

La maggioranza di coloro che hanno acquistato e montato le prime schede del nostro microcomputer, cioè scheda CPU, tastiera esadecimale, interfaccia tastiera, interfaccia cassette ed espansione della memoria, è convinta che avendo a disposizione solo questo materiale, il microcomputer stesso possa servire solo per provare alcuni semplici programmi didattici.

Noi invece cercheremo di dimostrarvi il contrario e quando avrete terminato di leggere questo articolo vi lasciamo liberi di trarre le vostre conclusioni. Quella che stiamo per presentarvi è infatti una vera e propria «gestione di magazzino» da effettuarsi su tale microcomputer, cioè tutta una serie

di programmi e subroutine mediante i quali è possibile tenere un archivio di un numero praticamente illimitato di articoli, purché suddivisi per categorie o per gruppi, effettuare i carichi e gli scarichi, conoscere la giacenza, la quantità entrata durante l'anno, la quantità uscita, il prezzo di acquisto e il prezzo vendita.

La gestione da noi realizzata è idonea per qualsiasi tipo di merce o componenti: per esempio in una farmacia potremmo utilizzarla per catalogare pillole, sciroppi, fiale ecc.; in un laboratorio di elettronica per tenere sotto controllo la giacenza e la vendita di integrati, transistor, condensatori; in una cartoleria per seguire molto speditamente gli acquisti e le vendite di quaderni, matite, compassi; in una ferramenta per tenere il conto di cacciaviti, martelli, bulloni ecc. In altre parole si tratta di un esempio di gestione «universale» che con un po' di fantasia può essere utilizzata anche per altre funzioni che non siano semplicemente un magazzino, ad esempio per fini amministrativi onde controllare il **dare** e l'**avere** di fornitori clienti.

Precisiamo che tale «gestione» non ha la pretesa di essere la migliore in assoluto che si possa

GESTIONE



realizzare con questo nostro microcomputer, anzi i più esperti «programmatore» non mancheranno di criticare talune scelte adottate, tuttavia occorre tener presente il fine che ci siamo proposti nel prepararla, cioè dimostrare al lettore che se non si hanno grosse esigenze, le schede del nostro microcomputer attualmente a disposizione consentono già di ottenere dei risultati pratici non indifferenti.

Non solo ma anche ammesso che a qualcuno non interessi la gestione in se stessa in quanto già pensa di effettuarla in linguaggio BASIC, quando avrà a disposizione la tastiera alfanumerica, il video ed il floppy disk, potrà sempre servirsi di

dati normalmente registrati su nastro (laddove cioè vi sono meno probabilità che possano andare perduti qualora per esempio venga a mancare la tensione di rete) e di richiamare ogni volta in memoria solo ed esclusivamente quel gruppo o quella categoria che ci interessa modificare.

Precisiamo che nelle due RAM disponibili sulla scheda CPU noi terremo memorizzati i programmi mentre nelle due RAM presenti sulla scheda di espansione terremo memorizzati i dati relativi ai vari componenti letti di volta in volta dal nastro.

In altre parole i programmi e le subroutine che troverete più avanti dovranno essere scritti nel primo «kappa» della memoria RAM, mentre tutte le

In questo articolo vi presentiamo un qualcosa di più di un semplice programma da sperimentare sul nostro microcomputer infatti si tratta di una vera e propria gestione di «magazzino» che vi consentirà, utilizzando le sole schede finora a disposizione, di tenere un archivio di oltre 1.000 articoli diversi, effettuare i carichi e gli scarichi, sapere quanti pezzi sono stati acquistati e quanti venduti durante l'anno, conoscere la giacenza attuale nonché il costo e il prezzo vendita.

di magazzino con il MICRO Z80

queste pagine per imparare ad usare determinate istruzioni che ancora non vi abbiamo descritto oppure per imparare alcuni trucchetti impiegati in tali programmi che facilitano notevolmente l'utilizzazione del microcomputer allo stato attuale.

QUALI SCHEDE SONO NECESSARIE

Vi abbiamo già detto che per poter effettuare questa «gestione di magazzino» occorrono tutte le schede del nostro microcomputer finora presentate, vale a dire:

- la scheda CPU siglata LX382
- la tastiera esadecimale LX384
- l'interfaccia per registratori LX385
- la scheda di espansione LX384 con almeno 1 «kappa» di RAM montato sopra
- il BUS e l'alimentatore
- un registratore e diversi nastri

Come avrete notato, nella versione «minima» del nostro programma non occorre neppure completamente la scheda di espansione della memoria in quanto due sole RAM (1 kappa) montate su questa scheda, sono sufficienti insieme alle due già presenti sulla scheda della CPU, per raggiungere lo scopo.

La nostra «gestione» infatti prevede di tenere i

informazioni relative ai vari componenti (cioè codice, giacenza, entrate, uscite, costo, prezzo vendita) verranno automaticamente memorizzate nel secondo «kappa».

INFORMAZIONI RELATIVE a ciascun COMPONENTE

Una volta appreso che cosa occorre per effettuare questa «gestione» automatizzata, il lettore si chiederà subito quante informazioni vengono memorizzate per ciascun componente e quanti componenti possono trovare spazio in 1 «kappa» di RAM.

A tale proposito vi risponderemo che per ciascun componente del nostro magazzino vengono memorizzati i seguenti dati:

Codice: composto da 4 numeri o lettere qualsiasi purché presenti sulla tastiera esadecimale, per un totale di **2 righe** di memoria (una riga ogni due cifre o caratteri).

Giacenza: da 0 a 999.999 unità (cioè 6 cifre) per un totale di **3 righe** di memoria.

Quantità entrata: da 0 a 999.999 unità (cioè 6 cifre) per un totale di **3 righe** di memoria.

Quantità uscita: da 0 a 999.999 unità (cioè 6 cifre max) per un totale di **3 righe** di memoria.

Prezzo vendita: da 0 a 999.999 lire (cioè 6 cifre max) per un totale di **3 righe** di memoria.

Costo: da 0 a 999.999 lire (cioè 6 cifre max) per un totale di **3 righe** di memoria.

In altre parole ciascun componente che noi inseriamo in memoria occupa **2 righe** per il codice e $3+3+3+3+3=15$ **righe** per le altre informazioni più una riga aggiuntiva che gli è stata assegnata per semplicità di programmazione, quindi **18 righe** in totale. Poiché in 1 «kappa» di memoria si hanno a disposizione complessivamente 1.024 righe di cui occorre tener presente che le ultime 10-20 servono al programma MONITOR per svolgere le sue funzioni, il numero massimo di componenti che potremo inserire su ciascun «kappa» sarà **55** (infatti $55 \times 18 = 990$).

A questo punto certamente vi chiederete: «se io ho un magazzino di 500-550 componenti, come faccio a gestirlo con 1 kappa di RAM quando avete appena detto che su 1 kappa si possono memorizzare solo 55 componenti?».

La soluzione da noi consigliata è la seguente: frazionare il magazzino in tante categorie (resistenze, condensatori, diodi, transistor ecc.) composte da un massimo di 55 componenti ciascuna, registrare queste categorie su tanti nastri diversi e richiamare in memoria di volta in volta solo la categoria che interessa.

PROGRAMMI DISPONIBILI

La nostra gestione di magazzino mette a disposizione dell'utilizzatore un totale di 5 programmi diversi e precisamente:

PROGRAMMA 0

Serve per inserire in memoria i codici dei componenti con i dati ad essi relativi, cioè giacenza, quantità entrata, quantità uscita, prezzo vendita e prezzo d'acquisto.

PROGRAMMA 1

Serve per visualizzare sui display la giacenza, la quantità entrata, la quantità uscita, il prezzo vendita e il prezzo d'acquisto relativi a ciascun componente da noi inserito in memoria.

PROGRAMMA 2

Serve solo per fare i «carichi» di magazzino, cioè per registrare le entrate dei vari componenti man mano che si acquistano.

PROGRAMMA 3

Serve solo per fare gli scarichi di magazzino, cioè per registrare le uscite dei vari componenti man mano che vengono utilizzati.

PROGRAMMA 4

Serve solo per modificare il prezzo vendita o il prezzo d'acquisto di un componente già inserito in memoria.

Ciascuno di questi programmi impiega sia delle subroutine contenute nel «monitor», sia delle subroutine appositamente realizzate per questo scopo che troverete riportate in evidenza nel «listing» totale della nostra gestione di magazzino.

In pratica la gestione si articola su un programma principale che non appena avviato ci chiede subito quale dei 5 programmi a disposizione vogliamo utilizzare, facendo comparire sui display la scritta:

The logo consists of the letters 'SPL' in a bold, white, sans-serif font, centered within a dark rectangular background.

che significa «Scegli il Programma Lavoro».

Rispondendo **0** oppure **1** oppure **2** oppure **3** oppure **4** a questa domanda noi facciamo la nostra scelta, cioè facciamo girare il solo programma selezionato escludendo automaticamente gli altri quattro. Tanto per fare un esempio l'operazione di scelta del programma può essere paragonata all'operazione che ciascuno di voi compie quando in un bar o in una pizzeria si accosta al juke-box e pigliando uno o due tasti di selezione fa in modo che venga suonato solo il disco che gli interessa escludendo automaticamente tutti gli altri.

Come procedere per scegliere il programma verrà comunque spiegato più dettagliatamente nelle pagine che seguono con alcuni esempi pratici.

Il vero problema che dovremo affrontare adesso è invece scrivere in memoria il programma completo (vedi pag. 121 e seguenti), un'operazione questa che vi porterà via un po' di tempo ma che con un minimo di attenzione può essere portata a termine con successo da chiunque si cimenti nell'impresa.

PER INSERIRE IL PROGRAMMA IN MEMORIA

Come già anticipato, il nostro programma di «gestione» viene scritto nel primo «kappa» della memoria RAM, vale a dire in quella porzione di memoria presente sulla scheda CPU i cui indirizzi vanno da 0000 a 03FF.

