THE D/FW DATA EXPANSION
TIMEX/SINCLAIR/AMSTRAD METROPLEX NEWSLETTER
COVERING: TIMEX, CP/M, AND MS-DOS
VOL 5, NUMBER 3<>MAR. 1988
EDITOR: DAVID BAULCH
CO-EDITOR: JOHN BATTEY

NOTES FROM THE EDITOR

I have become aware in the last few months of the demise of the Timex computers. I know that it truly took place about three years ago, but it seems to have risen again. I also know that I have commented about this before, but bear with me as I comment on it again.

We have had Sinclairists, Timexers, and Amstraders all moving away from 8-bit machines and to MS-DOS and 16-bit machines - not necessarily an Amstrad. With these supposed 'diehards' leaving the 8-bit world for greener pastures, what type of a future do we have for Timex? Other 'diehards' maintain there is life for our computers, yet! Who is right?

We now have the capability of having a universally connected disk system with the LK-DOS cartridge by Larry Kenny. More desktop publishing programs and improvements are forthcoming from Stan Lemke. Other operating systems other than BASIC are available, as CP/M from AERCO and ZEBRA (even though ZEBRA has discontinued the drive system). There are even more peripherals coming, larger RAM (256K) and more. Does this sound disparaging?

Yes, many user groups are beginning to fall on hard times, our own group included, but others are increasing as computers are being recycled to other people. The more that people predict that we are finally going under, the more the REAL diehards cling to their computers refusing to give up. I hope that this topic has finally been exhausted by now. I would like to have an MS-DOS machine, but not at the expense of giving up my 2068. What is wrong with it? I have an excellent disk system, a modem that has given me no problems, an excellent word processor (with improved versions still-to-come), a handy spreadsheet, graphics programs, databases, etc., all of with which I am familiar. Everything works, so why should I change to something different when there is no need?

I will get off my soapbox now and return to the work of the newsletter. - David

This newsletter is published monthly and is sent to all of our subscribers and to other user groups throughout the U.S., Canada, Mexico, and Europe. The monthly printing is about 125 copies. The Timex Sub-Board on the FWKUG MBBS & PSDE [8/1/N], (817 or 214) 540-4183, carries a number of TS 2068 and Spectrum downloads and is up 24 hrs. The FWKUG MBBS also carries a large

number of RP/M (CP/M 2.2) and MS-DOS downloads. The subscription rates for the newsletter are: $2.50 a month or $12.00 for six months. Anyone wishing to advertise, our rates are: $5.00 (minimum) for 1/4 page, $10.00 for 1/2 page, and $20.00 for a full page. Please send to: David Baulch, 4424 Geddes Avenue, Fort Worth, TX 76107.

FORT WORTH

PRESIDENT:CHUCK DAWSON
VICE-PRESIDENT:FRANK BOULDIN
TREASURER:ELLIS SAUNDERS
SECRETARY:FRED STOCKTON
EDITOR:DAVID BAULCH
PROGRAM EDITOR:CHUCK DAWSON
TECHNICAL EDITOR:FRANK BOULDIN

DALLAS

CHAIRMAN:GLYNN HARRIS
DDC ALTERNATE:JIM LYONS
TREASURER:ARTHUR EMERSON
CO-EDITOR:JOHN BATTEY

****

MEETING NOTES-FORT WORTH

The meeting convened in MEETING ROOM 'A' of the Fort worth Public Library at 300 Taylor Street in downtown Fort Worth. The next meeting is scheduled for April 9, 1988 in MEETING ROOM 'A'. The meetings are on the second Saturday of every month from 1:00 P.M. to 4:00 P.M. The meetings are open to all who wish to learn more about Sinclair, Timex, and Amstrad computers. This includes anyone using any type of disk system with their computers, or other computer languages. Any visitors are welcome to attend. Any other computer operator/owners are more than welcome as there are many different types of computers, peripherals, and programming languages that get discussed.

We had a very small turnout, possibly because this was the first time in a while that the weather was so nice. Ellis Saunders was unable to make the meeting because of a prior commitment. Others have just been lax in attendance. Chuck Dawson started us on the discussion section of the meeting anyway.

Fred Stockton has not gotten his 2068 set up as yet. He has finally figured out where it is going to go and it will not interfere with his wife's sewing. Before Fred gets it up where he wants it, he says he has to finish his income tax preparation first. (I can understand that myself-but I do use my 2068 for getting all of my costs and purchases, etc. together using Omnicalc2.) He says that the forms and the booklets are very confusing, as we all know, but he is getting closer to the end of completing the process. Once that is complete, he will set up his computer and get back to learning how it differs from his 'old' 1000. Fred Also told us that he was the treasurer for the senior citizens club in this area. He described some of the things necessary to carry on with his job. I think he just might like to keep track of all that with his computer, once he gets it all set up.

Frank Bouldin was out for ten days with the flu, but is finally getting over the last vestages of it. He also had a house guest for a while, his niece from Los Angeles. She wanted to go with Frank to one of those Friday luncheons where, according to Frank, the old 'fogeys' get together and talk computers and flying. Flabbergasted, Frank took her and she really enjoyed herself. She is quite intelligent and was able to 'hold' her own in the conversations. He has been spending about an hour a day with the satellite hook-up. He has also been keeping up a long-distance conversation with his brother-in-law, a priest, in St. Louis using StarText. His brother-in-law still is having trouble with his Tandy 1000. He cannot get the backspace key to work while he is on StarText. He might be typing a message, make an error, and cannot backup to erase it. StarText says it is Tandy and Tandy says it is StarText. Chuck suggested that he get a printout of all his keys using the SHOW.ALL keyword. If that was no help, try using the SHOW.CODE keyword. This allows you to press a key and see what the ASCII code for that key. If it is the right one, then there is a configuration problem. If it is not, then you can change it to the proper code.

Chuck has not been doing a great deal with his computers. He is using them at work, but has not been doing anything spectacular. He has been doing some downloading from many boards around the area. One that he has gotten a good bit of freeware from is the FWKUG MBBS & PSDE. Frank said that he has not done any downloading as yet, but will probably start checking it out to see what kind of software is available.

