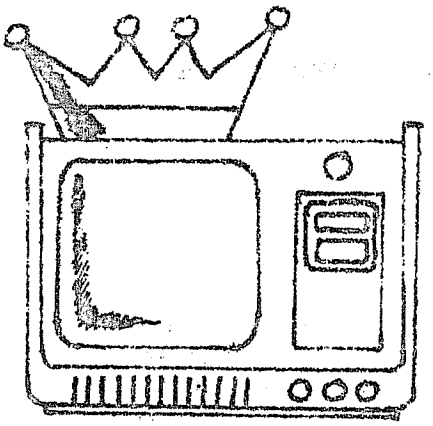


Kings Byte

The Kings County
TRS-80 Users Group

c/o MORTY LIBOWITZ
1063 EAST 84 STREET
BROOKLYN, NY 11236
(212) 763-4233

JUNE 1982 KINGS BYTE NEWSLETTER VOLUME 2 NUMBER 1

KINGS BYTE OFFICERS

President:	Steve Abramowitz
Vice President:	Kal Bergen
Treasurer:	Sid Epstein
Secretary:	Bernie Warren
Communications Officer:	Morty Libowitz

Editor: B. Warren
Arts Editor: Al Hartman
Technical Editor: Kal Bergen

PRINTING: MIDWOOD LEAGUE FOR THE LEARNING DISABLED - A SUBSIDIARY OF VIP FOUNDATION

AREA TRS-80 USER GROUPS

CLUB	TRS-80 GROUP	CONTACT	
METRO	TRUG	BARRY KORNFELD	(212)675-3831
LICA	LITRUG	FRED KEMPLER	(212)229-4040 EXT 6753

AREA TRS-80 BULLETIN BOARDS

BRIAN BOYLE'S B B	(212)933-9459	INNOVATION 80	(516)482-8491
CONNECTION 80 QUEENS	(212)441-3755	CONNECTION 80 HQ	(516)588-5836
CONNECTION 80 MANHATTAN	(212)991-1664	PHOTO 80	(201)790-6795
SISTER STATEN ISLAND	(212)442-3874	BULLET 80 HQ	(203)744-4644
CONNECTION 80 BROOKLYN	(212)277-5851	CONN 80 STONY PT	(914)942-2638
PROGRAMS UNLIMITED	(516)334-3134	BULLET 80	(914)277-0665

KINGS BYTE MEETINGS

Our meetings take place on the first Monday of each month at BAM (B'klyn Academy of Music) located near the Atlantic Ave. and Pacific St. subway stations. It is also near the L.I.R.R station.

Meetings start at 7:00 PM. Assembly language course at 7:30 PM. Lectures and/or demonstrations start at 8:15 PM. For up-to-date information call our Official Contact, Morty Libowitz at (212)763-4233.

Presidents Keyboard

by

Steve Abramowitz

In the course of working with various application programs, I have found that there are a few simple zaps or patches which help integrate different vendor's application programs and provide for smoother operation. I have some helpful bits and bytes that I have developed over the last few months.

Model I SCRIPSIT users know that if they want to use the program in conjunction with the spelling checker MICROPROOF, that the filename for SCRIPSIT must contain the /LC extender. This has caused unnecessary frustration when using the program in conjunction with DIRPICK (DP/CMD). DP sees the /LC extender and thinks it is executing a BASIC file. That /LC extender is a pain in the neck also in DOS READy mode, since the /CMD file extender is not present and cannot be called with the /CMD default. FLEXTEXT patched SCRIPSIT/LC, becomes SCP/LC which does MICROPROOF no good, since it is looking for SCRIPSIT/LC.

The solution is to patch CORRECT/CMD of MICROPROOF with a filespec which includes the /CMD file extender and then rename the SCRIPSIT/LC and SCP/LC files to SCRIPST/CMD.

The patch to CORRECT/CMD is :
File Relative Sector 5, at Relative Byte 30
change 49 54 2F 4C 43 0D to 54 2F 43 4D 44 0D
FLEXTEXT documentation states it does not care what the SCRIPSIT file name is, so that SCRIPST/CMD seems to work OK.

