

# 8520: Complex Interface Adapter

di Maurizio Damiani Chersoni, Luca Galeani e Giuliano Peritore

*Una nuova rubrica dedicata all'Amiga? Forse, nel senso che i presupposti ci sono tutti, la voglia naturalmente non manca, e gli argomenti che tratteremo sono (a mio sindacabilissimo giudizio) più che interessanti. Queste pagine della rivista ospiteranno, con cadenza quasi mensile (molti argomenti sono ancora allo studio), articoli riguardanti l'hardware dell'Amiga. Ma non solo: tutte le volte che sarà possibile, gli articoli saranno accompagnati da semplici progetti con i quali i lettori, saldatore in mano, potranno sbizzarrirsi a vedere in qualche modo applicati gli argomenti trattati quel mese. Sia come circuitini accessori, sia come vere e proprie modifiche alla macchina per migliorare alcune performance. E ne vedremo delle belle: purtroppo non posso anticiparvi di più. L'unica cosa che posso dirvi è che l'argomento di questo mese riguarda due bei chipponi a quaranta piedini di nome 8520 e cognome CIA che gestiscono da soli tutto l'I/O della macchina. Il progettino di questo mese è un simpatico albero di Natale con le lampadine intermittenti a ritmo di drive... Buona lettura*

adp

## 8520: informazioni generali

«8520 chi era costui?»

Così il buon Don Abbondio avrebbe esordito se accanto al breviario avesse avuto la raccolta completa di MC?

Ebbene stiamo parlando di un signor chip della bellezza di 40 piedini che praticamente assolve da solo il compito di interfacciare l'Amiga con il mondo esterno.

I chip custom dell'Amiga sono infatti affiancati, per quanto riguarda la gestione dell'I/O da due chip standard chiamati 8520, che vengono indicati usualmente con CIAA e CIAB, dove CIA è l'acronimo di Complex Interface Adapter. I due componenti di cui stiamo parlando sono parti fondamentali dell'Amiga in quanto, tranne la parte Video e quella Audio, gestiscono praticamente tutto. Ogni 8520 contiene infatti due porte parallele bidirezionali ad otto bit, due linee PC E FLAG per la gestione dell'handshaking, due timer a sedici bit, un TOD (acronimo di Time Of Day, letteralmente: Ora del Giorno), una porta seriale ad otto bit, ed ha inoltre la possibilità di generare cinque differenti tipi di interrupt.

Stavamo dicendo che questi due chip (... poni) sono punti chiave dell'architettura dell'Amiga in quanto sono utilizzati per la gestione delle porte seriale e parallela, della porta dei drive, di parte della porta giochi, per le sincronizzazioni, per lo switching fra i task, per gli interrupt e perfino per la gestione della tastiera.

Se vi interessa sapere come queste operazioni sono distribuite fra i due 8520 potete andare a vedere le tabelle 1,2,3 e 4 che riportano le mappe degli indirizzi e l'assegnamento delle connessioni.

Tuttavia, nonostante la gran mole di lavoro che questi due chip svolgono, la loro complessità non è eccessiva come

potrete rendervi conto tra breve leggendo le righe seguenti.

## Porte parallele di I/O

Abbiamo già visto che ogni 8520 contiene due porte parallele. Chi ha già dato un'occhiata alle tabelle 1-4 si sarà già fatto un'idea di come queste porte vengano usate nell'Amiga, ma dato che... repetita iuvant... descriviamo il tutto a parole in modo da rendere più chiare le tabelle.

D'ora in poi indicheremo le due 8520 con CIAA e CIAB. Dato che ogni 8520 contiene due porte parallele chiameremo anche queste PA e PB. La porta CIAA/PB è dedicata totalmente alla gestione della porta parallela (stampante, digitalizzatori, ecc.), e non vi è bisogno di particolari spiegazioni in quanto si tratta semplicemente di otto linee dati.

La porta CIAA/PA si dirama un po' per tutto l'Amiga.

Infatti i bit 6,7 sono dedicati alla lettura delle linee FIRE delle Gameport 0,1. I bit 5,4,3,2, sono dedicati invece alla gestione del drive e sono alcuni di quelli che useremo nel circuitino di questo mese. I compiti da loro svolti sono rispettivamente: controllo della linea Disk Ready, controllo del posizionamento della testina sulla traccia 00, controllo della tacca di protezione, controllo sui cambi di disco.

