numero, pubblichiamo la scheda di prenotazione dell'AMICO 2000 al fine di permettere alla ASEI. S.r.l. di evadere gli ordini con la massima tempestività. Una nota infine per i "secchioni" (quelli, per intenderci, che vogliono sapere tutto prima di tutti e più a fondo): degli ottimi testi che introducono al mondo del microprocessore vengono pubblicati dalla Jackson Italiana Editrice di Milano, P.le Massari 22. Quello che vi consigliamo è "Introduzione ai micro-computer", di prossima pubblicazione o "I microprocessori e le loro applicazioni" già disponibile.

ALCUNI DETTAGLI SULLA CPU

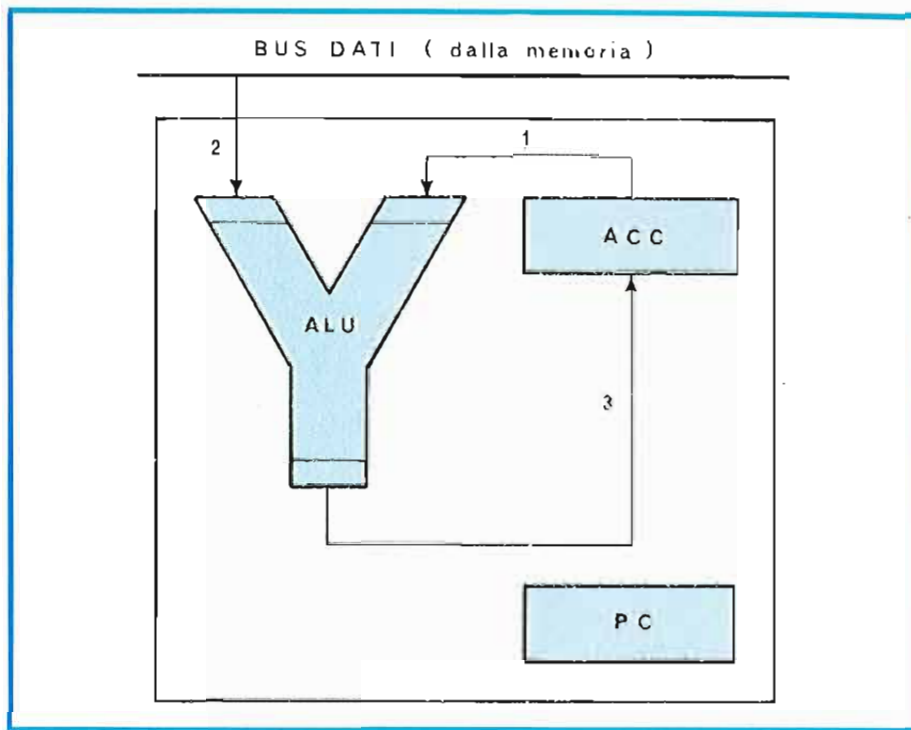
Sino ad ora abbiamo accennato alla presenza, all'interno della CPU, dei seguenti elementi:

ACCUMULATORE (ACC)
PROGRAM COUNTER (PC)
ALU (Arithmetic-logic unit, unità aritmetico-logica).

L'accumulatore ed il Program Counter sono due registri assai importanti, e l'ALU è il vero e proprio "cuore" della CPU.

Si può quindi per ora schematizzare la CPU come figura. Facendo riferimento al nostro esempio analizziamo il flusso dei dati all'interno della CPU durante l'esecuzione dell'istruzione ADC 07 (cioè la somma).

Il primo dato, contenuto nell'accumu-



Alcuni dettagli della CPU

latore, viene trasferito ad uno dei due ingressi dell'ALU (freccia 1), immediatamente dopo, attraverso il Bus Dati viene presentato (freccia 2) al secondo ingresso dell'ALU il dato contenuto nella settima locazione di memoria (infatti l'istruzione era ADC 07, che vuol dire somma il contenuto della settima locazione all'accumulatore). Avendo a

disposizione entrambi i dati necessari all'esecuzione dell'istruzione l'ALU esegue la somma al suo interno e trasferisce (freccia 3) il risultato dell'operazione nell'accumulatore, sostituendo il contenuto precedente. Tutte queste operazioni vengono eseguite in maniera automatica dalla CPU, e l'operatore non deve intervenire in alcun modo.

Tabella 1 - Traduzione delle istruzioni del microprocessore 6500 (che è la CPU utilizzata nel microcomputer AMICO 2000) da rappresentazione in CODICE ESADECIMALE a LINGUAGGIO SIMBOLICO. Sono riquadrate le prime istruzioni trattate in questo articolo. Questa tabella è da conservare e vi permetterà di verificare i vostri progressi nell'apprendimento delle istruzioni del microelaboratore.

