

TO:

COLOUR GENIE

AUCKLAND DISTRICT USER GROUP

NEWSLETTER NO: 17

JUNE, 1985

Hi,

First off, the date of our next meeting is Monday, 24th June.

This is being put at the start of the newsletter, due to those people who miss seeing it further on.

Last month was a fairly unusual meeting - one of our committee members (who shall remain nameless - but he knows who he is!) forgot about the meeting, incredible as this seems. So, we didn't have a speaker. However, Olwen opened the meeting, and Andy demonstrated his Data Base II program. By the way, when I said there was a demonstration program that came with the Data Base II, I find when I loaded it and looked at the program, that it is a demonstration file. And, guess what, I can't load the data into my machine, so until something else is worked out, everyone that has bought the program won't get the demonstration data file part. Sorry about this, blame these stupid Genies that we have got.

Disk Drives

There is no more news yet about these, Mark is still waiting for some parts to arrive from overseas. However, if anyone is desperate, we (or you) can order the interfaces direct from England. They will cost around \$300.00 (107.00 pounds plus 7.50 airmail postage converts to something around \$300.00. If you are interested in doing something now get in touch with me, and I can give you details.

Data Stabilisers

These are now freely available, let me know if you want one.

New Members

We would like to welcome the following new members to our Group :-

Nick Bartulovich, 13 Hendry Avenue, Hillsborough.

Rodney Adamson, 24 Cheval Drive, Glenfield, Auckland 10.

G.R. Anderson, 300 Gascoigne Street, Hastings.

Wayne Lester, 12 Henry Street, Waikouaiti.

SOFTWARE

Saug

This program has now been sent out to members who had requested it. I was finally able to load the updated program from the Goldies after cleaning the heads on the recorder and kicking it once or twice.

Moonbuggy/Triton Battle

I have decided to sell these two programs on the same tape - one on each side, for the same price as one program - \$15.00. The reason for this is that Triton Battle is very much like A10 Bomber, other than the talking part at the start.

Newsletter Tapes

These are the tapes that were programs from our newsletters, and we were putting three or four programs on the one tape, and selling them for about \$3.00 each. I find that it is too time consuming to do them for the price. It is much easier and quicker to put two programs the same on the tape, than three or four.

Supaprint

\$20.00

This program, for members with printers, performs LGR and FGR dumps, using the Genie's graphics. Chris has been using it to produce listings in the last couple of newsletters. The program is easy to use, and screen dumps of graphic drawings can be produced simply.

TAPE RECORDERS

The one I mentioned in the last newsletter is actually a 'Euromatic'. A few people seem to have these, and all say they have no problems with them. I understand from David Reid's that they are imported and they expect shipments throughout the year.

computerised Address program

\$3.50

this program has been sent to us by Herman Philipsen. It is a name and address file program, with an option of printing labels. the listing was going to be put in the newsletter until i found it was six pages long.

see you all at the meeting next week.

- nola huggins

SCREEN SCROLLER

Recently I needed a routine to scroll the VDU screen vertically and horizontally with a complete wrap-around in both cases.

The screen output on the Colour Genie is located in memory at 4400Hex-47BFHex. Each graphics position is represented by one byte of this memory. Therefore the contents of a byte can be Peeked, examined and/or modified, and loked back to its original position or to new screen location. The colour of each position is controlled in the same way by memory at F000Hex-F3BFHex. Peeking and Poking the 1920 bytes needed to shift both the screen and its colour is extremely slow and tedious in a Basic Programme. The obvious answer is machine code.

The machine code routine is shown in both assembler version and in Basic version and is located in the part of memory reserved for Hi Resolution graphics. If you wish to use the routine and the Hi Resolution screen then the code will need to be relocated. To the top of memory for example.

The routine is accessed by a call to 4900Hex from a Basic or machine code programme. If the clear key is pressed the routine will return to the original programme. The screen is rotated by pressing the appropriate Arrow key.

To slow the scrolling enough to enable only one line to be rotated when an Arrow Key is pressed, a delay had to be included. This is done by a call to 60Hex, which is a Rom decrement count routine. To reduce the amount of delay then the delay amount loaded into the BC registers must be reduced. The fastest scroll speed needs a delay amount of 1 and the slowest FFFFHex. The delay amount is loaded in the following lines of the Assembler Version:

```
051  
081  
117  
153
```

The next stage is to modify the code to enable a predefined window of the screen to be scrolled. Perhaps some enterprising machine coder would like to try this.

Peter Fisher

```

001      XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
002      SCREEN SCROLL AND ROTATION ASSEMBLER VERSION
003      FOR COLOUR GENIE 32k WITH OLD ROMS
004      (C) PETER FISHER 1985
005      UP/DOWN/LEFT/RIGHT ARROWS MOVE THE SCREEN
006      CLEAR KEY TO RETURN TO BASIC/PROGRAMME
007      XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
008

```

```

009 4900      ORG
010

```

```

011      XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```

```

012 4900 3A 40 F8 SCAN      LD A,(F840)      ;ARROW KEYS ADDRESS
013 4903 FE 02          CF 2              ;CLEAR KEY PRESSED
014 4905 C8            RET Z            ;YES-GO BACK TO PROGRAMME
015 4906 FE 08          CF 8              ;UP KEY PRESSED
016 4908 CA 1F 49      JF Z,UP          ;YES-PROCESS UP
017 490B FE 10          CF 16             ;DOWN KEY PRESSED
018 490D CA 6C 49      JP Z,DOWN       ;YES-PROCESS DOWN
019 4910 FE 20          CF 32             ;LEFT KEY PRESSED
020 4912 CA B9 49      JF Z,LEFT       ;YES-PROCESS LEFT
021 4915 FE 40          CF 64             ;RIGHT KEY PRESSED
022 4917 CA F8 49      JF Z,RIGHT      ;YES-PROCESS RIGHT
023 491A C3 00 49      JF,SCAN        ;GO BACK TO SCAN ROUTINE
024 491D 00            NOP
025 491E 00            NOP

```

