



COSTRUIAMOCI UN VERO MICROELABORATORE

HOME COMPUTER AMICO 2000

a cura della A.S.EL - parte diciottesima

In queste pagine descriviamo una scheda molto importante del sistema microcomputer Amico 2000, si tratta della scheda di I/O seriale/parallelo che ha la funzione di collegare il calcolatore con il mondo esterno. In particolar modo questa scheda permette a dispositivi cosiddetti periferici, videotermini, stampanti etc., di inviare segnali al microcomputer e di riceverne.

Come vedremo nel corso di questa trattazione esistono due tipi di trasmissione e ricezione di dati: il tipo seriale e il tipo parallelo.

Alcune periferiche, come videotermini, utilizzano una comunicazione di tipo seriale, mentre altre, come le stampanti, possono trasmettere e ricevere sia in modo seriale che in modo parallelo.

Fra le altre prestazioni attuabili con una scheda di tal genere, vi è anche la possibilità di collegare al computer

un'ulteriore scheda per la programmazione di EPROM, che è relativamente poco costosa e permette di programmare in casa questo tipo di memoria, senza dover ricorrere a laboratori esterni specializzati.

L'articolo dedicato a questa scheda, siglata A2000/19, è stato suddiviso in due parti.

Nella prima parte scendiamo in profondità sui concetti che riguardano il sistema di trasmissione seriale-asincro-

Caratteristiche generali della scheda di I/O seriale/parallelo

- Formato Eurocard 100x160 mm.
- Comunicazione seriale a 7 o 8 bit selezionabile da software
- Velocità di trasmissione a ricezione programmabile tra 75 e 4800 baud a mezzo di ponticelli
- Possibilità di programmazione software della velocità di trasmissione e ricezione
- Porta seriale completa di segnali di controllo secondo lo standard RS232
- 4 porte parallelo da 8 bit ciascuna completa di segnali di controllo handshake
- Ogni singolo bit può essere programmato come ingresso o come uscita
- 4 timer programmabili
- Possibilità di generare onde quadre o singoli impulsi
- Disponibilità di due registri seriali da 8 bit
- Possibilità di contare eventi esterni
- Occupazione di memoria di 256 Byte

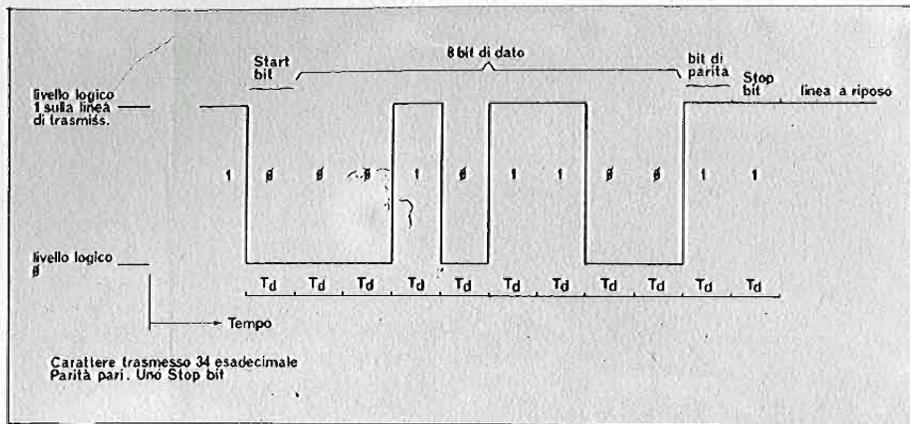


Fig. 1 - Trasmissione di tipo seriale.

no, in modo da far comprendere al lettore come esso sia stato concepito e come funziona; in particolare ci riferiremo all'integrato MC6850 che viene utilizzato in questa scheda e che adempie ai compiti di trasmissione e ricezione dei segnali.

In questa stessa parte trattiamo dello schema a blocchi di funzionamento della scheda e ne descriviamo il montaggio.

Nella seconda parte scenderemo in particolari nella descrizione della cosiddetta VIA (Versatile Interface Adapter), rappresentata nella nostra scheda dall'integrato 6522.

Essendo quest'ultimo integrato una periferica intelligente, come compo-

nente è possibile programmarla in modo da adattarla alle esigenze dell'utilizzatore. È quanto vedremo nella prossima parte, così come verrà spiegato come questa periferica sia in grado di fornire certi tipi di informazioni elaborate, senza che la CPU del sistema debba essere assolutamente interessata; ciò significa che il tempo macchina, cioè il tempo in cui la CPU lavora, non viene praticamente intaccato e non altera la velocità di elaborazione del sistema stesso.

La trasmissione e ricezione seriale

Un sistema molto usato per trasmettere dei caratteri per apparecchiature

elettroniche è quello asincrono-seriale.

Seriale significa che i bit che compongono il carattere da trasmettere vengono inviati sull'unico filo di trasmissione uno dopo l'altro, a una ben precisa velocità, quindi in serie.

Asincrono significa che il singolo carattere viene trasmesso in un istante qualsiasi, e che il ricevitore dall'altra parte della linea di trasmissione riesce, data la particolare forma di segnale, a sincronizzarsi e quindi a riconoscere ciò che gli è stato inviato.

Molti costruttori di semiconduttori, vista la diffusione di questo sistema di trasmissione, hanno progettato dei componenti particolari, i cosiddetti UART (Universal Asynchronous Receiver Transmitter), che in un solo integrato comprendono tutte le funzioni essenziali per la trasmissione e la ricezione dei dati, in particolare essi provvedono alla trasformazione serie-parallelo per la ricezione del carattere e a quella parallelo-serie per la trasmissione.

La CPU scrive in un registro dell'UART il carattere da trasmettere.

Questa ovviamente è una operazione parallela, infatti gli 8 bit di dato vengono trasferiti contemporaneamente dalla CPU all'UART tramite il BUS DATI del computer.

Appena l'UART riceve questo carattere comincia a trasmetterlo da un suo piedino di uscita, completandolo con dei segnali di controllo necessari al suo riconoscimento da parte del ricevitore.

Contemporaneamente da un suo piedino di ingresso l'UART può ricevere un segnale seriale simile, lo riconosce, lo trasforma in parallelo, e avvisa tramite un bit di stato la CPU di avere a disposizione un carattere; a questo punto la CPU può leggere in parallelo il carattere ricevuto.

Nella nostra scheda viene usato un integrato MC 6850, il classico UART della Motorola prodotto da almeno altri 3 costruttori di semiconduttori.

Composizione di un carattere seriale

Ogni carattere seriale è composto da 3 (o da 4) parti distinte:

- 1 - Uno Start bit
- 2 - Da 5 a 8 bit di dato
- 3 - Un bit di parità, pari o dispari, per il riconoscimento degli errori (opzionale)
- 4 - Uno, uno e mezzo o due Stop bit

In fig. 1 viene riportato un esempio di carattere seriale.

La caratteristica fondamentale della trasmissione ASINCRONA è la possibilità di trattare dati che si succedono in sequenza casuale.

