

## Più memoria per l'Amiga con la Janus... ma non solo

di Giuliano C. Peritore - Latina

Ma come, la Janus non era solo l'emulatore MS-DOS per l'Amiga? Ecco uno dei primi esempi di come può essere sfruttata la BridgeBoard a vantaggio dell'Amiga quando non si ha bisogno di utilizzare il PC. Ma non ci fermeremo qui. Vedremo anche quali sono gli altri trucchetti per recuperare qua e là un po' di memoria che fa spesso molto comodo quando si usano quegli innumerevoli programmi che mangiano quantità enormi di byte ma a cui non possiamo rinunciare (DPaint, Caligari, ecc. ecc.).

Ancora non è finita, vedremo anche che con la Janus possiamo aggiungere ad un Amiga con sola Chip Ram un pochettino di Fast Ram.

### Premessa

Prima di cominciare a parlare di tutte le tecniche descritte in questo articolo è doveroso specificare la configurazione

hardware con la quale ho realizzato il tutto. Infatti i dati numerici e le indicazioni tecniche che riporterò saranno quasi sicuramente differenti su macchine differenti. Vi consiglio quindi se non siete particolarmente esperti di pensare due volte prima di mettere mano al saldatore e invece di provare e riprovare per quanto riguarda il software. La configurazione raccomandata è la seguente:

#### Computer:

Amiga B2000 Rev. 6 con Fat Agnus (1MB Chip Ram) + DF1:

#### Processore:

MC68010

#### Bridgeboard:

Janus A2088 XT

#### DOS:

MS-DOS 3.30

#### Handler:

Janus Handler 2.63

#### Libreria:

Janus Library 33.1

#### Controller:

GVP SCSI+2 OK RAM

#### Hard Disk:

Seagate 80MB 19 MS diviso in due partizioni da 40MB

ga in quanto la gestione di grafica e suono implica la necessità di avere parecchi Kbyte sia di Ram che di memoria di massa. Talvolta accade inoltre di essere veramente ai limiti per quanto riguarda la Ram tanto che spesso un risparmio di pochi Kbyte può essere vitale per il completamento di alcune operazioni. Basta infatti pensare a rendering, compilazioni, elaborazioni varie per rendersi conto di quante volte abbiamo rimpianto una bella (e purtroppo costosa) espansione.

In questo articolo vedremo tre modi di recuperare memoria. Il primo, via software, il secondo via hardware, il terzo via... BridgeBoard.

### Memoria = software

Il recupero della memoria via software è usato da parecchi anni e consiste generalmente nel sottrarre un bitplane allo schermo del Workbench la cui memoria verrà aggiunta alla lista della Chip Ram libera. Ciò consente un risparmio di circa 21 Kbyte e può essere fatto sia con il programma «Add21K» che con il programma «OneColor».

Vi è poi anche un altro problema che affligge i possessori del Fat Agnus. Le vecchie versioni del DeluxePaint avevano un «bug» che impediva di creare animazioni se non si possedeva la Fast Ram. Le ultime versioni sono state corrette ma comunque se avete tali problemi basta che vi procurate il programma «KillChip» presente sul disco dell'A-MAX. Questo programma che deve essere chiamato all'inizio della startup-sequence spezza in due il megabyte di Chip Ram facendo credere al sistema di avere 512 Kbyte di Chip Ram e 512 Kbyte di Fast Ram.

Il problema è quindi risolto per quei programmi «legali» che usano la funzione AllocMem ma spesso vi sono dei programmi «illegali» che accedono direttamente a \$C00000 e purtroppo per essi non c'è nulla da fare.

### Memoria = hardware

Un Amiga con un Megabyte avrà 1024 Kbyte liberi. Tuttavia vediamo che un Amiga NTSC con un drive e senza

### Perché questo articolo

La memoria è uno dei problemi che più spesso assillano i possessori di Ami-

Numero	Scheda	Funzione
J102	AMIGA	Commutazione PAL/NTSC
J301	AMIGA	Attivazione DF1:
J4	GVP+2	Abilitazione del Boot da HD

◀ Figura 1 - Elenco dei jumper.