A questo punto ci rimangono i bit 0 ed 1. Il bit 0 è il Memory Overlay Bit, il bit 1 è quello che comanda la luminosità del led. Se siete proprio interessati al suo funzionamento provate la seguente linea in Assembler:

```
EORI.B #$02,$BFE001
```

Se ci spostiamo sulla tabella 4 troviamo le connessioni del CIAB. La porta CIAB/PA è dedicata totalmente alle linee di controllo della porta seriale, mentre la CIAB/PB contiene le altre linee di controllo del drive e più precisamente:

```

BFE001 /FIR1 /FIRO /RDY /TKO /WPRO /CHNG /LED /OVL
BFE101 Porta parallela
BFE201 DDR porta A (BFE001)
BFE301 DDR porta B (BFE101)
BFE401 CIAA Timer A byte basso
BFE501 CIAA Timer A byte alto
BFE601 CIAA Timer B byte basso
BFE701 CIAA Timer B byte alto
BFE801 60 Hz Event Counter LSB bits 7-0
BFE901 60 Hz Event Counter bits 15-8
BFEA01 60 Hz Event Counter MSB bits 23-16
BFEB01 Non usato
BFEC01 CIAA Registro Dati Seriali (Tastiera)
BFED01 CIAA Registro di controllo Interrupts
BFEE01 CIAA Registro di controllo A
BFEF01 CIAA Registro di controllo B

```

Tabella 1 - Mappa degli indirizzi del CIAA.

```

BFD000 /DTR /RTS /CD /CTS /DSR /SEL /POUT /BUSY
BFD100 /MTR /SEL3 /SEL2 /SEL1 /SELO /SIDE /DIR /STEP
BFD200 DDR porta A (BFD000)
BFD300 DDR porta B (BFD100)
BFD400 CIAB Timer A byte basso
BFD500 CIAB Timer A byte alto
BFD600 CIAB Timer B byte basso
BFD700 CIAB Timer B byte alto
BFD800 Hor Sync Event Counter LSB bits 7-0
BFD900 Hor Sync Event Counter bits 15-8
BFDA00 Hor Sync Event Counter MSB bits 23-16
BFDB00 Non usato
BFDC00 CIAB Registro dati seriali
BFDD00 CIAB Registro di controllo Interrupts
BFDE00 CIAB Registro di controllo A
BFDF00 CIAB Registro di controllo B

```

Tabella 2 - Mappa degli indirizzi del CIAB.

controllo del motore, selezione dei drive 1-3, selezione della testina, selezione della direzione del movimento della testina, controllo del motore testina.

Prima abbiamo definito le porte parallele come «bidirezionali». Ciascuna linea può essere infatti programmata sia in uscita che in entrata. Tale selezione avviene settando appropriatamente un apposito registro.

Appare evidente come il controllo di ciascuna porta può avvenire semplicemente attraverso due locazioni: il registro a 8 bit DDR (acronimo di Data Direction Register) ed il registro a 8 bit PDR (acronimo di Peripheral Data Register).

Prima di effettuare qualunque operazione bisogna ovviamente definire le linee da utilizzare come uscite e quelle da utilizzare come entrate. Tutto ciò si può effettuare settando ad 1 nel DDR i bit corrispondenti alle linee desiderate come uscite. A questo punto siamo pronti per iniziare la trasmissione che avviene semplicemente scrivendo i dati nel PDR. Per quanto riguarda l'input

bisogna fare attenzione solamente al fatto che il PDR in lettura ritorna gli stati di ogni pin indipendentemente dal valore del DDR.

### Timer

All'inizio dell'articolo abbiamo detto che ogni 8520 contiene due timer e quindi, dato che nell'Amiga vi sono due 8520, con la difficile formula  $2 * 2 = 4$  deduciamo che in tutto abbiamo 4 timer. Nonostante quattro timer possano sembrare un po' eccessivi, leggendo la documentazione ci accorgiamo che il programmatore di applicazioni li trova praticamente inutilizzabili a meno che non ricorra al blocco di determinate operazioni. Il sistema operativo sfrutta infatti in questo modo i quattro timer.

Il CIAA/TA viene utilizzato per la gestione delle comunicazioni seriali, il CIAA/TB per la sincronizzazione tra Blitter e VideoBeam, il CIAB/TA per la gestione della tastiera ed infine il CIAB/TB per lo switching e gli interrupt in Exec.

Bisogna comunque fare attenzione a non confondersi con la parola timer. Infatti più che a quattro normali orologi ci troviamo di fronte a quattro contatori alla rovescia a sedici bit. Il decremento di una unità del contatore può avvenire ad ogni impulso proveniente (come si può vedere nelle tabelle 5 e 6) da:

- 1) 2 cicli di Clock
- 2) Transizioni positive sul piedino CNT
- 3) Termine del conteggio del timer A
- 4) Termine del conteggio del timer A con CNT alto.

