

ORIC

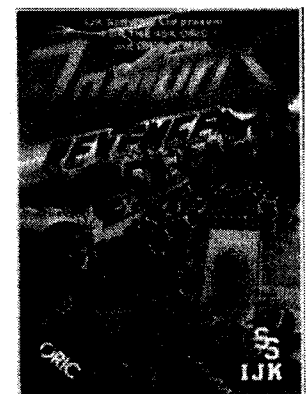
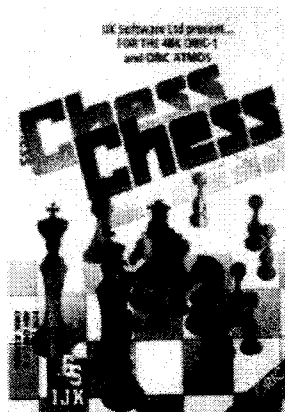
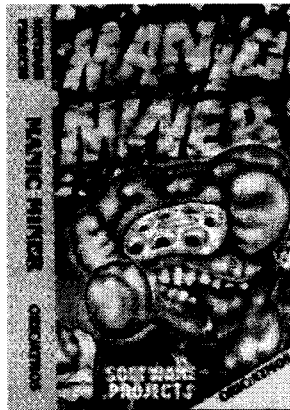
Number **142**

June 1999

USER MONTHLY

*Keeping the
Oric alive*

with Alternative Micros



Just 15 years on (part 7).....

The Editorial

Hi and welcome,

To the June issue, though it will probably be July before you get it.

A fair bit of variety in this issue, thanks to your input.

After much soul searching I have decided to sell off my complete Oric collection. My reasons are various. One is lack of time to ever use it. The other reasons include: a) Not much interest in it, and b) I don't want to go through life hoarding things just for the sake of it, when someone else might put it to better use.

I have decided to sell it all to Steve Marshall. A few of you expressed an interest, but I contacted Steve because I knew he was committed to the Oric, but also was not afraid to use the technology of the Internet and PCs. Therefore any programs he comes across in my vast collection will hopefully become available to each and every one of you.

It is with some sadness that I bow out of Oric-ing, but I will still use Euphoric (if I can get the latest version to bloody well work - if not, then out the window it goes. I have no intention of piddling about and wasting time).

And now (YES - starting with a conjunction) to better things - it's nearly MEET TIME.
See you all there on July 10th.

THE INDEX

Page 1 - THE COVER - by Jon Haworth.....Page 2 - THE EDITORIAL.....Page 3 - NEWS
Page 4 - E-MAILS TO THE EDITOR.....Pages 5/6/7 - M/CODE FOR THE ATMOS (Pt.85) -
Peter Bragg.....Page 8 - BITS'n'BOBS.....Pages 9/10 - RAMROM (Pt.111) with Jon H.
Page 11 -GAMES PROGRAMMING ON THE ORIC IN BASIC - Steve Marshall.
Pages 12/13 - BACK TO THE PAST WITH TANGERINE BASIC (Pt.1) - Fabrice Frances.
Pages 14/15 - AND NOW THE END IS NEAR..... - Francis Bolton
Pages 16/17 - BRIAN's PAGES - The Welsh Wizzard with more posers.

MENU OF THE DAY

Our local chip shop will be doing the catering at the Aylesbury Oric Meet. To help them it would be good of you to let me know your order in advance, either by e-mail, post or phone.

Menu is as follows:

CHIPS - 80p or 1.20.....COD/ROCK - 2.00.....PLAICE/HADDOCK - 2.20
SCAMPI - 3.00.....COD ROE/FISH CAKE - 60p.....JUMBO SAUSAGE - 90p
Steak & Kidney/ Chicken & Mushroom PIE - 1.20.....Quarter CHICKEN - 2.00
CHICKEN BURGER - 1.60.....Qt, HAMBURGER with bun & relish - 1.50
Qt. CHEESEBURGER - 2.00.....MUSHY PEAS or BAKED BEANS or CURRY SAUCE
or GRAVY - 60p.....Pickled Onions - 15p.....Bread & Butter - 25p.....Mixed Salad - 80p
Apple or Banana Fritters - 80p

PAY ON THE DAY.

JULY OUM

As we are again late with this issue, and I am away for nearly a week, I have decide to miss out a July OUM and go for a double issue marked July/Aug for going out on August 1st.

That means double articles from all our writers please.

Last date for inclusion in this bumper issue is July 19th.

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

ORIC MEET

The final Aylesbury ORIC MEET takes place on Saturday July 10th. This is your last reminder, as there will NOT be a July OUM.

The venue for the MEET is the MRCA (Mandeville Residents Community Association) on The Green, Simpson Place, off Harvey Road, Aylesbury. Telephone number of the club is: 01296 424731 - I will be there from about 9 a.m. My mobile number is: 0411 803237. My home number is: 01296 426050. I will be away on the coast from the Sunday until the Thursday of the week leading up to the MEET - so sort yourselves out now.

GRAND RAFFLE

The final Grand Raffle will be held on the day of the MEET. Tickets are £1 each, so send off for some now, if you are not attending. Profits from this will be used to give you a bumper final issue in September, to fund a complete CONTACT LIST for each and every one of you, and to send you an Oric newsletter in January 2000.

MEET EVENING

For those staying on after the MEET, it will be a good chance to have a few beers and generally socialise. It is hoped that Oricians and members of the Social club will get together for a sing-song.

We have a Karaoke machine at the club, and that will be wheeled out and given an airing. Yours truly will give his infamous renditions of HELLO MARY LOU (the old Rick Nelson classic), WHAT DO YOU WANT (whatever happened to Adam Faith), and PEGGY SUE (the even older Buddy Holly classic).

RHETORIC

Firstly the bad news. Although he was certain he was able to attend the Oric meet, sadly Simon Ulliyatt cannot, as his band is performing in Peterboro on the same day.

However, the good news is that we should have the first issue of **Rhetoric** available for the meet, as Jonathan Bristow has agreed to try and shift some!

Rhetoric WILL be going ahead. The price will be £10 per year for 12 issues, some with coverdisks full of Oric stuff (possibly quarterly).

NEW NUMBER

David Goodrum has a new telephone number. It is
01256 410124

OUM - onward until the end!!

e-mails to the Editor

Hi Steve/Dave,

I would have been in touch earlier, but I have been unable to log onto freeserve for about 2 months. Now I have managed to get back in.

I have not got any strong views on the future of the Oric User Group. I still retain my Oric computers and accessories, and will continue to do so. However, they are rarely used. Euphoric gets used a little more often, but still not much, just for the odd 10 minute games play.

If enough people, even perhaps a dozen, want to continue to support the Oric then I see no reason why they should not do so. If a user group of some kind continues to exist then I would probably support it, although my input would be minimal. My preference would be for communication electronically, although I know this would not be possible for all, so would probably support a snail-mail system, but it depends on the cost, quality, relevance etc. I must confess that the Oric comes well down my list of priorities these days.