Figura 2 - La memoria dual port dalla parte del PC.

```

C>debug
-ds000:0000 0080
000:0000 55 A8 10 EB 72 90 20 4A-61 6E 75 73 20 48 61 6E U...r. Janus Han
mmmmmmmm 64 8C 65 72 20 62 79 20-54 6F 72 73 74 65 6E 20 dler by Torsten
000:0010 42 75 72 67 64 6F 72 46-66 20 00 20 43 6F 70 79 Burgdorff, Copy
mmmmmmmm 72 69 67 6B 74 20 28 63-29 20 31 39 38 38 2C 20 right (c) 1988,
000:0020 43 6F 6D 6D 6F 64 6F 72-65 20 41 6D 69 67 61 20 Commodore Amiga
mmmmmmmm 49 6E 63 2E 0A 0D 20 20-20 20 20 20 20 20 41 Inc... A
000:0030 6C 6C 20 72 69 67 6B 74-73 20 72 65 73 65 72 76 ll rights reserv
mmmmmmmm 65 64 2E 20 20 20 00 FA-06 1E 56 57 55 52 51 53 ed. ....VWURO5
000:0040 50

```

HD restituisce (con il comando Avail) 927 Kbyte. Ciò significa che il sistema operativo ha bisogno 96 Kbyte per girare. Sono inclusi lo schermo del CLI, le strutture del DOS, di Intuition, la tavola dei vettori del 68010 e tutte le altre strutture di sistema. Fino a qui non c'è nulla da fare.

Vediamo adesso la situazione con un Amiga 1MB però PAL, con DF1: e Hard Disk diviso in due partizioni. Il numero delle partizioni è importante in quanto occupano memoria). Il comando Avail (sempre come primo comando eseguito) restituisce un totale di circa 759 Kbyte.

```
>m200000
00200000: 55AA10EB 7290204A 616E7573 2048616E 'Uä.ër. Janus Han
00200010: 646C8572 20627920 546F7273 74656E20 'dler by Torsten
00200020: 42757267 646F7266 66200020 436F7079 'Burgdorff . Copy
00200030: 72696768 74202863 29203139 38382C20 'right (c) 1988,
00200040: 436F6D6D 6F646F72 6520416D 69676120 'Commodore Amiga
00200050: 496E632E 0A0D2020 20202020 20202041 'Inc... A
00200060: 6C6C2072 69676874 73207265 73657276 'll rights reserv
00200070: 65642E20 202000FA 061E5657 55525153 'ed. .ü...VWRUSQ
00200080: 50 'p

>m220000
00220000: AA55EB10 90724A20 6E617375 48206E61 'äUë..rJ nasuH na
00220010: 6C647265 62202079 6F547372 6574206E 'ldreb yoTsret n
00220020: 75426772 6F646672 20662000 6F437970 'uBgrodfr f .oCyp
00220030: 69726867 20746328 20293931 3838202C 'irhg tcl 19188
00220040: 6F436D6D 646F726F 20656D41 67692061 'oCmmdoro emAgi ä
00220050: 6E492E63 0D0A2020 20202020 20204120 'niC... A
00220060: 6C6C7220 67697468 20736572 65737672 'llr gith seresvr
00220070: 6465202E 2020FA00 1E065756 52555351 'de . .ü...VWRUSQ
00220080: 8C '.
```

Figura 3 - La memoria dual port dalla parte dell'Amiga.

solo conseguenze negative. Infine se non volete forare il contenitore dell'Amiga un buon posto per installare gli interruttori sono i fori sullo chassis (lato anteriore) in modo da fare sbucare le levette dai fori longitudinali di areazione che si trovano alla sinistra del connettore della tastiera.

Buoni 928K.

### Memoria = BridgeBoard

Sarà la mancanza di documentazione, sarà il desiderio di smanettare ma da quando ho acquistato la BridgeBoard la sto davvero tartassando specialmente per quanto riguarda la comunicazione fra il mondo MS-DOS e il mondo di AmigaDOS. Sul manuale del-

C'è quindi una differenza di 168 Kbyte fra noi e i nostri colleghi statunitensi senza HD e senza DF1: Dov'è finito tutto questo ben di Dio? Vediamo: l'Hard Disk si è succhiato circa 137 Kbyte fra buffer, dati delle partizioni ecc. ecc. Lo schermo PAL e relative strutture si mangiano circa 10 Kbyte e le strutture (e relativi buffer) per il DF1: si ingoiano i rimanenti 21 Kbyte. Non c'è male. Vediamo i rimedi: brutalmente potremmo decidere di proseguire nel seguente modo:

1. Staccare la scheda dell'Hard Disk
2. Staccare il DF1:
3. Lanciare il programma TV-Boot (vedi MC99 Settembre 1990).

Ovviamente tutto ciò è assurdo. Per fortuna c'è un altro modo di procedere molto ma molto più comodo. Basta andarsi a comprare tre interruttori da attaccare su alcuni jumper. Vediamo quali sono (vedi figura 1).