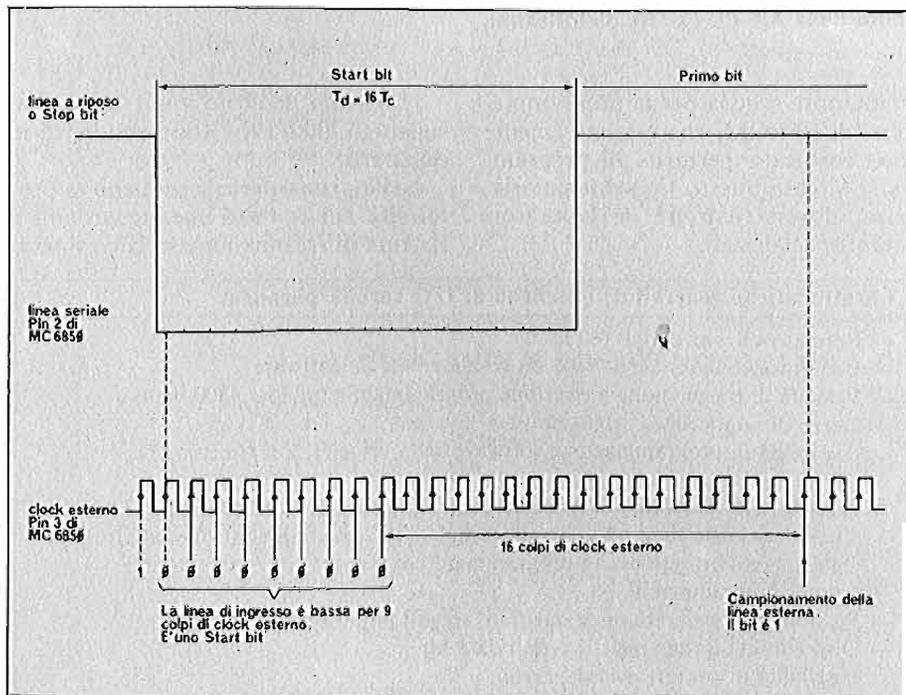


Fig. 2 - Sincronizzazione di un carattere in ricezione.

Infatti alla fine del carattere la linea di trasmissione va sempre alta per eseguire lo Stop bit, e rimane alta per un tempo indefinito, fino a che un nuovo carattere deve essere trasmesso. L'inizio del nuovo carattere è determinato dalla transizione alto-basso dello Start bit che lo precede (lo Start bit è sempre 0 o lo Stop bit sempre 1).

Il ricevitore si sincronizza ogni volta su questa transizione, per una corretta ricezione del carattere.

Il ricevitore dell'UART controlla continuamente che la linea a riposo sia in stato 1. Se la linea rimane sempre in stato 0 avvisa la CPU tramite un bit di errore dedicato, in un suo registro interno.

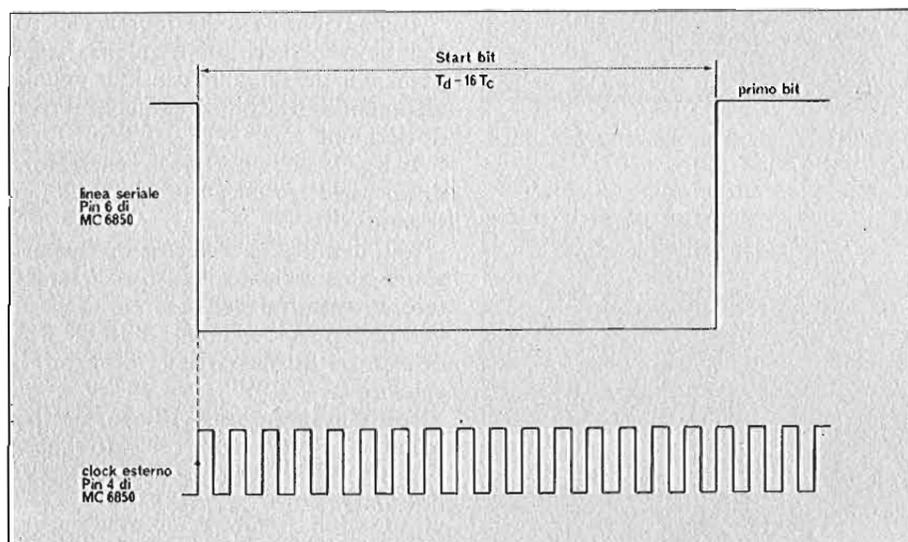
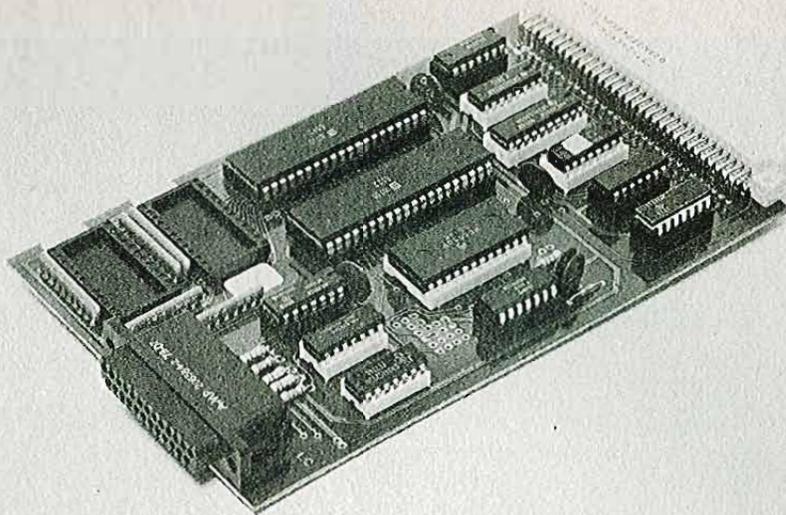


Fig. 3 - Generazione del segnale trasmesso.

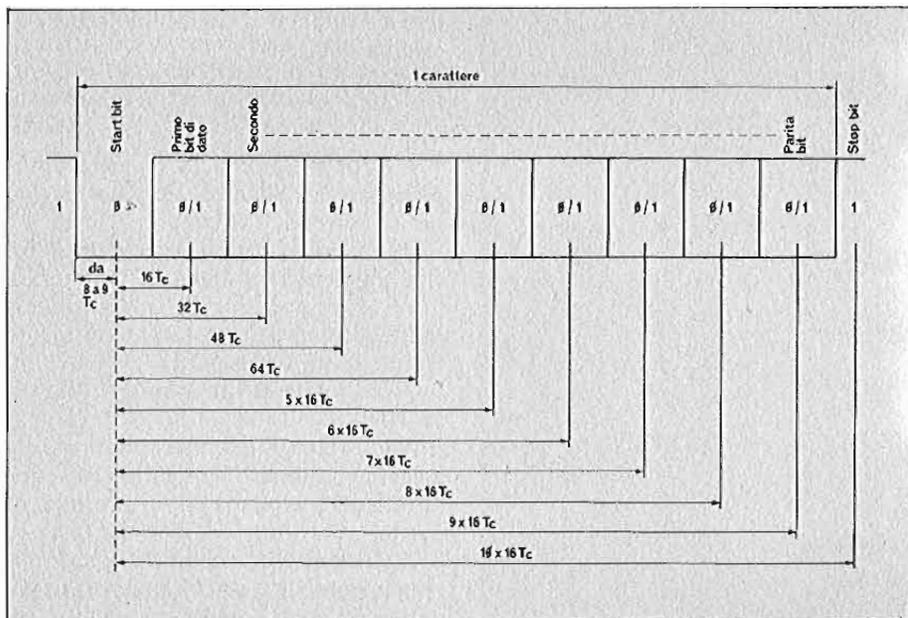


Fig. 4 - Configurazione e temporizzazioni di un carattere trasmesso.

▲ Scheda I/O seriale/parallelo. La sua funzione è di collegare il computer con l'esterno, cioè a dispositivi periferici.

Questo tipo di errore si chiama FRAMING ERROR. Quando la CPU trova questo bit a 1, deve scartare il carattere eventualmente ricevuto.

Per facilitare il riconoscimento di un carattere, il ricevitore usa un clock esterno (una onda quadra) ad una frequenza che è normalmente 16 volte la durata del bit T_d .

Può anche essere $T_d = 64T_c$ con $T_c =$ periodo del clock esterno, o $T_d = T_c$.

Nella nostra scheda si usa $T_d = 16T_c$, quindi descriveremo questo caso che è in effetti il più consueto.