00 - BRK	24 - BIT - Zero Page
01 - ORA - (Indirect, X)	25 - AND - Zero Page
05 - ORA - Zero Page	26 - ROL - Zero page
06 - ASL - Zero Page	28 - PLP
08 - PHP	29 - AND - Immediate
09 - ORA - Immediate	2A - ROL - Accumulator
0A - ASL - Accumulator	2C - BIT - Absolute
0D - ORA - Absolute	2D - AND - Absolute
0E - ASL - Absolute	2E - ROL - Absolute
10 - BPL	30 - BMI
11 - ORA - (Indirect), Y	31 - AND (Indirect), Y
15 - ORA - Zero Page, X	35 - AND - Zero Page, X
16 - ASL - Zero Page, X	36 - ROL - Zero Page, X
18 - CLC	38 - SEC
19 - ORA - Absolute, Y	39 - AND - Absolute, Y
1D - ORA - Absolute, X	3D - AND - Absolute, X
1E - ASL - Absolute, X	3E - ROL - Absolute, X
20 - JSR	
21 - AND - (Indirect, X)	

Segue Tabella 1

4Ø - RTI
 41 - EOR - (Indirect, X)
 45 - EOR - Zero page
 46 - LSR - Zero Page
 48 - PHA
 49 - EOR - Immediate
 4A - LSR - Accumulator
 4C - JMP - Absolute
 4D - EOR - Absolute
 4E - LSR - Absolute

5Ø - BVC
 51 - EOR - (Indirect), Y
 55 - EOR - Zero Page, X
 56 - LSR - Zero Page, X
 58 - CLI
 59 - EOR - Absolute, Y
 5D - EOR - Absolute, X
 5E - LSR - Absolute, X

6Ø - RTS
 61 - ADC - (Indirect, X)
 65 - ADC - Zero Page
 66 - ROR - Zero Page
 68 - PLA
 69 - ADC - Immediate
 6A - ROR - Accumulator
 6C - JMP - Indirect
 6D - ADC - Absolute
 6E - ROR - Absolute

7Ø - BVS
 71 - ADC - (Indirect), Y
 75 - ADC - Zero Page, X
 76 - ROR - Zero Page, X
 78 - SEI
 79 - ADC - Absolute, Y
 7D - ADC - Absolute, X
 7E - ROR - Absolute, X

81 - STA - (Indirect, X)
 84 - STY - Zero Page
 85 - STA - Zero Page
 86 - STX - Zero Page
 88 - DEY
 8A - TXA
 8C - STY - Absolute
 8D - STA - Absolute
 8E - STX - Absolute

9Ø - BCC
 91 - STA (Indirect), Y
 94 - STY - Zero Page, X
 95 - STA - Zero page, X
 96 - STX - Zero page, Y
 98 - TYA
 99 - STA - Absolute, Y
 9A - TXS
 9D - STA - Absolute, X

AØ - LDY - Immediate
 A1 - LDA - (Indirect, X)

A2 - LDX - Immediate
 A4 - LDY - Zero Page
 A5 - LDA - Zero Page
 A6 - LDX - Zero Page
 A8 - TAY
 A9 - LDA - Immediate
 AA - TAX
 AC - LDY - Absolute
 AD - LDA - Absolute
 AE - LDX - Absolute

BØ - BCS
 B1 - LDA - (Indirect), Y
 B4 - LDY - Zero Page, X
 B5 - LDA - Zero Page, X
 B6 - LDX - Zero Page, Y
 B8 - CLV
 B9 - LDA - Absolute, Y
 BA - TSX
 BC - LDY - Absolute, X
 BD - LDA - Absolute, X
 BE - LDX - Absolute, Y

CØ - CPY - Immediate
 C1 - CMP - (Indirect, X)
 C4 - CPY - Zero Page
 C5 - CMP - Zero Page
 C6 - DEC - Zero Page
 C8 - INY
 C9 - CMP - Immediate
 CA - DEX
 CC - CPY - Absolute
 CD - CMP - Absolute
 CE - DEC - Absolute

DØ - BNE
 D1 - CMP - (Indirect), Y
 D5 - CMP - Zero page, X
 D6 - DEC - Zero Page, X
 D8 - CLD
 D9 - CMP - Absolute, Y
 DD - CMP - Absolute, X
 DE - DEC - Absolute, X

