BETA

Volume II No 2

(Issue 12 May'83)

- Softreview
- Competition results
- Mactor Part I
- Letters

. . . . . and lots more!

At the moment there is a flood of new books coming onto the market, a good percentage of them published by Granada. One in particular, Introducing The BBC Micro by Ian Sinclair, refers to LASERBUG in its preface. Ian, who has written a little for us in the past, opens his book by talking about the survey we carried out last year which showed that out of all the people buying BBC Micros, 34% (just over one third) had absolutely no experience. When you think about it, that is an amazing figure. A third of BBC Micro owners were people that decided to take the leap into the world of microcomputing by spending £400 (as few model A's are sold any more). That might be good for Acorn but is it really wise? I agree without a doubt that the BBC Micro is the best home computer for under £500 and that it is an excellent machine for learning on for things like its structured BASIC and good error messages, etc. but is it really a good thing to spend £400 to see if you like a hobby? Computing is one of the fastest growing hobbies in the country but are people really so convinced that they'll take to it all just like that. If a person had any sense (and no previous experience) they should see if they like the idea of computing by attending a meeting of a local user group. The number of people selling second hand computers must be proof that it's not everyones cup of tea. Watching something such as the Computer Programme makes it all seem very easy – it isn't. We do publish a list of local user groups but the best place to find details of your local group is to contact the ACC c/o Rupert Steele, St. John's College, Oxford, OX1 3JP.

After putting you all off buying just like that (although most of you already have machines) I should do something to restore you all. It is a common misconception that "computers are for youngsters". Many adults will not even touch a computer. The children of today are growing up with TV games, electronic hand held games, etc. – a computer holds no mystery to them. The older ones among us were simply not used to anything of that sort. It is true that children have a much better learning capacity than adults – all that really means is that it will take an adult that much longer to learn something, not that they can't. The main problems adults have is their hesitance at actually using the machine. Once this is overcome, learning is no problem. The whole point of all this basically is to prove what I have just said needn't be completely true – not for everybody. Tom Measures of Lancaster Road, Notts took an early retirement. At the time of retiring he bought a BBC Micro – without knowing anything about computers. Today he is getting quite a dab hand at programming them. All it took was a little courage at first and of course patience. Once this is obtained, the rest if easy. Shouldn't the rest of us take a lesson from Tom? Well done Tom for taking the plunge.

OK, enough philosophising – lets come back down to earth. At last after many impatient calls to me (from members) and from me (to Acorn) we have our first delivery of 1.2 ROMs. Thanks to Acorn for supplying them (eventually) and I hope you are all pleased with the chips. There are two articles in this issue which should allow you to get the most out of them and a lot more out of your computer in general.

At the moment we are trying to arrange a super new prize competition – we'll print full details as soon as things are finalised. If we can sort everything out 12 people are going to be very happy over the coming year . . .

Paul Barbour

news

New Micro Magazine Launched

Yet another magazine for the BBC Micro has come onto the market. Building on their success of Personal Software and the two issues full of BBC Micro Programs, Argus Specialist Publications have produced A & B Computing. It is a bi-monthly magazine costing £1.85. It follows a different format to the usual magazines – more like that of Personal Software than Acorn User. The very first issue contained a reasonable balance of material. It is available from your local newsagent now.

Micro Market Survey

A new micro survey reveals some interesting facts – one house in 20 has a microcomputer and within the next two years that figure is expected to double! Young males are the probable users of the computers and 45% of them are less than 18 years old. 45% of computers were bought as gifts and 25% of the time spent using micros is for educational purposes. To compare this with the BBC Micro scene (as from our 1982 survey) 12% of BBC Micro Users are under 18 and 37% of the time the computer will be used for education. Out of interest we will be launching a new survey in the August LASERBUG.

Business Beeb

Who says that the BBC Micro is not a good machine for business. GSL now produce a Winchester Interface to allow the BBC Model B to be connected up to four 85Mb Winchesters giving it a total capacity of 340Mb storage. That is approximately 348 160 000 000 bytes!!! You can get more details from GSL, 2 North Way, Andover, Hants, SP10 5AZ.

How Many ROMs?

When you have your DFS fitted plus your wordprocessor you will find your ROM sockets filling up rapidly. So what do you do? The answer comes from Watford Electronics. Your computer has four sockets for paged ROMs – BASIC is one of them. The BBC Micro is in actual fact capable of supporting 16 ROMs – if it had the sockets. The device from WE plugs into one of the spare sockets and leaves you with an extra 13 sockets!!! It fits in quite neatly so we have heard and is undetectable in action. It costs £19.95 and you can get more details from Watford Electronics, 33 Cardiff Road, Watford, Herts. Phone Watford 40588.

Super Beeb

Supposing you could buy a super BBC Micro with speech synthesiser and VIEW wordprocessor plus the Disk and Econet interfaces as standard. Would you be interested? Well, that is the machine that will be sold over in the States. Apparently, America is such a different kind of market that this kind of thing is needed to make it sell. So, if you have $995 and fancy a trip over to America . . . (Just a warning – if you do go over to the States do not ever be tempted to buy the BBC Micro or indeed any other microcomputer, TV Game, etc. When you bring them back to England you will find quite simply that because of the different power supply and UHF modulator they will not work.)

Dull Amber

You may remember a number of issues ago we reviewed the Amber 2400 Printer. Well, since then Amber controls has gone into

contents

liquidation. We have had a number of enquiries asking where on earth people are going to get the ribbons? This would mean that unless another supply can be found, anyone who bought an Amber printer might as well throw it away! Any suggestions???

Biting The Hand . . .

Our review of Micronet 800 last month did a great deal for popularity – for Viewfax 258, a rival of Micronet!!! Many people, did not know of the existence of Viewfax until they read our review. Don't forget, for an alternative Micronet microcomputer news service enter *258#

Micronet Is Top Of The Pops

Prestel has its very own Top of the Pops – the most popular database on Prestel. For as long as people can remember Viewtel 202 has been number one – until now. Micronet 800 is now the most popular database on the whole of Prestel with 500 000 "accesses" in its first month!

LASERBUGs Own Top Of The Pops

Not to be out done, LASERBUG in conjunction with Micro Management is going to publish its own top ten best selling programs. The charts will be based on monthly sales with the first chart (May figures) appearing in the next issue.

Show Time Again

Next month it's the IPC Computer Fair at Earls Court. LASERBUG will be there of course and we hope to see as many of you as possible. The show is open between the 16th and 19th June (1pm–6pm Thursday, 10am–6pm Friday and Saturday, 10am–5pm Sunday) the entrance fee is £3.00 for adults, £2.00 for children and OAP's. Don't forget to visit Club Avenue . . .

Hands Off BBC

Obviously supporting a **BBC** Computer, many companies feature the BBC logo in their adverts – no more. The BBC has written to numerous advertisers and computing magazines warning that the BBC logo is the property of the BBC and they are not prepared to stand misuse of its symbol. The BBC story is that the use of the BBC logo implies that the products have been licensed by the corporation. This was also extended to the use of the BBC Owl – the symbol of the Computer Programme – which was used by Micronet for the telesoftware library. Acorn User took the step of altering every occurrence of the logo in the May edition themselves. To use the BBC logo in an advertisement, computer magazine or anywhere you first must get permission from BBC Enterprises.

Electron

Acorn User last month carried pictures of the Acorn Electron (an ugly looking thing). It will have 32k RAM as standard and single keyword entry. It has a proper keyboard and the power supply comes from an included adaptor. There is no MODE7 and will work very slowly in the higher MODEs (0–3). In other modes, the computer will work about two-thirds of the speed of your Beeb. It will be built in Singapore and "full scale production" will start in June which infers an August/September launch date. The first public glimpse of the machine is expected at the Acorn User exhibition.

Special Offers Expand

As you probably noticed the Special Offers section of LASERBUG has expanded – we now sell covers for the Epson printer, Microvitec Monitor and Single Disk Drive (and of course the computer) in PVC. We will, contrary to what we said last month, still be selling the Polyester Cotton covers. Also, a binder has been introduced to protect your valuable magazines.

But . . .

But even though we have expanded the special offers last month there was a mix up. Instead of sending out the Polyester/Cotton covers to people who had ordered them, some people were sent PVC covers instead. Now we would point out that the PVC Covers are 50p more expensive than the Polyester ones but even so, many people find them inferior. If you were sent a PVC Cover by mistake simply return the cover and we will replace it with a Polyester/Cotton one.

More Problems With Disks

The main problem with disks (the Acorn ones anyway) is that nobody can get them! Apart from that it seems that you now may have trouble getting any other companies disk drives to connect up to you Beeb. Why? It appears that there is a shortage of Disk Interfaces (groan!). When I first heard about this story I found it quite funny because at the time I was having a disk interface fitted into my machine but it seems that it's true. The problem is only partly due to Acorn – mainly because they choose an out of date chip, the Intel 8271. There is a worldwide shortage of these chips and it is uncertain when more will be available. There is even chances that Intel might stop producing the Floppy Disk Controller chip altogether – meaning that Acorn would have to have some specially made for them and thus putting prices up. It seems that until the summer it is unlikely that people will be able to have disk interfaces fitted! It makes me glad I got mine when I did!!!

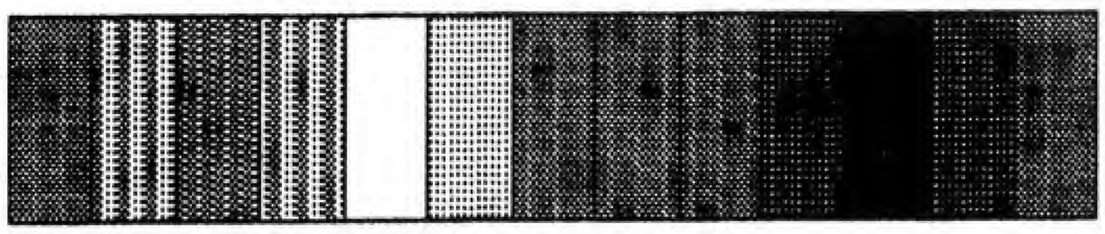
Micronet Soap Opera

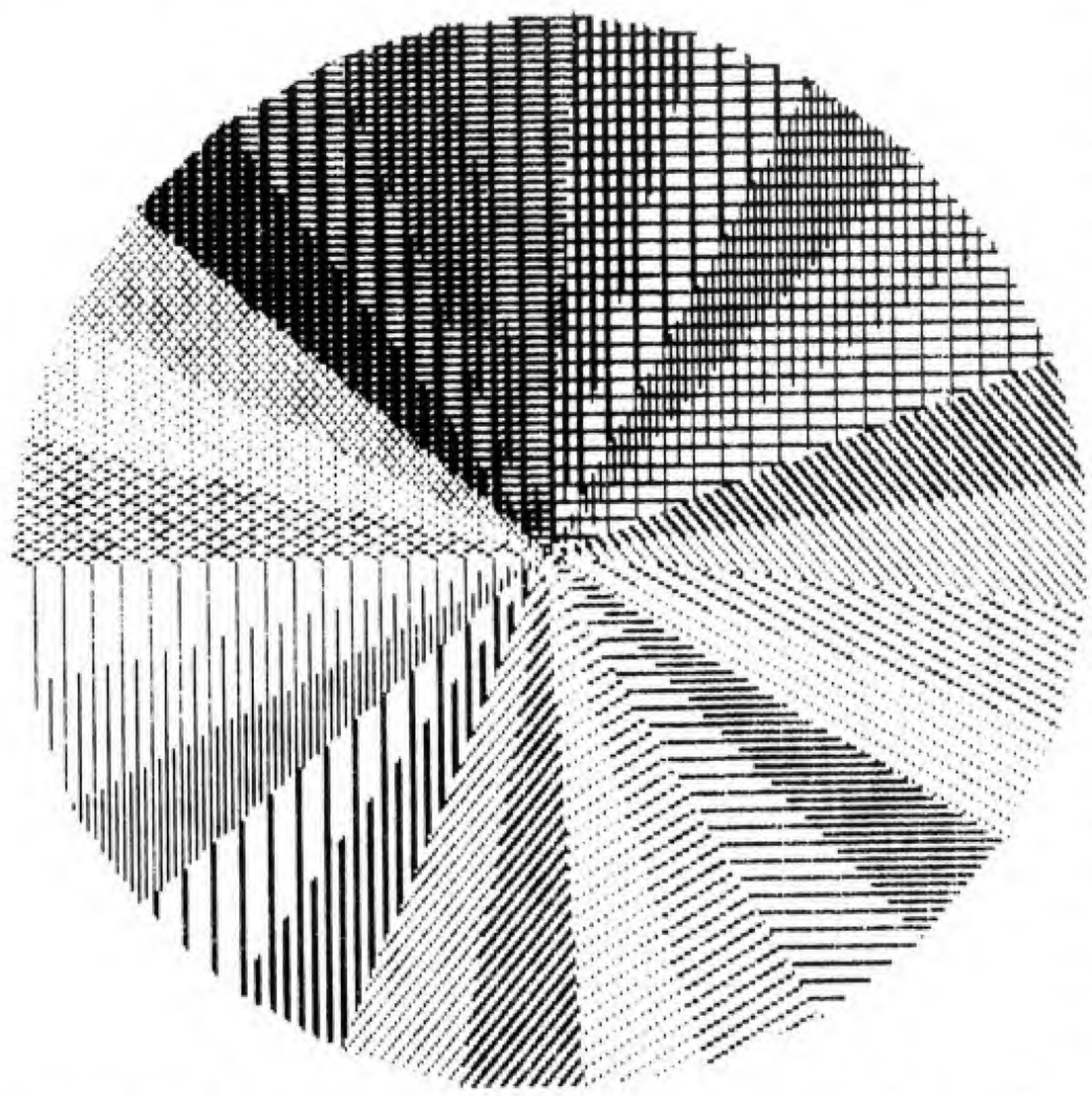
There is currently a lot going on with regards to Micronet/Soft Machinery. Soft Machinery are the company that produces the software to enable the BBC Micro to communicate with Prestel. Originally, Soft Machinery supplied Micronet with version 3.2 of the Telesoft program but in January gave them the much improved 4.0. Despite this, Micronet is supplying subscribers with version 3.2. The differences between the two versions is quite great. The main one is with regards to telesoftware – the format used on Prestel is changed every now and again. The new version at the moment just involves problems with file names – i.e. downloading GaIRACE instead of RACE. By July a new format will be used which will not be compatible at all with the old version. Version 4.0 is capable of handling both – version 3.2 can only deal with the old version. Version 4.0 also allows Frame Tagging, automatic saving, printing of frames for printers with block graphics and a good number of other features – making it 100% better than the original version. Bob Clark of Soft Machinery sent a message to all the Micronet members explaining about the situation on April 1st (although it wasn't an April fool). He urged Micronet members to pester Micronet for them to release the new version. Micronet in turn got upset and called Bob Clarks message just a "quick advert". Viewfax who don't miss a chance of getting at Micronet in turn has carried several stories about the problems. Micronets story is that they are making sure the program is bug free and that they will be releasing it soon as a downloadable program. Micronet will be doing this "pretty soon" (note: they said that on the 7th April, lets see how long "pretty soon" turns out to be). Bob Clark supplied me with a copy of version 4.0 and I can't find any bugs with it. So, even when Micronet do supply us all with version 4.0, how on earth are they going to send us the 46 page A4 manual that comes with it!!! If Micronet have had the software since January I personally think that they have had plenty of time to get it checked and out.

oddspot

```
L.    10 MODE2
      20 VDU23,1,0;0;0;0;
      30 FORX%=0TO360STEP4
      40   S%=SINRAD(X%)*400+640
      50   C%=COSRAD(X%)*400+512
      60   GCOL0,1
      70   MOVE0,0:DRAWS%,C%
      80   GCOL0,2
      90   MOVE1280,0:DRAWS%,C%
     100   GCOL0,3
     110   MOVE1280,1024:DRAWS%,C%
     120   GCOL0,4
     130   MOVE0,1024:DRAWS%,C%
     140   NEXT
     150 REPEATUNTIL0
```