Per abilitare e disabilitare il floppy abbiamo bisogno del jumper J301 che si trova sulle piastre dell'Amiga vicino al connettore per il floppy interno dal lato del connettore di alimentazione.

Per selezionare il modo NTSC o PAL ci serviremo del jumper J102 che si trova subito al di sopra del 68010 vicino ad un quarzo.

Infine per abilitare/disabilitare il boot dell'Hard Disk basta attaccarsi al jumper J4 situato sulla scheda GVP fra le EPROM di autoboot e alcune PAL. Mi raccomando di fare molta attenzione col saldatore e di utilizzare gli interruttori solo a computer spento. Usarli in altri momenti non avrebbe senso ma

LOWER	UPPER	FREE	ATTR	PRI	HUNK	NAME
200020	220000	49536	5	10	JanusMem	
0008e8	0fe800	533336	3	-10	Chip Memory	
Total:		582872				

Figura 4  
Lista dei blocchi di memoria.

```
.....
Mount PIPE: ;Prepara il device logico PIPE:
BindDrivers ;Connette le schede logicamente
run >nil: sys:pc/pcdisk ;Fa partire il ponte fra i DOS
type PIPE:Done ;Aspetta il file Done e lo stampa
AddJanusMem ;Aggiunge all'Amiga i 128K della Janus
.....
```

Figura 5 - La startup-sequence dell'Amiga.

```
ÄWRITE C:DONE.TXT PIPE:Done ;Spedisce il file Done all'Amiga
```

Figura 6 - L'AUTOEXEC.BAT del PC.

```
Janus Boot Done - Adding 128 KByte to Amiga's memory
```

Figura 7 - Il file DONE.TXT (PC).



la Bridge Board viene riportato che il ponte è effettuato tramite 128 Kbyte di Ram dual ported. Ciò significa che tale Ram è accessibile sia dal lato Amiga che dal lato PC. Il primo passo consiste quindi nel ricercare dove si trova tale memoria. Per quanto riguarda il PC dal manuale si ritrova facilmente che il segmento che ci interessa è il segmento E000. Infatti come si vede nella figura 2 se con il comando debug andiamo a leggere il segmento E000 vediamo, la stringa «Janus Handler by Torsten Burgdorff...» che lascia tutta supporre che sia qualcosa che riguarda l'Amiga. Se andiamo poi a cercare nella memoria dell'Amiga ritroviamo la stessa cosa in \$200000 (fig. 3). Fate attenzione perché se avete una espansione questa zona può essere dilocata altrove. Per esempio, nel caso abbiate 2 megabyte di fast Ram troverete il banco che ci interessa a \$400000. Ritourneremo co-

PC Tools Deluxe R4.21

```

-----System Information Service-----

The BIOS programs are dated - 02/29/88
Operating system - DOS 3.30
Number of logical disk drives - 7
Logical drive letter range - A thru G
Serial Ports - 0
Parallel Ports - 1
CPU Type - 8088
Relative speed (orig PC=100%) - 100%
Math co-processor present - No
User programs are loaded at HEX paragraph - 117E
Memory used by DOS and resident programs - 71648 bytes
Memory available for user programs - 452440 bytes
Total memory reported by DOS - 512K
PC Tools has found the total memory to be - 512K
Monochrome Display Adapter present
Color Graphics Adapter present
Additional ROM BIOS found at HEX paragraph - E000K

```

Figura 8 - La schermata Info dei PCTools.

munque sugli indirizzi più avanti. È importante notare che in \$220000 ritroviamo le stesse cose con i byte alti e bassi scambiati. La posizione dei byte alti e bassi è infatti la principale fonte di problemi di compatibilità fra Motoro-

la e Intel. Ecco quindi un buon modo di velocizzare l'interfacciamento. Se adesso andiamo a scrivere sia dalla parte del PC che dalla parte dell'Amiga vediamo che tale zona di memoria si comporta come normale Ram con l'unica differenza di essere condivisa. Praticamente possiamo leggere dall'Amiga quello che scriviamo con il PC e viceversa. Se però ci dimentichiamo del PC possiamo usare dalla parte dell'Amiga tale memoria come normale memoria Ram. Ovviamente se vogliamo utilizzare questi 128 Kbyte come espansione di memoria dobbiamo sia aggiungerla alla lista di sistema della memoria che fare in modo che il PC non vada a scriverci sopra. La prima idea che viene in mente è quella di aggiungere la memoria alla lista senza attivare (con il comando BindDrivers) la Janus. Ciò presenta due problemi:

- 1) impedirebbe di fare il BindDrivers per altre schede a meno che il file Janus.library non venisse tolto dalla directory Expansion.
- 2) Se la BridgeBoard non venisse fatta partire ogni tanto accedrebbe alla memoria Ram condivisa.