I hope these views are helpful to you in deciding on what to do.

Regards,

Trevor Shaw

Hi Dave,

I am actually in England at the beginning of July, wrestling as last year. However, my flight back to Norway is on the 10th July. I am sorry that I am not able to attend this year's meeting.

Anyway, I am very happy that I was able to attend last year's meeting. That was a good experience, and it was nice to meet David Wilkin, Peter Bragg, Peter Thornburn and you again. Not to mention all the other

people I met, such as Jon Bristow, Jon Haworth and all the other people, whose names have been mentioned several times in OUM.

Good luck and happy "retirement"!!!

Best wishes, Arnt Erik Isaksen
(Norway)

Hi Dave,

I e-mailed you a few months back and only learned later that OUM was going to bow out of the scene. I was gutted to hear this as I was thinking about re-subscribing. Is this due to falling subscription numbers or it being too much hassle to carry on with? The number of active subscribers must be fairly low now - I say this while knocking this out on a laptop.. still have my Atmos though.

Anyway just a quick note to say I thought you did a marvellous job as Editor of OUM, always managing to keep things interesting. Still, I've four years of back-issues to reminisce with. Sigh.

All the best to you and yours.

Denis Bonfield

Hello there,

I had no idea this great little machine was still being supported. It has made me get my Oric 1 out of the loft and fire it up for the first time in about 10 years!

I noticed in your cheat & tip section about disabling the autorun feature. How is this done?

Is it easy to connect an Oric to a Pc so i can download emulated software and useit on the real meccoy! Many thanks

Tony Hulkes

Hello Dave,

I'm happy to renew my OUM subscription (for the last time unfortunately...) for two reason: First of all its a good reading (even if I sometimes get some understanding difficulties) Secondly because it allows me to use very old "State of Guernsey" Pounds I got here for more than 10 years, from the times I lived in Cherbourg and was used to spend some Week-ends in the " Iles anglo-Normandes" :-))

btw: thank you for the great job you did during so many years as OUM editor.

best regards,

Dominique Pessan

Hi,

I'm searching for an Atmos with disk drive and a lot of tape/disk (prevalently tapes for my poor old real Oric1) Oric software (in Italy it is impossible to found!) for sale.

Can somebody help me?

Thanks all.- Cristiano Bei

...I'm building a new Oric site at

<http://skyscraper.fortunecity.com/daisywhell/46/index.html>

..work in progress!

Machine Code for the Oric Atmos

(Part 85)

Peter N. Bragg

The Story so far

We have been looking at how to install a computer mouse and interface on the Oric and have dealt in detail with the hardware and are now coming to the end of the software analysis. A complete listing was published in part 77 of the series (OUM August 98). The last couple of articles looked at the working of the largest routine in the mouse software, JSR 8210 "Operation Control" and its subroutine JSR 8161 "Button Operation?" which together, use data produced from the mouse input to control and update the screen display.

As mentioned at the end of the last article, that JSR 8210/JSR 8161 combination control the screen display, using seven short routines. Four of these control the cursor movement in the horizontal (left/right) and vertical (up/down) directions, so that the cursor on the screen mimics the mouse movement on its mat.

The other three routines are allocated, to the three mouse buttons. Each button produces a slightly different effect when pressed and the three routines produce those effects, one for each button.

The seven control routines are located in the software listing in area, #8110 to #815C inclusive and are each labelled according to their function. So let's have a look at those seven routines now.

The Seven Cursor and Button Effect routines

If you look at all of the seven routines in turn, you will soon notice that they are all nearly identical in listing and operation. This applies to both the button effects and cursor movement operations. In fact, the only difference between each of the seven routines, lies in the third instruction in each routine. This third instruction picks up a specific code from Parameter Block 8100 and puts it into Register "X". In each case, that is followed by a call in the fourth instruction to routine, JSR 8280 "Enter Mouse Code". The rest of the instructions in each of the seven routines, simply preserve the original contents of Register "X" on the Stack, so that each routine is "transparent" and Register "X" has the same contents on exit from the routine as it had when it entered the routine.

So let's look at the first of the seven routines, which is JSR 8110 "Move Left". The first two instructions preserve "X" on the Stack and then instruction 8112 fetches a copy of the code in Parameter 8102 and puts it into Register "X". That particular code is 08hex, which is a control code which also happens to be the same code that is produced when the Left (Arrow) Cursor key is pressed. The next instruction 8115 then calls the "Enter Mouse Code" routine at JSR 8280, which inserts the code into the Oric's keyboard buffer. The effect of that action is exactly the same as if you had pressed the Left Cursor key yourself. The remaining three instructions of JSR 8110 simply retrieve the original contents of Register "X" and then exit at instruction 811A.

Hopefully, you can now see that the effect of calling JSR 8110 "Move Left" is exactly the same as if you had pressed the Left Cursor key. So when you move the mouse sufficiently far enough to the left, it will cause JSR 8210 "Operation Control" routine to call up JSR 8110 "Move Left" routine, which will then move the cursor to the left to match your movement of the mouse.

If you now look at JSR 811B "Move Right", you will see that it is exactly the same operation again as that in JSR 8110. The only difference is that the third instruction at 811D, will now fetch the code from Parameter 8103. This code is 09hex, which is the code also produced by pressing the Right Cursor key. So once again, the effect of calling JSR 811B "Move Right" is exactly the same as pressing the Right Cursor key and of course the same applies to the next two routines JSRs 8126 and 8131, which move the cursor up and down respectively, in the same way, when they are called by the appropriate mouse movements.

This now brings us to the three Button routines, JSRs 813C, 8147 and 8152. In fact, these three also use the identical operation to that used by the four cursor movement routines and as you might guess pressing Button "E" simply prints an "E" on the screen, or a line of them if you hold the Button down. There, I did warn you that the Buttons did nothing much of interest.

For various reasons connected to further development, which hasn't happened as yet, Button "C" called up JSR 81C0 at instruction 817C in the original listing and as a result, doesn't appear to do much if that listing is used. However, assuming that you have a three button mouse and restore instruction 817C to call up JSR 8152 as listed in the last couple of articles it will operate in the same way as the other two Buttons and will print the letter "C" on the screen. Another small point. You may have noticed that the first instruction in each of the seven routines is "TXA/PHA". Strictly speaking, this is two instructions and like the tail end pair, "PLA/TAX" is simply written like that to save 14 lines of listing!

This leaves us with just one routine to explain, which is the Keyboard Buffer routine, JSR 8280 "Enter Mouse Code". This routine as we have seen, is called up by each of the seven "Cursor and Button Effect" routines. There is not much to it. The first instruction preserves the contents of the Status Register and the second saves a copy of the contents of Register "X". The Oric's operating system requires Bit 7 to be set to "1", before any code is put into the Keyboard Buffer and so the code in Register "X" is copied into the Accumulator by instruction 8284 and instruction 8285 sets Bit 7 using an "ORA" instruction. Instruction 8287 saves a copy of the modified code and instruction 828A puts another copy of it into the Keyboard Buffer at #02DF. That in effect completes the operation and the routine is then wound up by clearing Register "X" and retrieving the contents of the Status Register, before making an exit at instruction 8290.