La zona di memoria che occuperemo sarà comunque molto più ristretta rispetto a quella disponibile, infatti non vi sarà difficile notare che la prima istruzione del nostro programma deve essere inserita nella riga **0070** per terminare alla riga **0385** non solo ma anche nel corso del programma stesso vi sono delle zone di memoria inutilizzate (da **00Fd** a **00FF**, da **0261** a **0270**, da **028d** a **028F**, da **02E5** a **02FF** e da **0336** a **034F**) che potrete eventualmente impiegare per inserire altri programmi o subroutine da voi stessi realizzati a completamento di quelli già disponibili.

Le operazioni da compiere per l'inserimento in

memoria sono le seguenti:

- 1) Pigiare il tasto RESET
- 2) Pigiare sulla tastiera 0070
- 3) Pigiare i due tasti CONTROL - 0 ed automaticamente vedrete apparire sul display due numeri del tutto casuali che rappresentano il contenuto della locazione di memoria 0070.
- 4) Impostate sulla tastiera il codice E5 relativo alla riga 0070 poi pigiate i due tasti CONTROL - 0 per trasferire tale codice in memoria.
- 5) Sul display vi apparirà il numero di riga successivo, cioè 0071, seguito ancora da due numeri casuali.

6) Impostate sulla tastiera il codice 21 poi pigiate i due tasti CONTROL - 0 per trasferirlo in memoria.

7) Proseguite in questo modo fino ad arrivare alla riga 00FC dopo la quale, come già anticipato, si trova il primo spazio vuoto in memoria.

8) A questo punto, fino alla riga 00FF, noi memorizzeremo degli 00.

9) Giunti alla riga 0100 dovremo memorizzare un Cd (vedi tabella dei programmi) dopodiché procederemo con le righe successive fino alla 01F9.

10) La riga 01FA non viene sfruttata dai nostri programmi perciò anche in questa memorizzeremo uno 00.

11) Proseguiremo ancora con la riga 01Fb poi 01FC ecc. fino ad arrivare alla riga 0260.

12) Le locazioni da 0261 a 0270 non sono utilizzate dai programmi, perciò in tutte queste scriveremo degli 00.

13) Il successivo gruppo di istruzioni va scritto nelle righe dalla 0271 alla 028C, poi troveremo ancora uno spazio vuoto fra 028d e 028F che riempiamo con degli 00.

14) A partire dalla riga 0290 troviamo le istruzioni relative alla subroutine di «addizione», quindi dovremo scrivere tali istruzioni fino a raggiungere la riga 02C5, dopodiché troveremo le istruzioni del programma n. 4, quindi ancora uno spazio vuoto dalla locazione 02E5 fino alla 02FF che riempiamo con degli 00.

15) Da 0300 a 0335 trova alloggio la subroutine di «sottrazione» poi c'è ancora uno spazio vuoto dalla locazione 0336 alla 034F che riempiamo sempre con degli 00.

16) Infine dalla locazione 0350 alla locazione 0387 troveranno alloggio le istruzioni necessarie per far comparire sul display la scritta SPL ed a questo punto l'operazione inserimento può dirsi conclusa.

Inutile aggiungere che essendo molte le istruzioni da inserire in memoria, l'intera operazione deve essere eseguita con una cura del tutto particolare facendovi possibilmente aiutare da qualcuno che legga la tabella mentre voi scrivete.

Se vi accorgete di aver sbagliato un numero prima di pigiare CONTROL-0 potrete sempre riscriverlo in modo corretto; se no dovrete pigiare RESET e richiamare sul display la riga in cui avete commesso l'errore.

E inoltre consigliabile, non appena terminato di

scrivere il programma in memoria, di registrarlo subito su nastro ancor prima di provare se funziona perché se per caso avete commesso qualche errore, tentando di farlo «girare», potreste correre il rischio che vi si cancelli totalmente.

È pure consigliabile, man mano che inserite in memoria il vostro programma, controllare ogni 10-15 istruzioni che il numero di riga visualizzato sul display corrisponda effettivamente con quello indicato in tabella, perché se per caso saltaste inavvertitamente qualche istruzione, giunti alla fine della vostra opera dovrete praticamente ricominciare da capo.

REGISTRAZIONE DEL PROGRAMMA SU NASTRO

Come vi abbiamo accennato, quando avrete scritto tutto il programma, dovrete subito registrarlo su nastro ed a tale proposito noi vi diciamo che prima di passare alla registrazione, dovrete effettuare quest'ultima operazione che vi permetterà di far girare immediatamente tale programma quando lo andrete a rileggere.

1) Pigiare il tasto RESET.

2) Pigiare i due tasti CONTROL - 2 (quelli cioè che permettono di accedere ai registri della CPU).

3) Pigiare tante volte di seguito i due tasti CONTROL - 0 quante sono necessarie per veder apparire sul display la scritta PC seguita da quattro numeri casuali.

4) Impostate sulla tastiera il numero 0350 (corrispondente alla riga in cui è contenuta la prima istruzione del programma necessario per far comparire sul display la scritta SPL).



5) Pigiare i due tasti CONTROL - 0 (così facendo trasferirete tale numero all'interno del registro PC).

Giunti a questo punto potrete procedere alla registrazione vera e propria del programma seguendo le indicazioni testé fornite.

1) Collegate il registratore al microcomputer inserendo su di esso un nastro vergine senza coda completamente riavvolto.

2) Pigiare il tasto RESET sul microcomputer.

3) Pigiare i due tasti CONTROL - 5 in modo da far comparire la scritta:



4) Pigiare sul registratore i due tasti PLAY-REC in modo da predisporlo per la registrazione.

5) Pigiare sulla tastiera la lettera A o b a seconda dell'uscita a cui vi siete collegati.

6) Il registratore immediatamente si metterà in moto e dopo qualche istante vedrete sul display iniziare un conteggio molto rapido che partendo da 0000 terminerà a 03FF.

7) Attendete che il registratore si fermi poi estraete la vostra cassetta e scriveteci sopra in

bella vista «Programma Gestione Magazzino» in modo che non si possa confondere con le cassette che utilizzerete per i dati.

Nota: l'operazione che vi abbiamo fatto eseguire prima di registrare il programma su nastro, cioè scrivere PC = 0350, ci sarà estremamente utile quando rileggeremo tale programma dalla cassetta, infatti avendo già memorizzato nel registro PC l'indirizzo da cui vogliamo che inizi l'esecuzione, ci sarà sufficiente pigiare i due tasti CONTROL - 4 per veder comparire sui display la scritta SPL, senza doverci preoccupare ogni volta di riscrivere appunto il numero 0350 nel registro PC.

COLLAUDO DEL PROGRAMMA

Dopo aver registrato il programma su nastro, la prima operazione che vi consigliamo di fare è quella di provare a rileggerlo (senza spegnere il microcomputer) per assicurarvi che sia stato memorizzato alla perfezione.

- 1) Riavvolgete completamente il nastro.
- 2) Pignite il tasto RESET sul microcomputer.
- 3) Pignite i due tasti CONTROL - 6.
- 4) Pignite sul registratore il tasto PLAY in modo da predisporlo su «ascolto».
- 5) Pignite il tasto **A** oppure il tasto **b** a seconda dell'uscita a cui vi siete collegati.
- 6) Automaticamente il registratore si metterà in funzione e dopo qualche secondo sui display vedrete comparire dei numeri che partendo da 0000 aumenteranno progressivamente fino ad arrivare a 03FF.

Se tutto avviene come da noi descritto significa che il programma è stato registrato correttamente, quindi potrete subito iniziare il «collaudo»; se invece il microcomputer segnala un **errore** provate a registrarlo di nuovo oppure procuratevi una cassetta provvista di un nastro di miglior qualità.

Precisiamo che le cassette più idonee allo scopo sono quelle da 6+6 minuti, sprovviste di «coda» all'inizio, che già siamo in grado di fornirvi al prezzo di L. 1.500 cadauna.

Una volta letto il programma dal nastro la prima cosa che dovrete fare per collaudarlo è pigiare il tasto RESET, poi subito dopo pigiare i due tasti **CONTROL - 4**.

Automaticamente dovrà apparirvi sui display la scritta



cioè «Scegli il Programma di Lavoro» ed a questo punto dovrete pigiare il **tasto 0** per selezionare quel programma che vi permette di inserire dei «componenti» in memoria, infatti se non inseriamo nessun articolo in memoria gli altri quattro programmi non potranno mai funzionare.

Pigiando il tasto 0 automaticamente sui display vi apparirà la scritta



cioè il computer vi chiederà quale codice di «riconoscimento» volete assegnare al componente che state per inserire in memoria e qui è necessario aprire un capitolo a parte.

COME CODIFICARE UN ARTICOLO DI MAGAZZENO

Come in tutti i computer, per identificare all'interno della memoria i vari componenti inseriti, noi dovremo assegnare a ciascuno di essi un proprio «codice», cioè una particolare sigla che permetta al computer di ritrovare facilmente i dati di un componente ogni volta che gli vengono richiesti.

In altre parole, volendo per esempio utilizzare questa gestione di magazzino per dei componenti elettronici, noi non potremo inserire direttamente le sigle SN.7400 per gli integrati, o BC208 per i transistor o 120.000 ohm per le resistenze in quanto avendo ciascuna sigla una lunghezza diversa dall'altra si creerebbero dei problemi di «organizzazione» non indifferenti, bensì dovremo utilizzare per ogni componente un «codice» standard composto di 4 caratteri, non importa se alfabetici o numerici.

Per esempio potremo individuare ciascun componente con 4 numeri, cioè 0001-0002-0003-0004 ecc., oppure con una lettera seguita da tre numeri A001-A002-A003 o b001-b002-b003 (nota: le lettere che possiamo utilizzare sono quelle della tastiera esadecimale cioè A-b-C-d-E-F), oppure ancora due lettere e due numeri AF01-AF02-AF03-bd44-bC12 e così di seguito.

Nella scelta del codice dovremo possibilmente seguire un criterio che ci permetta di ricordarlo con estrema facilità, diversamente si perderanno gran parte dei vantaggi che il computer è in grado di offrire.

