

# PROGETTO

# ELEKTOR

7.8

luglio-Agosto 1988

e le sue pagine

Gruppo Editoriale  
**JCE**

IMMAGINE NELL'IMMAGINE

OROLOGIO ATOMICO DCF77

PREAMPLIFICATORE ESOTERICO

NUMERO DOPPIO  
BASTA  
CAR AMPLIFIER  
25W  
IN  
REGALO



INSERTO  
A.R.I.

L. 7.000

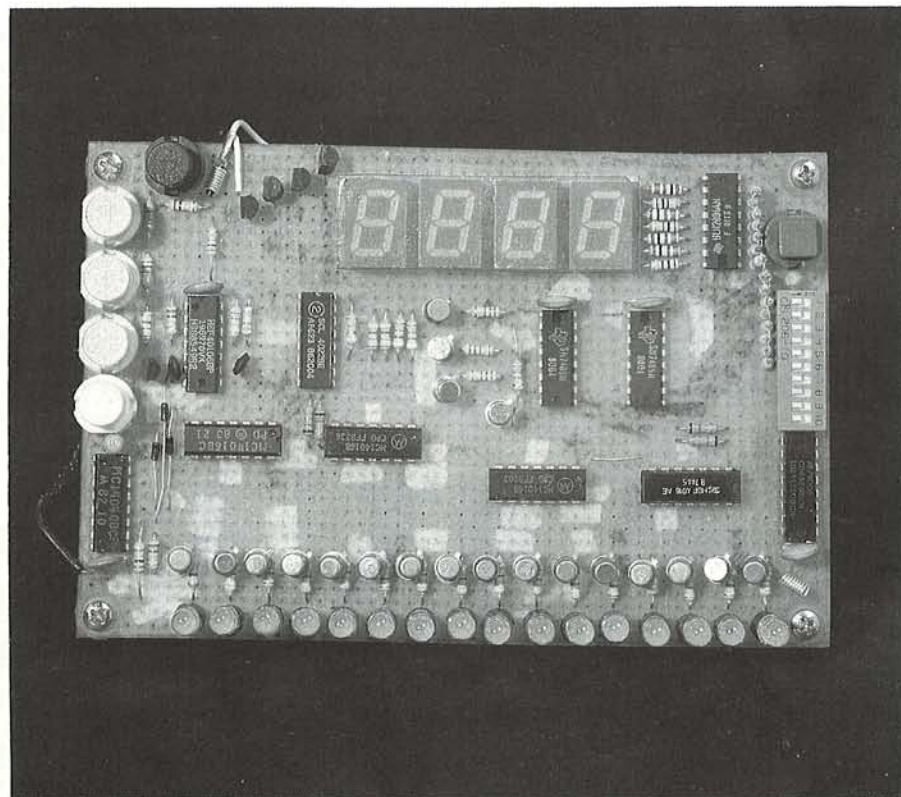
# CONVERTITORE "B.E.D."

Un estroverso lettore marchigiano si porta a casa l'Amstrad PC 1640 SD in palio questo mese: questo convertitore decimale-binario-esadecimale è stato realizzato con oltre 100 componenti!

di Valter Narcisi - S. Benedetto del Tronto

Il circuito che ho realizzato permette di convertire i tre codici numerici più diffusi in elettronica e informatica fino a un valore massimo di 65535 (FFFF) ma nella realizzazione pratica ho trovato molte difficoltà per ciò che riguarda la parte convertitrice in codice esadecimale. Inoltre non sono riuscito a

porte esterne di tipo nand/nor/not; il circuito avrebbe funzionato alla perfezione tenendo conto che a ingressi maggiori del codice 1001 si sarebbe spento automaticamente; tuttavia questa soluzione mi è sembrata una soluzione poco ottimale. Scartando le Eprom o le memorie di tipo volatile ho



reperire in commercio le decodifiche in esadecimale a sette segmenti e proprio per questo motivo ho dovuto realizzare sei prototipi prima di trovare la giusta combinazione di integrati per realizzare la parte esadecimale.

