ORIC

USER MONTHLY

with Alternative Micros

Keeping the

Oric alive

THE   EDITORIAL

HELLO AGAIN,
                    TO YET ANOTHER ISSUE OF O.U.M.
THE CRACKING ISSUE THAT I'D HOPED FOR LAST MONTH DIDN'T QUITE TRANSPIRE, BUT I THINK YOU'LL AGREE THAT THIS IS ONE OF OUR BEST.
I'M NOT GOING TO MAKE ANY PROMISES ABOUT CLEARING THE BACKLOG OR DESPATCHING THE NEXT O.U.M DISC,BECAUSE EVERY TIME I DO,AND SOME DISASTER BEFALLS ME.
IT'S TWO-FOLD THIS MONTH - JUST WHEN WORK WAS EASING DOWN AND I WAS FINISHING EARLIER (THANKS TO MEMORISING MOST OF CALLS),AND LO AND BEHOLD WE HAVE A MASSIVE SHAKE-UP AND FOR THE THIRD TIME IN A YEAR I AM HEAVILY INVOLVED - MORE WORK AND I'M EXPECTED TO START AN HOUR EARLIER (3.00 A.M). I AM NOW RESIGNED TO LOOKING FOR ANOTHER JOB,AS I CANNOT ENVISAGE ME GETTING UP BEFORE 2.30 A.M FOR THE NEXT 20 YEARS OF MY WORKING LIFE.

DISASTER NUMBER 2 CAME IN THE FORM OF MY ACQUISITION AT XMAS - THE AMSTRAD NOTEPAD NC100. I HAD PREPARED SOME OF THE CONTENT OF O.U.M ON IT AND HAD ALSO SPENT MANY,MANY HOURS INPUTTING SOME 500 SONG TITLES (PART OF MY COMPACT DISC COLLECTION). THE DATABASE (ADDRESS BOOK) HAD A FAST RESPONSE TIME AND SEEMED TAILOR-MADE. I COULD CHECK OUT A TITLE/PART-TITLE/ARTIST/EVEN ONE WORD IN UNDER 5 SECONDS. I EVEN USED IT AT A DISCO. IF SOMEONE ASKED FOR A TRACK THEN I COULD FIND IT JUST LIKE THAT WITHOT HAVING TO SEARCH THROUGH DISCS OR PAPER. THEN DISASTER STRUCK - FOR NO APPARENT REASON THE WHOLE MEMORY WIPED ITSELF. WHEN NEXT I LOOKED - NO FILES AND ALL THE MEMORY FREE. I WAS LIVID. I HAVE WRITTEN TO ALAN SUGAR AS HE WAS CO-AUTHOR OF THE SOFTWARE AND MANUAL. I WILL LET YOU KNOW WHAT TRANSPIRES,BUT MEANWHILE I NO LONGER HAVE THE LUXURY OF TYPING UP PAGES,WHILST WATCHING A BIT OF FOOTBALL.
AS I'D RECOMMENDED THE MACHINES TO FRANK BOLTON AND BOB TERRY; I JUST HOPE THAT MINE WAS JUST A ROGUE MACHINE AND THAT THERE IS NO DESIGN FAULT.

AND SO TO THE INDEX FOR THIS ISSUE:

THE INDEX
--- -----
===

MARCH O.U.M
----- -----
ARTICLES FOR INCLUSION IN THE MARCH ISSUE SHOULD REACH ME BY FEBRUARY 22nd.

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

THE MEET
--- ----

THE 1994 AYLESBURY MEET WILL TAKE PLACE ON SATURDAY JUNE 18th AT THE USUAL VENUE (RIVETS SPORTS AND SOCIAL CLUB). IT IS BEING HELD A MONTH EARLIER THAN USUAL,AND I HOPE THAT THOSE WHO ARE GENERALLY ON HOLIDAY IN MID-JULY WILL BE ABLE TO ATTEND.
IT MAY WELL BE THAT THE DATE IS VERY NEAR TO THAT FOR THE PARIS MEET,BUT UNFORTUNATELY I HAVE FOUND IT IMPOSSIBLE TO MAKE THE AYLESBURY MEET ANY LATER. I SINCERELY HOPE THAT OUR OVERSEAS ORICIANS WILL STILL BE ABLE TO MAKE THE JOURNEY - A GOOD CHANCE TO MAYBE TRY THE CHANNEL TUNNEL OR DO YOU FRENCH CALL IT LA MANCHE!
TICKETS WILL AGAIN BE JUST 2 POUNDS (1 POUND FOR UNDER 17's AND UNEMPLOYED). FOOD IS AGAIN BEING ARRANGED.
WHAT WE NEED NOW ARE DONATIONS FOR THE GRAND RAFFLE. DON'T FORGET THAT LAST YEAR A WEEK IN STEVE HOPPS'S VILLA WAS THE FIRST PRIZE. THE WINNER (ROB KIMBERLEY) AND HIS FAMILY WILL BE OFF TO SPAIN IN THE SPRING, AND I HAVE BOOKED FOR A COUPLE OF WEEK'S IN THE AUTUMN. PERHAPS SOMEONE WOULD LIKE TO DONATE A WEEK-END IN EURODISNEY FOR THIS YEAR'S TOP PRIZE? BEFORE IT IS CLOSED DOWN!

TICKETS FOR THE MEET AVAILABLE NOW! (WE HAVE TO PAY FOR THE HALL IN ADVANCE). ALSO AVAILABLE ARE RAFFLE TICKETS AT 1 POUND EACH - PRIZES? I'LL START THE BALL ROLLING BY PUTTING IN A BOX OF DISCS (3"/3.5"/5.25" - WHATEVER THE WINNER USES).
==

DRAGON AT AYLESBURY!
------ -- ---------

AN INVITE IS BEING SENT TO THE 'DRAGON' USER GROUP TO SEE IF THEY WOULD LIKE TO BRING A STAND TO OUR 'ORIC' MEET IN JUNE. I KNOW WE HAVE A FEW 'DRAGON' USERS. IF THERE IS ANYTHING ELSE YOU WOULD LIKE TO SEE REPRESENTED THEN PLEASE LET ME KNOW.
==

SEDORIC AND 'LORIGRAPH' AND 'MACADAM BUMPER'.
------- --- ---------- --- ------- ------

THOSE WHO RECIEVE THE C.E.O DISCS MAY BE WONDERING WHY THE AUTUMN AND WINTER DISCS WERE SENT OUT ON SEDORIC V1 AND NOT THE NEW V2. IT IS DUE TO THE DIFFERENCES BETWEEN V1 AND V2,WHICH CAUSE PROBLEMS WITH SOME SOFTWARE,NAMELY 'LORIGRAPH' AND 'MACADAM BUMPER'.
'LORIGRAPH' MUST BE BOOTED FROM SED. V1 (THUS A MAX. OF 55 TRACKS D/SIDED). DRAWINGS SAVED ON A SED.V2 DISC CAN BE LOADED INTO THIS V1 VERSION. YOU MAY ALSO LIKE TO KNOW THAT DRAWINGS CAN BE SAVED WITHOUT AN EXTENSION AND THE 'COM' EXTENSION BECOMES THE DEFAULT.
'MACADAM BUMPER' WILL RUN O.K ON A SED V2 DISC UP TO A POINT. IF YOU DESIGN YOUR OWN TABLE,THEN IT WILL SAVE IT,BUT NEXT TIME YOU BOOT THE DISC IT WILL NOT FULLY LOAD. THEREFORE THE GAME MUST BE STORED ON A SED.V1 DISC.
I HOPE THIS CLARIFIES THE SITUATION.

Machine Code for the Oric Atmos (Part 32)   Peter N. Bragg

Here we are, back again after the Christmas break. Have you broken all those
New Year resolutions, yet ? Are you like me ? As ever, full of plans and
good intentions, many of which will still be waiting to see the light of day,
when the Christmas tree goes up again. The New Year period is a good time
for exhibitions and the opportunities to get out and see what the rest of
the world does with it's spare time and cash (if there is any available).
The "Computer Fair" and the "Model Engineer" were worth visiting.
Interesting items included computerised tools and flight simulators, but
whatever happened to the CD ROMs we have heard so much about. Plenty of CD
players and hardware, but still not much choice in the way of CD disk
material after all these years. Ah well, on with the motley !

The Story so far
---------------- We have looked at the basic requirements for machine code
programming. A small selection of useful 6502 Instructions appeared in
Part 22 of the series. A few programming techniques, have also been covered.
The last couple of issues have concentrated on the subject of binary code
and we looked at three instructions that can be used to mask out specific
items of data. I left you with a brief illustration of a masking instruction.

The instruction used was "AND" and that example is repeated below, together
with the same setup using "ORA" and "EOR" instructions. Later on, we will
see how this sort of instruction can be made to do something useful. In the
meantime, take a look at the examples below and note how each instruction
has a different effect, when using the same data and mask.

```
          Binary        Hex
Data      1010 0101  =  A5 in Accu before.      This example shows the effect of
          ↓↓↓↓  ↓↓↓↓                            using "AND". The four bits at "0" in
                                                the mask, block the data bits above
Mask      0011 1100  =  3C in Operand.          them, thereby putting "0" into those
          ↓↓  ↓↓                                bits in the result.
Result    0010 0100  =  24 in Accu after.
```

```
          Binary        Hex
Data      1010 0101  =  A5 in Accu before.      This example shows the effect of
                                                using "ORA" and in effect reverses
                                                the mask. The four bits at "1" block
Mask      0011 1100  =  3C in Operand.          the data bits above them, thereby
          ↓↓  ↓↓                                putting "1" into those bits in the
Result    1011 1101  =  BD in Accu after.       result.
```

```
          Binary        Hex
Data      1010 0101  =  A5 in Accu before.      This example shows the effect of
                                                using "EOR". The four bits at "1" in
                                                the mask, reverse the four data bits,
Mask      0011 1100  =  3C in Operand.          before putting them in the result.
            INVERT                              The mask bits at "0" leave their
          ↓↓  ↓↓                                data bits unchanged in the result.
Result    1001 1001  =  99 in Accu after.
```

Theory is OK up to a point, but things are easier to understand when you can
actually see them work on the Oric, for real. Below is a short routine,
which demonstrates the "AND" instruction operation above, using an Immediate
version of the instruction. The end result of the masking operation is
copied into location 1002 and you can look at that address after you have
run the routine, to confirm that the results are indeed, those shown above.
To change the operation to "ORA" or "EOR" you only need to change the
instruction opcode in location 1013 from 29 for "AND" to 09 for "ORA", or 49
for "EOR" and once again you can confirm the result by PEEKing location 1002.

```
Oric                          Demo Routine                      19 Jan 94
----                          ------------                      ---------
[ CALL#1010 ]---------------[ Data Masking ]----------------------------[ / ]
                             ---Parameter Block 1001---
1001:A5          : "sample"  : Item for masking (binary value 1010 0101).
1002:            : "result"  : Storage for item produced by the instruction.
                             ---Load Item & Mask it---
1010:AD 01 10 : LDA 1001     : Load Accu with sample from location 1001 and
1013:29 3C    : AND# 3C      :      then mask it with binary value 0011 1100
1015:8D 02 10 : STA 1002     :      and store the result in location 1002.
                             ---Finish---
1018:60          : RTS       : Exit (at which point location 1002 contains 24).
            ---end---
```

The routine above, can be entered using your own favourite assembler or the
Hexloader program from Part 7 of the series. As it consists of just nine
bytes it could even be POKEd into the memory, one byte at a time.