GRAPHICS PROGRAM: Extended Colour Fill Graphics
REQUIRES: 16k
SUPPLIER: Gaelsett Software, 44 Exeter Close, Stevenage, Herts., SG1 4PW
PRICE: £10.00
DESCRIPTION: This program basically does exactly what it says – extends the colour fill (plotting filled in triangle) graphics to produce shades, etc. The advertising literature talks about 6561 colours from 16 colour MODE2. Well, I don't know about that but the package really is quite impressive. The actual routine is on the second side of the tape and is entered into your machine by using *RUN. On OS 0.1 this provided no problem but on the 1.2 when the program had finished loading it gave the message "Syntax Error". We were unable to find the cause but despite the message, the program still worked on the 1.2. Basically it extends the VDU19/PLOT81–85 to enable some quite impressive shading and colouring effects to be produced:





Remember the above pictures are just screen dumps done with an Epson printer and do not show the full capabilities of the program. The tape includes full details of how to use the commands and some excellent demonstration programs utilising these new facilities. The program was good and is probably well worth buying if you are particularly interested in graphics but £10? It seems a little bit overpriced for most people I fear.
USE OF GRAPHICS: ★★★★★
USEFULNESS: ★★★
LOADING PROBLEMS: No
VALUE FOR MONEY: ★★★

--o0o--

MISCELLANEOUS PROGRAM: Life Plus
REQUIRES: 16k
SUPPLIER: Mayday Software, 181 Portland Crescent, Stanmore, Middx., HA7 1LR
PRICE: £4.49
DESCRIPTION: If you don't know about the game of Life then I really don't think I have enough room here to explain about it. If you do then let me talk about this specific version. The game runs (obviously) in MODE7 and allows you to pick what colours will be used for both the foreground and background. You can determine what structure the program uses by using the cursor keys. Other features to allow you to build up the screen include turtle graphics and triangle/rectangle facilities. When up and running the computer can handle up to 10 frames a second. It has a hold facility to enable you to examine screens more closely and includes an instruction booklet giving some interesting shapes to try out. A good buy overall.
PRESENTATION: ★★★
SPEED: ★★★★★
LOADING PROBLEMS: No
VALUE FOR MONEY: ★★★★

ARCADE GAME PROGRAM: Secta Invaders
REQUIREMENTS: 32k (Joystick optional)
SUPPLIER: Secta Software, 187-195 Broad Street, Coventry, CV6 5BN
PRICE: £5.95
DESCRIPTION OF PROGRAM: Following on from the last review, if there is one games program that's bound to appear for a computer then it's Space Invaders. Surprisingly, the best Space Invaders I've seen to date is not from Acornsoft but from this company, Secta Software. There's not a lot anyone can see about the actual game itself suffice it to say that this version is excellent. The menu you get is extremely comprehensive and enables you to do everything from look at the instructions to turning the sound off (a handy facility) to SAVEing your hi-scores on tape to look at them again later. Well worth £6.
PRESENTATION: ★★★★★
USE OF GRAPHICS: ★★★★
ADDICTIVE QUALITY: ★★★
LOADING PROBLEMS: No
VALUE FOR MONEY: ★★★★

--o0o--

ARCADE GAME PROGRAM: Galactic Intruder
REQUIREMENTS: 32k
SUPPLIER: Software for All, 72 North Street, Romford, Essex
PRICE: £6.95
DESCRIPTION OF PROGRAM: Hmmm. This program is certainly colourful, noisy and fast but apart from that you can't say a lot about it! There is very little skill involved and the whole game seems rather pointless and confusing. You can *either* move your spaceship *or* fire at the super fast aliens – not both at the same time. The game may look good from a distance but trying it yourself it is deathly boring. Not worth buying at all.
PRESENTATION: ★★★
USE OF GRAPHICS: ★★★★
ADDICTIVE QUALITY: ★
LOADING PROBLEMS: No
VALUE FOR MONEY: ★

--o0o--

MISCELLANEOUS PROGRAM: Microtype
REQUIREMENTS: 32k
SUPPLIER: Kansas City Systems, Unit 3, Sutton Springs Wood, Chesterfield, S44 5XF
PRICE: £12.50
DESCRIPTION OF PROGRAM: Just as we were about to do this review we received a letter from Ralph Erskine. The majority of this review is made up of excerpts from his letter – I'm afraid I bought the Kansas City tape without seeing it reviewed which was foolish. However, I did so partly on the basis of your remark that they were a firm with a good reputation (they are and their program Galactic Firebirds is very good – Ed.) added to which I didn't think they would dare charge 25% more than Acornsoft unless their produce was better. As a typing teacher, Kansas compares unfavourably with another typing tutor I bought. Their advert claims that at first one is given the home keys to practice (ASD . . .)then in the first 100 impressions CVBN appear. In the next 100 most of the keyboard seems to be thrown in. Ralph's letter goes on but he concluded his letter with "I say no more except that I doubt if Microtype does anything good for Kansas's reputation and that at £12.50 it is very overpriced. I feel too strongly about it to review it myself, especially

as I think their advert may make incorrect claims for its contents!". Need we go to the effort of reviewing it?
PRESENTATION: ★★★
USEFULNESS: ★
LOADING PROBLEMS: No
VALUE FOR MONEY: ★

–o0o–

ARCADE GAME PROGRAM: Stratobomber
REQUIREMENTS: 32k
SUPPLIER: IJK Software, 9 King Street, Blackpool, Lancs
PRICE: £7.50
DESCRIPTION OF PROGRAM: I will describe this game the best I can. Basically you are a Spaceship in orbit of your planet. In a closer orbit than you is a spaceship and above you are three alien fighters. The alien spaceship is trying to build up enough of a charge to fire its beam and destroy your planet. Your mission is to bomb the spaceship until you get through to its power core. One bomb on the power core and the enemy is finished. The only problem is the three fighters – if they managed to get too close to the surface then you explode! You have now laser guns for the fighters – just a repulsor beam! If the game sounds interesting that's because it is! Well worth investing in.
PRESENTATION: ★★★
USE OF GRAPHICS: ★★★★
ADDICTIVE QUALITY: ★★★
LOADING PROBLEMS: Yes (Second tape recorder had to be used)
VALUE FOR MONEY: ★★★★

–o0o–

GAME PROGRAM: Sliding Block Puzzles
REQUIREMENTS: 32k
SUPPLIER: Acornsoft, 4a Market Hill, Cambridge, CB2 3NJ
PRICE: £9.95
DESCRIPTION OF PROGRAM: Before the Rubik's Cube there was these things – the sliding block puzzles. You must all know what I mean so I won't bore you with a long description. The Acornsoft version of these is very good. The actual picture you use can be selected from one of 6 different patterns and if you are ever stuck just press S and Y and the computer solves the problem for you. Good for filling up the odd spare hour here and there!
PRESENTATION: ★★★
USE OF GRAPHICS: ★★★★
ADDICTIVE QUALITY: ★★★★
RESPONSE SPEED: ★★★★
LOADING PROBLEMS: No
VALUE FOR MONEY: ★★★

–o0o–

ADVENTURE PROGRAM: Firienwood
REQUIREMENTS: 32k
SUPPLIER: MP Software & Services, 165 Spital Road, Bromborough, Merseyside, L62 2AE
PRICE: ???
DESCRIPTION OF PROGRAM: I seem to be lost for words about a number of programs this month, don't I. What can I say about adventures that hasn't been said. This one uses colour in teletext mode which livens things up a bit and plays a good game from what I've seen. Another one to add to your collection.
PRESENTATION: ★★
COMPLEXITY: ★★★
RESPONSE SPEED: ★★★★★
LOADING PROBLEMS: No
VALUE FOR MONEY: ?????

–o0o–

We would like to thank Gaelsett Software, Mayday Software, Secta Software, Software for All, Kansas City Systems, IJK Software, Acornsoft and MP Software & Services for supplying us with review tapes.

For those of you thinking that Mactor is a game, you had better find another article to read now. Mactor is in fact a "**mac**hine code tu**tor**" or in other words a simulation of a low level language like BBC BASICs built in assembler. So, if BBC BASIC has a built in low level language why use a program like Mactor? For those interested in learning a low level language Mactor provides a good general starting point. The low level language on the BBC Micro is 6502 assembler – there are many other types i.e. Z80 assembler. Mactor is neither one nor the other and thus is a sort of go-between.

Mactor is rather more friendly than a normal low level language. It has a main menu from which you can use its many functions. The main features of mactor are:

* 19 Commands including access to the VDU drivers
* 10 Memories
* Directory of commands in program giving syntax and meaning
* 10 Labels
* Printer listings
* Full use of function keys
* Compiler/Decompiler
* Full editing including insert, delete and change
* Saveing/Loading of Mactor programs
* 1024 program step memory

To help you, when you first start to use the program press SHIFT–f0 and you will get a list of the commands. If you have a printer these can be printed out for reference. Pressing SHIFT–f1 will take you into the program mode allowing you to enter your Mactor program. Each command is three letters long which when applicable is followed by a number. Memory and labels must be in the range 0–9, numbers should be in the range 0–255. A number over 999 will generate a syntax error message. When you have finished the program you should press RETURN. There is no way in the program mode of altering lines once they are entered. After entering the program you can list and check it using SHIFT–f2. Again you can have a printer listing if required.

If you find an error in the program you should enter the edit mode by pressing SHIFT–f3 which will allow you to insert, delete and change program steps. Once you are sure that the program is correct you must compile it. This means that the computer changes all the words you have entered into numbers to enable the program to be run correctly. Pressing SHIFT–f5 will enable you to run the program.

Should you wish to alter or list the program after it has been compiled you must decompile it using f6. You can LOAD or SAVE Mactor programs by pressing SHIFT–f7/8.

If a serious error should occur (including pressing the ESCAPE key) the memory will become corrupted and you will have to re-boot the system.

One thing that should be noted is compiling/decompiling programs. Certain modes only work depending on the state of the program:

PROGRAM MODE – enters a decompiled program and wipes out any previously compiled programs
LIST MODE – only operates on decompiled programs
EDIT MODE – only operates on decompiled programs
COMPILE MODE – only operates on decompiled programs
RUN MODE – only operates on compiled programs
DECOMPILE MODE – only operates on compiled programs
SAVE MODE – only operates on compiled programs
LOAD MODE – only loads a compiled program overwriting any previously decompiled programs.

In general the language is quite simple and we'll say more about it next month. The complete Mactor program takes up 12k memory so we will be printing it in two stages. The first half is below, the second part follows next month.

```
L.
   10 IFPAGE<>&E00THENPRINT"No Room - Please Reload At
&E00":END
   20 ONERRORPROChelp
   30 :
   40 REM  MACHINE CODE TUTOR
   50 REM       "MACTOR"
```

```
 60 REM    by Paul Barbour
 70 :
 80 REM        19/4/83
 90 :
100 REM      Version 1.0
110 :
120 REM Takes up 12.30k memory
130 :
140 REM  Requires 32k,  PAGE
150 REM   to be set to &E00
160 REM   and a series 1 OS
170 :
180 REM   Written on OS 1.2
190 :
200 REM   (c) LASERBUG 1983
210 :
220    :::::
230 :
240 *TV255
250 *OPT1,1
260 *OPT2,1
270 MODE7
280 VDU23,1,0;0;0;0;
290 DIMinstruct$(1024),loc%(1024)
300 DIMcompile%(1024)
310 DIMmem%(9),lbl%(9)
320 compile%=FALSE:acc%=0
330 :
340 PROCteletext
350 PROCtop
360 PROCmactor
370 END
380 :
390    :::::
400 :
410 DEFPROCteletext
420 bd$=" "+CHR$157
430 rd$=CHR$129
440 bl$=CHR$132
450 cy$=CHR$134
460 gr$=CHR$130
470 yl$=CHR$131
480 st$=yl$+CHR$157+bl$
490 er$=bl$+CHR$157+yl$
500 in$=rd$+CHR$157+yl$
510 mg$=CHR$133
520 ENDPROC
530 :
540 DEFPROCtop
550 CLS
560 PRINTbd$'bd$;rd$;"   MACTOR";bl$"the Machine Code
Tutor"'bd$
570 ENDPROC
580 :
590 DEFPROCmactor
600 :
610 PROCmenu
620 :
630 GOTO610
640 :
650 DEFPROCmenu
660 PROCtop
670 PRINTst$;"         STATUS: Menu Mode"
680 PRINT'cy$;TAB(7)"SHIFT-f0";gr$;"KEYWORD LIST MODE
"
690 PRINTcy$;TAB(9)"SHIFT-f1";gr$;"PROGRAM MODE"
700 PRINTcy$;TAB(11)"SHIFT-f2";gr$;"LIST MODE"
710 PRINTcy$;TAB(11)"SHIFT-f3";gr$;"EDIT MODE"
720 PRINTcy$;TAB(9)"SHIFT-f4";gr$;"COMPILE MODE"
730 PRINTcy$;TAB(11)"SHIFT-f5";gr$;"RUN MODE"
740 PRINTcy$;TAB(8)"SHIFT-f6";gr$;"DECOMPILE MODE"
750 PRINTcy$;TAB(10)"SHIFT-f7";gr$;"SAVE MODE"
760 PRINTcy$;TAB(10)"SHIFT-f8";gr$;"LOAD MODE"
770 PRINTcy$;TAB(8)"CTRL-f9";gr$;"RETURN TO BASIC"
780 PRINTcy$;TAB(8)"CTRL-TAB";gr$;"REBOOT SYSTEM"
790 REPEATfn$=GET$
800   UNTILfn$=CHR$128ORfn$=CHR$129ORfn$=CHR$130ORfn$
=CHR$131ORfn$=CHR$132ORfn$=CHR$133ORfn$=CHR$134ORfn$=CH
R$135ORfn$=CHR$136ORfn$=CHR$153OR(fn$=CHR$9ANDINKEY(-2)
)
810 IFfn$=CHR$153THENCLS:PRINT'"BBC Computer 32K"'"B
ASIC"':END
820 IFfn$=CHR$9THENCLEAR:RUN
830 fn%=ASC(fn$)-127
840 ONfn%GOTO850,860,870,880,890,900,910,920,930
850 PROCkeyword_list:ENDPROC
860 PROCprogram:ENDPROC
870 PROClist:ENDPROC
880 PROCedit:ENDPROC
890 PROCcompile:ENDPROC
```

```
 900 PROCrun:ENDPROC
 910 PROCdecompile:ENDPROC
 920 PROCsave:ENDPROC
 930 PROCload:ENDPROC
 940 :
 950 DEFPROCkeyword_list
 960 PROCtop
 970 PRINTst$;"     STATUS: Keyword List Mode"
 980 PRINT'er$;"      MACTOR: Printer On (Y/N)"
 990 REPEATpr$=GET$:UNTILpr$="Y"ORpr$="N"
1000 PROCtop
1010 PRINTst$;"     STATUS: Keyword List Mode"
1020 IFpr$="Y"THENVDU2
1030 PRINTyl$"    <num> 0-255 <mem> 0-9 <lbl> 0-9"
1040 PRINT'cy$"LAN <num>"gr$"Load Acc With Number <num
>"
1050 PRINTcy$"LAM <mem>"gr$"Load Acc With Memory <mem>
"
1060 PRINTcy$"SAM <mem>"gr$"Store Acc In Memory <mem>"
1070 PRINTcy$"AMA <mem>"gr$"Add Memory <mem> to Acc"
1080 PRINTcy$"SMA <mem>"gr$"Subtract Memory <mem> from
"'gr$"Acc"
1090 PRINTcy$"AND <mem>"gr$"Memory <mem> AND Acc"
1100 PRINTcy$"ORA <mem>"gr$"Memory <mem> OR Acc"
1110 PRINTcy$"NOT "gr$"NOT Acc"
1120 PRINTcy$"INC <mem>"gr$"Increment Memory <mem> by
1"
1130 PRINTcy$"DEC <mem>"gr$"Decrement Memory <mem> by
1"
1140 PRINTcy$"SKG <mem>"gr$"Skip Next Step If Memory"'
gr$"<mem> Is Greater Than Acc"
1150 PRINTcy$"SKL <mem>"gr$"Skip Next Step If Memory"'
gr$"<mem> Is Less Than Acc"
1160 PRINTcy$"SKZ <mem>"gr$"Skip Next Step If Memory"'
gr$"<mem> Is Zero"
1170 VDU3
1180 PRINTer$;"MACTOR: Screen Full - Press SPACE"
1190 REPEATUNTILGET$=" "
1200 PROCtop
1210 PRINTst$;"     STATUS: Keyword List Mode"
1220 PRINTyl$"    <num> 0-255 <mem> 0-9 <lbl> 0-9"'
1230 IFpr$="Y"THENVDU2
1240 PRINTcy$"LAK"gr$"Load Acc With Next Keyboard"'gr$
"Character Pressed"
1250 PRINTcy$"LSA"gr$"Load Screen With ASCII Character
"'gr$"In Acc"
1260 PRINTcy$"PRA"gr$"Print Number Held In Acc"
1270 PRINTcy$"LBL <lbl>"gr$"Define Label <lbl>"
1280 PRINTcy$"JMP <lbl>"gr$"Jump To Label <lbl>"
1290 PRINTcy$"END"gr$"End Mactor Program And Return To
"'gr$"Menu"
1300 VDU3
1310 PRINT'er$;"MACTOR: End Of Mode - Press SPACE"
1320 REPEATUNTILGET$=" "
1330 ENDPROC
1340 :
1350 DEFPROCprogram
1360 pointer%=0
1370 compile%=FALSE
1380 PROCtop
1390 PRINTst$;"        STATUS: Program Mode"
1400 PRINTTAB(0,4);in$;"MEMORY FREE: ";1024-pointer%;"
bytes"
1410 PRINTcy$;pointer%;CHR$135;" - ";gr$;
1420 VDU23,1,1;0;0;0;
1430 INPUT""instruction$
1440 VDU23,1,0;0;0;0;
1450 IFinstruction$=""THENENDPROC
1460 instruct$(pointer%)=LEFT$(instruction$,3)
1470 loc%(pointer%)=VAL(RIGHT$(instruction$,LEN(instru
ction$)-3))
1480 PROCerror
1490 IFerr%<>0THENPRINTer$;"ERROR: Syntax error in las
t line -"'er$;"        Please re-enter...":GOTO1410
1500 pointer%=pointer%+1
1510 IFpointer%>1024THENENDPROC
1520 PROCtop
1530 PRINTst$;"        STATUS: Program Mode"
1540 PRINTTAB(0,4);in$;"MEMORY FREE: ";1024-pointer%;"
bytes"
1550 FORprev%=10TO1STEP-1
1560   IFpointer%-prev%<0THEN1580
1570   PRINTcy$;pointer%-prev%;CHR$135;" - ";gr$;instr
uct$(pointer%-prev%);" ";loc%(pointer%-prev%)
1580   NEXT
1590 GOTO1410
1600 :
1610 DEFPROClist
1620 PROCtop
1630 PRINTst$;"        STATUS: List Mode"
1640 IFcompile%=TRUE THENPRINT'er$;"      ERROR: Progr
am Compiled"'er$;"          Press SPACE":REPEATUNTILG
ET$=" ":ENDPROC
1650 PRINT'er$;"      MACTOR: Printer On (Y/N)"
1660 REPEATpr$=GET$:UNTILpr$="Y"ORpr$="N"
1670 pointer%=0
1680 PROCtop
1690 PRINTst$;"        STATUS: List Mode"
1700 count%=1
1710 REPEAT
1720   IFpr$="Y"THENVDU2
1730   IFinstruct$(pointer%)=""THEN1750
1740   PRINTcy$;pointer%;" - ";gr$;instruct$(pointer%)
;" ";loc%(pointer%)
1750   pointer%=pointer%+1
1760   count%=count%+1
1770   UNTILcount%=20
1780 VDU3
1790 IFinstruct$(pointer%)=""THENPRINT'er$;"MACTOR: En
d Of Mode - Press SPACE":REPEATUNTILGET$=" ":ENDPROC:EL
SEPRINTer$;"MACTOR: Screen Full - Press SPACE":REPEATUN
TILGET$=" ":GOTO1680
1800 :
1810 DEFPROCerror
1820 err%=0
1830 IFLEN(instruction$)>7THENerr%=err%+1
1840 er2%=0
1850 RESTORE1900
1860 REPEAT
1870   READcommand$
```

```
1880   IFcommand$=instruct$(pointer%)THENer2%=er2%+1
1890   UNTILcommand$="E"
1900 DATALAN,LAM,SAM,AMA,SMA,AND,ORA,NOT,INC,DEC,SKG,S
KL,SKZ,LAK,LSA,LBL,JMP,END,PRA,E
1910 IFer2%=0THENerr%=err%+1
1920 IF(instruct$(pointer%)="LAM"ORinstruct$(pointer%)
="SAM"ORinstruct$(pointer%)="AMA"ORinstruct$(pointer%)=
"SMA"ORinstruct$(pointer%)="AND"ORinstruct$(pointer%)="
ORA"ORinstruct$(pointer%)="NOT")ANDloc%(pointer%)>9THEN
err%=err%+1
1930 IF(instruct$(pointer%)="INC"ORinstruct$(pointer%)
="DEC"ORinstruct$(pointer%)="SKG"ORinstruct$(pointer%)=
"SKL"ORinstruct$(pointer%)="SKZ"ORinstruct$(pointer%)="
LBL"ORinstruct$(pointer%)="JMP")ANDloc%(pointer%)>9THEN
err%=err%+1
1940 ENDPROC
1950 :
1960 DEFPROCedit
1970 PROCtop
1980 PRINTst$;"          STATUS: Edit Mode"
1990 IFcompile%=TRUE THENPRINT'er$;"      ERROR: Progr
am Compiled"'er$;"          Press SPACE":REPEATUNTILG
ET$=" ":ENDPROC
```