Now for an LDOS unofficial Patch:
To remove the password protection check from all files use FED/CMD, LZAP/CMD, SUPERZAP/CMD from NewDos 80 version 2, (single density only) or SUPER UTILITY. We must make a one byte fix to LDOS 5.1.2's system file SYS2/SYS.RSOLTOFF at Relative Sector 2 Relative Byte 18 change E1 28 2D to E1 18 2D. That is it. I am no longer troubled by 'FILE ACCESS DENIED'.

RSOLTOFF is the overall disk password for LDOS 5.1 and after. It works with every LDOS file except CONFIG/SYS which requires CCC but if you make the above patch you don't need to worry about RSOLTOFF (notice that the O's are slashed zeros Ø).

Finally, a fix for TASMON/CMD the Alternative Source's outstanding monitor. The appendix to the TASMON documentation includes a patch to the original TASMON so that single stepping can be sent out to the printer. The documentation is deficient in not clearly pointing out that the new addresses for the patched TASMON file still requires the 6000H entry point. If you install the stepper hardcopy printout patch and the suggested ASCII display with a disassembly listing, the new addresses for TASMON thus becomes 5D00H 8021H 6000H.

I found that the TASMON appendix documentation also failed to note that with these patches installed, there is a partial video garbage display, while single stepping. To prevent on the screen garbage, create a screen buffer in high memory with the K command. Use the K (Keep screen) command and a HEX address above your machine language programs but below any printer or other drivers, which you may have in high memory. I found E000H to work quite fine.

When you use TASMON, to change an address in the PC (Program Counter) register before Trace single stepping, with the Commands T I or T down arrow or T H I or T H down arrow (H for Hard copy), you must enter the change Register command R followed by PC and the new HEX starting address. TASMON smartly defaults to the C in PC without your entry. The C is a default after the entry of R P. The documentation indicates that all register pairs will default to the second letter, except the IX and IY registers.

For the first Monday in June, the June 7, 1982 meeting we will have a panel of speakers on various printers. The EPSON MX80, the EPSON MX100, the BASE 2, the Daisy Wheel II and some others. Kal will continue his course on programming in Z-80 assembly language and we will conduct a hardware and softwear clinic. We will receive nominations for officers from the nominating commitee chair and from the floor and there will be a report on our planned Computer Picnic this summer.

For July, due to the July 4-5 holiday weekend, we have tentatively rescheduled the meeting to the second Monday (July 12). Note this date will be confirmed at the June meeting, and is due to the holiday.

We will return to our first Monday of the month standard in August.

Happy Computing
Steve

***** ***** ***** ***** ***** ***** *****

THE TRS-80 MODEL I AND THE DISK DRIVE - PART III

by Bernie Warren

TRS-80 DISK FILE MANAGEMENT (CONT'D)

FPDE/FXDE SECTORS

These are relative sectors 2-9 on the directory track, track 17. FPDE means FILE PRIMARY DIRECTORY ENTRY. FXDE means FILE EXTENSION DIRECTORY ENTRY. This is where the file management information is stored, program name, attributes, passwords, size of the file (in sectors), END OF FILE, physical location on the disk, etc. So it is said that the FPDE's and FXDE's are stored here.

FILE ENTRIES IN THE FPDE SECTORS

When the DOS has to find the location of a file entry in an FPDE sector, it doesn't use the method shown above. This is done by using the HASH CODE entry for a specific file, which it finds in the HIT sector. Then the relative location of the byte of HASH CODE is converted to its equivalent 8-bit binary number, using a conversion table.

For example, assume that you are looking at the HIT sector with SUPERZAP (see the HIT sector above), and you see the HASH CODE, CC, at relative byte 04. You know that this is the code for a file named TAIPINST/BAS (Later it will be shown how to determine this.) The relative location, 04, is converted to binary code.

```
04 = 0 0 0 0 0 1 0 0
     ( 0 )     ( 4 ) + 2 = 6
```

The decimal value of the last three bits on the right is 4, so add 2 to get 6. Therefore, the FPDE entry for the file, TAIPINST/BAS, is in sector 6 of the directory. Just add 2 to the decimal value.