I have been downloading some software myself from the Kaypro board, but mine has been CP/M and not MS-DOS. I have quite a collection of Free software now, but some of it is not quite as useful as I expected it to be. Many times the programs require that other programs be present to work, like dBASE II. I may have to break down and write to Aston-Tate, MicroSoft and a few others to see if they still have some of these programs available in a Morrow format and what is the price. I also had to thank Chuck again for his help with the OVERLOADER 1.2 program. It seems amazing to me that someone can give me so much help with a system that he does not have. Now with my new-found knowledge, I have been changing numerous Timex commercial programs to work on the disk system. I also brought a number of the new newsletters for anyone to see and read. I did run into a new problem that I cannot figure out as yet. When I format a new disk, the system asks what type of system - Timex, Spectrum, or CP/M. I switched the toggle in the back to Spectrum and turned on the computer. The Sinclair sign came on, but the keyboard was locked up tight. I hope that I can figure out what is wrong. I tried typing in a few other programs for the Timex, but was having a terrible time in getting them to work properly. It is time to start 'debugging' again. This did not take long, but we took a short break before we came back to talk about the future of the newsletter and the user group.

We came back to quickly talk about the newsletter and the dues. We were running out of money, and with few new members

coming in, there had to be a change. The cost of printing was too high. This was met by deleting many people that were getting complimentary issues and other user groups that were not giving as good as they were getting in return. I then checked out the cost of going back to Xeroxing instead of using the school print shop. Now we are going to be back to where we should be on price. This will help greatly. We also talked about beginning to meet at each others houses, again or possibly for lunch somewhere and have the meetings there. We all agreed that it would be a nice change. We broke up the meeting a little earlier than usual.

****

AMSTRAD GETTING STRONGER

When I picked up the MARCH 1988 issue of the COMPUTER SHOPPER from my mailbox, What did I see on the cover but the new AMSTRAD PC 1640 and a banner proclaiming 'SIR CLIVE RETURNS WITH NEW LAPTOP'. Of course, the first thing I read was the review of the Amstrad 1640. Reviewer Rich Fields was, on the whole, very pleased with the new Amstrad model. He was a little down on the monitor - you still can only use the one provided by Amstrad and a lot down on the user's manual, but he did give good comments to many of the other things about one of the newest computers in the Amstrad line.

I then began to glance through the magazine on my way to reading about the new Z88 laptop computer. On my way to the review I saw a number of advertisements for the Amstrad 1512SDM and 1512SDC and the Amstrad PPC640 'Laptop' computer from Computer Warehouse in Augusta, Georgia. It seems that Alan Sugar of Amstrad is beginning to get to the American public in a big way.

The Amstrad 1640HD uses a more powerful version of the 8086 CPU, a true 16-bit chip running at 8MHz, and has the full compliment of 640K RAM. It even has a socket for an 8087 math co-processor to be installed. A very exceptional unit that measures up quite well on benchmarks against other like clones. It also costs less. The Amstrad PPC640 has 2-3.5 in 720K drives, 2400 baud modem, Super twist screen, three power modes, a 101-key enhanced keyboard, and much more. I, personally, believe that these new computers will catch on here in the US relatively fast, just keep your eyes open and see!

****

ASK CHUCK
By Chuck Dawson

QUESTION: I know about the HEX$ function in GWBASIC, which returns the hex value of a decimal argument. But what about going the other way? I want to start out with a hex value and end up with decimal. There is no DEC$ function, is there?

ANSWER: No, there isn't, but there is an easy way to do the same thing. Let's assume your hex value is in A$. Then Decimal=VAL('&H'+A$) This concatenates (add strings) the symbols

'&H' and the contents of A$ (for example, 2C) which would yield '&H2C'. Now, taking the VAL of this string gives the decimal value.

QUESTION: From within a BASICA program, I want to use the SHELL command to execute a DOS command. When I run the program, however, it does not work. I have checked the program over and over but cannot find any problem. Have you ever had the SHELL command fail?

ANSWER: It should not fail if there are no progam errors and a copy of COMMAND.COM is available in the current directory for loading. This second copy of COMMAND.COM (the first is the one loaded at boot-up) is necessary and if not found, will cause the command to fail. If you are operating from floppy drives, just copy COMMAND.COM to your BASICA (or GWBASIC) disk and try the program again. If you are operating from a hard disk, copying COMMAND.COM to the current directory will work, or just use the PATH command to tell the system where to look for COMMAND.COM if it is not found in the current directory.

QUESTION: I have a program that was written in BASICA and I want to run it using GWBASIC. It stops with an error at a BEEP command. Can I do anything other than deleting the line?

ANSWER: GWBASIC does not use the BEEP command. However, substituting PRINT CHR$(7) will give the same results.

QUESTION: I always get nervous when using the DEL command, especially when using a wild card. I am afraid I will delete more than I intended. Any suggestions?

ANSWER: Yes, this trick will let you have a 'dry run' to see what files will be affected before you actually delete. Let's say you want to delete all files that end in .DOC so you first enter DIR *.DOC to see the list of files. If it looks OK and you want to proceed, just enter DEL *.DOC and those files you just looked at will be deleted.

****

MBASIC TIPS

By David Haldeman

Reproduced from the Cincinnati Osborne Group's newsletter, COGWheels

Here are two hints and two routines which you might find useful.

HINT #1: If you need to print a whole line of one character, use the STRING$ function. It saves a lot of typing. It works like this:

If you want a whole line of asterisks (look it up in your ASCII table in the manual) which is ASCII code 42, just say (for 52 character screen display):

```
10 PRINT STRING$(52,42)
```

where 52 means print 52 characters with the ASCII code 42. You can print anything you like.

If you want to scroll of the screen, instead of a whole string of PRINT statements, just issue a:

```
10 PRINT STRING$(24,10)
```

where 10 is the ASCII character for a line feed. You can use it for anything, and it can be used in a graphics mode if you want fancy screen decorations. (Character 127 is especially nice as a shady character on some computers.)

HINT #2: Avoid using the CLEAR statement within a GOSUB routine. The statement will clear memory so well that the computer will forget where it is supposed to RETURN to!

CLEAR should be used with caution.

