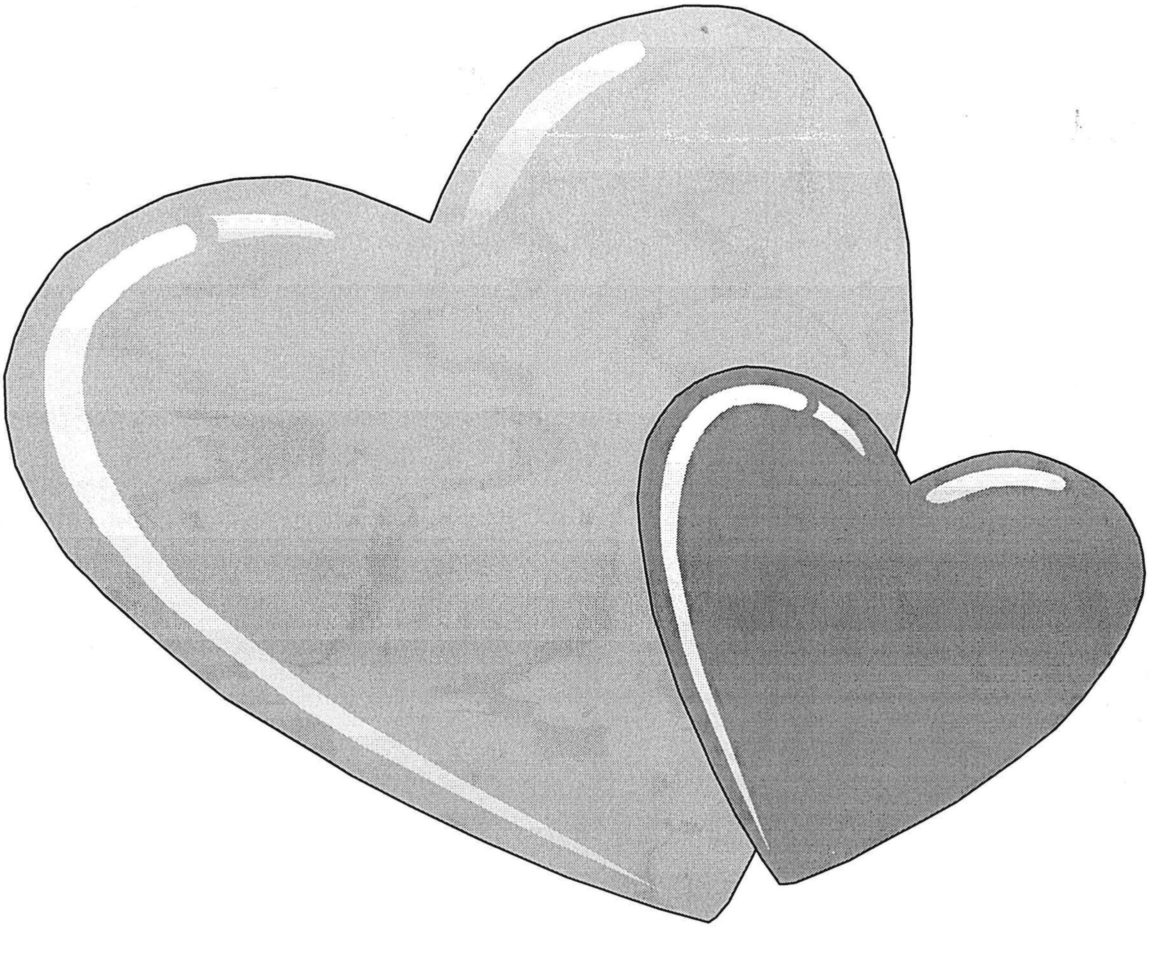
ØRIC USER MONTHLY

with Alternative Micros

Number 90

February 1995

Keeping the Oric alive



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

THE EDITORIAL

HELLO,

AND WELCOME ONCE MORE TO O.U.M.

AS WE GET INTO FEBRUARY ONE TENDS TO THINK OF LONGER DAYS AND WARMER WEATHER, AND OF COURSE - HOLIDAYS.

I HAVE DECIDED TO HAVE A BREAK AT 'BUTLINS' IN MINEHEAD AT THE BEGINNING OF APRIL. THIS MEANS THAT THERE WILL BE NO APRIL ISSUE OF O.U.M, BUT INSTEAD A DOUBLE ISSUE IN MAY.

THE BACKLOG IS STILL BEING WORKED ON!

THIS MONTH SEES THE APPEARANCE OF SOME VERY INTERESTING ARTICLES - NOT ALL ORIC RELATED. KEEP UP THE GOOD WORK.

I WILL SOON BE FIXING THE DATE OF THIS YEAR'S AYLESBURY ORIC MEET - SEE NEXT ISSUE.

AND SO TO THIS ISSUE:-

THE INDEX

==

OUMDISC#6

THE SIXTH OUMDISC IS ALL BUT COMPLETE - I HAVE TO TYPE UP THE INSTRUCTIONS ETC. INCLUDED ON THE DISC ARE: another version of MASTERMIND, an excellent DOTMAN,cheat version of GRAVITOR,cleaned up version of NHL ICE HOCKEY MANAGER, a version of MIND MADNEZ allowing you to design your own screens + lots,lots more.

Expected release date is March 1st.

===

MARCH O.U.M

Articles for inclusion in the March issue should reach me by February 24th - PLEASE!

===

RAFFLE TICKETS

Starting with this issue,you will be recieving raffle tickets - see BRIAN's PAGE for the full story. Look after these tickets with your life! I will NOT be keeping a note of who has what tickets. You will only be able to claim a prize if you have the ticket in your posession. YOU HAVE BEEN WARNED!

==

Important NEWS

WAVE

'WAVE' OF CUMBRIA ARE CURRENTLY OUT OF STOCK OF 3" DISCS.

A BUDDING PROGRAMMER

RODNEY TITCHENER'S SEVEN YEAR OLD SON NEIL HAS RECENTLY STARTED TO USE DAD'S ORIC, AND SEEMS VERY KEEN ON LEARNING PROGRAMMING. IT IS NICE TO SEE THE YOUNGER GENERATION GETTING INVOLVED.

DISPLAY ELECTRONICS

"DISPLAY ELECTRONICS" ARE SELLING THE 'ALPS' MECHANISMS FOR 49 POUNDS + V.A.T. THEY ALSO HAVE BOXES OF PENS FOR 5.50. ALAN BOWERS TELLS ME THAT HE HAS THE FULL TECHNICAL SPECS. ON THESE UNITS.

