

\*\*\*\*\*

MICROPOLIS RUMORS

by Buzz Rudow

There are stories being told of the cessation of production of Micropolis drives as we know them, that is, as subsystems, and specifically of the 1000 module. A subsystem is the packaging of a drive in an enclosure, with or without a power supply. Told in their worst form, one would deduce that we are in deep trouble pertaining to repair or replacement of current equipment, or ad-on of additional equipment. These stories scared me badly, so I did a bit of investigation. To give you the bottom line first, in general, we aren't in deep trouble.

Micropolis management is indeed considering whether they wish to continue producing the 1000 series of products, and whether or not they wish to continue making subsystems. They have been making an 1100 series of drives for about a year now. The 1100 drive is plug compatible with the 1000 series, and runs off the standard controller card. Production of the controller, just as we know it, will continue.

The new drive runs much faster, and quieter, and is less expensive to make. It's a better drive. So where's the problem?

Since they are plug compatible, who not just put the 1100 drives in the current boxes? The problem is that the 1100 drive and its on-board electronics are physically different from the 1000. The faceplate and disk locking mechanism has been changed. The drive can not be bolted from the bottom. The electronics to control two drives can not be contained in the one circuit board currently being used on the series 1000 dual module.

The new drives absolutely won't fit in the current double-drive enclosure. They could fit in the single drive enclosure, but re-working would be required.

Here are what the options of Micropolis seem to be.

- 1) Cease all 1000 series and subsystem production and concentrate on bare-drive 1100 series sales to OEMs.
- 2) Cease all 1000 series production, rework the single drive enclosure to produce an 1100 series subsystem.
- 3) Same as 2, but in addition, design a new dual-drive cabinet.
- 4) Continue making both the 1000 and 1100 drives, with only the 1000 series available as subsystems.

Obviously there are costs, sales projections, company philosophy, FCC regulations, and a bunch of other variables that Micropolis must consider before reaching a decision. I surely have no knowledge of this data, and since I'm not psychic, I won't predict the outcome. A better activity is for me to explain where we'll stand if Micropolis elects option 1 for implementation.

As I said, the controller board will still exist. The current software will still exist and will run the 1000 drives.

REPAIRS

There is some law that says a company must keep repair parts for a product for seven years. So, if we

need drive motors, heads, etc., for our 1000-type drives, we can get them. This is already the case with MOD I's. Production was stopped two years ago, but you can still get them repaired.

REPLACEMENT

If you are running your 1000 series drives in a custom enclosure, such as the Vector Graphic MZ box, you can simply remove the 1000 drive and replace it with an 1100 drive. Fact is, you should be doing that now.

If you are currently using a subsystem, there's a bit of a problem. You can't replace the drive, because it won't physically fit, and the electronics is different. You may be able to jury-rig it, if you have a single drive enclosure, and if you're into that sort of thing. Otherwise, you will need to get a separate enclosure from an outside source and mount the drive in it. You'll have to make your own cables, too. Priority One, as well as many other places, have all the needed parts.

ADDITIONAL EQUIPMENT

As in replacement, you'll have to get an enclosure from an outside source, and make your own cables.

SUMMING IT UP

While there may be some inconvenience for those of us running dual-drive subsystems, there isn't any great problem. The new drives are much better than the 1000 series and should be incorporated into any new system. The current S/W, be it MDOS or various versions of CP/M, is written for the 1000 series. It will work with the 1100's. It has wait times that are considerably too long for the 1100 series, however. (In most cases, they are too long for the 1000 series, too, but that's another story.) What we, as MUGers, need to do is to identify the wait loops and then create new versions of the S/W for the 1100 drives.

So don't panic. We'll be just fine. Micropolis will help us help ourselves, and we'll end up with better systems, too.

.....

BASIC PROGRAMMING TIPS

by Buzz Rudow

MONITORING RECORD TRANSFERS

One of rules of "user friendly" software is to never leave the user with a blank or inactive screen. The most flagrant violation of this rule occurs when the program is doing a disk-to-disk transfer of records. The operator has said "GO" and then everything is quiet. If monitoring of record transfers is not incorporated into your program, then after awhile, certainly with large files, the operator (1) wonders whether the program is working, or (2) wonders where he is in the transfer and how much longer it's going to take.

You can alleviate the problem by showing the operator where the program is in the transfer cycle. Since any printing to the screen is time consuming, I don't want to tell of each transfer - just often enough to instill confidence that the system is functioning properly. The following code uses a record count of 20 between signals. You can use a smaller or larger figure.

Micropolis Basic

```

10 FOR I%=1 TO 1000
20 ! Get record to be transferred
30 ! Write record to be transferred
40 IF MOD(I%,20)=0 THEN PRINT "Transferred through
Record";I%
50 NEXT I%

```

Basic-80

```

10 FOR I#=1 TO 1000
20 | Get record to be transferred
30 | Write record to be transferred
40 IF I# MOD 20 = 0 THEN PRINT "Transferred through
   Record";I#
50 NEXT I#

```

Basic/z

Basic/z code can be implemented exactly like the Micropolis code. Since the loop doesn't exceed 65000, you can use I# instead of I% and save a byte of memory.

What's more impressive is using the absolute cursor addressing capabilities of Basic/z. Again, any I/O to the terminal takes time. Rewriting the "Transferred through Record" phrase and scrolling of the screen are really things you don't want to do. Try it both ways and see the difference in the loop execution speed.

```

10 FORMFEED
20 PRINT "Transferred through Record";
30 FOR I#=1 TO 1000
40 | Get record to be transferred
50 | Write record to be transferred
60 IF MOD(I#,20)=0 THEN PRINT TAB(1,28);I#;
70 NEXT I#
80 PRINT

```

.....

THE \$29.95 SOFTWARE

=====

by Buzz Rudow

I received an interesting letter from Gary Van Cott this month that deserved an answer. The more I thought about it, the more I felt that the answer to the question would be of interest to more than just Gary.

Gary writes: "I gather you have some connection to the Basic/z language. I have a suggestion. Lower the price to \$59.95 or better yet \$29.95. Take out some big ads in Infoworld and similar places. Then I think that you will sell a lot of software and make some money. With two BASICS already, there is no way I am going to spring \$300 for another version. But if it is as good as you say, and is priced right, I probably would."

Perhaps you're right. The customers surely would be happy. I'm not sure about Bob and me, or any other of Bob's dealers. I probably should have discussed the following with Bob, but I didn't. These are just my thoughts, and don't reflect any of Bob's actual situation or operations. It's just a mental exercise to see what needs to be considered when marketing a piece of software.

You can do some figuring yourself. Suppose you wanted to make your living as a programmer, and needed \$25,000 a year. It takes a year to get the program and documentation complete. You're \$25K in the hole and ready to sell. Monthly costs, working out of your house, are maybe \$2000/month advertising (at least - my little 2 x 3 ad in Byte costs \$600 a month, for example), \$500/month printing and reproduction, and \$300/month long distance phone. You've spent \$33,600 this year, so after two years you're out \$83,600. Actual reproduction of the manual and disk probably cost \$25; shipping is another \$5 (I don't know what the \$29 Pascal manual is like but Basic/z' is BIG).