Tanto per fare un esempio, se dovessimo codificare dei condensatori, potrebbe essere consigliabile impiegare un codice composto da una «C» seguita dal valore di capacità espresso secondo il metodo giapponese, cioè:

C220 = 22 pF

C221 = 220 pF

C222 = 2.200 pF

C223 = 22.000 pF

C224 = 220.000 pF

dove l'ultimo numero che appare sulla destra sta ad indicare quanti «zero» occorre aggiungere alle prime due cifre per ottenere la capacità, cioè:

C220 = 22 + nessuno zero = 22 pF

C222 = 22 + due zero = 2.200 pF

C223 = 22 + tre zero = 22.000 pF

Per le resistenze potremmo impiegare un codice composto da una A seguita ancora da tre numeri che ci indichino gli ohm in codice giapponese, per esempio:

A010 = 1 ohm

A100 = 10 ohm

A101 = 100 ohm

A102 = 1.000 ohm
 A103 = 10.000 ohm
 A104 = 100.000 ohm
 A105 = 1 megaohm

Per i condensatori elettrolitici potremmo impiegare la lettera E seguita sempre da tre numeri che indichino la capacità, questa volta però espressa in «microfarad» anziché in «picofarad», quindi

E102 = cond. elettrolitico da 1.000 mF

E471 = cond. elettrolitico da 470 mF

e così di seguito.

Per gli integrati TTL della serie SN.74 potremo scrivere direttamente il numero tralasciando «SN», cioè:

7400 = integrato tipo SN7400

7490 = integrato tipo SN7490

e la stessa cosa dicasi anche per i C/MOS, cioè:

4011 = integrato tipo CD4011

4029 = integrato tipo CD4029

4528 = integrato tipo CD4528

Per ogni categoria di materiale, resistenze, condensatori, integrati, transistor ecc. utilizzeremo un **diverso nastrino**, così avremo meno possibilità di errore e risulterà più semplice ricercare i componenti.

Ribadiamo ancora che con un solo «kappa» di memoria è possibile inserire un massimo di **55 componenti** per volta, tuttavia se saprete suddividere in modo opportuno il vostro magazzino vi accorgete che questa non è una grossa limitazione anzi sotto certi punti di vista è un vantaggio in quanto avendo tutti i vostri dati memorizzati su nastro non correrete il rischio che questi vadano perduti ad esempio per un momentaneo black-out della tensione di rete.

Se invece tenessimo tutti i nostri dati in memoria, sarebbe sufficiente un impulso spurio captato per chissà quale motivo dal microcomputer per mandare in fumo il lavoro di giorni o di mesi.

Nota: questa limitazione non esisterà più quando avremo il floppy disk infatti in un solo floppy potremo memorizzare oltre 2.000 componenti.

Per non creare confusione su ciascuna cassetta dovreste scrivere in bella evidenza a quale categoria si riferisce, per esempio:

Cassetta n. 1 = resistenze 1/4 di watt

Cassetta n. 2 = integrati TTL

Cassetta n. 3 = integrati C/MOS

Cassetta n. 4 = condensatori a disco

Cassetta n. 5 = transistor

poi ogni sera o a fine settimana aggiornerete voce per voce facendo i carichi e gli scarichi.

Se durante la settimana avete acquistato una partita di transistor, per aggiornare le giacenze prenderete la vostra cassetta n. 5 e su di essa memorizzerete le «entrate» tramite il programma n. 2, poi dovreste nuovamente registrare il tutto su nastro per aggiornarlo con le variazioni effettuate.

Lavorando su una sola categoria di componenti ci vorrà meno tempo per trasferire i dati dalla cassetta alle RAM e lo stesso dicasi anche per l'operazione inversa, non solo ma se per caso commettete un errore di inserimento oppure vi si cancellasse qualcosa potreste rimediare molto più facilmente che non avendo in memoria tutto il magazzino al completo.

Per esperienza vi consigliamo sempre di registrare gli stessi dati su due nastri diversi oppure in due punti diversi del nastro in modo da conservare un «duplicato» da poter utilizzare in casi di emergenza.

Quando opererete la suddivisione in categorie del vostro magazzino, la prima operazione che dovrete compiere sarà quella di prepararvi un tabulato, come vedesi in fig. 1 nel quale, accanto ad ogni codice, scriverete la sigla del componente e nelle colonne adiacenti i dati ad esso relativi che dovrete poi inserire nel microcomputer, cioè «giacenza», «entrate totali», «uscite totali», «prezzo vendita» e «prezzo d'acquisto».

Nota: è ovvio che nella **descrizione** del componente potremo scrivere quello che vogliamo, per esempio le resistenze potrebbero diventare quaderni a quadretti, pillole, bulloni, scarpe, bottoni ecc..

L'importante è suddividere tutti questi componenti in categorie ed assegnare a ciascuno di essi un codice che sia il più mnemonico possibile, cioè che si possa ricordare facilmente senza dover ogni volta andarsi a consultare un'apposita tabella.

Ricordatevi che il computer si creerà a sua volta

Posizione riga	Codice	Resistenze 1/4 di watt	Giacenza	Entrate	Uscite	Prezzo vendita	Prezzo d'acquisto
1	A010	1 ohm	500	850	350	25	20
2	A100	10 ohm	25	100	75	25	20
3	A150	15 ohm	300	300	0	30	23
4	A470	47 ohm	325	380	55	30	23
5	A102	1.000 ohm	50	120	70	35	25
6	A473	47.000 ohm	20	80	60	35	28

Fig. 1 Prima di inserire tutti i vostri componenti di magazzino dentro al microcomputer preparatevi una tabella, con sopra riportato la «posizione di riga» in cui volete che ciascun componente venga collocato, il «codice» che volete assegnargli nonché tutti gli altri dati richiesti cioè giacenza, entrata, uscita, lire vendita e prezzo acquisto.

in memoria una tabella simile a quella di fig. 1, composta di 55 «righe», cioè una riga per ogni componente proprio come abbiamo fatto noi con le resistenze, pertanto per ogni componente che inserirete dovrete anche specificare la posizione in cui volete che venga sistemato entro tale tabella, diversamente esso non saprebbe dove collocarlo.

Per esempio se volete che il vostro componente venga memorizzato nella prima riga, dovrete dirgli «Posizione 1»; se invece volete che venga memorizzato nella 5ª riga dovrete dirgli «Posizione 5» e così di seguito.

COME SI PREPARA IL MAGAZZENO

Dopo aver suddiviso tutti gli articoli del vostro magazzino in categorie comprendenti ciascuna non più di 55 componenti ed aver preparato le relative tabelle con i codici e i dati da assegnare a ciascuno di essi, come vi abbiamo fatto vedere in fig. 1, giungerà finalmente il momento di inserire questi dati in memoria.

Per far questo prendete la cassetta in cui avete registrato il vostro programma e fatela leggere al microcomputer seguendo le indicazioni fornitevi in precedenza.

Appena terminato di leggere il programma pigiate il tasto RESET e come al solito vedrete apparire la scritta:



pigiate i due tasti CONTROL - 4 e sempre automaticamente il computer vi chiederà di scegliere il programma di lavoro facendo comparire sui display la scritta



Sapete già che il programma necessario per inserire in memoria i vostri dati è il **programma n. 0**, quindi alla domanda SPL rispondete pigiando il tasto 0. Immediatamente il computer farà comparire la scritta



per chiedervi appunto quale codice volete assegnare al vostro componente.

Come già detto il codice deve sempre essere composto di **4 caratteri** scelti fra quelli presenti sulla tastiera, cioè i numeri da 0 a 9 oppure le lettere da A a F.

Per esempio, rifacendoci alla tabella precedente, potremmo scrivere sulla tastiera **A010**, cioè il codice relativo alla resistenza da 1 ohm.



Se per caso ci sbagliassimo nello scrivere il codice, potremmo riparare all'errore riscrivendolo in

modo corretto senza dover pigiare nessun tasto particolare.

Una volta terminato di scrivere il codice, per trasferirlo in memoria e andare avanti col programma dovremo pigiare i due tasti CONTROL - 0 ed a questo punto possono verificarsi due eventi distinti:

1) se il codice da noi impostato non rientra nello standard di cui abbiamo appena discusso (cioè non è composto di 4 caratteri) oppure è già presente in memoria in quanto lo abbiamo assegnato in precedenza ad un altro componente, il computer non lo accetta e ritorna a chiedere «Codice?».

2) Ammesso invece che tutto risulti regolare, il computer farà comparire sui display la scritta



cioè «Posizione?», per richiederci in quale posizione vogliamo che si inserisca quel componente.

Nel nostro caso, trattandosi del primo elemento della categoria, lo collocheremo ovviamente in posizione 1, tuttavia questo non è un obbligo in quanto se ci fa piacere potremmo anche collocarlo in posizione 23, tenendo però presente in seguito che la posizione 23 è già occupata da un componente perché il microcomputer non lo segnala.

Nota: fate attenzione nel rispondere a questa domanda perché se per caso indicaste una posizione già occupata da un altro articolo, il precedente articolo verrà automaticamente cancellato da quello nuovo. Ad esempio se nella riga 5 avete già scritto il codice A102 e scrivendo A473 lo mettete ancora nella riga 5, il computer cancellerà A102 ed in suo luogo verrà memorizzato A473.

In pratica, come già anticipato, le posizioni disponibili sono in tutto 55 (infatti questo è il numero massimo di componenti che possiamo inserire in un «kappa» di RAM), quindi noi dovremo impostare sulla tastiera un numero compreso fra 1 e 55 (il computer è stato programmato in modo da non accettare numeri maggiori di 55 pertanto se tentassimo di scrivere un numero più alto, ci tornerrebbe a chiedere POS).

Impostate il numero 1 sulla tastiera



(oppure un qualsiasi altro numero compreso fra 1 e 55) poi pigiate i due tasti CONTROL - 0 e automaticamente sui display vedrete comparire:



cioè il calcolatore vi chiederà la «giacenza» di magazzino del vostro componente.

Precisiamo che come giacenza noi possiamo scrivere un numero qualsiasi compreso fra 0 e 999.999 con possibilità di correzione automatica in caso di errore. Ad esempio se per caso ci sbagliassimo a scrivere tale numero, potremmo pigiare tante volte di seguito il tasto 0 quante sono ne-

cessarie per farlo uscire completamente sulla sinistra del display e riscrivere quindi la giacenza corretta.