The data, mask and final result bytes, were
all converted to and from binary and hex using
the table shown again here. For example the
mask is 0011 1100. Notice that it is written
as two groups of four digits, rather than one
group of eight. That is not important, but it
does make binary numbers easier to handle.

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

Using the conversion table, the first group
0011 converts to 3 and the second group 1100
converts to C. Put them together and you get
3C for the mask in instruction 1013 operand. The results are converted in
the same way. The "AND" operation produces 24. The 2 in binary is 0010 and
the 4 is 0100, which put together as one byte is 0010 0100, for 24 hex.

Capital Punishment ?
------------------------ These masking operations really come into their own
with graphics, but they are also used for setting up hardware such as
Input/Output ports. They can have other uses. Have you ever fallen foul of
"CAPS LOCK". The Oric has an "on screen" CAPS indicator, but other machines
are not so well endowed. Software is often controlled by presing lettered
keys, usually capital letters. If the software requires you to press key "A",
you will find that keying little "a" will not do. It's very annoying when a
computer "hangs up" because you forgot to set the CAPS LOCK correctly,
before running the software. Keying "Caps Lock" while running may not solve
the problem, because the software may prevent you from doing that and anyway
strange though it may seem, it is not always obvious what the problem is, in
the heat of the moment. It is a much better idea to incorporate a routine
into your software that in effect, automatically disables the "Caps Lock" so
that the problem cannot occur in the first place.

Take a look at the ASCII character code (listed in most computer manuals)
you will find that the only difference between small (lower case) letters
and capital (upper case) lies in Bit 5 of each character code, which is set
to "1" for small letters and cleared to "0" for capitals. A simple masking
routine will ensure that the keyboard only produces capital letter codes.
More about that next time.......In the meantime, a party political message !

Surely John Major wouldn't have had all that trouble if he had used
        assembly language, instead of going "Back to Basics" all the time !!

THE   GAMESTER

ARCADE/STRATEGY CHARTS FOR 1993
------ -------- ------ --- ----

1 (1) DONT PANIC, 2 (2) ZEBULON, 2 (-) COLUMNS, 4 (3) TETRIX, 5 (4)
MANIC MINER, 6 (4) IJK INVADERS, 6 (8) XENON III, 6 (4) ORIC MUNCH, 6 (-)
MIND MADNEZ, 10 (-) GRENDEL.

SIX OF THE TOP TEN DISTRIBUTED BY O.U.M. ALISTAIR WAY'S "GRENDEL"
CREEPS BACK IN THE CHART. CAN WE PERSUADE HIM TO WRITE ANOTHER?
===

LATEST ON THE HI-SCORE FRONT
------- -- --- -- ------ -----

MY OLD FRIENDS, HENRY AND RENE MARKE CALLED IN RECENTLY. HENRY BOUGHT
HIMSELF A TATUNG MONITOR FROM TONY CLARK, WE PLAYED ON THE ORIC, AND LATER
WENT FOR A MEAL. NOT RECOMMENDED IS 'THE WOOLPACK' IN STOKE MANDEVILLE.
 HENRY HAS NO DOGS,JUST CATS,AND ENDED UP WITH A 'CATTY BAG'. I DO HOPE
THE CATS WEREN'T TOO ILL!
 HENRY AND RENE CONTINUE HAMMERING THE ORIC: -
        DON'T PANIC - 20,470 - HENRY
(WHEN I FIRST TOLD HENRY THAT A FRENCH GUY HAD SCORED OVER 12,000,HE
DIDN'T BELIEVE IT AND THOUGHT HE MUST HAVE A CHEAT VERSION - JUST SHOWS
YOU WHAT DIFFERENCE A MONITOR MAKES!)
        INTERTRON - 8,970 - RENE
        STRESS - 578 - HENRY (THIS GAME ON OUMDISC #4)

DON'T FORGET TO GET YOUR HI-SCORES TO ME OR PREFERABLY DIRECT TO STEVE
MARSHALL.

COLUMNS

A COUPLE OF NEW HIGHS ON COLUMNS FROM STEVE MARSHALL:
   O/N - 10,221
   O/H - 2,910

DON'T PANIC! - IT'S ONLY A BUG!
----- -----

JONATHAN BRISTOW WILL BE PLEASED (!) TO HEAR THAT OUR HENRY HAS FOUND A
BUG IN "DON'T PANIC". HENRY SAYS THAT AFTER FINISHING A GAME WHERE YOU
HAVE SCORED AROUND 10,000 OR MORE; THEN ON THE NEXT GAME THE CHARACTERS DO
NOT RE-DEFINE (I.E. A's,B's etc), AND THE BORDER IS CORRUPTED. THE ONLY
THING TO DO IS TO RE-BOOT. J.B - WE AWAIT YOUR COMMENTS - I SUPPOSE YOU'LL
HAVE TO PRACTICE A BIT TO GET TO 10,000 TO PROVE IT.

INTERTRON - A BUG

I PLEASED TO REPORT THAT HENRY MARKE HAS FOUND A BUG IN A FRENCH
GAME,NAMELY 'INTERTRON'!
 APPARENTLY IF YOU SCORE MORE THAN 6,000 THEN EVEN THOUGH YOU ARE CREDITED
WITH THE CORRECT SCORE AT THE GAME END; AN ERROR OCCURS ON THE HI-SCORE
TABLE. IT APPARENTLY POINTS TO LINE 30160. I'LL TRY AND FIND TIME TO LOOK
AT IT,UNLESS ANY OF YOU HAVE THE FIX.

SONIX SPECIAL   - A method of using Sonix

   Although Sonix comes with a good manual that deals with the various parts of
the program, it fails to outline a method for putting music into the program.
The following is one method of putting sheet music into Sonix, but I stress, it
is only ONE method, and not the only way of using Sonix. I have found the
method to be simple and effective. I hope that this article may solve any
problems you may be having.
GETTING STARTED. First of all we need a piece of sheet music to work from. I
suggest using music for piano, rather than guitar/buskers music which has just
the melody line and chords. Piano music (usually) has a bass line, melody (the
tune) and a harmony part. We can use one of our sound channels for each of
these parts. It is not essential to work from sheet music, but it makes things
a lot easier !
Also extremely useful, though not essential, is a 'chart' of the bass and
treble clefs which has the Oric octaves on it. This makes conversions of notes
from sheet music form to Sonix form, much easier. (I'll try and fit one in at
the end!)
OFF WE GO. If we have a look at the Sonix screen, the main section is for
'Patterns'. You will see that there is a section for channel A. This is then
repeated for channels B + C. Select Patterns from the main menu, and Edit from
the sub-menu. This puts our 'selector bar' in the channel A section of patterns.
 To put notes in, just type in the note, (in the form C-3, D-2 etc). To get a
sharp/flat use the ZX keys - flats must be converted to their corresponding
sharp name e.g. B flat = A sharp).
All very well, but how do we get notes of different lengths ? Have a look
through your music for the SHORTEST note. This will probably be a quaver, or
semi-quaver. Make this equal to one Sonix entry. To obtain longer notes, double
the number of Sonix entries accordingly. Say our shortest note is a quaver. A
crotchet would be entered twice, a minim four times, a semibreve eight times.
Use channel A for the bass notes, channel C for the melody. Channel B is used
for the harmony line, or for chords. There may be parts of the music where one
sound channel isn't being used. If there is a chord you can use this channel
for one of the chord notes. i.e. it is not necessary to use channel A (for
example) only for bass, it can be used for any other notes, if you have space
to fit them in.
Using this system, tap in your music. If, or when you come to the end of the
pattern, press ESC select a new pattern, and carry on. If a section of music is
completed but all of the pattern hasn't been used, then put a BAR at the end of
the section.   ·
Check through your music for any sections that are repeated. It is not
necessary to tap these in again. (Read on !)
Once you've tapped in a section of music, play it back,(by selecting Play from
the master menu, and Play again from the sub-menu). No sound huh ? There are
two reasons for this. 1) The default setting for the channels is OFF. This
seems a bit barmy to me and I think the default should be ON, as you aint gonna
get no sound otherwise! Go back to the master menu and turn the channels on. 2)
Still no sound? This is because you have yet to select a volume for your notes.
HEARING IS BELIEVING. Cast your eyes over the patterns part of the main screen
and you will see the letters SOV. This has nothing to do with Russia. S stands
for Sample, O for Ornaments and V for Volume. Look down this column and you
will see everything is set to zero. Enter the patterns editor and put the
volume up to about 'A'. When you've finished that, go back and play your tune.
You will now be able to hear what you've done, but it will probably be
repeating. This is because the Events section has (also) a default of zero.
You will have entered your music into the first pattern, which is pattern 0.
This will show you how the events section is used to repeat sections, (or
patterns), of music. To play other patterns, in what ever order you require,
and repeated as you wish, here is where it's done. Just enter the appropriate
pattern number(s).
Although you can now get your tune to play, it will probably not sound as you
would like it. It hums through the notes, rather than plink, plonk, bong - or
what ever. This is where Samples come in.
SAMPLES. A plucked or struck string will produce a sound that starts loud,
continues at a slightly quieter volume, and then gradually drops in volume, (to
nothing). This can be emulated in a sample, which controls the volume of a note.

Make a sample for your shortest note like the one drawn below (s1). For your longer notes you will need to make three other samples. The first is the same as the start of s1 with the volume level continued to its end. The second is the same as the later half of s1 with the middle part again extended. These two samples can be used one after the other to produce a note the same as s1, but twice as long. To allow for longer notes you need to 'splice' in a sample which is of a constant volume level throughout. See diagrams below. Using these should make your tune sound much better.

ORNAMENTS. If you want to add vibrato, or have the pitch of a note moving around this can be done here. Change the values up and down as required. (Keep the ups and downs regular for vibrato.) You select the ornament you want for each note in the patterns section. (The SOV bit.)

WARNING. If there are several notes that you DON'T want a sample or ornament on, then make sure that your first sample/ornament is 1. Otherwise you will have to go through your music, changing the values from 0 to 'switch them off'.

Once you have tapped in all your patterns, with the appropriate samples/ornaments, and entered the order you wish to play them in, (in the events section), enter the play sub-menu. Here make sure the start and end numbers are set to suit your music. The values correspond to the events numbers.

If your music ends at event 22, then set the music to end at 23 in the play section. The start is probably 0, but if you want to play just one or two sections of the music, you can change these values so that you don't have to listen to the whole thing each time.