consumer spot

This is a new section hinted at last week. What we aim to do here is to sort out your problems – not your programming ones, write to Queryspot for that – but any trouble you are having with an Acorn Dealer, Software House or indeed anybody that has something to do with the BBC Micro. It doesn't matter what your problem is, let us know and we'll have a go at sorting it out for you. We'll have our best shot at solving the problem. Firstly before we start on this months problem we would like to thank Microage Electronics for sorting out a query with regards to them. The problem was suitably replied to so we won't be going into details here. What we will be doing though is printing two moans against Bug-Byte Software.

The first complaint is with regards to Mr. John Shaw of Glasgow. To refresh your memory a little bit you can have a look at page 4 of Issue 8 (Jan'83). Back in August Mr. Shaw ordered Bug-Byte Chess. The money was paid in during August even though Mr. Shaw didn't receive his first copy of the program till September (not a good start!). When Mr. Shaw received the program it wouldn't work properly and so he sent it back. Since then Mr. Shaw has sent back the program 5 times and each time the program either would not load or gave a Bad Program error message. One time he was sent another program by mistake and to Mr. Shaw's disgust that would not load either. To date as far as we know Mr. Shaw still hasn't received a working copy of the program. To our minds Mr. Shaws problem was simple – firstly when he was getting the Bad Program message this was probably because he was entering CHAIN"" instead of *RUN. Secondly when the program wasn't loading this could be because the inlay tells you to enter *RUN"BEEB-CHESS" when the title is CHESS and so that would obviously stop the program loading. I can think of two reasons why the program doesn't work and I haven't looked at it since October last year – Bug-Byte deal with the program daily and they should know why Mr. Shaw is having problems. Surely a note with one of the replacement programs would have saved a great deal of trouble. Even so we have now answered Mr. Shaws problem and if he gets a fresh copy of the program he should have a better chance of getting it working – the problem now is that he hasn't got a copy. Bug-Byte still haven't replaced the package! We think that (i) in the first place when the replacement program was sent somebody at Bug-Byte should have written Mr. Shaw a note telling him what he was doing wrong. (ii) At this late stage either Bug-Byte should replace Mr. Shaws program a sixth time or give him his money back plus something to cover his postage costs (i.e. sending a tape back FIVE times).

The second complaint we have received, and this is from several people, is regards to the infamous magazine the Beebon. Everybody who subscribed to the Beebon paid £7.50 for 6 issues of the magazine – to date only three magazines have ever been published and those were late in the extreme (even worse than us last year!). Bug-Byte has not stopped trading obviously. They have taken people's money for 6 magazines and only ever produced three. No word was even given as to what happened to the Beebon. A new magazine that was to be called the Beebon (now BBC Micro User) was rumoured to take over from it but that has no connection at all to Bug-Byte. The case in hand is this. Bug-Byte took money for 6 issues of the magazine and only produced three. No effort has ever been made to explain what happened to the magazine and no person has received a refund. All the names of subscribers to the Beebon were held on file and so Bug-byte had no reason not to automatically give everybody concerned a £3.75 refund. They have certainly not made a reasonable effort to contact people. We think that everybody who subscribed to the Beebon should be sent a refund to the amount of £3.75. As all names are held on file this should prove no problem. Bug-Byte have absolutely no defence as far as the Beebon is concerned.

We gave Bug-Byte fair warning of what we were going to publish in a letter to them. The only response we received was a compliment slip saying "please phone us". We did so and managed to speak to the person who sent us the note. They did not have a clue as to what we were talking about, took our phone number and promised to get back to us later that day. To date no response has been received. In the first case Bug-Byte acted unfairly and has still not produced any goods, in the second case Bug-Byte blatantly acted without forethought by not providing goods and failing to refund monies paid. We hope to be able to report in the next issue that Mr. Shaw has finally a working copy of his program and that all subscribers to the Beebon will be receiving a refund. For an allegedly reputable software house Bug-Byte are acting very strangely. We will do our very best to get a satisfactory result.

The Computer Programme – Making The Most Of The Micro
Presented by Ian McNaught Davis
With John Coll
Produced by David Allen
Episode IV – Introducing Graphics

The fourth programme opens by showing a patient with a badly formed jaw discussing a forthcoming operation to correct his problem. The surgeons work out what they are going to do in the operation by looking at an X-Ray and deciding how much to move the patients bones and what he will look like afterwards. It is explained to us how hard it is to judge a 3 dimensional figure as complex as the human facial structure by looking at a 2 dimensional picture. So, how can computers help?

A cast of a head is shown being scanned by a camera and the information being fed into a mini-computer. Slowly the image is built up on the screen. Once the process is finished a complete representation of the head is in the computers memory and it can now be rotated at willl and so look at every possible angle in the face. In the future this will allow surgeons to show a patient what he will look like after an operation before he has even stepped into an operating theatre. All of this shows one of the practical applications of computer graphics.

Back to the studio to look at graphics on the BBC Micro. Mac explains "today, even the humblest personal micro is capable of displaying graphics; in other words pictures . . . even if you look at a single character on the screen it's made up of lots of dots or pixels". Moving over to another computer we find John Coll. John goes on to say that there are two ways the BBC Micro can handle graphics – one as character graphics and the other as high resolution graphics. Firstly – character graphics. John Coll says (incorrectly) that there are 256 characters available on the BBC Micro (that is true but ASCII codes 0–31 are used to control the micro – not for printing as characters as such). Using a pack of cards with the letters on he explains all about the ASCII system and shows it working using the CHR$ command. Using a specially written program they look at some of the pre-defined characters and then go on to define their own character – that of a small man. Using PRINTTAB they print the man on the middle of the screen and develops it to print the man all over the screen in random colours. Defining two more characters they make the man jump up and down on the spot. Taking user defined characters to their limit they run a program that draws a field of flowers and make two bees (made up of characters) buzz around the screen. To show a real use of character graphics (or block graphics to be more precise) the Acornsoft Mode 7 Invaders is shown.

Now onto high resolution graphics. A picture of Bjorn Borg is shown (which you may have seen in Acorn User a few months ago) in differing resolutions – 154 pixels, 320, 616, 1 280, 2 700, 9 200 (about teletext resolution), 20 480, 36 400 (slightly lower resolution than MODEs 2/5), 81 920 (MODEs 1/4), 324 680, 655 536, 1 300 000, 2 620 000 and finally 5 000 000 for a colour picture. Out of interest I have just worked out that to store that 5 000 000 pixel picture in full colour would need about 4½Mb!!!

For another example of high res graphics a circle is drawn on the screen first on a ZX81 using letter Is for each point (which is wrong because the ZX81 is capable of a better resolution than that) then a ZX Spectrum taking up about 10k of memory and finally a much, much finer one of the BBC Micro taking up 20k. Three charts are shown illustrating the difference between program memory and graphics memory in MODEs 7, 4 and 1 (which shows how extremely little the available memory is when using the hi-res modes). MOVE, DRAW and PLOT85 are also shown.

Next we move back out into the real world and look at a graphics studio trying to impress their prospective client (from Rolls-Royce of all places) how good their services are. They digitise a photograph into the computers memory and then alter the colours around to suit the company concerned – the idea being in the end to produce an advertisement. The graphics studio also does other work, mainly producing charts, etc. to show company sales, profits and so on.

Not forgetting our trusty BBC Micro John Coll goes on to show us a three dimensional chart of a companies sales. He explains that the program is slow because it is written in BASIC and if it was necessary to speed the process up then it would have been written in machine code – to illustrate the point he loads in Monsters from disk (a very business like example?).

When the second series of the Computer Programme was announced it was thought that it would be a teach BASIC kind of programme. That kind of thing would look very boring on television. What the programme tries to do is illustrate the possibilities of the computer without going too much into the nuts and bolts of the operation. Once more I was fairly pleased with the program from this side of things and would make a good introduction before you were taught how to use graphics on the BBC Micro.

Next Month: Keeping A Record

Paul Barbour

hardreview

"EASIREADER"

Typing in listings is tiresome: magazines slip and books will not stay open. Papers placed flat on a desk can induce neck-strain. The Easireader bookstand has been designed to overcome such problems.

The stand is made from clear acrylic plastic. It comes in three separate parts, which take only a few seconds to fit together. The stand itself is about 18" x 12", with a good lipped edge to prevent pages from slipping off or fanning. A full width cursor helps to hold papers and acts as a line marker (especially useful with multi-statement lines). A non-adjustable back prop keeps the stand at about 35 degrees from the vertical.

The stand can hold material ranging from small single pages up to double A4 pages, including books up to ¾" thick. I found it a real help with Ian Birnbaum's book on assembly language, which is otherwise difficult to keep open when typing its listings.

The Easireader has been carefully designed. It is simply, but soundly, made and serves its purpose admirably. If you are looking for a bookstand and are happy with plastic, it should meet your needs well. As to price, the only widely available comparable stand of which I am aware is sold by Inmac and costs over £25.

It is available by mail order only from Bugbear, Nicholson Buildings, Templetown, South Shields, Tyne & Wear NE33 5RZ, at £12.99 + £1.25 postage (VAT included).

Ralph Erskine

micronet update

The last issue was rather dominated by Micronet 800. This month there is just this small article. In Micronet 800 Part III I complained that only 20% of connections to Micronet were OK in my experience. Well, after pestering BT (British Telecom, not Breakfast Television) they had another look at my phone line, kicked the phone twice and hey presto I get a perfect line almost every time. One of the causes of a bad line can be operator error – make sure that you do not push the receiver into the modem too hard otherwise you'll get unequal pressure or something like that and garbage will appear on the screen only. Likewise, don't have it too loose so that you can hear the noises the computer is making.

After being on line to Micronet for almost two months now one disappointing thing still is the slow rate at which items are updated/brought into line. As an example when I wrote the review originally the Micronet 800 Programming Course and an Adventure game were both due to start "very soon" – they still haven't been put on!!!

Never mind, despite all this it is a good system and a large number of LASERBUG members are already members of Micronet. If you send in your mailbox numbers perhaps we'll start up a Micronet Contacts page in LASERBUG. Let us know your details via mailbox number 91 999 1005.

At the moment we're still discussing details of the mini-LASERBUG on Micronet with the bosses – by next month we'll have some more information.

