

DIALOGO TRA COMPUTER

Se disponete di un C/64 ed un M-24, o un qualsiasi altro computer che disponga di interfaccia RS-232C, può capitare la necessità di collegarli tra loro

di F. Sorgato e G. Mariani

Il Commodore 64, nonostante la sua non giovanissima età, continua a stupire per la flessibilità che lo caratterizza; oltre, infatti, a mostrarci favolosi giochi e stupendi programmi di utilità, spesso si presta a scopi ben più seri.

Le due porte di cui dispone permettono, l'una, l'espansione del sistema; l'altra (la porta utente) la trasmissione dei dati.

Purtroppo quest'ultima non è standard da un punto di vista hardware, e non risulta utilizzabile senza alcuni piccoli accorgimenti.

La user port del C/64 funziona quasi come una normale RS-232C, che è l'interfaccia diffusa su tutte le macchine in commercio; non è comunque possibile una connessione diretta con questa, dal momento che la porta del C/64 lavora con livelli logici TTL (0 e 5 Volt) mentre la RS-232C standard richiede tensioni di 12 Volt positive e negative.

Per connettere il C/64 con un altro computer dotato di RS-232C, sarà sufficiente costruire un semplicissimo adattatore per convertire la tensione in ingresso e uscita della User Port ai li-

SCHEDA TECNICA

Hardware e software per comunicazioni tra computer

Hardware: C/64, C/128, Vic-20, e computer che dispongano di interfaccia Rs-232C

Anche il programma per C/64 pubblicato in queste pagine è contenuto nel disco "Directory" di questo mese.

velli richiesti.

Diamo ora uno sguardo al funzionamento della RS-232C. Osservando un computer che monta questa interfaccia, si nota che questa comunica con l'esterno tramite un connettore Canon a 25 poli.

Naturalmente non tutti i contatti sono utilizzati; anzi, gli unici essenziali per il funzionamento dell'apparato sono solo due: Il terminale di trasmissione (TxD) e quello di ricezione (RxD), oltre, naturalmente, alla linea di massa (GND).

Le altre sono linee di handshaking, cioè permettono la sincronizzazione con le altre periferiche collegate. Generalmente queste linee consentono il collegamento con periferiche più lente del computer, in modo che questi dispositivi possano "informare" l'elaboratore di sospendere momentaneamente la trasmissione dei dati fino a quando non saranno in grado di riceverne altri.

Siccome in questo articolo ci occupiamo della interconnessione tra due computer, le uniche linee che interesseranno saranno quelle di trasmissione e di ricezione.

Abbiamo realizzato l'interfaccia nel modo più semplice possibile, così che anche i lettori meno esperti possano realizzarla senza eccessive difficoltà. Il dispositivo è costituito essenzialmente da due circuiti integrati (con specifiche funzioni) e pochi altri componenti discreti.

I due integrati sono poco più di semplici invertitori, e le loro sigle sono: MC 1489 e MC 1488.

Il primo è necessario per il collegamento da RS-232C a C/64 e si alimenta con la sola tensione di +5V, disponibile sulla User Port.

Il secondo serve per il collegamento opposto, e quindi necessita delle due tensioni +/- 12 Volt indispensabili per pilotare il dispositivo RS-232C.

Queste tensioni vengono ricavate dall'uscita a 9 Volt alternati del C/64, attraverso l'uso di quattro condensatori ed altrettanti diodi, che provvedono ad elevare la tensione ai livelli necessari, secondo lo schema classico riportato in qualsiasi manuale di elettronica.

Non ci dilungheremo certo in parti-

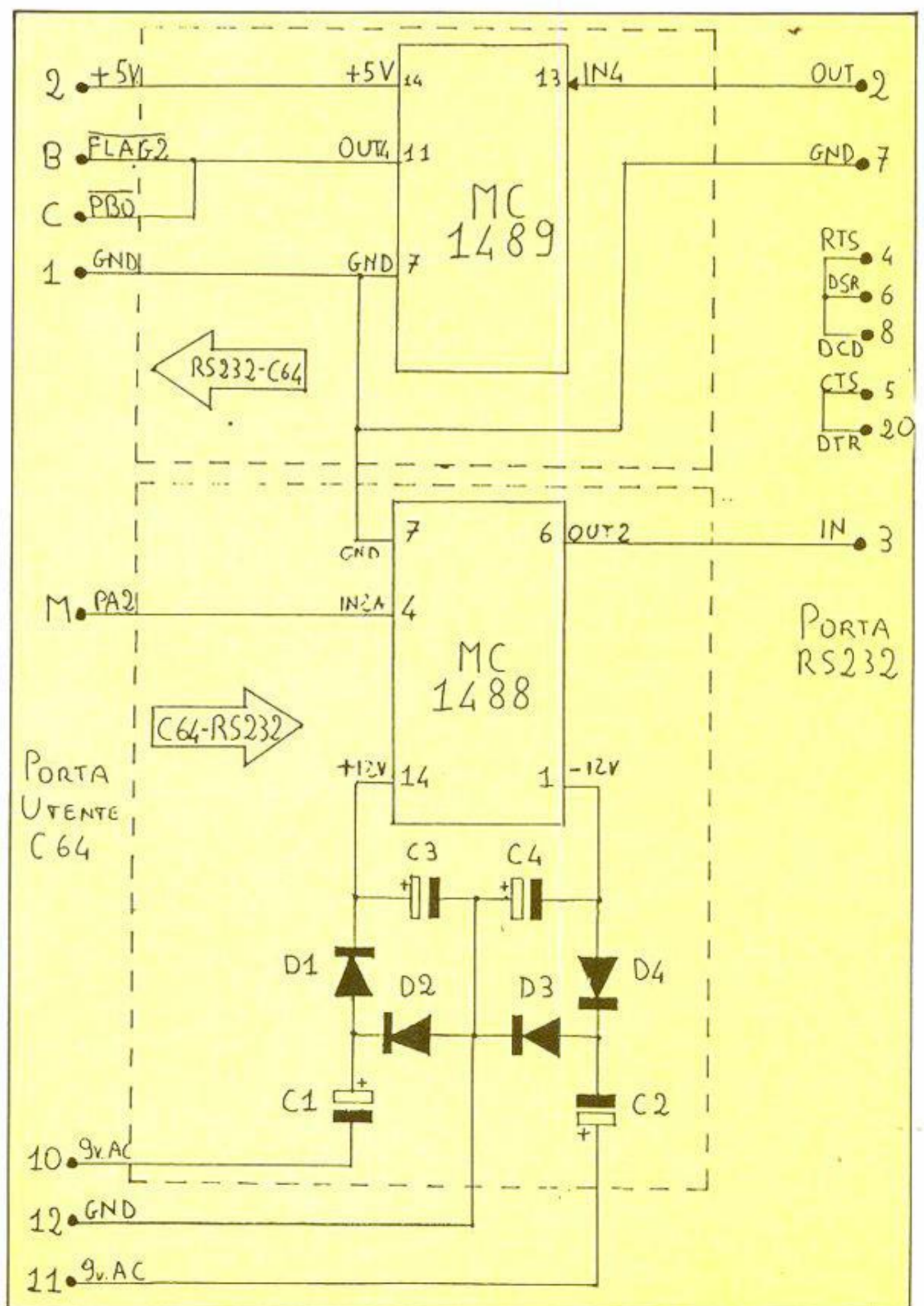
colari tecnici che potrebbero confondere il lettore, ma passeremo subito alla realizzazione pratica del circuito.