Qualora pigiassimo una lettera al posto di un numero, il computer è stato programmato per non accettarla.

Dopo aver scritto la giacenza relativa al vostro componente, per esempio 500, pigiate come al solito i due tasti CONTROL - 0 per trasferirla in memoria ed automaticamente vi apparirà:

E.

che in pratica significa «Entrata Totale», cioè se ieri sono entrati 350 pezzi e oggi ve ne giungono 130, il microcomputer vi deve dire che in totale ne avete acquistati 480.

In questo modo voi alla fine di ogni mese o di ogni anno potrete sapere quanti pezzi di un determinato componente sono entrati in magazzino.

E' ovvio che ogni volta che voi farete un «carico», cioè registrerete un'entrata di magazzino, questa voce verrà automaticamente aumentata così come verrà automaticamente aumentata la giacenza.

Per esempio se in un mese avete acquistato 480 pezzi e ne avete venduti 470, quindi nella voce giacenza ne esistono 10, entrando oggi altri 130 pezzi, nella giacenza ci ritroveremo automaticamente $10 + 130 = 140$ pezzi e nella quantità entrata totale $480 + 130 = 610$ pezzi.

Dopo aver inserito in memoria la quantità entrata, pigiate i due tasti CONTROL - 0 ed automaticamente vi apparirà:

U.

che significa «Uscita totale», cioè la quantità che avete venduto a partire dall'inizio dell'anno o del mese fino al giorno in cui andrete a controllare questo dato.

Ad esempio se in magazzino all'inizio del mese avevamo giacenti 350 pezzi e in giorni diversi ne abbiamo venduti $15 + 22 + 8 + 34 = 79$ pezzi, controllando le uscite totali troveremo appunto questo numero (cioè 79), mentre controllando la giacenza, se non vi sono state delle entrate nel frattempo, troveremo $350 - 79 = 271$ pezzi.

Il nostro programma infatti è concepito in modo che eseguendo uno scarico, automaticamente diminuisce la giacenza e la quantità che viene tolta dalla giacenza va ad incrementare il totale delle uscite.

Quando avremo impostato il «totale uscite», dovremo come al solito pigiare i due tasti CONTROL - 0 ed automaticamente ci apparirà:

L.

che significa «Lire del prezzo vendita».

Questo è un numero che una volta inserito si può modificare solo tramite il programma n. 4.

Avendo a disposizione nella voce U il quantitativo totale venduto durante l'anno possiamo anche sapere quanto abbiamo incassato in totale vendendo questo componente, infatti per raggiungere lo scopo ci basterà moltiplicare il numero che troviamo nella voce U per il numero che troviamo nella voce L.

Dopo aver scritto il prezzo vendita, pigiate ancora i due tasti CONTROL - 0 ed automaticamente vedrete apparire:

P.

che significa «Prezzo d'acquisto».

Anche questo è un numero che una volta inserito si può cambiare solo con il programma n. 4, numero che ci sarà molto utile per calcolare quanto abbiamo speso in totale durante l'anno per l'acquisto del componente ed eventualmente ricavare il guadagno.

Una volta scritto il prezzo d'acquisto pigeremo ancora i due tasti CONTROL - 0 ed automaticamente vedremo comparire sul display la scritta SPL, cioè «Scegli il programma di lavoro», quindi potremo scegliere di nuovo il programma che ci interessa.

Ovviamente, dovendo inserire altri componenti in memoria, sceglieremo ancora il **programma n. 0** e cominceremo a scrivere i dati del secondo componente della tabella, poi proseguiremo col terzo e così di seguito fino ad inserirli tutti.

Come potrete notare il nostro programma, così come è concepito, potrebbe servire anche per scopi diversi da una semplice gestione di magazzino, per esempio potremmo utilizzarlo per tenere la contabilità di clienti e fornitori.

Basti pensare che assegnando un codice a ciascun fornitore, cioè A001 = Ferramenta FR A002 = Ditta Pellicani ecc. ecc. potrete inserire nella voce E. le entrate (in migliaia di lire), nella voce U. le uscite (sempre in migliaia di lire) e ritrovarvi automaticamente nella voce G la differenza fra il dare e l'avere.

PER CONTROLLARE I DATI INSERITI (Programma n. 1)

Ammettiamo per esempio di aver inserito in memoria, tutti i nostri componenti e di averli registrati su nastro. Se ora volessimo conoscere i dati ad esempio del condensatore poliesteri da 47.000 pF a cui abbiamo assegnato il codice C473 prenderemo la cassetta dove sono registrati i condensatori e li trasferiremo in memoria sul microcomputer dopodiché pigeremo CONTROL - 4 e vedremo apparire la scritta

SPL

ciòè «Scelta programma Lavoro» e poiché vogliamo vedere dei dati non dovremo fare altro che pigliare il **tasto 1** per selezionare appunto il programma n. 1.

Immediatamente sui display comparirà la domanda:

infatti il calcolatore per andare a cercare in memoria i nostri dati ha bisogno che gli si fornisca il codice del componente.

Impostate sulla tastiera il codice C473 oppure il codice che avete assegnato al componente inserito in precedenza, quindi pigiate i due tasti CONTROL - 0.

A questo punto possono verificarsi due ipotesi:

1) Se il codice da voi impostato è inesatto oppure non esiste ancora in memoria perché non lo avete mai inserito, il microcomputer tornerà a chiedervi «Codice?».

2) Se il componente esiste in memoria, vedrete invece apparire sui display la lettera G. seguita dalla giacenza che voi avete memorizzato, per esempio:

Dopo aver controllato se la giacenza è esatta, pigiate un tasto qualsiasi ed automaticamente vedrete comparire sui display la lettera E. seguita dalle «entrate» che voi avete memorizzato, per esempio:

Controllate se questo numero è esatto poi pigiate un tasto qualsiasi ed automaticamente vedrete apparire sui display la lettera U. seguita dalle «uscite» che voi avete memorizzato, per esempio:

Controllate se il numero è esatto poi pigiate un tasto qualsiasi ed automaticamente vi apparirà la lettera L. seguita dal prezzo vendita del componente, per esempio:

Pigiate un tasto qualsiasi ed automaticamente vi apparirà la lettera P. seguita dal costo del componente, per esempio:

Pigiate ancora un tasto qualsiasi ed automaticamente vi apparirà la scritta SPL; a questo punto, se volete vedere i dati relativi ad un altro componente, pigiate 1 per scegliere il programma n. 1, se no pigiate il numero relativo al programma che vi interessa far «girare».

PER FARE UN CARICO DI MAGAZZENO (Programma n. 2)

Come già detto, una volta inseriti in memoria i codici e i dati relativi ad un certo numero di componenti, noi possiamo aggiornare la giacenza di questi componenti servendoci dei programmi n. 2 e n. 3 il primo dei quali ci permette di registrare le «entrate» di magazzino ed il secondo le «uscite».

Supponiamo per esempio di aver acquistato 1.000 pezzi del nostro condensatore da 47.000 pF la cui giacenza precedente era di 350 unità.

Per registrare questo «carico», quando sui display compare la scritta

noi dovremo pigiare il **tasto 2** in modo da selezionare il **programma n. 2**. Automaticamente sui display comparirà la domanda:

infatti anche in questo caso il computer ha bisogno del codice per poter ritrovare in memoria i dati relativi al componente.

Impostate sulla tastiera il vostro codice, cioè C473 oppure qualsiasi altro da voi utilizzato, quindi pigiate i due tasti CONTROL - 0.

Automaticamente sui display vi apparirà la scritta:

per indicarvi che il computer è in attesa che voi scriviate la quantità **entrata**.

Impostate sulla tastiera il numero di componenti che sono entrati in magazzino (1.000 nel nostro esempio), quindi pigiate i due tasti CONTROL - 0 per trasferire tale numero in memoria.

Come già anticipato il numero 1.000 che voi avete inserito con questo programma verrà automaticamente addizionato sia alla giacenza precedente (voce G.), sia alle entrate totali (voci E.).

Sui display vi apparirà la scritta SPL ed a questo punto se volete controllare se l'addizione è stata effettuata in modo corretto, potrete selezionare il programma n. 1 e rivedervi tutti i dati del componente seguendo le indicazioni fornite nel precedente paragrafo: constaterete così che la giacenza G. è passata da 350 pezzi a 1.350 pezzi mentre le «entrate totali» che in precedenza erano 3.000, adesso sono 4.000. Tutti gli altri numeri invece rimarranno invariati, cioè avremo:

G. = 1.350

E. = 4.000

U. = 2.650

L. = 150

P. = 100

PER REGISTRARE LE USCITE (programma n. 3)

Supponiamo a questo punto che entri un cliente nel nostro magazzino ed acquisti per esempio 230 condensatori da 47.000 pF; ovviamente occorre scaricare questa merce per avere la giacenza aggiornata e per ottenere questo si può procedere come segue:

- 1) Quando il calcolatore vi chiede



premete il **tasto n. 3** per selezionare appunto il programma relativo agli «scarichi».

- 2) Automaticamente sui display vi apparirà la domanda



- 3) Scrivete sulla tastiera il codice C473 quindi premete CONTROL - 0.

- 4) Sui display vi apparirà la scritta



per chiedervi la quantità **uscita**.

- 5) Scrivete sulla tastiera il numero 230, cioè i componenti che avete venduto quindi premete i due tasti CONTROL - 0.

A questo punto il computer eseguirà la sottrazione dei 230 pezzi dalla giacenza precedente e nello stesso tempo addiziona questo valore alle «uscite complessive», cioè alla voce U.

Subito dopo farà comparire sui display la solita scritta SPL per invitarvi a scegliere un nuovo programma, oppure ancora questo programma nel caso vogliate effettuare un secondo scarico.

Come già detto, qualora voleste controllare se lo scarico è stato effettuato in modo corretto, potrete selezionare il programma n. 1 e andare a rivedervi i dati relativi al vostro condensatore.

