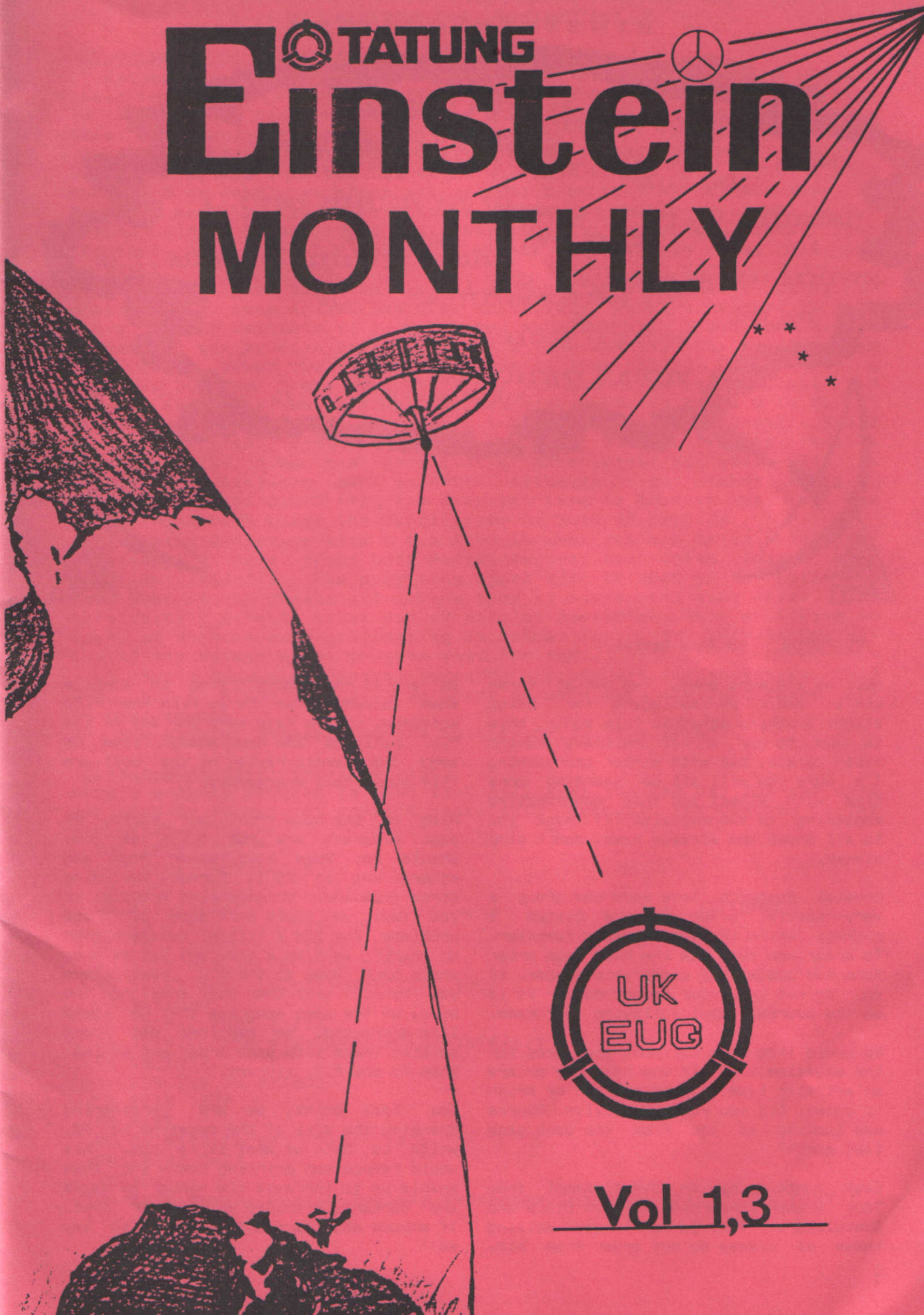
TATUNG Einstein MONTHLY



UK
EUG

Vol 1,3

— EINSTEIN MONTHLY —

EDITORIAL (Graham Bettany)

Well as the song goes, 'Two out of three ain't bad'. We apologise for being slightly behind schedule this month, this is due to our annual holidays having taken place, but with winter approaching I'm sure you will all be spending more time with Albert and thus contributing something to the magazine!!!! Thank you to all those who already have, don't stop please.

Crystal Research have made the group a very special offer on the System 5 package, which includes a Basic compiler. To order you must use the back page order form and quote your membership number. If you do not know your member number it is on the address label following your name.

We would like some cartoons and jokes for the magazine, so if you have any or are an artistic type please put pen or brush to paper. Did you realise Marilyn Monroe was looking at you from the back page last month?

Last month we had an advertisement from Tony Brewer concerning Silicon Discs and Speculators, Tony is no longer selling these so please do not order from him.

We are at the moment trying to compile some documentation to go with the P.D. software, if you have sent for any of the P.D. packages and have enough time to send Jim a small review of the software this would help enormously.

Also on the help front, our copies of past programs are not quite as they should be. Some have minor bugs and others are in ASCII format including words from their accompanying article. If you feel you could sort some of these problems and add a little documentation on paper, send me a disc and let me know which back isues of the U.K. newsletters you have. I will then copy some, not too many, of the past programs that we would like corrected. We can then set about putting these into some order and adding them to the P.D. library.

One last point, we are considering changing the size of the magazine to A5, which is half of what it is now. This would reduce our printing costs and thus enable us to increase the number of pages and content, provided we get your input. If anyone objects now is the time to say so!!

MY GOLDEN DISC AWARD (Vic Day)

Having aired my grouses at Tatung's lack of support for the 256 and their refusal to produce a silicon ram patch utilising the VRAM I wish to announce my award of the Golden Disc (or should it be Golden Ram Disc) to Mr A J Cornish if Crystal Research.

Some weeks ago I wrote to Crystal Research saying that I couldn't use their System 5 on the 256 as it wouldn't run on 80 columns. I also mentioned my complaint about the silicon disc problem. Within days Mr Cornish wrote giving me details of a patch to enable me to run in 80 column mode. He mentioned that they (Crystal Research), hadn't ben involved with the DOS or BASIC of the 256 although they did the original.

Mr Cornish then said that he had produced an experimental patch to enable you to have a 128k Ram disc using VRAM. He also offered to give me a copy to evaluate if I would send my disc. this I did, and within a week I got my disc back, not with a 128k Ram disc but a 176k Ram disc (that makes the 112k of the Amstrad look silly). He produced a modified DOS, DOS 2.05e especially for the 256 which included all the patches for 80 column working. The Ram disc works in modes 0 to 2.

On booting the disc I found that I had a 176k disc on Disc2! Users of the original Einstein silicon disc will need no telling that a silicon (ram) disc has very little advantages if used with an interpreted language such as BASIC, although it only takes about 2 seconds to load BASIC from Ram disc you have to load it into the Ram disc first which takes the normal time of some 15 seconds.

The main advantage with a Ram disc is when you are using a language compiler. I have tried both Nevada fortran and Turbo Pascal with excellent results, the compiling speed must be seen to be believed, with none of the creaks and groans from an overworked disc drive. The only thing I haven't tried yet is with an Assembler. This will have to wait as machine code programming is not my forte, but perhaps one day I will try smeone elses program just for fun.

I don't know what Crystal Research's plans are for marketing this DOS, but no doubt they will be doing so once they are sure there are no bugs in it.

ED. See Crystal Research advert on the inside back page.

HAVE A NICE DAY? (Neville Last)

Here's a short program in XBAS that will give you the day for any date you care to enter. It doesn't take into account the changes of the Gregorian Calender.

```
 10 INPUT "Day ";D
 15 INPUt "MONTH ";M
 20 INPUT "YEAR ";Y
 25 Z=Y-1:LEAP=0
 30 IF Y MOD 4<>0 THEN 45
 35 IF Y MOD100<>0 THEN LEAP=1
 40 IF Y MOD400<>0 THEN LEAP=1
 45 X=INT(Z/4)-INT(Z/100)+INT(Z/400)
 50 X=X+D+Y-1
 55 IF M=2 THEN X=X+3
 60 IF M>2 THEN X=X+LEAP+INT(2.61*(M+1))
 65 RESTORE
 70 FOR I=1 TO 1+XMOD7
 75 READ DAY$
 80 NEXT
 85 PRINT;D;"/";M;"/";Y;" is a ";
 90 PRINT DAY$
 95 RUN
100 DATA Sunday,Monday,Tuesday,Wednesday,Thursday,Friday,Saturday
110 end
```

SEPTEMBER

M	T	W	T	F	S	S
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

EINSTEIN ASSEMBLY LANGUAGE COURSE

Dennis Hain has been looking into the Glentop disc/book publication.

It never occurred to me before, but all technical reviews in any magazine are written by experts, or at least someone who knows the subject. I HAVE noticed that experts have difficulty in spotting what is not clearly explained because, of course, they already understand it. Surely the best test is to give the material to a nerd and find out whether he learns anything.

On the strength of reviews in magazines I bought Mike Bayliss' course. I have used Basic for some time, but have always been baffled by so called 'beginners articles' on Assembly Languages.

I must agree at once with the reviewers that it is a good course and very clearly and simply explained. However, I am not so keen on what seem to be too many errors in the text book. (Or were they put in to see if I was paying attention?).