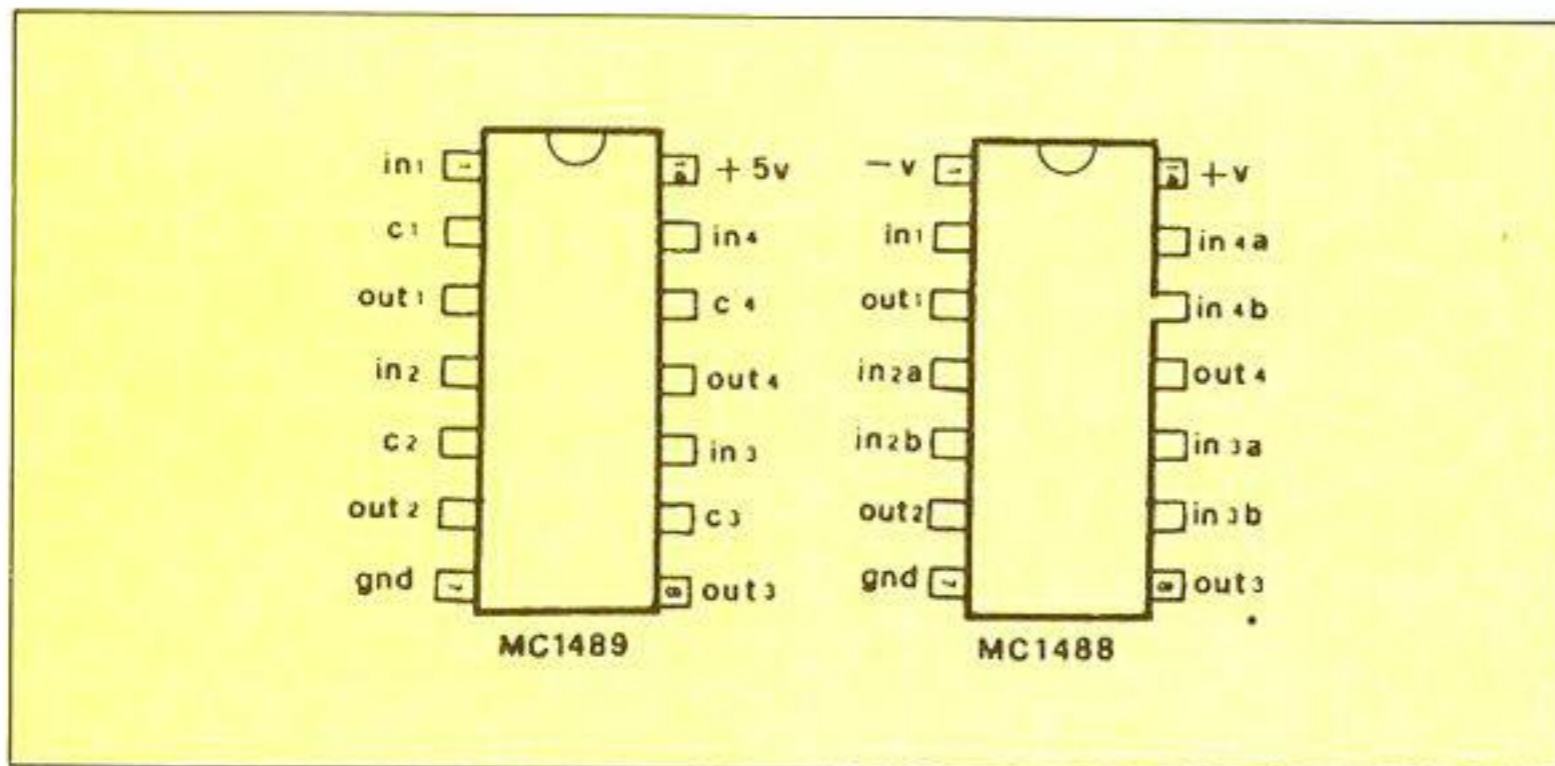
Il circuito

Grazie alla semplicità dello schema elettrico, ed al ridotto numero di collegamenti necessari, è stata realizzata l'interfaccia su di una basetta del tipo millefori, con passo per integrati, corredata di pettine con passo 3,96 mm.

