

REFEEBUG

FOR THE BBC MICRO

B

WINDOWS
AND
ICONS

1



U

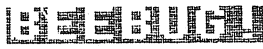
Type
1

E

Type
2

DEMO

INPUT DEMO
with $N\% = 7$



E

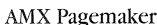
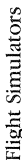
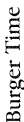
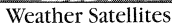
Window
type
3

G

Pointer
Sensitivity



B



BEEBUG

VOLUME 4 NUMBER 9
MARCH 1986

GENERAL CONTENTS

- 3 Editorial Jottings
- 4 Postbag
- 5 BEEBUGSOFT Forum
- 6 News
- 7 The Master 128 in Practice
- 10 Weather Satellites
- 13 AMXTRAS
Pagemaker, Super Art and 3D
Zicon
- 16 Windows and Icons
- 21 Adventure Games
- 22 Ramsafe
Dynamic Backup to Sideways RA
- 25 Preserving Screens
- 26 Points Arising
- 27 Getting a Better View
- 30 Angels at 12 o'clock
Three Flight Simulators
- 32 Games Reviews
- 34 BEEBUG Workshop
The 6522 Timers
- 36 Three Modems
- 39 Writing Your Own Compiler
(Part 3)
- 44 First Course
Using the EVAL Function (Part 2)
- 46 Burger Time

PROGRAMS

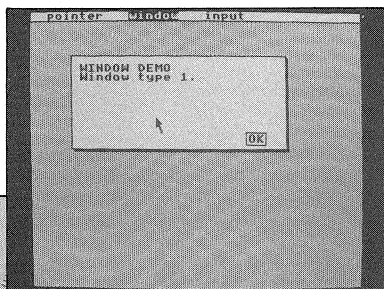
- 16 Windows and Icons
- 22 The USI Routines and Demo
- 22 Ramsafe Utility
- 25 Preserving Screens
- 34 Workshop Routines Using the 6522 Timers
- 39 Basic Compiler (complete)
- 44 First Course EVAL Example Programs
- 46 Burger Time Game

INFORMATION

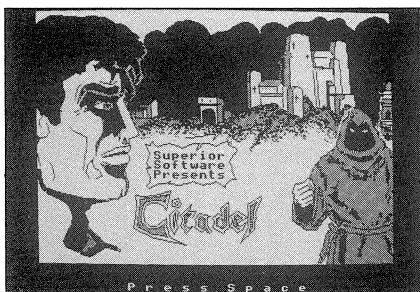
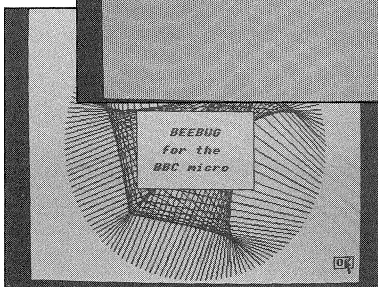
- 46 View or Wordwise?



The Master 128

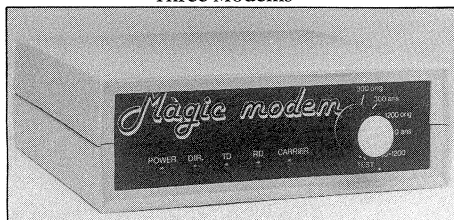


Windows and Icons



Games Reviews

Three Modems



EDITORIAL JOTTINGS

INCREASED SUBSCRIPTIONS

We very much regret that as a result of a continuing rise in our costs, the subscription rates to BEEBUG are being increased by £1, except for the six month UK subscription which goes up by 50p. The new rates apply to all subscriptions from April. We have managed to maintain the previous subscription rate since October 1984, a period of just over 18 months, and the increase now is in line with inflation over that period. Subscription rates to the magazine cassette and disc will continue to be held at existing levels.

BEEBUG MAGAZINE

We have had a number of letters and phone calls praising the Full Screen Editor published in the December issue (Vol. 4 No. 8). It is thus even more regrettable that we omitted to give the author of this excellent utility. He is John Cole.

The AMX Mouse has set new standards for user-friendly screen displays on the Beeb and in this issue we start a new series of articles all about the use of windows and icons in your own programs. Although the magazine series provides an excellent starting point, readers who are particularly interested in this type of presentation should consider BEEBUGSOFT'S Icon Master. This is not only a very comprehensive package, but by its own use of windows and icons, makes the designing of professional-style displays extremely easy.

ACORN MASTER SERIES

As promised, we have included a much more detailed look at the Master 128 in this issue. We have had a machine since early January and can only say that the more we get to know it the more we like it. Many visitors to our St Albans shop have expressed considerable interest in the Master series and we have a machine continuously on display for any member who would like to find out more. We are now eagerly anticipating the other models in the Master range.

PROGRAM CLASSIFICATION

All programs in the magazine, and on the magazine cassette/disc, are marked with the symbols shown below. An uncrossed symbol indicates full working, a single line through a symbol shows partial working for that configuration (normally some modifications will be required), and a cross through a symbol indicates a program that will not work on that type of system. There is a symbol for the B+ which includes the 128K version, and a symbol for the new Master series.

Basic I	I	Electron	⦿	Model B+	+
Basic II	II	Disc	⦿	Master 128	M
Tube	⦿	Cassette	⦿		



POSTBAG



POSTBAG

Speaking Logically Again

There is surely nothing wrong with the way in which Basic interprets the line in question ('Logically Speaking', Postbag, BEEBUG Vol.4 No.5). Since IF is in a multi-statement line, everything after the IF is conditional, and that includes the NEXT.

All that is required is to put NEXT on a separate line:

```
10 FOR J=1 TO 5:PRINT J,J M
OD2:IF J MOD2 PRINT"Odd" EL
SE PRINT"Even"
20 NEXT
```

When the NEXT is included in the same line, the first number, not being MOD2, never sees a NEXT and so the program stops.

N.P.Fay

Mr Fay is quite right in what he says, and his code offers a very good alternative to the solution given. The major point remains the undesirability of including key words like NEXT and UNTIL in isolation in IF-THEN-ELSE statements. Further, the example showed that the IF statement could be avoided altogether.

Differing Accounts

Following Mr Kidd's suggestions (Postbag, BEEBUG Vol.4 No.4) for your DOMAC program, the following alterations show the date the information was saved, and alter the date when new information is added.

Delete line 1050 PROCdate.
Amend line 2450 to read:

```
2450 VDU26,12:PROCdate:
CLS:PROCColour(1)
```

A date is then only

inserted when the data saving section is used.

Alan Wright

This was not the only letter received. It seems as though Mr Kidd has solved a minor problem for many users of DOMAC.

White Knight to the Rescue

I recently bought the disc version of White Knight Mk12 and to my horror I discovered that it will not run on my machine which has a Solidisk DFS 2.1 (1770). I get the dreaded message 'Bad Program'. Now, I cannot tell whether the program is corrupted in some way or whether the game and my DFS are incompatible.

I would be grateful if you could tell me what to do. I don't want my money back; I'd rather have a working version of the game.

Graham Conridge

We checked with BBC Soft and they say that they can supply Graham Conridge, or anyone else in the same situation, with a version of White Knight 12 that will work with the 1770 disc controller chip and associated DFS. Just send the disc, preferably in its original packing, to BBC Soft, 1-2 Marylebone High Street, London.

In fact, David Atherton, their Software Manager, told us that they are always prepared to try and supply a suitable working version of their software if the customer contacts

them direct. They cannot guarantee to do so in all cases, but will at least always refund the purchase price as a last resort.

Hard Words

I am writing to express my appreciation on completing the most recent EPROM Programmer Project (BEEBUG Vol.4 Nos.4 & 5). It is also quite gratifying to receive a magazine which continues with high standards of presentation, software and hardware.

My interests have now been directed to a modem. Can you recommend a kit or unit to satisfy the average user's requirements. Communications is generally neglected in the average book. Indeed, how about a regular hardware page?

J.Battersby

Despite some problems in publishing the articles, we have now heard from a good many members who have successfully built the EPROM Programmer we featured. For those who may still have some problems, we do have a set of useful test notes available on receipt of an A5 SAE.

The only modem we know of in kit form can be had from Maplins for about £50, but this is limited and not really recommended. For ready-built units, see our review in this issue, while the Pace Nightingale with Commstar was favourably reviewed by us about twelve months ago, and is available on special offer from BEEBUGSOFT.



BEEBUG SOFT FORUM

Masterfile Interface with Wordwise and View

Masterfile II integrates extremely well with the Wordwise, Wordwise Plus and View word processors. Standard Masterfile print-out and specialized form design layouts may be spooled to disc ready for loading directly into Wordwise or View, making the preparation of very attractive reports an easy matter.

Similarly, in the other direction, information created using Wordwise may be combined with the Masterfile printed output when using Masterfile's 'Form Design' option. This means that personalized letters, forms and records may easily be produced. For example, bills may be sent with the appropriate messages:

Mr. Smith, please pay
the outstanding amount
of £500.

or

Mr. Jones, £4126.83 is
now overdue.

or

Mr. Brown pay invoice
No. 1023 NOW or we'll
send the bills around.

Masterfile II B+ Modification

Certain versions of Masterfile II require a simple modification to enable this program to run successfully on the BBC model B+. This should be performed by typing in the following, exactly as given:

```
*ACCESS LF <Return>
LOAD"LF" <Return>
150 *DRIVE0 <Return>
SAVE "LF" <Return>
*ACCESS LF L <Return>
```

Screen Dumps

The easiest way to produce a screen dump from BEEBUGSOFT software such as Icon Master, Paintmaster, Design, or Teletext is to use the in-built save feature to save a copy of the required screen to disc or tape. Users of the Hershey Characters utility can save the screen display by incorporating a *SAVE command in their own program.

The saved screen may then be dumped to your printer by writing a simple Basic program. For example users of the Dumpmaster ROM could use the following program to print out a mode 2 screen previously saved from within Paintmaster:

```
10 MODE 2
20 *LOAD screen 3000
30 *PRINT P5
```

Line 10 should initiate the mode required, line 20 should contain the appropriate start address for the screen mode concerned, and line 30 should call your printer dump routine. The mode start address may be found from your User Guide,

the BEEBUG reference card, or by typing *H SCREEN if you have the BEEBUGSOFT HELP ROM.

Epson JX-80 Printer Dump

The Dumpmaster ROM has now been re-written to include the new Epson JX-80 colour printer and the Centronics GLP printer (not Epson compatible).

Existing users of Dumpmaster who wish to upgrade should return their ROM only with £5.00 to the St. Albans address.

Tube Compatibility

Both RomIt and Icon Master are now available in fully tube compatible versions.

Disc Errors

We receive a considerable number of queries from members confused by the unhelpful nature of the disc error messages issued by the DFS. Owners of the BEEBUGSOFT HELP II ROM will find an explanation of the most common disc error messages by typing *H DISC.

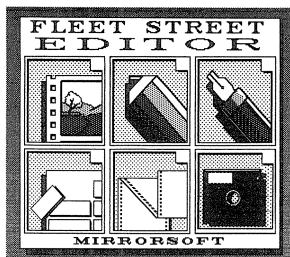
DFS Users Guide

A number of BEEBUG members have had their BBC micros upgraded with disc filing systems but have been given no information on how to use the new DFS commands at their disposal.

We have negotiated a special price on Cumana's excellent DFS User Guides. This usually costs £6.00 but it is available to members at the special price of £1.50. Details are on the order form.



News News News News News News Ne



Headline News

Hot on the heels of the AMX Pagemaker (see this issue) comes the long awaited Fleet Street Editor from Mirrorsoft. For just £39.95 this disc-based package gives you everything you need to produce your own professional-looking printed pages - covering all the skills of library research, graphics, copy, page make-up, and so on. A review of Fleet Street Editor will appear in next month's BEEBUG. Meanwhile, Robert Maxwell will answer all your questions on 01-377 4644.

Back at AMS things are hotting up with a new and improved version of Pagemaker. Pagemaker Plus has many new features and the ability to use the cursor keys or a joystick for control, enabling even mouseless Beeb owners to enjoy playing at being newspaper barons.

Power of Words

Members with a long memory will remember an early word processor for the Beeb called Wordsworth. Now the author, Ian Copestake has released an

updated version called Wordpower. Supplied on ROM for £39 or on disc (for 2nd processor owners) for £35, Wordpower offers the usual text editing features along with mail merging with Viewstore files, automatic backup, printer spooling, split screen, compatibility with the B+, Aries boards, Econet, and even the Electron, and document size limited only by the size of disc. Further details can be obtained from Ian Copestake on 04867-4755.

More Power

If your Beeb has a tendency to overheat under the strain of powering the two disc drives, ROM board, EPROM programmer, and other bits and pieces you may have connected to it, then Pace may have the answer for you. For £44.85 Pace will supply a separate power supply with four auxiliary power output sockets capable of powering a couple of separate disc drives and other equipment with no effort from your Beeb. Further details from Pace on 0274-488211.

Fly Catching

The Spider interface is a boon for computer control enthusiasts. Combining both sideways RAM and ROM software, the Spider board provides a set of process timers and practical real time input/output software tools. The budget Spider uses the Printer and User ports and costs £65. More extravagant web spinners can use the 1MHz bus with the £115 Spider. Further

details from the manufacturer, Paul Fray Ltd. on 0223-66529

Replications

Clares' tape-to-disc utility - Replica - is now available in yet another updated form - Replica III. The latest version costs £15 and promises to handle just about everything. Also from Clares, Fontwise gives your Epson compatible printer a range of several new and attractive letter styles, accessible from Basic or Wordwise. Fontwise costs £12 and Clares is on 0606-48511.

New Books

Latest in the ironically named BBC 'Master' series comes 'Mastering the Disc Drive'. All you ever wanted to know about using and abusing your disc drives for £7.95. If you want to master the written word then '15 Hour Word Processing' from the NEC is more in your line. This tutorial comes in two versions for Beeb users with View or Wordwise, each at £3.95.

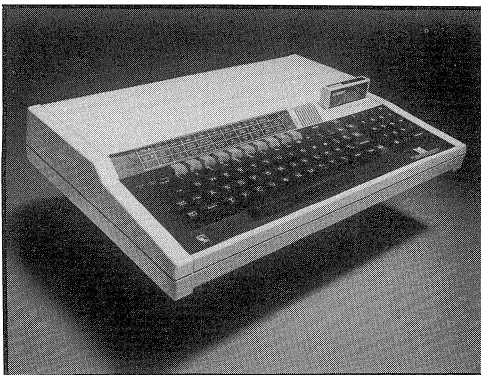
Adventure fans will be interested in 'Adventure Description Language' from Sigma Press. The book is really a description of a language for the BBC to enable less able programmers to write good adventures. The book covers defining the language, compressing string data, and writing the interpreter to run the game. All that will cost you £8.95.



THE MASTER 128

IN PRACTICE

Following its launch in January, the editorial team have been taking a close look at the Master 128, and now report on their experiences in using the new machine.



This month we take a look at the key features of the Master 128, concentrating on the differences and changes from the model B and its software systems. Even so, there is much more detail than we have room for. So at the end of this article we have listed those back issues of BEEBUG containing relevant articles and reviews (on the ADFS for example).

VIEW AND VIEWSHEET

View and Viewsheets as provided with the Master are, as far as one can tell, identical to the versions currently being sold for the B and B+, that is version 3.0 for View and version 1.0 for Viewsheets. Thus Viewsheets is as we reviewed it in BEEBUG Vol.3 No.3. View 3.0 has been available from Acornsoft since the Autumn of 1985 and has only a few differences from the previous version 2.1 (see article 'Getting a Better View' in this issue).

BASIC

A number of improvements and additions have been incorporated in Basic IV, the version used in the Master. A major factor in the new version is the obvious increase in speed as a result of using the 65C12 CPU chip instead of the standard 6502 as in the model B. Simple benchmark tests show a speed improvement of between 30 and 40 percent on what was already a notably fast version of Basic.

The language has a number of new keywords or constructions. For structured programming enthusiasts there is now a form of the 'case' statement:

ON <expression> PROC1,PROC2,PROC3,....
This is similar to ON-GOSUB, but allows branching to one of a series of named

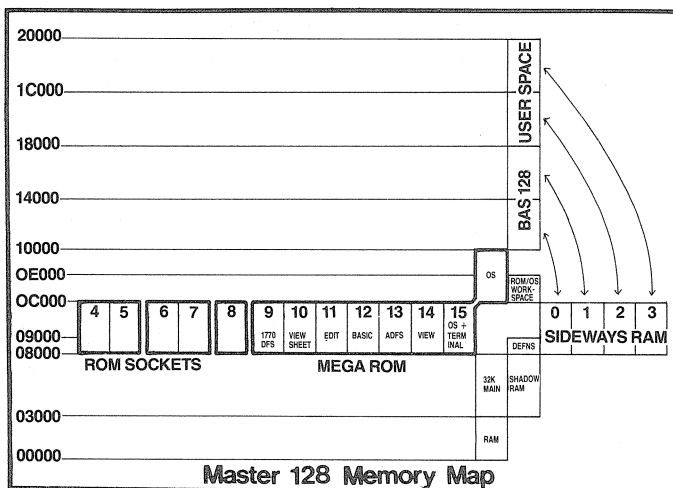
procedures depending on the value of the expression.

