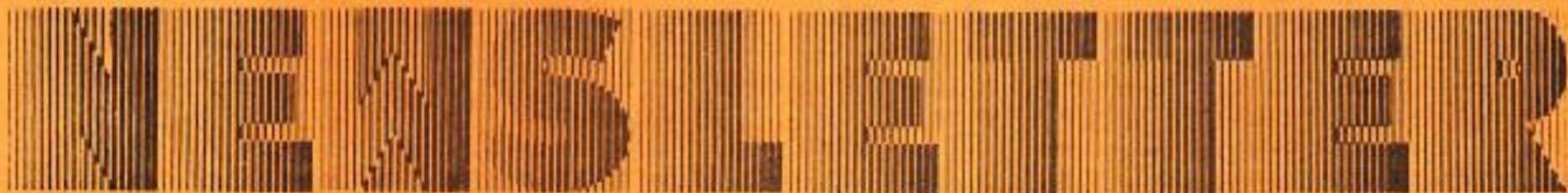


SVI & MSX

SPECTRAVIDEO



AUSTRALASIAN
USERS GROUP



REGISTERED BY AUSTRALIA POST PUBLICATION No. TBH 0917 CATEGORY "B"

ISSUE NO.

3 - 5/6

ANNUAL SUBSCRIPTION

AUSTRALIA	\$20.00
OVERSEAS	\$25.00
OVERSEAS AIRMAIL ...	\$30.00
YEAR BOOK 83/84	\$20.00

DATE

FEB-MAR 1985

CONTENTS

INTRODUCTION	2
LETTER	3
MSX DISK BUG FIX	4
DSKMOD.BAS (programs)	5
EXPLORING BASIC Pt-16	7
TRACK / STATS (programs)	10
ADD ON DISK FOR SVI-738	12
JOUST (program)	13
CURSOR ALTERATION (CP/M program)	17
MSXSWEET.BAS (program)	21
LIBRARY NOTES	29
BUY, TRADE & SELL	33

NEWSLETTER CORRESPONDENCE

S.A.U.G.,
P.O. BOX 191,
LAUNCESTON SOUTH,
TASMANIA, 7249.

(003) 442493

LIBRARY CORRESPONDENCE

S.A.U.G. LIBRARY,
1 CONRAD AVENUE,
GEORGE TOWN,
TASMANIA, 7253.

(003) 822919

INTRODUCTION

The Editor (P.W. Deckert.)

It has been two month since you received the January Newsletter. However since we had a 3 week delay with our Printer on that newsletter, it was becoming almost impossible to print the next newsletter. So for my piece of mind and to get the newsletter back on time, I cheated and held over the February newsletter and included it with the March newsletter.

Dont worry you have a two times larger newsletter and next month things will be back to normal. Sorry for any problems this newsletter has created.

Well with a bigger newsletter you get bigger articles and there is some great software and hardware information in this one.

Many thanks again to L.A. Dunning who has helped bring our newsletter to its current quality with his many articles. How will we ever repay this dedication?

Also this month you again see a regular contributor S.W. McNamee who is a very talented member of our group. His articles and programs are always received with great excitement.

This newsletter is created not only by me, but with the invaluable help of two other hard working people. Mr. J. Collins who you know as our Librarian, and Mr. A. Kellner who is our membership manager. Although you do not hear as much from Mr. Kellner, it is his job to make sure you are on the list and you are reminded when your renewal is due. Both have articles in the newsletter. To all other contributors many thanks and keep it all coming in.

While I am on the subject of money. The cost of living has again caught up with us and unfortunately from the 1st of MAY 1986 we will be forced to increase membership from \$20 to \$25, this is a must to allow us to keep up the Quality of our Newsletter.

YEAR BOOK

The Year Book for 1985 is now available and can be purchased by writing to the address on the front cover. HOWEVER please NOTE that the cost of the Year Book is still \$20, but as from the 1st of MAY the price will also increase to \$25. So be early and buy one now.

That is all from me for this month. Next month we will be starting a new Department called DEAR SAUG, where we select letters we receive from members who have problems that need answering and answer them in the Newsletter for all members benefit.



SPECTRAVIDEO INTERNATIONAL LTD.

Head Office

Room 507-509, New World
Office Building (West Wing),
Tsimshatsui, Kowloon, Hong Kong.
Tel: 3-685123 Tlx: 36929 SVL HX
Fax: 3-7216620

U.S.A. Office

3300 Seldon Court,
Fremont, Calif. 9459, U.S.A.
Tel: (415) 490-4300-2
Telefax: (415) 490-5897

Your ref:

Our ref:

SVM507/85

Date:

20 Jan, 1986

MR. ALFRED BUATTI
16 MERRIGUM CRESCENT
SOUTH CLAYTON 3169
MELBOURNE, VICTORIA
AUSTRALIA

Dear Mr. Buatti

With reference to your Telex 701/1, we like to thank you for your patronage on SVI products.

The problem you encountered on 738 Floppy Disk Drive was a software bug in the Disk Control ROM originally released by Microsoft. Soon after we were aware of such a bug, we strongly raised our protest to Microsoft. Under our continuous pressure, they finally rectified this problem and released a modified EPROM. All machines made from now on have this EPROM installed and disk problem should not happen again.

We apologize for the inconvenience that might incur. However, Microsoft passed the bug to us and Spectravideo in turn suffered too.

Taking this opportunity, we like to thank you once again for your support to SVI and wish you enjoy using our products.

With Best Regards,
SPECTRAVIDEO INTERNATIONAL LTD.

Daisy Kwong
Sales Executive

M.S.X. Disk Bug Fixed !!

By. S.W. McNamee

After the editor's glowing report on the new SVI X'Press I could no longer resist the urge, so last September I proudly took possession of my new 738. Several late nights later I finally got around to playing with MSX-DOS (I am an inveterate CP/M hacker), and was unable to copy a disk without getting several disk errors. I solved that by using the CP/M utility BACKUP (Good old CP/M). Over the next few days I lost several files and a couple of times even wiped out the directory track. It finally dawned on me that something was wrong with MSX-DOS. Back to Rose Music went the X'Press and after a couple of weeks they sent it back saying there was nothing wrong with it. They gave me a new master disk in case there was something wrong with the old one. You guessed it ... same problem. Back to Rose Music who sent it to Melbourne. Another couple of weeks and back came the message "Nothing wrong with it!". Well I jumped up and down in the one spot and said there is definately something wrong somewhere. Finally in Janurary I got a brand new X'Press swapped for mine and major frustration - IT had the same problem.

In the mean time I had the use of a third machine for a short while which also had the fault. Several other X'Press owners I had talked to also experienced trouble with MSX-DOS. It soon sank through the old grey matter that there was a bug somewhere in the disk control software (ROM).

Out came the trusty dissassembler (DISZILOG) and after several dozen pages of print out and 2 days of feverish activity I finally came to the conclusion that the problem was a lack of head settling time in the write sector routine. A small patch to fix this was written and I started looking for somewhere to put it. I then discovered that MSX-DOS leaves 24 bytes of RAM unused at FFD7. Wonder of wonders the patch is 23 bytes long - I seemed to be in business. I loaded it in, gingerly fired up COPY and lo and behold got a clean disk copy. To date I have been running the patch for about 6 weeks and have not had a single disk error - end of story.

There are two BASIC programs to implement the patch.

1. DSKMOD.BAS is for use when DISK BASIC is booted up without going through MSX-DOS. It should be run each time you turn on the computer.
2. DSKMOD1.BAS is a BASIC program that creates a file called DSKMOD.COM which is a command file for use when booting up MSX-DOS which MUST be run each time you boot. You only have to use this program once. When you have a copy of DSKMOD.COM you can copy it to any disks you need.

Note that if you enter DISK BASIC from MSX-DOS after running DSKMOD.COM you need not run DSKMOD.BAS.

The easiest way to implement the patch is to use DSKMOD1.BAS to create DSKMOD.COM and then make this command part of an AUTOEXEC.BAT file. The patch will then be loaded every time you boot MSX-DOS.

DSKMOD.BAS (m.s.x.)

by : S.V. Mc Namee.

This Program may be entered using the 'INPUT' program from Newsletter 2 - 2 (NOV. 84.) or The Year Book.

```
CI      10  ' *****
AB      20  ' **** PROGRAM: DSKMOD.BAS ****
CK      30  ' ****
CL      40  '
BE      50  ' AUTHOR: Steve McNamee
AB      60  ' DATE   : 18-1-86
AK      70  ' USE    : This is a patch to fix
AC      80  ' the bug in the SV-73B X-Press
FI      90  ' disc ROM which corrupts
EJ     100  ' sectors.
AJ     110  '
GK     120  ' This program MUST be run every
HL     130  ' time you boot up DISC BASIC.
AG     140  '
AF     150  '
CC     160 AD=&HFFE6
BO     170 FOR N=0 TO 23
CJ     180 READ D$:D=VAL("&h"+D$)
FP     190 POKE AD+N,D
CJ     200 NEXT N
CP     210 NEW
AL     220 END
AI     230 '
AH     240 '
BC     250 DATA E5,F5,3A,B9,7F,21,FE,FF
CK     260 DATA BE,77,28,08,21,00,40,2B
AH     270 DATA 7C,B5,20,FB,F1,E1,C9,00
END
```

DSKMOD1.BAS (m.s.x.)

by : S.V. Mc Namee.

This Program may be entered using the 'INPUT' program from Newsletter 2 - 2 (NOV. 84.) or The Year Book.

```
CI      10 ' *****
FA      20 ' ***** PROGRAM: DSKMOD1.BAS *****
CK      30 ' *****
CL      40 '
BE      50 ' AUTHOR: Steve McNamee
AB      60 ' DATE : 18-1-86
CP      70 ' USE : This program creates a
DB      80 ' .COM file for use under MSX-DOS
AF      90 ' This command is used to load
BM     100 ' the patch to fix the bug in the
BJ     110 ' SV-738 X-Press disc ROM. The
DM     120 ' command MUST be run every time
DO     130 ' MSX-DOS is booted. The easiest
JH     140 ' way to do this is to make it
AH     150 ' part of an AUTOEXEC.BAT file.
BL     160 ' The name of the command is
BG     170 ' DSKMOD.COM
AC     180 '
AB     190 '
EH     200 OPEN "DSKMOD.COM" FOR OUTPUT AS #1
CJ     210 FOR N=0 TO 38
DA     220 READ D$:D=VAL("&h"+D$)
AF     230 PRINT #1,CHR$(D);
CN     240 NEXT N
BJ     250 CLOSE #1
DE     260 NEW
BA     270 END
BG     300 DATA 00,00,00,21,0F,01,11,E6
CI     310 DATA FF,01,17,00,ED,B0,C9
BD     350 DATA E5,F5,3A,B9,7F,21,FE,FF
CJ     360 DATA BE,77,28,0B,21,00,40,2B
AG     370 DATA 7C,B5,20,FB,F1,E1,C9,00
END
```

Exploring Basic Pt-16

by L.A. Dunning

This part deals with disk drives and how to avoid making errors while using them. It also covers techniques with interrupts.

