VOL. 2 ISSUE 8 MAY 1986

MEMOTECH OWNERS CLUB

CIRCA ...247

M.O.C.

VOLUME 2 ISSUE NUMBER 8

CONTENTS

o o o o o o o o o

If anyone has any good graphis designs for a front cover then we would love to see them!!!

E D I T O R I A L (May 1986)

Phil Eyres
23 Denmead Road
Harefield
Southampton
SO2 5GS

This month we have an excellent assembler program (see pages 4 & 5), it was sent in by Dave Dulson, who obtained the original (non-working) code from us, after we had been sent it by a member. Dave has done a great job in producing the Function Key Definer, which I hope you all agree is a program we have been in need of for some time.

Alan Dobsons Basic program (on pages 2 - 4) is also an excellent offering, well worth typing in. The program, called CardTric will deal three piles of cards on the screen, while it is dealing pick a card and remember it, when the computer has finished dealing tell it what pile the card was in, after telling it which pile your card is in three times - like magic - the computer will tell you which card you had picked!!. Oh! The graphics are nice as well.

Over the past couple of years in which we have been running the club we have come to know our postman fairly well, he has done a truly tremendous job delivering your mail, but, a couple of weeks ago he informed us that whilst on his round, his post bag had been stolen, this happened before he had delivered any mail to us, whilst he is not sure that he had any mail to deliver to us, we do normally get some mail every day, so the chances are a letter or two may not have arrived, if anyone is awaiting a reply and have been doing so for two or more weeks then please get in touch because we may not know that you've written.

We've had a very good response to 'ast month's request for High - Scores, to see if your'e still on top, go to the Letters Page, page 9. Ed-> I must try and find a game no-one else plays so that I can get in the High Scores table!.

Thanks to everyone who has used our Hotline on Monday evenings between 6 & 7pm, the number to phone now is (0703) 466106, ask for Phil. If we keep Mondays as Hotline night then I can be sure of being in. However, feel free to phone any evening after 6pm, if I'm not in the my Mum (good old Mum!!) will take any calls.

If anyone would like back issues they are available for the small remittance of 80p. At present there are 17 back issues, 10 for volume 1 and 7 for volume 2.

