ØRIC USER MONTHLY

with Oric Enthusiasts

Europe's longest running Oric Magazine

Number 59
July 1992

O.U.M. MEETING

Saturday 18th July

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

THE EDITORIAL

BONJOUR,HELLO , AND WELCOME TO THE JULY ISSUE.
I ONLY HOPE I HAVE A BETTER JULY THAN JUNE IS TURNING OUT TO BE.

A NEW DISC INTERFACE THAT WOULDN'T WORK; A NEW 3" DRIVE THAT WOULDN'T FORMAT DISCS - IT READS THEM ALL RIGHT; THE DATA FILE ON PAGE 3 OF THIS ISSUE GOT CORRUPTED (a quick bit of editing,but still not perfect). THEN TO TOP IT ALL I INJURED MY BACK.

IN THE END EVERYTHING HAS TURNED OUT REASONABLY WELL. THANKS TO YOUNG MATTHEW FOR COLLATING THIS AND THE JUNE ISSUE OF OUM - 2 HOURS BACK-BREAKING WORK PER ISSUE,FOR WHICH HE TOOK A BRIBE! THANKS ALSO TO CHRIS HEARN FOR GIVING UP A SUNDAY AFTERNOON TO HELP ME SORT OUT MY STOCK AND FOR THE SOLDERING WORK. DOES ANYONE WANT TO BUY A CAR RADIO/CASSETTE IN NON-WORKING ORDER!

A QUICK 'CTRL Z'. NOW A 'CTRL X'. SO FAR SO GOOD - THIS PAGE SHOULD COME OUT O.K.!

THANK YOU FOR THE LETTERS ABOUT THE LAST ISSUE. THREE OUT OF FOUR SAID THAT IT DID TURN OUT O.K. THE OTHER ONE HAS A CAT THAT DOES NOT LIKE 'WHISKAS'.

ALL ORDERS SHOULD BE DEALT WITH BY THE END OF JUNE AND MOST OF YOUR LETTERS.

I'M AFRAID BACK ISSUES WILL HAVE TO WAIT ANOTHER MONTH. MOST LETTERS WILL HAVE BEEN DEALT WITH.

IT IS HOPED THAT FROM SEPTEMBER THAT FRANK BOLTON WILL TAKE OVER THE JOB OF CASSETTE DUPLICATION ON 'MIRAGE' TITLES. THANKS FRANK - THE CHEQUE IS IN THE POST.

THIS ISSUE WILL I'M AFRAID CONTAIN ONLY 21 PAGES. NO A-Z OF SOFTWARE HOUSES, NO LISTINGS, AND NOT MUCH IN THE WAY OF GAMES/ADVENTURE HELP. ALL WILL BE PUT RIGHT FOR THE SUPER DUPER DOUBLE 5th BIRTHDAY ISSUE DUE OUT AT THE BEGINNING OF SEPTEMBER.

DO NOT FORGET - YOU WILL NOT RECIEVE AN AUGUST ISSUE.

OUM DISC

I HAVE HAD A FEW REPLIES TO THE SUGGESTION ABOUT AN OUM DISC. I HAVE ALSO GATHERED SOME PROGRAMS FOR IT. UTILITIES/LISTINGS THAT FRANK BOLTON HAS BEEN COLLECTING,PLUS SOME ITEMS THAT HE HAS WRITTEN. THERE WILL ALSO BE MANY OTHER ITEMS NEW TO MOST OF YOU. I THINK IT WILL BE WORTH THE EFFORT,BUT ONLY IF A FEW MORE OF YOU WRITE TO SAY THAT YOU WANT IT. DROP ME A LINE OR TELL ME AT THE 'ORIC MEET'.

AU REVOIR

WELL THAT'S YER LOT. I'M OFF TO ENJOY MY VACATION. JUST IN CASE WE GET SOME WET EVENINGS,WE ARE TAKING THE ORIC. NOT FOR WORK,BUT PURELY FOR PLEASURE. MATTHEW WANTS TO COMPLETE 'ZEBULON'. THE WIFE WANTS TO COMPLETE 'MAHJONG'. LOUISE WILL BE HAPPY ZAPPING 'EM UP ON 'ULTIMA ZONE', AND I WILL PROBABLY NOT GET A LOOK IN!

PERHAPS WE HAD BETTER TAKE TWO!

INDEX

BULLETIN BOARD

I recently recieved a telephone call from Chris Dalby of Bradford. He had dug my number out of an old issue of the now defunct - 'New Computer Express'.

Chris is setting up a Bulletin Board for non-mainstream computers. He runs a QL and is able to read Data files on 3.5" disc. He is offering to put on an ORIC section. He is already running a test board. I've agreed to send him the OUM pack and details of the next ORIC MEET, just in case some Oric users happen to log on. If you want an Oric section on the board, then please write to me. It is no good putting in a lot of effort for just one or two. If this takes off, then we will need a volunteer to edit the Oric section.

The system runs on VIEWDATA (same as Prestel), and I would assume that the Prestel software available on Public Domain would allow one to log on. The board is a 'ring back' one - ring twice and ring back. Number is the same as Chris's telephone number - 0274 487601.

Chris's address is : 74 St.Leonards Rd., Bradford, West Yorkshire. BD8 9QF.

Why not try him out and DON'T FORGET to let me know if you are interested. I have a few of the Oric Modems in stock.

OUM OFFICE

OUM IS NOW CLOSED UNTIL JULY 13th. IF YOU HAVE MONIES TO SEND FOR ORDERS / SUBSCRIPTIONS etc., PLEASE TIME THEM TO ARRIVE ON OR AFTER THAT DATE.

PRIZE WINNER

THIS MONTH'S LUCKY WINNER IS MEMBER NO.135, WHO IS DENIS BONFIELD FROMF DEAR OLD CRICKLEWOOD. HE RECIEVES A FREE COPY OF OUM.

ORIC MEET UPDATE

THE 4th ORIC MEET IN AYLESBURY ON SAT.JULY 18th LOOKS LIKE BEING THE BIGGEST EVER. SWELLING THE ATTENDANCE ARE JOHN HURLEY FROM SOMERSET AND HENRY MARKE AND HIS WIFE FROM PORTSMOUTH. WITH QUITE A FEW WIFES/GIRLFRIENDS COMING TOO ,IT LOOKS AS THOUGH A SHOPPING TRIP WILL TAKE PLACE. HENRY'S WIFE (RENE) WIL PROBABLY SETTLE HER SELF DOWN AT AN ORIC PLAYING' TETRIX' ALL DAY

HAVE RECIEVED FROM TREVOR SHAW AN AUDIO TAPE CONTAINING HIS 'POPCORN' PIECE ,WHICH WAS A WINNER IN THE 'COMPOSER' COMPETITION. ALSO ON THE TAPE ARE OTHER WINNERS, INCLUDING A RENDITION OF 'THE SABRE DANCE'. I'LL BRING ALONG THE HI-FI SO THAT YOU CAN ALL LISTEN TO THEM OVER LUNCH.

RON EVANS FROM WOOD GREEN IS TRYING TO PERSUADE HIS WIFE TO COME WITH HIM.

GRAEME BURTON POPPED IN RECENTLY. AMONGST THE GOODIES HE LEFT FOR THE RAFFLE WAS A BOTTLE OF 'OLD F..T'. SEE RAMROM FOR FULL SPELLING.

PARALLEL PORT

STUART WRIGHT ASKS IF THESE PARALLEL PORT HARD DRIVES CAN BE USED WITH OUR ORICS. HE THINKS IT MAY BE A TASK FOR Dr. RAY!

CAR MADNESS

'CAR MADNESS' FROM STAALE EIKBRAATEN OF NORWAY HAS JUST ARRIVED. IT IS NOT YET A FINISHED GAME. SIMILAR IN IDEA TO 'DRIVER' etc. I WILL KEEP YOU POSTED

RAMBLING IN THE ROM - 40

Those were the days...

I mentioned some while ago that I had had a chat with Paul Kaufman about the early days of Oric, in preparation for a new edition of 'The Story So Far' (still available at £1.50, etc...). The chances of a new edition in the near future are remote, so I thought that this month I might share a few of Paul's revelations about the formative years.

When Tangerine decided to produce an integrated design computer, the original aim was to go for an executive desktop machine which would link to Prestel and compute. Paul wrote a memo listing what he thought were the right features for the 'Microtan 2' - sound and graphics, a modulator and so forth. The result was a design in late 1981 for the 'Tangerine Tiger', a desktop machine with three processors - a Z80 for CP/M, a 6809 for I/O, disc and printing, and a graphics chip. In the end this design was sold off to a company called H.H. Electronics, and was never in fact produced. The Microtan 2 plus a Prestel capability was the basis for the Oric, a name incidentally that <u>does</u> come from 'Micro'. Paul says that they spent hours trying to think up a name, much of it toying with anagrams of 'micro'. Someone came up with 'Oric', which sounded good. Then they realised they had left out the 'M', but it still sounded good, and the name was born!