```

026      XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```

```

027 491F 21 00 44 UP      LD HL,4400      ;TOP LEFT SCREEN ADDRESS
028 4922 11 37 4A      LD DE,BUFFER1  ;40 BYTE BUFFER ADDRESS
029 4925 01 28 00      LD BC,28       ;LINE LENGTH TO SCROLL
030 4928 ED B0          LDIR          ;TOP LINE TO BUFFER
031 492A 21 00 F0      LD HL,F000     ;TOP LEFT COLOUR ADDRESS
032 492D 11 62 4A      LD DE,BUFFER2  ;40 BYTE BUFFER ADDRESS
033 4930 01 28 00      LD BC,28       ;LINE LENGTH TO SCROLL
034 4933 ED B0          LDIR          ;TOP LINE COLOUR TO BUFFER
035 4935 21 28 44      LD HL,4428     ;2ND LINE LEFT ADDRESS
036 4938 11 00 44      LD DE,4400     ;TOP LEFT SCREEN ADDRESS
037 493B 01 98 03      LD BC,0398    ;NUMBER OF BYTES TO MOVE
038 493E ED E0          LDIR          ;MOVE LINE N TO N-1
039 4940 21 28 F0      LD HL,F028     ;2ND LINE COLOUR ADDRESS
040 4943 11 00 F0      LD DE,F000     ;TOP LEFT COLOUR ADDRESS
041 4946 01 98 03      LD BC,0398    ;NUMBER OF BYTES TO MOVE
042 4949 ED B0          LDIR          ;MOVE COLOUR LINE N TO N-1
043 494B 21 37 4A      LD HL,BUFFER1  ;40 BYTE BUFFER ADDRESS
044 494E 11 98 47      LD DE,4798    ;BOTTOM LINE SCREEN ADDRESS
045 4951 01 28 00      LD BC,28       ;BYTES TO MOVE
046 4954 LD BC          LDIR          ;BUFFER TO BOTTOM LINE
047 4956 21 62 4A      LD HL,BUFFER2  ;BUFFER ADDRESS
048 4959 11 98 F3      LD DE,F398    ;BOTTOM LINE COLOUR ADDRESS
049 495C 01 28 00      LD BC,28       ;BYTES TO MOVE
050 495F ED E0          LDIR          ;BUFFER TO BOTTOM COLOUR LINE
051 4961 01 FF FF      LD BC,FFFF    ;DELAY AMOUNT
052 4964 CD 60 C0      CALL C060     ;ROT. DECREMENT COUNT
053 4967 C3 00 49      JF,SCAN        ;RETURN TO SCAN ROUTINE
054 496A 00            NOP
055 496B 00            NOP

```

```

056      XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```

```

057 496C 21 98 47 DOWN   LD HL,4798     ;BOTTOM LEFT SCREEN ADDRESS
058 496F 11 37 4A      LD DE,BUFFER1  ;40 BYTE BUFFER ADDRESS
059 4972 01 28 00      LD BC,28       ;BYTES TO MOVE
060 4975 ED B0          LDIR          ;BOTTOM LINE TO BUFFER

```

```

061 4977 21 98 F3      LD HL,F398      ;BOTTOM LEFT COLOUR ADDRESS
062 497A 11 62 4A      LD DE,BUFFER2  ;40 BYTE BUFFER ADDRESS
063 497D 01 28 00      LD BC,28       ;BYTES TO MOVE
064 4980 ED B0          LDIR           ;BOTTOM LINE COLOUR TO BUFFER
065 4982 21 97 47      LD HL,4797     ;BOTTOM LINE+1 RIGHT ADDRESS
066 4985 11 BF 47      LD DE,47BF     ;BOTTOM RIGHT SCREEN ADDRESS
067 4988 01 98 03      LD BC,0398    ;BYTES TO MOVE
068 498B ED B8          LDDR           ;MOVE LINE N TO N-1
069 498D 21 97 F3      LD HL,F397     ;BOTTOM LINE-1 RIGHT COLOUR
                                ;ADDRESS