If you sell at \$345, you make \$315, assuming you sell it yourself and don't have to pay a dealer like me. To make the \$38,600 recurring cost, you need to sell 186 programs. You have to sell 80 more to make up for the year-one development cost. As I see it, there's no way to sell Basic/z at \$30, since there's zero profit. Could you sell 236 at \$345? I hope so, let's suppose so.

So, what about something in between? Possibly. But the real world isn't as I've described it above. A S/W publisher such as System/z does have dealers that it has to pay, does have people helping with the clerical and routine S/W chores, does have computer equipment to pay for, and has to worry about health and life insurance costs and retirement. One also (I speak from experience) tends to underestimate costs. I'm also sure that the \$25K salary is low, if one is deciding whether to go-it-alone or work for industry, and one has the talent of a Bob Zale.

What you need to gross is more like \$100K a year. At \$345, or \$315 gross/package, that's 317 sales. At \$50, or \$20 gross/package, that's 5000 sales. One's costs don't stay the same when you sell 5000 units, though. You'd require more support personnel to answer questions, and more clerical personnel to send out the merchandise.

How many can you sell at each of these prices? I don't know. One would have to do a market survey. Why not try a low price and see? Because your costs go up (support and clerical) to service the increased sales and you may dry up the market. For instance, how many refrigerators would you buy, even if they were \$10 a piece?

It wouldn't surprise me if Basic/z' price comes down over the years. Not that Zale has told me it would, but because that's the way our system works. A company develops a product and initially sells it at price X to: (1) recoupe development costs, and, (2) to service those buyers who "need" it worse than others. What I mean is that a professional application S/W developer may decide that Basic/z is going to make him 50% more productive, so the \$345 price doesn't phase him. After awhile, when you begin to understand your market, you probably lower your price to attract the more price conscious buyer.

It's all quite complicated, and to be frank, sometimes scary, as you may know if you've ever gambled your money and future on an idea that may or may not work out. If Bob and I were geniuses we'd both retire to the Caribbean and be able to tinker on our computers all day, giving our products to the public domain without worrying about making a living. Fantasies are nice.

JRT PASCAL

Is the JRT Pascal a good buy? There's quite a discussion going on, with Dr. Dobb's Journal in the middle of it. Dr Dobb's Journal, by the way, is a well respected magazine for the advanced computerist, a magazine which has been around since the beginning of hobby micro-computers. Dr Dobb's review of JRT's product in the November '82 issue and found it "unacceptable at any price" and "overpriced at \$29.95". The December issue contained a letter from JRT stating that many reviewers thought JRT Pascal was great, and that they would no longer do business with Dr Dobb's. It's true that there have been good reviews; Info-World, for instance. The December issue also contained a reader's letter, code and output testing which showed that JRT Pascal calculates  $X, Y * 0.0 = X.Y$ . Not too neat. In January, Dr Dobb's backed off a bit. Now their review conclusions are: "If you write programs only for your own use, and if you expect never, ever, to move to different hardware, and if you enjoy puzzling out and coding around compiler failures, then you will find JRT Pascal an acceptable product. If you, like me, want the ability to write (or use) portable code, or if you believe as I do that a compiler should compile, regularly and reliably, then you will find JRT Pascal Not Acceptable just as I did."

ELLIS' SOFTWARE

Ellis computing was the second company to go the \$29.95 route with programs which previously sold for \$120 to \$200. The Ellis product seems OK and hasn't received any bad press yet. What irritates me is that Mr. Ellis did this switch after he had

sold me some versions at dealer discount from the old full list.

So now I'm stuck with products I paid \$100 and up for, that are worth \$29.95. Well, I don't need them. Do you want them? First come first serve, one copy only of each, and you won't have to wait 3 months to receive them. Just don't ask me to support them.

Nevada Cobal       \$29.95  
Nevada Pilot       \$29.95  
Nevada Edit        \$29.95  
Plus shipping.

.....

#### MICROPOLIS ON AN 8085/8088

by Melvin M. Dalton  
7826 West 80th Street, Playa Del Rey, CA 90291

I have recently completed (in so far as such things are ever really done) the transition from a SOL-20 based MICROPOLIS system to one using the COMPUPRO 8085/8088 CPU. Replacement costs and their fine hardware/software performance dictated the need to retain the MICROPOLIS drives.

The COMPUPRO CPU has a switch selectable clock rate for the 8085. The choices on my board are 2Mhz or 6Mhz. Integration began at 2Mhz and proceeded with no more than the usual hitches. A working interim version of a monitor called MELOS was designed (by borrowing heavily on SOLOS and EXTCCP7), debugged and integrated with a TELEVIDEO 950 terminal. The remainder of the equipment consists of two RAM 20 32K memory cards, a COMPUPRO enclosure/power supply, a CROMEMCO TUART to talk to the 950 and the EPSON MX-80, a SSM PB-1 EPROM board for the MELOS PROM, and of course the MICROPOLIS controller and drives.

Now for the punch line. The MICROPOLIS software is designed for use at 2 or 4 Mhz, but NOT at 6 Mhz. If I was to get the most out of my new CPU, I had to make some changes. A few phone calls later (including MICROPOLIS) told me I was on my own. Well not quite. MUG newsletter #22 contains George Shaw's juicy article. Chapter 6 of the MICROPOLIS Manual is also very helpful.

I will not attempt a blow by blow discussion of the design process. Eventually I found it necessary to make changes in two functional areas. The most obvious (with the help of Shaw's article) is to change the routine at 1336H to provide 1 ms of delay at 6 Mhz. This routine is called from eight different locations in RES and provides the delay time for various mechanical things to occur in the MICROPOLIS drives.

The second functional area where changes are necessary for reliable operation are the read, write, CRC check and the byte by byte verify routines. The timing constraints specified in Section 6.4.4 paragraph 5 and Section 6.4.5 paragraph 5 of the MICROPOLIS Users Manual are not to be ignored if reliable operation is expected. The existing software timing meets the requirements at 2 & 4 Mhz BUT NOT at 6 Mhz.

An upper limit of 25 usec. for the software to put or get a byte to/from the controller is easier to meet the higher the clock rate. However, the lower limit of 6-7 usec. is not met by the existing PDS vs. 4.0 code for clock rates above about 5 Mhz. The solution is to replace the four existing routines with ones which are padded to require about 40 clock times to execute. The padding is done with an initial JMP and then NOP's as needed. The JMP occurs after the SYNC byte is handled. The JMP plus extra NOP's if needed are used to reach about 40 clock times from sync to first byte. Additional NOP's are inserted as needed to reach about 40 CT's between each data byte and from the last data byte and reading/writing the CRC byte. A JMP returns the program flow to the original code after the CRC

byte is handled and the timing is no longer so critical.