In Chapter 2, there is a very detailed explanation of the JP instruction; very easy to follow. Then I did something a knowledgeable reviewer probably would not, I keyed in the short program to see it work - it crashed! After a lot of work with pencil and paper counting bytes, it appears that JP 271 and JP 272 should have been 261 and 262 respectively. The former addresses were repeated throughout SIX pages of explanations. The latter addresses worked o.k. This sort of thing is discouraging to a member of the older generation struggling to keep up.

Other possible errors (I say possible because I could be wrong - I am a nerd, remember) are - Program 2.5: note after JP NZ,NXT (B=0) should be (B<>0), and surely DEC B should follow the INC A; Exercise 3.3: the solution given at the back of the book uses different X & Y co-ordinates to the problem; the example of masking with AND (page 5-6) has a wrong binary result; the third paragraph under FILL ROUTINES (page 10-5) seems to have a line missing as it makes no sense as it is.

In spite of these deterrents, I hope to try my hand at writing a program using the Assembler. Which raises another problem, all the course programs are stored at &4F00 (using an Offset) and run from &0100 (ORG 256).
So what do I do if I want to call the routine from BASIC which is stored from &0100? I know about using the CLEAR command to lower RAMTOP for MC routines, but how do I assign an Offset and ORG addresses to cater for this? The book does not seem to say.
This latter point is more important than the mipsrints, since it is ok to call routines from DOS in order to test them, but it is vital to be able to USE them from BASIC in order to make it worthwhile doing them in the first place. I wanted to write a small test routine in BASIC so that I could call the examples in the course as I worked through them.
Apart from these shortcomings, it is a good course, the book is well printed (not from dot matrix copy) and worth the cost, once I have found out how to use it from BASIC.
ED. The comments on the errors are all valid. To use your code from Basic, set an offset such that the program will run at say 8000Hex, save it and under DOS rename the file to make its extension .OBJ instead of .COM, you can then from Basic write a program that calls the object file. This would be something like the following:

```
10 CLEAR &8000
20 LOAD "TEST.OBJ"
30 CALL &8000
```

Note. The Basic manual tells you to Clear a location 1 less than where you wish the object file to be located, in this case it would be &7FFF however I have found that I have to Clear from the same address I wish the code to be loaded to?

NEWS..NEWS..NEWS..NEWS..NEWS..NEWS..NEWS..NEWS..NEWS..NEWS..NEWS

A NEW USER GROUP IN THE WEST COUNTRY

Chris Conyon is setting up an Einstein user group in the West country. He tells us that it will be a fairly local group for both learners and the more experienced in the Saltash, Plymouth, Tavistock, Callington and Tor Point area. The group hopes to produce hardware for controlling external devices via the user ports. There will be a charge of £1.00 to cover a quarterly newsletter and they will run a P.D. library. If there is enough response there will be local meetings and Chris is looking for enough members to arrange a coach trip to the U.K.E.U.G. National Exhibition in Birmingham. If you're interested drop a line to : CHRIS CONYON, 13 ST., ANNES ROAD, SALTASH, CORNWALL, PL12 6EJ.

U.K.E.U.G Bulletin board: Due to the holiday season being upon us, work on the bulletin board has been delayed. The software is now ready and it is just a matter of getting together the hardware and testing it. Start date to be announced.

ONELINERS

Kevin Lyons: (BBCBASIC)

```
10 INPUT"ANGLE INC. VALUE";A:B=133:C=100:CLS:PLOT-1,512,368:MOVE0,0:FORI=0TO36000 STEP
   A:J=2*I:X=J/COS(RAD(I)):Y=J/B*SIN(RAD(I)):DRAW X,Y:NEXT I:Z=GET:GOTO 10
```

Graham Bettany: (XBAS)

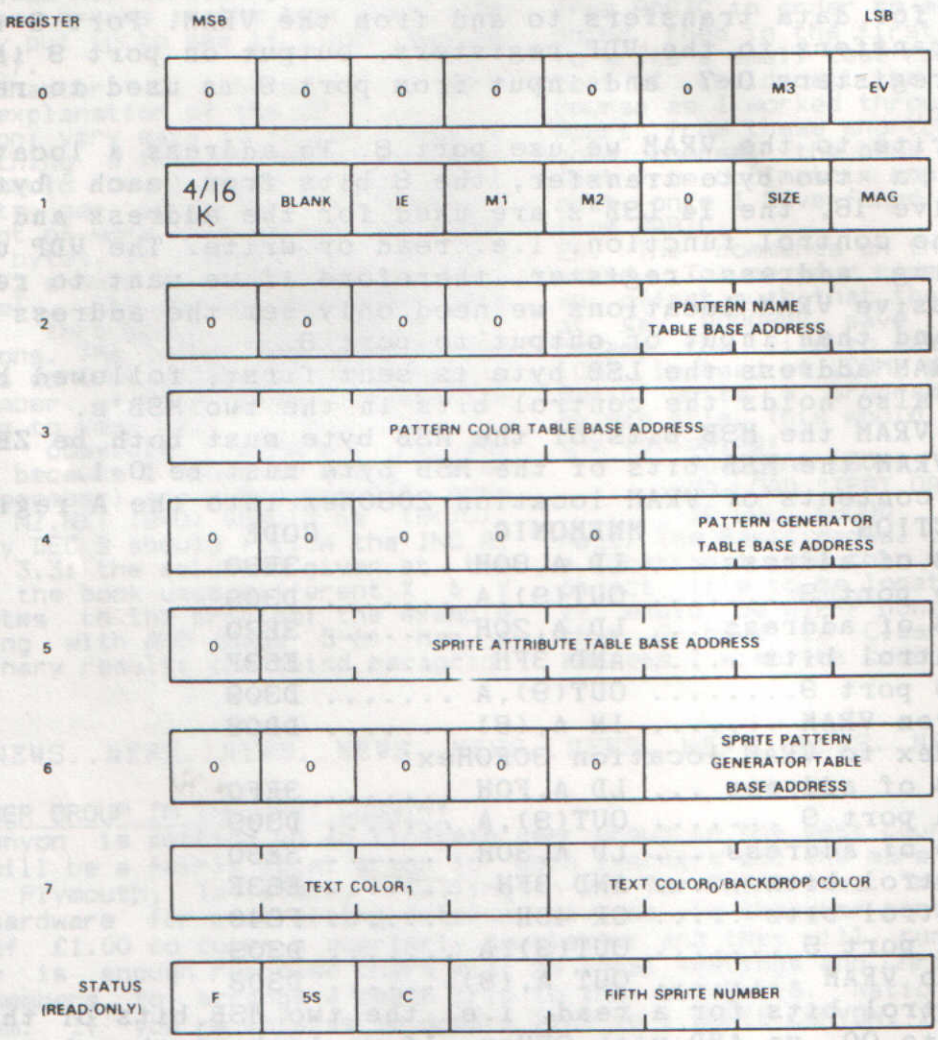
```
1 BCOLO:ORIGIN99,85:A=3:B=6:C=8:FORC=2TO15:FORI=0TO250:XX=Y-SGN(X)*SQR(ABS(B*X-C)):
  YY=A-X:X=XX:Y=YY:PLOTX*5,Y*5:GCOLC:NEXT:NEXT
```

To write to any of registers 0-7 requires a two byte transfer. The first byte contains the data to be written into the register and the second byte contains the register number we wish the data to be input to. The register number is held in the 3 LSB's, the MSB must be a 1 with all other bits set to ZERO.

	MSB							LSB
	: 7 :	6 :	5 :	4 :	3 :	2 :	1 :	0 :
BYTE 1/DATA	: ? :	? :	? :	? :	? :	? :	? :	? :
BYTE 2/REGISTER No ...	: 1 :	0 :	0 :	0 :	0 :	REG. No.		:

To read the STATUS register we read the value input from port 9.

Below is a diagram of the registers; they will be covered in more detail next month.



The VDP mode of operation is controlled by the Mode bits M1, M2, M3.

	M1	M2	M3
GRAPHICS MODE 1 ...	0	0	0
GRAPHICS MODE 2 ...	0	0	1
MULTICOLOUR	0	1	0
TEXT MODE	1	0	0

Last month's article gave a listing to save and load the screen from XBAS, it took several seconds to complete. Here is a machine code version that does it before you blink, well nearly! For those of you who find machine code a jungle there is a Hex dump and instructions on how to save it as an .OBJect file and CALL it from XBAS.

