ØRIC USER MONTHLY

with Alternative Micros

Number 99

November 1995

Keeping the Oric alive

Edited and Distributed by Dave Dick, 65 Barnard Crescent, Aylesbury, Bucks HP21 9PW

EDITORIAL

HELLO AND HERE WE GO AGAIN,
TO ANOTHER ISSUE OF ORIC USER MONTHLY.

AS SUMMER FADES AND THE NIGHTS DRAW IN, IT'S TIME TO SETTLE DOWN WITH YOUR TRUSTY OLD ORIC.

WITH THIS ISSUE YOU WILL RECIEVE THE LAST OF THE RAFFLE TICKETS FOR BRIAN KIDD'S GRAND RAFFLE. THE DRAW WILL TAKE PLACE SHORTLY AND WINNING NUMBERS WILL BE REVEALED IN THE NEXT OUM.

I HAVE JUST BEGUN TO LEARN TO USE 'ENVISION' - DESKTOP PUBLISHER FOR MY PC-XT. EXPECT SOME PROFESSIONAL LOOKING INPUT FROM ME IN ABOUT 3 YEARS TIME!

NOW LET'S GET TO THE INDEX.

INDEX

FOR SALE

CHRISTOPHER PHILLIPS HAS CERTAIN ORIC GOODIES FOR SALE, NAMELY,:

ORIC MCP40 PLOTTER/PRINTER
ORIC-1 - 48K
3" MICRODISC SYSTEM

OFFERS TO CHRIS AT:
19 ALEXANDRA PARK, SCARBOROUGH, NORTH YORKSHIRE.
YO12 5JN (TEL: 0723 368475)

NEWS NEWS NEWS

CEODISC 22

THE 22nd DISC FROM CLUB EUROPE ORIC ARRIVED RECENTLY, AND CONTAINED SOFTWARE FROM THE 'INTERNET'.

'SLIME' IS A FUN GAME.

'SOKOBAC' IS THE ANDRE CHERAMY GAME SHOWN AT THE AYLESBURY MEET, WITH EACH SCREEN DEDICATED TO A CEO MEMBER. UNFORTUNATELY,THE INSTRUCTIONS ARE IN FRENCH, BUT THE RULES ARE EASY ENOUGH TO GET THE HANG OF.

USING THE CURSOR KEYS TO CONTROL YOUR LITTLE MAN, YOU HAVE TO MOVE THE PACKETS INTO THE SHADED AREAS, WITHOUT GETTING BLOCKED IN BY THE WALLS.

'J' = PLAY
'S' = NEXT LEVEL
'T' = CHOOSE THE LEVEL

THIS IS DEFINITELY ONE TO GET YOU THINKING.

AS WE GO TO PRESS, I HAVE MANAGED TO CLEAR LEVELS 2 & 3, AND AM ALL READY TO HAVE A CRACK AT LEVEL 4 (CALLED 'POOR LAURENT'). IT'S ONLY A PITY THAT I CAN'T CLEAR LEVEL ONE.

ALSO ON THE DISC IS A 'HIRES' PICTURE (007!), AND THE 'LINKER' DATABASE PROGRAM. THE 'LINKER' PROGRAM INTERESTED ME, BUT EACH TIME THAT I GOT TO ABOUT THE TENTH LINE OF INPUT I GOT A 'BAD SUBSCRIPT ERROR' IN LINE 3200 - SORT IT OUT FELLOWS!

===

THE ONE HUNDREDTH ISSUE

NEXT MONTH WILL SEE THE APPEARANCE OF OUM No.100.

WHAT A LANDMARK! WHO WOULD OF THOUGHT THAT THE ORIC MAGAZINE STARTED BY ROBERT COOK OVER 8 YEARS AGO, WOULD STILL BE AROUND, AND IN FACT GOING FROM STRENGTH TO STRENGTH?

TO CELEBRATE,WE ARE PUTTING TOGETHER A VERY SPECIAL ISSUE. WE NEED YOUR INPUT. ANY ODD LITTLE ORIC RELATED STORIES WILL DO.

WE ALREADY HAVE A NICE LITTLE ARTICLE FROM SIMON ULLYATT. NOW IT IS YOUR TURN.

GET THOSE ARTICLES IN BY NOVEMBER 20th PLEASE.

===

EINSTEIN

THOSE NICE PEOPLE WHO PUBLISH THE 'EINSTEIN MAGAZINE & ALL MICRO NEWS', SENT ME THEIR LATEST BI-MONTHLY ISSUE.

WHAT A PITY THAT I COULD HARDLY READ IT! THEY ARE STILL HAVING ENORMOUS PROBLEMS WITH DUPLICATION. THE RE-PRODUCTION REALLY IS HORRENDOUS. IT'S A GREAT SHAME, AS THERE IS OBVIUSLY A LOT OF THOUGHT PUT INTO THE MAGAZINE,WITH PLENTY OF VARIETY IN THE ARTICLES.

LET'S HOPE THAT THEY CAN GET THE PRINTING SORTED OUT, AS 8-BIT ENTHUSIASTS NEED ALL THE INFO THAT THEY CAN GET.

MAGNETIX *(HERE AT LAST !!!)*

It has taken some time but now, at long last, Jon Bristows eagerly awaited new game ***Magnetix*** is available, (for a measly sum so you've got no excuses for not sending off your order to the eagerly awaiting Dave - now have you ???)

What's it all about then, what's all the fuss about and why has it taken all this time to write ?? Read on and I might get round to revealing these and other mind - boggling questions that have been bugging you recently. (Like whatever happened to clutch pencils?)

SCENARIO. You're in space, you've got a job to do and the bad guys are after you. (OK, so you can't be original ***all*** the time !) There you are in stuck within a maze where you have to move power blocks together to form a larger block of 5 x 6. Happily there are vortexes, (or should that be vortices?), to whisk you from one part of the maze to another. Various nasties charge about as you battle against time to complete your task.

GAMEPLAY AND STUFF. The above scenario may not sound too exciting to some of you, but try to imagine playing Tetris when someone's bashing on your head, you're late for work/school and the phone's ringing. This games a bit like that, (but in a nice way)!