The new routines require more memory than the old ones. The only place in RES to put them that I found was at the end of the space reserved for console/printer I/O. The apparent blank spot at 1180H was tried but later I found it is used by disk drives 2 & 3. I placed the new code at 700H. Others may need to relocate their I/O drivers (and don't forget to change the jump tables) to provide room. The four subroutines are independent so you can put one here and another there if it suits your fancy. The code I used is shown in Listing 1.

Now comes the problem of installing the changes to RES. First, make sure your system works at 2 Mhz. Second, load the patches to RES by executing a LOAD "FASTRESO" or whatever name you gave the scatter load object file resulting from the assembly of the source code in Listing 1. Third, test the new RES with several load and save operations to the disk. It should work without a hitch IF the speed jumper is OUT of the MICROPOLIS controller. Fourth, save the new RES in place of the old one per the instructions in the manual. Fifth, install the speed jumper on the controller board and switch the CPU to 6 Mhz. Boot the system up and test it at 6 Mhz.

Problems??? Don't forget that some EPROM boards need 1 or 2 wait states before your monitor works reliably. Your I/O card may need wait states (the COMPUPRO dual CPU card has a DIP switch for this feature). Another possibility is a data entry error. It never hurts to double check! There may be a problem with the cold boot load program on the controller card. Mine worked fine at 6 Mhz but it does contain a disk read routine which doesn't meet the minimum time requirements given earlier.

Are we through with the conversion to 6 Mhz? Not quite yet. The DISKCOPY utility needs all four disk subroutines and the delay routine. Unfortunately, it does not use the ones in RES even though they are undisturbed after DISKCOPY overlays MDOS. The subroutines are strung together in a different way by the DISKCOPY programmer. The sections of interest to us are verbatim copies from RES so we simply need to patch DISKCOPY to use them. I could not find a way to SAVE an overlay file. It is easy enough to LOAD a copy of DISKCOPY into memory above 2B00H and then patch it as needed. If I SAVE the new version without first moving it down to its overlay position, I can not get the correct load and execute addresses on disk. If I move it first (using my monitor since the MDOS move routine will be overlaid before the move is complete) then MDOS can not be used to SAVE the program.

My solution is as follows:

1. Assemble Listing 2 to create scatter load file COPYPATCHO.
2. FILL 2B00 33FF 0 ;clean slate to write on
3. TYPE "DISKCOPY" B ;temporary to allow offset load.
4. LOAD "DISKCOPY" 2B37 ;offset load
5. LOAD "COPYPATCHO" ;insert move routine and patches
6. SAVE "COPYDISK" 2B00 33FF 18 ;new 6 Mhz version
7. TYPE "DISKCOPY" C ;back to overlay file type

COPYDISK should now function the same as the old DISKCOPY. After testing to be sure, you can SCRATCH "DISKCOPY" and RENAME "COPYDISK" "DISKCOPY".

The results are very pleasing. MICROPOLIS BASIC really zips along. A ten minute Shell sort of 650 names in a mailing list now runs in about three minutes. The computer only takes 25 sec. for a move in QUBIC as apposed to the old 90 seconds.

When I make my next revision to MELOS, I plan to include a cold boot loader to replace the one on the MICROPOLIS controller. My next step, however, is to modify the BDOS in CP/M on MICROPOLIS version 1.411 from LIFEBOAT ASSOCIATES to run at 6 Mhz. That will really help the compile and link times in PL/I-80.