That should be enough information for you to get your Oric singing your favourite tunes. If you come up against any other problems, let me know. Jonathan Bristow has put a lot of time and effort into creating a Sonix, and it is well worth spending a little time and effort yourself to understand and use the program. Please get in touch rather than giving up and hiding the program away some where. It's worth a lot more than that and could prove a vital addition to the software catalogue of the Orician.

GOOD LUCK !

THE MUSO

THE SAMPLES

THE 'SPLICE'

ORIC octave 4
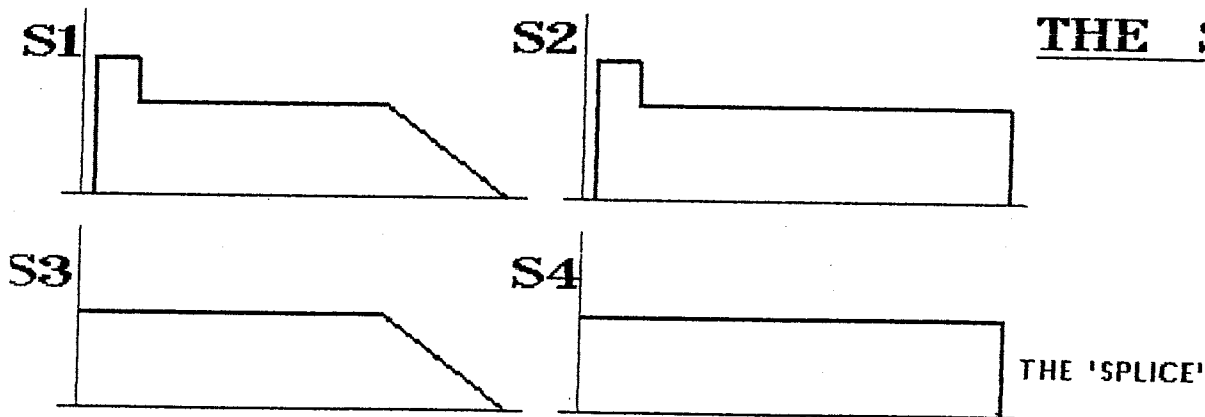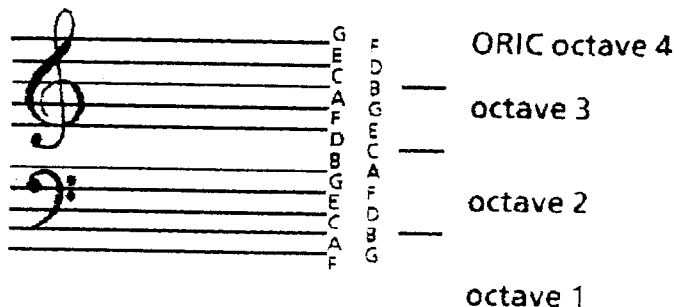
octave 3

octave 2

octave 1

RESUMÉ - We have looked at harmonics and know that a vibrating string 'divides' into uniform parts. The frequencies of these are:- f,2f,3f,4f etc. We also know that an octave is twice the frequency of its origin.(In the harmonic series, the 2nd harmonic is an octave up from the first.)

NOTES AND SCALES. Long ago, Pythagoras took time off from fiddling around with triangles to study the effect of plucking pairs of strings for sounds that were pleasing to the ear.He found that two notes sound pleasant when the ratio of the lengths of the vibrating strings is of two small integers. (He also discovered that, providing all other factors remain constant, the frequency of a stretched string is inversely proportional to its length.) The reason that one note has an integer ratio with another is that notes are derived from the harmonic series.
Let me show you an example. We will start with a note of 100Hz. The second harmonic is an octave up from this - 200Hz, but the third harmonic is not an octave up, being 300Hz. We now have a new note. If we want to create a sequence of notes between our first,(100hz), and its octave,(200Hz), we reduce the frequency of our new note, in octave jumps, until it is within the 100-200Hz frequency range. In this case we need only to jump down one octave to produce a frequency of 150Hz. (300/2).
If we had a string tuned to the first note, we can now tune a second string to our new note. The third harmonic of 150Hz will now give us an additional note - 450Hz, (150*3).We need to take this down two octaves to bring it into our 100-200 range. This gives us a note of 112.5Hz.
This process can be continued until we reach 200Hz. We then have all the notes in the musical scale, 13 in all. (12 plus our octave note). There is a big problem with this however - our octave note is calculated as being 202.71Hz, instead of 200. Our octave has been expanded!
Our notes were, in fact, a fifth apart from one note to the next.(More on this later). Today, piano tuners slightly reduce fifths so that the octaves are not expanded. If we had calculated our notes going downwards, we would have been calculating notes a fourth apart, and our octave would have been contracted. And so, modern piano tuners expand fourths to maintain the octave.

A musical scale is a seqence of notes, usually in ascending order. One of the earliest musical scales had only eight notes,(hence octave). This had several names including, the natural,ideal,just or pure diatonic scale. A musical scale on this system may be CDEFGABC - all naturals, no sharps or flats, hence 'the natural scale'. The earliest keyboards had only naturals and would be tuned to suit the music of the time - music which was vastly different from that of today. Harmony seldom consisted of more than fifths and octaves. Plain-song, still used in many religious services, springs from this time. Such instrumental accompaniments as existed would normally be performed in unison with,(at the same pitch), or in octaves with, the human voice.
The problem with the natural scale is that, if you want to play in a different key, the relationship between notes becomes different, and so sounds wrong. With such instruments as flutes, a new flute had to be made to suit a new key. (This situation didn't change until fairly recently.)
As music developed, new notes were added to the musical scale,(Sharps and flats), and the method of tuning instruments also changed. At one time A sharp and B flat were two different notes, so that, if you were playing ascending notes, you might use A sharp, and when descending B flat might be used. This was also problematic when trying to play in different keys. Eventually it was decided to make the interval between one note and the next,(i.e. a semitone), the same, whatever the starting note. This means that A sharp,(for instance), is now the same as B flat. A tune will sound 'the same' whatever key it is played in, as the relationship between notes is always the same. This is the 'equal-tempered' scale. It is true that notes in this scale are slightly 'out of tune', but any temperment is a compromise, and the equal-tempered scale is the best compromise of all.

I'll finish off with a few more notes on written music.

Where two notes are the same, but joined by a curved line, each note is NOT played separately, but only one note is played that lasts the length of the two notes. This is known as a tie and looks thus:-

A TIE

Where the notes are different, as below, the curved line is called a slur. The first note is 'slurred' into the next, via means which vary according to the instrument played.

A SLUR

As well as the time-signature (see part 3) there is also the key-signature at the start of a piece of music. There may be one or more sharps or flats written on the lines or spaces corrosponding to notes. What this means, is that every time you come to a note which has a sharp/flat in the key-signature, it is played as a sharp/flat.e.g. There may be a two sharps in the key signature. These will be a F sharp, and a C sharp, so all F's and C's are played as F sharps and C sharps. The exception is where the natural sign is used ♮ . The natural is then played for the remainder of that (one) bar, wherever the note occurs again, after which it again becomes a sharp.

STEVE THE MUSO

ALTERNATE MICROS

'SCUBA DIVE' FOR THE SPECTRUM

Steve Marshall recently picked up the Spectrum version of that old ORIC favourite 'SCUBA DIVE' (Durell Software). His comments follow:- " .. supposed to be better than the ORIC version - NO WAY ....dreadful controls make it virtually unplayable.."

ATARI LYNX revival!

One in every ten of the adverts for the new JAGUAR will feature the ATARI LYNX - perhaps we will see more software for this neat little games machine!

JAGUAR

The JAGUAR has been priced at 400 pounds, 200 pounds, and the latest is 350 pounds.
One of the companies working on software for it is LORICIEL - the French company responsible for many of the great ORIC titles.

ST

'DOGGY' from LORICIELS was a popular title for the ORIC. It's author (Eric Chahl) has moved onto the ST and has written 'ANOTHER WORLD' and 'FLASHBACK'. Both are reported to have wonderful graphics.

DRAGON AT OSSETT

Peter Thornburn has recently been in contact with the DRAGON USER GROUP and has learnt that the OSSETT Meet for this year has been scrapped,due to ack of Trade Stands needed to cover the cost of the hire of the hall.
They are hoping to stage a smaller Meet in Liverpool later in the year and would be pleased to see an ORIC stand there. If anyone is interested in manning this stand,then would they please contact me. Perhaps it is ear enough for Allan Whitaker to get involved on our behalf!

SOFTWARE CHARTS FOR 1993
———————— —————— ——— ————

THERE WERE 227 UNITS SOLD IN 1993,AND THE TOP 8 LISTED BELOW MADE UP 30% OF
THESE.
THERE WAS A TIE FOR No.1 BETWEEN DR.RAY'S "COMPILER" & JONATHAN BRISTOW'S
"DON'T PANIC". THIS BRACE JUST OUTSOLD JONATHAN'S "ZEBULON" AND NICK
HAWORTH'S "COLUMNS".
THE FIRST SIX IN THE CHART WERE ALL DISTRIBUTED THROUGH O.U.M;THUS PROVING
THAT NEW SOFTWARE REALLY CAN SELL.

POSITION	LAST TIME	TITLE	COPIES SOLD
Jt. 1	1	DON'T PANIC	11
Jt. 1	3	COMPILER	11
Jt. 3	2	ZEBULON	10
Jt. 3	—	COLUMNS	10
5	4	SONIX	9
6	6	TETRIX	7
7	4	WORDWORTH	6
8	6	3D OXO/BACKGAMMON	5

NOW FOR THE SEPARATE CATAGORIES,EXCEPT FOR 'ARCADE/STRATEGY',WHICH APPEAR
ON THE GAMESTER PAGE.

UTILITIES
————————— 1 (1) COMPILER, 2 (2) SONIX, 3 (2) WORDWORTH, 4 (—)
ASSEMBLER, 5 (—) ACCOUNT BOOK/CALC.

ADVENTURE
—————————
         1 - LAND OF ILLUSION, Jt.2 - HELLS TEMPLE & ENCHANTED.

SIMULATION
——————————
         1 - 3D OXO/BACKGAMMON, Jt.2 - IJK CHESS, TRICKSHOT, & FRIGATE
COMMANDER.

OUM DISCS
——— —————
  OUM DISC # 1 SOLD 43, #2 SOLD 41, & #3 SOLD 42.