Sul pettine è saldato il connettore 12 + 12 poli, da inserire nella porta utente del C/64. Osservando lo schema del circuito, si notano due parti ben distinte, racchiuse in rettangoli tratteggiati, che rappresentano, rispettivamente, la zona di ricezione e quella di trasmissione.

Poichè sono completamente indipendenti l'una dall'altra, chi non avesse la necessità di scambio bidirezionale dei dati potrà limitarsi a costruire solo la parte che interessa, tralasciando l'altra.





Sullo schema si notano anche i numeri dei piedini degli integrati (secondo lo schema classico di lettura, in base alla tacchetta di riferimento) e la nomenclatura dei contatti delle varie porte, in modo da non commettere errori nella realizzazione del collegamento.

Bisognerà ricordarsi di effettuare il ponticello sul connettore 12 + 12 poli tra il piedino B (Flag 2) ed il C (PBO), ed anche i collegamenti sul connettore Cannon a 25 poli RS-232C tra i piedini 4 (RTS), 6 (DSR), 8 (DCD) ed tra i contatti 5 (CTS) e 20 (DTR), come indicato, appunto, sullo schema.

Questi ponticelli simulano l'handshaking; omettendoli non si riuscirà nè a trasmettere nè a ricevere.

Il prototipo realizzato ha funzionato egregiamente mettendo in contatto tra loro un C/64 ed un Amstrad PC-512 compatibile IBM, che possiede un'interfaccia RS-232C standard; tuttavia la disposizione dei contatti sul connettore potrebbe cambiare in altri computer; ne consegue che, prima di costruire l'interfaccia, sarebbe opportuno consultare la documentazione fornita con il computer e di annotare eventuali variazioni.

Nei disegni si notano l'elenco dei piedini della user port, della RS-232C e degli integrati MC-1488 e MC-1489, oltre all'elenco dei componenti necessari alla realizzazione dell'interfaccia.

A questo elenco vanno aggiunti sia il connettore 12 + 12 (accertatevi che i contatti siano separati!), sia quello Cannon a 25 poli maschio o femmina

a seconda del connettore presente sul computer.

Il Basic e la RS-232C

Il basic dei computer Commodore è dotato di istruzioni per la gestione di un'interfaccia RS-232C tramite le quali impostare vari parametri di trasmissione:

- Parità
- Baud
- Bit di stop
- Bit di dato

Impostando diversi valori è possibile variare la velocità di trasmissione (Baud), il numero di bit da cui sarà composto un byte (bit di dato), il controllo d'errore (parità) e il sincronismo (bit di stop).

Per aprire un canale di comunicazione tramite RS-232, è sufficiente usare la seguente sintassi:

`Open fn,2,0, Chr$(a) + Chr$(b)`

in cui FN è il numero del file che dovrà essere usato in tutti i successivi comandi riferiti alla RS-232, "2" è il numero del dispositivo che identifica automaticamente l'interfaccia, mentre A e B controllano i parametri, specificati nella tabella.

Per combinare assieme più di un valore, bisogna eseguire tra gli stessi un'operazione di somma oppure di OR logico.

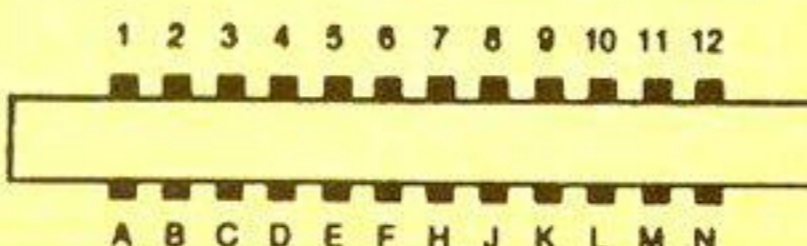
Se, per esempio, volessimo comunicare ad una velocità di 600 baud, con 7 bit di dati ed 1 bit di stop, "A" sarebbe uguale a: 7+32+0 (7=600 baud, 32=7 bit di dati, 0=1 bit di stop).

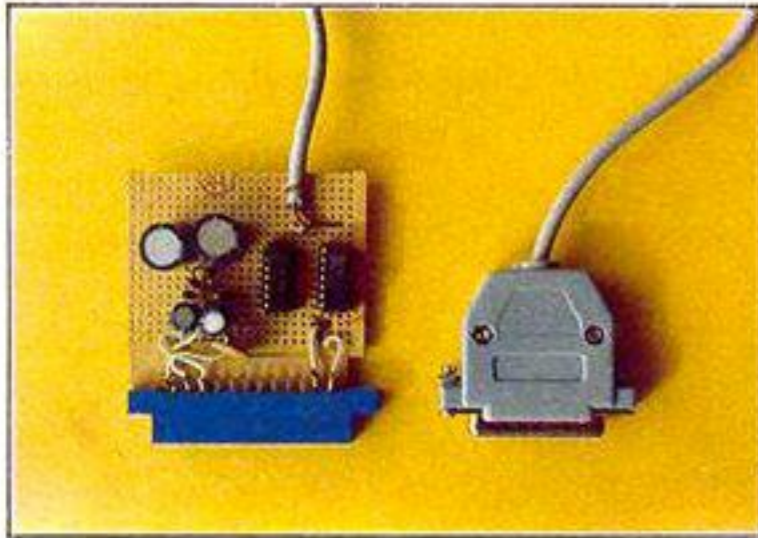
Dopo aver aperto il canale si possono scambiare i dati con la RS-232 tramite le normali istruzioni usate nel trattamento di file, quali Print#, Input#, Get# e Cmd. È importante ricordare che all'apertura e alla chiusura di un file RS-232 verrà eseguito automaticamente un CLR, quindi TUTTE le variabili ed i vettori verranno irrimediabilmente cancellati.