Ho ritenuto utile costruire una pseudo-decodifica tramite un 4511 ad alcune

pensato di impiegare alcune RAM statiche la cui programmazione risulta molto semplice. Per la costruzione di questo convertitore ho cercato di rendere il circuito alla portata di tutti, evidenziando anche alcune funzioni che nella norma vengono svolte elettronicamente.

Dato l'impiego appena descritto delle RAM statiche appare chiaro che nel momento in cui viene a mancare l'alimentazione, ogni contenuto si cancella definitivamente; quindi, è opportuno collegare l'alimentazione del circuito a una piccola pila comandata da una serie di interruttori; in questo modo possiamo escludere il funzionamento dei display ma garantire la programmazione delle RAM. Un piccolo consiglio: sarebbe utile utilizzare delle RAM CMOS del tipo 74C89 al solo scopo di non scaricare in breve tempo la batteria. Chi volesse optare per la soluzione priva di batterie può farlo ricordandosi che all'accensione dovrà effettuare la programmazione.

## Schema elettrico

Si prenda in considerazione lo schema relativo alla sezione di controllo, binario e multiplexer, visibile in Figura 1. Il codice binario è generato da U1 e U2: il primo è un contatore BCD doppio a 4 digit mentre il secondo è sempre un contatore ma a 12 bit. Ogni impulso (con fronte negativo) che noi applichiamo al pin 2 di U1, farà avanzare il conteggio visualizzato dai led LD1-LD16 ed i relativi transistor usati come buffer.

Oltre che pilotare i led, questi due integrati pilotano gli ingressi di U4, U5, U6 e U7, tutti integrati CD 4016 (quadrupli interruttori bidirezionali) usati in questo progetto in funzione di DATASELECTOR.

Le loro uscite, infatti, vengono multiplexate dall'integrato U3, un contatore Johnson a 8 uscite, usato appunto come generatore di multiplexer per i 4016 e i transistor Q22-Q25: la frequenza di scansione è generata dal gate U7/E (inverter 40106) e il suo valore si aggira sui 240 Hz. L'integrato U3, mediante le sue uscite, abilita in sequenza il gruppo di interruttori dei 4016 e il transistor relativo all'integrato selezionato.

Mediante i transistor Q18-Q21, i dati selezionati di volta in volta verranno applicati sui piedini 1, 15, 14, 13 delle memorie RAM (SN 7489), corrispondenti ai pin di indirizzo delle memorie (A0-A3). Alle uscite troveremo i dati precedentemente programmati che verranno applicati all'ULN 2003 al cui interno troviamo 7 buffer invertenti che pilotano i segmenti dei display ad anodo comune di tipo TIL 321.

Avrete certamente intuito che le memorie leggono i dati provenienti dal DATASELECTOR a gruppi di 4 bit partendo da quelli meno significativi: i display visualizzano i risultati provenienti