Oric

12 Jan 91

"M" OUN 6

Param Block 8100	
8100 FF	"█" Page Marker
8101 00	
8102 08	Left
8103 09	Right
8104 0B	Up
8105 0A	Down
8106 45	"E"
8107 4D	"M"
8108 41	"C"
8109 00	
810A 00	Bit7 set
810B 00	Bit7 unset
810C 10	Horizontal
810D 10	Vertical
810E 00	Count
810F 04	Exec

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

			Move Left JSR 8110
8110	8A 48	TXA/PHA	Preserve contents of Index X
8112	AE 02 81	LDX 8102	Fetch item from Param 8102
8115	20 80 82	JSR 8280	and display it, via the keyboard buffer. FINISH
			Move Right JSR 811B
811B	8A 48	TXA/PHA	Preserve contents of Index X
811D	AE 03 81	LDX 8103	Fetch item from Param 8103
8120	20 80 82	JSR 8280	and display it, via the keyboard buffer. FINISH
			Move Up JSR 8126
8126	8A 48	TXA/PHA	Preserve contents of Index X
8128	AE 04 81	LDX 8104	Fetch item from Param 8104
812B	20 80 82	JSR 8280	and display it, via the keyboard buffer. FINISH
			Move Down JSR 8131
8131	8A 48	TXA/PHA	Preserve contents of Index X
8133	AE 05 81	LDX 8105	Fetch item from Param 8105
8136	20 80 82	JSR 8280	and display it, via the keyboard buffer. FINISH
			Button "E" JSR 813C
813C	8A 48	TXA/PHA	Preserve contents of Index X
813E	AE 06 81	LDX 8106	Fetch item from Param 8106
8141	20 80 82	JSR 8280	and display it, via the keyboard buffer. FINISH
			Button "M" JSR 8147
8147	8A 48	TXA/PHA	Preserve contents of Index X
8149	AE 07 81	LDX 8107	Fetch item from Param 8107
814C	20 80 82	JSR 8280	and display it, via the keyboard buffer. FINISH
			Button "C" JSR 8152
8152	8A 48	TXA/PHA	Preserve contents of Index X
8154	AE 08 81	LDX 8108	Fetch item from Param 8108
8157	20 80 82	JSR 8280	and display it, via the keyboard buffer. FINISH
			Enter Mouse Code JSR 8280
			Enter with item already in Register "X"
8280	08	PHP	Preserve contents of the Status register.
8281	8E 0B 81	LDX 810B	Save a copy of the Register "X" item.
8284	8A	TXA	Now copy Register "X" item to Accumulator.
8285	09 80	ORA# "1"-----	Set Bit 7 in the Accumulator item.
8287	8D 0A 81	LDX 810A	and save a copy of the result
828A	8D DF 02	STA 02DF	and then put a copy into the keyboard buffer. FINISH
828D	A2 00	LDX# 00	Clear Register "X" to zero and then
828F	28	PLP	retrieve contents of the Status register
8290	60	RTS	Exit.

			Enter Mouse Code JSR 8280
			Enter with item already in Register "X"
8280	08	PHP	Preserve contents of the Status register.
8281	8E 0B 81	LDX 810B	Save a copy of the Register "X" item.
8284	8A	TXA	Now copy Register "X" item to Accumulator.
8285	09 80	ORA# "1"-----	Set Bit 7 in the Accumulator item.
8287	8D 0A 81	LDX 810A	and save a copy of the result
828A	8D DF 02	STA 02DF	and then put a copy into the keyboard buffer. FINISH
828D	A2 00	LDX# 00	Clear Register "X" to zero and then
828F	28	PLP	retrieve contents of the Status register
8290	60	RTS	Exit.

I have to say at this point that several of those instructions are not essential to the operation of JSR 8280 "Enter Mouse Code" and were put in for the purpose of future development. The essential instructions are 8280 and 828F, which preserve and retrieve the contents of the Status Register and instructions 8284 and 8285, which set Bit 7 and of course 828A, which puts the modified code into the Keyboard Buffer, before leaving the routine, via instructions 828F and 8290. That completes the description of the listing for the computer mouse software.

So where do we go from here ?

It has taken quite time to describe both the hardware and software in such detail, for the Oric computer mouse and even if you never put one together for yourself, I hope that it has provided a useful insight into the technique of writing software and constructing and setting up hardware. That was the main aim behind the project. I should like to point out that constructing the hardware and tapping in the software was far easier and a lot quicker than writing up the description for these articles, so don't be put off by the length of the description.

The articles on the computer mouse start from Part 66 in the series and commenced with a description of the hardware and operation and then went on to describe the construction of an interface for the Oric computer mouse. The essential software for the mouse was listed in Part 77 of the series and a detailed description of that listing has been the main subject this series of articles since then. If you have all the OUM magazines from issue 119, to the current issue, they should hopefully provide you with enough information to produce a computer mouse for your Oric.

As for the Oric mouse system as described, I am aware that moving the text cursor and printing a few letters across the screen may not seem very ambitious, but it does give you all the essential basics of a computer mouse system for the Oric. Think about any computer mouse system for a moment. It produces a very simple end result compared with say, a game or a piece of office software. All it needs to do, is to produce four movements on command, which are specifically, left, right, up and down, plus any button effects on command, of which there are usually three at the most. As a result, it is fairly simple to connect up your own software to the system described in this series, because you only have to change the address of a maximum of seven simple calls in order to link it up to your own mouse software. All seven of this system's calls and their addresses are listed here below and if you write any routines that make use of the mouse movements and/or buttons, you only need to change the appropriate instructions to point to your own routines for that particular mouse function. As a result, it should be fairly easy to improve the system, so that the mouse can be used to control other things such as the Hires screen for example.

Software Links to the Mouse system

These are the instructions that call the seven routines, described above and they are the key to linking up any software to the Oric mouse system described in this series. The seven routines are called up by instructions located at addresses, 8236, 8244, 8262 and 8270 for cursor movement and at addresses, 816C, 8174 and 817C for the Button effects. These are the most useful of all the instructions to remember, because you only need to change the calls in those instructions in order to make them call up your own routines instead.

Possible Uses

Now let's push a few ideas about. For example, if you wish to use the mouse Button "E" as a "Fire Button" in a game, it would probably be best to change instruction 816C, so that it calls your own "Fire" routine. Likewise, the same applies if you wish to use the mouse as a joystick. You should be able to link the mouse up, by using the four instructions 8236, 8244, 8262 and 8270 to call up your own movement routines. It should also be possible to modify some of the key presses in comercial software, so that you can use the mouse as well as the keyboard.