Next, the decimal value of the first three bits on the left is 0, so this means that the FPDE entry for TAIPINST/BAS is the zeroith entry in the sector. Sector 6 of the directory is below, and note that the zeroith entry is for TAIPINST/BAS.

```
111600  1020 00E1 0054 4149 5049 4E53 5442 4153 .....TAIPINSTBAS
111610  9642 9642 1500 0701 0D22 FFFF FFFF FFFF .B.B.....".......
111620  0E00 00FD 0043 4F50 5920 2020 2043 4D44 .....COPY....CMD
111630  9642 9642 0500 1020 FFFF FFFF FFFF FFFF .B.B............
111640  0000 0000 0000 0000 0000 0000 0000 0000 ................
111650  0000 0000 0000 0000 0000 0000 0000 0000 ................
111660  0000 0000 0000 0000 0000 0000 0000 0000 ................
111670  0000 0000 0000 0000 0000 0000 0000 0000 ................
111680  0000 0000 0000 0000 0000 0000 0000 0000 ................
111690  0000 0000 0000 0000 0000 0000 0000 0000 ................
1116A0  0000 0000 0000 0000 0000 0000 0000 0000 ................
1116B0  0000 0000 0000 0000 0000 0000 0000 0000 ................
1116C0  1000 00A6 0042 4C43 4B4A 4143 4B42 4153 .....BLCKJACKBAS
1116D0  9642 9642 0B00 1301 0120 FFFF FFFF FFFF .B.B............
1116E0  0000 00A4 0042 494E 5452 4545 2020 2020 .....BINTREE....
1116F0  9642 9642 0F00 1320 1701 FFFF FFFF FFFF .B.B............
```

Also note that each entry for a file has 32 bytes of code. In addition, each file entry starts on one of the relative positions 00, 20, 40, 60, 80, A0, C0, E0, as indicated previously. If there is not enough room to store the data for a file, DOS uses one of the 32 position areas that is not occupied. It can be seen that each directory

sector can contain data for up to 8 files.

Now the FPDE for the file, TAIPINST/BAS, will be copied below from sector 6 above.

```
1020 00E1 0054 4149 5049 4E53 5442 4153
9642 9642 1500 0701 0D22 FFFF FFFF FFFF
```

All of the file information is coded into the machine code that you see above for the file entry. Look at the first group (1020) and remember that each digit is four bits, so that 1020 has an equivalent binary code of 16 bits. The first 8-bit byte (relative byte 00) is converted to binary.

10 = 0 0 0 1 0 0 0 0

Relative bit 7 on the extreme left is a "0", which indicates that this is an FPDE, a primary entry. This bit would be "1" for an FXDE, an extension of another entry somewhere in the directory.

Bit 6 is 0, because this is not a system file. If it were, the bit would be "1".

Bit 5 is not used.

Bit 4 is "1" because this is an active file. For an inactive file, it would be 0.

Bit 3 is "0" because this is a visible file. For an invisible file, it would be 1.

The last three bits give the protection status of the file. The code is below:

CODE	PROTECTION LEVEL
111	No access
110	Execute only
101	Read & execute
100	Write, read & execute
011	Not used
010	Rename, write, read & execute
001	Kill, rename, write, read & execute
000	No restrictions

From this we learn that 1020 (or 1000) means an active file, 0020 (or 0000) would be an inactive (killed) file, and 1820 (or 1800) would be an active, invisible file.

Now going to relative bytes 01 and 02 of this FPDE, if this is an FPDE entry, then they are not used, and these bytes will be 00 and 00. If it is an FXDE entry, then byte 1 is the DEC (DIRECTORY ENTRY CODE) that points back to the FPDE. DEC will be explained later. Byte 2 is always 00, since it is not used.

Relative byte 03 (E1) is the END OF FILE (EOF) BYTE, and is the relative byte position of the last byte of the file in the last relative sector of the file. ALL BYTES REFERRED TO BELOW WILL BE RELATIVE BYTES.

Byte 04 is the LOGICAL RECORD LENGTH of the file. It is always 00 and is 256. This is not actually used by the system.

Bytes 05 - 0C is the filename in code.