Constaterete così che la giacenza G. da 1.350 pezzi è scesa a 1.120 pezzi (infatti $1.350 - 230 = 1.120$) mentre le uscite totali (voce U.) sono salite da 2.650 pezzi a 2.880 pezzi (infatti $2650 + 230 = 2.880$). Tutti gli altri numeri saranno invece rimasti invariati, cioè avremo:

G. = 1.120
E. = 4.000
U. = 2.880
L. = 150
P. = 100

PER MODIFICARE IL PREZZO VENDITA E IL PREZZO D'ACQUISTO DI UN QUALSIASI COMPONENTE (programma n. 4)

Supponiamo di aver già inserito in memoria i dati (cioè giacenza, entrate, uscite, prezzo vendita, prezzo acquisto) relativi ad un certo numero di resistenze e di averli trasferiti su nastro.

Un determinato giorno il prezzo di acquisto di

un componente, per esempio della resistenza da 1.000 ohm 1/4 watt il cui codice supponiamo che sia A102, viene aumentato da 15 a 20 lire.

È ovvio che noi dovremo registrare tale aumento anche all'interno del computer, non solo ma per mantenere inalterato il margine di guadagno dovremo ritoccare anche il prezzo vendita.

Per ottenere questo le operazioni da compiere sono elementari.

- 1) Prendete la cassetta relativa alle resistenze da 1/4 watt e caricate il contenuto nel computer.

- 2) Premete il tasto RESET e subito dopo premete i due tasti CONTROL - 4.

- 3) Sui display vi apparirà come al solito la scritta



cioè «Scegli il Programma di Lavoro».

- 4) Premete il tasto 4 per selezionare il programma che vi permette di cambiare il prezzo vendita e il prezzo d'acquisto.

- 5) Automaticamente sui display vi apparirà la scritta



- 6) Scrivete sulla tastiera il codice A102 oppure il codice del componente a cui volete cambiare il prezzo, poi premete i due tasti CONTROL - 0.

- 7) Automaticamente sui display vi apparirà la scritta



per chiedervi il nuovo prezzo vendita di tale componente.

8) Ammettendo per esempio che il nuovo prezzo che volete stabilire per la resistenza da 1.000 ohm sia 35 lire, scrivete sulla tastiera il numero 35 poi premete i due tasti CONTROL - 0 per trasferirlo in memoria.

- 9) Sui display vi apparirà la scritta



per chiedervi il nuovo prezzo di acquisto di tale componente.

10) Nel nostro caso, essendo il nuovo prezzo d'acquisto della resistenza da 1.000 ohm uguale a 20 lire, dovremo scrivere sulla tastiera il numero 20, poi dovremo premere i due tasti CONTROL - 0 per trasferire questo numero in memoria.

Dopo aver premuto CONTROL - 0 sui display tornerà a comparire come al solito la scritta SPL ed a questo punto noi potremo selezionare quello dei 5 programmi disponibili che ci interessa ancora utilizzare. Se per esempio selezionassimo il programma 1 e andassimo a vedere i dati della resistenza da 1.000 ohm, troveremo:

L. = 35
P. = 20

cioè i nuovi prezzi da noi inseriti in precedenza.

PER REGISTRARE SU NASTRO I DATI RELATIVI AD UN GRUPPO DI 55 COMPONENTI

Una volta inseriti in memoria tutti i componenti di una determinata categoria, oppure anche dopo un'operazione di carico o di scarico, per registrare questi dati su nastro dovremo seguire una procedura leggermente diversa rispetto a quella utilizzata per la memorizzazione del programma, infatti bisogna tener presente che i dati stessi sono contenuti nel secondo «kappa» della memoria RAM, non nel primo come il programma.

L'esatto modo di procedere in questi casi è comunque il seguente:

- 1) Pigiare il tasto RESET.
- 2) Inserite sul registratore la cassetina con il nastro completamente riavvolto.
- 3) Scrivete sulla tastiera il numero 0022.
- 4) Pigiare i due tasti CONTROL - 0.
- 5) Automaticamente vi appariranno sui display due numeri casuali che rappresentano il contenuto della locazione di memoria 0022.
- 6) Scrivete sulla tastiera 00 quindi pigiate i due tasti CONTROL - 0 per trasferire tale numero in memoria.
- 7) Sui display vi apparirà il numero di riga successivo, cioè 0023 seguito ancora da due numeri casuali.
- 8) Scrivete sulla tastiera 04 quindi pigiate ancora i due tasti CONTROL - 0 per trasferire tale numero in memoria.
- 9) Pigiare i due tasti CONTROL - 5, quelli cioè relativi alla registrazione.
- 10) Pigiare sul registratore i tasti PLAY-REC in modo da predisporlo per registrare.
- 11) Pigiare il tasto A oppure il tasto B a seconda dell'uscita a cui vi siete collegati sull'interfaccia.
- 12) Immediatamente il registratore si metterà in moto e dopo alcuni secondi sui display vedrete iniziare un conteggio progressivo che partendo da 0400 terminerà a 07FF.
- 13) Giunti a questo punto l'operazione «scrittura» può considerarsi conclusa, tuttavia prima di riporre il nastro vi consigliamo sempre di provare a rileggerlo per vedere se la registrazione è avvenuta

nel migliore dei modi. Precisiamo che per rileggere i dati, una volta registrati su nastro, si può seguire la stessa procedura indicata in precedenza per il programma, cioè non è necessario scrivere all'inizio nelle locazioni 0022-0023 il numero 0400 in quanto il computer è in grado di capire autonomamente in quale «kappa» i dati stessi debbono venire caricati, senza che dobbiamo dirglielo noi.

CONCLUSIONI

Giunti a questo punto crediamo proprio di avervi detto tutto quello che si poteva riguardo la nostra gestione di magazzino, quindi ci congediamo senz'altro da voi con la speranza che tutto questo chiacchierare non vi abbia annoiato, bensì vi sia servito per comprendere che il microcomputer non è solo un oggetto per «giocare» ma è anche e soprattutto un prezioso collaboratore per le nostre attività quotidiane.

Precisiamo che questo è solo un esempio molto limitato delle capacità di tale macchina, infatti quando avremo a disposizione il floppy-disk, il video, la tastiera alfanumerica e la stampante, vedrete che si riusciranno a fare delle gestioni molto più «raffinate», conservando in memoria un numero molto maggiore di articoli e soprattutto un numero molto maggiore di informazioni per ciascun articolo.

Oggi però, con quello che avevamo a disposizione, non ci si poteva certo chiedere di più né era nostra intenzione darvi di più in quanto, come precisato all'inizio, tutti i programmi che vi abbiamo fornito hanno solo ed esclusivamente uno scopo dimostrativo per stimolarvi a realizzare voi stessi dei programmi di pratica utilità. Possiamo solo aggiungere, nel caso a qualcuno interessi provare questo programma ma non abbia intenzione di scriversele tutto da solo in quanto l'operazione richiederebbe troppo tempo, che sono disponibili un certo numero di cassetine con l'intero programma già registrato, pronto per essere caricato in memoria e fatto girare (semplicemente pigiando CONTROL - 4).

Chi ne fosse interessato potrà richiederle alla nostra redazione al prezzo di L. 2.500 cadauna.

Nelle pagine che seguono troverete riportate tutte le istruzioni che è necessario inserire in memoria per poter effettuare la gestione di magazzino descritta in questo articolo.

Tale gestione è stata volutamente suddivisa in sottoprogrammi e subroutine per facilitarne la comprensione.

Nota: nell'inserire le istruzioni in memoria controllate sempre che il numero di riga che compare sui display corrisponda con quello da noi indicato in tabella.

Quando troverete scritto per esempio «da 0261 a 0270 riempire con 00» significa che le righe dalla 0261 alla 0270 non sono utilizzate dal programma, quindi in ognuna di esse va posto uno 00.



Subroutine per visualizzare la parola CODICE sui display

Riga	Dati	Mnemonico	Descrizione
0070	E5	PUSH HL	Salva il contenuto della coppia di registri HL nello Stack, poi carica nella locazione 0007 il numero esadecimale 0C corrispondente alla lettera C.
0071 0072 0073	21 07 00	LD HL,0007	
0074 0075	36 0C	LD(HL),0C	
0076	2d	DEC L	Carica nella locazione 0006 il numero esadecimale 00 corrispondente alla lettera O.
0077 0078	36 00	LD(HL),00	
0079	2d	DEC L	Carica nella locazione 0005 il numero esadecimale 0d corrispondente alla lettera d.
007A 007b	36 0d	LD(HL),0d	
007C	2d	DEC L	Carica nella locazione 0004 il numero esadecimale 01 corrispondente alla lettera I.
007d 007E	36 01	LD(HL),01	
007F	2d	DEC L	Carica nella locazione 0003 il numero esadecimale 0C corrispondente alla lettera C.
0080 0081	36 0C	LD(HL),0C	
0082	2d	DEC L	Carica alla locazione 0002 il numero esadecimale 0E corrispondente alla lettera E.
0083 0084	36 0E	LD(HL),0E	
0085	2d	DEC L	Carica nella locazione 0001 il numero esadecimale 11 che equivale a display spento.
0086 0087	36 11	LD(HL),11	
0088	2d	DEC L	Carica nella locazione 0000 il numero esadecimale 45 che equivale ad una T sdraiata.
0089 008A	36 45	LD(HL),45	
008b 008C 008d	Cd EC 80	CALL 80EC	Visualizza il tutto sui display.
008E	E1	POP HL	Ripesca dallo stack il precedente contenuto della coppia di registri HL, poi ritorna ad eseguire il programma principale
008F	C9	RET	

Subroutine per leggere i dati dalla memoria e visualizzarli sui display

0090	C5	PUSH BC	La CPU salva il contenuto dei registri che deve utilizzare trascrivendolo nello Stack
0091	d5	PUSH DE	
0092	E5	PUSH HL	
0093	F5	PUSH AF	