Bisogna quindi fare in modo di aggiungere la memoria alla lista dell'Amiga dopo che il PC ha fatto il Boot. Non possiamo inoltre attivare le PCWindow in quanto la memoria dual ported serve a noi.

Resterebbe quindi da aspettare un tempo «ad occhio» e lanciare il comando AddJanusMem (vedi listato 1). Ma questa non è una soluzione accettabile e pulita. Possiamo invece far partire PCDisk dopo il BindDrivers, effettuare su Amiga il comando Type Pipe: Done e inserire alla fine della Autoexec.Bat del PC il comando Awrite C:Done.TXT Pipe:Done. Il file Done.TXT conterrà un messaggio del tipo:

«Janus Boot Done - Adding 128 K to Amiga system memory»

Tipo Memoria	Numero di iterazioni	Tempo	Fattori	
Chip RAM	6.000.000 circa	10 sec	1.0	0.7
Janus RAM	6.000.000 circa	14 sec	1.4	1.0

Figura 10  
BenchMark  
su Amiga Chip Ram.

```

BenchMark c 1988 Lake Forest Logic
Free chip: 558K Free fast: 127K
Total free: 685K Total RAM: 1143K

Devices:
PIPE: DF0: PRT: PAR: SER: RAW:
CON: RAM: DH0:

Directories:
ROLLDOWN: quad: include: lib: lc:
S: L: C: FONTS: DEVS: LIBS:
SYS:

Volumes:
RAM DISK: DH1(40MB): DH0(40MB):

CPU Elapsed time: 2.8 seconds
CPU Performance: 1.1

```

Figura 9  
Differenze di velocità  
fra le due memorie.

```

BenchMark c 1988 Lake Forest Logic
Free chip: 590K Free fast: 99K
Total free: 689K Total RAM: 1143K

Devices:
PIPE: DF0: PRT: PAR: SER: RAW:
CON: RAM: DH0:

Directories:
ROLLDOWN: quad: include: lib: lc:
S: L: C: FONTS: DEVS: LIBS:
SYS:

Volumes:
RAM DISK: DH1(40MB): DH0(40MB):

CPU Elapsed time: 4.00 seconds
CPU Performance: 0.8

```

Figura 11  
BenchMark  
su Janus Ram.



che apparirà sulla finestra CLI attiva dell'Amiga. Ovviamente dal lato Amiga ci sarà una pausa fino a quando non arriverà il file Done che sarà spedito dal PC alla fine del Boot. A questo punto il PC è fermo con il prompt C> del DOS e rimarrà in attesa all'infinito perché noi non faremo partire PCWindow mentre la Ram sarà disponibile per il comando AddJanusMem che vedremo fra poco. La situazione che avremo alla fine sarà un Amiga con:

1024 Kbyte di Chip Memory  
128 Kbyte di Fast Memory.

Ci troveremo nella situazione della figura 4 che rappresenta la lista dei blocchi di memoria disponibili (ottenuta con lo Xoper). Se avete comunque dei problemi a cercare la zona in cui si trova la memoria dual ported della Janus oppure se volete essere soltanto sicuri dei vostri conti potete far partire il programma del listato 4 (FindJanusMem) che non fa altro che tirare fuori dalla struttura JanusBase la locazione iniziale del segmento di memoria che ci interessa. In questo modo sarete sicuri di non sbagliare assolutamente.

### Il programma AddJanusMem

Per effettuare il tutto avremo quindi bisogno di una startup-sequence e di una autoexec.bat apposite. Potete vederle in figura 5 e 6. Quello che rimane ancora da vedere è il comando AddJanusMem. Cominciamo! Come si vede al primo colpo d'occhio il programma AddJanusMem (listato 1) è brevissimo. La funzione è infatti solamente (!) quella di aggiungere alla lista di memoria dell'Amiga i 128 Kbyte della Janus. Il meccanismo è molto semplice. Basta preparare tutti i dati di cui abbiamo bisogno e chiamare la funzione AddMemList ed il gioco è fatto. Iniziamo con una chiamata a Forbid() che disabilita il MultiTasking. Questa operazione è necessaria in quanto altrimenti qualcuno potrebbe manipolare la lista contemporaneamente a noi ottenendo o al limite fornendoci una lista inconsistente con la conseguenza di poter provocare un possibile crash del sistema.