TECH.SPECS.

ALAN BOWERS HAS ASSORTED AMSTRAD SPECIFICATIONS FOR THE CPC,PPC1640 AND PC1640. HE IS ON THE LOOK OUT FOR A PC1640 KEYBOARD.

ALAN, FROM THE ISLE OF WIGHT, HAS RECENTLY CHANGED JOB LOCATIONS FROM LIVERPOOL TO SOUTH DORSET AND CAN BE CONTACTED ON: 01202 782839 (MON-FRI DAYTIME) OR 01202 733266 (EVENINGS). ALTERNATELY YOU CAN 'E-MAIL' HIM AT: ALAN@BOWERS.DEMON.CO.OK.

ONCE AGAIN ON THE SCROUNGE (HIS WORDS,NOT MINE); ALAN IS ON THE LOOK-OUT FOR A PARTICULAR MODEM,NAMELY,: US ROBOTICS WS 3000 AND IT MUST RUN AT 2400 BPS.

SEDORIC UPDATE

BOTH COLIN COOK AND JONATHAN BRISTOW HAVE BEEN IN CONTACT WITH ME REGARDING THEIR THOUGHTS ON THE SEDORIC UPDATE BEING CARRIED OUT BY DR.RAY. AMONG THE POINTS RAISED IS THE PROVISION FOR SUB-DIRECTORIES. THIS FEATURE ALREADY EXISTS IN 'RANDOS',AND ALLOWS ONE TO COLLECT ALL THE SIMILAR FILES ON A DISK TOGETHER UNDER ONE HEADING.

THE GOOD NEWS IS THAT IT IS DONE.

DR.RAY INFORMS ME THAT THE NEW SEDORIC HAS THIS FACILITY, AND IS SIMILAR TO THAT USED BY 'MSDOS'.

MICROWAVES

MATTHEW COATES was intrigued by the method of increasing execution speed (mentioned in MICROWAVES in the January RAMROM). However,when he tried it out he discovered that it also changes the speed of the internal clock (as used for the timings in his current article).

Readers should be warned of this side-effect in case they try to use it in programs,which rely on the clock for accurate timings.

==

LATE ADDITIONS TO OUMDISC#6

Thanks to Matthew Coates the next OUMDISC now has a couple of extra programmes. The first is a utility for WORDWORTH users - a word count. The second is a mortgage repayment program.

PROGRAMMERS BIND

4

by Matthew Coates

Hello again. I should start with an apology for my long absence from the pages of OUM. No excuses, I've just been doing other things. Thankyou Dave for being so patient and for your occasional, very gentle, reminders.

Since my last article I have switched from Wordworth to Word-Speed and, so far, I haven't looked back. Operation is beautifully smooth and there are lots of little things that help to make life easier but the main thing I appreciate is not having to decide in advance on the page-width since I don't usually know how much space a document will take up until I have finished writing it. This also means that the words wrap around properly on screen, regardless of the width of the eventual printed document.

Speed Trials

In the past I've gone on quite a lot about the importance of good structure and how, ideally, programs should be easy to follow for anyone who needs to understand them at a later date. This is all very well, but on occasions we need to squeeze as much performance as possible out of this tired old machine of ours and elegant coding is the least of our concerns. What I would like to do this time is demonstrate how the requirements of speed and good structure often coincide, or at least do not conflict.

To this end I have carried out a little research to compare the efficiency of various alternative methods of achieving the same results. In each case I added the test code to the skeleton shown below, which gave a time in seconds for the execution of the code, allowing direct comparisons between the various alternatives.

```
        100 N=1000
        110 DOKE £276,65000

             ...

        900 PRINT (65000-DEEK(£276))/100
```

The variable, N, is used to control the number of repeats and this has to be large in order to obtain a measureable time period for the comparisons. In each case the code under test was inserted between lines 110 and 900 shown above. Occasionally, additional variables have to be set before the main routine starts, and these are set before line 100 so that they do not add to the measured execution times. The measured times are shown to the right of each listing.

Note that I have deliberately not expressed comparisons between times in terms of percentages, since this might be misleading. It is better to think in terms of the time added or saved by the different options over N cycles. Also, the examples given are the simplest possible cases and the percentage differences are larger than would be attainable in most real programs.

Loops

The three main forms of loop available for use in Oric BASIC are FOR...NEXT, REPEAT...UNTIL and the returning IF statement. Examples are shown below.

```
1      200 FOR I=1 TO N
       210 NEXT                              1.33

2       10 I=1
       200 REPEAT
       210 I=I+1
       220 UNTIL I>N                         7.93

3       10 I=1
       200 I=I+1
       210 IF I<=N THEN 200                 12.58
```

From our knowledge of BASIC, we would expect the first two to be better than the last because for these the starting address is stored on the stack so that the interpreter knows where to return to when the end of the loop is reached. Because the third option doesn't make use of this facility, the interpreter has to search through the whole program (from the beginning) until it finds the specified line number. This accounts for the difference between options 2 and 3, but why is REPEAT...UNTIL so much slower than FOR...NEXT? To find out, try adding the line 205 J=J+1 to listing number 1.

Subroutines

Because GOSUB makes use of the stack to keep track of the position from which the subroutine was called, a RETURN is much quicker than a second GOTO.

```
4      200 FOR I=1 TO N
       210 GOTO 950
       220 NEXT
       910 END
       950 GOTO 220                         11.59

5      200 FOR I=1 TO N
       210 GOSUB 950
       220 NEXT
       910 END
       950 RETURN                            7.30
```

An important point to remember about GOSUBs is that the initial subroutine call still has to search for a line number, just like the GOTO. By placing the most frequently used (or the most critical) subroutines near the start of a listing, the search time on each call can be considerably reduced. Compare the following with listing number 5.

```
6       10 GOTO 100
        20 RETURN
       200 FOR I=1 TO N
       210 GOSUB 20
       220 NEXT                              5.67
```

You will notice that in the case of both subroutines and loops the GOTO comes off worst in terms of speed (remembering that line 210 in listing 3 contains an implied GOTO). This is the most common source of inefficiency and should be avoided where possible. For example, say we have a section of code that is only executed when a certain condition is true. The obvious approach is to include the code in the middle of the routine and to use an IF (with an implied GOTO) to skip this section when it is not needed, as in example 7.

```
7      200 FOR I=1 TO N
       210 IF I<N THEN 230
       220 PING
       230 NEXT                            9.96
```