10 arcade game high scores

This months high scores have seen both Ian Cook and Koon Loong Chan beat one of their previous scores and a new addition – that of Andrew Graham. This high score feature is proving very popular with you all but the charts are still dominated by Ian and Koon – isn't there anyone else out there with quick reactions and a strong trigger finger? If you can beat any of the high scores below let us know (i) the game, (ii) if it has difficulty levels what game you were playing on, (iii) if you have the option whether you were using the keyboard or joysticks, (iv) how long it took you, (v) the score itself (that would help slightly – Ed.) and (vi) the signature of somebody who witnessed you.

Arcadians (1)	* 22 750 *	15 Mins (J)	Ian Cook
Atlantis (2)	* 32 300 *	???	Andrew Graham
Meteors (1)	16 800	7 Mins	Koon Loong Chan
Monsters (1)	104 650	30 Mins	Ian Cook
Planetoid (1)	159 775	30 Mins	Ian Cook
Rocket Raid (1)	* 129 510 *	25 Mins	Koon Loong Chan
Snapper (1)	170 190	15 Mins	Koon Loong Chan

KEY:
J – Joysticks
1 – Acornsoft
2 – IJK Software

the complete *FX list

There have been many partial lists of the *FX commands – even the user guide does not have anywhere near the complete list. For the first time in any magazine, LASERBUG publishes the complete list of *FX commands known. Every command is documented to some extent but full details on most can be obtained by looking in the user guide (pages 421–441). Those commands marked by a # do not appear in the user guide and we give full details of their operation.

*FX0	Prints what OS version you are using
*FX1,x,y	Is reserved for user supplied routines
*FX2,x	Selects the input device
*FX3,x	Selects the output device
*FX4,x	Selects function of the editing keys
*FX5,x	Selects printer type
*FX6,x	Sets printer ignore character
*FX7,x	Sets the RS423 receive rate
*FX8,x	Sets the RS423 send rate
*FX9,x	Sets flash rate of first colour
*FX10,x	Sets flash rate of second colour
*FX11,x	Sets auto repeat delay
*FX12,x	Sets auto repeat period
*FX13,x	Disables certain events
*FX14,x	Enables certain events
*FX15,x	Flushes defined internal buffer
*FX16,x	Selects number of ADC channels used
*FX17,x	Forces conversion of defined ADC channel
*FX18	Resets the user defined keys
*FX19	Waits until the television field synchronization pulse for next frame for use with animation
*FX20,x	Selects character mode
*FX21,x	Flushes specified internal buffer
*FX117	# Returns the VDU status byte in the X register
*FX118	# If CTRL key is held down carry bit will be set, if SHIFT key then the negative bit is set
*FX123	# Is used in a user print routine (see *FX5) to tell the computer that it has finished
*FX124	Resets the ESCAPE flag
*FX125	Sets the ESCAPE flag
*FX126	Acknowledge the detection of ESCAPE
*FX127,x	Tests EOF status of specified file
*FX128,x	Reads ADC channel (ADVAL)
*FX129,x,y	Reads keyboard within a fixed time limit (INKEY)
*FX130	Read higher order address
*FX131	Read first available RAM address (PAGE)
*FX132	Read bottom of screen display address (HIMEM)
*FX133,x	Returns HIMEM for a specified MODE
*FX134	Returns in X and Y the co-ordinates of the cursor (POS/VPOS)
*FX135	Read character at current cursor position
*FX136	# Reads the X and Y co-ordinates of the light pen
*FX137,x,y	Switches cassette motor on/off and can define whether for read or write operations (for a dual cassette system)
*FX138,x,y	Inserts character y into buffer x
*FX139,x,y	Sets file options (*OPT)
*FX140,x	Selects tape speed (*TAPE)
*FX142,x	Executes ROM number x
*FX144,x,y	Alters vertical position of screen/interlace setting (*TV)
*FX145,x	Takes characters out of buffer x
*FX146,x	Read FRED (&FC00 + x)
*FX147,x,y	Write y to FRED (&FC00 + x)
*FX148,x	Read JIM (&FD00 + x)
*FX149,x,y	Write y to JIM (&FD00 + x)
*FX150,x	Read SHEILA (&FE00 + x)
*FX151,x,y	Write y to SHEILA (&FE00 + x)
*FX152,x	# Reads buffer x without altering it. Carry set if buffer clear, carry empty if anything in buffe, next character in Y
*FX153,x,y	# Insert character y into buffer x where x handles interrupt character
*FX156,x,y	# Alters the 6850 to the rule (previous value AND y) EOR x
*FX158	# Read from the speech chip
*FX159,x	# Write x to the speech chip
*FX196	# Reads auto repeat delay
*FX197	# Reads auto repeat period
*FX200,x	# Provides the ultimate form of program protection. If x=1 the escape key is disabled, if x=2 then the memory is permanently cleared if BREAK (or any BREAK function) is pressed and both if x=3
*FX202,x	# Effects CAPS LOCK/SHIFT LOCK function. If x=0 CAPS LOCK/SHIFT LOCK lights on (SHIFT LOCK function), x=16 SHIFT LOCK on, x=32 CAPS LOCK on, x=48 CAPS LOCK/SHIFT LOCK off, x=144 SHIFT LOCK light on (reverse SHIFT LOCK function i.e. normal SHIFT LOCK except when SHIFT is pressed), x=160 CAPS LOCKlight on (reverse CAPS LOCK function)
*FX210,x	# Turns sound generator on (x=0) or off (x=1)
*FX211,n	# Alter channel of bell to n
*FX212,x	# If bit 7 of x is 0 then x is used as the envelope number. If it is 1 (i.e. 128) then bits 0 and 1 are used for synchronization control, bit 2 is the flush control and bits 3, 4, 5 and 6 control the volume of the sound.
*FX213,x	# Sets the pitch of bell to x
*FX214,x	# Sets the length of bell to x
*FX219,x	# Redefines TAB key as character x
*FX220,x	# Sets ESCAPE condition to occur on character x
*FX224	Clear the VDU queue
*FX225,x	Sets the base code for function keys. Same rule applies as with *FX156
*FX226,x	Sets the base code for SHIFT-function keys. Same rule applies as with *FX156
*FX227,x	Sets the base code for CTRL-function keys. Same rule applies as with *FX156
*FX228,x	Set the base code for SHIFT-CTRL-function keys. Same rule applies as with *FX156
*FX229,x	Sets ESCAPE key. If x=0, ESCAPE acts as usual. If x=1 ESCAPE produces ASCII code 27. Same rule applies as with *FX156
*FX230,x	Alter ESCAPE function. If x=0 all buffers are flushed (default). If x=1 then no buffers are flushed. Same rule applies as with *FX156
*FX231,x	Enable/disable 6522 IRQ. Same rule applies as with *FX156
*FX232,x	Enable/disable 6850 ACIA IRQ. Same rule applies as with *FX156
*FX233,x	# Enable/disable system 6522. Same rule applies as with *FX156

*FX235 # Tests whether speech synthesis is fitted or not. Returns 255 if it is
*FX236 # Reads current output buffer
*FX237 # Reads current edit state
*FX241 # Reads the *FX1 value(s)
*FX245 # Reads the printer type
*FX246 # Reads the printer ignore character
*FX247,x # If this value is unaltered (or x=0) then usual BREAK function occurs. Otherwise, this byte is used as a machine code instruction (i.e. JMP or 76)
*FX248,x # This can be used to add a second machine code instruction for *FX247 (i.e. the lsb of a memory location)
*FX249,x # This can be used to add a third machine code instruction for *FX247 (i.e. the hsb of a memory location)
*FX252 # Reads ROM number of current language
*FX253 # Returns what kind of BREAK last occurred. If x=0 it was a soft BREAK, if x=1 it means no break has occurred since the machine was switched on and if x=2 it was a hard break
*FX254,x # If x=64 on a model B it convinces the computer after the next MODE change that it only has 16k memory. This is only reset when x=128. It is not effected by a hard break
*FX255,x Allows you to read the current keyboard link value. It can be written to but that value will be reset to the hardware determined version on a hard break. For reference bits 1, 2 and 3 determine what screen mode will be used, bit 4 controls the auto-boot and bits 5 and 6 control the disks

There are approximately 90 calls detailed above. Is that all of them? The answer I am afraid is no! As far as we can work out there are 66 calls still unknown (119, 120, 121, 122, 141, 143, 154, 155, 157, 160, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 198, 199, 201, 203, 204, 205, 206, 207, 208, 209, 215, 216, 217, 218, 221, 222, 223, 234, 238, 239, 240, 242, 243, 244, 250 and 251). What are they? Well, you should be able to access every one of the BBC Micros functions using *FX rather than use ? or !. For instance instead of using ?216/?602 to alter the CAPS LOCK/SHIFT LOCK functions you should use *FX202.

Most of the calls only apply to the series 1 OS but what are these extra calls that are still undocumented. Well, if you can find out what any of the 66 unknown calls are, do please let us know and we will award a prize to all replies published. Hopefully within a few months we'll be able to have a completely comprehensive list of calls.

Lets all get experimenting (you might find a copy of the BBC Micro Revealed handy – as far as possible most memory locations described in the book should be alterable by an FX call).

basic BASIC

My apologies for the name but I thought something you'd remember was better than say Programming Course in BASIC for Beginners. I'll stick to Basic BASIC. BASIC stands for Beginners All purpose Symbolic Instruction Code and is the language your computer works in. Over the coming months we'll be having a series all about BASIC. Hopefully it'll be able to take an utter novice from the beginning and by the end produce a fluent programmer. Now obviously where to start this is a problem. Most of you already know programming to some kind of standard but there is one thing I always try to make clear – there is the person who is able to program and there is the person who can program properly. At the beginning of each article we'll tell you what we are going to cover. Simply join in when we are covering parts which you don't know about/would like to know more.

Far, far too many programming courses are written by programmers who already know the language and introduce the subject how they think it should be done, not the correct way. The method below has been extensively tried. I personally teach a class of 11/12 year olds and they find it suitable (hello all those at St. Mary's!). As well as this I've managed to turn several hesitant adults of all types into programmers by the same method. Bear with us, join in when you think you should and you'll be able to get a lot more out of your micro.

PART I

BASIC words covered this month: RND, PRINT, INPUT, IF, THEN, ELSE, REPEAT, UNTIL, RUN, CLS, LIST, NEW

OK, now everybody reading Part I of this article should be complete novices. As such there are a few introductory things I should say. Firstly a computer is not some magical device that can do anything you desire. Compare it to a pile of bricks. A pile of bricks by themselves can do absolutely nothing – if you manipulate them around you can have yourself a house! A computer is similar, by itself it does nothing but when somebody is there to program it the only limit is the programmers imagination.

Just one more note before we start anything – remember that to enter anything at all into the computer you have to press the big key marked RETURN. The best way to understand what is said below is quite simply to type it in on your BBC Micro and see what happens. Although most of what will be said applies to most home computers, some of the words are only available on your Beeb.

Right, let's start! The best way to actually write your first program is to take a real life example and transfer that onto the computer. Think of a simple addition sum, no harder than 10+10 i.e. what is 3+4. I answer 7 and you tell me that I'm right. Think of another one, say 2+7. I answer 10 and you tell me that I am wrong. Now what you have got to do to transfer that onto the computer is to break what you did down exactly. Believe it or not when you thought of that question you went through 6 steps which were:

(i) Think of a number between 1 and 10
(ii) Think of a second number between 1 and 10
(iii) Work out what the answer is and remember it
(iv) Ask me the question
(v) Get my answer
(vi) if my answer is the same as the real answer tell me I'm right otherwise tell me I'm wrong.

With your computer set up we'll try and get the computer to do the same thing. The first thing we did is think of a number between 1 and 10. As it is the first thing we did we can enter 1. The number we thought of was the first number so we can call that FIRSTNUMBER (no space between FIRST and NUMBER). The number we thought of could have been anything between 1 and 10 i.e. 3, 6, 3, 6, 2, 8, 1, 9, 10, 7, 4, 5. These numbers have no real pattern and so are called random. So what we want to do is think of the FIRSTNUMBER and give that a value between 1 and 10 so it has to be a RANDOM NUMBER BETWEEN 1 AND 10. Well all that is rather a mouthful so it can be abbreviated. RANDOM NUMBER becomes RND (RaNDom number). We replace the between parts by enclosing the values in brackets. Also the computer automatically knows that the number has got to start from 1 so we just have to put in the 10 part. RANDOM NUMBER BETWEEN 1 AND 10 becomes RND(10). So the whole of the first thing we did can be simulated on the computer by the line:

```
1 FIRSTNUMBER=RND(10) <RETURN>
```

(note that you should press the RETURN key when you see the <RETURN>, not type it in!).

The second thing we do is similar except that this time we want to think of a second number so enter:

```
2 SECONDNUMBER=RND(10) <RETURN>
```

Third is to work out the answer. Just as if we were writing an equation down we can put:

```
3 ANSWER=FIRSTNUMBER+SECONDNUMBER <RETURN>
```

The fourth step is where it starts getting complicated. We want to print a message on the screen and so quite simply we use the word PRINT. If you are writing down what someone said you should really put it in quotation marks – in BASIC you have to put quotation marks around what the computer will print up on the screen. The easiest way to explain fully about what the fourth line to contain is to put it down and then explain. So, the fourth thing we do is expressed on the computer as

4 PRINT"WHAT IS";FIRSTNUMBER;"PLUS";
SECONDNUMBER
The actual words that will be printed on the screen are "What is" and "plus" because they are in quotation marks. We actually want to ask what is the firstnumber plus the secondnumber. We can't enclose the whole thing in quotation marks like say PRINT"WHAT IS FIRSTNUMBER PLUS SECONDNUMBER" because the computer would print out the word FIRSTNUMBER and SECONDNUMBER. What you really want it to do is print out the numbers that **you have called** FIRSTNUMBER and SECONDNUMBER. This is why we have the funny arrangements of quotation marks, semi-colons and words. We actually want to print out the words WHAT IS and so they are enclosed in quotation marks. Instead of the word FIRSTNUMBER we want the number we called FIRSTNUMBER so instead of enclosing it in quotation marks, we use semi-colons (;). The reason for this will be explained in a future part but for the time being it is simpler if you just accept the fact.

Fifth thing. We want to enter the persons answer. Unfortunately we can't use the word enter but instead use INPUT. We have to INPUT something which is the persons answer but we have already used the word answer in line 3. We might as well use the word guess. So, the fifth thing (or line 5) is
5 INPUT GUESS

Now the sixth and final thing we do is the part that makes the computer seem intelligent. The thinking you do goes something like "if the answer they told me is the same as the correct answer then tell them that they're right otherwise tell them they're wrong". You can't put that in but instead try:
6 IF GUESS=ANSWER THEN PRINT"RIGHT" ELSE PRINT"WRONG"
You should be able to compare the sentence above with the "computer sentence" exactly. To summarise you should now have a program in your computer something like:

```
LIST
    1 FIRSTNUMBER=RND(10)
    2 SECONDNUMBER=RND(10)
    3 ANSWER=FIRSTNUMBER+SECONDNUMBER
    4 PRINT"WHAT IS ";FIRSTNUMBER;" PLUS ";SECONDNUMBE
R
    5 INPUT GUESS
    6 IF GUESS=ANSWER THEN PRINT"RIGHT" ELSE PRINT"WRO
NG"
```

To actually make the program work you have to type in the word RUN. If you do that something like the following should happen:

```
>RUN
WHAT IS 2 PLUS 10
?12
RIGHT
>RUN
WHAT IS 1 PLUS 6
?7
RIGHT
>RUN
WHAT IS 1 PLUS 4
?6
WRONG
>RUN
WHAT IS 8 PLUS 9
?13
WRONG
>RUN
WHAT IS 6 PLUS 1
?6
WRONG
```

except that the questions might be different (as they are RaNDom). If you get an error message i.e. "***** at line nn" the ***** might say No Such Variable and the nn might be 3 then don't worry, simply re-enter the line that the computer says is wrong. Whatever you do you cannot harm the computer (unless you take a hammer to it!).

So now we have written our first program. The best thing to do now is to look carefully at what you have written and try to understand exactly what is happening. If you haven't a mathematical background then an important principle you might find hard to grasp is the way that we can assign a number to a name in the way the first 3 lines do.