A new pseudo-variable, `TIMES`, allows the real-time clock to be read or reset from Basic. This provides, once it has been correctly set up, the time of day, date, and day of the week. Another new facility is `EXT#` (on the left of an expression) which allows a disc file to be extended to a new specified length.

For program development, `LIST IF <string>` allows listing of lines that contain a specified string. Line number ranges can also be used with this if required. The interpreter will now automatically strip any trailing spaces from a line entered or copied. Leading spaces will also be removed from lines entered while a non-zero `LISTO` command is in operation. Thus copying from a formatted listing will no longer increase the number of leading spaces. Along with this, `AUTO` no longer inserts a space after each line number as before.

There are a large number of new VDU commands available to Basic (and some old ones are officially documented for the first time) - see operating system section. A useful feature with many VDU commands is that the `|` character can be used at the end of a VDU string to insert the correct number of remaining zeros.

The built in assembler now incorporates all the additional instructions recognised by the 65C12 processor. It is the use of these additional machine code instructions by Basic that provides much of the speed increase referred to earlier.



Master 128 Memory Map

THE EDITOR

The Master comes with a built in editor (Edit) in addition to its wordprocessor (View). Edit provides no formatting capability and its prime function is the creation and editing of programs, though simple text files can be dealt with.

Once a program has been loaded or typed in using Basic, entering the editor automatically produces an equivalent ASCII file in the editor's work space, while Edit provides for a return to the language of your choice, converting the ASCII text to tokenised Basic if appropriate.

All the features you might expect for creating and editing programs are present and are controlled by the function keys as in View. Two aspects worth mentioning are the on-screen 'help' information, which can be reduced or removed altogether if you wish, and the use of a scrolling window, smaller than the displayed window, allowing program lines above or below the extreme cursor positions to be seen. Again this is under user control. Like View and Viewsheets, Edit will work in any mode and automatically uses shadow RAM.

OPERATING SYSTEM

The Master 128 has a new series 3 O.S. This incorporates many new features and enhancements. The user-defined character and function key workspace now resides in 12K of private sideways RAM leaving pages &B and &C free for the Econet workspace.

12K of private RAM is made up as in the table below.

The reason for the discontinuity in the addressing is so that the ROM workspace is not paged at the same address as the ROMs themselves.

The 12K private workspace is one of the best new features on the Master 128. It enables you to have two DFSs present without having to raise the value of PAGE from &E00. It gives you 4 times as much space for function key definitions and all of the 224 characters (from ASCII 31 to ASCII 255) can be re-defined without

exploding the character memory.

&8000 - &83FF	Soft-key buffer
&8400 - &87FF	VDU workspace
&8800 - &88FF	VDU variables
&8900 - &8FFF	Character definitions
&C000 - &DCFF	Paged ROM workspace
&DD00 - &DFFF	MOS workspace

The operating system now incorporates two new OSWORD routines to read and write to the battery-backed clock. On top of that there are new commands to configure the machine on power-up, show the contents of the function keys, list and disable ROMs and turn shadow memory on and off.

The operating system also has enhanced graphics capabilities (basically the Acornsoft graphic extensions without the sprites), and has room for even further extensions. The new extensions give the user facilities such as stippled fill, parallelograms and rectangles with solid fill, circles and arcs and ellipses with fast fill operation. It is also possible to disable scrolling when a character is printed in the final character position on the screen. The new features are accessed via Basic's VDU 23 and VDU 25 commands.

FILING SYSTEMS

There are principally four filing systems available on the Master. These are the standard Cassette and ROM filing systems found on any other BBC micro, and two DFSs, the 1770 DFS and the ADFS (see

UPGRADING TO MASTER CLASS

Following the launch of the Master series, around half a million owners of a BBC micro model B or B+ will be looking at their machine, whether it is new or old, in a new light. All would probably wish they now owned a Master.

The problem is that a Master 128 costs more than just a little money and the second hand market for old Beebes is not the most lucrative. So if you would like a Master, what chances are there for you to upgrade your present machine?

Several features of the Master put it in front of the old Beeb. Firstly, of course, there is the 128K. B+128K owners already have this feature and B+64K owners can easily upgrade for about £40. Model B owners will not find it so easy. Attaining 64K is easy enough with an Aries B-32 board (£92) or similar, but Aries has no plans at the moment for a B-96. As the further 64K on the 128 machines is in the form of sideways RAM, you could simply add a Watford ROM/RAM board to your Beeb (£70) but this would not give you the option of using this memory for Basic programs as you can with the Master and B+128. There is no Bas128 Basic available for this. Nor is even the loading and saving of sideways ROM 'images' handled in the same manner.

The Master's disc system can be easily imitated. B+ owners already have a 1770 disc interface and a DFS to suit. An ADFS can be bought as an upgrade for £30. However, model B owners must purchase a 1770 disc upgrade as well, for £50.

More difficult for existing Beeb owners is the real-time clock, 50 bytes battery-backed RAM and cartridge sockets of the Master. A real-time clock is electronically a simple add-on and models for the Beeb have appeared in the past for about £40. However, the cartridge sockets

are not available in any form (yet...).

Even with such a clock, your Beeb still won't have the software to look after it in the same way as the Master. You cannot use PRINT TIMES in a Basic program nor can you implement the start-up configuration system of the Master. You won't have the other extensions and improvements that make up Basic 4, either.

What is readily available is the other software that is included with the Master. View 3 (£90) and Viewsheet (£53) are available separately and Acornsoft's Graphics Extension ROM (£30) will give your Beeb the extra screen handling facilities of the Master. Acornsoft's Termulator (£30) will provide the terminal emulation facilities and the Basic Editor substitutes reasonably for the Master's Edit for £30 as well.

Despite the fact that the Master is (according to Acorn anyway) a 'new' machine, the old Beeb can be upgraded to a system remarkably close in specification. However, such a course is going to cost you a large amount more and still leave you short of the target.

Upgrade	model B	B+128K
BBC micro	£330	£460
Aries B-32	£ 90	-
Watford ROM/RAM 64K	£ 70	-
1770 disc interface	£ 50	-
ADFS	£ 30	£ 30
View 3	£ 90	£ 90
Viewsheet	£ 55	£ 55
Basic Editor	£ 30	£ 30
Termulator	£ 30	£ 30
GXR	£ 30	£ 30
Total	£805	£725
(All prices approximate)		

the recent review in BEEBUG).

The introduction of sideways RAM also adds new commands to the operating system. *SRLOAD loads a file into sideways RAM and *SRSAVE saves the sideways RAM to a specified file. Other commands allow areas of sideways RAM to be reserved for RAM filing, or for Basic 128.

For further information see the following back issues:

Aries B-32	Vol.4 No.8
64K upgrade for B+	Vol.4 No.7
1770 disc interface	Vol.4 No.7
ADFS	Vol.4 No.7
Viewsheet	Vol.3 No.3
Basic Editor	Vol.4 No.4
GXR	Vol.4 No.7

WEATHER SATELLITES

Weather satellites have become familiar to us through the pictures shown in TV weather forecasts. Now, the BBC micro user can get in on the act at modest cost. Mike Williams reveals all.



Meteosat infra-red picture

Since the first Sputnik was launched back in the fifties, the use of satellites has developed enormously. Pictures from weather satellites in particular have become a familiar sight in TV weather forecasts. In the past these pictures have required very expensive hook-ups to be received. Now the home BBC micro owner can join in the fun. If you're bored with other applications and seeking new worlds to conquer why not investigate satellites?

In fact, owning any kind of micro is not essential to receive satellite broadcasts, but it does pave the way, and with the help of suitable software provides flexible control of the signals received, and in many cases the opportunity to store pictures for display and further processing. The Beeb's own memory and its various ports under software control can replace much of the hardware of more professional systems, providing opportunities for weather satellite reception at a much lower cost. First of all, we'll take a look at the types of weather satellite which can be readily received by amateurs.

WEATHER SATELLITES

Weather satellites have been in operation since 1960, sending back to Earth pictures of the latest cloud formations and other information. From the beginning the same APT (Automatic Picture Transmission) system has been used with only negligible changes, and is used not only

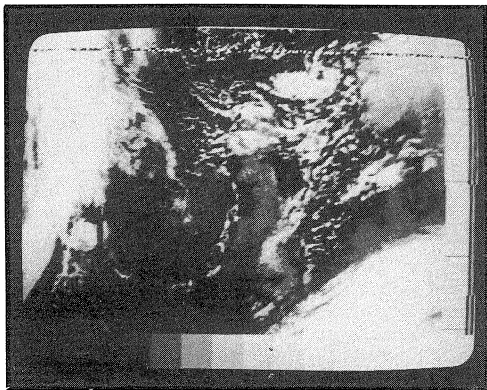
by American and West European satellites, but also those launched by the Soviet Union. Nowadays, two different types of weather satellite are in use, both receivable by the amateur. These are the polar-orbiting satellite and the geostationary satellite.

Polar-orbiting satellites (for example the US NOAA8 and NOAA9) have nearly circular orbits at a height of approximately 850km passing over both the north and south poles. Such a satellite orbits the Earth in a little over 100 minutes and in that time the Earth will have rotated on its axis by about 25 degrees. As the satellite moves overhead, it continuously scans a strip of ground below. On successive orbits the strips overlap, least near the equator, most near the poles. Thus every point on the Earth's surface is scanned at least twice every 24 hours.

At the latitude of the British Isles it is usually possible to pick up at least two consecutive passes giving at least four passes per day, sometimes as many as six. Not all passes will be overhead: some will take place at a lower angle with a consequent loss of signal strength. On each pass, the satellite transmits both a visible-light and an infra-red picture (at a nominal frequency of 137.5MHz), the receiver displaying both side by side, or just one as selected.

Geo-stationary satellites have very high orbits (36,000km above the Earth) giving an orbital period of 24 hours, equal to the Earth's own period of rotation. Such a satellite, positioned above the equator, thus appears stationary to an earthbound observer. In Europe, it is Meteosat 2 which can be received, and this is situated above the equator just off the west coast of Africa. Other geo-stationary satellites cover other parts of the globe.

From Britain, Meteosat can be received 24 hours a day. Unlike polar-orbiting satellites, Meteosat transmits complete pictures every four minutes according to a fixed schedule, and at the much higher frequency of 1694.5MHz. The pictures cover visible light, infra-red and water vapour (at different times) but the infra-red picture which includes the British Isles is transmitted regularly twice an hour. However, from a height of 36,000km above the equator, the quality of received pictures is not as good as with the NOAA series of polar-orbiting satellites. This is despite the fact that Meteosat pictures are first transmitted to a computer at the European Space Agency for processing, and adding landmass outlines, then transmitted back to the satellite and broadcast for general reception (10 minutes later). Pictures from polar-orbiting satellites are received direct.

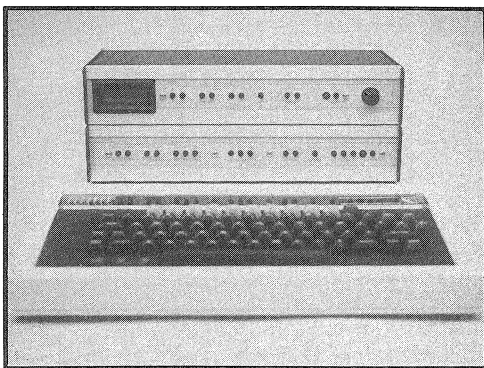


NOAA infra-red picture

Infra-red pictures from either type of satellite are often preferable, giving good pictures of cloud formations at night and during the winter periods of low visible light.

RECEIVING WEATHER PICTURES

The weather pictures transmitted by the orbiting satellites provide the best results, and these transmissions are also the easiest and cheapest to receive. For this you need a fairly simple fixed aerial that can be fixed reasonably high up (I found a flat garage roof quite adequate), a receiver and a decoder/interface.

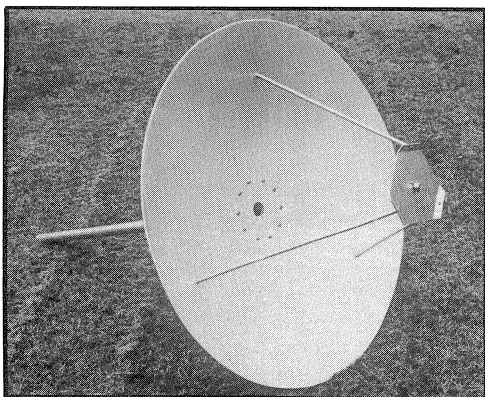


Timestep receiver and decoder

Timestep Electronics sell an excellent package for this purpose that will get you started for £276 inc VAT. This includes all the hardware, including aerial, fixed band receiver, decoder/interface and excellent controlling software in ROM format. The main units can also be purchased more cheaply as pre-tested modules (for boxing, fitting and connecting yourself), or in complete kit form. Although much cheaper, the latter is not recommended unless you are very competent at constructing electronics kits and have a certain amount of test gear available. Timestep say that if you need to enquire whether this is suitable for your skills then the answer is quite definitely 'No'.

This setup will allow you to receive pictures from NOAA8 or NOAA9 for display and storage. If you want the added fun of tuning in to different satellites, including some of the Russian ones, then you will need a scanner costing a further £109 (if purchased at the same time). A scanner also gives you the exciting opportunity of being perhaps the first to pick up any new and unknown Russian satellite.

If you want to receive Meteosat, then you will need a complete dish antenna and converter (to change the frequency of the



Typical dish antenna for Meteosat

received signal to that accepted by the existing receiver). This is comparatively expensive and will cost a further £335 if ordered at the same time, slightly more if added later.

Another company offering equipment for weather satellite reception is Feedback Instruments Ltd. Their equipment is extremely professional, very easy to use and well supported by excellent documentation. Options are again available for receiving NOAA or Meteosat signals as before. The main difference between the two companies is that Feedback offers a complete self-contained weather satellite receiver system that provides high resolution displays.

The Feedback system can optionally be provided with a computer interface which can be connected to the BBC micro allowing the micro to control most of the receiver's functions, and to transfer pictures from the receiver, which has its own picture memory, to the micro for storage on floppy disc. Such images can then be further processed by computer if required. Such high quality professional equipment does not come cheap with a price tag of over £2000 for a basic system. Regretably, the software supplied is neither very flexible nor easy to use, and in my view is poorly written. Apart from these criticisms, the Feedback system is a highly professional product.

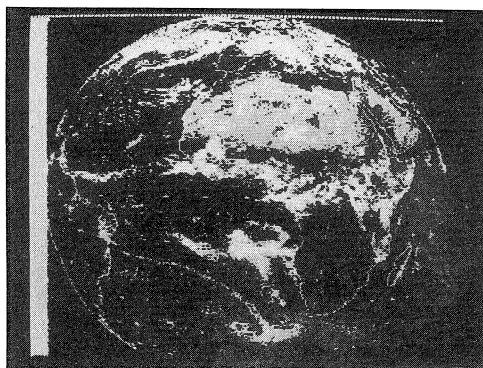
PICTURE QUALITY

Both types of satellite broadcast pictures of a size and resolution that cannot

be fully displayed on a typical home monitor. For example, a Meteosat picture consists of 800 graphics lines each containing 600 pixels. Instead, the pictures can either be condensed or just part of the picture displayed. The Timestep system provides a certain degree of control through the ROM software, while the Feedback system gives very complete control through various switches and counter settings.

With polar-orbiting satellites this is much less of a problem, and it is really just a question of displaying both visible and infra-red pictures together, or selecting just one of these for display.

Both Timestep and Feedback appear to provide good customer support, and will provide information on Meteosat schedules and pass times of the NOAA satellites. Indeed, Timestep operate a viewdata system (Tel. 0440-820002) giving this, plus other news and information on satellites, and details of Timestep products.



Whole Earth view from Meteosat

I can only say that I found the whole experience quite fascinating, and, given the time and opportunity to search out more satellites, an application of real interest. If you want to find out more, the addresses and telephone numbers of both Timestep and Feedback are below.

Timestep Electronics Ltd.,
Wickhambrook, Newmarket, Suffolk CB8 8QA.
Tel. 0440-820040

Feedback Instruments Ltd.,
Crowborough, Sussex.
Tel. 08926-3322



< < AMXTRAS > >

AMX PAGEMAKER 16+8K EPROMS + 2 discs £49.95

The AMX mouse has proved to be such a versatile animal that many software houses have been keen to write programs for it. The latest batch of AMX releases only confirms the high quality of software and design which have become the company's hallmark.