EØ - CPX - Immediate
 E1 - SBC - (Indirect, X)
 E4 - CPX - Zero Page
 E5 - SBC - Zero Page
 E6 - INC - Zero Page
 E8 - INX
 E9 - SBC - Immediate
 EA - NOP
 EC - CPX - Absolute
 ED - SBC - Absolute
 EE - INC - Absolute

FØ - BEQ
 F1 - SBC (Indirect), y
 F5 - SBC - Zero Page X
 F6 - INC - Zero Page, X
 F8 - SED
 F9 - SBC - Absolute, Y
 FD - SBC - Absolute, X
 FE - INC - Absolute, X

UK770



UNITÀ DI COMMUTAZIONE PER GIRADISCHI UK 770

Un accessorio semplicissimo ma indispensabile che consente di commutare a volontà due piatti giradischi stereo oppure qualsiasi altra coppia di trasduttori sonori su un'unica uscita, prelevando il segnale sulla prima o sulla seconda entrata a seconda delle necessità; indispensabile per qualsiasi tipo di regia.



L'applicazione di questo dispositivo si rivela utile in discoteche, trattenimenti musicali, alberghi, sale d'aspetta, stazioni radio, in casa ecc. Le prese di entrata e di uscita schermate ed il contenitore metallico, assicurano l'accurata schermatura da tutti i campi elettromagnetici di disturbo esterni.



CARATTERISTICHE TECNICHE

CUP: microprocessore 6502
Memoria RAM: 1 K byte
Memoria ROM contenente il monitor
Tastiera esadecimale
Visualizzatore LED a 6 cifre
Interfaccia parallelo
Prediposto per interfaccia per telescrivente e per registratore a cassette.
Regolatore di tensione incorporato
Alimentazione 5 V, 800 mA max
Circuito stampato professionale doppia faccia in vetronite

(*) Per alimentarlo basta una tensione raddrizzata e filtrata compresa fra 7 e 12 V in grado di fornire 1000 mA.

Prezzo in kit Lit. 195.000 (**) IVA compresa.
Montato e collaudato (garanzia 3 mesi) Lit. 235.000 IVA compresa.

(**) **IMPORTANTE:** Il kit è comprensivo di una speciale garanzia per cui in caso di mal funzionamento o insuccesso nella realizzazione è possibile inviare la piastra, con tutti i componenti, al costruttore, che la sostituirà con una montata e collaudata dietro il pagamento di una quota fissa di Lit. 50.000.

MODULO DI PRENOTAZIONE PER IL MICROELABORATORE "AMICO 2000"

È possibile prenotare sin d'ora l'acquisto del microelaboratore AMICO 2000. Chi invierà il presente modulo di prenotazione entro 30 giorni dalla data di copertina avrà diritto ad uno sconto di Lit. 15.000 al momento dell'acquisto sui prezzi indicati fra parentesi.

Vorrei prenotare un microelaboratore AMICO 2000 (barrare il quadratino corrispondente).

- in scatola di montaggio (Lit. 195.000)
 montato e collaudato (Lit. 235.000)

(scrivere in stampatello)

Nome _____
Cognome _____
Via _____ n. Tel. _____
Cap. _____ Città _____

NON INVIARE DENARO

Ritagliare e spedire in busta chiusa a:

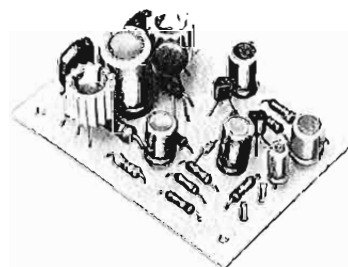
A.S.E.L. s.r.l. - Via Roma, 43 - 20041 Cavenago Brianza

UK145A



AMPLIFICATORE DI BASSA FREQUENZA 1,5 W UK 145-A

Circuito miniaturizzato di uso universale per applicazioni su apparecchi radio portatili, fonovaligie, microregistratori ecc.
Ottimo per funzionare in combinazione con il sintonizzatore UK 521.
Ottima fedeltà e sensibilità.



CARATTERISTICHE TECNICHE

Alimentazione: 9 Vc.c.
Assorbimento (pot. Uscita = 0) 14 mA
Assorbimento (Pot. Uscita = 0,5 W) 130 mA
Sensibilità Ingresso (Pu = 0,5 W) 45 mV
Impedenza d'ingresso: 100 kΩ
Impedenza d'uscita: 8Ω
Risposta di frequenza (-3 dB) 50-25.000 Hz
Distorsione armonica (Pu = 0,5 W) 1%