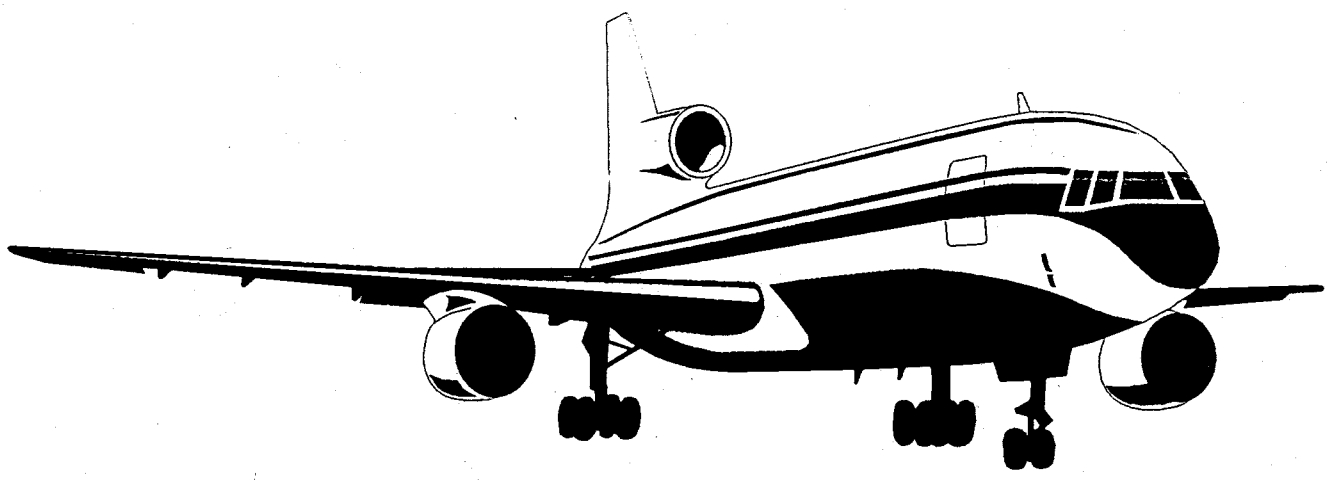
ORIC USER MONTHLY

with Oric Enthusiasts

Europe's longest running Oric Magazine

Number 57

May 1992



FLIGHT SIMULATOR

ON DISC OR CASSETTE FROM C.E.O.

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

T H E E D I T O R I A L

HELLO AND WELCOME,
TO YET ANOTHER PACKED ISSUE.
THIS,THE LAST WEEK OF APRIL,IS MY FIRST WEEK'S VACATION SINCE LAST JUNE. HOPEFULLY I WILL CATCH UP WITH ALL ORIC RELATED MATTERS; HAVING FIRST MADE VISITS TO PORTSMOUTH AND LONDON AND OF COURSE DONE A LITTLE DECORATING.
THE WARMER WEATHER HAS ALSO FUELLED MY ENTHUSIASM FOR ROAD RUNNING. LOOK OUT FOR MY 'ORIC' T-SHIRT WHEN THERE IS A RUNNING EVENT NEAR YOU AND DON'T FORGET TO HAVE AN OXYGEN TENT READY FOR MY USE!!
DUE TO SPACE , SOME ARTICLES HAVE HAD TO BE OMITTED THIS TIME AROUND. I WILL FIND A WAY OF OVERCOMING THIS PROBLEM. IT IS NICE TO SEE ITEMS COMING IN TO THE OFFICE AND PLEASE KEEP IT UP. THERE ARE NO 'LISTINGS' IN THIS ISSUE,BUT THERE WILL BE A GOOD SELECTIO N NEXT TIME. ALSO HELD OVER IS BERNHARD GRONE'S CONTINUATION OF THE 65802 ASSEMBLER.
NEWS ON OTHER ITEMS WILL BE FOUND ELSEWHERE.
THE MAILSHOT HAS BEEN A HUGE SUCCESS. ABOUT 25 REPLIES TO DATE AND STILL COMING IN. HAVE YOU SENT YOURS OUT YET?
IF TIME PERMITS THEN SOME MORE OF YOU WILL BE ASKED TO PLAY THEIR PART THIS MONTH. THE APRIL ISSUE WENT OUT TO 105 READERS.
WELL,HERE WE GO WITH WHAT YOU'LL FIND TO INTEREST YOU THIS TIME AROUND.

INDEX

==========

M S D O S

As stated in the last issue, I have recieved an article from Bernhard Grone entitled 'ORIC READS AND WRITES MSDOS FILES'.
Being 8 pages long; I feel it is too long to serialise and anyway, who wants to wait months to be able to use this information?
Therefore to those who want a copy; then here is what to do: A) Readers in the United Kingdom should send stamps to the value of 70 pence to cover photocopying and postage. B) Readers in other parts of Europe will have the number of issues of OUM owed to the m reduced by one.

MORE ABOUT THE ARTICLE

There now follows the introduction from Bernhard,followed by his summing up. It is the bit in the middle that is missing. -----
THIS IS NOT A RUNNABLE SOURCE CODE - MAIN PARTS ARE NOT TESTED!
IT WILL ONLY READ THE ROOT DIRECTORY. FILES IN SUB-DIRECTORIES CAN'T BE FOUND OR WRITTEN TO.
IT SHALL BE A HELP HOW TO CODE THE PROGRAM.
AS I AM NOT KEEN IN FORMAL LANGUAGE FOR THE DESCRIPTION OF ALGORITHMS,I FORMULATE THE THING IN 'FORTH'. IT IS WELL SUITED FOR THIS PROBLEM BECAUSE OF IT'S MODULARITY. I ONLY DESCRIBE THE WAY TO GET DATA FROM A MSDOS DISC AND HOW TO PUT DATA ON IT,BUT NOT WHERE YOU GET IT FROM. I ONLY USE 3 ROUTINES OF THE CONTROLLER - ROM (see article) SO YOU CAN USE THE PROGRAM WITH ANY DOS.....
.....
..... PHEW, THAT WAS IT. IT IS A LOT OF TEXT AND IT'S A BIG TASK TO CODE AND GET IT TO RUN. I REPEAT: THIS SOURCE IS NOT TEST ED, I DON'T THINK IT WILL RUN WITHOUT MODIFICATIONS.
I DIDN'T GIVE YOU THE SPECIFIACATION OF MSDOS DISC FORMST,FOR THERE EXISTS ENOUGH LITERATURE.
NOW, 'FORTH' PROGRAMMERS,GET TO YOUR KEYBOARDS AND TRY YOUR BEST!
ANY FEARLESS CAN CODE IT IN MACHINE CODE IF HE REALLY WANTS TO.
THE ARTICLE OFFERS A VERY SIMPLE MS DOS. YOU ARE FREE TO EXTEND IT (e.g RENAME or DELETE etc.)
IF ANYBODY MAKES A PROGRAM WITH THE HELP OF THIS SOURCE; PLEASE LET ME KNOW.
THIS TEXT WAS WRITTEN BY BERNHARD GRONE - 18/19 MARCH 1992.

THE FOURTH ORIC MEET

Get your tickets now for the next ORIC meet in Aylesbury on Saturday July 18th.

Price is 2 pound each. If you are coming, PLEASE send in now, as we have to pay the balance on the hire fee for the hall shortly.

Already lined up to appear are : Jon Haworth,Allan Whitaker,David Wilkin, Frank Bolton,Graham Shaw,Robert Cook,Vincent Talvas,Bob Terry,John Mckay and the Lincoln Lads,Peter Bragg ++++. We have also sent out an invitation to Dr. Paul Johnson.

The venue is the usual - The RIVETS sports and social club (Terrace Room).

It is situauted on the South side of town on the Stoke Mandeville/ Princes Risborough road. It is half a mile north of Stoke Mandeville Hospital - therefore if you end up at the hospital,you are near it. The sign at the entrance to the social club is marked - "CLAYHITHE PLC - Bifurcated steel and tube co.).

The meet commences at 10.a.m.

For those coming by road, there is a map elsewhere in this issue.

Trains from London (Marleybone) run every hour and journey time is just under the hour. The railway station is about 1.5miles from Rivets, as is the bus station. If you use a taxi from the railway station, ask for RIVETS, MANDEVILLE ROAD. Fare will be approx. 1.50 pounds.

DON'T leave it to the last minute as I cannot be held responsible if you don't recieve your tickets in time. I will be away for part of June/July - See last issue.

==========

H-PLANS

H-PLANS is an addition to the ASTRONOMY program adapted from PC to ORIC.

The original ASTRONOMY program has been on PUBLIC DOMAIN for some time (Disc only).

Judy Simms spent a long time typing in the H-PLANS program,but could not get it to run. Rob Kimberley,who did all the work on the Oric Astronomy version,has sorted out most of the bugs; BUT it is still not quite right. I tried it out and it worked 3 times out of 6.

A volunteer is needed to iron out the problem. Anyone interested should write to me (Dave Dick) and I will send them the program on disc + PC listing + ORIC listing.

The program deals with visual aspects of the planets.

THE NEXT INTERVIEW

I have recently been in contact with Dr. Paul Johnson (designer of the Oric) and probably the only ex-ORIC director to hang onto his big house in Ascot.

Paul now has a company in Camberley and has agreed to answer my interview on his return from Helsinki.

On behalf of Dave Utting I quizzed him about the ULA circuit diagram. He said he couldn't really pass on information relating to it as he does not know who owns it. I would assume that it has ended up in France. Perhaps our French readers could enlighten us !

==========

ALL FORMATS COMPUTER FAIRS

The next batch of venues for these popular shows are set out below. The fairs are open between 10 a.m. and 4 p.m. Further details are available on 0926 613047.

MAY 16th - LONDON - Horicultural Hall,Greycoat St.,Westminster.
MAY 17th - WEST - Brunel Centre,Temple Meads,Bristol.
JUNE 7th - SCOTLAND - City Hall,Candleriggs,Glasgow.
JUNE 14th - NORTH WEST - Haydock Park,J23,M6

MAN ON THE MOVE

JONATHAN BRISTOW of CHARED/OBED/MUSED fame has move home. Queries on his software and love letters should now be sent to him at : FLAT 5a, MIDDLE ST. ,WATTON,THETFORD,NORFOLK. He advises that his old address (his parents), should still be used when sending bulky items!!