DISK ERRORS

When Disk Basic is loaded, it also adds a set of errors than can only occur when you use a disk. These are necessary if you want to avoid losing data, files, or overwriting disks. The extra errors are:

- 61 Bad Allocation Table
- 62 Bad Drive Number
- 63 Bad Track/Sector
- 64 File still open
- 65 Disk not mounted
- 66 Deleted Record
- 67 File already exists
- 68 Disk full
- 69 File Write Protected
- 70 Disk I/O error
- 71 Disk offline
- 72 Rename accross Disk

Most of these are fairly obvious in meaning and you should have encountered them already, especially Bad Allocation Table or File already exists. Anyone who can tell me how to get errors 65, 66 or 71 will get a certificate of observation from me. Whenever my drive doesn't have a disk in it, or the catch isn't down, it just hangs. So how do you get errors 65 or 71? There doesn't appear to be anyway to 'delete' a record. You can write over it, but not delete it.

Errors 61 to 63 are straightforward. You will get error 64 if you try to KILL a file that is currently open. Error 67 occurs when you try to rename a file and the new name is the name of an existing file. Error 80 happens when all available tracks on the disk are used. Error 69 occurs when you try to write to the disk and either the file or disk is set for Write Protect ("P"using SET).

You get error 71 when there is something physically wrong with either the disk drive or disk and the drive cannot read/write properly to the disk. You also get this error by breaking the program during such activity. Error 72 merely means that you've put the wrong disk number on the new file name during a NAME.

As with most errors, they can be used to your advantage. The technique is to force a possible error before it would ruin your program and then redirect the flow to handle the problem. For instance, suppose you want to know if a certain file (called "test") exists. You can do FILES and see directly however this requires the user to alter their choice after seeing the result and might disrupt your display. One way of avoiding this is to use:

```
NAME"test" AS "test"
```

This will produce either a File not found error (53) if the file is not on the disk or File already exists (67) if it is there. In the error trapping subroutine, you set up a sequence like:

```
IF ERR=53 then RESUME ##### ELSE IF ERR=67 THEN RESUME 0000
```

where ##### is the starting linenumber for the routine that executes if the file doesn't exist, and 0000 is the start of the subroutine executed when it does exist.

Disk I/O errors are the hardest errors to handle when they are due to hardware/firmware faults. On some cheap disks you may get what are called bad or corrupted sectors. These sectors are not reliable for use and the drive will hang, attempting to read/write to them. SETting a disk to read after write ("R") means that the system will check that what is in a sector matches with what is written to it. If the two don't match, another write is done until the two do match. A file SET to "R" will force the same procedure, but only for sectors belonging to that file.

On other systems, bad sectors can be "locked out" of a directory and for all intensive purposes don't exist. This loses data but saves your nerves. On the SV however, this can't be done on such a basis. If you have a track that has too many bad sectors, or has them in bad locations (like sector 1) then you can block out that track by giving it an FEH entry on the Files Allocation Table (FAT). This value is used to reserve both the system file and directory track. The track will then not be used for a File and the bad sectors avoided. If you have bad sectors on the directory track, the best bet is to copy what files you can to another disk and use the disk for CP/M, or another computer, or anything.

In the above case, copying might prove difficult. If LOAD & SAVE & COPY don't work properly because the DIR or FAT is corrupted, then you can use DSKO\$ & DSKI\$ to transfer a file on a sector by sector basis. This is a long and tedious affair. You will also have to do some detective work using either DISK SCANNER (see part 14) or similar utility to see exactly which tracks and sectors the file is contained in. The listing FILE STATS will list full details of all files on a normal disk. After a file is transferred (to a formatted empty disk) you then use DISK SCANNER (hereafter called DS for brevity) to modify both the DIR and FAT on the new disk. If you have been following the last two parts, you should have an idea how this is done.

Let's take an example of an imaginary bad disk and a file called "test". The FAT tables are beyond repair and we want to get a copy of test because it's the only copy you have and it took you 20 hours to design and type in.

First, you create a new formatted disk using the CP/M format program to format it and the basic format program to create a vacant directory track. Using DS you switch to ASC and MOD. mode to look at the DIR on track 20, sector 1. You find the entry for the program. The first six bytes show "test ". The next byte is 200 octal indicating it is a binary save and the next byte is set to 19, indicating that the first track used for the program is track 19. You go to track 19 and there is the start of "test". Checking along, one sector at a time, you discover that "test" uses all 17 sectors of the track and seems to end abruptly on the 17th. This implies that it is continued on another track.

After 10 minutes search you find what appears to be the rest of "test" on track 36. It occupies sectors 1 to 6. At this point you use DS to transfer the sectors from one disk to another (if you have two drives) or write a small program that uses DSKI\$ to get a sector and then (after you have placed the new disk in the drive) uses DSKO\$ to dump it to the new disk. It will transfer all sectors of track 19 and the first 6 sectors of track 36 from the first to the second disk. The listing TRACK COPIER will allow you to copy tracks and sectors between either two drives or even the same drive, with one or two disks. After the sectors have been dumped you use DS to alter the DIR. On the first entry you place "test " followed by 200 octal and 19. On sector 15 of track twenty you alter the FAT so that entry 19 reads 36 (it then points to entry 36) and on 36 you place 306 octal, indicating the end of file and the first six sectors are used. Sector 15 is then copied to sectors 16 and 17. If you exit DS and do a FILES you will see "test" listed. With luck, if you load "test" you will get your original program restored to you.

It doesn't always work because there can be some doubt as to where a file ends, but at least you recover the bulk of your program. The key to the procedure is the amount of detective work you do. You need to have a clear idea of what you are looking for before you can find it. If either the DIR or FAT is blown, you use the other to help find the file. If both are blown, you have to spend time looking at sector 1's of each track to see if it looks like the start of your file. Good luck!

NEXT MONTH

In next months part, I'll talk about menus, subroutines and interrupts.

TRACK

by : L.A. Dunning

This Program may be entered using the 'INPUT' program from Newsletter 2 - 2 (NOV. 84.) or The Year Book.

```
AC 10 REM TRACK COPIER
DM 20 :
CC 30 SCREEN0,0:CLS:WIDTH39:PRINTTAB(13)"TRACK COPIER":CLEAR5000:DEFIN
TA-Z:DEFSTRV:DIM V(17,1),VB(1):VE=CHR$(27):FIELD#0,128ASVB(0),12
BASVB(1)
KB 40 LOCATE0,1:INPUT"How many drives on line";ND:IFND>20RND<1GOTO40
HO 50 LOCATE0,4,1:PRINT"SECTORS COPIED FROM":PRINT"----- ----- -----"
BN 60 LOCATE0,6:PRINTUSING"Disk           (1 or #) ";ND;;INPUTD1:IFD1<1
ORD1>NDGOTO60
BL 70 LOCATE0,7:INPUT"Track       (0 to 39)";T1:IFT1<0ORT1>39GOT070
FM 80 LOCATE0,8:INPUT"First Sector ( 1 to 17)";SA:IFSA<10RSA>17GOT080
DP 90 PRINT" to":SB=0
AO 100 LOCATE0,10:PRINTUSING"Last Sector (## to 17)";SA;;INPUTSB:IFSB=
0THENSB=SA
BG 110 IFSB<SA OR SB>17GOTO100
EC 120 LOCATE0,12,1:PRINT"SECTORS COPIED TO":PRINT"----- ----- ---"
BH 130 LOCATE0,14:PRINTUSING"Disk           (1 or #) ";ND;;INPUTD2:IFD2<
10RD2>NDGOTO130
BF 140 LOCATE0,15:INPUT"Track       (0 to 39)";T2:IFT2<0ORT2>39GOT070
BL 150 SC=0:SM=17-SB+SA
IO 160 LOCATE0,16:PRINTUSING"First Sector ( 1 to ##)";SM;;INPUTSC:IFSC=
0THENSC=SA
BB 170 IFSC<1 OR SC>SMGOTO160
CK 180 SD=2:IFD1<>D2GOTO200
DI 190 LOCATE0,18,0:PRINT"Are you copying to the same disk <Y/N>":GOSUB
270
DJ 200 LOCATE0,20,0:PRINT"Insert original disk in drive "D1:PRINT"Press
<ESC> when ready":GOSUB290:NC=SB-SA:ON SD GOTO 220,210
AG 210 IFD1=D2GOTO240ELSEGOSUB280
DA 220 FORS=0TONC:V=DSKI$(D1,T1,SA+S):DSKO$D2,T2,SC+S:NEXT:GOTO250
DI 230 GOSUB280:FORS=0TONC:V=DSKI$(D1,T1,SA+S):FORE=0TO1:V(S,E)=VB(E):N
EXT:NEXT
BH 240 GOSUB280:FORS=0TONC:FORE=0TO1:LSETVB(E)=V(S,E):NEXT:DSKO$D2,T2,S
C+S:NEXT
EF 250 LOCATE0,18:PRINTVE"J";:PRINT"Do you want to copy more sectors <Y
/N>":GOSUB270:ONSDGOTO260,300
KF 260 PRINT"Use the same original specs      <Y/N>":GOSUB270:LOCATE0,1
8:PRINTVE"J":ONSDGOTO120,50
AI 270 V=INPUT$(1):SD=INSTR("YyNn",V)/2+.5:IFSD=0GOTO270ELSERETURN
HO 280 LOCATE0,22:PRINT"Insert copy    disk in drive "D2:PRINT"Press <
ESC> when ready";
DH 290 V=INPUT$(1):IFV<>VEGOTO290ELSERETURN
CL 300 CLS:SCREEN0,1:LOCATE,,1
END
```