Successivamente allocheremo una zona di memoria adatta a contenere il nome del nuovo nodo. Ci va bene qualunque memoria per cui basta scegliere come attributi Public e Clear per ottenere una zona pulita.

Tale allocazione è necessaria perché se avessimo messo il nome nel programma con un dc.b "JanusMem", 0 e avessimo passato l'indirizzo con Lea

```

;AddJanusMem      Public Domain by Giuliano C. Peritore
;                Via Amaseno, 6
;                04100 Latina - ITALY

_LVDForbid        EQU     -132
_LVDPermit        EQU     -138
_LVDAAllocMem     EQU     -198
_LVDAAddMemList   EQU     -618

AddJanusMem:
    move.l 4,a6                ;ExecBase
    jsr   _LVDForbid(a6)       ;Disabilita il MultiTasking

    move.l #12,d0              ;Lunghezza del nome della memoria
    move.l #$10001,d1          ;CLEAR | PUBLIC
    jsr   _LVDAAllocMem(a6)    ;Alloca la memoria
    move.l #0,a1

    move.l #'Janus',0(a1)      ;Scrivo il nome della zona
    move.l #'sMem',4(a1)

    move.l #$20000,d0          ;Dimensione del blocco di memoria
    moveq #4+1,d1              ;Attributi: FAST | PUBLIC
    move.l #10,d2              ;Priorità
                                ;Mettere -15 se si vuole dare la
                                ;precedenza alla Chip Memory
    move.l #$200000,a0         ;Indirizzo iniziale della zona
    jsr   _LVDAAddMemList(a6) ;Aggiungi la nuova memoria

    jsr   _LVDPermit(a6)      ;Riabilita il MultiTasking

    moveq #0,d0                ;Pulisce il flag d'errore
    rts                        ;Esci

;Message from Amiga to Janus ---> "Thank You!"

```

Listato 1 - Il programma AddJanusMem.

JanusName,a1 avremmo passato l'indirizzo di una zona di memoria che non sarebbe rimasta allocata all'uscita del programma ma che sarebbe andata persa. Avremmo quindi perso anche il nome.

Una volta scritto il nome "JanusMem" bisogna passare ad AddMemList() i seguenti parametri:  
in A1 il puntatore alla stringa,  
in A0 l'indirizzo di inizio del nuovo banco di memoria,  
in D0 la lunghezza del banco  
in D1 i suoi attributi  
in D2 la sua priorità di allocazione.

A questo punto la memoria è già disponibile. Basterà quindi chiamare il Permit() corrispondente al precedente Forbid() per riabilitare il MultiTasking e mettersi in condizione di poter uscire e il gioco è fatto.

Per assemblare il programma basta il DevPac oppure direttamente un

GenLm2 AddJanusMem.5

```

00050000      move.l  #$0000005A,D1
00050006      moveq   #$FF,D0
00050008      dbf     D0,$00050008
0005000C      dbf     D1,$00050006
00050010      rts

```

Listato 2 - Ciclo di circa sei megaiterazioni in Chip Ram.

Si otterrà un eseguibile di soli 112 byte che andrà messo nella directory c: del disco di boot dell'Amiga.

Ovviamente nulla ci vieta di fondere insieme i programmi del listato 1 e del listato 4 per ottenere (a scapito delle dimensioni) un programma (JanusMem) che trova da solo il segmento di memoria in cui è allocata la Janus e lo aggiunge alla lista libera di memoria. È proprio quello che fa il programma del listato 5.