W O R D S P E E D

Product Description: A Disc ONLY wordprocessor on the SEDORIC system.
Author: Dr.Ray McLaughlin. Software House: Rayzorsoft
Outlet: ONLY via Allan Whitaker at ORIC ENTHUSIASTS, 8 Staley Hall Road, Stalybridge, Cheshire. SK15 3DT.
Price: Software - 50 pence for registered SEDORIC users, 1 pound for non-registered SEDORIC users or for 7.50 you get the software,the manual and registration.
You need to send your own disc or alternately add 2.10 for a disc.
(In other words , if you are a registered SEDORIC user,send 9.60 for the full works and a new disc - I THINK !!

Well I have finally decided to get around to reviewing this WP from Dr. Ray.
"Not before time ", I hear you mutter.
The problem that I have found is that having once learnt to use a program (as in the case of EASYTEXT),and then to try to learn another one is an arduous task. You tend to want to run before you can walk!
My first attempt was a disaster. My second a little better. Then armed with a disc from Allan containing some product info. stored with WORDSPEED; I tried to work out what codes he had used and why.
One of the beauties of the program is that the Embedded Commands are displayed in the text in green,so as to stand out.
As an excercise, I decided to write a letter to Dr. Paul Johnson (of ORIC fame).
The 'emphasised print' worked well.
My 'full indent to the right margin' looked nice - ON THE LEFT!
Underscoring of a heading didn't just underscore the heading,but everything to the left of it as well. This also happened when I printed off an article of Allan's and therefore I suppose it could be the printer.
I wasn't at ease using ESCAPE codes.
The 20 page manual is of professional quality. Each feature of the program has narrative to explain what the term means - an excellent aid to new users of WP's.
It is menu driven with a sub-menu,which allows one to get a directory of some/all files,delete/rename files or select a new current drive.
I did not find an option to go into Graphics mode.
There are numerous editing commands and embedded commands. The editing ones include: Cursor right one word, Cursor to position after last character of text,change case of of alpaha character under cursor (e.g 's' becomes 'S' - brilliant that!), Wordcount,Move marked section of text to cursor position, etc etc. Forty editing commands in all,controlled mainly with the FUNCT, CTRL and SHIFT keys.
Over 70 embedded commands are covered and they include: Justification, Centre text,Emphasised print,Double strike,Print current page number and redefine pound character.

With the package you also get Allan's utility to convert AUTHOR and WORDWORTH files .

MY OPINION
-- -------
Perhaps I should of taken more notice of Dr.Ray's demo at the last Oric meet.
Though it has many excellent features; I don't honestly believe that I will be switching from EASYTEXT. The 'juggling' with the FUNCT key and embedded commands baffled me. If I had more time then I might consider switching. I'm quite happy to hit CTRL 'B' on Easytext and know that I've centred the text on EASYTEXT rather than to have to worry if I have to do FUNCT 1 CE and FUNCT 2 or is it FUNCT 1 or RETURN.
I have also got used to the idea of formatting and un-formatting my text at will.
In summing up - From what I have seen of AUTHOR and WORDWORTH; this is a far superior programme and probably no more complex to use.
If anyone out there is using WORDSPEED ,perhaps they would like to voice their opinions. I know Allan, Rob K and Jon H have critically acclaimed it.

- DAVE DICK

FOR SALE

IN GOOD CONDITION AND COMPLETE WITH P.S.U, LEADS,MANUAL and a few GAMES, there is FOR SALE an ORIC 1 (48k).
OFFERS around 15 pound WILL SECURE.
CONTACT: Mr. KENTISH-SMITH on LUTON 598896 or write to him at: 12 WIVETON CLOSE, LUTON, BEDS. LV2 7DA

MODEL RAILWAYS

For those who want to use an ORIC to control their Model Railway System; the book to get is,according to Peter Thornburn, from the pen of R.A. PENFOLD and titled - "ELECTRONICS CIRCUITS FOR THE COMPUTER CONTROL OF MODEL RAILWAYS"
'ORICIANS' may remember Mr.Penfold FOR his book entitled: "AN INTRODUCTION TO PROGRAMMING THE ORIC 1"

MESSAGES

I often get messages from one user to pass onto another and often I may wish to send a message,without the need to write a full letter. Therefore a new feature in O.U.M will be the MESSAGE COLUMN. There is one to follow.

MESSAGE TO PETER BRAGG

Peter Thornburn asks if Peter Bragg could please bring his MICROTAN '65 to the ORIC MEET in July.

ONE OVER THE EIGHT - NEARLY!

Robert Crisp has recently added to his Oric collection. After a recent visit to a Car Boot Sale, Robert is the now the proud owner of 4 ORIC 1's and 4 ATMOS's.

HIRES TIP

To save a HIRES screen,you need to save it in HIRES and type:
CSAVE"title",A#A000,E#BF3F.
When loading,type: HIRES:CLOAD""
Alternately use 40960 and 48960 as the start/end addresses

MESSAGE TO BOB TERRY/DAVID WILKIN/PETE THORNBURN

If you are going to the Paris Oric meet get in contact with each other and 'sort it out'. Whoever is going is also asked to contact the C.E. O for the finer details. I do not have time to to make plans for you.

THE ORIC ANGELS

Marilyn Bell,who headed the team of ORIC ANGELS who visited retail outlets to promote the ORIC,is currently working with Dr.Paul Johnson (Brains behind the design of the ORIC).

WANTED

Timothy Grimwood is a little grim these days as he is still trying to get hold of a piece of software from TOPAZ SOFTWARE,which was called : 3D BATTLESTAR DEFENCE. Apparently it was a sort of Star Wars game.
Tim is to be found at : 103 LOWER LAMBRICKS, RAYLEIGH,ESSEX SS6 8BX, and like many other standed users out there has been in touch following our recent mailshot.

Machine Code for the Oric Atmos (Part 15)

Peter N. Bragg

The Story so far

We have looked at the basic requirements for machine code programming on the Oric and now have a small table (subset) of sixteen useful instructions. We have also had a look at how the keyboard can be used to control operations. The last issue gave a brief summary of how we covered the above items for the benefit of those who might be missing some issues.

The "In's and Out's" of Oric

Two items are essential to any microcomputer. These are, a display, in order to see what the little silicon menace is up to and a keyboard to keep it under control. Of course there are other input and output devices such as printers and joysticks for example, but these are usually added to the system later, as and when we need them.

This means that the first essential of any programming language (or system) is that it must provide an easy way to write to the display screen and also read the keyboard, when required. Of course, this applies to machine code and assembly programming too. However, we dont have to bother to write anything to do these jobs for us. The Oric, like most computers, already has the required routines installed in its Operating System ROM. It could not function very well without them. These routines are also available to us and can be easily called up for use in our own programs.

The two essential calls are "GTORKB" which reads the keyboard and "VDU" which writes to the screen display. We have already used "GTORKB" to read the keyboard in parts 11 to 13 of the series. Now we will look at how the "VDU" call can be used to update and write to the display screen.

The Display

First of all, I should mention that earlier in the series we used a routine that wrote items to the text screen. As a simple demonstration, this could fill the display screen with crosses or any other display item. It operated by selecting screen addresses (in the #BBA8 - #BFDF area) and poking display code into them. This is OK for blanking the screen and producing simple backgrounds, but it obviously has its limitations if we want to display anything else, such as input, on the screen.

To do that, we need something to handle the screen addressing for us and deal with any display codes which affect the display format (ie. Line Feed and Carriage Return). It needs to operate automatically particularly when we reach the end of the last line at the bottom of the display.

Fortunately, the Operating System has the same requirements in this respect and so it is not surprising that it contains just the routine, that we need, which is named "VDU". All we have to do is to load the required display code and call up the "VDU" routine using the instruction JSR F77C. This will then display the code at the current cursor position and then deal with any other effects that the displayed code might produce.

So now there are two methods of writing to the screen display. You can just POKE them straight into a screen address or you can use the "VDU" routine. If you experiment with these two methods, you will find that some codes, particularly those in the 00 to 1F range, produce different results, depending on whether you use the "VDU" call, or POKE them directly into the screen. Essentially, the main advantage of the "VDU" call is that it enables you to feed input straight into the cursor position without having to set an address.

The routine below provides a simple illustration of how the "VDU" call operates. You can put it into the Oric, using an assembler or the "Hexloader" (from Part 7) if you wish. To use it, just CALL#1010 and then tap the space bar a few times. One by one, the three codes in locations 1001 to 1003 will be displayed at the cursor. Change the middle code in 1002 for something else, like 0A or 0B for example and you will see that "VDU" handles these differently.

A "VDU" Display Demo

Now we will look at the routine and see how it works. Essentially the routine is made up from just four different instructions.

```
Oric - Demo        Display Operation          8 Apr 92
-----------        -----------------          --------
[ CALL#1010 ]-----------[ DISPLAY THREE ITEMS ]-----

                        ---Parameter Block 1001---
1001:41       : "A"     :  Three ASCII code
1002:42       : "B"     :   items for the
1003:43       : "C"     :     display.

         ---start---    ---Pause to check screen---
1010:20 78 EB : JSR EB78 : Wait for a keypress.
1013:10 FB    : BPL 1010 : (read keyboard until then)


                        ---Display Item 1 ---
1015:AE 01 10 : LDX 1001 : Load item 1 into Reg "X"
                           and use
1018:20 7C F7 : JSR F77C : OS "VDU" to display it.
                           Pause to check screen,
101B:20 78 EB : JSR EB78 : wait for a keypress.
101E:10 FB    : BPL 101B : (read keyboard until then)

                        ---Display Item 2 ---
1020:AE 02 10 : LDX 1002 : Load item 2 into Reg "X"
                           and use
1023:20 7C F7 : JSR F77C : OS "VDU" to display it.
                           Pause to check screen,
1026:20 78 EB : JSR EB78 : wait for a keypress.
1029:10 FB    : BPL 1026 : (read keyboard until then)

                        ---Display Item 3 ---
102B:AE 03 10 : LDX 1003 : Load item 3 into Reg "X"
                           and use
102E:20 7C F7 : JSR F77C : OS "VDU" to display it.
                           Pause to check screen,
1031:20 78 EB : JSR EB78 : wait for a keypress.
1034:10 FB    : BPL 1031 : (read keyboard until then)

                        ---Finish---
1036:60       : RTS     :  Exit back to Basic.
          ---end---
```

Let's look at the "Wait for a Keypress" instruction pair first. There are four of them and their sole purpose is to slow the operation down, so that you can see what is happening at each stage, instead of having it all happen at once. We have already looked at the "GTORKB/BMI" keyboard operation in detail. This time we are not bothering to save or use the input from the keyboard. The pair of instructions simply provide a "Press any Key to Procede" type of control. This applies to all of the "Wait for a KeyPress" instruction pairs in the routine at 1010, 101B, 1026 and 1031.