There are a couple of other words that you should know while we're here. By now the screen probably looks quite a mess. If we want to CLear the Screen we can simply enter CLS. We have already met RUN but if we want to list out the program that we typed in we simply use the word LIST. Simple!

When you are sure that you understand fully what the first program does enter the word NEW into your computer. What this has now done is to clear the old program from memory. You now have a completely fresh computer. You can now attempt to program something into the computer yourself!

The program above asked an addition question no harder than 10 + 10. You try to write a similar program but instead of an addition sum try a multiplication one. Because of this you want the question to be able to go as high as 12 x 12 and so the first two lines have to be altered. The third line must also be changed because of the different sign. The sign for multiplication incidently is not X or x as these are impossible to distinguish from a small and large letter X. On the computer we use * to represent multiply. The only other line that needs altering is the fourth one and that is just to change the word "plus" to "multiply". Remember that with line four when you actually want to print words on the screen these must be enclosed in " whereas numbers **represented** by words have to be enclosed by ; – the only exception is that you don't need a ; at the end of a variable (the word we give to numbers represented by words) if the variable itself is at the end of a line. If you get stuck don't panic – refer back to the listing of the first program.

Try your best to produce something – you shouldn't find it too hard. We'll be printing the answer in next months magazine.

With an almost identical program above to help with that problem we'll carry on further. With the program above we put down 1 for the first thing we did (thought of the first number), 2 for the second and so on. This method is all very good but it does have some problems. Consider the following program made up just of simple PRINT statements:

```
LIST
    1 PRINT"PLUG IN COMPUTER & TV"
    2 PRINT"CONNECT COMPUTER TO TV"
    3 PRINT"SWICTH ON COMPUTER & TV"
    4 PRINT"RUN PROGRAM"
>RUN
PLUG IN COMPUTER & TV
CONNECT COMPUTER TO TV
SWICTH ON COMPUTER & TV
RUN PROGRAM
```

You can see that there should be an extra step between step 3 and step 4 (type in program) but where are we going to add it, step 3½?

Originally, we started at line 1 and went up in 1's just to demonstrate how the first step in the program represents the first thing that you'd do in real life, the second line the second thing you'd do in real life and so on. When you are writing your own program it is best to start at line 10 and go up in steps of 10. Thus if we have to add something in (as in this case) you can use a "middle" number. We want to add something between lines 30 and 40 and so we can enter it as 35:

```
>LIST
   10 PRINT"PLUG IN COMPUTER & TV"
   20 PRINT"CONNECT COMPUTER TO TV"
   30 PRINT"SWICTH ON COMPUTER & TV"
```

```
  40 PRINT"RUN PROGRAM"
>35 PRINT"TYPE IN PROGRAM"
>RUN
PLUG IN COMPUTER & TV
CONNECT COMPUTER TO TV
SWICTH ON COMPUTER & TV
TYPE IN PROGRAM
RUN PROGRAM
```

Let's try another program. First of all read the following question:
Q. A man goes up to his boss and says next month he would like to be paid differently. He suggests that for the first day all he wants to be paid is 1p, the second day 2p, the third day 4p and so on, the amount doubling each time. Pretend that you are the mans boss and work out whether it is a good deal or not. Assume the man works 7 days a week and that the month is 31 days long.

Now what you should try to do is to program that question into the computer by again breaking it down into its separate steps and transferring each step onto the computer. The steps can be described as:
(i) It is the first day
(ii) He has no money to start off with
(iii) His pay is originally 0.01
(iv) Add his pay to his money
(v) Double his pay
(vi) Add one to the day
(vii) If it isn't the end of the month (i.e. the day is still less than 32) go back to step (iv)
(viii) Print out how much money he made.
The actual program works out to be:

```
>LIST
  10 DAY=1
  20 MONEY=0
  30 PAY=0.01
  40 MONEY=MONEY+PAY
  50 PAY=PAY*2
  60 DAY=DAY+1
  70 IFDAY<32THEN40
  80 PRINT"£";MONEY;"P"
```

Again if you compare the steps described in the article with the steps in the listing you can see how they compare. One thing you might find hard to understand is the way lines 40, 50 and 60 are. To take line 40 as an example how this reads is "his money will be what he had before plus his pay". Similarly line 50 reads "his pay will be what he had before doubled" and so on.

When you think you understand the program RUN it and the answer should come out to be £21 474 836.50p!!!

Now if you look carefully at the listing you should be able to see that there are three lines that we want to keep on repeating until it is the end of the month. At the moment to keep it going we use the messy looking thing at line 70 reading "if it is less than day 32 (i.e. not the end of the month) then goto line 40".

There is a much neater way of repeating this section. It is lines 40-60 we want to repeat so we mark the beginning by using the word REPEAT at line 35 and the end by overwriting line 70 with UNTIL DAY>31 i.e. keep on doing lines 40-60 until it is the end of the month:

```
>35 REPEAT
>70 UNTILDAY>31
>RUN
£21474836.5P
>LIST
  10 DAY=1
  20 MONEY=0
  30 PAY=0.01
  35 REPEAT
  40 MONEY=MONEY+PAY
  50 PAY=PAY*2
  60 DAY=DAY+1
  70 UNTILDAY>31
  80 PRINT"£";MONEY;"P"
```

If you think about it by using the REPEAT and UNTIL does show what you are doing a lot clearer.

Are you totally confused now? I certainly hope not. If you are don't worry – go back to the beginning and start again. You have a month before you can start Part II so you have plenty of time to digest if all. I won't leave though without setting you something else to do – another programming problem that you've got to solve by yourself.
Q. Write a program to print out all the odd numbers between two values entered by the user. To save you working out the mathematics:
ODD=NUMBER
ODD=ODD–INT(ODD)
IFODD=0.5 THENPRINTNUMBER
If you incorporate these lines into your program if NUMBER is an odd number it will print it out. As this is the first proper question I have set I'll give you some more guidelines on how to do it:
(i) Enter the two values in the program
(ii) Test the number to see if it is odd, if so print it out (use the three lines above)
(iii) Add one to the first number
(iv) Keep going back to (ii) until the second number entered is greater than the first one.
Try using REPEAT and UNTIL in your program. See how well you can do. PLEASE DO NOT SEND IN THE ANSWERS TO US – THERE IS NO PRIZE. We are though, of course, always pleased to receive correspondence from you on any subject.

More next month . . .

Paul Barbour

discounts

If you are a member of LASERBUG you will have a membership card – that card not only proves that you belong to LASERBUG but also entitles you to special discounts that certain companies have offered. At present, Dracal Ltd. offer a 5% discount, Futura Software offer a 25% discount and Silent Computers offer a 10% discount. Dracal Ltd. produce many different things for the BBC Micro including design sheets and monitor stands. They can be contacted at Dracal (North West) Ltd., P.O. Box 130, Warrington, WA1 4QB. Phone Padgate 815419. Futura Software produce a large number of games for the BBC Micro on which you can get a 25% discount. More details from Futura Software, 63 Lady Lane, Chelmsford, Essex, CM2 0TQ. Silent Computers manufacture another monitor stand made of aluminium with room for a monitor/TV and Dual Disks or a Single Disk and a 2nd Processor. 10% is the discount on offer and you can find out more from Silent Computers, 27 Wycombe Road, Tottenham, London, N17 9XN. Full details on these three offers were given in issues 6, 9 and 10 respectively. Please quote your membership number to these companies both when ordering goods and corresponding with the companies. We are not acting as agents for these companies and cannot be held responsible for their actions or services they provide in any way whatsoever.

softspot nuisance alarm

At one time or another we have all seen the nuisance who goes to the nearest computer and types in:
10 PRINT"ARSENAL ARE BEST"
20 GOTO 10
RUN
This kind of thing can be seen anywhere there is a computer free to prying hands. Quite rightly so, this kind of "nuisance" annoys a lot of people. If your computer is accessable to such people try typing in the following program and give those idiots a fright! It's also good at catching out unweary people!

```
LIST
   10 REM    Nuisance Alarm
   20 REM    by Paul Barbour
   30 :
   40 REM        27/3/83
   50 :
   60 REM      Version 1.0
   70 :
   80 REM Takes up 0.80k memory
   90 :
  100 REM   Suitable for 16k
  110 :
  120 REM   Written on OS 0.1
  130 :
  140 REM   (C) LASERBUG 1983
  150 :
  160    :::::
  170 :
  180 ONERRORGOTO310
  190 *KEY10OLD|MGOTO310|M
  200 *FX15,1
  210 *TV0
  220 MODE7
  230 VDU7
  240 PRINT'"BBC Computer 32K"
  250 PRINT'"BASIC"
  260 PRINT'">";
  270 REPEATA%=GET
  280   VDUA%
  290   IFPOS=0ANDVPOS=5THENPRINTTAB(0,5);">";
  300   UNTILA%=13
  310 CLS
  320 PROCP("THIS COMPUTER IS TEMPORARILY OUT OF",3)
  330 PROCP("ORDER - NORMAL SERVICE WILL BE",6)
  340 PROCP("RESUMED AS SOON AS POSSIBLE.",9)
  350 PROCP("FOR THE TIME BEING PLEASE LEAVE THE",12)
  360 PROCP("COMPUTER ALONE.  THANK-YOU",15)
  370 FORsiren=0TO5
  380   FORX%=0TO253
  390     SOUND&11,-15,X%,1
  400     SOUND&12,-15,X%+1,1
  410     SOUND&13,-15,X%+2,1
  420     NEXT
  430   NEXT
  440 RUN
  450 DEFPROCP(A$,Y)
  460 X=(40-(LEN(A$)+2))/2
  470 X=X-1
  480 PRINTTAB(X,Y);CHR$134;CHR$141;A$;TAB(X,Y+1);CHR$
134;CHR$141;A$
  490 ENDPROC
```

variables listing

I haven't seen anywhere a listing for 'X-reference' utility which prints out all the variables used in a programme and the line numbers they appear on. The nearest thing that I've found is in Jeremy Ruston's BBC Micro (p. 133) in his excellent analysis of how the Beeb handles variables. Unfortunately the program doesn't work in quite the way it states – not on a 1.2 OS anyway, maybe it does on the 0.1. Nevertheless it's a useful programme and worth persevering with; it lists all the active variables, not with their line numbers but with their present values (resident static variables excepted).

As he describes, the Beeb doesn't store the initial letters of any variables. Instead, it keeps a sort of alphabetically indexed address book in page 4 from &482 to &4F4, which he takes as the starting point. Altering MODE or PAGE has no effect on this address-book but entering LOAD or RUN cancels it completely, resetting it all to zero. So the act of loading his program cancels the data it needs to operate on! No doubt in the second edition of his book he'll provide an elegant solution to this but meanwhile two possibilities occur to me. Firstly, if you have plenty of memory available, you can renumber this programme to high line numbers, append it to your own (p. 402, User Guide) by the "dirtier" method and having ESCAPED from your own, GOTO his. This is simple and effective. My problem was being in MODE 2 with only a handful of bytes spare (his takes up over 1½k!) and so unable to append. Getting round this is a bit devious. First thing is to program a fkey to copy the contents of &482 to &4F4 into a high bit of RAM when it becomes available. Then, prefix Ruston's program with a line or two that copies it back again, and SAVE it.

Now the main program can be run, activating its variables . . . Change to MODE 7, making space available, and press the fkey. Reset PAGE near the top of memory and type NEW. LOAD and RUN the utility. (I've incorporated those last two steps into the fkey 7 routine.)

```
*FX 18 : REM Clears all fkeys
*KEY7 MODE7| MF%=&482:G%=HIMEM-&1200:FORK%=
OTO114STEP4:G%!K%=F%!K%:NEXT:PAGE=HIMEM-
&1000| MNEW| MPRINT"PAGE=&";  PAGE:CHAIN
"VARLIST"| M
```

Then add to the front of Jeremy's program:

```
5 F%=&482:G%=HIMEM-&1200
6 FOR K%=OTO114STEP4:F%!K%=G%!K%:NEXT
```

and SAVE the whole lot as "VARLIST". It's useful also to SAVE the fkey setting into a separate VARLIST LOADER programme.