My own ideas involved fitting a low resolution camera on to a simple stand, which would be driven by two servos, to provide pan and tilt, all of which would also be controlled by the mouse. I got most of the way there, having produced both a successful camera system, which was operated using the same VIA 6522 Expansion Unit and of course, the basic mouse system. I even have the servos, however committments and time available have not been sufficient, to put the whole lot together up to now. Since Dave announced the end of OUM, a lot has happened and I now have a lot of new projects in the pipeline. The changes at Acorn have also have had a big effect and as a result, the Oric part of my own computer system has seen very little use in the last few months. I am not too sure if I will get around to Oric camera and mouse system in the near future. However, I have no plans to retire the Oric at the present time and it will remain set up and ready to go, as part of my computer system for the foreseeable future. Next month, I aim to have a look at a "crib card" for the 6502 instruction set. See you then.



Bits'n'Bobs

PROGRAM ODDITY

The following short program seems to highlight an oddity:-

```
20 REPEAT:REPEAT
30 GET A:PRINT "A is A";
35 IF A=1 THEN PRINT " ":PING
50 UNTIL A<>1
60 PRINT "Wrong Number":UNTIL X=1
```

Run the program and pressing any number key except 1 will result in printing the number followed by 'Wrong number', pressing 1 will result in printing the number only and of course pressing any letter key will result in the error message 'TYPE MISMATCH ERROR' ??? not so!! If you press E then the result is 'E=0 Wrong Number'

Perhaps Dr.Ray can shed light on this anomaly from Arthur Crawford.

OUM DISC 7

I am afraid I cannot see it happening. Anyone who has paid - please ask for a refund.

FOR SALE

Ex-OUM reader Ken Evans from Cheltenham has some Oric goodies for sale. Prices are as follows:

3" Microdisc system + Atmos + 20 disks of software + joystick interface - £50

As above + 12" Tandata colour monitor - £60

Various Oric books are also available at £5 the lot, and include THE Geoff Phillips book.

All the above options are subject to an addition for postage at cost, or buyer picks it up from Ken's.

Also available is the system of the late Jack Lupton as advertised on page 8 of the May OUM.

Prices as follows:

Atmos + 3" Microdisc system + tapes + cassettes + Oric-1 - £40 + postage at cost.

As above + MCP 40 - £50 + postage

As above + Citizen printer - £60 + postage

The above systems are subject to Cristiano Bei deciding which if any he is having.

For the late Jack's system please call me or his daughter Jeanette on: 01928 718733

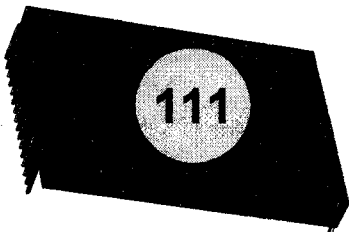
Ken Evans is on: 01242 245030..... e-mail : ken_s_evans@yahoo.com

WANTED

New contact Paul Donegan is on the look out for an Oric-1.

Contact me or e-mail Paul at: genfrank@btinternet.com

SEE YOU AT THE MEET



RAMBLING IN THE ROM



Rambling on....

With the display routines.....

'CHAR' (COMMAND)

Entry: #2E1-#2E2 contains the ASCII code of the character to display
#2E3-#2E4 contains the number of the character set
#2E5-#2E6 contains the FB code

Exit: #2E0 incremented if parameter error

F0A5	LDX 02E2	F12D	LDX 02E2	if code too large or negative
F0A8	BNE F0E7	F130	BNE F16D	error
F0AA	LDX 02E1	F132	LDX 02E1	take required code
F0AD	CPX #20	F135	CPX #20	if control code, error
F0AF	BCC F0E7	F137	BCC F16D	
F0B1	CPX #80	F139	CPX #80	if >128
F0B3	BCS F0E7	F13B	BCS F16D	error also
F0B5	LDA #02	F13D	LDA #02	
F0B7	LDX #E3	F13F	LDX #E3	index #2E3
F0B9	JSR \$F264	F141	JSR \$F2F8	character set 0 or 1 ?
F0BC	BCS F0E7	F144	BCS F16D	no, error
F0BE	LDA #04	F146	LDA #04	
F0C0	LDX #E5	F148	LDX #E5	index #2E5
F0C2	JSR \$F264	F14A	JSR \$F2F8	FB code less than 4 ?
F0C5	BCS F0E7	F14D	BCS F16D	no, error
F0C7	LDA 0219	F14F	LDA 0219	take horizontal coordinate
F0CA	CMP #E9	F152	CMP #EB	compare to 235 (240-6 (character width))
F0CC	BCS F0E7	F154	BCS F16D	if above, error
F0CE	LDA 021A	F156	LDA 021A	vertical coordinate
F0D1	CMP #C1	F159	CMP #C1	compare to 193 (200-8 (character height))
F0D3	BCS F0E7	F15B	BCS F16D	jump if error
F0D5	JSR \$F0EB	F15D	JSR \$F171	calculate character address
F0D8	JSR \$F115	F160	JSR \$F19B	display the character
F0DB	LDX 0219	F163	LDX 0219	
F0DE	LDY 021A	F166	LDY 021A	take cursor coordinates
F0E1	JSR \$EFA6	F169	JSR \$F049	recover address and shape of cursor
F0E4	JMP \$F0EA	F16C	RTS	
F0E7	INC 02E0	F16D	INC 02E0	
F0EA	RTS	F170	RTS	

CALCULATE ADDRESS OF A CHARACTER

F0EB	CLD	F171	CLD	
F0EC	LDA 02E5	F172	LDA 02E5	take FB code
F0EF	STA 0212	F175	STA 0212	and save
F0F2	JSR \$EDED	F178	JSR \$EDED	and adjust FB code
F0F5	LDA 02E1	F17B	LDA 02E1	take ASCII code

F0F8	STA 0C	F17E	STA 0C	and save
F0FA	LDA #00	F180	LDA #00	high byte = 0
F0FC	STA 0D	F182	STA 0D	
F0FE	LDX #03	F184	LDX #03	3 shifts = *8
F100	ASL 0C	F186	ASL 0C	
F102	ROL 0D	F188	ROL 0D	
F104	DEX	F18A	DEX	calculate displacement in table
F105	BNE F100	F18B	BNE F186	(code * 8)
F107	LDA 02E3	F18D	LDA 02E3	take number
F10A	ASL A	F190	ASL A	*2 = 0 or 2
F10B	ASL A	F191	ASL A	*4 = 0 or 4 (high byte, so #0000 or #0400)
F10C	CLC	F192	CLC	
F10D	ADC #98	F193	ADC #98	add to table base address
F10F	CLC	F195	CLC	
F110	ADC 0D	F196	ADC 0D	and to displacement (low byte unchanged)
F112	STA 0D	F198	STA 0D	
F114	RTS	F19A	RTS	