STATS

by : L.A. Dunning

This Program may be entered using the 'INPUT' program from Newsletter 2 - 2 (NOV. 84.) or The Year Book.

```
BI      10 REM FILE STATS
FO      20 REM Lists Name, Type, last sector           used and tracks used
       by file.
DN      30 :
FB      40 CLS:MAXFILES=2:CLEAR2000:DEFINTA-Z:DEFSTRC,S:DIM S(16),C(40),N(4
       0)
EF      50 A=1:B=0:D=0:X1=VARPTR(#0)+9
KF      60 FIELD#0,16ASS(0),16ASS(1),16ASS(2),16ASS(3),16ASS(4),16ASS(5),16
       ASS(6),16ASS(7),16ASS(8),16ASS(9),16ASS(10),16ASS(11),16ASS(12),
       16ASS(13),16ASS(14),16ASS(15)
FB      70 LOCATE0,10:PRINT"List FILES to <S>creen":PRINTTAB(11)"or <P>rint
       er"
A6      80 C=INPUT$(1):OP=INSTR("SsPp",C):IFOP=0GOTO80
MC      90 IFOP>2THENSF="LPT"ELSESF="CRT"
IK      100 OPENSF+":FILES"FOROUTPUTAS#1
GF      110 PRINT:PRINT#1,"Name      Type Last,Tracks used":PRINT#1, "
       Sctr
CE      120 D$=DSKI$(1,20,A)
CI      130 S(16)=S(B)
BK      140 TP=ASC(S(16)):IFTP=255GOTO260
BD      150 IFTP=0GOTO240
CK      160 D=D+1:CC=LEFT$(S(16),9)+" "
HE      170 TP=ASC(MID$(S(16),10)):FM=TPAND&0707:EX=TPAND&070:IFFM=0THENCC=C
       C+"ASC"
CE      180 IFFM=1THENCC=CC+"MCR"
BE      190 IFFM=128THENCC=CC+"BIN"
BH      200 C1=" ":"IFTPAND&010THENC1="R"
BF      210 C2=" ":"IFTPAND&020THENC2="P"
BA      220 C3=" ":"IFTPAND&040THENC3="*
DF      230 C(D)=CC+C1+C2+C3:N(D)=ASC(MID$(S(B),11))
CD      240 B=B+1:IFB>15THENB=0:A=A+1:GOTO120
AC      250 GOTO130
CB      260 IFD=0GOTO310
GM      270 D$=DSKI$(1,20,15):FORA=1TOD:CC="":J=N(A)
BJ      280 CC=CC+", "+RIGHT$(STR$(J),2):J=PEEK(X1+J):IFJ<192GOTO280
AP      290 C(A)=C(A)+(" "+RIGHT$(STR$(JAND63),2)+"")"+CC:NEXT
HD      300 FORA=1TOD:PRINT#1,C(A):NEXT:GOTO320
BF      310 PRINT#1,"NO FILES"
AH      320 CLOSE
END
```

ADD ON DISK DRIVE FOR SVI-738 X'PRESS

Add on double sided 80 track 5 1/4 inch disk drives are now available for the X'Press. These drives come complete with CUSTOM BIOS and store a massive 790 kBytes when running CP/M.

The custom BIOS supplied allows reading, writing and formatting many different disk formats. Currently 28 formats can be read/written including Spectravideo S/S and D/S, all 5 MicroBee formats (including the 3 1/2 inch format), Amstrad, Kaypro 11, and Osborne double density. There are 13 types able to be formatted.

This BIOS also supports 3 different terminal emulations. ADM-3A, ADM-31 and VT-52. A configuration utility is supplied that allows various features of the BIOS to be user modified. Eg. the RS-232 baud rate, the screen colours and the function key contents.

The disk supplied also has several useful public domain utilities supplied free of charge.

The drives are fully compatible with MSX-DOS and when running this operating system the standard 360 kBytes is available.

Prices:	SAUG Members Drive plus software	\$450.00
	Software alone	\$ 50.00

Non Members Drive plus software	\$525.00
Software alone	\$ 75.00

To order send a cheque or money order with ORIGINAL CP/M master disk to :

S.W. McNamee,
5/15 Stuckey Rd.,
Clayfield 4011.
Ph. 835-8683 Home
262-1127 Work

***** NOTE *****

Custom double sided 40 track drives are still available for Spectravideo SV-328. Prices as previously advertised.

JOUST

by : R. Moore (310/328)

This Program may be entered using the 'INPUT' program from Newsletter 2 - 2 (NOV. 84.) or The Year Book.

```

AF 1 ' = JOUST = by Ross Moore
AA 2 ' Instructions:
KB 3 ' Object: to joust with the coloured rider, and pop the eggs that appear. (You lose points if they hatch.)
AC 4 ' Spacebar/Fire makes your bird fly upward, joystick ↔ controls direction To win a joust you must be facing and higher than opponent.
KA 5 ' As your score increases, barriers and hazards appear; don't fall in the fire!
MG 6 ' ' ' ' ' ' ' ' ' ' ' ' ' ' ' ' ' '
FK 10 ONSTOPGOSUB475:STOPON:GOTO485' Intro
HF 15 COLOR,1,1:SCREEN1,2:DEFINTA-C,E-Z:DEFSNGD:CLICKOFF
IO 20 PLAY"T255L64":SOUND6,31:SOUND7,&B101010:B$="T255L64V1507ED#DV12E
D#DV8ED#D":K$="V1406CER6CG3":GOSUB390'Read Sprites
JE 25 CLS:STOPON:COLOR15:LINE(0,170)-(255,191),13,BF:LOCATE110,183:PRIORITY"Hi:"HS:LOCATE198,183:PRINT"Live":S=3:X=120:Y=153:A=240:B=90
:DA=-5:DB=1.8:N=RND(-TIME):S1=1:S2=4:S3=3:LI=5:GOSUB310:GOSUB385
:C=10
FM 30 ONSPRITEGOSUB175:GOTO90
AL 35 IFDY>-2THENDY=DY-5ELSEDY=DY-2:IFDY<-7THENDY=-7
BE 40 RETURN
AG 45 IFPOINT(X+17,Y+4)=13ORPOINT(X+17,Y+11)=13THENPLAYB$:DX=-DX
LF 50 RETURN ' Bounce-off routines
AH 55 IFPOINT(X-2,Y+4)=13ORPOINT(X-2,Y+11)=13THENPLAYB$:DX=-DX
BG 60 RETURN
HH 65 IFPOINT(X+8,Y+22)=13THENDY=1
GE 70 IFPOINT(X+8,Y+17)=13THENDY=0:IFDX<0THENS1=2:RETURN135ELSE1=1:RETURN135
EI 75 IFPOINT(X+8,Y+13)<>BANDPOINT(X+8,Y+9)<>8THENRETURNELSEOUND9,15:
PUTSPRITE0,,15,8:FORN!=0TO31STEP.05:SOUND6,N!:NEXT' Fall in fire
DA 80 FORN=0TO4:SOUND9,12-3*N:PUTSPRITE0,,,N+9:FORT=1TO100:NEXTT,N
BO 85 LI=LI-1:GOSUB385:X=120:Y=153:DX=0:DY=0:RETURN : ** Main Loop follows
JC 90 PUTSPRITE0,(X,Y),15,S1:PUTSPRITE1,(A,B),C,S:A=A+DA:B=B+DB:IFB>15
3THENB=153:DB=-DBELSEIFB<0THENB=0:DB=-DB
AJ 95 IFA<-8THENA=248ELSEIFA>248THENA=-8
CM 100 ST=STICK(0)+STICK(1):SR=STRIG(0)+STRIG(1):IFST=20RST=30RST=4THEN
DX=DX+1ELSEIFST=60RST=70RST=8THENDX=DX-1
EF 105 IFDX>7THENDX=7ELSEIFDX<-7THENDX=-7
BB 110 IFS=5ANDTIME>600THENGOSUB260
AM 115 IFSRTHENGOSUB35
DN 120 DY=DY+1:IFDY>7THENDY=7
AC 125 IFDY>0THENGOSUB65
AK 130 IFDX>=0THENS1=S2:GOSUB45ELSE1=S3:GOSUB55
BA 135 X=X+DX:Y=Y+DY:IFY<0THENY=0:DY=1
AM 140 IFX<-8THENX=248ELSEIFX>248THENX=-8
66 145 IFRND(1)<5E-03THENDA=-DA:IFS<5THENS=7-SELSEIFS>5THENS=13-S' Change Dir.
CD 150 IFRND(1)<.01THENDB=-DB
PK 155 IFDY<=0THENIFPOINT(X+8,Y-2)=13THENDY=6
BG 160 SM=1-SM:IFSMTHEN SOUND9,8ELSE SOUND9,4' Switch Wing Sound

