

ORIC

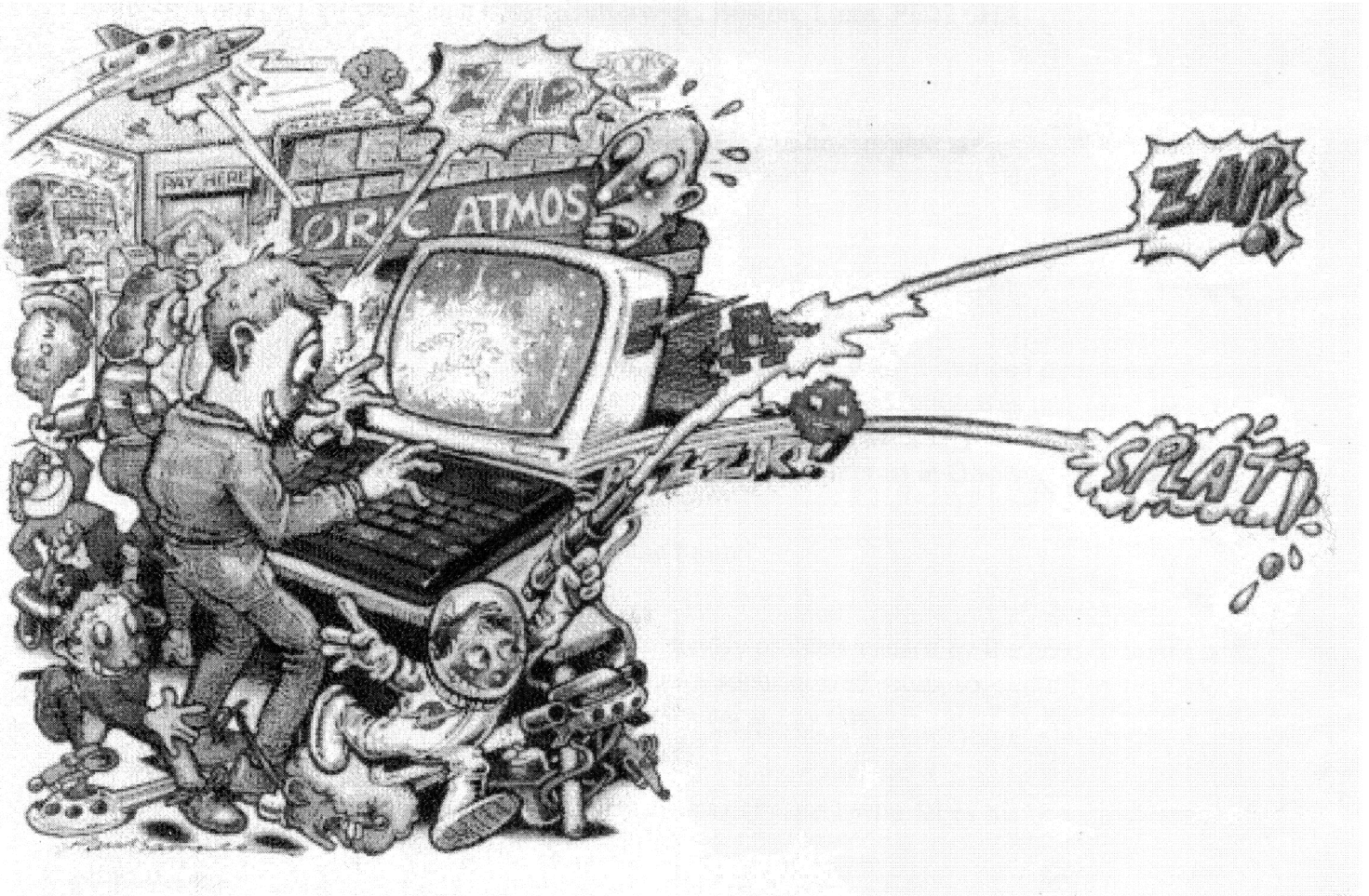
Number **140**

April 1999

USER MONTHLY

Keeping the
Oric alive

with Alternative Micros



Just 15 years on (part 5).....

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

The Editorial

HI, this OUM is late, and thus straight to it!

The Index

Page 1 - THE COVER from Jon Haworth.....Page 2 - EDITORIAL..... Page 3 - NEWS
 Page 4 - E-MAILS TO THE EDITOR..... Pages 5/6 - LETTERS TO THE EDITOR
 Page 7 - ON THE NET Page 8 -BITS'n'BOBS.... Page 9 - COMPILER
 Pages 10/11/12 - MACHINE CODE FOR THE ORIC ATMOS (Pt. 83) - Peter Bragg
 Page 13 - GAMES PROGRAMMING - Muso Pages 14/15 - BRIAN's BASIC.....
 Page 16 - MY ORIC STORY - John Foggin..... Pages 17/18 - RAMBLING IN THE ROM with
 Jon Haworth.....Page 19 - THE BACK PAGE

ON THE MOVE

David Goodrum has moved to Hampshire, and is now resident at:

31 Van Dyck Close, Black Dam, Basingstoke. RG21 3QJ He also has a new e-mail address
 : d.goodrum@cwcom.net

ALREADY MOVED

Simon Ulyyatt moved a few months back, but I've been sending his OUM's to an old address.

Simon is now at:32 Peter Paine Close, Butterwick, Boston, Lincs. PE22 0HA

Tel 01205 761558 or 0411 395622

ON THE NET

Dave Utting has joined the ranks of Internet users. He can be e-mailed at:

dave.utting@btinternet.com

BACK IN TOUCH

Denis Bonfield, who subscribed to OUM until mid- 1995 is back in contact. Denis is best remembered for his musical pieces and hi-scores in TETRIX.

Also back in touch is County Cork resident David Clifford, who subscribed until the end of 1997. In David's e-mail he wrote:"My reason for stoppin my subscription was that I just didn't have the time for the Oric anymore. I enjoyed practically all the articles in your mag and I would recommend it to anyone, whether they want to keep up their interest in Oric matters or just for the 8-Bit programming side.

Keep up the good work, and best wishes for the future"

PROBLEM SOLVED

Peter Bragg informs me that he has solved the display problem caused by the need to upgrade to a new RiscPC. The computer system is now quite a sophisticated setup, topped off by the Oric Atmos of course, and the whole lot is fully operational at any time.

BEST OF LUCK

Best of luck to Turkish subscriber Oguzhan Yilmaz (Peacer) who takes his post-graduation medical specification examination on April 10th.

E-mails to the Editor

Hi,

I've just received OUM 138/9 and read Jim Groom's most interesting "Email to the Editor".