Dallo schema di fig. 9B si può vedere

che $f_c = \frac{1}{T_c}$ è ricavata dal clock di mac-

china (02) tramite una catena di divisione formata da IC 9 e IC 10. Tramite ponticelli si può cambiare la f_c , e di conseguenza la velocità di ricezione e di trasmissione del 6850.

Un altro ponticello permette di ricavare il clock di trasmissione e di ricezione dal Timer presente nell'integrato IC12.

Parleremo di questa possibilità in un prossimo articolo, in cui descriveremo la sezione "parallelo" della scheda.

In fig. 2 viene descritto il sistema di sincronizzazione usato per ricevere e trasmettere dati seriali asincroni.

A ogni fronte di salita di f_c il circuito interno analizza lo stato del filo di ingresso (0 o 1). Quando trova 9 zeri consecutivi dopo un 1, sa che è arrivato uno Start bit.

A questo punto comincia a considerare la linea di ingresso ogni 16 colpi del

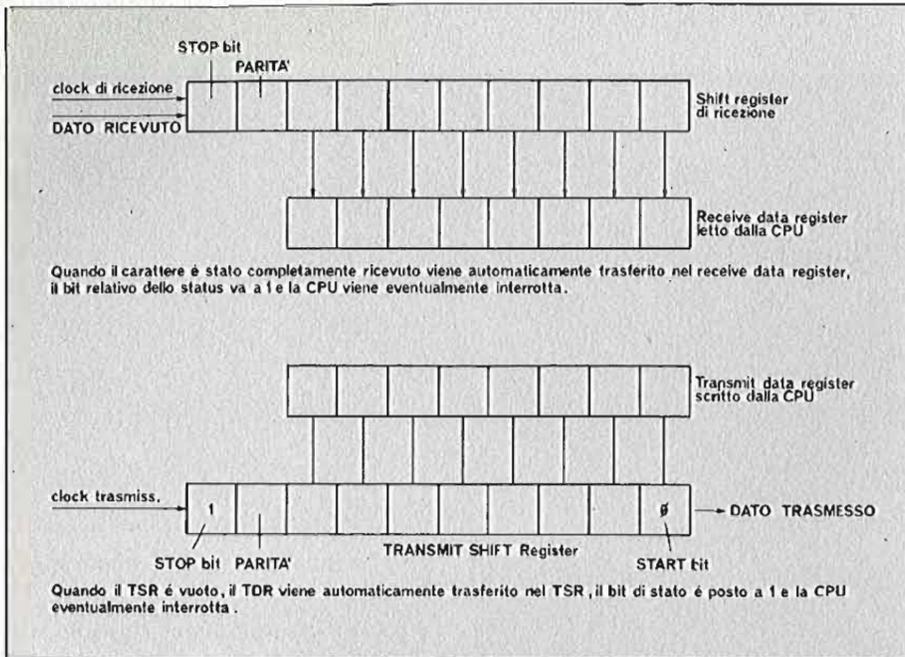


Fig. 5 - Come avviene la doppia bufferizzazione.

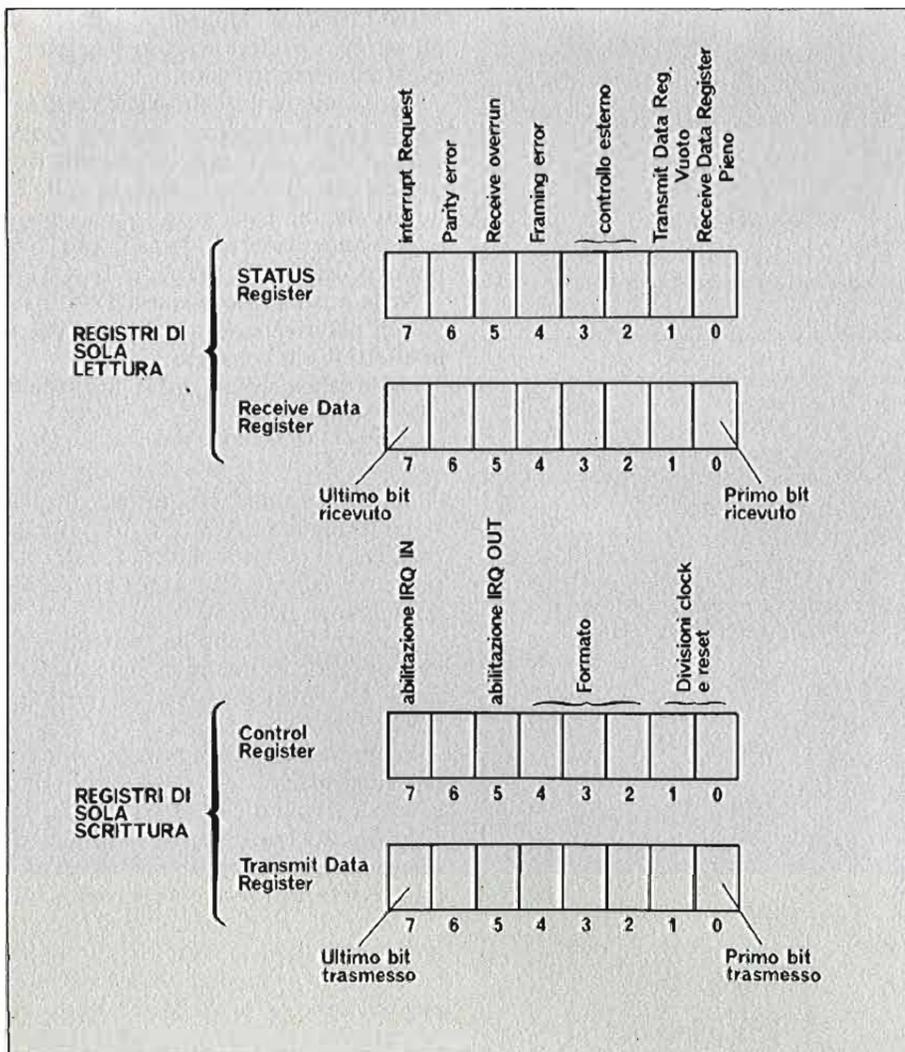


Fig. 6 - Configurazione dei registri di lettura e scrittura.

clock esterno per vedere se il filo di ingresso è alto o basso.

Facciamo ora due importanti considerazioni:

- 1 - Un disturbo in linea che faccia diventare basso il filo di ricezione per un tempo inferiore a $9T_c$ non fa partire il riconoscimento del carattere.
- 2 - Il punto di mezzo dello Start bit viene determinato con una sufficiente precisione, per permettere una buona sincronizzazione.

Dopo la sincronizzazione (determinazione del punto di mezzo dello Start bit) l'ingresso del ricevitore viene letto ogni 16 colpi di clock (T_c) in modo di andare a vedere se il filo di ingresso è a 0 o a 1 all'istante di mezzo di ogni singolo bit.

È chiaro che il ricevitore e il trasmettitore devono avere una frequenza di generazione dei bit il più possibile uguale, altrimenti si possono commettere errori di ricezione.

In fig. 3 viene riportata la generazione di un carattere da parte dell'UART in trasmissione.

Per definire la velocità di trasmissione consideriamo il caso di trasmettere 10 caratteri per secondo. Se ogni carattere richiede 11 bit (1 Start bit, 8 bit di stato, 1 bit di parità, 1 Stop bit) la velocità è:

11 bit per carattere X 10 caratteri per secondo = 110 bit per secondo (baud)

Il tempo disponibile per ogni bit è ovviamente

$$T_d = \frac{1}{\text{mS}} = 0,0091 \text{ secondi} = 9,1$$

110 (baud)

È chiaro che per ricevere e trasmettere a 110 baud si userà una frequenza esterna data da

$$f_c = 16 \cdot \frac{1}{9 \cdot 10^{-3}} = 1760 \text{ Hz}$$

L'UART esamina in maniera automatica se la sequenza ricevuta ha una parità corretta, e se ciò non è, mette a 1 un bit di errore nel suo registro di stato, che poi viene letto dalla CPU.