070 4990 11 BF F3      LD DE,F3BF     ;BOTTOM RIGHT COLOUR ADDRESS
071 4993 01 98 03      LD BC,0398    ;BYTES TO MOVE
072 4996 ED B8          LDDR           ;COLOUR LINE N TO N+1
073 4998 21 37 4A      LD HL,BUFFER1  ;40 BYTE BUFFER ADDRESS
074 499B 11 00 44      LD DE,4400     ;TOP LEFT SCREEN ADDRESS
075 499E 01 28 00      LD BC,28       ;BYTES TO MOVE
076 49A1 ED B0          LDIR           ;BUFFER TO TOP LINE
077 49A3 21 62 4A      LD HL,BUFFER2  ;40 BYTE BUFFER ADDRESS
078 49A6 11 00 F0      LD DE,F000     ;TOP LEFT COLOUR ADDRESS
079 49A9 01 28 00      LD BC,28       ;BYTES TO MOVE
080 49AC ED B0          LDIR           ;BUFFER TO TOP COLOUR LINE
081 49AE 01 FF FF      LD BC,FFFF     ;DELAY AMOUNT
082 49B1 CD 60 00      CALL 0060      ;ROM DECREMENT COUNT
083 49B4 C3 00 49      JP,SCAN        ;RETURN TO SCAN ROUTINE
084 49B7 00            NOP
085 49B8 00            NOP
086                    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
087 49E9 01 18 00      LEFT LD BC,18     ;NUMBER OF LINES TO MOVE
088 49BC 11 00 44      LD DE,4400     ;TOP LEFT SCREEN ADDRESS
089 49BF 21 01 44      LD HL,4401     ;TOP LEFT SCREEN ADDRESS+1
090 49C2 C5            LOOP1 PUSH BC       ;SAVE LINE COUNT
091 49C3 1A            LD A,(DE)     ;SAVE LEFT BYTE OF LINE
092 49C4 01 27 00      LD BC,27       ;BYTES TO MOVE
093 49C7 ED B0          LDIR           ;BYTE N TO N-1
094 49C9 12            LD (DE),A     ;SAVED BYTE TO RIGHT OF LINE
095 49CA 13            INC DE        ;NEXT LINE LEFT ADDRESS
096 49CB 23            INC HL        ;LINE LEFT ADDRESS+1
097 49CC C1            POP BC        ;GET LINE COUNT
098 49CD 0B            DEC BC        ;DECREMENT LINE COUNT
099 49CE 78            LD A,B        ;LINE COUNT=0
100 49CF B1            OR C
101 49D0 C2 C2 49      JP NZ,LOOP1    ;NO-MOVE NEXT LINE
102 49D3 01 18 00      LD BC,18       ;LINES TO MOVE
103 49D6 11 00 F0      LD DE,F000     ;TOP LEFT COLOUR ADDRESS
104 49D9 21 01 F0      LD HL,F001     ;TOP LEFT COLOUR ADDRESS+1
105 49DC C5            LOOP2 PUSH BC       ;SAVE LINE COUNT
106 49DD 1A            LD A,(DE)     ;SAVE LEFT BYTE OF LINE
107 49DE 01 27 00      LD BC,27       ;BYTES TO MOVE
108 49E1 ED B0          LDIR           ;MOVE BYTE N TO N-1
109 49E3 12            LD (DE),A     ;SAVED BYTE TO RIGHT OF LINE
110 49E4 13            INC DE        ;NEXT LINE LEFT ADDRESS
111 49E5 23            INC HL        ;NEXT LINE LEFT ADDRESS+1
112 49E6 C1            POP BC        ;GET LINE COUNT
113 49E7 0B            DEC BC        ;DECREMENT LINE COUNT
114 49E8 78            LD A,B        ;LINE COUNT=0
115 49E9 B1            OR C
116 49EA C2 DC 49      JP NZ,LOOP2    ;NO-MOVE NEXT LINE
117 49ED C1 FF FF      LD BC,FFFF     ;DELAY AMOUNT
118 49F0 CD 60 00      CALL 0060      ;ROM DECREMENT COUNT

```



```

10 'SCREEN SCROLL AND ROTATION BASIC VERSION
20 'FOR COLOUR GENIE 32K WITH OLD ROMS
30 '(C) PETER FISHER 1985
40 'UP/DOWN/LEFT/RIGHT ARROWS MOVE SCREEN
50 'CLEAR KEY TO RETURN TO BASIC/PROGRAMME
60 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
70 FOR A=&H4900 TO &H4A36
80 READ B:POKE A,B
90 NEXT A
100 DATA 58,64,248,254,2,200,254,8
110 DATA 202,31,73,254,16,202,108,73
120 DATA 254,32,202,185,73,254,64,202
130 DATA 248,73,195,0,73,0,0,33
140 DATA 0,68,11,55,74,1,40,0
150 DATA 237,176,33,0,240,11,98,74
160 DATA 1,40,0,237,176,33,40,68
170 DATA 11,0,68,1,152,3,237,176
180 DATA 33,40,240,11,0,240,1,152
190 DATA 3,237,176,33,55,74,11,152
200 DATA 71,1,40,0,237,176,33,98
210 DATA 74,11,152,243,1,40,0,237
220 DATA 176,1,255,255,205,96,0,195
230 DATA 0,73,0,0,33,152,71,11
240 DATA 55,74,1,40,0,237,176,33
250 DATA 152,243,11,98,74,1,40,0
260 DATA 237,176,33,151,71,11,191,71
270 DATA 1,152,3,237,176,33,151,243
280 DATA 11,191,243,1,152,3,237,184
290 DATA 33,55,74,11,0,68,1,40
300 DATA 0,237,176,33,98,74,11,0
310 DATA 240,1,40,0,237,176,1,255
320 DATA 255,205,96,0,195,0,73,0
330 DATA 0,1,24,0,11,0,68,33
340 DATA 1,68,197,26,1,39,0,237
350 DATA 176,18,19,35,193,11,120,177
360 DATA 194,194,73,1,24,0,17,0
370 DATA 240,33,1,240,197,26,1,39
380 DATA 0,237,184,18,19,35,193,11
390 DATA 120,177,194,220,73,1,255,255
400 DATA 205,96,0,195,0,73,0,0
410 DATA 1,24,0,17,191,71,33,190
420 DATA 71,197,26,1,39,0,237,176
430 DATA 18,27,43,193,11,120,177,194
440 DATA 1,74,1,24,0,17,191,243
450 DATA 33,190,243,197,26,1,39,0
460 DATA 237,184,18,27,43,193,11,120
470 DATA 177,194,27,74,1,255,255,205
480 DATA 96,0,195,0,73,0,0

```