     THANK YOU FOR YOUR SUPPORT - KEEP BUYING AND OUR WRITERS WILL KEEP
WRITING.
——

A NEW LOOK FOR O.U.M
— ——— ———— ——— —————

YOU MAY OF NOTICED THE NEW LAYOUT ON PAGE 2 - MORE CONTENT PER PAGE.
THOSE WITH 'WORD-SPEED' MAY LIKE TO KNOW HOW IT WAS ACHIEVED.
THE LEFT-HAND SIDE WAS TYPED UP AS A SEPARATE PAGE,USING A LINE MARGIN OF
ZERO AND LINE LENGTH OF 64. IT WAS THEN PRINTED IN CONDENSED (17 CPI) MODE.
THEN AFTER THE RIGHT-HAND PAGE WAS TYPED ,THE PAPER WAS WOUND BACK TO TOP
OF PAGE AND THE SECOND HALF WAS GIVEN A LINE MARGIN OF 72 AND A LINE LENGTH
OF 64,AND AGAIN PRINTED IN CONDENSED MODE.
  WOULD YOU RATHER HAVE SMALLER OR LARGER PRINTER?
==

Rambling on...

Continuing with the part of the ROM that deals with variables, we now turn to arrays. As conventions, I shall use the terms 'dimension' and 'element'; thus A(2,2,2) has three dimensions, and A(1,2,1) specifies the element of the array containing that variable. Next time we shall spend a little more time on the structure of arrays in memory.

Treat an array

Principal:
This is done by always having on top of the stack the variable name, its type, and the number of dimensions, including the value of each dimension. The flags and array components are constantly interchanged, both needing 2 bytes. It is also necessary to save the name and type of the variable, since the evaluation of elements modifies these system variables.

RETURN	H
ADDRESS	L
INTEGER FLAG	
STRING FLAG	
2nd character	
1st character	

RETURN	H
ADDRESS	L
element	H
0	L
element	H
1	L
INTEGER FLAG	
STRING FLAG	
2nd character	
1st character	
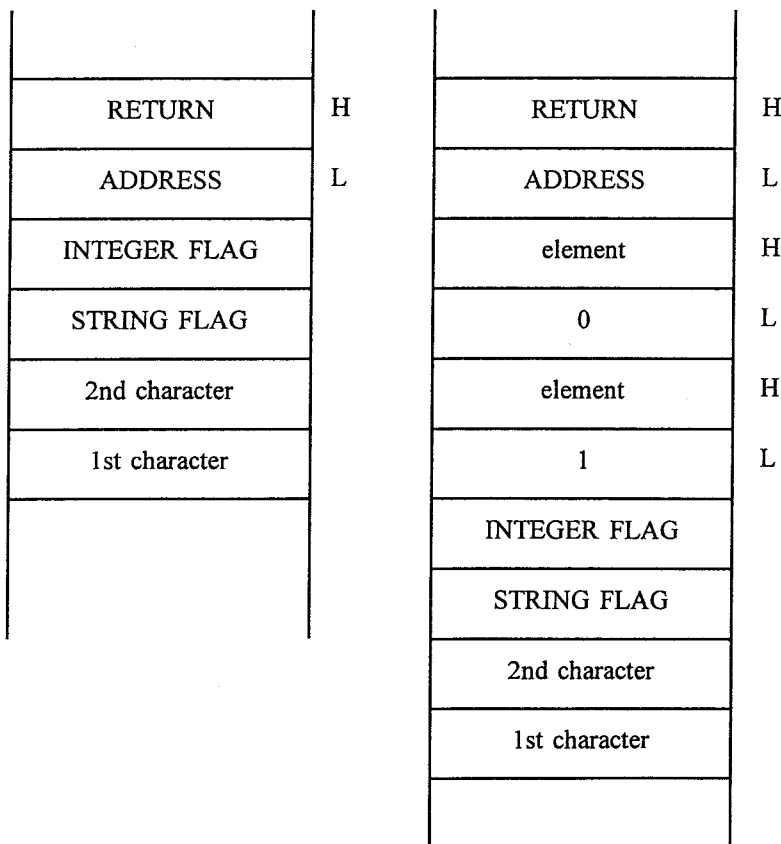
State of the stack during evaluation of the components of an array

Addr	Instr	Addr	Instr	Description
....	....	D2BB	LDA 2B	Take STORE/RECALL flag
....	....	D2BD	BNE D306	if not nul (#40), just start address
D229	LDA 27	D2BF	LDA 27	check flag
D22B	ORA 29	D2C1	ORA 29	at same time as b7 says if integer
D22D	PHA	D2C3	PHA	and save it
D22E	LDA 28	D2C4	LDA 28	String flag
D230	PHA	D2C6	PHA	is saved as well
D231	LDY #00	D2C7	LDY #00	initialise dimension counter
D233	TYA	D2C9	TYA	
D234	PHA	D2CA	PHA	save current dimension
D235	LDA B5	D2CB	LDA B5	
D237	PHA	D2CD	PHA	and array name
D238	LDA B4	D2CE	LDA B4	
D23A	PHA	D2D0	PHA	(both characters)
D23B	JSR $D20A	D2D1	JSR $D29C	Take element of current dimension
D23E	PLA	D2D4	PLA	
D23F	STA B4	D2D5	STA B4	and recover the variable name
D241	PLA	D2D7	PLA	
D242	STA B5	D2D8	STA B5	
D244	PLA	D2DA	PLA	
D245	TAY	D2DB	TAY	and the number of the current dimension
D246	TSX	D2DC	TSX	
D247	LDA 0102, X	D2DD	LDA 0102, X	Take the integer/check flag
D24A	PHA	D2E0	PHA	amd make it more accessible
D24B	LDA 0101, X	D2E1	LDA 0101, X	and the same for the string flag
D24E	PHA	D2E4	PHA	and place on top of the stack
D24F	LDA D3	D2E5	LDA D3	replace it by the element
D251	STA 0102, X	D2E7	STA 0102, X	high byte
D254	LDA D4	D2EA	LDA D4	
D256	STA 0101, X	D2EC	STA 0101, X	then low byte
D259	INY	D2EF	INY	Prepare for next dimension
D25A	JSR $00E8	D2F0	JSR $00E8	and take the current character
D25D	CMP #','	D2F3	CMP #','	another dimension?
D25F	BEQ D233	D2F5	BEQ D2C9	yes, start again
D261	STY 26	D2F7	STY 26	no, save the dimension number
D263	JSR $CFD3	D2F9	JSR $D05F	and demand a ')'
D266	PLA	D2FC	PLA	recover the string flag
D267	STA 28	D2FD	STA 28	
D269	PLA	D2FF	PLA	and the mixed integer/check flag
D26A	STA 29	D300	STA 29	only b7 is used for the integer flag
D26C	AND #7F	D302	AND #7F	on the other hand b7 must not affect
D26E	STA 27	D304	STA 27	the check flag

Find the array

Principal:
As simple as for a simple variable, except that you jump from one array to another according to the length of each.

Addr	Instr	Addr	Instr	Description
D270	LDX 9E	D306	LDX 9E	
D272	LDA 9F	D308	LDA 9F	Take start of array zone in memory
D274	STX CE	D30A	STX CE	
D276	STA CF	D30C	STA CF	and save as a pointer
D278	CMP A1	D30E	CMP A1	compare to the end of array zone
D27A	BNE D280	D310	BNE D316	not yet reached, carry on
D27C	CPX A0	D312	CPX A0	and compare the low byte
D27E	BEQ D2B9	D314	BEQ D355	at the end, not found, so create it
D280	LDY #00	D316	LDY #00	prepare index
D282	LDA (CE), Y	D318	LDA (CE), Y	first character
D284	INY	D31A	INY	

D285	CMP B4	D31B	CMP B4	already used, check second character
D287	BNE D28F	D31D	BNE D325	
D289	LDA B5	D31F	LDA B5	second character
D28B	CMP (CE), Y	D321	CMP (CE), Y	
D28D	BEQ D2A5	D323	BEQ D33B	found it!
D28F	INY	D325	INY	pass to the next; index the length
D290	LDA (CE), Y	D326	LDA (CE), Y	
D292	CLC	D328	CLC	
D293	ADC CE	D329	ADC CE	and add to the current pointer
D295	TAX	D32B	TAX	low byte in X
D296	INY	D32C	INY	
D297	LDA (CE), Y	D32D	LDA (CE), Y	and the same for the high byte
D299	ADC CF	D32F	ADC CF	
D29B	BCC D274	D331	BCC D30A	unconditional: continue the search
D29D	LDX #6B	D333	LDX #6B	'BAD SUBSCRIPT ERROR'
D29F	BYT #2C	D335	BYT #2C	
D2A0	LDX #35	D336	LDX #35	'ILLEGAL QUANTITY ERROR'
D2A2	JMP $C485	D338	JMP $C47E	

The array has been found

D2A5	LDX #78	D33B	LDX #78	Prepare 'REDIM'D ARRAY ERROR'
D2A7	LDA 27	D33D	LDA 27	check flag
D2A9	BNE D2A2	D33F	BNE D338	error if declaration
....	:................	D341	LDA 2B	authorisation flag
....	....	D343	BEQ D347	jump if OK
....	....	D345	SEC	if you only want the address, exit
....	....	D346	RTS	C=1 indicates array found
D2AB	JSR $D1F4	D347	JSR $D286	calculate actual address of array
D2AE	LDA 26	D34A	LDA 26	take the dimension number
D2B0	LDY #04	D34C	LDY #04	
D2B2	CMP (CE), Y	D34E	CMP (CE), Y	and compare to true value
D2B4	BNE D29D	D350	BNE D333	if not equal, error!
D2B6	JMP $D343	D352	JMP $D3EB	if OK, calculate the element address

DIMENSION AN ARRAY

....	....	D355	LDA 2B	Take STORE/RECALL flag
....	....	D357	BEQ D361	OK, continue
....	....	D359	JSR $E93D	reconfigure the VIA
....	....	D35C	LDX #2A	'OUT OF DATA ERROR'
D2B9	JSR $D1F4	D361	JSR $D286	calculate start of array elements
D2BC	JSR $C448	D364	JSR $C444	and verify there is enough room
D2BF	LDA #00	D367	LDA #00	initialise length (low byte)
D2C1	TAY	D369	TAY	prepare index
D2C2	STA E1	D36A	STA E1	and set length high byte to 0 for the moment
D2C4	LDX #05	D36C	LDX #05	initialise length of a real element
D2C6	LDA B4	D36E	LDA B4	first character of name
D2C8	STA (CE), Y	D370	STA (CE), Y	and save it
D2CA	BPL D2CD	D372	BPL D375	if real, save length
D2CC	DEX	D374	DEX	element length = 4 for the moment if integer
D2CD	INY	D375	INY	
D2CE	LDA B5	D376	LDA B5	second character of name
D2D0	STA (CE), Y	D378	STA (CE), Y	and save it
D2D2	BPL D2D6	D37A	BPL D37E	if real, save length
D2D4	DEX	D37C	DEX	
D2D5	DEX	D37D	DEX	if not real, 3 for string and 2 for integer
D2D6	STX E0	D37E	STX E0	save the length (low byte)
D2D8	LDA 26	D380	LDA 26	take number of dimensions