oooOOOooo

```
10 REM  ******************
20 REM  ******************
30 REM  **                **
40 REM  **  CARD  TRICK   **
50 REM  **                **
60 REM  **      by        **
70 REM  **                **
80 REM  **  ALAN DOBSON   **
90 REM  **                **
100 REM ******************
110 REM ******************
120 VS 4: COLOUR 2,2: COLOUR 0,2: COLOUR 1,1: COLOUR 4,10: CLS
130 GENPAT 1,133,76,210,82,82,82,82,236,0
140 GENPAT 0,ASC("^"),90,189,102,219,219,102,189,90
150 GENPAT 1,130,0,16,56,124,254,124,56,16
160 GENPAT 1,131,0,108,254,254,254,124,56,16
170 GENPAT 1,132,16,56,124,254,254,254,16,56
180 GENPAT 1,134,16,56,16,84,254,84,16,56
190 GENPAT 1,140,60,60,60,60,0,0,0,0
200 GENPAT 1,141,0,0,0,0,60,60,60,60
210 CSR 10,0: PRINT "CARD TRICK"
220 PRINT : PRINT " 12 cards will be shown-one at a time-in three pile
s."
230 PRINT : PRINT " While this is happening, please pick one of the ca
rds.": PRINT : PRINT " Then please tell me which pile  it is in."
240 PRINT : PRINT " This will occur 3 times.  After the third time I w
ill show you  the card you chose."
250 PRINT : PRINT : PRINT : PRINT " Press any key to start....
"
260 DIM A$(8,5),TEMP$(21,2)
270 DIM T$(11,5)
280 LET A$(1)=" ^^^ "
290 LET A$(2)="     "
300 LET A$(3)=" "+CHR$(140)+CHR$(141)+CHR$(140)+" "
310 LET A$(4)=" "+CHR$(140)+" "+CHR$(140)+" "
320 LET A$(5)=" "+CHR$(141)+" "+CHR$(141)+" "
330 LET A$(6)="  "+CHR$(141)+"  "
340 LET A$(7)="  "+CHR$(140)+"  "
350 LET A$(8)="    "
360 LET T$(1)="11112"
370 LET T$(2)="68682"
380 LET T$(3)="76872"
390 LET T$(4)="48842"
400 LET T$(5)="46842"
410 LET T$(6)="45842"
420 LET T$(7)="35842"
430 LET T$(8)="44442"
440 LET T$(9)="43442"
450 LET T$(10)="34342"
460 LET T$(11)="86882"
470 DIM C(21)
480 DIM S(21)
490 IF INKEY$="" THEN GOTO 490
500 CLS
510 PRINT "^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^"
520 FOR X=0 TO 22
530 CSR 0,X: PRINT "^": CSR 31,X: PRINT "^"
540 NEXT X
550 CSR 0,23: PRINT "^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^";
560 CSR 10,0: COLOUR 0,15: COLOUR 1,1: PRINT " CARD TRICK ": COLOUR 0,
2
570 CSR 5,10: PRINT "PLEASE WAIT - SHUFFLING"
580 DIM W$(22,2)
590 LET Z$="A23456789"+CHR$(133)+"JQK"
600 LET D$=CHR$(134)+CHR$(130)+CHR$(131)+CHR$(132)
610 FOR Q=1 TO 21
620 LET Z=INT(RND*13)+1: LET C(Q)=Z
630 LET K=INT(RND*4)+1: LET S(Q)=K
640 SOUND 0,16*Q,15
650 LET O$=Z$(Z)
```

```
660 LET L$=D$(K)
670 LET G$=O$+L$
680 FOR N=1 TO 21
690 IF W$(N)=G$ THEN GOTO 620
700 NEXT N
710 LET W$(O)=G$
720 NEXT O
730 SOUND 0,0,0
740 CSR 5,10: PRINT "                                "
750 CSR 6,5: PRINT "<1>      <2>      <3>"
760 FOR PL=1 TO 3
770 LET A=1: LET AA=1
780 LET C=C(A)
790 LET S=S(A)
800 LET C$=LEFT$(W$(A),1)
810 LET S$=RIGHT$(W$(A),1)
820 IF C>10 THEN LET C=1
830 IF S=1 OR S=4 THEN COLOUR 1,1
840 IF S=2 OR S=3 THEN COLOUR 1,6
850 COLOUR 0,15
860 CSR (AA-1)*8+5,7: PRINT " ";C$;" ";S$;" "
870 CSR (AA-1)*8+5,8: PRINT "     "
880 FOR Y=1 TO 5
890 SOUND 1,100*Y,15
900 CSR (AA-1)*8+5,8+Y: PRINT A$(VAL(T$(C,Y)))
910 NEXT Y
920 SOUND 1,0,0: PAUSE 1500
930 LET A=A+1: LET AA=AA+1: IF AA=4 THEN LET AA=1
940 IF A=22 THEN GOTO 970
950 GOTO 780
960 STOP
970 COLOUR 0,2: COLOUR 1,1: CSR 8,18: PRINT " Which Pile ? "
980 LET KE=VAL(INKEY$)
990 IF KE<1 OR KE>3 THEN GOTO 970
1000 CSR 8,18: PRINT "                  "
1010 FOR AA=1 TO 3
1020 FOR Y=7 TO 0 STEP -1
1030 CSR (AA-1)*8+5,6+Y: PRINT "      "
1040 NEXT Y: NEXT AA
1050 ON KE-1 GOSUB 2000,3000,4000
1060 NEXT PL
1070 FOR AA=1 TO 3
1080 FOR Y=7 TO 0 STEP -1
1090 CSR (AA-1)*8+5,6+Y: PRINT "      "
1100 NEXT Y: NEXT AA
1110 CSR 6,5: PRINT "                        "
1120 COLOUR 0,15
1130 FOR FI=1 TO 11
1140 LET C=C(FI): LET S=S(FI)
1150 LET C$=LEFT$(W$(FI),1): LET S$=RIGHT$(W$(FI),1)
1160 IF S=1 OR S=4 THEN COLOUR 1,1
1170 IF S=2 OR S=3 THEN COLOUR 1,6
1180 CSR 13,7: PRINT " ";C$;" ";S$;" "
1190 CSR 13,8: PRINT "     "
1200 FOR Y=1 TO 5
1210 SOUND 1,100*Y,15
1220 CSR 13,8+Y: PRINT A$(VAL(T$(C,Y)))
1230 NEXT Y
1240 SOUND 1,0,0: PAUSE 1500
1250 NEXT FI
1260 COLOUR 1,1: COLOUR 0,2
1270 CSR 5,18: PRINT "THIS IS THE CARD CHOSEN"
1280 PAUSE 5000
1290 CSR 2,18: PRINT "PRESS ANY KEY FOR ANOTHER GO"
1300 IF INKEY$<>"" THEN GOTO 1300
1310 IF INKEY$="" THEN GOTO 1310
1320 RUN
1330 STOP
```

```
2000 REM ************************
2010 REM ************************
2020 REM **                    **
2030 REM **  CHOOSE PILE ONE   **
2040 REM **                    **
2050 REM ************************
2060 REM ************************
2070 LET T=1
2080 FOR TE=2 TO 20 STEP 3
2090 LET TEMP$(T)=W$(TE): LET T=T+1
2110 NEXT TE
2120 FOR TE=1 TO 19 STEP 3
2130 LET TEMP$(T)=W$(TE): LET T=T+1
2150 NEXT TE
2160 FOR TE=3 TO 21 STEP 3
2170 LET TEMP$(T)=W$(TE): LET T=T+1
2190 NEXT TE
2300 GOTO 6000
3000 REM ************************
3010 REM ************************
3020 REM **                    **
3030 REM **  CHOOSE PILE TWO   **
3040 REM **                    **
3050 REM ************************
3060 REM ************************
3070 LET T=1
3080 FOR TE=1 TO 19 STEP 3
3090 LET TEMP$(T)=W$(TE): LET T=T+1
3110 NEXT TE
3120 FOR TE=2 TO 20 STEP 3
3130 LET TEMP$(T)=W$(TE): LET T=T+1
3150 NEXT TE
3160 FOR TE=3 TO 21 STEP 3
3170 LET TEMP$(T)=W$(TE): LET T=T+1
3190 NEXT TE
3200 GOTO 6000
4000 REM ****************************
4010 REM ****************************
4020 REM **                        **
4030 REM **  CHOOSE PILE THREE     **
4040 REM **                        **
4050 REM ****************************
4060 REM ****************************
4070 LET T=1
4080 FOR TE=1 TO 19 STEP 3
4090 LET TEMP$(T)=W$(TE)
4100 LET T=T+1
4110 NEXT TE
4120 FOR TE=3 TO 21 STEP 3
4130 LET TEMP$(T)=W$(TE)
4140 LET T=T+1
4150 NEXT TE
4160 FOR TE=2 TO 20 STEP 3
4170 LET TEMP$(T)=W$(TE)
4180 LET T=T+1
4190 NEXT TE
4200 GOTO 6000
5000 SAVE "Card Trick"
5010 RUN
6000 FOR TE=1 TO 21
6010 LET W$(TE)=TEMP$(TE): LET C(TE)=VAL(LEFT$(W$(TE),1))
6020 IF ASC(W$(TE))=133 THEN LET C(TE)=10
6030 IF ASC(RIGHT$(W$(TE),1))=132 THEN LET S(TE)=1