DISPLAY A CHARACTER

Entry: #0C-#0D contains its address in the character table
Exit: nothing special

.....	F19B	CLD	
F115	LDY #00	F19C	LDY #00	
F117	STY 0F	F19E	STY 0F	indicate first byte of the definition
F119	LDA (0C), Y	F1A0	LDA (0C), Y	take character shape
F11B	STA 0E	F1A2	STA 0E	and save
F11D	JSR \$F2C3	F1A4	JSR \$F35D	save cursor shape
F120	ROL 0E	F1A7	ROL 0E	
F122	ROL 0E	F1A9	ROL 0E	shift (avoiding b7 and b6 pointless)
F124	LDX #06	F1AB	LDX #06	there are 6 pixels per byte
F126	ROL 0E	F1AD	ROL 0E	take out a pixel
F128	BCC F12D	F1AF	BCC F1B4	if 0, do nothing
F12A	JSR \$EF5B	F1B1	JSR \$F024	display the pixel in the pattern
F12D	JSR \$ F004	F1B4	JSR \$F0A1	and shift to the right
F130	DEX	F1B7	DEX	for the 6 pixels
F131	BNE F126	F1B8	BNE F1AD	
F133	JSR \$F2D4	F1BA	JSR \$F36E	recover shape of first pixel
F136	JSR \$ EFE6	F1BD	JSR \$F089	shift lower
F139	LDY 0F	F1C0	LDY 0F	take number of byte
F13B	INY	F1C2	INY	prepare the next
F13C	CPY #08	F1C3	CPY #08	is it the last?
F13E	BNE F117	F1C5	BNE F19E	no, start again
F140	RTS	F1C7	RTS	

See you all at the Meet.....

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

Jon Haworth
3 Petersfield Road
Duxford
Cambridge
CB2 4SF

GAMES PROGRAMMING IN ON THE ORIC IN BASIC

```

10 CLS
20 FORI=46856TO46895:READ CH:POKE I,CH:NEXT
30 DATA 1,1,1,3,23,61,40,40,32,32,32,48,58,47,5,5
40 DATA 0,0,1,3,23,61,41,41,0,0,32,48,58,47,37,37
70 DATA 255,255,255,255,255,255,255,255
80 A$=" ":B1$="ab":B2$="cd"
100 X=3:Y=20
110 PLOT 6,17,"eeeeee":PLOT16,17,"eeeeee":PLOT26,17,"eeeeee"
120 PLOT 6,18,"eeeeee":PLOT16,18,"eeeeee":PLOT26,18,"eeeeee"
130 PLOT 6,19,"e e":PLOT16,19,"e e":PLOT26,19,"e e"
140 PLOTX,Y,B1$
150 REM
160 KPS=KEYS
200 XM=20:YM=5:YR=YM*0.9:XR=XM*0.6
210 AA=INT(RND(1)*3)+1:BB=INT(RND(1)*3)+1
220 GF=0
230 REM ----- REPEAT ----
231 KPS=KEYS
235 PLOTX,Y,B1$
240 GF=GF+0.05
250 P=XM+XR*SIN(GF*AA)*SIN(7*GF)
260 Q=YM+YR*SIN(GF*BB)*SIN(7*GF)
270 PLOTXX,YY," ":PLOT P,Q,"0":XX=P:YY=Q
280 IFKPS=""THEN230ELSE510
300 GOTO230
500 REM --- KEY PRESS ----
510 IF KPS="" THENZAP:GOSUB620
520 IFKPS=CHR$(8)ANDX>3ANDSCRN(X-1,Y)<>101THEN530ELSE540
530 PLOTX+1,Y," ":X=X-1:GOTO230
540 IFKPS=CHR$(9)ANDX<36ANDSCRN(X+2,Y)<>101THEN550ELSE560
550 PLOTX,Y," ":X=X+1:GOTO230
560 IFKPS=CHR$(10)ANDY<22ANDSCRN(X,Y+1)<>101THEN570ELSE580
570 PLOTX,Y,AS:Y=Y+1:GOTO230
580 IFKPS=CHR$(11)ANDY>15ANDSCRN(X,Y-1)<>101THEN590ELSE600
590 PLOTX,Y,AS:Y=Y-1:GOTO230
600 REM
610 GOTO230
620 PLOTX,Y,B2$
630 ML=Y-1
640 REPEAT
650 ML=ML-1
660 PLOTX,ML,".."
670 PLOTX,ML+1," "
680 UNTILSCRN(X,ML-1)<>32ORML=1
690 IFSCRN(X,ML-1)=79ORSCRN(X+1,ML-1)=79THENEXPLODE:PLOTX,ML-1,"*"
700 PLOTX,ML," "
750 RETURN

```

Back to the past with Tangerine Basic (part 1)

Looking inside a code reveals more than a simple sequence of instructions. This is where programming reaches other forms of intellectual processes known as Arts. Right, the Art of Programming... Like painting or music, you can say a piece of code is either dull, beautiful or ugly, and sometimes you can even shout "What a genius!". I hope you will agree each author has its own style and this style can easily be seen in a program. Last but not least, I would also say some programs exhibit another dimension: time. Like a cathedral which has been build over several centuries, and which shows evidence of the different eras, a program can tell its history and tell us about the people who worked on it...

That's what I discovered when looking at Microtan 65's BASIC, from which our familiar Tangerine BASIC 1.0 and 1.1 emerged. You may remember I was wondering at some time if Tangerine really had the Microsoft BASIC sources. This interrogation came from the fact some parts of Oric-1's BASIC really look like patched from a binary dump, and not cleanly assembled. For example, everyone can ask why the routine responsible for getting the next token in a program is split in two parts, one part in zero page at location \$E2 and the second part in ROM. This routine is heavily called during execution, so it would be fair it is as fast as possible. We can imagine there was not enough room in zero page for such a routine, but nevertheless, the JSR at the end of it really looks like an ugly patch. This is rather surprising when you see how optimised some other parts of the code are. In the excellent book "L'oric à nu", there are few places where Fabrice Broche flags Microsoft code snippets as "badly optimised". The most evident ones lie in the floating point routines, when a group of six instructions could be replaced by a single rotate instruction at several locations. All these suspect codes can be explained when tracing through the code back the origins...

Looking at Microtan 65's BASIC is really interesting, because it reveals not only what Tangerine changed to Microtan 65's BASIC in order to produce Tangerine BASIC 1.0, but also what Tangerine changed to Microsoft BASIC65 in order to have Microtan's BASIC. Finally, it also gives some information about the own evolution of BASIC65 at Microsoft's. Let's go there, travelling through time a step at a time...