The Pagemaker package, designed for anyone who wants to produce professional-looking newsletters, magazines, manuals or posters, is both their most complex and rewarding package to date. It consists of 24K of machine-code, held in two EPROMs, together with discs holding further programs, sample text fonts and page layouts. The package starts with a couple of nicely animated title and credit screens, and then displays a main menu of four graphics windows. These show the main modules in the program: setup, font designer, pagemaker and previewer.

The setup module adjusts the display on the screen and configures the program for one of eight printers, mostly Epson compatibles, and each drawn on screen when selected!

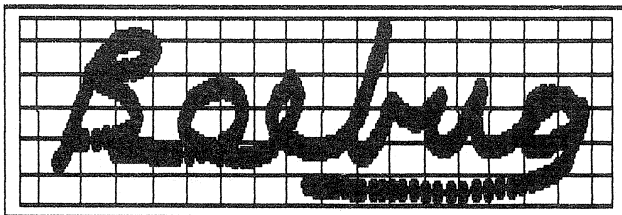
The font designer allows you to load in any of 15 pre-defined fonts, or design your own. The complete font is displayed actual size and clicking the mouse while pointing at a character will copy

it to the 64x64 design grid. Each font contains upper and lower case characters, numbers and punctuation, but no pound sign.

The main bulk of the work is done by the Pagemaker module itself. This is a very sophisticated program which displays everything in mode 0 - you'll need a monitor to do it justice. The screen shows roughly half of an A4 page but can be scrolled up and down and updated straight from disc, thus using the disc as a kind of slow RAM.

The facilities available for laying up your page are comprehensive. You can define a window of any size to take text, AMX art pictures (although not SuperArt ones) or digitised photos (if you have the necessary equipment) and everything will be kept within the bounds of the window. This means you can set up areas of the page independently of each other, define multiple columns and spread text around illustrations.

The text can be typed in directly onto the screen, or loaded from Wordwise or View files. Whichever technique is used, Pagemaker automatically word-wraps it and will justify, centre or display it literally, whichever mode is selected. It does this in whichever font is selected.



Products : AMX PAGEMAKER £49.95
: AMX 3D ZICON £24.95
: AMX SUPERART ROM £49.95
Supplier : Advanced Memory Systems Ltd,
Green Lane, Appleton,
Warrington WA4 5NG
Tel: 0924-413510

These three new AMX packages are reviewed by Simon Williams.

Fonts are called off disc as required and may be adjusted for height, width and vertical and horizontal spacing. The smallest text is the standard 80 column mode 0 script, and the largest is huge!

Graphics can be cut and pasted from one place on the page to another, mirrored, stretched, compressed and zoomed to pixel level for fine adjustments.

Once you've put together a page or two, and saved them onto a specially formatted disc, you can call the fourth module to view three pages on screen at once. The text is unreadable in this mode, but it's intended for looking at the overall layout. You can call further pages from disc to replace those displayed.

There are two qualities of print available, depending on how long you're prepared to wait. The title page of this review (which is reduced from an A4 original) was printed in medium quality.

The manual runs to 90 pages and is spiral bound. It's well laid out, but misses a full index, although the contents page is fairly detailed. I found it tricky to dip into until I knew most of the functions of the package.

Overall, Pagemaker is one of the most absorbing pieces of software I've used. Some people will have little use for it, but I suspect it could have all kinds of uses not yet thought of. A marvellous piece of software.

AMX 3D ZICON

The windows and icons environment lends itself to many different applications - indeed some new micros tailor their whole operating systems around the idea, but one application not often covered is 3D drawing.

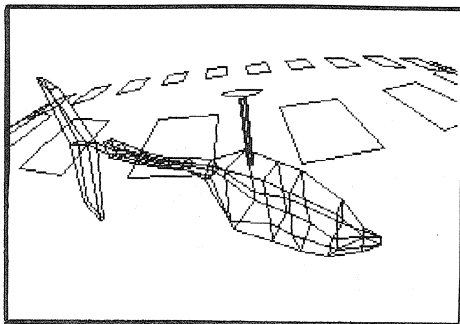
There are several reasons for this, one of the main ones being insufficient resolution on micros (even the likes of the BBC Micro) to produce any useful work. While jagged 'straight' lines may be acceptable for sketches, they're usually quite unacceptable for technical drawing. This accounts for the high price of dedicated Computer Aided Design (CAD) packages.

3D Zicon tries to get around some of these problems by providing a dump routine for the Penman plotter, which will draw straight lines, as well as for Epson compatible printers, which won't. How well does it fare in other departments?

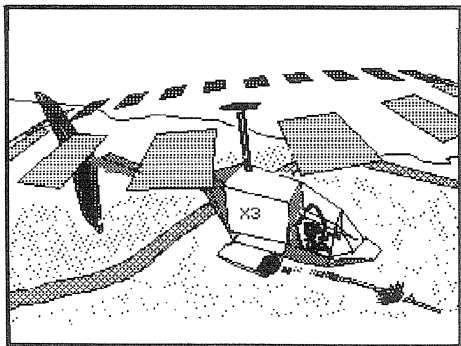
The package is disc-based and operates in mode 4 - the only way it can reserve sufficient space for the co-ordinate information. The facilities of the program are divided between four screens, all looking much the same but referred to as Black, White and Grey Zicon, and Zedit. You work using the mouse (or joystick/keyboard if using the SuperArt ROM), positioning points on the 3D screen which takes up the majority of the display.

Above and below this screen are strips of icons which allow you to rotate your drawing in any direction and to zoom in and out of the picture. Each move takes a while to recalculate, so things don't happen in real time.

Drawings are wire frame, with no hidden line removal (see BEEBUG Vol.4 No.6 for an example of hidden line removal), and with the standard BBC micro you are limited in the number of points you can enter. This printout of a helicopter gunship, taken from a demonstration file, is probably close to the limit.



With a 6502 second processor, however, there's a lot more room, and things speed up a bit, too. You can take 3D Zicon pictures and feed them into AMX Art to fill and embellish them. The second screen dump was produced in this way.



There are a couple of special features which make Zicon easy to use. The first of these is the 'GLUE' option. This allows you to create components of a drawing and then fasten them together to produce a finished object. Thus you could design the wings, fuselage and tail of an aircraft and then bring the modules together using GLUE.

The second feature is 'tubemaker', which does what its name suggests and generates tubes from their cross sections. This can be a great short-cut to producing 3D objects.

3D Zicon does its job well, but I'm left wondering where its market lies. It's a good package for manipulating 3D shapes but seems to offer little for producing dimensioned or scaled drawings. In the end, is it any more than an interesting introduction to the subject?

AMX SUPERART

Anybody who has bought the AMX mouse since its release over a year ago will have been impressed with the AMX Art package which accompanied the rodent. This package made use of a black and white mode 4 screen to display a work area together with windows for 'tools' (such as pencil, spray gun, paint roller and rubber), sizes

and textures of spray and a series of different stipples and patterns with which to fill drawings.

The package was operated entirely with the mouse, the only use of the keyboard being to type in file names. Later in the year, the same company introduced its utilities package, which added facilities to copy one area of a picture to another, to 'zoom' in and tidy up a picture pixel-by-pixel and to use icons and user-defined patterns in Art drawings. In addition there was a full-colour sketch program in mode 2.

What has all this to do with a review of SuperArt? Quite simply, AMX have integrated the Art and Utilities programs into one handy package, have moved the whole thing to mode 1 to offer four main colours and have put all the basic features into the ROM so you can use it in a stand-alone form.

Anything you could do with Art (except fills), can be done with SuperArt, and without even inserting a disc. If you do use the accompanying disc you get colour, stipple and pattern fills, all the other features of the utilities package and grey-scale or colour printer dumps. Unfortunately though, AMX now assume we all have Epson or Integrex printers. Loading the disc adds an extra pull-down menu to the top of the screen.

If you've designed programs using AMX ROM commands, these will work with the SuperROM. Ten new commands are added, however. These include *DEVICE, which lets you use the ROM with keyboard or joystick as well as mouse, *LOOP which will loop around a section of program until one of eight defined actions occurs, and *SPRAY, which will spray a pattern or colour onto the screen.

The manual is well illustrated and clearly laid out, but still lacks an index. A detailed contents is not nearly as useful.

If you use the original Art package, then you really should consider upgrading. If you haven't a mouse, it's worth buying SuperArt to use with joystick or keyboard. The man with the Mac looks pretty wet!

WINDOWS AND ICONS

Windows and Icons have become very popular with micro users. Alex Kang, in the first part of a new series, has written a set of routines that will help you write your own user-friendly programs with visual appeal.

The AMX mouse has shown how windows and icons may be used on the BBC micro to provide really user-friendly interfaces for many applications and utilities. Now, with the aid of the routines described and illustrated in these articles, you can provide your own programs with similar user-friendly displays, and no mouse or external ROM is needed either.

The User System Interface (USI) is a series of machine code routines designed to handle the creation of windows and icons, and the movement of a pointer around the screen. Once running, all selections can be made using the cursor keys and the space bar.

Although in machine code, the USI routines may be readily used within Basic programs as the demo programs will show, and no knowledge of machine code programming is required for their use.

This month's article contains the listing of the USI routines, and a complete demonstration program. As a consequence, we shall be concentrating on the demonstration program and the procedures it uses so that you may see the potential of the USI routines. Next month's article will show how to modify and extend the demo program, something you might like to try first yourself, and also provide detailed information on the USI routines. Part 3 of the series will present a complete application program based on the USI routines.

ASSEMBLING THE USI ROUTINES

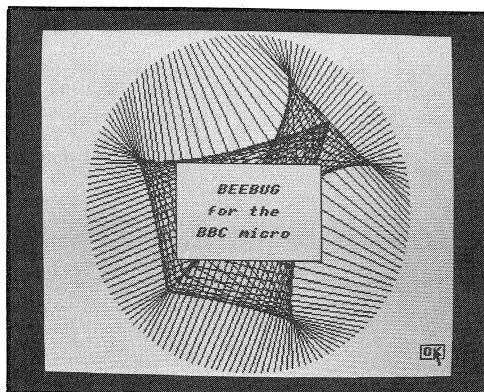
Type in the USI Utilities, taking particular care as most of it is in machine

code. All lines starting with a '\' may be omitted as these are just comment lines. Cassette users should also insure that PAGE is first set to &1900, as the code is assembled below this address. Please keep to the line numbering used, to allow for future additions. Save the program before attempting to run it as a mistake in the machine code could easily corrupt memory.

Run the program and the machine code will be assembled. If all is well, the program will display a list of the USI routines and give instructions for saving the assembled code. Save this with the filename 'M.USI'.

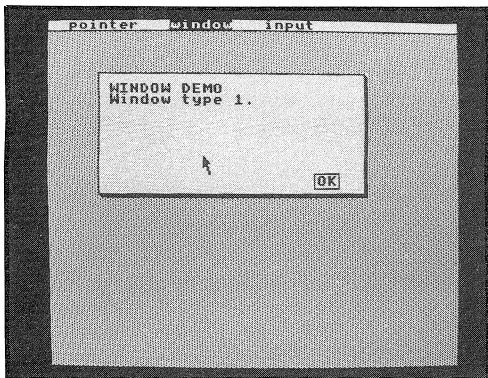
RUNNING THE USI DEMO

Type in the demonstration program, save it and run it. Cassette users should again ensure that PAGE is set to &1900 before loading and running this program. You should see a mode 4 screen with four headings and an arrow pointer. You can move the arrow pointer around the screen with the cursor keys, and make a selection from the top of the screen by pointing to a heading and pressing the space bar. The program demonstrates the three different types of window available, the use of the 'OK' box, how to alter pointer sensitivity, and how to use the text input routine.



The "pointer" option enables you to find the most comfortable setting for the pointer speed (T%). Once you have found your preferred response, you can initialise T% in PROCinit with that value.

The "window" option demonstrates the different window frames available. Type 1 gives a 3D edge, type 2 a double-edged



outline, and type 3 a single-edged one.

The "input" option allows user input (as a character string) with a maximum length determined by *n%*. In this case, *n%* has been set to 7, so the routine will only accept up to 7 characters.

THE PROCEDURES

In order to simplify the task of using the USI utilities, many of them have been incorporated into Basic procedures, which may be used in your own programs. PROCinit must also be included in any program that uses the USI utilities as it declares the addresses of the labels, as well as initialising the pointer co-ordinates (?&70 and ?&71) and pointer speed (T%). The assembled USI routines are also loaded from the file M.USI by this procedure.

PROCopen and PROCclose open and close respectively a pop-up window. PROCopen is followed by five parameters: (*a%*,*b%*) being the lower left-hand corner text co-ordinates of the window, and (*c%*,*d%*) the top right-hand corner; *w%* is the window type. PROCclose simply closes the current window. Only one pop-up window may be opened at any one time.

PROCqb provides a boxed option meant for selection with the pointer, for example the "OK" option in the demonstration program. It has three parameters: *qx%* and *qy%* are the x and y co-ordinates of the option and *q%* is the option descriptor (*q%* <= 39 characters).

PROCip fetches the user input to the maximum length specified by *n%*. It has three parameters: *n%* is the input length;

ix% and *iy%* are the x and y co-ordinates of the input field.

PROCital provides an italic font. It has three parameters: *A%* is the italic string; *x%* and *y%* are the x and y co-ordinates for positioning the string. Characters converted to their italic equivalent are stored in character 253.

ADDITIONAL NOTE

When using the USI routines, HIMEM must be set to &4D00, since the pop-up window uses memory from &4D00 to &5800 as a storage buffer. This is handled at line 110 in the demo program.

```

10 REM USI Utilities
20 REM Listing 1 Version B1.0
30 REM Author Alex Kang
40 REM BEEBUG March 1986
50 REM Program subject to copyright
60 :
100 MODE7:HIMEM=&7000
110 DIM start% 1536
120 PROCassemble
130 PROCcur:PROCcurdef(0):CLS:VDU31,0,
1:PRINT"Code assembled at "&";~start%;"C
ode length = "&";~(P%-start%);" bytes""
ENTRY POINTS:"
140 PRINT"open      = "&";~open
150 PRINT"close     = "&";~close
160 PRINT"pointer   = "&";~pointer
170 PRINT"prptr     = "&";~printptr
180 PRINT"delptr    = "&";~delptr
190 PRINT"input     = "&";~input
200 PRINT"str       = "&";~str
210 PRINT"curoff    = "&";~curoff
220 PRINT"curon     = "&";~curon
230 PRINT"qubox     = "&";~qubox
240 PRINT"oscli     = "&";~oscli
250 PRINT"italic    = "&";~italic
500 PRINT'"To save the code, type:"""
"*SAVE <filename> "&";~start%;" "+"&";~(P%-st
art%);""""
510 END
520 :
1000 DEFPROCassemble
1010 oldxcu=&70:oldycu=&71
1020 newxcu=&72:newycu=&73
1030 noparm=&72:edsad=&72:sbf=&72
1040 sad=&74:loptr=&76:nbyt=&76:numy=&7
7
1050 zsad=&78:texpar=&7A:amtdisp=&7F
1060 bitval=&7F:parbuf=&80:zpbk=&80
1070 zpar=&8E:parblk=&0600:oswrch=&FFEE
1080 osbyte=&FFF4:osrdch=&FFE0
1090 osword=&FFF1
1100 start%=&1300
1110 FORI%=0TO2STEP2