```
5 CLS:COLOUR5
10 PRINT@454,"MOONLANDING"
15 PRINT@604,"DO YOU WANT INSTRUCTIONS (Y/N)"
20 A$=INKEY$: IF A$="N" GOTO 30 ELSE IF A$="Y" GOTO 2000:ELSE 20
2000 CLS:PRINT@44,"Your rate of descent varies with"
2010 PRINT@84,"the engine thrust."
2020 PRINT@124,"Use the numerical keys 0 to : to"
2030 PRINT@164,"apply increasing leves of thrust"
2040 PRINT@204,"from zero to ten."
2050 PRINT@244,"Keep watching your rate of descent"
2060 PRINT@284,"and your hight above the lunar"
2070 PRINT@324,"surface"
2080 PRINT@404,"Running out of fuel is a big"
2090 PRINT@454,"NO NO !"
2100 PRINT@527,"PRESS RETURN TO CONTINUE"
2110 G=PEEK (-1984)
2120 IF G AND 01 THEN GOTO 30 ELSE GOTO 2110
```

MOONLANDING instruction page
by BERT HARRIS

SWAP

GENIE INVADERS
& LINK(old roms only)
for anything except HEKTIC

PHONE: 486-504 weekdays after 4.30 p.m.

The Genie and the #-1 Case

As I said last month, I have been looking into the problem of data saving with the Colour Genie using the PRINT#-1 and INPUT#-1 statements. If I had foreseen the difficulties that would arise with the #!%&' data, I would not have used those statements in my program. I have another routine in the program which saves the bulk of the data and it would have been easier in retrospect to have used it to save all of the data. However, it is an ill wind.....

What were the facts I had to work on? After the evidence had been sifted from the chaff, I was left with the following. Some folks cannot read back data tapes. The ROM uses the same routines for SAVING and LOADING SYSTEM, BASIC and data. The LOADING volume level is much more sensitive with data than with BASIC or SYSTEM tapes. Long strings of variables in PRINT#-1 and INPUT#-1 are more difficult than single variables.

How does the Genie save bytes onto tape? Each byte presented to the WRITE-BYTE routine is broken up into 8 bits of binary information. Each of these bits has a preceding clock pulse recorded onto the tape to let the machine know that a recorded bit of data is coming. The data bit follows about 600 microseconds later. The routine then collects the next bit and records a clock pulse and the data bit. This goes on until all 8 bits have been recorded. They look something like this diagrammatically:

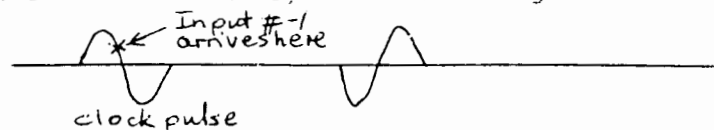


By the time that the usual cassette electronics have dealt with the signal, it looks like this:



When the data is being LOADED back into the cassette input port, The Genie waits until a clock pulse comes along. It then expects the data bit to follow and gets it. This is stored while the machine waits for the next clock pulse. This goes on until it has all 8 bits which then gets stored as a loaded byte of data by the BASIC, SYSTEM or INPUT#-1 routine. This process is repeated until a specified sequence of bytes tell it that the load has been completed.

What goes wrong with data SAVES and LOADS? My theory is that the processing time between bytes on an INPUT#-1 is greater than the space between bytes on the tape and the first clock pulse is arriving at the tape head just a bit (pun there!) too early as in the diagram.



Now these clock and data pulses get amplified inside the Genie so that the machine can distinguish between the pulses and other tape noise. Most Genies can handle this but a few have slightly weaker amplifying stage (a normal thing with IC's) and they miss the first clock pulse. This upsets

normal thing with IC's) and they miss the first clock pulse. This upsets the whole timing sequence and the result is a garbaged load. I believe that the PRINT#-1 process works properly and data is SAVED to tape correctly. It is just that some machines cannot read their own tapes whereas another machine could! Don't destroy data tapes if you can't read them. Someone else could re-record them so that you can read them.

How can we cure this timing problem? It would be too difficult to speed up the INFUT#-1 procedure as this would require a rewrite of the ROM. However, the designers of grand-pappy TRS80 Model 1 were very farsighted and built CALL's to the RAM into the ROM. These enable the ROM procedures to be intercepted and an additional routine inserted. Thus we can insert a delay between each byte being sent from the PRINT#-1 to the tape output port. This is what the following patch does.

```
10 FOR X=0 TO 8:READ B:POKE&H4181+X,B:NEXT
20 POKE&H41C3,65:POKE&H41C2,129:POKE &H41C1,195
30 DATA 58,156,64,183,240,71,16,254,201
```

Naturally, WP2 has had this patch incorporated but owners need not bother Nola by sending their tapes back for an update. Make the following alterations to lines 388 and 389 of WP2.

```
388 FORL=0TO55: .....(add at end of line):POKE&H41C3,65:POKE&H41C2,129:
POKE&H41C1,195
```

Add this data at the end of line 389.

```
389 .....58,156,64,183,240,71,16,254,201
```

The patch can be added to any program and will remain in memory from then on even when a new program is loaded. It disappears on switching off. This patch illustrates the advantage of a hybrid word processing program as it would be very difficult to add it to a machine code program.

WP2 owners making this alteration could make another cosmetic change to line 366 at the same time. Change "..THEN2ELSECLS" to "..THEN1ELSECLS". This will prevent a premature end to some functions under a certain unusual combination of commands.

Does the patch work? This article has been TAFED and LOADED several times with it in place but I normally don't have any problem in this respect. I will only know the results IF YOU WRITE OR PHONE ME. I do like to hear good news sometimes. In fact I'd like to hear anything from anybody about my programs or articles.

- Allan Clarke

WF2 - Word Processor Notes

This month I shall discuss underlining with WF2 and then have a few words about taping data files with the Genie. (But also see the separate article about PRINT#-1/INFUT#-1.)

One of the least understood functions in WF2 has been the underlining function. Instead of using markers in a line, I adopted the WYSIWYG (what you see is what you get) principle when designing the word processor. Lines typed like this:

```
19          HEADING
```

```
20
```

will be printed like this:

```
          HEADING
```

The line containing the underlining characters MUST end with an underline character. Otherwise the line will be printed as a line of CHR\$(95)'s.

Centred lines or headings are best underlined by tabbing the cursor to the position just under the characters to be underlined rather than trying to centre an underline string.

The method of underlining used in the program is back-space underlining as this allows the greatest flexibility with different printers. It is also the smoothest method with the Super 5/CP 80 family. Some printers are not happy with this method and the following program alteration may suit them better as it does my NEC 8023. The alternative requires one line to be replaced and one additional line. However, you would probably have to 'tune' line 396 to suit your particular printer. This line contains printer dependent variable values. To change to the alternative, type in the following two lines:

```
338 FORI=1TOLEN(A$(P)):IFI<=LEN(A$(P+1))IFASC(MID$(A$(P+1),I))=95  
LPRINTU$;MID$(A$(P),I,1);V$;ELSELPRINTMID$(A$(P),I,1);ELSE  
LPRINTMID$(A$(P),I,1);
```