What the "LDX" ?

The second item of note in this routine is the "LDX" instruction, which is used to load the display code for the "VDU" call. This is something new and obviously a little bit of explanation is called for. You may recall that at the beginning of the series, I mentioned that the 6502 uses six storage units called Registers. These are used to store and process data. So far, we have dealt with three of these, the Program Counter, the Status Register and the Accumulator. We have since been making use of them in our programs. These three Registers ensure that instructions are executed in the right order, correct decisions are made and data is processed as required. The remaining three Registers are essentially used to make programming more versatile and efficent as you will see when we look at them in more detail. In the meantime, it is sufficent to say at this point, that Register X is one of those three remaining Registers and that to a limited extent it can be loaded and used in the same way as the Accumulator Register, although it's contents are quite separate from those of the Accumulator.

In the routine presented here, Register "X" is being used in exactly the same way as the Accumulator would be. It can hold a maximum of one byte, which as we already have found, can be any value from 00 to FF.

Register X is loaded with the required code for display, which is fetched by "LDX" instructions from Parameter Block 1001, before making the calls to the Operating System "VDU" using JSR F77C, to display each item.

As a matter of interest, other computer systems (eg. Apple, BBC etc) do use the Accumulator for the display code, but otherwise the operation is very similar to Oric's "VDU" call.

So to sum up, this routine fetches the three ASCII codes and displays them one at a time, waiting between each one, so that you can see what effects each code produces. For a start, we are using just "A", "B" and "C" (41 to 43), but of course there is the whole range between 00 to FF, many of which dont just display a character, but produce other effects as well.

I should of course mention that all of this so far applies to the TEXT Mode display. The HIRES Mode operates in a different way. It is in fact not very difficult to write programs for the HIRES display, but the TEXT Mode probably gives us more options at this point in time.

The initial aim of this series has been to provide the smallest possible set of instructions and Operating System calls, as a kind of "starter pack". The idea was to give anyone who would like to try their hand at assembly (machine code) programming, a simple basic "kit of bits" which could be used to experiment and build up from. We will of course be covering a lot more of the 6502 instructions, but now it is possible to produce reasonable machine code programs using the instructions and system calls, we have covered so far. If you havent already done so, why not give it a try for yourself. For a start, take another look at the demo routine right here in this article. OK, so it does work, but is it really good enough ?

The only problem now is that the result is not very useful, because the routine only displays three codes. We really need a minimum of six codes to do the job properly, as some display items need several codes to show their full effect. Of course, I could extend the routine simply by adding on more "Display Item" operations, just like the first three. This would work of course, but it's a very poor solution. The routine shown here is really not very well thought out. There must be a better way, using the instructions that we have covered so far. What we really need is something that can display half a dozen display codes in exactly the same way as this routine, but a bit more compact. Maybe I should look at Part 10 of the series to see if that is any help.

With that thought, I will leave you..Next time, more ways to banjax the Oric !

AN AID FROM ROBERT CRISP

After reading Peter Bragg's article on CRIB CARDS in O.U.M issue 49, Robert Crisp decided to create a srt of his own.
Here in his own words is what he came up with.
As i prefer to write in machine code rather than basic,and am not very familar with ORIC basic;my Crib Cards needed to include more than just machine code instuctions. I have sent you (Dave Dick) a sample and,as you can see,I got carried away with creating a set of Crib Cards.
My cards contain the following subjects:

BASIC COMMANDS
BASIC ERROR MESAGES
ORIC DOS COMMANDS
ORIC DOS ERROR MESSAGES
A TOKEN TABLE
PAGE 0 AND PAGE 2 MEMORY ALLOCATION
ESCAPE CODES
CONTROL CHARACTERS
MEMORY MAP FOR 48K MACHINE
MEMORY MAP FOR TEXT DISPLAY AREA
MEMORY MAP FOR HIRES SCREEN
TEXT AND HIRES SCREEN LAYOUTS
VIDEO ROM BIT PATTERN
ROM ROUTINE ENTRY POINTS
DERIVED FUNCTIONS
FORMAT OF DATA ON A CASSETTE
EXPANSION & PRINTER PORT CONNECTIONS
MACHINE CODE INSTRUCTIONS
EXTRA MACHINE CODE INSTRUCTIONS
NEW ADDRESSING MODES
MACHINE CODE INSRUCTION SUMMARY
CONVERSION TABLE

One ASCII character is missing from the ASCII column in the conversion table. My word-processor/print cannot print the ASCII character for the copyright symbol,which is code 60 (hex).
I use 2 sets of these Crib Cards. One set on A4 listing in plastic wallets in a 4-ring binder is always next to my computer. The other set is printed on thin card,which is cut down to A5 size. This is stored in thick plastic wallets,which originally held 5.25" discs,and stored in an A5 size Olivetti 3-ring binder,which has its own rigid clear plastic case. This copy is often carried around by me,so that I can write programs and learn about the Oric when I am away from home. The clear plastic case also has just enough space to take a few sheets of paper to write on and a pen and a slim calculator. I use a Texas TI-68 calculator. This has a dot-matrix display, which I find better for displaying alphabetic characters,and can perform conversions and arithmetic in Binary,Octal,Hexadecimal and Decimal.
While I was compiling these Crib Cards, I came accross an error in "THE ORIC ADVANCED USER GUIDE". On page 255 in the column for ROM V1.0 start addresses,the keywords PING and ZAP have the same start address (#F41B). The start address for PING should be changed to #F412.
If any OUM readers would like a copy of the Crib Cards; I would be willing to print them a copy. They would have to send me an S.A.E with stamps to the value of 1.20 pounds, and tell me which size they wanted (i.e: A4 listing or the smaller card version).
All you need now is my address:
44 BENTLEY GROVE, MEANWOOD, LEEDS. LS6 4AT

- Robert Crisp

NOTE FROM THE EDITOR

I duly recieved my cards from Robert and was very impressed. It is nice to have all the information in one place,rather than have to go from book to book etc.

If any overseas readers are interested in a copy; they should write direct to me. I will work out the price of mailing abroad and deduct the price from their subscription. i.e if the cost is 2.50,then they will recieve 2 issues less of OUM before they have to re-subscribe. - Dave Dick

THE TIME MACHINE

Solution by James Groom

START - NORTH,WEST,SOUTH,NORTH,GET GLOVES,WEAR GLOVES,WEST,BREAK WINDOW (Can use: BREAK WIND !),GO WINDOW,EXAM PAINTING,GET KEY,OPEN DRAWER,EXAM DRAWER,GET PISTOL,GET CROWBAR,EAST,OPEN DOOR,GO DOOR,GET FLASHLIGHT,GO MACHINE,REMOVE GLOVES,DROP GLOVES,DROP KEY,PRESS (BUTTON),LEAVE MACHINE.
You will now be randomly transported to th cellar,Brig,Sphynx,or Swamp. The Sphynx and Brig must be completed first. To move to the required location keep pressing the button til you get there.

THE SAILING BRIG

-Carrying only a crowbar.

NORTH,GET ROPE,GET SAIL,SOUTH,SOUTH,CLIMB RIGGING,GET POUCH,DOWN,NORTH,GO PORTAL,EXAM POUCH,DROP POUCH,GET PRISM,INSERT PRISM,DROP ROPE,DROP SAIL, LEAVE MACHINE,EAST,OPEN DOOR,GO DOOR,EAST,OPEN CHEST,GET HAMMER,WEST,WEST, GET BISCUITS,EAST,SOUTH,GET NEEDLE,NORTH,UP,WEST,GO PORTAL,DROP CROW,PRESS (BUTTON).

THE SPHYNX

- Carrying Flashlight and Pistol

PUSH STONE,GO OPENING,LIGHT FLASHLIGHT,GET ROCK,SOUTH,PULL LEVER,JAM LEVER, NORTH,NORTH,GO DOOR,CLIMB STATUE,GET PRISM,DOWN,SHOOT DOG,GO OPENING,SOUTH, GET SHOVEL,GO PORTAL,INSERT PRISM,DROP PISTOL,DROP FLASHLIGHT,PRESS (BUTTON).

THE SWAMP

- Carrying rope,sail,biscuits,needle + thread,hammer and shovel.

NORTH,NORTH,EAST,FEED BRONTOSAURUS,GO CAUSEWAY,FIX BOAT,GO BOAT,DIG,DROP SHOVEL,GET PRISM,GO BOAT,WEST,WEST,SOUTH,SOUTH,GO PORTAL,INSERT PRISM, PRESS FOR (4 TIMES),GET CROW,LEAVE MACHINE,LEVER PLATE,GO SHAFT,WEST,BREAK GENERATOR,EAST,GO ARCHWAY,GET DOCTOR

FINISH

WIN A FORTUNE!

Those who have SUPERFRUIT from IJK may like the odds stacked in their favour when playing on this fruit machine simulation. Thanks to James Groom we now print the reel sequence.

REEL 1	REEL 2	REEL 3
POUND	ORIC	CHERRY
PLUM	LEMON	SEVEN
ORANGE	APPLE	ORANGE
BELL	ORANGE	APPLE
ORANGE	GRAPE	GRAPE
CHERRY	POUND	BELL
LEMON	PLUM	CHERRY
GRAPE	APPLE	LEMON
SEVEN	LEMON	CHERRY
ORANGE	PLUM	SEVEN
CHERRY	SEVEN	ORANGE
APPLE	BELL	APPLE
GRAPE	APPLE	CHERRY
LEMON	LEMON	APPLE
POUND	PLUM	GRAPE
ORIC	ORIC	ORIC
ORANGE	PLUM	BELL
LEMON	CHERRY	APPLE
CHERRY	POUND	PLUM
LEMON	APPLE	POUND

CEO QUARTERLY DISC

The first quarterly disc for 1992 duly arrived from Club Europe Oric in time for a look at over Easter. Seven items on the menu,which were as follows:-

A) THE FORGOTTEN ISLE (L'ILE OUBLIEE) from the pen of Patrice Lemitru is one of those nice graphical adventures that the French do so well. After leaving a bunker,you set out for the woods. You can then guide yourself toward the cabin,which is full of tools or head west from the woods toward the coconut tree and onto the port and clamber aboard a small boat. Each time that you make headway; the next screen picture is read from disc. To help you out there is a French to English vocabulary sheet from Jon H. A most interesting adventure.