Ma tale inconveniente è facilissimo porre rimedio: sarà sufficiente fare in modo che il comando Open appaia per primo e Close per ultimo.

I/O Utente

Pin	Tipo	Nota	Pin	Tipo
1	GND		A	GND
2	+5V	MAX. 100 mA	B	FLAG2
3	RESET		C	PB0
4	CNT1		D	PB1
5	SP1		E	PB2
6	CNT2		F	PB3
7	SP2		H	PB4
8	PC2		J	PB5
9	SER. ATN IN		K	PB6
10	9 VAC	MAX. 100 mA	L	PB7
11	9 VAC	MAX. 100 mA	M	PA2
12	GND		N	GND





Il Basic sul PC

Anche il Gw-Basic del PC è munito di un set di istruzioni che permettono la gestione della RS-232.

Innanzitutto bisogna aprire un canale di comunicazione tramite il comando Open, in questo modo:

Ricezione:

Open "com1: baud, parity, bit, stop, asc" For Input As #fn

Formato	Val. "A"
8 bit di dati	0
7 bit di dati	32
50 baud	1
75 baud	2
150 baud	5
300 baud	6
600 baud	7
1200 baud	8
2400 baud	10
1 bit di stop	0
2 bit di stop	128

Formato	Val. "B"
Assenza di par.	0
Parita' pari	32
Parita' dispari	96

Trasmissione:
Open "com1: baud, parity, bit, stop, asc" For Output As #fn

Oppure, in caso di comunicazione bidirezionale:

Open "r", #fn, "com1: baud, parity, bit, stop, asc", 1

Spieghiamo i vari parametri:

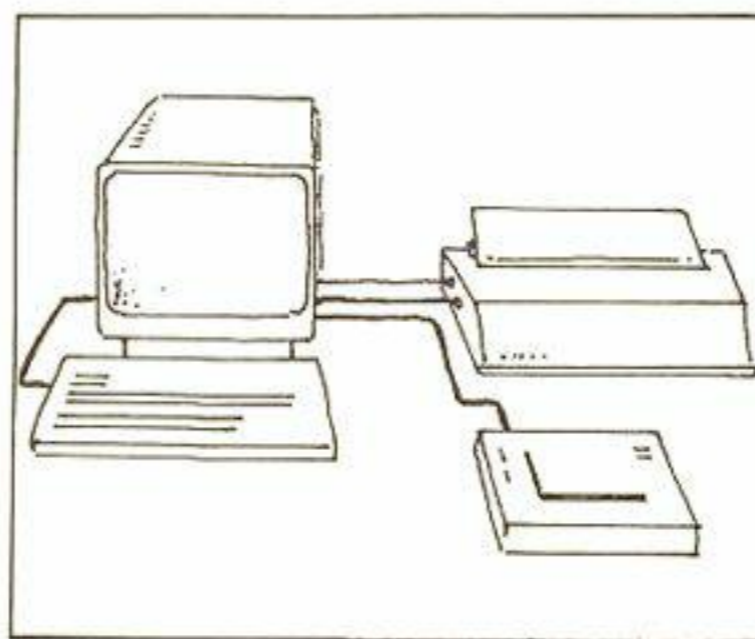
Baud:
è la velocità di trasmissione misurata in bit per secondo, che può valere: 75, 110, 150, 300, 600, 1200, 1800, 2400, 4800, 9600

Parity:
specifica la parità e può essere: "n" (assenza di parità), "o" (parità dispari), "e" (parità pari)

Bit:
è il numero di bit che compongono il dato, e può essere 7 oppure 8.

Stop:
specifica il numero di bit di stop, e può essere 1 oppure 2.

Asc:
è un parametro che specifica che i da-



ti trasmessi sono nel formato Ascii. Omettendolo si avrà la trasmissione di dati in formato binario.

Fn:
specifica il numero del file.

Esempio: per trasmettere ad una velocità di 600 baud, con 7 bit di dato, 1 di stop ed assenza di parità, bisognerà dare il comando:

Open "r", #fn, "com1: 600, n, 7, 1, asc"

Elenco componenti

C1: cond. elettrol. 100 micro f. 25V
C2: come C1
C3: cond. elettrol. 220 micro f. 25V
C4: come C3
C5: con. ceramico 0.1 micro far.
D : diodi 1N4002

Le istruzioni per ricevere e trasmettere dati sono quelle normali del Gw-Basic, ossia Print#, Input# e Input\$.

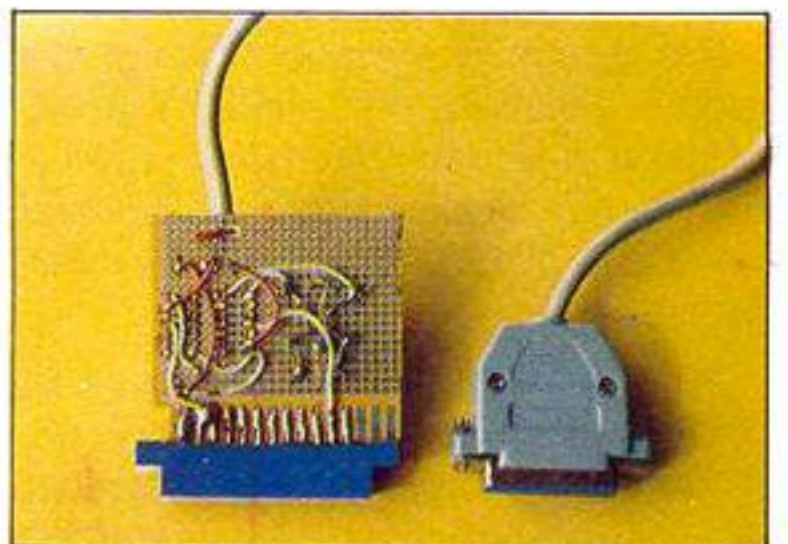
Il listato Data Comm

Insieme al progetto dell'interfaccia abbiamo sviluppato un programma che permette la comunicazione tra un Commodore 64 ed un PC compatibile con Gw-Basic.