.....

```

0010 *
0020 *****
0030 *   PATCHES TO 'RES' FOR OPERATION AT 6Mhz
0040 *
0050 *   BY MELVIN M. DALTON
0060 *   7826 WEST 80th ST.
0070 *   PLAYA DEL REY, CA
0080 *   90291
0090 *
0100 *   NON-EXCLUSIVE RIGHTS DONATED
0110 *   TO MICROPOLIS USERS GROUP
0120 *****
0130 *
0140 *
0150 *   ORG   2B00H   ;DUMMY INSTRUCTION
0160 *           ;TO FORCE
0170 *           ;OVERLAY LOADING OF
0180 *           ;REST OF ROUTINES
0190 *   NOP
0200 *
0210 *   UNUSED AREA OF 'RES'
0220 *
0230 *   ORG   700H
0240 *
0250 *
0260 *   NEW WRITE LOOP
0270 *
0280 * WLOOP  MOV   A,M   ;FROM BUFFER
0290 *       NOP           ;PAD FOR 6Mhz
0300 *       STAX  D      ;TO DISK
0310 *       ADC   B       ;COMPUTE CRC
0320 *       MOV   B,A     ;SAVE CRC
0330 *       INX   H       ;NEXT BUFFER SLOT
0340 *       MOV   A,M     ;FROM BUFFER
0350 *       NOP           ;PAD FOR 6Mhz
0360 *       NOP           ;MORE PAD
0370 *       NOP           ;STILL MORE PAD
0380 *       STAX  D      ;NOW TO DISK
0390 *       ADC   B
0400 *       MOV   B,A
0410 *       INX   H
0420 *       DCR   C       ;COUNT BYTE PAIRS
0430 *       JNZ  WLOOP   ;REPEAT
0440 *       MOV   A,B     ;GET CRC
0450 *       STAX  D      ;SEND TO DISK
0460 *       JMP   138BH  ;RETURN TO 'RES'
0470 *
0480 **   NEW READ LOOP
0490 *
0500 * RLOOP  LDAX  D      ;READ DISK
0510 *       MOV   M,A     ;WRITE BUFFER
0520 *       INX   H
0530 *       ADC   B
0540 *       MOV   B,A     ;CRC BACK IN 'B'
0550 *       NOP           ;PAD FOR 6Mhz
0560 *       NOP           ;MORE PAD
0570 *       NOP           ;STILL MORE PAD
0580 *       LDAX  D      ;READ DISK AGAIN
0590 *       MOV   M,A     ;INTO BUFFER
0600 *       INX   H
0610 *       ADC   B
0620 *       MOV   B,A
0630 *       DCR   C       ;COUNT BYTE PAIRS
0640 *       JNZ  RLOOP   ;REPEAT
0650 *       LDAX  D      ;GET CRC FROM DISK
0660 *       JMP   13BFH  ;RETURN TO 'RES'
0670 *
0680 *   NEW CHECK LOOP
0690 *
0700 * CKLOOP NOP           ;PAD FOR 6Mhz
0710 *       MOV   A,M     ;READ DISK
0720 *       STAX  D      ;PUT IN BUFFER
0730 *       ADC   B
0740 *       MOV   B,A
0750 *       INX   D
0760 *       NOP           ;PAD FOR 6Mhz
0770 *       NOP           ;MORE PAD
0780 *       NOP           ;STILL MORE PAD
0790 *       MOV   A,M
0800 *       STAX  D
0810 *       ADC   B
0820 *       MOV   B,A
0830 *       NOP           ;PAD FOR 6Mhz
0840 *   CLOOP1 NOP          ;LOOP TO HERE
0850 *       NOP           ;FOR RIGHT DELAY
0860 *       NOP           ;FOR 6Mhz
0870 *       MOV   A,M     ;READ DISK
0880 *       ADC   B
0890 *       MOV   B,A
0900 *       PUSH  PSW     ;PAD FOR 6Mhz

0910 *   NOP           ;MORE PAD
0920 *   POP   PSW       ;STILL MORE PAD
0930 *   MOV   A,M       ;READ DISK
0940 *   ADC   B
0950 *   MOV   B,A
0960 *   DCR   C         ;COUNT BYTE PAIRS
0970 *   JNZ  CLOOP1
0980 *   NOP           ;PAD FOR 6Mhz
0990 *   NOP           ;MORE PAD
1000 *   NOP           ;STILL MORE PAD
1010 *   MOV   A,M       ;READ CRC BYTE
1020 *   JMP   13COH    ;RETURN TO 'RES'

1030 *
1040 *   NEW VERIFY LOOP
1050 *
1060 * VLOOP  NOP           ;PAD FOR 6Mhz
1070 *       NOP           ;MORE PAD
1080 *   VLOOP1 LDAX  D      ;READ DISK
1090 *       XRA   M       ;SAME AS BUFFER?
1100 *       JNZ  141DH   ;'RES' ERROR EXIT
1110 *       INX   H
1120 *       NOP           ;PAD FOR 6Mhz
1130 *       NOP           ;MORE PAD
1140 *       NOP           ;STILL MORE PAD
1150 *       LDAX  D
1160 *       XRA   M
1170 *       JNZ  141DH   ;'RES' ERROR EXIT
1180 *       INX   H
1190 *       DCR   C
1200 *       JNZ  VLOOP1
1210 *       LDAX  D      ;READ CRC
1220 *       JMP   1417H  ;RETURN TO 'RES'
1230 *       ;NO ERROR
1240 *
1250 *   WRITE PATCH -- REPLACE OLD CODE
1260 *   WITH JMP TO NEW CODE AND NOP'S
1270 *
1280 *   ORG   137BH
1290 *   JMP   WLOOP     ;GO WRITE 'C' BYTE
1300 *           ;PAIRS, AN CRC BYTE
1310 *           ;AND JMP 138BH
1320 *   FILL  138BH-$,0 ;NOP UNNEEDED CODE
1330 *
1340 *   READ PATCH -- REPLACE OLD CODE
1350 *   WITH JMP TO NEW CODE AND NOP'S
1360 *
1370 *   ORG   1380H
1380 *   JMP   RLOOP     ;GO READ 'C' BYTE PAIRS
1390 *           ;AN CRC BYTE & JMP 13BFH
1400 *   FILL  13BFH-$,0 ;FILL UNNEEDED CODE
1410 *
1420 *   CHECK PATCH -- REPLACE OLD CODE
1430 *   WITH JMP TO NEW CODE AND NOP'S
1440 *
1450 *   ORG   13E2H
1460 *   JMP   CKLOOP   ;GO READ TRACK, SECTOR,
1470 *           ;CRC, & JMP 13COH
1480 *   FILL  13FCH-$,0 ;FILL UNNEEDED CODE
1490 *
1500 *   VERIFY PATCH -- REPLACE OLD CODE
1510 *   WITH JMP TO NEW CODE AND NOP'S
1520 *
1530 *   ORG   1406H
1540 *   JMP   VLOOP     ;GO READ & CHECK
1550 *           ;BYTE FOR BYTE BETWEEN
1560 *           ;DISK AND BUFFER
1570 *   FILL  1417H-$,0 ;NOP UNNEEDED CODE
1580 *
1590 *   ONE MILLISECOND TIMER
1600 *   W9 JUMPER ON CONTROLLER
1610 *   OUT=2Mhz, IN=6Mhz
1620 *   'DE'=NUMBER OF MILLISECONDS
1630 *
1640 *   ORG   1336H   ;PUT IN PLACE OF OLD ONE

1650 *
1660 * TIME  PUSH  H
1670 *       PUSH  B
1680 *       LHLD  0882H  ;POINT TO CONTROLLER
1690 *       MOV   A,M   ;GET STATUS
1700 *           ;& RESET 4 SEC. TIMER
1710 *       ANI  20H    ;CHECK SPEED JUMPER W9
1720 *       MVI  B,56   ;56*(36 CT's)=1ms @ 2Mhz
1730 *       JNZ  T1     ;NZ=2Mhz, Z=6Mhz
1740 *       MVI  B,167  ;167*(36 CT's)=1ms @ 6Mhz
1750 *   T1   MOV   A,B
1760 *   T2   PUSH  H     ;DO NOTHING
1770 *       POP   H     ;FOR 21 CT's

```

```

1780      DCR  A      ;COUNT = 5 CT's
1790      JNZ  T2     ;LOOPING HERE = 36 CT
1800      DCX  D      ;NUMBER OF MILLISECONDS
1810      MOV  A,D
1820      ORA  E
1830      JNZ  T1
1840      POP  B
1850      POP  H
1860      RET
    
```

MICROPOLIS MAINTENANCE  
 =====

by Buzz Rudow

Another company that does Micropolis Repair has been brought to my attention. Those of you on the west coast might try Computer Peripheral Service, 3541 Old Conejo Rd., Newbury Park CA 91320, (805) 497-6577.

Aida Pineda runs the place. He tells me he's doing some of the quick turn-around work of Micropolis. Aida's normal turn-around time is 2-3 days, though you can get 8-hour service for a 30% additional charge. He also repairs 8" floppy drives. Computer Peripheral Service warranties their work for 90 days.

Micropolis repair prices are:

- Single Drive \$65 + parts
- Dual Drive \$140 + parts
- B Controller Board \$75
- Diagnosis only \$35