Bytes 0D - 0F is the file extension. The "/" is not used here.

Bytes 10 & 11 is the UPDATE PASSWORD. This is two-byte HASH CODE for the password that is specified when the DOS command ATTRIB is used.

Bytes 12 & 13 is the ACCESS PASSWORD. This is a tw-byte HASH CODE for this type of password.

Bytes 14 & 15 is the END OF FILE (EOF) SECTOR, which contains a count of the number (in HEX) of sectors in the file. The two bytes are needed for files that are very long and take up, for example, as much as 400 sectors. 400 (DECIMAL) equals 0186 (HEX). This two byte count would be 8601, with the positions of the bytes reversed, which is the way that the computer handles machine code.

Bytes 16 - 17, 18 - 19, 1A - 1B, 1C - 1D and 1E - 1F are the five extents, EXTENT 1, 2, 3, 4 and 5. An EXTENT is a two-byte entry that indicates that there is an extension of the file which is located elsewhere on the disk. The code for each EXTENT gives the information as to track and contiguous granules for this part of the file. As an example, look at sector 6 above, and note that EXTENT 1 (relative bytes 16 - 17) for this file contains the code 0701. The first byte is the hexidecimal track number, or track 07H. The second byte is broken down into its binary equivalent.

01 = 0 0 0 0 0 0 0 1

The first 3 bits on the left can be either 000 or 001. 000 indicates that the first sector of this part of the file starts at sector 0, and for 001, it starts at sector 5. Therefore, for this part of the file, its first sector starts at sector 0 of track 07H.

Now look at the 5 remaining bytes. The decimal equivalent of these bytes, plus one, is the number of contiguous sectors for this part of the file. In this case it is one plus one, or two.

If the two bytes in the first extent were 1A29, then the file would start on track 1A (HEX) or 27 (DECIMAL), and the decoding of the 2nd byte would look like this:

```
29 (HEX) = 0 0 1 0 1 0 0 1
           ( 1 ) (   9   ) + 1 = 10 (DECIMAL)
```

The first 3 bits are 001, so the file starts at relative sector 5 of track 27. The last 5 bits equal 9 DECIMAL, so the file occupies 10 contiguous granules.

Going back to the FPDE entry of code, an FF means that it is the end of the extents. IF THE USER ADDS TO AN ALREADY EXISTING FILE, AND THE NEXT ADJOINING SECTOR ALREADY HAS DATA, THEN DOS WILL FIND SOME FREE GRANULES, USING THE GAT TABLE, AND THEN DOS WILL INSERT AN EXTENT INTO THE DIRECTORY WHERE THE FPDE FOR THIS FILE EXISTS. There may be up to 5 extents for a file in an FPDE entry.

FXDE ENTRIES

If an FXDE exists for a file, it will be coded in the last EXTENT, EXTENT 5 of the FPDE entry. The code will be FE followed by a DIRECTORY ENTRY CODE (DEC) pointing to an FXDE entry. For example, look at the following FPDE entry with an FXDE code:

1000 002B 0044 4953 4B4F 5247 2050 434C
9642 9642 4200 2023 0124 0500 0701 FE40

The FXDE code (FE40) is in extent 5 and 40 is the DEC that points back to the location of the FXDE entry in the directory, for this file. To decode it, get the binary equivalent of 40 as below:

40H = 0 1 0 0 0 0 0 0

First find the decimal equivalent of the last 3 bits on the right, 0 in his case, and then add 2. Here, the result is 2, so this is the relative directory sector in which the FXDE is located. The next 2 bits to the left are not used.