```

D6 165 SPRITEON:S=10-S:S2=10-S2:S3=10-S3:GOTO90' Switch Sprite Shapes **
End Loop
FP 170 ' Sprite Collision Routines
CG 175 SPRITEOFF:SOUND9,0:IFS=5THEN250ELSEIFX=120ANDY=153THENRETURN
DJ 180 IFY>B-3ANDY<B+3THEN210
CD 185 IFSGN(DA)=SGN(DX)THENPLAYB\$:GOTO220
DH 190 IFY>BTHEN195ELSEPLAYK\$:S=5:SC=SC+150:B=B+20:DY=DY-3:DB=1.8:GOSUB
310:TIME=0:RETURN90
AB 195 SOUND5,3:SOUND10,14
EO 200 FORD=YTO238:PUTSPRITE0,(X,D),,8:SOUND4,D:NEXT:SOUND10,0
BD 205 LI=LI-1:S1=1:Y=153:X=120:DX=0:DY=0:GOSUB385:RETURN90
CF 210 PLAYB\$:IFSGN(DA)=SGN(DX)THEN220
CE 215 DA=-DA:DX=-DX:GOTO230
AC 220 IFABS(DA)>ABS(DX)THENDA=-DA:DX=2*DX:GOTO230
DM 225 DX=-2*DX:GOTO235
AM 230 IFS<5THENS=7-SELSEIFS>5THENS=13-S
BN 235 X=X+2*DX:A=A+2*DA
DE 240 IFB>YTHENB=B+8ELSEB=B-B
BK 245 RETURN90
EA 250 A1=A:B1=B:SC=SC+300:B=RND(1)*150:PLAY"07V14GG+AV15A+B08C":IFX<12
0THENA=240:DA=-5:S=3ELSEA=0:DA=5:S=4
AE 255 GOTO265
EA 260 A1=A:B1=B:SC=SC-100:PLAY"S11MB00007B2":IFDA=5THENS=4ELSES=3
AL 265 C=C-2:IFC=1THENC=10ELSEIFC=8THENC=7
CE 270 FORN=9T013:PUTSPRITE1,(A1,B1),N-5,N:FORT=1T050:NEXTT,N
EG 275 GOSUB310:RETURN90:*****
BI 280 C=15:PUTSPRITE0,,0:PUTSPRITE1,,0:IFSC>HSTHENHS=SC
KK 285 CLS:N1=43:PLAY"V14":FORN=1T06:N2=N1-3:N3=N1-6:N4=N1-9:PLAY"N=N1;
N=N2;N=N3;N=N4;":N1=N1-3:NEXT:N4=N4-3:PLAY"L2N=N4;L64"
FM 290 IFSTRIG(0)+STRIG(1)THENSC=0:GOTO25
BK 295 C=C-1:IFC=1THENC=15
EA 300 COLORC:LOCATE73,20:PRINT" G A M E O V E R ":LOCATE40,60:PRINT"Y
ou scored"SC:LOCATE40,80:PRINT"High score:"HS
BE 305 LOCATE40,150:PRINT"PRESS 'FIRE' FOR NEW GAME.":GOTO290
EC 310 LINE(26,183)-(56,190),13,BF:COLOR15:LOCATE20,183:PRINTSC:ONSC/50
0 GOSUB315,320,325,330,335,340,345,350,355,360,365,370:RETURN '
Display Score
AK 315 LINE(75,80)-(175,87),13,BF:RETURN
BJ 320 MA=0:LINE(0,40)-(75,47),13,BF:LINE(175,40)-(255,47),13,BF:RETURN
BH 325 LINE(0,118)-(75,125),13,BF:LINE(175,118)-(255,125),13,BF:RETURN
CK 330 LINE(0,170)-(50,180),8,BF:LINE(205,170)-(255,180),8,BF:RETURN
BJ 335 LINE(50,170)-(100,180),8,BF:LINE(155,170)-(205,180),8,BF:RETURN
BI 340 LINE(100,170)-(155,180),8,BF:LINE(125,170)-(130,173),13,BF:IFMA=
0THENGOSUB380:MA=1:RETURNELSERETURN
DI 345 LINE(75,80)-(175,73),8,BF:RETURN
AH 350 LINE(0,40)-(75,33),8,BF:LINE(175,40)-(255,33),8,BF:RETURN
DO 355 LINE(0,118)-(75,111),8,BF:LINE(175,118)-(255,111),8,BF:RETURN
DD 360 LINE(0,0)-(7,170),13,BF:LINE(255,0)-(248,170),13,BF:IFMA=1THENGO
SUB380:MA=2:RETURNELSERETURN
BB 365 LINE(122,35)-(128,80),13,BF:RETURN
BI 370 LINE(0,0)-(255,6),13,BF:IFMA=2THENGOSUB380:MA=3:RETURNELSERETURN
FG 375 ' Extra Life
JC 380 LI=LI+1:SOUND5,0:SOUND10,14:FORN=1T03:FORD=255T00STEP-8:SOUND4,D
:NEXT:FORD=5T0255STEP8:SOUND4,D:NEXTD,N:SOUND10,0
CD 385 LINE(240,183)-(245,190),13,BF:COLOR15:LOCATE234,183:PRINTLI:IFLI
=0THEN280ELSERETURN ' Display Lives
AN 390 RESTORE:FDRS=1T07:IFS=5THENNEXTELSES\$=""":FORN=1T032:READA:S\$\$=S\$\$+
CHR\$(A):NEXT:SPRITE\$(S)=S\$:NEXT

ED 395 S\$="" : FORN=1 TO 6 : READA : S\$=S\$+CHR\$(A) : NEXT : SPRITE\$(5)=S\$
 BA 400 S\$="" : FORN=1 TO 32 : READA : S\$=S\$+CHR\$(A) : NEXT : SPRITE\$(8)=S\$
 FG 405 FORS=9 TO 13 : S\$="" : FORN=1 TO 29 : READA : S\$=S\$+CHR\$(A) : NEXT : SPRITE\$(5)=
 S\$: NEXT : RETURN
 HE 410 DATA 6,6,4,14,14,14,127,31,127,255,127,31,2,2,2,3,0,0,6,15,8,8,2
 55,8,152,248,240,128,0,0,0,128 : ' Stand R
 AC 415 DATA 0,0,96,240,16,16,255,16,25,31,15,1,0,0,0,1,96,96,32,112,112,
 112,254,248,254,255,254,248,64,64,64,192 : ' Stand L
 DA 420 DATA 0,0,96,240,16,16,255,16,25,31,15,3,0,0,0,0,96,96,36,124,124,
 124,254,252,254,255,254,248,224,0,0,0 : ' Fly L(U)
 BD 425 DATA 6,6,4,14,14,14,127,31,127,255,127,63,63,60,112,64,0,0,6,15,8
 ,8,255,8,152,248,240,192,0,0,0,0 : ' Fly R(D)
 CO 430 DATA 6,6,36,62,62,62,127,63,127,255,127,31,7,0,0,0,0,0,6,15,8,8,2
 55,8,152,248,240,192,0,0,0,0 : ' Fly R(U)
 BC 435 DATA 0,0,96,240,16,16,255,16,25,31,15,3,0,0,0,0,96,96,32,112,112,
 112,254,248,254,255,254,252,252,60,14,2 : ' Fly L(D)
 GE 440 DATA 0,126,255,255,126,60 : ' Egg
 DL 445 DATA 2,7,7,15,15,15,255,143,143,6,6,7,3,0,0,0,0,64,64,192,248,255
 ,251,192,64,64,64,64,248,76,76,72 : ' Dive
 BE 450 DATA 0,0,0,0,1,5,0,12,0,5,1,0,0,0,0,0,0,0,64,0,96,0,64,0,0,
 0 : ' P sml
 AF 455 DATA 0,0,0,1,9,4,0,24,0,4,9,1,0,0,0,0,0,0,0,32,64,0,48,0,64,32,
 0,0 : *2
 AE 460 DATA 0,0,1,17,8,0,0,48,0,0,8,17,1,0,0,0,0,0,0,16,32,0,0,24,0,0,32
 ,16,0 : *3
 AG 465 DATA 0,1,1,16,8,0,0,96,0,0,8,16,1,1,0,0,0,0,0,16,32,0,0,12,0,0,32
 ,16,0 : *4
 DJ 470 DATA 1,1,32,16,0,0,0,192,0,0,0,16,32,1,1,0,0,0,8,16,0,0,6,0,0,
 0,16,8 : *5
 BA 475 IF INKEY\$>"" THEN 475
 GM 480 BEEP : COLOR 15,4 : SCREEN 0,1 : CLICKON : END
 BK 485 DATA 50,79,83,83,0,45,79,79,82,69,0,80,82,69,83,69,78,84,83,14,14
 ,14
 EF 490 COLOR 4,15 : SCREEN 0,0 : LOCATE,,0 : DIM L(63) : RESTORE 485 : FORN=21 TO 42 : R
 EADL(N) : NEXT : PLAY "T200L30SBM2000" : N1=30
 FG 495 FOR A=0 TO 20 : PLAY "N=N1;" : N1=N1+1
 HM 500 FOR N=1 TO 22 : VPOKEN+448, L(N+A) : NEXT
 GM 505 PLAY "N=N1;" : N1=N1+1 : FORN=1 TO 21 : VPOKE 19+40*N, L(N+A) : NEXT N, A
 AG 510 PLAY "S13M750T11005F4L24CCL16C04A05CC04A05CF3R10", "S13T11005C4L24
 04AAL16AFAAFA05C3R10"
 EC 515 PLAY "L16AGFAGFAGFG12R16C24C4RB", "L16FECFECFECE12R1604G24G4RB"
 CK 520 COLOR 13,1,1 : SCREEN 1 : LINE(47,116)-(77,123),,BF : LINE(60,124)-(67,1
 49),,BF
 FM 525 DRAW "BM47,123D5EU3RD2EURBM77,123D5HU3LD2HULBM41,149R7"
 FP 530 CIRCLE(54,149),6,,3.1416,0 : CIRCLE(54,149),13,,3.1416,0 : PAINT(49,
 160)
 DO 535 CIRCLE(94,139),8,,,1.333 : CIRCLE(94,139),15,,,1.15 : PAINT(94,150
)
 DF 540 LINE(114,124)-(121,141),,BF : LINE(133,124)-(140,141),,BF
 EN 545 CIRCLE(127,141),6,,3.1416,0 : CIRCLE(127,141),13,,3.1416,0 : PAINT(1
 27,152)
 EL 550 CIRCLE(160,137),6,,0,4.713 : CTRCLE(160,137),13,,0,4.713 : CIRCLE(16
 1,157),6,,3.1416,1.571
 CG 555 CIRCLE(161,157),13,,3.1416,1.571 : DRAW "BM167,137R7BM147,157R7" : PA
 INT(160,169)
 H6 560 LINE(177,116)-(210,123),,BF : LINE(190,124)-(197,160),,BF : DRAW "BM1
 77,123D5EU3RD2EURBM210,123D5HU3LD2HUL"

```
BE    565 COLOR15:LINE(30,179)-(221,181),,BF:DRAW"BM105,178L7UR3L3UBM105,1
      82L7DR3L3DBM222,180R2"
PJ    570 PLAY"C4L2004AAL10A05CF4R8L76F65A16A2", "A4L20FFL10FA05C4R8L7EDE5F
      16F2"
IF    575 DRAW"C15BM62,30H2LUL3GR2L2GR6FL5G2D9G2LGLGL2GL2GD6RUFRFL5U6H2L2H
      L2HLHLH2EREREREREU7EU5ER2FD3G2F2D2F2GDFDF2RFRE3UEU6R2G":PAINT(35
      ,40):LINE(30,35)-(67,35)
DC    580 LINE(121,70)-(135,15):LINE-(150,30):LINE-(131,30):PAINT(140,23)
IH    585 COLOR10:DRAW"C10BM193,30E2RUR3FL2R2FL6GR5F2D9F2RFRFR2FR2FD6LUGLG
      R5U6E2R2ER2ERERE2HLHLHLHU7HU5HL2GD3F2G2D2G2FDGDG2LGLH3UHU6L2F"
      :PAINT(220,40):LINE(225,35)-(188,35)
EF    590 LINE(134,70)-(120,15):LINE-(105,30):LINE-(124,30):PAINT(117,23)
EE    595 DEFUSR=18312:A=USR(0):FORN=1TO2000:NEXTN:SCREEN1:GOT015
FF    600 IFPLAY(1)THEN600ELSESCREEN1:GOT015
END
```

S.A.U.G.**1985****YEARBOOK**

THE S.A.U.G. 1985 YEAR BOOK IS
NOW AVAILABLE FROM THE USERS GROUP

PRICE \$20

PRICE ONLY CURRENT FOR MARCH AND
APRIL. PRICE MUST GO UP SOON.

NOW**AVAILABLE**

Cursor Alteration Program

By. A. Kellner (CP/M)

***** CURSOR ALTERATION PROGRAM *****

; This program allows the user to alter the type and size of cursor
; when using
; the spectravideo 328 and the 80 column card.