You have the choice of two types of key sets or joystick interfaces to choose from to play the game and so your craft is easy to control. As you move around the screen you will soon get used to jumping into a vortex, (often necessary), to see the screen smoothly scroll across to another vortex which you pop out of. This a game which is dead easy to play. Your only trouble is how to play the game effecitvely so that you complete your task of sticking those power blocks together. This is very easy on the first level but later levels will have you scratching your head and muttering under your breath. (A useful feature of this game is that it allows you to start at whatever level you wish - provided you have the password given on completing the previous screen!) This is certainly not a shoot 'em up game but it does have some elements of that as you blast the aliens. For this you will need to make sure to collect the charge crystals hidden around the maze. Firing results in different effects ranging from zapping baddies to protecting your ship and you need to be careful how you use the four options if you want to stay alive!

GRAPHICS If you thought the Oric was slow, wait 'til you see this baby move. The screen scrolls around whilst the foreground and background move at different rates, giving a 3D effect whilst various aliens tear past you at break neck speed - *AMAZING.* We haven't mentioned the intro yet. The TWILYTE logo appears with a shimmering effect and then your confronted with a screen of animated stars. These scroll away to give the main title screen with more twinkling stars dashing across the screen. Various options can be chosen, but pressing a key doesn't cause anything to happen for a while as the stars scroll off screen out the way. Leave the title screen and you will get the Hi-scores displayed. It's all very impressive stuff when you sit back and think that it is an Oric in front of you. Some of the graphics within the game are amazing when you try and think how they could be achieved without real sprites.

SOUND Throughout the intro screen you get to hear the 25 minute (!) long Sonix composed music. This varies as it goes along and changes in style. If one bit isn't to your taste then another bit will be. What ever your musical preferences are this is a great example of what can be achieved using Sonix and should sell a few more copies of another Bristow masterpiece.

Sound effects in the game are numerous and excellent with a pair of headphones strapped to your head. Everything seems to be animated and has its own sound.

CONCLUSION This is without doubt one of the best pieces of programming to have appeared on the Oric. Everything - sound, graphics, gameplay , is all well thought out and, if you look at that screen and realise what's happening, will understand why it must have taken so long to program.

I'm not going to tell you whether or not I think this game is the best piece of Oric software. I'm not going to tell you if I think it has the best graphics or the best sound. All these things are for you to decide - and there is only one way for you to find out ......

FOOTNOTE. I won my copy of Magnetix in a competition. Ha ha ha ha ha ha ha ha ha !!!!

What you have done is to place a blue ink attribute to the left of the cursor position, where text appears. In fact, you can use PLOT to place strings (letters) or numerical values (attributes, letters or graphics characters) anywhere on the text screen, and also in inverse colours (but more of that later). Try typing PLOT 0,0,"just a line of text", and your coloured text will disappear and be replaced by that message.

One of the limitations of your machine, however, is that you cannot have a line of text with each letter in a different colour to your choosing, you have to leave a space between the letters to hold the attribute value, whether for background or foreground colour. It is possible to get four colours adjacent in text by using inverse colours, about which more later when you have become more advanced!

PLOT enables you to place a value on the screen, and there is a corresponding command, PRINT SCRN(X,Y), which returns the value found at column X, row Y. This is much easier than using PRINT PEEK (address), but I did not introduce PLOT and SCRN first because in some ways it is better to regard the screen as a block in memory, which can be filled with values lying between 0 and 255. Curiosity is then given a chance to act, prompting the programmer to ask "What happens if I place a value higher than 127 in a square?" To find out, you are just going to have to wait yet again, because it is now time to look at the third and fourth ways to introduce colours, by using the PRINT command. Now we are in unfamiliar territory, because this command has a logic all of its own, and some special effects such as double printing are available which cannot be obtained other ways.

The PRINT command can produce all of the colour effects described above, as well as cursor control, character printing and line blanking, and so is often the method preferred by programmers. What it cannot do is place values higher than 127 on the screen. If we try a line like:

```
PRINT CHR$(145) <return>
```

we will see our familiar red bar across the screen, so obviously a red background attribute has been printed there. What the Oric has done is to subtract 128 from the value requested and to place the remainder on the screen. The order of colour attributes is just the same as for PLOT or POKE, so it is easy to calculate what value needs to go into the PRINT command to have the desired effect. This can be very useful, because it is possible to construct a string, e.g. A$, from individual characters, any of which can be either an alphanumeric or an attribute (+128), and then print the entire string to the screen. It can thus contain embedded attributes, and since A$ can be printed anywhere (reasonably!) on the screen, your colourful message can be re-written all over without having to laboriously reconstruct the sequence of attributes and characters. Try the following program:

```
10 FOR I=1 TO 33
20 READ A
30 A$=A$+CHR$(A)
40 NEXT:END
50 DATA148,84,131,72,129,69,128,149,82,132,65,135,73,128,78,132,66,135
60 DATA 79,128,87,132,146,129,79,132,82,128,73,133,67,32,144
```

Try PRINT A$ anywhere on the screen, now, either by cursor positioning or PRINT@. Where the whole message appears in one line, the effect is predictable, but if the message is split between lines, some colours may be missed, because attributes only affect the line (row) they are on. So coloured titles and suchlike have to be prepared with a good deal of planning for positioning, and I use squared paper for this job. Some care has also to be taken with colour choice, because some combinations do not make good contrast, and any lettering can be hard to read.

Strings such as A$ can also be PLOTted, of course, and the effects are much the same as PRINT except that there is no double printing (a short cut for double height characters), and the individual elements of A$ should be below 128 if you want the same effects as PRINT. Try PLOTting the A$ you created in the previous program, to see what effects it creates. You will notice some sudden changes of colour where the attributes lie, and the effect at this stage is rather untidy. The reason for the colour change is that PLOT and POKE both accept values higher than 127, and their effect is to INVERT the colours of any printable characters or attributes. What happens on the TEXT (or LORES) screens, and as we shall see, on the HIRES screen too (but not with characters in HIRES), is that 128 is subtracted from the PLOTted or POKEd value, and the numerical value of the remainder, as its ASCII character, is placed on the screen. The difference between this and what was shown earlier is that with PLOT and POKE, the colour values for background (PAPER) and foreground (INK) are inverted, that is, they are subtracted from 7 and shown on screen in the resulting colour number. Try the following example:

```
 5 CLS:PRINT@ 0,10;""
10 PLOT 0,0,20: REM BLUE BACKGROUND
20 PLOT 1,0,6:  REM CYAN FOREGROUND
30 PLOT 2,0,"ORIC"
40 PLOT 0,2,148:REM BLUE B/G BUT SQUARE IS YELLOW
50 PLOT 1,2,134:REM CYAN F/G ON YELLOW SQUARE INVERTED B/G
60 PLOT 2,2,"ORIC"
70 M=SCRN(3,2)+128:PLOT 3,2,M
80 M=SCRN(5,2)+128:PLOT 5,2,M
```

You will notice than in both displays of "ORIC", the background colour goes to the end of the line, but in the 2nd "ORIC", the attribute squares are displayed in their inverse colours, although they have the same effect on the rest of the line. Only the squares holding values greater than 128 have their colours inverted, so it is possible to create some fancy displays where the background colour is inverted and the characters have their normal colours, or vice versa. This means that the maximum number of different colours in adjacent squares is four, where one set is the inverse of the other. To get more variations, we have to insert attributes for other colours, which means leaving a blank space between characters, although we can invert the actual colour shown where the attribute lies, as in the above program. (A little artistic restraint is advisable to avoid too much colour in plain text, although it allows more pleasing graphics.)