Now get the decimal equivalent of the last 3 bits on the left (0 1 0), which is 2. This means that the FXDE 32-byte directory entry for this file is relative entry 2 (remember to start counting from zero, i.e. 0, 1, 2, etc.). To understand this, look at relative directory sector 2 below. Relative entry 2 starts at relative byte 40, for the sector, and the first two bytes for this entry are 90 and C7.

```
011200  5E00 0000 0042 4F4F 5420 2020 2053 5953 .....BOOT....SYS
011210  EB29 210E 0300 0000 FFFF FFFF FFFF FFFF .)!.............
011220  0000 0000 0000 0000 0000 0000 0000 0000 ................
011230  0000 0000 0000 0000 0000 0000 0000 0000 ................
011240  90C7 0000 0000 0000 0000 0000 0000 0000 ................
011250  0000 0000 0000 0821 FFFF FFFF FFFF FFFF ................
011260  0000 0000 0000 0000 0000 0000 0000 0000 ................
011270  0000 0000 0000 0000 0000 0000 0000 0000 ................
011280  0000 0000 0000 0000 0000 0000 0000 0000 ................
011290  0000 0000 0000 0000 0000 0000 0000 0000 ................
0112A0  0000 0000 0000 0000 0000 0000 0000 0000 ................
0112B0  0000 0000 0000 0000 0000 0000 0000 0000 ................
```

```
0112C0  0000 0000 0000 0000 0000 0000 0000 0000 ................
0112D0  0000 0000 0000 0000 0000 0000 0000 0000 ................
0112E0  0000 0000 0000 0000 0000 0000 0000 0000 ................
0112F0  0000 0000 0000 0000 0000 0000 0000 0000 ................
```

To explain again in more detail, in the sector above, the first 32-byte directory entry starts at relative byte position 00. The next 32-byte entry would be relative entry 1, starting at relative byte position 20, however, there is no entry here. Relative entry 2 starts at relative byte position 40. Also, looking again at the DEC code, 40, this indicates that the FXDE entry starts at relative byte 40. The first 2 bytes are 90C7, and the first byte, 90, is decoded exactly as the first byte of an FPDE. Byte C7 at relative byte 41 is the DEC that points back to the FPDE for this file. This can be decoded the same way as the DEC for the FXDE.

HOW TO FIND THE HASH CODE FOR A SPECIFIC FILE

The question arises as to how the user can follow through with the information in this article on disk file management, in order to find the location of a specific file on a disk. Well, when DOS looks for file management information for a file, it looks first at the hash codes in the HIT sector for starters. If a user looks at the hash codes, how does he know which hash code pertains to a specific file? To determine this, use a disk that has been formatted, and with nothing else put on it. Go into BASIC mode and type any one-line program. Then save it with the filename that you want to locate on the other disk. Next use SUPERZAP to look at the HIT sector of this disk, sector 1. There will be just three hash codes on it, one will be A2 for BOOT/SYS, another will be 2C (C4 if your system is NEWDOS/80) for DIR/SYS, and a third hash code for the filename that is being sought. Now go back to the other disk and look for this hash code in the HIT sector.

This explains the file management methods used on a disk in a TRS-80 Model I system. The system disk uses these methods to store and read files on a disk.

ATTENTION ALL B A S F DRIVE OWNERS!!!

NO MORE CLUNK CLUNK.

While at the EIGHTY APPLE show last month at the STATLER, I stumbled across a very nice gentleman who was selling BASF drives. Since I had participated in the club's group purchase of BASF drives recently, I purchased a service manual and told the man what a good drive I thought it was, except for the "CLUNK CLUNK", i.e. the head loading & unloading on each disk access. He said "that's because you didn't buy them from me". Whereupon I closely questioned him and found that it was possible to change some of the internal jumpers and cause

the heads to load at motor-on time and stay loaded until motor-off time. When I got home I tried the new jumper arrangement and sure enough, it not only worked but improved the performance of my drives (both of which had been dropped on the street). The man at the show had predicted improved performance due to the disk not constantly changing speed due to the head loading & unloading.

For JJ2, JJ3, and JJ4, only the following pins should be connected:

JJ2	3-4	JJ3	5-6	JJ4	3-4
	5-6		7-8		5-6
			9-10		
			13-14		

KAL

P.S. The booth where I got this valuable information was:
Software Support
One Stalker Lane
Framingham, MA 01701
(800) 343-8841

K.B.

CHANGE OF MEETING DAY FOR JULY

The meeting day for July has been tentatively changed from the first Monday in the month to the second Monday. The reason for the change is that the first Monday, July 5, is a holiday (the day that independence day is celebrated). Keep the meeting day for the July meeting, July 12, in mind.