```
340 NEXTI:P=P+1
```

This gives a true underlining but was not as smooth on the Super 5. U\$ and V\$ are the underline-on and off character strings in line 396. Anyone who has difficulty in setting up their printer should contact me, preferably with a copy of their printer codes.

It was not very difficult to deal with the faulty data tapes created with WF2 as requested in last month's bulletin. There were none! Does this mean that nobody has any problem any more? I doubt it. I think that the latent bug existing in the Genie has made us much more aware of tape care than otherwise would have been the case. One more suggestion for those with difficult data loading and saving is not to use automatic volume controls on tape recorders. This produces unsaturated tapes.

- Alla Clarke

PRINTING DEMONSTRATION

USING A SUPER 5 PRINTER

```
10 FOR J=0 TO 10:READ K:POKE 16422+J,K:NEXT
20 DATA 45,64,67,0,0,80,82,121,195,235,4
30 CLS:PRINT@449,"PRINTING DEMONSTRATION"
40 FOR X=1 TO 1000:NEXT
50 CLS:PRINT@58,"MENU"
60 PRINT@130,"FICA - TYPE 'P'
ALICS TYPE 'I'"
70 PRINT@250,"DOUBLE WIDTH - TYPE'D'
MPRESSED - TYPE'C'"
80 INPUT A$
90 IF A$="P" THEN GOSUB 500 ELSE IF A$="E" THEN GOSUB 510 ELSE IF A$="I" THEN GOSUB 520 ELSE 100
95 GOTO 110
100 IF A$="D" THEN GOSUB 530 ELSE IF A$="U" THEN GOSUB 540 ELSE IF A$="C" THEN GOSUB 550 ELSE 80
110 INPUT "TYPE A LINE";B$
120 LPRINT B$
130 LPRINT CHR$(27);CHR$(64):GOTO 50
500 LPRINT CHR$(27);CHR$(80);CHR$(01):RETURN
510 LPRINT CHR$(27);CHR$(80);CHR$(0):RETURN
520 LPRINT CHR$(27);CHR$(52):RETURN
530 LPRINT CHR$(27);CHR$(87);CHR$(01):RETURN
540 LPRINT CHR$(27);CHR$(45);CHR$(01):RETURN
550 LPRINT CHR$(15):RETURN
```

} Allan Clarke's patch to enable Super 8
etc printers to accept CHR\$(0).

ELITE - TYPE 'E'	IT
UNDERLINE - TYPE 'U'	CO

A. R. McGill.
Super 5 EN-P1090

```

1 CLS
10 A=RND(6)
100 IF A=6 THEN FOR X=X TO X+5 STEP2:PRINT@X,CHR$(202):NEXT X:FOR
  X=X+160 TO X+164 STEP2:PRINT@X,CHR$(202):NEXT X
110 IF A=5 THEN PRINT@1,CHR$(202):PRINT@5,CHR$(202):PRINT@161,CHR
  $(202):PRINT@165,CHR$(202):PRINT@83,CHR$(202)
120 IF A=4 THEN PRINT@1,CHR$(202):PRINT@5,CHR$(202):PRINT@161,CHR
  $(202):PRINT@165,CHR$(202)
130 IF A=3 THEN PRINT@1,CHR$(202):PRINT@83,CHR$(202):PRINT@165,CH
  R$(202)
140 IF A=2 THEN PRINT@1,CHR$(202):PRINT@165,CHR$(202)
150 IF A=1 THEN PRINT@83,CHR$(202)
180 FOR X=1 TO 1000:NEXT:CLS
200 GOTO 10

```

THIS PROGRAM GIVES LARGE DICE
 GRAPHICS AND CAN BE MODIFIED FOR USE IN LARGER
 PROGRAMS

PRINTER'S NOTE

Page 1

MODIFICATION TO WORDMATE

(PICA)

The following extra lines I have added to Allan Clarke's Wordmate
 enable me to use Pica, Elite and Italic fonts. It may be of use to
 others.

(ELITE)

My printer is a Super 5/EN-P1090.

(ELITE
ITALICS)

Keith McGill.

```

251 GOSUB 400
400 INPUT"FONT - P/E/I";XP$
402 LPRINT CHR$(27);CHR$(64)
404 IF ASC(XP$)=80 THEN 406 ELSE IF ASC(XP$)=69 THEN 408 ELSE IF ASC(XP$)=73 THEN 410 ELSE 400
406 LPRINT CHR$(27);CHR$(80);CHR$(01):RETURN
408 LPRINT CHR$(27);CHR$(80);CHR$(0):RETURN
410 LPRINT CHR$(27);CHR$(52):RETURN

```

```

10 INPUT "No. of grid lines"; N
20 DIM CX(5), CY(5), LX(5, N+1), LY(5, N+1)
30 CLS
40 FCLS: FGR: FCOLOUR4
50 FOR I=1 TO 5
60 READ CX(I), CY(I)
70 NEXT I
80 DATA 9, 9, 9, 89, 149, 89, 149, 9, 9, 9
90 FOR I=1 TO 4
100 PLOT CX(I), CY(I) TO CX(I+1), CY(I+1)
110 NEXT I
120 FOR I=1 TO N
130 LX(1, I) = CX(1) + (CX(2) - CX(1)) * I / N
140 LY(1, I) = CY(1) + (CY(2) - CY(1)) * I / N
150 NEXT I
160 FOR I=1 TO N
170 LX(3, I) = CX(4) + (CX(3) - CX(4)) * I / N
180 LY(3, I) = CY(4) + (CY(3) - CY(4)) * I / N
190 NEXT I
200 FOR I=1 TO N
210 LX(2, I) = CX(2) + (CX(3) - CX(2)) * I / N
220 LY(2, I) = CY(2) + (CY(3) - CY(2)) * I / N
230 NEXT I
240 FOR I=1 TO N
250 LX(4, I) = CX(1) + (CX(4) - CX(1)) * I / N
260 LY(4, I) = CY(1) + (CY(4) - CY(1)) * I / N
270 NEXT I
280 ROW=1: GOSUB 310
290 ROW=2: GOSUB 310
300 K$=INKEY$: IF K$="" THEN 300 ELSE FCLS:CLS:LGR:RUN
310 REM
320 FOR I=1 TO N
330 PLOT LX(ROW, I), LY(ROW, I) TO LX(ROW+2, I), LY(ROW+2, I)
340 NEXT I
350 RETURN
360 REM: ADAPTED FOR CG by K. McGill FROM USBORNE BOOKS 'BETTER BASIC' .

```

THIS PROGRAMME
 produces a grid on
 screen of the inputted
 number of grid lines.
 Press any key to re-start.