At this stage they saw what Sinclair had done, and their financial backers, British Car Auctions, wanted higher volumes from the mass market to be the target. Thus was the Oric-1 born, although the first mock-up retained its executive image with a teak and apple-green colour scheme! Paul has a photograph to prove it.

The ULA was designed by Dr. Paul Johnson, the draughtsman for the design work being a chap called Roger, still believed to be living near Cambridge. The prototype was built from discrete logic chips, with many hours being spent trying to get it to go. It was produced for Oric by California Devices Inc., of Silicon Valley.

Tangerine had a Microsoft licence for the Microtan, but didn't bother to obtain extra permission to use Microsoft Basic for the Oric. It was all kept hush-hush during development, and when Microsoft got wind of what was happening, Oric were unable to release the details of the ROM as they might have intended. That is why we had all the refusals to detail the ROM at the time, and had to wait for enterprising authors like Leycester Whewell to tell us what was in there! As for the authors, Peter Halford ('Ratters') used to work in a TV shop in Northampton, and wrote the cassette routines, described by Paul justifiably as 'shoddy'. He also did Oric Mon, but Geoff Phillips had to spend weeks debugging it. Andy Brown and Chris Shaw did most of the ROM, they were professionals in every sense. Paul himself wrote the music routines, on a Microtan in Forth (they had an Oric emulator running on the Microtan). He hand-coded them into machine code, and passed them to Andy, who put them in the ROM. Originally he had 20 to 30 different sounds, and had to pick the best six. For some reason he didn't chose his favourite, a superb rendition of a fart!

The plastic case was designed by outside professionals. Apparently there were great rows about the mock speaker holes. The idea was to have holes, but the moulding machine couldn't cope, so dummies were the compromise. The advertising and publicity designs were done by Paul Sample, who did the illustrations for the Tom Sharpe books.

And that folks, is just a small insight into the heady days. One day, I promise, there will be a second edition of 'The Story...'

And so we go rambling 'on'...

'ON' (COMMAND)

Principal:

Having calculated the displacement, the GOSUB or GOTO code is saved and then the search commenced, jumping from comma to comma. If the displacement is too long, a return is made to the end of the instruction and an exit occurs without any action.

If it is nul, the first DEC #D4 makes it #FF, and it becomes a too lengthy displacement.

This method of simulating a simple GOTO or GOSUB by diverting directly to the interpreter is clever.

All line numbers are evaluated until the required one is found, which necessarily involves a loss of speed if you are waiting for the last number...

```
CA78  JSR $D80D   CAC2 JSR $D8C8   Evaluate displacement in #D4
CA7B  PHA         CAC5 PHA         save the code (GOSUB or GOTO)
CA7C  CMP #9B     CAC6 CMP #9B     GOSUB?
CA7E  BEQ CA84    CAC8 BEQ CACE    yes, jump
CA80  CMP #97     CACA             CMP #97     GOTO?
CA82  BNE CA75    CACC             BNE CABF    no, error
CA84  DEC D4      CACE             DEC D4      Take following number
CA86  BNE CA8C    CAD0 BNE CAD6    if not the correct one, continue
CA88  PLA         CAD2 PLA         recover the instrcution code
CA89  JMP $C900   CAD3 JMP $C917   and execute the command

CA8C  JSR $00E2   CAD6 JSR $00E2   jump ',' or GOTO/GOSUB (first time)
CA8F  JSR $CA98   CAD9 JSR $CAE2   get the line number
CA92  CMP #','    CADC             CMP #','    if it is followed by a comma then
CA94  BEQ CA84    CADE             BEQ CACE    continue the search
CA96  PLA         CAE0 PLA         readjust stck if displacement too great
CA97  RTS         CAE1 RTS         and exit
```

EVALUATE A LINE NUMBER IN #33–#34

Entry: C and A must be set as on the exit from a JSR #00E2 or #00E8: 0 if a number, 1 if not.

Exit: #33–#34 contains the value, an error is generated if the number is greater than 64000.
Y is untouched by this routine
TXTPTR points to the first non–numeric character, and the exit is made by a JSR #00E2 (A, N and Z correctly set).

Principal:

Each character is multiplied by 10 to ensure that one ends up with a whole number. The test for an excessive number is done using multiplication. #1A00=6400=64000/10. 6400 is practical because you only have to test the high byte to know if the limit is exceeded. 64000=#FA00 is also a 'whole' number, but it would have been necessary to add a LDA $34 to test the high byte after the multiplication by 10. The test is therfore done before multipilcation.
It is a fortunate chance that 64000 and 6400 are 'whole' numbers in hexadecimal as well as decimal...

```
CA98  LDX #00      CAE2  LDX #00      Set the result to 0...
CA9A  STX 33       CAE4  STX 33
CA9C  STX 34       CAE6  STX 34
CA9E  BCS CA97     CAE8  BCS CAE1     Exit if not numeric character
CAA0  SBC #2F      CAEA               SBC #2F    reduce to 0–9 (C=0 so subtract #30)
CAA2  STA 24       CAEC               STA 24     and save
CAA4  LDA 34       CAEE  LDA 34       take high byte of result
CAA6  STA 91       CAF0  STA 91       and save it
CAA8  CMP #19      CAF2  CMP #19      if result >= #1900=6400, too high a number so
CAAA  BCS CA80     CAF4  BCS CACA     'SYNTAX ERROR' (A cannot have value of #97)
CAAC  LDA 33       CAF6  LDA 33       Take low byte of result
CAAE  ASL A        CAF8  ASL A        x2
CAAF  ROL 91       CAF9  ROL 91       and high byte
CAB1  ASL A        CAFB  ASL A        x4
CAB2  ROL 91       CAFC  ROL 91       and high byte
CAB4  ADC 33       CAFE  ADC 33       +original: x5
CAB6  STA 33       CD00  STA 33
CAB8  LDA 91       CD02  LDA 91       recover high byte
CABA  ADC 34       CD04  ADC 34       add original and transfer
CABC  STA 34       CD06  STA 34       x5, so
CABE  ASL 33       CB08  ASL 33       x2x5=x10
CAC0  ROL 34       CB0A  ROL 34       not forgetting the high byte
CAC2  LDA 33       CB0C  LDA 33       add the newcomer
CAC4  ADC 24       CB0E  ADC 24       which takes the place of the units
CAC6  STA 33       CB10  STA 33
CAC8  BCC CACC     CB12  BCC CB16
CACA  INC 34       CB14  INC 34       still not forgetting the high byte
CACC  JSR $00E2    CB16  JSR $00E2    Take the next character
CACF  JMP $CA9E    CB19  JMP $CAE8    and recommence
```

No room for a tailender this month – see you at the Meet

Jon Haworth

Machine Code for the Oric Atmos (Part 17)   Peter N. Bragg

The Story so far

We have looked at the basic requirements for machine code programming on the Oric. Last time in Part 16 of the series, all that we have seen so far was summarised and put together, to provide a small, simple machine code programming kit. This consisted of an Instruction Subset, that is a list of just five essential instruction types, plus two Operating System calls that make it possible to read the keyboard and write to the display screen. Also included, was a reference columm to locate information on the various items, if required.

Back to the Drawing Board

A couple of issues ago, back in Part 15 of the series, we looked at the Operating System call "VDU" which can display codes put into Register "X". A short demonstration routine was shown which would load and display three codes one after another, pausing between each, to allow you to see what effect each one had.

As I pointed out at the time, the routine worked well enough, but could hardly be considered a good program. For those who do not have Part 15 of the series handy this moment, I will recap briefly. Essentially, each item was displayed using four seperate instructions, (1) to fetch the item (2) display it (3) read the keyboard and (4) wait for a key to be pressed. Instructions (3) and (4) created the pause for observation. These four instructions were repeated each and every time a new code was fetched for display. This meant that no less than twelve instructions were needed to display just three codes.

It should be obvious that if the routine was extended in the same way to display many more items, it would soon become enormous and clumsy and waste a lot of your time and Oric's memory.