D2DA	INY	D382	INY	adjust Y
D2DB	INY	D383	INY	
D2DC	INY	D384	INY	to number of dimensions
D2DD	STA (CE), Y	D385	STA (CE), Y	and save it
D2DF	LDX #0B	D387	LDX #0B	initialise default number of elements (11)
D2E1	LDA #00	D389	LDA #00	high byte as well
D2E3	BIT 27	D38B	BIT 27	implicit declaration or DIM?
D2E5	BVC D2EF	D38D	BVC D397	jump if implicit (keep 11)
D2E7	PLA	D38F	PLA	recover element (low byte)
D2E8	CLC	D390	CLC	and adjust (element no. 0)
D2E9	ADC #01	D391	ADC #01	
D2EB	TAX	D393	TAX	and set low byte
D2EC	PLA	D394	PLA	take high byte
D2ED	ADC #00	D395	ADC #00	finally adjust
D2EF	INY	D397	INY	
D2F0	STA (CE), Y	D398	STA (CE), Y	and store high byte
D2F2	INY	D39A	INY	
D2F3	TXA	D39B	TXA	
D2F4	STA (CE), Y	D39C	STA (CE), Y	then low byte (important!)
D2F6	JSR $D3A5	D39E	JSR $D44D	calculate current length of array
D2F9	STX E0	D3A1	STX E0	and save it
D2FB	STA E1	D3A3	STA E1	
D2FD	LDY 91	D3A5	LDY 91	recover Y (saved by #D3A5/#D44D)
D2FF	DEC 26	D3A7	DEC 26	pass to next dimension
D301	BNE D2DF	B3A9	BNE D387	if there is one...
D303	ADC C8	B3AB	ADC C8	adjust end address (C=0)
D305	BCS D364	B3AD	BCS D40C	if passed, 'OUT OF MEMORY ERROR'
D307	STA C8	B3AF	STA C8	save the high byte
D309	TAY	B3B1	TAY	and save it in Y
D30A	TXA	D3B2	TXA	take low byte
D30B	ADC C7	D3B3	ADC C7	and adjust as well
D30D	BCC D312	D3B5	BCC D3BA	not forgetting the possible carry
D30F	INY	D3B7	INY	
D310	BEQ D364	D3B8	BEQ D40C	if passed, error
D312	JSR $C448	D3BA	JSR $C444	is there room?
D315	STA A0	D3BD	STA A0	
D317	STY A1	D3BF	STY A1	yes, adjust top of arrays
D319	LDA #00	D3C1	LDA #00	prepare to fill with 00
D31B	INC E1	D3C3	INC E1	adjust length of zone to clear (DEC: BNE)
D31D	LDY E0	D3C5	LDY E0	low byte of length
D31F	BEQ D325	D3C7	BEQ D3CE	if 0, no part page of memory
D321	DEY	D3C9	DEY	
D322	STA (CE), Y	D3CA	STA (CE), Y	
D324	BNE D321	D3CC	BNE D3C9	blank to the end of the page
D326	DEC C8	D3CE	DEC C8	next page pointer
D328	DEC E1	D3D0	DEC E1	and counter as well
D32A	BNE D321	D3D2	BNE D3C9	there's more, continue
D32C	INC C8	D3D4	INC C8	adjust because decremented once too often
D32E	SEC	D3D6	SEC	
D32F	LDA A0	D3D7	LDA A0	end of array
D331	SBC CE	D3D9	SBC E	minus start = length of array zone
D333	LDY #02	D3DB	LDY #02	
D335	STA (CE), Y	D3DD	STA (CE), Y	and save it
D337	LDA A1	D3DF	LDA A1	not forgetting the high byte
D339	INY	D3E1	INY	
D33A	SBC CF	D3E2	SBC CF	
D33C	STA (CE), Y	D3E4	STA (CE), Y	which is saved as well
D33E	LDA 27	D3E6	LDA 27	test if a DIM
D340	BNE D3A4	D3E8	BNE D44C	yes, finished
D342	INY	D3EA	INY	no, index the element number

Jon Haworth

SOFTWARE SOUNDS - 2
The second of a series of articles originally appearing
in 'SOFT & MICRO' between October 1984 and June 1985
written by Jean-Marie Cour
translated by Jon Haworth

We saw in the last article the traditional musical scale of twelve notes, repeated octave by octave from low to high.

The Atmos computer represents these notes as numbers; the programmer can control the three voices of the sound generator.

The Oric computers (Oric-1 and Atmos) have a simple Basic command to play a musical note:

```
MUSIC c, o, n, v
```

where the parameters *o* and *n* respectively specify the **octave**, a number from 0 to 7, and the **note**, a number from 1 to 12.

No doubt the author of the ROM hoped that the musician would recognise here the familiar octave on the piano. The octave is repeated six times on the piano, from the lowest on the left to the highest on the right. In between, the musician uses such expressions as 'A4' to describe the note A in the fourth octave; on the Atmos it is the pair of numbers: 4,10.

Musicians are people of tradition, and will probably be happy with such a way of coding. But for the computer user, it's not so clever...
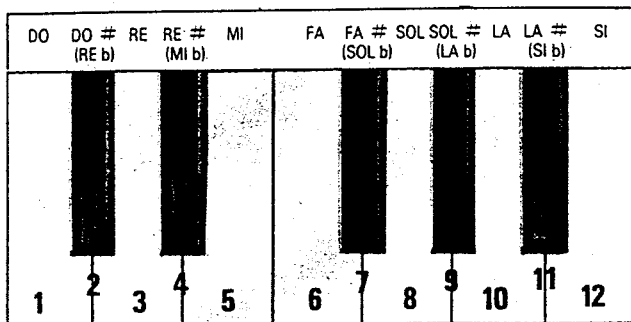
The Octave

Octave and note: one number

In Oric code, it is therefore necessary to use *two* variables, let's say O and N, to represent *one* note. Moreover, the note values of 1 to 12 oblige the programmer to use complicated routines for very simple 'musical' operations.

For example, let's write a subroutine, given our variables O and N, to pass to the next note. It's easy for the notes from C to B♭, you just add 1 to N; but with B, which is followed by C (=1) of the next octave (+1) we have a special case. So we need something like this:

```
1000    REM Next note
1010    N=N+1
1020    IF N,13 THEN RETURN
1030    N=1: O=O+1
1040    RETURN
```

In writing computer programs, economy of code is a virtue which leads to quicker programs. Certainly it would be preferable to use only the one variable N, reducing our subroutine to:

```
1000 REM Next note
1010 N=N+1: RETURN
```

Origin zero; transposition

The most natural code is always that starting from zero (as any mathematician will confirm), taking as its origin the 'smallest' element there is. For playing notes on the Atmos, this is the C of octave 0. You then

take the notes as they come, assigning them to successive whole numbers. The C♯ of the same octave will be note 1, the D note 2, etc.

Setting out the codes of all the notes is just the sort of task at which the computer excels. With the following program a table is produced on your printer which numbers the seven octaves of the Atmos from 0 to 95, with their traditional names.

```
10   REM Note table
12   LPRINT
13   LPRINT "** Note numbers **"
14   LPRINT "Octave   Note
     Number   Classic name"
16   LPRINT
20   GOSUB 200
40   FOR OCTAVE=0 TO 7
50   : FOR N=1 TO 12
60   : LPRINT OCTAVE,
     N, (OCTAVE*12)+(N-1), N$(N)
70   : NEXT
80   NEXT
90   END
200  REM Table of note names
210  DIM N$(12)
220  FOR K=1 TO 12
230  READ N$(K)
240  NEXT
250  RETURN
300  DATA C, C♯/D♭, D, D♯/E♭, E, F,
     F♯/G♭, G, G♯/A♭
310  DATA A, A♯/B♭, B
```

Transposition is one of those disheartening exercises the secret of which is firmly held in music schools. And with traditional notation, who would argue!

The task is this. A piece of music is written in a certain key. Here, for example, are the first notes of 'Tambourin' (by Jean-Philippe Rameau)


written in the original key of E minor, as indicated by the F sharp sign immediately after the clef. For any one of a number of reasons, such as the range of a particular singer's voice, it may be necessary to shift the music higher or lower, i.e. to reproduce each note at a given interval above or below its original position. This adapts the piece for playing in a higher or lower voice.

The transposition of 'Tambourin' three semitones higher changes its key to G minor. The novice will recite in his head the three elementary intervals (E, F, F♯, G).

Arithmetic notes							
Oct-ave	Note	Number	Name	Oct-ave	Note	Number	Name
0	1	0	C	4	1	48	C
0	2	1	C♯/D♭	4	2	49	C♯/D♭
0	3	2	D	4	3	50	D
0	4	3	D♯/E♭	4	4	51	D♯/E♭
0	5	4	E	4	5	52	E
0	6	5	F	4	6	53	F
0	7	6	F♯/G♭	4	7	54	F♯/G♭
0	8	7	G	4	8	55	G
0	9	8	G♯/A♭	4	9	56	G♯/A♭
0	10	9	A	4	10	57	A
0	11	10	A♯/B♭	4	11	58	A♯/B♭
0	12	11	B	4	12	59	B
1	1	12	C	5	1	60	C
1	2	13	C♯/D♭	5	2	61	C♯/D♭
1	3	14	D	5	3	62	D
1	4	15	D♯/E♭	5	4	63	D♯/E♭
1	5	16	E	5	5	64	E
1	6	17	F	5	6	65	F
1	7	18	F♯/G♭	5	7	66	F♯/G♭
1	8	19	G	5	8	67	G
1	9	20	G♯/A♭	5	9	68	G♯/A♭
1	10	21	A	5	10	69	A
1	11	22	A♯/B♭	5	11	70	A♯/B♭
1	12	23	B	5	12	71	B
2	1	24	C	6	1	72	C
2	2	25	C♯/D♭	6	2	73	C♯/D♭
2	3	26	D	6	3	74	D
2	4	27	D♯/E♭	6	4	75	D♯/E♭
2	5	28	E	6	5	76	E
2	6	29	F	6	6	77	F
2	7	30	F♯/G♭	6	7	78	F♯/G♭
2	8	31	G	6	8	79	G
2	9	32	G♯/A♭	6	9	80	G♯/A♭
2	10	33	A	6	10	81	A
2	11	34	A♯/B♭	6	11	82	A♯/B♭
2	12	35	B	6	12	83	B
3	1	36	C	7	1	84	C
3	2	37	C♯/D♭	7	2	85	C♯/D♭
3	3	38	D	7	3	86	D
3	4	39	D♯/E♭	7	4	87	D♯/E♭
3	5	40	E	7	5	88	E
3	6	41	F	7	6	89	F
3	7	42	F♯/G♭	7	7	90	F♯/G♭
3	8	43	G	7	8	91	G
3	9	44	G♯/A♭	7	9	92	G♯/A♭
3	10	45	A	7	10	93	A
3	11	46	A♯/B♭	7	11	94	A♯/B♭
3	12	47	B	7	12	95	B

Re-writing

Our musical novice hasn't finished yet! He must learn more or less by heart the accidentals corresponding to the different keys. For G minor, two flats will replace the sharp of E minor:
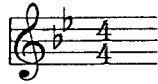
He must also transcribe each note three semi-tones higher, taking account of the accidentals. This gives:

Come on, experienced musicians, be honest: how much time and trouble has it taken to 'see' at first glance that it's the same melody?