A Little More About Inverse Colours

There is a logic to the order of colour numbers used by our computer, and it lies somewhere in the hardware of the ULA (Uncommitted Logic Array) used to drive the various (i.e. two) screen modes. The Oric's designers have chosen a 3-bit colour register, which gives us 2^3 (i.e. 8) colours on screen. If we imagine our three bits (binary digits) in a row, creating a binary number, we can see that colour number 0 has a binary representation of 000. Our colour monitor or TV has three electron guns in the tube which are capable of lighting up three different coloured phosphors on the screen, by means of the "shadow mask", which is a metal screen with holes in. This arrangement ensures that each of the three guns only shoots electrons at one phosphor, so that each gun creates its own image in only one colour: red, green or blue. What the ULA does is to, in effect, switch on or off these three guns according to the binary pattern it has in its memory at the time. If it finds a value of 000, it switches them all off. The next value up, 001, enables it to switch on the red gun only, leaving the green and blue off. This is why a value of 1 on the screen is interpreted as a signal for switching on the red gun and showing everything scanned afterwards in red foreground. The next value up is 010 (2 in decimal), which switches off the red gun and switches on the green. Then comes 011 (dec 3), which to the surprise of some readers, (probably), gives a yellow colour and not a greenish-red. (This is just the way our brain interprets colour.) Next comes 100 (dec 4) which switches on the blue gun and switches off the other two, and so on up to 111 which has all the guns on and creates a passable imitation of white. The thing about inverting, then, is that if any of these patterns (e.g. 101 for magenta) is inverted (0s become ones and vice-versa), the effect is the same as if the number were subtracted from 7 (binary 111). The colours also become inverted in the sense that they are subtracted from white. Knowing this, you only have to remember the numbers for red, green and yellow, and you can calculate the numbers for the rest.

While we are on the subject of colour perception, did you know that the human eye can detect as little as a 0.2% change in the intensities of any of the three primary colours of red, green and blue? This gives 500 different levels of intensity in each of the three primaries, and therefore a total of 125 million discernible colours. (The exact number is debatable, and people's eyesight varies somewhat.) Computers that are capable of displaying photographic quality images need a register to signal those 125 million or so. Our Oric's 3-bit register can signal 2^3 colours, so 125 million requires around 27 bits (it's all done with logarithms!). I'm told they make do with 24 bits, which is 3 bytes for every point on the screen, and they have many times higher resolution than our 240*200.

To get a little practice in the creation of coloured text, we now go on to the fourth and last method of producing attributes, and that is using so-called Escape Sequences. Most printers use these to control their modes of output (such as italics), but the Oric uses them for screen effects. An Escape Sequence is any sequence of ASCII characters which starts with CHR$(27), which is the code for the ESC key on your keyboard. Only one character at a time can follow ESC on the Oric, so every time we want to put an attribute on the screen, we have to print ESC first. Pressing this key, or printing CHR$(27) changes the way other characters are output onto the screen, subtracting 64 from the ASCII code of the character. We can test the effects of this in immediate mode on the computer, just after powering up when the machine is waiting for Basic instructions. Type <ESC> <A> ORIC, but don't press <return>, and you will see the word ORIC printed on the screen in red letters. Press CTRL-X now to clear your input, or the Oric will remember you've typed it and come up with a "? SYNTAX ERROR" message later. What has happened is that 64 has been subtracted from the code for A which is 65, leaving 1, which is then placed on the screen. A value of 1 in a square is an attribute for red ink on that line. The rest of the Escape codes and their effects, can be found in the Oric handbook. Here is a little program that enables you to print away to your heart's content on the screen without causing too many problems. Control keys (CTRL+key, pressed together) may also be used, including CTRL+C to escape the program. Remember that certain control keys disable the screen or make it judder, but the useful CTRL keys are as follows:

C	QUIT
D	DOUBLE PRINT ON/OFF TOGGLE
L	CLEAR SCREEN
M	CARRIAGE RETURN
N	CLEAR LINE
Q	CURSOR ON/OFF TOGGLE
T	CAPITALS ON/OFF TOGGLE
]	PROTECT COLUMNS ON/OFF TOGGLE

Double printing is used mainly to create double height characters, for which the escape sequence requires the letter J, unless you want them to flash, when an L is required. Double height characters only appear correctly when the printing is in odd-numbered rows. The status line can also be used for them, although the ASCII values for the characters, including the value of 10 for double height, have to be POKEd there. Here is the program; beware of scrolling!:

```
10 CLS
20 GET A$:IF A$=CHR$(3) THEN END
30 PRINT A$;
40 GOTO 20
```

The 8-bit A to Z PART 7

Well here we go again with a look at the heady world of home computers, looking back at an era when computers were exciting, BASIC ruled and people were arguing over whether the Z80 or the 6502 was the better CPU chip. Well, the Z80 was developed into the 8080 and then the 8286, 8386, 8486 and so became the pentium chip also known as the '586. Yep! the chip that was in the ZX81 was developed into the chip(s) now used in most PCs thoughout the world. The humble old 6502 was developed also and became the 68000 - as used in Apple computers and Atari STs. Chips developed from the 68000 now sit inside the current Apple Macs, two fingers raised at the boring old clones sitting on the opposite shelves.

Nowadays nobody seems to care too much what is inside their white plastic box, just so long as it can play the latest gory 'beat em up' with wonderful geometrically shaped figures ! Ha ! I've seen more realistic graphics on a Speccy !!

