Issue No.67

Nov / Dec 1993

ALL MICRO MAGAZINE

(for users of the Einstein and other golden oldies micros)

published bi-monthly by the Steam Computer Society
at Ivy Cottage, Church Road, New Romney, Kent. TN28 8TY
and available only by subscription
(opinions herein are not necessarily those of the publisher)

****

CONTENTS:-

****

IT'S COMPETITION TIME AGAIN, FOLKS !!!

This issue's compo is another wordpuzl from John Briggs, who remains an enthusiastic Einsteiner, despite not actually having owned one for some years! Find the hidden words, circle them on (a copy of) the grid, and send it in, with your name and address. First correct answer out of the hat will get a £10 voucher, and as Xmas is rumoured to be around in these parts, we'll give a runner-up prize to everyone submitting a correct answer.

S	C	I	H	P	A	R	G	N	U	S	F	P
P	N	I	E	T	S	N	I	E	I	O	C	P
R	P	O	R	T	A	M	O	D	I	F	I	C
I	E	L	I	F	T	O	L	P	S	C	E	E
N	E	B	A	T	A	B	L	E	T	C	E	R
T	R	E	B	L	A	T	L	M	X	P	R	A
E	S	U	O	M	D	C	R	D	I	W	G	N
R	C	G	I	D	O	A	I	E	I	P	T	E
P	C	G	D	M	L	S	X	F	S	M	N	E
E	A	L	B	E	C	P	U	S	I	U	E	R
I	G	N	I	S	I	T	I	G	I	D	D	C
R	L	O	T	G	E	R	A	W	T	F	O	S
P	I	N	T	U	R	S	E	U	O	M	R	M

~~ALBERT~~ ~~AMIGA~~ ~~ARM~~ ~~DIGITISING~~
~~DISC~~ ~~DUMP~~ ~~EINSTEIN~~ ~~GRAPHICS~~
~~MODIFICATIONS~~ ~~MOUSE~~ ~~PLOTFILE~~
~~PORT~~ ~~PRINTER~~ ~~RODENT~~ ~~SCREEN~~
~~SOFTWARE~~ ~~TABLE~~ ~~USER~~

****

THE FORGOTTEN SCREEN by Chris Pickles

(An Introduction to the Einstein's Multi-Colour Screen)

Having mentioned the multi-colour screen in a previous article, I thought I might take this opportunity to expand on the subject more fully.

As we all probably know the Einstein has a number of different screen configurations, nearly all of them easily summoned from higher command levels, but for some reason, when the operating system was developed, it was decided to ignore the full capabilities of the graphics chip and not implement the 64 x 48 multi-colour graphics mode. This is a pity, for whatever the reasons for ignoring the capability, I am certain that a screen displaying the full 15 colours without the horizontal restriction of 8 pixels one colour, no matter how low the resolution, would have been useful - especially in the games area.

There are many games that require simple graphics but would benefit from the unrestricted use of colour. Unfortunately, as far as I am aware, there are none available.

In an effort to overcome this I began delving into the inner workings of the machine and succeeded in writing a short Assembly Code routine that produced a small .COM program that called up the multi-colour screen and cycled through the range of colours - but that was as far as it went. Try as I might I couldn't find any way to address the individual pixels and my pleas for help have fallen on deaf ears.

On top of that, in the last two or three years, I have become increasingly involved in PCs, and what little insight I had gained into manipulating the inner workings of Albert appears to have dissipated completely. I cannot even find the listing for the Assembly program mentioned above.

However, for those with a hour or two to spare and an interest in at least taking a cursory look at the 'hidden treasures', I can offer a Machine Code listing and instructions on how to create your own multi-colour screen program.

First you need to boot up the machine and place a non-write-protected disk in whichever drive you are using. Then, at the DOS prompt type MOS <ENTER> to take you (and it) down to the Machine Operating System.

Once at the > prompt type M 0100 to modify the RAM (random access memory) starting at memory location 0100. The figures 0100 will appear on the screen and then the following must be typed in accurately (including spaces), pressing <ENTER> at the end of each line:-

```
0100 CD 0D 01 00 CD 9C 01 CD
0108 2A 01 C3 03 01 21 1E 01
0110 01 80 08 7E D3 09 79 D3
0118 09 0C 23 10 F6 C9 00 EB
0120 0F FF 03 20 00 FF 3E 00
0128 06 FF 21 00 14 F5 CD 4F
0130 01 21 00 0C F1 CD 42 01
0138 3C 11 FF FF CD 59 01 10
0140 E9 C9 F5 D3 08 2B 7C B5
0148 28 03 F1 18 F5 F1 C9 7D
0150 D3 09 7C F6 40 D3 09 C9
0158 C9 1B F5 7B B2 20 02 F1
0160 C9 F1 C3 59 01 21 00 14
0168 3E 22 06 30 F5 CD 4F 01
0170 F1 E5 21 40 00 CD 42 01
0178 E1 11 40 00 19 3C 11 FF
0180 DF CD 59 01 10 E6 C9 21
0188 60 15 CD 4F 01 21 01 00
0190 3E 01 CD 42 01 11 FF FF
0198 CD 59 01 C9 21 40 14 CD
01A0 4F 01 21 00 0C 3E 00 CD
01A8 42 01 CD 87 01 C9 FF FF
01B0 FF .
```