Per esempio, se si usa una parità dispari, il trasmettitore automaticamente trasmette un bit di parità tale per cui il numero totale di uni trasmesso sia dispari.

Allo stesso modo il ricevitore deve leggere un numero totale di uni dispari, altrimenti mette a 1 il bit di errore di parità.

Il clock esterno necessario per il funzionamento dell'UART può essere generato in molti modi: sulla scheda A 2000/19 ciò viene fatto tramite un divi-

Tabella 1 - Predisposizione del baud rate

	4	2	1	6	3	1	1	E
	8	4	2	0	0	5	1	X
	0	0	0	0	0	0	0	T
X1	x							
X2	x							
X3		x						
X4			x					
X5				x				
X6					x			
X7							x	
X8								x
X9	x	x	x	x	x	x	x	x
X10	x	x	x	x	x	x	x	x
X11								x

Tabella per la scelta dei ponticelli

						0
						X11
0	0	0	0	0	0	0
X1	X2	X3	X4	X5	X10	
0	0	0	0	0	0	0
0	0	0	0			
X6	X7	X8	X9			
0	0	0	0			
0	0	0	0			

Posizionamento fisico ponticelli

Tabella 2 - Collegamenti al connettore di comunicazione seriale RS232

- 1 GND
- 2 Receive Data
- 3 Transmit Data
- 4 D. C. D.
- 6 + 12 V (3,3 kΩ in serie)
- 7 GND
- 8 R. T. S.
- 9 + 12 V (3,3 kΩ in serie)
- 18 - 12 V (3,3 kΩ in serie)
- 20 C. T. S.

sore programmabile (IC 9 e IC10) e con una serie di ponticelli che servono a determinare la frequenza (vedere tabella 1).

Sempre sulla nostra scheda il *baud rate* (velocità di trasmissione espressa in baud) può essere ricavato, tramite ponticello, dal timer presente sull'integrato IC 12 (VIA) e programmato da software. Vedremo in un prossimo articolo questa possibilità.

Sia il divisore programmabile che la VIA ricavano la base di conteggio dal clock di macchina a 1MHz, che è quarzato e quindi molto stabile.

Il 6850 comprende al suo interno sia il ricevitore che il trasmettitore di dati, e ha i pin del clock del ricevitore e del trasmettitore separati fra di loro. Le nostre applicazioni però prevedono che sia il Tx che l'Rx funzionino sempre alla stessa frequenza, per cui i due pin (il 3 e il 4) sono cortocircuitati fra di loro.

È tipica inoltre nella scheda che descriviamo la doppia bufferizzazione. Ciò significa (vedere fig. 5) che il dato ricevuto viene fatto entrare nel registro a scorrimento di ingresso (Receive Shift Register) utilizzando il clock di ricezione, che è stato sincronizzato con i dati nel modo che abbiamo già descritto.

Quando il carattere è stato ricevuto, insieme alla parità (se c'è) e allo Stop bit, avvengono due cose:

- 1 - Il dato viene automaticamente trasferito nel registro di ricezione RDR (Receive Data Register).
- 2 - Il bit di stato del 6850 che dichiara la avvenuta ricezione di un carattere, viene posto a 1 e, se abilitato, viene attivato l'interrupt (pedino \overline{IRQ} dell'UART a \emptyset).

Appena è stato eseguito il trasferimento nel RDR, l'UART riprende a testare l'ingresso alla ricerca di un nuovo Start bit.

La CPU ha dunque a disposizione tutto il tempo di ricezione di un carattere (circa 10 ms a 1200 baud) per prelevare il carattere e analizzarlo.

Se la CPU non esegue questo prelievo, e nel frattempo è arrivato un altro carattere, viene posto a 1 da parte del 6850 un bit dello Status che si chiama "Overrun error".

La stessa cosa avviene in trasmissione.

La CPU scrive nel cosiddetto TDR (Transmit Data Register) il dato da far uscire serialmente, immediatamente il 6850 lo trasferisce nello SR (Shift Register) di uscita e comincia a farlo uscire.

A questo punto il 6850 è in grado di ricevere immediatamente un altro dato nel TDR. Se la CPU scrive subito un

carattere, la UART mette a \emptyset il bit dello status Transmit Data Register EMPTY per segnalare che ha occupati sia lo SR di uscita che il TDR.

Appena il primo carattere è stato trasmesso e il TDR è stato trasferito nello SR, il bit viene posto a 1 e se abilitato, l' \overline{IRQ} viene generato.

La CPU così è in grado di sapere che può scrivere nell'UART un nuovo carattere.

Un altro segnale importante che entra dall'esterno è il CTS (Clear To Send) che serve a dire all'UART che chi è abilitato a ricevere il segnale seriale è pronto a farlo.

Useremo questo segnale essenzialmente per le stampanti che, se sono occupate a stampare e non possono con-

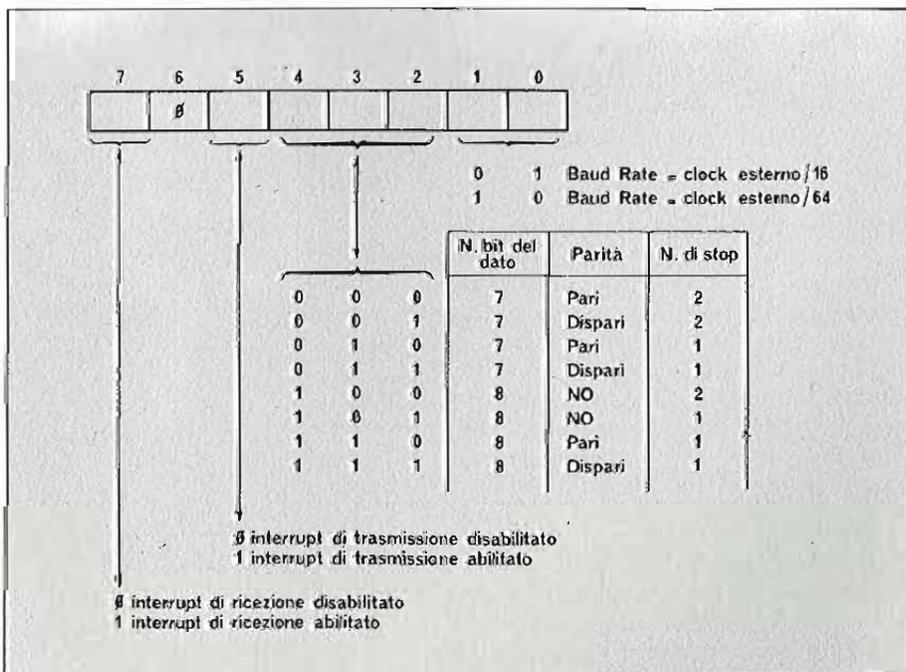


Fig. 7 - Significato dei bit del Control Register.

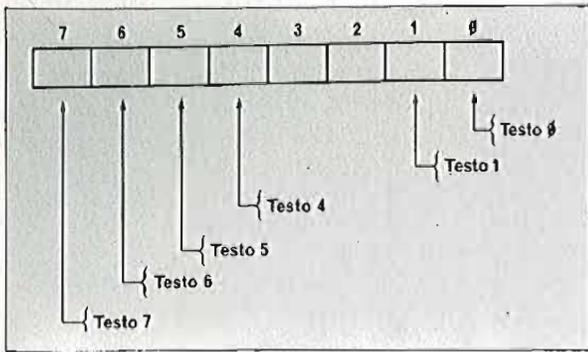


Fig. 8 - Significato dei bit dello Status Register.