; Written by A. Kellner, George Town, Tas. SVI & MSX Users
; group. 22/12/85

***** ORG 0100H ;Base address of program

***** SYSTEM CONSTANTS *****

BOOT	EQU	0000H	;Reboot cp/m
BDOSS	EQU	0005H	;System call
CONIN	EQU	1	;Console input function
CONOUT	EQU	2	;Console output function
CONIO	EQU	6	;Console i/o function
PSTR	EQU	9	;Print string at console function

***** PROGRAM CONSTANTS *****

CLS	EQU	0CH	;Clear screen control code
CR	EQU	ODH	;Carriage return control code
LF	EQU	OAH	;Line feed control code
ESC	EQU	01BH	;Escape control code
IVON	EQU	070H	;Inverse video on [p]
IVOFF	EQU	071H	;inverse video off [q]

CRTPT	EQU	050H	;Address reg. select port [80 col card]
DATAPT	EQU	051H	;Control reg. data port "
RTOP	EQU	0AH	;Reg. for the top of the cursor
RBOT	EQU	OBH	;Reg. for the bottom of the cursor

NOBL	EQU	00H	;Value for a non blinking cursor
NOCS	EQU	020H	; " " no cursor
FASTBL	EQU	040H	; " " fast blinking cursor
SLOWBL	EQU	060H	; " " slow blinking cursor

***** SELECT TYPE OF CURSOR REQUIRED *****

SELECT:	LXI	D,MENU1	
	CALL	PRINT	;Print first menu
	CALL	INPUT	;Get users response
	LDA	CIN	;Load users input
	CPI	031H	;test which option selected
	JZ	NBK	;non-blinking
	CPI	032H	
	JZ	NCS	;no cursor
	CPI	033H	

```
JZ      SBK      ;slow blink
CPI    034H
JZ      FBK      ;fast blink
JMP    SELECT   ;If not valid option loop
;
***** SET TOP POSITION OF CURSOR *****
;
SIZE:  LXI    D,MENU2
CALL   PRINT   ;Print second menu
CALL   INPUT   ;Get users response
LDA    CIN
CPI    030H   ;Test if input is in range
JC     SIZE    ;if not loop
CPI    039H
JNC    SIZE    ;Not in range loop
SUI    030H   ;Remove ascii offset
STA    TOPSL   ;Store top scan line selected
MOV    B,A
LDA    CDAT   ;Load selected cursor type & add
ADD    B       ;top scan line
STA    CDAT   ;Store back in memory
;
***** SET BOTTOM POSITION OF CURSOR *****
;
SIZE2: LXI    D,MENU3
CALL   PRINT   ;Print third menu
CALL   INPUT   ; Same as above
LDA    CIN
CPI    030H
JC     SIZE2
CPI    039H
JNC    SIZE2
SUI    030H
MOV    B,A
LDA    TOPSL
CMP    B       ;Check that bottom line no.
JZ     STORE   ;is = or > top line no.
JNC    SIZE2   ;If not loop
STORE: MOV    A,B
        STA    BOTSL   ;Store bottom line no. in memory
;
***** SET CURSOR TO USER INPUT VALUES *****
;
SETCRS: MVI   A,RTOP  ;load reg no. for top of cursor
        OUT   CRTPT  ;send to address port
        LDA   CDAT   ;load value to set
        OUT   DATAPT ;send value to data port
        MVI   A,RBOT  ;load reg no. for bottom of cursor
        OUT   CRTPT  ;send to address port
        LDA   BOTSL  ;load value to set
        OUT   DATAPT ;send value to data port
        JMP   SELECT  ;Go back to main menu.
```

```
;  
***** VALUES FOR TYPES OF CURSOR SUB-ROUTINES *****  
;  
NBK:    MVI      A,NOBL ;Non-blinking  
        STA      CDAT  
        JMP      SIZE  
;  
NCS:    MVI      A,NOCS ;No cursor  
        STA      CDAT  
        JMP      SETCRS  
;  
SBK:    MVI      A,SLOWBL;Slow blink  
        STA      CDAT  
        JMP      SIZE  
;  
FBK:    MVI      A,FASTBL;Fast blink  
        STA      CDAT  
        JMP      SIZE  
;  
***** UTILITY SUBROUTINES *****  
;  
***** EXIT ROUTINE *****  
;  
EXIT:   LXI      D,CLEAR ;Clear the screen  
        CALL     PRINT  
        JMP      BOOT    ;Go back to cp/m  
;  
***** PRINT STRING AT CONSOLE ROUTINE *****  
;  
PRINT:  PUSH    H  
        PUSH    B  
        MVI     C,PSTR ;Print string function  
        CALL    BDOS  
        POP     B  
        POP     H  
        RET  
;  
***** PRINT SINGLE CHARACTER AT CONSOLE ROUTINE *****  
;  
PCHR:   MVI     C,CONOUT;Console output function  
        CALL    BDOS  
        RET  
;  
***** GET CONSOLE INPUT ROUTINE *****  
;  
INPUT:  MVI     C,CONIO ;Console input function  
        MVI     E,OFFH  
        CALL   BDOS  
        ORA     A      ;Test for input  
        JZ      INPUT  ;If none loop  
        CPI     ESC    ;If escape code  
        JZ      EXIT   ;return to cp/m  
        STA     CIN    ;Store input in memory  
        MOV     E,A  
        CALL   PCHR   ;Print input at console  
        RET
```

```
;  
;***** STRING CONSTANT *****  
;  
CLEAR: DB      CLS,CR,LF,'$';String to clear the screen  
;  
;***** SELECTION MENU *****  
;  
MENU1: DB      CLS,LF,CR  
DB      ','          ====== ',CR,LF  
DB      ','          CHANGE CURSOR SHAPE PROGRAM',CR,LF  
DB      ','          ====== '  
DB      CR,LF,LF  
DB      ','          ',ESC,IVON,'SELECT TYPE OF CURSOR REQUIRED'  
DB      ESC,IVOFF,CR,LF,LF  
DB      ' 1)  - Non - blinking cursor ',CR,LF  
DB      ' 2)  - No cursor ',CR,LF  
DB      ' 3)  - Slow blinking cursor ',CR,LF  
DB      ' 4)  - Fast blinking cursor ',CR,LF  
DB      ' ESC) - To exit program ',CR,LF,LF  
DB      ','  
DB      ESC,IVON,'ENTER YOUR SELECTION :',ESC,IVOFF,' '$'  
;  
;***** SELECT TOP OF CURSOR MENU *****  
;  
MENU2: DB      CLS,LF,CR  
DB      ','          SELECT SIZE OF CURSOR REQUIRED',CR,LF  
DB      ','          -----',CR,LF,LF  
DB      ','          ',ESC,IVON,'TOP SCAN LINE POSITION',ESC,IVOFF  
DB      CR,LF,LF  
DB      'Enter [ 0 ] to start cursor at top of block ',CR,LF  
DB      'to [ 8 ] to start at the bottom or ESC to exit.',CR,LF  
DB      'E.G. set top & bottom scan line to 8 for a thin',CR,LF  
DB      'underline or top to 0 and bottom to 8 for the usual',CR,LF  
DB      'large block type of cursor. ',CR,LF,LF  
DB      ','  
DB      ESC,IVON,'ENTER A NUMBER 0 TO 8 :',ESC,IVOFF,' '$'  
;  
;***** SELECT BOTTOM OF CURSOR MENU *****  
;  
MENU3: DB      CR,LF,LF  
DB      ','          ',ESC,IVON,'BOTTOM SCAN LINE POSITION',ESC,IVOFF  
DB      CR,LF,LF  
DB      'Enter [ 0 ] to finish cursor at top of block ',CR,LF  
DB      'to [ 8 ] to finish at the bottom or ESC to exit.',CR,LF  
DB      'Note: The bottom scan line value must be equal to',CR,LF  
DB      'or greater than the top scan line value',CR,LF,LF  
DB      ','  
DB      ESC,IVON,'ENTER A NUMBER 0 TO 8 :',ESC,IVOFF,' '$'  
;  
;***** DATA STORAGE AREA *****  
;  
CIN:   DS      1      ;Storage area for:- console input  
CDAT:  DS      1      ;value for type of cursor  
TOPSL: DS      1      ;    "    "    top of cursor  
BOTSL: DS      1      ;    "    "    bottom of cursor  
;  
=====
```

END

MSXSWEET.BAS" (m.s.x.)

by : S.V. - Mc Namee.