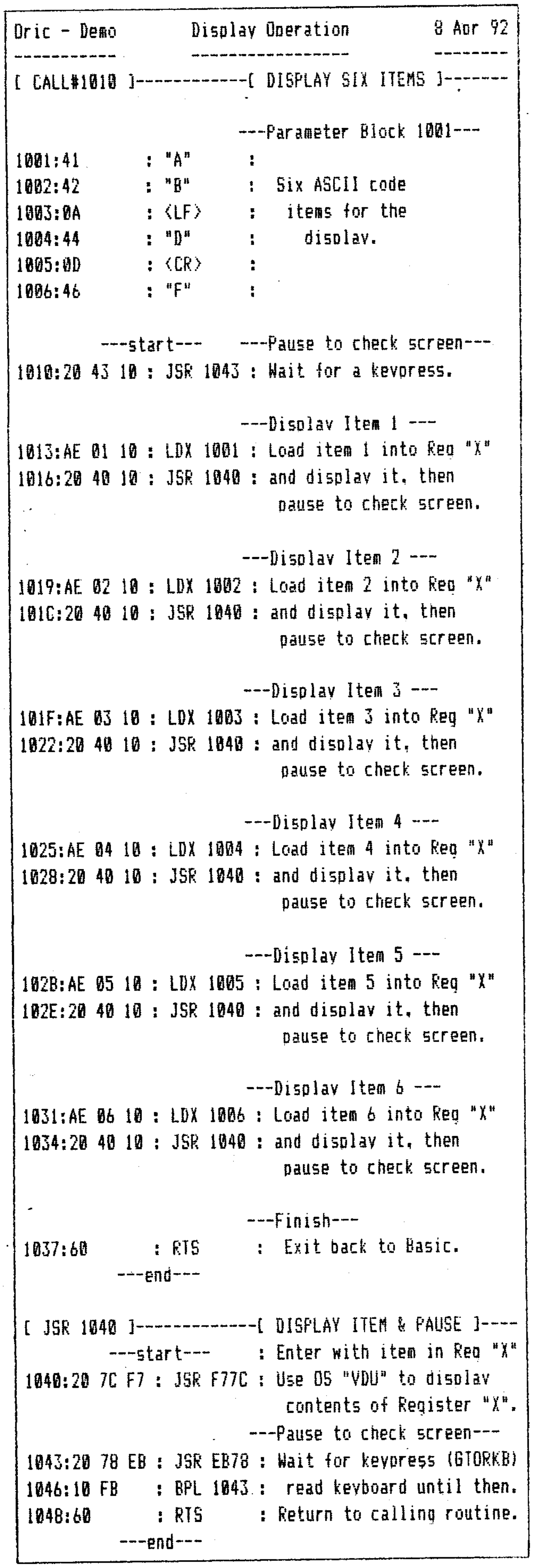
Now if you look at that original routine, you will see that only the "fetch" instruction changed, the remaining instructions (2), (3) and (4), which display the item and provide the pause, are exactly the same for each item displayed. It would be far better to make those three into a short subroutine, which could then be called up with a single instruction. After all, that is what we would we would do if we were writing Basic.

This precisely what has been done in the new listing presented here. You can enter it into the Oric using an assembler, or in hex code form, using the "Hexload" program from Part 7. Once again, to use it, just CALL#1010 and then tap the Space Bar a few times to display the contents of Parameter Block 1001 at the cursor. The display codes used in this sample, illustrate how the "Newline" operation is done, but of course, you can try other combinations in the Parameter Block, to see how the "VDU" call handles them.

Why make changes ?

The main point of this article was to show how programming can be made easier, in this case by using a subroutine. For a start you have less writing to do and there is a saving of memory space. More important, it should make the program easier to understand as well.

Of course this "Display Operation" is only a small demo program and the improvement is small too. Most of our demo programs need to be small so that they can be accompanied by an explanation of what actually is being demonstrated. Even so this version displays twice as many items as the original version, for only a small increase in size.



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

                         ---Parameter Block 1001---
1001:41        : "A"      :
1002:42        : "B"      :  Six ASCII code
1003:0A        : <LF>     :   items for the
1004:44        : "D"      :     display.
1005:0D        : <CR>     :
1006:46        : "F"      :

        ---start---    ---Pause to check screen---
1010:20 43 10 : JSR 1043 : Wait for a keypress.

                         ---Display Item 1 ---
1013:AE 01 10 : LDX 1001 : Load item 1 into Reg "X"
1016:20 40 10 : JSR 1040 : and display it, then
                            pause to check screen.

                         ---Display Item 2 ---
1019:AE 02 10 : LDX 1002 : Load item 2 into Reg "X"
101C:20 40 10 : JSR 1040 : and display it, then
                            pause to check screen.

                         ---Display Item 3 ---
101F:AE 03 10 : LDX 1003 : Load item 3 into Reg "X"
1022:20 40 10 : JSR 1040 : and display it, then
                            pause to check screen.

                         ---Display Item 4 ---
1025:AE 04 10 : LDX 1004 : Load item 4 into Reg "X"
1028:20 40 10 : JSR 1040 : and display it, then
                            pause to check screen.

                         ---Display Item 5 ---
102B:AE 05 10 : LDX 1005 : Load item 5 into Reg "X"
102E:20 40 10 : JSR 1040 : and display it, then
                            pause to check screen.

                         ---Display Item 6 ---
1031:AE 06 10 : LDX 1006 : Load item 6 into Reg "X"
1034:20 40 10 : JSR 1040 : and display it, then
                            pause to check screen.