La gestione di un timer avviene tramite tre registri, due a sedici bit ed uno di controllo ad otto bit. Uno dei due registri a sedici bit è chiamato timer Counter ed è in esso che deve essere inserito il valore iniziale del conteggio che verrà poi congelato nell'altro registro a sedici bit chiamato Timer Latch. La lettura del Timer Counter ritorna infatti il valore attuale del timer e non il valore di partenza. Il registro più importante per la gestione dei timer è quello di controllo ad otto bit. Esso consente di effettuare una grande quantità di ope-

```

PA7 DTR Data Terminal Ready
PA6 RTS Request to Send
PA5 CD Carrier Detect
PA4 CTS Clear to Send
PA3 DSR Data Set Ready
PA2 SEL Centronics Control
PA1 POUT Paper Out
PA0 BUSY Busy
SP BUSY Commodore
CNT POUT Commodore
PB7 MTR Disk Motor
PB5 SEL3 Selects external 3rd drive
PB5 SEL2 Selects external 2nd drive
PB4 SEL1 Selects external 1st drive
PB3 SELO Selects internal drive
PB2 SIDE Side select
PB1 DIR Head Direction
PB0 STEP Step
PC Not Used
F INDEX Disk Index

```

Tabella 3 - CIAA Assegnamento della porta.

```

PA7 Gameport 1 pin 6
PA6 Gameport 0 pin 6
PA5 RDY Disk Ready
PA4 TKO Disk Track 00
PA3 WPRO Disk Write Protected
PA2 CHNG Disk Change
PA1 LED Led Light (0 Bright, 1 Dim)
PA0 OVL Memory Overlay Bit
SP KDAT Keyboard Data
CNT KCLK
PB7 P7 Centronics parallel interface data
PB6 P6
PB5 P5
PB4 P4
PB3 P3
PB2 P2
PB1 P1
PB0 FO
PC DRDY Centronics control
F ACK

```

Tabella 4 - CIAB Assegnamento della porta.

razioni. Tramite esso si può far partire e fermare il timer, si può far apparire l'output sul piedino PB6 per il timer A e sul piedino PB7 per il timer B, si può definire se l'output debba presentarsi come una inversione di stato della linea (EOR) o come un semplice impulso, si può definire se il timer debba ripetere il conteggio all'infinito o effettuarlo una sola volta, ed infine inserire tramite un bit di controllo (Force Load) un nuovo valore nel Timer Counter.

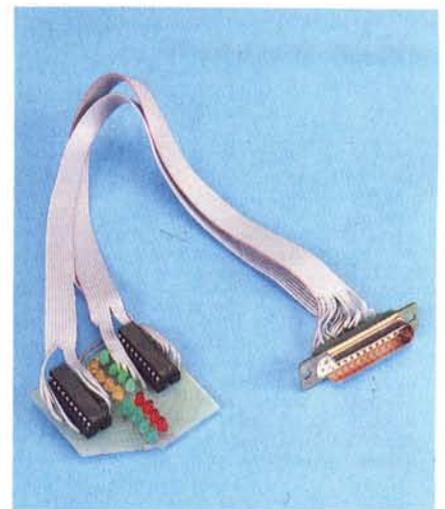
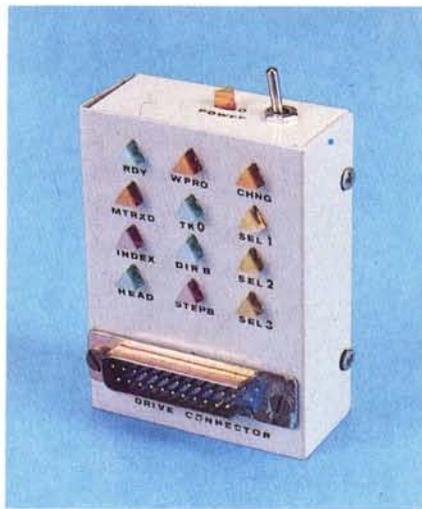
### TOD - Time Of Day

Il TOD non è altro che un contatore a 24 bit che viene incrementato ad ogni impulso che arriva sul piedino TOD. È anche possibile definire un orario di allarme che consente di generare un interrupt al raggiungimento di un determinato valore. Il TOD non è usato dall'Amiga e la presenza di determinate funzioni del sistema operativo consente al programmatore di evitare l'accesso diretto a questo componente dell'hardware.

### Porta seriale

La gestione della porta seriale dell'8520 (non dell'Amiga) è molto semplice. Basta distinguere i due casi di input ed output. Quando ci si trova in input ad ogni impulso che arriva sul piedino CNT il dato presente sul piedino SP viene shiftato in un apposito registro denominato (ma guarda un po'...) per l'appunto Shift Register. Dopo la ricezione di otto bit lo Shift Register viene copiato nello SDR (Serial Data Register) e viene generato un interrupt per indicare che il dato è pronto.