```

10 CLS:PRINT@51,"H O R S E  R A C E":PRINT:PRINT
20 CHAR4
30 PRINT"THE ONLY CONTROLS ARE THE SPACE BAR AND THE 'M' KEY. PRESS EITHER OF TH
ESE TO  START THE RACE"
40 PRINT"PRESS ANY KEY TO CONTINUE"
50 K$=INKEY$:IF K$="" THEN 50
60 DIMA(15):GOTO 370
70 CLS:FOR A=1 TO 7 :READ A(A):NEXT A
80 ST=-3072:FOR AD=ST TO ST+7:READ D:POKE AD,D:NEXT AD
90 ST=-3064:FOR AD=ST TO ST+7:READ D:POKE AD,D:NEXT AD
100 DATA 85,165,245,325,405,485,565
110 DATA 0,2,5,248,120,132,66,36
120 DATA 36,18,33,30,31,160,64,0
130 PRINT@13,"* C.G.STAKES *"
140 CHAR 2
150 PRINT@720, "WHICH HORSE DO YOU WANT TO BET ON";:INPUT H
160 CLS
165 PRINT@13,"* C.G.STAKES *"
170 FOR X=48 TO 688 STEP 40:PRINT@X,"a";:NEXT
180 FOR C=1 TO 7:GOSUB 470:NEXT C
190 F=80:FOR C=1 TO 7:PRINT@F,C:F=F+80:NEXT C
200 PRINT@845,"GET SET...!";
210 FOR X=1 TO 1000:NEXTX:FOR X= 48 TO 688 STEP 40 :PRINT@X," ":NEXT:PRINT@845,"
AND THEY'RE OFF!";
220 FOR C=1 TO 7
230 V=C:GOSUB 450:IF E$="M" THEN 240 ELSE 270
240 ZS=RND(35):IF C=ZS THEN GOSUB 530:E$="":GOTO 290
250 REM
260 REM
270 W=RND(2):IF W=2 AND C<>ZS THEN A(C)=A(C)+1
280 IF A(C)<80 THEN A(C)=0:PRINT@A(C)," ";:PRINT@A(C)+2," ";:PRINT@A(C)+1," "
;
290 NEXT C
300 FOR V=1 TO 7:GOSUB 450: A(C) =A(V):GOSUB 470
310 NEXT V
320 GOTO 220

330 FOR R=34 TO 834 STEP 40:PRINT@R,"*";:NEXT R
340 PRINT@845,"THE RACE IS OVER!";:FOR X=1 TO 1000:NEXT
350 FOR X=1 TO 1000:NEXT
360 CLS:FOR P=1 TO 250:NEXT P
370 CLS:PRINT"WINNER:":PRINT"HORSE NO. ";M
380 IF H=V THEN PRINT@440,"YOU WON $";H*5:ELSE PRINT"TOO BAD - YOUR HORSE LOST"
390 RESTORE
400 PRINT"PRESS SPACER OR 'M' KEY TO RESTART"
410 E$=INKEY$
420 IF E$=" " THEN CLEAR:DIMA(15):GOTO 70
430 IF E$="M" THEN CLEAR:E$="M":DIMA(15):GOTO 70
440 FOR P=1 TO 250:NEXT P:GOTO 410
450 M=V
460 RETURN
470 CHAR2:PRINT@A(C)," ";:PRINT@A(C),CHR$(128)
480 PRINT@A(C)," ";
490 PRINT@A(C)+1,CHR$(128)
500 IF A(C)=592OR A(C)=112 OR A(C)=192 OR A(C)=272 OR A(C)=352 OR A(C)=432 OR A(
C)=512 THEN GOSUB 450:GOTO 330
510 RETURN
520 GOTO 70
530 CHAR1:PRINT@A(C),CHR$(129);" HORSE FELL";
540 A(C)=0
550 RETURN
560 END

```

CONVERTED TO C.G. BY KEITH MCGILL
From a programme in '55 Advanced

ELIZA