In cases like this, where the routine in question is skipped more often than it is executed, a better solution is shown below.

```
8      200 FOR I=1 TO N
       210 IF I=N THEN GOSUB 950
       220 NEXT
       910 END
       950 PING
       960 RETURN                          5.09
```

Of course in this case it would be far simpler to avoid both of these and use something like

```
       210 IF I=N THEN PING
```

However, I was interested to discover that if a longer sequence of commands is required, as in examples 9 and 10, a separate subroutine can still be marginally faster than appending the commands to the IF statement. Of course, in this example, the statements are only executed once in 1000 cycles. As the number of subroutine calls increases there will be a break-even point when the cost of searching for the subroutine will start to outweigh the benefit of not having to scan through the unused commands every time.

```
9      200 FOR I=1 TO N
       210 IF I=N THEN PING:PRINT "END OF
           GOSUB TEST ROUTINE"
       220 NEXT                            5.71

10     200 FOR I=1 TO N
       210 IF I=N THEN GOSUB 950
       220 NEXT
       910 END
       950 PING:PRINT "END OF GOSUB TEST
           ROUTINE"
       960 RETURN                          5.14
```

REMarks and Compaction

As I have mentioned before, I consider compaction of code an abhorent practice which makes programs all but unreadable and has little impact on execution time. Similarly, the omission of comments in the name of speed hinders readability and is often not terribly effective anyway. Having stated my opinions I feel I ought to present some sort of evidence to support them. First of all, compare the following (which is given in full) with listing 5.

```
11     100 N=1000:DOKE£276,65000
       200 FORI=1TON:GOSUB950:NEXT
       900 PRINT(65000-DEEK(£267)):END
       950 RETURN                          7.06
```

Now compare examples 12 and 13 with listing 1.

```
12     200 FOR I=1 TO N
       210 REM
       220 NEXT                            1.78

13     200 FOR I=1 TO N
       210 REM  USEFUL TEXT
       220 NEXT                            2.08
```

The first thing to note is that the differences we find here are relatively small when compared with those we have seen up to now. Remember that the comparisons should be made using absolute times rather than percentages. Note also that the length of the comment makes a difference here, with the comparable execution time increasing to 3.45 seconds for a line of the maximum length. This effect can be minimised by keeping comments short and/or by placing them outside the most critical parts of the program, so that the processor does not have to keep scanning the same "useless" code. For example, try placing the REM statement in line 199 instead of line 210.

One other point to remember is that the length of the text following a REM makes no difference to the search time for a GOTO or GOSUB. This is because the processor only reads the line number before skipping to the start of the next line. Therefore, it is the number of lines, not their length, which matters most in any statements which appear before a line which is the object of a GOTO or GOSUB. The length of a comment is only significant when the surrounding statements are actually being executed, as in example 13. Thus, if a REM occurs in a part of the program that is never executed (eg between a RETURN and the start of the following routine), the length of the comment is irrelevant.

And Finally

So, it seems that the requirements of structure and speed can be the same. For loops use FOR...NEXT where possible, or failing that use REPEAT...UNTIL. Subroutines using GOSUB are very efficient, even if they are only ever called by one other routine and always return to the same position. Avoid repeated use of GOTOs, including implied GOTOs in IF statements. Do use REMs but think carefully about where you put them.

There are bound to be special cases where these recommendations will not be optimal. For example, how many exclusive IFs can there be in a sequence before it becomes more efficient to use a GOTO that returns control to a point after the subsequent IFs than to use a GOSUB that returns control to the next IF statement (which must be false and therefore need not be evaluated)? When is it better to create an indicator variable and use ON GOSUB? Questions such as this can only be answered for each specific case.

One last point which I was surprised to discover in the course of my research: If additional spaces are typed at the end of a statement, these are not stripped out but are retained in the listing. This can slow down execution significantly if they are in the wrong place, so take care.

MARSHALL'S MUSIC

It's been quite a while since this series started, and now it is time to finish off. If there are any subjects that call for the resurrection of this article, then that will happen, so if you want any help with music - (sound advice ?) - then write in.

For now we are going to give you a special treat and enlighten you on a subject that has been a topic of debate for musicians and non-musicians alike - the secrets of Stradivari.

ANTONIO STRADIVARI (1644 - 1737) is probably the most famous maker of musical instruments. His violins have sold for record prices and his name is synonymous with instruments of the best quality. But what is so special about a Stradivarius violin? Why has a mystique surrounded his work making him the 'Mona Lisa' of music?

Stradivari was from Cremona in Italy. At this time the best stringed instruments were said to come from Italy, and the best Italian instruments came from Cremona. Members of the Amati family were amongst the best of these luthiers, (it was a trade very much passed down from father to son). Stradivari became an apprentice under Nicolo Amati and later went on to found his own workshop in Cremona and began to use his own labels in about 1667.

Now let's look at what a workshop such as this would be like. The romantic image of a chap fiddling around carving at bits of wood is a little bit far from the truth. The typical workshop would have quite a few apprentices doing all the hard work with the master craftsman doing the finishing touches and discarding any of the apprentices' work that wasn't up to scratch. They would work about fourteen hours every day. Contrary to popular belief, a workshop like Stradivari's would have produced huge quantities of instruments and was very much a mass-production industry. Stradivari, like the other luthiers, did not just produce violins. He also made 'cellos and a few guitars as well as other stringed instruments. The pattern and shape of the violins that Stradivari made wasn't a fixed pattern that he used, although makers today will make violins which are essentially copies of one of Stradivari's instruments. He was gradually changing the design - lengthening the instruments in an attempt to improve the sound. He eventually went back to the shorter length of instrument as we know it today. He had got the right idea here, but he was ahead of his time and what stopped the larger instruments from working properly were the strings.

A modern violin is a quite different machine to the instruments Stradivari made. In fact all Stradivarius violins in use today have been modernised. The angle of the neck and the length of the fingerboard have been changed to allow for the newer metal-wound strings which are under a greater strain (If you put modern strings on an unmodernised violin, it would collapse under the tension. This has happened quite a few times!). So what we have is an instrument that sounds quite different to the way it originally sounded. (It is true that violins improve with age as well.) But these violins are still better than the rest. What is it that makes the difference? Modern luthiers are forever trying to reproduce the quality obtained by the old masters when woodwind and brass makers are constantly improving their instruments? This is something that has frustrated makers and produced a whole mythology of its own. But now science has revealed the secrets of the past although there are many who still refuse to believe the truth and want to maintain the mystique.