****

MAILBAG-EDITOR'S MAIL

I received some interesting things in the mailbag this week. I have been on the Kaypro board keeping in contact with our old friend Charles Stelding and Ed Grey. I had asked Charles if the AERCO system would really READ a Morrow MD3 format as it is advertised to do. It is nice to have someone who has alreay been through this. Ed Grey has been 'missing' from the mailbag for a while, but he brings some interesting news. I also answered a message about dBASE II and some books that I had seen at Lolir #5, here in Fort Worth. The answer I got back was from Norman Peavy and Jim Coburn. Jim, in particular, was the originator of the message. I also suggested to him, since he was working a DOS version of dBASE III+ that he might like to try one of the fieldless databases, Findex or DataFax. His reply was interesting. I would like to see his program at work.

[][][][][][][][][]

Msg#: 174 Date: 02/26/88
From: CHARLES STELDING (Private)
To: DAVID BAULCH
Re: MORROW MD3 DISKS

David,

Yes, Morrow MD3 disks will work but you must do something to add the system disk to each disk so that it will boot up. At the moment I can't remember the proper procedure. It has been quite a while since I have used the system. I'll look it up and let you know.

Thanks for the message!

[][][][][][][][][]

Msg#: 180 Date: 02/26/88
From: ED GREY (Private)

To: DAVID BAULCH
Re: PRESS RELEASE

Hello David,

I hope that all is well with you. I have uploaded a press release (in the T/S file area) for your publication. It is called PRESS.WS. Take care.

Ed Grey

P.S. Hmm, I almost forgot. I have my own BBS up and running now. It is called Grey Matter BBS & RCP/M (213-971-6260). Naturally there is T/S file section.

[][][][][][][][]

Msg#: 181 Date: 02/26/88
From: JIM COBURN
To: DAVID BAULCH
Re: DBASE II

I've been using Condor DBMS for 5+ yrs with no problems since I can write interstructered programs from within Condor with Menu's that append to other components or programs (user written) from Condor. Plus it has the ability to create data files in 5+ formats for transfer to w/star or supercalc and reversed also. Their Dos version is nice since you get a 2nd page of fields to also create. The program never caught on in Texas because of lack of representation by Condor but the Government uses the program tremendously because of its flexibility. There are about 28 separate commands i.e append , join, sort, define up to 32 fields, calculate, write ASCII files on sorted fields, delete etc. Would be nice to have a Dos based machine but I never found any reasons to switch. Once wrote a complete program for my ex that had 28 menus, 10 choices per menu that automatically brought up 280 prgms with 42 fields and the list goes on. inter language is English.

****

JRT PASCAL-FINALLY> A PROGRAM

By David Baulch

When writing in Pascal, you should use what is called the 'top-down' design method using modular building blacks. The top-down method is building your programs in modules using levels of work. It is difficult to explain, but the method looks something like this:

(1) Analyze the problem
(2) Write the main module
(3) Write the remaining modules
(4) Resequence and revise as necessary

So, following the proscribed plan, the first thing we need to do is to understand what we want to accomplish. Whe must know what is given - the input, and what is required - the output. List any assumptions that you might have and, from all of this

information, develop an overall algorithm. When this is accomplished, begin writing as 'English' code for your main module. This will eventually be translated into Pascal. If this module is longer than fifteen statements, go back and try again. You are at a level too low to write a competent main module. There is no law stating how many levels you must have, but a good 'rule of thumb' is to start with a main module of fifteen or less statements. If the main module can be refined later providing more clarity and finer details, do so, but at a later time. This is modular programming where you can change the module without having to re-write the entire program.

Now that we have something to work from, we need to figure out a program to write. Being a high school teacher, the most valuable program, to me, is the T/S GRADER program by Robert Fischer. I have modified his program, somewhat, to run on a disk system using a 'dual' save making one a back-up copy. I added the 80 column printer driver, and all works quite well. However, ther are certain things I want it todo, but ther is no room in the code. Even if I were to use disk access for each class, this taking more time than the original, I could have the program do more and contain more information PLUS a full 80 column printout without using a COPY statement as in the Timex version. This seems to be a perfect candidate for a Pascal program.

The first thing that the program needs is a menu module that will tell all of the things that the program will do. One of the things it needs to do is to actually start by writing classes. I think it would be a good idea to have a sort module to alphabetize the names, but that can come later. It would be nice if it would 'weight' certain grades, have a 'search' mode, and a number of other things that I believe would be handy. Well, figuring out exactly what I want in the first module may not be so easy after all. Maybe I should follow the procedure like I am suppose to and write down input and output. This may take some time. Well, we have enough to start with so next time I guess I will begin figuring out the actual MENU section which will be the MAIN MODULE of the program. From that module I should be able to figure out all the other modules that will be necessary for the conversions to grades, the names and classes, the weighting of the grades, a quick sort, an alphabetical arrangement, etc. You get the idea. If any of you get any more ideas, please send them to me. I might like to add them in, just add in another module and quickly re-do the MAIN MODULE.

****

TS/1000 VERIFY

By David Nowotnik
From Nov/Dec 87 RAMTOP of Greater Cleveland

With only 8K of ROM in the TS/1000 it's little wonder that it hasn't a verify command. This little program will take care of that omission.

This routine is based on the LOAD routine in ROM. The change is when a byte is read off the tape. Instead of putting

the byte into the approprite place in RAM, it is compared with the current byte at that address. If their is not a match, then the routine exits with an error message (R/O).

If all bytes match and the verification was successful, with no mis-match, then the 0/0 message will be returned at the end of the routine.