```

```

1120 P%=start%
1130 [OPTI%
1500 \-----PARAMETER HANDLING-----
1510 .pbtozp
1520 LDA parblk,X:STA parbuf,X:DEX
1530 BPL pbtozp:LDA parbuf:CMP noparm
1540 BEQ finish:JMP badparm
1550 .tparm
1560 LDA (parbuf+1,X):STA texpar,Y
1570 DEX:DEX:DEX:DEY:BPL tparm:RTS
2000 \-----WINDOW HANDLING-----
2010 .open
2020 LDX #15:LDA #5:STA noparm
2030 JSR pbtozp:LDX #&C:LDY #4
2040 JSR tparm:JSR wdimen
2050 .svrow LDY #0:LDX nbyt
2060 .svbgd LDA (sad),Y:STA (sbf),Y
2070 INY:DEX:BNE svbgd:DEC numy:BEQ cls
2080 JSR nxrow:JMP svrow:cls LDA #12
2090 JSR oswrch:JSR drawwin:RTS
2100 .close
2110 LDX #12:LDA #4
2120 STA noparm:JSR pbtozp:LDX #9
2130 LDY #3:JSR tparm:JSR wdimen
2140 LDA #26:JSR oswrch:clw
2150 LDY #0:LDX nbyt
2160 .restscn LDA (sbf),Y:STA (sad),Y
2170 INY:DEX:BNE restscn:DEC numy
2180 BEQ finish:JSR nxrow:JMP clw
2190 .finish RTS
2200 .wdimen LDA #26:JSR oswrch
2210 LDA texpar+2:SEC:SBC texpar
2220 CMP #3:BCC badparm:CMP #31
2230 BCS badparm:LDA texpar+1:SEC
2240 SBC texpar+3:CMP #2:BCC badparm
2250 CMP #31:BCS badparm
2260 LDA texpar+1:SEC:SBC texpar+3
2270 STA numy:INC numy:LDA #0:STA sbf
2280 LDA #&4D:STA sbf+1:LDX texpar
2290 LDY texpar+3:JSR getsadd
2300 JSR dftxwin:LDA #&34C:STA nbyt:RTS
2310 \---Bad parameter values error---
2320 .badparm BRK:] ?P%=61:P%=P%+1:$P%="
Bad parameters!":P%=P%+LEN$P%: ?P%=7:P%=P
%+1: ?P%=0:P%=P%+1:[OPTI%
2330 \-----EXECUTE VDU28,x,y,x1,y1-----
2340 .dftxwin LDA #28:JSR oswrch:LDX #0
2350 .vdu28 LDA texpar,X:JSR oswrch
2360 INX:CPX #4:BNE vdu28:RTS
2370 .nxrow
2380 LDA sad:CLC:ADC #&40:STA sad
2390 LDA sad+1:ADC #1:STA sad+1:LDA sbf
2400 CLC:ADC nbyt:STA sbf:LDA sbf+1
2410 ADC #0:STA sbf+1:RTS
2420 \-----gets screen address-----
2430 .getsadd
2440 JSR prttab:LDA #34A:STA sad
2450 LDA #34B:STA sad+1:RTS
2460 \-----TEXT TO GRAPHICS-----
2470 \-----COORDINATES CONVERSION-----
2480 .txtogrp LDX #0:LDY #0
2490 .x32 LDA texpar,Y:PHA
2500 ASL A:ASL A:ASL A:ASL A:ASL A
2510 STA gr,X:PLA:LSR A:LSR A:LSR A
2520 STA gr+1,X:INX:INX:INX
2530 CPY #4:BNE x32
2540 .grcoord
2550 LDY #0:JSR subcd:LDY #3:JSR subcd
2560 LDX #4:LDA #31:CLC:ADC gr,X
2570 STA gr,X:INX:LDA #0
2580 ADC gr,X:STA gr,X:RTS
2590 .subcd
2600 LDX mcdata,Y:LDA mcdata+1,Y:SEC
2610 SBC gr,X:STA gr,X:LDA mcdata+2,Y
2620 SBC gr+1,X:STA gr+1,X:RTS
2630 .mcdata:]P%=P%+45:[OPTI%
2640 .gr:]P%=P%+16:[OPTI%
2650 \-----DRAWS WINDOW FRAMES-----
2660 .drawwin
2670 JSR txtogrp:LDA texpar+4
2680 CMP #1:BEQ wtype1:CMP #2
2690 BEQ wtype2:CMP #3:BEQ wtype3:RTS
2700 \-----3D window frame-----
2710 .wtype1 LDA #8:STA amtdisp
2720 JSR gcdisp:LDY #21
2730 .lp3d JSR vdukxy:CPY #45
2740 BNE lp3d:JMP inwin
2750 \-----Double outline frame-----
2760 .wtype2 LDA #12:STA amtdisp
2770 JSR gcdisp:JSR box1:LDX #15:LDY #7
2780 .nbox LDA gr,X:STA gr,Y:DEX:DEY
2790 BPL nbox:.wtype3 JSR box1
2800 .inwin INC texpar:INC texpar+3
2810 DEC texpar+1:DEC texpar+2
2820 JMP dftxwin
2830 \-----Single outline frame-----
2840 .box1 LDY #6
2850 .bxlp JSR vdukxy:CPY #21:BNE bxlp
2860 RTS
2870 \-----Execute PLOT k,x,y-----
2880 .vdukxy
2890 LDA #25:JSR oswrch:LDA mcdata,Y
2900 JSR oswrch:INX:JSR vduloop
2910 INY:JSR vduloop:INX:RTS
2920 .vduloop
2930 LDX mcdata,Y:LDA gr,X:JSR oswrch
2940 LDA gr+1,X:JSR oswrch:RTS
2950 \-----displace graphics coord.-----
2960 .gcdisp
2970 LDX #0:LDY #8:JSR addn
2980 JSR addn:JSR subn:JSR subn:RTS
2990 .addn
3000 LDA gr,X:CLC:ADC amtdisp:STA gr,Y
3010 INX:INX:LDA gr,X:ADC #0
3020 STA gr,Y:INX:INX:RTS
3030 .subn
3040 LDA gr,X:SEC:SBC amtdisp:STA gr,Y
3050 INX:INX:LDA gr,X:SBC #0
3060 STA gr,Y:INX:INX:RTS
4000 \-----POINTER CONTROL-----

```

```

4010 .pointer
4020 LDA #0:STA &044C
4030 LDA #26:JSR oswrch:JSR printptr
4040 .delay LDY &0450:LDX #0
4050 .incx
4060 INX:BNE incx:DEY:BNE incx
4070 LDA oldxcur:STA newxcur
4080 LDA oldycur:STA newycur
4090 \-----Cursor keys test-----
4100 .left
4110 LDX #&E6:JSR inkey:BEQ right
4120 LDA newxcur:BEQ right:DEC newxcur
4130 .right
4140 LDX #&86:JSR inkey:BEQ down
4150 LDA newxcur:CMP #39:BEQ down
4160 INC newxcur
4170 .down
4180 LDX #&D6:JSR inkey:BEQ up
4190 LDA newycur:CMP #31:BEQ up
4200 INC newycur
4210 .up
4220 LDX #&C6:JSR inkey:BEQ xycmp
4230 LDA newycur:BEQ xycmp:DEC newycur
4240 .xycmp
4250 LDA newxcur:CMP oldxcur:BNE newptr
4260 LDA newycur:CMP oldycur:BEQ eq
4270 .newptr
4280 JSR delptr:LDA newxcur:STA oldxcur
4290 LDA newycur:STA oldycur
4300 JSR printptr
4310 .eq LDX #&9D:JSR inkey
4320 BNE spacebar:JMP delay
4330 .spacebar
4340 LDA oldxcur:STA newxcur
4350 LDA oldycur:STA newycur
4360 JSR delptr:LDA #15:LDX #1
4370 JSR osbyte:RTS
4380 \-----INKEY Key Test-----
4390 .inkey LDY #&FF:LDA #&81
4400 JSR osbyte:CPX #0:RTS
4410 \-----Prints pointer-----
4420 .printptr
4430 LDX oldxcur:LDY oldycur
4440 JSR getsadd:LDA sad:CLC:ADC #&40
4450 STA loptr:LDA sad+1
4460 ADC #1:STA loptr+1
4470 .cur LDY #7
4480 .lol
4490 LDA (sad),Y:STA sptr,Y
4500 LDA (loptr),Y:STA sptr+8,Y
4510 DEY:BPL lol:LDY #7
4520 .lo2
4530 LDA sptr,Y:AND pdef,Y
4540 ORA pdef+16,Y:STA (sad),Y
4550 LDA sptr+8,Y:AND pdef+8,Y
4560 ORA pdef+24,Y:STA (loptr),Y
4570 DEY:BPL lo2:RTS
4580 .sptr:]P%=P%+16:[OPTI%
4590 .pdef:]P%=P%+32:[OPTI%
4600 \-----Deletes pointer-----

```

```

4610 .delptr LDY #7
4620 .lo3 LDA sptr,Y:STA (sad),Y
4630 LDA sptr+8,Y:STA (loptr),Y
4640 DEY:BPL lo3:RTS
5000 \----CHARACTER INPUT ROUTINE-----
5010 .input
5020 LDX #2:LDA #4:JSR osbyte
5030 LDX #1:LDA #15:JSR osbyte
5040 JSR curon:LDY #0:LDA #&0D
5050 STA str,Y:LDX #0
5060 .readchar
5070 JSR osrdch:CMP #&1B:BNE notesc
5080 LDA #&7E:JSR osbyte
5090 .notesc
5100 CMP #&0D:BEQ endip:CMP #9
5110 BEQ readchar:CMP #&7F:BNE testmax
5120 CPX #0:BEQ beep:DEX:JSR oswrch
5130 BPL readchar
5140 .testmax
5150 CPX &0438:BEQ beep:JSR oswrch
5160 STA str,X:INX:JMP readchar
5170 .beep
5180 LDA #7:JSR oswrch:JMP readchar
5190 .endip
5200 LDA #&0D:STA str,X:JMP curoff
5210 .str:]P%=P%+50:[OPTI%
5220 \----cursor off/on routine-----
5230 .curoff LDX #0
5240 .offlp LDA curdata,X:JSR oswrch
5250 INX:CPX #4:BNE offlp:LDX #6
5260 .zerolp LDA #0:JSR oswrch
5270 DEX:BNE zerolp:RTS
5280 .curon LDX #4
5290 .onlp LDA curdata,X:JSR oswrch
5300 INX:CPX #8:BNE onlp:BEQ zerolp
5310 .curdata
5320 ]!P%=&200A0017:P%=P%+4:!P%=&670A00
17:P%=P%+4:[OPTI%
6000 \-----QUERY BOX ROUTINE-----
6010 .qubox
6020 LDX #6:LDA #2:STA noparm
6030 JSR pbtozp:LDX #3:LDY #1:JSR tparm
6040 LDA #26:JSR oswrch:LDX texpar
6050 LDY texpar+1:STY texpar+3
6060 JSR prttab:JSR atomstr:DEY:DEY:TYA
6070 CLC:ADC texpar:STA texpar+2
6080 JSR txtogrp:LDA #8:STA amtdisp
6090 LDX #0:LDY #0:JSR subn:JSR subn
6100 JSR addn:JSR addn:JMP box1
6110 \--printtab (vdu31,x,y) routine--
6120 .prtab LDA #31:JSR oswrch:TXA
6130 JSR oswrch:TYA:JSR oswrch:RTS
6140 \-----prints $str on screen-----
6150 .atomstr LDY #0
6160 .printstr LDA str,Y:JSR oswrch
6170 INY:CMP #&0D:BNE printstr:RTS
7000 \-----OSCLI ROUTINE-----
7010 .oscli LDX #(str AND &FF)
7020 LDY #(str AND &FF00):DIV &100
7030 JSR &FFF7:RTS

```



```

8000 \-----ITALICS ROUTINE-----
8010 .italic LDX #&80:LDY #0:LDA #&A:JS
R osword:CLC:LSR &81:LSR &82:ASL &86:ASL
&87:ASL &88:RTS
10000 ]
10010 NEXT:ENDPROC
10020 :
10030 DEFPROCcur
10040 DATA&7100,C0A090888482818684B2D212
0909090600406070787C7E78784C0C0C06060600
10050 RESTORE10040:READA%,A$:FORi%=0TO3
1:A%?i%=EVAL ("%"+MID$(A$,1+i%*2,2)):NEXT
10060 DATA02E00306FF03040002050402050406
05000605000204080204000A550402550C0A5504
0E550C0605000605000A
10070 READA$:FORi%=0TO44:modata?i%=EVAL (
"%"+MID$(A$,1+i%*2,2)):NEXT
10080 ENDPROC
10090 :
10100 DEFPROCcurdef(n%)
10110 FORi%=0TO15:pdef?i%=-1EOR ?(&7100+
i%+n%*32):NEXT:FORi%=16TO31:pdef?i%=?(&7
100+i%+n%*32):NEXT
10120 ENDPROC

```

```

10 REM USI UTILITIES DEMO
20 REM Version B1.0
30 REM Author Alex Kang
40 REM BEEBUG March 1986
50 REM Program subject to copyright
60 :
100 *TV255,1
110 MODE4:HIMEM=&4D00
120 DIM str 50:H%=4
130 PROCinit:PROCbgd:CALL cufoff
140 REPEAT:S%=0:CALL pointer
150 S%=0:CALL pointer
160 RESTORE250:i%=0:REPEAT:i%=i%+1:REA
D 1%,r%:IF (?&70>=1% AND ?&70<=r%) AND ?
&71=0 THEN S%=i%
170 UNTILS%>0 OR i%=H%
180 UNTILS%>0:PROCfb(1%,0,r%-1%+1)
190 IF S%=1 PROCadj
200 IF S%=2 PROCdemoWin
210 IF S%=3 PROCdemoinput
220 IF S%=4 VDU26,12:MODE7:END
230 PROCfb(1%,0,r%-1%+1)
240 GOTO140
250 DATA 2,8,13,18,23,27,32,35
260 :
1000 DEFPROCinit:*LO."M.USI"
1010 open=&131D:close=&134F:pointer=&15
54:prptr=&15ED:delptr=&165C:input=&166C:
str=&16C7:cufoff=&16F9:curon=&1711:qubox
=&1728:oscli=&1787:italic=&178F:T%=00:??&
70=20:??&71=16
1020 VDU19,0,6;0;19,1,0;0;23,255,255,25
5,255,255,255,255,255:ENDPROC
1030 :

```

```

1040 DEFPROCopen(a%,b%,c%,d%,w%):A%=a%:
B%=b%:C%=c%:D%=d%:W%=w%:CALL open,A%,B%,
C%,D%,W%:ENDPROC
1050 :
1060 DEFPROCclose
1070 CALL close,A%,B%,C%,D%:ENDPROC
1080 :
1090 DEFPROCqb(qx%,qy%,q%)
1100 $str=q$:CALL qubox,qx%,qy%:ENDPROC
1110 :
1120 DEFPROCip(n%,ix%,iy%):N%=n%:VDU26,
31,ix%,iy%:PROCfb(ix%,iy%,n%):*FX200,1
1130 ?&D3=255:CALLinput:??&D3=0:*FX200,0
1140 ENDPROC
1150 :
1160 DEFPROCital(A$,x%,y%):VDU31,x%,y%:
FORi%=1TOLEN$A$:??&80=ASC(MID$(A$,i%,1)):C
ALL italic:VDU23,253:FORj%=1TO8:VDUj%?&8
0:NEXT:VDU253:NEXT:ENDPROC
1170 :
2000 DEFPROCfb(X%,Y%,Z%):X%=X%*32:Y%=10
23-32*Y%:VDU5:GCOL3,1:MOVEX%,Y%:PRINTSTR
ING$(Z%,CHR$255):GCOL0,1:VDU4:ENDPROC
2010 :
2020 DEFPROCd(d%):tm%=TIME:REPEATUNTILT
IME>tm%+d%:ENDPROC
2030 :
2040 DEFPROCadj
2050 PROCopen(0,8,25,1,2):PRINT"POINTER
SPEED""30 (fast) - 150 (slow)""Curre
nt value = ";T%:CALL curon:INPUT"New val
ue: " newt%:CALL cufoff:IF newt%>29 AND
newt%<151 T%=newt%
2060 PROCclose:ENDPROC
2070 :
2080 DEFPROCbgd
2090 VDU23,254,17,68,17,68,17,68,17,68
2100 ?&D0=2:FORj%=1TO31:PRINTTAB(0,j%)S
TRING$(40,CHR$254):NEXT:MOVE0,991:DRAW1
279,991:VDU31,0,2:??&D0=0:PROCital("point
er window input exit",2,0):ENDP
ROC
2110 :
2120 DEFPROCdemoWin
2130 FORw%=1TO3:PROCopen(12,8,30,1,w%):
PRINT"WINDOW DEMO""Window type W%=";w%:
PROCok(26,7):PROCclose:NEXT:ENDPROC
2140 :
2150 DEFPROCdemoinput
2160 PROCopen(5,15,30,5,2):PRINT"INPUT
DEMO""with N%=7.""It will only accept"
""up to 7 characters."":PROCip(7,8,11):PR
OCok(26,14):PROCclose:ENDPROC
2170 :
2180 DEFPROCok(okx%,oky%)
2190 PROCqb(okx%,oky%,"OK"):??&70=okx%+1
:??&71=oky%:REPEAT:CALL pointer:UNTIL(??&
70>=okx% AND ??&70<=okx%+1) AND ??&71=oky%:
PROCfb(okx%,oky%,2):PROCd(20):ENDPROC

```



by Mitch

ADVENTURE GAMES ADVENTURE GAMES

TERRORMOLINOS £7.95

HAMPSTEAD £6.95

**Supplier: Melbourne House,
Castle Yard, Richmond TW10 6TF.**

Both these games were originally written for the Spectrum using the Quill adventure writing generator, which has resulted in slightly shorter than usual games with a very similar format. Both games lean heavily on their humour rather than puzzles to occupy the player and the same pair of authors are responsible for both.

The Terrormolinos plot involves you and your tatty family in a tacky package tour to Spain where you must survive the food, excursions and the inhabitants. To accomplish your mission you must return having taken ten photographs to prove you really did do all those horrid things the British do abroad on holiday. The problems contained within are fairly simple and suitable for a newcomer to adventures.