OK - FACT 1: Wood of the quality of that used in Cremona violins is not available any more. It has all gone - long since chopped down, and there ain't any more growing that can replace it. To re-establish a forest like the one needed will take centuries.

FACT 2: For a violin to sound good it has to be a sort of irregular shape. Strads were in fact, 'poorly assembled' which sort of makes it more irregular, inadvertently improving the sound of the instrument. A good modern luthier will make an instrument more accurately - all squares and right angles.

So, you make a violin out of the best available wood and make it a bit irregular and it should sound something like a Stradivarius. It's been tried, and it doesn't work. This is where a German scientist with an interest in music solved the problem. He managed to scrounge a tiny shaving off a Strad. and stuck it under an electron-microscope.

FACT 3: Stradivarius violins are made from wood that is 'wet-seasoned' - ie the wood is seasoned in water, unlike anything today. Wood nowadays is usually 'seasoned' in a kiln. Wood used to be left outside stacked in boards for a number of years until it had dried and settled down so that it would not warp.

Under the electron-microscope, small 'pores' in the wood of a Strad. violin are filled in with the saps and natural juices in the wood. This helps the sound to travel through the wood. Wood that is not wet-seasoned has air in these pores.

Stradivari would have got his wood from Venice, the nearest port, about 120 miles from Cremona. Here wood would certainly have been left in water for some time.

OK, so the German scientist seasoned some quality wood in water himself, and made a violin to a Strad pattern. It was very good, but still not as good as a genuine instrument. He realised that there must be something to the polish on the instrument that effects the sound. Now violin polishes are something akin to fish batters. Each luthier has his secret recipe. Old recipes contain some very obscure substances including frankincense and myrrh. Making these potions can be very hazardous and more than one luthier has blown himself up in the process.

After a few years of begging, the scientist managed to obtain a tiny shaving of Stradivarius polish and found something quite surprising.

FACT 4. The polish on Stradivarius violins contains ground down semi-precious stones. This makes the polish rigid and kills the upper overtones produced by the violin giving its sound a wonderful richness.

The scientist again made a violin using wet-seasoned wood and polish containing ground semi-precious stones. The result was one of the finest violins made for centuries. He took his two violins to one of the greatest teachers in the world and asked him to compare them to a real Stradivarius.

He agreed that the violin was the best he had heard for years, though he still favoured the genuine instrument. (Well,like I said, they do improve with age!)

So was Stradivari a genius - a superb craftsman? Well, no. Of the hundreds, if not thousands of instruments he made, only the best would have survived through to this time - and there are relatively few surviving. All of these instruments have been modernised. He would have got his wood from the local supplier which just happened to be Venice. He would not have made the polish himself, but would have bought it from his local chemist. The chemist would have used whatever recipe had been passed down to him. So Stradivari was just in the right place at the right time.

All this has been proven, but still it is not generally accepted by the music world and Stradivari is still hailed as the ultimate master craftsman, and violin makers still argue about what is so special about his instruments. Violins will never be as good as those of the past despite what they think.

Well I hope you have found this article and the rest of the series to be interesting. I have shown that any silly bugger can contribute to OUM if they set their mind to it. So why don't YOU have a go and see what you can come up with.

Steve Marshall

LOOKING AT CLASSIC GAMES

with Arnt Erik Isaksen
Part 4

Loriciels

Part 4 of "Looking at Classic Games" is dedicated to the most famous French software company, Loriciels. Only one of their titles has been reviewed in an earlier part of my series, Crocky (part 1.1).

SUPER JEEP. Loriciels 1984. C.Perconti.
This is a version of the true classic game called Moon Buggy. You drive your patrol car on the moon and must avoid or fire at obstacles on the horizontal scrolling screen. Watch out for bombs from space-ships in the air and fire rockets against them to blow them away. There are three levels.
Super Jeep is a good Moon Buggy version with nice and colourful graphics and good sounds.
O:** G:***** S:**** A:***

FRELON. Loriciels 1984. Marc Duren.
Fly a helicopter and pick up blocks for transportation to a bridge, which is being built. Other helicopters and tanks are certainly not your friens - watch out! Ensure that you have enough fuel.
A quite popular game. However, in my opinion is the graphics not smooth enough and the sounds are a little bit too noisy. I must admit that Frelon is not my favourite game.
O:***+ G:*** S:***+ A:***

STANLEY. Loriciels 1984.
Flip a switch to release a disc and then push it down across a bar. There are six bars and you must collect three or more discs on every bar. Many creatures are after you to make Stanley's life hard to live. Did you get the idea?
Stanley is a very original game that is fun to play. You have a lot of options, i.e. make Stanley move faster/slower etc.
O:***** G:**** S:***+ A:****

DOGGY. Loriciels 1984. Eric Chani.
A dog is running home to his dog house. You must help him on the dangerous road home. Holes in the road, traps, logs and other obstacles make it difficult for you. Fight other dogs to score bonus points if you have enough energy. Bones that lay in the road can be eaten to gain energy.
What about this game? The animations of Doggy are just superb, while the music is excellent at the same time. Doggy was one of the first French games I ever saw, and I played it frequently during a period in 1987.
O:***+ G:*****+ S:*****+ A:****

SORVIVOR. Loriciels 1984.
Sorvivor is a pure shoot'em-up-game. Fire at the enemy's space-ships that move down the screen.
The graphics in Sorvivor move very smooth and is the best thing about the game. Sorvivor lacks originality and playing it a few times is enough.
O:*+ G:****+ S:*** A:*+

JIMMY POUBELLE. Loriciels 1985.
Move up and down ladders and collect items on the screen to move on to the next level. A lot of obstacles and moving machines must be avoided.
I must admit that I, for some reason I don't know, have not played this game very much. Jimmy Poubelle seems to be a good choice for those who love ladder games.
O:*** G:**** S:**** A:***

GASTRONON. Loriciels 1983.
Gastronon is like Sorvivor a shoot'em-up-game. This game has, however, similarities with The Ultra and The Hellion but is a lot poorer.
It is not much to say about Gastronon according to my notes.
O:** G:**+ S:** A:*

VIDEO FLIPPER. Loriciels 1984.
Your Oric can be turned into a pinball machine by loading Video Flipper. This version includes options for elasticity, friction, angle and rebound.
Two other pinball versions exist for the Oric, Macadam Bumper (ERE Informatique) og Cobra Pinball (Cobra Soft). All three versions has some advantages. Video Flipper has the options as its best side.
O:** G:**** S:**** A:***