B) MUSIC - a bit of 'SERENADE 1 HAVANNA' and utterly boring it is too!

C) PENGORIC - a 1983 LORICIELS title and not one of their best. Kill off the opposition by pushing blocks towards them. Similar in idea to 'PIERROT' from a previous CEO disc. Unashamed fun.

D) HIRES SCREEN - an abstract design,similar to Alain Weber's design for the front cover of JEOMAG 3.

E) COCORIC - Michel Cremon with a chickens/eggs arcade game. A lot of instructions in French and I haven't found time yet to translate it. If YOU have;please let me know.

F) ODYSSEY 2000 - by Christian Baribaud. A colourful introduction to this 'Invaders' type game, which is in English. You decide the speed/firing power of the enemy. Good zapping fun.

G) CATALDISC - Yann Legrand's disc catalogue,which has been listed in the JEOMAG. Nice menu routine. Do a directory on a disc. Choose the file names that you want to store, give it a category e.g ARC = Arcade (the extension changes on screen e.g .COM become .ARC); then give it a number and store the record. Something which I intend to delve further into.

In summing up; well worth the price for the adventure alone.

==========

SOFTWARE CHARTS

No response for my request for your favourites (see April issue) , and therefore no chart this time around.

==========

BRAINTEASERS

The current JEOMAG reviews two new titles from Laurent Chiacchierini. MAHJONG and ORIC-SWAP look very interesting strategy games indeed.

==========

SAUSAGES

I have recently recieved a demo of SAUSAGES,the latest Quilled adventure from Paul Baker. I will report next time around. Meanwhile, don't forget that you can get his GRANDAD only from OUM for 3 pounds.

==========

LAND OF ILLUSION

Iif anyone has solved this Tansoft adventure,would they please send the solution to OUM.

==========

DRAPEAUX

This 'Nations Flags' listing from an old THE'ORIC magazine had some bugs in it. Ron Evans has tried his utmost to sort them out , but to no avail. If anyone has solved the problem; PLEASE write to OUM.

==========

PROTEK

Now in stock are a couple of PROTEK programmable joystick interfaces,which originally cost about 30 pounds. They go into the expansion port and I have had them working with the likes of ORICMUNCH and ULTRA. Price inclusive of post/pkg. is 8 pound with the main benefactor being the OUM funds. Thanks to Steve (have you booked your holidays yet !) Hopps.

==========

ORIC MEET RAFFLE

NOBODY VOLUNTEERED TO GET THE RAFFLE GOING FOR THE NEXT ORIC MEET. THEREFORE I WILL SET IT IN MOTION, BUT WOULD ASK THAT SOMEONE ELSE WILL SELL TICKETS ON THE DAY. FOR THOSE NOT ATTENDING, TICKETS CAN BE BOUGHT FROM ME AT O.U.M IN ADVANCE. TICKETS ARE 1 POUND EACH. FOR OVERSEAS READERS THEY CAN HAVE 2 TICKETS IN EXCHANGE FOR 1 ISSUE FROM THEIR OUM SUBSCRIPTION. SO SEND IN NOW. ESPECIALLY OUR GERMAN FRIENDS,AS THEIR POSTAL SERVICE WILL PROBABLY TAKE WEEKS TO DELIVER ANYTHING DUE TO THE THREATENED STRIKES.

WHAT ABOUT PRIZES - YES, THAT'S A GOOD IDEA. I WILL DONATE A MODEM AND SOME SOFTWARE. IF YOU OUT THERE HAVE ANYTHING TO DONATE,PLEASE SEND IT IN OR BRING IT ON THE DAY. YOUR RESPONSE WOULD BE APPRECIATED,PARTICULARLY IF YOU WANT THIS MAGAZINE TO SURVIVE!

THE 'HURLEY BURLEY' LIFE OF AN ORIC OWNER'

JOHN HURLEY from Yeovil has been kept rather busy lately,ever since I suggested that he should communicate with Keith Lester with regard to their SEKEISHO SP-1900 printers. I would say 'Hello' to Keith from John, but our intrepid Keith is not a member and so I won't.

John has sent me the program CODES for the printer. If anyone wants a copy, or indeed would like to finish it off,then John thinks Keith "will sleep again". "Hopefully for good!" - who said that - not me John - not me Dave - must be Gremlins in the printer again!

Write to OUM for a copy

Also from John I have parts 2 & 3 of the ORIC DEMO (John - why are they titled parts 3 & 4?).

I will ask Jon H if we can put all the parts together as a suite on PD.

Again from John the O.U.G 'ORIC SHOW-OFF' - a Doppelganger production from ORIC COMPUTING magazine. A nice demo and the prelude to 'SHOWING OFF AGAIN'.

Finally on the disc was COLDITZ,an adbenture from THEORIC magazine. It's partly in French and partly in English and partly 'cocked-up' by the looks of it.

Example:-

My instruction - N (North or Nord).
Reply: I don't understand "GO".

If anyone wants to have a go at sorting it out for John,please drop me a line and I will send you it on disc or cassette.

GOODY GOODY GOODRUM

David Goodrum is probably a happy man at present. His favourite artist, Marc Almond, is storming up the Pop charts with his rendition of 'THE DAYS OF PEARLY SPENCER'. - Note for more elderly readers e.g.David Wilkin. The word 'Pop' relates to popular music and not a kind of fizzy drink !!

Anyway,every so often I suggest updates to David Goodrum to improve his excellent SOFTINDEX (to be found on Public Domain).

The updated version - V4.3 has duly dropped through my letter-box. Ver good it is too. As well as some nice cosmetic changes, searches can now be done by 'Part Program title' ,&'Part supplier'.

Very handy that when you want to find a French title that you can't spell and can only remember that it starts with 'SH...'.

The program has also been speeded up again by on average 40%.

Also speeded up is David's MUSICINDEX.

Well done and thank you D.G. even if I think that David McWilliams versior of PEARLY is the better.

JUNE O.U.M

Articles for inclusion/exclusion with regard to the June issue of O.U.M should reach me by May 27th. AT THE LATEST.

PLEASE ALSO NOTE - due to holidays - articles for the JULY issue MUST be with me by JUNE 17th at the latest. It has to then to allow me to get it together.

THE ORIC SERIAL PORT Part 2

In part one we examined the uses that a serial interface can have and the way that the Oric's serial port (i.e. the modem interface) should be connected to other computers or devices. In this part I shall explain about the way the interface can be programmed. To this end I have included two BASIC programs which together will allow you to experiment with the port.

The programs are given in listings 1 and 2. You can of course simply type these in without learning how they work. But since they are pretty straightforward then it is probably worth your while trying to understand how they do work. I must first of all say that these programs will only run properly when the Oric modem interface is connected. Without it the computer will crash. The computer will also crash if you try to run listing 2 without having first run listing 1. It is also important to realise that the two programs should not be combined into one because running the first program more than once (without a total system reset) will also cause the computer to crash because of the altered interrupt vectors. Okay so let's have a look at these listings and see what they do.

The subroutine at line 100 of listing 1 loads a small piece of machine code into page 4 of the Oric's memory (below the Oric DOS link code) and lines 10 to 90 redirect the operating system interrupt handler routine. This is necessary because the serial port has been wired-up so that it makes use of the 6502 interrupt line (IRQ). Therefore whenever the serial port is used it triggers an interrupt and listing 1 is needed to handle these. Without the interrupt handler the computer will crash. If anyone wants to understand more about interrupts and how to handle them "The Advanced Oric User Guide" by Leycester Whewell gives a reasonable explanation of this topic on the Oric. Essentially, interrupts provide a way for computers to be alerted when something happens which they must give priority to. The 6502 CPU has to respond to interrupts, it has no choice which is why an interrupt handler routine is needed for the serial interface. An interrupt is generated each time a character is received and are normally used with serial interfaces to ensure that any characters being received are not missed. The serial port interrupt handler can therefore be made to save the received characters in a buffer. The routine provided here does not do this however, it simply handles the interrupts so that the Oric does not crash.

```
1 REM ** LISTING 1: HANDLES INTERRUPTS  **
5 GOSUB100       'MACHINE CODE LOADER
10 AA=DEEK(#FFFE)
20 AB=DEEK(AA+1)
30 DOKE#420,AB
40 AC=PEEK(#30E)
50 POKE#30E,AC-128
60 DOKEAA+1,#400
70 POKE#30E,AC
80 PRINT"6551 INTERRUPT HANDLER INSTALLED"
90 !LOAD"RS232.BAS"
95 END
97 '
98 '
99 REM ** SET UP M/C CODE **
100 FOR X=#400 TO #41F
110 READ DTA:POKE X,DTA
120 NEXT
130 DATA #8D,#22,#4   'STA #0422
140 DATA #AD,#81,#3   'LDA #0381
150 DATA #29,#80      'AND #$80
160 DATA #D0,#6       'BNE #06
170 DATA #AD,#22,#4   'LDA #0422
180 DATA #6C,#20,#4   'JMP #0420
190 DATA #8E,#23,#4   'STX #0423
200 DATA #8C,#24,#4   'STY #0424
210 DATA #AD,#22,#4   'LDA #0422
220 DATA #AE,#23,#4   'LDX #0423
230 DATA #AC,#24,#4   'LDY #0424
240 DATA #40          'RTI
250 RETURN
```

Line 90 of listing 1 causes the second program, listing 2, to be loaded into the computer. If you are using a cassette-based machine you will obviously need to alter this line accordingly. After typing in each program they should of course be saved and this can be done so that they will auto-run on loading.

Now on to listing 2. This is the particularly interesting program because it is the one which sets up the serial interface and operates it. I have structured it into three main parts: the main body, subroutine 100 and subroutine 200. The main body firstly calls subroutine 200 which sets up the internal registers of the serial interface main chip (6551). Line 20 then configures the screen after which subroutine 100 is called which actually makes the interface work. This subroutine first tests to see if a character has been received at the interface (line 120). If one has then it is printed to the screen. Otherwise the routine gets a character from the keyboard and sends this to the interface (lines 150-170), unless the character is ESC in which case the subroutine returns to the main body, resets the serial interface and the CAPS (in case you have toggled it off) and exits (lines 40-90).

After typing in these programs you can test them by connecting together pins 2 and 5 of the 7-pin DIN plug, as pointed out in part 1. You should find that the characters you type appear on the screen. However, note that pressing [RETURN] will take the cursor back to the start of the same line. This is not a fault. You can move down a line by pressing the down arrow.