This Program may be entered using the 'INPUT' program from Newsletter 2 - 2 (NOV. 84.) or The Year Book.

```
CI      10 ' *****
AD      12 ' **** PROGRAM: MSXSWEET.BAS ****
CE      14 ' ****
CC      16 '
CA      18 ' AUTHOR: Steve McNamee
AC      20 ' DATE : 16-2-86
CJ      22 ' USE : This program is a com-
AP      24 ' prehensive file manipulation
BC      26 ' utility. It is designed to
FC      28 ' implement most of the features
DO      30 ' of the CP/M program NSWP2.COM
BD      32 ' The features not supported are
BE      34 ' the file print, file status
BJ      36 ' and squeeze/unsqueeze options.
CC      38 '
CG      40 ' IMPORTANT VARIABLES:
FP      42 ' NM$() - holds the file names
AG      44 ' AT() - holds the attributes
AK      46 ' for each corresponding file
BK      48 ' name. 0=untagged, 1=tagged,
JO      50 ' 3=file copied...possible retag
EC      52 ' and 3=deleted
DG      54 ' LN() - holds the length in kBytes
BM      56 ' of each corresponding file name
DF      58 ' CF - is the current file poin-
FE      60 ' ter
FJ      62 ' TL - is the total length in k
GD      64 ' bytes of all tagged files
BK      66 ' CM - is the maximum file number
AE      68 ' DN$ - is the logged drive name
CB      70 ' DN - is the logged drive number
CM      72 '
EK      74 ' NOTE: This program has only been
BL      76 ' tested on a SV-73B X-Press.
BG      78 ' It should work OK on any MSX
BL      80 ' machine that has at least
BI      82 ' 32k but this cannot be guar-
AG      84 ' anteed.
CJ      86 '
CH      88 '
DA      90 '
CC    100 CLEAR 4000,&HD000
DE    110 ONSTOPGOSUB 9900:STOP ON
CP    120 GOSUB9000 'load machine code into memory
DI    130 DEFINT A,L,C,N :DIMNM$(256),LN(256),AT(256)
BE    135 WIDTH40:LOCATE,,1
CP    140 DN$="A:";DN=1:POKE&HD0A0, DN 'set drive number in FCB
BC    200 'SEARCH FOR FIRST &HD000
EK    210 'SEARCH FOR NEXT &HD020
CN    215 'COMPUTE FILE LENGTH &HD090
AE    220 DEFUSR=&HD000:DEFUSR1=&HD020:DEFUSR2=&HD090
```

```
DD 230 CF=1:TL=0:CM=0:X$=USR(0):IF X$="" THEN 500
BE 240 NM$(CF)=X$
AD 245 X=USR2(0) 'compute file length in 128 byte blocks
DO 250 LN!=(PEEK(&HD100+34)*256+PEEK(&HD100+33))*128
CH 255 L1=INT(LN!/1000):IFLN!-L1<>0THENL1=L1+1
EI 260 LN(CF)=L1:AT(CF)=0
CB 270 CM=CM+1:CF=CF+1
AD 280 X$=USR1(0)
DN 290 IF X$<>"" THEN 240
AM 300 '
GO 500 CLS:PRINT"MSX-DOS Sweep Version 1.0"
BL 520 PRINT"By Steve McNamee":PRINT
CH 540 PRINTDSKF(DN); "kBytes free on Drive ";DN$:PRINT
FF 550 CF=1:GOTO8500'print file
BN 560 GOSUB8700 'this is start of command loop
NL 580 IF I$="t"ORI$="T"THEN1000
NJ 590 IF I$="u"ORI$="U"THEN1500
MG 600 IF I$="m"ORI$="M"THEN3000
NK 610 IF I$="v"ORI$="V"THEN4000
LK 620 IF I$="e"ORI$="E"THEN5000
LJ 630 IF I$="c"ORI$="C"THEN7000
LH 640 IF I$="b"ORI$="B"THEN8200
LC 650 IF I$="a"ORI$="A"THEN2000
NK 660 IF I$="w"ORI$="W"THEN2500
HO 670 IF I$=?ORI$=/THEN5500
NI 680 IF I$="s"ORI$="S"THEN6500
NI 690 IF I$="r"ORI$="R"THEN7500
LO 700 IF I$="l"ORI$="L"THEN3500
LH 710 IF I$="d"ORI$="D"THEN6000
EO 720 IF I$=CHR$(13)ORI$=" "THEN8000
NL 730 IF I$="x"ORI$="X"THEN9900
NM 740 IF I$="x"ORI$="X"THEN9900
LI 750 IF I$="f"ORI$="F"THEN1200
EO 1000 'Tag File
CK 1020 '
AI 1040 IFAT(CF)=1THEN1120
AC 1060 AT(CF)=1
BL 1080 TL=TL+LN(CF)
GO 1100 PRINT" * ";
DF 1120 GOSUBB300
EH 1140 CF=CF+1
CP 1160 GOTO8500'print next file
CG 1200 '
DN 1220 'Find file
CK 1240 '
EL 1260 PRINT:PRINT:PRINT"Name of file to find";
DA 1280 INPUTFI$
AD 1300 IFMID$(FI$,2,1)="""THENFI$=RIGHT$(FI$,LEN(FI$)-2)
FH 1320 DP=INSTR(FI$,"."):IFDP=0THENFI$=FI$+STRING$(11-LEN(FI$)," "):GOTO13
60
AI 1340 F1$=LEFT$(FI$,DP-1):F2$=RIGHT$(FI$,LEN(FI$)-DP)
BJ 1350 FI$=F1$+STRING$(8-LEN(F1$)," ")+F2$+STRING$(3-LEN(F2$)," ")
CG 1360 FORN=1TO11:C=ASC(MID$(FI$,N,1)):IFC>96ANDC<123THENC=C-32
EO 1380 MID$(FI$,N,1)=CHR$(C):NEXTN
AB 1400 FORN=1TOCM
BL 1420 IFAT(N)=3THEN
GJ 1440 IFNM$(N)=FI$THENCF=N:GOTO1480
```

```
AC 1460 NEXTN:PRINT:PRINT"File ";FI$;" not found"
DA 1480 GOTO8500'print next file
CD 1500 '
DM 1520 'Untag
CH 1540 '
EM 1560 IFAT(CF)<>1THENCF=CF+1:GOTO8500'print next file
CK 1580 AT(CF)=0:TL=TL-LN(CF)
HH 1600 PRINTCHR$(8);CHR$(8);"u ";'erase "*" and replace with "u"
AE 1620 GOTO1120
CJ 2000 '
DM 2020 'Retag
CN 2040 '
JF 2050 PRINT:PRINT:PRINT"Tagging:"
AK 2060 FORN=1TOCM
CH 2080 IFAT(N)<>2THEN2180
CI 2100 '
EA 2120 AT(N)=1:TL=TL+LN(N)
EL 2140 PRINT"--> ";DN$;NM$(N);
CO 2160 GOSUB8300'print length of tagged files
FF 2170 PRINT
GE 2180 NEXTN
EE 2200 CF=CM
DD 2220 GOTO8500'print next file
CE 2500 '
FK 2520 'Wild card tag
CI 2540 '
LG 2560 FI$="":PRINT:PRINT:PRINT"Tag What";:INPUT FI$
BF 2565 IF FI$=""THEN8500
HP 2570 PRINT:PRINT"Tagging:"
DA 2580 DP=INSTR(FI$,".")
BM 2600 IFDP=0THENF1$=FI$:F2$=""":GOTO2660
AD 2620 F1$=LEFT$(FI$,DP-1)
DB 2640 F2$=RIGHT$(FI$,LEN(FI$)-DP)
MO 2660 P=INSTR(F1$,"*"):IFP=0THENF3$=F1$+STRING$(8-LEN(F1$)," "):GOTO2700
DD 2680 F3$=LEFT$(F1$,P-1)+STRING$(9-LEN(F1$),"?")+RIGHT$(F1$,LEN(F1$)-P)
LK 2700 P=INSTR(F2$,"*"):IFP=0THENF4$=F2$+STRING$(3-LEN(F2$)," "):GOTO2740
DI 2720 F4$=LEFT$(F2$,P-1)+STRING$(4-LEN(F2$),"?")+RIGHT$(F2$,LEN(F2$)-P)
BM 2740 F$=F3$+F4$
AD 2760 FORN=1TOCM
FB 2780 IFAT(N)=3THEN2940'skip if file erased
CB 2800 '
AK 2820 FORL=1TO11
DH 2840 C$=MID$(NM$(N),L,1):C1$=MID$(F$,L,1):C=ASC(C1$)
CG 2850 IFC<123 AND C>96 THENC=C-32'capitalise
CD 2860 C1$=CHR$(C):IFC$<>C1$ANDC1$<>"?"THEN2940'files not equal so exit
FL 2880 NEXTL
DP 2900 PRINT"--> ";DN$;NM$(N);
EP 2910 D=N
IB 2920 IFAT(N)<>1THENAT(N)=1:TL=TL+LN(N)
DJ 2930 GOSUB 8300'print length of tagged files
EE 2935 PRINT
FI 2940 NEXTN
HH 2960 PRINT:CF=D:GOTO8500'print next file
CK 3000 '
EB 3020 ;Mass transfer
FG 3048 ONERRORGOTO3400
```

```
BD 3080 PRINT:PRINT:PRINT"Copy to ";
EP 3100 I$="" : INPUT I$
GC 3110 IF I$="" THEN 8500
BA 3120 IF I$="A:" OR I$="a:" THEN 3220
BG 3140 IF I$="A" OR I$="a" THEN I$="A:" : GOTO 3220
AM 3160 IF I$="B:" OR I$="b:" THEN 3220
BN 3180 IF I$="B" OR I$="b" THEN I$="B:" : GOTO 3220
BK 3200 GOSUB 8800 : PRINT : GOTO 3080
AF 3220 FOR N=1 TO M
DC 3240 IF AT(N)<>1 THEN 3320
FE 3260 PRINT
BE 3280 PRINT "Copying "; DN$; NM$(N); " to "; I$;
CG 3300 COPY DN$+NM$(N) TO I$
ED 3310 AT(N)=2: TL=TL-LN(N)
FN 3320 NEXT N
CA 3340 ONERROR GOTO 0
AE 3360 CF=1: PRINT
CN 3380 GOTO 8500 ' print next file
BO 3400 IF ERR=66 THEN PRINT:PRINT "Disk full": RESUME 3340
FE 3420 IF ERR=68 THEN PRINT":PRINT "DISK WRITE PROTECTED": resume 3340
AB 3440 IF ERR=69 THEN PRINT":PRINT "DISK OFFLINE": resume 3340
EE 3460 PRINT:PRINT "Copy Error on file "; DN$; NM$(N): RESUME 3320
CF 3500 "
BJ 3520 ' Log new drive
CJ 3540 "
NG 3560 PRINT:PRINT "Which drive to log on";
LM 3580 GOSUB 8900 ' get drive specs
DB 3600 DN$=D1$: DN=D1: POKE &HD0A0, DN ' set drive byte in FCB correctly
EJ 3620 CLS
BD 3640 GOTO 230 ' load new directory
CL 4000 "
FC 4020 ' View
CP 4040 "
FD 4050 CLS
DD 4060 OPEN DN$+NM$(CF) FOR INPUT AS#1
FI 4100 LINEINPUT #1, A$
DF 4120 PRINT A$
DD 4140 I$=INKEY$
BK 4240 IF I$=CHR$(24) THEN 4320 ' ^X
DF 4300 IF NOT EOF(1) THEN 4100
FA 4320 PRINT
EL 4340 CLOSE #1
CO 4360 GOTO 8500 ' print next file
CM 5000 "
HP 5020 ' Erase files
DA 5040 "
FG 5060 ONERROR GOTO 5400
PA 5080 PRINT:PRINT:PRINT "Erase (T)agged or (U)ntagged ? ";
KA 5100 GOSUB 8700 ' get char
BM 5120 IF I$="u" OR I$="U" THEN T=$: GOTO 5200
BN 5140 IF I$="t" OR I$="T" THEN T=1: GOTO 5200
CO 5150 IF I$=CHR$(13) THEN PRINT: GOTO 8500
BI 5160 GOSUB 8800 ' bell
AN 5180 GOTO 5100
AF 5200 FOR N=1 TO M
GN 5210 IF T=0 THEN IF AT(N)=2 THEN N5240
FF 5220 IF AT(N)<>T THEN 5300
```

```
FE 5240 PRINT
DM 5260 PRINT"Erasing --> ";DN$;NM$(N);
BO 5280 KILLDN$+NM$(N)
FN 5300 NEXTN
CA 5320 ONERRORGOTO0
BI 5340 GOTO 230 'load new directory
CI 5400 '
LG 5420 PRINT:PRINT"File deletion error on ";DN$;NM$(N)
BK 5440 RESUME5300
CH 5500 '
EK 5520 'Help
CL 5540 '
FA 5560 CLS
DL 5580 PRINT"           COMMAND MENU"
EM 5600 PRINT
BA 5620 PRINT"<A>-Retag files"TAB(20)"<B>-Back one file"
DC 5630 PRINT"<C>-Copy file"TAB(20)"<D>-Delete file"
ED 5640 PRINT"<E>-Erase files"TAB(20)"<F>-Find file"
FF 5650 PRINT"<L>-Log new drive"TAB(20)"<M>-Mass transfer"
AF 5660 PRINT"<R>-Space on drive"TAB(20)"<S>-Space on drive"
BB 5665 PRINT"<T>-Tag a file"TAB(20)"<U>-Untag a file"
NE 5670 PRINT"<W>-Wildcard tag"TAB(20)"<X>-Exit program"
AI 5690 PRINT"<?>-This help menu"TAB(20)"<CR>-NEXT FILE"
GD 5700 PRINT:PRINT"NOTES:
CG 5710 PRINT"      1. * indicates a tagged file
CK 5720 PRINT"      2. # indicates a file that may be"
HO 5730 PRINT"          retagged with <A>"
EJ 5900 PRINT
DI 5910 GOTOB500'print next file
CN 6000 '
DG 6020 'Delete file
DB 6040 '
FG 6060 ONERRORGOTO6400
FL 6080 PRINT
LI 6100 PRINT"Deleting --> ";DN$;NM$(CF);
BL 6120 KILLDN$+NM$(CF)
FL 6140 IFAT(CF)=1 THEN TL=TL-LN(CF):AT(CF)=3
CH 6160 ONERRORGOTO0
BA 6180 PRINT:AT(CF)=3:CF=CF+1
DB 6200 GOTOB500'print next file
LA 6400 PRINT:PRINT"Deletion error on ";DN$;NM$(CF)
BL 6420 RESUME6200
CI 6500 '
CP 6520 'Space on drive
CM 6540 '
CM 6560 PRINT" Which drive ";
EM 6580 GOSUB8900'get drive spec
AN 6600 PRINT:PRINTDSKF(D1);"k free on ";D1$
DD 6620 GOTOB500'print next file
CO 7000 '
GB 7020 'Copy file
DC 7040 '
FF 7060 ONERRORGOTO7300
FL 7070 PRINT
AE 7080 PRINT:PRINT"Copy to ";
FD 7100 I$=""":INPUTI$
GG 7110 IF I$=""":THENB500
```