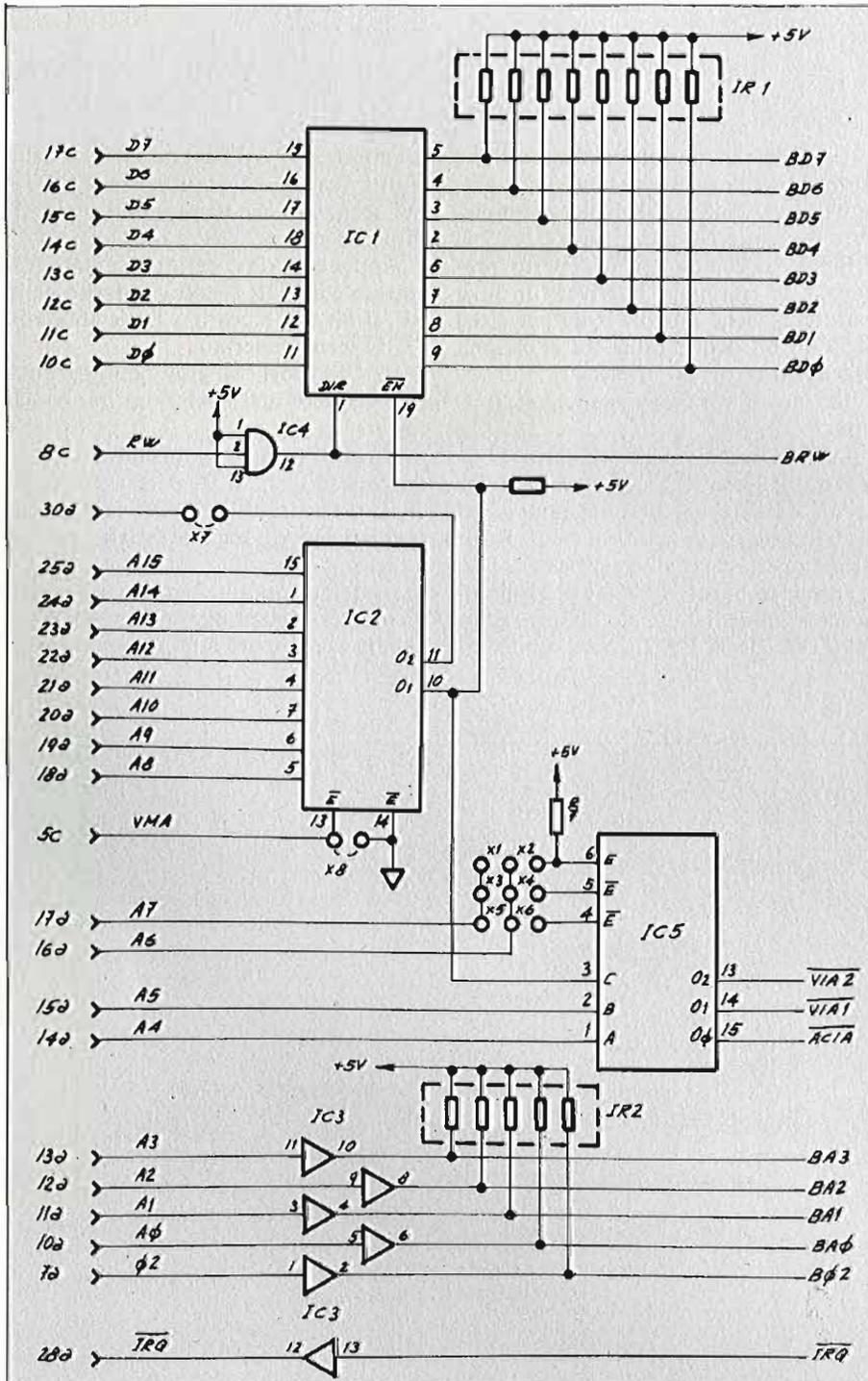


Fig. 9A - Schema elettrico: Interfaccia Bus.

temporaneamente ricevere i dati, segnalano questa loro non disponibilità tramite una uscita.

Automaticamente l'UART blocca la trasmissione per riprenderla quando la stampante è ancora in grado di ricevere.

Nella tabella 2 vengono riportate le denominazioni e i collegamenti al connettore di comunicazione seriale.

I Registri del 6850 sono due che possono essere o solo letti o solo scritti.

All'indirizzo 0 (più la base di indirizzamento che nel nostro caso è FA00) in lettura c'è il registro di stato, e in scrittura quello di controllo.

All'indirizzo 1 (FA01) si trovano in lettura il Receive Data Register e in scrittura il Transmit Data Register.

Volendo ricevere o trasmettere un carattere bisognerà eseguire le seguenti funzioni:

- 1 - Reset dell'UART scrivendo 03 nel Control Register.
- 2 - Set del modo di funzionamento scrivendo nel Control Register quanto voluto.
- 3 - Per trasmettere si scrive un dato nel Transmit Data Register quando il bit 1 dello Status Register è a 1.
- 4 - Per leggere un carattere ricevuto lo si legge dal Receive Data Register quando il bit 0 dello Status Register è a 1.

Montaggio della scheda di I/O

Nessun problema per il montaggio di questa scheda che non presenta difficoltà particolari. Come di consueto si comincia con la saldatura dei componenti dimensionalmente più bassi, ovvero le resistenze.

Seguono gli zoccoli per gli integrati (fare attenzione all'orientamento seguendo la serigrafia dei componenti riportata sul circuito stampato come appare in fig. 10). È importante saldare tutti i piedini, anche quelli che sembrano isolati, perchè si tratta di un circuito stampato a doppia faccia.

Saldare ora i diodi D1 e D2, badando alla corretta polarità, i condensatori CF e i 7 array di resistenza IR1 - IR7.

Si passi ora a fissare bene nella propria sede il connettore del bus stando attenti, prima di saldare, che tutti i terminali siano bene inseriti nello stampato: attenzione che una volta saldato il connettore non può essere più rimosso. Fare le stesse operazioni per il connettore J1.

Controllate ora tutte le saldature e procedete ad inserire gli integrati negli zoccoli corrispondenti sempre badando bene al corretto orientamento.

A questo punto il montaggio è ultimato e la scheda di I/O può essere su-

bito inserita in uno dei posti del mother board per il collaudo.

Collegamento di una stampante

Come abbiamo detto all'inizio, una funzione tipica di questa scheda è quella di permettere il collegamento del computer ad una stampante che può essere usata sia sotto il controllo del "monitor" che pilotata con istruzioni BASIC.

La prima operazione da fare è il collegamento fisico fra stampante e scheda. Nell'esempio che descriveremo in

queste pagine facciamo riferimento alla stampante modello Sara 10 costruita dalla Honeywell ISI. Ci si dovrà procurare due connettori a vaschetta da 25 contatti, di cui uno maschio (da collegare al corrispondente femmina presente sulla scheda) e un altro adatto alla stampante utilizzata. I collegamenti dei vari fili, nel caso della Sara 10, sono riportati nella tabella 3: due connessioni sono fatte in modo diretto (terminali 3 e 7) e riguardano i segnali dei dati trasmessi; il collegamento per i segnali di