There are 4 memory locations concerned with the serial port. These are all in page 3 of the memory map (reserved for input/output). The four locations are &380-&383. I shall explain more about these next time. If any of you wish to use this routine to link to a bulletin board then it should work fairly well. However, you will probably find that you lose some characters. One reason for this is the slow speed of BASIC (compare it with Oricomms), another is the fact that the interrupt handler here does not save the incoming characters to a buffer.

Well, that's all for now, I hope you have fun trying out these routines. If anyone has any problems or ideas they would like to discuss then you can write to me but I would appreciate it if you enclose a stamped-addressed envelope if you want a personal reply (i.e. rather than through OUM). Next time I shall explain how the serial interface internal registers can be programmed to do different things.

Trevor Shaw.

```
5 REM  ** LISTING 2: SIMPLE RECEIVE/SEND PROGRAM **
10 GOSUB 200   'SET UP INTERFACE PARAMS
20 PAPER0:INK7:CLS
30 GOSUB 100   'TRANSMIT/RECEIVE ROUTINE
40 POKE#382,0      'RESET RS232
50 POKE#20C,255    'SET CAPS ON
90 CLS:END
97 '
98 '
99 REM ** TRANSMIT/RECEIVE ROUTINE **
100 PRINT"          PRESS 'ESC' TO FINISH"
120 IF(PEEK(#381)AND8)THENPRINTCHR$(PEEK(#380));
130 IF(PEEK(#381)AND16)GOTO150
140 GOTO120
150 IN$=KEY$
160 IFIN$=CHR$(27)THENRETURN
170 IFIN$<>""THENPOKE#380,ASC(IN$)
180 GOTO120
197 '
198 '
199 REM ** SET UP INTERFACE PARAMETERS **
200 POKE#382,0      'RESET RS232 INTERFACE
210 POKE#383,22     'SET BAUD RATE ETC
220 POKE#382,5      'SET PARITY AND INTERRUPTS ETC
230 RETURN
```

DEAR DAVE,

Okay,you have found me out! Yes,I am the same Trevor Shaw who jointly won the 'COMPOSER' competition in 1984 for my arrangement of 'POPCORN'.
I bought an ORIC 1 in '84 and a copy of the program just 2 weeks before the closing date for the competition. It was partly the musical potential of the ORIC which persuaded me that it was the microcomputer I should buy - the other factors were the price and the features it offered for th price. If I could have afforded one I would probably have bought a BBC B micro, but I have never regretted buying an ORIC,although the BASIC is a little slow,it lived up to my expectations! (That's the marketing plug over with ! !)

I'll bore you with the history of my association with the ORIC computer sometime if you like!!

- TREVOR SHAW (CLEVELAND)

DEAR TREVOR,

Perhaps you could send us a copy of your rendition of POPCORN. We could use it as the opening music at the next ORIC MEET. We could perhaps put a lyric to it; something like - "Nice one Trevor,nice one son. Nice one Trevor, let's have another one". We will perhaps even hire a KARIOKE for the day!!

I am SURE an article relating to your association with the ORIC wouldn't bore anyone and would be happy to print it.
Thanks for the articles on the SERIAL PORT. I and others find it most interesting.

- DAVE

DEAR DAVE,

have you heard of a game from around 1984 called 'F.S.S LIBERATOR' by NEBULAE SOFTWARE?

- TIM GRIMWOOD (RAYLEIGH)

DEAR TIM,

Although I've heard of NEBULAE,who were famous for their CHARACTER/SOUND PACKAGE; I must admit to not knowing anything about LIBERATOR.
Well readers, does anyone out there know anything?

- DAVE

DEAR DAVE,

An interesting thing happened to me recently. I went into the Cancer Research Campaign shop in Dudley and there on a shelf were several Oric games. These were as follows:-

NOWOTNIK PUZZLE - 75 pence
GREEN X TOAD - 75 pence
LIGHT CYCLE - 50 pence
TEACH YOURSELF ORIC 1 BASIC - 50 Pence
ATMOS INTRO. TAPE - 50 pence
CIS (may have programs on it) - 50 pence.

I can get these for any reader who is interested,having checked first to see if they load. Whoever wants them should just send me the money and I will pay the postage.

- JAMES GROOM of 129 NARROW LANE,HALESOWEN,WEST MIDLANDS. B62 9NX

DEAR JAMES,

thanks for thinking of the readers and your offer for free postage. The games are a good cross-section to newer Oric users and of course 'CANCER RESEARCH' would benefit.

- DAVE

S O F T W A R E H O U S E S

BRIAN KIDD continues his list of British Software Houses who released titles for the ORIC. The Editor adds his own comments in brackets.

CDS MICROSYSTEMS - BOZY BOA (A really naf snake game), REVERSI (board game), WORDSEARCH (an excellent version of the word game that appears in the Daily Express).

CIROSOFT - CLUSTER CONTROLLER,HANGMAN/MATCH,MERCHANT PRINCE, NIGHTRIDER (quite a good one that), PARACHUTIST/FRUIT.

COMPUSOUND - MOLE CAPTURE

COMPUTAMASTER - FRUIT/PONTOON

COMPUTASOLVE - MINER

CRL - ESCAPE FROM MANHATTAN,RESCUE,SPACE MISSION (All quite fair)

DORMERE - ORICAL BACKGAMMON, ORICAL INVADERS (A goodish version).

DREAM - BLOCKBUSTER,FROGEE,GRAPHICS PACKAGE,MATHS GAMES

DURELL - ASS/DISSASS, CHICKEN, DEMON, GALAXY FIVE (4 games i think!), HARRIER ATTACK (Their best known), JUNGLE TROUBLE (Was it released!), LYNX ATTACK (I don't think this was released),ORICSTAR, SCUBA DIVE (underwater fun), STARFIGHTER (the best thing about this was the fact that it never got updated for the Atmos).

ELEPHANT - FARMER HORACE, JERICO 2 (Was there a JERICO 1 ?), JUMPER, PARAGRAM, RINGO (A puzzle with rings), VANQUISHER.

ELTRONICS - GRAPHICS DESIGNER

EMERALD - DELTA 4 (from the pen of Andy 'GRUN' Green of ULTIMA ZONE fame This was also something to do with the NECTARINE label)

EPSILON - 10K OTHELLO;10K STAR TREK,12K BACKGAMMON,3D NOUGHTS & CROSSES, BATTLESHIP CONTROL.

EXPRESS - BANDIT,BRAD THE PROFESSOR,BREAKOUT,MONTE CARLO RALLY,SPACE QUEST.

FBC SYSTEMS - WRIGGLER

FGC/HGC - BUCCANEER,CHUCKFORD,FLY FOR YOUR LIFE,FOOTBALL (From the pen of Alistair Way),FGC UTILITIES,SAFE STOP/AUTO STOP,GOLDMINE,WINTER GAMES, WORDWORTH,STARWIND,MAZE RALLY,SNAKE VENOM,THE LAST WARRIOR,LEAGUE CHAMPIONS, INDIANA JONES (From the pen of Graeme Burton).

G S TEAL - TAPE COPIER

GEM - GOLF,MASTER MINE,SULTANS MAZE

GEMINI - DATA BASE,SPREAD SHEET

GILSOFT - THE QUILL (Aid To writing adventures)

HARESOFT - HARE RAISER (Was there a sequel?)

HEADFIELD - CHAR/GRAPHIC GENERATOR,CLICK,CODE BREAKER,DEATH TRAP,GALATIC DEFEND,KNIGHTS,MASTER MYNDE,THE WORLD,TOADY

HONEYFOLD - ASSEMBLY LANG.,BASIC PROGRAMMING,BEGINNERS ASSEMBLY LANGUAGE

..... AND AS THE CREDITS ROLL, WE TAKE OUR LEAVE WITH THIS QUOTE FROM THE OLD HIT T.V. SERIES - "To be Continued..."

Rambling on... with GOTO, POP, RETURN and DATA

'GOTO' (COMMAND)

Principal:

The search for the destination line number starts at the following line if the number is greater than the current line number, so as to accelerate the search.
Anything following the destination line number is ignored and does not produce a syntax error.

Bug: The test to see if the search should start at the next line is done only on the high byte! So in the following example the search willl start at the beginning of the program, since the high bytes of both line numbers are the same!

Example: 200 GOTO 250 – the search begins at the start of the program; compare
200 GOTO 260 – the search starts on the next line!
In any event it's a curious short–cut to program into the ROM.

C9B3	JSR $E79D	C9E5	JSR $E853	Evaluate the line number in #33–#34
C9B6	JSR $CA1F	C9E8	JSR $CA51	and find the end of the line
C9B9	LDA A9	C9EB	LDA A9	Take high byte of current number
C9BB	CMP 34	C9ED	CMP 34	and compare to the number sought
C9BD	BCS C9CA	C9EF	BCS C9FC	If lower or equal, start at the beginning
C9BF	TYA	C9F1	TYA	If not, adjust TXTPTR
C9C0	SEC	C9F2	SEC	to the next line
C9C1	ADC E9	C9F3	ADC E9	
C9C3	LDX EA	C9F5	LDX EA	to enable the search to start
C9C5	BCC C9CE	C9F7	BCC CA00	at the next line rather than the beginning
C9C7	INX	C9F9	INX	of the program
C9C8	BCS C9CE	C9FA	BCC CA00	Unconditional
C9CA	LDA 9A	C9FC	LDA 9A	Take start of program as
C9CC	LDX 9B	C9FE	LDX 9B	start of search
C9CE	JSR $C6E8	CA00	JSR $C6BD	and look for the line address
C9D1	BCC C9F1	CA03	BCC CA23	if not found, error
C9D3	LDA CE	CA05	LDA CE	adjust TXTPTR
C9D5	SBC #01	CA07	SBC #01	to the start of the line
C9D7	STA E9	CA09	STA E9	
C9D9	LDA CF	CA0B	LDA CF	(#CE–#CF points to after
C9DB	SBC #00	CA0D	SBC #00	the 0 of the start of the line)
C9DD	STA EA	CA0F	STA EA	
C9DF	RTS	CA11	RTS	

'POP' (COMMAND)

Principal:

The POP command is treated as a RETURN, the only difference being the positioning of TXTPTR. When the command is issued, Y contains the duplicate of the BASIC token. This permits the routine to distinguish two commands with the same ROM address (as here). This procedure is also used by the sound and graphics commands, and by PULL/UNTIL.

After adjusting the stack, execution simply continues without touching TXTPTR.

'RETURN' (COMMAND)

Principal:

The routine looks for a GOSUB block on the stack and recovers the values of TXTPTR and the line number. It then recovers the return address to the interpreter which was left in place by GOSUB.