(N.B.:- In this
(listing all
(occurrences of
(1 and 0 are
(numbers,
(not letters

A full stop followed by <ENTER> terminates the input. Having completed the inputting of the machine code enter G at the > prompt to return to the DOS level, then type SAVE 1 MULTICOL.COM <ENTER> and the program will be saved to disk. The 1 represents the number of sectors and there is a single space between SAVE and 1, and between 1 and the program name. Having got the program to disk it is now just a matter of typing in its name for it to run, and (if you've typed it in correctly) away it will go.

You should see a screen divided into 64 vertical stripes with background and foreground colours cycling through the full available range. If you don't it is likely that you made a mistake typing in the machine code, in which case reset the computer, and with the program disk in the drive type LOAD MULTICOL.COM <ENTER> followed by MOS <ENTER> and M 0100 <ENTER>. Now, by pressing ENTER you will be able to step through the machine code checking and correcting as you go. When all is satisfactory enter the full stop before typing G <ENTER> and SAVE1 MULTICOL.COM.

This is a somewhat primitive program, with no means of exiting other than re-setting the computer, but hopefully I might be able to re-discover or re-create the assembly listing, and (if the missing links turn up) even develop it further. If not, perhaps some other poor benighted soul might discover some way of exploiting the 'forgotten screen' to the benefit of all Einsteiners, everywhere.

FIG4

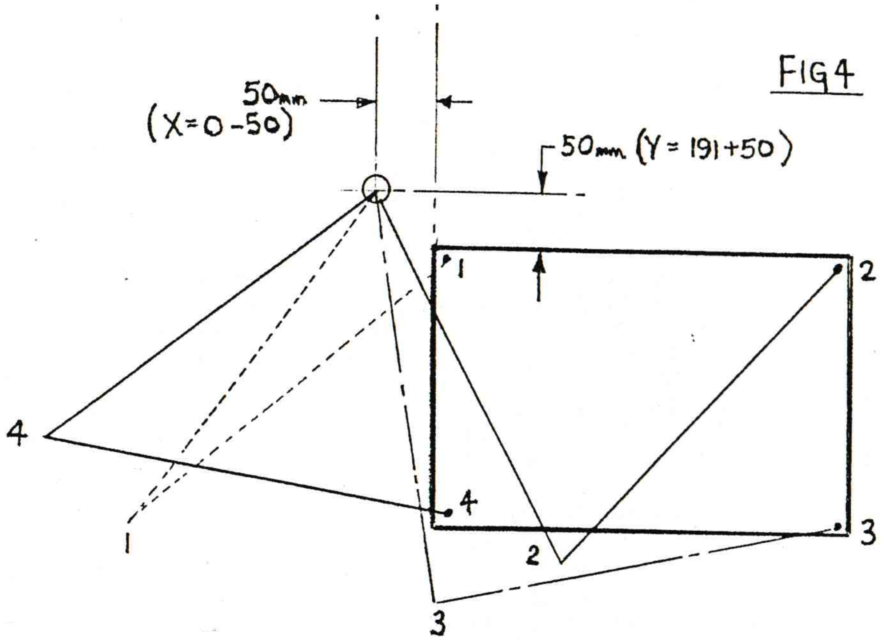


FIG 5A

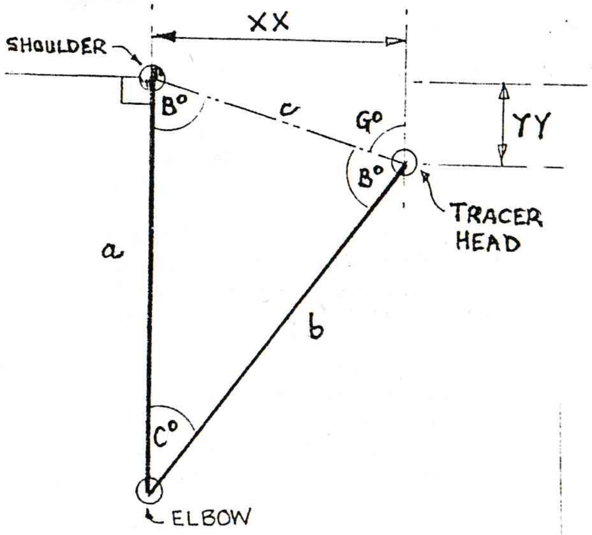
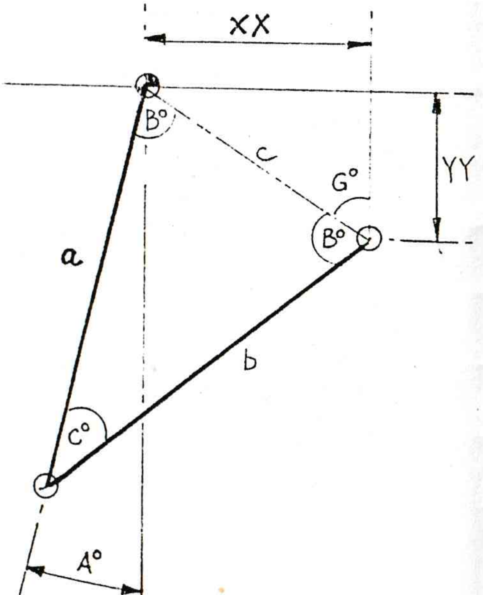