Tabella 3 - Cavo di interfaccia per la stampante SARA 10

Lato interfaccia	Lato stampante
3 ○	○ 3
7 ○	○ 7
4 ○	○ 20
	○ 8
	○ 21

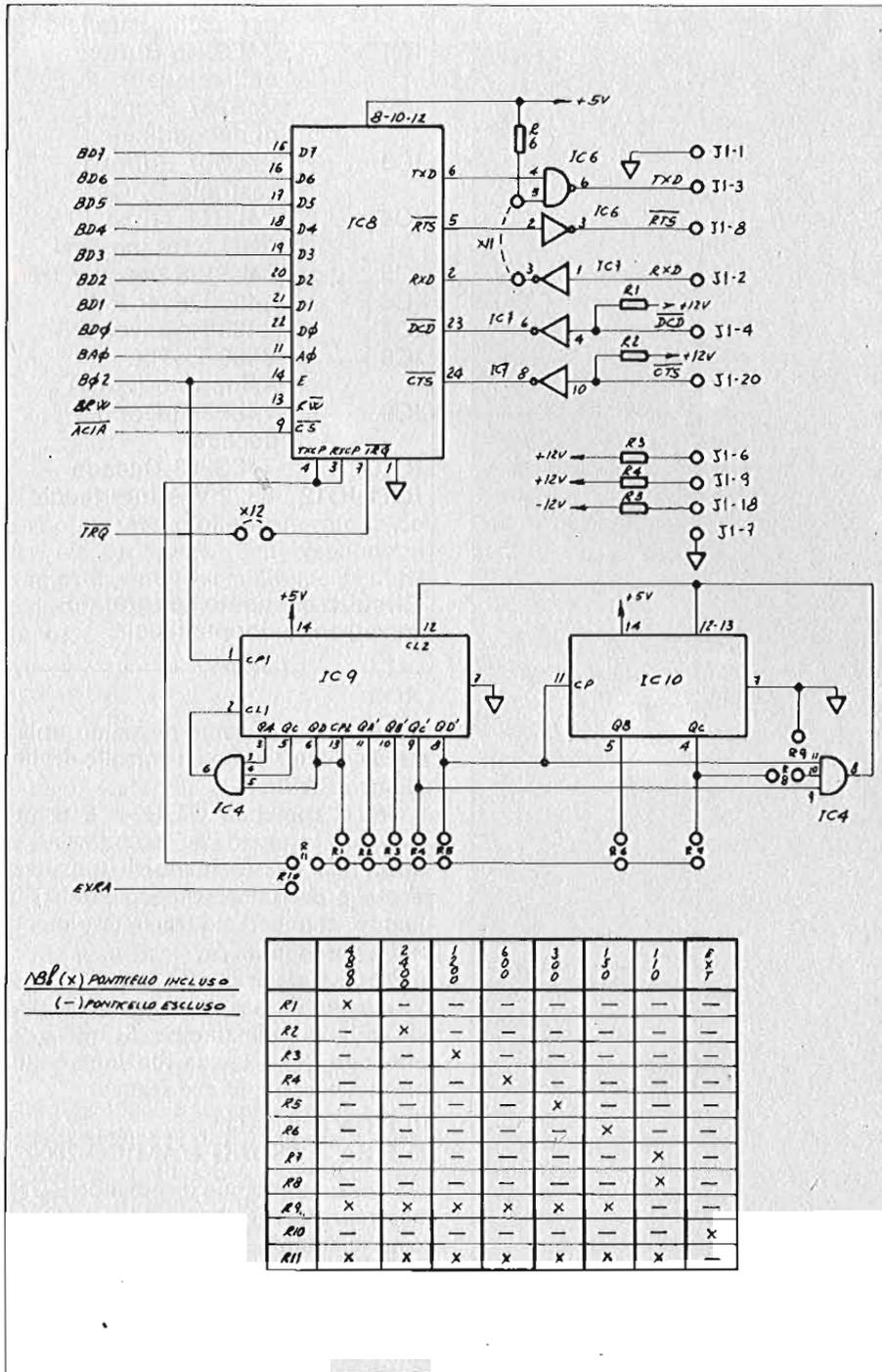


Fig. 9B - Schema elettrico: ACIA e baud rate generator.

controllo dipende invece dal tipo di stampante usata. Si noti comunque che il segnale di controllo della stampante deve essere in ogni caso connesso al contatto 20 del connettore della scheda I/O.

Fatto il cavo di collegamento ci si deve assicurare che la scheda comunichi alla stampante alla stessa velocità alla quale quest'ultima è abilitata a ricevere. Per quanto riguarda la scheda vi rimandiamo a quanto detto in questa trattazione circa il baud rate, mentre per la stampante bisognerà procedere ad una accurata lettura del proprio manuale d'uso.

Nel caso specifico della Sara 10 il cursore degli interruttori S03 e S04 deve essere posizionato verso il basso, predisponendo così una ricezione alla velocità di 1200 baud.

A questo punto possiamo accendere l'AMICO e la stampante, lanciare a F400 il Monitor Video, premere il pulsante ON/OFF LINE sulla stampante in modo da accendere il led di ON LINE e quindi procedere al collaudo.

Prima di tutto si preme il tasto CTRL e il tasto P e sul video appare:

> ON

Ripremendo nuovamente i due tasti otterremo:

> ON OFF

La scritta che di volta in volta appare, indica lo stato della stampante in quel momento.

Ci riporteremo nuovamente nella situazione ON e procederemo nella nostra prova eseguendo il comando T per stampare una zona di memoria a nostro piacere.

Se tutto funziona bene otterremo la stampa della memoria contemporaneamente sul video e sulla stampante; se qualcosa non funziona controllate le connessioni e ricominciate daccapo.

Di seguito riportiamo una routine che può essere usata durante l'esecuzione di un programma per stampare qualsiasi carattere ASCII contenuto nell'accumulatore.

A8
A9 02
2C 00FA
F0 FB
68
8D 01FA

ENTRO
STAMPA

PHA
LDA # \$02
BIT ACIAST
BEQ STAMPA
PLA
STA ACIADA

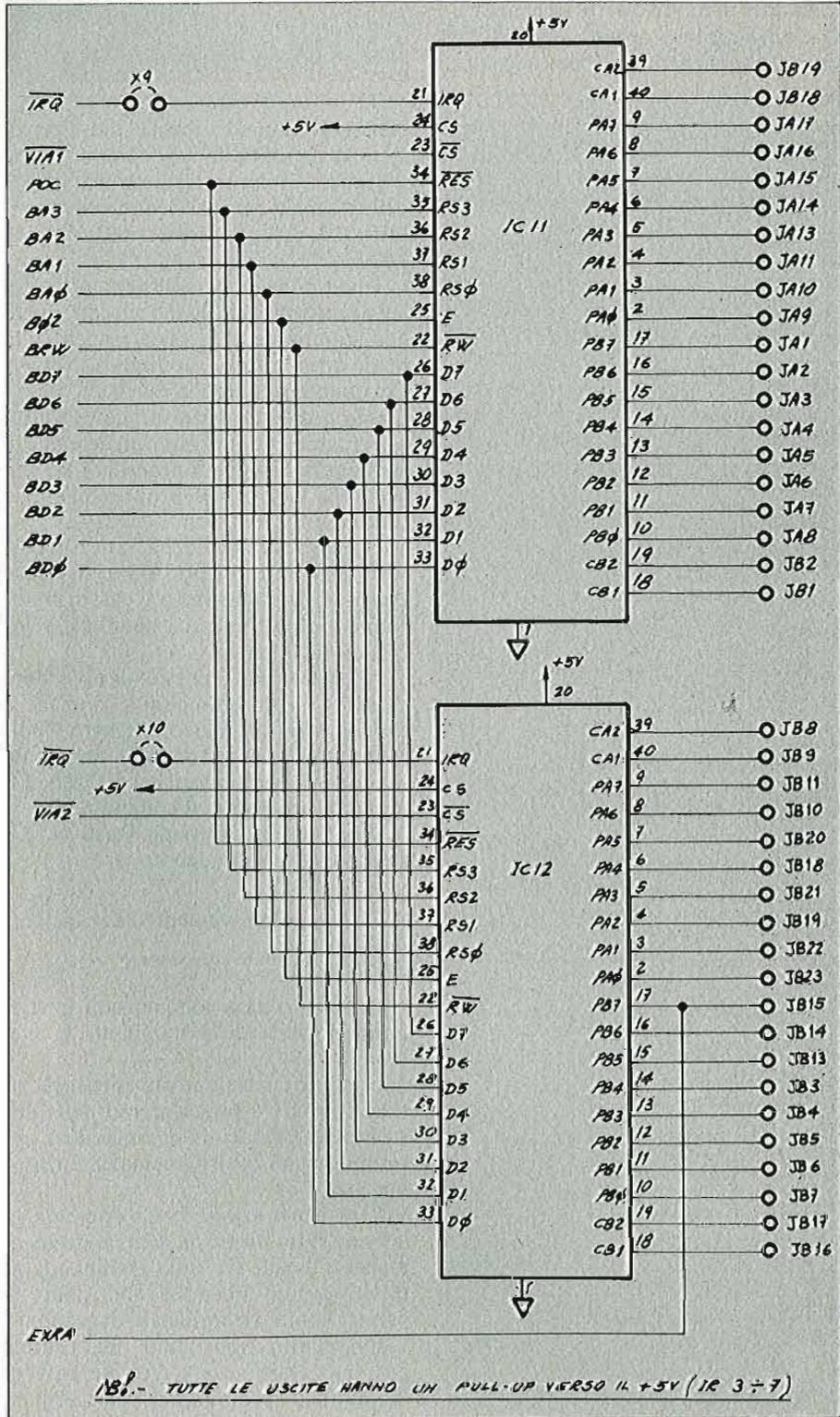
Salvo il carattere
Presetto l'accumulatore
Testo lo status dell'ACIA
Aspetto il libero
Recupero il carattere
Deposito il registro
di uscita
Ritorno