6040 IF ASC(RIGHT$(W$(TE),1))=130 THEN LET S(TE)=2
6050 IF ASC(RIGHT$(W$(TE),1))=131 THEN LET S(TE)=3
6060 IF ASC(RIGHT$(W$(TE),1))=134 THEN LET S(TE)=4
6070 IF ASC(LEFT$(W$(TE),1))=65 THEN LET C(TE)=1
6080 IF C(TE)=0 THEN LET C(TE)=1
6090 NEXT TE
6100 RETURN
```

```
5 REM FUNCTION KEY DEFINER
10 CODE

8022 START:   CALL TRANSF          ;transfer memory routine
8025          CALL SETINT          ;set up interupts routine
8028          CALL LOWMEM          ;lower memory top routine
802B          CALL INTON ;turn interrupts on routine
802E          CALL DELETE          ;delete program from memory routine
8031          RET        ; return to basic.
8032 KEYS:    DI ;this is the routine that
8033          LD A,(£FD7D)         ;handles the function keys and
8036          CP 128     ;is called by the MTX interrupt
8038          JR C,OUT   ;system 64 times a second
803A          CP 144
803C          JR NC,OUT
803E FUNCT:   SUB 127    ;convert the ascii character to the funkey num
8040          LD B,A     ;load B with A for loop counter
8041          LD DE,FK0  ;load DE with the base address of funkeys
8044 FINDLP:  LD A,(DE)  ;scan through funkey area until
8045          INC DE     ;the required key is found
8046          CP 255     ;funkey address now held in DE
8048          JR NZ,FINDLP
804A          DJNZ FINDLP
804C PRINT:   LD A,8     ;delete last character typed this is the
804E          RST 28     ;funkey character
804F          DB £AC
8050 PRLOOP:  LD A,(DE)  ;print out character for required funkey
8051          CP 255
8053          JR Z,OUT
8055          RST 28
8056          DB £AC
8057          INC DE
8058          JR PRLOOP
805A          RET
805B OUT:     LD A,0     ;places value of 0 in the last character typed
805D          LD (£FD7D),A         ;to avoid repetition  of function key
8060          EI ;enable interrupts
8061          RETI       ;return from interupt routine.
8063 TRANSF:  LD DE,(STPROG)       ;this routine moves the
8067          LD HL,KEYS ;handler routine KEYS into
806A          LD BC,(STSIZE)       ;high memory so it is invisible
806E          LDIR       ;to the user
8070          RET        ;return from routine
8071 LOWMEM:  LD DE,(STPROG)       ;this routine lowers the top of
8075          DEC DE     ;memory so the program doesnot
8076          LD (£FA92),DE        ;get over run by variables or
807A          RET        ;other program data
807B DELETE:  LD DE,£4000          ;this routine deletes the
807E          LD (£FAA7),DE        ;program from lower memory
8082          RET        ;return from routine
8083 SETINT:  LD A,£C3   ;this routine sets up an interupt
8085          LD (£FA98),A         ;table to point to the
8088          LD HL,(STPROG)       ;routine KEYS in high memory
808B          LD (£FA99),HL
808E          RET        ;return from routine
808F INTON:   LD A,£FF   ;this routine turns
8091          LD (£FD5E),A         ;the MTX interupt system on
8094          RET        ;return from routine
8095 INTOFF:  LD A,£0F   ;this routine turns the
8097          LD (£FD5E),A         ;MTX interupt system off
809A          RET        ;return from routine
809B MAP:     LD DE,FK1  ;this routine will print a list
809E          LD B,16    ;of all the functiom keys
80A0          LD A,"0"   ;currently set up
80A2          LD IX,£F0AE          ;this routine is completely
80A6          LD (IX+1),A          ;optional and can be left out
80A9          LD A," "   ;to save space if needed
80AB          LD (IX+0),A
80AE          RST 10     ;clear screen
```

```
80AF            DB 129,12
80B1 MLOOP:     LD A,(IX+1)
80B4            INC A
80B5            LD (IX+1),A
80B8            CP 58         ;if overflow occurs then set
80BA            JR NZ,SKIP ;ascii chars for keys above 9
80BC            LD A,"0"
80BE            LD (IX+0),A
80C1 SKIP:      RST 10        ;print out function key num
80C2            DB 137,"KEYS "
80C8 NUM1:      DB " "
80C9 NUM:       DB "0= "
80CC MLOOP2:    LD A,(DE)   ;this code prints out the
80CD            INC DE        ;value of the function key
80CE            CP 255
80D0            RST 28
80D1            DB £AC
80D2            JR MLOOP2
80D4            JR MLOOP2
80D6 BACK:      RST 10        ;prints out a return and
80D7            DB 130,13,10        ;new line character
80DA            DJNZ MLOOP
80DC            RET
80DD FK0:       DB " ",255 ;base area for function keys each key
80DF FK1:       DB "ASSEM ",255     ;is a line of ascii characters
80E6 FK2:       DB "PANEL ",255     ;followed by an end byte of 255
80ED FK3:       DB "USER SAVE ",34,255      ;as the char : and " can't
80F9 FK4:       DB "USER LOAD ",34,255      ;be used in assembler
8105 FK5:       DB "LIST",255       ;use the codes 58 and 34 instead
810A FK6:       DB "SAVE ",34,"PROG",34,58,"VERIFY ",34,"PROG",34,255
8124 FK7:       DB "RUN ",255
8129 FK8:       DB "USER ",255
812F FK9:       DB "USER DIR",255
8138 FK10:      DB "USER STAT ",34,255
8144 FK11:      DB "USER  ERA ",34,255
8150 FK12:      DB "USER COPY ",34,255
815C FK13:      DB "USER READ ",34,255
8168 FK14:      DB "USER WRITE ",34,255
8175 FK15:      DB "USER FORMAT ",255
8182 FK16:      DB "USER REN ",34,255
818D STPROG:    DW £F015    ;start of progr in upper memory
818F STSIZE:    DW £0200    ;length of program
8191            RET
8192            RET

Symbols:
TRANSF   8063     START    8022
SETINT   8083     LOWMEM   8071
INTON    808F     DELETE   807B
KEYS     8032     OUT      805B
FUNCT    803E     FINDLP   8044
PRINT    804C     PRLOOP   8050
STPROG   818D     STSIZE   818F
INTOFF   8095     FK1      80DF
MAP      809B     MLOOP    80B1
SKIP     80C1     NUM1     80C8
NUM      80C9     MLOOP2   80CC
BACK     80D6     FK0      80DD
FK2      80E6     FK3      80ED
FK4      80F9     FK5      8105
FK6      810A     FK7      8124
FK8      8129     FK9      812F
FK10     8138     FK11     8144
FK12     8150     FK13     815C
FK14     8168     FK15     8175
FK16     8182
```