FIG 5B



GRAPHICALLY SKETCHING (Dave Arts continues his series)

Chapter 3 - The Graphic Arm. (Part 2)

DYNAMIC RANGE OF THE ARM

The arm can comfortably span 255mm. horizontally and 192mm. vertically. This means that each square mm. can be translated to a pixel position on the screen. The optimum position of the arm with respect to the working area is shown in FIG.4.

The sweep angle (SPAN) of each potentiometer is approximately 75 degrees, which is a little over a third of the effective track.

Initially I had misgivings about the resolution that the graphic arm could provide. In actual practice it proved to be quite acceptable within the constraints of the limited resolution of Albert's VDU display.

PRINCIPLE OF OPERATION

The arms of the Graphic Arm form the two sides of an isosceles triangle, the Base being the distance between the Tracer Head and the main pivot. This of course will be different for differing angles between the arms. Once the Base length has been found, it is an easy task to find its position from the fixed pivot by simple geometry. Consider FIG 5A:

From the cosine rule: c*c=2*a*b-2*a*b*cosC

If we assume angle C is 30 degrees and a=b=305mm., substituting in the above formula we have:

c*c=2*(305*305)-2*(305*305)*0.866 c being 157.88

Since the triangle is an isosceles with side a = to side b, then the two opposite angles to C are equal (B degrees).

ANGLE B= (180-30)/2 = 75 degrees
ANGLE G=B (corresponding angle)

Therefore XX=c*SIN 75 degrees
YY=c*COS 75 degrees

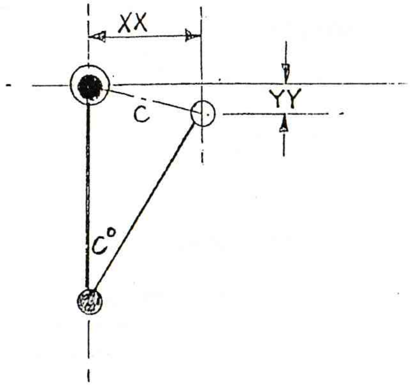
As the elbow moves so the values of XX and YY will change in accordance with the angle C.

This is OK if side a is always perpendicular to the horizontal, but this is not always the case as the shoulder pivot can also move and the situation can occur as in FIG.5B

Since the shoulder arm has moved through 10 degrees (A degrees), G degrees must have decreased by 10 degrees.

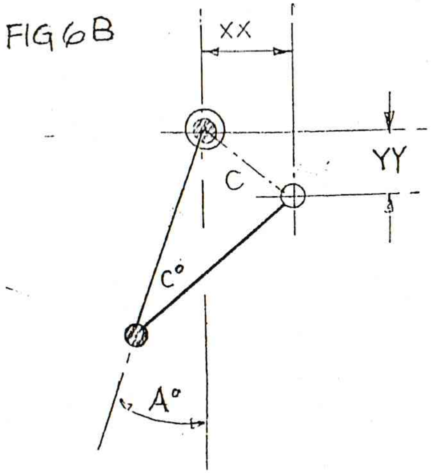
However note that B degrees is still 75 degrees as the elbow joint has not moved (C = 30 degrees as before).
From this we can see G=B-A (G=B only when A=0)

FIG6A



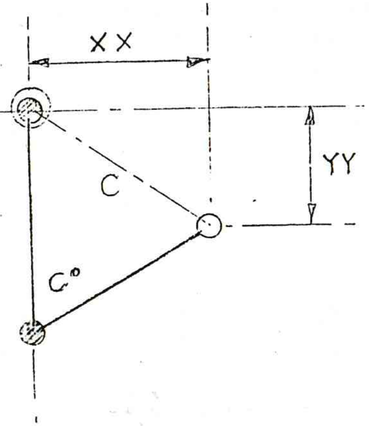
```
VALUE FOR CO IN DEGREES 30
VALUE FOR AO IN DEGREES 0
  C=  157.88
 XX=  152.5
 YY=  40.8622
```