Anyway, I digress. If you think I was forgetting the Commodores C16 then you were wrong - it's here below !!! (EEEeeeeekKkkk!) * **COMMODORE C16** I'm not quite sure why they decided to build this machine, but they did and they managed to sell the thing as well. This was another of those cases where a manufacturer stepped back and developed a machine that was inferior to they're latest model. This model came in the same case as the VIC 20 and early C64 except this time it was a horrible dark grey colour. I don't have the spec. on this model other than that it is 16K, but it is probably very similar to the VIC 20. A later model was released as the C16 +4. It had a new case (- still grey), but I'm not sure about other modifications. Nowadays you would be far better getting a C64 which will not cost you much more and has lots more software to spend your hard earned cash on.

* **DRAGON 32** As I was jabbering on about earlier processor chips were developed into better and faster ones. The 6502 family also contained the 6809E which was used in the Dragon. Supposedly one of the best 8-bit chips ever, but came onto the scene too late and so was not used in many machines.

The Dragon was a truly British machine being developed in Cambridge with the assistance of two (British) Universities, had components made in Scotland and was assembled in Wales and being financially backed by the Welsh Development Agency and the Prudential Group.

Released just in time for Christmas 1982, the machine offered a decent version of Microsoft BASIC and a 'full 32K of RAM'. The graphics handling is good but the commercial games I've got are often lacking in colour and the games are quite slow despite the capabilities of the CPU. Only one sound channel to play about with so all in all you're better off with your Oric and should look else where for an alternative. Is the user group still going ???

SPEC. 6809 CPU 32K RAM, 16K ROM. SCREEN 32 x 24 text. 256 X 192(max) *graphics.8 colours.INTERFACES Centronics, expansion and provision for two joysticks. 53 WP keys which make a funny clonking noise*

The Tandy Colour Computer is so similar that much of the software will work on both machines. There were plenty of books available which you can pick up at your local boot sale. Disk drives were available and the machine was expandable to 64K which leads us to the almost inevitable new machine ...

* **DRAGON 64** You guessed it, a 64K Dragon! For some reason they decided to give the Dragon a better keyboard, but miss off four of the keys like ESCAPE and CONTROL keys. the circuit board has been modified to allow space for the extra RAM and a serial port.

On switching on you are faced with a standard Dragon 32 to allow use of all previous software. A command moves the BASIC in memory to allow about 45K free - the full 64K is only available under other operating systems. Graphics are the same which is a shame as the maximum 256 X 192 only allows the use of 2 colours. The text display (which allows only a green or orange background is very limiting for a machine which was hoping to appeal to the business market.

Dragons disk unit comes with either one or two drives and has a small DOS included.

Dragon did however go to great lengths to make the OS9 operating system available which allows multi-tasking and multi-user facilities which may be interesting to some of you. If you consider buying it for this reason make sure you get the manuals.

READERS LETTERS

DEAR DAVE,

ALLY AND I ARE WORKING ON SOMETHING FOR THE 100th ISSUE OF 'ORIC USER MONTHLY'.

ENJOYED COLIN COOK's ARTICLE VERY MUCH,AND HOPE THAT IT WILL COVER MORE ON GRAPHICS BESIDES THE USE OF COLOUR. I THINK THIS IS AN AREA OF **ORIC** USEAGE THAT HAS RECIEVED LITTLE ATTENTION IN THE **ORIC** MAGAZINES.

- STEVE 'THE MUSO' MARSHALL.

DEAR STEVE,

LOOKING FORWARD TO YOUR ITEMS FOR THE 100th ISSUE. HERE AT O.U.M, I AM PLANNING A FEW SURPRISES. PERHAPS OTHER READERS COULD COME UP WITH THEIR OWN IDEAS.

GLAD YOU LIKE COLIN's '**RAINBOW** ORIC' ARTICLE. I THINK THAT YOU WILL AGREE THAT PART TWO (SEE PAGE 5), AND THE FINAL INSTALMENT (NEXT MONTH) ARE OF GREAT INTEREST.

- DAVE

DEAR DAVE,

I DON'T KNOW ANYONE WHO WOULD HAVE AN **ORIC** HERE IN **FINLAND**. IT DIDN'T EVER BECOME POPULAR HERE. I REMEMBER ONLY ONE GUY WHO HAD ONE BACK IN 1983-84, BUT HE GOT BORED WITH IT BECAUSE THERE WASN'T ENOUGH SOFTWARE AVAILABLE. I HAD A TIME TOO WHEN I WAS ABOUT TO ABANDON MY LITTLE MACHINE,BUT LUCKILY GOT IN CONTACT WITH PEOPLE LIKE ALLAN WHITAKER,STEVE HOPPS,ARNT ERIK ISAKSEN, YOU AND MANY OTHERS, AND SO I CHANGED MY MIND. I JUST WISH I HAD MORE TIME AND ENERGY (I AM A FATHER OF TWO GIRLS NOWADAYS) TO WRITE PROGRAMS. NOW IT SEEMS THAT ALL I DO IS PLAY GAMES WITH IT,BUT THAT IS BETTER THAN NOTHING,I GUESS.

I THOUGHT ABOUT WRITING AN ARTICLE ONCE,BUT GAVE UP,BECAUSE I COULDN'T THINK OF A SUBJECT I WOULD KNOW ENOUGH ABOUT EXCEPT MAYBE **"ORIC IN FINLAND"**, BUT IT WASN'T A GOOD IDEA,BECAUSE THAT WAS A SHORT STORY,AND A SAD ONE TOO.

ANYWAY,I WOULD LIKE TO SAY THAT I ENJOY **O.U.M** VERY MUCH,AND I AM GOING TO KEEP SUBSCRIBING.

- RAUL HAKLI (HELSINKI)

DEAR RAUL,

EVEN THOUGH YOU HAVEN'T COME UP WITH AN ARTICLE AS SUCH; YOU HAVE SENT US A LETTER OF INTEREST. IT IS ALWAYS NICE TO HEAR LITTLE SNIPPETS FROM OUR SUBSCRIBERS, ESPECIALLY THOSE FROM OUTSIDE THE U.K.

THERE IS NOTHING WRONG WITH JUST PLAYING GAMES ON THE **ORIC** - TO EACH HIS OWN. WHERE WOULD THE GAMES PRODUCERS BE WITHOUT THE GAMES PLAYERS. IT IS JUST A PITY THAT YOU ARE ONLY CASSETTE BASED, AS YOU MISS OUT ON SOME OF THE NEWER TITLES.

BY BUYING THE '**ADVANCED USER GUIDE**' BOOK, IT PROVES THAT YOU HAVE MORE INTERESTS THAN JUST GAME PLAYING, AND HOPEFULLY YOU WILL FIND MORE TIME IN THE FUTURE TO DO OTHER THINGS. EDUCATIONAL SOFTWARE FOR YOUR DAUGHTERS WOULD BE A GOOD PLACE TO START. IT WULD HELP YOU AND THEM.