. . . This Is Only A Test
Part II

In the first part of this article published in the February Magazine, we said that programming problems could be broken down into three types : numbers, characters and applications. We then went on to describe two number type tests, the Prime numbers problem and the Smith numbers problem.

This month we move on to character problems, these should appeal more to the less mathamatically minded people, as they purely involve 'mucking about' with character strings.

Below there are two problems, the first is designed to be fairly simple (just to give you some confidence!) and the second a bit more difficult (just to deaden any confidence!).

Problems
Exercise 1 - Number of Symbols

Definition: A "string" is any combination of blanks, letters, digits and/or special characters.

Problem: Calculate and print the number of different symbols used in a given string. A blank space counts as one symbol.

Input: Any String. The length is limited by 60 characters. Use the following string as test data, "kiss me good bye!"

Output: The number of different symbols. For example, if the input string is "mississippi", the output should be "Number of Symbols = 4"

Exercise 2 - Word Frequency

Definition: A string is any combination of blanks, letters, digits and/or special characters. Words are groups of characters separated by blanks.

Problem: Count the frequency of use of each word in a string that consists of several words.

Input: Any string of words. The length of the string is limited to 60 characters. To simplify the exercise assume that the string contains only letters, the length of a word is less than 20 characters and there are no more than 20 words. Use the following test data, "so do la fa mi do re so do la si do re do".

Output: The format of the output should list each different word and the frequency of use of that word. Print each word on one line. There is no special order; you do not have to sort the list. For example, if the input string is "never say never again" the output should be:
never 2
say 1
again 1
Ed-> As before a prize for the quickest, most compact program!!

Advertisment

62 Characters on a line

This program gives 62 characters on a single line in the VS 4 mode. You can mix both 32 and 62 characters on the same line, and there is full CSR and colour control.

You can use it in both new and old programs, by only changing the print statement, and add a single line to the old program. Full M/Code.

TAPE D.KR. 70,00

MINITEKST (Word-processing-program)

This program makes use of the program described above, and other features are:

- Change Fonts as you wish on the same line. Also on screen.
- Straight right margin
- Edit in an existing line
- Save and load data blocks.
- Block moves
- Erase Blocks
- Find and change a letter in a word.
- Build a printer driver to work with any printer.
- Mostly Menu direct.
- Full M/Code.