Patrick Dowling

softspot maths race

Cast your mind back twice – firstly to Issue 6 (November'82) and our Maths Race program and to Issue 9 (February'83) and a letter from Mr. R.A. Smith, the headmaster of Blue Bell Hill Junior School. Mr. Smith wrote in saying that Maths Race was good but the questions were too hard for the program to be of general use in schools and suggested that the user should have the option to put what standard of questions he wants. I suggested that Mr. Smith tried altering the program himself and sending us in his result. He did this and the outcome appears below. Thanks Mr. Smith!

```
>LIST
   10 REM        MATHS RACE
   20 REM     by Paul Barbour
   30 REM     and Mr.R.A.Smith
   40 REM   (Headmaster of Blue
   50 REM Bell Hill Junior School)
   60 :
   70 REM  Basic Program by Paul
   80 REM  Additional features by
   90 REM         Mr.Smith
  100 :
  110 REM      February'83
  120 :
  130 REM      Version 2.0
  140 :
  150 REM  Requires 5.26k memory
  160 :
  170 REM      Requires 32k
```

```
180 :
190 REM    Written on OS 0.1
200 :
210 REM    (C) LASERBUG 1983
220 :
230    :::::
240 :
300 *TV255
310 MODE7:VDU23;8202;0;0;0;
320 VDU23,224,28,28,8,28,42,8,20,34,
330 VDU23,225,0,24,24,0,255,0,24,24
340 PROCintro
350  CLS
360  PROCchoice
370 CLS
380 PROCintro_test
390 MODE2:VDU23;8202;0;0;0;
400 PROCdraw_track
410 PROCgame
420 MODE7:VDU23;8202;0;0;0;
430 PROCcontinue
440 GOTO 390
450 DEFPROCintro_test
460 PRINT"REMEMBER:";CHR$136;"DELETE";CHR$137;"will
rub out the last""          thing you entered."""
      You must press";CHR$136;"RETURN";CHR$137;"after
""          you have put the answer to a""
question."""
470 C%=0
480 FOR C=1 TO 10
490    IF Q=1 GOSUB 680:GOTO570
500    IF Q=2 GOSUB 750:GOTO570
510    IF Q=3 GOSUB 800:GOTO570
520    IF Q=4 GOSUB 860:GOTO570
530    IF Q=5 ON RND(2) GOSUB 680,750:GOTO570
540    IF Q=6 ON RND(2) GOSUB 800,860:GOTO570
550    IF Q=7 GOTO 560
560   ON RND(4) GOSUB680,750,800,860
570   PRINTCHR$129;C;"  What is ";number1;function$;
number2;" ?";CHR$133;
580   INPUT""answer
590   IF answer=real_answer THEN PRINT'CHR$130"CORRE
CT !":C%=C%+1:PROCright
600   IF answer<>real_answer THEN PRINT'CHR$131"WRON
G !  It was ";real_answer:PROCwrong
610   PRINT''
620   NEXT
630 PRINTCHR$134;"You scored ";C%;" out of 10"
640 PRINT'''CHR$131;CHR$157;CHR$132;"Press the Space
Bar to start game..."
650 REPEATA=GET:UNTIL A=32
660 ENDPROC
670 :
680 REPEATnumber1=RND(Z)
690   number2=RND(Z)
700   UNTIL number1>number2
710 real_answer=number1+number2
720  function$=" + "
730 RETURN
740 :
750 GOSUB680
760 real_answer=number1-number2
770 function$=" - "
780 RETURN
790 :
800 number1=RND(Z)
810 number2=RND(Z)
820 real_answer=number1*number2
830 function$=" x "
840 RETURN
850 :
860 number2=RND(7)
870 real_answer=RND(Z)
880 number1=number2*real_answer
890 :
900 IF?871=7 THEN function$=" ^ ":ELSE function$=" "
+CHR$225+" "
910 RETURN
920 DEFPROCintro
930  PRINTCHR$157TAB(13);CHR$141;CHR$132"MATHS RACE"
'CHR$157TAB(13);CHR$141;CHR$132"MATHS RACE"'CHR$157TAB
(14);CHR$132"_____ ____"
940 PRINT''CHR$134;"     In this game you must race
against"CHR$134"the computer over 800 metres."
950 PRINT'CHR$134;"     You move by answering mental
"'CHR$134"arithmetic questions which could be on"'CHR$
134"addition,  subtraction,  multiplication"CHR$134"or
 division."
960  PRINT'CHR$134;"     The quicker you answer the"
'CHR$134"questions the more you move.  The"'CHR$134"co
mputer moves 20m per question.  If"'CHR$134"you answer
 within 5 seconds you move"
970 PRINTCHR$134;"40m,  within 10 seconds 20m and an
y"'CHR$134"longer than that 10m.  If you get a"'CHR$13
4"question wrong then you do not move at"'CHR$134"all.
"
980 PRINT''CHR$131;CHR$157;CHR$129;"Press RETURN to
choose   ";CHR$156
990 REPEATA=GET:UNTILA=13
1000 ENDPROC
1010 DEFPROCdraw_track
1020 COLOUR131
1030 CLS
1040 VDU29,640;512;
1050 FOR X%=0 TO 360 STEP4
1060   S=SINRAD(X%):S1=SINRAD(X%+30)
1070   C=COSRAD(X%):C1=COSRAD(X%+30)
1080   SOUND&11,-15,X%/2,5:SOUND&12,-15,(X%/2)+4,5:SO
UND&13,-15,(X%/2)+8,5
1090   MOVES*440,C*312
1100   DRAWS*640,C*512
1110   PLOT85,S1*640,C1*512
1120   DRAWS1*440,C1*312
1130   PLOT85,S*440,C*312
1140   NEXT
1150 GCOL0,6
1160 MOVE0,300:DRAW0,512
1170 ENDPROC
1180 DEFPROCright
```

```
1190 S=RND(100)
1200  FOR X=1 TO 5:SOUND1,-10,5,2:SOUND0,-10,1,2:FOR
D=1 TO 200:NEXT:SOUND1,-10,20,2:SOUND0,-10,4,2:FOR D=1
 TO 200:NEXT:NEXT X
1210 ENDPROC
1220 DEFPROCwrong
1230 FOR noise=1 TO 50 STEP.1:SOUND&10,-15,2,1:NEXT
1240 ENDPROC
1250 DEFPROCgame
1260 P=0:C=0
1270 VDU28,5,21,14,10
1280 COLOUR134:CLS
1290 VDU5
1300 GCOL0,1
1310 MOVE SINRAD(0)*590-30,COSRAD(0)*462+10:PRINTCHR$
224
1320 GCOL0,4
1330 MOVE SINRAD(0)*490-30,COSRAD(0)*362+10:PRINTCHR$
224
1340 VDU4
1350  IF Q=1 GOSUB 680:GOTO1430
1360  IF Q=2 GOSUB 750:GOTO1430
1370  IF Q=3 GOSUB 800:GOTO1430
1380  IF Q=4 GOSUB 860:GOTO1430
1390 IF Q=5 ON RND(2) GOSUB 680,750:GOTO1430
1400  IF Q=6 ON RND(2) GOSUB 800,860:GOTO1430
1410  IF Q=7 GOTO 1420
1420 ON RND(4) GOSUB680,750,800,860
1430 CLS
1440 COLOUR1:PRINT'" What is:"
1450 PRINT''" ";number1;function$;number2
1460 TIME=0
1470 INPUT''" ? "answer
1480 VDU5
1490 GCOL0,7
1500 MOVE SINRAD(P/2.2)*590-30,COSRAD(P/2.2)*462+10:P
RINTCHR$224
1510 MOVE SINRAD(C/2.2)*490-30,COSRAD(C/2.2)*362+10:P
RINTCHR$224
1520 VDU4
1530 IF answer=real_answer THEN CLS:PRINT'''''"CORREC
T  !"
1540 IF answer=real_answer AND TIME<500 THEN P=P+40
1550 IF answer=real_answer AND TIME>=500 AND TIME<100
0 THEN P=P+20
1560 IF answer=real_answer AND TIME>=1000 THEN P=P+10
1570 IF answer<>real_answer THEN CLS:PRINT'''"WRONG
'"'"  IT WAS"'"    ";real_answer
1580 C=C+20
1590 VDU5
1600 GCOL0,1
1610 MOVE SINRAD(P/2.2)*590-30,COSRAD(P/2.2)*462+10:P
RINTCHR$224
1620 GCOL0,4
1630 MOVE SINRAD(C/2.2)*490-30,COSRAD(C/2.2)*362+10:P
RINTCHR$224
1640 GCOL0,6
1650 MOVE0,300:DRAW0,512
1660 IF answer=real_answer THEN PROCright:ELSE PROCwr
ong
```

```
1670 IF P>=800 OR C>=800 THEN ENDPROC
1680 GOTO 1340
1690 DEFPROCcontinue
1700  FOR J=1 TO 2:B=128
1710    FOR X=1 TO 8:FOR Y=0 TO 24:PRINTTAB(0,Y);CHR$
B;CHR$157:NEXT Y:SOUND0,-15,RND(50),2:B=B+1:NEXT X:NEX
T J:CLS
1720  IF P>=800 THEN PRINTTAB(12,3)CHR$141;CHR$129;"W
ELL DONE !"'TAB(12)CHR$141;CHR$130;"WELL DONE !"''TAB
(3)CHR$131;"You managed to beat the computer."
1730 IF C>=800 THEN PRINTTAB(13)CHR$141;CHR$129;"BAD
LUCK !"'TAB(13)CHR$141;CHR$130;"BAD LUCK !"''CHR$131;"
Unfortunately the computer beat you."
1740  RESTORE 1750:FOR MUSIC=1 TO 11:READA,Z:SOUND1,-
15,A+48,Z:NEXT MUSIC
1750  DATA33,9,49,3,41,3,33,3,49,9,41,9,33,15,61,1,69
,1,77,1,81,18
1760 PRINT''TAB(6)CHR$133"Would you like another go ?
"'TAB(11)CHR$133"(Press 'Y' or 'N')"
1770  A$=GET$:IF A$="Y" THEN ENDPROC
1780 IF A$<>"N" THEN 1770
1790  IF A$="N" THEN END
1800  END
1810 DEFPROCchoice
1820 PRINTTAB(12,3);CHR$141;CHR$130;CHR$157;CHR$132;"
CHOOSE  ";CHR$156
1830 PRINTTAB(12,4);CHR$141;CHR$130;CHR$157;CHR$132;"
CHOOSE  ";CHR$156
1840  PRINTTAB(3,7);CHR$129;"1. Addition only"
1850  PRINTTAB(3,8);CHR$130;"2. Subtraction only"
1860 PRINTTAB(3,9);CHR$131;"3. Multiplication only"
1870  PRINTTAB(3,10);CHR$132;"4. Division only"
1880 PRINTTAB(3,11);CHR$133;"5. Addition and Subtract
ion"
1890 PRINTTAB(3,12);CHR$134;"6. Multiplication and Di
vision"
1900 PRINTTAB(3,13);CHR$135;"7. Everything"
1910  PRINTTAB(0,17);CHR$131;"What is the biggest num
ber you"'CHR$131"want to work with? (2_12)";CHR$130;:I
NPUTTAB(27,18)Z:IF Z<2 OR Z>12 PRINTTAB(27,18);SPC(40)
;:GOTO 1910
1920  PRINTTAB(0,20);CHR$134;"What kind of sums would
 you like?";CHR$133;:INPUTTAB(35,20)Q
1930 PRINTTAB(0,23);CHR$131;CHR$157;CHR$129;"Press RE
TURN for 10 sample questions"
1940 REPEATA=GET:UNTILA=13
1950 ENDPROC
```

what to do with your 1.2

(Apologies for the title!) With the offer on the back page of LASERBUG ,many people now are upgrading their 0.1 OS to a 1.2. What do you get for your money? Well, continue reading and you'll find out about a whole host of features you didn't know you had!!!

There area two primary benefits from having the 1.2 OS. The first one is that the annoying (and destructive) cassette bug is fixed – never again will you SAVE a program and find then when you try to load it in again you have block 0 missing! The other facility you have is that of paged ROMs. Neither of these features are worth describing here as there is very little to say!

When programming there is one major difference which can be put to use right away, that of turning the cursor off and on. On OS 0.1 you use VDU23;8202;0;0;0; to switch it off and switching it on again requires some other complex VDU statement to switch it on again. On OS 1.2 you can use VDU23,1,0;0;0;0; to switch it off and VDU23,1,1;0;0;0; to put it back again. This is much more logical as you can see – 0 for off, 1 for on.

The keyboard has several other features with the new OS. The first is the use of the function keys to produce control codes. Just pressing a function key would produce the same effect as on OS 0.1. If you press SHIFT–f1 you get the teletext code for RED. SHIFT–f2

continued on page 22

computer camp '83

In 1982 when the idea of holding a 'Computer Holiday Camp' at the University of Southampton was first suggested the proposal met with a certain scepticism.

'Hi de Hi Campers' this is your computer speaking, was the predictable headline that appeared in 'Computer Weekly'. Then there was the young lady reporter who, after some aggressive questioning, suddenly demanded to know if I really believed computing would replace sex as a major holiday pastime.

If I had been organising a holiday camp for say musicians or model railway enthusiasts I doubt that the TV cameras would have spent the day filming their activities or that I would have been interviewed by a succession of more than sixty journalists.

The 'Mighty Micro' is news but that does not explain why so many of the Press found it hard to believe in the existence of people who enjoy computers, who want to learn how to make better use of the Micro or simply want to spend time with others who share their interest and enthusiasm.

That said, all publicity was gratefully received, the 'Computer Camps' were a great success and we all learnt a lot from the experience. This year the 'Computer Holiday' idea has been developed and extended and we have arranged:-

1) Computer holidays devoted specifically to computer graphics. The BBC Micro has tremendous potential as a graphics machine even more so now that devices such as light pens, pantographs, graph tablets, etc. are becoming available.
2) For the 'Camps' to become International by promoting them throughout Europe and the Americas. Last year people came from places as far apart as Belgium and Brazil and the different perspective they provided was an unexpected bonus. Also, more than half the people who came subsequently bought themselves a BBC Micro so the 'Holidays' provide a good opportunity to promote British goods and services. The possibility of establishing similar 'Computer Camps' at suitable locations on the Continent is being actively investigated.
3) Special 'Computer Holiday' weeks for GPs, Teachers, Dentists, Business Users, Managers and other groups. With Discs, the Z80 chip and the CP/M operating system the BBC Micro becomes a viable high performance business machine and at a price which is very competitive. My company currently has a contract to supply 130 computers for use in a chain of retail shops. After a thorough evaluation, the BBC Micro is at the top of the short list.

Throughout the whole of the season we shall be running computer holidays for beginners and experienced programmers open to everyone age 9 years old and over. Chalk and talk is kept to a minimum and tuition is based on the individual working through a series of carefully graded practical projects. These are related to his or her interests and experience and allows everyone to work at their own pace.

Tuition, accommodation and the use of the computers costs £140 per week (£115 Non-Resident). If you want to bring friends or relations who'd just like to sit in the sun, walk in the New Forest or go sightseeing they can come to for the price of the accommodation, £35 for a week. There's a £5 discount for members of Laserbug and a further £10 discount if they bring and use their own machines.

For more information write to Dr. Lionel Wardle, Computer Holidays, 37 University Road, Southampton SO2 1TL. (Large s.a.e. please) or Telephone 0703 558621.