### Eventuali problemi

L'uso della memoria dual ported che abbiamo destinato all'Amiga potrebbe provocare dei problemi. Tale zona è infatti vista dal Bios del PC come parte di Rom Bios addizionale come si vede con il comando info dei PCTools (vedi figura 8). Inoltre viene anche usata come zona di comunicazione per la tastiera. Purtroppo la mancanza di documentazione non mi ha permesso di capire

```

00210000      move.l  #$0000005A,D1
00210006      moveq   #$FF,D0
00210008      dbf     D0,$00210008
0021000C      dbf     D1,$00210006
00210010      rts

```

Listato 3 - Ciclo di circa sei megaiterazioni in Janus Ram.

quale sia la zona riservata all'Input-Output e quindi può accadere che le normali scritture di dati dell'Amiga provochino un flusso di caratteri verso il PC causando un riempimento del buffer di tastiera. Tutto questo, che tra l'altro accade molto ma molto raramente, non ha generalmente altra conseguenza che quella di produrre dei beep sull'altoparlante del PC che indicano il buffer di tastiera pieno. Non dovrebbe invece accadere che il PC scriva su questa Ram (che lui vede come Rom) e del resto non mi è mai capitato che l'Amiga sia andato in Guru o in blocco per inconsistenze dei dati nella fast.

### Considerazioni finali sulla velocità

Ho provato a confrontare le velocità della memoria dell'Amiga e di quella della Janus. Allo scopo ho messo in memoria i due programmini dei listati 2 e 3 attraverso un normale monitor. Questi programmi che vanno lanciati con j 50000 e 210000 e che non fanno altro che un ciclo di 5.898.240 iterazioni mi hanno dato risultati diversissimi (vedi figura 9). Il primo, riguardante la Chip Ram ha girato per 10 secondi. Il

```

/*FindJanusMem      Public Domain by Giuliano C. Peritore
                   Via Amaseno, 6
                   04100 Latina - ITALY */

/* Compilare con lc -L FindJanusMem - (Lattice 5.0) */

#include "exec/types.h"
#include "exec/libraries.h"
#include "exec/interrupts.h"

struct JanusBase {
    struct Library      ja_LibNode;
    ULONG              ja_IntReq;
    ULONG              ja_IntEna;
    UBYTE              *ja_ParamMem;
    UBYTE              *ja_IoBase;
    UBYTE              *ja_ExpanBase;
    APTR               ja_ExecBase;
    APTR               ja_SegList;
    struct Interrupt   **ja_IntHandlers;
    struct Interrupt   ja_IntServer;
    struct Interrupt   ja_ReadHandler;
};

struct JanusBase      *JanusBase;

void main()
{
    JanusBase=(struct JanusBase *)OpenLibrary("janus.library",0);
    if(JanusBase==NULL)
    {
        printf("Can't open janus.library\n");
        exit();
    }
    printf("Janus Dual Ported Memory is at %X081X\n",JanusBase->ja_ExpanBase);
    CloseLibrary((struct Library *)JanusBase);
}

```

Listato 4 - FindJanusMem.

secondo, riguardante la Janus Ram, ha girato per 14 secondi. Ciò significa che un programma che girerà nella Janus Ram andrà 1.4 volte più lento di un programma uguale che giri nella Chip

Ram. Ovviamente il sistema non proteggerà per la lentezza di questa «Fast» Ram. Suggestisco quindi di mettere -15 come precedenza della Janus Ram in modo che questa memoria sia usata solo in casi di estrema necessità, pena il rallentamento di alcune operazioni.

È da notare che il miniciclo interno del listato 2 e 3 si avvale della piccola cache del 68010 e quindi con un Amiga con 68000 otterrete tempi leggermente superiori.

Anche il programma BenchMark della Lake Forest Logic evidenzia la differenza di velocità. L'ho fatto girare una volta in Chip Ram dando la priorità -15 alla Janus Ram ed i risultati sono in figura 10. La seconda volta l'ho invece fatto girare in Janus Ram dandole la precedenza +15 ed i risultati sono in figura 11. Come si vede la differenza è nettissima. Ciò tuttavia non toglie nulla al valore di questa memoria aggiuntiva che vi sarà utile parecchie volte.

Ed anche per questo mese è tutto. Buon lavoro e buon divertimento con questi nuovi 128 K di pseudo Fast Ram!