Il programma è stato sviluppato in due versioni: una per il C/64 e l'altra in Gw-Basic per il PC.

Per il funzionamento del programma è sufficiente dare uno sguardo al menu che viene presentato dopo il RUN.

L'unica avvertenza è che, per il corretto trasferimento dei dati, è necessario impostare i parametri di comunicazione (velocità, parità, e così via) uguali su entrambi i computer.




```

10 REM ** Programma di comunica
   zione tra C/64 e RS-232
20 REM ** versione GWBASIC IBM
30 REM
90 KEY OFF
100 CLS: DIM U$(150)
110 COUNT=1
120 BAUD$="600": PARITY$="n":
   DATABIT$="8": STOPBIT$="1":
   CONFIG$=BAUD$+", " + PARITY$
   + ", " + DATABIT$ + ", "
   + STOPBIT$
150 CLS
160 PRINT"PC Terminal by Mariani
   G. & Sorgato F.":PRINT
170 PRINT"1- Configurazione
   programma"
180 PRINT"2- Ricezione dati"
190 PRINT"3- Trasmissione dati"
200 PRINT"4- Salvataggio dati su
   disco"
210 PRINT"5- Resetta (new file)"
220 PRINT"6- Visualizzazione
   dati"
230 PRINT"7- Terminale"
240 GOSUB 990
250 A$=INKEY$: IF A$<"1" OR A$>"7"
   THEN 250
260 IF A$="1" THEN GOSUB 520
270 IF A$="2" THEN 820
280 IF A$="6" THEN GOSUB 890
290 IF A$="5" THEN RUN
300 IF A$="4" THEN GOSUB 770
310 IF A$="7" THEN 420
320 IF A$="3" THEN 350
330 GOTO 150
340 REM ** trasmissione file
350 OPEN "com1:"+CONFIG$+",asc"
   FOR OUTPUT AS #2
360 CLS: INPUT" File da
   trasferire";N$:PRINT
370 OPEN N$ FOR INPUT AS #1
380 A$=INPUT$(1,#1): IF EOF(1)
   THEN 400
390 PRINT #2,A$;:PRINT A$;:
   GOTO 380
400 CLOSE #1:CLOSE #2:GOTO 150
410 REM ** terminale
420 CLS:PRINT"<esc> per finire"
430 OPEN "r",#1,"com1:"+CONFIG$+
   ",asc",2
440 B$=INPUT$(1,#1): IF ASC(B$)
   =27 THEN 510
445 IF ASC(B$)<>0 THEN A$=B$:
   GOSUB 840:B$=A$: IF E=1
   THEN 460
450 PRINT B$;
460 A$=INKEY$: IF A$="" THEN 440
490 IF ASC(A$)=27 THEN 510
500 PRINT A$;:GOSUB 852: IF E=0
   THEN PRINT #1,A$;
505 GOTO 460
510 PRINT #1,CHR$(1):CLOSE #1:
   GOTO 150
520 REM ** configurazione prog.
530 CLS:PRINT"Scegli la veloci
   ta'":PRINT:R$=CHR$(13)
540 PRINT" 150 baud"
550 PRINT" 300 baud";R$;" 600
   baud <cr> (consigliata)";
   R$;"1200 baud"
560 PRINT"2400 baud";R$
570 INPUT "Digita il valore:";A$
   :IF A$="" THEN A$="600"
580 IF A$<>"150" AND A$<>"300"
   AND A$<>"600" AND A$<>"1200"
   AND A$<>"2400" THEN 530
590 CLS:PRINT"Scegli i bit di
   dato:"
600 PRINT R$;" 8 bit di dato
   <cr> (consigliato)"
610 PRINT" 7 bit di dato";R$
630 INPUT "Digita il valore:";B$
640 IF B$="" THEN B$="8"
650 IF B$<>"7" AND B$<>"8"
   THEN 590
660 CLS:PRINT"Scegli i bit
   di stop:";R$
670 PRINT" 1 bit di stop <cr>
   (consigliato)";R$;" 2 bit
   di stop"
680 PRINT R$:INPUT "Digita il
   valore:";C$:IF C$="" THEN
   C$="1"
690 IF C$<>"1" AND C$<>"2"
   THEN 660
700 BAUD$=A$:DATABIT$=B$:
   STOPBIT$=C$
710 CLS:PRINT"Scegli la pari