FIG6B



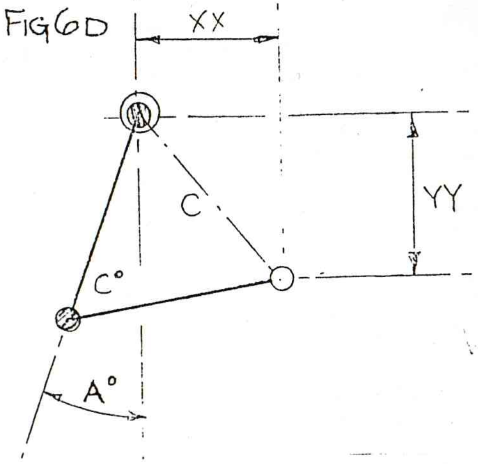
```
VALUE FOR CO IN DEGREES 30
VALUE FOR AO IN DEGREES 30
  C=  157.88
 XX=  111.638
 YY=  111.638
Ready
```

Fig6c



```
VALUE FOR CO IN DEGREES 60
VALUE FOR AO IN DEGREES 0
  C=  305
 XX=  264.138
 YY=  152.5
```

FiG6D



```
VALUE FOR CO IN DEGREES 60
VALUE FOR AO IN DEGREES 30
  C=  305
 XX=  152.5
 YY=  264.138
```

It follows then that the length of side c can always be derived from the Cosine Rule as long as we know C degrees.

Likewise co-ordinates XX and YY can always be derived from the angle C using simple geometry as long as we know the angle of the arm from the perpendicular A degrees:

```
XX=c*SIN(B-A)
YY=c*COS(B-A)
```

ALBERT's PROGRAM

The following program calculates the co-ordinates XX and YY from inputted values for Angles C and A. From now on we will represent angles by the suffix O (letter O) i.e. angle C= CO and angle A= AO, angle B=BO etc.

In the listing c*c is represented by the variable z. Upon finding this value we can derive c by using c=SQR(z)

Notice also that if we input angles in degrees we must use the form RAD to prefix the angle.

```
  5 REM "GEOM1"
 10 INPUT "VALUE FOR CO IN DEGREES ";CO
 20 INPUT "VALUE FOR AO IN DEGREES ";AO
 30 LET Z=2*(305*305)-2*(305*305)*COS(RAD(CO))
 40 C=SQR(Z)
 50 LET BO=(180-CO)/2
 60 XX=C*SIN(RAD(BO-AO))
 70 YY=C*COS(RAD(BO-AO))
 80 PRINT "  C= ";C
 90 PRINT "  XX= ";XX:PRINT "  YY= ";YY
100 END
```

Figures 6A to 6D represent 4 different attitudes of the graphic arm with varying angles for AO and CO. When the program is run with these values the answers for base length c and co-ordinates XX and YY are calculated. GEOM1 derived from the Cosine Rule is at the heart of the Graphic Arm project.

Of course XX and YY are co-ordinates from the pivot and NOT from the SCREEN ORIGIN. What we have to do is determine this point in relation to the Screen Origin.

I have decided after much thought to place my Pivot at absolute co-ordinates X=-50 andY=241. (See FIG.4 again.) Therefore in order for us to establish X we merely subtract 50 from XX, and in order for us to establish Y we subtract YY from 241 (191+50)

MODELLING THE ARM

The program "MODEL" will allow us to plot points on the screen which simulate inputs from the Graphic Arm without it even being connected. Enter the program and allow it to run. The program mathematically models what would appear on the screen if we opened angle C by 2 degrees each time starting at 8 and finishing at 80 degrees, & repeats this for angles of A between -20 and 60 degrees, again in steps of 2 degrees. (perpendicular is 0 degrees). It is interesting to see that we can easily determine where the pivot position lies as the picture builds up.

```
  5 REM "MODEL"
 10 CLS
 20 DRAW0,0 TO 0,191 TO 255,191 TO 255,0 TO 0,0
 30 FOR AO=-20 TO 60 STEP2
 40 FOR CO= 8 TO 80 STEP2
 50 LETZ=2*(305*305)-2*(305*305)*COS(RAD(CO))
 60 C=SQR(Z)
 70 LET BO=(180-CO)/2
 80 XX=C*SIN(RAD(BO-AO))
 90 YY=C*COS(RAD(BO-AO))
100 LETX=XX-50
110 LETY=241-YY
120 PLOT X,Y
130 NEXT CO
140 NEXT AO
150 END
```

****

SHOPPING LIST:- Dave says he's sorry that this was missed from Part 1 of his article due to his being so intent on checking the technical content and the details of construction to eliminate any errors before you started.