REVERSI CHAMPION. Loriciels 1984.
This is a version of the ever popular board game Othello and gives you a test against the computer or a human friend.
Compared to many other versions of Othello for the Oric, I find this very good. Reversi Champion is extremely fast compared to Reverse (IJK Software), Othello (Kenema) and Othello (CEO). Reversi Champion has not as many options as Othello (CEO), i.e. 3D display etc.
No ratings

I will give a summary of Loriciels's games from Your Oric's charts for June/July 1987:

12.Super Jeep
15.Frelon
20.Doggy
25.Star

Loriciels did not only release arcades and the top 3 adventures in Your Oric's charts were from them, "Le Diament De L'ile Maudite", "Secret du Tomb" and "L'Aigle D'or". Another remarkable program is Master Paint, which is one of the best drawing programs for the Oric.

Next month we look at the games from one of the Oric's best or maybe the best author, John Marshall. Look forward to that. See you then.

READERS LETTERS

DEAR DAVE,

I have just written over a Sedoric disc file with another file, with the SAVEO command. Both files were in Basic,but were different. Is there a way that I can recover the original file?

- JONATHAN BRISTOW

DEAR JONATHAN,

there is a program on Public Domain that will allow one to recover a file. Unfortunately I do not think that this will work after the SAVEO command has been used, as the second file would have been saved on the same Track/Sector. You will,I'm afraid,have to write the program again.

- DAVE

DEAR DAVE,

I'm pleased to read that you are up and going ready to face another year with the ORIC. My days of retirement would be very boring if it were not for being able to switch on the trusty machine for an hour or so any day.

I have finally added a 5.25" drive to my existing 3" one - the 5.25" drives are still available from GREENWELD for 5.95. They are 360K and the model is:Z5971/FD55BR.

I have been making my collection of stored programs even better lately. By making the new discs Menu Driven,so that on booting up every file is able to be read from the opening screen,and selected with a single button push. Also I have a Master 3" disc for the OPELCO drive that immediately transfers control to the 5.25" drive and calls the menu screen ready for use. My daughter is very impressed.

- JOHN HURLEY (Yeovil)

DEAR JOHN,

nice to hear that you have got the second drive up and running. Good also to see that you are geting to grips with Sedoric.

I assume your 3" Master Disc has it's INIST set up as: B-MENU or B-!MENU (if Slave or Gameinit).

- DAVE

DEAR DAVE,

"MONTANA PATIENCE" is so addictive that I've taken to dealing a pack of cards out on a table to play the game properly. Echoing Steve Marshall's remarks,but not wishing to nag, I'd also like to see a better shuffling routine,easy replay,,and if the layout were split into 2 halves,there would be space to get some colour attributes in for the suits,which would make it easier to find the cards.

Glad that my "MASTERMIND" is enjoyed. My high score for this is a paltry level 3,but I'm sure someone has done better by now,so let the boasting begin!

- COLIN COOK (Pitsea)

DEAR COLIN,

perhaps Nick will update Montana Patience for us. Meanwhile I wonder how far Henry Marke has got with MASTERMIND!

- DAVE

Machine Code for the Oric Atmos (Part 43) Peter N. Bragg

It's only money

Editor Dave's recent excursions on behalf of OUM, into the logic of large numbers, reminds me of a time, when a friend was doing the rounds at work, trying to flog tickets for the Directors Ball. It was hard going, until he mentioned that they were raffle tickets, not dance tickets !!

I am not a gambler, having never filled in a pools coupon or been in a betting shop. However, like most people, I usually join in sweepstakes and club raffles. All good clean fun. However, the "Lottery" is a whole new ball game. There is the chance of winning millions and my feelings are that if you are going to have fun gambling, it is best to go for the "big one". Greedy, thats me !! My only reservation on the subject is, never ever gamble anything you are not prepared to lose.

I have not yet bought a ticket for the "Lottery", but have read all the info. It inspires the thought. Could one write a computer program to predict the winning line ? We have a very easy to use computer. Would it be possible to use the Oric to predict a set of Lottery numbers for us ?

True, that is not a particularly original thought. There have been software packages to predict racing form and football pools. How successful are they ? Well, if they were really that successful, the writers wouldn't need to sell their software, they would be raking far more money, by using the software themselves !

Flights of Fantasy

However, providing you don't take it too seriously, there could be quite a lot of fun to had, trying to write a program to predict the weekly lottery result. Sorry to say, that I haven't done anything myself in this particular field, as yet, having been a little bit too busy to do much computing in the last few months. So I haven't got a neat little machine code routine to make us all multi-millionaires, but we can dream, can't we ?

The question arises, surely there are programmers out there who are better than us and machines (perish the thought !) that are faster and better equipped than the Oric ? Surely they have a much better chance of success, predicting Lottery numbers ?

That doesn't necessarily follow. Expert programmers are probably too busy writing safe and saleable software packages to earn their daily crust and few of them would really want to spare the time for a "fun" project. Secondly, what the Oric lacks in speed and memory, it more than makes up in ease of programming and operation, so we have an advantage there. Don't forget, most computer users now, have very little ability or interest in programming anyway, where as we do. It is our interest that has kept the Oric going so successfully for so long.

One in 14 Million

Those are the odds on winning the jackpot. So how do you pick the numbers and how much do you stake ? First consideration, the stake. In theory, if you have confidence in your ability, a single £1 coin, is all you need, however you don't need binary code to tell you that a second £1, will lop a cool seven million off the odds against you. Thats good value at £2, because if you doubled up again, it would only knock off another three and a half million from the odds, which is less of a bargain.

That's the easy bit providing you have the money, of course. Picking the the right numbers is where we need the trusty Oric. Everybody probably has their own ideas of how to go about this. As usual, my own feelings are to concentrate on keeping it all nice and simple. Not being experienced in this particular field, I would probably go about it in the following way........

The "White Stick" Technique
------------------------- First thing is to make a note of the most recent results. You don't need many, say about four consecutive weeks results. Ignoring the "Bonus" numbers leaves four sets of six winning numbers each. Look at them in another way, to see that you have six sets of four winning numbers in sequence and what you need is the 5th number for each of the six sets. In theory you have a choice of 1 to 49 for each number. However, as you need to choose six numbers, the lowest cannot be higher than 44 and the highest cannot be lower than 6 and the other four numbers are restricted in a similar fashion, which must improve the odds somewhat.