I must say this is one of the most sensible/reasonable contribution (if not THE MOST) that I've seen so far toward producing an English CEO-MAG.

I just would like to tell you that you have my full support (at least moral support, because my workload does not allow to be of much more help any longer).

And I sincerely hope that others will support you for the greatest benefit of non - French speaking Oric fans.

Orically yours,

Laurent Chiacchierini - CLUB EUROPE 



Hi Dave,

The Big Issue certainly was excellent - one of the best OUM's for a while. I tried out Jon's mirror website, but have been finding it slower than connecting to the French site - the net is a fickle beast.

Regarding Euphoric, I have found this latest release to be the best. No problems at all. Ah, well, at least with so many versions out there you were able to find ONE that worked for you. I see you've seen the Microtan 65 emulator on Fabrice's site. Thank God they invented the

Oric.....

Colin Stein, who was hoping to get hold of some old Oric tapes, still hasn't got back to me with an address, so it looks like thats a dead end.

For Paul Farnesse's information, there IS a light pen available for users of Euphoric. Once selected it is controlled by the PC mouse and is displayed on the screen as such. In fact, under Euphoric, the light pen effectively becomes a mouse. Have tried it out - it works quite well.

Haven't tried it with any software however. Anyone fancy writing a point and click adventure?

- Jim Groom



Hi Dave,

My real Atmos isn't with a diskdrive. (I'm planning to connect my old 40MB IDE harddisk to Oric..)

Jani Tiainen (Finland)



Dear all,

Just a short note to say latest check up in Bristol happened today. No signs of any problems (even though it was a dreadful x-ray) and it's back again in three months.

At least is was a day off work...

- David Goodrum



Hi Dave,

Sadly, all maps on my site are in colour, thus making re-production in OUM difficult. I will have to redraw the maps again at some point. Damn! Looks like this avenue is shut for a bit, sorry!

- Jim Groom



OUM - all the news from your e-mails.

Letters to the Editor

Dear Dave,

my wife recently found her way into the loft and discovered my Oric in its original box with a case of software. She appears to have promised all of this to the School Caretaker, who is collecting old computers at the School. She is able to justify this on the grounds that everybody has access to PCs at work and at home, and at School, although Neil seems to think that the latest MAC machine is streets ahead of the rest of the market and his home PC is now obsolete and needs replacing. Perhaps we should replace our Orics with the latest MACs. It appears that PCs are obsolete as soon as you get them out of the box, and connect them up to the Internet. In the mean time, I am going to Washington & New York to check up on the latest computer packages.

- Rod Titchener (South Croydon)

Dear Rod,

if your local school caretaker is looking for any other computers, then let me know, and I'll put an advert in OUM.

I must admit that the MAC certainly gets good revues from both press and users.

Hope the trip went well - we hope to visit a few parts of the States next year.

- Dave

+++++

Hi Dave,

It was a bit of a shock to see the announcement of your Oric retirement, and I was saddened to think of you leaving the scene in such a manner. Reading between the lines and various articles in OUM and CEO about the amount of work, sweat time and even money that OUM has cost you over the years, it becomes easier to see why you have taken the decision that you have. The general lack of interest from the membership may not of helped either, but in comparison to a Union branch I was in once, we do participate quite well.

As a suggestion has been put forward to form a new magazine, mainly for those of us who are not on the Net, I would be happy to subscribe and even hopefully contribute something or two. On the issue of a new magazine maybe we could borrow an idea from an electronics magazine I get . They publish a printed version of their commercial magazine every month and then after a couple of weeks the core of the magazine is available for downloading, via credit card transaction. Possibly we could make arrangements for any new magazine to be put on a secure web site, or encrypted for which you have to buy a new key or password each year. I don't know if this is feasible or not, maybe some one with the correct knowledge could comment, but if it is then the membership could be enlarged significantly.

At the same time I would be happy for my name, rank and telephone number to go a new contact list for those of us whose means of communications only include telephone and Royal Hail.

Oric CD.

As an interim project has anyone had any thoughts on copying tapes to minidisc in audio format rather than onto CD in digital format. The recorders are dearer than an Oric Disk, but would be a way of storing cassette software in a somewhat more robust format, that would take longer to deteriorate and be a way of possibly saving some software.

ORIC mk2 (or 3, son of Statos ?)

I am still of the mind that the take up on this project will be low on this side of the channel, at least.

However the electronics magazine that I mentioned earlier is the middle of a series of articles about a teaching computer. This costs in total about £130 and uses a pre-programmed micro-controller with 64K of low power static ram and led readouts. It is programmed from a Windows PC and uses a MAX232CPF RS232 port and a 82C55A2 port. The board is about 6 ins by 8 ins amd is double sided and costs £15. The CD-ROM and the programmed microcontroller both cost £15, so we have a guide at the price of a new ORIC.

Letters to the Editor

The specifications of the new ORIC have improved since the project was first announced and have answered some of the doubts about the original, but not all.

I would estimate that the board needs to be about 6 ins by 8 ins and will probably need to be double sided. We can get a full 128K of memory in one 32 pin SRAM chip and a flash ROM in another 32 pin chip. This would cut out the need for refresh from a ULA, possibly enabling the use of a somewhat smaller newer field programmable chip instead of the ORIC ULA.

With the operating systems and BASIC being on flash ROM, no eraser or programmer would be needed, all that would be needed would be the ROM with a single system installed and a disk containing all of the various operating systems, which could be uploaded as required using a small utility. This semi-loaded flash ROM, the disk of operating systems, the board and a possible ULA, new or old, could hopefully be provided as a kit to interested parties.

If full onboard decoding of page 3 is included in the design then a separate expansion board should be possible. This could consist of a board offering buffered 8 bit slots to in line with the PC ISA slots. Then if anyone wanted to add a separate, say, sound card to his machine he would have an easy route to connecting it. If we had two serial ports on the main board then perhaps the current series on the ORIC mouse could be extended to cover the use of one of these ports.

- John Foggin (Ashington, Northumberland)

Dear John,

thanks for the interesting letter. What the readers don't know is that you wrote it just before Xmas, and sent it to me with another one dated mid - February. I'll print that other letter after this reply.

I'll leave it to you guys to sort out whether there is life after OUM, and what form, if any, it takes. Mini disc sounds good to me. I purchased one back in November, and have been merrily copying my old vinyl records to minidisc - scratches and all! I find the machine has better facilities than a CD recorder. There might be a problem attaching a minidisc player to a PC, so that Euphoric users can benefit.