TAPE D.Kr. 170,00

Please write to : Leif Mortensen
Binnitsevej 8
DK 4930 MARIBO
DENMARK.

- -

INTERFACING PROJECTS

Why not make a break which will lead you into the exciting world of micro electronics. Infact what better way to start than with an MOC D.I.Y. kit. Everything you need is supplied, except a soldering iron, wire cutters and of course a few hours of your time!!. So why not order now.

Interface price list

A full set of components and instructions for the LED kit. -->£6.95
A full set of components and instructions for the Speech Synthasiser kit-->18.00
Connecting cable for the internal port (needed for projects) -->£4.50
All prices are fully inclusive. Please allow 14 days for delivery and make cheques payable to MOC.

YOUR LETTERS

** Games High Scores Table **

AGROVATOR	61828	A.DOBSON
ASTROMILON	30830	T.NEAL
ASTROPAC	69390	A.DOBSON
BLOBBO	71233	T.PICKSTONE
B.BILL	219610	A.DOBSON LEVEL 1
B.BILL	158334	A.DOBSON LEVEL 9
CHAMBEROIDS	19 MINS	P.ERIKSSON
COBRA	5634	A.DOBSON
CONT RAID	10810	M.GILL
CRYSTAL	35507	*A.LYNCH
DR FRANKY	14925	*N.CRIGHTON
D.DANGER	8087	*V.STEPNEY
D.DESTROYER	3380	T.NEAL
EMERALD ISLE	725	R.SIDDALL
E. ZARKOS	90 OBJ	R.SIDDALL
F. DEEP	1420	A.LYNCH
FELIX	20600	P.COUGHLAN
F.FREDDIE	15560	*M.FIDLER
FLUMMOX	25700	T.NEAL
GOLDMINE	6308	*M.FIDLER
HAWKWARS	15850	P.CRIGHTON
HUNCHY	5681	T.NEAL
ICEBURG	17431	A.DOBSON
JUMP' J FLASH	2970	T.NEAL
KARATE KING	1300	T.NEAL
KILOPEDE	35275	*N.CRIGHTON
KNUCKLES	488650	P.CRIGHTON
L OF TIME	950	R.SIDDALL
MAXIMA	501250	R.SIDDALL
MINER DICK	22520	R.SIDDALL
MISS ALPHA	53320	*P.CRIGHTON
M OMEGA	4400	T.NEAL
NEMO	14650	*P.CRIGHTON
OBLOIDS	80110	*P.CRIGHTON
PHAID	5285	*M.FIDLER
P PETE	39630	A.DOBSON
QUAZZIA	41020	*V.STEPNEY
QOGO	11440	*M.FIDLER
QOGO 2	255000	R.SIDDALL
ROLLA BEAR	27741	*V.STEPNEY
SEPULCRI	6175	*V.STEPNEY
S.M.G.Rt	26280	*V.STEPNEY
S.M.G.Lt	11830	*V.STEPNEY
SNAPPO	79300	P.ERIKSSON
SNOWBALL	1000	P.COUGHLAN
S OF PETE	10542	P.ERIKSSON
STAR COMM	131690	*P.CRIGHTON
SUPERBIKE	20.7KM	*A.FIDLER
S M/FIELD	829	M.GELDER
S SCANNER	7340	A.DOBSON
T FIGHTER	3260	*V.STEPNEY
TAPEWORM	168515	A.DOBSON LEVEL 1
TAPEWORM	150500	A.DOBSON LEVEL 9
T ZONE	7610	P.ERIKSSON
TOADO	107549	N.GOODING
TURBO	23030	M.GELDER

* Denotes new high score

A Request For Help

1.Dave Copeland of the Australian Embassy, Switzerland has put forward a request for help.

He has a professional applications package called NOVA, which he bought from a firm in America. The package is designed to run on CP/M systems, which as we all know is a bit of a farse!. The program has an Installation routine, much like many other commercial application programs, this is mainly to set up the correct control codes for writing to the screen. The installation program comes with several options, none of which suit the Memotech, there is an option to declare your own Control codes, but this seems to fall over when you try and tell it the Memotech requires a screen offset of 32 in order to keep things correct. This has stifled both Dave and us, could someone therefore help, it may require the use of DDT in order to patch some code in. Dave is willing to recompense anyone who can find a solution for him.

If you would like to have a go, please ring or send a letter to Phil (MOC [0703] 466106) and I'll forward the diskette.

2. A request for help with the VDP.
Can anyone supply a routine that will print ascii characters on a graphics screen using assembler??

All routines gratefully received.

3. This is a request from MOC.
Quite often when we are making a magazine we have to combine a Basic (or Assembler) program with some text, this has to be done in two separate steps, first make a master of the text, leaving room for the program listing to be inserted, them insert the program. It would be more versatile if we could create an Ascii file which looks identical to the program, which could then be merged with a Newword document. Can anyone produce, preferably a disc based routine, that would create an ascii look-a-like file of a program??