Using our code, the same notes of 'Tambourin' are *represented* by these numbers:

```
40   42   43   42   40   47   47   45...
E    F#   G    F#   E    B    B    A
```

To 'number' a score, the numbers can be written below each note on the score. This can be done very quickly with the help of the table we got. Bearing in mind that in the original key the stave with the G clef covers octaves 3 and 4, and taking account of the global F#, we can see that:

E	on octave 3	is 40
F#	on octave 3	is 42
G	on octave 3	is 43
etc.		

Transposition therefore becomes, literally, child's play!

By definition, to transpose a piece is to transpose each note by moving it a certain number of semi-tones. Since a transposition of one semi-tone means add or subtract 1 in our code, it follows that the transposition of an entire piece is done by adding to each note the required number (of semi-tones).

So, 'Tambourin' in numbers is transposed into G minor with an increment of +3:

E minor			G minor
(E) 40 +3	=		43 (G)
(F#) 42 +3	=		45 (A)
(G) 43 +3	=		46 (B♭)
and so forth.			

The result is faultless. And it's easily done on the computer; just repeat an instruction such as N=N+T. So now brave readers can attempt to program transposition in Oric 'octave/note' notation.

The attentive will already have had the passing thought: This is all very well, but it's still going to be necessary to break down the single number in order to use the MUSIC command. Quite so.

In Atmos Basic there is a useful type of variable which is ideal for our purposes, the integer variable (one with the suffix %). Suppose then that we have a coded note, $C\%$, and that we want to find:
- its octave, $O\%$
- the Atmos note $N\%$, in that octave.

The octave is obtained by **integer division** of $C\%$ by 12:

$$O\% = C\%/12$$

There is no remainder problem with this kind of variable. The remainder of this division gives:

$$N\% = C\%-(O\%*12)$$

which will give 0 for C, 1 for C#, etc. You then simply add 1 to get the Atmos note number. Here is a simple subroutine to achieve this:

```
10   C%=45
20   GOSUB 1000
30   MUSIC 1,O%,N%,8
40   PLAY 1,0,0,0
50   END
100  REM Convert code to octave/note
110  O%=C%/12: N%=1+(C%-(O%*12))
120  RETURN
```

Towards a complete code

In the first article, we used, without much thought, DATA lines to represent the score of 'Frère Jacques'. Now let's try to achieve as pleasant a representation as possible. By 'pleasant' the computer programmer means:

- a representation as condensed as possible in terms of data
- without pointless repetition
- which is easy to modify ('edit')
- and if possible 'readable'.

We have already made progress in that we have one number instead of two to represent the pitch of a note. We now have to represent its **length**, not forgetting rests.

To illustrate our approach to a single code, the author has in his weakness written a short piece, named 'Atmos' (below).

The numbering of the pitch of the notes is shown, and there is no need to repeat how that is done. As for their length, musicians use the crotchet as their point of reference, giving it a conventional length of 1. This gives a length of two for the minim, and ½ for the quaver.

In the computer, that doesn't help much. For timing, it only knows the WAIT instruction, expressed in hundredths of a second. Thus to use fractions causes 'floating point' calculations, which are costly in terms of speed of execution.

A good way to provide for such contingencies is take as the base unit the lowest common denominator of the note lengths, i.e. the note length of which all others are multiples. In our 'Atmos' piece, no note is shorter than a quaver. So, our lowest common denominator is the quaver, which we give the length 1, making the crotchet 2 and the minim 4 units in length.

Truly compact code...

Rests are calculated in the same way. In our code so far, we have no note higher than 95. So we can take the opportunity to include in one number both the length and pitch of each note using the formula:

$$(100 * length) + pitch$$

The length therefore appears as the first number of our new code, and the first four codes for 'Atmos' will be:

452   143   142   143   145

The rest is not truly a note, and so we give it a number 'outside' our code numbers, 99. So the first rest is coded as:

299

.. and without repetition

Let's now be even more economical. Often in music you get successive notes which have the same length. As we write the notes in the order of the score, why repeat the same length as we have just assigned to the preceding note? Put another way, we only need a 'hundreds

prefix' if the length of a note is different from that of its predecessor. Our new more compact code therefore reads as follows:

$$452, 143, 42, 43, 45, \ldots$$

for the first bar of the score.

We now only have to mark the end of the score; why not with a negative number? Here,, therefore, is how to write our 'Atmos' piece in the form of DATA:

```
5000   REM   Compacted 'Atmos' code
5010   DATA 452,143,42,43,45
5020   DATA 299,47,45,47
5030   DATA 154,52,54,57,250,52
5040   DATA 152,50,48,47,250,250
5080   DATA 852
5090   DATA 150,48,47,54,440
5100   DATA 140,40,42,43,43,43,45,47
5110   DATA 299,52,51,50
5120   DATA 140,40,42,43,840
5130   DATA -1
```

There is one line of DATA for each bar to simplify reading, editing and correcting the program.

One point to note: in the last line we interpret the two minims which are tied as a semibreve equal to a length of 8. The last rest is omitted, because it is at the end!

The Barrel Organ...

To play the piece, we need an instrument, in our case the computer's sound generator. But it cannot read our compacted code. The chip works via the MUSIC command rather like a Barrel Organ, which opens a pipe according to the note and octave required. You hold the note for the length required by WAITing (There are better ways, but they must await a later article).

At each beat we need three values: octave, note and length. They can sensibly be placed in three tables. Thus our driver sub-routine for our organ can be written as follows:

```
1000   REM Music driver
1010   FOR I=1 TO FINISH
1020   PLAY 0,0,0,0
1025   IF O%(I)=99 THEN 1050
1030   MUSIC 1, O%(I), N%(I), 8
1040   PLAY 1,0,0,0
1050   WAIT D%(I)
1060   NEXT
1070   RETURN
```

Before this driver can play a piece we have coded, we must convert our encoded numbers into three tables, one each for the octave O%, the Atmos notes N% and the length D%.

Suppose a variable RHYTHM contains (in hundredths of a second) the length of our base note, here the quaver. We convert our code with a sub-routine such as this:

```
2000   REM Conversion
2010   DIM O%(100), N%(100), D%(100)
2020   I=0: T%=1: REM T% is note length
                         variable
2030   I=I+1: READ C%: IF C%=-1 THEN
              RETURN
2035   FINISH=I
2040   D%(I)=RHYTHM*(C%/100)
2050   IF D%(I)=0 THEN D%(I)=T%
2060   T%=D%(I): REM save the length
2070   C%=C%-100*INT(C%/100): REM rest
2072   IF C%<99 THEN 2075
2074   O%(I)=99: GOTO 2030
2075   REM ****** transpose here *****
2080   O%(I)=C%/12
2090   N%(I)=1+(C%-12*O%(I))
2100   GOTO 2030
```

Rests are marked as 'octave 99' and detected as such in the driver. Now we only have to play the piece... or any other that you transcribe!

```
10   RHYTHM=20
20   GOSUB 2000
30   GOSUB 1000
40   PLAY 0,0,0,0
50   WAIT 500: GOTO 30: REM replay
```
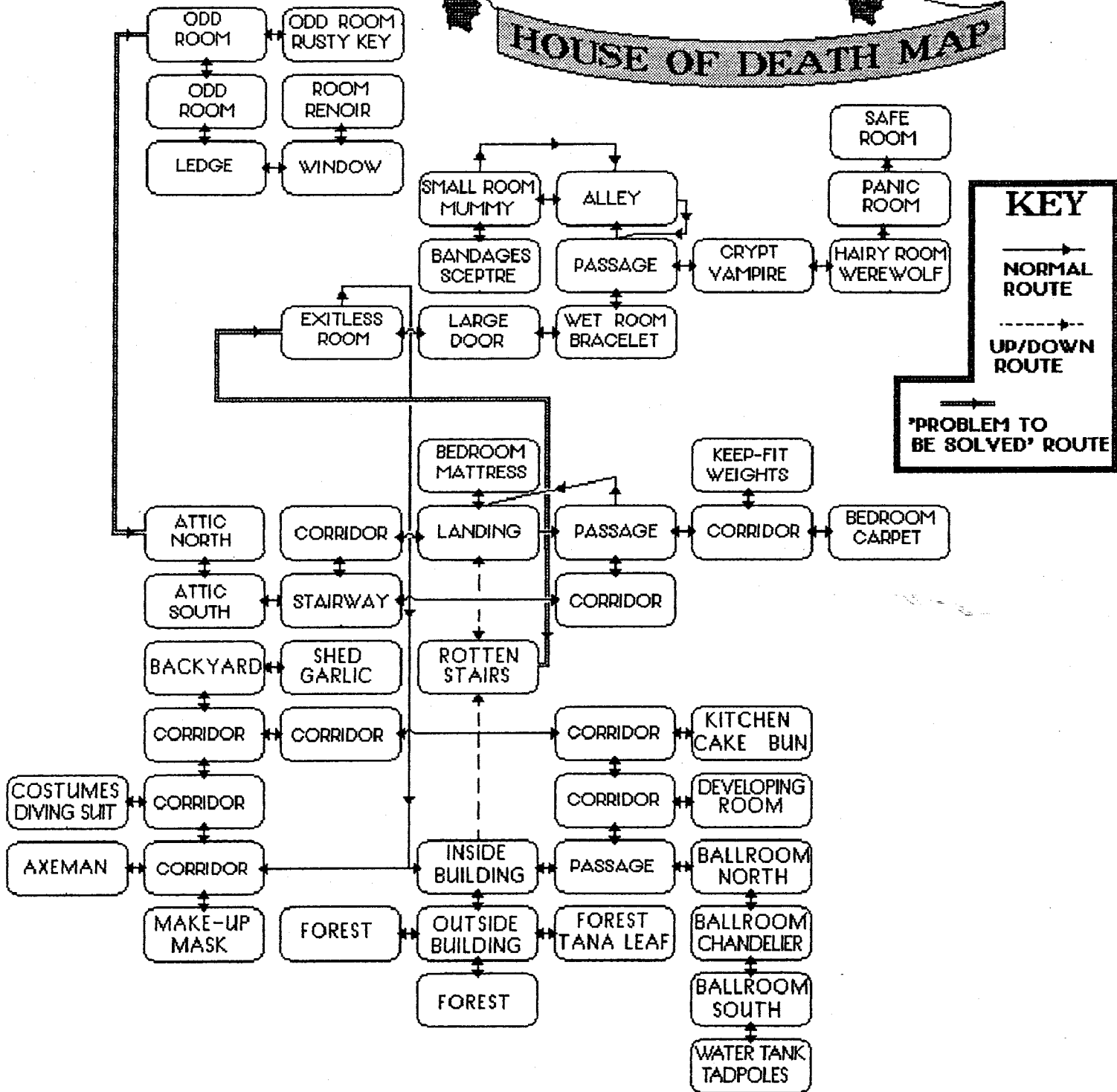
Now transpose

To end here is a question for you to reflect on: what instruction should be put in line 2075 to transpose the piece by X semitones? By now, the answer must be clear to you...