The equally edifying plot of Hampstead is your journey from Council-House Prole to Hampstead Yahoo via the Dole Office, Betting Shop and the West-End. Remember always to wear the right clothes, eat the right food and no doubt kiss the right place on the right person!

In both games the dubious, barbed and mocking humour leaves a smell in the nostrils reminiscent of my dragon's sleeping bag. In too many places I felt the authors were sneering rather than laughing. I recommend that some wholesome adventurer escorts these "Closet Yuppies" to the centre of Collosal Cave and leaves them for the trolls.

BORED OF THE RINGS £6.95

Supplier: Silversoft Games.

For Tolkien lovers - who are trying to break the hobbit - this game will kill or

cure. Not so much an adventure, more a mobile story, this game is thin on puzzles but full of hilarious text. The Pythonesque dialogue and descriptions pour onto the screen oblivious of any attempt you make to solve anything. The result is a truly witty skit which will delight Tolkien fans but frustrate game players. Early versions of the game have serious programming faults which is very frustrating but I'll chuckle over many of the game's witticisms for some time to come.

Incidentally, If you are stuck for help in the willow tree, send an SAE to 'CRY FOR HELP' at the editorial address.

THE WORM IN PARADISE


£9.95 (cassette), £11.95 (disc)

**Supplier: Level 9 Computing,
P.O. Box 39, Weston-Super-Mare, Avon.**

In an attempt to get back to where men are men and the princesses are glad of it, I plunged straight into this final episode of the Silicon Dream trilogy. Like its predecessors, laughs are thin on the ground in this game and the 200 plus locations complete with plastic pizzas and cups of lentil custard soon wiped the smile off my face. This future planet is complete with Dream Park, Casino and Robot police (Fuzbots) who control the mindless, pleasure-seeking inhabitants. To help you save the world there is a cuddly robot dog called Dagget, (late of Battlestar Galactica) who it appears has an unhealthy attraction to your leg!

A nice advance in Level 9 software is shown by this typical response to your attempt to affect a fountain,

"Fountain:- scenery only, ignore it". This useful reply could save you three hours trying to find three coins to chuck into it!

There is a complex travel system within the city using the Resistor Colour Code. If you don't know what "Bad Boys do to Our Young Girls" I suggest you find out from some electrical trog, or you and your Dagget will never make it home. This text-only game is as devious and professional as all Level 9 adventures, and as I have solved a lot more of this than I did with the two previous episodes - it may be just slightly easier. 



Ramsafe

Dynamic Backup to Sideways RAM

How many times have you been developing a program only to lose the contents of memory through a power cut or software failure? Derek Floyd's RAMSAFE routine will automatically and regularly back up your current program to sideways RAM.

More and more BBC micro owners are investing in sideways RAM, to use as buffers, RAM discs or temporary homes for their ROM programs. Sideways RAM can also be used to hold a copy of main memory, thus protecting a Basic program from accidental erasure. If the sideways memory is battery-backed this protection can even be extended to cover power failures.

Sideways RAM can take many forms, one of which is the special sideways RAM board sold by companies such as Watford Electronics. The second, and most popular option, is to buy a ROM expansion board which allows the addition of sideways RAM (see review in BEEBUG Vol.3 No.6). ROM boards such as the ATPL 'Sidewise' allow 16k of RAM to be fitted as two 8k RAM chips.

RAMSAFE is a utility which allows the contents of main memory to be saved to and loaded from sideways RAM at any time. This is done using the BBC micro's interrupt system. The RAMSAFE machine-code is itself held in sideways RAM, but occupies less than .5K.

The RAMSAFE routine is designed to load itself into sideways RAM and pretend to the ROM operating system that it's a ROM. Indeed, if you type *HELP once you've installed the routine, it will report itself like any true ROM!

USING THE PROGRAM

Type in the program listing, paying particular attention to PROCassemble and the value of 'last' in line 1100. This should be &BF for 16K sideways RAM and &9F for 8K. Check that your listing is exactly as printed, and save a copy to tape or disc before running it.

Run the program and press any key to install the assembled machine-code into sideways RAM. Press Break to initialise the RAMSAFE 'ROM' and you can then use any of three new RAMSAFE commands:

*RSAVE saves the contents of memory between PAGE and TOP into sideways RAM from &8200 upwards. If your program is longer than your available sideways RAM, it can't all be saved, and RAMSAFE reports 'Ram full !'. Otherwise the program will beep to show a successful save.

*RLOAD loads a previously saved program from sideways RAM into main memory.

*TSAVE automatically starts a repeating countdown timer, and *RSaves memory every 10 minutes, when the timer reaches zero. This is done 'transparently', so you can continue to work without giving any further instructions.

If you're using *TSAVE on a micro with battery-backed sideways RAM, and suffer a power failure, you can recover your program by re-running RAMSAFE and using *RLOAD to reload the contents of sideways RAM.

PROGRAM NOTES

RAMSAFE uses 'events', the BBC Micro's 'packaged' interrupt facility, to take control of the processor for saving and loading memory. The routine is very compact, but tends to dominate the event system. This means that it may be incompatible with other programs which use the event vector.

BEEBUG Retail can supply 8k RAM chips at £4.50 each. Please enclose 50 pence p&p along with your cheque / credit card number to BEEBUG Retail, Dolphin Place, Holywell Hill, St.Albans, AL1 3YS. Alternatively you may telephone your order on St.Albans 40303 using your credit card.

```
10 REM PROGRAM RAMSAFE
20 REM Version 1.20
30 REM Author D.R. Floyd
40 REM BEEBUG March 1986
50 REM Program subject to copyright
60 :
100 REM Main Program
110 PROCtitle
120 PROCdata
```

```

130 PROCassemble
140 END
150 :
1000 DEFPROCdata
1010 REM Data Set
1020 page=&1D      :REM PAGE
1030 top=&01       :REM TOP
1040 from=&70      :REM data
1050 to=&72        :REM for
1060 length=&74    :REM move.
1070 timeset=&75   :REM timerbase
1080 timeread=&7A
1090 start=&82
1100 last=&BF:REM last=&9F for 8K RAM
1110 size=last-start
1120 osrdch=&FFE0
1130 oswrch=&FFEE
1140 osnewl=&FFE7
1150 osbyte=&FFF4
1160 osword=&FFF1
1170 ENDPROC
1180 :
1190 DEFPROCassemble
1200 FOR I%=0 TO 2 STEP 2
1210 P%=&8000
1220 [OPT I%
1230 .ram
1240 OPT FNEQUB(&0)
1250 OPT FNEQUB(&0)
1260 OPT FNEQUB(&0)
1270 JMP sentry \ Service entry
1280 OPT FNEQUB(&82)
1290 OPT FNEQUB(&15)
1300 OPT FNEQUB(&12)
1310 .title
1320 OPT FNEQUS("RAMSAFE") \ ROM title
etc
1330 OPT FNEQUB(&20)
1340 OPT FNEQUS("1.20")
1350 OPT FNEQUB(&00)
1360 OPT FNEQUS("(C) 1985 BEEBUG")
1370 OPT FNEQUB(&00)
1380 .sentry
1390 PHP:CMP #4 \ Unrecognised command
1400 BEQ unreccom
1410 CMP #9 \ Help call
1420 BEQ help
1430 PLP
1440 RTS
1450 .out \ Restore registers & return
1460 PLA:TAY:PLA:TAX:PLA:PLP
1470 RTS
1480 .exit \ Set Acc to zero & return
1490 PLA:PLA:PLA:PLA:LDA #0
1500 RTS
1510 .list
1520 OPT FNEQUS("TSAV") \ List of new
1530 OPT FNEQUB(&C5) \ commands.
1540 OPT FNEQUS("RSAV") \ Last character of

```

```

1550 OPT FNEQUB(&C5) \ each has bit
7
1560 OPT FNEQUS("RLOA") \ set.
1570 OPT FNEQUB(&C4)
1580 .unreccom \ Unrecognised command
1590 PHA:TXA:PHA:TYA:PHA \ Save regs.
1600 LDX #0 \ Beginning of list
1610 JSR search \ Search for TSAVE
1620 BNE nottsave
1630 JMP tsave
1640 .nottsave
1650 PLA:PHA:TAY \ Reset pointer 'Y'
1660 JSR search \ Search for RSAVE
1670 BNE notrsave
1680 JMP rsave
1690 .notrsave
1700 PLA:PHA:TAY \ Reset pointer 'Y'
1710 JSR search \ Search for RLOAD
1720 BNE notrload
1730 JMP rload
1740 .notrload
1750 JMP out \ Nothing found
1760 .search \ Search for word
1770 TXA:PHA \ Save 'X'
1780 .searchloop
1790 LDA list,X \ Check list
1800 BMI gotit \ against
1810 CMP (&F2),Y \ command.
1820 BNE nothere \ If not here, leave
1830 INY:INX
1840 JMP searchloop
1850 .nothere \ Set 'X' to
1860 PLA:CLC:ADC #5:TAX \ next word
1870 RTS \ and retn.
1880 .gotit \ Mask last character
1890 AND #&5F \ and check
1900 CMP (&F2),Y \ for match.
1910 BNE nothere \ Branch if no match
1920 PLA:LDA #&0 \ Match! Set 0 flag
1930 RTS
1940 .help \ Save all registers
1950 PHA:TXA:PHA:TYA:PHA
1960 JSR osnewl \ Send <CR> to VDU
1970 LDX #0 \ Fetch rest of
1980 LDA (&F2),Y \ command.
1990 CMP #&0D \ Is it <CR>?
2000 BNE helpx
2010 .titleloop
2020 LDA title,X \ Yes, so send ROM
2030 JSR oswrch \ title to VDU.
2040 INX:CPX #7
2050 BNE titleloop
2060 JSR osnewl \ Send <CR> to VDU
2070 JMP out \ Return to system
2080 .helpx \ No, so check for
2090 LDX #0 \ ROM title.
2100 .testloop
2110 LDA (&F2),Y
2120 CMP #&20 \ Ignore spaces
2130 BEQ space

```

```

2140 CMP title,X
2150 BEQ space
2160 JMP out \ No match - return
2170 .space
2180 INY:INX:CPX #7
2190 BNE testloop
2200 LDX #0 \ Check ROM OK.
2210 JSR word \ Write list
2220 JSR word \ of words
2230 JSR word \ to VDU.
2240 JMP out \ Return to system
2250 .word
2260 LDA #&20 \ Write a word,
2270 JSR oswrch \ starting with
2280 JSR oswrch \ two spaces,
2290 .wordloop \ then a
2300 LDA list,X \ character,
2310 BMI endword \ until end
2320 JSR oswrch \ of word.
2330 INX:JMP wordloop
2340 .endword
2350 INX:AND #&7F \ Mask last char.,
2360 JSR oswrch \ send it to VDU,
2370 JSR osnewl \ then a <CR>.
2380 RTS
2390 .tsave \ Command TSAVE
2400 LDA #&16 \ Set timebase
2410 STA timeset+1 \ to
2420 LDA #&FF \ give
2430 STA timeset \ delay
2440 STA timeset+2 \ of
2450 STA timeset+3 \ ten
2460 STA timeset+4 \ minutes.
2470 LDA #&30:STA &220 \ Set event vec
2480 LDA #&FF:STA &221 \ to ext vect.
2490 LDA #&A8:LDX #&00 \ Fetch address
2500 LDY #&FF \ of ext. address
2510 JSR osbyte \ area, and store
2520 STY &81:STX &80 \ it in page 0.
2530 LDY #&30
2540 LDA #isave MOD 256
2550 STA (&80),Y \ Set ROM addr. lsb.
2560 INY
2570 LDA #isave DIV 256
2580 STA (&80),Y \ Set ROM addr. msb.
2590 INY
2600 LDA &F4 \ Fetch ROM number
2610 STA (&80),Y \ and put it here.
2620 LDA #&0E \ *FX 14,5
2630 LDX #&05 \ enables timer
2640 JSR osbyte \ event.
2650 JSR timeinit
2660 JMP exit \ Return to system
2670 .timeinit
2680 LDA #&04 \ OSWORD 4 resets
2690 LDX #timeset MOD 256 \ the
2700 LDY #timeset DIV 256 \ timer.
2710 JSR osword
2720 RTS
2730 .rsave \ Command RSAVE

2740 JSR dosave
2750 JMP exit
2760 .rload \ Command RLOAD
2770 LDA #&00:STA &70:STA &72 *
2780 LDA page:STA &73 \ Set TO & FROM
2790 LDA #&82:STA &71 \ addresses, &
2800 LDA &81FF:STA length \ length.
2810 JSR move \ Move block
2820 JMP exit \ Return to system
2830 .dosave \ Save routine
2840 JSR beep \ Make beep
2850 LDA #&00:LDX #00:STA &70:STA &72
2860 LDA page:STA &71 \ Set FROM & TO
2870 LDA #&82:STA &73 \ addresses for
2880 LDA top:SEC:SBC &71 \ saving.
2890 STA length:INC length
2900 LDA length:CMP #size
2910 BMI roomok
2920 JSR noroom \ No room, so say so
2930 .roomok
2940 LDA length \ Get length and
2950 STA &81FF \ save in RAM.
2960 JSR move \ Move block
2970 RTS
2980 .isave \ Ext. vector here
2990 PHP:PHA:TXA:PHA:TYA:PHA
3000 JSR dosave \ Do normal save
3010 JSR timeinit \ Reset timer
3020 JMP exit \ Return to system
3030 .move \ Block move routine
3040 LDX length \ Outer loop
3050 .outerloop \ counts pages.
3060 LDY #&00 \ Inner loop
3070 .innerloop \ counts bytes.
3080 LDA (from),Y \ Transfer
3090 STA (to),Y \ one byte.
3100 INY:BNE innerloop \ Next byte
3110 INC from+1 \ Page done -
3120 INC to+1 \ bump pointers.
3130 DEX:BNE outerloop \ Next page
3140 RTS
3150 .noroom \ Run out of space
3160 LDX #00
3170 .noroomloop
3180 LDA ramfull,X \ Say 'RAM full'
3190 JSR oswrch \ to VDU.
3200 INX:CPX #&0B
3210 BNE noroomloop
3220 LDA #size:STA &74
3230 JSR osnewl
3240 RTS
3250 .ramfull
3260 OPT FNEQUS("Ram full !")
3270 OPT FNEQUB(&0D)
3280 .beep
3290 LDA #&7 \ Make a noise
3300 JSR oswrch
3310 RTS
3320 ]
3330 NEXT I%

```


PRESERVING SCREENS

If you ever wished you could emulate some of the commercially available programs which display a graphics screen, then mode 7 text and then restore the original graphics, Paul Ganney's routines could be just what you need.

Producing a reasonably complex display on the screen can take some time. If your program needs to follow this with a screen of text and then return to the graphics display, much time-consuming re-drawing is usually involved. However, this is all unnecessary. It is quite possible for your program to produce an attractive graphics screen, switch to mode 7 for a text display, and then to restore the original graphics display. The short routine listed here can be incorporated in your own programs to achieve just this effect.

The utility is contained in PROCCHANGE. This should be added to your own program, and called once near the start of the program. It sets up the machine code routine to swap between mode 2 and mode 7. Alternatively, the code could be assembled and *LOADED in separately. To do this, type in or load PROCCHANGE. Then enter
PROCCHANGE <Return>
*SAVE mode 900 +111 <Return>

To use the utility, all that is required is to set up your mode 2 screen, and then change screens as you wish by using the instructions 'CALL TWO' and 'CALL SEVEN' (or 'CALL &900' and 'CALL &9E1' if pre-assembled). Note that as the routine stands, the mode 7 screen is set up and cleared using VDU22,7 so as not to re-set HIMEM (set in line 120 to preserve the 1K picture buffer).

PROGRAM NOTES

The basic idea used in this routine is as follows. The operating system's 'mode' change routine begins at &CB1D and is entered with the accumulator set to the required mode number. After setting the registers of the 6845 with the appropriate

values, it then clears the screen. Thus, by copying the first part of this routine to RAM, we can CALL this and effect a mode change without clearing the previous screen. This is achieved in line 1060.

Modes 7 and 2 use 1K of common memory, and this must therefore be preserved elsewhere when changing modes. This is done by transferring the 1K of the mode 2 screen to memory locations &2B00 to &2F00 when entering mode 7, and transferring it back to restore the picture. This is done in lines 1270 to 1420.

To use modes different from 2 and 7, you will need to insert the appropriate mode values into lines 1040 and 1240. Note that a mode other than 7 will require a larger buffer area and thus require further alterations to the program. If for example you wish to use modes 5 and 6, first select mode 5 and set HIMEM to &3800 (see table below). Alter the values at lines 1040 and 1240 to read 5 and 6 respectively. Then the values at lines 1100 and 1190 should be changed to &60 (second value in table) as this is the high byte of the mode 6 screen start, and lines 1120 and 1170 have to be changed to equal the high byte (first two digits) of the value of HIMEM (&38). Please note that it is impractical to swap a 20K mode with any mode other than 7 as this would require at least 28K of screen memory even without the program!