4. A request from Leif Mortensen from Denmark

I have a problem with a program, I hope you can help me. It is about saving programs on cassette. I have two nearly separate M/C programs, with the main program before the

sub program. The sub program is used first then never again, so instead of saving it on tape together with the main program, my idea was, in the main program, to set the system variable FAA4 and FAA5 to the new top of program, then save it on tape from a Basic line after the sub program, to get it to auto-run after loading from cassette, But it will not work!!. If I do it in direct mode, it works but not in the program. Now, the big question, what system variable determines the top of what is to be saved on tape, and how can I control it??

5. Lastly A request From Peter Crighton.

Could you pass on to members via the magazine, my request for self-adhesive cassette labels on tractor feed paper. If any one can obtain small quantities could they please contact me.

Pete's address is, 14 Stornaway Strand, Gravesend, Kent.
Ed-> The club would also be interested to find a supplier.

Hints And Tips
1. Some AGROVATOR Pokes from Paul Wood of Worcester
Load in program and when running, press <BRK> then press the following keys, B,Y,ESC,B,0

```
POKE 16640,X   X= No of Bullets
POKE 16619,X   X= No of Lives
POKE 16568,X   X= Speed of Player (originally 24)  1 is
               fastest
POKE 16573,X   X= Speed of Ghost  (originally 23) 255 is
               slowest

Then either RUN - SAVES program with MODS
     or GOTO 10 - RUNS modified program
```

2.Two tips from Peter Crighton of Kent.
a) When writing a program, if an error occurs in a line containing RETURN, check the line that called it. If this is a multi-statement line and the GOSUB is not the last statement, then the error may lie in that line (after the GOSUB), although the error detector brings down the line containing the return.

b) When a virtual screen is created, scrolling or paging mode can be selected by printing characters. In the scrolling mode the bell always sounds when the screen fills - particularly annoying if you want to auto-scroll. Anyway true to Memotech style it is user selectable (one of the things I like about the Memotech is that everything seems to be user controlled). If you look in the manual at the appendix containing system variables there are 15 bytes for each virtual screen from FF5D to FFD4. The first byte of this 15 contains various bits controlling the screen. Poking this byte (i.e. FF5D, FF6C, FF7B etc.) with 0 will set the screen to auto scroll. I am not sure which bit effects this, but if you require to set one of the other controls, then I suggest you play around with the screen data (as I have done) until the screen does as you require.

Interest
1.This new release might be of use to many, ...a new, long awaited operators manual with contents as follows:-
Sound,Graphics; High Res & Sprites, Noddy, MTX Assembly, Technical Appendix, Memory Maps, Z80 Programming Summary, MTX Disc ROM Expansion, DMX80 Printer, PAL Listing, System Variables.

This manual is available from :-
Phoenix Publishing Associates LTD
14 Vernon Road
Bushey
Herts
WD2 2JL
For £8.95 + £1 P&P. This price includes a free CRIB card with every order.

2.This is a better version of last month's Pascal PRIME Numbers program, written by Andy Capon.

```
PROGRAM PRIMES
VAR
  SCAN,NUMBERS : INTEGER;
BEGIN
  NUMBER := 101;
  REPEAT
    SCAN := (NUMBER DIV 2) + 2;
    REPEAT
      SCAN := SCAN - 1;
    UNTIL (SCAN = 2) OR (NUMBER MOD SCAN = 0);
    IF SCAN = 2
    THEN
      WRITE (NUMBER);
    NUMBER := NUMBER - 2;
  UNTIL NUMBER < 0;
END.
```

This ran in approximately half a second, as well as being faster, it is easier to read and understand.

THE "O" OPTION

by Geoff Gardiner

The FDX Operator's Manual contains a good summary of the CP/M disc operation system, but it is not comprehensive. It was sensible to leave out the the details for using "ED", as no-one who has the facility of using Newword in non-document mode to write a program is going to want to use the complex and inconvenient "ED".

But the ommission of a description of the "O" option for PIP is serious, and will have caused problems for those who have given a file extension type "BAS" to their programs written in disc basic, or indeed any file extension type other than "COM", for the chances are that such files cannot be copied with "PIP".

The trouble arises from the fact that CNTRL Z, or 1A in Hex, is interpreted as an end of file marker or data terminator when PIP is copying, transmitting, or receiving an ASCII type file. But in an object code file such as a program 1A is likely to turn up anywhere in the file, for it is the hex code for the frequently used Z80 source statement "LD A,(DE)". "PIP" knows that this problem exists with "COM" type files and when copying these files does not come to a halt when it encounters "1A". But with any other type file "PIP" has to be told to ignore the 1A's, and this is done by using the "O" option. It is very simple: when you have typed in your normal PIP instruction, continue with space, open square bracket, O, close square bracket. Probably it is already your practice to put V in square brackets at the end of your PIP instruction in order to have the transfer verified (even though Gary Kildall, the inventor of CP/M reckons there is no need to use the "V" option) so in future put VO in the brackets when copying program files. Of course there is an alternative: you could give all your program files the COM extension, but that is likely to cause confusion.