First, enter this machine code loading routine, with 135 spaces or characters in the REM statement.

```
10 REM ........(135 SPACES).....
20 LET X=16514
30 INPUT A$
40 IF A$="S" THEN STOP
50 LET J=16*(CODE A$-28)+CODE A$(2)-28
60 IF PEEK X=27 THEN POKE X,J
70 LET X=X+1
80 GO TO 30
```

Next, RUN the program and enter these HEX digits:

```
CD 23 0F 37 11 00 00 CB 12 CB
0A CD 10 7C 18 FB 0E 01 06 00
3E 7F DB FE D3 FF 1F 30 49 17
17 38 28 10 F1 F1 BA D2 7A 7C
62 6B CD 10 7C CB 7A 79 20 03
BE 20 D6 23 17 30 F1 FD 34 15
21 09 40 50 CD 10 7C 00 CD 6C
7C 18 F6 D5 1E 94 06 1A 1D DB
FE 17 CB 7B 7B 38 F5 10 F5 D1
20 04 FE 56 30 B2 3F CB 11 30
AD C9 7A A7 28 BB CF 0C EB 21
7C 40 37 ED 52 30 06 1A B9 28
02 CF 1A 13 2A 14 40 37 ED 52
EB D0
```

Now the routine is in the long REM statement, and you can delete lines 20 through 80. DO NOT delete line 10. After these lines are deleted, add the next lines, 20 through 70. Then SAVE this before you go further, it is your VERIFY program.

```
20 LER X=16514
30 FOR I=31744 TO 31878
40 POKE I,PEEK X
50 LET X=X+1
60 NEXT I
70 NEW
```

To use the VERIFY routine, it must be loaded into your TS/1000, above RAM-TOP, before any other program. First, lower RAM-TOP with these three direct commands:

```
POKE 16388,123 -- ENTER
POKE 16389,255 -- ENTER
```

Now you may LOAD and RUN your VERFIY routine.

You now can type in your BASIC program. When you are ready to SAVE it, just SAVE it to tape as you would normally SAVE it. To VERIFY, rewind the tape to the start of the program, and type in, in direct command:

RAND USR 31744

And press play on your recorder and ENTER on your computer.

If you saved the program with variables, the CLEARed them before verifying, or changed the variables in any way, then you may get a verify error (R/O). Otherwise, if all is well, you'll get an 0/0 message to tell you that your program has been VERIFIED.

****

THE Z280 CHIP

By Michael Rulison

Reproduced from the ROCC (Raleigh Other Computer Club) newsletter

CP/M is dead; long live CP/M.

Just as the funeral peroration for CP/M was fading in our ears, ther came a small jubliant horn toot out ot the south-west and west (Dallas and Loas Altos) reporting sightings of unusual news from Winnipeg about products from Redding and CAmbell (California, you all!).

To wit: Zilog, Inc., has (will?) release(d) the Z280 chip: a super extension of the Z80, 40+ new instructions, clock rates exceeding 10 megaherts, 16 Megabytes of memory space addressable, 8-bit Z80 bus or 16-bit Zilog Z-bus. A true 16-bit chip, but runs Z80 code.

This chip is the basis of the Ultraboard retrofit for the Kaypro Z80 CP/M systems (1, 2-64, 4-84, 2X, 10) that provides 1-16 Mbytes of RAM, 12 Mhz instead of 4 Mhz clock speed, RAM disk partition, cache memory for approximately 256 bytes of recently-used instructions or data, and many other features of a technical nature.

The Ultraboard includes a new screen driver to keep up with the Z280, and RGB connector for external color monitor and 1 meg of ROM capacity, and HANDYMAN, a desk top utility. Ultraboard will support multi-tasking and networking - when an appropriate operating system is developed. Price should be about $500.

Although some of the above may be vaporware, if it proves correct, the Z280 may make a CP/M machine comparable to an XT clone. Certainly something to cogitate on!

Ultraboard: High Tech Research, 1135 Pine St., #107, Redding CA 96001, 800/446-3220 (outside CA).

Z280: Messrs: Magill, Davies, or Hampton, Product and Technical Marketing, Zilog, Inc., 210 Hacienda Ave. Campbell CA

95008. 408/370-8000 or 5166.
****

CONVERSATION-MAIL BAG
David Baulch/Chuck Dawson

I read the 'new' ASK CHUCK column for this month. I was curious about the DEL.EXE or ERA.EXE which is similar to my DEL.COM and ERA. I, too can call up only certain files on the directory if I ask for those. Then I can erase them all at one time, or one at a time if there are programs that I wish to keep among those I wish to delete. However, in one particular program that I use, albeit infrequently, I can also DELETE a number of programs, but I can add an extra command in using it to request that I confirm each deletion before the system actually deletes that particular file. If I do not place that extra command when I call the program, it will delete or erase all those programs. This can be a mixed blessing. If it always did this, the program would take too long to function properly for speed. Since I can add it if necessary, it is a boon for dispensing with unnecessary files. Chuck sent me his answer, but also gave me some intereesting information about the Timex version for the Zebra FDD that I was not familiar with.

Date: 2/19/88 Time: 12:21 PM
From: Chuck Dawson -- Mail Code: 4579

David,

In MS-DOS, you can use 'ERASE' or 'DEL' interchangably. I just use DEL because it is less letters to type. No, in MS-DOS you do not get a chance to look at each one before it is erased. In Timex Operating System, you can. Guess it is better.

[If you would like to hear about more of the differences between MS-DOS and CP/M and how they match up with the Timex systems-either the Zebra FDD (Chuck's system) or the AERCO (my system) please let me know and we will be glad to oblige.]
****

FURTHER REALIZATIONS ABOUT CP/M

During this month I have, upon occasion, found myself coming across some problems and not being able to figure them out. But, little by little, I am using my computer and learning a bit more about using the CP/M disk system. I had asked Charles Stelding and Chuck Dawson about how you go adding the system tracks to the PD software that you receive. I had to wait for an answer, but before I received it, I had figured it out myself - by reading the FILES.DOC that comes with the boot disk. The reason that I thought about it, one of the articles in the new FOGHORN for CP/M was about a guy that did the exact same thing that I would have done. Placed the software in the computer and tried to boot it. Without the system image on the disk, this would not be possible.

CP/M people use SYSGEN.COM, which I do not have. But I do have two other programs, (1) GETSYS.COM and (2) PUTSYS.COM. You need to use the GETSYS first, bringing up the 'system image' from the A drive (usually) into the TPA. Now that the 'image' is there, call the PUTSYS to place it on the appropriate disk in the specified drive, in my case, the B drive. Now that I had figured it out, I formatted a disk and left off the system tracks - on purpose! I used NWSP.COM, a program I have mentioned before (love that program!), and placed a large number of files on that disk. Then I went through the above procedure, reset the system, and attempted to boot the new disk. No problem!

Little by little, I am learning the subtleties of operating in CP/M, a fascinating operating system. I am going to attempt to 'modify' the operating system with a program called ZBDOS that 'improves' the BDOS and I/O. I think I can afford 'one' disk to 'trash' if I find out that this will not work. I do not agree with a number of other people that shall remain nameless, that CP/M is not only dead, but it should never have been born in the first place. Just because it is not graphics intensive does not mean it is worthless. For a graphics machine, MS-DOS is probably better, but for me - CP/M is just fine.