When comparing Microtan's BASIC and Oric's BASIC, it becomes obvious Tangerine really had the Microsoft sources when developing Oric's BASIC. This was confirmed by Peter Halford recently, who remembers the smart (and sometimes humorous) comments in it. Insertions in a program are not easy to do when you don't have the sources, because you not only have to move the code but also to change the operands of many instructions. There are many insertions in Tangerine BASIC 1.0, of course new commands have been inserted and not only appended to the end of code, but also individual instructions have been inserted in many places. Mostly, these instructions play with flags in page 2 : Microtan's Basic doesn't use variables in page 2, how could it do since page 2 is screen memory ? As for new commands, here are the tokens added in Tangerine BASIC 1.0 :

EDIT, TRON, TROFF, HIMEM, GRAB, RELEASE,
 POP, PULL, REPEAT, UNTIL, ELSE,
 TEXT, HIRES, LORES, CLS, @, PLOT, SCRN, POINT,
 INK, PAPER, INVERSE, NORMAL,
 LLIST, LPRINT,

MUSIC, PLAY, SOUND, SHOOT, ZAP, PING, EXPLODE,
 PATTERN, DRAW, FILL, CIRCLE, CHAR, CURSET, CURMOV,
 CALL, !, &, AUTO,
 KEYS\$, HEX\$, DEEK, DOKE, LN, PI, TRUE, FALSE

As you can see, these keywords mainly deal with graphics and sound; very few of these commands have an interaction with the heart of the BASIC interpreter. The most noticeable exceptions have all led to bugs in Tangerine BASIC 1.0 : remember the bugs related to the much-desired ELSE keyword, or the interpretation of hexadecimal values. Does this work justify the renaming of Microsoft BASIC to Tangerine BASIC ? The core of the BASIC did not change at all (except the addition of the ELSE and the quote, which both were bugged). Sure, the tokens recognised by Microtan's BASIC were rather minimal, and would not have attracted customers of a colourful / soundful home computer :

END, FOR, NEXT, DATA, INPUT, DIM, READ, LET, GOTO, RUN, IF, RESTORE,
 GOSUB, RETURN, REM, STOP, ON, NULL, WAIT, LOAD, SAVE, DEF, POKE, PRINT,
 CONT, LIST, CLEAR, GET, NEW,
 TAB(, TO, FN, SPC(, THEN, NOT, STEP, +, -, *, /, ^,
 AND, OR, >, =, <, SGN, INT, ABS, USR, FRE, POS, SQR,
 RND, LOG, EXP, COS, SIN, TAN, ATN, PEEK, LEN, STR\$, VAL, ASC,
 CHR\$, LEFT\$, RIGHT\$, MID\$, GO

Here are some comments about these keywords:

NULL defines the number of null ASCII chars to be sent with carriage return (in these days, output peripherals often need time to do a carriage return (think of a printer))

WAIT waits for bits changes at a memory location (usually an IO port)

LOG is the natural logarithm. With the Oric-1, LOG was renamed LN, and a decimal LOG was introduced (this only requires a call to the natural logarithm followed by a call to the division)

GO TO written as two words is an accepted syntax. It was either forgotten by Tangerine or intentionnally dismissed, but then, why did the GO keyword survive ?

STR\$ is not bugged: first char of positive numbers is a space (\$20), it became \$02 with the Oric-1. This is rather strange error, which could make us believe Tangerine typed the Microsoft BASIC from a source listing, but Peter Halford mentioned the source was on floppy too...

So, again, I let you judge if the part added by Tangerine to the Microsoft core justify the removal of the Microsoft copyright, and the removal of the string "WRITTEN BY WEILAND AND GATES" which can be still read in Microtan's BASIC. Instead, you all know the famous "Software by Peter Halford and Andy Brown". Well, asked about that point, Peter Halford admitted he was only 17 at that time...

What I am not sure about, in the other hand, is whether Microsoft's source code was exhibiting the hidden signature or instead masking it... Fabrice Broche discovered these MICROSOFT characters spelled backwards with the most significant bits changed so that they look like two 5-bytes floating point values, as if they were two of the sine polynomial coefficients. I would say the source code was masking these bytes, otherwise there are great chances they would have disappeared. There are two other interesting things about this signature, I will talk about one of them next time, but the first one is that it is different on the Apple2 BASIC. This sounds like a serial number, could it have been a way for Microsoft to track illegal copies ? Well, next time, we will step back from Microtan's BASIC to Microsoft's BASIC65.

Cheers, **Fabrice Frances**

And now, the end is near I did it MY way. (FRANK BOLTON, - bowing out.)

As this will almost certainly be my last contribution to OUM I am using small print and asking Dave to give me both sides of a page to make my farewell, instead of the usual one side. I would like to thank all the people who have read my articles and been kind enough to express their interest and satisfaction. I would also thank the people who have sent in questions and queries, often questioning my judgement (and my grammar). The whole point about communication is that we should constantly be reviewing it and looking for better and more precise means of passing on our thoughts with the least possibility of ambiguity.

.Which brings me to John Hurley's letter in this month's OUM. Thank you John, for your kind words, and especially the comment of your wife, which gives me the opportunity to clear up the confusion.

Words like "and", "but", and "so" are called conjunctions because they are designed to join two closely related thoughts. It was at one time the rule that they were only to be used before the final item in a list (bell, book and candle), or to join two co-ordinate or two subordinate clauses, sitting like piggy in the middle and doing nothing else. (E.g. He came into the room and she stood up. The book, which belonged to Peter and had a red cover, was lying on the table.) But despite this rule, and from the earliest written records, usage has consistently defied the law that a sentence cannot begin with a conjunction. To explain this clash between what is and what should be, I must distinguish between GRAMMAR and USAGE. Grammar gives us the law, and usage is the application of that law. It is as if grammar says "Thou shalt not kill" and usage then says, "Ah, but what about self-defence, defending one's country in war-time and the slaughter of animals for food" This is only an analogy but I hope it will serve. I have seen, in the course of my 75 years, strict grammatical laws eroded by usage. At first the pedants shake their heads sadly at what they call "bad grammar" only to find, ten years later, that they are out of step with the rest of the world and must change. The law that an infinitive can never be split went up in smoke when man landed on the moon. An infinitive is a verb preceded by "to" (to go, to speak, to eat etc.) In my youth it was a crime to split the two words and say things like "to never go" instead of "never to go". Writers have always done it and critics have always condemned it. Now, with space travel we have the motto, "To boldly go where no man has gone before" (or something similar) and bang goes the split infinitive rule. Americans never obeyed it in any case. And why should they? It served no purpose, and splitting the infinitive causes no ambiguity. Indeed, it often avoids clumsiness of expression.

So it is with "and" or "but" when used to begin a sentence or a paragraph. Here, (and forgive me for using a sledge hammer to make my point), are a few examples. I took down at random from my shelves three novels by popular writers who know their craft.

Morris West (The Salamander) begins sentences with "AND" on pages 6,9,10,12,14,15 and so on.

Hammond Innes (The Mary Deare) starts sentences using "BUT" on page 1, and "AND" twice on page 2.

A.J.Cronin (Adventures in Two Worlds) "BUT" on page 1. "AND" (twice) and "BUT" once on page 2.

But why go on...? (says he, beginning a sentence with "BUT".)

Turn to the most literary work ever written, - the Bible. Look at the first chapter, Genesis. You will find that more than half the sentences, (called verses) begin with "and".

The New Testament is the same, "And it came to pass...." "And some fell on stony ground...."