MORE RS232 TROUBLES

By Geoff Gardiner

Last November (vol2 Issue3 Page 9) I revealed that the software for the FDX does not follow the true RS 232C standard, and that the Write Register 5 of the Dual Asynchronous Receiver Transmitter (DART) is fed with the wrong control byte with the result that the two serial ports (SIOA and SOIB) do not function if the cabling connecting them to peripherals is wired in accordance with the RS232 convention. I gave details of the patches that one needs to make in the CONTACT.COM and BAUD.COM programs in order to correct them.

Having got my serial connections to work I decided to use CONTACT.COM and it's associated H.COM to download a program in INTEL HEX format to my printer so that I could study the code at my leisure. I decided to make my task as difficult as possible by connecting the printer (a Brother EM200 typewriter) to channel B instead of channel A which is the customary printer connection. It was an opportunity to test my mastery of RS232 by using the channel that has all the handshake lines; moreover channel B and the interface to the printer are both configured as "Data Terminals", which means that the wires have to be crossed over,pin 2 of one having to be connected to pin 3 of the other,etc. All worked perfectly until I tried to stop and start the flow using the "z" command. Pressing "z" stopped the flow alright but would not start it again. I had set half duplex mode so the output to the printer was also going to the screen and I could see that there was a flow but it was not getting through to the printer. Had I bungled the wiring or was there another software problem? Having had the most expert guidance on RS232 from my son who designs modems, I decided to look at the software which meant using DDT to examine the 10000 bytes or so of CONTACT.COM.

This is not such hard work as it sounds as the fact that DDT cannot read Z80 code provides a telltale. The programs are almost wholly 8080 code but the Z80 "OUT (C),A" instruction is often used, and I was looking for output to port OF of the hex numbers 05 and 68. When DDT cannot disassemble code it displays query signs followed by the code it cannot read, so all I had to do was to look for the query signs. Incidentally the new Z80 codes are all two,three, or four bytes, and altough DDT cannot disassemble the first byte it mistakenly thinks it knows what the remaining bytes of the instruction mean so it preceeds to display garble. Succeeding instructions can be corrupted as well by a knock on effect. I soon found two "OUT" instructinos for hex 78 and 68. I tried to see if the code for "compare z" led to these instructinos but the program was to involved for me to understand and I had to trust that I had guessed correctly the location of the errors.

But why was "78" being output to write register 5 of the DART? An examination of the Mostek Data Book showed that it set the same bits 68 plus the "break" bit.I had wondered what the break bit did and I now deduced that its effect is to switch off the data flow even though all the other bits for data transmission remains set. It is a way of interrupting the flow temporarily by altering one bit alone. Pressing Z must result in 78 being sent to WR 5, and pressing it again sends out 68. These I changed to FA and EA respectively, using the DDT's "s" instruction. The locatios are 1EA7 for FA, and 1EBB for EA. The changes had the desired effect.

The routine is to call DDT CONTACT.COM, and when this has been loaded type S1EA7 (RET). The screen will now display 1EA7 78. Type FA (RET), and the screen will acknowledge by redisplaying 1EA7 FA. Type a full stop followed by (RET). Now you can make the second amendment by typing S1EBB (RET), followed by EA (RET) and . (RET). Then press CNTRL-C to exit from DDT. To save the amended program wait for the prompt and then type SAVE 40 CONTACT.COM, and your old program will be overwritten and replaced by the new one, provided you have not write protected your disc!. My manual does not tell me how to use DDT's instructions, nor many other CP/M instructions, so I have acquired at the ghastly expense of £15.95 the Osborne/McGraw-Hill CP/M User Group, Third Edition. I am afraid that it is essential.

When I am using Contact and "H", I find I have to set "Screen Filter" and "Flow Control" OFF. If I want to start printing part way through a file I set HALF DUPLEX and with the printer disconnected set the flow going, watch the flow on the screen, and stop it with "z" some way in front of the point at which I want to start printing. One cannot stop it very accurately because one has to switch to the menu (with CNTRL-G) before "z" will work, and meantime the data is streaming through. Then I change to flow control ON, press "z", now the data is displayed one line at a time, using line feed, and although there is a lot of fiddle switching between the menu and the contact screen, one can eventually get to the exact place one wants. Press "z" switch on the printer, switch flow control off, press "z" again, and away we go. I can stop printing to change paper by the "select" switch on my interface, and this causes my RS232 handshaking lines to do their job, that is tell the DART to "lay off".

Our Swiss friends usually end the articles in their mag with the words "Viel Spass", I will use our equivalent, "Have lots of fun!".

P R O G R A M L I B R A R Y
£1.20 Per Cassette, 2 Programs per Cassette

Two new offerings this month, a Skittle's program from Paul Wood, and a cassette label routine from Peter Crighton. Please keep sending in your programs for the library, they are appreciated by everyone.

1.Basic & Assembler Programs

All programs available on cassette, 2 programs per cassette, £1.20 per cassette. Or on disc, £2.50 per disc, please enclose a disc, stating capacity.(Some programs are only available on cassette!!).

Reviews of all programs are available, please send a large SAE. All Swiss User Group programs are prefixed with 'Sw'.

The first 15 programs have had to be omitted this month due to lack of space, please refer to previous magazines for full details.