2 off. POTENTIOMETER, 5K linear track with splined shaft.
1 off. RESISTOR, half watt, 4.7K.
2 off. SWITCH, momentary push-button (push on, release off).
1 off. DIN PLUG, 7-pin, to connect to Analogue 1 Socket on RH side of Albert (ADC 0 and ADC 1)
1 off. POWER SUPPLY UNIT, 13.8 volt (old CB unit, see text)
1 off. TOY SKETCHAGRAPH, Mk.2 preferred, but older Mk.1 OK.
3 off. INSTRUMENT KNOBS, 2 to fit the potentiometers, plus 1 to fit the pointer shaft.
OTHER ITEMS:- Araldite, solder, terminal wire (16 s.w.g.), baseboard (approx. 720mm. * 320 mm.), small piece of Veroboard (approx. 125mm. * 10 mm.), cut-down knitting needle for pointer shaft, size to suit instrument knob.

SCHIZOSCRENIA by Ted Cawkwell

I had a thought the other day - it does happen occasionally! - that it was a sad reflection on modern life that one needs to be reminded about things which should really be quite easy to remember. In the days before TV (giving my age away) we used to play cards, and it was no great problem to remember what had happened to some 20 or 25 picture cards during a game of whist. So it does seem trivial to have to keep in mind that e.g. "S" saves a record, "C" changes a record, "R" reads one, etc., but it is certainly easier if these keys are on permanent show at the bottom of the screen.

This thought was in fact prompted by a program I was developing which would clearly benefit from a permanent bottom line. The question was.... how to do it in BASIC on a 40 column screen?

My first attempt was to use the POS command, as POS(2) holds the current row position of the cursor. As the inputs move down the screen the program cycles through a line like
'100 if POS(2)=19 THEN CLS:GOTO 200'
where 200 reprints the bottom line and starts again at the top of the screen. This works OK as long as you remember never to go right to the end of row 23 (if you do, you force a line feed), and to reset the PRINT position to the top. A working example of this is "UPSCROLL" (Listing 1).

The next attempt was using SCRN$(n); try the following:-

```
 10 FOR J=0TO 19:PRINT@3,J;J;"SCROLL TEST":NEXT
 20 PRINT@3,23;"BOTTOM LINE";@0,20;
 30 Y$=INCH$:PRINT" "
 40 IF POS(2)=>21 THEN GOSUB 100
 50 GOTO 30
100 FOR J=0 TO 19:PRINT@0,J;SCRN$(J+1):NEXT
110 RETURN
```

Note that lines at the top of the screen may also be retained. Try changing J in line 100 to 7 to 19. This is a fairly sluggish method though.

So far the interruption and flicking on and off is not very elegant. Clearly some sort of selective scroll routine is required and this means the dreaded machine code. Attempts at writing a suitable program brought the predicted disasters and the workings of the VDP remain a mystery to me, but I then came across the Screen save routine in the Compendium, and saw that this could be modified to scroll the screen by saving a number of rows to RAM and then loading them back one row higher.

The result of my modifications is SCROLL20.OBJ (Listing 2), and it works OK in a BASIC routine like the following:-

```
10 CLS:CLEAR &A000
20 LOAD "SCROLL20.OBJ"
30 REM Your Print to Screen routine
40 REM which cycles every line through
50 REM a line like:-
60 IF POS(2)=>19 THEN CALL &A000:PRINT@
   0,18;
70 GOTO 30
```

This seems to work all right as long as line 19 is all spaces and text is added at the bottom and scrolls off the top. The routine works on the VDP by loading Lines 1 to 19 into RAM and then back to the VDP as Lines 0 to 18 which then appear on the screen. It is very quick. A simple working example is Listing 3.

The first time I tried it was not a success. I filled the screen with numbered rows of graphics and CALLed the machine code. To my amazement (but not surprise!) the lines of text started jumbling up into three sections down the screen and after a few passes it was chaos. Back to the VDP book!
It turns out that the VDP does in fact set up the screen in three sections of 8 lines by 40 characters, and some lines were jumping from one block to another by some (to me) obscure mechanism. The secret seems to be in making sure that the last line of the scrolled section is blank.

I was eventually able to make use of this fact in Listing 4 which divides the screen into 3 work areas and a permanent bottom line. You can switch from one to another of the work spaces, two of 6 lines and one of 4 lines, by inputting the parallel line symbol next to BREAK and then selecting T,M or B for top, middle or bottom. The program as given lets you type into the work areas as required but it is quite easy to input data from an array or even straight from a file.

I have tried to combine Listing 4 with the use of WIDTH and ZONE to give some sort of windowing without success, possibly the use of PRINT@ might be the answer. SCHIZO is quite entertaining to play around with but I must confess that so far I have not thought of a proper use for it! Perhaps a reader can come up with something?