****

Ed Grey Enterprises
P.O. Box #2186
Inglewood, CA 90305
(213) 759-7406 voice
Grey Matter BBS (213) 971-6260

Press Release: February, 1988

The sales and service formally offered by Grey & Clifford COMPUTER PRODUCTS, to the Timex/SINCLAIR community, is going to continue under the name of Ed Grey Enterprises. Not much will change, except there is a new BBS online to support our customers. The Grey Matter BBS & RCP/M (213-971-6260) started on January 24, 1988. It operates 24 hours a day, 8 data bits/ 1 STOP bit/ no parity, and it runs at 300/1200 and 2400 bps. Dave Clifford's TIME-=X=-CHANGE will continue to be online, but orders and request for sales information should now be directed to the Grey Matter BBS (phone number and mailing address have not changed). Although I hope to expand the product line with new and innovative Timex products, I will also offer many non-Timex specific products. Catalogs are FREE, just ask by any of the methods mentioned above.

ANNOUNCING.....The Z-SI/O Bare Board Package:

The Z-SI/O RS-232c Serial Interface Card has been recognized as the best RS-232 I/F available for your T/S2068 computer. The assembled and tested version of the Z-SI/O card was priced at $79.00, and although well worth that price, many were reluctant to spend that amount for the product. Now, Ed Grey Enterprises has a Z-SI/O Bare Board Package that includes the 2 needed boards plus complete (step by step) assembly instructions. This package will allow you to construct your

Z-SI/O as either a STOCK, JLO or AERCO version card. Most of the parts needed for contructing the card can be found on a 2050 modem card, especially the expensive MOTOROLA modem and modem filter chips. While supplies last, a 2050 modem card will be included FREE with each purchase. The price for this Z-SI/O Bare Board Package (with complete documentation and instructions) is $24.50 + $2.50 S&H. THE ASSEMBLED AND TESTED VERSION OF THE Z-SI/O CARD IS NO LONGER AVAILABLE.

It seems that our announced release of the TMX-64 BBS software was a bit premature. Even though developement of this software continues, unexpected problems have delayed its completion. I have refunded all money (with my apology) to those who sent prepaid orders for TMX-64 BBS software. Send name and address if you want to be added to the 'inform when ready' list.

****

DIRECTORY TRACK READER

By Michael C. Finn

Vol.4, No.1, TDM, pg.11

The following program was written on an old model Zebra Disk Drive. Owners of the newer FDD 3000 may wish to run this proram and report on the results. I don't know whether there are any software differences between the two models.

The disk drive controller formats the 3 inch disk into 40 tracks, numbered 0 to 39, each containing 4K bytes. Each track is subdivided into 16 sectors, numbered 0 to 15, each containing 1/4K or 256 bytes. The first four tracks (0 to 3) are reserved for the operating system. Track 4 contains the directory. Tracks 5 through 39 contain the files you saved to disk. When a disk is intially formatted, TOS sets up 40 tracks, writes a copy of TOS to tracks 0 through 3, palces the disk name in the first sector of track 4 and sets all unused bytes to 0E5h (229d).

The Zebra Disk Drive Technical Manual remarks that the contents of the directory track can be read using the extended Basic command INPUT *#0. After some trial and error, I found a suitable method to read this track using a Basic program.

I numbered this program beginning with line 200. You may wish to merge this program with a utility program which reads disk headers. One such program is Chuck Dawson's DISKREAD, which was published in Vol.1, No.1, of the (now defunct) T.O.P.S. newsletter. DISKREAD, or a similar utility, could be used to obtain further details on the directory contents.

```
 200 REM DIRECTORY READER By Mike Finn
 205 PRINT AT 10,0;"SELECT PRINT OUT DEVICE :"''''"1  TV
SCREEN"'''"2  PRINTER"
 208  INPUT LINE Z$: IF Z$<"1" OR Z$>"2" THEN GO TO 208
 210 LET Z=1+VAL Z$
 215 CLS : CAT *
 220 PRINT #Z'''"THIS PROGRAM WILL ITEMIZE THE  ENTRIES ON TRACK
4 OF THE DISK."
```

```
 225 PRINT #Z
 230 INPUT 'PRESS ENTER TO CONTINUE. '; LINE B$: CLS
 240 RESTORE *#0
 300 FOR I=1 TO 128
 310 REM INPUT *#0;A$;AT I
 320 IF CODE A$(1)=229 THEN PRINT #Z;'RECORD ';I;' IS EMPTY.':
GO TO 720
 325 PRINT #Z''RECORD: ';I
 330 PRINT #Z;'LEVEL: ';CODE A$(1);' ';'DISK NAME' AND (CODE
A$=255)
 340 PRINT #Z;'NAME: ';A$(2 TO 9)
 350 LET A=CODE A$(10): IF A>=128 THEN LET A$(10)=CHR$ (A-128):
PRINT #Z;TAB 6;'PROTECTED CATALOG ENTRY'
 360 LET A=CODE A$(11): IF A>=128 THEN LET A$(11)=CHR$ (A-128):
PRINT #Z;TAB 6;'INVISIBLE CATALOG ENTRY'
 370 PRINT #Z;'TYPE: ';A$(10 TO 12)
 380 IF A$(10 TO 12)='DIR' THEN PRINT #Z''THIS DIRECTORY
CONTAINS ALL     FILES AND SUBDIRECTORIES WITH   LEVEL NUMBER
';CODE A$(17);'.': GO TO 700
 390 PRINT #Z;'EXTENSION #: ';CODE A$(13)
 400 PRINT #Z;'BYTES LAST SECTOR: ';CODE A$(14)
 410 LET S=(256*CODE A$(15)+CODE A$(16))/2
 420 PRINT #Z;'# SECTORS WITH DATA: ';S
 430 PRINT #Z;'BLOCK ALLOCATION: '
 440 FOR J=17 TO 24
 450 PRINT #Z;TAB 6;CODE A$(J);TAB 16;CODE A$(J+8)
 460 NEXT J
 500 LET T=(S-1)*256+CODE A$(14)+(256 AND (CODE A$(14)=0))
 510 PRINT #Z;'TOTAL FILE BYTES: ';T
 520 LET A1=INT (S/4)
 530 LET A2=S-4*A1
 590 PRINT #Z;'UNUSED BYTES IN ALLOCATION: ';1024*(A1+(1 AND
(A2<>0)))-T
 600 LET A1=INT (CODE A$(17)/4)
 610 LET TRACK=4+A1
 620 LET A2=CODE A$(17)-4*A1
 630 IF A2=0 THEN LET A2=4
 640 LET SECTOR=16-4*A2
 650 PRINT #Z;'ALLOCATION BEGINS:''TAB 6;'TRACK ';TRACK;TAB
16;'SECTOR ';SECTOR
 700 INPUT 'PRESS ENTER OR KEYWORD STOP '; LINE B$: IF B$=CHR$
226 THEN GO TO 800
 710 CLS
 720 NEXT I
 800 RESTORE *#0
 810 STOP
```