The table shows the necessary values for various mode combinations.

Mode	1st Mode				
	0,1,2	3	4,5	6	7
0,1,2	N/A	N/A	N/A	N/A	&2C00
2					&30
n 3	N/A	N/A	N/A	&2000	&3C00
d				&40	&40
4,5	N/A	N/A	&3000	&3800	&5400
M			&58	&58	&58
o 6	N/A	&2000	&3800	&4000	&5C00
d		&60	&60	&60	&60
e 7	&2C00	&3C00	&5400	&5C00	&7800
	&7C	&7C	&7C	&7C	&7C

The program as listed contains a small demonstration of the PROCCHANGE routine.

10 REM PROGRAM PRESERVE
20 REM VERSION B0.1

```

30 REM AUTHOR P GANNEY
40 REM BEEBUG MARCH '86
50 REM PROGRAM SUBJECT TO COPYRIGHT
60 :
100 ON ERROR GOTO 5000
110 PROCCHANGE
120 MODE2:HIMEM=&2B00
130 FORI%=&3000TO&7FFF
140 ?I%=I%
150 NEXT
160 PRINTTAB(7,10)"MODE 2"
170 PRINT'"PRESS ANY KEY "
180 REPEAT
190 IFGET
200 CALL SEVEN
210 PRINTTAB(10,10)"MODE 7"
220 PRINT'"PRESS ANY KEY FOR MODE 2,"
"OR ESCAPE TO END"
230 IFGET
240 CALL TWO
250 UNTIL0
260 :
1000 DEF PROCCHANGE
1010 P%=&900
1020 [OPT2
1030 .TWO
1040 LDA#2
1050 ]
1060 FORN%=&0TO&D2:N%?&902=N%?&CB1D:NEXT
1070 FORI%=&0TO2STEP2
1080 P%=&9D5
1090 [OPTI%
1100 LDA#&7C
1110 STA&71
1120 LDA#&2B
1130 STA&73
1140 JSR SCREFN
1150 RTS
1160 .SEVEN
1170 LDA#&2B
1180 STA&71
1190 LDA#&7C
1200 STA&73
1210 JSR SCREEN
1220 LDA#22
1230 JSR&FFEE
1240 LDA#7
1250 JSR&FFEE
1260 RTS
1270 .SCREEN
1280 LDA#0
1290 STA&70
1300 STA&72
1310 LDY#0
1320 .LOOP
1330 LDA(&72),Y
1340 STA(&70),Y
1350 INY
1360 BNE LOOP
1370 INC&71
1380 INC&73
1390 LDA#&80
1400 CMP&73
1410 BNE LOOP
1420 RTS
1430 ]
1440 NEXT
1450 ENDPROC
1460 :
5000 ON ERROR OFF:MODE7
5010 REPORT:PRINT" at line ";ERL
5020 END

```

← 24

```

3340 CLS:PRINT "RAMSAFE installed - pre
ss <BREAK>"
3350 ENDPROC
3360 :
3370 DEFPROCtitle
3380 CLS
3390 PRINT TAB(15,2)CHR$(141);"RAMSAFE"
3400 PRINT TAB(15,3)CHR$(141);"RAMSAFE"
3410 PRINT TAB(11,7)"for sideways RAM"
3420 PRINT TAB(14,12)"written by"
3430 PRINT TAB(15,14)"D.R.Floyd"
3440 PRINT TAB(7,21)"Press any key to i
nsta11":G=GET
3450 ENDPROC
3460 :
3470 DEFFNEQUB(n)
3480 ?P%=n:P%=P%+1:=0
3490 :
3500 DEFFNEQUS(n$)
3510 $P%=n$:P%=P%+LEN(n$):=0

```

POINTS ARISING POINTS ARISING POINTS ARISING POINTS

A number of mostly minor errors crept into the December issue of BEEBUG (Vol.4 No.7). In Personal Diary Manager there is an occasional problem in the handling of the 28th/29th February. This can be corrected by adding the line:

```
2155 IF X%=60 AND L%=0 THEN M%=3
```

In Chinese Chequers insert a space between 'cr' and 'MOD' in line 1900 (i.e. put cr MOD and not crMOD). Line 100 should also be replaced by: '100 MODE 1'.

In the Fuzzy Commands program in the January/February issue (Vol.4 No.8) add:

```
1665 LDA &18:STA block+3:LDA #0:STA block+2:STA block+6
```

MORFIX COMPETITION RESULTS

The Morfix competition set in the December supplement brought another good response from members. The program was featured on the magazine cassette/disc, and the competition involved answering six questions about the game.

The six questions are repeated below, with the relevant answers.

1. Who can be seen on TV? (Answer: Morfix)
2. What football club does Morfix support? (Answer: LFC or Liverpool)
3. What time is shown by the cuckoo clock? (Answer: 3 o'clock OR 12.15)
4. What time is shown by the round clock? (Answer: 3 o'clock)
5. How many ships or boats are there? (Answer: 3 - 2 of them are in pictures)
6. What happens if you succeed your quest? (Answer: Screen flashes, a tune plays, and a love heart appears.)

The first name out of the BEEBUG balaclava was that of Nathan Pearce of Reading and the prize of £50 will be on its way to him. Thank you to everybody who entered the competition, and we hope you had as much fun playing the game over Christmas as we did.

ATTENTION ALL CIRCUIT DESIGNERS!!

LOW COST ELECTRONICS C.A.D.

IBM PC BBC MODEL B and SPECTRUM 48K

'Analyser' computes the AC Frequency Response of linear (analogue) circuits. Gain and Phase, Input Impedance, Output Impedance and Group Delay (except Spectrum version) are calculated over the frequency range required. The effects on performance of Modifications to the circuit configuration and component values can be speedily evaluated.

Circuits containing Resistors, Capacitors, Inductors, Transformers, Bipolar and Field Effect Transistors and Operational Amplifiers can be simulated - up to 150 components (IBM version).

Ideal for the analysis of Active and Passive Filter Circuits, Audio Amplifiers, Loudspeaker Cross-Over Networks, Wide-Band Amplifiers, Tuned RF Amplifiers, Aerial Matching Networks, TV IF and Chroma Filter Circuits, Linear Integrated Circuits, etc. etc.

'Analyser' can greatly reduce or even eliminate the need to breadboard new designs.

Used by Industrial R&D Departments and Universities world-wide. Very Easy to Use. Prices from £20 ex VAT. Access or American Express welcome.

For further details and example computation or for details on our New Draughting Program, please write, phone or telex

NUMBER ONE SYSTEMS LIMITED
Dept B B, 9A Crown Street, St Ives
Huntingdon, Cambs PE17 4EB, UK

Tel: (0480) 61778 Telex: 32339

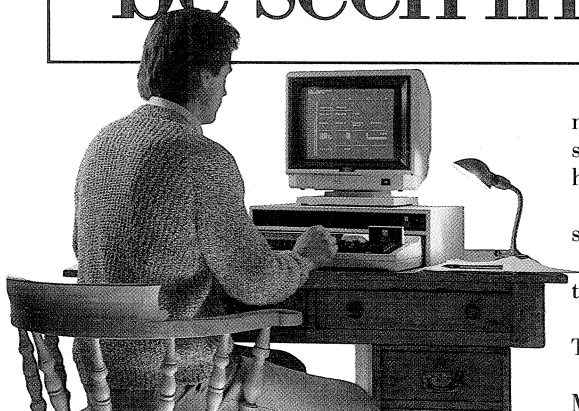
ADVERTISING IN BEEBUG

For advertising details please
contact:

Lynn Lloyd Yolanda Turvelo
on or on
(0923) 677222 (0727) 40303

or write to:
P.O. Box 50
St. Albans, Herts.

The new BBC microcomputer series will be seen in all areas.



The BBC Master Series is a continuation and development of the highly popular BBC Micro. As such, the series retains all of the BBC Micro's traditional virtues including BBC BASIC and a superior choice of software and peripherals.

Though never before has one microcomputer encompassed such a wide range of applications or offered so much potential.

THE HOME.

BBC Micros have gained a reputation as powerful, versatile home computers.

The Master 128 will prove an even more useful addition to the household.

Your child is probably already familiar with it from school.

In which case, our educational programs will ensure that learning continues at home.

By using a Teletext adaptor, up-to-the-minute information can be called up instantly.

And via a Modem, you can control all the money in your bank account, paying bills and standing orders from the comfort of your own home.

You can even write and send letters on screen by Electronic Mail.

Yet despite these advances, the price remains the same at £499 inc. VAT.

To speed up other tasks, there is the Master TURBO.

For an extra £125 inc. VAT, it improves the Master 128's benchmark from 9.2 seconds to a record-breaking 4.67 seconds using BBC BASIC.

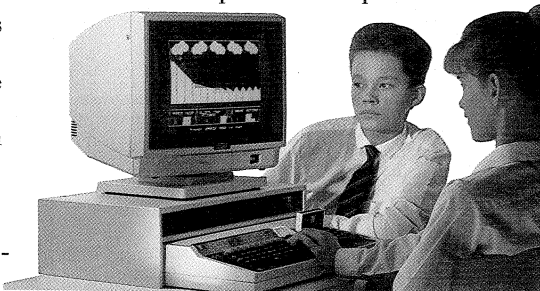
THE SCHOOL.

The BBC Micro is already to be seen in 75% of our schools.

New features on the Master 128 make it particularly useful in the schools area.

The word processing package makes it ideal for creative writing.

While its sophisticated spreadsheet will



THE MASTER 128. 128K RAM 2 x 256K ROM CAPACITY CARTRIDGE SOCKETS. REAL TIME CLOCK/CALENDAR. VIEW, VIEWSHEET, EDITOR AND TERMINAL. THE MASTER ET. 128K RAM AND 64K ROM. ADVANCED NETWORK FILING SYSTEM. 32K BYTES OPERATING SYSTEM. 16K BYTES BBC BASIC. *THE MASTER TURBO. RAM. HARDWARE FLOATING POINT. SEND FOR FULL DETAILS FROM BOB COATES, ACORN COMPUTER

prove invaluable in any Maths Class.

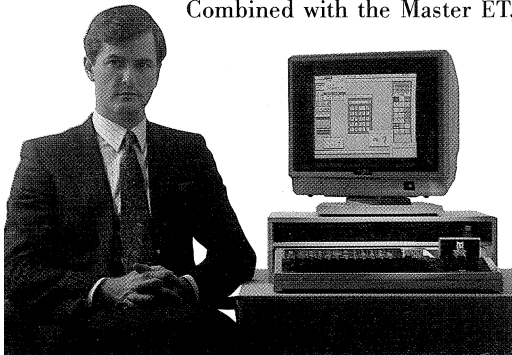
As an upgrade of the 128, the Master 512 gives higher levels of education access to MS-DOS compatible applications.

For under £500.

Or for only £348.26 exc. VAT you have the Master ET – a powerful, custom-built networking terminal.

Our ECONET networking facility is probably the world's best selling, low cost network.

Combined with the Master ET,



it affords considerable cost saving without limiting your expansion options, ensuring the BBC Micro's place throughout our education system.

PROFESSIONAL.

Business dictates that a microcomputer is both efficient and easy to operate.

The Master 512 puts all the power of 16-bit computing under your control, for under £1,000.

This control comes courtesy of the GEM Collection which provides an icon-driven, simple and easy to use environment, as well as a word processor and painting package.

Not only does this allow you to write documents easily, but also to insert charts and graphs into them. Thereby improving your business presentations.

Though business is only one of the areas in which the Master 512 will be taking control.

In the Medical field, it could be taking the drudgery out of repeat prescriptions.

At the same time monitoring dosage for optimum patient care.

Existing BBC Micros are capable of running an entire factory.

The more sophisticated Master 512 could check every stage in a manufacturing process from beginning to end.

Of course, your immediate needs may not be on quite this scale.

So you could buy the Master 128 now and easily upgrade to the Master 512 at a later stage.

THE LABORATORY.

As a simple upgrade to the Master 128, the Master Sc. provides a powerful scientific computer for under £1,500.

A tenth of the price of its nearest mainframe equivalent.

The Master Sc. forms a natural and compatible link with the first in the extremely powerful Acorn Cambridge Work Station range.

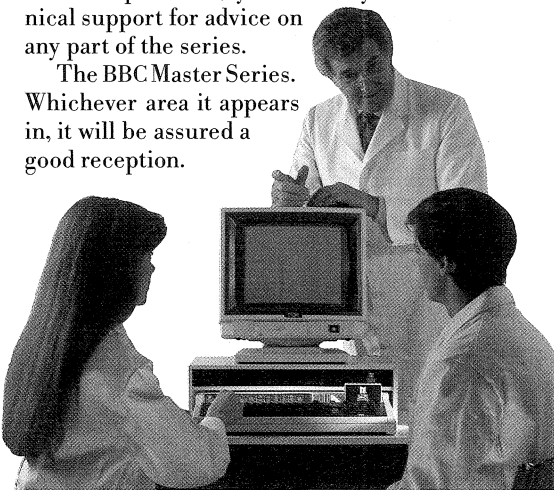
Naturally it uses all the important scientific languages – ISO PASCAL, FORTRAN 77, C and 32016 ASSEMBLER as well as BBC BASIC.

For anyone who is teaching languages, the Master Sc. is a particular asset.

You can now afford to teach more by practice rather than theory.

After purchase, you can rely on our technical support for advice on any part of the series.

The BBC Master Series. Whichever area it appears in, it will be assured a good reception.



BRITISH BROADCASTING CORPORATION
MASTER SERIES MICROCOMPUTER

The BBC Master Series.

Acorn
The choice of experience.

ADVANCED DISK FILING SYSTEM. *THE MASTER 512. 512K RAM 128K ROM CAPACITY. 80186 PROCESSOR – GEM COLLECTION FROM DIGITAL RESEARCH. MOUSE. TRA 64K RAM CMOS PROCESSOR. HI-BASIC. HI-EDIT AND PRINTER-BUFFER EXTENDER. *THE MASTER SC. NATIONAL SEMICONDUCTOR 32016 PROCESSOR. 512K CAMBRIDGE TECHNOPARK, DEPT (BB1), 645 NEWMARKET ROAD, CAMBRIDGE CB5 8PD.

*Check availability with local dealer.

FLEET STREET

CITY OF LONDON EC4

Editor

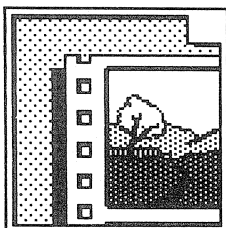
MIRROSOFT

Welcome to the world of personal publishing on your BBC Micro! **FLEET STREET EDITOR** gives you everything you need to produce professional-looking printed materials – in one package!

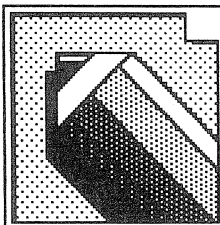
Just £39.95 including VAT!

A QUICK TOUR ROUND THE DEPARTMENTS

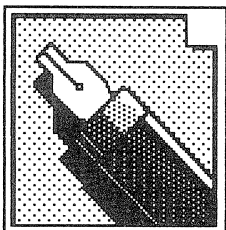
Each Fleet Street Editor page is constructed as a series of panels, with up to 8 panels per A4 page. Here's where you construct those panels:



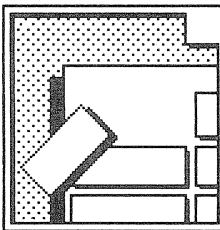
GRAPHICS LIBRARY
200K (about 400 pieces) of ready-to-use art – illustrations, symbols, and display typefaces which can be incorporated into your own work.



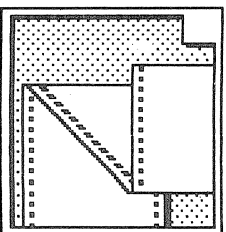
STUDIO
Where you modify existing art from the Library or your own collection, or create your own. All the graphics in the Library were created here.



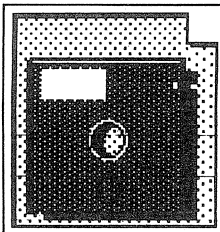
COPYDESK
Input your text here, using either the program's word processor or existing View or Wordwise files, and watch it wrap automatically round your graphics. Use different fonts, too.



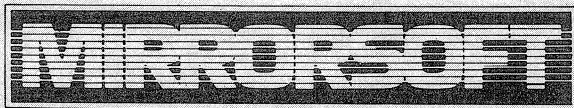
PAGE MAKE-UP
Define the format for your page and construct it from the catalogue of panels you've built up. Some panels are provided ready-made for you to experiment with.



PREVIEW AND PRINT
Preview a reduced representation of your completed page on screen, approve it, and send it to your printer using one of the built-in printer-drivers – you can write your own, too.