About the only thing that immediately comes to mind is comparing the contents of 2 or three Arrays, one in each section and I have tried this with success. It is merely a matter of arranging for branching to separate routines at the GOSUB 500 and then RETURNing to Line 50.

I note that an improved Screen Save including the colour table has been published in Einstein Monthly and I should think it would be convertible in the same way but as yet I have not got around to dissembling it to see. The one used here is the BASIC version and of course, no longer saves the screen to disk.

LISTING 1: UPSCROLL.XBS

```
  5 REM        ==UPSCROLL==
 10 DIM A$(25):X=0:CLS
 20 FORJ=0TO24:A$(J)="*******":NEXT
 30 PRINT@4,23;"ANY KEY FOR NEXT LINE";:PRINT@0,0
 35 PRINTX,A$(X):X=X+1
 40 Y$=INCH$:IF Y$="" THEN 50
 50 IF POS(2)=>19 THEN GOSUB 100
 60 GOTO 35
100 PRINT@0,23;CHR$(10);CHR$(11)
110 PRINT@4,23;"ANY KEY FOR NEXT
LINE";@0,22;SPC(40):PRINT@0,18;
120 RETURN
```

LISTING 2: SCROLL20.OBJ

```
0100 21 00 B0 22 00 D0 3E 00 !.0".P>.
0108 D3 09 3E 01 E6 3F D3 09 S.>.f?S.
0110 06 14 C5 06 00 DB 08 2A ..E..[.*
0118 00 D0 77 23 22 00 D0 10 .Pw£".P.
0120 F4 C1 10 EE 00 11 00 B0 tA.n...0
0128 21 00 00 06 14 C5 06 00 !....E..
0130 E5 D5 7D D3 09 7C F6 40 eU}S.|v@
0138 D3 09 D1 1A D3 08 E1 23 S.Q.S.a£
0140 13 10 ED C1 10 E7 C9 EE ..mA.gIn
0148 C9 11 00 B0 21 00 00 06 I..0!...
0150 18 C5 06 00 E5 D5 7D D3 .E..eU}S
0158 09 7C F6 40 D3 09 D1 1A .|v@S.Q.
0160 D3 08 E1 23 13 10 ED C1 S.a£..mA
0168 10 E7 11 00 C8 21 00 20 .g..H!.
0170 06 18 C5 06 00 E5 D5 7D ..E..eU}
0178 D3 09 7C F6 40 D3 09 D1 S.|v@S.Q
0180 1A D3 08 E1 23 13 10 ED .S.a£..m
0188 C1 10 E7 C9 A.
```

LISTING 3: SCROTEST.XBS

```
    10 CLS:CLEAR &A000
    20 LOAD"SCROLL20.OBJ"
    30 FOR  J=0TO19:PRINT@3,J;J;"SCROLL TEST":NEXT:PRINT
  SPC(40):PRINT@3,22;"SCROLL TEST":PRINT@3,19:
    40 CALL &A000
    45 Y$=INCH$:IF Y$="Q" THEN END
    50 PRINT "NEXT";@3,19;:GOTO 40
```

LISTING 4: SCHIZO.XBS

```
    10 REM ******SCHIZO*******
    20 CLEAR &A000:LOAD"SCROLL20.OBJ":L=6:B=6:M=1:C(1)=2
:C(2)=2:C(3)=2
    25 CLS:PRINT@0,7;MUL$("4",40);@0,15;MUL$("4",40):REM
 GRAPH 4 IN QUOTES
    30 PRINT@4,23;CHR$(23);"¦change:Top:Middle:Bottom:Quit"
;CHR$(23):GOTO 200
    40 GOSUB 500:IF C(M)=0 THEN C(M)=40
    50 Y$=INCH$:IFY$="" THEN 50
    60 IF Y$="T"OR Y$="t" THEN 200
    70 IF Y$="M"OR Y$="m" THEN 300
    80 IF Y$="B"OR Y$="b" THEN 400
    90 IF Y$="Q"OR Y$="q" THEN END
   100 GOTO 50
   200 POKE&A00B,&01:POKE&A011,&07:POKE&A02A,&00:POKE&A02C
,&07
   210 M=1:L=6:B=6:PRINT@0,L+1;SPC(40);@C(M)-1,L-1:GOTO
 40
   300 POKE&A00B,&09:POKE&A011,&07:POKE&A02A,&08:POKE
&A02C,&07
   310 M=2:L=14:B=6:PRINT@0,L+1;SPC(40);@C(M)-1,L-1:GOTO 40
   400 POKE&A00B,&11:POKE&A011,&06:POKE&A02A,&10:POKE&A02C,
&06
   410 M=3:L=20:B=4:PRINT@0,L+1;SPC(40);@C(M)-1,L-1:GOTO 40
   500 A$=INCH$:PRINT A$;
   510 IF A$="¦" THEN C(M)=POS(1):RETURN
   520 IF POS(2)=>L THEN PRINT@0,L-B;SPC(40):CALL &A000:
PRINT@0,L-1;SPC(40);@0,L-1
   530 GOTO 500
```