Frere Jacques

drawn by
Ally Scott
1994

HOUSE OF DEATH MAP

ODD ROOM	ODD ROOM RUSTY KEY
ODD ROOM	ROOM RENOIR
LEDGE	WINDOW

SMALL ROOM MUMMY — ALLEY

BANDAGES SCEPTRE — PASSAGE — CRYPT VAMPIRE — HAIRY ROOM WEREWOLF

SAFE ROOM
PANIC ROOM

EXITLESS ROOM — LARGE DOOR — WET ROOM BRACELET

KEY

→ NORMAL ROUTE

---▶- UP/DOWN ROUTE

➤ 'PROBLEM TO BE SOLVED' ROUTE

BEDROOM MATTRESS

KEEP-FIT WEIGHTS

ATTIC NORTH — CORRIDOR — LANDING — PASSAGE — CORRIDOR — BEDROOM CARPET

ATTIC SOUTH — STAIRWAY — CORRIDOR

BACKYARD — SHED GARLIC — ROTTEN STAIRS

CORRIDOR — CORRIDOR — CORRIDOR — KITCHEN CAKE BUN

COSTUMES DIVING SUIT — CORRIDOR — CORRIDOR — DEVELOPING ROOM

AXEMAN — CORRIDOR — INSIDE BUILDING — PASSAGE — BALLROOM NORTH

MAKE-UP MASK — FOREST — OUTSIDE BUILDING — FOREST TANA LEAF — BALLROOM CHANDELIER

FOREST — BALLROOM SOUTH

WATER TANK TADPOLES

THE MAP FOR 'HOUSE OF DEATH' (TANSOFT) WAS SENT INTO O.U.M A FEW YEARS BACK (WHEN ROBERT COOK WAS EDITOR). UNFORTUNATELY ROBERT COULDN'T REMEMBER WHO SENT IT. ORIGINALLY IN PENCIL FORM;IT HAS NOW BEEN RE-DRAWN ON A P.C BY ALLY SCOTT,WHO IS STEVE MARSHALL'S CURRENT PASSION! OUR SINCERE THANKS TO ALLY FOR HER EFFORTS,AND AS A THANK YOU WE PRINT HER PICTURE ON ANOTHER PAGE OF THIS ISSUE.

FRANK & SPAIN & ROMANIA

Frank Bolton is an extraordinary fellow,who I have got to know through letters and via the telephone.

A retired school teacher, Frank is probably busier now than any 'working man'. He has set up Spanish schools to teach English to the Spaniards,visits Spain, and more recently has been to Romania on voluntary work. In fact in the past year Frank has been to Spain four times and Romania twice. In that time he has also moved twice and spent spells in hospital.

Why am I telling you all this?

I'll quote from one of Frank's recent letters -

" WE HAVE ADOPTED A SCHOOL IN THE VILLAGE OF CURTEA DE ARGES IN ROMANIA. WE VISITED THEM IN THE SUMMER HOLIDAYS TO FIND THE TEACHERS WORKING - SCRUBBING THE FLOORS,PAINTING THE WALLS,AND REPAIRING THE DESKS THAT LOOKED LIKE SOMETHING OUT OF 'DICKENS'. THEY TOLD US THAT THE GOVERNMENT ALLOWS THEM NEXT TO NO MONEY FOR RUNNING THE SCHOOL ,AND PAYS THEM THE EQUIVALENT OF 50 DOLLARS A MONTH. AS MEAT AND FISH ARE ALMOST AS EXPENSIVE THERE AS HERE,THE WAGE DOESN'T GOT FAR. IT'S A MIRACLE THAT THEY STILL HAVE ENTHUSIASM. I PROMISED TO SEE THAT THEY HAVE A LANGUAGE LABORATORY THIS YEAR WITH AN ATMOS, A T.V, VIDEO,DISK DRIVE,CASSETTE SYSTEM AND MY ENGLISH COURSE WITH SOUND ON CASSETTE (I USE MOTOR ON/MOTOR OFF TO SYCHRONISE THE SOUND AND IMAGE), AND ALSO MANY OF THE LESSONS RECORDED ON VIDEO WITH ACTION PICTURES FILMED ON MY VIDEO CAMERA AND WITH SPEECH DUBBED IN AFTERWARDS AND SUB-TITLED AS WELL USING A COMBINATION OF CAMERA AND COMPUTER. IT IS FASCINATING WORK AND IS GOING TO TAKE ME A YEAR TO COMPLETE,BUT IT IS WELL WORTH IT.

ROMANIA IS SO POVERTY STRICKEN THAT ANY HELP WE SEND IS ONLY LIKE TRYING TO FILL A BUCKET WITH A HOLE IN THE BOTTOM. IF THERE IS ANY HOPE FOR THE COUNTRY TO LIFT ITSELF OUT OF 45 YEARS OF COMMUNISM,IT IS BY HELPING THE YOUNGER GENERATION TO COMPETE IN THE MODERN WORLD. AND TO DO THAT THEY NEED LANGUAGES. LET'S HOPE THAT BEFORE I 'KICK THE BUCKET' THAT I CAN BE OF SOME USE IN GIVING THEM A 'LEG-UP'."

HOW CAN WE HELP?
--- --- -- ----

WELL FRANK HAS INDEED 25 ORIC'S SCATTERED AROUND (SPAIN,ROMANIA,AND LEICESTER). MANY WILL BE USED TO TEACH SPANISH STUDENTS IN THE SUMMER.

SOME NEED MENDING - WILL YOU HELP?

SOME NEED TO BE UPGRADED FROM ORIC-1 TO ATMOS - WILL YOU HELP?

IT APPEARS TO ME THAT FRANK IS DOING MORE THAN HIS SHARE,FOR FRANK IS ALSO CURRENTLY TRAINING LUIS (A SPANIARD) TO RUN HIS OWN ENGLISH SCHOOL.

ASTHMA AND ARTHRITIS AND 'FATHER TIME' DO NOT HELP FRANK'S CAUSE.

I WILL SEE THAT THE BRAND NEW ATMOS DONATED BY NIGEL ALEFOUNDER IS SENT TO FRANK. INDEED I HOPE TO MEET UP WITH FRANK IN THE SPRING TO DELIVER WHATEVER ELSE CAN BE SPARED,AND OF COURSE TO TALK ORIC AND SUP BRANDY.

WHAT ELSE CAN YOU DO?

YOU KNOW THE TASK INVOLVED - HAVE YOU SPARE ORIC'S,DRIVES,VIDEO RECORDERS AND RELATED ITEMS e.g. - DISCS,DATA TAPES,VIDEO TAPES,AUDIO TAPES,PRINTERS,PAPER ETC. - USE YOUR IMAGINATION.

IF YOU HAVE NEVER DONE ANYTHING WORTHWHILE IN YOUR LIFE,THEN NOW IS YOUR CHANCE. NOT ONLY WILL YOU BE HELPING A GOOD CAUSE,BUT YOU WILL HELP TO KEEP THE ORIC ALIVE BY PUTTING IT ON THE MAP IN ROMANIA.

THIS PLEA IS NOT JUST FOR THE BRITISH - IT IS ALSO MEANT FOR OUR FRENCH,GERMAN,SCANDINAVIAN,DUTCH AND WELSH READERS. YOU ALL CAN HELP - EVERY LITTLE BIT COUNTS!

I WOULD ADVISE THAT YOU CONTACT FRANK BEFORE DESPATCHING OR DELIVERING ANYTHING,OR ALTERNATELY YOU CAN CONTACT ME. FRANK IS IN LEICESTER,WHICH IS QUITE CENTRAL AND I'M SURE WE CAN ARRANGE TRANSPORT IN SOME FORM OF SHUTTLE IF WHAT YOU HAVE TO OFFER IS BULKY. THE FRENCH CAN ALWAYS SEND THINGS OVER WITH THEIR REPRESENTATIVES TO OUR NEXT ORIC MEET.

DON'T PUT THIS ARTICLE DOWN AND FORGET IT. PLEASE DO SOMETHING - BE IT A DONATION OF GOODS, A DONATION OF CASH (TO BUY DISC SYSTEMS etc.) OR A DONATION OF YOUR TIME TO PERFORM SOME TASKS.

FRANK BOLTON IS AT: 23 MELTON AVENUE,LEICESTER LE4 7SE (TEL: 0533 662056).

A FINAL PLEA - FRANK WOULD APPRECIATE ANY EDUCATIONAL PROGRAMMES FOR THE ORIC THAT MIGHT HELP ROMANIAN KIDS TO LEARN ENGLISH.

AND FINALLY - I MAKE NO APOLOGY FOR USING THE PAGES OF O.U.M TO MAKE THIS APPEAL. IF THERE ARE ANY OBJECTIONS - THEN TOUGH!

- DAVE DICK

LATE NOTE
---- ----

THERE ARE NOW 2 ORICS AT THE ROMANIAN SCHOOL,AND FRANK TELLS ME THAT ALL THE FIVE YEAR OLDS ARE 'HOOKED' ON 'FISHERMAN SAM' (PUBLIC DOMAIN).

MCP 40 HIRES DUMP

THE FOLLOWING PROGRAM FROM BRIAN KIDD WAS USED FOR THE 'GIRL' PICTURE ON THIS PAGE. IT TOOK A LONG WHILE TO PRINT,AND I HAVE OTHERS,WHICH WE WILL PRINT AS AND WHEN.

```
1 HIRES
2 !GIRL.SCR
8 LPRINTCHR$(18)
9 LPRINT"I"
10 FOR F=0 TO 199
11 FOR Z=1 TO 2
12 LPRINT"H"
13 LPRINT "RO,-1"
14 LPRINT "I"
20 FOR G=0 TO 239
30 P=POINT(G,F)
35 IF P= -1 THEN P=1
40 GOSUB 1000*(P+1)
50 NEXT G
55 NEXT Z
60 NEXT F
70 LPRINT"A"
80 STOP
1000 LPRINT"J2,0"
1010 RETURN
2000 LPRINT"R2,0"
2010 RETURN
```
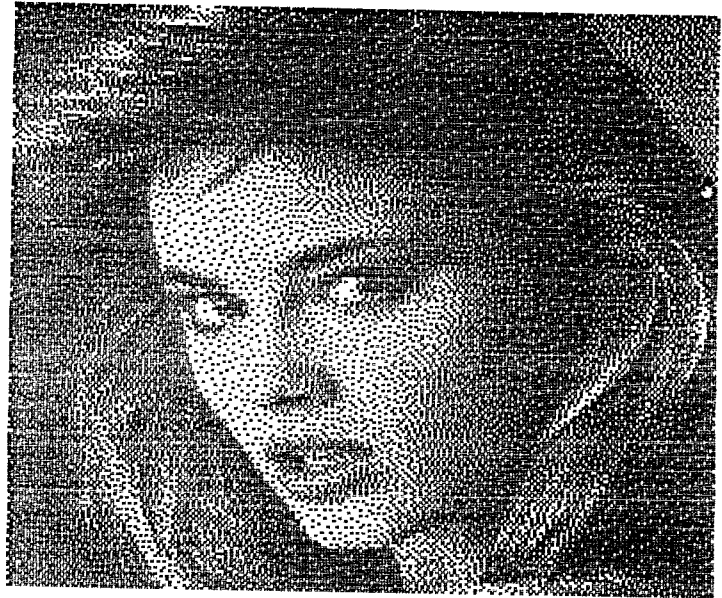

"ALLY-OOPS!"