B6 7120 PRINT"Copying "DN\$;NM\$(CF);" to ";I\$
BD 7130 IFMID\$(I\$,2,1)="" THENCOPYDN\$+NM\$(CF) TO I\$:GOT07160
BC 7140 COPYDN\$+NM\$(CF) TO DN\$+I\$
CI 7160 ONERROR GOTO0
CH 7180 GOT08500'print next file
AI 7300 IFERR=66THENPRINT"Disk Full":RESUME7160
BI 7320 IFERR=68THENPRINT"Disk write protected":RESUME7160
EE 7340 PRINT"Copy error"
CJ 7360 RESUME7160
CJ 7500 '
AJ 7520 'RENAME
CN 7540 '
FF 7560 ONERRORGOTO7800
FG 7570 PRINT
KM 7580 PRINT:PRINT"New name ";
BD 7600 FI\$="" INPUTFI\$:IFMID\$(FI\$,2,1)="" THENF1\$=FI\$:GOT07640
BB 7610 IF FI\$="" THEN8500
DL 7620 IFMID\$(FI\$,2,1)="" THENFI\$=RIGHT\$(FI\$,LEN(FI\$)-2)'truncate drive na
me
FA 7630 DP=INSTR(FI\$,"."):IFDP=0 THENFI\$=FI\$+STRING\$(11-LEN(FI\$)," "):GOT076
60
AF 7640 F1\$=LEFT\$(FI\$,DP-1):F2\$=RIGHT\$(FI\$,LEN(FI\$)-DP)
BM 7650 FI\$=F1\$+STRING\$(8-LEN(F1\$)," ")+F2\$+STRING\$(3-LEN(F2\$)," ")
DI 7660 FORN=1TOLEN(FI\$)
BL 7670 C=ASC(MID\$(FI\$,N,1))
CM 7680 IFC>96 AND C<123 THENC=C-32'capitalise
FC 7690 MID\$(FI\$,N,1)=CHR\$(C):NEXTN
BI 7700 NAME DN\$+NM\$(CF) AS FI\$
AI 7710 NM\$(CF)=FI\$
BO 7720 ONERRORGOTO0
DC 7730 GOT08500'print next file
AK 7800 'ONERRORGOTO0
EF 7810 IFERR<>65THENPRINT:PRINT"Rename Error":RESUME7720
NA 7820 PRINT:PRINTDN\$;FI\$;" already exists ...":PRINT"Erase (Y or N) ? ";
AJ 7830 GOSUB8700'char in
BL 7840 IFI\$="Y" OR I\$="y" THEN7880
CH 7850 IFI\$="N" OR I\$="n" THENPRINT:RESUME7720
AN 7860 GOSUB8800'Bell
AF 7870 GOT07830
CK 7880 KILLFI\$
AH 7890 FORN=1TOCM:IFNM\$(N)=FI\$THENAT(N)=3:NEXTN
BB 7900 RESUME7700
CP 8000 '
BA 8020 'ENTER or SPACE keys
DD 8040 '
FB 8060 CF=CF+1
CF 8080 GOT08500'print next file
CN 8200 '
GM 8220 'B key
DB 8240 '
EN 8260 CF=CF-1
CH 8280 GOT08500'print next file
CM 8300 '
IO 8310 'Print length of tagged files
CO 8320 '
DM 8330 PRINT TAB(24);
BL 8340 PRINTUSING"Tagged= ### k ";TL;

```
BK 8350 RETURN
CK 8500 '
BA 8520 'Print next file
CD 8540 '
FG 8560 PRINT
IA 8570 IF CM=0 THEN PRINT "No files": GOT0560
LD 8580 IF AT(CF)=3 THEN CF=CF+1 'next file if deleted
HD 8590 IF CF=0 THEN CF=CM: PRINT
FB 8600 IF CF>CM THEN CF=1: PRINT
DI 8610 PRINT USING "& ### k"; DN$+NM$(CF), LN(CF);
HD 8620 IF AT(CF)=1 THEN PRINT " * ";
GM 8630 IF AT(CF)=2 THEN PRINT " # ";
DC 8640 GOT0560
CI 8700 '
EL 8720 'Character input
CM 8740 '
DD 8760 I$=INKEY$
GE 8780 IF I$="" THEN 8760
BK 8790 RETURN
CH 8800 '
EM 8820 'Bell
CL 8840 '
CK 8860 PLAY "M200005S1G"
BI 8880 RETURN
CG 8900 '
AG 8910 'Get drive name
CI 8920 '
BF 8930 INPUT D1$
BF 8940 IF D1$="A": ORD1$="a": THEND1=1: RETURN
BD 8950 IF D1$="B": ORD1$="b": THEND1=2: RETURN
GK 8960 IF D1$="A" OR D1$="a": THEND1=1: D1$="A": RETURN
GL 8970 IF D1$="B" OR D1$="b": THEND1=2: D1$="B": RETURN
GD 8980 GOSUB 8800: PRINT "Invalid drive name ... Try again"
AE 8990 GOT08900
FN 9000 AD=&HD000: FOR N=0 TO 207
GH 9010 READD$: D=VAL("&h"+D$): POKE AD+N, D
GG 9020 NEXT N
CN 9025 '
BB 9030 ' THE DATA FOR THESE TWO POKES SHOULD BE THE LOW BYTE
CL 9040 ' AND THE HIGH BYTE OF THE ADDRESS TO CALL FOR DOS CALLS
JM 9060 POKE &HD051, &H7D 'LOW BYTE
BH 9080 POKE &HD052, &HF3 'HIGH BYTE
DJ 9090 '
CP 9100 '
BO 9490 RETURN
EG 9500 ' D000 Search for first
FL 9510 DATA 11, 00, D1, 0E, 1A, CD, 50, D0, 11, A0, D0, 0E, 11, CD, 50, D0
GI 9520 DATA C3, 40, D0, 00, 00, 00, 00, 00, 00, 00, 00, 00, 00, 00, 00, 00
CO 9530 '
BK 9550 ' D020 Search for next
FJ 9560 DATA 11, 00, D1, 0E, 1A, CD, 50, D0, 11, D0, A0, 0E, 12, CD, 50, D0
GN 9570 DATA C3, 40, D0, 00, 00, 00, 00, 00, 00, 00, 00, 00, 00, 00, 00
DD 9580 '
BC 9600 ' D040
GC 9610 DATA B7, 3E, 00, C2, 60, D0, 3E, 0B, C3, 60, D0, 00, 00, 00, 00, 00
FH 9620 DATA C3, 00, 00, 00, 00, 00, 00, 00, 00, 00, 00, 00, 00, 00, 00
CN 9630 '
```

```
BF 9650 'D060
GP 9660 DATA 32,B0,D0,3E,03,32;63,F6,21,B0,D0,22,F8,F7,C9,00
EM 9670 DATA 00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00
DC 9680 '
EE 9700 'D0B0 Return parameter area D090 Get file length
EH 9710 DATA 0B,01,D1,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00
GJ 9720 DATA 11,00,D1,0E,23,CD,50,D0,C9,00,00,00,00,00,00,00,00,00
CM 9730 '
CN 9750 'D0A0 DPB
FN 9760 DATA 00,3F,3F,3F,3F,3F,3F,3F,3F,3F,00,00,00,00,00
EL 9770 DATA 00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00
EM 9780 DATA 00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00
CH 9900 '
HK 9920 CLS:CLEAR200,&H8000+24455
END
```

There are two new CP/M utility disk offers in this listing. GRAF.COM is a full disk of software which will allow you to produce graphical displays and add enhancements to your printouts when combined with any of the current crop of Epson compatible printers. This is a powerful set of routines and well worth having in any CP/M library. There is a very powerful demo program on the disk to show you how it is done. NULU.COM is a set of routines which will allow you to Squeeze your CP/M files, combine them in libraries, and store them on disks in this compressed form. On the disk with NULU.COM is NSW.COM which is probably the most powerful disk utility available. These two programs are worth their weight in gold for any CP/M user. Full documentation files are on the disk for NULU and NSW and these files printout to give full operating instructions for both programs.

SVI X/Press and 728 M.S.X. owners can order any of the above CP/M Software on 3.5 inch or 5.25 inch disks to suit their needs. 3.5 inch disks must be supplied by the users as we are not carrying stock owing to the high price.

LIBRARY NOTES

by J. Collins.