Source code for screen save and re-load, written with ZEN.

```
ORG 0100H
LOAD 4000H
SCRATCH:EQU 0D000H
LD HL,0B000H        ;START LOCATION IN RAM
LD (SCRATCH),HL     ;POINTER TO RAM
LD A,00             ;LSB OF ADDRESS
OUT(9),A            ;SEND TO VDP
LD A,00             ;MSB OF ADDRESS
AND 3FH             ;SET CONTROL BITS
OUT(9),A            ;SEND TO VDP
LD B,24             ;24 ROWS ON SCREEN
LOOP1:PUSH BC       ;ROW NUMBER
LD B,0              ;COUNT OF 256 PIXELLS
LOOP2:IN A,(8)      ;READ FROM VDP
LD HL,(SCRATCH)     ;UPDATE
LD (HL),A           ;THE
INC HL              ;RAM
LD (SCRATCH),HL     ;POINTER
DJNZ LOOP2          ;256 PIXELLS DONE?
POP BC              ;GET NEXT ROW
DJNZ LOOP1          ;DO 256 PIXELLS
RET                 ;FINISH OF SCREEN SAVE
LD DE,0B000H        ;START OF SCREEN LOAD
LD HL,0             ;FROM RAM TO VRAM
LD B,24             ;24 ROWS
LOOP3:PUSH BC       ;SAVE ROW COUNT
LD B,0              ;256 PIXELLS
LOOP4:PUSH HL       ;SAVE VRAM ADDRESS
PUSH DE             ;SAVE RAM ADDRESS
LD A,L              ;LSB OF VRAM ADDRESS
OUT(9),A            ;SEND TO VDP
LD A,H              ;MSB OF VRAM ADDRESS
OR 40H              ;SET CONTROL BITS
OUT(9),A            ;SEND TO VDP
POP DE              ;GET RAM ADDRESS
LD A,(DE)           ;GET DATA FROM RAM
OUT(8),A            ;SEND TO VDP
POP HL              ;UPDATE
INC HL              ;VRAM POINTER
INC DE              ;RAM POINTER
DJNZ LOOP4          ;256 PIXELLS DONE?
POP BC              ;NEXT ROW
DJNZ LOOP3          ;DO NEXT 256 PIXELLS
RET                 ;FINISH SCREEN LOAD
NOP
END
```

The routine to save the screen uses the VDP's auto-incrementing address register, having set up the start address as VRAM 0000, each memory location of the pattern table is read into main RAM starting at B000Hex. We end up with a copy of the 6k bytes from VRAM 0000-17FFHex, (remember this is copy of the screen), in RAM locations B000-C7FFHex. The screen load routine is written in a different way, we are not using the auto-incrementing VDP address register but setting the VRAM address every time. This was just to show a different approach. The screen load program is very similar to the save, we set the VRAM address, read from RAM the data to be displayed and send it to the pattern table, and so on until all 6k bytes have been transfered. To use the screen load routine after having saved a screen in RAM requires a call to the start location. This is 0125H for the .COM file.

NOTE: the VDP requires an 8 mico-second delay between successive read or write operations usually this is not a problem but if in line code is used a delay must be included.

O.K. so you did'nt understand the machine code but you would like to use the routine from XBAS, this is what you do. Enter DOS with a CTRL-BREAK, then type MOS<E>.

1 .. M0100<E> you will see location 0100H and it's present contents being displayed on the screen. What you need to now do is enter the following code, I usually enter lines of 8 bytes followed by ENTER. After you have finished entering the code exit the modify by using .<E> (full stop then enter). You should now check that everything is correct by tabulating the memory you have just altered,

2 .. T 0100 0180<E> you should see the same Hex dump as is printed below, if it is not the same then go back and modify the differing locations. When you have completed entering the code the next step is to save it as a file so it can be used from XBAS. The type of file we need is a .OBJect file, so enter DOS with a CTRL-BREAK and type

3 .. SAVE 1 SCREEN.OBJ<E> We now have a machine code file that can be LOADed into a basic program and CALLed. If you check the DIRectory you will see the file SCREEN.OBJ now present, the 1 after the SAVE command tells the DOS how much code to save in 256 byte blocks.

Hex dump of ths screen save/load program;

```
0100 21 00 B0 22 00 D0 3E 00
0108 D3 09 3E 00 E6 3F D3 09
0110 06 18 C5 06 00 DB 08 2A
0118 00 D0 77 23 22 00 D0 10
0120 F4 C1 10 EE C9 11 00 B0
0128 21 00 00 06 18 C5 06 00
0130 E5 D5 7D D3 09 7C F6 40
0138 D3 09 D1 1A D3 08 E1 23
0140 13 10 ED C1 10 E7 C9 00
```

To use the program from XBAS we have to be careful where we load the SCREEN.OBJ file. The program is using memory locations B000Hex to C7FFHex to save the VRAM, XBAS uses memory upto approximately 4000Hex so we can use memory somewhere between the two to locate our .OBJ file. The example uses A000Hex which leaves plenty of room for any basic programing. All we need to do is to CLEAR &A000 and CALL &A000 to run the first part of the program, the screen save. To run the second part, the screen load we need to CALL &A025. Having saved a screen we can then save the RAM as another .OBJ file for use later on.

```
SCREEN SAVE:
10 CLS
20 CLEAR &A000
30 LOAD "SCREEN.OBJ"
40 ELLIPSE 100,100,60
50 CALL &A000
60 SAVE "DEMO.OBJ",&B000,&C7FF
```

```
SCREEN LOAD:
10 CLS
20 CLEAR &A000
30 LOAD "SCREEN.OBJ"
40 CLEAR &B000
50 LOAD "DEMO.OBJ"
60 CALL &A025
```

Line 40 in the save program is where your artistic impressions would go. This principle could be used to flash between two or more screens saved in RAM or a form of windowing could be designed. It will not work with colour as it stands but a very similar routine for the colour table could be written. I'm sure you could improve this routine and it is really included to stimulate ideas so lets have some!!

COMPETITION RESULT

Last months competition in the Trolls Lair returned several entries with answers ranging from 10 to over 2000 diamonds! The reason many ended with the answer as 999 coins was the pressumtion that all the Trolls coins came from and were returned to the coin store. This was not the case for transactions and was only true in the last cavern. The actual answer was 407 coins taken, this gave 65 diamonds. The two winning entries were from Kevin Turner and Doug Barradine, Kevin wins Apollo II and Doug the Einstein Primer book.

Both programs were very similar and this is Kevin's;

```
 5 RST:CA=0:DA=0
 10 FOR CS=1 TO 999
 15 C=CS:A=0:E=0:D=0:B=0:DO=0:CT=0
 20 A=INT(C/10):C=C MOD 10
 30 E=E+1:D=D+C
 40 B=D:D=0:D=C
 50 B=B-1:IF B>0 THEN C=C+D:GOTO 50
 70 IF E>0 THEN E=E-1:GOTO 40
 80 CT=CT+C:C=0:DO=DO+D:D=0:IF A=0 THEN GOTO 100
 90 C=A:A=0:GOTO 20
100 IF CT+(1000-CS)=1000 AND DO>DA THEN DA=DO:CA=CS
110 NEXT CS
120 PRINT CA;" COINS GIVE YOU A MAXIMUM OF ";DA;" DIAMONDS"
```

```
VARIABLES: CA .. COINS ANSWER   DA .. DIAMONDS ANSWER
           CS .. COINS START    C ... COINS
           A ... APPLES         E ... EMERALD
           D ... DIAMONDS       B ... BEADS
           DO .. DIAMONDS OUT   CT .. COINS TAKEN BY TROLL
```

RANDOM JOTTINGS (Mystron)

The birth was loud and pronounced and great things were foretold for this new life. Many rushed to give it aid believing that they would be rewarded with words of wisdom. Alas it was not to be, a long drawn out agony was turning to a slow and painful end. One last desperate attempt was made to save it, new blood was injected but all in vain. Those who had brought it forth melted away into the night and those who had given aid waited in the unknown. The monitor beeps had stopped and two words were written on the screen, BRAIN DEATH.

U.K. EINSTEIN SHOW

WHEN . . 14th of NOVEMBER.
10am — 5pm.

WHERE . NATIONAL MOTORCYCLE MUSEUM, BIRMINGHAM.

Your chance to meet SCREENS, MIKE BAYLISS, B+H and more. See what is available for Albert. Addmision will be by program at 50p with a draw ticket giving you the chance to win one of several great prizes. There will be a bar from 11am and a good day all round is envisaged. If any company or individual would like to exhibit at the show then please contact the U.K.E.U.G. (address on the back page).

GAMES_PLAY (Harry The Hacker)

This month I am giving over the first part of this piece to a game review after which I shall be telling you how to get infinite lives on JET SET WILLY.

HOUSE OF USHER - MERLIN SOFTWARE £12.95
(Reviewed by Alice Jackson, Aged 10)

It's a game a bit like Jet Set Willy but in 3-D and is very risky like if you go through a door, unless you go gently, you might fall down and you might never get up again. Also if you jump up some boxes which get higher and then jump off the other end you have to risk never going back.

Quite good but once you're stuck that is the end of that life and you only have ten lives to start with. It's quite hard to go up on the boxes and go through a door. Those are the things that are not so good but the graphics are brilliant!

I think it is very good value for money and is exciting - my Dad says that the idea of a Monk having to find his way round a labyrinth to find treasures while avoiding all sorts of hazards reminded him very much of the film "Name of the Rose" so he kept imagining he was Sean Connery.

It's very easy to load and is good with a joystick, all in all brilliant!

My thanks to Alice for that review, I've tried Usher and the one critisism I would make is that the programmer hasn't made a great use of sound.

Now to JSW and how to get infinite lives, on the Spectrum version you just held down a few keys and not only did you get infinite lives but you could select rooms. This doesn't work on the Einstein version unless they have changed the code, anyone out there know?

Never mind we can still have infinite lives, what I did was to LOAD JETSET.COM after first loading in a FIND routine, I used the routine sent in by Tom Crossfield that we shall be publishing in next months mag. The FIND routine enabled me to search through the JSW code until it found a 3E07 this as some of you may know is machine code for load the A register with 7. But I'm not here to teach you machine code, the 7 is the number of lives you have at the start of JSW so they must be stored somewhere. Next I loaded in a disassembler and looked at the code following the 3E07 I soon came across the address in memory where the 7 was stored, then using FIND again I found where in the program the 7 was decremented and put in a NOP (No operation) instead.