****

XBAS 5-LINER ***** "JOSEPH" by David Williams

You'll need colour to appreciate this fully, but it's pretty in mono too.

```
    10 REM JOSEPH by David Williams
    20 BCOL1:TCOL15,0:CLS40:FORA=1TO60
    30 C=RND(14)+2:GCOLC,1:ELLIPSE3*A,1.8*A,A
    40 NEXT:PRINT@2,3;"AGAIN (Y/N):";
    50 A=INCH:IFA=89THEN20
```

ODDS AND ENDS DEPARTMENT

Graham Bettany and Jim Ellacott have both given up on you and your machine as a lost cause this year, so we're very relieved (& delighted) that our hunch was right and that all you really needed was a mixture of encouragement and TLC. (For anyone not familiar with the term TLC, it's simply a mixture of "Mummy kiss it better" and a good cuddle.)

When we took over in August there was virtually no input to the magazine, subscribers were lapsing at every opportunity, printing costs massively exceeded subscription income, and the user group was defunct. We've revived the user group, cut printing costs to the bone whilst trying to improve the actual magazine content, made a start on keeping existing members AND on recruiting new ones, and you've supported our efforts magnificently by renewing or extending your membership and also by providing input to the magazine.

Thanks to you the situation is now no longer an untenable one. Short-term it is now reasonably sustainable, though the number of Einsteins - and thus of Einsteiners - will inevitably continue to fall (since no new ones are being made), and keeping the user group viable long-term will almost inevitably involve supporting other golden-oldies machines and their users as well as the Einstein.

Many thanks to each one of you - new Einsteiners and old - who responded to our request for information and input. Some of you provided conventional articles, and some of these appear in this issue. Other than very short items we like to make a series of mini-articles of them where we can, to give maximum spread of coverage, proficiency level and subject interest in each issue. Many of you sent letters in about yourselves and your computing. In almost every case these are both interesting and informative, and would either form the basis of a future article, or are fit to publish, just as they are. If you've written in since we took over, could we PLEASE have a copy on 3" disk? We'd love to share what you've told us with your fellow-Einsteiners, but Fred the office cat still has sore paws after keying in the whole of our first "utility" issue single-pawed, and no-one else in the editorial office knows how to work a keyboard!

Whilst you've responded magnificently to our appeal to you to help us save your magazine and rebuild your user group, this restored level of input from you DOES have to be sustained if our long-term aim is to be realised, of working with you to provide the mutual self-support that will enable the Einstein to continue to productively earn its keep well into the next millennium.

Many of you have asked what you can do to help us to help you. The most obvious answer is that you can provide input to the magazine. Your combined knowledge, experience and insight into Einsteining (and computing generally) make up an incredibly valuable resource if it can be aggregated, and even the most humble and inexperienced user holds a very valuable piece of this treasure. We'd love you to share it with the rest of us. If you don't feel up to writing an article, just write a letter. Or send us a query or problem. We'll take it from there. But please DO send us a copy on 3" disk if it's more than three or four lines long. Don't worry, you won't lose the disk. We'll make a copy and send yours back to you. Also, as a bonus, we credit you with the issue that your contribution appears in, or (if you prefer) we'll send you a new disk instead, as well as returning your old one.

We do prefer a print-out of what you're sending us, but don't worry about this if you don't have a printer - just a short note telling us what's on the disk, and what word-processor you used to create it, is just fine. In fact, you can write a letter or article for us even if you have no printer at all, by sending it on disk instead.

We have problems with some other formats, so we do prefer contributions on 3" Einstein disk. A copy in your word-processor's native format please, and ALSO a copy in ASCII format if you can. Please DON'T put in formatting codes such as underlines, italics, etc. It causes us no end of trouble. For the same reason please DON'T right-justify your text if your word-processor allows ragged-right text. Simply put headings in BLOCK CAPITALS (on a new line?), and indent new paras one or two columns. In emergency you can always use your BASIC editor to create a text file by putting REM at the start of each line, and if you don't own a word-processor at all, we should soon be able to supply a commercial-quality one from the Einstein software library.

Which brings us to another point. Jim Ellacott set up the EAEA software library, and absorbed the existing UKEUG software when the two groups merged. The resulting UKEUG pd library was always presented as an integral part of the user group, but Jim says that he created it at his own expense and with his own labour, and it has always been his own personal property.

Having given up on you himself, he agreed to allow us to use his duplicate disk set instead, but we can't read them on our disk drives. Instead he offered to sell out to us so he can concentrate on his IBM shareware. He'd told us that you'd already repaid in copying charges what it had cost him to set up the library, but we thought it prudent to pay the hundreds of pounds he was asking to buy him out without quibbling, since second-hand 3" disks have a ready market as data disks for users of the PCW & other machines!