— DAVE

LETTER TO THE EDITOR,

Following the publication of a photograph of one of our members (OUM - Sept/Oct.'95 - page 39), we, the Countrymen Enhanced Organisation of Muslim Armed Groups (CEOMAG), solemnly demand the Royal Family (including cats,dogs,canaries and hamsters) to publicly convert to Islam.

Lacking this, we will deposit a bomb:

a) 65 Barnard Crescent,Aylesbury b) 3 Madingley Road,Cambridge
c) Frank Bolton's place (we will find out)

Moreover, Waterloo Station will be renamed Gare d'Austerlitz and subsequently, the subway Bakerloo Line will become Berlitz Line.

- Allah Akbar

DEAR ALLAH,

we know who you really are - the postmark was a giveaway. Nice to see that you haven't lost your sense of humour, Jean. For your information, Frank Bolton lives at: - Kapow!, Bang!.....

DEAR DAVE,

I have had a good hunt, but I am afraid that my last commercial title - "AIR-SEA RESCUE" has probably been lost forever. I can't find it anywhere.

- STEPHEN HAIGH (Knottingley)

DEAR STEPHEN,

it's a great shame, but such is life! At least we can still share your other titles,such as GHOSTMAN and GRAVITOR.

- DAVE

DEAR DAVE,

I have not yet got the start-up disc for my MACINTOSH, but I've come across a telephone number for the Yorkshire and Humberside MACINTOSH Users Group, and so I might be able to get a stsrt-up disc from them.

I thought the latest issue (the double issue) of OUM was one of the best ever issues. I enjoyed the first part of the series on graphics. Graphics is one area which appeals to me but is something that I hardly know anything about.

- ROBERT CRISP (Meanwood)

DEAR ROBERT,

perhaps you could let us have further details of the MACINTOSH group, as it may be of interest to others.

- DAVE

DEAR DAVE,

I enclose my cheque for renewal to OUM. I aplogise for not taking a more active part in the OUM proceedings. I've not been an active 'ORICER' for some years now,but I do enjoy the club-like atmosphere that comes through the pages of the magazine. My thanks to you and the other regulars for all the hard work that must go into the mag.

I've recently taken delivery of the excellent Emulator (OH to have Fabrice Frances' ability!) in the hope of re-kindling the ORIC spark.

Watch this space!

Keep up the good work.

- PAUL FARNESSE

DEAR PAUL,

I HAVE SAVED YOU A SPACE! - DAVE

BITS 'N' BOBS

EUPHORIC TIPS

SOME TIPS FOR 'EUPHORIC' USERS FROM JIM GROOM:-
A) Rename EUPHORIC.EXE as O.EXE to save typing!
B) WINDOWS users - save your favourite Oric game screenshot using EUPHORIC!'s built in screen capture routine. Then use the resulting BMP file as wallpaper in your WINDOWS environment. This is especially effective with WINDOWS 95.

Now for a poser from Jim - he asks if Fabrice can come up with a better way of changing between the ROMs?

==

MAGNETIX and EUPHORIC

Jonathan Bristow is currently working on a Demo version of MAGNETIX for the INTERNET. In jonathan's game, the transfer to another machine is rather involved.

==

GET WELL SOON

An appointment with a surgeon meant that Derek Smith was unable to make it to this years' MEET. We wish him a speedy recovery.

==

NICE ONE CHARLIE!

Thanks to answering Peter Thornburn's plea for info on the SORD and LYNX go out to Charlie Wallis. I have passed the info on.

==

RENE DOES IT AGAIN!

RESIDENT SUPER GAMER RENE MARKE HAS STRUCK GOLD AGAIN. RENE SCORED 197,500 ON THE FRENCH PINBALL GAME - 'MACADAM BUMPER'.
WELL DONE RENE - EAT YOUR HEART OUT HENRY!

==

Machine Code for the Oric Atmos (Part 50)   Peter N. Bragg

The Story so far

The past few issues have looked fairly intensively at the subject of Zero Page instructions. These can be very useful instructions that combine the operations of indexing and indirect addressing, which allows them to be used to fetch, store or modify data in blocks or scattered locations. The last article, included a short demo of both "Pre-Indexed" and "Post-Indexed" Zero Page instructions together. The demo routine used the Accumulator to fetch an item from data memory and put copies of it onto the screen display.

Hopefully, breaking the instructions down into a number of small and simple operations and then dealing with those separately, before building up again to the full blown instruction operation, makes more complex instructions, such as Zero Page, easier to grasp, in the long run.

We have now more or less covered the subject of Zero Page instructions. They are probably the most complex of the Oric's 6502 machine code collection, so if you have got those "sussed out", you are unlikely to have trouble with any of the other machine code instructions for the Oric.

So What Now ?

This series has concentrated on explaining the Oric's 6502 Instruction Set and I must confess that it has taken a lot longer to cover the subject than I first thought. The first few articles in particular, had to fill in a lot of essential detail. Although such detail is common knowledge to some of us, there are many who find the lack of know-how on the basic elements of programming and how a computer works, a severe handicap.

Often it is the simple things that produce the stumbling block. Way back when I started computing, I was always stumbling over an instruction that required an "offset" (occasionally referred to as a "displacement").

I hadn't a clue what an "offset" was and the only explanation I could find, appeared to be some nutty exercise in binary maths and punctuation. This was in the early days and there was no friendly user group available for advice and my knowledge on the subject was very sketchy. The only way to find out, was to enter the instruction and some test values into the computer and note the results.

Now that I am quite familiar with that instruction, it all seems so simple. The problem instruction was the Branch and the "offset" (or "displacement) was the value that just set the distance and direction of the relative jump, that the Branch instruction makes, if the conditions are right. Any value from 00 to 7F would produce a forward jump of that number of locations. Values of FF down to 80 would do the same in the opposite direction, starting with FF and going furthest back with value 80. It may seem simple now, but I can well remember the frustration of trying to work out just what it did, from the rather vague information in the manual. The manual's description still seems difficult to comprehend, even now.

So why did I continue to use machine code, in view of such problems ? Well the Basic language was an expensive extra on many of the early micros, when I started. So like many people at the time, I made do with machine code, which was faster and didn't need so much expensive memory. Later, Basic became more widely available. However, it was less attractive, because it was slow and I found the variations between different versions irritating.

The other popular language, in the early nineteen eighties, was Pascal, which was faster than Basic and more portable, which meant that software written in Pascal could be run on other machines, without the need for extensive changes, to cater for different versions.