IC7 BIS = 40105

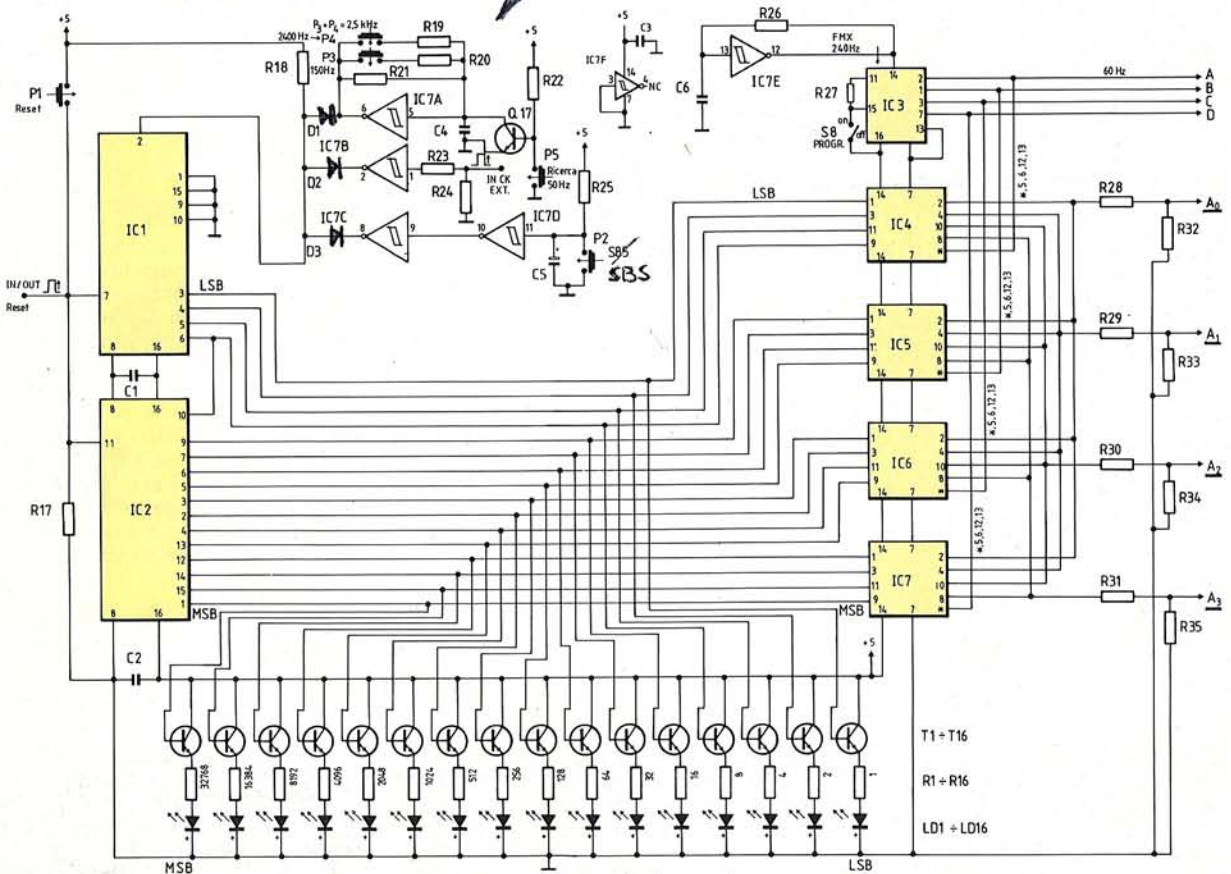


Figura 1. Schema elettrico sezione controllo. *bus e multiplex*

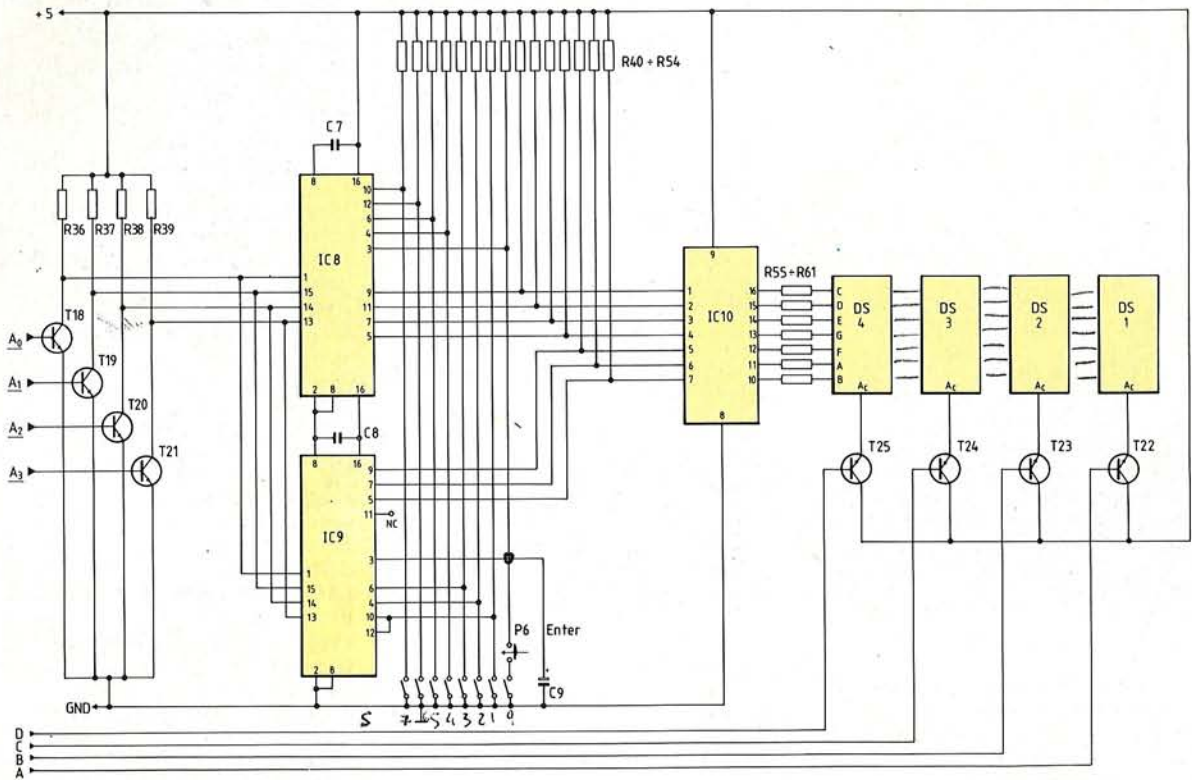


Figura 2. Schema elettrico sezione display. *programmazione*

dalle memorie alternandosi a loro volta in modo che il valore dei primi 4 bit venga visualizzato sul display DS1, il valore del secondo gruppo sul secondo display e via dicendo: questo ciclo si ripete per circa 60 volte al secondo in modo tale da non permettere all'occhio umano di seguire le variazioni. Mne-monicamente, dunque, si potrebbe as-sociare l'integrato U4 al display DS1, l'integrato U5 al DS2, l'U6 al DS3 e l'integrato U7 al DS4.

Il codice decimale si ricava prendendo in considerazione i led accesi e som-mandone il valore (riportato sullo schema o Tabella 1).

La boccola IN/OUT Reset unitamente a quella IN CK/EXT sono utilizzate per l'allaccio con dispositivi esterni.