STRATOS 2000 or whatever it was going to me called, is now I think long dead and buried as a project, unless I've missed some important announcement sent to the 'comp.sys.oric' newsgroup. I must admit I've only just re-subscribed to the group, and have missed a few months of hot news!

- Dave



Dear Dick,

since my other letter I've been sorting out a present for myself, an upgrade to my Amiga, which has now gone in a case with a 60Mhz Pentium.

After re-reading the January issue of OUM, I have sent a separate short letter to Steve Marshall, as I really would like to see an English Oric magazine in print, for at least a little while yet. I am still happy for my name and address to be published for future contact and maybe more of us could pass around our details at the Meet.

As regards the John Hughes suggestion of a constructors corner with Peter Bragg; I would be interested in any information or demo of his Oric 'ROM camera' that was mentioned in an early CEO Mag. Maybe now is the time to sort out the list of suitable projects available in some of the 80s electronic magazines, just in case anybody would be interested. Well, you never know, do you!!

- John Foggin (Ashington)

Dear Foggin,

seeing as we are now on surname terms! No real reply - it's all down to the readers.

- Dave

ON THE NET

ON THE NET WITH Software Warehouse.

Further to my note on page 7 of the last issue, I can now give you an update on the service offered by SOFTNET to their Gold customers. The help-line is now a free service. And every month you can choose one from three software titles worth £20 each absolutely free, and then download them. I've just taken PC-Cillin 98 (virus scanner), and can get free updates.

The server wars is certainly hotting up!!!!!!!!!!!!

It's not really ORIC - but!!!!

I know it's not an Oric topic, and I did say only Oric-related this issue, but I'd only forget by next month. FLOUNDER as mentioned on page 13 last month. Issue 1 was quickly followed by the issues 2 and later number 3. You get it as an e-mail with the text attached in 'Notepad' form.

It's not really 8-Bit, it's hardly computer orientated, but it's a bit of a giggle!!!

To get a free subscription, e-mail flounder@spheroid.demon.co.uk

Here's an example of what to expect:
Computer-related Section:

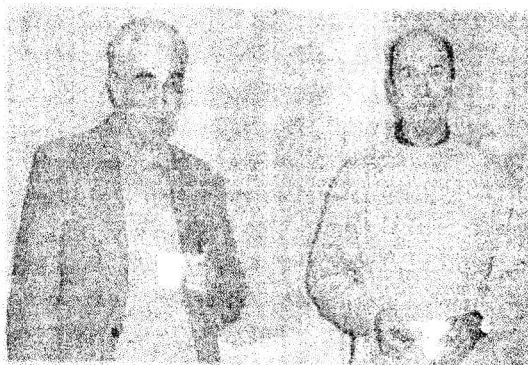
1) We have managed to obtain a small number of the Real Catskin screen-cleaner mitts which were advertised in issue 19 of 8BIT magazine. Simple to put on (simply insert your hand into the hole situated just below the tail) and use. No polish required; contains natural lanolin. Choice of grey, black, or tabby.

2) Best use for an 8-bit computer. Frisbee; yes; doorstep; yes. But why no mention of the greatest practical use for one; skeet shooting?

An 8-bit computer's keyboard stands out really well against the sky as it is spinning, yet it still offers real sport to a marksman because of its distinctive irregular shape and the flashing in flight of its simple range of contrasting keyboard colours. A most satisfying spread of shrapnel can be obtained by a direct hit because of all the tiny little discrete electrical components on the motherboard which fly off so easily when the case is split.

Any model *can* be used but an Amstrad CPC 664, with its distinctive "Cambridge blue on black" keyboard colour scheme gives as much pleasure and satisfaction as picking off a Master of Foxhounds by his pinks at 100 paces.

Roger Barbier and Alain Weber



Meetings on the Net.

The Oric Internet meetings are now a monthly affair starting at 20.00 hrs G.M.T on the first Friday of each month, with a topic to be announced the previous day. For more frequent on-line contact with fellow Oricians you can go through ICQ.

Check out <http://oric.cjb.net> for further details. You'll need to scroll to the bottom of Peacer's Baris Manco page to get to the Oric section, and then choose IRC from the menu.

What a site!

Jonathan Bristow has revamped his web site. The new site supports FRAMES, and includes RHETORIC! Check out: www.twilghte.freemove.co.uk

OUM Site

No real updates to my site for a long time, but it is still receiving its fair share of hits.

As we go to press (gee - is it really April 13th. Already!), there have been 2384 hits the counter was introduced. WINDOWS '95 is top system, with EXPLORER being most favoured browser, though NETSCAPE is also extremely popular. Busiest time is between 7 and 10 p.m., whilst 3 a.m. is the most unfavourite time to pay a visit. Uk residents have made 675 visits, with the French second with 357 visits. Over the past 5 weeks the average is 36 visits per week. The most popular day to visit is Monday (387), and the least popular is Saturday (301).

BITS'n'BOBS

3.5" Drive

Over time I have printed some of your requests for help - some have been resolved, whereas others have fallen on deaf ears. There are others that I may not of answered personally or put into print. Here's an oldie , which I don't think was resolved.

John Hurley has acquired a 3.5" disc drive to add to his system, which already comprises 3" and 3.5" drives. He wants the new uncased unit to become drive C. Unfortunately it has no visible binary toggles or links to enable him to assign it a drive number. He knows it will start and stop on command, but will not talk to the computer.

I know that some drives were made that were factory pre-set to run just as drive B, so perhaps this is the answer. If anyone else has any ideas then they can ring John on: 01935 423194

May OUM

It's obviously gonna be late, but please get your articles to me by April 30th.

New server

Italian reader, Cristiano Bei has changed servers, after changing jobs. He can now be e-mailed at: bei_cristiano@alitalia.it

FOR SALE

Ex- OUM reader Ken Evans from Devon has an Atmos with an original Atmos cased 3 inch >drive, colour monitor, joystick interface etc. Spare Hitachi drive (unused), lots of software. Lots of books on Oric

I am still waiting for Ken to get back to me with his address, but you can telephone him on:

01242 245030

or e-mail him at: ken_s_evans@yahoo.com

THE Y2K CRISIS

Taken from a memo received at a Fortune 500 company -

To: VP, Corporate Administration

I hope I haven't misunderstood your instructions, because this Y to K problem makes no sense to me.

Be that as it may, I have completed the conversion of the corporate calendar for the year 2000, per my understanding of the instructions.

The months now read as follows:

Januark

Februark

March

April

Mak

June

Julk

etc.

Please let me know if there is anything else that needs to be done in preparation for the year 2000.

COMPILER.

The COMPILER from Dr. Ray McLaughlin uses Drive 'A' for most of the time, and I have found that this can cause problems if used often and erasing has to be done many times.