Take "The Lord's Prayer". "And forgive us our trespasses as we forgive them that trespass against us. And lead us not into temptation..."

The poem by William Blake, now a sacred song "Jerusalem"

"And did those feet in ancient time walk upon England's mountains green."

Turn to Dickens, Shakespeare, Shaw, Wilde, --- the list is endless. They all begin sentences with "AND".

And if it's good enough for them, it's good enough for me.

I quote now from Longman's English Usage. **It is often perfectly legitimate and very effective to begin a sentence or even a paragraph with "and".** It's a pity some teachers don't know it or accept it.

Let me give you two common and valid uses. First, in conversation. Someone says, "I've invited George", and someone else asks, "And what about Mary?" What's wrong with that? And as Dave Dick has already said, to provide a complete break in thought we can begin a new paragraph by writing: - "And now for something new".

Man invented language and set out grammar as a guide. English, more than any other language I know, has constantly allowed common sense to modify grammatic structures, and as long as no ambiguity results we should be quite happy to accept the situation. However, although it is, and must continue to be, a changing language, where change brings doubt and confusion it must be resisted. For example, we have a 100% rule that the letters "CI" (as in CIGAR) give the sound of "SI" and not "KI", so people who write ORICING are writing "orricising" and not "orricking". Is ORICIAN to be pronounced ORISHAN or ORICKIAN? This kind of inconsidered change is unacceptable because of the problems it will introduce in the longer term. I once wrote an article begging readers not to invent new words derived from BASIC or from ACRONYMS like O.R.I.C and P.C.U. Imagine "I have been PCUING all night"? It's ugly and impossible to pronounce. So although ORICKING is better than ORICING it should never have been conceived in the first place, or should have been strangled at birth.

Here are a few of the things that bring me close to tears at how our language is used by professionals:-

I have before me an article from last Saturday's Daily Telegraph about the death, by hanging, of a young boy. The mother, surely distraught by the tragedy, cannot have been much comforted to read in the newspaper that she had murdered her son. The writer, unaware of his ambiguity, had written:-

"A 12-year-old boy died after being found hanged from his bedroom door by his mother."

This is misrelation of the worst kind. The boy was not hanged by his mother. He was found by his mother.

"A 12-year-old boy was found dead by his mother, hanged (or better still, "hanging") from his bedroom door."

That is clear English. The active mood is even better than the passive. "A mother found her 12 year old son dead, hanging from his bedroom door."

Incidentally, did you know that there are two verbs "to hang"? One means to suspend an article. The other means to kill by suspension. Their past tense and past participles are different. The present tense has the same word.

Present tense:- We hang our clothes on a line. They hang murderers in some countries.

Past tense:- I hung the clothes. The jailor hanged the man.

Past participle:- She has hung the washing. He has hanged himself.

The Liberal-Democrats have just pushed a leaflet through my door urging me to vote Lib-Dem in the European elections and saying to the voters, after a Lib-Dem increased majority in the local elections:-

"Thank you!! We couldn't have done it without you!"

What a stupid remark! Of course they couldn't have done it without the electors who voted for them. Or did they think they might get in by magic in spite of the electoral system? Why can't people think before they write?

Let me repeat the principles by which I have taught English all my life.

The spoken word is less formal than the written word. We can make reasonable short cuts, and take more liberties.

"What's the matter? Got a toothache?" is bad grammar but good English. It should be "Have you got?" but in spoken English we are more direct and less formal. We couldn't write "You coming out?" instead of "Are you coming out?" but we constantly say it, and rightly so. The voice clears up any chance of ambiguity.

Our word "Goodbye" is a contraction of "God be with ye!" which became "God be wi' ye" and hence "Goodbye."

The word "between" is derived from "by twain" or "by two", and when I was young you could not say or write "between the three of us". Even "between two" was senseless repetition because it meant "by two two"

So we had to say "Between hills" when there were two, and "among the hills" when there were more than two.

Now BETWEEN has lost its original meaning. BETWEEN and AMONG are synonymous, and only the pedants keep on grumbling at the change in usage. "PRESENTLY" used to mean "immediately". Now it means "in a while". We all understand the modern meaning. It has become part of modern English usage.

In my youth, the sentence "I found it among the rubbish" was considered incorrect. If it was "in the middle" it was AMIDST and not AMONG. If it was "common to all", like a secret, or money divided, it was AMONG (or AMONGST which means the same). "Share this money among(st) you. The village nestles amidst the hills.

Now nobody gives a damn about the one-time-distinction and we understand each other perfectly.

On the other hand we ought never to accept a statement like, "I live in the house on the road with the wide entrance" unless we mean that it is the road with the wide entrance and not the house.

Many things are accepted in spoken English which would be intolerable in a written form.

A recent habit is to say "Enjoy!" instead of "Enjoy yourself!" or "Enjoy the film!". But according to grammar, ENJOY is a transitive or reflexive verb and needs an object. Enjoy yourselves. Enjoy the dance. Not simply "Enjoy". That is what grammar says. Usage says differently and at the moment usage is gaining acceptance in spoken English, although nobody would ever say. "Last night I really enjoyed" without saying what.

It will come, given time. But although "Enjoy!" is now accepted in the spoken word it will not be so readily accepted as the written imperative form of the verb to enjoy.

Now to end my time with you all, here are a couple of intriguing items. Can you think of a situation where the word "AND" is written five times in succession in a single sentence? Or can you imagine the word "HAD" written eleven times in succession in a single piece of writing? Here they are.

The Pig and Whistle needed a new sign. The painter painted: PIG AND WHISTLE.

The landlord said, "There is too much space between "PIG" and "AND" and "AND" and "WHISTLE".

In an exercise John wrote "She had her breakfast before Peter arrived". Mary wrote "She had had her breakfast before Peter arrived."

So, Mary, where John had had "HAD", had had "HAD HAD". "HAD HAD" had had the master's approval.

I'd like to think that some of you will still keep in touch after Dave takes a well earned rest and lets the reins drop.

Thank you Dave, for all you have done for so many of us. It's been a constant joy to be a member of OUM.

If any of you have problems of grammar or usage that I might be able to resolve, then remember, I am on E-mail:-- pepa.frank @ diamond.co.uk. But don't delay too long. There aren't many more years left in these old bones.

Goodbye for now, and thanks to all of you. It's been great fun.