Riga	Dati	Mnemonico	Descrizione
0094 0095 0096	21 01 00	LD HL,0001	Queste due istruzioni servono alla CPU, insieme a quelle di riga 00C1 e seguenti, per ricercare in memoria la posizione in cui si trovano i dati relativi al componente in esame
0097 0098	10 28	DJNZ,DIS = 28	
0099	1A	LD A,(DE)	La CPU legge le prime due cifre del numero dalla memoria e le visualizza sui display 0-1
009A 009b 009C	Cd 3C 80	CALL 803C	
009d	13	INC DE	La CPU legge le due cifre centrali del numero dalla memoria e le visualizza sui display 2-3
009E	1A	LD A,(DE)	
009F	23	INC HL	
00A0	23	INC HL	
00A1 00A2 00A3	Cd 3C 80	CALL 803C	La CPU legge le ultime due cifre del numero dalla memoria e le visualizza sui display 4-5
00A4	13	INC DE	
00A5	1A	LD A,(DE)	
00A6	23	INC HL	
00A7	23	INC HL	La CPU provvede a spegnere tutti gli «0» non significativi posti davanti al numero vero e proprio ponendo un 11 nelle relative locazioni di memoria
00A8 00A9 00AA	Cd 3C 80	CALL 803C	
00Ab	23	INC HL	La CPU provvede a spegnere tutti gli «0» non significativi posti davanti al numero vero e proprio ponendo un 11 nelle relative locazioni di memoria
00AC	36	LD (HL),11	
00Ad	11		
00AE 00AF	06 05	LD B,05	
00b0	2b	DEC HL	
00b1	7E	LD A,(HL)	
00b2	b7	OR A	
00b3 00b4	20 07	JR NZ,DIS = 07	
00b5 00b6	36 11	LD (HL),11	
00b7	05	DEC B	
00b8	78	LD A,B	
00b9	b7	OR A	
00bA 00bb	20 F4	JR NZ,DIS = F4	
00bC	F1	POP AF	La CPU ripescava dallo Stack il vecchio contenuto dei registri poi ritorna ad eseguire il programma principale
00bd	E1	POP HL	
00bE	D1	POP DE	
00bF	C1	POP BC	
00C0	C9	RET	

Riga	Dati	Mnemonico	Descrizione
00C1 00C2 00C3	21 10 00	LD HL,0010	Vedi riga 0094 e seguenti
00C4	19	ADD HL,DE	
00C5	54	LD D,H	
00C6	5d	LD E,L	
00C7 00C8	18 Cb	JR DIS = 18	
00C9 00CA 00Cb	Cd 90 00	CALL 0090	La CPU visualizza il numero che ha letto dalla memoria sui display facendolo precedere da un simbolo tipo G.-E.-U.-L.-P. dipendente dal contenuto del registro A
00CC 00Cd 00CE	Cd EC 80	CALL 80EC	
00CF 00d0	0E F7	LD C,F7	
00d1 00d2	Ed 79	OUT (C),A	
00d3 00d4 00d5	Cd AA 80	CALL 80AA	
00d6	C9	RET	La CPU attende che si pigi un tasto qualsiasi poi ritorna ad eseguire il programma principale
Programma n. 4 per modificare prezzo vendita e costo			
00d7	78	LD A,B	Se il codice da noi impostato non è presente in memoria nella relativa tabella, la CPU segnala l'errore chiedendo di nuovo «CODICE»
00d8	b7	ORA	
00d9 00dA	20 03	JR NZ,03	
00db 00dC 00E1	C3 00 01	JP 0100	La CPU carica nei registri DE l'indirizzo da cui inizia la tabella dei dati in memoria poi gli addiziona il numero 0009
00dE 00dF 00E0 00E1	Ed 5b 4b 01	LD DE,(014b)	
00E2 00E3 00E4	21 09 00	LD HL,0009	
00E5	19	ADD HL,DE	
00E6	Eb	EX HL,DE	
00E7 00E8	3E 47	LD A,47	La CPU aspetta che noi impostiamo il nuovo prezzo vendita sulla tastiera, poi lo trasferisce in memoria
00E9 00EA 00Eb	Cd 21 02	CALL 0221	
00EC 00Ed 00EE	Cd Fb 01	CALL 01Fb	
00EF	13	INC DE	La CPU aumenta di 1 il contenuto dei registri DE poi carica uno 0C nel registro A
00F0 00F1	3E 0C	LDA,0C	

Riga	Dati	Mnemonico	Descrizione
00F2 00F3	06 01	LD B,01	La CPU aspetta che noi impostiamo il nuovo prezzo d'acquisto sulla tastiera poi lo trasferisce in memoria
00F4 00F5 00F6	Cd 21 02	CALL 0221	
00F7 00F8 00F9	Cd Fb 01	CALL 01Fb	
00FA 00Fb 00FC	C3 50 03	JP 0350	La CPU ritorna a far comparire sui display la scritta SPL
00Fd 00FF 00FF	00 00 00		Righe non occupate dal programma
Routine per l'ingresso ed il controllo del «codice»			
0100 0101 0102	Cd 70 00	CALL 0070	La CPU fa comparire sui display la scritta CODICE poi si mette in attesa che venga pigiato qualche tasto.
0103 0104 0105	Cd AA 80	CALL 80AA	La CPU salva il contenuto del registro A ricopiandolo su B, esegue la subroutine 8067 (vedi relativo articolo) poi torna a copiare il contenuto di B su A.
0106	47	LD B,A	
0107 0108 0109	Cd 67 80	CALL 8067	La CPU controlla se il tasto pigiato rientra fra quelli consentiti, diversamente ritorna alla riga di programma 0100, cioè pone di nuovo la domanda «codice?».
010A	78	LD A,B	
010b 010C	d6 11	SUB 11	La CPU controlla se è stato pigiato CONTROL-0 e in caso affermativo capisce che il codice è completo e passa ad eseguire la riga di programma 0120; se no prosegue con la riga 0114.
010d 010E	30 F1	JR NC,F1	
010F	78	LD A,B	La CPU visualizza il numero da noi impostato sul display 3, spostando contemporaneamente di una posizione verso sinistra i numeri già presenti.
0110 0111	FE 10	CP,10	
0112 0113	28 0C	JR Z,0C	La CPU si mette in attesa che pigiamo un nuovo tasto e quando ciò avviene carica il relativo numero nel registro B poi ritorna alla riga 010b per controllare se il tasto pigiato rientra nella norma.
0114 0115 0116	Cd 0C 80	CALL 800C	
0117 0118 0119	Cd EC 80	CALL 80EC	
011A 011b 011C	Cd AA 80	CALL 80AA	
011d	47	LD B,A	
011E 011F	18 Eb	JR,Eb	

Programma per ricercare la posizione del codice nella tabella interna.

Riga	Dati	Mnemonico	Descrizione
0120 0121 0122	3A 06 00	LD A,(0006)	La CPU controlla se il codice è composto da 4 caratteri come richiesto se no ritorna ad eseguire la riga di programma 0100, cioè ritorna a chiedere «codice?»
0123 0124	FE 11	CP 11	
0125 0126	28 d9	JR Z d9	
0127 0128 0129	Cd 55 80	CALL 8055	La CPU carica il codice che noi abbiamo impostato nella coppia di registri DE
012A 012b 012C	21 00 04	LD HL,0400	La CPU carica nella coppia di registri HL il numero 0400, indirizzo iniziale della tabella dei codici in memoria.
012d 012E	06 01	LD B,01	La CPU esplora tutta la tabella dei codici in memoria alla ricerca del codice da noi impostato sulla tastiera. Se lo trova scrive nel registro B il numero corrispondente alla posizione di tale codice in tabella. Se non lo trova scrive nel registro B uno 00. Terminata la ricerca passa ad eseguire l'istruzione contenuta nella riga 0143
012F	7E	LD A,(HL)	
0130	bb	CP E	
0131	23	INC HL	
0132 0133	20 04	JR NZ,04	
0134	7E	LD A,(HL)	
0135	bA	CP D	
0136 0137	28 0b	JR Z,0b	
0138	04	INC B	
0139	78	LD A,B	
013A 013b	FE 38	CP 38	
013C 013d	28 03	JR Z,03	
013E	23	INC HL	
013F 0140	18 EE	JR EE	
0141 0142	06 00	LD B,00	
0143 0144 0145	C3 75 01	JP,0175	Salta alla riga 0175 o ad altre righe a seconda del programma selezionato.

Programma n. 1 per visualizzare i dati relativi a ciascun componente

0146	78	LD A,B	La CPU controlla se il codice da noi impostato è presente in memoria, se no ci richiede: «CODICE»
0147	b7	OR A	
0148 0149	28 b6	JR Z,b6	
014A 014b 014C	11 70 04	LD DE,0470	La CPU carica nella coppia di registri DE il numero 0470

Riga	Dati	Mnemonico	Descrizione
014D 014E	3E 42	LD A,42	La CPU legge in memoria la giacenza e visualizza il numero sul display preceduto da una G
014F 0150 0151	Cd C9 00	CALL 00C9	
0152	13	INC DE	La CPU legge le entrate complessive in memoria e visualizza il numero sul display facendolo precedere da una E
0153	13	INC DE	
0154	13	INC DE	
0155 0156	3E 06	LD A,06	
0157 0158 0159	Cd C9 00	CALL 00C9	La CPU legge le uscite complessive in memoria e visualizza il numero sul display facendolo precedere da una U
015A	13	INC DE	
015b	13	INC DE	
015C	13	INC DE	La CPU legge il prezzo vendita in memoria e lo visualizza sul display facendolo precedere da una L
015d 015E	3E 41	LD A,41	
015F 0160 0161	Cd C9 00	CALL 00C9	La CPU legge in memoria il costo e lo visualizza sul display facendolo precedere da una P
0162	13	INC DE	
0163	13	INC DE	
0164	13	INC DE	La CPU ritorna a visualizzare la scritta SPL
0165 0166	3E 47	LD A,47	
0167 0168 0169	Cd C9 00	CALL 00C9	La CPU controlla se il codice esiste già in memoria e in caso affermativo chiede nuovamente «codice?»
016A	13	INC DE	
016b	13	INC DE	
016C	13	INC DE	
016d 016E	3E 0C	LD A,0C	La CPU controlla se il codice esiste già in memoria e in caso affermativo chiede nuovamente «codice?»
016F 0170 0171	Cd C9 00	CALL 00C9	
0172 0173 0174	C3 50 03	JP 0350	