A way to avoid this would be to use TWO disk's, the MASTER and a SLAVE for producing the COMPILED FILES.

FORMAT a disk for Drive 'A' and make it a MASTER/SLAVE. (ie) full system.

Now COPY over from the COMPILER MASTER disk the files ADDRESSES.ASM, LIBRARY.ASM and DATA.ASM as all of these will be used later. (do not add protection)

HERE WE NOW GO INTO THE WORLD OF S P E E D.

BOOTUP the COMPILER master disk in drive 'A'. Select No.1 COMPILER.

When the TITLE PAGE and READY prompt appear, REMOVE the COMPILER MASTER disk. Now is the time to load the BASIC program you wish to compile. The BASIC program should be loaded using the 'J' (join extension). You can load it from drive 'A' or drive 'B' if you have both

If you use drive 'B' then put your newly formatted disk into drive 'A' ready to accept the COMPILED files. If you only have the one disk drive? then you will have to load the BASIC program as before using 'J' and 'N' to inhibit AUTO RUN.

With using drive 'B' for the BASIC program, the COMPILING will start automatically. However, if you use drive 'A' then you will have to enter COMPILE or COMPILE"name.ext" (the name of the Basic program you have just entered).

Lengthy BASIC programs will take some time to COMPILE as two passes have to be made. Don't give up, WAIT.

When completed, the TEXT errors or zero errors appears and the program exits to the READY prompt. Now is the time to view what has been saved to your SLAVE disk. Any files with CODE A, B or C ending with BIN should now be DELETED as they will corrupt the next stage of ASSEMBLY.

Power down when done and RE-BOOT the COMPILER MASTER disk in drive 'A'.

Select No.2 ASSEMBLER. The title screen and DRIVE SELECTION prompt appears.

Remove the COMPILER MASTER and put back in the SLAVE disk with the COMPILED files on. ENTER 'A' for drive selection.

The ASSEMBLER program is AUTO running. Leave until the TEXT message 'ERRORS, or zero Error' appears.

The message 'PRESS any key for MENU' should now be on screen. REMOVE the SLAVE disk and replace it with the COMPILER MASTER disk before you do.

The MENU page will now appear. Select No.3 LINKER to complete the last part of the change.

When the TEXT 'Enter drive letter' appears? remove the COMPILER MASTER for the last time, and put in the SLAVE disk.

Enter 'A' for drive 'A'. All files will now come together to produce the final coded program.

Follow on screen instructions to RUN the new version of your BASIC listing.

You will probably find now that the newly compiled file will be longer than the original?

Don't compile game programs as the new speed will be horrendous. It is best suited to thinking programs like MAXIT, BACKGAMMON, CHESS or any strategy programs against the computer.

Easy ain't it~

- John Hurley

Note from the Editor.

The COMPILER program is still available from OUM on 3" or 3.5" disc for the princely sum of just £3.

Machine Code for the Oric Atmos

(Part 83)

Peter N. Bragg

The Story so Far

We have looked at how to install a computer mouse and interface on the Oric and have reached the software stage. A complete listing was published in part 77 of the series (OUM August 98) and we are now looking at that listing in more detail to see how it works. The last couple of articles described how the data input generated by the mouse being moved, was converted into two values, which were stored in a parameter block, ready for use by the operating system, when it updated the screen display. While we were looking at that, it was noted that there were in fact two lots of data input from the mouse. In addition to the data generated by moving the mouse, there is also data generated by pressing the mouse buttons and this is what we are going to look at here.

The computer mouse should be a parallel port type, preferably with at least two buttons. I use a three button mouse, but it doesn't matter too much as a lot depends on what you intend to do with it. The three buttons on my mouse were originally labelled from left to right, "Execute", "Menu" and "Cancel". It doesn't matter if the buttons on your mouse have different names, because that set of labels is purely arbitrary and I shall be referring to the Buttons simply as Button "E", Button "M" and Button "C".

To recap briefly, the job of sorting "button press" input from "mouse movement" input was done by the JSR 80E0 "Read for Move or Button Pressed" routine which passed the results of its sorting operation on to one of two selected routines, depending on which of the two types of input it found. Mouse movement data is passed on to be dealt with by JSR 8099 and we looked at how that worked, in the last issue. Now we can go on to the other routines which deal with the data input, which is generated by pressing the mouse buttons.

Mouse Buttons Input

The mouse buttons don't generate an interrupt, as was pointed out earlier. This is because it is not essential to deal with button press input straight away. Unlike the input generated by mouse movement, button press input can afford to wait around for a few thousandths of a second, until a normal Oric operating system interrupt comes along to deal with system interrupt requests in general. Our mouse routines have simply tacked the mouse buttons input routine on to the end of the queue where it can wait its turn to be dealt with. You are not likely to notice the fraction of a second delay, between pressing a mouse button and the effect of that being seen. In fact, it is actually necessary to slow down the response to the mouse buttons to prevent repeated signals, from individual button presses, from cluttering up the system. This is because even a short button press of say, half a second, is a long time for a computer system, time enough for many calls to the interrupt system, each of which will see the same short button press as a new button press. As a result, a single very short button press may well register as a dozen or more repeat button presses, which will cause much confusion. The easy way to deal with this problem, is to build a delay into the operation. This will slow down the button repeats and keep the resulting screen updates to reasonable rate. The delay is built into the first part of the routine that we are going to look at now.

"JSR 8210 "Operation Control" (for the mouse buttons)

This is the main routine and every mouse movement and button press, will end up being processed by it. It also calls up and uses a shorter subroutine labelled "Button Operation ?" at JSR 8161. The routine and its subroutine at JSR 8161, deal with all of the output from the mouse and together, call up a set of seven very small routines to produce the appropriate action required. The seven small routines are those listed between locations 8110 and 815C and if you look at the original listing, you will not be surprised to find that four of the seven deal with the four mouse movements and the other three are for each of the three buttons. These seven small routines are the key to attaching your own software, for operation with the Oric computer mouse system.

As we are at present dealing with the Button input and as JSR 8210 is quite a long routine, we will cover the part that deals with the Button input, which is instructions 8210 to 8220 inclusive, plus the subroutine at JSR 8161, in this article and go on to rest of JSR 8210, which deals with the mouse movement in the next article. So now lets start to look at JSR 8210.