Per quanto riguarda l'output viene utilizzato il timer A per generare gli impulsi corrispondenti ai diversi Baud Rate ed i dati vengono estratti dallo SDR ed inviati sul piedino SP alla metà della velocità di underflow (termine del conteggio) del



A sinistra il nostro progettino montato e «inscatolato», a destra in versione «economica».

timer A. Anche in questo caso dopo la trasmissione di otto bit viene generato un interrupt per indicare che il sistema è disponibile per l'invio di altri dati.

Se vi siete mai chiesti perché sul cavo della tastiera vi sono proprio quattro linee, sappiate che due sono la massa e l'alimentazione, una è il KCLK e cioè il CNT, e l'altra è il KDAT e cioè l'SP.

### Interrupt

Abbiamo detto che gli 8520 hanno anche la capacità di generare degli interrupt, e più precisamente cinque differenti tipi di interrupt:

- 1) Termine del conteggio del timer A
- 2) Termine del conteggio del timer B
- 3) Allarme del TOD
- 4) Serial Port piena o vuota
- 5) Piedino F.

Due registri ad otto bit consentono di gestire gli interrupt: si tratta del Mask Register e del Data Register.

Quando si verifica un interrupt basta leggere il Data Register (che viene azzerato ad ogni lettura) per individuare il tipo di interrupt che è stato generato. Il Mask Register (vedi tabella 7) serve

invece ad abilitare o disabilitare determinati interrupt. Dato che il registro è ad otto bit, che gli interrupt sono 5 e che sarebbe molto utile poter abilitare o disabilitare un dato interrupt senza influenzare gli altri, i progettisti dell'8520 (che si sono rivelati persone serie) hanno avuto la brillante idea di utilizzare il settimo bit del registro di mascheramento come bit di controllo. Il tutto avviene in questo modo: bisogna decidere quali bit della maschera si vogliono accendere (o spegnere) e metterli ad uno, e poi mettere ad uno (o zero) il settimo bit. I bit posti a zero non verranno influenzati dall'operazione. Se per esempio volessimo accendere i bit 0 e 2 dovremmo utilizzare il valore 10000101, mentre se volessimo spegnere sempre i bit 0 e 2 dovremmo utilizzare il valore 00000101.

### Individuazione degli indirizzi

Volendo calcolare l'indirizzo di uno qualunque dei registri degli 8520 si possono seguire due strade: la prima è quella di utilizzare le tabelle 1 e 2, la seconda è quella che ora spiegheremo.

Il CIAA si trova alla locazione binaria

Bit	Funzione
0	Timer A 1 Start 0 Stop
1	Timer A 1 Uscita su PB6 0 PB6 normale
2	Timer A 1 Uscita a cambiamento di stato 0 Uscita a impulso
3	Timer A 1 Conteggio unico 0 Conteggio ripetuto
4	Force Load
5	Timer A 1 Uso di CNT come impulso 0 Uso di 2 cicli di Clock
6	Serial 1 Output con CNT come Clock 0 Input con clock esterno richiesto
7	0 60 Hz 1 50 Hz

Tabella 5 - Registro di controllo A.

Bit	Funzione
0	Timer B 1 Start 0 Stop
1	Timer B 1 Uscita su PB7 0 PB7 normale
2	Timer B 1 Uscita a cambiamento di stato 0 Uscita a impulso
3	Timer B 1 Conteggio unico 0 Conteggio ripetuto
4	Force Load
6,5	Selezione impulsi del Timer B 0,0 Uso di 2 cicli di clock 0,1 Uso di CNT come impulso 1,0 Uso degli underflows del Timer A 1,1 Uso degli underflows del Timer A con CNT alto
7	TOD 1 Scrivendo al TOD si setta l'allarme 0 Scrivendo al TOD si setta il TOD

Tabella 6 - Registro di controllo B.