I must apologise to all those members who have ordered software etc from the Library over the past two months for the delays in getting most orders away. I've been ferociously busy at my normal place of work and just when things settled down we had our X/press go down and out with a dry joint on one of the hairy little legs of the Z80. However I'm back in full swing now and ready to process those orders, so roll 'em up.

As a special note I must repeat my request that you include a stamped addressed envelope for replies, when you are writing for information. If you are sending for disks or cassettes of software you should use a Padded Jiffy-Bag but don't put two hundred staples in the top. When ordering for the first time from the library please print your name and address very clearly. I will send back some official order forms with your order for future use and these should then be used exclusively for any orders.

CP/M PUBLIC DOMAIN LIBRARY SOFTWARE

Items listed in this category are suitable for 318/328 owners who have an expander unit and one or two disk drives. Also suitable for MSX owners with disk drives and CP/M capability. (not all MSX machines support CP/M).

The label "PUBLIC DOMAIN" refers to programs written and then released in order to be used by anyone, for no charge, by the author. The charges we make for these programs reflect the direct cost to the library in terms of labour, time, and resources. This represents the cheapest possible way of getting high-class utility software. All disk-drive owners should make full use of this section of our library. At the moment I can supply CP/M software on the following disk formats.

STANDARD SINGLE-SIDED DOUBLE DENSITY (early 318/328)
STANDARD DOUBLE-SIDED DOUBLE DENSITY (later 318/328)
CUSTOM DOUBLE-SIDED DOUBLE DENSITY (altered BIOS)
SVI-MSX SINGLE-SIDED DOUBLE DENSITY (5.25inch disk)
*** XPRESS SINGLE-SIDED DOUBLE DENSITY (3.5 inch disk)

ALL CP/M PROGRAMS CURRENTLY ADVERTISED IN THE NEWSLETTER WILL BE AVAILABLE FOR OWNERS OF MSX MACHINES WHICH SUPPORT CP/M, ON BOTH 3.5 inch and 5.25 inch SINGLE-SIDED DOUBLE DENSITY DISKS.

*** While on the subject of disks and disk formats we have decided, because of the high cost of 3.5 inch disks, that we will not keep stocks of blank 3.5 disks on hand. This means that if you own an Xpress 738 and want software on disk you will have to supply the disks yourself.

IMPORTANT NOTICE

When supplying disks for CP/M software you should FORMAT each disk and SYSGEN an image of your CP/M system onto each disk, and remember when ordering to clearly state if this has been done and in what format. If you don't do this you may have to wait longer for your order. I can not put a CP/M system on your disks as this would contravene copyright laws.

Response to the Competition Software Package offers has been good and I would advise that the special offer prices are for a limited period only. Don't miss out because you delayed sending off your order.....

SOFTWARE COMPETITION PACKAGE OFFERS

All those programs which were submitted for inclusion in our recent competition have been combined into three disks or (three cassettes) and are offered for sale as follows :-

Single Disks or Cassettes containing twelve or more programs will cost \$18.00 per disk or \$16.00 per cassette.

\$45.00. All three disks, containing over 36 programs, will cost \$45.00. A saving of \$9.00 on single disk price.

\$40.00. All three cassettes, containing over 36 programs, will cost \$40.00. A saving of \$8.00 on single cassette price.

IN THIS OFFER WE RESERVE THE RIGHT TO SUPPLY THE DISKS OR CASSETTES AND TO CHOOSE THE PROGRAMS WHICH WILL BE ON EACH DISK OR CASSETTE. DO NOT SEND YOUR OWN DISKS OR CASSETTES AND PLEASE DON'T REQUEST PARTICULAR PROGRAMS INSIDE THIS SPECIAL OFFER. Either buy a single disk or cassette or order the package. This is the only way the software will be available. When ordering please refer to the disk, cassette or package you want, as follows :-

COMPETITION DISK NUMBER ONE/CASSETTE NUMBER ONE
COMPETITION DISK NUMBER TWO/CASSETTE NUMBER TWO
COMPETITION DISK NUMBER THREE/CASSETTE NUMBER THREE
COMPETITION DISK PACKAGE (3 disks)
COMPETITION CASSETTE PACKAGE (3 cassettes)

Here's your chance to build up your library of Games and utilities with programs going for less than a dollar each. When you consider that cost of media is also included in this bonus price it must rate as a real bargain. **RUSH YOUR ORDERS NOW**

GROUP AUTHOR SOFTWARE-LIBRARY PROGRAM LIST

ASKING PRICE		OUR MEDIA		YOUR MEDIA	
Includes Pack and Post.		CASSETTE	DISK	CASSETTE	DISK
3D-MAZE \$5.00		9.00	11.00	5.00	5.00
CALENDARS \$3.00		7.00	9.00	3.00	3.00
MURDER \$10.00	##	14.00	16.00	10.00	10.00
MYSTERIOUS MANOR \$5.00		9.00	-----	5.00	-----
COUNT DRACULAR \$10.00		14.00	-----	10.00	-----
CRUNCH \$10.00	##	14.00	16.00	10.00	10.00
DISASSEMBLER \$5.00	##	9.00	11.00	5.00	5.00
ELIZA \$10.00	##	14.00	16.00	10.00	10.00
MARVYN \$10.00	##	-----	16.00	-----	10.00
MIGHTY MORMAR \$5.00		9.00	11.00	5.00	5.00
HOUSE OF FRANKENSTEIN \$5.00		9.00	11.00	5.00	5.00
PACMAN \$10.00		14.00	16.00	10.00	10.00
SUPER IMP/ED \$10.00		14.00	16.00	10.00	10.00
JOYSTICK SPRITE \$10.00	##	14.00	16.00	10.00	10.00
FILES \$5.00		9.00	11.00	5.00	5.00
RUBIKS CUBE \$10.00		14.00	16.00	10.00	10.00
X'BERT \$10.00		14.00	16.00	10.00	10.00
FIVE GAME PACK \$6.00		10.00	12.00	6.00	6.00
ASMED/LOADER \$11.00		15.00	-----	11.00	-----
WP318/WP328 \$5.00		9.00	11.00	5.00	5.00
MSX GAMES PACKAGE \$7.50		11.50	13.50	7.50	7.50
BASIC UTILITIES \$6.50		-----	12.50	-----	6.50
DRAW-2 \$7.50		11.50	13.50	7.50	7.50
SVI ARTIST \$7.50		11.50	13.50	7.50	7.50

COMPETITION DISK ONE	-----	18.00	-----	-----
COMPETITION DISK TWO	-----	18.00	-----	-----
COMPETITION DISK THREE	-----	18.00	-----	-----
THREE DISK PACKAGE DEAL	-----	45.00	-----	-----
COMPETITION CASSETTE ONE	16.00	-----	-----	-----
COMPETITION CASSETTE TWO	16.00	-----	-----	-----
COMPETITION CASSETTE THREE	16.00	-----	-----	-----
THREE CASSETTE PACKAGE DEAL	40.00	-----	-----	-----

The programs shown with two hash marks after the name are available for both SVI318/328 computers and also have been ported across to MSX. Please be sure when ordering that you specify which computer you want the software for.

The two BASIC adventure programs Mysterious Manor and Count Dracular are too long to fit in memory of standard 318 computers. Owing to problems which have shown up in these two programs they are now only being supplied on cassette.

CP/M LIBRARY SOFTWARE

LINERA.COM \$10.00

ADHEX.COM \$10.00

DISK.COM \$10.00

CP/M PUBLIC DOMAIN SOFTWARE

ADVENT.COM * \$5.00

MODEMZ.COM * 55 00

VAM.COM * \$5.00

XDIR.COM \$5.00

ZCBBZ \$5.00

CATALOGUE OF THE 15-20

CP/M GAMES DISK \$6.00

GRAE.COM * \$15.00

NULL.COM \$10.00

All CP/M software is suitable for DISK use only and the prices shown DO NOT INCLUDE PRICE OF DISK. If you don't supply disks please add normal \$6.00 media charge to the price shown against the item you want. Those CP/M items with an asterisk after the program name fill a normal SS/DD disk and other items can not be included on the disk. Example.....if you want MODEM7 and XDIR.COM you will need two disks.

buy, trade & sell

HOME OFFICE

Package containing three programmes:

WRITER: Word processor (wordwrap, print controls, indent, RMJ etc.)
FILER: Data Base (multi-search, tab printout, sum/average, sort)
CHEQUE: Cheque account/budget programme (Category search, printouts, auto-number, verify etc.)

COST \$30.00+\$2.00 post & pack
(\$35.00+\$2.00 for disc)

Write specifying MSX or SV to:-
DOWNS HOME COMPUTING
HOOPER CENTRE
HUME ST.,
TOOWOOMBA, QLD., 4350

FOR SALE:

COMBINATION RS232/MODEM TO SUIT SV-328 SYSTEM.
CCITT & BELL FREQUENCIES. 300 FULL DUPLEX.
1200 & 600 HALF DUPLEX WITH 75 BAUD BACK CHANNEL.
ALL OPTIONS ARE SOFTWARE SELECTABLE: NO SWITCHES TO SET.
AUTO-DIAL/AUTO-ANSWER. INCLUDES COMMUNICATIONS
SOFTWARE (INCLUDING VIATEL SOFTWARE). PLUGS DIRECTLY
INTO EXPANDER. PRICE = \$350.

CONTACT:

BRIAN PARKER
19 BARCELONA ST,
WISHART QLD 4122,
PHONE (07) 3490315.

SPECTRAVIDEO 318 324 209. 2 AM
512K RAM - 80K DATA-EXPANDER.
Approx. 150 PROGRAMS INCLUDING
GAMES, DATABASE, DBase, ACCOUNTS,
MSX - ETC. ALL IN 5.25" DISKS.
\$265 ONC. REPLY TO: WATSON PHONE:
STD 071-442917 OR 22 HAZARD ST,
KWINNA CUTTERS, QLD 4575

buy, trade & sell

FOR SALE:
SV-328 WITH SV-903 CASSETTE DECK AND THREE PROGRAMS = \$290.
TWO SV-902 DISK DRIVES = \$250 EACH.

CONTACT:
BRIAN PARKER,
19 BARCELONA ST,
WISHART QLD 4122,
PHONE (07) 3490315.

Sell's (318/328 Games) Tetra Horror
Turboot } \$20 each o.n.o.
Ninja
SV-105 Graphic Tablet
(Comes with graphicspro-} \$120 o.n.o.
gramme & manual)
Frantic Freddy } \$30.
(on cartridge)

PETER J. WHITE
P.O. Box 35
Macquarie
A.C.T. 2614

For Sale
1 SVI Graphics tablet for \$120
1 game for Spectra - trader
Write to: for \$15.00
16 Cameron Donnelly
3084, VIEWBANK
VIC.