****

MORE FROM THE EDITORS MAILBAG

These were inadvertantly left out from the original MAILBAG. I think they needed to be added in. I, again, am attempting to show many people how busy and informative the FWKUG MBBS & PSDE really is. When you have this type of a BBS in your area, I highly suggest that you patronize on a regular basis. These people, the SYSOP included, place a great deal of information and help at your disposal, all you need to do is

ASK?

Msg#: 196 Date: 03/02/88
From: JIM COBURN
To: DAVID BAULCH
Re: CONDOR

The only problem with programs written with condors inter language is that it only works with condor. The great feature is creating screens / menu's / helps /. The language is all in English, only 6 commands if/end/endif/etc. I was even able to get someone to enter in their name/password to allow the program to run or else it aborts back to DO> , I'm sure there is mucho stuff out there in DOS land, but who needs the price, never had any limits set on size of files. Once had a database of invoices that was 122K and, yes, I automatically had a backup copy of anything I did. It only took one power surge back in 81 to teach me that one.

[][][][][][][][]

Msg#: 211 Date: 03/06/88
From: CHARLES STELDING (Private)
To: DAVID BAULCH
Re: MORROW FORMAT

David,

To use the Morrow formatted disk, the instructions are on the original boot disk. I can't remember the exact instructions, but it should be there. If not, I can send you a copy of the instructions. Good luck. Sounds like you are using the CP/M to the max!

Charles Stelding
(Luckily I figured out how to do this before I received Charles' letter. All I had to do was to read the instructions that came on the FILES.DOC. Next time I will READ first and ASK second.)

[][][][][][][][]

Msg#: 212 Date: 03/07/88
From: ED GREY
To: DAVID BAULCH
Re: PBBS

David,

I am running PBBS software on a TeleVideo TS803h. There are many PD BBS software packages available for CP/M, I'm sure one could be made to run on the AERCO RP/M system. Take care.
Ed Grey
(I am highly tempted to try that myself, just for the fun of it. A real RCP/M BBS running from an AERCO dual-drive 2068 running RP/M.)

****

QL VAL STRING FUNCTION

By Robert D. Hartung

Vol.4, No.2, TDM, Pg. 32

A long time ago I saw a note by Mike de Sosa on the lack of the VAL sting finction on the QL. At the time I didn't have one, but witih the recent price cuts I've added a QL to my 'home for orphan computers'. Although I ordered one with software and QL User Guide, it came sans manual (later received), so I was digging through all my old Sinclair mags for info when I ran across Mike's comment again.

No doubt other users have found purely software approaches since then, although I have found no one published ones in my non-QL specific mags except the one Mike gave which utilizes Microdrive commands. The problem seems to be that while SuperBASIC does correctly interpret the value of an expression discretely entered as a program line, for example:

LET a$=2^3*PI or LET a=2^2*PI/C

It seems to 'see' only the first item (2) if the same expression is entered in response to INPUT a:

LET a$=a: PRINT a$/C

While it is a clumsy substitute for what can be done so routinely with 'lowly' TS1000/2068 BASIC, the following process will give the correct output for an expression entered as the definition of either a numeric variable or as a string variable definition without quotes or asterisks:

```
  10 expr=0: REM or expr$=0
  20 WHEN ERROR
  30 RETRY
  40 END WHEN
  50 CLS: CLS #0: PRINT 'expr = ';expr: PRINT 'Enter expression,
the r for result': EFIT 10: STOP
  60 DEFine PROCedure r
  70 RUN
  80 END DEFine r
```

****

RP/M UTILITIES

RPMPIP -CONTINUED

Archive Mode

The 'A' parameter puts RPMPIP into archive mode. For example,

RPMPIP C:=B:*.ASM[A]

will copy from the disk in drive B all .ASM files that have their archive attribute bit t3' reset to zero. After such source file is copied, the t3' attribute bit in the source file directory entry is set (1) to mark the file 'archived.'

In archive mode occurence of a full output disk receives special treatment. First, the output file is closed. Then the output file is renamed to indicate that it is only a portion of a file that must be divided onto two or more dismountable disks. After closing the output file, RPMPIP logs out the output drive, then asks for a fresh disk and continues the copy operation on the new disk. The renaming procedure is best illustrated by example. If the output disc becomes full while copying the file named SLMRAM.ASM, the output file is renamed SLMRA#01.ASM. The remainder of the file will be copied onto the new disc under the name SLMRA#02.ASM. This process may continue onto yet another disk under the name SLMRA#03.ASM, and so on, for as many output disks as required to copy all of the source file SLMRAM.ASM.

Archive Reload

Assuming that the example file SLMRAM.ASM was archived onto two floppy disks under the file names SLMRA#01.ASM and SLMRA#02.ASM, the command to reload the file, and thereby restore the file to the original state, would be

```
RPMPIP B:=SLMRAM.ASM=C:SLMRA#*.ASM[R2]
```

The 'R' parameter is followed by a decimal value stating the number of input files to be concatenated. When the end-of-information is reached on the file SLMTA#01.ASM, the program will ask for the disk containing the file SLMRA#02. ASM. This file concatenation process will continue for the number of disks stated in the R parameter.