Initially, I would probably make some use of Oric's Random (RND) number routines to generate a continuous sequence of numbers. The Random number routine is located at #E34F (or #E34B for the Oric-1). Of course, initial experiments could be tried using Basic, but I would probably use hex values and assembly language. You can in fact use decimal values in machine code programs, the instruction SED (code F8h) sets it up in the 6502 Status Register and instruction CLD (code D8h) clears it back to hex operation again. However, you do have careful when using Decimal operation, that nothing in the machine (including the Operating System), clears the Status Register back to hex operation, when you are least expecting it.

Taking the above example of six sets of four numbers each, I would set up the Random number routine to generate a continuous output, with the aim of producing a matching set of four numbers for each of the six sets of four winning numbers. This could be done by passing the random output through a set of four CMP (test) and BNE (branch) instructions, in much the same way as we have used in the past, to look for a specific address. This would be set up to indicate when the Random number generator has produced a matching set of four numbers, at which point, the next random number that follows the sequence of four, is the magic "5th Number" that we are looking for.

The next thing is to repeat the above operation for each of the six sets of four winning lottery numbers to produce a new set of six numbers (the "5th Numbers") for the next weeks Lottery ticket.

Of course, in theory the numbers should be in the same numerical sequence as the original six sets of four winning numbers. It means that the "5th Number" generated for the lowest set of four should be the lowest of the new set of "5th Numbers" and so on, in sequence up to the highest value set, which should produce the highest "5th Number". It might be necessary to generate quite a few sets of the magic "5th Numbers" in order to produce the desired sequence of values. However, it should be possible to write a routine to test for such a sequence, which would enable the whole thing to be done automatically. The Oric is able to generate and test vast quantities of numbers, far faster than anything done by hand. It should be possible to achieve quite a lot with comparatively simple routines.

OK, so it's a far out idea ! but would it really work ? I have not tried it myself yet, but who knows ? One thing, it is a great way to gain programming experience. As for the Lottery ticket, it is more interesting than using birthdates or house numbers and you never know...........................

BRIAN'S PAGE - FEB. ISSUE O.U.M. 1995

FIRSTLY THIS MONTH , PLEASE NOTE THAT YOU HAVE FOUND A RAFFLE TICKET WITH THIS ISSUE - THESE ARE FOR USE WITH SOME PRIZES I AM/HAVE COLLECTED . A DRAW WILL BE MADE IN TIME FOR THE NOVEMBER EDITION . IF YOU WIN THEN CONTACT ME AND I WILL ENSURE YOUR PRIZE REACHES YOU BEFORE XMAS. THERE IS NO CHARGE FOR THESE TICKETS AND FURTHER TICKETS WILL BE ISSUED WITH EACH MAGAZINE DELIVERED TO YOU. YOU CAN FURTHER ENHANCE YOUR CHANCE OF WINNING BY SENDING YOUR SOLUTIONS TO MY POSERS FOR EACH REPLY I WILL GIVE THE CORRESPONDAND A FURTHER RAFFLE TICKET - SO WRITE TO ME !
SECONDLY , LAST MONTHS ANSWER , WHICH IS 775 * 33 = 25575 - DO IT BY LONGHAND AND YOU WILL SEE IT'S CORRECT . SHOULD ANYONE WISH TO SEE HOW THESE POSERS ARE WORKED OUT THEN SIMPLY SEND AN SAE WITH MEDIA , AND I WILL FOREWARD THE SOLUTION .

NOW FOR THIS MONTHS POSER - SIMPLY WORK OUT THE VALUE OF 'INCH' WHERE :

INCH * INCH = ****INCH - EASY IF YOU PUT YOUR MIND TO IT.

NOW FOR THE END PIECE , WHICH AS PROMISED LAST MONTH IS TO DO WITH BINGO - THIS PROGRAMME WILL GENERATE ALL THE NUMBERS IN A RANDOM SEQUENCE , DISPLAYING THE CURRENT NUMBER IN DOUBLE HEIGHT , AND ALSO CURRENT NUMBER OF NUMBERS DRAWN , AND ALSO ALL PREVIOUS DRAWN NUMBERS. I HOPE TO EXPAND IT FURTHER IN THE FUTURE TO INCLUDE 1) SELF PRINTING OF CARDS (WITH CODES) 2) CHECKING OF SAID CARDS FOR 1/2 LINES AND FULL HOUSES (AS IN ALL BINGO CLUBS , BUT ONLY THE ORIC WAY!) - SO HERE GOES WITH THE LISTING - NO DOUBT DAVE WILL ALSO PUT IT ONTO THE NEXT OUM DISC FOR THOSE TOO LAZY TO TAP IT IN !

```
100 TEXT:PAPER 0:INK 7                     105 POKE 48036,0:POKE #26A,10
110 DIM A(90),B(90)                        115 FOR F=1 TO 90:B(F)=F:NEXT F
120 CLS:?"PRESS ANY KEY TO RANDOMIZE"      125 REPEAT:Z=RND(1):UNTIL KEY$<>""
130 CLS:?"RANDOMIZING THE NUMBERS"         135 FOR F=90 TO 1 STEP-1
140 Z=INT(RND(1)*F)+1                      145 A(F)=B(Z)
150 IF Z=F THEN 170                        155 FOR G=Z+1 TO F
160 B(G-1)=B(G)                            165 NEXT G
170 NEXT F                                 175 CLS
180 REM NUMBERS CHOSEN..                   185 ?"WHEN READY TO PLAY - HIT A KEY"
190 Z$=KEY$:GET Z$                         195 CLS
200 FOR Q=21 TO 22:PLOT 13,Q,14:PLOT 14,Q,0:PLOT 15,Q,"GAME PAUSED":NEXT Q
205 PLOT 11,25,"SPACE BAR TO PAUSE"       210 PLOT 15,26,"^ TO EXIT"
215 FOR F=15 TO 16;PLOT 1,F,7:PLOT 2,F,10:PLOT 11,F,"NUMBER :       ="
225 PLOT 26,F,1:PLOT 24,F,7:PLOT 19,F,4 230 NEXT F
235 I=1                                    240 FOR F=1 TO 10
245 PLOT 1,F,I:I=I+1                       250 IF I=7 THEN I=1
255 NEXT F                                 260 REM SHOW NUMBERS
265 FOR F=1 TO 90                          270 Q$=""
275 IF A(F)<10 THEN Q$="0"
280 B$=STR$(A(F)):B$=MID$(B$,2,LEN(B$)-1):Q$=Q$+B$
290 FOR G=15 TO 16                         295 PLOT 20,G,STR$(F)
300 PLOT 28,6,Q$                           305 NEXT G
310 C=INT(A(F)/10)                         315  D=C+1:IF  D=10  THEN  D=9:E=36:GOTO
325
320 E=3*(A(F)-(*10))+6                     325 PLOT E,D,Q$
330 C=0                                    335 REPEAT
340 C=C+1:R=RND(1):K$=KEY$                 345 UNTIL C=75 OR K$<>""
350 IF K$="^" THEN 430                     355 IF K$=" " THEN 390
360 IF K$<>"" THEN 340                     365 NEXT F
370 PING:WAIT 100                          375 PLOT 11,25," END OF SEQUENE. "
380 PLOT 11,26,"PLAY AGAIN (Y/N) ?"        385 GOTO 440
390 FOR Q=21 TO 22:PLOT 14,Q,5:NEXT Q      395 PLOT 24,25,"CONT."
400 Q$=KEY$:REPEAT                         405 Q$=KEY$:UNTIL Q$=" " OR Q$="^"
410 FOR Q=21 TO 22:PLOT 14,Q,0:NEXT Q      415 IF Q$="^" THEN 430
420 PLOT 24,25,"PAUSE":GOTO 365            425 :
430 CLS:?:?"EXIT CALLED":?                 435 ?"PLAY AGAIN (Y/N) ?"
440 A$=KEY$:GET A$                         445 IF A$="Y" THEN RUN
450 IF A$<>"N" THEN 440                    455 CLS:END
```