"Mouse6"

Read for Move or Button Pressed JSR80E0

```

80E0 20 99 80 JSR 8099    Read/Update if required trackball/mouse move.
                               Button/key status ?
80E3 AD E0 03 LDA 03E0    Fetch IRB/ORB and mask out } Read status of PB5, PB6
80E6 29 E0     AND# "1111" } all but Bits 5, 6 and 7. } and PB7 Lines and then
80E8 8D 01 80 STA 8001    save a copy in Param 8001 (Button input).
80EB 20 10 82 JSR 8210    Test and use input as required.
                               FINISH
80EC 60       RTS         Exit.
    
```

Operation Control JSR8210

```

8210 EE 0E 81 INC 810E    DELAY OPERATION
                               Add 01 to Operation Counter in Param 810E.
8213 AD 0E 81 LDA 810E    Fetch Counter contents into Accumulator.
8216 CD 0F 81 CMP 810F    Test - Is it at count end yet ?
8219 D0 08     BNE "8223"
                               No - so skip Button operation this time.
                               Yes - so now
                               reset the Counter to zero (00).
821B A9 00     LDA# 00
821D 8D 0E 81 STA 810E
-----
8220 20 61 81 JSR 8161    BUTTON OPERATION
                               Test for specific Button pressed.

-----
                               UPDATE CURSOR POSITION - HORIZONTAL and VERTICAL
8223 } Instructions 8223 to 8278 inclusive, update the cursor position
      } on the screen display, using mouse movement input data, which was
      } read and stored in Params 8008/0B, as described in the last issue.
      } (We will look at this display operation, next time.)
8278 }

-----
827B 60       RTS         FINISH
                               Exit.
    
```

Button Operation ? JSR8161

```

8161 AD 01 80 LDA 8001    Fetch Button input from Param 8001 for test.
                               Button pressed ?
8164 C9 E0     CMP# "None" Test - for any Button pressed.
8166 F0 17     BEQ "817F"
                               No Button pressed - so skip to Finish.
                               Yes a Button was pressed - so...
                               Which Button ?
8168 C9 C0     CMP# "E"     Test - Was it Button "E" ?
816A D0 04     BNE "8170"
                               No - so skip to try Button "M".
                               Yes - so...
816C 20 3C 81 JSR 813C    Use "Button E" routine and then...
816F 60       RTS         Exit.

8170 C9 A0     CMP# "M"     Test - Was it Button "M" ?
8172 D0 04     BNE "8178"
                               No - so skip to try Button "C".
                               Yes - so...
8174 20 47 81 JSR 8147    Use "Button M" routine and then...
8177 60       RTS         Exit.
8178 C9 60     CMP# "C"     Test - Was it Button "C" ?
817A D0 03     BNE "817F"
                               No - so skip to Finish anyway.
                               Yes - so...
817C 20 52 81 JSR 8152    Use "Button C" routine and then...
817F 60       RTS         FINISH
                               Exit.
    
```

Param Block 8000

```

8000 FF "█" Page Marker
8001 00 Button Input
8002 00 lsb } Prog Counter Addr
8003 00 msb }
8004 4E lsb } Offset
8005 00 msb }
8006 00 lsb } New Vector Addr
8007 00 msb }
8008 00 Lo { Horizontal }
8009 00 Hi { ("X") } Mouse
800A 00 Lo { Vertical } movement
800B 00 Hi { ("Y") }
800C 4C JMP instruct
800D 00 }
800E 00 } Oric INTSL/IRQ copy
800F 00 }
    
```

Param Block 8100

```

8100 FF "█" Page Marker
8101 00
8102 08 Left } Vertical
8103 09 Right } Movement
8104 0B Up } Horizontal
8105 0A Down }
8106 45 "E" } Button effects
8107 4D "M" }
8108 41 "C" }
8109 00
810A 00 Bit7 set } Mouse input code
810B 00 Bit7 unset }
810C 10 Horizontal } Cursor movement control
810D 10 Vertical }
810E 00 Count } Button rate control
810F 04 Exec }
    
```

Simple Binary Table

0 = 0000	8 = 1000
1 = 0001	9 = 1001
2 = 0010	A = 1010
3 = 0011	B = 1011
4 = 0100	C = 1100
5 = 0101	D = 1101
6 = 0110	E = 1110
7 = 0111	F = 1111

Note - The full listing was published in Part 77 of the series (OUN August 98).

When a button press input has been detected by JSR 80E0, it passes that input straight on to JSR 8210, which controls operation to update the screen display. Most of JSR 8210 is concerned with keeping the cursor position updated on the screen, but it also handles the mouse button operation and provides the above mentioned delay operation.

The first part of the routine, provides the controlled delay, using six instructions, 8210 to 821D inclusive. These six instructions produce the simple counter, using two parameters located in Parameter Block 8100. The delay period is controlled by the value stored in location 810F.

Each time JSR 8210 is called, the first instruction increments (adds 01 to) the counter value in location 810E. The next two instructions (8213/16) then compare that value in 810E to the value held in 810F. If the two values are different, instruction 8219 will ensure that instructions 821B/20 are skipped, which will result in that particular input from the mouse buttons, being ignored. However, if the values in 810E and 810F are the same, the counter (810E) will be reset to zero by instructions 811B/1D and that will be followed by instruction 8120, which will take the button input on to JSR 8161 to be dealt with by that routine.

If you look at the two counter parameters 810E and 810F, you will see that you can control the mouse button repeat delay, by setting the value in 810F. If you double the value in location 810F, you will double the repeat time delay and if you halve the value, you will halve the repeat time delay. At this stage we have reached the point where we can take the mouse button input on to JSR 8161 to sort out which of the buttons was pressed, so now lets have a look at that.

"JSR 8161 "Button Operation ?"

We already have a copy of the Button input, which was saved automatically in Param 8001, when we passed through the JSR 80E0 routine. The first instruction of JSR 8161, fetches a copy of that Button input from Param 8001, into the Accumulator for testing to see which button has been pressed. The Button input is a copy of the contents of the IRB/ORB register (VIA 6522), which is located in #03E0. We are only interested in the top three bits of that value, so the rest of the byte has already been masked out by JSR 80E0 before it saved the copy in Param 8001.

The reason for that ? Each mouse button is connected to a VIA 6522 data pin line. The computer mouse uses lines PB5 for Button "E", PB6 for Button "M" and PB7 for Button "C". These three lines set each of the appropriate bits in IRB/ORB to "1", so if no button is pressed, the Button input, will be 1110 0000 in binary, but of course, we will read that as hex code E0h.

If a button is pressed, the appropriate bit in IRB/ORB (and the Button input copy) will change to "0". So if we press the middle mouse button "M", it will use line PB6 to change IRB/ORB contents in Bit 6 and as a result the Button input copy will become 1010 0000 or hex code A0h.

The other two buttons have a similar effect. Button "E" will produce C0h (1100 0000 bin) and Button "C" produces 60h (0110 0000 bin), when pressed.