The GOSUB put TXTPTR on the stack before evaluating the line number. This explains why one calculates the end of the instruction via DATA.

Since GOTO and RETURN cannot be followed by any other instruction, one directly finds the end of the line via DATA.

The search for the first GOSUB block is done avoiding any FOR blocks by simulating the search for a FOR block which would have an index address above #FF00, which is impossible. Thus the search stops on the first block which is not a FOR block.

```
C9E0  BNE C9DF     CA12  BNE CA11     Exit if parameters present
C9E2  LDA #FF      CA14  LDA #FF      Specify an address
C9E4  STA B9       CA16  STA B9       that is impossible
C9E6  JSR $C3CA    CA18  JSR $C3C6    and so jump all redundant FOR blocks
C9E9  TXS          CA1B  TXS          and adjust S above it
C9EA  CMP #9B      CA1C  CMP #9B      is it a GOSUB block?
C9EC  BEQ C9F9     CA1E  BEQ CA2B     yes, Ok
C9EE  LDX #16      CA20  LDX #16      'RETURN WITHOUT GOSUB ERROR'
C9F0  BYT $2C      CA22  BYT $2C      jump next instruction
C9F1  LDX #$5A     CA23  LDX #$5A     'UNDEF'D STATEMENT ERROR'
C9F3  JMP $C485    CA25  JMP $C47E
C9F6  JMP $CFE4    CA28  JMP $D070    'SYNTAX ERROR'
C9F9  PLA          CA2B  PLA          Take GOSUB code off the stack
C9FA  PLA          CA2C  PLA          and low byte of line number
C9FB  CPY #0C      CA2D  CPY #0C      is it POP? (#86*2=#10C--> #0C)
C9FD  BEQ CA18     CA2F  BEQ CA4A     yes, jump
C9FF  STA A8       CA31  STA A8       save line number low byte
CA01  PLA          CA33  PLA
CA02  STA A9       CA34  STA A9       and high byte
CA04  PLA          CA36  PLA          then TXTPTR low byte
CA05  STA E9       CA37  STA E9
CA07  PLA          CA39  PLA          and high byte
CA08  STA EA       CA3A  STA EA
```

'DATA' (COMMAND)

Principal:

The DATA instruction does nothing save to slow down program execution; it is necessary only to jump it. READ permits the data to be read. It is therefore better to put the DATA lines at the end of a program.

```
CA0A JSR $CA1C   CA3C JSR $CA4E   Move to the next instruction
CA0D TYA         CA3F TYA         and adjust
CA0E CLC         CA40 CLC
CA0F ADC E9      CA41 ADC E9      TXTPTR in consequence
CA11 STA E9      CA43 STA E9
CA13 BCC CA17    CA45 BCC CA49
CA15 INC EA      CA47 INC EA
CA17 RTS         CA49 RTS         and return to the interpreter
```

'POP' (continued)

```
CA18 PLA         CA4A PLA         Take off the stack for adjustment
CA19 PLA         CA4B PLA
CA1A PLA         CA4C PLA         and simply exit
CA1B RTS         CA4D RTS
```

Tailender...

An apology to all who have had to wait for orders; I've been exceptionally busy at work this last couple of months, and what with CEO mags and discs, plus covers and RAMROM, something had to suffer. It's Easter as I'm writing this, and I promise all will be up to date forthwith. As I've said before, if you've not received something you should have, drop me a line, I usually feel so guilty I add a little something...

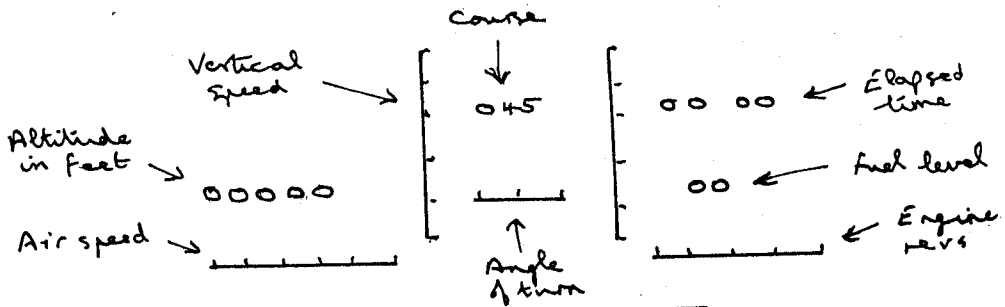
And another...

This month's cover returns to the OUM days of yore with an ad for software. Flight Sim is probably as good a simulator as we're going to see on the Oric. It has a true area to fly over, with line drawn 3D objects to fly to, round and under (!), full controls for flying, and it all runs at a good speed – more than can be said of one program with a similar name! To whet your appetite, the control panel scheme is illustrated on the next page. On disc it costs £8.99 together with Mr. President and Robinson Crusoe, on cassette it's £4.99 with Mr. President only. **New** is CEOSOFT 5, MAH-JONG and ORIC SWAP, two quite different board games played on a 24 x 10 squares ..er..board. Good time-absorbing stuff from Laurent Chiacchierini. And the price – £3.99 on cassette, £4.99 on disc. Three and a half inchers deduct £1 from the disc price as usual. Orders to me at 3, Madingley Road, Cambridge CB3 0EE, cheques payable to me please.

I reckon JoJo's MUSED is worth a look too...I hope you had a relaxing Easter

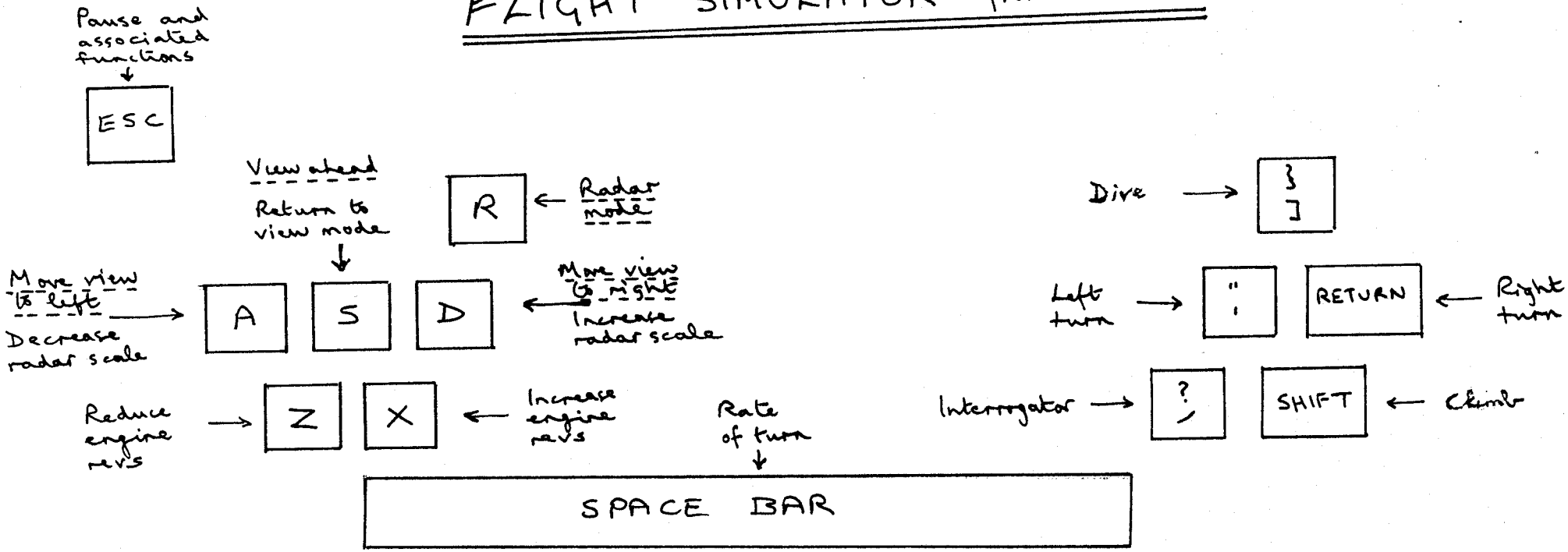
Jon Haworth

Above the screen window is shown the direction of view :
FO = FORWARD
RI = RIGHT
LE = LEFT
RE = REAR



CONTROL PANEL DISPLAY

FLIGHT SIMULATOR from CEO



KEY

1) Radar mode
2) View mode

KEYBOARD LAYOUT

ORIC Enthusiasts (OUM 57)

ORIC DEVELOPMENT

Thanks go to Robert Crisp for an interesting letter. He too, would wish to see the development of the ROM to provide a separate Machine Operating System (MOS or BIOS if you are talking IBM PC parlance) with extra ROMs providing a structured BASIC, machine code assembler/disassembler, etc. Additionally, Robert is keen on seeing the introduction of utilities for the transfer of information to and from the ATMOS to the IBM PC. Although it is still early days, there is an emerging preferred path for the development of the ATMOS, so as from May I will be starting putting a draft proposal together for this work. I will keep the interested parties informed of the progress.

BOOKS

Continuing with the series on books available from my price list. The next title is "Games To Play On Your ORIC-1", by Czes Kosniowski and published by Shiva. Guess when this book was first published! Yep, you got it! 1983. It was obviously a good year for publishing books about the ORIC-1. Or was it? Did many books teach the eager hobbyist trying to get to grips with his trusty micro and its questionable programming manual? Well if they did not, now is the chance to catch up.

This book is aimed fairly and squarely at the emerging programmer and aims to teach structured BASIC programming by the use of games listings. My judgement is that it accomplishes this task very well. All the listings are clear and easy to follow. The author makes good use of the structured commands within ORIC BASIC, refrains from the dreaded GOTO command and makes use of variables in pointing to sub-routines. This means that programs are legible and can easily be modified if renumbering is carried out. The book claims that each program is 100% bug free and robust. Although I haven't been able to key in all the programs, study of the listings does suggest that it is not an idle boast.

Each listing is accompanied by a general description of the program and its aim, a representation of the screen output, followed by more detailed accounts of the main routine and its associated sub-routines. All in all, this is one of the best books for the ORIC, in terms of layout of programs and educational value when it comes to teaching how to program in BASIC.

The book contains 128 pages with listings for 30 programs covering four types of games; arcade-type, demonstrations (using sound and colour), puzzles and brain drains. Appendices cover how to save the contents of a HIRES screen, a renumber utility, information on the standard and alternate character sets and finally, how to produce music from the ORIC.