SEDORIC - A BUG!

Colin Cook has found a bug in SEDORIC. We start with Colin's letter:

There appears to be a bug in SEDORIC. Try typing in the following program:-

```
10 DIM A%(100),B%(100),C%(100):SIZE=100
20 FOR I=1TO100
30 A%(I)=RND(1)*256:B%(I)=RND(1)*256:C%(I)=RND(1)*256
40 NEXT
50 !OPEN S,"TEST",1:!PUT 1,SIZE
60 FOR I=1TO100
70 !PUT 1,A%(I),B%(I),C%(I)
80 NEXT:!CLOSE 1
```

On running this, I get the following: "?FILE NOT OPEN ERROR IN 50". It crashes at the !PUT statement. The same happens after a QUIT. If a dummy line:-

```
!OPEN S,"TEST1":!CLOSE
```

is inserted before line 50, the message is: "?BAD SUBSCRIPT ERROR IN 70". This error seems to occur because array A%() loses its contents and its dimension,so anything indexed >10 gives the BAD SUBSCRIPT error. The same problems are encountered with two integer arrays,or even just one,but they do not occur with real arrays.

I've encountered this problem before, using ORIC DOS V1.1, but it does not occur under RANDOS, (with reservations). There is obviously conflict between the operating systems and variable storage,because no error is generated if the disk is not accessed.

This is an extremely serious problem for me,because my novelty hires word processor program "Nova-Ryt" was developed on Randos and is so memory hungry that it has to use integer arrays. The only way I can publish this now is if you put it on side 2 of a 3"disk, because this is the only format that can have 2 different operating systems, and of course you have to put most of your programs on side 1 under Sedoric. I'm also planning another byte-hungry program,"Keyboard Karaoke", which will run up against the same problems. Can one of our gurus please have a look at Sedoric and integer arrays? Let's get this skeleton out of the cupboard!

- Colin

THE EDITOR STEPS IN

Me thinks this is one for Dr.Ray, and thus a phone call or two.

DR.RAY REPLIES

After running Colin's program, the Doc encounters the same problem. This jolts something in Ray's memory, and the answer is forthcoming:-

It is caused by a Pseudo array that file handling sets up automatically. This array is effectively called FI%.

When Sedoric comes accross commands such as OPEN or PUT etc., it looks for a long Integer Array. In the case of Colin's program it sees that the first one is a long one and assumes it to be FI%.

The solution is to make the first one short or not Integer. That is to say - a dummy one. Dimension a short Real Array e.g: DIM A(1), or a String Array e.g: DIM A$(1).

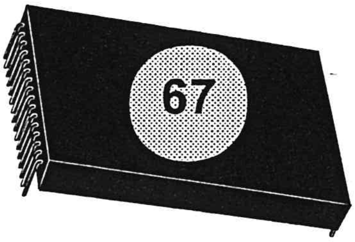
WHAT OF THE FUTURE?

Dr.Ray's says that Sedoric will not be updated as yet to put this right. However, when his Sedoric is available on EPROM,then the matter will be resolved.

AND FOR NOW!

I suggest that Sedoric users make a note of this in their manuals. I would also reccomend that Jon Haworth mentions the bug in the update to the Sedoric manual that he is working on.

- Dave



RAMBLING IN THE ROM

Rambling on...

'HEX$' (FUNCTION)

Principal:

Simple enough, though the function is a little complicated by the fact that one has to ignore any 0's in fronot of the number if it has less than 4 significant digits. The indicator #2F (nil) is used when a significant digit is not found.

Bug: In V1.0 if the number is 0 the result is just #, which is unsightly.

```
D917  JSR $D867          D9B5  JSR $D922      ACC1-->#33-#34
D91A  LDX #00            D9B8  LDX #00        prepare index
D91C  STX 2F             D9BA  STX 2F         indicate nil for the moment...
D91E  LDA #'#'           D9BC  LDA #'#'       hex identifier
D920  STA FF             D9BE  STA FF         at the start of the string
D922  LDA 34             D9C0  LDA 34         take high byte
D924  JSR $D8F5          D9C2  JSR $D993      and convert
D927  LDA 33             D9C5  LDA 33         take low byte
D929  JSR $D8F5          D9C7  JSR $D993      and convert as well
........ ................ D9CA  TXA            is the number 0?
........ ................ D9CB  BNE D9D3       no, jump
........ ................ D9CD  LDA #'0'       yes, put a 0 (it looks better)
........ ................ D9CF  STA 0100, X    in the buffer
........ ................ D9D2  INX            and adjust to place the 0
D92C  LDA #00            D9D3  LDA #00        take terminator
D92E  STA 0100, X        D9D5  STA 0100, X    and end the string
D931  JMP $D4E0          D9D8  JMP $D59B      and end as STR$
D934  JMP $CFE4          D9DB  JMP $D070      'SYNTAX ERROR'
```