```



```

ta':";R$
720 PRINT" Assenza di parita'=
'n' o <cr> (cons.)"
730 PRINT" Parita' dispari= 'o'"
;R$;" Parita' pari= 'e'";R$
740 INPUT"Digita il valore:";
PARITY$:IF PARITY$=""
THEN PARITY$="n"
750 IF PARITY$<>"n" AND PARITY$<>
"o" AND PARITY$<>"e" THEN 710
760 CONFIG$=BAUD$+", "+PARITY$+", "
+DATABIT$+", "+STOPBIT$:RETURN
770 REM ** salva dati
775 CLS:INPUT"Nome ";N$
780 PRINT:PRINT"Attendi...":OPEN
N$ FOR OUTPUT AS #1
785 K=1:C=1
790 PRINT #1,MID$(V$(K),C,1);:
C=C+1:IF C>LEN(V$(K)) THEN
C=1:K=K+1:IF K>COUNT
THEN 800
795 GOTO 790
800 CLOSE #1:RETURN
820 REM ** ricezione dati
821 IF COUNT>150 THEN 150
822 CLS:INPUT"Devo visualizzare
i dati (s/<cr>) ";X$
823 IF X$<>"s" AND X$<>" "
THEN 822
824 PRINT:PRINT"Attendere...":
PRINT:OPEN "com1:"+CONFIG$
+",asc" FOR INPUT AS #1
826 A$=INPUT$(1,#1):IF A$=
CHR$(0) THEN 835
827 GOSUB 840:IF E=1 THEN 826
828 V$(COUNT)=V$(COUNT)+A$:
IF LEN(V$(COUNT))=255 THEN
COUNT=COUNT+1:IF COUNT>150
THEN 835
829 IF X$="s" THEN PRINT A$;
ELSE RR=1-RR:LOCATE 3,14:
PRINT RR
830 GOTO 826
835 CLOSE #1:GOTO 150
839 REM ** conversione ascii
c64-pc
840 E=0:D=ASC(A$):IF D=13 THEN
RETURN
841 IF D<32 THEN E=1:RETURN
842 IF D<65 THEN RETURN
844 IF D<91 THEN A$=CHR$(D+32):
RETURN
845 IF D<95 THEN RETURN
846 IF D<124 THEN A$=CHR$(D-32)
:RETURN
848 IF D>192 AND D<219 THEN A$=
CHR$(D-128):RETURN
850 E=1:RETURN
851 REM ** conversione ascii
pc-c64
852 E=0:D=ASC(A$):IF D=13
THEN RETURN
854 IF D<32 THEN E=1:RETURN
856 IF D<65 THEN RETURN
858 IF D<91 THEN A$=CHR$(D+128)
860 IF D<95 THEN RETURN
862 IF D<124 THEN A$=CHR$(D-32):
RETURN
864 E=1:RETURN
890 REM ** visualizzazione
895 CLS
900 K=1:C=1
905 PRINT MID$(V$(K),C,1);:C=C+1
:IF C>LEN(V$(K)) THEN C=1:
K=K+1:IF K>COUNT THEN 940
910 A$=INKEY$:IF A$="" THEN 905
915 A$=INKEY$:IF A$="" THEN 915
920 IF A$=" " THEN 905
930 RETURN
940 IF INKEY$="" THEN 940
950 RETURN
990 REM ** stampa configurazione
1000 PRINT:PRINT" Baud
: ";BAUD$
1010 PRINT" Bit di dati: "
;DATABIT$
1020 PRINT" Bit di stop: "
;STOPBIT$
1030 PRINT" Parita' : "
;:IF PARITY$="n" THEN B$
="nessuna"
1040 IF PARITY$="o" THEN
B$="dispari"
1050 IF PARITY$="e" THEN
B$="pari"
1060 PRINT B$
1070 PRINT:PRINT" Memoria
libera: ";(151-COUNT)*255
;"bytes"
1080 RETURN
1090 END: REM Software by
G.Mariani & F.Sorgato

```



```

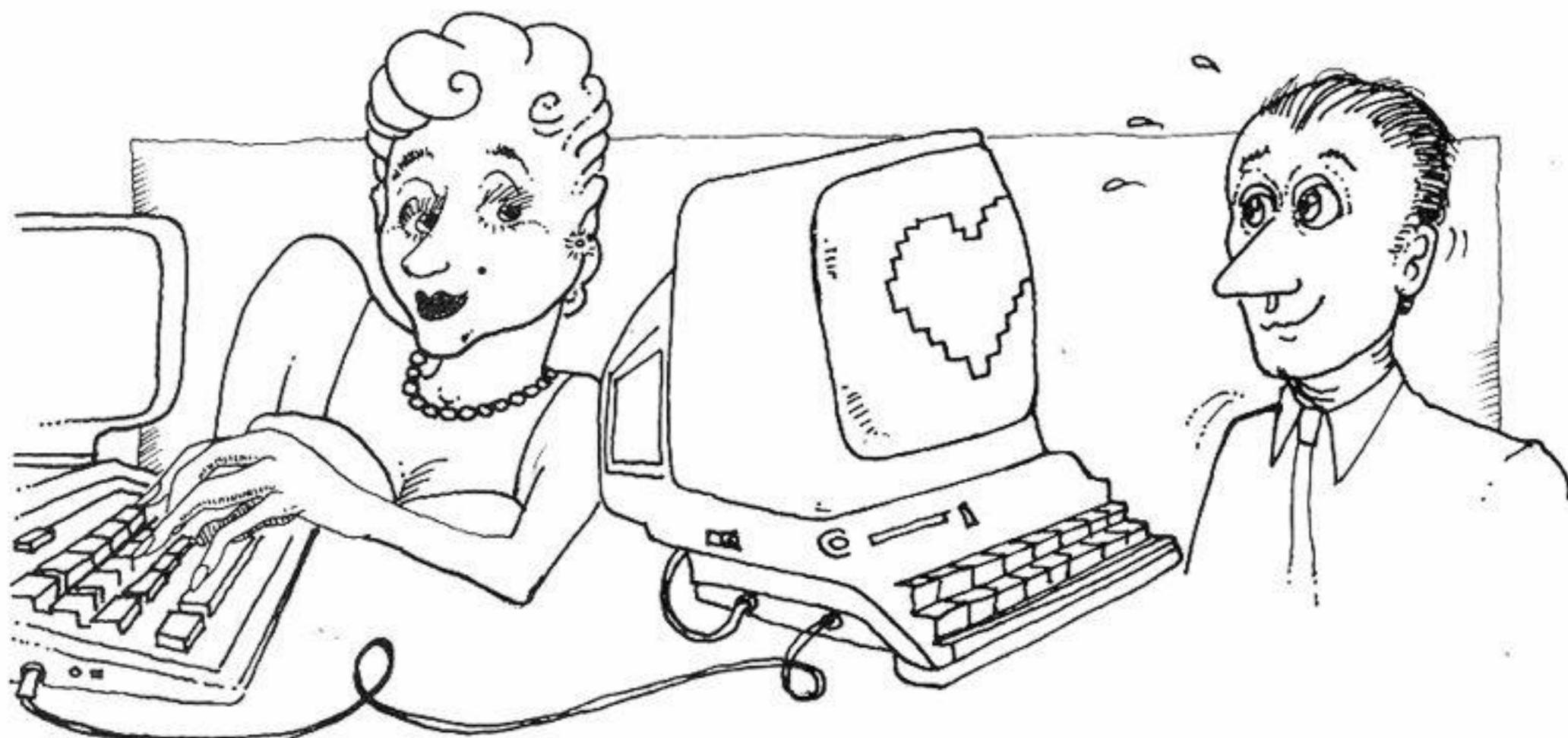
10 rem ** programma di comuni
cazione tra c64 e rs232
20 rem ** versione commodore
64
30 rem
100 poke 56,30:poke 55,255:clr
:print"[clear]"
110 poke 251,0:poke 252,31:re
m ** inizio area memorizz.
120 poke 53000,7:poke 53001,0
:if peek(49152)=162 and
peek(49153)=2 then 140
130 read a:if a<>-1 then pok
e 49152+t,a:t=t+1:goto 13
0
140 poke 53280,0:poke 53281,0
:print"[verde2]"
150 print"[clear]"chr$(14)chr$(
8);
160 poke 56,30:clr :print"C64
Terminal by Mariani G. & So
rgato F.":print
170 print"1- Configurazione pro
gramma"
180 print"2- Ricezione dati"
190 print"3- Trasmissione dati"
200 print"4- Salvataggio dati s
u disco"
210 print"5- Resetta (nuovo fil
e)"
220 print"6- Visualizzazione da
ti"