                         ---Finish---
1037:60       : RTS      :  Exit back to Basic.
        ---end---

[ JSR 1040 ]-------------[ DISPLAY ITEM & PAUSE ]----
        ---start---      : Enter with item in Reg "X"
1040:20 7C F7 : JSR F77C : Use OS "VDU" to display
                            contents of Register "X".
                         ---Pause to check screen---
1043:20 78 EB : JSR EB78 : Wait for keypress (GTORKB)
1046:10 FB    : BPL 1043 :  read keyboard until then.
1048:60       : RTS      : Return to calling routine.
        ---end---
```

Using subroutines for repetitive operations in programs, will considerably reduce the program size, particularly when larger programs are being written. However, more useful to us at this moment, is how a subroutine like this can also be used to simplify the programming.

This new version of "Display Operation" now only needs two instructions for each code item displayed and furthermore, the essential description for each item has been cut down too. A good program description is essential, but it is best to keep it as small as possible. The "Display & Pause" operation is described in the subroutine and need not be repeated elsewhere. When the subroutine is called into use, we only need to note briefly how it is being used, which is easier. This of course applies to all subroutines, they only have to be described in detail once, any instruction that calls one, only needs a brief note of how that subroutine is being used, which makes the program easier to write and understand.

While this version of "Display Operation" is an improvement on the original shown in Part 15, I should of course point out that it is far from perfect and could still be improved upon. However, that is not the point of the excercise. The aim was to show that some improvements can be quite easy to make.

It is well worth looking at your program with a critical eye, with the aim of making some improvements, but dont expect to write super efficent programs all the time and if you are just starting out on machine code programming, dont even bother to try for perfection, it isn't worth the effort. The most important thing to concentrate on when writing machine code programs, is to make it work first. Cleverly written programs are useless if they dont work. Once you have a working program and have saved it, you can then try improving on that. Stick to making simple improvements like the one shown here, usually they are the most effective, anyway. If you are too clever, you will have a heck of a job when you want to update and improve your software later.

Next time will not be next time, it'll be the month after, if you see what I mean !!

THE ORIC SERIAL PORT Part 3

In part two I provided listings in BASIC which show how easy and simple it is to make use of the Oric serial port. I did not explain much about the interface and its registers but I intend to put this right now. However, the jargon may cause some problems for those of you who are not familiar with serial communications. I shall give explanations but by necessity these will be brief: you may need to do some further reading in order to get the most out of this article!

Okay, let's have a closer look at what the programming last time did to the serial port registers. The subroutine at line 200 of listing 2 sets up the interface. The settings given in that program configure the interface to operate at 300/300 baud with a word length of 8 bits and 1 stop bit (line 210: POKE £383,22). In addition, parity mode was disabled, RTS was set to low and interrupts were enabled (line 220: POKE £382,5).

Perhaps I better explain what some of those terms mean:

Baud is the label given to the measure of the rate of data transmission.

Word length is the length of each unit (data package) sent. For an ASCII character this need be no more than 7-bits long. For characters with codes of more than 128 (e.g. machine code files) then 8-bits will be needed. These will normally be the only word lengths you will need, however, the Oric interface can cope with word lengths of between 5 and 8 bits. 5-bit words are not often used except by radio amateurs for RTTI communications (which I know next to nothing about).

Stop bits are added to the end of each character sent. Now since a serial interface sends data sequentially each package of data (usually a character) has to be preceded by a start bit (for asynchronous communications) so that the receiving end can unscramble all the bits and correctly interpret the characters. Start bits are preset but the number of stop bits added is set by the user: you can have either 1 or 2 stop bits (actually you can also have 1½ stop bits if you are using a word length of 5-bits, but we shall ignore that!). So each package of data might look like this:-

//StartBit/Bit1/Bit2/Bit3/Bit4/Bit5/Bit6/Bit7/Bit8/StopBit//....

i.e. 1 start bit, 8-data bits, 1 stop bit. The total length in this case is 10 bits (hence a speed of 300 baud does not directly translate to a particular number of characters per second, since it depends upon word length and number of stop bits. In this example 300 baud would work out at about 30 characters per second. i.e. 300/10 if parity is disabled).

Parity can be set to odd, even or no parity. This is a simple means of automatically checking data integrity through the generation of a parity check bit. However, it is frequently not used (i.e. set to no parity/disabled) since software methods (e.g. splitting the data into discrete blocks and using checksums) have been found to be more effective.

The important thing to realise is that the system receiving the data has to be set up the same as the one transmitting (i.e. same baud rate, word length, number of stop bits and parity settings), otherwise all that will be received will be nonsense! Therefore anyone who dabbles in the murky world of serial ports and data transmission has to have some knowledge of baud rates, word length, stop bits and parity. Any communications software, including Oricomms, should allow you to easily alter these settings to achieve different serial port configurations. In practice, two configurations are more often used than others. These are 8-bit word length, No parity, 1 stop bit (i.e. 8N1) as used by many bulletin boards; and 7-bit word length, Even parity, and 1 stop bit (7E1) as used by viewdata services such as Prestel. The most frequently used baud rates are 300/300, 1200/75, 1200/1200, 2400/2400 and 4800/4800. Higher rates tend to be used for direct computer-to-computer data transfer only.

Okay, so what about the Oric serial interface internal registers. The 'heart' of the interface is the 6551 Asynchronous Communications Interface Adapter (ACIA). This is a very sophisticated chip designed specifically to be used with the 6502 microprocessor (the 'heart' - and brain! - of the Oric). The 6551 has four registers and, as stated last time, there are four memory locations in page 3 of the Oric's memory map which give you access to them. I shall describe each of these in turn:-

£380: THE TRANSMITTER & RECEIVER DATA REGISTER

This is where your transmitted data is POKEd to, or your received data is PEEKed from. Incoming data is overwritten/lost if it is not read frequently enough. This is why it is best to use interrupts to ensure reading this register is given priority, especially when operating at the faster baud rates.

£381: THE STATUS REGISTER

This allows you to find out about the status of the 6551 chip. If bit 7 is high this indicates that the 6551 has caused the processor interrupt. This bit is cleared when the status register is read.
Bit 6 can be ignored since DSR which it represents is not used in the Oric serial interface.
Bit 5, when low, indicates that a data carrier has been detected (DCD).
Bit 4 indicates whether or not the transmitter data register is full or empty (high = empty). This bit is cleared when a new character is written to the transmitter data register.
Similarly, bit 3 indicates the status of the receiver data register (high = full).
The final three bits give an indication of the integrity of the received data but do not produce an interrupt. Bit 0 (parity error), bit 1 (framing error) and bit 2 (overrun) would normally be checked when the receiver register is read (if they are checked at all). A 1 (bit high) in any of these indicates that there may be a problem with the character received. Bits 0-3 are cleared automatically when the receiver register is read.

£382: THE COMMAND REGISTER

This allows you to control specific modes and functions provided by the 6551 e.g whether or not parity checking is used.
Bits 7-5 are concerned with parity. Bit 5 controls whether or not parity checking will be used. Set bit 5 to 0 to disable parity and to 1 to enable it. It doesn't matter what bits 7 & 6 are set to if bit 5 is 0 (parity disabled). Otherwise, set bits 7 & 6 both to 0 for odd parity, or to 0 and 1 for even parity.
Bit 4 should normally be set to 0. If set to 1 it provides a simple way of testing some aspects of serial port software since then transmitted characters are echoed back, not transmitted. For this to work bits 2 & 3 must both be low.
Bits 3 & 2 control the RTS line and the transmitter interrupts. When both are set to 0 then the RTS line is low (i.e. indicates to connected modem etc that the computer is not ready to send data). Bit 3 should be set to 0 and bit 2 set to 1 for RTS to be high with transmitter interrupts enabled, or vice-versa for RTS high with transmitter interrupts disabled.
Bit 1 controls whether or not the 6551 will generate a processor interrupt. Set to 1 for interrupts enabled, 0 for interrupts disabled.
Bit 0 controls the DTR line of the serial port. In the Oric set-up this is used to turn the Oric Modem on-line. Set to 1 to turn on-line, 0 to drop the line.

£383: THE CONTROL REGISTER

This allows you to set the baud rate, the word length, and the number of stop bits.
Bit 7 controls the number of stop bits: when set low, 1 stop bit is generated; when set high, 2 stop bits are generated (but 1 stop bit for 8-bit word lengths with parity).
Bits 5 & 6 control the word length: both should be low for 8-bit words, bit 5 high & bit 6 low for 7-bit word lengths.
Bit 4 controls the source of the receiver clock (necessary for when the receive & transmit rate are different e.g. 1200/75). This is usually set to 1, but if the receiver clock rate is different to the transmit rate then this should be set to 0. Note however, that unless the Oric Modem is being used with the interface then a clock signal will need to be connected to the RxC pin of the interface (this is covered in the Oricomms manual).
Bits 0-3 allow the selection of different baud rates. Only the most common are given below:-

	Bit 3	2	1	0
75 baud	0	0	1	0
300 "	0	1	1	0
1200 "	1	0	0	0
2400 "	1	0	1	0
4800 "	1	1	0	0
9600 "	1	1	1	0

Bytes & Bits: For those of you who may be baffled by the idea of 'bits', I shall give a quick explanation. The Oric is an 8-bit micro. This means that each memory location can hold one byte (= 8 bits - binary) of data. Each bit can only be high (1) or low (0) but represent the different decimal values shown below. These can be tested separately by PEEKing the memory location with the AND logical operator and an appropriate number according to which bit(s) are being tested. 8-bits can represent decimal numbers between 0 and 255. Each bit when high will represent the following values, or 0 when low:-

Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0	
128	64	32	16	8	4	2	1	Total = 255

For example, in listing 2 given in part two, lines 120 and 130 use 8 and 16 respectively to AND with PEEK(£381). This tests bit 3 and bit 4 respectively of the status register.

Using the information provided above together with the BASIC programs in part 2, it should be possible for you to work out how to expand/change the subroutine beginning at line 200 to alter the interface configuration, and perhaps to develop other features in the transmit/receive routine. If anyone is really keen and would like more information on the 6551 chip registers, I can supply it (SAE please).

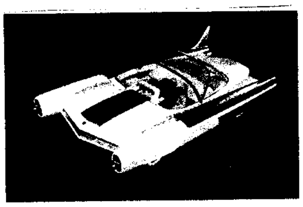
Next time I shall discuss transferring files between the Oric and another computer, and will suggest a structure for a file transfer program.

Trevor Shaw.

FX - ATMOS 1954

FX-ATMOS 1954

Patriotically finished in a colour scheme of red, white and blue, the FX-Atmos was the star of Ford's stand at the 1954 Chicago Automobile Show and featured ideas "so advanced that they had formerly been seen only in Ford styling studios". The FX-Atmos, boasted Ford Division chief Lewis D. Crusoe, represented "one of many avenues which styling could take in the future". Strange devices resembling giant hypodermic needles projected from the front of the nacelle-like front wings, a curious bubble canopy protected the passengers from the elements and the driver sat centrally, guiding the car with tiny hand controls. The power source, again, was conveniently unspecified...



FX-ATMOS 1954

For a full
18" x 22"
Colour Photo:

a)Nip to a Ford dealer

b)STEAL their Calendar

And

c)Get JON.H. to defend you in Court !!!

THE FUNCTION KEY AND SEDORIC DOS - Dave Dick

LIKE MANY MANUALS, THE SEDORIC MANUAL,THOUGH VERY WELL WRITTEN,CANNOT FOR OBVIOUS REASONS DETAIL EVERYTHING IN FINE DETAIL. OVER THE NEXT FEW MONTHS I HOPE TO PASS ON TO NEW AND OLD USERS ALIKE THE TIPS THAT I HAVE PICKED UP.

IF YOU HAVE ANYTHING OF RELEVANCE;THEN PLEASE SEND IT IN.

WE START WITH THE USE OF THE 'FUNCTION' KEY. MANY MOONS AGO I GAVE A LIST OF SOME OF THE COMMANDS AND ROUTINES THAT ARE PRESET IN YOUR DOS. IT IS SO MUCH EASIER TO DO A 'FUNCT] ' INSTEAD OF TYPING IN !DIR EVERYTIME. WHEN ONE IS TYPING IN A PROGRAM,IT IS FAR EASIER TO HIT 'FUNCT J' RATHER THAN TYPE IN'CURSET'.

I WILL FIRSTLY LIST THE USES OF THE FUNCTION KEY FOLLOWED BY A KEY, AND THEN GO ONTO THE USE OF 'FUNCTION' + 'SHIFT' + A KEY.

FUNCT +

1 or ; or , > or < or : or " or $ or % or ^ or (or £ = ?HEX$ (DEEK(#
...

(note that in some instances the SHIFT key will need to be used e.g. to get the % sign,then you have of course to use SHIFT 5)

2 = PAPER 0:INK 7
3 = DOKE #24E,#108 ..... 4 = DOKE #245,#EE22
5 = DOKE #245,#484 .... 6 = DOKE #23C,#EB78
7 or Z or #= CALL #F8D0 .... 8 = RENUM 1000
9 = RENUM 1000,,1000 .... 0 = NUM END

OKAY! A quick look along the top row of keys to see what else we can use. How about: FUNCT @ (i.e SHIFT and 2) for RESET, FUNCT ! (SHIFT and \) for LIST 1000- . Then there is FUNCT) (SHIFT 0) for SAVE". We drop down a row for FUNCT for SAVEU" and down a further row for FUNCT ' which gives EXT. Right,that's the first 16. I am not implying that you are all thick by stating that you have to use SHIFT 5 to get %. It's just that I don't know which of the characters will print out and this way you will get a clue. I can't see how to print the - you know,those squiggly brackets. If I hold down SHIFT and press one of them,nothing happens,but if I press them twice then an odd character appears on screen!

Anyway, I'm just going to make a cuppa tea and then back to this.

Now the most commonly used,which is FINCT and right square bracket. This gives !DIR and of course hits the RETURN key for you.

Now some more : -

FUNCT +

- or ESC or + or left squiggly bracket = OLD ..... = (equals) = RUN
\ = LIST ..... Q = SOUND ... W = PATTERN ... E = ELSE .. R = REPEAT
T = THEN .... Y = DRAW ... U = UNTIL .. I = IF ... O = ON ... P = PEEK
A = AND ... S = STR$ ... D = DEEK ... F = FOR ... G = GOTO ... H = HEX$
J = CURSET ... K = KEY$ ... L = LEFT$ ... X = CLOAD .... C = CLS... V = VAL
B = PAPER ... N = NEXT ... M = MID$ ... / = TRON ... Left Cursor = CALL
Down Cursor = NEW ... SPACE BAR = CTRL A 6 Times (i.e. COPY 6 CHARACTERS)
Up Cursor = EDIT ... Right Cursor = LIST ... & = HIRES ... * = TEXT

By now you should of started to remember a few. Especially G for goto and E for ELSE etc.

And so to the last of them,utilising the same keys as above,but bringing the SHIFT key into use.

FUNCT + SHIFT +

Q = PLAY ... W = CHAR ... E = END ... R = RETURN ... T = TO ... Y = CIRCLE
U = TEXT ... I = RIGHT$ ... O = SAVE" ... P = TROFF (also obtained with ?)
A = ASC ... S = SCRN ... D = DOKE ... F = FILL ... G = GOSUB ... H = HIMEM
J = CURMOV ... K = CLEAR ... L = RIGHT$ ... X = CSAVE ... C = CHR$
V = LEN ... B = INK ... N = NOT ... M = MUSIC ... Up Cursor = CONT
Right Cursor = AUTO ... Z = DATA
Left Cursor,Down Cursor OR SPACE BAR WILL ALL GIVE ?HEX$(DEEK(#, same as FUNCT + 1

WELL I TRUST YOU CASSETTE USERS WERE SUITABLY BORED BY THAT. THERE WAS IN FACT A F.G.C. UTILITY FOR TAPE USERS WHICH DID SIMILAR THINGS.

IF ANY DISC USERS DESIGN A THINGY TO GO OVER THE KEYS FOR USE WITH THE FUNCTION KEY,THEN I WOULD BE PLEASED TO RECIEVE ONE. THEN MY ATMOS KEYBOARD COULD RESEMBLE A 'SPECCY'.

===

TITLE: ZEBULON   AUTHOR: JONATHAN BRISTOW
AVAILABILITY: ON 3" DISC FROM JULY 18,& ON CASSETTE SOME TIME IN AUGUST.
PRICE: TO BE AGREED.

This new arcade/strategy came virtually out of the blue. And what a pleasant surpise it was. It is loosely based on the likes of 'Secret du Tombeau','Boulderdash'etc.There are 20 levels to complete.On each level there are a set number of doors and switches. Each switch opens or shuts a number of doors on that level. Keys can be used to open any door, but once used the key is forfeited.Several switches may operate the same door. On some of the trickier levels ,you will find a bank of switches .Your task is then to find the correct combination to allow you to proceed. Watch out for the moving walls. Matthew insists that on one of the levels,the cursor keys do not send you in the direction you expect. Other items to look out for are a THUNDERBOLT, which you will need to slay the ZORKS. Food has to be taken ,or else you die.To actually complete the game,you need to pick up the 36 puzzle pieces,which are spread about the 20 levels.These puzzle pieces make up a picture.

The lay out on the screen is split between the following:The game area,the hint window ,the registers displaying energy etc,and the puzzle picture.

The game makes full use of the ORIC graphics and sound.Machine code routines were written by ALISTAIR WAY.

For two days solid either MATTHEW,LOUISE or ANN have been trying to complete this brainteaser. Therefore as penance,I have insisted that ANN types up this review for me.

Definitely a winner is this one.Don't you agree MATTHEW!.

"Here mum and dad,hurry up and get off that machine so that I can get that last puzzle piece"- shouts MATTHEW.

"SHUT UP AND GO TO BED its nearly 11 o'clock" -shouts dad.

Oh! I nearly forgot. To get from one level to the next you need to land on the OPIUM. Now we know what Jonathon is taking.

- DAVE DICK (with typing and wisecracks from the wife.)

++

PREVIEW

Title : PIXED 92   Author : Jonathan Bristow. Availability : Coming soon to an Oric near you.

In just under 3K of memory,this utility allows you to edit any of the standard or alternate character sets with the flexability of being of being able to design the screen at the same time.

PIXED presents you with a single screen with which you can place your characters upon.Should you wish to trim (Edit) a character during plot mode,simply press 'E' and you will enter the pixel editor mode. The screen doesn't change ,but instead of moving a character over the screen,you have control of a pixel within the last character position. You may now simply move over that character or to a character next to it and press Space to plot/unplot the pixel.

There are numerous options in and when I can get the thing to run;then I will report further. But,how does he know what he has written to date is correct,Ihear you mutter!.Well, the words are straight out of the manual and a well presented manual it is! It states that at the start that the program will ask for a filename and that this filename will be used for both saving and loading files. Perhaps Jonathan would like to send me the 'cheat' version i.e the version that will allow me to type in a file name and get the bloody thing to run. Sometimes I DESPAIR!

Hang on a minute.Even though the screen appears to of hung up,perhaps it hasn't. It says in the manual that 'FUNCT' + 'C' will clear the screen. Oh! so it does. I think this could be a long job.

- Dave Dick

===

PREVIEW

Title: MASTERMIND   Author : Jonathan Bristow (with acknowledgement to the original idea).   Availability: Coming soon

This is almost ready for release.

To describe it as colourful would be an understatement. I thought for a minute that i was playing one of those French pinball games. Even the music is more akin to MACADAM BUMPER than a serious thinking man's/woman's game that MASTERMIND was originally marketed as. Jonathan has put life into this game. I won't bore you with the rules as they are the usual for this peg-board routine.

- Dave Dick

==

'SAM COUPE'

Jonathan Bristow has a SAM COUPE - no it's not a baby buggy,it's a Miles Gordon computer. It is 512K and comes with a 3.5" internal drive and a few games. Jonathan wants to swap it for a Spectrum 128 (the one with the same sound chip as the Oric). You can write to Jonathan direct or leave a message for him to ring you by telephoning his parents on 0953 881212

==

CONTACT LIST

You will notice that I am not printing names/addresses. To find them,just check out the contact list - less work for me- more work for you! - MORE SPACE IN OUM.

RAFFLE TICKETS

Raffle tickets for the OUM GRAND DRAW still available. See last issue for details. Additional prizes include some special Leicester beer donated by Graeme Burton - HIC!

+++

MESSAGE TO STEVEN FRIEND

Frank Bolton has a copy of the Gary Marshall book that you are after. Just send him a few stamps to cover postage. Alternately buy him a raffle ticket at the ORIC MEET.

++

ACCOMODATION

For those wishing to crash out in Aylesbury on the Friday or Saturday of Oric Meet; I have found a cheapo Bed and Breakfast. It is 'Homely', 127 Tring Rd, Aylesbury. Tel: 0296 20828. Price is 10 pound per person per night. Those who can't afford THAT can bring sleeping bags and will be given the key to my garage!

==

ROYAL MAIL

The postal system certainly excelled with the last issue of OUM. All magazines were sent out second class on a Monday afternoon. By late Tuesday morning, Ken Austin was on the phone asking for the second hand software packs advertised. During the day another 3 of you rang.

I will try and put together some more cheapo packs.

The Opelco twin drive system h been sold. Offers are being taken for the MICRODISC. Put in a bid to exclude the Atmos and speech thingy. Start around 50 pounds and let me know your limit. We will auction it at the meet.

==

PRIZE WINNER

NEW SUBSCRIBER COLIN COOK HAS A SURPRISE GIFT ON IT'S WAY TO HIM FOR INTRODUCING ARTHUR CRAWFORD TO THE GROUP.

WELCOME ABOARD GENTLEMEN.

KANGAROOS

JUDY SIMMS will knit anyone a jumper with the ORIC logo on it. For price details please contact Judy direct.

ORIC Enthusiasts (OUM 59)

This month sees a temporary departure away from the usual format in that I have decided to present a complete article on Geoff Phillips' book in order to complete the program contained in it. I will continue with other articles next time round.

ORIC ATMOS and ORIC-1 GRAPHICS & MACHINE CODE TECHNIQUES

Chapter 3 - BASIC (continued)
copyright of Geoff Phillips

3.10 Machine code advice (cont'd)

INCREMENT AND DECREMENT Important points cont'd:

3 When using INC or DEC with several bytes, remember that you can safely do one set of INC or DEC instructions at a time. The following example employs such faulty logic:

```
      INC 42
      INC 42
      BNE A
      INC 43
   A  NOP
```

RETURN FROM INTERRUPT Remember to use RTI to finish an interrupt routine. The only difference between RTI and RTS is that with RTI the 6502 saves the processor flag on the stack. This means that an interrupt routine need not save the processor status register.

SUBROUTINES When the jump to subroutine instruction is executed, the return address is saved on the stack. This address is saved high byte followed by the low byte (this follows the 6502 convention of a low address being stored in the lower location). This return address on the stack is always one less than the real return address - the 6502 adds one to the program pointer before executing each instruction.

SEI AND CLI On the ORIC and interrupt can occur at any time. If you want to disable interrupts (which will stop the keyboard from being scanned and the cursor flashing) you can use the SEI instruction. CLI (clear interrupt disable) enables the interrupts again. Note that SEI does not stop the 6522 clocks from running, but it does prevent interrupts from being generated when the clocks reach zero. SEI should be used when your program is using the stack area in a non-standard way.

3.11 Using the ! extension command

The ! command allows you to create your own BASIC command. When BASIC encounters the ! token it jumps to the address stored at #2F5,6, assuming it to be a normal subroutine.

PASSING DATA PEEK and POKE provide one way to send data between your extension subroutine and BASIC, but a better way is to put the data after the ! command, as you would do for any other BASIC command. The pointer #E9,#EA will be identifying the byte following the ! command as you enter your subroutine. You can (and must) use this pointer to extract all the data pertaining to the command. When you exit from the subroutine #E9,#EA must be pointing to the byte following the last byte in your command. In order to look at each character, you can call subroutines at #E2 (which increments #E9,#EA) or #E8 (which does not increment #E9,#EA). After each call the next character is passed in the accumulator. This can be used to pass over delimiters, such as commas.

USING THE FORMULA EVALUATION ROUTINE If you want the extension command to work with expressions (such as X + Y) as well as fixed-format data, you may need to call the ROM subroutine which evaluates the expression. This subroutine (at #CF17 for V1.1 ROMs or #CE8B for V1.0 ROMs) only needs the #E9,#EA pointer to be set up. At the end of the subroutine the #E9,#EA pointer will be correctly set to the character following your expression. Note that the expression evaluated can contain the normal BASIC functions, e.g. !X*SQR(Y), but be warned that the subroutine assumes that all words have been compacted into tokens - including such things as the +,-,* and / operators. As in BASIC, expressions must be terminated with a comma, colon or #00 (i.e. the end of a BASIC line). There are two possible types of answer returned:

1 A string of characters. The information about this string is stored in an area of memory pointed to by the address #D3, #D4. In this temporary area there are three bytes: length (one byte) and address of string (two bytes). When the formula results in a string, location #28 is set to #FF. Once you have finished with the string, you must release the temporary area it used by calling #D7CD (V1.1) or #D712 (V1.0).

2 A floating-point number. This number is stored in the floating-point accumulator (see Chapter 6). Location #28 is set to zero to indicate a numeric result. If you want to convert the number into a signed 2-byte integer, you can simply call #D92C (V1.1) or #D871 (V1.0). This will return Y as the low byte and A as the high byte. For an example of using ! see Chapter 4.

3.12 Using the & extension function routine

Whereas ! can only be passed data, the & function not only expects data to be passed but also returns a value. The & facility assumes that #2FC,#2FD points to the machine code routine.

PASSING DATA There are two types of data that can be passed - a string of characters or a number. In both cases, & must have an argument following, surrounded by parenthesis. For example, &(A$), &(4.3+S). The formula evaluation takes place automatically on the argument, and the results are exactly the same as described in section 3.11. When a number is passed, you can either take it or leave it, but a string requires extra action. If your subroutine has been passed a string, you must call subroutine #D8AC (V1.1) or #D7F1 (V1.0) in order to free up the temporary string space. This will also extract the necessary information, storing the length in the accumulator and the address of the string in #91,#92.

RETURNING DATA Returning data will usually be the final thing that the subroutine does. Location #28 should be set to zero if you are returning a number, or #FF if the result is a string. To return a number you simply leave that number in the floating-point accumulator at #D0 to #D5 - see Chapter 6. Returning a string is a little more complicated, since you must first allocate an area for it. This is done by putting the length (in bytes) into the accumulator and calling #D5AB (V1.1) or #D4F0 (V1.0). This will leave the address of the new string at #D1,#D2. Once you have put the string at this address, you must finish the subroutine with either of these:

```
        PLA                   PLA
        PLA                   PLA
        JMP   #D5F4 (V1.1)    JMP   #D539 (V1.0)
```

When returning a floating-point number, you exit with the usual RTS instruction.

EXAMPLE: THE INSTR FUNCTION On some computers you will find the 'INSTR' function. This searches for a string of characters within another string, returning its position, if found. For example, INSTR("ABCD","BC",1) is 2 (the last parameter, 1, indicates the start position of the search). The subroutine of Program 3.2 simulates the INSTR function. The function is called by a statement such as: A=&("T$,S$,N"). String S$ is searched for within string T$, starting at position N. The quotes are used since & can only take one parameter; this means that you can only use simple variables (such as A$) in the actual statement.

The listing will work unchanged for V1.1 owners, but users of V1.0 ROMs should make the following adjustments:

```
9800:  JSR $D7F1
981D:  JSR $CE8B
982D:  JSR $CE8B
983D:  JSR $CE8B
9840:  JSR $D871
987B:  JSR $D3ED
```

To use INSTR, you must first type DOKE#2FC,#9800.

```
9800:  20 AC D8  JSR $D7F1     981D:  20 17 CF  JSR $CF17
9803:  A0 09     LDY #$09      9820:  A0 02     LDY #$02
9805:  B9 33 00  LDA $0033,Y   9822:  B1 D3     LDA ($D3),Y
9808:  48        PHA           9824:  99 35 00  STA $0035,Y
9809:  88        DEY           9827:  88        DEY
980A:  10 F9     BPL $9805     9828:  10 F8     BPL $9822
980C:  A5 E9     LDA $E9       982A:  20 E2 00  JSR $00E2
980E:  48        PHA           982D:  20 17 CF  JSR $CF17
980F:  A5 EA     LDA $EA       9830:  A0 02     LDY #$02
9811:  48        PHA           9832:  B1 D3     LDA ($D3),Y
9812:  A0 01     LDY #$01      9834:  99 38 00  STA $0038,Y
9814:  B1 D3     LDA ($D3),Y   9837:  88        DEY
9816:  85 E9     STA $E9       9838:  10 F8     BPL $9832
9818:  C8        INY           983A:  20 E2 00  JSR $00E2
9819:  B1 D3     LDA ($D3),Y   983D:  20 17 CF  JSR $CF17
981B:  85 EA     STA $EA       9840:  20 2C D9  JSR $D92C
```

```
9843:  38        SEC          986A:  C5 38     CMP $38
9844:  A5 35     LDA $35      986C:  D0 EC     BNE $985A
9846:  E5 38     SBC $38      986E:  A4 33     LDY $33
9848:  86 35     STA $35      9870:  C8        INY
984A:  E6 35     INC $35      9871:  D0 06     BNE $9879
984C:  C6 33     DEC $33      9873:  E6 33     INC $33
984E:  A5 33     LDA $33      9875:  D0 D7     BNE $984E
9850:  85 3B     STA $3B      9877:  A0 00     LDY #$00
9852:  C5 35     CMP $35      9879:  A9 00     LDA #$00
9854:  B0 21     BCS $9877    987B:  20 99 D4  JSR $D499
9856:  A9 00     LDA #$00     987E:  68        PLA
9858:  85 3C     STA $3C      987F:  85 EA     STA $EA
985A:  A4 3B     LDY $3B      9881:  68        PLA
985C:  B1 36     LDA ($36),Y  9882:  85 E9     STA $E9
985E:  A4 3C     LDY $3C      9884:  A0 09     LDY #$09
9860:  D1 39     CMP ($39),Y  9886:  68        PLA
9862:  D0 0F     BNE $9873    9887:  99 33 00  STA $0033,Y
9864:  E6 3B     INC $3B      988A:  88        DEY
9866:  E6 3C     INC $3C      988B:  10 F9     BPL $9886
9868:  A5 3C     LDA $3C      988D:  60        RTS
```

Program 3.2 INSTR

3.13 A real-time clock Program 3.3 is a short program to give your programs a clock that can return the current time of day. The listing will work unchanged for V1.1 owners, but users of V1.0 ROMs should make the following adjustments:

```
                                   46B:   STA $230,X
```

To start the clock, CALL#45C.