### MC-Link

Se c'è qualcuno interessato alla Janus o che ha sviluppato o al limite conosce qualcosa di interessante su tale scheda mi contatti postalmente o sulla casella MC6040 di MC-Link. Grazie.

```

/* JanusMem      Public Domain by Giuliano C. Peritore
                   Via Amaseno, 6
                   04100 Latina - ITALY */

/* Compilare con lc -L JanusMem - (Lattice 5.0) */

#include "exec/types.h"
#include "exec/libraries.h"
#include "exec/interrupts.h"

struct JanusBase {
    struct Library      ja_LibNode;
    ULONG              ja_IntReq;
    ULONG              ja_IntEna;
    UBYTE              *ja_ParamMem;
    UBYTE              *ja_IoBase;
    UBYTE              *ja_ExpanBase;
    APTR               ja_ExecBase;
    APTR               ja_SegList;
    struct Interrupt   **ja_IntHandlers;
    struct Interrupt   ja_IntServer;
    struct Interrupt   ja_ReadHandler;
};

struct JanusBase      *JanusBase;

void main()
{
    JanusBase=(struct JanusBase *)OpenLibrary("janus.library",0);
    if(JanusBase==NULL)
    {
        printf("Can't open janus.library\n");
        exit();
    }

    Forbid();
    AddMemList(0x20000,5,-15,JanusBase->ja_ExpanBase,NULL);
    Permit();

    printf("Janus Dual Ported Memory at %X081X added to system memory\n",
        JanusBase->ja_ExpanBase);

    CloseLibrary((struct Library *)JanusBase);
}

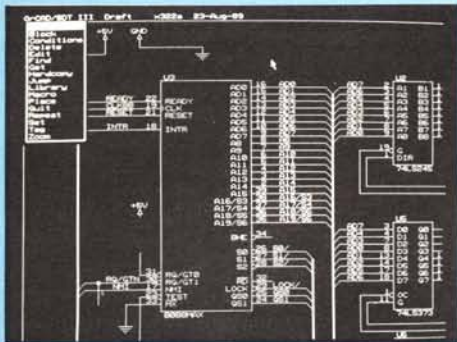
```

Listato 5 - JanusMem.



# CAD-CAE ad alte prestazioni

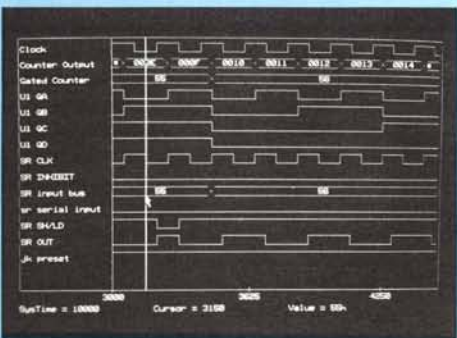
S  
D  
T



numero uno per il disegno di schemi elettrici per velocità, semplicità d'uso e potenza.

- Librerie con più di 6000 simboli espandibili con editor grafico.
- Driver video (più di 100), plotter e stampanti (più di 75).
- Lista parti e net-list per i CAD-PCB più diffusi.
- Schemi gerarchici: fino a 4000 fogli con 200 livelli di gerarchia.
- Potenti funzioni grafiche: rotazione e ribaltamento simboli, spostamento blocchi, drag dei collegamenti, import export di blocchi e file di testo.
- Numerazione automatica componenti, controllo degli errori, gestione collegamenti a bus.
- Macrocomandi definibili dall'utente.

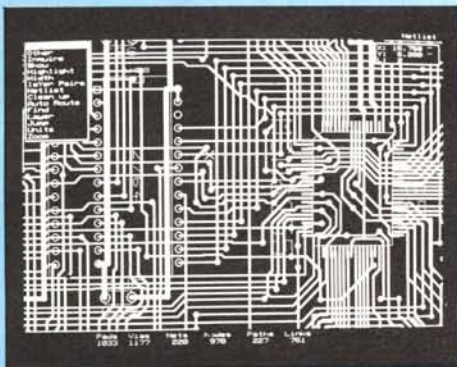
V  
S  
T



la potenza di una workstation su PC.  
In simulatore digitale ad alte prestazioni: veloce, potente e facile da usare, per verificare la funzionalità del circuito man mano che si sviluppa il progetto.

- Libreria completa di componenti: TTL, CMOS, Memorie; ampliabile dall'utente.
- Simulatore a 12 stati.
- Velocità di elaborazione 65.000 eventi al secondo.
- Visualizzazione dei segnali graficamente come su analizzatore logico, possibilità di raggruppare segnali in bus.
- Marker (3) per la misura di intervalli di tempo.
- Possibilità di inserire fino a 10 break point come AND-OR di 16 segnali.

P  
C  
B



- Lo strumento più evoluto su PC per il layout di schede.
- Griglie di lavoro selezionabili da 50 a 5 mils.
  - Schede multistrato (fino a 16) e componenti SMD.
  - Autorouter indirizzabile: per pad, net, moduli o blocchi.
  - Router manuale con controllo isolamenti in linea DRC.
  - RIP-UP e ottimizzatore per ridurre il numero di fori.
  - Selezione di ampiezza piste, forma e dimensione pad.
  - Possibilità di modificare dopo il router automatico: ampiezza piste e dimensione pad.
  - Gestisce zone ramate, zone proibite e senza fori.
  - Master del circuito, serigrafia, maschera per solder, piano di foratura; su plotter, stampante o fotoplotter.
  - drill tape per macchine automatiche di foratura.