A FEW NOTES CAME WITH THE LISTING THAT I HAVE NOT HAD TIME TO CHECK OUT E.G: INCLUDE LINE TO STRIP ATTRIBUTES -
```
36 IF P<>1AND PC<>0THEN P=0
```

```
ALTERNATIVE: 30 P=POINT
   35 P=ASC(P)
   36 IF P>1 THEN P=0
```
===
'ELSE' BUG

AS PREVIOUSLY MENTIONED IN O.U.M,THERE IS A BUG IN 'ELSE'. DR.RAY HAS COME UP WITH THE SOLUTION. THE FIRST LISTING,WHICH WE WILL CALL "ELSEBUG",DEMONSTRATES THE BUG. THE SECOND LISTING,WHICH WE WILL CALL "ELSEOK",IS THE CORRECTION. THE CURE IS THE SINGLE QUOTE (FOR COMMENT) IMMEDIATELY BEFORE "ELSE".

ELSEBUG

```
100 PRINT "Enter a letter ";
105 GET A$
110 IF A$>"M" THEN PRINT A$;  ELSE PRINT "LOW";:PRINT " EXTRA"
```

ELSEOK

```
100 PRINT "Enter a letter ";
105 GET A$
110 IF A$>"M" THEN PRINT A$;  'ELSE PRINT "LOW";:PRINT " EXTRA"
```

M O R E   B I T S ' N '   B O B S

A CHAT WITH DR.RAY

OVER THE PAST FEW MONTHS I HAVE RECIEVED SEDORIC UPDATES AND SOME HANDY UTILITIES FROM RAY. TIME CONSTRAINTS AND NOT FULLY UNDERSTANDING SOME OF HIS NOTES THAT CAME WITH THEM;HAS MEANT THAT THESE WERE SET ASIDE.
   I THOUGHT IT IN MY AND OTHERS BEST INTERESTS TO SORT THROUGH THEM,AND THANKS TO A LENGTHY PHONE CALL WITH RAY,HAVE COME UP WITH SOME ITEMS THAT MAY BE OF HELP TO OTHERS. AT THE SAME TIME I HAVE GIVEN RAY A FEW MORE THINGS TO WORK ON.

SECTMAP

A 'FIX' FOR 'SECTMAP' WAS PUBLISHED IN AN EARLIER ISSUE OF 'O.U.M' AND I HAVE ALSO RECIEVED THE UPDATE ON DISC FROM RAY. UNFORTUNATELY THE ONLY DISC THAT IT WILL READ IS THE ONE WITH THE 'SECTMAP' PROGRAM ON IT. THUS IF YOU WANT TO READ ANOTHER DISC,THEN YOU FIRST HAVE TO TRANSFER THE PROGRAM TO THAT DISC. IT APPEARS A FAIRLY ROUTINE JOB FOR RAY TO AMEND AND WE WILL KEEP YOU POSTED.

LISTINPUT

THIS HANDY UTILITY HAS ALLOWED ME TO TRANSFER A BASIC LISTING INTO A 'WORD-SPEED' DOCUMENT. THE LISTINGS SHOWN ON THE PREVIOUS PAGE WERE DONE IN THIS WAY. A GREAT BOON TO ME,AS I CAN NOW TYPE AN INTRODUCTION TO A LISTING AND THEN LOAD THE LISTING INTO THE DOCUMENT. THE METHOD USED WITH THE UTILITY IS VERY SIMPLE.

   I WILL DELVE FURTHER INTO THESE PROGRAMS FROM RAY AND WILL PUBLISH THEM IN OUM IF THEY ARE IN BASIC OR ELSE PUT THEM ON AN OUM DISC.
   SINCERE THANKS TO RAY FOR HIS TIME AND EFFORT.
==

LATE 'PANIC' NEWS

AS WE GO TO PRESS,A FINAL UPDATE FROM HENRY MARKE.... STOP PRESS... LATEST HI-SCORE ON 'DON'T PANIC' IS 25,490... IT TOOK HIM ABOUT HALF AN HOUR....HE IS TO HAVE A REST FROM THE GAME NOW TO CONCENTRATE HIS EFFORTS ON 'COLUMNS'......WATCH THOSE HI-SCORES TUMBLE 'MUSO'.

A NEW GOLF GAME FOR THE ORIC!

   SPURRED ON BY THE PATHETIC EFFORT OF 'R&R' TO WRITE A GOLF GAME FOR THE ORIC; KIERON SMITH HAS BEEN WORKING ON ANEW GOLF GAME FOR THE ATMOS. KIERON SAYS THAT THE GRAPHICS AREN'T BRILLIANT,BUT IS STILL MORE LIKE THE FAMILIAR VERSIONS FOUND ON PC'S AND SEGA ETC. IT WILL CONTAIN LAKES,STREAMS,TREES AND BUNKERS ETC.
   I LOOK FORWARD TO SEEING IT AND WILL HOPEFULLY RELEASE IT THROUGH OUR 'MIRAGE' LABEL.
===

MESSAGE TO JAMES GROOM FROM KIERON SMITH

   THANKS FOR THE SYMPATHY FOR MY POSITION. I AM SURE THAT WE WILL BOTH FIND SOMETHING SOON.

MESSAGE TO ANYONE FROM KIERON

I WASN'T IMPRESSED AFTER PLAYING A SEGA MEGA DRIVE OVER XMAS. I PREFER THINGS LIKE 'ANT ATTIC' ON THE SPECCY. HOW ABOUT SOME MASOCHIST OUT THERE WRITING A VERSION FOR THE ORIC!
   - I'D BUY IT.

OUM DISC # 4

AS I HAVE ACHIEVED WHAT I SET OUT TO DO WITH THIS ISSUE (AND ON TIME), THEN I AM BEAVERING AWAY ON OUMDISC #4. ALL 3.5" DISCS ARE FORMATTED TO THE NEW SEDORIC V2.1 NAD I AWAIT A CONSIGNMENT OF 3" DISCS. THUS 3.5" DISCS WILL BE SENT OUT WITH THIS ISSUE AND 3" WILL BE DESPATCHED SHORTLY AFTERWARDS. SINCERE APOLOGIES FOR THE DELAY.
  INCLUDED ON THE DISC ARE UPGRADES TO SEDORIC WRITTEN IN SUCH A WAY THAT ONLY REGISTERED SEDORIC USERS WILL BE ABLE TO BENEFIT AND INCLUDE 'GAMEINIT' FOR 82 TRACK DOUBLE SIDED. ALSO INCLUDED ON THE DISC ARE: TETRISGB, KINGDOM, A SPLIT SCREEN, THE MUSICAL PIECE '12 DAYS OF XMAS' (STANDARD AND COMPILED), MIDNIGHT FEAST, SHOPPER, WORLD MAP, PALLIDA MORS, TILE SWOP, OUM TOWERS, CARD TRICK, CALENDAR, STRESS AND A FEW SURPRISES. NOW WASN'T THAT LITTLE LOT WORTH WAITING FOR!
===

POPCORN

OUR 'MUSO' TELLS ME THAT HE HAS PUT THE TUNE 'POPCORN' ONTO THE ATMOS VIA 'SONIX'. I AWAIT A COPY TO COMPARE IT WITH THAT YEARS OLD VERSION BY TREVOR SHAW,WHICH WAS INPUT VIA 'COMPOSER'.
  STEVE CAME ACROSS ONE OF THE DOCUMENTED BUGS IN 'SONIX',WHICH LED TO HIS DATA BEING LOST AT ONE STAGE. JONATHAN BRISTOW WAS UNABLE TO GET TO GRIPS WITH SEDORIC'S ERROR TRAPPING ROUTINES. SUBSEQUENTLY IF YOU TRY AND SAVE YOUR COMPOSITION ON A DISC THAT IS WRITE-PROTECTED,THEN YOU RETURN TO BASIC AND LOSE YOUR WORK,WHEREAS AN ERROR MESSAGE SHOULD BE DISPLAYED AND YOU SHOULD BE INVITED TO TRY AGAIN.
  I WILL PERSONALLY TRY AND SORT IT OUT.
  WHEN (AND ONLY WHEN) THIS IS DONE, I WILL SEND A FULL COPY TO THE C.E.O.
  I CAN'T BEAR THE C.E.O HAVING ANOTHER GO AT J.B!
==

THE 'GEOFF PHILIPS ' BOOK

AS WE NEAR THE END OF JANUARY, I HAVE HAD ONLY TWO REQUESTS FOR 'THE' BOOK.
  IF I HAVE NOT HAD ANY MORE LETTERS IN THE NEXT FEW DAYS THEN IT WILL BE GOING TO THE NORTH EAST OR THE SOUTH EAST.
  TO BE HONEST, I HAD EXPECTED MORE RESPONSE,ESPECIALLY AS MANY OF YOU HAD WRITTEN IN THE PAST ASKING IF I COULD GET IT FOR YOU. DON'T BOTHER AGAIN!
==

OUM CASSETTE NO. 1

THE CASSETTE IS NOW AVAILABLE DIRECT FROM BRIAN KIDD. PRICE INCLUDING POSTAGE IS 2 POUNDS,WITH 50 PENCE GOING TO O.U.M FUNDS. DETAILS OF THE TAPE WERE GIVEN IN THE DECEMBER O.U.M
==

AN APOLOGY

THERE WAS IN FACT ONE ENTRY TO OUR COMPETITION (TOP 3). ALTHOUGH NOT QUITE QUITE CORRECT, I THINK THE ENTRANT SHOULD STILL RECIEVE A PRIZE. WELL DONE STEVE MARSHALL.

==

PRICE INCREASE

  DUE TO EVER INCREASING COSTS IT WILL BE NECESSARY TO INCREASE SUBSCRIPTION RATES. FULL DETAILS IN THE NEXT ISSUE.
  TO BEAT THE PRICE RISE,WHY NOT RE-SUBSCRIBE EARLY. YOUR EXPIRY DATE IS SHOWN ON THE STICKER ON THE ENVELOPE THAT THIS TOME CAME IN!