That was the theory anyway. However, in practice even the commercial software writers ran into a lot of problems with Pascal, particularly when debugging software. From the professional users point of view, we found that commercial software was a poor advertisement for languages such as Pascal, which is why I continued to use the machine code/assembly language, for writing my own software.

Other languages have also had their brief spell of popularity, some have even been hailed as the ultimate magic language which will make writing and debugging software an easy job. None of them have achieved that goal. The great survivor appears to be "C" and it's variants, but nobody could call it a simple to use and friendly language.

Something New ?

--------------- So what has all this to do with machine code ? Well there appears to be an interesting development on the software front, that I have recently read about. This is an operating system called "Taos", which has been under development for six years. From the description that I saw in a recent copy of "Acorn User" it appears to be a new version of an old idea.

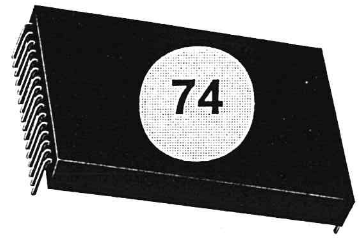
I wonder how many remember "Basicode". Those of us who tried "Basicode" in the mid-nineteen eighties had great fun passing Basic software on tape, between the Apple, ZX81, BBC Micro and others. The trick was done by installing the same Basic and filing system in all of the different computers. The main snag was that it tended to slow the faster machines down and for obvious reasons, the lowest common denominator prevailed, so all the computers tended to look like the old ZX81 in operation. It was a very clever idea for the time. There was even a version for the Oric although I wasn't able to get hold of that one.

The information that I have seen so far about "Taos" is a bit sketchy. However, it seems that it is not so much based on a language this time, but on a machine code system that acts as a kind of pseudo microprocessor.

Apparently, what it does, is to install it's own machine code system as an interface to the microprocessor currently installed. As a result, many different types of microprocessor will be able to use exactly the same machine code language. So it follows that the same software written for the system will run on any machine that has the system installed and it will be able to run on many different computers with different microprocessors.

The advantages of a system like this, are that it can be quite small. As a result, it is unlikely to slow down the overall computer speed. More to the point, because it works at the microprocessor's level, it is less affected by hardware. Computers with better performance and hardware, will not be dragged down to a common level by less well equipped machines.

A single machine code language system, would affect all software including other languages and make software writing much simpler. Snags ? Well dont hold your breath, waiting for a 6502 version. The developers have decided that an eight bit version is not worth developing. Never mind, it does mean that knowledge of machine code operation could be very useful in the future, as most microprocessors seem to have the same base functions...............



RAMBLING IN THE ROM

Euphoric... just keeps improving!

Fabrice hasn't let me down. The latest version is v.0.8b, and is now fully optimised to run on a 386 as well as a 486 PC. That means you 386 owners can run all forms of the emulator, including Sedoric and a Telestrat. Also, as mentioned in last month's OUM, there was a bugged version of 'writedsk', the utility to write an emulator disk image to an Oric formatted disk, distributed by me. I discovered the problem, mailed Fabrice, and the debugged version was on the server a day later! I would suggest anyone who uses the emulator regularly sends their disk back to me for an upgrade - no charge, just an S.A.E. to return it to you.

Here, then is the recent version history:

V 0.7alpha

- divergent release...no bank switching mechanism (so no disk system), but a new 6502 emulation optimized for 386, a faster keyboard reading, and a faster clock code. Should be the base for a complete release optimized for 386, slow with bank switching (like 0.6alpha1) but fast anywhere else (to do: a modified Sedoric with no programmed function keys).

V 0.7a (from 0.6e)

- faster clock code and keyboard reading
- first TeleStrat emulation ! Partial emulation of the second VIA, no ACIA emulation yet (will come quite soon...)

V 0.7b (486)

- new 6502 emulator, I have made bencharks and it is faster thanks to fewer cache misses (despite additional jumps)

V 0.7c (486)

- Restores the correct aspect of the screen (during a power reset) lost with 0.7
- Handles differently the F7 and F8 key for the Telestrat
- Modification of the FDC 1793 module to work around a nasty behaviour of Stratsed: selecting a drive AFTER sending a read or write command and BEFORE the actual

transfer (not always the case, so... a bug ?)
- first emulation of the ACIA, but still lacks interrupts handling.

V 0.7d (486)

- Oops, I forgot to copy files from Linux to Dos, so neither the screen aspect nor the F7/F8 keys were corrected under Dos...
- first emulation of the real time clock extension (with Telestrat for now)
- updates the Bios timer when exiting from the Dos version (no need for such a hassle with unix of course. Hey, have you tried Linux ?)

V 0.7e (486)

- Repair the broken dump/restore mechanism
- Generously give an RS232 interface to Oric1/Atmos users (but I have not tested it yet), and a real-time clock extension.

V 0.7f (486)

- Patch for international keyboards on the Telestrat
- Fix the RS232 interface, it works in polling mode...(interrupts not implemented yet)

V 0.7g (486)

- Move the IO address of the RS232 interface on the Oric 1 and Atmos

V 0.7h (486)

- Serial interrupts are in! The 6551 ACIA emulation is now complete, the communication software of the Telestrat runs identical to the original.

V 0.8 (386) (01/10/95)

- At last! A cross-over of 0.7alpha and 0.7h achieving the speed of versions anterior to the introduction of the disk controller. Owners of fast 386 are able to enjoy it (and now... rushing to the 286...
- PS/2 compatible

V 0.8a

- Oops, I forgot to modify the Real Time Clock module, sorry...

V 0.8b

- Added additional cycle on absolute indexed and indirect indexed read addressing modes, and branch instructions, when crossing a page boundary. Timings should now be exactly similar to a real 6502.
- Improved the B flag implementation (still have to better implement N and V flags in decimal mode).

And if that's not enough...

Amoric arrives!

Now available is an Oric emulator for the Amiga. You can download it directly from the Ensica server - FTP or Web - the Web address is URL:
http://arlesienne.ensica.fr/LOCAL/ORIC/oric_emglish.html

...and now the software

One effect of the emulator is that a number of 'old' Oric owners have returned to their favourite (virtual) machine. An interesting feature is that programs can be written in the C language, compiled, and crossed to the Oric. They then run as stand-alone Oric programs, notable for their speed and sophistication. King of the crop at the moment is an arcade game, 'Slime', written by Alexios Chouchoulas, of Athens and Edinburgh. Excellent graphics and great gameplay are the features. There is also a character editor, CAP 1.0, with some very sophisticated pixel scrolling routines, and a VT100 screen terminal emulator, a sophisticated piece of software to enable you to communicate with other computers over a serial connection. And finally there is Fabrice's 'Tea for Two', a chess game playable between two Orics over a serial link. You can plan and enter your next move without the other player knowing of it, and so in a timed game create panic in the opposition!