On the east coast, as stated in the October MUG, is Jack MacGrath of Tercentennial Technical. Contact Jack for your Micropolis repairs at 70 Tercentennial Dr., P.O. Box 5, Billerica MA 01821, (617) 667-8272 (after 6:30 PM EST).

```

0010 *
0020 *****
0030 *
0040 * REVERSE DISKCOPY UTILITY
0050 * TO RUN AT 6Mhz
0060 *
0070 * BY MELVIN M. DALTON
0080 * 7826 WEST 80th ST.
0090 * PLAYA DEL REY, CA
0100 * 90291
0110 *
0120 * NON-EXCLUSIVE RIGHTS DONATED TO
0130 * MICROPOLIS USERS GROUP
0140 *****
0150 *
0160 * 1) FILL 2B00 33FF 0
0170 * 2) LOAD "DISKCOPY" <start address>
0180 * start address = LOADHERE
0190 * 3) LOAD "COPYPATCHO"
0200 * 4) SAVE "COPYDISK" 2B00 33FF 18
0210 *
0220 OLDSTART EQU 1572H ;START OF DISKCOPY
0230 OLDEND EQU 1DA7H+1 ;END OF DISKCOPY+1
0240 LENGTH EQU OLDEND-OLDSTART
0250 *
0260 ORG 2B00H
0270 *
0280 COPYPAT LXI H,LOADHERE ;SOURCE
0290 LXI D,OLDSTART ;DESTINATION
0300 LXI B,LENGTH ;LENGTH
0310 MOVE MOV A,M ;GET FROM SOURCE
0320 STAX D ;PUT TO DESTINATION
0330 INX H
0340 INX D
0350 DCX B ;LENGTH=LENGTH-1
0360 MOV A,B
0370 ORA C
0380 JNZ MOVE ;MORE TO MOVE
0390 *
0400 * WHEN EXECUTED, THE NEXT SECTION
0410 * OF CODE WILL CHANGE THE NEW 'RES'
0420 * EXIT JMP'S SO THAT THEY RETURN
0430 * TO THE OVERLAYED DISKCOPY
0440 *
0450 * IT ASSUMES THAT THE NEW 'RES'
0460 * ROUTINES ARE ORG'ed AT 700H
0470 *
0480 LXI H,1CBCH
0490 SHLD 715H ;WRITE LOOP EXIT
0500 LXI H,1CFAH
0510 SHLD 72AH ;READ LOOP EXIT
0520 LXI H,1CFFH
0530 SHLD 74FH ;CHECK LOOP EXIT
0540 LXI H,1D76H
0550 SHLD 768H ;VERIFY LOOP EXIT
0560 LXI H,1D7DH ;DISKCOPY ERROR EXIT
0570 SHLD 756H
0580 SHLD 75FH
0590 JMP OLDSTART
0600 *
0610 LOADHERE EQU $ ;LOAD OLD DISKCOPY
0620 *
0630 ;WITH OFFSET TO HERE
0640 OFFSET EQU LOADHERE-OLDSTART
0650 ORG OFFSET+1CACH
0660 JMP 700H ;TO REVISED WRITE LOOP
0670 FILL 0DH,0 ;NOP UNNEEDED CODE
0680 ORG OFFSET+1CEBH
0690 JMP 717H ;TO REVISED READ LOOP
0700 FILL 0CH,0 ;NOP UNNEEDED CODE
0710 ORG OFFSET+1D33H
0720 JMP 72CH ;TO REVISED CHECK LOOP
0730 FILL 17H,0 ;NOP UNNEEDED CODE
0740 ORG OFFSET+1D65H
0750 JMP 751H ;TO REVISED VERIFY LOOP
0760 FILL 0EH,0 ;NOP UNNEEDED CODE
0770 ORG OFFSET+1C4EH
0780 JMP 1336H ;TO REVISED TIMER
0790 FILL 1355H-1336H-3,0 ;NOP UNNEEDED C
ODE
    
```

CP/M VER. 3.0 FOR MICROPOLIS  
 =====

by Buzz Rudow

There is a sizable interest being expressed by the MUG members for information on Digital Research's newest CP/M, Version 3.0, also known as CP/M Plus. Version 3 is said to run 4 to 10 times faster than 2.2. It rids us of the RESET, or log every new disk, problem. It supposedly allows extents of larger size, which does away with the time consuming problem of opening and closing extents when using a pointer file. However, it also takes 4 to 5 K more of the TPA area, if you are just working with 64K. CP/M 3 is really designed to run with multiple, bank-switched, memory boards.

CP/M 3.0 lists for \$300 and is being sold in the INTEL MDS configuration on 8" SSSD disks. The probability of MUG members being able to use this CP/M version varies with their hardware. If you're running Vector Graphic with Tandon drives, chances are pretty good that you'll be able to get an update from Vector Graphic. Several people have stated that using 3.0 shouldn't be a huge problem for Micropolis drives. All one has to do is locate the 8" disk access routines in 3.0, and replace them with your Micropolis routines contained in your 2.2 system. Of course, you'll also have to replace your console and printer I/O routines.

I don't know what the probability is for getting a "configurable" version, but I'm working on it. I'm also still working on the double-sided CP/M 2.2 problem. If anyone has knowledge of either piece of software, let me know.

```

*****
*                                     *
*      BASIC/Z SPECIAL INTEREST GROUP   *
*                                     *
*****

```

BASIC/Z CORNER #1

by Steven Guralnick  
 375 South Mayfair Ave., Suite 205  
 Daly City, CA 94015

This is the first of what I hope will be a continuing series of articles about BASIC/Z. Before I go any further, it might be helpful to describe who I am, and the particular slant that my background will bring to these columns.

I am an attorney, in a general law practice partnership in a suburb of San Francisco. Because of the obvious need, I commenced to bring microprocessing to our operation, starting in 1979. Originally, we had an Exidy, with two Micropolis drives. Recently, we have upgraded and now have Compupros, running with double sided, double density Qume drives, accessed with Televideo 925 terminals.

Originally, the purpose of our computers was word processing. That is something lawyers do a lot of and we have more than our fair share of paper pushing. However, a couple of years ago I joined MUG and Buzz steered me to a great guy and fellow member, Jerry Lenz. Jerry produced a number of software packages for our office, written in Micropolis BASIC. The last major package that Jerry and I worked on was a client ledger program and it was a resounding success.

At the beginning of 1982, I anticipated that we would be converting totally to CP/M and proceeded to convert our Micropolis BASIC software to CBASIC, using SPELLBINDER. By and large that was successful. However, when I ran into the client ledger program, it was obvious to me that converting the Micropolis BASIC files to CBASIC files was going to be a monumental task, particularly in handling disk I/O. This is because CBASIC will not report the number of logical records in a given file.

I had this vision that I would be married to MDOS forever and in desperation I called Buzz and asked for suggestions to get that package running in a CP/M compatible language. His suggestion was BASIC/Z. I bought the package from him and had that enormous program running within three evenings (On the old system, with lengthy compiling times involved). In fact, the conversion was so successful that I have since enhanced the program and am giving serious consideration to exploiting it commercially.

During the course of the last several months of working with BASIC/Z, I have come to have a good awareness of its present advantages and shortcomings. Buzz has been an enormous help, and Jerry (who has since purchased the package himself) has been just as much help. So, I suggested to Buzz that I might do a series of columns.

So much for who I am. Here is what I am not. I am not an experienced BASIC programmer, unless you call one year or so experienced. Also, I do not use BASIC for scientific work. To the best of my knowledge I have never written a program that uses the mathematical functions, COS, TAN and so forth. My slant on all of this is purely as a business user. I have not done extensive benchmarking with BASIC/Z. What I am good at is learning fast and that I have been doing a lot of for the last three and one-half years.

In the business of microprocessors, the biggest problem I see is the failure to communicate between those who have information and those who need it. It seems to be a chronic problem and I have done whatever I can to make it less so. However, this

is a two-way street. I want help and I solicit it. I do not want to be a voice in the wilderness with this column; in fact, I will not maintain this column unless I get regular input from my fellow MUGers.

For those of you who are not using BASIC/Z, but who are using Micropolis BASIC, remember, the languages have an enormous amount of commonality. So, I am sure that many of the routines which you have developed to do a fast job faster will be very much appreciated, even if the languages are not completely the same. In that regard, I plan a regular feature which I will shamelessly steal from Hewlett Packard. For many years I subscribed to their "Key Notes" for programmable calculators. Their editor had a feature called "25 Lines or Less". It was a valuable idea which consisted of short routines that not only made the wheel rounder but turning faster. I don't know how many dozen routines I experimented with, but it was a lot, and I want very much to implement that feature in this column. If you have a pet routine in some other BASIC, send it along to me in clearly readable hardcopy and if I cannot get it running on BASIC/Z I will drag in some help (Attention: Jerry Lenz, Attention: Buzz Rudow).