18 softspot one arm bandit

Before Arcade Games in Pubs were . . . the fruit machines. This program simulates a one-armed bandit using numbers on each reel. Full instructions are given in the program. Try altering it so that instead of numbers, pictures are shown in each wheel.

```
>L
L.
   10 REM    ONE-ARMED BANDIT
   20 REM    by Paul Barbour
   30 :
   40 REM         9/2/83
   50 :
   60 REM      Version 1.0
   70 :
   80 REM Takes up ~6.05k memory
   90 :
  100 REM   Uses Modes 1 and 7
  110 :
  120 REM      Requires 32k
  130 :
  140 REM    (c) LASERBUG 1983
  150 :
  160    :::::
  170 :
  180 GOSUB410
  190 PROCinitialise
  200 ONERRORPROCerror
  210 *KEY10OLD|MCLS|M
  220 PROCtitle
  230 PROCintro
  240 GOSUB420
  250 PROCmachine
  260 PROCcash
  270 REPEAT
  280   PROCpull
  290   PROCspin
  300   PROCwin
  310   PROCcash
  320   PROChold
  330   UNTILcash<0.1
  340 GOSUB410
  350 PROCbroke
  360 END
  370 :
  380    :::::
  390 :
  400 :
  410 MODE7:VDU23;8202;0;0;0;:RETURN
  420 MODE1:VDU23;8202;0;0;0;:RETURN
  430 :
  440    :::::
  450 :
  460 DEFPROCinitialise
  470 DIMreel$(3),hold(3)
  480 VDU23,224,-1,-1,-1,-1,-1,-1,-1,-1
  490 ENVELOPE1,3,0,0,0,0,0,0,121,-10,-5,-2,120,120
  500 reel$(1)="0":reel$(2)="0":reel$(3)="0"
  510 top$=STRING$(13,CHR$224)
  520 space$=CHR$224+"   "+CHR$224+"   "+CHR$224+"   "
+CHR$224
  530 open$=CHR$224+" "
  540 mid$=" "+CHR$224+" "
  550 close$=" "+CHR$224
  560 hold(1)=FALSE
  570 hold(2)=FALSE
  580 hold(3)=FALSE
  590 dble$=CHR$141
  600 randomise=RND(-RND(10000))
  610 whtb$=" "+CHR$157
  620 blue$=CHR$132
  630 red$=CHR$129
  640 ylwb$=CHR$131+CHR$157
  650 cyan$=CHR$134
  660 grn$=CHR$130
  670 ENDPROC
  680 :
  690 DEFPROCtitle
  700 PRINTTAB(11,10)dble$;"ONE-ARMED BANDIT"
  710 PRINTTAB(11)dble$;"ONE-ARMED BANDIT"
  720 PRINTTAB(11,12)dble$;"by Paul Barbour"
  730 PRINTTAB(11)dble$;"by Paul Barbour"
  740 null=INKEY(300)
  750 ENDPROC
  760 :
  770 DEFPROCintro
  780 CLS
  790 PRINTwhtb$;blue$;dble$;"   ''''' ONE-ARMED BANDI
T ''''''"'whtb$;blue$;dble$;"   ''''' ONE-ARMED BANDIT
''''''"
  800 PRINTylwb$;red$;"by Paul Barbour - (c) LASERBUG
1983"
  810 PRINT'cyan$;"Press";grn$;"P";cyan$;"to pull the
handle"
  820 PRINTcyan$;"Press";grn$;"W";cyan$;"to collect yo
ur winnings (i.e."'cyan$;"to end the game)"
  830 PRINT'cyan$;"It costs 10p per go"
  840 PRINT'cyan$;"A zero in any window wins 10p"
  850 PRINTcyan$;"A pair wins 20p"
  860 PRINTcyan$;"Three of a kind (3 0/A/K) wins 50p a
s"
  870 PRINTcyan$;"does a Run i.e. 123,  765 etc."
  880 PRINTcyan$;"Three 0's wins '1.00 !!!"
  890 PRINT'''cyan$;"How much money do you want to put
 in"'cyan$;"the machine (in pence and no more than"
  900 PRINTcyan$;"'1 please. ?";grn$;
  910 INPUT""cash
  920 IFcash>1000Rcash<=9THENPRINTTAB(14,20);SPC(40);:
INPUTTAB(14,20)""cash:GOTO920
  930 cash=((INT(cash/10))*10)/100
  940 start=cash
  950 ENDPROC
  960 :
  970 DEFPROCmachine
  980 VDU19,0,4,0,0,0
  990 FORoff=1TO3:VDU19,off,4,0,0,0:NEXT
 1000 COLOUR2
 1010 PRINTTAB(15,3)"ONE-ARMED"
 1020 PRINTTAB(17)"BANDIT"
 1030 PRINTTAB(13,7)top$
 1040 PRINTTAB(13)space$
 1050 PRINTTAB(13,9)open$;:COLOUR3:PRINTreel$(1);:COLO
```

```
UR2:PRINTmid$;:COLOUR3:PRINTreel$(2);:COLOUR2:PRINTmid
$;:COLOUR3:PRINTreel$(3);:COLOUR2:PRINTclose$
 1060 PRINTTAB(13)space$
 1070 PRINTTAB(13)top$
 1080 PRINT''TAB(12)"0 ANYWHERE  10p"
 1090 PRINTTAB(12)"2 PAIR      20p"
 1100 PRINTTAB(12)"3 O/A/K     50p"
 1110 PRINTTAB(12)"RUN         50p"
 1120 PRINTTAB(12)"0 0 0    '1.00p"
 1130 PRINT'''TAB(12)"PRESS P TO PULL"
 1140 PRINTTAB(17)"HANDLE"
 1150 PRINT'TAB(14)"OR W TO TAKE"
 1160 PRINTTAB(16)"WINNINGS"
 1170 PRINT''TAB(15)"10p per go"
 1180 GCOL0,1
 1190 FORX%=0TO50STEP10
 1200   MOVE320+X%,0+X%:DRAW320+X%,1023-X%
 1210   DRAW924-X%,1023-X%:DRAW924-X%,0+X%
 1220   DRAW320+X%,0+X%
 1230   NEXT
 1240 MOVE924,288:DRAW1024,288
 1250 MOVE974,268:DRAW974,888
 1260 VDU5
 1270 MOVE958,888:VDU224
 1280 VDU4
 1290 VDU19,1,6,0;0;19,2,7,0;0;19,3,5,0;0;
 1300 ENDPROC
 1310 :
 1320 DEFPROCcash
 1330 COLOUR2
 1340 PRINTTAB(0,15)"You have:"
 1350 COLOUR3
 1360 cash=(INT((cash*100)+.5)/100)
 1370 accurate=INT((cash-INT(cash))*10)
 1380 cash=INT(cash)+((accurate*10)/100)
 1390 pounds=INT(cash)
 1400 pence=INT((cash-INT(cash))*100)
 1410 IFpence=0THENpence$="00":GOTO1430
 1420 pence$=STR$(pence)
 1430 PRINT"£";pounds;".";pence$;"p   "
 1440 ENDPROC
 1450 :
 1460 DEFPROCpull
 1470 REPEATA$=INKEY$(0):UNTILA$="P"ORA$="p"ORA$="W"OR
A$="w"
 1480 IFA$="W"ORA$="w"THENVDU22,7,23;8202;0;0;0;:HIMEM
=&7C00:PROCwinnings
 1490 cash=cash-.1:PROCcash
 1500 VDU5
 1510 FORX%=888TO368STEP-8
 1520   SOUND0,-10,3,1:SOUND1,0,(X%-368)/2.03921569,1
 1530   GCOL0,1
 1540   MOVE958,X%:VDU224
 1550   GCOL0,0
 1560   MOVE958,X%:VDU224
 1570   NEXT
 1580 FORX%=368TO888STEP32
 1590   SOUND0,-10,3,1:SOUND1,0,(X%-368)/2.03921569,1
 1600   GCOL0,1
 1610   MOVE958,X%:VDU224
```



```
 1620   MOVE974,X%:DRAW974,268
 1630   GCOL0,0
 1640   MOVE958,X%:VDU224
 1650   NEXT
 1660 GCOL0,1
 1670 MOVE958,888:VDU224
 1680 MOVE974,888:DRAW974,268
 1690 *FX15
 1700 VDU4
 1710 ENDPROC
 1720 :
 1730 DEFPROCspin
 1740 FORX=1TO10
 1750   IFhold(1)=FALSE THENreel$(1)=STR$(RND(10)-1)
 1760   IFhold(2)=FALSE THENreel$(2)=STR$(RND(10)-1)
 1770   IFhold(3)=FALSE THENreel$(3)=STR$(RND(10)-1)
 1780   COLOUR2:PRINTTAB(13,9);CHR$224
 1790   PRINTTAB(13,9)open$;:COLOUR3:PRINTreel$(1);:SO
UND&11,1,116,25:COLOUR2:PRINTmid$;:COLOUR3:PRINTreel$(
2);:SOUND&12,1,108,25:COLOUR2:PRINTmid$;:COLOUR3:PRINT
reel$(3);:SOUND&13,1,100,25:COLOUR2:PRINTclose$
 1800   PROCdelay(10)
 1810   NEXT
 1820 hold(1)=FALSE
 1830 hold(2)=FALSE
 1840 hold(3)=FALSE
 1850 ENDPROC
 1860 :
 1870 DEFPROCwin
 1880 IFreel$(1)="0"ANDreel$(2)="0"ANDreel$(3)="0"THEN
```

```
win=1:PROCfanfare(255):GOTO1990
 1890 IFreel$(1)=reel$(2)ANDreel$(2)=reel$(3)THENwin=.
5:PROCfanfare(150):GOTO1990
 1900 reel1=VAL(reel$(1))
 1910 reel2=VAL(reel$(2))
 1920 reel3=VAL(reel$(3))
 1930 IFreel1=(reel2+1)ANDreel2=(reel3+1)THENwin=.5:PR
OCfanfare(150):GOTO1990
 1940 IFreel3=(reel2+1)ANDreel2=(reel1+1)THENwin=.5:PR
OCfanfare(150):GOTO1990
 1950 IFreel$(1)=reel$(2)ORreel$(2)=reel$(3)ORreel$(1)
=reel$(3)THENwin=.2:PROCfanfare(50):GOTO1990
 1960 IFreel$(1)="0"ORreel$(2)="0"ORreel$(3)="0"THENwi
n=.1:PROCfanfare(10):GOTO1990
 1970 ENDPROC
 1980 PROCdelay(50)
 1990 FORbell=0TOwin STEP.1
 2000   SOUND1,1,200,10
 2010   NEXT
 2020 cash=cash+win
 2030 ENDPROC
 2040 :
 2050 DEFPROCdelay(time)
 2060 TIME=0:REPEATUNTILTIME=time
 2070 ENDPROC
 2080 :
 2090 DEFPROCfanfare(pitch)
 2100 FORXX=0TOpitch
 2110   SOUND&11,1,XX,1:SOUND&12,1,XX,1:SOUND&13,1,XX,
1
 2120   NEXT
 2130 ENDPROC
 2140 :
 2150 DEFPROChold
 2160 IFRND(10)<>1THENENDPROC
 2170 PRINTTAB(12,21);"---------------"
 2180 PRINTTAB(12,22);"------HOLD-----"
 2190 PRINTTAB(12,23);"---------------"
 2200 FORreel=1TO3
 2210   PRINTTAB(12,25);"Hold reel ";reel;"   ?"
 2220   PRINTTAB(12,26);SPC(12)
 2230   PRINTTAB(14,27);"(""Y"" or ""N"")"
 2240   REPEATA$=GET$:UNTILA$="Y"ORA$="N"
 2250   IFA$="Y"THENhold(reel)=TRUE:PROCfanfare(5):ELS
EPROCfanfare(1)
 2260   NEXT
 2270 COLOUR2
 2280 PRINTTAB(12,21)"               "
 2290 PRINTTAB(12,22)"PRESS P TO PULL"
 2300 PRINTTAB(12,23)"     HANDLE    "
 2310 PRINTTAB(12,25)"  OR W TO TAKE "
 2320 PRINTTAB(16,26)"WINNINGS"
 2330 PRINTTAB(12,27);"              "
 2340 ENDPROC
 2350 :
 2360 DEFPROCwinnings
 2370 PRINTcyan$;"Had enough then ?  Well,  in the tim
e"
 2380 PRINTcyan$;"you were playing you managed to make
"
 2390 cash=(INT((cash*100)+.5)/100)
 2400 accurate=INT((cash-INT(cash))*10)
 2410 cash=INT(cash)+((accurate*10)/100)
 2420 pounds=INT(cash)
 2430 pence=INT((cash-INT(cash))*100)
 2440 IFpence=0THENpence$="00":GOTO2460
 2450 pence$=STR$(pence)
 2460 PRINTTAB(16);dble$;grn$;"'";pounds;".";pence$;"p
"'TAB(16);dble$;grn$;"'";pounds;".";pence$;"p"
 2470 PRINTcyan$;"The difference between when you star
ted"cyan$;"and finished is ";cash-start
 2480 PRINT''''ylwb$;red$;dble$;"Next person press RE
TURN to start"'ylwb$;red$;dble$;"Next person press RET
URN to start"
 2490 REPEATUNTILINKEY(0)=13:RUN
 2500 :
 2510 DEFPROCbroke
 2520 cash=(INT((start*100)+.5)/100)
 2530 accurate=INT((cash-INT(cash))*10)
 2540 cash=INT(cash)+((accurate*10)/100)
 2550 pounds=INT(cash)
 2560 pence=INT((cash-INT(cash))*100)
 2570 IFpence=0THENpence$="00":GOTO2590
 2580 pence$=STR$(pence)
 2590 PRINTcyan$;"Well,  the gambling bug proved too m
uch"cyan$;"for you and you have lost all '";pounds;"."
;pence$;"p !"
 2600 GOTO2480
 2610 DEFPROCerror
 2620 IFERR<>17THENREPORT:PRINT" at line ";ERL:END
 2630 VDU22,7,23;8202;0;0;0;:HIMEM=&7C00
 2640 PRINTcyan$;"If you can't take the pressure you"'
cyan$;"shouldn't of started - there's no need"
 2650 PRINTcyan$;"to press ESCAPE.  You can stop playi
ng"'cyan$;"but I'll keep all the money."
 2660 GOTO2480
```

competition results

Wasn't that March competition easy? The "code" was quite simply a list of ASCII codes which when decoded came up with the message "LASERBUG FOR THE BBC MICRO". So many of you came up with the right answer that rather than try to find the best program we stuck them all in a hat and came up with the winner who is Barry Kadleck from Bitterne Manor, Southampton. £5 is on its way.

The particular program Barry used was:

```
10 INPUT CODE$
20 FOR CHAR=1 TO LEN(CODE$)-1 STEp 2
30 PRINT CHR$(VAL(MID$(CODE$,CHAR,2)));
40 NEXT
50 PRINT
```

GOLEM LTD
COMPUTER SERVICES

77 QUALITAS · BRACKNELL · BERKSHIRE RG12 4QG · Telephone: BRACKNELL (0344) 50720

Laserbug
10 Dawley Ride
Colnbrook
Slough
SL3 0QH

7 April 1983

Dear Sirs,

Thank you for your magazine Issue 10 and your invoice.

We were generally pleased with the review but we think the review was not done by a teacher as we find teacher are particularly enthusiastic about CUBECOUNT. We object to the sentence describing Cubecount - "Sometimes this can involve pure guesswork and I can't really see how this program is educational???".This program teaches children visual perception, addition and multiplication. Pure guesswork should not be involved at all. It is like saying counting 2 + 2 involves pure guesswork!!!

Would you please suspend further advertising till further notice.

Yours faithfully,

Eva Koskuba

Directors: K. KOSKUBA E J KOSKUBA

Registered Office: 77 Qualitas Bracknell, Berkshire RG12 4QG. Registered in England No: 1570049
VAT Registration No: 314 8360 69

All the LASERBUG Softreviews are done by qualified people – all educational reviews are carried out by teachers. The review is purely one teachers experience with the program – the teacher conserned did test the program out with her own class and if one teacher dislikes this particular program on the tape the chances are that others will. The last sentence of your letter was interesting – I am afraid that we will not bias a software review for any reason even if the company is advertising. If you can't take the heat . . .

Dear LASERBUG,

A BBC Micro local user group is being formed in the Wakefield area and I would appreciate it if you would mention us in LASERBUG magazine. Anyone interested should contact me at 1 Wavell Garth, Sandal, Wakefield, West Yorkshire, WF2 6JP. Tel. Wakefield 25515 for further information.

Richard Sterry, Wakefield, West Yorkshire.

Dear LASERBUG,

Further to my telephone call concerning the alleged "Bug" in the Acornsoft Meteors program I would like to clarify a couple of points. Although Acornsoft refused to acknowledge that the program might contain a bug, at no time did they refuse to talk to me. I am still of the opinion that if an "Asteroid is not destroyed when a Missile appears to strike the leading face then the game is degraded. I would point out also that the "Space-ship" will often self destruct when materialising after a trip in "Hyper Space". It is because of these two malfunctions (not bugs?) that I do not think that the game compares favourably with other versions.

I have recently received a free 1.2 OS ROM from Acorn to replace the much maligned OS 0.1 ROM. Further, I found an Acorn dealer to fit (just in case) and test it without charge. I would add that my 0.1 OS was in ROM, not EPROM.

It should not be necessary for anyone to pay for the supply and installation of a series 1.0 OS ROM in order to bring their computer to specification. It is only necessary to point out to the Vendors that the computer does not meet the specification implied on page 4 of the BBC Information Sheet G2 (December 1981) i.e. that the computer will support four paged ROMs such as a Word Processor e.g. "Wordwise". The computer fitted with the 0.1 OS will not do this! In other words as I see it (as so incidentally does the local Trading Standards Office) any one possessing a BBC Micro fitted with a series 0.1 OS is entitled to a replacement 1.0 operating system free of charge! Free fitting too!

Michael Dunning, Potters Bar, Herts.

Dear LASERBUG,

I think your members should know that it **is** possible to get free 1.2 ROMs from Acorn. Write (if necessary 2 or 3 times) to Mrs. J.R. Pullen or try the BBC. Unfortunately this fact has not been made public and I hope you will do so.

ANONYMOUS???

Dear LASERBUG,

After spending many nights wrestling with Philosopher's Quest I decided to cheat and dig the keywords and phrases out of the program. Acornsoft had anticipated my insidious plans and scrambled their strings. I have written a program that unscrambles and prints out the required information in a little over 1 hour.

It probably isn't "proper" to publish the program in your magazine and I do not want to spend money advertising a program of such limited interest, so I have written in the hope that you will let people know it is available. I will supply a copy of the listing for 40p plus a SAE. If anyone has very lazy typing fingers then I could supply it on tape for £1 including postage and packing.

David Coldicott, 101 Bryanston Road, Solihull, W. Midlands, B91 1BP.

Dear LASERBUG,

BBC BASIC may well become a standard dialect and I am not sure anyway whether it would be wise to base a user group like LASERBUG on one high level language. For example, Joe W. Brown has recently provided an excellent implementation of FORTH for the Beeb on EPROM and I believe it is much more important to give space in the magazine to these exciting developments rather than expanding the coverage to other users of BASIC. With wider availability soon of "tube" extensions many other languages will probably become popular which are at present inhibited by the lack of storage capacity at the "terminal".

It seems to me that it is the Beeb architecture and operating system that make it particularly special within its price range and the excellent LASERBUG should go forward where these features lead it. For a start may I recommend that the Editor 'goes FORTH' as soon as possible. It takes quite a re-orientation if my experience is anything to go by, to realign oneself back to BASICs, even one with the structural features of the BBC formulation!

James Helder, Wokingham, Berks.

(If you care to take over the running of LASERBUG for a month then I will have time to go FORTH. Seriously though we do want to encourage a Languagespot article in LASERBUG – we have Nick Goodwin working on such a thing now. Instead of writing letters on FORTH and the like why not send us in articles on the subject? – Ed.)

Dear LASERBUG,

One of your articles – on tape recorders (Tape Recorder Talk – pages 5/6, Issue 10) – suggests that no company offers software on disk. Obviously you've been caught by the time lag between writing and getting it to members, but one thing that has impressed me is the amount of software already available on disk. My hunch is that a large percentage – very posibly most – BBC Micros in the next year will be disked based. I know one program I would very much like for my disks – a disk indexing program, It would be resident, with its files, on one disk. You could structure the records to include program or file name, disk name and number, keywords (subject, etc.), perhaps length – all to allow you to search for any relevant programs (very important for word-processing for us academics with all our notes on disk). The way it would work is you would initially load it into the BBC, take out the disk, and insert disks not indexed. The program would then automatically read the disk directory into itself (in the BBC memory), then you could change the disk and do all your disks. Then replace the original disk and it would let you add other details to each record, then save it on the resident disk.