So all JSR 8161 needs to do, to find which button was pressed, is to test for those three codes and if you look at the operation of instructions 8168/6F, you will find that those four instructions do just that. Instruction 8168 tests for the value C0h and will cause instruction 816A to skip the next two instructions and go on to the next test at instruction 8170 if it doesn't find that value. On the other hand, if instruction 8168 does find that value C0h, it will go to the specific routine for Button "E", do that and then exit.

The other two operations using instructions 8170/77 and 8178/7F, use exactly the same operation to detect a button press from Buttons "M" or "C". Instructions 8064/66 simply confirm that a button has been pressed and was in fact a bit of an overkill at the time.

If a mouse works with the Oric interface, but uses a different set of values for it's buttons, it can still be used by simply reading the value produced by IRB/ORB, when its buttons are pressed and then changing the three values used in this routine to those used by the mouse currently in use.

So that's how the Buttons are sorted. They don't actually do much when you press them at the present time, but you can use them to call up your own routines and the same applies to the mouse movement routines, which we will be looking at next time.....see you then.



GAMES PROGRAMMING IN ON THE ORIC IN BASIC

OK, so we know that BASIC isn't the best language to program in, but on the ORIC it scores on two fronts. Firstly, everybody that has an ORIC has BASIC to play with. Secondly, it's quite easy to get *something* happening on your screen. It just takes a bit more effort to get reasonable results.

BASIC really is too slow for programming animated games, but if you look at games such as 'Xenon I', 'Don't Panic' and the like, you will find large chunks of BASIC used. Sometimes a BASIC program is used to keep track of what's happening and send the program to different machine code routines. We can use the same sort of program but have the MC bits in BASIC as well. We can also cheat a bit, thanks to the great Dr. Ray, by using his compiler. This can greatly speed up any of our slow programs. What's important for me is that you guys get your hands on the Oric keyboard and have a go. Go on, switch the little bugger on!

This little project was started some time ago. It was shelved for some time, resurrected and abandoned again when I found half of the little programs I'd written had got corrupted. I'm revitalising things again 'cause there might not be the opportunity again - so make the most of it and join in the fun. I hope to get some feed back to include in future articles - your improvements on my duff programming, or an entirely different approach that you have thought of. Hopefully some good will come of it before the end of the publication.

OK, what we are about to do is just delve straight in and chuck out a bit of program for a space invaders type game. This is entirely the wrong way of going about things - but what the heck. If you don't do it wrong you won't know how to do it right!

The first defines some simple characters using the DATA info. Poked in with the FOR/NEXT loop. The strings pair the characters together. Changing from the first set (line 80) to the second (line 90) will give us a simple animation. Lines 110-140 plot in our 'Space Invaders' shields & 140 displays our base. 150-250 is the keyboard control and the last bit is for shooting.

Tap it in and you will see how basic it is. See if you can improve things. We will get critical next time!!!

```
10 CLS
20 FORI=46856T046927:READ CH:POKE I,CH:NEXT
30 DATA 0,1,1,1,1,1,3,7,0,32,32,32,32,32,48,56
40 DATA 7,21,60,44,40,40,0,0,56,42,15,13,5,5,0,0
50 DATA 0,0,0,0,0,1,3,7,0,0,0,0,0,32,48,56
60 DATA 7,21,61,45,41,41,0,0,56,42,47,45,37,37,0,0
70 DATA 255,255,255,255,255,255,255,255
80 AS="  ":B1$="ab":C1$="cd"
90 B2$="ef":C2$="gh"
100 X=3:Y=20
110 PLOT 6,17,"iiii":PLOT16,17,"iiii":PLOT26,17,"iiii"
120 PLOT 6,18,"iiii":PLOT16,18,"iiii":PLOT26,18,"iiii"
130 PLOT 6,19,"i i":PLOT16,19,"i i":PLOT26,19,"i i"
140 PLOTX,Y,B1$:PLOTX,Y+1,C1$
150 REM
160 KPS=KEYS
170 IF KPS=" " THEN ZAP:GOSUB240
180 IF KPS=CHR$(8) AND X>3 AND SCR(X-1,Y)<>105 THEN 181 ELSE 190
181 PLOTX+1,Y," ":PLOTX+1,Y+1," ":X=X-1:GOTO140
190 IF KPS=CHR$(9) AND X<36 AND SCR(X+2,Y)<>105 THEN 191 ELSE 200
191 PLOTX,Y," ":PLOTX,Y+1," ":X=X+1:GOTO140
200 IF KPS=CHR$(10) AND Y<22 THEN Y=Y+1:PLOTX,Y-1," ":GOTO140
210 IF KPS=CHR$(11) AND Y>15 THEN Y=Y-1:PLOTX,Y+2," ":GOTO140
220 IF KPS="" THEN GOTO150
230 GOTO150
240 PLOTX,Y,B2$:PLOTX,Y+1,C2$
250 FORML=Y-2 TO 0 STEP-1
260 PLOTX,ML,".."
270 PLOTX,ML+1," "
280 NEXT
290 PLOTX,Y,B1$:PLOTX,Y+1,C1$
300 EXPLODE
310 RETURN
```

M
11/15

Basically Yours

Page #14

```
40 REM GAME SELECTED :
45 REM GO ENTER PLAYERS NAMES
50 GOSUB 8000
60 REM SET UP NEW BOARD PARAMS etc
70 GOSUB 8200

8000 REM ENTER NAMES
8005 CLS
8010 GOSUB 9800
8020 PRINT:PRINT"Name - Player #1 ";
8030 INPUT P$(1)
8040 CLS
8050 PRINT:PRINT"Name - Player #2 ";
8060 INPUT P$(2)
8065 CLS:PLOT 22,12,1:PLOT 22,14,1
8070 Q$="#1 - "+P$(1):PLOT 23,12,Q$
8080 Q$="#2 - "+P$(2):PLOT 23,14,Q$
8090 CLS:PRINT:PRINT"Selecting Player to go first : "
8100 WAIT 500 :R=INT(RND(1)*2)
8110 S=INT(R+1)
8120 PRINT"PLAYER #";S;" - ";P$(S);" plays first .":WAIT 100
8130 CLS:PRINT:PRINT"SETTING UP PLAY....."
8140 WAIT 300
8150 RETURN
8200 REM NEW BOARD PARAMS ETC
8201 REM UDG'S FIRST
8202 M=#B400+8*(ASC("^")):Q$="12121212121812"
8203 FOR F=1 TO 8:G=VAL(MID$(Q$,F*2-1,2))
8204 POKE M+F-1,G:NEXT F
8205 M=#B400+8*(ASC(">")):Q$="0004026161020400"
8206 FOR F=1 TO 8:G=VAL(MID$(Q$,F*2-1,2))
8207 POKE M+F-1,G:NEXT F
8208 :
8210 MOVES=0
8215 FOR F=1 TO 3
8220 :   FOR G=1 TO 3
8225 :     PS(F,G)=0
8230 :   NEXT G
8240 NEXT F
8245 J1=1:J2=1
8250 AA(1)=6:AA(2)=9:AA(3)=12:AA(4)=2
8260 DA(1)=2:DA(2)=5:DA(3)=8:DA(4)=6
8265 AA(5)=AA(J1):DA(5)=DA(J2)
8270 PLOT AA(5),AA(4),"^"
8280 PLOT DA(5),DA(4),">"
8290 RETURN
```

Basically Yours

Page #15

So what's happening with the programme to date ?