60

RTS

Elenco componenti scheda I/O	
R1	: 3,3 kΩ
R2	: 3,3 kΩ
R3	: 2,2 kΩ
R4	: 2,2 kΩ
R5	: 2,2 kΩ
R6	: 3,3 kΩ
R7	: 4,4 kΩ
IR1-IR7	: ARRAY da 4,7 kΩ
D1-D2	: 1N4148
CF	: 47 nF
J1	: Connettore 25 poli
J _A / J _a	: Zoccoli 24 pin per uscita parallelo
IC1	: 74LS245 Buffer bidirezionale
IC2	: 74S287 Prom di decodifica
IC3	: 74LS07 Buffer sestuplo O. C.
IC4	: 74LS11 Tripla AND a tre ingressi
IC5	: 74LS138 Decoder 1/8
IC6	: 1488 Driver EIA
IC7	: 1489 Receiver EIA
IC8	: 6850 Tx/Rx Asincrono (UART)
IC9	: 74LS393 Doppia decade
IC10	: 74LS293 Decade
IC11-IC12	: 6522 VIA Interfaccia parallelo

Circuito stampato serigrafato in vetronite doppia faccia.



Vediamo ora come possiamo utilizzare la stampa sotto il controllo del linguaggio BASIC.

Con il comando CTRL + P si può stampare tutto ciò che viene inviato al video, ma questo modo di funzionamento è decisamente poco pratico in quanto stamperemo ragionevolmente molte cose inutili.

Per ovviare a questo inconveniente potremo utilizzare un comando del BASIC relativo alla stampa. Si tratta del comando "!" e la sua funzione è illustrata nell'esempio che segue:

```
10 PRINT "CIAO"
20 PRINT! "SONO L'AMICO 2000"
```

Se ora, utilizzando il comando RUN, facciamo partire il programma otterremo la stampa sul video di entrambi i messaggi, mentre solo il secondo verrà inviato alla stampante.

L'invio selettivo dei dati alla stampante viene controllato dallo stato del bit 6 della locazione di memoria F8.

Utilizzando questa particolarità pos-

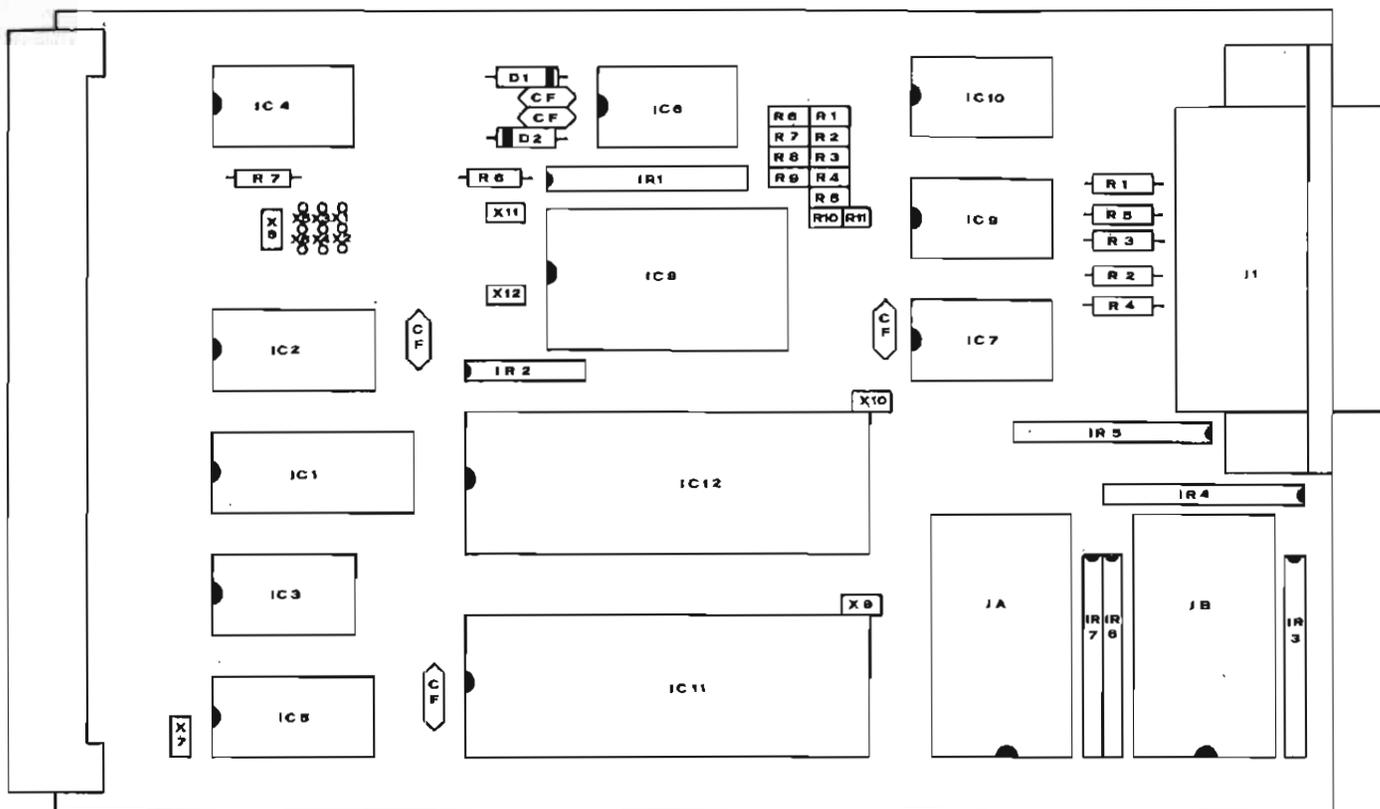


Fig. 10 - Serigrafia disposizione componenti sul circuito stampato scheda interfaccia I/O.

siamo ovviare all'inconveniente di dover far procedere ogni comando di stampa da un "!" semplicemente utilizzando la routine che riportiamo di seguito:

A9	40	TOGGLE	LDA	#\$40	Presetto l'accumulatore
45	F8		EOR	\$F8	Commuto il bit 6
85	F8		STA	\$F8	Deposito il risultato
60			RTS		Ritorno

Altro primo piano della scheda I/O seriale/parallelo. ▼

Ogni volta che chiameremo, impiegando il comando di BASIC `USR`, la routine in questione otterremo lo stesso effetto del comando `CTRL + P` senza però l'inconveniente di stampare le scritte `ON` e `OFF`.