Also, I plan to see to it that a copy of this column is sent to Bob Zale at System/z for his comments. Bob is a nice guy who has been willing to listen to ideas for changes and improvements to BASIC/Z. He has a powerful program there and I am sure he knows it, but some changes are going to have to be made if it is going to be a top-flight product. I will be discussing those suggestions as the column goes along.

Your comments will be actively solicited at all times and I am more than willing to turn over the entire column to a guest contributor if something comes in that is too comprehensive to be excerpted. However, I will not fill up this column with lines and lines of source code. If one of you has a program out there and you are willing to part with your source code, then I am going to establish a policy that you be willing to furnish a copy of it upon a receipt of a dollar or two and a stamped, self-addressed envelope.

Compiler Basics are fun. The user gets an enormous amount of speed and a higher level language to write in. However, I have discovered that what I do not know about basic programming could fill a big book and I am sure that all of us could profit by a series of exchanges on what we do know.

Whether this column will be stillborn or be around next month will largely depend on the response either I or Buzz receives over the next four weeks. I would particularly welcome your observations on what you would like to see in an ongoing series.

Best Wishes and a Happy New Year!  
 .....

```

*****
*                                     *
*      VECTOR GRAPHIC SPECIAL INTEREST GROUP   *
*                                     *
*****

```

IS ANYONE INTERESTED IN GRAPHICS?

by Stephen D. Maegerlein  
 P.O. Box 60, Williams IN 47470

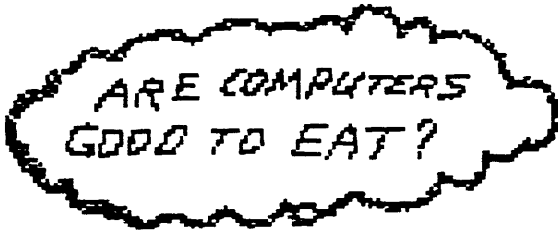
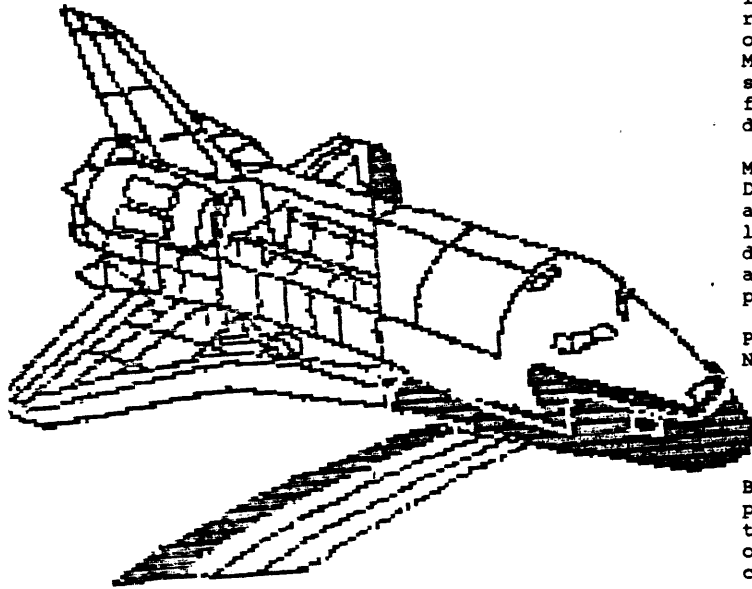
I have been working on a machine language program called COPYIT for several months. COPYIT allows transfer of graphics between the Vector Graphic's high resolution graphics screen and disk storage, or printing the picture on the IDS 440G dot matrix printer. The COPYIT program also allows merging of pictures using combinations of logic functions. Examples of the graphics are at the end of the article.



The pictures were created with Houston Instruments' HIPAD digitizer pad, Vector Graphic's GRAPHIT software and a small basic program called TRACEIT.

Is there enough interest in graphics for the IDS 440G printer and Vector Graphic's System B computer to warrant including the programs COPYIT and TRACEIT in the MUG library? The files used to generate the pictures would be included with the programs.

Although the programs are written for CP/M, the interface routines could be converted to MDOS.



LETTERS

Hates CP/M

Buzz: I feel that CP/M should receive only scant attention in the group, because we are, after all, a 'Micropolis' group. I realize that some members are running CP/M on Micropolis. But the only common ground is the hardware, and beyond controller boards and the addressing of such, there is no compatibility.

I do run CP/M 2.2 on my Micropolis, both machine code and Micro Soft Basic. I find the major problem of CP/M is that it is user unfriendly. I reckon it's downright dangerous. The thing has a habit of reverting to system whenever a small problem occurs. You can't easily swap disks around, etc. My wife is getting sick of hearing "I hate CP/M" shouted about. In fact, it once allowed two disk files with the same name and type on one disk. I don't know how it did that.

Micropolis is the best, safest, and most integrated DOS I have ever seen. It's been designed to cover almost every eventuality that may arise, and still let the user recover and carry on, with no loss of data. It's just not safe, in my humble opinion, to allow anyone without programming skills operate programs running in CP/M.

Pete Gorton, 44, Francis St., Castle Hill 2154, N.S.W. Australia.

Textwriter II

Buzz: I would like to find out why I am having a printing problem with my MDOS Textwriter program that I use in conjunction with Edit/S. The mistakes only occur when I invoke the QUME5 or DIABLO10/20 commands.

I seem to be getting false line feeds and the letters appear to move 1 increment to the right on each new line. If I run the program with my T.I. 820 printer, I don't have any problem. I am using Vector's MDOS Version 4.00.

Richard Herz, Aurora Manufacturing Corp. 212/388-4281. 74 White Ave., Brooklyn NY 11211

Richard: I spoke with Burks Smith about this. He runs a Diablo 630 on a Vector with Textwriter. Burks says you must use the custom print routines, and not go through MDOS (or CP/M) for the I/O. I haven't tried to do this, so I don't know how hard it is. Perhaps Burks could explain the procedure in one of his articles. A lot of people in the group have Textwriter. -- Buzz

CP/M Q-Files

Buzz: I'm having a problem with MUG CP/M disk 1006. Except for LIB.DOC, there are no files for any of the other programs. I do not know what the files are that have extensions with a Q in the middle, nor how to use them. Using CP/M TYPE with these files, for instance BSORT.DQC, gives me some sort of garbage on the screen .

Richard Hamburger, Germantown MD 20874

Richard: The Q files are compacted, or squeezed, files. You need the utilities on disk 1005 to TYPE them (TYPESQ), or unsqueeze them (USQ). The DQC is a squeezed DOC, etc.-- Buzz

"BOUND-UP" DRIVE MOTORS

Buzz: When you have to get work out today, and the motor on the disk drive is bound up, the day can still be saved. Remove the motor, and with a

hypodermic syringe, put about one half drop of WD-40 oil on the shaft bearing on each end of the motor. The motor can then be run (it takes a hand twist to start it). In about 10 minutes the oil will have worked its way into all the bearings. Then replace the motor and you're ready to go. I've run 20 hours per week for two months on this repair.