Programma n. 0 per inserire in memoria i dati di un nuovo componente

0175	78	LD A,B	La CPU controlla se il codice esiste già in memoria e in caso affermativo chiede nuovamente «codice?»
0176	b7	OR A	
0177 0178	20 87	JR NZ,87	

Riga	Dati	Mnemonico	Descrizione
0179 017A 017b	Cd 67 80	CALL 8067	La CPU fa comparire sui display la scritta POS, cioè «Posizione» per chiederci in quale posizione vogliamo collocare il nostro codice in tabella. Rispondendo per esempio a questa domanda con il n. 5, il codice impostato in precedenza verrà memorizzato nella 5ª riga di tale tabella sulla RAM
017C 017d	3E 17	LD A,17	
017E 017F 0180	32 07 00	LD(0007),A	
0181 0182	3E 05	LD A,05	
0183 0184 0185	32 05 00	LD(0005),A	
0186	AF	XOR A	
0187 0188 0189	32 06 00	LD(0006),A	
018A 018b 018C	Cd EC 80	CALL 80EC	
018d 018E 018F	Cd 34 80	CALL 8034	
0190 0191 0192	Cd AA 80	CALL 80AA	
0193 0194	FE 10	CP 10	La CPU controlla se è stato pigiato CONTROL-0 e in caso affermativo salta alla riga di programma 01A3, se no prosegue con la riga 0197
0195 0196	28 0C	JR Z,0C	La CPU controlla che sia stato pigiato un numero e non una delle lettere presenti sulla tastiera, quindi fa comparire tale numero nel display 0 spostando il contenuto precedente del display 0 sul display 1. Se avessimo battuto per errore una lettera si tornerebbe alla riga 0179
0197 0198 0199	Cd 03 80	CALL 8003	
019A 019b	d6 0A	SUB 0A	
019C 019d	30 db	JR NC,db	
019E 019F 01A0	Cd EC 80	CALL 80EC	La CPU viene rimandata ad eseguire la riga di programma 0190
01A1 01A2	18 Ed	JR Ed	
01A3 01A4 01A5	Cd 24 80	CALL 8024	La CPU carica nel registro A il numero da noi impostato; se questo è uguale a 0 torna a chiedercelo in quanto non è regolare; se no prosegue
01A6	b7	OR A	
01A7 01A8	28 d0	JR Z,d0	
01A9	F5	PUSH AF	La CPU trasforma il numero decimale relativo alla «posizione» in un numero esadecimale
01AA 01Ab	E6 F0	AND F0	
01AC	0F	RRCA	
01Ad	0F	RRCA	
01AE	0F	RRCA	

Riga	Dati	Mnemonico	Descrizione
01AF	0F	RRCA	A conversione avvenuta carica tale numero nel registro B
01b0	3C	INC A	
01b1	47	LD B,A	
01b2	AF	XOR A	
01b3 01b4	10 0F	DJNZ,0F	
01b5	47	LD B,A	
01b6	F1	POP AF	
01b7 01b8	E6 0F	AND 0F	
01b9	80	ADD A,B	
01bA	47	LD B,A	
01bb 01bC 01bd	3A 3b 01	LD A,(013b)	Se la posizione da noi scritta è superiore a 55, cioè alla capacità massima della tabella dei codici, la CPU torna a chiedercela; se no prosegue con la riga di programma 01C8
01bE	3d	DEC A	
01bF	90	SUB B	
01C0 01C1	38 b7	JR C,b7	
01C2 01C3	18 04	JR 04	Queste due istruzioni appartengono alla routine di conversione da decimale a esadecimale vista in precedenza
01C4 01C5	C6 0A	ADD A,0A	
01C6 01C7	18 Eb	JR Eb	
01C8 01C9 01CA	2A 2b 01	LD HL,(012b)	La CPU trascrive il nuovo codice nella posizione della tabella che noi gli abbiamo assegnato. Se tale posizione era già occupata il vecchio codice viene cancellato dal nuovo. Il contenuto dei registri BC viene salvato caricandolo nello Stack
01Cb	C5	PUSH BC	
01CC	05	DEC B	
01Cd 01CE	28 04	JR Z,04	
01CF	23	INC HL	
01d0	23	INC HL	
01d1 01d2	18 F9	JR F9	
01d3	73	LD(HL),E	
01d4	23	INC HL	
01d5	72	LD(HL),D	
01d6 01d7	3E 42	LD A,42	La CPU visualizza sui display la scritta G., cioè «Giacenza?» quindi attende che noi scriviamo sulla tastiera la giacenza stessa
01d8 01d9 01dA	Cd 21 02	CALL 0221	
01db	C1	POP BC	

Riga	Dati	Mnemonico	Descrizione
01dG 01dd 01dE 01dF	Ed 5b 4b 01	LD DE,(014b)	Pigiando CONTROL-0 la CPU carica in memoria nella posizione richiesta la giacenza che noi abbiamo impostato sulla tastiera
01E0 01E1 01E2	Cd Fb 01	CALL 01Fb	
01E3 01E4	3E 06	LD A,06	La CPU visualizza sui display la scritta E., cioè «Entrate» e aspetta che noi scriviamo il relativo numero; quando pigiamo CONTROL-0 trasferisce tale numero in memoria
01E5 01E6 01E7	Cd 57 02	CALL 0257	La CPU visualizza sui display la scritta U., cioè «Uscite» e aspetta che noi scriviamo tale numero; quando pigiamo CONTROL-0 lo trasferisce in memoria
01E8 01E9	3E 41	LD A,41	
01EA 01Eb 01EC	Cd 57 02	CALL 0257	La CPU visualizza sui display la scritta L., cioè «Lire» e aspetta che noi impostiamo il prezzo vendita; quando pigiamo CONTROL-0 lo trasferisce in memoria
01Ed 01EE	3E 47	LD A,47	
01EF 01F0 01F1	Cd 57 02	CALL 0257	La CPU visualizza sui display la scritta P., cioè «Prezzo acquisto» e aspetta che noi gli forniamo tale prezzo; quando pigiamo CONTROL-0 lo trasferisce in memoria
01F2 01F3	3E 0C	LD A,0C	
01F4 01F5 01F6	Cd 57 02	CALL 0257	La CPU ritorna ad eseguire la riga 0350, cioè torna a chiederci SPL
01F7 01F8 01F9	C3 50 03	JP 0350	
01FA	00		

Subroutine per trasferire in memoria i dati presenti sui display

01Fb	C5	PUSH BC	La CPU salva, trascrivendolo nello Stack, il contenuto dei principali registri
01FC	E5	PUSH HL	
01Fd	F5	PUSH AF	
01FE 01FF 0200	21 01 00	LD HL,0001	Vedi descrizione riga 0219 e seguenti
0201 0202	10 16	DJNZ,16	
0203 0204 0205	Cd 2A 80	CALL 802A	La CPU trasferisce in memoria il contenuto dei primi due display, cioè 0-1
0206	12	LD(DE),A	
0207	13	INC DE	
0208	23	INC HL	
0209	23	INC HL	

Riga	Dati	Mnemonico	Descrizione
020A 020b 020C	Cd 2A 80	CALL 802A	La CPU trasferisce in memoria il contenuto dei display 2-3
020d	12	LD(DE),A	
020E	13	INC DE	
020F	23	INC HL	
0210	23	INC HL	La CPU trasferisce in memoria il contenuto dei display 4-5
0211 0212 0213	Cd 2A 80	CALL 802A	
0214	12	LD(DE),A	
0215	F1	POP AF	La CPU va a riprendersi il contenuto precedente dei registri dallo Stack poi ritorna ad eseguire il programma principale
0216	E1	POP HL	
0217	C1	POP BC	
0218	C9	RET	Con queste istruzioni la CPU va a cercarsi in memoria la posizione esatta in cui deve trascrivere il numero presente sui display
0219 021a 021b	21 10 00	LD HL,0010	
021C	19	ADD HL,DE	
021d	54	LD D,H	
021E	5d	LD E,L	
021F 0220	18 dd	JR dd	

Subroutine per inserire i dati del componente tramite la tastiera

0221	E5	PUSH HL	La CPU spegne tutti i display poi fa comparire sul solo display 7 il simbolo corrispondente al contenuto del registro A, cioè G.-E.-U.-L.-P.
0222	F5	PUSH AF	
0223 0224 0225	Cd 67 80	CALL 8067	
0226	F1	POP AF	
0227 0228 0229	21 05 00	LD HL,0005	
022A 022b 022C	Cd EC 80	CALL 80EC	La CPU controlla se è stato pigiato CONTROL-0 e in caso affermativo salta alla riga 0248, se no prosegue
022d 022E 022F	Cd CF 00	CALL 00CF	
0230 0231	0E 06	LD C,06	
0232 0233	FE 10	CP 10	
0234 0235	28 11	JR Z,11	

Riga	Dati	Mnemonico	Descrizione
0236 0237	d6 0A	SUB 0A	Se per sbaglio avessimo pigiato una lettera invece di un numero la CPU non la considera valida
0238 0239	30 08	JR NC,08	
023A 023b	C6 0A	ADD 0A	
023C 023d 023E	Cd 15 80	CALL 8015	La CPU visualizza l'ultimo numero impostato sul display 0 spostando tutto ciò che era presente sui display di una posizione verso sinistra
023F 0240 0241	Cd EC 80	CALL 80EC	
0242 0243 0244	Cd AA 80	CALL 80AA	La CPU aspetta che si pigi un tasto poi ritorna ad eseguire la riga di programma 0232
0245 0246	18 Eb	JR Eb	
0247	7E	LD A,(HL)	La CPU pone automaticamente degli 0 davanti al numero impostato se questo è più corto di 6 cifre
0248 0249	FE 11	CP 11	
024A 024B	20 09	JR NZ,09	
024C	AF	XOR A	
024d	77	LD(HL),A	
024E	b5	OR L	
024F 0250	28 04	JR Z,04	
0251	2d	DEC L	
0252 0253 0254	C3 47 02	JP 0247	
0255	E1	POP HL	
0256	C9	RET	Gruppo di istruzioni utilizzato dal programma n. 0 per richiederci la giacenza, il totale entrate, il totale uscite ecc. e per trasferire in memoria il numero da noi impostato sulla tastiera.
0257	13	INC DE	
0258 0259 025A	Cd 21 02	CALL 0221	
025b 025C	06 01	LD B,01	
025d 025E 025F	Cd Fb 01	CALL 01Fb	
0260	C9	RET	