```

10 CLS: CLEAR 5000
20 CLEAR 5000
30 PRINT$55,"$ ELIZA $"
40 PRINT$124,"$ YOUR PERSONAL PSYCHOTHERAPIST $"
50 PRINT:PRINT:PRINT"TYPE IN YOUR FEELINGS, AND ELIZA WILL THINK FOR A MOMENT, THEN ANSWER YOU"
60 PRINT:PRINT
70 DIM S(36),R(36),N(36)
80 M1=36:M2=14:M3=112
90 FOR I=1 TO M1+M2+M3:READ Z$:NEXT I
100 FOR I=1 TO M1
110 READ S(I),L:R(I)=S(I):N(I)=S(I)+L-1
120 NEXT I
130 PRINT"HELLO. I'M ELIZA. HOW CAN I HELP YOU ?"
140 INPUT I$
150 I$=" "+I$+" "
160 FOR L=1 TO LEN(I$)
170 IF MID$(I$,L,1)="" THEN I$=LEFT$(I$,L-1)+RIGHT$(I$,LEN(I$)-L):GOTO 170
180 IF L+4<LEN(I$) THEN IF MID$(I$,L,4)="SHUT" THEN PRINT"DON'T YOU TELL ME TO SHUT UP, YOU LITTLE TWIT!":END
190 NEXT L
200 IF I$=P$ THEN PRINT"PLEASE DON'T REPEAT YOURSELF!":GOTO 140
210 RESTORE
220 S=0
230 FOR K=1 TO M1
240 READ K$
250 IF S>0 THEN 290
260 FOR L=1 TO LEN(I$)-LEN(K$)+1
270 IF MID$(I$,L,LEN(K$))=K$ THEN S=K:T=L:F$=K$
280 NEXT L
290 NEXT K
300 IF S>0 THEN K=S:L=T:GOTO 320
310 K=36:GOTO 530
320 REM
330 RESTORE:FOR I=1 TO M1:READ Z$:NEXT I
340 C$=" "+RIGHT$(I$,LEN(I$)-LEN(F$)-L+1)+" "
350 FOR X=1 TO M2/2
360 READ S$,R$
370 FOR L=1 TO LEN(C$)
380 IF L+LEN(S$)>LEN(C$) THEN 430
390 IF MID$(C$,L,LEN(S$))<>S$ THEN 430
400 C$=LEFT$(C$,L-1)+R$+RIGHT$(C$,LEN(C$)-L-LEN(S$)+1)
410 L=L+LEN(R$)
420 GOTO 470
430 IF L+LEN(R$)>LEN(C$) THEN 470
440 IF MID$(C$,L,LEN(R$))<>R$ THEN 470
450 C$=LEFT$(C$,L-1)+S$+RIGHT$(C$,LEN(C$)-L-LEN(R$)+1)
460 L=L+LEN(S$)
470 NEXT L
480 NEXT X

```