If these programs are a sign of things to come, then we are in for some amazing new progams for our trusty Orics. Incidentally, subscribers to the CEO software discs will get these new programs as a matter of course - if you are interested in a subscription there are some special offers to tempt you:

CEOmag July - December 1995 (all the Internet happenings) just £5.00
As above, with the Autumn and Winter 3½" CEO software discs (Internet software) £10.00
As above, with the Autumn/Winter 3" CEO software disc £12.00

Or, take out a subscription now for 1996 and get the November and December CEOmags free. Subscription rates are:
CEOmag only £11.00 - plus 2 double 3" discs £22.00
- plus 4 single 3½" discs £20.00

If possible, please order 3½" discs - they are cheaper, easier to come by, and you get one a quarter rather than a 3" disc (which contains all the same software) twice a year.

If you are interested, my address is at the end of this article.

Rambling on in the maths...

'/' (OPERATOR) ACC2/ACC1 --> ACC1

Principle:

As for multiplication, the calculation is reduced to the division of mantissas, and their justification. To subtract exponents the opposite of the exponent in ACC1 is calculated: to add the opposite is to subtract.

The division of mantissas is more complicated, but the result is the difference between the multiplicand and the multiplier.

Programming:

The routine lacks refinement - numerous branchings could have been avoided. It doesn't affect the speed of execution, but a good ten bytes could have been saved.

```
DDE3  BEQ DE5B     DDE7  BEQ DE5F     if ACC1=0, 'DIVISION BY ZERO ERROR'
DDE5  JSR $DEEC    DDE9  JSR $DEF4    AACC1-->ACC1
DDE8  LDA #00      DDEC  LDA #00
DDEA  SEC          DDEE  SEC
DDEB  SBC D0       DDEF  SBC D0       complement the exponent
DDED  STA D0       DDF1  STA D0
DDEF  JSR $DD78    DDF3  JSR $DD7C    then calculate the sum of the exponents
DDF2  INC D0       DDF6  INC D0       adjust
DDF4  BEQ DDA0     DDF8  BEQ DDA4     and Overflow if falls to 0
DDF6  LDX #FC      DDFA  LDX #FC      index ACC3 (#98+#FC+1=#195, i.e.#95)
DDF8  LDA #01      DDFC  LDA #01      initialise for 8 shifts
DDFA  LDY D9       DDFE  LDY D9
DDFC  CPY D1       DE00  CPY D1       compare ACC1 and ACC2
DDFE  BNE DE10     DE02  BNE DE14     jump if different
DE00  LDY DA       DE04  LDY DA
DE02  CPY D2       DE06  CPY D2
DE04  BNE DE10     DE08  BNE DE14     same for byte 2
DE06  LDY DB       DE0A  LDY DB
DE08  CPY D3       DE0C  CPY D3
DE0A  BNE DE10     DE0E  BNE DE14     same for byte 3
DE0C  LDY DC       DE10  LDY DC
DE0E  CPPY D4      DE12  CPY D4       same for byte 4

DE10  PHP          DE14  PHP          save sign comparison (in C)
DE11  ROL A        DE15  ROL A        assemble the result
DE12  BCC DE1D     DE16  BCC DE21     jump if not yet 8 shifts
DE14  INX          DE18  INX          adjust index, and set N and Z
DE15  STA 98, X    DE19  STA 98, X    save resulting byte
DE17  BEQ DE4B     DE1B  BEQ DE4F     index=0, the extension is passed
DE19  BPL DE4F     DE1D  BPL DE53     index positive, end
DE1B  LDA #01      DE1F  LDA #01      indicate new 8 shifts
DE1D  PLP          DE21  PLP          recover comparison
DE1E  BCS DE2E     DE22  BCS DE32     Dividend > divisor: subtraction
```

Shift dividend (ACC2), that is divide by 2

```
DE20  ASL DC       DE24  ASL DC       byte 4
DE22  ROL DB       DE26  ROL DB       byte 3
DE24  ROL DA       DE28  ROL DA       byte 2
DE26  ROL D9       DE2A  ROL D9       byte 1
DE28  BCS DE10     DE2C  BCS DE14     1 exit: ACC2 must > ACC1
DE2A  BMI DDFA     DE2E  BMI DDFE     makke comparison, uncertain
DE2C  BPL DE10     DE30  BPL DE14
```

Tailender

Don't forget the new Sedoric manual, £5.00 and help OUM funds - and look for something special next month...

Jon Haworth
3 Madingley Road
Cambridge
CB3 0EE

Email: jon@cam.dungeon.com
Oric Mailing List: oric@cam.dungeon.com

BRIANS PAGE - O.U.M. NOVEMBER 1995

- WELCOME TO ALL

STRAIGHT INTO BUSINESS FOR THIS MONTH AND HERE ARE SOME POSERS FOR YOU TO HAVE A GO AT.

1) WINTER IS UPON US , RAIN ET AL - BUT WHAT ABOUT THE SUNSHINE OF THE SUMMER ? FOR THIS POSER THINK OF THE ABOVE TYPES OF WEATHER AND REPLACE THE '---' SEQUENCE WITH A MALE NAME , TO PRODUCE A WONDEROUS SIGHT :

THE SEQUENCE .... ' - - - G B I V '

2) STARTING WITH THE WORD 'HUGE' AND CHANGING ONLY ONE LETTER AT A TIME , TO PRODUCE KNOWN WORDS , CAN YOU , IN FIVE STEPS , MAKE THE WORD 'TINY' (Ie Six words in total Incl. Huge & Tiny) ?

3) THE BROTHER OF AN ALCHOLIC DIED , YET THE MAN WHO DIED HAD NO BROTHER ! HOW IS THIS POSSIBLE ?

4) ASK A FRIEND TO THINK OF ANY NUMBER UNDER TEN MILLION (10,000,000).WHAT IS THE MAXIMUM NUMBER OF 'YES' OR 'NO' GUESSES WOULD YOU REQUIRE TO FIND THE EXACT NUMBER ? TEN MILLION OFFERS AN AWFUL LOT OF POSSIBILITIES BUT THE ANSWER IS SUPRISINGLY SMALL. CAN YOU PROGRAMME ORIC TO HANDLE THIS BINARY SORTING ?