The titles are as follows. TREASURE HUNT (find the treasure behind the screen characters); DON'T CRY WOLF (avoid the forest wolves and reach home); MOON LANDING (an instruments only version of this popular early game); BIRD AND CATERPILLAR (the proverbial bug-eater, mushroom mania, caterpillar game); TWENTY-FIVE SQUARES (a puzzle where you rearrange the letters of the alphabet); INTELLIGENCE TEST? (crack the sequences generated by the ORIC); HYPNOTIC (graphic/sound demo); DANCING MEN (variation on the theme of the caterpillar game); GRAPHICS SHOW (line graphics demo with an equation for producing arcs); SQUARE NIM (game of strategy based upon the 'pick up sticks' game); OH! SUSANNA (music/graphics demo): STOP 'EM (a game to test your reactions); ROUND AND ROUND (graphics demo showing polar plots - poles apart from the GRAPHICS SHOW!); CASCADE (catch the balls that drop from the top of the screen, sometimes bouncing off obstacles); LA CUCARACHA (music demo with simple graphics); CLOCK PUZZLE (a sliding blocks type puzzle); BLANK OUT (simple hand-eye co-ordination program); SURFACES (3D surface plots); BLUE DANUBE (music demo); SNAKE (another variation on the caterpillar program); SPACE CHASE (Asteroids type game); SPELL IT (memory game); SKETCH PAD (ETCH-A-SKETCH type program); BOUNCE AWAY (a BREAKOUT game); SPIRALS (graphics demo); KNEES UP MOTHER BROWN! (well I never! this is a karaoke program on the trusty ORIC - going back to 1983); MEMORY CARDS (memory game); HANGMAN (the proverbial word game); FIGURE IT OUT (the proverbial Mastermind game) and DON'T BE GREEDY (dice game).

All in all, a variety of games to while away the hours but none at a standard that will take your breath away, but then that is not the intent. The book is designed to get the user to see how games are written in order to develop the masterpieces of their own. I must issue the usual proviso about the applicability of this book to the V1.1 ROM but since all th programs are written in BASIC it is an easy task to convert the PLOT commands. Remember to add 1 to the 'x' position in the PLOT command so that the PAPER attributes are not overwritten unintentionally. The book is available at £1.00, inclusive of postage and packaging, (original price was £4.95)o

ORIC ATMOS and ORIC-1 GRAPHICS & MACHINE CODE TECHNIQUES

Chapter 3 - BASIC (continued)

3.10 Machine code advice As mentioned previously, a book on machine code is essential, not only to teach the subject but as a constant guide to the 6502. This section covers some of the more error-prone areas of programming, in the hope that you may learn from my own mistakes!

BRANCHES The following observations may be useful:

1 Any branch will depend on one bit within the processor status register. Branch instructions work in pairs, e.g. BEQ, BNE; BCS, BCC.

2 The operand in the branch instruction gives the number of bytes, forward or backward, to jump. If this number is between 0 and #7F the branch is forward in memory; otherwise the jump is to a previous location. When a backward branch is required the operand is #100 minus the number of locations that you are jumping. For example: 1200 BNE 11C2 results in an operand of (#100-(#1202-#11C2)) i.e. #C0.

Any good machine code monitor will work out branch offsets for you. An assembler will allow you to enter either an absolute address or a meaningful label.

COMPARE A newcomer to 6502 programming can become confused with the CMP instruction when testing less-than or greater-than conditions. The compare instruction works in a similar way to subtract as regards the use of the carry flag. When a subtraction is done, the carry flag is used to indicate a borrow when the value being subtracted is greater than the accumulator. The advantage of the compare instruction is that the A, X and Y registers are not affected. When writing a compare instruction do a mental subtraction of the value given in the instruction from the register value (A, X or Y). If the result is zero, the zero flag is set. If the result is positive, including zero, the carry flag is set; otherwise it is cleared.

THE BIT INSTRUCTION BIT is probably the least used of all the instructions - CMP is often used instead. Like the compare instruction, BIT only alters flags in the process status register. If you wanted to examine a number of locations, picking out one bit, then you would load the accumulator with the bits to examine and just use BIT with each address. If you used the AND instruction, you would need to keep re-loading the accumulator. BIT also traps bits 6 and 7 of the location you are examining, reflecting them in the overflow and negative flags. Because BIT does not affect the A, X and Y registers, you can use BIT in a sneaky way to conserve memory.

Consider the program:

```
TRY1      LDA    #1
          BNE    CARRY-ON
TRY2      LDA    #2
          BNE    CARRY-ON
TRY3      LDA    #3
CARRY-ON
```

This can be replaced by:

```
TRY1      LDA    #1
          #2C
TRY2      LDA    #2
          #2C
TRY3      LDA    #3
CARRY-ON
```

The '2C' is the opcode for the 3-byte version of BIT. Here we use the fact that BIT does not alter the accumulator in order to skip past one or two load instructions. You will find this kind of confusing programming when you disassemble the ORIC's ROM. The saving is so small as to be not worth the trouble, but it does demonstrate an interesting programming technique.

THE STACK When using the stack remember:

1 In a sub-routine you must leave the stack as you find it. This means that if you execute 5 PHA instructions, you must balance them with 5 PLA instructions. This is important because the RTS instruction will be expecting a return address on the stack.

2 To follow up the last point, here is a common mistake:

```
1000 PHP
1001 JSR 1234
1234 PLP ; attempt to pass processor stack.
```

3 When saving all the registers on the stack, use a sequence such as:

```
PHP  PHA  TXA  PHA  TYA  PHA
```

When you want to restore the registers, remember to reverse the order:
PLA TAY PLA TAX PLA PLP.
If you are saving an area of memory on the stack you will need to reverse the loop when loading back from the stack. For example, if this is your save routine:

```
      LDX #F
A     LDA 2E0,X
      PHA
      DEX
      BPL A
```

then the reverse procedure is:

```
      LDX #0
A     PLA
      STA 2E0,X
      INX
      CPX #10
      BNE A
```

The stack provides the only way of examining the complete processor status register:
PHP PLA
Similarly, to set up the processor status register in one go:
LDA #47 PHA PLP

DECIMAL INSTRUCTIONS When a program goes unaccountably wrong always consider the state of the decimal flag. The normal state for the decimal flag is off. Many ROM sub-routines will expect the decimal flag to be cleared, so remember the CLD instruction. The decimal flag is only recognises when using either the ADD or SBC instructions, whereas INC and DEC will always work in binary.

SHIFT AND ROTATE When using any of the shift or rotate instructions, remember:

1 There is always one bit coming away from the byte. This is always saved in the carry flag.

2 There is always one bit coming into the byte. This is either zero for shift instructions or the old carry flag for rotate instructions.

3 The rotate instructions work on 9 bits at a time. Therefore, if you rotate 0000 0001 to the right, the 1 will not appear on the left until a further rotate instruction.

CLEAR CARRY AND SET CARRY Two simple rules apply here:

1 Clear the carry flag before doing an addition. If adding numbers longer than 8 bits, leave carry alone after the first clear carry instruction; for example:

```
      CLC
      LDA 0
      ADC 2
      STA 0
      LDA 1
      ADC 3
      STA 1
```

2 When subtracting, set the carry first of all. The carry will be automatically set or cleared to indicate a borrow (if clear then there is a borrow).

INCREMENT AND DECREMENT Important points:

1 INC and DEC take no notice of the decimal flag - they always work in binary.

2 INC and DEC do not either use or alter the carry flag. If you want to increment a 16-bit value, use a branch instruction, as in:

```
      INC 42
      BNE B
      INC 43
B     NOP
```

When decrementing numbers, you will have to use a compare instruction:

```
      DEC 42
      LDA 42
      CMP #FF
      BNE C
      DEC 43
C     NOP
```

Next month -- completion of 3.10 & some more of Chapter 3

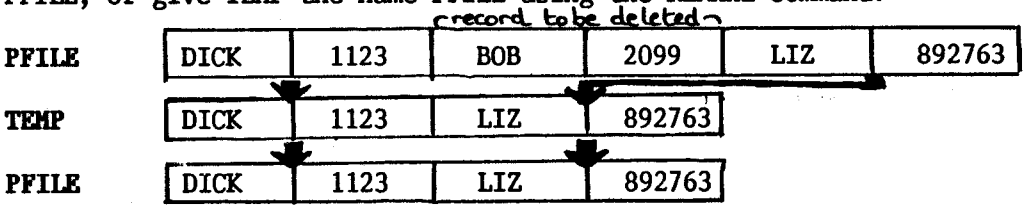
DISC FILE HANDLING TECHNIQUES USING BASIC

UNSORTED SEQUENTIAL FILES (continued)

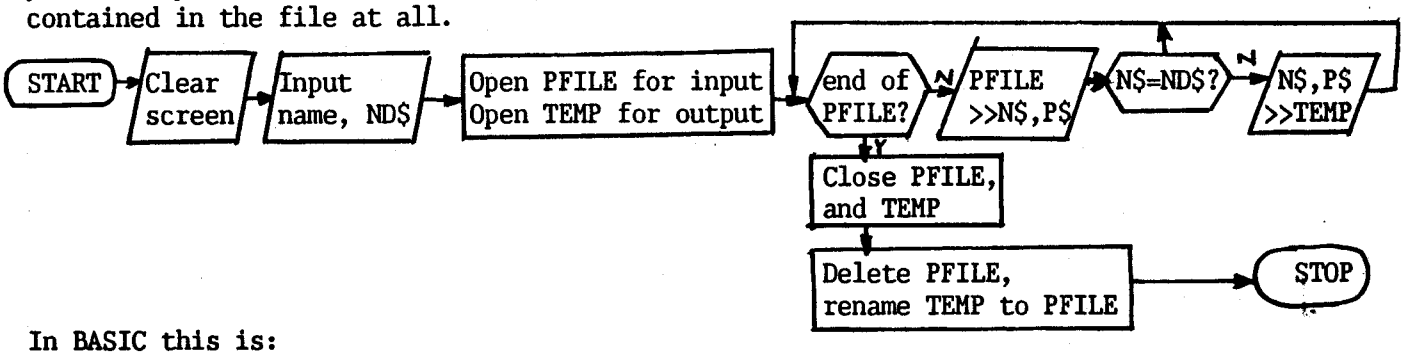
This month sees the completion of the Search routine and the Delete Record routine. The program lines to complete the Search routine are :-

```
MICROSOFT    4050 INPUT#1,N$,P$
RANDOS       4050 !GET 1,N$:!GET 1,P$
SEDORIC      4050 TAKE 1,P$,N$
             4060 IF N$<>S$ THEN 4040
             4070 PRINT "FOUND:":PRINT N$,P$
             4080 GOTO 4100
             4090 PRINT "NAME NOT FOUND"
             4100 INPUT "CONTINUE (Y/N)";A$
             4110 IF A$<>"Y" THEN 4100
             4120 GOSUB 1300:        REM CLOSE PFILE
             4130 RETURN
```

Deleting a Record To remove a record from the file PFILE, you open PFILE for reading and also an auxiliary file TEMP for writing. You read a record and compare it with the one to be deleted. If they are the same, you move on to the next record; if not, you write the record to the auxiliary file. Finally, you can either copy all the records from TEMP to PFILE, or give TEMP the name PFILE using the RENAME command.