Once the decrement had been taken out all that remained was to save the modified program. So this is it, first make a Backup of your JSW disc. With the copy in drive 0 reset the machine and type the following:

```
LOAD JETSET.COM <E>
MOS <E>
M6051 <E>
OO. <E>
Y <E>
```

Now you should be back in DOS so type SAVE 108 JSW.COM <E> and there you have it an infinite life Willy.

Happy exploring and watch out for the Quirkafleeg.

ANALOGUE/DIGITAL CONVERSION. (Peter Moon.)

The Einstein's analogue/digital conversion capability gives a useful means of monitoring external equipment, provided that what is going on externally can be can be expressed as a dc voltage signal to the computer. The A/D converter has four independent channels, each capable of accepting a dc signal and turning it into an integer proportional to the voltage. In XBASIC the output is 0 - 255 for an input of 0 - 2.1 volts (the maximum). One limitation is immediately apparent; the smallest voltage change detectable is 1/255 x 2.1 volts or about 8 millivolts, so a smoothly changing voltage signal will appear to move in jumps of 8 mV.

The other drawback shows up as soon as the A/D converter is connected to a steady voltage source such as a battery. The digitised reading jumps about all over the place, varying by 60 mV or more, because the circuit is very "noisy", and even using screened cable and putting a capacitor across the signal input doesn't improve matters.

The only way to improve the accuracy is to sample the signal many times and take an average. The table below shows the results of twelve repeat tests on on a steady voltage signal from a 1.5 V battery, taking from 1 to 1000 readings in each of the twelve tests. Standard deviation is a measure of the scatter in the results. A departure of twice the S.D. from the mean can be expected about once in every 20 readings and three times the S.D. once in a hundred.

No. of readings in each of the 12 tests	Standard deviation of the 12 means (volts)
1	0.033
10	0.0101
100	0.00382
1000	0.00108

As you would expect, the more readings in each test, the less variation there is between the averages of the tests. In fact, the scatter improves as the square root of the number of readings in each test, so for a tenfold improvement you need to take 100 times as many readings. This need not bother you, as 1000 readings can be taken and averaged in about 10 seconds, and compared with 10 readings, the S/D is improved from 0.0101 to 0.00108 as a result. Expressed in practical terms, the shakiness of the measurement (taken as 2 x S.D) is cut down from +/-20mV to +/-2mV by this means.

Practical measurements.

The Einstein has two 7-pin DIN sockets (MO14 and MO15, see "An Introduction to Einstein", page 216) each handling two A/D channels. Thus you can monitor up to four sources using variables ADC(0) to ADC(3).

As well as monitoring a voltage source you can measure the position of a potentiometer. The Einstein provides a reference voltage of 5V which can be connected to earth via a potentiometer, forming a potential divider. The wiper takes up a voltage depending on its position along the resistance track of the potentiometer. This is the basis of joystick control. The "fire" switches are monitored by BTN(0) and BTN(1). What the book doesn't tell you are the important bits.

Connections.

In all applications, connect pins 2 and 6 together.

Voltage source: Positive to pin 1, negative to pin 6. Max. voltage 2.1.

Potentiometer: Track ends to pins 5 and 6, wiper to pin 1. To avoid overloading the reference voltage source, the resistance of the whole potentiometer track should be not less than 47k.ohms.

Fire button: Connect pins 2 and 4 when "firing". BTN(0) changes from 1 to 0 when this connection is made.
Programme.

Using the above information you can run the programme below to plot on the screen a battery discharge curve and obtain a printed graph of the result.

```
 10 REM ADCGRAPH, TO PLOT VOLTAGE INPUT VIA A/D CONVERTER.
 20 REM DOS 1.11, XBAS 4.12.
 30 REM REQUIRES GDUMP2. Peter Moon 30 July 1987.
 40 CLS
 50 CLEAR &A000
 60 REM DRAW AXES
 70 ORIGIN 25,25
 80 DRAW 0,0 TO 0,166
 90 DRAW 0,0 TO 231,0
100 REM GRADUATE X AXIS
110 FOR X=0 TO 231 STEP 4: REM 4 PLOTTING UNITS=12 MINUTES = 1/5HR.
120 DRAW X,0 TO X,-2
130 NEXT X
140 REM PUT LONG MARKS EVERY HOUR.
150 FOR X=0 TO 231 STEP 20
160 DRAW X,0 TO X,-7
170 NEXT X
180 REM GRADUATE Y AXIS
190 FOR Y=0 TO 166 STEP 3: REM 3 PLOTTING UNITS = 0.01 VOLT.
200 DRAW 0,Y TO -2,Y
210 NEXT Y
220 REM PUT LONG MARKS EVERY 0.05 VOLT
230 FOR Y=0 TO 166 STEP 15:REM 30 PLOTTING UNITS = 0.1 VOLT
240 DRAW 0,Y TO -5,Y
250 NEXT Y
260 REM CALIBRATE A/D CONVERTER
270 INPUT "TRUE VOLTAGE = ";VT
280 PRINT@ 0,20;"0.8"
290 PRINT@ 0,17;"0.9"
300 PRINT@ 0,13;"1.0"
310 PRINT@ 0,9;"1.1"
320 PRINT@ 0,5;"1.2"
330 PRINT@ 0,1;"1.3"
340 PRINT@ 1,3;"V"
350 PRINT@ 4,22;"0"
360 PRINT@ 10,22;"2"
370 PRINT@ 17,22;"4"
380 PRINT@ 24,22;"6"
390 PRINT@ 30,22;"8"
400 PRINT@ 36,22;"10h"
410 REM SAMPLE VOLTAGE ON CHANNEL 0 AND AVERAGE THE RESULTS
420 SIGA=0
430 FOR I=1 TO 1000
440 SIGA=SIGA+ADC(0)
450 NEXT I
460 A=SIGA/1000
470 F=VT/A
480 FMT 1,4
490 PRINT@ 28,0;F;" F"
500 TM=0
510 TI$="000000"
```

```
520 IF MID$(TI$,3,2)<>"03" THEN 540 ELSE 580
530 REM IF VOLTAGE IS LOW, STOP MONITORING AND PLOT GRAPH.
540 IF ADC(0)>50 THEN 520
550 LOAD "GDUMP2.OBJ"
560 CALL &A000
570 GOTO 690
580 TM=TM+3
590 TI$="000000"
600 PRINT CHR$(7)
610 SIGV=0
620 FOR I=1 TO 1000
630 SIGV=SIGV+F*ADC(0)
640 NEXT I
650 V=SIGV/1000
660 PRINT@ 28,1;V;" V"
670 PLOT TM/3,300*(V-0.8)
680 GOTO 520
690 END
```

The programme allows you to input battery voltage, as measured by an analogue voltmeter, at the start of the experiment so as to calibrate the A/D converter in terms of volts per computer unit (F). It uses the TI$ function to measure intervals of 3 minutes, at the end of which the voltage is sampled 1000 times, averaged, displayed on the screen as V, and plotted. When the battery goes flat or you want to end the test, disconnect it and the programme will call up one of the GDUMP routines (Newsletter Vol. 2 No.3, Jan. 1978, page 48) and print the screen contents. Alternatively you could arrange to end the test by connecting pins 2 and 4, and replacing line 540 by 540 IF BTN(0)=1 THEN 520.

JOYSTICKS FOR THE MASSES (Steve Cooper)

Steve Cooper has sent us this hardware project for those of you who want to use the Commodore/Atari type joysticks. Give it a try it or get someone to do it for you it really is worth the effort.
The only non-unidirectional components are the transistors as you can see from the drawings these particular ones are 'D' shaped, which makes life easier.

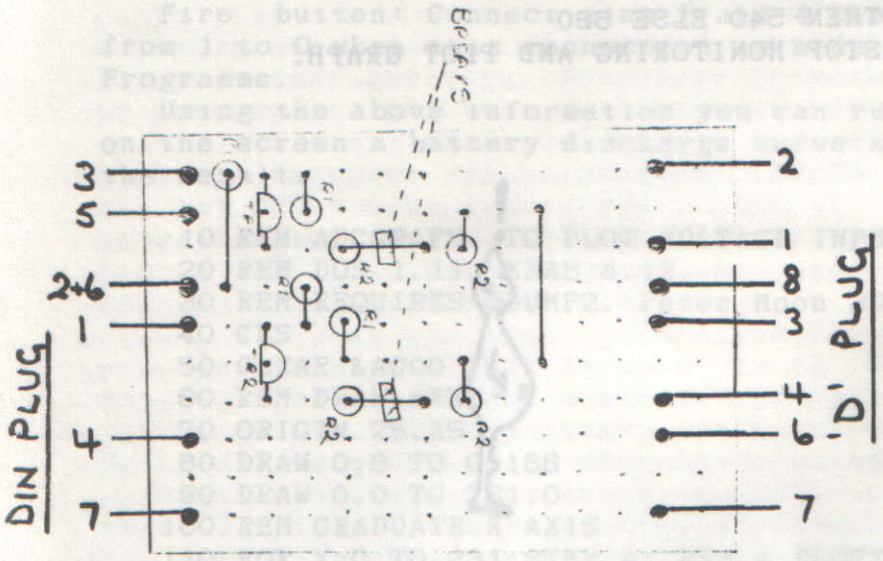
All the components are common items available Maplins or any reasonable electrical shop.