5) FINALLY A WORD SEARCH - HIDDEN IN THE FOLLOWING GRID ARE TWENTY-THREE (23) BOARD GAMES - SOME CURRENT SOME DEAD AND BURRIED - THEY CAN BE VERTICAL , HORIZONTAL OR DIAGONAL . CAN YOU FIND THEM ?

```
--- - - - - - - - - - - - ---
  E K E N S I N G T O N T Y
  D C D S N S L U D O E M C
  P A I E A R E E O G N A A
  H M R T K E V H O N O S M
  Y P R T E K D H C I A T O
  N A I L S C R A B B L E L
  N I K O A E A L X A U R P
  E G S T N H U M Y C M M I
  P N N Y D C G A L K R I D
  A O O L L E H T O G O N J
  H K O U A S T V P A F D I
  E E E I D E S K O M S R G
  V D D B D N S E N M E E S
  O S M E E I H I O O N D A
  H H I D R H C S M N T H W
  S Y U A S C A R E E R S X
--- - - - - - - - - - - - ---
```

GOOD LUCK WITH THEM - AND SEND SOME REPLIES DOWN/UP TO ME !

NOW I APOLOGISE FOR MISSING LAST MONTH , UNFORTUNATELY MY DAUGHTER HAD SEVERAL HOSPITAL VISITS , WHICH RESULTED IN HER HAVING TO UNDERGO MINOR HEART SURGERY - THANKFULLY ALL WENT WELL , AND SHE IS BACK AT SCHOOL NOW. BEING FOUR AND A HALF YEARS OLD THOUGH AND REGISTERED HANDICAPPED , OCCASIONALLY THIS WILL HAPPEN , BUT HOPEFULLY NOT FOR A LONG TIME AGAIN !

I HOPE YOU ARE ALL KEEPING THOSE TICKETS , FOR NEXT MONTH - OUM'S 100TH ISSUE - THE DRAW WILL BE MADE - IF YOUR A WINNER FOLLOW THE INSTRUCTIONS THAT I WILL PRINT ALONG WITH THE WINNING TICKET NUMBERS AND THE MATCHING PRIZE.ANY UNCLAIMED PRIZES WILL BE GIVEN TO DAVE TO DO WHAT HE WANTS WITH.

BRIAN - CONTINUED

DO YOU LIKE THE IDEA OF A FREE DRAW , AND WOULD YOU LIKE ONE NEXT YEAR ? PLEASE LET ME KNOW . ALSO REMEMBER I DO THIS ON MY OWN BACK , PURELY FOR THE BENEFIT OF OUM SUBSCRIBERS , AND AS A THANKYOU FOR STAYING LOYAL TO OUR ORIC COMPUTER .

NOW FOR THE ANSWERS TO THE LAST SET OF POSERS I SET .....

6 DIGIT NUMBER POSER .... ANSWER = 999999

(999999 / 7 = 142857 : 142857 * 6 = 857142)

ATTENDEE'S AND HANDSHAKES...

LAST YEAR 85 PEOPLE = 3570 HANDSHAKES

THIS YEAR 120 " = 7140 HANDSHAKES

THE UNLUCKY GAMBLER....

WELL HE STARTED WITH SIXTY TWO POUNDS AND 32 PENCE , LOSING 32 POUNDS 66 PENCE BETTING + 50 PENCE DOWN THE DRAIN , GIVING A TOTAL LOSS OF 33 POUNDS 16 PENCE = EXACTLY HALF OF HIS ORIGINAL AMMOUNT (33.16 * 2 = 66.32) - DID YOU SOLVE IT ?

FINALLY - HAVE ANY OF YOU EVER PLAYED MY P.D. GAME 'HIRE SALT CITY' - IF YOU HAVEN'T - WHY NOT ?! WELL FOR ALL YOU TAPPERS HERE IS THE ORIGINAL GAME , FROM WHICH MY GAME WAS CONCEIVED .

```
1   Q$ = CHR$(8)
5   IF Y = 0 THEN Y = Y + 1 ELSE IF Y > 10 THEN Y = 1
10 D = 1 : A = 120 : B = 149 : GOSUB 500 : REPEAT
20 A$ = KEY$ : IF A$<>"" THEN ON D GOSUB 100 , 120 , 140 , 160
30 ON D GOSUB 200 , 210 , 220 , 230
40 V = POINT (A,B) : IF V=-1 THEN 300
50 CURSET A , B , 1 : Z = Z + 1
60 UNTIL Z = 1500 : P = P + Z : Y = Y + 10 : PING : GOTO 5
100 IF A$ = Q$ THEN D = 4 ELSE D = 2
110 RETURN
120 IF A$ = Q$ THEN D = 1 ELSE D = 3
130 RETURN
140 IF A$ = Q$ THEN D = 2 ELSE D = 4
150 RETURN
160 IF A$ = Q$ THEN D = 3 ELSE D = 1
170 RETURN
200 B = B - 1 : RETURN
210 A = A + 1 : RETURN
220 B = B + 1 : RETURN
230 A = A - 1 : RETURN
300 EXPLODE : FOR I=1 TO 7: PAPER I : CIRCLE I,1:WAIT 25: NEXT I: WAIT 200
305 L = L + 1 : IF L < 3 GOTO 5
310 TEXT: PRINT "YOU ARE DEAD BUT YOU MANAGED TO SCORE " ; P + Z ; " POINTS"
320 END
500 HIRES : INK 1: CURSET 60 , 50 , 1 : DRAW 0 , 100 , 1 : DRAW 120 , 0 , 1
505 DRAW 0 , -100 , 1 : DRAW -120 , 0 , 1
510 IF Y / 200 = INT ( Y / 200 ) THEN 550
515 FOR C = 1 TO Y + 50: CURSET RND(1) * 113 + 60 , RND(1) * 92 + 50 , 1
520 CHAR RND(1) * 90 + 32 , 1 , 1 : NEXT C
530  FOR C = 1 TO 104 STEP 8 : READ X : CURSET C + 70 , 40 , 0: CHAR X , 0 ,
1 : NEXT C : RESTORE : RETURN
540 DATA 72 , 105 , 114 , 101 , 115 , 97 , 108 , 116 , 32 , 67 , 105 , 116 ,
121
550 FOR C = 62 TO 178 STEP 3 : CURSET C , 54 , 1 : DRAW 0 , 92 , 1 : NEXT  C
: GOTO 530
```