Brian's Page

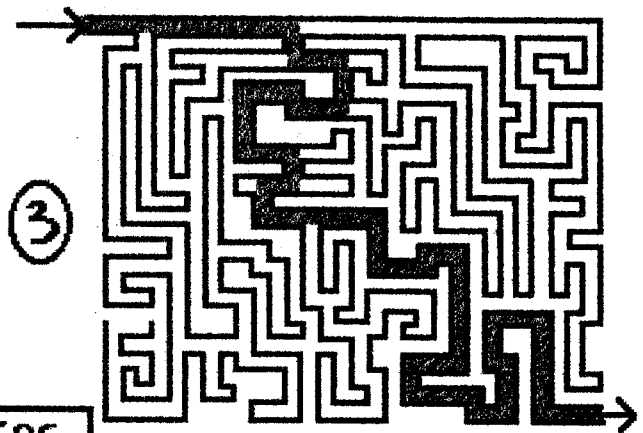
OLIM # 142 - June 1999 - Page # 16

Answers firstly to the posers set last issue

①

▲ = 3 ⬡ = 7

● = 11 ■ = 5



② Clare

POSERS

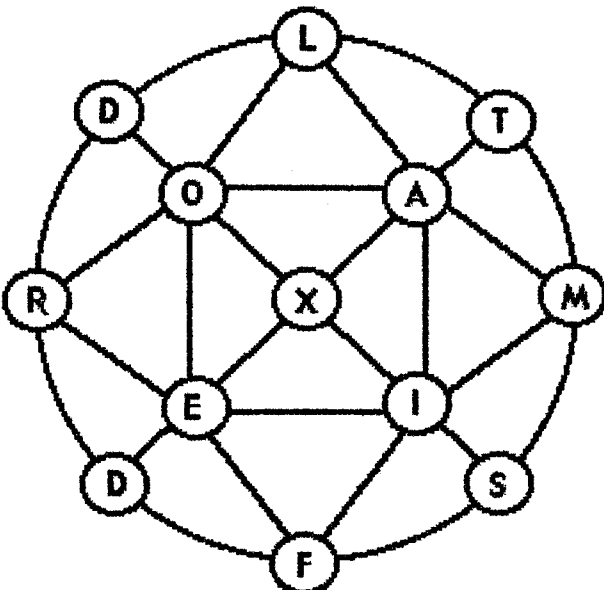
①

1							
2							
3							

Solve the clues , then re-arrange the letters in the shaded squares , to reveal the mystery word :

- 1) Go Forward
- 2) You , me , in fact anyone
- 3) They say these turn into mighty rivers

② Move from letter to letter along the lines to spell out as many words as you can . Each word must have at least one X in it. The arcs around the circumference count as lines .



Prize Pool

Two winners this month : Stan Holden , who wins a copy of "The 10 Dimensional Maze" , and Ken Duddle , who wins the game "Balax".

To replace these prizes , we have 1) Rubiks Clock Puzzle , 2) Digital Travel Alarm Clock.

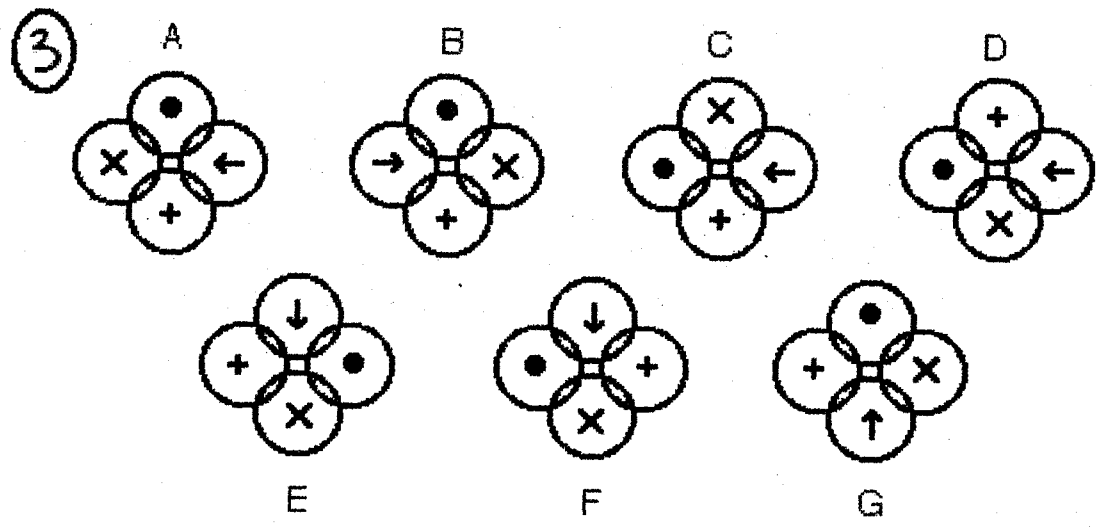
Remember to stake a prize claim , simply correspond with me . Any answers submitted needn't be correct . Just name yourself and required prize.

Brian Kidd , 49 Harlequin Drive , Newport , S. Wales . NP20 5GJ

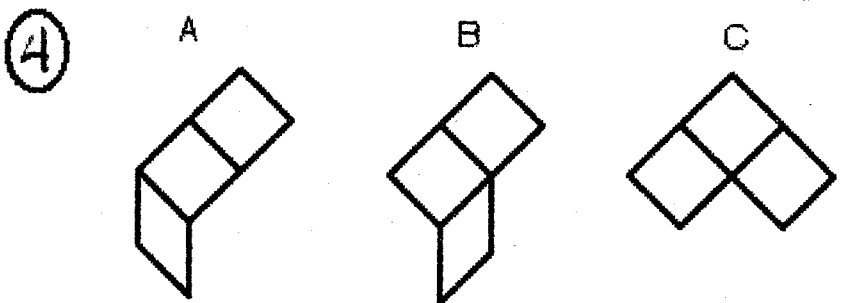
Basically Yours

is no more.....

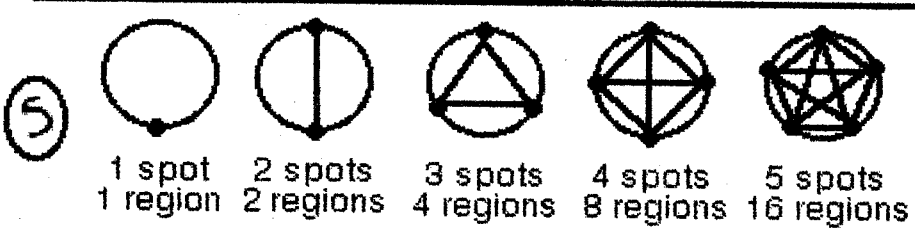
Lack of readership support, along with personal difficulties, have made my mind up, with regards to ending this section of OUM. For those who would like a copy of the project undertaken, I will ensure it's completion, prior to the closure of OUM, and make available copies, free of charge. The 'Brian's Page/s' will continue until the end, and for this issue, I will provide some addition posers for you to ponder. Next month will see the last of the additions to the prize pool. If all prizes are not claimed by August issue, then September will see the 'free for all' clearout. Sorry if this causes any upset (doubtful), but the decision is final.



WHICH OF THE ABOVE IS THE ODD ONE OUT ?



WHAT COMES NEXT IN THE SEQUENCE ?



IF 6 SPOTS WERE ADDED TO A CIRCUMERENCE AND JOINED HOW MANY REGIONS WOULD WE GET ?

Brian Kidd, 49 Harlequin Drive, Newport, S. Wales. NP20 5GJ