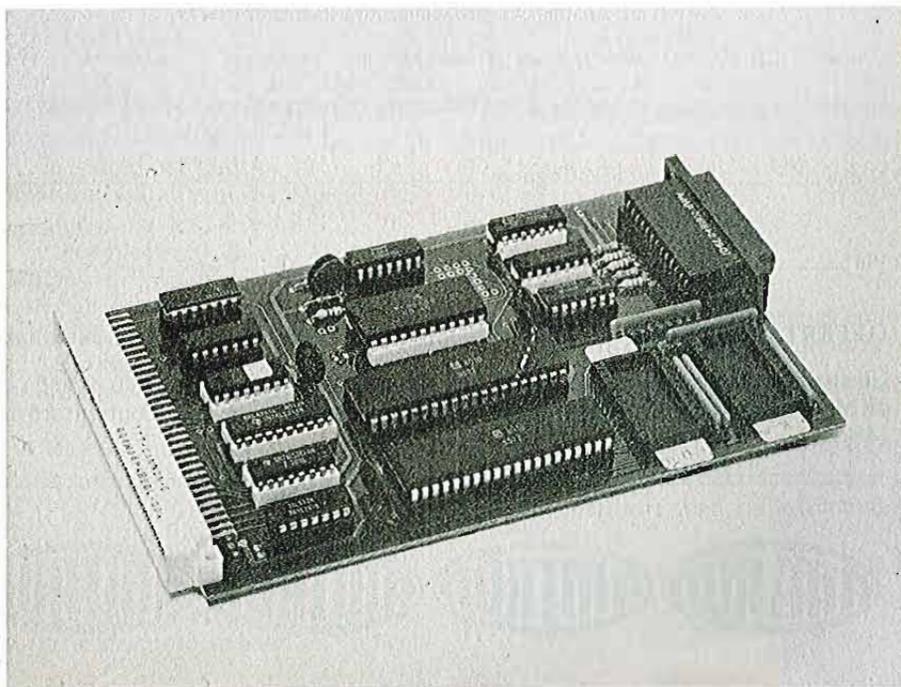
La sequenza corretta per chiamare dal programma BASIC la routine in questione (precedentemente memorizzata alla locazione `02000`) è la seguente:

```
XXXX GOSUB 10000
```

alla riga 10000 e seguenti avremo la seguente sequenza di istruzioni:

```
10000 POKE 4,0 : POKE 5,2
10005 Z = USR (Z1)
10010 RETURN
```

Riteniamo con questi esempi di avervi messi in grado di utilizzare convenientemente la stampante e vi consigliamo di provare e di riprovare per prendere confidenza con questa nuova periferica.



Nuovo listino

Inviatemi a stretto giro di posta il seguente materiale:

- (quantità) _____ AMICO 2000/1 K in scatola di montaggio (completo di 1 K byte di RAM e interfaccia per registratore a cassette). Lit. 249.500 (+ IVA)
- (quantità) _____ AMICO 2000/2 montato e collaudato (con 1K byte di RAM e interfaccia per registratore a cassette) Lit. 305.300 (+ IVA)
- (quantità) _____ art. A2000/3K Alimentatore da 1A in kit adatto per alimentare il microcomputer. Lit. 16.500 (+ IVA)
- (quantità) _____ art. A2000/6 Scheda per espansione sistema (accetta fino a 9 schede formato EUROPA) completa di buffer dati e indirizzi Lit. 93.000 (+ IVA)
- (quantità) _____ art. A2000/7K Alimentatore di potenza per il sistema espanso (+5V/8A, ±12V/0,8A, -5V/0,5A) in kit montaggio Lit. 114.000 (+ IVA)
- (quantità) _____ art. A2000/7 (come sopra montato e collaudato) Lit. 144.000 (+ IVA)
- (quantità) _____ art. A2000/9K Contenitore per il sistema completo in kit (completo di interruttori e minuterie) Lit. 144.000 (+ IVA)
- (quantità) _____ art. A2000/10 Contenitore per il sistema completo di scheda per espansione (art. A2000/6) e alimentatore (art. A2000/7), tutto montato e collaudato. Lit. 350.000 (+ IVA)
- (quantità) _____ art. A2000/11K Scheda di interfaccia video in kit. Lit. 224.000 (+ IVA)
- (quantità) _____ art. A2000/11 come sopra montata e collaudata Lit. 249.500 (+ IVA)
- (quantità) _____ art. A2000/14K tastiera ASCII completa di contenitore e cavo di collegamento, in kit di montaggio. Lit. 129.000 (+ IVA)
- (quantità) _____ art. A2000/14 come sopra montata e collaudata Lit. 144.000 (+ IVA)
- (quantità) _____ art. A2000/16BK Scheda RAM/ROM completa di linguaggio BASIC standard, in scatola di montaggio. La scheda è fornita con 4Kbytes di RAM e 8Kbytes di ROM (BASIC). È possibile montare fino a 16Kbytes di ROM o EPROM la cui decodifica avviene tramite una PROM. Lit. 269.000 (+ IVA)
- (quantità) _____ art. A2000/16B Come la 16BK, montata e collaudata. Lit. 299.000 (+ IVA)
- (quantità) _____ Sole EPROM del BASIC da 8K e nuova EPROM di Monitor video, per l'aggiornamento della scheda di Mini-BASIC. Lit. 120.000 (+ IVA)
- quantità _____ art. A2000/18K 16 Scheda di RAM dinamica da 16Kbytes (ampliabile sino a 32 K) in scatola di montaggio. Lit. 299.000 (+ IVA)
- quantità _____ art. A2000/18K 32 Scheda di RAM dinamica da 32Kbytes in scatola di montaggio. Lit. 399.000 (+ IVA)
- quantità _____ art. A2000/18-16 Come la /18K 16, montata e collaudata. Lit. 319.000 (+ IVA)
- quantità _____ art. A2000/18-32 Come la /18K 32, montata e collaudata. Lit. 419.000 (+ IVA)
- quantità _____ art. A2000/19K Scheda di Input/Output seriale e parallela. Dispone di un port di comunicazione seriale RS 232 con baud rate programmabile tra 110 baud a 4800 baud; quattro port da 8 bit completi di segnale hand shake utilizzabili (con altra espansione) per la programmazione di EPROM. Adatta per interfacciamento a stampante o terminale video. Lit. 134.000 (+ IVA)
- quantità _____ art. A2000/19 Come la /19K, montata e collaudata. Lit. 154.000 (+ IVA)

Per il pagamento scelgo la forma:

- anticipato a mezzo assegno circolare o vaglia (spese di spedizione a carico della ASEL);
- in contrassegno alla consegna del pacco - spese di spedizione a carico del Committente.

IMPORTANTE: La merce viaggia a rischio e pericolo del Committente; è possibile assicurarla aggiungendo Lit. 6.000 per le prime 100.000 di capitale assicurato. Per pacchi di valore superiore consultare le tariffe presso gli uffici postali.

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

Inviare il presente modulo in busta chiusa con allegata copia della ricevuta del vaglia alla:

A.S.E.L. s.r.l. - Via Cortina D'Ampezzo, 17
Milano (Tel. 02/5695735)

PREZZI VALIDI DALL'1-4-81

Nome _____ Cognome _____ Tel. _____

Via _____ Codice Fiscale _____ CAP _____ Città _____

OFFERTA SPECIALE per chi vuole cominciare direttamente con il Personal Computer.

Un sistema completo (art. A2000/P) composto da: CPU AMICO 2000, scheda BASIC da 8K, scheda Interfaccia video, 4Kbytes di RAM, Alimentatore di potenza, tastiera alfanumerica e contenitore per tutto il sistema. Lit. 1.195.000 (+ IVA 14%).