This option also allows the user the option of auto-fire, where the over the counter ones don't.

SHOPPING LIST

ITEM	QUANTITY	APPROX. COST EACH
RESISTOR (1K)	4	12pence.
RESISTOR (100K)	4	12pence.
TRANSISTOR (2N3702)	2	60pence.
7 PIN "DIN" PLUG	1	35pence.
9 PIN "D" PLUG (MALE)	1	50pence.
SMALL VERO BOARD	1	30pence.
CABLE (Any lightweight cable provided it has at least 7 cores)	1yd	35pence.
SMALL PROJECT BOX	1	90pence.
	Approx. total	£3.24

ALL RESISTORS ARE 1/4 WATT 5%
THE COLOUR CODES FOR THE RESISTORS ARE AS FOLLOWS:
R1 (100K) BROWN - BLACK - YELLOW - GOLD
R2 (10K) BROWN - BLACK - RED - GOLD



THINGS TO REMEMBER

1. The two small breaks in the veroboard tracks. Use the corner of a screwdriver.
2. Get the right wires on the right pins. If you look closely at the back of the plug they are numbered.
3. Make sure no excess solder shorts out pins or tracks.
4. Make sure the transistors are the right way round.

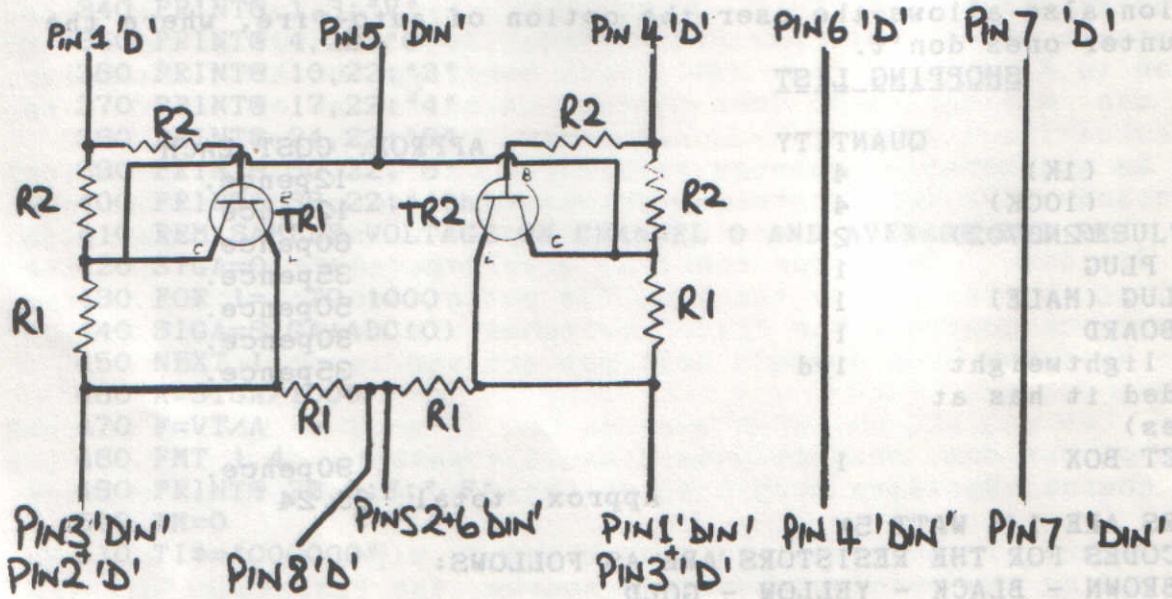
Once you have put it all together you need to try it out so here is a test program.

```
10 CLS:PRINT @10,11;"ANALOGUE 1 TEST BED"
20 PRINT @3,7;"ADC(0) (LEFT & RIGHT) "
30 PRINT @3,12;"ADC(1) (UP & DOWN)   "
40 PRINT @3,17;"BTN(0) (FIRE BUTTON)  "
50 A=ADC(0):B=ADC(1):F=BTN(0)
60 PRINT @30,7;A"  ":?@30,12;B"  ":?@30,17;F:GOTO 50
```

With the joystick and adapter plugged into analogue 1 you should get the following results when the program is run:-

JOYSTICK CENTRAL	--	ADC (0+1) 110 TO 120 - BTN(0) 1
FIRE BUTTON PRESSED	--	BTN(0) 0
JOYSTICK RIGHT	--	ADC(0) >250
JOYSTICK LEFT	--	ADC(0) <5
JOYSTICK UP	--	ADC(1) >250
JOYSTICK DOWN	--	ADC(1) <5

Our thanks to Steve for a good hardware project, is there any more out there? if so don't keep them to yourselves let us all in on the act.



COMPETITION

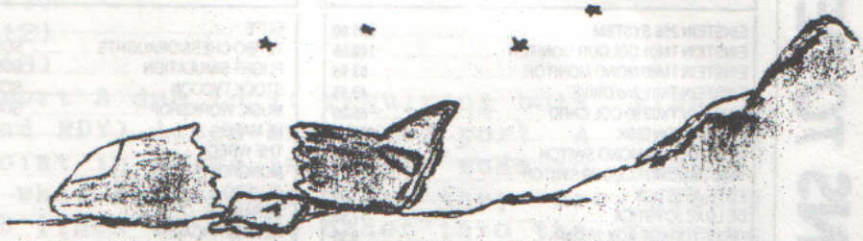
Our thanks to SCREENS for supplying the prize for this month's competition, which is a copy of the KUMA package 'COMMUNICATIONS with VIEWDATA'. Add this piece of software to your modem and get into the world of Prestel and Bulletin Boards. See the advertisement in this issue for SCREENS address and telephone number.

It is the 21st century and Ted, who is a space agricultural scientist is starting out on what seems to be a routine mission. He has with him samples of a new strain of food plants and is setting out in Galaxy 3 to find a suitable planet to test the growing parameters.

After lift off a malfunction in the organic fuel system propels Ted's craft out of Galaxy 3 and into unchartered space. Unable to control the craft Ted watches as he crashes down onto a very barren terrain. The craft is completely destroyed, Ted is miraculously thrown clear in the explosion. He is carrying a small amount of food and some of the food plants.

Ted has:

Food for 10 days
15 Parma seeds
10 Destro seeds
5 Gansy seeds



In this enviroment;
Parma seeds yield 2 meals after 7 days but die after 9 days.
Destro seeds yeild 3 meals after 9 days but die after 12 days.
Gansy seeds yeild 4 meals after 12 days but die after 16 days.
Meals from a plant last 4 days including the day they were picked, if the plant dies the food turns poisonous and is fatal if eaten.
Ted knows that with these strains of food you can only plant one seed a day and each successive planting must be of a different strain seed. Ted needs 2 meals a day, how long can he survive?

Send in your program to solve the above problem, written in XBAS or BBCBASIC. The end product being a list of what day Ted should plant what seed, taking the day he crash landed as day 1, and the total days Ted can expect to survive. Closing date 14th October.

****
Could you set a competition? We have a copy of House of Usher, kindly given by B+H and reviewed in this issue which we will give to anyone sending in a competition which we can use.
****
Still in competition mode Crystal Research have offered a really superb prize of System 5, including the Basic compiler, plus their book Albert Revealed, to the best article published in the next 2 issues. If the winner already has either or both of the above they will refund the cost. Albert Revealed is due to be published at the end of September and will be priced at £14:95 plus £1:00 P+P. Worth some effort I think, so GO FOR IT.
****

SCREENS SCREENS SCREENS

LARGEST EINSTEIN CENTRE IN THE UNITED KINGDOM

OFFICIALLY APPOINTED TATUNG DISTRIBUTORS

IBM COMPATIBLE PC XT AT COMPUTERS

Check our specifications, check our prices, SCREENS cannot be beaten!
Every model is exclusive to SCREENS in the United Kingdom!

ST100 BEGINNER SYSTEM — £399!
● 8088-2 CPU (4.77/8 MHz) ● 512K RAM ● 360K Disk Drive ● Printer Port ● Serial Port ● 12" Amber Monitor ● MS-DOS ● Turbo Processor.

CL-2D TURBO DESKTOP PC — £599!
● 8088-2 CPU (4.77/8MHz) ● 640K RAM ● Dual Disk Drives ● 2 x Printer Ports ● 2 x Serial Ports ● Games Port ● 12" Monitor ● MS-DOS/GW Basic ● Price includes one year on-site maintenance in UK.

CL-20 DESKTOP TURBO XT — £945!
● 8088-2 CPU (4.77/8 MHz) 640K RAM ● 360K Disk Drive ● 20MB Hard Disk ● Printer/Serial/Games Port ● Real-time clock with battery backup ● One year maintenance included

ST 320 ENHANCE BABY AT — £1499!
● 80286-10 CPU (6/10/12 MHz) 1MB RAM ● 1.2MB Floppy Drive ● 30MB Hard drive ● 14" Monitor ● Clock and battery backup.

ST386 16MHz DESKTOP AT — £2995!
● 80386 CPU (16MHz) 2MB RAM ● 1.2MB Floppy drive ● 40MB fast access Winchester ● 14" Monitor ● Numerous ports.

ALL PRICES EXCLUSIVE OF VALUE ADDED TAX

REQUEST OUR EXTENSIVE PRICE LIST TODAY!