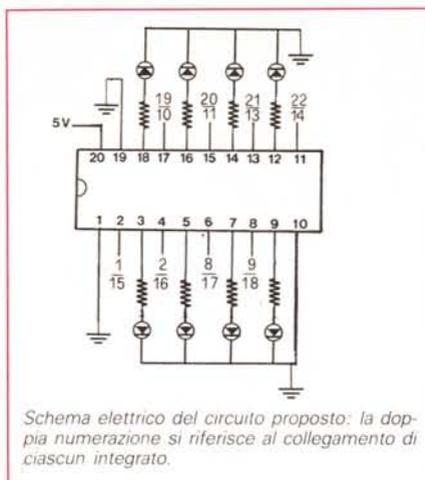
101x xxxx xx01 rrrr xxxx xxx0 (BFer01 in esadecimale) ed il CIAB alla locazione binaria 101x xxxx xx10 rrrr xxxx xxx1 (BFDr00 in esadecimale). Sapendo che lo stato dei bit x non ha importanza e considerando la tabella 8 per i bit r, il calcolo di un indirizzo si svolge molto velocemente. Per esempio, volendo calcolare la locazione di controllo del led basta seguire questo ragionamento: il bit che controlla il led si trova nel PRA del CIAA. Sappiamo che l'indirizzo del CIAA è BFER01. Allora andiamo a guardare la tabella 8 e vediamo che il PRA corrisponde a 0000 (bin) e 0 (hex). A questo punto sostituiamo questo valore a BFER01 ed otteniamo la locazione cercata BFE001.

### Il circuitino di questo mese

Ora che abbiamo le idee più chiare sul funzionamento degli 8520, possiamo iniziare a parlare del nostro progetto hardware attraverso il quale cercheremo di far conoscere ai lettori come funzionano le interfacce utilizzate dall'Amiga per gestire gli I/O del sistema. Raggiungeremo questo scopo con la costruzione di una semplice interfaccia in grado di visualizzare lo stato delle linee di controllo del drive. I componenti che più risaltano nello schema dell'interfaccia sono due integrati. Essi appartengono alla stessa famiglia dei TTL e riportano la sigla SN74LS240. Anche se questa sigla può risultare a prima vista incomprensibile, non vi è il bisogno di scoraggiarsi anche perché con un po' di impegno è molto semplice riuscire a sapere con quale strano «attrezzo» si sta avendo a che fare. Andando infatti a ricercare questa strana sigla su un Data Book dei componenti della famiglia TTL, troveremo subito la dicitura «Unidirectional Octal buffer or line driver», il che vale a dire che i chip di cui stiamo parlando svolgono la funzione di buffer unidirezionali ad 8 bit e sono in grado di pilotare direttamente con le loro uscite una linea di dati oppure un display (come nel nostro caso).

Vi è un particolare motivo per il quale è stato necessario ricorrere all'utilizzo di tali chip. La ragione è che quando si lavora con un qualsiasi strumento elettronico, e specialmente se poi si è alle prime armi, è molto facile danneggiarlo. I due 74LS240 hanno infatti il compito di separare i segnali di uscita del computer da qualsiasi altra periferica, in modo tale da non far gravare quest'ultima direttamente sull'interfaccia utilizzata dall'Amiga per comunicare con il mondo esterno (la famigerata 8520).

I nostri due chip 74LS240 hanno però una caratteristica molto importante,



quella di «negare» tutto ciò che gli viene detto. Spiegandosi meglio, ed è possibile riscontrarlo dalle tabelle pubblicate in queste pagine, se noi tentassimo di inserire in uno degli otto ingressi del 74LS240 un dato relativo ad uno zero all'uscita troveremmo, al contrario di quanto aspettato, invece di uno 0 logico (0 volt) un valore logico 1 (5 volt), e questo per tutti gli ingressi del 74LS240.

Infatti tutti i valori inseriti, vengono presentati all'uscita invertiti di stato logico.

Abbiamo deciso di utilizzare tali chip in quanto l'Amiga mette a disposizione sulle uscite del connettore del drive segnali con valori logici «normalmente» alti. Ciò significa che, anche se il drive non è attivo, tali segnali rimangono allo stato alto commutando allo stato basso solo quando il drive riprende a funzionare.

Se invece di tali chip (i 74LS240) avessimo usato dei normali buffer non invertenti (74LS244) i diodi led sarebbero rimasti sempre accesi e ciò, oltre ad essere antiestetico, avrebbe rappresentato un carico di corrente non indifferente per il computer.

Arrivati a questo punto è piuttosto semplice capire il funzionamento della scheda: ciascun dato proveniente dal

connettore a ventitré pin del drive viene inviato al rispettivo ingresso di uno dei due 74LS240. Le uscite, a loro volta, non faranno altro che andare a pilotare il led di monitoraggio della scheda che però non possono funzionare direttamente a 5 volt. Infatti se così fosse il diodo, pur emettendo una luce forte, non durerebbe molto a lungo. Ciò è dovuto al fatto che anche i diodi led sono dei semiconduttori costituiti da elementi di silicio, la cui tensione nominale di conduzione si aggira attorno a 0.5-0.6 volt. Sarebbero quindi necessari solo 1.5-2 volt per ottenere un'illuminazione sufficiente senza correre il rischio di bruciare qualche diodo e, per fare in modo che ciò avvenga, è necessario inserire tra il catodo comune dei diodi e la massa una resistenza del valore di 470 ohm.