Archive All

The 'B' parameter may be used to force all matching files to be archive copied regardless of the state of the archive attribute bit t3' in the source.file directory entry. For example

```
RPMPIP C:=B:*.ASM[AB]
```

will copy all .ASM files whether or not previously archived. Interactive selection of files to be archived may be achived by adding a 'C' parameter,

```
RPMPIP C:=B:*.ASM[ABC]
```

The program will ask for the confirmation before processing each file.

Concatenation

Files may be concatenated, or chained end-to-end, by a command such as

```
RPMPIP SLM.ASM=SLMMVM.ASM,B:LIB.ASM,C:DSD.ASM
```

which will combine three source files into one output file. Parameters may be imbedded in the source list. For example, the

command

```
RPMPIP T8S.HEX=T8SMOV.HEX[1:00 control Z], T8SAWT.HEX
```

will concatenate two intel format .HEX files and remove from the first file all the zero length records.

Ignore Strings

The general form of the 'I' parameter is Is where s is a character string ending in control-Z. The program will omit all characters starting with each occurence of the string s and extending through the next end of line (cr,lf).

Delete Columns

The 'D' parameter may be used to extract a stated midportion of each line of an Ascii file. For example,

```
RPMPIP COL3.TMP=REPORT.PRN[D35-45]
```

will copy eleven characters starting at character position 35 from each line of the file REPORT.PRN. The two decimal values seperated by '-' following the D parameter designate the starting and ending character positions. When only one decimal value follows the D parameter it designates the ending character position. For example,

```
RPMPIP CHOP.TMP=REPORT.PRN[D40]
```

will truncate all lines to 40 characters.

Console Echo

The 'E' parameter will cause transferred data to be echoed to the console display

Delete Formfeeds

The 'F' parameter causes formfeeds to be omitted from the output.
For example, the command

```
RPMPIP CHOP.TMP=REPORT.PRN[D40EF]
```

will truncate lines to 40 characters, ignore formfeeds, and echo to the console all data transferred to the output file.

Upper and Lower Case Translation

The 'L' parameter causes all alphabetic characters to be translated to lower case. The 'U' parameter causes all characters to be translated to upper case. If both are stated, parameter L prevails.

Clear Parity

The 'Z' parameter clears the parity, or high order, bit of each character copied.

Line Numbers

The 'N' parameter may be used to prefix a line number to each line of an Ascii file.

Object Code

When the filetype of the file being copied is .COM all line editing operations are disabled. Line editing operations can be disabled for any file by using the 'O' parameter. In the absence of an O parameter while copying a file that is not file type .COM, data transfer will terminate at the first control-Z (1A hex) encountered.

Start and Quit Strings

The start 'S' parameter and the quit 'Q' parameter may be used to extract a portion of an Ascii file. For example.

```
RPMPIP MMC.TMP=LIB.ASM[SMMC control ZQCCO control Z]
```

will copy all data starting at the first occurence of the string 'MMC' to, but not through, the next occurence of the string 'CCO'. In order to process lower case characters in start and quit strings, RPMPIP must be called in console mode.

Console Mode

When RPMPIP is called without arguments it comes up in console mode and displays '*' to prompt for input of an argument string. Case folding does not occur in console mode, so the command sequence

```
RPMPIP
*MMC.TMP=LIB.ASM[SMove control ZQout. control Z]
```

may be used to extract all text from 'Move' to 'out.'

Verify Disk Copy

Verification of a disk copy is invoked by the 'V' parameter. For example,

```
RPMPIP C:=B:CHART.*[V]
```

will transfer all files matching the ambiguous prototype and will verify each copy by reading back the data written to disk.

Printer

Destination device names LST: and PRN: may be used to cause the program to copy a file to the list device. For example,

```
RPMPIP PRN:=HELLO.PRN
```

is equivalent to

```
RPMPIP LST:=HELLO.PRN[NT8P60]
```

Lines will be numbered, tabs will be expanded spaced 8, and a formfeed will be emitted after each 60 lines. The command

```
RPMPIP LST:=HELLO.PRN
```

lists the file verbatim with tab characters ignored.

Page Skip

The 'P' parameter may include a page skip value. For example,

```
RPMPIP PRN:=HELLO.PRN[P60-3]
```

will set page size to 60 lines and begin printing at page 3.

Console

The console device CON: may be either a destination device or a source device. For example,

```
RPMPIP CON:=HELLO.ASM
```

will display the file name on the console.

Text files may be created by using CON; as a source device Following the command

```
RPMPIP MAILADR.IMP=CON:
```

the program will display thew character '_' to prompt for text lines from the console keyboard. Entry of control-Z will end input and the program will then write the data to disk.

Compare Files

Two disk files may be compared by a command such as

```
RPMPIP D:LEDGER.BAS::C:LEDGER.BAS
```

The general form of the compare command is

```
RPMPIP file1 :: file2
```

Compare operations involve no writing. The two files are read and compared byte-for-byte. If the two files are identical, the program will so state; otherwise the message

```
Files  MISMATCH
```

will be issued to the console. The E parameter causes mismatching records to be echoed to the console in both hex and

Ascii. The B parameter, when used with the 'E' parameter, causes all records to be echoed to the console.
****

CATXRF Rev. 1.2
by Simon J. Ewins,
676 Oriole Parkway,
Toronto, Ontario.
Phone: (416) 484-9427 >> Voice
(416) 484-9663 >> Data (occasionally)

Features of Rev. 1.2:
---

* Handles up to 2000 MAST.CAT entries.
* Handles any number of file extension types.
* Under 6K in size.
* Printer on/off option.
* Choice of 3 or 6 columns.
* Choice of normal/condensed type with Epson MX-80.
* Up to 9 space margin on hard-copy printouts.
* Assembler source code.
* Will run 'as-is' on most Z80 systems.
* Will run 'as-is' on ALL Osbornes.
* Optional disk file routing.
* ^S toggle of printout to start/stop.
* Uses currently logged drive.
* Auto-sequence to print sorted listings for all three options.

CATXRF will sort by:

1: Disk number.
2: File name.
3: File extension.

All three are internally alphabetized. The disk numbers are extracted from the .FRE files generated by the NCAT program at the head of the MAST.CAT file.

The listing is sent to the screen and the printer at the same time plus, if so desired, the option of sending the listing to a disk file is available.