Two issues ago , we wrote the instructions for the screen display , and these were explained at that time .

Last month saw the first user interactivity , by simple choice of seeing instructions , quitting or playing the game . The quit routine is very basic , but suffices (9415) . The instructions were again very simplistic . For this game , instructions are simple and quite few , so it was decided to plot them to the relevant screen area . It would have been nice to have a text window , but Oric's programmers only did a half hearted job , so there is little for us to work with . The code for these instructions is found at 9425-9680 . A separate subroutine was created (9800-9880) to clear the text window , and plot the next colour code (z) .

This issue sees us enter the players names using variables P#(1-2) . Q# is a temporary variable (8000-8080) . Then it is decided who will go first (8090-8120) , using s as the variable . Finally in this issue , we see some UDG's designed (8201-8207) , the grid positions allocated values of 0 , using the variable PS(1-3,1-3) , and variables set for the UDG positions : J1 & J2 are the current row / column positions . AA(1-4) & DA(1-4) are used as actual plot positions , while AA(S) & DA(S) are used to correctly set up for the plotting of the UDG's (AA(S)=AA(J1) & DA(S)=DA(J2) .

Lets hope your following that ! Anyway back to normality next month , with more posers and more listing / explanation .

This issue is dedicated to Dad , who sadly passed away on 16/4/99 , aged just 59 . He will be remembered fondly and greatly missed . Why mention him here - well - I introduced him to home computing , and his great joy for years was an Oric

My Oric Story - John Foggin

I thought it might interest a few in my early Oric history, as a customer with Tangerine Computer Systems. We tend to know them better as ORIC or TANSOFT. But back around 1980 they produced a single board machine called the Microtan 65, one of which I bought, in kit form, along with the full ASCII keyboard and the graphics option. The Microtan cost £79.95, the keyboard £69.95 and the graphics option £7.50

In hind sight I would have been better off buying the lowercase option instead, as I could have got the chips for the graphics option from other sources.

A home built power supply and hexadecimal keypad completed the set up, which did lack certain refinements, such as tape storage.

An interesting side line to the Microtan story was the publishing, in 1980, by an electronics magazine ETI of a single board Space Invasion game to play on your TV.

This was a slightly modified Microtan board with a different 2K ROM. A full circuit diagram was also included. The next month they gave plans for a sound effects board, followed in Jan. 1981 by the details to turn it into the full Microtan.

I never did get the Tanex expansion board as events overtook finances, as they have a habit of doing, and the Oric 1 went on sale.

As the Oric offered more, for less I paid out, as I remember, £139.90 for a 48K computer complete with cassette interface, from John Menzies. The shop has since gone but I still have the receipt.

This came with two, what I would call low density EPROMs, and after seeing the Dec 1994 issue of the Oric owner I loaded up a double density ROM. This was fitted in one of the sockets and a Atmos ROM in the other, with an external switch. This switch was fitted in a small box with a volume control and two push buttons which gave me a warm and a cold reset without turning the Oric over.

As I was working nights for a small firm repairing home computers at this time, an Atmos key board was added next. Then I got an Oric printer, with possible another Oric, and then some where an Atmos disc drive came along with a full Atmos. At this point I had to refit a small I.C. I had removed in my modifications so the disc drive would work. A second bare 3 in drive from a firm called Matmos was connectd up, to make life easier.

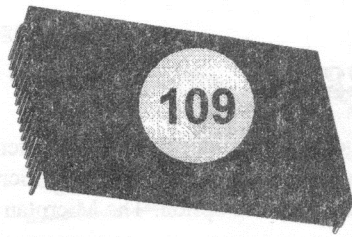
About Jan 1985 I joined the Oric Club and sent off my order, just in time for the company to go bust. I later bought an Amiga!! The company that I bought my 386 board from, went bust, and then I suscribed to OUM, so it's all my fault really. In 1986 I went to Saudi Arabia taking good old Oric with me. It went in a brief case with the power suply and the printer. There it got put into a home made case with a numeric keypad wired across the normal keyboard and connection to a joystick. This was again wired across the keyboard to replicate the cursor keys and return, for fire. The numerics keypad was intended for directional control in a program called The Valley. Also in the same case were two 3in drives and the Oric printer and a thomebrewed power supply powering everything.

The Oric was mainly used in the Middle East for writing back to the U.K. using the built in printer and for producing credits and titles for home videos. initially the programme to use the Oric as a typewriter was produced using just the printer, write a few lines blind, LLIST then rewrite and so on. It made life that little bit easier when I did get a TV.

The Oric and parts survived to come back to England, and it currently lives an an ex-CBM PC case with two 3.5in drives. Power is taken from the PC supply, which also feeds the drives. The output goes via a small interface board to a, EGA monitor which works at the rate scan rate. Inverse video comes at the touch of a switch. It will be interesting to see if the PC keyboard interface from the Statos 2000 could be fitted to the current Oric board. I'll see.!!

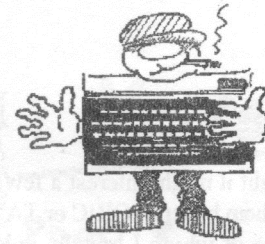
Dave and time permitting, I may send a missive on problems etc that I've had..

OUM - keeping in touch with the grass roots!!!!



RAMBLING

IN THE ROM



Rambling on....

With another dose of the graphics routines written in-house at Tangerine.....

CALCULATE PATTERN AND ADDRESS OF PIXEL

Entry: X and Y contain the horizontal and vertical coordinates of the desired pixel

Exit: #10-#11 contains the address of the six byte block includes the pixel
#0215 contains the mask of the pixel in the seven byte block
A and Y unchanged.