16.RELOC Relocs Assembler Properly!!
17.Character Editor Yepp!! Another Sprite Gen!!
18.Quasimodo Excellent Arcade Game
19.Planner YAS6 (Yet Another Sprite Generator)
20.Hanoi Classic Puzzle (Brilliant simple use of
21.Nobie Simple Text Game Graphics)
22.Hi-Lo Just like Bruce's Play Your Cards Right
23.Composer Our First Sound Generator!!
24.Anova Applications Program
25.CASHFLOW Applications Program
26.RenumIII Utility !!!26,27 & 28 cassette only!!!
27.Merge Utility
28.Money Manager Applications program
29.Word Word Processor
30.Reversi Strategy Board Game
31.Full Time Football Manager Game
32.PANEL3 Panel extensions
--- The Second Disc Starts Here ---
33.Texted Word-pro
34.SwMice Swiss Arcade game Written in Basic
35.TNTTIM Assembler arcade game.
36.Sw3D-FUNC.1 First of two. Saturn!!!
37.Sw3D-FUNC.2 Second of Above. Sinpr?.
38.SwSpr-Ed YAS6.
39.SwZ-Wandl Number Base Convertion Prog.
40.OXO Noughts & Crosses.
41.Solitaire Strategy Game.
42.Cross-Num Excellent strategy game!!
43.Avoid Seven Dice Game
44.Numerology Analyse your name!!
45.Chemin Another Dice Game!
46.Dice Another, Another Dice Game!!
47.SwMathe Arithmetic Tester.
48.Reversi2 Assembler of no. 30. Great!!
49.ISOT A really good maze game.
50.DBaseI Simple Data Base
51.DBaseII Requires MTX Util Tape
52.Money 2.1 An update of no 28
53.Ram Disc Better than sliced bread
54.RDisc Source Source of above.
55.Diary Diary & Address program
56.Terminal Em. Comm's via Rs232 & Modem.
57.Skittles *** NEW *** Keeps league tables
58.Card-Ind *** NEW *** Produces Card Index's

2.Programs/Procedures in Pascal

(Available on disc. Please provide sufficient postage to cover club costs!!)
Note the compiled Pascal programs will run without the use of Pascal, ie they are .COM programs. All you need is CP/M
1. DBASE for Disc Turbo Pascal
1(a). Comprehensive Create File Procedure
1(b). Simple Display File Procedure
1(c). Add More Data To File
2. Pretty Disney Characters.
2(a). Thumper.
2(b). Bambi.
2(c). Mickey.

3.Articles From Previous Magazines

(Available as listings, please provide sufficient postage to cover club costs. TA!)
1.PANEL2 Utility. An updated version of PANEL1, which includes a second feature.
2.Undocumented Neword dot commands.(Vol1 Iss.7)
3.Hisoft Pascal Review (vol1 Iss.8)
4.Neword Rom Review (Vol1 Iss.5)
5.RST10 Codes Explained (Vol1 Iss.3)
6.VDP Explained Using assembler (vol1 Iss4,5,6)
7.System Variables (Not Previously Published!!)

5.Reviews

a)Skittles :- By Paul Wood
This is a really nice menu driven program that allows you to set up a skittles league, full features include, saving and loading data, adding more data and printing out league tables.
b)Card-Ind :- By Peter Crighton
A well written program that will produce cassette labels for you.

Abridged Terms & Conditions (Downloads)

Disclaimer

www.primrosebank.net, (***the website***) is provided by Dave Stevenson as a service to the public, is provided "as is" and carries no warranties, expressed or implied, of any kind.

Dave Stevenson is not responsible for, and expressly disclaims all liability for, damages of any kind arising out of use, reference to, or reliance on any information contained within the website or made available for download. Whilst the information contained within the website site is periodically updated, no guarantee is given that the information provided on the website is correct, complete, and up-to-date.

A number of articles on the website contain technical data and practical guidance which may be of use in testing and maintaining various items of vintage computer and electronics hardware. Such articles are not intended to cover all aspects of the tasks involved and may omit essential information, including necessary safety precautions. Performance of the tasks described may risk damage to equipment and/or people. The reader is responsible for ensuring that he/she is capable of performing the tasks described and well as assessing the inherent risks involved and taking appropriate measures to mitigate such risks.

Dave Stevenson expressly disclaims all liability for, damages to equipment or injury of any kind arising out of use of such technical data and guidance.

As far as possible, information included on the website from other sources has been credited to the respective author and/or publisher. The majority of content on the website is derived from material first published in the 1980s. ***This material is likely still under copyright of the original author and/or publishers.*** The authors and/or publishers may not have given express permission to copy, transmit or make this information available for download, but I believe that they would have no objection to this archive information being placed into the public domain.

However, should the author and/or publisher of the original material find any content on the website for which they wish to assert their rights, they should notify Dave Stevenson (by e-mail to: webmaster@primrosebank.net) who would be pleased to enter into a dialogue to agree a satisfactory resolution of their concerns.

If you obtained this file as part of a paid-for package, you have been scammed! I suggest that you request a refund from the seller, please also advise Dave Stevenson at the e-mail address above.