```

```

230 print"7- Terminale"
240 gosub 990
250 get a$:if a$<"1" or a$>
"7" then 250
260 if a$="1" then gosub 52
0
270 if a$="2" then 820
280 if a$="6" then gosub 89
0
290 if a$="5" then run
300 if a$="4" then gosub 77
0
310 if a$="7" then 420
320 if a$="3" then 350
330 goto 150
340 rem ** trasmissione file
350 open 2,2,0,chr$(peek(53000
))+chr$(peek(53001))
360 print"[clear]":input " File
da trasferire";n$:print
370 open 1,8,0,n$+",s,r"
380 get #1,a$:if st<>0 then
400
390 print#2,a$;:printa$;:goto
380
400 print#2,chr$(0):close 1:clo
se 2:goto 150
410 rem ** terminale
420 print"[clear] ← Per finire"
430 open 2,2,0,chr$(peek(53000
))+chr$(peek(53001)):poke 2

```

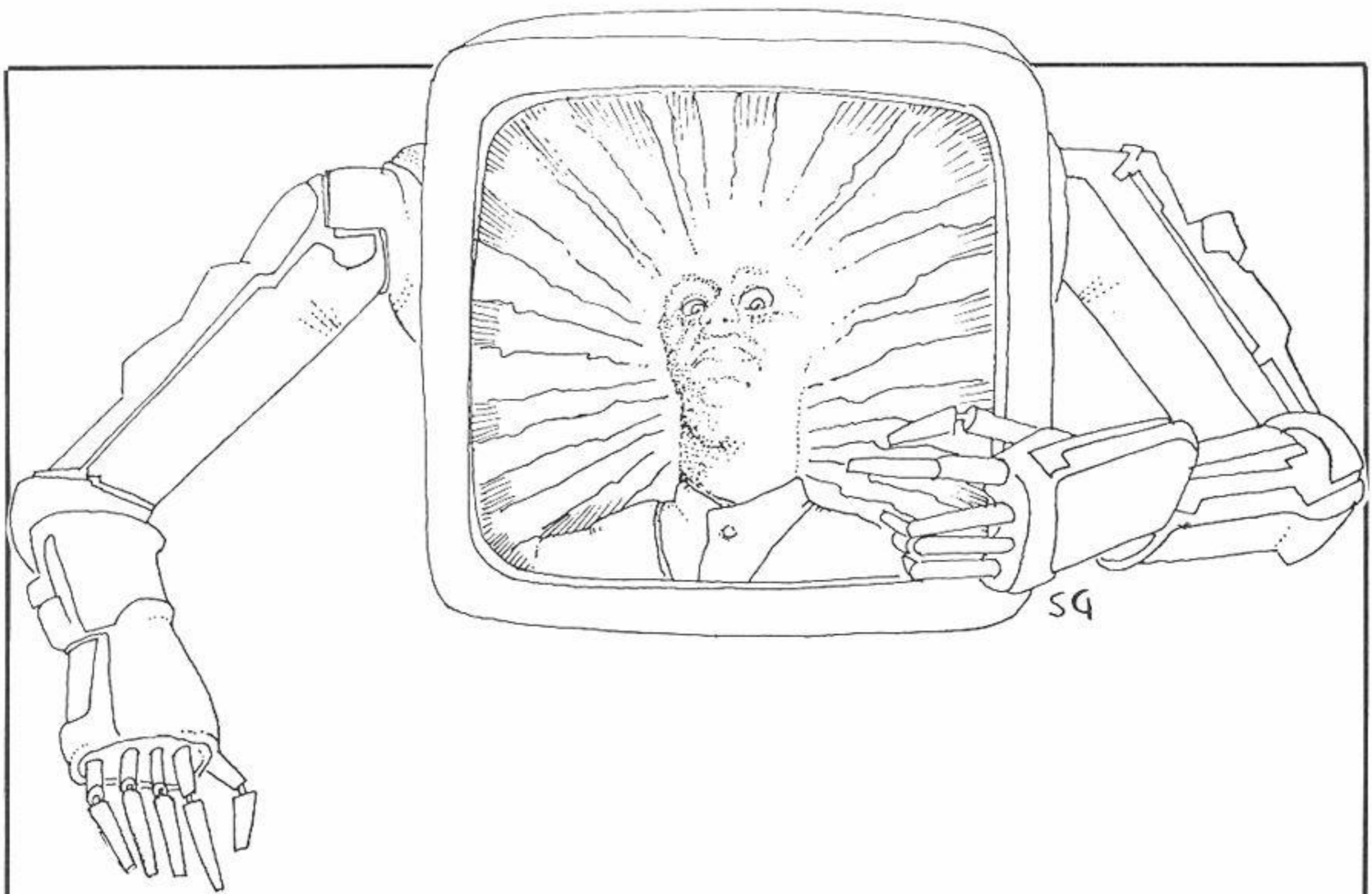


SIMONE GALLINA

```

04,0
440 get #2,b$:if b$=chr$(13)
then poke 204,1:print" "c
hr$(20);:poke 204,0
445 if b$=chr$(1) then 510
450 print b$;
460 get a$:if a$="" then a$=c
hr$(0):goto 500
470 if a$="[down]" or a$="[up]"
or a$="[left]" or a$="[rig
ht]" or a$=chr$(148) or a
$=chr$(20) then 460
480 if a$=chr$(13) then poke
204,1:print" "chr$(20);:pok
e 204,0
490 if a$="←" then 510
500 poke 212,0:printa$;:print#
2,a$;:goto 440
510 print#2,chr$(27):close 2:p
oke 204,1:goto 150
520 rem ** config. prog.
530 print"[clear][down]Scegli l
a velocita': ":print:r$=chr
$(13)
540 print" 75 baud= 2"r$" 150
baud= 5"
550 print" 300 baud= 6"r$" 600
baud= 7 o <cr> (consigliata
)"r$"1200 baud= 8"
560 print"2400 baud=10"
570 a=0:input "[down]Digita il
valore";a
580 if a=0 then a=7
590 if a<>2 and a<>5 and a<>6 a
nd a<>7 and a<>8 and a<>10
then 530
600 print"[clear][down]Scegli i
bit di dato:"
610 printr$" 8 bit di dato= 0 o
<cr> (consigliato)"
620 print" 7 bit di dato=32"
630 b=-1:input "[down]Digita il
valore";b
640 if b=-1 then b=0
650 if b<>0 and b<>32 then
600
660 print"[clear][down]Scegli i
bit di stop:"
670 printr$" 1 bit di stop= 0 o
<cr> (consigliato)"r$" 2 b
it di stop= 128"
680 c=0:input "[down]Digita il
valore";c
690 if c<>0 and c<>128 then
660
700 a=a or b or c
710 print"[clear][down]Scegli l