John Roberts, Cheyenne Wyoming

John: I've had similar problems, and just carefully sprayed WD-40 on the bottom bearing of the drive while it's still in the unit. Also worked.

I have motors and belts, however, for those of you who know that these fixes are temporary. Next month I'll list all prices and descriptions of the Micropolis parts I carry. --- Buzz

DISK EMULATORS

Buzz: I'd like to see someone write an article for MUG on the disk emulators that are now on the market. How does one interface them into the S-100 buss. Can we use them with MDOS? How do you go about reading and writing to them?

Bob Bradford, Philadelphia PA

CLASSIFIED

WANTED: Two Micropolis 1015 MOD II drives. Must be in nearly new condition with late serial numbers.

Gene Riding, 714/276-6224  
2227 Chicago St., San Diego CA 92110

WANTED: One Micropolis MOD I add-on drive.

Reg Brearley, 514/697-3660  
137 Broadview, Pointe Calire Que., H9R 3Z5 Canada

WANTED: Used Vector Graphic System B or VIP. Will consider printer, but not desired. Must have Mem-rite III.

H. Spierer, 203/322-3564  
71 Big Oak Road, Stamford, CT 06903

FOR SALE: Vector Graphic S-100 boards (guaranteed to run): 8" FDC, \$175; Micropolis 1015-II disk controller, \$100. 48K RAM, \$149. 64K RAM, \$200. Bitstreamer I/O, \$75. 8K PROM/RAM, \$50. Peachtree Software, \$300 (G/L, AP, AR, PR, INV). CBASIC, \$50. MDOS with BASIC \$40.

Does not work, ZCB single board computer, \$100. Z80 board, \$60.

David Paden, 205/595-6792 (leave message).  
P.O. Box 292, Trussville AL 35173

WANTED: Any information of the Space Byte 8085 CPU board. Also want to know of anyone doing H/W or S/W maintenance on the LATAH computer system.

John Roberts, 307/638-6161  
1025 East 15th St., Cheyenne Wyoming 82001

WANTED: Information on how to use the Shaw Labs Micropolis Disk Enhancement Kit with Exidy Sorcerer. Also want info on opening up Exidy to

Bob Bradford  
1020 Winchester St., Philadelphia PA 19111

WANTED: A SDK-85 development board.

Benoit Carrignan  
533 Darveau, Longueuil, P.Q. J4L 2V3 Canada

LIBRARY DISKS

I've been hit on the head enough. Some members feel that the new system of having disks half-full of categorized software is unreasonable. So, I'm going to move incoming software to a new set of disks. They will be labeled 900 and up for MDOS, and 1900 and up for CP/M. They will be "full", that is, approximately 70 tracks, or two MOD Is full.

These disks will be temporary, in the sense of perhaps six months availability. During their six months life, I will move them to the categorized disks, which will also be expanded to be "full". Then the temporary, mixed input, disks will be removed from the library.

The aim is to attain the best of both library concepts. Old members may get a copy of the new temporary disk and move the software to old disks, or to anywhere they wish. New members can still purchase a disk which exclusively contains software in their interest area, such as Home, Business, System Enhancements, etc.

The first of these temporary disks are listed below. MOD I users must, again, submit a second disk, or pay \$4 extra for me to supply the extra disk.

The first disk, 1900, is a large set of I/O routines, written by George Smith, which do communications (send and receive) from a bunch of different I/O boards. Communications is a tough area of computing. Besides having to understand I/O chip operation in general, one has to understand a half dozen varieties of I/O chips, different ways of interfacing them into systems, and program around all the timing problems, too. This is a great set of software which all system hackers will want to have.

Space doesn't permit me to list the contents, but there are also: CP/M Disk 1901, Miscellaneous Games, and Utilities; CP/M Disk 1902, Forth; MDOS Disk 900 and 901, Miscellaneous Games, Utilities, and Applications, and MDOS Disk 902, an AMWAY dealers accounting system.

MUG CP/M Library Disk 1900, Revision 00, FEB 83  
COMMUNICATION  
All by George Smith

NAME	TYP	RV	SZE	CAT	NAME	TYP	RV	SZE	CAT
APL-DNLD	ASM	00	008		MEM-IN	I/O	00	002	
APL-DNLD	HEX	00	002		MEM-OUT	HEX	00	002	
APL-TRM	COM	00	002		MEM-OUT	I/O	00	002	
APL-UPLD	ASM	00	004		MORACO	I/O	00	004	
APL-UPLD	COM	00	002		MORR-MIO	I/O	00	006	
APL-UPLD	HEX	00	002		OSBORNE	I/O	00	004	
APPLE	DOC	00	002		PUN/RDR	I/O	00	006	
APPLE	I/O	00	004		RCV-IT	COM	00	002	
BITBUCKT	HEX	00	002		SAMPLE	I/O	00	004	
BITBUCKT	I/O	00	002		SND/RCV	DOC	00	014	
BITST-I	I/O	00	004		SNDIT	COM	00	004	
BITST-II	I/O	00	004		SUPR-RCV	ASM	00	018	
BUCKET	COM	00	004		SUPR-RCV	HEX	00	006	
CROMEMCO	I/O	00	006		SUPR-SND	ASM	00	030	
DUM-TERM	HEX	00	002		SUPR-SND	HEX	00	008	
HEATH	I/O	00	004		UPLOAD	COM	00	002	
KPHILIPS	I/O	00	004		ZSM	COM	00	008	
MEM-IN	HEX	00	002		ZSM	DOC	00	014	



# WORD PROCESSING — PLUS SPELLBINDER

A Word Processor for CP/M and MS-DOS Systems, with built-in mail list, sorts by zips, alpha and cues, forms generator, column addition and more.

<b>LIST</b>	<b>PLAN-A</b>	<b>PLAN-B</b>	<b>PLAN-C</b>
\$495	\$356	\$321	\$285

Other CP/M, MS-DOS and Apple software available with same terms. Write or call for full spec sheets or further information.

## PLANS:

- A - Phone support, exchange privilege, 90 days
- B - Phone support, exchange privilege, 30 days
- C - Support limited to supplied documentation, no exchange except for bad disk replacement.

Additional support available at \$20/hour.

## TERMS:

Prices include cash discount. Add 4% for charge or COD orders. Add \$5 shipping and handling.

Suite 14  
3322 Mem.  
Pkwy., S.W.



(205) 883-8113  
Huntsville,  
AL 35801

Master Programmer Pavel Broder Does It Again! NEW SUPER POWER! (version 1.3)  
PUTS YOU IN CONTROL OF CP/M. Now for CP/M 86 and MP/M 86, too.

"POWER IS A GREAT PROGRAM" - InfoWorld Software Review Nov 8/82

## POWER!

The first super program that puts you in control of CP/M.\*

POWER! works with CP/M or MP/M on any computer.