```
0410:  48        PHA          0443:  69 01     ADC #$01
0411:  18        CLC          0445:  8D 07 02  STA $02C7
0412:  F8        SED          0448:  C9 18     CMP #$18
0413:  AD C4 02  LDA $02C4    044A:  D0 0E     BNE $045A
0416:  69 01     ADC #$01     044C:  A9 00     LDA #$00
0418:  8D C4 02  STA $02C4    044E:  8D C7 02  STA $02C7
041B:  D8        CLD          0451:  18        CLC
041C:  AD C5 02  LDA $02C5    0452:  AD C8 02  LDA $02C8
041F:  69 00     ADC #$00     0455:  69 01     ADC #$01
0421:  8D C5 02  STA $02C5    0457:  8D C8 02  STA $02C8
0424:  C9 3C     CMP #$3C     045A:  68        PLA
0426:  D0 32     BNE $045A    045B:  40        RTI
0428:  A9 00     LDA #$00     045C:  A2 04     LDX #$04
042A:  8D C5 02  STA $02C5    045E:  A9 00     LDA #$00
042D:  18        CLC          0460:  9D C4 02  STA $02C4,X
042E:  AD C6 02  LDA $02C6    0463:  CA        DEX
0431:  69 01     ADC #$01     0464:  10 FA     BPL $0460
0433:  8D C6 02  STA $02C6    0466:  A2 02     LDX #$02
0436:  C9 3C     CMP #$3C     0468:  BD 72 04  LDA $0472,X
0438:  D0 20     BNE $045A    046B:  9D 4A 02  STA $24A,X
043A:  A9 00     LDA #$00     046E:  CA        DEX
043C:  8D C6 02  STA $02C6    046F:  10 F7     BPL $0468
043F:  18        CLC          0471:  60        RTS
0440:  AD C7 02  LDA $02C7    0472:  4C 10 04  JMP $0410
```

Program 3.3 Clock

The time can be set up (and read back) using PEEK and POKE from the following locations:

```
                       #2C5   seconds
```

#2C6	minutes
#2C7	hours
#2C8	days

Location #2C4 is used to store one-hundredth second intervals - but this is not in a suitable form for reading.

ACCURACY The clock will stay fairly accurate, except when certain commands are used. The most serious problems will arise when doing any tape saving or loading. A minor loss of time can happen during any sound command and when scrolling occurs, on V1.0 machines.

Next time, the end of Chapter 3 with the Relocator program.

SOFTINDEX

TWO FROM DAVID GOODRUM ON Public Domain

```
50
A

SOFTWARE OF DAVE DICK'S
-----------------------
Status: C = Cassette; I = Instructions J = Joystick OR/PRO/ALT

No:   Title:                          Supplier:   Disk No & Status:
.........................................................................
 25   ANTICS                                      BE1  OR/AT C      GAME
 40   ATTACK OF THE CYBERMEN          IJK         AH1  OR/AT C J    GAME
 66   ACHERONS RAGE                   SOFTEK      CB1  OR/AT C      GAME
 69   ATTACK OF THE CYBERMEN          IJK         CB1  OR/AT C J    GAME
139   A VIEW TO A KILL                DOMARK      CJ1  OR/AT C    I ADVENTURE
140   A VIEW TO A KILL                DOMARK      CJ1  OR/AT C    I ADVENTURE
172   ANTICRACK                                   AF2  OR/AT        UTILITY
180   AFRICA                                      BA2  OR/AT        ADVENTURE
223   ALTAI JOYSTICK ROUTINE          LISTING     CM2  ATMOS C J I UTILITY
234   ATLANTID                        SPRITES     CR2  OR/AT        GAME
253   ACCOUNTS/CALC                   SOFTBACKS   CW1  ATMOS      I BUSINESS
274   ARRAYSORT                       LISTING     DA1  OR/AT        UTILITY
281   AWARI                           LISTING     DA1  OR/AT C      STRAT/SIM
```

MUSIC INDEX

```
DAVE GOODRUM'S MUSIC INDEX
--------------------------
F indicates format - numbers correspond to:
 1 -CA/S7   2 -S12UK   3 -S12IM   4 -S10   5 --   6 -ALBUM   7 -LIVE
 8 -F/CLB   9 -OTHER

No:  Track                          Artist             Album Title
......................................................................
 11  A LOVER SPURNED:               SOFTCELL/MARC AL   MEMORABILIA
 28  A MAN COULD GET LOST:          SOFT CELL          NON STOP ECSTATIC
 33  ANGELS:                        MARC AND THE MAM   UNTITLED
 49  ART OF FALLING APART:          SOFT CELL          ART OF FALLING AP
 66  A MILLION MANIAS:              MARC AND THE MAM   TORMENT AND TORER
 94  ALWAYS:                        MARC/WILLING SIN   STORIES OF JOHNNY
120  ANGEL IN HER KISS:             MARC/WILLING SIN   MOTHER FIST AND H
134  A WOMANS STORY:                MARC/WILLING SIN   SINGLES 1984-1987
155  ALONE:                         MARC ALMOND        JACQUES
168  A LOVER SPURNED:               MARC ALMOND        ENCHANTED
196  A WOMANS STORY:                MARC/WILLING SIN   $TGLOW212
198  A SALTY DOG:                   MARC/WILLING SIN   $TGLOW212
```