EFA6	CLD	FP49	CLD	not decimal (once more !)
EFA7	PHA	F04A	PHA	save A
EFA8	TYA	F048	TYA	
EFA9	PHA	F04C	PHA	save Y
EFAA	JSR \$F400	F04D	JSR \$F731	calculate 40*A
EFAD	CLC	F050	CLC	
EFAE	ADC #00	F051	ADC #00	very useful ! (because of a label ??)
EFB0	STA 10	F053	STA 10	add #A000
EFB2	TYA	F055	TYA	and #0-#11 contains the start address
EFB3	ADC #A0	F056	ADC #A0	of the requested line
EFB5	STA 11	F058	STA 11	
EFB7	LDA #00	F05A	LDA #00	
EFB9	STA 0D	F05C	STA 0D	
EFBB	STA 0201	F05E	STA 0201	
EFBE	STX 0C	F061	STX 0C	#0C-#0D=X
EFC0	LDA #06	F063	LDA #06	
EFC2	STA 0200	F065	STA 0200	#200-#201=6
EFC5	JSR \$EEFF	F068	JSR \$EFC8	calculate X/6, as the line position
EFC8	CLC	F06B	CLC	
EFC9	LDA 0C	F06C	LDA 0C	take the position
EFCB	ADC 10	F06E	ADC 10	add to the address of the line
EPCD	STA 10	F070	STA 10	obtaining the pixel address
EPCF	LDA #00	F072	LDA #00	echo in the high byte
EFD1	ADC 11	F074	ADC 11	(BCC/INC 11 would have had more effect !)
EFD3	STA 11	F076	STA 11	
EFD5	LDA #20	F078	LDA #20	prepare pattern for pixel to right
EFD7	LDY 0E	F07A	LDY 0E	take the remainder (0 à 5)
EFD9	BEQ EFD9	F07C	BEQ F082	if nul, the position is correct, exit
EFD9	BEQ EFD9	F07C	BEQ F082	
EFDB	LSR A	F07E	LSR A	pattern for point to right
EFDC	DEY	F07F	DEY	remainder -1
EFDD	BCC EFD9	F080	BCC F07C	unconditional: continue...
EFDF	STA 0215	F082	STA 0215	save pattern of the pixel
EFE2	PLA	F085	PLA	
EFE3	TAY	F086	TAY	recover Y
EFE4	PLA	F087	PLA	recover A
EFE5	RTS	F088	RTS	

DISPLACE LOWER

Action: calculate the address and the mask of the point below the current point

EFE6	CLD	not decimal mode
EFE7	CLC	F089	CLC	
EFE8	LDA 10	F08A	LDA 10	
EFEA	ADC #28	F08C	ADC #28	
EFEC	STA 10	F08E	STA 10	address of the six byte block + 40 (next line)
EFEE	LDA 11	F090	BCC F094	
EFF0	ADC #00	F092	INC 11	
EFF2	STA 11	
EFF4	RTS	F094	RTS	

DISPLACE HIGHER

Action: calculate the address and the mask of the point above the current point.

EFF5	CLD	
EFF6	SEC	F095	SEC	
EFF7	LDA 10	F096	LDA 10	
EFF9	SBC #28	F098	SBC #28	
EFFB	STA 10	F09A	STA 10	address of the six byte block + 40 (previous line)
EFFD	LDA 11	F09C	BCS F0A0	
EFFF	SBC #00	F09E	DEC 11	
F001	STA 11	
F003	RTS	F0A0	RTS	

DISPLACE TO THE RIGHT

Action: calculate the address and the mask of the point to the right of the current point

F004	LSR 0215	F0A1	LSR 0215	shift the pattern to the right
F007	BCC F014	F0A4	BCC F0B1	if the pixel doesn't 'exit', that's it
F009	LDA #20	F0A6	LDA #20	
F00B	STA 0215	F0A8	STA 0215	if not, pattern = the leftmost pixel
F00E	INC 10	F0AB	INC 10	but of the next six-byte block
F010	BNE F014	F0AD	BNE F0B1	
F012	INC 11	F0AF		
F014	RTS	F0B1	RTS	

DISPLACE TO THE LEFT

Action: calculate the address and the mask of the point to the left of the current point

F015	ASL 0215	F0B2	ASL 0215	shift the pattern to the left
F018	BIT 0215	F0B5	BIT 0215	and test if the pixel 'exits' to the left
F01B	BVC F02C	F0B8	BVC F0C7	no, exit
F01D	LDA #01	F0BA	LDA #01	
F01F	STA 0215	F0BC	STA 0215	yes, pattern = the rightmost pixel
F022	DEC 10	
F024	LDA 10	
F026	CMP #FF	
F028	BNE F02C	but of the preceding six-byte block
F02A	DEC 11	
.....	F0BF	LDA 10	
.....	F0C1	BNE F0C5	
.....	F0C3	DEC 11	
.....	F0C5	DEC 10	
F02C	RTS	F0C7	RTS	

Jon Haworth
3 Petersfield Road
Duxford
Cambridge
CB2 4SF

Email: jon@cambr.force9.co.uk
Oric Mailing List: oric@lyghtforce.com

April 1st. 1999

Rhetoric Magazine number #01

Edited and Compiled by J.C.Bristow

Menu

The Editorial

Welcome to Rhetoric Magazine, the very first one of many (Hopefully). These pages have been collated with the help of just me at present, hopefully, some Oricians will come forward to offer such help in the near future

This is only a sample at present, and will not properly commence until September. All names within this magazine have been changed to protect the innocent. Remember, do not be afraid to send any stuff to me concerning this Magazine, if this magazine is to continue, then it is up to your contributions alone. I can easily hide your name from these pages, if you wish not to be known on the WEB

So not much to say other than happy Oricing

Send any Rhetoric Contributions to

arc@twilichte.freemove.co.uk

Retro-Reviews

Retro-Reviews is the section that concerns a look back at Oric titles of Yesteryear

The intention is to cover all of the games ever produced for the Oric, with reviews from me or any other member willing to put forward his or her own views on the title.

Reviews/ Previews

This section is intended to review/ preview any new software being developed or just released. If no software is available, it will take a look at software released within the last year that hasn't been seen before here.

Radar Warnings

The Radar Warning is where you can find about new stuff soon to become available and release schedules.

Hints/ Cheats/ Tips

ZipnZap

If a closed door is adjacent to the BUILDER tile, move over it to build, then collect just one tile before moving back across to the closed door. Repeat this procedure for as many tiles as are required to complete the level

Twilights Tip of the Month

Ok, for all you Budding BASIC Programmers out there, another tip for setting up really groovy sounds without the penalty of using up any precious processor time. Uuu... soon!