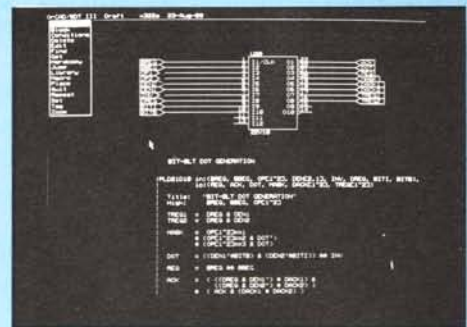
ORCAD



MICRODATA SYSTEM S.R.L.

LA SPEZIA - Tel. 0187/966123 - Fax 0187/988322  
 TORINO - Tel. 011/7496549 - Fax 011/7492986  
 MILANO - Tel. 02/4238296 - Fax 02/72023634  
 PADOVA - Tel. 049/8801466 - Fax 049/591479  
 FIRENZE - Tel. 055/8825600 - Fax 055/8825921  
 ROMA - Tel. 06/9322391 - Fax 06/9322853

P  
L  
D



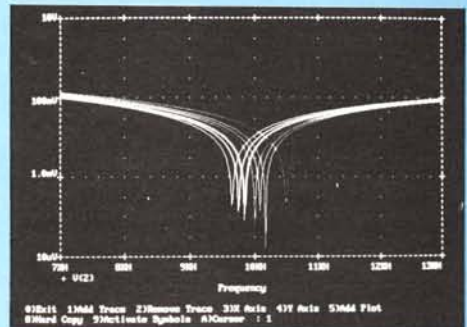
Un nuovo strumento flessibile e completo per progettare e realizzare PLD partendo da:

- Schema circuitale disegnato con ORCAD/SDT;
- Equazioni logiche Booleane;
- Equazioni con indici;
- Tabelle di verità;
- Mappe ed equazioni numeriche;
- Macchine a stati logici;
- Pseudo forme d'onda.

500 calcolatrici omaggio portate questo annuncio al ns. Stand F27 Pad. 16 - BIAS '90

- omnisce:
- File in formato JEDEC per programmatore di memorie;
  - Vettori di test per la verifica del chip dopo la programmazione;
  - Documentazione di progetto integrabile in ORCAD/SDT.

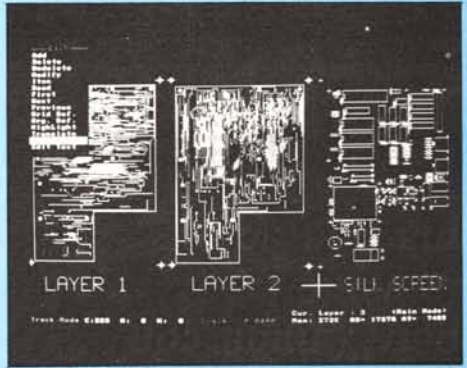
P  
S  
P  
I  
C  
E



Il simulatore analogico più diffuso su Personal Computer:

- Analisi in DC e AC;
- Risposta ai transitori;
- Risposta in frequenza;
- Analisi del rumore;
- Analisi in funzione della temperatura;
- Ampia libreria di moduli espandibile col modulo PARTS;
- Analisi visualizzabili in forma grafica (modulo PROBE);
- Blocchi funzionali simulabili inserendo la risposta in frequenza o la trasformata di Laplace (modulo ANALOG BEHAVIORAL);
- Analisi del circuito variando statisticamente i parametri (modulo MONTE CARLO).

A  
L  
S  
V  
I  
E  
W



Il naturale complemento per qualunque CAD-PCB: un sofisticato Editor grafico ed ottimizzatore per file di fotoplotteratura in formato GERBER;

- Permette di visualizzare, editare, modificare file di fotoplotteratura;
- Ottimizza la lunghezza del file;
- Permette di comporre sullo stesso film i vari piani della scheda o più copie dello stesso piano;
- Permette di modificare il set di apertura e visualizzare il risultato;
- Converte file da formato plotter: HPGL e DMPL;
- Possibilità di uscita su stampante grafica e laser.