Per selezionare il numero da convertire, si fa uso dei pulsanti da P2 a P5: in particolare, P2 incrementa il conteggio di una unità ogni volta che si preme

sogna incrementare il conteggio pre-mendo il pulsante P2 (SBS): così facen-do si possono programmare i 16 simbo-li alfanumerici che verranno via via vi-sualizzati dal display in base all'indiriz-zo che si presenterà sugli appositi pin delle memorie.

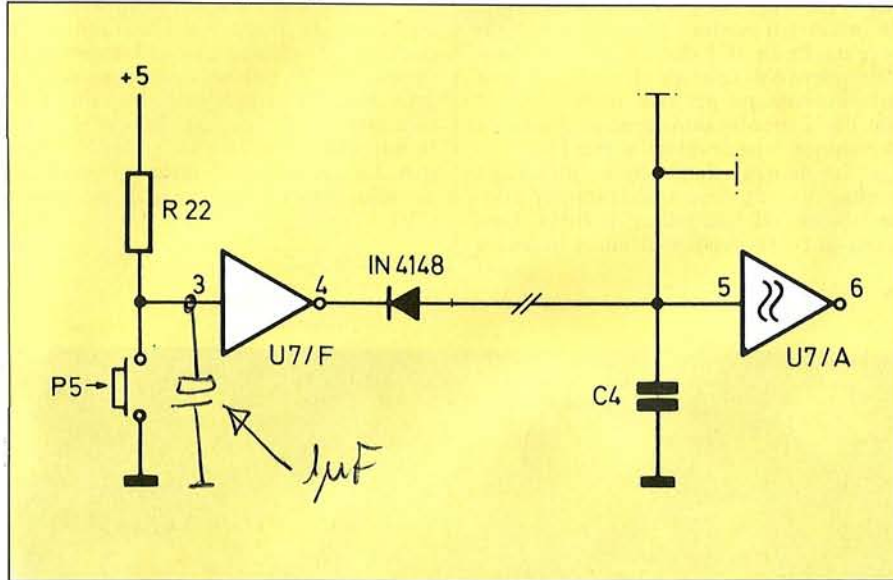
Per abilitare la programmazione occor-re chiudere gli interruttori S9 e S8: chiudendo quest'ultimo, si blocca il funzionamento del multiplexer che, ri-manendo fisso sulla prima uscita del 4022, abiliterà solo ed esclusivamente l'integrato U4 dal quale preleveremo i 16 indirizzi per la procedura di pro-grammazione: durante questa fase sarà in funzione solo il display DS1 (il primo da sinistra).