The flow chart is as follows, with the name to be deleted assigned to the variable ND$. If you wish you could have the message NAME NOT FOUND appear on the screen if the name is not contained in the file at all.

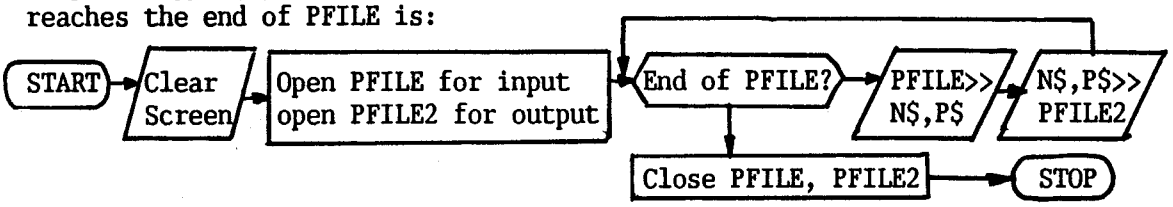


In BASIC this is:

```
             5000 REM ----- DELETE NAME AND NUMBER -----
             5010 CLS
             5020 INPUT"NAME TO DELETE";ND$
             5030 GOSUB 2200:    REM OPEN PFILE FOR INPUT
             5040 GOSUB 5300:    REM OPEN TEMP FOR OUTPUT
MICROSOFT    5050 IF EOF(1) THEN 5100
RANDOS       5050 IF IF PEEK(#4FF)=29 THEN 5100
SEDORIC      5050 IF (-&(1)) THEN 5100
MICROSOFT    5060 INPUT#1,N$,P$
RANDOS       5060 !GET1,N$,P$
SEDORIC      5060 TAKE1,N$,P$
             5070 IF N$=ND$ THEN 5050
MICROSOFT    5080 PRINT#2,N$:PRINT#2,P$
RANDOS       5080 !PUT2,N$,P$
SEDORIC      5080 PUT2,N$,P$
```

```
            5090 GOTO 5050
            5100 GOSUB 5400:   REM CLOSE TEMP
            5110 GOSUB 5500:   REM DELETE PFILE
            5120 GOSUB 5600:   REM RENAME TEMP TO PFILE
            5130 RETURN
            5140 REM
            5300 REM ----- OPEN TEMP FOR OUTPUT -----
MICROSOFT   5310 OPEN "O",#2,"TEMP"
RANDOS      5310 !OPEN "TEMP",1,W
SEDORIC     5310 OPEN S,"TEMP",1
            5320 RETURN
            5400 REM ----- CLOSE TEMP -----
MICROSOFT   5410 CLOSE 2
RANDOS      5410 !CLOSE2
SEDORIC     5410 CLOSE2
            5420 RETURN
            5500 REM ----- DELETE PFILE -----
MICROSOFT   5510 CLOSE 1:KILL F$
RANDOS      5510 !CLOSE 1:!DEL F$
SEDORIC     5510 CLOSE1:DEL F$
            5520 RETURN
            5600 REM ----- RENAME TEMP TO PFILE ------
MICROSOFT   5510 NAME "TEMP" AS F$
RANDOS      5510 !REN "TEMP" TO F$
SEDORIC     5510 REN "TEMP" TO F$
            5520 RETURN
```

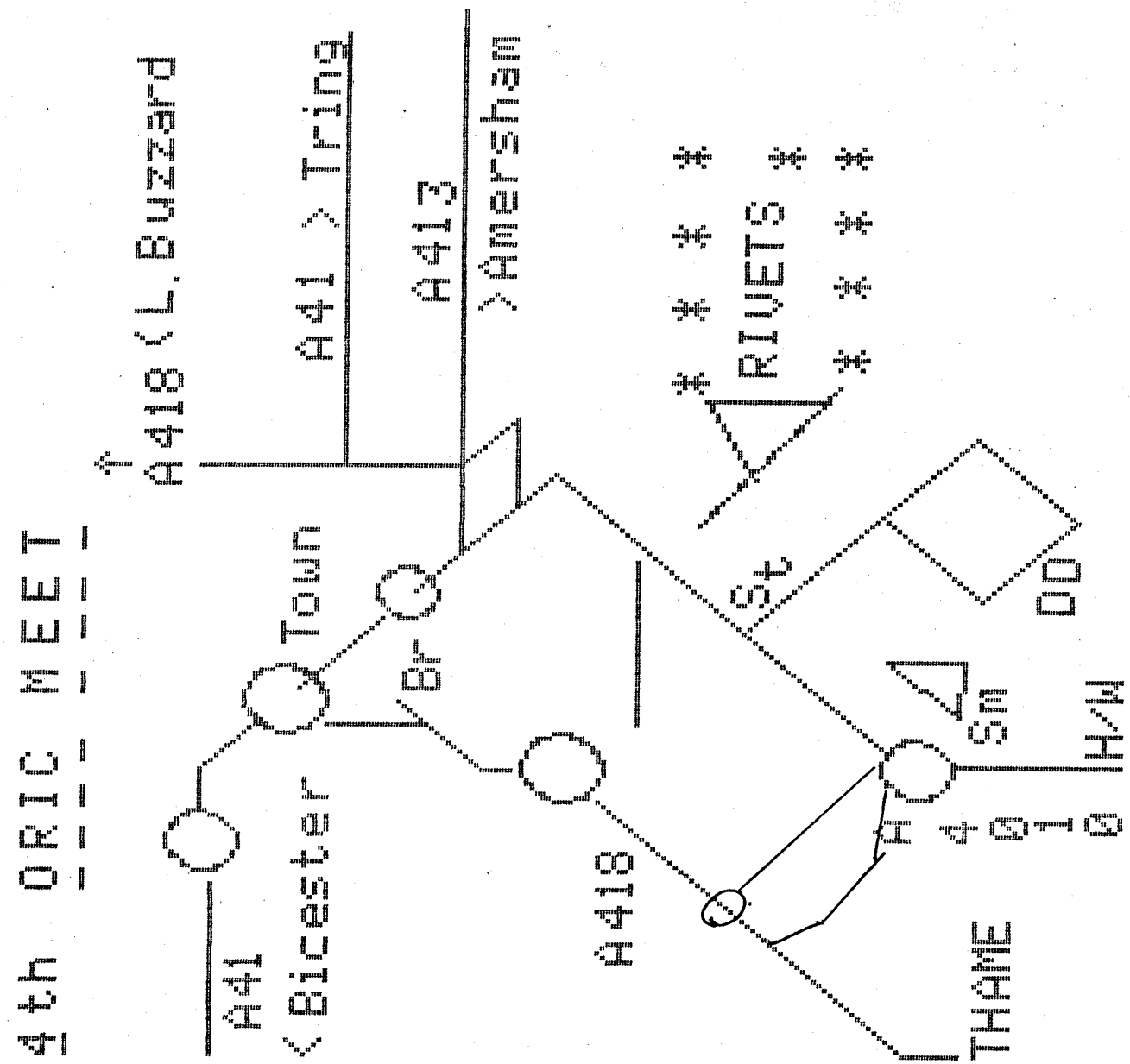
Copying the file PFILE At the end of this chapter we will now consider how to copy the file PFILE into a second one called PFILE2. This is important from the point of view of data security. If you have a second disc drive, you can develop the program to copy the file PFILE2 onto a separate disc, which can be kept in a safe place. The flow diagram for a simple copying process which continues to write records from PFILE to PFILE2 until it reaches the end of PFILE is:



As you will probably have realised this seems an unnecessary way of producing a copy when SEDORIC has a 'copy file' command. The fact of the matter is Microsoft BASIC does not! The book I'm using as reference gives a program that matches the flow diagram above but this is really ludicrous. It would be better to use the SHELL command and make use of the MS-DOS utility COPY to do the work much quicker. So I have written the program as follows:

```
            6000 REM ----- COPY PFILE TO PFILE2
            6010 CLS
MICROSOFT   6020 SHELL
MICROSOFT   6021 COPY PFILE PFILE2
MICROSOFT   6022 EXIT
RANDOS      6020 !COPY F$ TO F$+"2"
SEDORIC     6020 COPY F$ TO F$+"2"
            6030 RETURN
```

Well, this concludes the program that se set to write to demonstrate the use of unsorted sequential data files. You should now have a crude but working program that can create a file, read and print the file, append new records to it, write a record, delete a record and, finally, copy a file to another. Next month I'll look at the program to see if I can tidy it up and perhaps emphasise some of the strengths of SEDORIC DOS.



\==========

AYLESBURY IS THE POINT WHERE ALL ROADS SEEM TO MEET AND CAN BE A LIITLE DAUNTING WHEN DRIVING THROUGH IT.
THE A 418 COMES FROM LEIGHTON BUZZARD IN THE NORTH ,WHICH LEADS TO MILTON KEYNES AND THE M1. THE A418 CONTINUES SOUTH TO THAME AND THEN MEETS UP WITH THE M40 , JUST EAST OF OXFORD.
THE A41 IS FOR DRIVERS COMING OFF THE M40 FROM BIRMINGHAM AT JUNCTION 9. DRIVERS FROM LONDON AND THE SOUTH EAST CAN TAKE EITHER THE A41 VIA TRING OR THE A4010 (ONTO THE M40 AT HIGH WYCOMBE) OR THE A413 VIA AMERSHAM.

KEY

*** RIVETS *** = MEETING PLACE
St = STARS NEWSAGENTS (ON THE CORNER OF HARVEY ROAD).
DD = DAVE DICK's HOUSE AT 65 BARNARD CRESCENT, WHICH CONTINUES FROM HARVEY RD.
Sm = STOKE MANDEVILLE HOSPITAL
Br = BRITISH RAIL STATION AT AYLESBURY.