In tal modo la corrente che circola nei diodi è molto più bassa e non è più in grado di bruciarli, ovviando agli inconvenienti sopra accennati.

È importante anche ricordare che i diodi hanno una polarità ben precisa, il che vale a dire che per accenderli, è necessaria una tensione positiva sull'anodo (A) (il terminale più lungo), rivolgendo il catodo (K) verso la massa.

Per quanto riguarda il montaggio degli integrati, è consigliabile l'impiego di zoccoli onde evitare possibili danneggiamenti dovuti al surriscaldamento dei

Bit	Funzione
0	Fine conteggio Timer A
1	Fine conteggio Timer B
2	Allarme
3	Serial Port
4	Flag
5	Non usato
6	Non usato
7	Set/Clear Bit

Tabella 7 - Registro di mascheramento dell'interrupt.

r3	r2	r1	r0	HEX	Nome Registro
0	0	0	0	0	PR Registro Dati Porta A
0	0	0	1	1	PR Registro Dati Porta B
0	0	1	0	2	DDRA Registro direzione porta A
0	0	1	1	3	DDRB Registro direzione porta B
0	1	0	0	4	TALO Timer A registro basso
0	1	0	1	5	TAHI Timer A registro alto
0	1	1	0	6	TBLO Timer B registro basso
0	1	1	1	7	TBHI Timer B registro alto
1	0	0	0	8	LSB dell'Event Counter
1	0	0	1	9	Bits 8-15 dell'Event Counter
1	0	1	0	A	MSB dell'Event Counter
1	0	1	1	B	Nessuna connessione
1	1	0	0	C	SDR Registro dati seriali
1	1	0	1	D	ICR Registro controllo interrupts
1	1	1	0	E	CRA Registro di controllo A
1	1	1	1	F	CRB Registro di controllo B

Tabella 8 - Calcolo degli indirizzi.

chip durante la saldatura. Tali zoccoli andranno montati con la tacca di riferimento nello stesso verso di quella degli integrati, non perché questo possa in qualche modo compromettere la funzionalità del circuito, ma perché può provocare errori nel montaggio causando l'involontaria inversione dei chip.

La conseguenza di tale operazione è facilmente intuibile (ka-boom!!!).

Infatti bisogna prestare molta attenzione al fatto che tutti i circuiti integrati hanno un preciso verso di montaggio.

Tale verso è indicato da una tacca praticata sul contenitore, e, in corrispondenza di tale tacca, da un punto posto sulla sinistra che indica il pin numero uno.

Non rimane ora che effettuare le connessioni tra il connettore a 23 poli maschio con le relative piazzole di contatto della scheda e controllare, servendosi della tabella 9, che non vi siano errori nel montaggio della scheda o nel disegno del circuito stampato.

Se tutto è a posto si può collegare il DB23 della scheda sul relativo DB23 posto sul retro dell'Amiga (per chi non possiede drive esterni), oppure sul DB23 di derivazione posto su ogni drive. Ovviamente è meglio effettuare tale operazione a computer spento.

Se sorgono difficoltà nel reperimento dello spinotto a 23 poli, è sempre possibile utilizzarne uno a 25 (DB25) avendo l'accortezza di tagliare gli ultimi due piedini (il 25 ed il 13) e parte della lamiera di contorno.

A questo punto accendete l'Amiga. Se non udite rumori, quali esplosioni e

1	RDY	Disk Ready
2	DKRD	Disk Data Ready
3	GND	Ground
4	GND	
5	GND	
6	GND	
7	GND	
8	MTRXD	Disk Motor
9	SEL2B	Select 2nd Drive
10	DRESB	Disk Reset
11	CHNG	Disk Change
12	+5	+5 volt
13	SIDEB	Disk Side (1 Inf/0 Sup)
14	WPRO	Disk Write Protected
15	TKO	Disk Track 00
16	DKWEB	Disk Write Enable
17	DKWDB	Disk Write Data
18	STEPB	Stepper motor
19	DIRB	Direction (0 Int/1 Ext)
20	SEL3B	Select 3rd Drive
21	SEL1B	Select 1st Drive
22	INDEX	Disk Index Pulse
23	+12	+12 volt

Tabella 9 - Connessioni del DB23 del drive.

scricchiolii strani, potete essere certi che il vostro circuito è in grado di funzionare. Non rimane a questo punto che caricare un programma qualsiasi e... buon divertimento!

### Utilizzazioni pratiche (e teoriche) della scheda

Una volta montata la nostra bella scheda, pomposamente chiamata Disk Monitor, verremo presi da un dubbio atroce: abbiamo costruito qualcosa di veramente utile oppure solamente un bel gingillo adatto solo a fare scena?