A fourth option allows printing of all three above, without user intervention.

The program skips the (fn.typ) at the head of the MAST.CAT file thereby including ALL files in MAST.CAT in the print-outs. The ++++++.FRE files are also sorted and printed to give the user a record of how much space is free on each disk.

The printout of each record is as so:

filename.ext-###

where ### is the disk number. It is necessary that the number of the disk be in the form:

-xxxxxxxx.###

I catalog my disks as: -CTLG.001 etc.

It is important to note that if you select a 6 column printout the screen will get rather messy looking since each column is 16 characters with 2 spaces seperating the columns. This means that a 6 column printout is 106 characters wide (+ up to 9 for left margin = 116). 3 column printouts are 52 characters wide (+ up to 9 for left margin =62).

If you have a 132 character wide printer then the 6 column option will look fine in the 'normal' mode. If not, then you had better choose the 3 column option.

The 'condensed' option sends the Epson MX-80 the codes to put it into the condensed mode before printing begins. I find that the 3 column condensed option produces printouts that make fine disk-jacket labels when sorting by disk number.

The O: option in the printer menu is provided for those occasions when the printer may be unavailable. If a printer is NOT connected the program will 'hang' so use the O: option.

The option of sending the printout to a diskfile will open a file on the currently logged drive called CATALOG.XRF. A formatted copy of the printout is sent to this file.

The screen will echo ALL printouts regardless of other routing selections.

This program uses Richard Conn's SYLIB.REL library (Rev. 2.1) and the Microsoft M80 macro assembler. CATXRF is written in Zilog Z80 mnemonics. Some minor reworking will be needed in order to get it to run on an 8080 based machine.

This program was written on an Osborne-1 DD 80 column computer. The decision has been made to assume 80 columns, however, there are two printer options:

1. 6 or 3 columns
2. Normal or Condensed type

Option #2 is for Epson MX-80 printers. If you are using another type of printer the codes used in this program may do strange things to your printer.

The clear-screen & form-routines may be patched to include your terminal's character codes for clearing, and your printer's for form-feed. Starting at location 0105H are the 5 data bytes available to you for the clear function and 4 bytes for the form function.

As written the sequence is for the Osborne-1 which uses ASCII 26 for clearing the screen, and an Epson MX-80 which uses 12 for form feed.

So the sequence looks like:

```
0105 1A  \
0106 00   \
0107 00    > These 5 bytes for Clear-screen code(s).
0108 00   /
0109 00  /
010A ---> MUST be left as 0FFH
010B 0B  \
010C 00   \ These 4 bytes for Form-feed code(s).
010D 00   /
010E 00  /
010F ---> MUST be left as 0FFH
```

If the .MAC file isn't available to you or you do not have M80 you can use DDT.COM to change these bytes followed by a save of the appropriate number of pages as CATXRF12.COM. (The number of pages is calculated by taking the number marked as NEXT when DDT first says hello and subtracting 100h from it. Divide the result by 256 and there's the number of pages to save.)

[This program has come in extremely handy for finding out what I have on each one of my disks. As stated, using the 'condensed' option for my printer gives a veeasier program to use for the CP/M than the catalog mode of the Customized MSCRIPT. I have to do a great deal of extra work and typing to get the same thing that this program provides. One of these days, I may attempt to 'convert' this to the Timex version for disk systems. That may take a while, but with a little luck, I may attempt it later this year. - David]

****

BENCHMARK
From DATSN, JAN/FEB 1988

In the days of David Ahl's Creative Computing, they occasionally published a test of computation speed and accruacy. The algorithm is repeated here in Sinclair Basic. Timex/Sinclair computers are not speed demons, but they are not bad as number crunchers.

```
 10 REM AHL'S SIMPLE BENCHMARK
 20 FOR N=1 TO 100: LET A=N
 30 FOR I=1 TO 10
 40 LET A=SQR(A): LET R=R+RND
 50 NEXT I
 60 FOR I=1 TO 10
 70 LET A=A^2: LET R=R+RND
 80 NEXT I
 90 LET S=S+A: NEXT N
100 PRINT ABS(1010-S/5)
110 PRINT ABS(1000-R)
```

The above program was modified by using the fast integer for-next loop of the 2.41 version of John Oliger's disk system. The SAFE loop was used on the two inner loops. Alternate code is encouraged for comparison.

```
 10 LET L=0
 15 LET X=1: LET R=0: LET S=0
 18 FOR X=1 TO 100: LET A=X
 20 FOR /1 TO 10
 30 LET A=SQR(A): LET R=R+RND
 40 NEXT
 50 FOR /1 TO 10
 60 LET A=A*A: LET R=R+RND
 70 NEXT
 80 LET S=S+A: NEXT X
 90 PRINT ABS(1010-S/5)
100 PRINT ABS(1000-R)
```

The following chart shows comparison with other computers:

COMPUTER	TIME	ACCURACY
Cray 1	0:00.01	.0000000014
IBM PC	0:24	.01159668
Kaypro II	1:36	.187805
Commodore 64	1:53	.0010414
Apple IIe	1:53	.0010414
Timex/Sinclair (2068)	2:48 (JLO)	.0004131794
TS 1000 (FAST)	2:53	.00041294
TI 99/4A	3:46	.00000011
Sinclair ZX81	4:23	.00066852
Sinclair Spectrum	4:39	.00066852
Tandy Model 100	4:54	.0000002058
TS 1000 (SLOW)	16:55	.0004129408

In the accuracy measure, the smaller the better; likewise with time. I'd like to see how the QL and Z88 would fare. According to JLO, a 2068 computer using the fast SAFE for/next loop can loop faster than QL or C128 in fast mode!

****

ZX81/TS 1000 TIPS
By Don Lamen, SINCUS
March/April 1988

Here is a little function, which may be placed at the beginning or within a machine code program, to stop the program until the tape player starts inputting data.

```
XXXX DBFE      HOLD: IN A,(FE)
     CB7F            BIT 7,A
     28FA            JR Z, HOLD
```

Where: XXXX represents the address.

This is a routine to set RAMTOP and then install your machine code above it. As an example, let's say that you have 80 bytes of machine code in 1 REM and you want to install it as address 30000 [7530 hex].

BASIC PART:
==========

UPLOAD MACHINE CODE:
====================