(NB: Resettare prima della program-mazione col pulsante P1-Reset). Di se-guito è riportata la serie di 16 sequenze ognuna delle quali contiene la numera-zione degli interruttori da chiudere per

Tabella 1. Valore per calcolo codice decimale

LD	VALORE
1	1
2	2
3	4
4	8
5	16
6	32
7	64
8	128
9	256
10	512
11	1024
12	2048
13	4096
14	8192
15	16384
16	32768

Tabella 2. Modifica Q17-U7/F



mentre P5 abilita la ricerca veloce della cifra: unitamente a P5, si possono pre-mere i pulsanti P3 e/o P4 per aumenta-re ulteriormente la frequenza di ricerca. Il gate U7/F è stato volutamente lascia-to libero per permettere una eventuale Uscita o Ingresso per chi volesse appor-tare modifiche al circuito: se non venis-se usato, consiglio di fare la modifica riportata in Tabella 2 che permette di eliminare un transistor (Q17).

L'assorbimento di tutto il circuito oscil-la fra 250 e 350 mA.

Come già accennato, la program-mazione delle due RAM risulta alquanto facile.

Bisognerà solo tenere presente che, do-po aver impostato gli interruttori di programmazione, si deve battere il pul-sante P6, simbolicamente (ma non troppo!) denominato "ENTER".

Dopo aver programmato e messo in memoria un simbolo alfanumerico, bi-

programmare il simbolo relativo:

Interruttori in ON	Simbolo relativo programmato
1-2-3-5-6-7	0
2-7	1
2-3-4-5-6	2
2-3-4-6-7	3
1-2-4-7	4
1-3-4-6-7	5
1-3-4-5-6-7	6
2-3-7	7
1-2-3-4-5-6-7	8
1-2-3-4-6-7	9
1-2-3-4-5-7	A
1-4-5-6-7	b
1-3-5-6	C
2-4-5-6-7	d
1-3-4-5-6	E
1-3-4-5	F

Finita la procedura di programmazione aprire gli interruttori S8-S9.

### Elenco componenti

#### Semiconduttori

- Q1 ÷ Q16: 1W8907 o BC237
- Q17: BC237
- Q18 ÷ Q21: 1W8907 o 2N2222 o 2N708
- Q22 ÷ Q25: BC337
- U1: CD 4520-
- U2: CD 4040-
- U3: CD 4022-
- U4 ÷ IC7: CD 4016-
- U8, UC9: SN 7489 (Ram 16 × 4 St.)
- U10: ULN 2003 o 2004
- D1 ÷ D3: 1N4148
- LD1 ÷ LD16: led rossi
- DS1 ÷ DS4: display FND 507 o TIL 321
- P1 ÷ P6: pulsanti N.A.
- S1 ÷ S9: serie interruttori unipol.

#### Resistori

- R1 ÷ R16: 330 Ω
- R17, R18, R22, R24: 10 kΩ
- R19: 47 kΩ
- R20: 1 MΩ
- R21: 2,2 MΩ
- R23: 1 kΩ
- R25: 22 kΩ
- R26: 220 kΩ
- R27, R32 ÷ R35, R40 ÷ R47: 4,7 kΩ
- R28 ÷ R31: 6,8 kΩ
- R36 ÷ R39, R48 ÷ R54: 2,2 kΩ
- R55 ÷ R61: 100 Ω

#### Condensatori

- C1 ÷ C3, C7, C8: .047 μF
- C4: .01 μF
- C5, C9: 1 μF, tantalio
- C6: .022 μF