Nota: da 0261 a 0270 riempire con 00

Programma n. 2 per fare i carichi di magazzino

0271	78	LD A,B	Se il codice da noi impostato non è presente in memoria nella relativa tabella, la CPU segnala l'errore chiedendo di nuovo «CODICE»
0272	b7	OR A	
0273 0274	20 03	JR NZ,03	

Riga	Dati	Mnemonico	Descrizione
0275 0276 0277	C3 00 01	JP,0100	Salta alla riga 0100
0278 0279 027A 027b	Ed 5b 4b 01	LD DE,(014b)	La CPU somma la quantità entrata alla giacenza precedente e trasferisce il tutto in memoria
027C 027d	3E 06	LD A,06	
027E 027F 0280	Cd 21 02	CALL,0221	
0281 0282 0283	Cd 90 02	CALL 0290	
0284	13	INC DE	La CPU aggiunge la quantità entrata alle entrate totali precedenti e trasferisce il tutto in memoria
0285 0286	06 01	LD B,01	
0287 0288 0289	Cd 90 02	CALL 0290	
028A 028b 028C	C3 50 03	JP,0350	La CPU ritorna a far comparire sui display la scritta SPL

Nota: da 028d a 028F riempire con 00

Subroutine di «addizione»

0290	C5	PUSH BC	La CPU salva il contenuto dei registri principali trascrivendolo nello Stack
0291	E5	PUSH HL	
0292	F5	PUSH AF	
0293 0294 0295	21 01 00	LD HL,0001	La CPU cerca in tabella la posizione in cui sono contenuti i dati relativi al codice impostato
0296 0297	10 26	DJNZ,26	
0298 0299 029A	Cd 2A 80	CALL 802A	La CPU aggiunge le prime due cifre in memoria alle prime due cifre presenti sul display
029b	47	LD B,A	
029C	1A	LD A,(DE)	
029d	80	ADD A,B	
029E	27	DAA	
029F	12	LD(DE),A	
02A0	08	EX AF,AF'	
02A1	13	INC DE	
02A2	23	INC HL	
02A3	23	INC HL	
02A4 02A5 02A6	Cd 2A 80	CALL 802A	

Riga	Dati	Mnemonico	Descrizione	
02A7	47	LD B,A	La CPU addiziona le due cifre centrali del numero contenuto in memoria alle corrispondenti cifre del numero impostato sui display	
02A8	08	EX AF,AF'		
02A9	1A	LD A,(DE)		
02AA	88	ADC A,B		
02Ab	27	DAA		
02AC	12	LD(DE),A		
02Ad	08	EX AF,AF'		
02AE	13	INC DE		
02AF	23	INC HL		
02b0	23	INC HL		
02b1 02b2 02b3	Cd 2A 80	CALL 802A	La CPU addiziona le ultime due cifre del numero contenuto in memoria alle corrispondenti cifre del numero presente sui display	
02b4	47	LD B,A		
02b5	08	EX AF,AF'		
02b6	1A	LD A,(DE)		
02b7	88	ADC A,B		
02b8	27	DAA		
02b9	12	LD (DE),A		
02bA	F1	POP AF		
02bb	E1	POP HL		
02bC	C1	POP BC		
02bd	C9	RET	La CPU va a riprendersi dallo Stack il contenuto precedente dei registri poi torna ad eseguire il programma principale.	
02bE 02bF 02C0	21 10 00	LD HL,0010	Queste istruzioni insieme a quelle di riga 0293-0296 servono alla CPU per ricercarsi nella tabella dei dati in memoria l'esatta posizione in cui deve andare a leggersi ciò che gli necessita	
02C1	19	ADD HL,DE		
02C2	54	LD D,H		
02C3	5d	LD E,L		
02C4 02C5	18 Cd	JR Cd		
Programma n. 3 per fare gli scarichi di magazzino				
02C6	78	LD A,B'		Se il codice da noi impostato non è presente in memoria nella relativa tabella, la CPU segnala l'errore chiedendo di nuovo: «CODICE»
02C7	b7	OR A		
02C8 02C9	20 03	JR NZ,03		
02CA 02Cb 02CC	C3 00 01	JP 0100		
02Cd 02CE 02CF 02d0	Ed 5b 4b 01	LD DE,(014b)	Carica nella coppia di registri DE il contenuto della riga 014b	

Riga	Dati	Mnemonico	Descrizione
02d1 02d2	3E 41	LD A,41	La CPU sottrae la quantità uscita dalla giacenza precedente quindi trasferisce il risultato in memoria
02d3 02d4 02d5	Cd 21 02	CALL 0221	
02d6 02d7 02d8	Cd 00 03	CALL,0300	
02d9	13	INC DE	
02dA	13	INC DE	
02db	13	INC DE	La CPU addiziona la quantità uscita alle uscite totali quindi trasferisce il tutto in memoria
02dC	13	INC DE	
02dd 02dE	06 01	LD B,01	
02dF 02E0 02E1	Cd 90 02	CALL,0290	
02E2 02E3 02E4	C3 50 03	JP 0350	
Nota: da 02E5 a 02FF riempire con 00			

Subroutine di «sottrazione»

0300	C5	PUSH BC	La CPU salva il contenuto dei registri che dovrà usare trascrivendolo nello Stack
0301	E5	PUSH HL	
0302	F5	PUSH AF	
0303 0304 0305	21 01 00	LD HL,0001	Vedi riga 032E e seguenti
0306 0307	10 26	DJNZ,26	
0308 0309 030A	Cd 2A 80	CALL 802A	La CPU sottrae le prime due cifre presenti sul display dalle prime due cifre del numero conservato in memoria
030b	47	LD B,A	
030C	1A	LD A,(DE)	
030d	90	SUB B	
030E	27	DAA	
030F	12	LD(DE),A	
0310	08	EX AF,AF'	
0311	13	INC DE	
0312	23	INC HL	
0313	23	INC HL	
0314 0315 0316	Cd 2A 80	CALL 802A	Vedi descrizione riga 0318 e seguenti
0317	47	LD B,A	

Riga	Dati	Mnemonico	Descrizione	
0318	08	EX AF,AF'	La CPU sottrae le due cifre centrali del numero presente sui display dalle due cifre centrali del numero conservato in memoria	
0319	1A	LD A,(DE)		
031A	98	SBC A,B		
031b	27	DAA		
031C	12	LD(DE),A		
031d	08	EX AF,AF'		
031E	13	INC DE		
031F	23	INC HL		
0320	23	INC HL		
0321 0322 0323	Cd 2A 80	CALL 802A		La CPU sottrae le ultime 2 cifre sul display dalle ultime due cifre del numero conservato in memoria
0324	47	LD B,A		
0325	08	EX AF,AF'		
0326	1A	LD A,(DE)		
0327	98	SBC A,B		
0328	27	DAA		
0329	12	LD(DE),A		
032A	F1	POP AF	La CPU va a riprendersi dallo Stack il contenuto precedente dei registri poi torna ad eseguire il programma principale	
032b	E1	POP HL		
032C	C1	POP BC		
032d	C9	RET		
032E 032F 0330	21 10 00	LD HL,0010	Queste istruzioni servono alla CPU per ricercarsi nella tabella dei dati in memoria l'esatta posizione in cui deve andare a leggerci ciò che gli necessita	
0331	19	ADD HL,DE		
0332	54	LD D,H		
0333	5d	LD E,L		
0334 0335	18 Cd	JR,Cd		

Nota: da 0336 a 034F riempire con 00

Routine per la scelta del programma

0350 0351 0352	Cd 67 80	CALL 8067	Con queste istruzioni la CPU fa comparire la scritta SPL cioè «Scelta Programma Lavoro» sul display 7-6-5.
0353 0354	3E 05	LD A,05	
0355 0356 0357	32 07 00	LD(0007),A	
0358 0359	3E 17	LD A,17	

Riga	Dati	Mnemonico	Descrizione	
035A 035b 035C	32 06 00	LD(0006),A	Il numero esadecimale 05 che carichiamo nella locazione 0007 equivale a una S, il numero 17 che carichiamo nella locazione 0006 equivale a una P e il numero 16 che carichiamo nella locazione 0005 equivale a una L	
035d 035E	3E 16	LD A,16		
035F 0360 0361	32 05 00	LD(0005),A		
0362 0363 0364	Cd EC 80	CALL 80EC		
0365 0366 0367	Cd AA 80	CALL 80AA		La CPU si mette in attesa che si pigli uno dei tasti 0-1-2-3-4 per scegliere il programma; se per errore si piglia un tasto diverso la CPU ritorna a fare la domanda, cioè compare di nuovo la scritta SPL sui display
0368 0369	d6 05	SUB 05		
036A 036b	30 F9	JR NC,F9		
036C 036d 036E	21 88 03	LD HL,0388		La CPU aggiunge il doppio del numero da noi impostato al contenuto dei registri HL e sfrutta quindi il risultato di tale addizione come indirizzo per andarsi a leggere nella tabella dei dati riportata in coda al programma (locazioni 037E e seguenti) quel numero che gli necessita per ricordarsi in seguito quale programma noi vogliamo che esegua. La CPU annota tale numero nelle locazioni 0144-0145 della memoria RAM
036F	87	ADD A,A		
0370	85	ADD A,L		
0371	6F	LD L,A		
0372	7E	LD A,(HL)		
0373 0374 0375	32 44 01	LD(0144),A		
0376	23	INC HL		
0377	7E	LD A, (HL)		
0378 0379 037A	32 45 01	LD(0145),A		
037b 037C 037d	C3 00 01	JP_0100	La CPU passa ad eseguire la routine che inizia alla riga 0100	
037E 037F	75 01	Programma n. 0	TABELLA dei DATI necessari per «dirottare» la CPU ad eseguire il programma da noi prescelto	
0380 0381	46 01	Programma n. 1		
0382 0383	71 02	Programma n. 2		
0384 0385	C6 02	Programma n. 3		
0386 0387	d7 00	Programma n. 4		