La risposta giusta è ovviamente la seconda... (scherzetto!) ...la prima (altrimenti ci saremmo trovati nello stesso caso di quello strano essere che inventò lo scaricabatterie), in quanto il fatto di

poter vedere come si comportano le varie linee adibite al controllo del drive, mentre quest'ultimo è in funzione, è molto importante. Chi si è chiesto il motivo per cui programmi quali il FastLightning della VesaliaSoft, i caricatori di Barbarian e Obliterator, siano così veloci può rendersi conto di come la gestione del drive sia effettuata in modo totalmente diverso da quello standard semplicemente osservando i led del DiskMonitor. Per esempio si può vedere subito che nel FastLightning le testine dei due drive vengono spostate simultaneamente (con lo stesso comando) con grande risparmio di tempo rispetto agli spostamenti tradizionali indipendenti che comportano un numero doppio di istruzioni.

Non vi rimane che sperimentare tutto lo sperimentabile e attendere che esca il prossimo MC.

### Applicazioni (!)

Il tema principale di questo articolo era quello di illustrare come l'Amiga comandi le sue periferiche. Chi ha letto tutto l'articolo avrà potuto vedere come gran parte delle linee delle 8520 siano dedicate alla gestione del drive. Siamo in grado, come applicazione, di controllare il drive come più ci piace fino addirittura a controllarlo per farlo... SUONARE!!!

È proprio vero, infatti il programma del listato 1, scritto in Assembler per motivi di velocità, svolge proprio questa funzione. Il principio di funzionamento è molto semplice: se noi muoviamo la testina del drive avanti e indietro, con spostamenti piccolissimi, ma molto veloci, possiamo pensare ad essa come ad una corda in vibrazione che emetterà quindi un suono.

Tuttavia data la velocità del 68000 è stato necessario addirittura inserire nel programma dei cicli di ritardo per rendere il suono udibile. La presenza della chiamata alla funzione Disable() di Exec si è resa necessaria per dare al tutto, disabilitando gli interrupt, una certa costanza e fluidità. Le varie istruzioni move.b verso la locazione \$BFD100, rintracciabile nella tabella 2, non fanno altro che pilotare le linee SIDE, STEP e DIR della porta del floppy. Ovviamente potete vedere l'effetto di questo programma con il Disk Monitor che avete appena montato.

Il programma non fa altro che emettere un fischio (avvicinatevi al drive per sentirlo), la cui frequenza cala costantemente. Per uscire basta premere il tasto sinistro del mouse.

Buon ascolto!!!

```

        move.l 4,a6          ;Legge execbase
        jsr  -120(a6)       ;Chiamata a Disable()

        move.l #2,d3        ;Inizializzazione contatori
        move.l d3,d4
loop:   move.b #112,$bfd100 ;Muove testina 0
        move.b #113,$bfd100 ;verso l'interno
        move.b #116,$bfd100 ;Muove testina 1
        move.b #117,$bfd100 ;verso l'interno
att:    move.l d4,d2        ;Ritardo
        subi.l #1,d2
        bne att
        move.b #114,$bfd100 ;Muove testina 0
        move.b #115,$bfd100 ;verso l'esterno
        move.b #118,$bfd100 ;Muove testina 1
        move.b #119,$bfd100 ;verso l'interno
        move.l d4,d2        ;Ritardo
label:  subi.l #1,d2
        bne label
        add.l #1,d3
        cmp.l #200,d3
        bne skip
        clr.l d3
        add.l #1,d4
skip:   andi.b #$40,$bfe001 ;Mouse premuto ?
        bne loop           ;No, allora riinizia

exit:   jsr  -126(a6)       ;Chiamata a Enable()
        rts                ;Uscita
END

```

Listato 1 - Suoniamo con il drive.

# DI QUALE PARTNER HAI BISOGNO?

## **PARTNER** **Trade**

Una azienda al servizio di grossi consumatori di prodotti Ms Dos. Un tramite per importare dal mondo intero senza problemi restando comodi in ufficio!

## **PARTNER** **H.T. Diffusion**

Una completa gamma di Personal Computer Ms Dos, periferiche e add on in consegna pronta, a prezzi e condizioni commerciali eccezionali! Acquisti spot, mandati di concessione e/o di distribuzione, rapporti d'agenzia.

## **PARTNER** **Assistance**

24 ore per assistere in maniera completa il vostro hardware in ambiente Ms Dos. Garanzia su tutte le riparazioni. Assistenza sul posto per utenti finali. Condizioni speciali per i rivenditori.