However, this has created a severe cash-flow problem for us, as we'd just spent all the available spare cash on attending the Stafford Show, in getting in a bulk supply of 3" disks, and in buying the last legitimate copy of dBASE II from B&H in hopes of negotiating a non-profit distribution licence to keep it available to you.

We'd therefore appreciate any donations of cash (or software,) (or any additional membership renewals or extensions,) that you can afford, to help us out of this temporary cash-flow crisis that we couldn't very well have foreseen or avoided. We realise that with Xmas upon us you may very well have your own temporary cash-flow crisis, so we'll hold the 1993 membership subscription rates - existing members only - until the next magazine is issued, i.e. £16 for 12 issues/2 years, or £24 for 18 issues/3 years.

You can also help us to help you in two other ways. There are lots of people out there who have acquired second-user Einsteins. Often the system has been sold off in bits and pieces to get the best price, so they may have little or no computer knowledge, an incomplete system, and no software or manuals. Often they are struggling against impossible odds, or have given up and parked their 256 or Albert in the loft. We need each and every one of you to seek out ways in your community of contacting these invisible people, and letting them know that we exist to provide support for them.

Also we need to be visible in the contact point or editorial sections of the computer magazines, and to appear regularly in the free ads of PC Mart, Micro Mart, & etc. If anyone's willing to organise this side of things, it would take a lot of pressure off your "Jack-of-all-trades" secretary/editor.

Similarly, we didn't make any money out of the Stafford Show, but it did provide a valuable opportunity to make contacts and become visible. We can't give you the traditional report on it, since we were short-staffed and didn't manage to get more than ten feet from the UKEUG stand in any direction before having to hurry back to deal with enquiries. We'd very much welcome help at future shows (so we can all get to see the show), or any offers to run stands at shows for the group. Other groups have developed specialist teams to do this very successfully, either locally or wider afield. Someone with a van (or a caravan or camper-van) would have tremendous scope for this; but a car, bike or moped - or even a suitcase! - would be a good start.

APOLOGIES TO ALL NOVICE EINSTEINERS and any others without manuals - or who can't understand them. We'd intended to include Part 1 of A Beginner's Guide To The Einstein by A.C. McROBBIE in this issue, but cash and printing constraints won't let us increase the number of pages at present, and we've had to hold it over to the next issue.

Issue No.67

Nov / Dec 1993

ALL MICRO MAGAZINE

BACKPAGE INFO

ALL MICRO MAGAZINE is published bi-monthly by subscription only. One copy of each issue is mailed free of charge to current members of the U.K. EINSTEIN USER GROUP (UKEUG).

Contributions, subscriptions and enquiries should be addressed to Ivy Cottage, Church Road, New Romney, Kent, TN28 8TY. Phone enquiries cannot usually be dealt with.

Membership of the user group is available at £10 per year (6 issues), with discounted rates of £18 per 2 years (12 issues) and £27 per 3 years (18 issues). Members residing outside the UK pay slightly more to cover extra postage costs. An information pack / membership form is available.

The magazine and user group are run in their spare time by unpaid enthusiasts on a very tight budget. If you require a reply please include a S.A.E., AS WE MAY NOT BE ABLE TO AFFORD TO PAY THE POSTAGE BEFORE MAILING YOUR REPLY!

MAGAZINE BACK NUMBERS are available (post-free) at £1 each for odd copies, £5 for 6 copies, or £20 for 30 copies.
The following issues are currently available:-
EINSTEIN MONTHLY volume 1: 5,6,7,8,9,10,11,12
EINSTEIN MONTHLY volume 2: 1,2,3,4,5,6,7,8,9,10,11,12
EINSTEIN MONTHLY volume 3: 1,2
ALTERNATIVE MICRO NEWS volume 1: 1,2,3,4,5
ALL MICRO NEWS volume 1: 1,2,3,4,5,6,7,8,9,10,11,12
ALL MICRO NEWS volume 2: 1
ALL MICRO NEWS: #65,#66

EINSTEIN USER MAGAZINE: B&H Computers at Halifax publish this magazine. Their Sept '93 Flyer said £1 each or £10 for "THE FULL SET of 19". Check with them whether this means that they've discontinued publication -- it's hard to tell with a "quarterly" magazine that only comes out once in two years!
We do have a few copies of most issues - at our usual prices - if you want to include them in an order.

EINSTEIN SOFTWARE LIBRARY: This is in the process of being integrated into the user group and re-organised, so only very limited service is currently available. See next issue.

ALL BANK DRAFTS, CHEQUES, POSTAL ORDERS, etc., PAYABLE TO EINSTEIN USER GROUP PLEASE.

Orders from non-members will be accepted ONLY on the clear understanding that the first item supplied will be subject to a SURCHARGE of £10, in payment of 1 year's membership.