SCREENVISION

TURNS YOUR TM01 MONITOR INTO A COLOUR T.V. SET!

By simply connecting your TM01 Monitor to SCREENVISION using your same lead your monitor becomes a Colour Television Set! Now you can have dual use from your monitor. Each SCREENVISION has its own speaker, volume, colour, contrast and tone controls and gives clear reception on all channels when connected to your aerial system.

£59.95 INC VAT

HURRY! Only available at new low price while stocks last

PANASONIC

KX-P1081 NLQ Printer

120 cps Draft 24 cps NLQ. Built in buffer. Tractor & friction feeds. ASCII, italic & international characters. Centronics parallel & IBM compatible. Our price includes Einstein TC01 printer cable. A superb quality NLQ printer at an exceptional price.

£179.95 INC VAT

HARDWARE

EINSTEIN 256 SYSTEM	365.00
EINSTEIN TM01 COLOUR MONITOR	149.95
EINSTEIN TM80 MONO MONITOR	83.95
EINSTEIN TK01 2nd DRIVE	49.95
EINSTEIN TK02 80 COL CARD	65.00
256K SILICON DISK	85.95
40/80 COLUMN MONO SWITCH	12.95
40/80 COLUMN COLOUR SWITCH	21.95
ELITE JOYSTICK	11.95
DE LUXE JOYSTICK	17.95
DISK STORAGE BOX 20 CAP	9.95
DISK STORAGE BOX 30 CAP	13.95
PRINTER CABLE	10.95
TC01 DUST COVER inc MONITOR	8.95
13 AMP ANTI-SURGE PLUG	14.95
WS 4000 MODEM + SOFTWARE	195.00
EINSTEIN PRIMER BOOK	9.95
RELATIVELY BASIC BOOK	9.95
HARDWARE MANUAL EINSTEIN	25.00

SOFTWARE BEST BUYS

ELITE	FIREBIRD	19.95
TURBO CHESS/DRAUGHTS	SCREENSOFT	12.00
FLIGHT SIMULATION	SCREENSOFT	12.00
STOCK TYCOON	SCREENSOFT	9.00
MUSIC WORKSHOP	SCREENSOFT	22.95
LE MANS	ELECTRIC	12.00
THE WRECK	ELECTRIC	13.00
MONOPOLY	L. GENIUS	13.95
CLUEDO	L. GENIUS	13.95
SCRABBLE	L. GENIUS	13.95
KONAMI ARCADE 4	KONAMI	13.95
SUSPENDED	INFOCOM	20.00
DEADLINE	INFOCOM	20.00
HITCHICKERS GUIDE	INFOCOM	23.00
PAKMAN/MILLIPEDE	KUMA	11.95
ISLAND OF ARTAN	KUMA	14.95
SPACE TRAP	SOLO	11.95
JET SET WILLY II	S. PROJECTS	14.95

TASWORD W.P. inc 256	TASMAN	19.95
TASPRINT	TASMAN	12.95
CADMASTER 256	SCREENSOFT	69.95
HOME BUDGET	KUMA	14.95
SUPER STORE DATABASE	KUMA	19.95
DISMOD	KUMA	16.95
VIEWDATA COMMS	KUMA	19.95
NEVADA COBOL	NEVADA	39.95
CPM 2-2 + 80 col BASIC	TATUNG	89.95
AMTAT CPM EMULATOR	SYNTAXSOFT	35.95
CRACKER SPREADSHEET	SOFT TECH	57.50
PERSONAL ASSISTANT	FYDLER	39.95
INVOICING	TATUNG	17.95
HISOFT C	HISOFT	35.95
DATASTAR DATABASE	EINSOFT	99.95
BBC BASIC + MANUAL	EINSOFT	35.00
MASTERMAIL MAILING	FYDLER	29.95
STOCK CONTROL	BOWDENE	109.00

3" CF2 FLOPPY DISKS

£28 FOR 10! | £14.50 FOR 5! | £2.95 FOR 1!

SCREENS SCOOP!

BUY NOW AND SAVE!

FIDELITY 14" COLOUR PORTABLE T.V.
Full screen styling. Push button control. Various colour cabinets. Amazing price only as stocks lasts. **£129.95!**

FIDELITY 14" REMOTE CONTROL T.V.
As above but including latest infra-red remote control. **£159.95!**

CAD-MASTER 256

BY P. T. SWATTON

Cad-Master is a full functional computer aided design system for engineering, electrical, architectural drawing & poster design. Designed to make operation as simple as possible on your Einstein 256 with pop up menus and various screen areas. Incorporates floating point package to incredible 15 decimal places and built in "Zoom", "Text", "Circle", "Points", "View", "Rotate", "Mirror", and much more. Cad-Master incorporates a plotter driver enabling drawings to be plotted to the scale of users choice. **An incredibly powerful package at an unheard of price From SCREENSOFT.**

£69.95 INC VAT

MUSIC WORKSHOP

BY DR. D. SALVAGE

Designed for both TC01 & 256 computers. Music Composer allows music to be composed, edited, saved and replayed. Unlike other products you may also print your music in manuscript form from your Tatung or Epson printers. Music composer is supplied with a comprehensive manual and a step by step guide to music creation. Sample tunes are also included to demonstrate the amazing capabilities of your Einstein computer.

£22.95 INC VAT

DEAL WITH THE BIGGEST, DEAL WITH THE BEST! HUGE EINSTEIN PRODUCT LIST SENT FREE ON REQUEST. REMEMBER NO ONE SELLS CHEAPER THAN SCREENS. MANY NEW TITLES DUE UNDER SCREENS OWN SCREENSOFT LABEL. BIG DISCOUNTS!

SCREENS MICROCOMPUTERS & ELECTRONICS LTD

TRUST SCREENS TO DELIVER

MAIN AVENUE · MOOR PARK · NORTHWOOD · MIDDLESEX · ENGLAND

TEL: SALES 09274 20527 TEL: EXPORT 09274 20664 TLX: 929224 SCREEN G FAX: 0923 240402

HOW TO ORDER

PERSONAL CALLERS WELCOME MONDAY-SATURDAY 9am-5.30pm EARLY CLOSING WEDNESDAYS. OPPOSITE MOOR PARK UNDERGROUND STATION. COFFEE ALWAYS HOT! **TELEPHONE** QUOTING CREDIT CARD NUMBER FOR FAST DESPATCH. **WRITTEN** ORDERS WITH PAYMENT PLEASE ALLOW 14 DAYS DELIVERY. **DELIVERY** ORDERS BELOW £200 ADD £4.00 OVER £200 ADD £6.
ALL PRICES INCLUDE VAT EXCEPT WHERE SHOWN. ALL OFFERS SUBJECT TO AVAILABLITY.

SCREENS TRUST SCREENS TO DELIVER SCREENS

THE Z80 PIO (Dave Harvey)

In writing this article I have had to assume the reader has a knowledge of m/c code, this is necessary since interrupts are involved which need to be handled in m/c code.

The Z80 PIO chip has two "sides" on one side is port A, which is used for the printer port, and on the other port B, which is the user port. They are more or less the same to the programmer, however port B may directly drive Darlington Transistors, this is usful for control purposes.

The pinnouts for the port are shown in the back of the Einstein Intro Book (page 218 in mine). There are 8 data lines (D0 - D7), a data ready line (RDY) and a strobe line(STB) the rest are either +5 or 0 volts. It must be stressed that any experimenting must of course be done with great caution as not only could the user port be damaged but the printer port would also go being part of the same chip! Having said that it is as well to have a use for the port or it's pointless being able to program it, so if your not to up on wiring up a transistor or two or using logic I.C.s then beware and get someone to help. The uses I have put mine to are a link to my Dragon 32 and to run a speech synth', although I must admit I never use either of them, all the fun was finding out how.

The PIO can be programmed in any one of 4 modes, they are:

OUTPUT MODE (0)
INPUT MODE (1)
BIDIRECTIONAL MODE (2)
CONTROL MODE (3)

Mode 2 can only be used with port A due to it requiring both side A and side B handshake (STB and RDY) lines and since port A is the printer port there is little point in going into this mode to deeply.

I'd better first explain what Handshakes are, they are simply signals sent on the RDY and STB lines to force order into the passing of data from PIO to peripheral and back. Here is what happens with the Z80 PIO, first in mode 0, when data is sent to port &32, the RDY line will go from low to high to tell the peripheral device that data is awaiting its attention (on the 8 data lines) when the device sees this and reads the data it will lower and then raise the STB line to say the data has been taken. At this point the PIO will lower its RDY line and this cycle may then be repeated to send the next byte. The sequence for mode 1 is as follows, the peripheral device will lower and then raise the STB line to say it has data for the PIO and the PIO will lower its RDY line to say it is busy taking the data, when it does in fact take the data the ready line will return to high and the cycle is complete and may be repeated. Although the Interrupt, which is generated by the rising STB line in both cases has not yet been mentioned these are the handshake sequences for the input/output modes 0 and 1.

It is important to note that these handshake sequences are not suitable, as they are, for centronics type operation, as the Z80 PIO doesn't follow the normal protocol of waiting for the device to become ready and then sending the data followed by a strobe it in fact does it backwards to the other chips I have programed, the Motorola 6821 PIA and the 6522 VIA which,especially the later, are programmable to suit the peripherals requirements. The Einstein has an extra chip added to port A to make the handshake logic appear the other way round, this is what I had to do with the speech synth chip which was made to require the centronics type sequence.