ADMINISTRATION
Set up the program to suit your system configuration, format disks, integrate graphics from most Mode 4 programs, and further manipulate Fleet Street Editor files.



"FLEET STREET EDITOR is THE package for anyone with access to a BBC Micro who wants to produce printed materials that include graphics... The Graphics Library is a marvellous idea... The Studio is a really creative area with all the tools you need... Without a doubt, this is a wonderfully powerful and flexible package... FLEET STREET EDITOR has been cleverly conceived and brilliantly executed."

— Dave Futcher, *Educational Computing*

WRITE FOR BROCHURE AND FURTHER DETAILS

MIRRORSOFT LTD
FREEPOST BS4382, Paulton,
Bristol BS18 5BR

FLEET STREET EDITOR

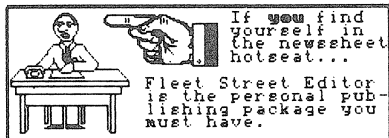
Issue No. 1

January 1986

GIVES YOU WORDS ... AND PICTURES

In any way you wish - together

or separately ...



If you find yourself in the newsheet hotseat...

Fleet Street Editor is the personal publishing package you must have.



You can use text in many ways: regular, bold, ~~underline~~ and, of course, with fancy

Headlines & Titles
in your text.

All this, and more, can be done in this one package - with nothing extra to buy!

Mirrorsoft



BEEBUG SHOWROOM AND MAIL ORDER

Items in this catalogue are available in our St. Albans Showroom and on Mail Order from the St. Albans address, at the time of going to press. If you are making a special journey, please telephone us first if there is anything you particularly want to see.

Come along and see us. Discuss your requirements with our staff. Ask for a demonstration of hardware or software (including the new release Beebugsoft range). Or simply browse amongst the items on display.

- **Computers**
- **Printers**
- **Monitors**
- **Disc Drives**
- **Consumables**
- **Budget Discs**
- **Software**
- **Books**
- **etc, etc**

● We stock the complete range of Beebugsoft products as well as back issues of BEEBUG. Our stocks also include a wide range of BBC Micro-computer accessories including monitors, disc drives, printers, software and books.

● We take both Access and Visa credit cards, simply telephone us on our new multi-line telephone number 0727 40303 quoting: Cardholder's Name, Cardholder's Address, Card-

holder's Telephone number, Credit card number, expiry date, order details.

NOTE: that the mail order despatch address must be the same as the cardholder's address.

● Our new mail order operation from St Albans will be in ADDITION to our current mail order service from High Wycombe. If you are ordering ONLY Beebugsoft products or special offers, then please send your order to High Wycombe as usual.

BEEBUG MAIL ORDER

Dolphin Place,
Holywell Hill, St. Albans,
Herts. AL1 3YS
Tel: 0727 40303
Multi-lines

HOURS

9 - 5.30 Monday - Saturday
(Switchboard closes at 5pm.
Answerphone 24 hours.

COMPUTERS

BBC Master Series

The Master 128 from Acorn is now available and offers super value for money, coming with VIEW, VIEW-SHEET, BASIC EDITOR and TERMULATOR software "bundled" in. Come and see the new-look Master in our Showroom. There are also limited stocks of the Master Turbo, with second processor on board and "Hi-" versions of the software.

BBC B+

There are still limited supplies of these machines available at special prices. Please contact us for our stock situation & current prices.

Code	Description	Price inc. VAT
0200	BBC Master 128 Micro	499.00
0203	6502 Second Processor	189.00
0204	Z80 Second Processor	389.00
0205	Acorn Prestel Adaptor	134.00
0206	Acorn Teletext Adaptor	144.00
0212	Acorn Speech Upgrade	46.00
0209	Acorn 1.2 DNFS ROM	19.50
0217	1770 Disc Controller Upgrade Kit	49.00
0218	ADFS ROM for 1770 DFS	29.00
0223	Acorn 1.2 DNFS disc Interface Kit	89.00
0213	Wafford DFS ROM + Manual	26.50
0214	Wafford DDFS Kit + Manual	98.00
0216	Wafford DFS Manual	6.95

DISC DRIVES AND DISCS

Beebug continue to recommend the Cumana range of drives.

Beebug Dynamic Discs are made by the world's leading magnetics company for us. Every pack includes a FREE library box or storage box. Each disc is guaranteed for life.

Code	Description	Price inc. VAT
0600	Cumana 100K CSX100 40T	97.00
0610	Cumana 100K CS100 40T inc PSU	127.00
0601	Cumana 200K CDX200 40T Dual	187.00
0602	Cumana 400K CSX400 40/80T	132.00
0603	Cumana 400K CS400 40/80T inc PSU	152.00
0604	Cumana 800K CD800S 40/80T inc PSU	282.00
0650	Memorex Discs S/S D/D 48tpi	15.50
0651	Memorex Discs D/S D/D 96tpi	25.50
0652	Memorex Disc Head Cleaning Kit	9.65
0653	Library Case — 10 discs	2.25
0654	Disc Storage Box — up to 50 discs	9.00
0655	Lockable Disc Box — 35	13.80
0656	Lockable Disc Box — 80	17.25

BEEBUG DYNAMIC DISCS

0657	10 × 48tpi SS DD Discs + Free Box	*14.90
0658	10 × 48tpi DS DD Discs + Free Box	*20.50
0659	10 × 96tpi SS DD Discs + Free Box	*20.50
0660	10 × 96tpi DS DD Discs + Free Box	*21.90
0661	25 × 48tpi SS DD Discs + Free Box	*34.90
0662	25 × 48tpi DS DD Discs + Free Box	*46.20
0663	25 × 96tpi SS DD Discs + Free Box	*46.20
0664	25 × 96tpi DS DD Discs + Free Box	*49.90
0665	50 × 48tpi SS DD Discs + Free Box	*59.30
0666	50 × 48tpi DS DD Discs + Free Box	*82.40
0667	50 × 96tpi SS DD Discs + Free Box	*82.40
0668	50 × 96tpi DS DD Discs + Free Box	*93.50

OTHER HARDWARE

ROMS & RAMS

Our 2764 8K EPROMs are still just £2.30 and the 27128 16K version is £3.45.

"Shadow" RAM to increase user memory in Modes 0 to 7 can be added with the ARIES B32 RAM Board. The Wafford 32K RAM card can sit inside the Beeb together with most ROM expansion Boards. Remember Beebug's special price of £39.00 for the ATPL Sidewise ROM board. A version for the B+ is now available at £29.00.

Code	Description	Price inc. VAT
0700	Aries B20 RAM card	86.50
0701	Aries B32 RAM card	91.00
0702	Wafford 32K RAM card	69.00
0703	AMX Mouse	79.95
0704	ATPL Sidewise ROM Board for Model B	*39.00
0735	ATPL Battery Backup Kit	17.00
0746	ATPL Sidewise ROM Board for B Plus	29.00
0705	2764 8K EPROM	2.30
0706	27128 16K EPROM	3.45
0707	6264 8K RAM	4.00
0708	EPROM Extractor Tool	5.00
0709	EPROM Eraser (UV)	32.00
0710	Softlite EPROM Blower	75.90
0711	Pace Nightingale Modem	120.00
0712	Pace Commstar ROM & Manual	33.00
0713	2 into 1 Phone Adaptor	6.32
0714	BB-Key Function Key Labels	2.00
0717	Wafford Zideways ZIF Socket	17.25
0718	Voltmace Delta 3B Single Joystick	11.99
0719	Voltmace Delta 3B Twin Joystick	19.95
0720	Voltmace Delta 14B Joystick/Keypad	14.95
0721	Voltmace Delta 14B/1 Interface	14.95
0742	Voltmace Delta Driver Disc 40T	9.00
0743	Voltmace Delta Driver Disc 80T	9.00
0722	4-way 13 amp Switchable Plug Block	10.35
0724	Antex CN17 Soldering Iron	6.79
0726	8-way DIP switch	0.98
0727	RS 432 "Domino" Plug	0.46
0728	RGB 6 pin DIN Plug	0.46
0729	Cassette 7 pin DIN Plug	0.30
0730	3-core 6 amp Mains Cable (per m)	0.35
0731	4-core RS432 Cable (per m)	0.20
0732	6-core RGB or Cassette Cable (per m)	0.25
0734	BBC Model A or B Replacement Keyboard	46.00

MONITORS

Colour Monitors

The Phillips CT2007 and CT1114 14" TV/RGB Monitors continue to be very popular and the Microvitec RGB Monitors are, of course in great demand. To take the place of the discontinued Sanyo 3115 Ultra High Resolution Monitor, Beebug are pleased to announce the TAXAN SUPERVISION 3, a 12" Monitor which, using state of the art technology, produces quite astonishing resolution and clarity.

The TAXAN Monochrome Monitors are superb value for money and offer a range of three phoshors to suit exacting requirements.

Code	Description	Price inc. VAT
0500	Taxan 1201 12" Green, Short Persist.	103.50
0501	Taxan 1202 12" Green, Long Persist.	120.75
0502	Taxan 1203 12" Amber, Long Persist.	120.75
0535	Tilt/Swivel Stand for Taxan 1201-3	25.00
0530	Microvitec 1431 Standard Res.	218.50
0531	Microvitec 1451 Medium Res.	276.00
0536	Taxan Supervision 3 Very Hi-Res.	420.00
0535	Tilt/Swivel Stand for Supervision 3	29.00
0532	Phillips CT2007 TV/RGB Monitor	225.00
0534	Phillips CT1114 TV/RGB Monitor	243.00
0550	RGB Monitor Lead, 6 pin DIN	5.75
0551	Mono Monitor Lead, BNC, for Taxan	3.45
0537	Supervision 3 Monitor Lead	8.00
0552	Single Plinth for Monitor	12.65
0553	Double Plinth for Monitor + Drives	21.85
0554	Memorex TV and VDU Cleaning Kit	4.51

PRINTERS

Letter Quality

The Brother HR15 printer has easily changeable printwheels and will accept sheet feeders, tractors and even a keyboard! At the budget end of Letter Quality is the Ibico LTR1 at just £138.00.

Colour Printers

The CANON PJ1080A Full Colour printer still surprises everyone with the clean, almost dot-free printout. It does not need special paper and is perfect for Graphics Freaks! The Epson JX100 is ideal for both coloured listings & documents and graphics dumps.

NLQ Matrix Printers

Both the Epson LX80 and the Taxan KP810 offer a Near Letter Quality printing mode. We now stock the amazing "Fontaid" package for the Taxan which gives you at least an extra 8 NLQ fonts. An extra disc provides even more, including Script and handwriting fonts and even 24cpi fonts! NOTE: "Fontaid" needs a RAM chip (our code 0707) fitted to the Taxan printer.

Code	Description	Price inc. VAT
0300	Brother HR15 Daisywheel Printer	373.75
0301	Canon PW1080A 10" NLQ Matrix	340.00
0302	Canon PW1165A 15" NLQ Matrix	405.00
0303	Canon PW1080A 10" Full Colour	460.00
0304	Epson LX 80 NLQ Matrix	270.00
0313	LX 80 Tractor Feed Unit	23.00
0305	Epson RX100	392.00
0307	Epson FX100	490.00
0308	Epson JX80 Colour	569.00
0309	Ibico LTR1 Print Band Printer	138.00
0311	TAXAN KP810 NLQ Matrix	263.35
0312	Linear Graphics Plotmate	333.50
0400	Platen Cleaner Kit for Printers	6.00
0401	2m BBC — Centronics Printer Lead	10.00
0402	Ribbon for Epson FX/RX 80 series	5.00
0403	Ribbon for Epson FX/RX 100 series	10.00
0413	Ribbon for Epson LX80	5.50
0405	Brother HR15 Replacement Printwheel	20.50
0408	Ribbon for Taxan KP810/Canon PW1080A	5.75
0411	Brother HR15 Correctable Ribbon	4.30
0412	Brother HR15 Nylon Ribbon	4.30
0414	Ibico LTR1 Ink Roller	4.60
0415	Colour Ribbon for Epson JX80	14.95
0419	"Fontaid" ROM & Disc 40T	30.00
0420	"Fontaid" ROM & Disc 80T	30.00

Code	Description	Price inc. VAT
0421	"Fontaid" Extra Disc "A" 40T	15.00
0422	"Fontaid" Extra Disc "A" 80T	15.00
0450	Listing Paper 9.5" × 66 lines 2000	14.50
0451	3.5" × 9 lines labels, 2-across 1000	5.00
0452	"Real A4" W/P Paper 80g 2000	28.75
0453	2-part NCR Listing Paper 2000 sheets	23.00

SOFTWARE

BEEBUGSOFT's HELP II ROM has become an indispensable tool for those who can't remember all the *FX and VDU commands and BASIC syntaxes etc. 28,000 characters of text about the Beeb crunched into a 16K EPROM together with special search routines!

"ROMII" is an Official Ram Filing System (RFS). It allows you to use sideways RAM or even user RAM to produce ROM versions of your programs. You can use it as a printer buffer or as a 16K "Silicon Disc". Send for our BEEBUGSOFT software catalogue for full details.

Code	Description	Price inc. VAT
1000	Acornsoft VIEW Word Processor ROM	53.00
1021	Acornsoft VIEWINDEX Disc	14.95
1001	Acornsoft VIEWSHEET Spreadsheet ROM	53.00
1002	Acornsoft HI-VIEW ROM & Disc	55.00
1019	Acornsoft VIEWSTORE Database ROM	53.00
1022	Acornsoft VIEW 3 ROM & Disc	89.70
1003	Acornsoft LISP ROM	49.00
1004	Acornsoft BCPL ROM	56.00
1005	Acornsoft COMAL ROM	49.00
1006	Acornsoft LOGO ROM	69.00
1007	Acornsoft ISO PASCAL ROMs	69.00
1008	Acornsoft BASIC EDITOR ROM	29.00
1013	VIEW Printer Driver Generator Tape	9.90
1014	VIEW Printer Driver Generator Disc	11.50
1012	Acornsoft ELITE Disc	17.00
1023	Acornsoft ELITE Tape	14.95
1015	Acornsoft REV5 Disc	17.00
1024	Acornsoft REV5 Tape	14.95
1016	Acornsoft AVIATOR Disc	17.00
1017	Acorn Graphics Extension ROM B	29.00
1018	Acorn Graphics Extension ROM B+	29.00
1009	AMX Desk Disc	21.95
1010	AMX Utilities Disc	13.45
1020	AMX Paint Pot Disc	13.45
1011	BBCsoft UltraCalc 2 Spreadsheet ROM	75.00
1118	BBCsoft White Knight 12 Chess Disc	19.00
1119	BBCsoft White Knight 12 Chess Tape	9.50
1102	C.C. Wordwise ROM	*35.00
1100	C.C. Wordwise Plus ROM	*46.00
1112	C.C. Hi Wordwise Plus Disc	* 5.75
1106	C.C. Accelerator ROMs	*56.00
1109	C.C. Disc Doctor ROM	*28.00
1103	C.C. Graphics ROM	*25.00
1110	C.C. Intersheet Spreadsheet ROMs	*47.70
1111	C.C. Hi Intersheet Disc	* 5.75
1114	C.C. Interchar ROM	*29.90
1104	C.C. Printmaster (Epson) ROM	*28.00
1105	C.C. Printmaster (Star) ROM	*28.00
1107	C.C. Speech ROM	*28.00
1108	Speech Processor (Only with 1107)	*10.00
1122	Superior Software SPEECH1 on Disc	11.45

BOOKS

We also stock a wide range of books, please send for a list.

CARRIAGE

Please **add £1.00** carriage to all orders except Computers, Monitors, and Printers **add £7.00 per item.**

Disc Drives, 2nd Processors, Modems etc **add £5.00 per item.**
Listing Paper **add £2.00** per box.

ITEMS MARKED *

Price shown includes postage.

OFFICIAL ORDERS

Official orders from education or government bodies are welcome.

CREDIT CARDS

Access and Barclaycard orders are welcome, and may be placed by telephone.

FURTHER INFORMATION

Please give our help line a call if you require further information.

TELEPHONE

St.Albans (0727) 40303

OVERSEAS

Please enquire for special postage rates.



Order Form

BEEBUG MAIL ORDER

Dolphin Place, Holywell Hill,
St.Albans, Herts. AL1 1EX
(Dolphin Place is opposite 18 Holywell Hill
in the centre of St.Albans)

CODE	QUANTITY	DESCRIPTION	UNIT PRICE	TOTAL

NAME:	SUB TOTAL	£
ADDRESS:	POSTAGE	£
	TOTAL	£

POSTCODE:	Membership No*
	<input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/>
DAYTIME TELEPHONE No:	Card Expiry Date
	<input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/>

ACCESS/BARCLAYCARD NO:	<input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/>
------------------------	--

In accordance with our competitive pricing structure prices are subject to alteration at any time.
Mail order despatch address must be same as cardholders address.

*Membership number may be found on magazine address label.