So, I know what I want – but have no idea how to do it. Any ideas? Perhaps you could suggest it to any software genius you know.

Doug Weller, The University of Birmingham.

Dear LASERBUG,

It may be of interest to readers of the problems I encountered when using the RS 423 input on the BBC Micro. According to the

continued on page 23

continued from page 17

produces GREEN. Using CTRL–fn produces other codes. In short SHIFT–fo to SHIFT–f15 produces ASCII codes 128 to 143 and CTRL–f0 to CTRL–f15 produces ASCII codes 144 to 159. The only things to note are that f11–f15 only work when anabled with *FX4,2 (see later) and that SHIFT–f10 and CTRL–f10 will not produce any code as f10 is in actual fact the BREAK key and SHIFT–BREAK provides an auto-boot and CTRL–BREAK provides a hard reset. This SHIFT–fn has another use – more of which later.

The credit for the next function must go to David Prideaux for discovering it. If you press SHIFT–CAPS LOCK you get a reversed function i.e. the normal CAPS LOCK action occurs except that when you press SHIFT and a letter key you get the lower case letter instead. In other words when you have the SHIFT–CAPS LOCK function on if say you wanted to type DEFPROCone instead of typing DEFPROC then taking the CAPS LOCK off and entering in the one or even taking the CAPS LOCK off, typing DEFPROC with the SHIFT key to get upper case and entering one you simply use SHIFT–CAPS LOCK function and whenever you need lower case (which is likely to be no where near as often as you will need upper case) use the SHIFT key to obtain it. This is much easier because it is convenient to leave this function on all the time.

If you use a printer with your computer then Acorn has made your life slightly harder – or to be more precise are making things stricter. Before if you wanted to send a code to your printer without doing anything else you could use say VDU1,15. However, on the new OS you cannot send any characters to the printer unless it is enabled so now you would have to use VDU2,1,15,3.

Although still compatible with the 0.1, redefining characters has been altered on the 1.2. Now there are two "modes" you can be in – imploded character moode and exploded character mode. The default is imploded.

With imploded characters like on 0.1 you have 32 characters to redefine. However characters 128–255 are for redefinition. With the imploded mode character 128 is the same as 160 is the same as 192 is the same as 224 and so on. In other words if you redefine character 224 (as you would on 0.1) this also effects 128, 160 and 192. This is how the two OS are compatible here. Basically on the 1.2 you should only redefine 129–159 (in imploded mode). The changing of the character numbers may just seem awkward but now you can access the characters from the keyboard. How? Well, SHIFT–f0 CHR$128, SHIFT–f1 produces CHR$129 which in teletext mode is the code for RED but in the other modes produces the character you have defined.

The other mode is exploded where you can redefine all characters i.e. 32 – 255. If you only want to use characters 128–159 then no problem occurs but if you want to alter any other characters you "must leave the memory up to OSHWM + &3FF clear of his program". If you're an expert programmer you might understand this – if not (and people who redefine characters are sometimes quite novice) then just looking at that sentence from the user guide will give you a headache. Turn to page 431 and enter the two line program there. Disregard the last two digits and the first two. With the example given this leaves you with 0E00. Now, before you start entering your program enter PAGE=&0E00+&400:NEW. This will leave the room that the other characters need. If you have a disk system for instance you might be left with 1900 so use PAGE=&1900+&400:NEW. Without altering PAGE you might find that your program will crash for no apparent reason. You can switch between these two modes with the use of *FX21 – more later.

One problem you will have with the new OS is the difference in memory locations. You may remember that if you entered ?216=32 this affected the CAPS LOCK/SHIFT LOCK functions. ?216 on the 1.2 produces funny effects with the prompt. On the 1.2 the memory location is 602 instead of 216. Full details are:

?602=0	CAPS LOCK/SHIFT LOCK on
?602=16	SHIFT LOCK on
?602=32	CAPS LOCK on
?602=48	Both off
?602=144	Reverse SHIFT LOCK on (i.e. normal SHIFT LOCK function but old function restored whilst SHIFT key is pressed down)
?602=160	Reverse CAPS LOCK on (i.e. normal CAPS LOCK function but lower case given when SHIFT is held down)

One of the other locations we have detailed in LASERBUG before is 875 which when set to 2 stops scrolling completely until reset by setting the location to 0. Try as we could we were unable to find an equivalent on the 1.2. If anybody can inform us of this location we would be extremely grateful.

Steven Mason of West London has informed us of some extra fill instructions which are not documented in the manual. These are 72–79 which the user guide says are "reserved". In actual fact these provide a much needed fill command. Unfortunately it isn't as good as it could be – on some home computers you just have to specify the point to start from and the colour and the whole area bounded by whatever shape is filled. On the BBC Micros fill command you have to write your own little routine. Steven sent us two programs which show the fill instruction working:

```
L.
   10 MODE5
   20 GCOL0,1
   30 MOVE100,100
   40 DRAW1000,100
   50 DRAW1000,1000
   60 DRAW100,1000
   70 DRAW100,100
   80 GCOL0,2
   90 FORY%=1000TO100STEP-4
  100   PLOT77,500,Y%
  110   NEXT
```

```
L.
   10 MODE5
   20 GCOL0,1
   30 MOVE640,1024
   40 FORX%=0TO360STEP15
   50   DRAWSINRAD(X%)*511+640,COSRAD(X%)*511+512
   60   NEXT
   70 GCOL0,2
   80 FORY%=1024TO0STEP-4
   90   PLOT77,640,Y%
  100   NEXT
```

Perhaps we will try to do a whole article on these fill commands another time if we find out more information on them.

On the 1.2 there are numerous FX commands that aren't available on the 0.1. Elsewhere in the magazine we have an article on the complete range of *FX commands known. You would be well advised to look through it as many are of use to everybody.

Well, looking through this article you might be pleased that you got a 1.2 ROM – you certainly have a number of new features to play around with. If you want to disassemble the 1.2 ROM you should look at locations &C000 – &FFFF. Hopefully over the coming months we'll be going into more detail about the more complex and interesting facilities you now have.

Paul Barbour

- P B Frazer,
  "Moon Hollow",
  Steeple Gidding,
  Huntingdon,
  Cambs. PE17 5RG
  0487 831342
- Michael Taylor,
  37 Hawthorn Crescent,
  Caddington, Nr. Luton,
  Beds. LU1 4EQ
  Luton 27585
- Peter Reay
  Romford 751954
- Dave Lister
  061 430 2060
- R R Henderson
  Ashley (063 087) 287
- Miss M C Dwyer,
  Fairmile,
  Haggs Lane, Cartmel,
  Grange over Sands,
  Cumbria LA11 6HD
  Cartmel (044 854) 450
- Tony Evans
  Penketh (092 572) 5078
- R W Darlington,
  1 Fells Grove,
  Worsley,
  Manchester M28 5JN
- A V Corker,
  The Old Farmhouse,
  Cavendish Road, Matlock,
  Derbyshire D54 3GY
  Matlock (0629) 57172

continued from page 21

manual it is a 5 pin DIN plug. Firstly it was very difficult to obtain. Radio Spares do not even stock it. However, Maplin do and keep ample stocks. Wiring these plugs is no problem but insertion is – the plug can go in two ways with possible spectacular results. one up for Acorn!

A Moulder, Rainham, Essex.

RS 423 plugs are available from CJE Microcomputers at 65p for two or from Happy Memories at 60p each. With the RS 423 cable that comes with Micronet, the gap in the plug should face the bottom of the computer.

Dear LASERBUG,

I am searching for routines to dump graphics screens from my BBC B to a Centronics 739 printer. I must have written 20 letters, all enclosing stamps, to receive about 2 replies, both negative. Can you make any suggestions? – I would be very grateful even to be pointed in the right direction. We don't all own Epsons!

A.I. Cameron, Dores, Inverness.

Dear LASERBUG,

Have you any information on connecting a Honeywell Terminett 300 Baud (serial) to the BBC B?

Mr. B.E. Gooch, Sidcup, Kent.

Can anybody out there help? We have no such suitable information (and don't even know what a Honeywell Terminett is!).

Dear LASERBUG,

Could you mention the Skye and Lochalsh Computing Society, of which I am Membership Secretary. The Society has members with various machines but the BBC Micro predominates. We are happy to give advice and information, with no obligation to join.

C.J. Manvell, Tigh na Pairc, 25 Breacais Iosal, Isle of Skye, IV42 8QA.

Dear LASERBUG,

On the subject of Micronet you may be interested in an idea a friend of mine Ian Gibbons came up with after owning his adaptor for a couple of weeks. Apparently the telephone handset does not fit exactly into the Micronet adaptor so he uses a strong rubber band to hold it in place. He reports that it has improved 'reception' of PRESTEL significantly. Just thought you might like to know. Keep up the good work with LASERBUG – it gets better every month.

Mark Cook, Braintree, Essex.

Dear LASERBUG,

As you have an Epson article this month (or last month now!) I thought I would make a small point about Nick Goodwins listing on page 15.

VDU1,27,1,82,1,0
would result in LDA#2 rather than LDA£2.
VDU1,27,1,82,1,3
restores the English version.

Alan Stoner, Coulsdon, Surrey.

You obviously have OS 0.1 – to send a control code to the printer on the series 1 OS you must enable it first so your VDU1,27,1,81,1,0 would become VDU2,1,27,1,18,1,0,3

classified

Got something to sell or buy? If so then LASERBUG classified Ads are for you. Rates are 10p/Word (minimum 15 words) or £10 per column centimetre (minimum 2 centimetres). Send your cheques/PO's made payable to LASERBUG to Classified Ads, LASERBUG, 10 Dawley Ride, Colnbrook, Slough, Berks., SL3 0QH. We cannot guarantee your ads will go in any particular issue but will do our best to ensure a prompt service.

credits

LASERBUG is edited by Paul Barbour.

The contributers this month were Paul Barbour, David Prideaux, Ralph Erskine and Patrick Dowling.

Articles and programs are always welcome. Please make sure that your work is original and has not been copied from elsewhere nor submitted to any other organisation. Payment for articles is at the rate of £5 or £10 per contribution depending on content, etc. Other rates can be negotiated for work of high standard. All contributions should be typed or computer printed with double spacing and at least a 1" margin. Hand written material may be subject to delay and error. All programs longer than 10 lines should be submitted on cassette both at 1200 and 300 baud. If a listing is supplied the computer should be set to WIDTH34 and LIST07 for 80-column listings. The first lines of the program should be REMed in the normal LASERBUG standard.

All articles published in LASERBUG become the property of LASERBUG, as does the copyright.

LASERBUG is typeset and printed by Prestige Press (UK) Ltd., East Street, Chesham, Bucks.

All correspondence should be sent to LASERBUG, 10 Dawley Ride, Colnbrook, Slough, Berks, SL3 0QH. If in difficulty phone 02812-3064 during office hours only. Prestel/Micronet Mailbox Number 919991005.

Advertising rates available on request.

ISSN 0264-5297

laserbug orders

MEMBERSHIPS

Membership to LASERBUG is £12.00 for one year or £6.00 for 6 months. This includes 12 (or 6) copies of the magazine and a membership card enabling you to take advantage of our many special offers. Overseas subscriptions are also available at both surface and air mail rates. Forces memberships are at UK prices.

MS01 UK 12 Month Membership £12.00
MS02 UK 6 Month Membership ..... £ 6.00
MS03 Overseas Surface Mail Membership £14.00
MS04 Postal Band A Air Mail Membership £18.00
MS05 Postal Band B Air Mail Membership £20.00
MS06 Postal Band C Air Mail Membership £22.00

DUST COVERS

There is nothing worse than an unprotected BBC Micro – solve this problem by buying some LASERBUG dust covers. We have dust covers for the BBC Micro, Microvitec Monitor, Epson MX80 Printer and the BBC Single Disk Drives available in PVC. We also stock Polyester/Cotton covers for the BBC Micro only.

DC01 BBC Micro PVC Cover ..... £4.00 + 20p P&P
DC02 Microvitec PVC Cover ..... £5.00 + 30p P&P
DC03 MX80 PVC Cover ..... £4.00 + 20p P&P
DC04 BBC Single Disk Drive Cover ..... £3.00 + 20p P&P
DC05 BBC Micro Polyester/Cotton Cover ..... £3.50 + 20p P&P

BACK COPIES

Missed out on a copy of LASERBUG? If so then use our back copy service and catch up on your reading. We have stocks of all magazines except issue 1 which is being re-printed shortly. A complete index to articles was given in issue 11 but briefly:

BC01 Issue 1 (16 pages) BEING REPRINTED SHORTLY
BC02 Issue 2 (16 pages) Sound/Envelope, *FX Part I, Teletext Part II, User Definable Keys, Pontoon
BC03 Issue 3 (24 pages) Programmers Corner, *FX Part II, Epson Screen Dump, Softreview, Telesoftware
BC04 Issue 4 (20 pages) Hardspot, What Printer?, Bookreview, Oddspot, Pixel Power, Moving Things
BC05 Issue 5 (32 pages) Wordprocessor, Machine Code, Disassembler, Seikosha Screen Dump, RGB Conversions
BC06 Issue 6 (24 pages) Software Protection Part I, Puzzle Program, How To Use Joysticks, Queryspot
BC07 Issue 7 (24 pages) Software Protection Part II, Alphabet Tester, Questionnaire Results, Club Reports
BC08 Issue 8 (24 pages) Memory Analyser, Assembler Programming On The BBC Micro Part I, Diskspot
BC09 Issue 9 (24 pages) Wallball, ADC Corner, Make The Most Out Of Sound, Grand Prix, Nine Dice
BC10 Issue 10 (24 pages) Tape Recorder Talk, Instant Memory Scan, Software Protection Part III
BC11 Issue 11 (28 pages) Micronet Review, Epson In Depth, Arcade Game High Scores, Year 1 Index

Each back copy costs £1.25 + 20p P&P.

BOOKS

Interface have kindly allowed us to offer you a discount on both books they sell for the BBC Micro. The first is Making The Most Of Your BBC Micro by Tim Hartnell and the other is the BBC Micro Revealed by Jeremy Ruston. Both books have been reviewed in LASERBUG.

IN01 Making The Most Of Your BBC Micro by Tim Hartnell ..... £5.50 + 40p P&P
IN02 The BBC Micro Revealed by Jeremy Ruston ..... £6.50 + 40p P&P

CASSETTE LEADS

Desperately need a cassette lead? There's no need to pay £5 for one when LASERBUG can offer their own cheap cassette leads to you. We sell both kinds of leads for compatibility with your personal tape recorder.

CL01 7 pin DIN – DIN/Remote Jack ..... £2.50 + 20p P&P
CL02 7 pin DIN – 3 Jacks ..... £2.50 + 20p P&P

1.2 ROMs

MEMBERS ONLY. If you are prepared to upgrade your computer yourself, LASERBUG can supply you with 1.2 ROMs for only £5.50 + postage – the cheapest price anywhere. They come with full fitting instructions and require no soldering. If you fit the chip and your computer doesn't work properly afterwards you can take your micro to the local Acorn dealer and he will check it over for you for the fee of £6.90. If the chip is found to be faulty you will not be charged but if you have damaged your computer the dealer will fix it but charge you for the repair at normal rates. We cannot guarantee a delivery date as we are dependant on Acorn for supplies.

RM01 1.2 ROM and fitting instructions ..... £5.50 + 20p P&P

BINDERS

Need something to hold your valuable LASERBUGs in? If so then purchase a LASERBUG binder. Each binder holds 12 issues and comes with the LASERBUG logo attractively printed on the front.

BD01 LASERBUG Binder ..... £4.00 + £1.00 P&P

CODE	DESCRIPTION	PRICE	P&P
	SUB TOTAL		
		TOTAL	

NAME ..... ADDRESS .....

.....

MEMBERSHIP NUMBER ..... DATE .....

ALL ITEMS NORMALLY EX STOCK PLEASE ALLOW 7 DAYS FOR DELIVERY *

* Allow 4–6 weeks for 1.2 ROMs

THIS ORDER FORM SUPERCEDES ALL OTHERS

If you run out of room or do not want to cut up your LASERBUG you may order on a separate piece of paper or take a photocopy of this form.