```
>>> Record number  235

>THE DESPERATE HOURS
-ORCHESTRAL VERSION
Artist >MARC ALMOND
Album Title >$12RPD6252
Format >S12UK
```

READERS LETTERS

DEAR DAVE,

I thank you for sending my software order. I'm just beginning to get into some of the programs. TETRIX looks particularly good.

'ORISCOPE' caught me out,as I was expecting something to do with celestial movements rather than astrology,but never mind,as I cannot get it to run anyway without working out what "DDHMM" means for latitude. What I am looking for is a program which shows the night sky from any point on the Earth's surface at a chosen time,complete with planetary positions. Perhaps I could have a go at wtiting one myself,but it would involve a lot of time and the purchase of Star tables etc.

People like me,who like to write programs,prefer to do it when the results can be shared,discussed,improved. The User group holds us all together,but as a born worrier,I would feel more comfortable if there were some kind of network of users with modems,to share interesting computing work,news etc. when you felt like doing the same,or even taking it easy;instead of having to get a magazine out. (NOTE FROM THE EDITOR:- I think Colin feels that this would benefit me in as much as articles etc. could be pushed down the wires).

Do you know if there are any users with Modems,and if modems can be used while a disc system remains connected. (Is a dual-interface available ?). If there is a network going then I will buy and join!

- COLIN COOK (Pitsea)

DEAR COLIN,

like many titles, ORISCOPE came without any instructions. For you and others who have written,here courtesy of Chris Hearn from Tring is what to do. --

LATITUDE - DD (enter a number in the range 00 to 90), H (enter N for North or S for South), MM (enter the minutes in the range 00 to 60).

LONGITUDE - DDD (enter in the range 000 to 180), L (enter E for East or W for West), MM (enter the minutes in the range 00 to 60).

As an example for those born in Aylesbury,they would enter for LATITUDE: 51 N 48 and for LONGITUDE: 000 W 48.

The latitude/longitude information is printed on the side edge of an ORDNANCE SURVEY MAP.

As for a program on night skies, I am informed by Rob Kimberley that the Oric could not cope with this. There is probably a program out for the BBC and most definitely one for the PC.

Regarding modems, see the new Bulletin Board news in this issue. To find out who has a Modem,just check out the contact list. I know there are quite a few and probably even more who didn't bother to fill in our old questionnare. There are certainly modem users around who do not subscribe to OUM. Trevor Shaw is arguably the most knowledgable on the subject,whilst David Goodrum has put it to great use. One thing to watch for of course is the telephone bill.

According to Trevor a modem can be connected whilst the drive is in situ. He has rigged himself up a mother board. By far the cheapest method is to add a 34 way IDC connector to the one between deive and Atmos. It should be connected as near as possible to the Atmos. This works with the ORIC MICRODISC and I assume would work with your OPELCO, though some of the leads supplied with the Opelco were a little short to say the least. The Opelco will not work with the long lead as supplied with the Microdisc,but does work with the one's that are about 8" long that Steve Hopps supplies now.

Well Colin, I think between us we have turned a letter into an article.

- DAVE

===

DEAR DAVE,

I've been meaning to write with this query for some time. I have an Atmos and a CITIZEN 120d printer and use AUTHOR from Tansoft. My problem is that I cannot find out how to send different printer styles from the program to the printer (e.g. Italics). I have to set the printer up first before I print. I believe someone else wrote to you about the Citizen,but I cannot find the relevant issu of OUM.

Also do you know of any other word processors that can do the above plus underlining tabs etc and are available on cassette format?

- KEN DUDDLE (Leicester).

DEAR KEN,

you may be wondering why I am printing the letter/reply when I have already sent you a reply. The reason is threefold: a) the information may be of some help to others, b) I may have found out something else relvant since I wrote , and c) there is probably someone out there more knowledgeable than myself on the subject.

Anyway, a list of Citizen owners can be found by looking through the contact list. One who I am putting you in contact with is Tim Phoenix,who uses it with AUTHOR. I would assume that to get the printer to do italics from within a program, then the CTRL codes would have to be used as defined in the Citizen manual. David Wilkin is our resident expert on AUTHOR.

Other cassette based word processors currently available are SCRIBE from Public Domain and WORDWORTH direct from me. I am told that WORDWORTH is not 100% Epson compatible.

I have recently been sent a lot of information on the SEIKOSHA SP1900,but as yet have nothing on the CITIZEN. Perhaps someone out there could pass on any tips. In closing, I notice from your letter that you have not yet mastered the art of right justification. This information is on it's way.

- DAVE

===

CONTACT LIST UPDATES

WELCOME BACK TO THE GROUP TO GEORGE POPLE. GEORGE IS AT FLAT 11, LEONARD HOULDEN COURT,off DORCHESTER ROAD,TAUNTON TA2 7LN....Tel: 0823 336897

HIS INTERESTS INCLUDE 'RTTY'

DENIS HUGHES CAN BE REACHED ON 0745 886271

WELCOME TO NEW READER BRIAN ODURNY OF 'THORNHILL',MAIN ROAD,HUNTLEY . GLOUCS. GL19 3EA ... OUM 194.... ATMOS/CASS. /PARALLEL TO RS232 I/FACE....INTERESTED IN FORTH,CALC,AUTHOR AND THE HARDWARE SIDE.

ON THE MOVE

FRANK BOLTON WILL SOON BE ON THE MOVE. MOVING DATE UNKNOWN,BUT DESTINATION IS - IT WILL BE: 204 ELIZABETH HOUSE, 2 WATERLOO WAY, LEICESTER. LE2 0QE

FRANK STARTED HIS INTEREST IN COMPUTERS AT THE TENDER AGE OF 61. WITH ONLY THE ATMOS MANUAL AND THE TRUSTY ATMOS. FRANK NOW USES THE MACHINE MAINLY FOR TEACHING 'ENGLISH' TO SPANIARDS.

HAVE A GOOD MOVE FRANK AND KEEP UP THE GOOD WORK.

FOUR FROM COLIN

RECENTLY RECIEVED FROM COLIN COOK ARE 4 TITLES FOR EVALUATION. THEY ARE ANOTHER RENDITION OF 'MASTERMIND', A FRACTALS PROGRAM, A PUZZLE GAME, AND A COLOUR HIRES DUMP FOR THE PLOTTER. CHRIS HEARN IS TESTING OUT THE LATTER. MORE AS AND WHEN.

LETTER FROM DENIS

AS WE GO TO PRINT I HAVE RECIEVED A LETTER FROM DENIS BONFIELD AND THOUGHT THAT SOME OF THE POINTS MAY BE OF INTEREST TO SERIOUS USERS AND GAMESTERS ALIKE. FIRSTLY THE MAIN POINTS FROM THE LETTER AND THEN THE EDITOR REPLIES.

THE LETTER

a) HAVE INVESTED IN A SECOND-HAND 'AMSTRAD DMP3160 PRINTER AND MY ATMOS IS NOW SWITCHABLE ROM. IT SWITCHES VIA THE 'RESET BUTTON'.

b) THE PROBLEM OF MAINS VOLTAGE SPIKES CRASHING THE ORIC SEEMS TO BE CROPPING UP QUITE A LOT IN O.U.M OF LATE. IT ALWAYS EXISTED WITH MY ATMOS/MICRODISC,BUT I FOUND THE INTERRUPTIONS INFREQUENT ENOUGH TO MAKE THEM TOLERABLE. FRIDGES AND ELECTRIC COOKERS SEEM TO BE THE BIGGEST PROBLEM. THE INSTALLATION OF ANOTHER FRIDGE IN MY PLACE OF RESIDENCE, COUPLED WITH THEIR INCREASED USE IN THE SUMMER MONTHS HAS MADE THE PROSPECT OD ATTEMPTING ANY SERIOUS WORK A NIGHTMARE. THE FINAL STRAW WAS REACHED THE OTHER EVENING HOWEVER,WHEN DURING A SESSION OF 'TETRIX' AND HEADING FOR AN ALL TIME PERSONAL BEST (1996 pts. AT THE LAST SCORE),THE FRIDGE DECIDED TO SWITCH ON. I TAKE MY 'TETRIX' VERY SERIOUSLY.

I HAVE SPENT MANY HOURS OVER THE LAST FEW DAYS ATTEMPTING TO REMEDY THIS PROBLEM,ALAS,ALL TO NO AVAIL. I AM NOW STARTING TO WONDER IF A 'DIRTY MAINS' REALLY IS THE CAUSE. COULD IT BE POSSIBLE THAT ELECTROMAGNETIC INTERFERENCE CAUSED BY THE ELECTRICAL APPLIANCE IS BEING ABSORBED BY THE DRIVES RIBBON CABLE OR EVEN THE DRIVE ITSELF. HELP!!

c) I'VE BEEN WORKING ON A MORSE TUTOR OFF AND ON FOTR THE LAST 8 MONTHS. IT'S ALL IN ASSEMBLER AND TO DATE IS ABOUT 80% COMPLETE.........

d) I SEEM TO RECALL YOU ASKING IN THE O.U.M RECENTLY IF THERE WERE ANY 'DRAGON' OWNERS AMONG US 'ORICIANS'. WELL,NOT ONLY CAN I ANSWER YES TO THAT (DRAGON 64),BUT I ALSO HAVE :- ZX81,ZX SPECTRUM,SINCLAIR QL,ENTERPRISE 128,EINSTEIN 64, ATARI 520STFM. WELL,SOME PEOPLE COLLECT STAMPS... I HAVE VERY LITTLE SOFTWARE FOR THE MORE OBSCURE MACHINES ,SO MOST OF THEM JUST GATHER DUST THESE DAYS;THE EXCEPTION BEING THE 'ATARI'.

- DENIS

DEAR DENIS,

a) WHAT'S AN AMSTRAD?

WHAT HAPPENS IF YOU WANT TO USE THE 'RESET'!

b) I TOO HAD PROBLEMS WITH MY OLD MICRODISC AND THE FRIDGE. DAVID WILKIN BUILT ME A BOX OF TRICKS TO TAKE THE HEAT AWAY FROM THE PSU. IT SOLVED THE PROBLEM. WHY NOT DROP DAVID A LINE OR GIVE HIM A BUZZ. I DON'T KNOW ENOUGH ABOUT ELECTRICS TO SAY WHETHER THE DRIVE RIBBON CABLE CAN BE ABSORBING WHAT IT SHOULDN'T. I KNOW MY OLD ATMOS/MICRODISC USED TO DO ODD THINGS. e.g IF THE THING HAD HUNG UP WHEN I WAS USING A W.P, THEN I COULD SWITCH IT OFF AND ON AND GET BACK THE DATA FILE. IT MUST OF BEEN HELD IN THERE SOMEWHERE. ODDLY WITH THE FRENCH MICRODISC,THE ATMOS ITSELF IS POWERED VIA THE RIBBON CABLE.

I'M GLAD YOU TAKE YOUR 'TETRIX' SERIOUSLY. THOSE BLOCKS MUST BE COMING DOWN AT SOME SPEED AT 1996 pts. I STRUUGLE AT ABOUT 1100. I INFORMED ONE OF OUR OTHER 'TETRIX' MANIACS (HENRY MARKE OF POMPEY) OF YOUR SCORE AND HE NOW SAYS HE HAS JUST TOPPED 2000. HENRY IS COMUNG TO THE MEET,SO WHY NOT COME ALONG AND HAVE A TOURNAMENT BETWEEN YOU. A BOTTLE OF WINE TO THE WINNER! I KNOW FOR A FACT THAT I MADE A MISTAKE WITH ONE OR TWO VERSIONS OF 'TETRIX' THAT WERE SENT OUT. IF YOU ARE GRTTING A 'BREAK ON BYTE', THEN THIS COULD BE THE REASON. THE 'INIT' STATEMENT OF THE DISC SHOULD READ :- !QUIT:! TETRIX. IF THE 'QUIT' COMMAND IS MISSING THEN A PROBLEM WILL ARISE. IT CAN BE CHECKED OUT BY DOING A !INIST. I'M NOT SURE WHETHER YOU HAD THE 3" OR 3.5" VERSION,BUT IF YOU PREFER YOU CAN SEND THE 3" MAKE TO ME FOR CHECKING OR THE 3.5" TO JON.

c) KEEP UP THE GOOD WORK WITH THE MORSE TUTOR. DID YOU KNOW OF 'CW MORSE' FROM NO MANS LAND. CHRIS HEARN TELLS ME THAT IT SOUNDS SIMILAR TO WHAT YOU ARE WORKING ON AND HE HAS PUT IT TO SOME USE. I HAVE ALSO HEARD OF AN ATMOS USER (QUITE LOCAL TO ME), WHO USES HIS ATMOS WITH 'CW MORSE' - THE ONLY USE HE HAS FOR THE OLD ORIC.

d) I THINK IT WAS JOHN PEACH WHO HAD THE 'DRAGON'. JOHN ALSO HAS A QL AND ZX81 AND I BELIEVE OTHER MACHINES. THERE ARE PROBABLY USERS OUT THERE OF SOME OF YOUR OTHER MACHINES. PERHAPS THEY HAVE SOME SOFTWARE TO OFFLOAD!

- DAVE