a parita':"
720 printr$" Assenza di parita'
= 0 o <cr> (cons.)"
730 print" Parita' dispari= 32"
r$" Parita' pari= 96"
740 d=0:input "[down]Digita val
ore";d
750 if d<>0 and d<>32 and
d<>96 then 710
760 poke 53000,a:poke 53001,d
:return
770 rem ** salva dati
780 print"[clear]":input " Nome
:";n$
790 print"[down] Attendi..":ope
n 1,8,1,n$+",s,w"
800 for k=31*256 to peek(251
)+peek(252)*256:print#1,chr
$(peek(k));:next
810 close 1:return
820 rem ** ricezione dati
830 a$="":input "[clear][down]D
evo visualizzare i dati (s/
<cr>)"a$
840 if a$<>"s" and a$<>" " t
hen 830
850 poke 2,0:if a$="s" then
poke 2,1
860 print"Attendere..[down]":op
en 2,2,0,chr$(peek(53000))
+chr$(peek(53001))
870 get #2,a$
880 sys 49152:poke 53280,0:clo
se 2:goto 150
890 rem ** visualizzazione
900 print"[clear]"
910 k=31*256
920 print chr$(peek(k));:k=k+1:
get a$:if a$=" " then 9
60
930 if k<=peek(251)+peek(252)*
256 then 920

```

```

940 get a$:if a$="" then 94 1090 if a=32 then b$="dispari
0
950 return 1100 print b$
960 get a$:if a$="" then 96 1110 print:print" Memoria libe
0 ra:";40959-(peek(251)+peek(
970 if a$=" " then 920 252)*256);"bytes"
980 return 1120 return
990 rem ** stampa config. 1130 data 162,002,032,198,255,1
1000 print:print" Baud : 60,000,032,207,255,240,251,
";:a=peek(53000) and 15 201,013,240,023,201
1010 if a=2 then a=4 1140 data 032,048,051,201,065,0
1020 if a=1 then b=50:goto 1 48,015,201,091,016,004,009,
050 128,208,007,201,096
1030 if a=10 then a=9 1150 data 048,035,056,233,032,2
1040 b=150*2^(a-5) 38,032,208,145,251,166,002,
1050 printb:print" Bit di dati 240,003,032,210,255
:";7-((peek(53000) and 32)= 1160 data 230,251,208,016,230,2
0) 52,165,252,201,160,208,008,
1060 print" Bit di stop:";1-(( 169,002,234,234,234
peek(53000) and 128)=128) 1170 data 076,204,255,169,000,1
1070 print" Parita' : ";:a= 33,253,169,255,133,254,032,
peek(53001):if a=0 then 078,241,170,208,183
b$="nessuna" 1180 data 198,253,208,246,198,2
1080 if a=96 then b$="pari" 54,208,242,240,224,-1

```