Remaining with modes 0 and 1 I will now go on to the interrupts which are as I said before generated when the strobe line goes up. The Z80 PIO is used via interrupts, in order to explain how it all works, we will stick with output mode(0) for now, and go through the procedure for setting the port up for that mode and sending a byte of data.

The first thing we should do before setting up the port is to get our interrupt routine in position and the vector at address FB12/3 pointing to it. The interrupt routine may do one of two things, it may handle the sending of data from a file on it's own or it may simply set a flag in memory to tell us that a strobe has been recieved and therefore that the peripheral is ready for the next byte (this would be like polling). We will choose the simpler of the two, the latter. The routine looks like this:

```
INTRPT:PUSH AF
       LD A,1
       LD (FLAG),A
       POP AF
       EI
       RETI
```

This routine would be the same for input and output note that all reg's used must be preserved as the Z80 doesn't do this for us, also the interrupts are re-enable on exit from the routine (EI). Because this is an interrupt all our program needs to do is clear the flag, send the data to the port and then check the flag to see if the data has been taken yet. So, we have our interrupt routine in position and are ready to set up the vector and the port here is a routine to do it:

```
PIOOUT:PUSH HL
       PUSH AF
       LD HL,INTRPT
       LD (FB12),HL
       LD A,12H
       OUT (33H),A
       LD A,0FH
       OUT (33H),A
       LD A,87H
       OUT (33H),A
       POP AF
       POP HL
       RET
```

You can see that 3 bytes have been sent to the port B port control register, the first is the ports half of the interrupt vector address the higher half is supplied by the Z80 I register (it's already set to FB) so when the interrupt is called we get the Whole address FB12, FB supplied by the I register and the 12 supplied by the port, bit 0 is always 0 to tell the port that this is the vector byte. The next byte is the mode, actually bits 6 and 7 contain the mode, bits 0,1,2 and 3 are all set to 1, this tells the port that this is the mode byte, bits 4 and 5 are unused and I have set them to 0. The next byte is the interrupt control byte, bit 7 set enables interrupts, bits 0,1,2 and 3 tell the port that this is the interrupt control byte. The port is now set up to send data to a peripheral which can use it's handshake logic all we do is clear the flag, send our data via an OUT instruction to the port B data register 32H and wait for the flag to go non zero signifying that the data has been taken. Here is a routine to send a byte in mode 0 :

```
SEND:PUSH AF
     LD A,0
     LD (FLAG),A
     POP AF
     OUT (32H),A
POLL:LD A,(FLAG)
     CP 0
     JR Z,POLL
     RET
```

The data is expected to be in A .

Now on to input mode 1. First the handshake sequence, this starts by the peripheral sending a strobe (down then up on the STB line) to tell our PIO that data is ready to be read, and as we have seen this causes an interrupt, the PIO then puts it's ready line down to show it is busy and cannot take any communications, when we finaly read the port this will cause the ready line to rise telling the peripheral that we are ready for a further byte. The interrupt routine is as for mode 0. The set up routine is as follows:

```
PIOIN:PUSH HL
      PUSH AF
      LD HL,INTRPT
      LD (FB12),HL
      LD A,0
      LD (FLAG),A
      OUT(33H),A
      LD A,4FH
      OUT(33H),A
      LD A,87H
      OUT(33H),A
      POP AF
      POP HL
      RET
```

You can see that it's almost the same as for mode 0 exept the mode byte (the second) is of course different,and the FLAG byte is set to 0 so as to be immediately ready before any data can arrive.The routine to read the port is as follows:

```
READ:LD A,(FLAG)
     CP 0
     JR Z,READ
     LD A,0
     LD (FLAG),A
     IN A,(32H)
     RET
```

The routine begins by polling the flag (remember this was initialy cleared by the set up routine), when it goes non zero the flag is cleared and the port read. This concludes the section on modes 0 and 1.

The next section is a brief rundown on mode 2, the bidirectional mode. It is unuseable from the chip in the Einstein due to it requiring all four handshake lines and since not only are port A lines tied up with the printer, they are tampered with to make them work as a centronics port. Mode 2 is designed to work as both input and output from side A of the port it is a combination of mode 0 and mode 1 logic with port A handshake lines used for output control and port B handshake lines used for input control, the logic on them is identical to that of mode 0 and 1 exept that data is only be allowed out when the STB line is low and the peripheral should only output data when it lowers its STB this avoids contention on the data lines.

When using this mode, port B must be set to mode 3 with its mask register set to inhibit all bits (don't panic mode 3 is next), this is because the same interrupt vector is returned for both modes so you should operate mode 3 in polled mode when port A is in mode 2.

Now for mode 3 operation this mode is designed around control applications it doesn't use handshakes (although it does effect the ready lines), it can be set up so that some of the 8 data lines can be input and the others output the port can be written to or read from at any time and interrupts if enabled will be generated after a change in the state of the register has occured,and a logic function (OR or AND)has produced a true result. This mode does take a little more thought to set up than the other modes but it will probably be the mode which most people will find easiest and best to play around with. I shall go through the control bytes to be sent to to the port (&33) and what they do.

The first byte as with the other modes sets the mode and bits 0,1,2 and 3 are set to 1s bits 4 and 5 don't matter and bits 6 and 7 make the mode number making the first byte CF to set up the mode.

The next byte must be the input/output select byte which determines which of the data lines will be input and which will be output a 1 selects input therefore if we send 49H (00101001) lines 0,3 and 5 will be inputs and the rest outputs.

The next byte must be the interrupt vector and as with the other modes bit 0 must be set to 0, this is the same as for the other modes, it provides the lower half of the interrupt vector address.

The next byte is the interrupt control byte bits 0,1,2 and 3 are set to 7 to signify this, bit 7 as with the other modes is interrupt enable (1=enabled) bit 6 is a new one it selects the logic to be used OR (0) or AND (1) if OR is selected then if any of the monitored bits change to the active level (selected by bit 5) an interrupt will be requested. If AND is selected all the monitored bits must change to the active state before an interrupt is requested. Bit 5 selects the active state of the monitored lines (high or low). Bit 4 tells the chip if a mask byte follows (1 = yes).

The next byte (if bit 4 of the previous byte was 1) must be the mask. This masks off any unused bits if the mask is D6 (11010110) then only bits 0,3 and 5 will be monitored.

Setting the interrupt vector is as before at FB12. If for example the active state is set to high and the logic is OR, an interrupt will be generated if any of the monitored lines goes high. Your interrupt routine must read the data from the port and take any action your application requires an example might be a sensor requesting a light on the sensor would be connected to an input line and the light switch to an output line so that you then send a 1 to that bit to turn on the light on.

Really at this point even the manual stops because the next action really depends on your application you may only need to send a byte to the port data register (32h) or you may need to reset the port completely for a new test. If you don't enable the interrupts the data register may be polled this may be easier to start with.

If you intend to use the port, I would at least buffer the lines through something like a simple buffer I.C. good luck D.A.H....

APOLLO11 - MERLIN SOFTWARE £16.95 (Reviewed by Mike Jackson)

The author of this educational package has obviously been burning the midnight oil and you will have a problem to avoid getting similarly hooked by the fascinating results Richard Hamer has achieved!

His stated aim is "to help you to appreciate and understand the fun in using the graphics yourself". The "graphics" referred to are those accessible from Albert's XTAL BASIC and whether you consider yourself novice or expert you will find plenty to amuse and inform you among the component parts. Loading can be simply achieved from BASIC by RUN"MENU" which results in being offered five choices but initially I would recommend you RUN"GRAPHICS" since this gives a few words of explanation on each of the choices before offering you the main menu.

First choice is "APOLLO11" and triggers quarter of an hour or so of the most amazing moving cartoon-style graphics depicting the various stages of the first moon landing. Sounds of differing degrees of realism are also emitted but were evidently a secondary consideration here and we must eagerly await the promissed sound package. Some of the scenes are rather "noddy" while others are remarkably detailed but the opportunity is taken with each scene to illustrate some new point. An accompanying 50 page manual describes how each scene was achieved and comments on problems encountered and how they were resolved.

Next on the menu is "TUITION" which provides illustrated explanations of the BASIC graphics commands such as DRAW, FILL, ELLIPSE, POLY, PRINT@, PLOT, BCOL, TCOL and GCOL. The third menu choice is "SPRITES" which extends the discussion onto SHAPE, SPRITE and MAG graphics commands. Remarkably that list represents about half of the total commands used in producing APOLLO11 so that the novice can take comfort from the fact that by using only a small subset of Albert's commands superb results can be achieved. The manual refers confusingly on Page 2 to inserting Side B of the disk in order to run "TUITION" but the review disk had everything contained on Side A and that advice was ignored. Additional TUITION is given on the implications of using high and low resolution screens and graphic blocks. Fourth on the menu is "STARBURST" which is a somewhat longwinded way of illustrating the implications of the constraint that only one "ink" colour at a time can appear on each 8 pixel line in a graphic block. We are told that for the picture to emerge completely it has to be left running for about two hours but if, like me, you have had enough after 5 minutes I suggest you SHIFT/BREAK into it, add a new line as follows:- 445 FOR Y=0 TO 24:PRINT@0,Y,MUL$(CHR$(255),40);:NEXT Y and restart it with GOTO 440. This will short circuit the two hour wait so that you can see the ultimate effect more or less immediately.