NOMINATIONS AND ELECTIONS

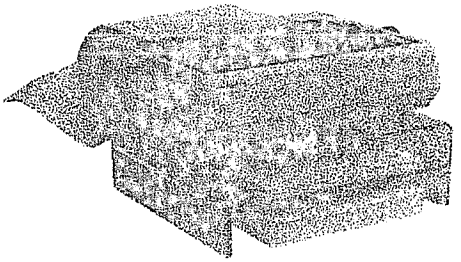
Nominations for officers of KINGS BYTE will take place at the June membership meeting. Any members interested in running for office should contact Arthur Gore, chairman of the election committee. Elections for officers will be held at the following meeting in July.

VERBATIM DATALIFE DISKS

At our next meeting in June, Bob Rosen will have Verbatim Datalife disks available in boxes of 10 for $22. This is a good price, so if you are in the market for disks, make sure that you don't miss the June meeting.

BT ENTERPRISES introduces

Printer Stands For IBM Personal Computer



End the paper mess on your computer desk. Our printer stand allows your paper to be fed from under the printer, making room for the used paper to stack behind the print out of the way. Available with an optional removable shelf (Shown) for easy computer forms change. Available in Large size also, for MX-100 and other large printer users also. Also available with center slot for bottom feed printers. (Large only).

Regular Stand	(300010)	$29.95
Regular w/shelf	(300011)	$44.95
Large Stand	(300020)	$34.95
Large w/shelf	(300021)	$49.95
Large w/slot	(300050)	$49.95

Ribbon Reloads for MX 70/80/100 IBM Personal Computer printer

Reload your old ribbon cartridge for much less then replacing the cartridge! Quick and easy to do, takes only about 2 minutes each to reload. Includes complete instructions. Available in Black and 4 popular colors!

Black Reload	(500000)	$3.95 ea.	Black Reload	(500001)	$39.95 dz.
Red Reload	(500010)	$4.95 ea.	Red Reload	(500011)	$49.50 dz.
Blue Reload	(500020)	$4.95 ea.	Blue Reload	(500021)	$49.50 dz.
Green Reload	(500030)	$4.95 ea.	Green Reload	(500031)	$49.50 dz.
Brown Reload	(500040)	$4.95 ea.	Brown Reload	(500041)	$49.50 dz.

Ribbon Cartridges for MX 70/80 IBM Personal Computer Printer

Replacement cartridges for your printer. Complete cartridge for your printer at great savings! Available in Black and 4 popular colors! Buy in 3 packs and save!

Black Cartridge	(500050)	$10.95 ea.
Red Cartridge	(500060)	$11.95 ea.
Blue Cartridge	(500070)	$11.95 ea.
Green Cartridge	(500080)	$11.95 ea.
Brown Cartridge	(500090)	$11.95 ea.
Black Cartridges	(500051)	$29.95/3
Red Cartridges	(500061)	$31.95/3
Blue Cartridges	(500071)	$31.95/3
Green Cartridges	(500081)	$31.95/3
Brown Cartridges	(500091)	$31.95/3

BT ENTERPRISES

171 Hawkins Rd.
Centereach, N.Y. 11720
(516) 981-8568 (voice)
(516) 588-5836 (modem)

Dealer Inquiries Welcome
N.Y.S. Residents Add Tax
Add $2 shipping — handling
Prices subject to change

VISA WELCOME

Application for membership in KINGS BYTE TRS-80 USERS GROUP

You don't have to live in Brooklyn to become a member.

NAME.....

ADDRESS.....

CITY.....STATE.....ZIP.....

Enclosed is fifteen dollars ($15) for membership in KINGS BYTE TRS-80 USERS GROUP.

Make check payable to: KINGS BYTE COMPUTER CLUB.

Send this form (or copy) to:

TREASURER
KINGS BYTE TRS-80 USERS GROUP
c/o MORTY LIBOWITZ
1063 EAST 84 STREET
BROOKLYN, NY 11236

KINGS BYTE NEWSLETTER
c/o MORTY LIBOWITZ
1063 EAST 84 STREET
BROOKLYN, NY 11236

F I R S T C L A S S M A I L

F I R S T C L A S S M A I L