```

490 IF MID$(C$,2,1)=" " THEN C$=RIGHT$(C$,LEN(C$)-1)
500 FOR L=1 TO LEN(C$)
510 IF MID$(C$,L,1)="!" THEN C$=LEFT$(C$,L-1)+RIGHT$(C$,LEN(C$)-L):GOTO 510
520 NEXT L
530 REM
540 RESTORE:FOR X=1 TO N1+N2:READ Z$:NEXT X
550 FOR I=1 TO R(K):READ F$:NEXT I
560 R(K)=R(K)+1:IF R(K)>N(K) THEN R(K)=S(K)
570 IF RIGHT$(F$,1)<>"*" THEN PRINTF$:P$=I$:GOTO 140
580 PRINTLEFT$(F$,LEN(F$)-1);C$
590 P$=I$:GOTO 140
600 DATA "CAN YOU", "CAN I", "YOU ARE", "YOU'RE", "I DONT", "I FEEL"
610 DATA "WHY DONT YOU", "WHY CANT I", "ARE YOU", "I CANT", "I AM", "IM "
620 DATA "YOU ", "I WANT", "WHAT", "HOW", "WHO", "WHERE", "WHEN", "WHY"
630 DATA "NAME", "CAUSE", "SORRY", "DREAM", "HELLO", "HI ", "MAYBE"
640 DATA " NO", "YOUR", "ALWAYS", "THINK", "ALIKE", "YES", "FRIEND"
650 DATA "COMPUTER", "NOKEYFOUND"
660 DATA " ARE ", " AM ", " WERE ", " WAS ", " YOU ", " I ", " YOUR ", " MY "
670 DATA " IVE ", " YOUVE ", " IM ", " YOURE "
680 DATA " ME ", " !YOU "
690 DATA "DON'T YOU BELIEVE THAT I CAN?"
700 DATA "PERHAPS YOU WOULD LIKE TO BE ABLE TO?"
710 DATA "YOU WANT ME TO BE ABLE TO?"
720 DATA "PERHAPS YOU DON'T WANT TO?"
730 DATA "DO YOU WANT TO BE ABLE TO?"
740 DATA "WHAT MAKES YOU THINK I AM?"
750 DATA "DO YOU WANT TO BELIEVE I AM?"
760 DATA "PERHAPS YOU WOULD LIKE TO BE?"
770 DATA "DO YOU SOMETIMES WISH YOU WERE?"
780 DATA "DON'T YOU REALLY?"
790 DATA "WHY DON'T YOU?"
800 DATA "DO YOU WISH YOU COULD?"
810 DATA "DOES THAT TROUBLE YOU?"
820 DATA "TELL ME MORE ABOUT THESE FEELINGS"
830 DATA "DO YOU OFTEN FEEL?"
840 DATA "DO YOU ENJOY FEELINGS?"
850 DATA "DO YOU REALLY BELIEVE I DON'T?"
860 DATA "PERHAPS IN GOOD TIME I WILL?"
870 DATA "DO YOU WANT ME TO?"
880 DATA "DO YOU THINK YOU SHOULD BE ABLE TO?"
890 DATA "WHY CAN'T YOU?"
900 DATA "WHY ARE YOU INTERESTED IN WHETHER OR NOT I AM?"
910 DATA "WOULD YOU PREFER IT IF I WERE NOT?"
920 DATA "PERHAPS IN YOUR FANTASIES I AM?"
930 DATA "HOW DO YOU KNOW YOU CAN'T?"
940 DATA "HAVE YOU TRIED?"
950 DATA "PERHAPS YOU CAN NOW?"
960 DATA "DID YOU COME TO ME BECAUSE YOU ARE?"
970 DATA "HOW LONG HAVE YOU BEEN?"

```

380 DATA "DO YOU BELIEVE IT IS NORMAL TO BE?"
990 DATA "DO YOU ENJOY BEING?"
1000 DATA "WE WERE DISCUSSING YOU - NOT ME"
1010 DATA "OH, I...?"
1020 DATA "YOU'RE NOT REALLY TALKING ABOUT ME, ARE YOU?"
1030 DATA "WHAT WOULD IT MEAN TO YOU IF YOU GOT?"
1040 DATA "WHY DO YOU WANT?"
1050 DATA "SUPPOSE YOU SOON GOT?"
1060 DATA "WHAT IF YOU NEVER GOT?"
1070 DATA "I SOMETIMES ALSO WANT?"
1080 DATA "WHY DO YOU ASK?"
1090 DATA "DOES THAT QUESTION INTEREST YOU?"
1100 DATA "WHAT ANSWER WOULD PLEASE YOU MOST?"
1110 DATA "WHAT DO YOU THINK?"
1120 DATA "ARE SUCH QUESTIONS ON YOUR MIND OFTEN?"
1130 DATA "WHAT IS IT THAT YOU REALLY WANT TO KNOW?"
1140 DATA "HAVE YOU ASKED ANYONE ELSE?"
1150 DATA "HAVE YOU ASKED SUCH QUESTIONS BEFORE?"
1160 DATA "WHAT ELSE COMES TO MIND WHEN YOU ASK THAT?"
1170 DATA "NAMES DON'T INTEREST ME"
1180 DATA "I DON'T CARE ABOUT NAMES-PLEASE GO ON"
1190 DATA "IS THAT THE REAL REASON?"
1200 DATA "DON'T ANY OTHER REASONS COME TO MIND?"
1210 DATA "DOES THAT EXPLAIN ANYTHING ELSE?"
1220 DATA "WHAT OTHER REASONS MIGHT THERE BE?"
1230 DATA "PLEASE DON'T APOLOGIZE!"
1240 DATA "APOLOGIES AREN'T NECESSARY"
1250 DATA "WHAT FEELINGS DO YOU HAVE WHEN YOU SAY THAT?"
1260 DATA "DON'T BE SO DEFENSIVE!"
1270 DATA "WHAT DOES THAT DREAM SUGGEST TO YOU?"
1280 DATA "DO YOU DREAM OFTEN?"
1290 DATA "WHAT PEOPLE APPEAR IN YOUR DREAMS?"
1300 DATA "ARE YOU DISTURBED BY YOUR DREAMS?"
1310 DATA "HELLO-TELL ME WHAT BROUGHT YOU TO ME"
1320 DATA "YOU DON'T SEEM QUITE CERTAIN"
1330 DATA "WHY THE UNCERTAIN TONE?"
1340 DATA "CAN'T YOU BE MORE POSITIVE?"
1350 DATA "YOU AREN'T SURE?"
1360 DATA "DON'T YOU KNOW?"
1370 DATA "ARE YOU SAYING 'NO' JUST TO BE NEGATIVE?"
1380 DATA "YOU'RE BEING A BIT NEGATIVE"
1390 DATA "WHY NOT?"
1400 DATA "ARE YOU SURE?"
1410 DATA "WHY NO?"
1420 DATA "WHY ARE YOU CONCERNED ABOUT MY?"
1430 DATA "WHAT ABOUT YOUR OWN?"
1440 DATA "CAN YOU THINK OF A SPECIFIC EXAMPLE?"
1450 DATA "WHEN?"
1460 DATA "WHAT ARE YOU THINKING OF?"
1470 DATA "REALLY, ALWAYS?"
1480 DATA "DO YOU REALLY THINK SO?"
1490 DATA "BUT YOU ARE NOT SURE YOU?"
1500 DATA "DO YOU DOUBT YOUR?"
1510 DATA "IN WHAT WAY?"
1520 DATA "WHAT RESEMBLANCE DO YOU SEE?"
1530 DATA "WHAT DOES THE SIMILARITY SUGGEST TO YOU?"
1540 DATA "WHAT OTHER CONNECTIONS DO YOU SEE?"
1550 DATA "COULD THERE REALLY BE SOME CONNECTION?"
1560 DATA "HOW?"
1570 DATA "YOU SEEM QUITE POSITIVE"
1580 DATA "ARE YOU SURE?"
1590 DATA "I SEE"
1600 DATA "I UNDERSTAND"
1610 DATA "WHY DO YOU BRING UP THE TOPIC OF FRIENDS?"
1620 DATA "DO YOUR FRIENDS WORRY YOU?"
1630 DATA "DO YOUR FRIENDS PICK ON YOU?"
1640 DATA "ARE YOU SURE YOU HAVE ANY REAL FRIENDS?"
1650 DATA "DO YOU IMPOSE ON YOUR FRIENDS?"
1660 DATA "PERHAPS YOUR LOVE FOR FRIENDS WORRIES YOU"
1670 DATA "DO COMPUTERS WORRY YOU?"
1680 DATA "ARE YOU TALKING ABOUT ME IN PARTICULAR?"
1690 DATA "ARE YOU FRIGHTENED BY MACHINES?"
1700 DATA "WHY DO YOU MENTION COMPUTERS?"
1710 DATA "WHAT DO YOU THINK MACHINES HAVE TO DO WITH YOUR PROBLEM?"
1720 DATA "DON'T YOU THINK COMPUTERS CAN HELP PEOPLE?"
1730 DATA "WHAT IS IT ABOUT MACHINES THAT WORRIES YOU?"
1740 DATA "DO YOU HAVE ANY PSYCHOLOGICAL PROBLEMS?"
1750 DATA "WHAT DOES THAT SUGGEST TO YOU?"
1760 DATA "I SEE"
1770 DATA "I'M NOT SURE I UNDERSTAND YOU FULLY"
1780 DATA "CAN YOU CLARIFY THAT A LITTLE?"
1790 DATA "CAN YOU ELABORATE ON THAT?"
1800 DATA "THAT'S QUITE INTERESTING"
1810 DATA 1,3,4,2,6,4,6,4,10,4,14,3,17,3,20,2,22,3,25,3
1820 DATA 28,4,28,4,32,3,35,5,40,9,40,9,40,9,40,9,40,9
1830 DATA 49,2,51,4,55,4,59,4,63,1,63,1,64,5,69,5,74,2,76,4
1840 DATA 80,3,83,7,90,3,93,6,99,7,106,6
1850 REM: ADAPTED FOR COLOUR GENIE BY A.K.McGILL
1860 REM: FROM A PROGRAMME IN DAVID AHL'S "MORE BASIC COMPUTER GAMES"