The final choice on the main menu is "SCENERY" which simply sketches in a background scene for you to practise your skill with sprites.

Inevitably there are one or two aspects which could have been improved if the sole had been approaching as nearly as possible continuous cartoon film animation. For example at the start of the "Rendezvous" scene you wait for a few seconds while a segment of the moon is drawn which is interesting if unrealistic!

Again the explanations and advice are not always fully compehensive as when on Page 41 you are advised that you can re-use a command such as RUN 20000 so long as it remains on screen, by moving the cursor back to that line and re-ENTERing (computerwise, not in the Command Module of course!). What you are not told is that Albert is clever enough to recognise such a command (or line of BASIC) even if you have edited it on screen and it occupies more than one line on the screen. It can even be only partly visible on the screen so long as you move the cursor into line and press ENTER.

Pages 22 and 23 of Tatung's describe this technique admirably and we are advised on the first page of the APOLLO11 manual to work through it in conjunction with the User's Manuals. Also, when moving onto the second example on Page 41, the line number 20010 is changed to 20005 for no obvious reason.

The TUITION and SPRITES programs provide a much more entertaining approach to learning about Albert's graphics than simply reading the manuals but although error trapping is used extensively it is not 100%. For example just to be perverse I entered a value of 1234567890 for T when ELLIPSE was being covered and a value of T>1.33 was requested to get a tall ellipse. Although accepted the program crashed with "Qty Error in 5420" when it came to draw the ellipse. Similarly one could criticize the SPRITES program for telling you to type SHAPE144,'1C 1C 08 1C 2A 08 1C 22' when to be absolutely accurate you type SHAPE144,"1C 1C 08 1C 2A 08 1C 22" if you see the difference?

Such criticism is so carping as to illustrate, I hope, what a hard task it has been to find any criticism of a product which combines so well education and fun. Thoroughly recommended as an aquisition for all ages and expertise levels.

LETTERS

Dear Mike,

Just a few questions for you to answer. I've got a 64 column print routine, how would you like me to send it on disk or in printed form?

The P.D. software, does each one fit onto a single side of a disc or is a complete disc needed for each of the P.D. discs? C. Whitehead.

ED. Thanks for the offer I'm sure it will be useful to the members. A point here, anyone sending in articles of any length, say more than one A4 page we really need them on disc, we would appreciate return postage. Typing in articles is very time consuming and prone to mitsakes. Each P.D. disc fills both sides.

Dear Graham,

Thanks for issue 2 of E.M. which arrived this morning. I hope to be contributing something soon. In the meantime you are welcome to publish my name and address for other Einstein users with musical interests and I will be delighted to discuss ideas and projects with them. I will probably not be able to help with problems unless I have encountered and solved them myself, but I don't mind being asked, so long as 'I don't know' is acceptable some of the time. Best wishes and keep up the good work. Dr D.R. Salvage. (Author of 'Music Workshop') 38 Priestman Point, Rainhill Way, London

Dear E.M.

I am a second year student at Manchester Polytech studying for a B.ENG(Hons) degree in Electrical and Electronic Engineering. I have written a piece of software which may be of use to Mathematicians, Engineers and students. I was wondering how to:-

A: make the existance of the program generaly known?

B: how to sell the program?

C: how to make lots of money from it?

Also how do you make it known to members that their subscription is due for renewal? K.N. Sykes

ED. Well this is a good one, your first step may be to advertise in Tatungs Einstein User and of course E.M. As you would be advertising for commercial gain you would be illegible for free advertising in E.M. Our current rates are £25/Full page, £15/Half page or 5p per word. We would also consider marketing members software on a commission basis. Subcription renewal notification goes out with a members final magazine.

Dear E.M,

I purchased my Einie' in '85 and I'm still in love with it. It makes me sad to see something like the Spectrum+3 being released, knowing it will be snapped up but you know what they say... imitation is the sincerest form of flattery. Long live the Einstein and its supporters.

Well that's the patriotic part out of the way so down to the practical bit. I loved the article in E.M. issue 1 about loading other Spectrum software to use with the Speculator. The problem is, how do you read the headers on the programs to find their length etc. The author says Break and list etc to read the header, is this on the Einstein or the Spectrum? After reading the article about ten times I assume the header reading must be done on the Spectrum.

I can do all the bits to convert the information so can someone lend me a Spectrum. Help! I know I live in the South West (ooh arr) but I'm having a job getting any of the original games on the Speculator disc. So far I've only got three, JETPAC, TTL and HUNCHBACK. All the dealers around here seem to think they are no longer available. Help!

The competitions are a great idea, only my last copy of E.M. arrived on July 16th and the closing date was July 19th and I just didn't have time to... just you wait until the next issue.

A.F. Cutler, 42 Oldway, Cudleigh,
Newton Abbot, TQ13 0JA.

ED. We are hoping that since the Spectrum+3 uses 3" discs and CPM it will prove useful to Einstein users. Yes the headers are on the Spectrum tapes and here is a Spectrum program to read headers.

NOTE... THIS IS A SPECTRUM PROGRAM... IT WILL NOT RUN ON ALBERT...
Type this into your Spectrum and save as "HEADREADER"LINE 50

```
50 CLS
100 CLEAR 32499
150 FOR Z=32500 TO 32509: READ Y: POKE Z,Y: NEXT Z
200 DATA 175,55,221,33,08,127,205,86,5,201
250 LET Y=32520: DEF FN Z(A)=PEEK (Y+A)+256*PEEK (Y+A+1)
300 RANDOMISE USR 32500
350 LET X=PEEK Y
400 IF X>3 THEN GO TO 300
450 PRINT "FILENAME: ";
500 FOR Z=Y+1 TO Y+10: PRINT CHR$ PEEK Z;: NEXT Z
550 PRINT : PRINT "TYPE : ";
600 GO SUB 1000+100*X
650 PRINT : PRINT
700 POKE Y,255
750 GO TO 300
1000 PRINT "PROGRAM"
1010 PRINT "TOTAL LENGTH: ";FN Z(11);" BYTES"
1020 PRINT "PROG. LENGTH: ";FN Z(15);" BYTES"
1030 IF FN Z(13)>9999 THEN PRINT "LOAD ONLY": RETURN
1040 PRINT "RUNS FROM LINE ";FN Z(13)
1050 RETURN
1100 PRINT "NUMBER ARRAY"
1110 LET A$="": GO TO 1220
1200 PRINT "CHR... ARRAY"
1210 LET A$="$"
1220 PRINT "ARRAY LENGTH: ";FN Z(11);" BYTES"
1230 LET W=PEEK (Y+14)
1240 PRINT "ORIG. ARRAY NAME: ";CHR$ (64+32*(W/32-INT (W/32)));A$
1250 RETURN
1300 IF FN Z(11)=6912 AND FN Z(13)=16384 THEN PRINT "SCREEN STR$":
     RETURN
1310 PRINT "BYTES"
1320 PRINT "START ADDRESS: ";FN Z(13)
1330 PRINT "LENGTH: ";FN Z(11);" BYTES"
```

MARKET PLACE

TM80 Green Screen monitor plus TK02 80 column card ... £50 Telephone 0827 893593.

48k Centronics Print Buffer, brand new includes power supply £35 Tel. U.K.E.U.G. 0473 49507

MONEYPLAN ACCOUNTING SYSTEM. Simple to use menu driven accounts system for the home/small business user. Requires 80 column card with twin internal disk drives. Cost? Only £15 (includes P+P and user manual). For further details write to: Phil Croshaw, The Avenue, Sark, Via Guernsey, Channel Islands.

FILE HANDLING FROM BASIC. A 50 page book written to compliment the Basic reference manual as supplied with the machine covering disc file handling. Examples are used throughout covering sequential, random and indexed files. £5 inc. P+P from the U.K.E.U.G.

BACK ISSUES. E.A.E.A. newsletters ..... Vol 1,1 - 1,3 ..£3 the set
U.K.E.U.G. newsletters ... Vol 1,1 - 1,12 .£1:25 each
Vol 2,1 - 2,4 ..£1:25 each
Einstein Monthly ..... Vol 1,1 - 1,2 ..£1:25 each

ADVERTISING RATES. Private ads are free, trade advertisements are at 5p per word, £15 per half page or £25 a full page.

SUBSCRIPTION RATE. £15 for 12 issues or £9 for 6 issues.

P.D. SOFTWARE. Available from Jim Ellacott, 39 Parkside, Westcliff-on-sea, Essex. SS0 8PR. If you require a list send Jim a SAE plus 2*13p stamps to cover the cost of photocopying. A full list was published in Vol 1,2.

MEMBERSHIP
BACKCOPIES
ADVERTISING
COMPETITIONS

GRAHAM BETTANY
80 DALES ROAD
IPSWICH
SUFFOLK
IP1 4JR
Tel: 0473 49507

ARTICLES
QUESTIONS
COMMENTS

MIKE SMALLMAN
7 SHEEPEN PLACE
COLCHESTER
ESSEX
CO3 3LD
Tel: 0206 540540

TELEPHONE CALLS BETWEEN 6pm and 9pm ONLY.....PLEASE.

****

SEE OVER FOR SPECIAL OFFER ON SYSTEM 5

Printed by Gipping Press, Needham Market, Ipswich.