'LORES' (COMMAND)

Use: Passes into TEXT mode, and arrives at #D93D/#D9E4 with the parameter (0 or 1) in X

```
D937  JSR $EE9A9         D9DE  JSR $EC21      do a TEXT
D93A  JSR $D80D          D9E1  JSR $D8C8      take a value in X
D93D  TXA                D9E4  TXA
D93E  BEQ D946           D9E5  BEQ D9ED       if it's 0, jump
D940  DEX                D9E7  DEX
D941  BNE D934           D9E8  BNE D9DB       if it's not 1, error
D943  LDA #09            D9EA  LDA #09        attribute for alternate characters
D945  BYT #2C            D9EC  BYT #2C        jump next character
```

```
D946  LDA #08       D9ED  LDA #08       attribute for normal characters
D948  LDX #10       D9EF  LDX #10       black paper attribute
D94A  STX 02F8      D9F1  STX 02F8      and save
D94D  LDX #1B       D9F4  LDX #1B       for the 27 lines
D94F  PHA           D9F6  PHA           save attribute
D950  TXA           D9F7  TXA           line in A
D951  JSR $D965     D9F8  JSR $DA0C     calculate address of the A th line
D954  LDA 02F8      D9FB  LDA 02F8      take colour attribute
D957  LDY #27       D9FE  LDY #27       for the 40 characters of the line
D959  STA (1F), Y   DA00  STA (1F), Y   and write it
D95B  DEY           DA02  DEY           to the end of the line
D95C  BNE D959      DA03  BNE DA00
D95E  PLA           DA05  PLA           recover attribute characters
D95F  STA (1F), Y   DA06  STA (1F), Y   and put in column 0
D961  DEX           DA08  DEX           pass to next line
D962  BNE D94F      DA09  BNE D9F6
D964  RTS           DA0B  RTS
```

CALCULATE ADDRESS OF LINE A (TEXT MODE)

Entry: A conatins a line number

Exit: #1F-#20 = address = #BB80+40*A
X unchanged. In V1.0 Y neither.
A also contains the low byte of the result, which can be useful (though not used in the ROM!)

Bug: On V1.0 the same multiplication routine as at #F700 is repeated. Curious!

Calculate 40*A

```
D965  PHA           ........ ................  save the line
D966  LDA #00       ........ ................
D968  STA 20        ........ ................  initialise high byte of result
D96A  PLA           ........ ................  take the line
D96B  STA 1F        ........ ................  and save for future addition
D96D  ASL A         ........ ................
D96E  ROL 20        ........ ................  *2
D970  ASL A         ........ ................
D971  ROL 20        ........ ................  *4
D973  CLC           ........ ................
D974  ADC 1F        ........ ................
D976  BCC D97A      ........ ................
D978  INC 20        ........ ................  + original = *5
D97A  ASL A         ........ ................
D97B  ROL 20        ........ ................  *10
D97D  ASL A         ........ ................
D97E  ROL 20        ........ ................  *20
D980  ASL A         ........ ................
D981  ROL 20        ........ ................  *40
D983  STA 1F        ........ ................  and save the result
........ ................  DA0C  JSR $F731   AY = 40*A
........ ................  DA0F  STY 20      save high byte
D985  CLC                  DA11  CLC
D986  ADC #80              DA12  ADC #80     add low byte of base of screen
D988  PHA                  DA14  PHA         and save it
D989  STA 1F               DA15  STA 1F      especially in #1F
D98B  LDA #BB              DA17  LDA #BB     and the same for the high byte
D98D  ADC 20               DA19  ADC 20
D98F  STA 20               DA1B  STA 20
```

```
D991   PLA          DA1D  PLA          recover the low byte
D992   RTS          DA1E  RTS
D993   JMP $D807    DA1F  JMP $D8C2    'ILLEGAL QUANTITY'
```

TAKE TWO COORDINATES (PLOT...)

Entry: TXTPTR correctly positioned

Exit: #2F8 and X contain the horizontal coordinate, #1F-#20 addresses the required line.

Bug: In V1.0 1 is addded to the horizontal coordinate, which has no purpose at all (2 would have enabled the protected columns to be ignored). Perhaps it was done to compensate for the fact that the comparison prohibits access to column 39?

```
D996   JSR $DA6B    DA22  JSR $DAF6    prohibit HIRES mode
D999   JSR $D80D    DA25  JSR $D8C8    take a coordinate in X
D99C   CPX #27      DA28  CPX #28      and verify not too long
D99E   BCS D993     DA2A  BCS DA1F     if outside the screen, error
D9A0   INX          ......... .................  curious!!
D9A1   STX 02F8     DA2C  STX 02F8     and save the horizontal coordinate
D9A4   JSR $CFD9    DA2F  JSR $D065    look for ','
D9A7   JSR $D80D    DA32  JSR $D8C8    take second coordinate
D9AA   CPX #1B      DA35  CPX #1B      and verify not more than 27 lines!
D9AC   BCS D993     DA37  BCS DA1F     yes, error
D9AE   INX          DA39  INX          +1 because line 0 is invalid
D9AF   TXA          DA3A  TXA          save in A
D9B0   JSR $D965    DA3B  JSR $DA0C    to calculate the line address
D9B3   RTS          DA3E  RTS
```

'SCRN' (FUNCTION)

Principal: Clear...

Bug: Moved as for PLOT in V1.0 (see previous routine).

```
D9B4   JSR $CFD6    DA3F  JSR $D062    look for '('
D9B7   JSR $D996    DA42  JSR $DA22    take coordinates
D9BA   JSR $CFD3    DA45  JSR $D05F    look for ')'
D9BD   LDY 02F8     DA48  LDY 02F8     take position on the line
D9C0   LDA (1F), Y  DA4B  LDA (1F), Y  take the character on the screen
D9C2   TAY          DA4D  TAY          in Y
D9C3   JMP $D3FD    DA4E  JMP $D4B6    Y-->ACC1 (unsigned)
```

Club Europe Oric

This is the time of year when members are renewing their annual subscriptions. If you are not a Club member, can I try and tempt you. There is a monthly magazine in English, which has drawn many complimentary comments this past year. In addition there are four optional software discs a year, full of excellent French and English software plus original utilities, music and clip art. Drop me a line or give me a ring if you're interested in subscribing :

Jon Haworth
3 Madingley Road
Cambridge
CB3 0EE

Phone: 01223 68761