## Scegli pure la tua Partner e telefonaci!

# Partner

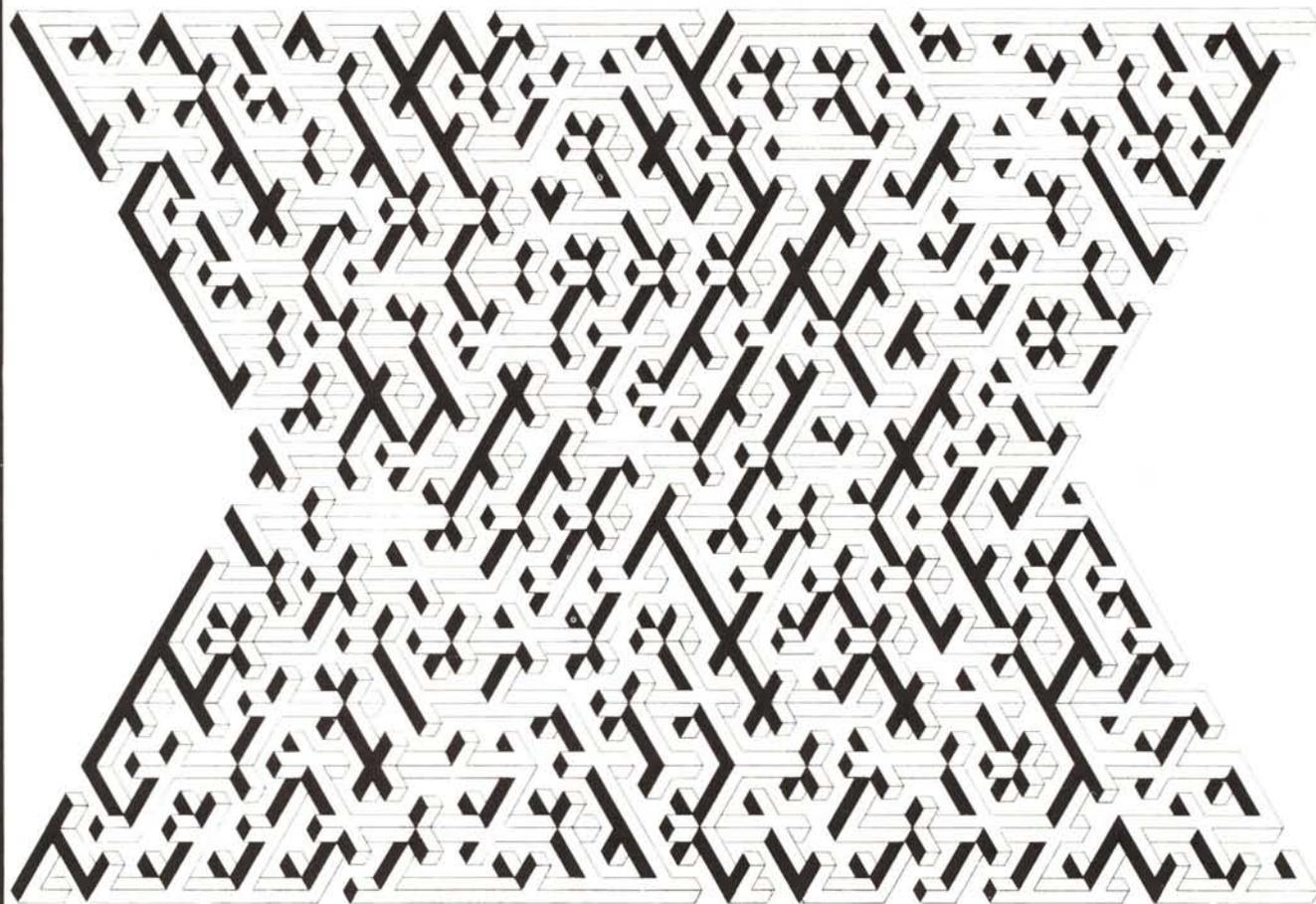
00144 Roma - Viale Cesare Pavese 410  
Tel.06.5003136 ra Fax.06.5002383 Tlx.610366

Società del Gruppo Panaviation

Partner importa e distribuisce

Friendly Murata Nec Star Samsung Philips

# ESCI DAL LABIRINTO CON LA SOLUZIONE GIUSTA



6° Salone  
dell'Office Automation,  
Informatica  
e Servizi per le Aziende

**Firenze 25 - 28 Novembre 1988 Fortezza da Basso**



REGIONE TOSCANA

Patrocinio



COMUNE DI FIRENZE

Organizzazione **UNICEST** Via Rossellino, 9/B - 50047 PRATO (FI) Tel. (0574) 596861/2