POWER! gives you complete control over CP/M!

Ever accidentally erased a file? POWER! restores erased files!

Ever fiddled with PIP in copying files? POWER! replaces PIP and is faster and easier. You simply pick files to be copied from a numbered menu. POWER! feeds the names to CP/M for you - no need to type file names, no typing errors. ever!

Tired of CP/M's scrolling through text files? POWER! goes through files for you, page by page, file by file, or line by line with instant halt at your finger tip.

Ever lost data on a glitched disk? POWER! tests disks and fixes glitched disks.

Damaged Directory? POWER! allows you to repair the directory!

Afraid of HEX numbers? POWER! automatically converts HEX to DECIMAL, BINARY & ASCII.

Need to patch or change a program? POWER! searches memory, displays memory, and lets you change memory wherever you want.

Want to locate a file? POWER! sorts the directory, searches all disks or all user areas automatically for files for you.

Annoyed at having to keep a system disk in Drive A? POWER! doesn't require a system disk in any drive.

Renamed a file using = and all that typing? POWER! lets you pick files from a numbered menu and prompts for every action.

Ever accidentally overwritten a file? POWER! checks first and asks permission.

Need to manipulate data on a disk? POWER! reads and writes any track or sector independently.

Ever make a mistake in the DDT? POWER! loads disk data to ANY memory address, not just 100, and writes to the disk from any memory address. POWER! Single-Steps through memory, moves memory compares memory sectors, tests memory, allows you to change memory and saves to disk using Decimal numbers.

NOW POWER! permits you to securely lock any file with your password to protect sensitive information from prying eyes. PASSWORD program included FREE with every POWER! order.

Hate DOS errors? POWER! ends DOS errors, and gives you a way out.

Trouble identifying files? POWER! marks original files and their copies for you. POWER! also compares files and finds identical copies regardless of name.

Can't remember odd file or program name abbreviations? POWER! lets you deal with disk files by number. Never type or mistype file names again.

POWER! does more...NEW version of over 55 command utility programs is the only CP/M housekeep or you will ever need to really get control of your computer. A great buy, too, at less than \$2.75 each.

## MORE THAN

55  
CP/M®  
UTILITIES  
IN ONE  
PACKAGE

ONLY \$149 (\$2.75 EA. UTILITY)

POWER! frees your disk space since it takes less than 15K.

POWER! versions for CP/M or MP/M on any computer.

TRY IT ON US!  
MONEY BACK GUARANTEE



### JOHN OTHER POWER USERS:

E I Dupont	Kerox Corp	AMF
Sperry Univac	Conn Gen Life	Syracuse Univ
NY Stock Exchange	Princeton Univ	Olivetti
Livermore Labs	ITT	New Mexico State
Union Carbide	Dow Chemical	Monsanto Chemical
UC Berkeley	Advanced Logic Sys	Univ Minnesota
UC San Francisco	Charleston Univ	US Dynamics
Bendis Corp	Univ Helsinki	City Bank
Fort Motor Co	Honeywell	

LIST	PLAN-A	PLAN-B	PLAN-C
\$149	\$159	\$140	\$124

SEE EXPLANATION OF PLANS AND TERMS ON REVERSE SIDE OF PAGE.

DAMAN Suite 14, 3322 Memorial Parkway, S.W., Huntsville, AL 35801 (205) 883-8113  
Computer Programs • Systems Enhancements • Data Management • Support Services

Listed to the right is a set of Super-Soft software which is on sale through February 28. If you'd like further information, call or write for fliers on any of the packages.

POWER, AND SPELLBINDER

See the ads for Power and Spellbinder on the reverse side of this sheet. I'm pleased to be able to offer the award winning POWER Utility Package. Spellbinder has always been my favorite word processor, as well as it being the best value.

PRICES DETERMINED BY SUPPORT

Note our new Plan structure. If you know what the package is, and don't need any help, buy under Plan C. If you aren't sure it's going to do the job, use Plan A or B. The exchange privilege means you can turn the package back in for one that does a similar function. That means you can trade one financial spread-sheet for another, but can't trade a spread-sheet for a word processor. Any price difference will be computed on the current price of the alternate product under the same Plan (A or B). Any documentation must be in "new" condition, of course.

ASK FOR A QUOTE

One of the problems in retailing software is being able to support all the packages. Some amount of money must be included in a retail price for that support.

DAMAN's new PLAN-C is intended to ease my problem. I don't necessarily need to understand software sold under PLAN-C. So, if you have a software package in mind, and if you don't need support, call for a price quote.

I can get most software. I can only offer PLANs A and B on a select group of programs which I own and can operate, so that I can find answers to buyer's questions.

	LIST	PLAN-A	PLAN-B	PLAN-C
***LANGUAGES				
BASIC COMPILER	200	160	145	129
C	250	199	180	161
C with BCD	400	318	287	256
BCD	300	239	216	193
FORTH	200	160	145	129
FORTRAN	275	219	197	176
RATFOR	100	79	71	64
FORTRAN/RATFOR	350	277	251	224
LISP	150	121	109	98
TINY PASCAL	85	69	62	56
***BUSINESS				
DATA VIEW	200	160	145	129
INVESTMENT TAXPAC	200	160	145	129
OPTIMIZER	200	160	145	129
SCRATCH-PAD	295	234	211	189
STAT-GRAPH	200	160	145	129
***WORD PROCESSING				
CORRECTOR	200	160	145	129
FFS	85	69	62	56
STAR EDIT	225	180	162	145
***UTILITIES				
DIAGNOSTICS II	125	98	89	79
DISK DOCTOR	100	79	71	64
DISK EDIT	100	79	71	64
ENCODE/DECODE	100	79	71	64
SYSTEM CHECKER	75	60	54	48
UTILITIES I	60	48	43	39
UTILITIES II	60	48	43	39
***GAMES				
ANALIZA II	50	40	36	33
DUNGEON MASTER	40	33	29	26
NEMESIS	45	36	33	29

\*\*\*\*\* PLEASE NOTE \*\*\*\*\*

All software available in most CP/M formats, as well as for MS-DOS (IBM PC) and CPM/86.

PLANS:

- A - Phone support, exchange privilege, 90 days.
- B - Phone support, exchange privilege, 30 days.
- C - Support limited to supplied documentation, no exchange except for bad disk replacement.

Additional support available at \$20/hour.

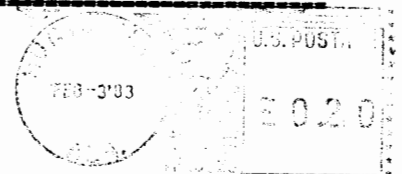
TERMS:

Prices include cash discount. Add 4% for charge or COD orders. VISA and Master Card accepted. For shipping and handling, add \$4 first piece, \$3 each additional (North America).

FIRST CLASS MAIL

FIRST CLASS MAIL

DAMAN  
Suite 14, 3322 S. Memorial Pkwy.  
Huntsville AL 35801  
(205) 883-8113



FIRST CLASS MAIL