ROMIT — Put your own basic programs into Rom

- Create Your Own Silicon Disc
- A Complete Filing System with up to 16K capacity (per bank of Ram)

ROMIT adds a new feature to your BBC Micro — A Ram Filing System. This provides over 20 new commands which enable you to:

1. Create a silicon disc with its own filing system commands.
Type **★RAM**, then you can catalogue the "disc", save and load files to it, storing as many files as memory will allow. **★DISC** (or **★TAPE**) takes you back to your normal filing system. You will need sideways Ram to use this facility.
2. Put your own Basic or assembler programs (or any files) into Ram and then make them into Eproms. Whether it's your favourite game, your function key definitions or any program that you use frequently, you can now put it into Eprom so that it's always instantly available. Multiple copies can also be produced to enable you to pass on your programs to other people.

Getting your own Basic program into Rom couldn't be simpler. Just load your program into memory and type:

★RAM ★TRANSFER "prog" ★DOWNLOAD

and either connect it to a blower, or send the disc (or tape) to us, together with your **RomIt** registration number, and we will Rom it for you. (There is a small charge for this service).

3. Use your sideways Ram as a printer buffer so that you can continue to use your computer for other work at the same time as printing long documents.

```

>#HELP ROMIT

★RAM          ★SAVE          SAVE
★TRANSFER     ★LOAD         LOAD
★ACCESS       ★SPDOL        OPENOUT
★DELETE       ★TITLE        PRINT#
★RENAME       ★HELP         BRUTH
★SNAP         ★DOWNLOAD     #END
★RELOAD       ★UPLOAD       *PAD
★EXEC         ★INFO         *PRINT

★RAM
★INFO

MY PROGRAM L  FF1900  FFB023 1CDE 806C
MY DATA   L  FFC0C0  000000 0100 0223
1BOOT      L  000000  000000 007C 0341
1HELP      L  000000  000000 0044 0309

End of Rom: 8A439   Free bytes : 81CC6

>*PAD
>*DOWNLOAD
Ready to blow ROM data at 81900
>

```

OVER 20 NEW ★COMMANDS ARE PROVIDED BY ROMIT

An Eprom Blowing Service is offered with this pack.

To get the most from **RomIt** you will need sideways Ram fitted to your computer, eg just a single Ram chip plugged into your ATPL (or other) Rom board.

**MEMBERS PRICE
£25.50**

NEW

Classified Ads

Rates are 18p (inc VAT) per word for members' personal ads. Members' business ads cost 30p (inc VAT) per word - and if you are selling any item in quantities rather than one-off, then that counts as business. Send cheque with order to **PERSONAL ADS, PO BOX 50, ST. ALBANS, HERTS AL1 3YS**. It is essential to include your membership number. Please note that we cannot arrange for box numbers.

MARKS AND STATISTICS for teachers and lecturers. Revolutionizes the handling of marks. Simple to use spreadsheet format. 330 students per file. Up to 300 subjects. Sorts, analyses, normalizes, calculates means, applies weighting factors. Prints histograms and lists. Model B. O.S.1.2. Full documentation. Disc (40 or 80 track) £16. In-Form, 73 Woodfield Park, Colinton, Edinburgh EH13 0RA.

Problems? try R-SOFT utilities!

1. **HOW-TO:** An essential collection of software and instructions for frustrated new disc owners who want to move their programs to disc.

3. **ROMPULL + TAPE_DUMP.**

4. **AUTOMATIC DISC MENU:** Includes sideways-RAM version, can boot from ROM, works with 2nd 6502.

5. **AUTOSOL:** Will automatically boot your Solidisk with a pre-selected list of ROMs.

6. **SWROM*:** Puts your Basic / machine code programs in ROM format.

7. **RFS-GENERATOR:** Generates ROMs for the *ROM filing system. This does not use the DFS workspace and is an ideal tool to run nasty programs from disc.

All above packages £5.00 each.

8. **D-MASTER:** Superb disc backup program (unprotected): £7.00.

Many other utilities (BBC only - sorry!): send for our list (sae please). R-SOFT, 22 Marriotts Close, Felmersham, Bedford, MK43 7HD (0234) 781730.

Aries-B20 shadow RAM and View 2.1 with manuals, £69. Ring 0865 723989.

Wanted - View expert to help over problem of getting printer to print out what is on disc. Contact Sam Jaffa c/o Room 3066, BBC Broadcasting House, Portland Place, London. Tel. 01 927 4477. Fee considered.

UMI IB 16 track polyphonic Midi sequencer and hundreds of DX7 sounds on disc. £295. Epping 76672.

The foods you eat - a program to calculate the calories, fat, protein, carbohydrate, vitamins and minerals in 470 foods, then compare your intake with the recommended levels. Disc only - please specify 40 or 80 tracks. £12.50 from David Bender, 17 Hightrees Court, Manor Court Road, London W7 3HD.

Have your own programs put on EPROM. Accessed by *command, no waiting for tapes to load. Up to 8K - £10, 8 to 16K - £15, plus 75p p&p. Send your program on tape together with your address, *command and payment to J.Izat, 16 Longlands Court, Winslow, Bucks. MK18 3QA. No copyright or protected programs please.

Acorn Z80 second processor with latest edition software and manuals. Complete and in mint condition. £250.00. Tel. Ascot (0990) 25868.

Sale. Pace Nightingale modem c/w Commstar ROM. All leads, manual. £100. Tel. 01 455 8831.

BBC B, DFS, Wordwise, 100L disc drive, AP100A printer, books, lots of software on disc, joysticks. Excellent condition. £400 the lot ono. Will split. Tel. 0209 719312.

Educational software. Wide variety of titles, pre-school to adult. Fllet street editor, Island Logic Music System. BBC Soft, all BESA titles, etc. Catalogue from TalkTapes (BBG1), 13 Croftdown Road, London NW5 1EL.

Wanted. To buy or hire; Which? 'Taxcalc' for financial year 1983-84. Clayton, 40a Birkbeck Road, London NW7 4AA. Tel. 01 959 7376.

Programmer required, familiar with beeb systems including Teletext and Prestel, for development of data acquisition and financial program. Would consider help, co-development. Please tel. 01 851 7266 (S.E. London), evenings only.

For sale: Star DP510 dot-matrix printer, boxed, cable for BBC, good condition. £140.00. Tel. Watford (0923) 52475 or Hatfield 75784 (day).

Customised ROMs from £12. A new concept in sideways ROMs. Command clashes impossible as you decide up to 25 commands and we implement them for you. Options available include printer driving (e.g. *ITALIC), function keys set up (e.g. *KEYS1). Switch-on options also available to set up your choice of soft keys, start in lower case, do *TV, alter the beep, show a message on the screen etc. etc. Send for further details and application form from Braeburn Software, Hawthorn Bank, Scott's Place, Selkirk.

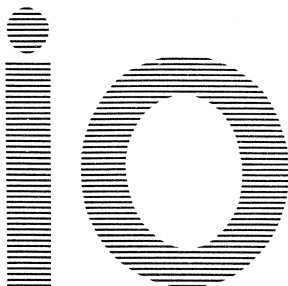
Advanced Printer Drivers for Wordwise or View

Two sophisticated printer drivers for BBC word processors and daisywheel or other printers.

VERSATILE drivers (for View) provide up to 20 word processing functions, including underscore, shadow, subscript, superscript, extra characters, blank, overstrike, change of printwheel in mid line, etc (depending on printer). Most functions can be easily redefined by the user to suit other makes of printer. Detailed manual supplied. Price £9.00

PROPORTIONAL MICROSPACING drivers (for daisywheel printers) give the ultimate in professional quality presentation. Text is smoothly spaced and right margin justified even when using proportional spacing printwheels. Price £12.00

HALF PRICE INTRODUCTORY OFFER. Buy two printer drivers, and pay only half price for the cheaper one. "Versatile" and "Proportional Microspacing" for just £16.50



Prices are for cassette; add £2.00 for dual format disc.
Please add 50p postage and packing on all orders.

Send sae for print samples.

Large range of printwheels available for most machines.

Credit card orders can be placed by phone.

Specify - Printer make and model

- Processor (Wordwise or View)

- Driver (Versatile or Proportional Microspacing)

Io Computer Communications (R)

7 Great George Street, Glasgow G12 8PD

041 - 339 5481

FREE printer initialisation program with every order.

NEW HIGH SCORES

Supplier	Game	Score	Player
A&F	Planes	57485	A. Butcher
ACORNSOFT	Carousel	44670	E. Somerville
ACORNSOFT	Labyrinth	510300	Cthulhu
ACORNSOFT	Revs	1:24.6	D. Whittam
BEEBUG MAG	Hedgehog	32760	C. Eberhardt
BEEBUG MAG	Xmas Pengo	25240	Angus
FIREBIRD	Gold Digger	25127	E. Somerville
ICON	Drain Mania	80400	Cthulhu
MICRO POWER	Laser Command	1010525	C. Eberhardt
SOFT SPOT	Fruity Freddy	24600	E. Somerville
SUPERIOR SOFT	DeathStar	223050	Cthulhu

Getting A Better

VIEW (Part 1)

For a professional approach to wordprocessing, View has much to commend it to the discerning user. Chris Wragg shows how to make the most of this product and reveals some of its more useful secrets.

INTRODUCTION

Acornsoft's View wordprocessor is one of the best available for the BBC micro. It has, however, received a somewhat critical press, which is not totally deserved, primarily because many of its best features are not readily apparent until you have used it for some time. Whilst the documentation supplied is very good, many of these facilities are only covered briefly, or not at all. This article sets out to redress the balance by providing a wealth of useful advice on making the most of View, and by showing how to overcome some of the limitations of earlier versions.

The examples given assume that your system includes a disc drive. However, there is still much of interest to those just using a cassette system.

EXEC FILES

Most disc owners will be familiar with setting up the !BOOT file to initiate the calling of a Basic program on pressing Shift-Break. However, its use is not limited to this. It can also be used to set up your chosen initial conditions for View. Example 1 shows a typical !BOOT file used for this purpose.

EXEC files can also be used at any time from the command mode of View, as any * command may then be entered. Indeed anything that you can key in from command mode can be put into an EXEC file, which thus may contain both View commands and OS * commands. This feature is a very

A TYPICAL !BOOT FILE

<u>Command</u>	<u>Purpose</u>
*FX3,2	Turn off VDU.
*FX21,0	Flush keyboard buffer.
*FX18	Clear function keys.
*TV255,1	Adjust screen display.
MODE 7	
*WORD	Enter View.
*DIR :0,\$	Set drive/directory.
*LIB :0,\$	Set library ditto.
A. :0.W.	Unlock workfiles.
*FX228,1	Enable Ctrl/Shift function keys.
*LOAD "KEYDEF1"	Load function keys.
*Mprint	Load routine to allow printing from memory.
MODE 3	{Set mode 3, so default
NEW	{ruler will be 74.
PRINTER EpsonFX	Load printer driver.
*FX202,48	Turn off Caps Lock.
*FX3,0	Turn on VDU.
MODE 7	40 character display.
*TYPE :0.\$,TITLE	Type my title page.

Example. 1

powerful way of carrying out complex, frequently used, functions. For example, the following should change abbreviated months to their full spelling:

```
CHANGE Jan. January
CHANGE Feb. February
CHANGE Mar. March
CHANGE Apr. April
CHANGE Jun. June
CHANGE Jul. July
CHANGE Aug. August
CHANGE Sept. September
CHANGE Oct. October
CHANGE Nov. November
CHANGE Dec. December
```

This might be useful, for example, in expanding out abbreviated data generated in a report from a database package. A file with these commands could be created using *BUILD <filename>. The file could then be called and executed from within View by typing *EXEC <filename>. As always with View, the text will require reformatting afterwards, though the FORMAT command could always be included at the end of the EXEC file above.

SPOOLING

View contains no facility for directly

```
*FX5,0      Set printer sink
*SPOOL <spool-file>
PRINT <file-to-be-spoiled>
*SPOOL
*FX5,1      Reset to parallel printer
```

However, View text files are just ASCII files anyway, so there is often no need for the above sequence. Just use the `SAVED` View file as it is.

View includes a SCREEN command which allows for the previewing of text, but only from the beginning of a document. If one needs to preview a single page in a long document this can be somewhat inconvenient. The Printer Driver Generator package includes a Screen Driver, but this aims to simulate the effect of some standard highlight codes and thus cannot be used in mode 7.

As an alternative, the printer output can be redirected to a sink (i.e. lost) using `*FX5,0`. SHEETS and PRINT may now be used to obtain a preview. Normal printer output may be restored with `*FX5,1` (parallel printer) or `*FX5,2` (serial printer).

View does not provide directly for generating documents in a dual column format, like BEEBUG Magazine. However, a function key may be programmed to provide an equivalent, e.g.:

*KEY 4 |!"5|!#1|!|Q|!#2|!|Z|!|P|!#5|!"6
|!|Q|I|!|\|!|Z|!|P|!#6

first line of the left-hand column. Set marker 5 at the start of the line that is to become the first line of the right-hand column. Each press of Ctrl/Shift-f4 will now move one line into dual column format.

Earlier versions of View contain a fault in the way in which it handles TABbed sections of text following those containing a highlight code. View counts each highlight as one byte, whereas it occupies no space on printing. This results in the TABbed section being printed to the left of the correct position by the number of highlight codes preceding it on the same line. Thus:

Highlighted text Tabbed text

The way round this is to include extra spaces after the TAB, equal to the number of highlights that preceded it. This fault, which has now been cured in View A3.0, also affects centred and right justified text containing highlights.

Highlights do not appear to work within headers and footers. However, any highlight in operation on reaching the end of a page, will affect printing of both the header and footer. Additionally, it should be remembered that the format of a header is dictated by the ruler in force immediately prior to the PE command.

When printing, using the SHEETS command, difficulties may occur if a page is deliberately missed (with M). Any highlights in the missed page will not have been processed and may result in undesired effects in the resultant printout. Similar effects can occur with other embedded commands, such as SR, or changes in rulers. I know of no way of overcoming this, though the problem no longer exists in View version 3.

The input buffer, for the commands PRINT, SHEETS, SCREEN, SAVE, LOAD and READ, will only accept eleven characters following the command. Thus commands such

as PRINT :0.\$.<name> will fail, if <name> is seven characters long. The form of the failure varies:

- + PRINT, SHEETS or SCREEN just results in the => prompt without the file being printed
- + LOAD or READ causes a syntax error
- + SAVE causes a syntax error, if "Editing No File", otherwise it will result in the name following "Editing...." to be overwritten.

The way round this is to use filenames no longer than six characters, if you wish to include the drive and directory. Otherwise, set the drive and directory, using *DIR, and issue the command without including them. View version 3 deals with this correctly, allowing a full 12 characters when used with the DFS, and up to 19 when used with the ADFS.

RECOVERING FROM KEYBOARD LOCK-UP

View is a very robust piece of software and failures are very rare indeed. However, occasionally you may find that the keyboard locks up on you, with Break the only way out. Typing OLD should then recover the situation, but sometimes the dreaded "No text" message will result - all, however, is not lost. The text will still be in memory, starting at OSHWM + &100 (= PAGE + &100), i.e. &1A00 for the Acorn DFS, &F00 for the cassette filing system, and &800 with second processor.

Typing *SAVE PANIC 1A00 +length, or equivalent, where length is adequate to include all the text, will save it to disc or tape. Re-entering View and READING in the PANIC file will recover your text, which should only need minor tidying up.

PRINTER DRIVERS

A wide range of printer drivers is now available on the market, including a

Printer Driver Generator from Acornsoft (a configurable View printer driver generator was also published in BEEBUG Vol.4 No.1). It should therefore be unusual for anybody to need to write their own, except in special circumstances. The protocols, and formats to be followed, are given in the manual "Into VIEW" published with version A1.4. It does not appear in the A2.1 version.

A fault exists in View A2.1 which causes the printer driver, once loaded to be lost under certain circumstances. This is easily overcome by ensuring that on entering View, you type NEW before loading the printer driver, and then all should be well.

NUMBER REGISTERS

Those who wish to write software using the number registers will find them held (in View A1.4 and A2.1) as low-byte, high-byte starting with register A at &798, &799 going up to register Z at &7CA, &7CB. Acornsoft confirm that these locations remain the same in version 3 (and will remain fixed in any later versions that may appear).

PRINTING FROM MEMORY

View A1.4 requires that files be saved, and then printed from disc, cassette etc. They cannot be printed from memory. This restriction is particularly awkward for cassette users. The Printer Driver Generator package includes a utility, Mprint, to overcome this. Mprint, is loaded at page &C, and when a PRINT, SHEETS or SCREEN "" command is issued causes the file in memory to be printed in conjunction with the selected printer driver.

Next month we will look in detail at the opportunities available in View for customised programming of the function keys, a valuable facility often overlooked by View users.



HINTS HINTS HINTS HINTS HINTS HINTS HINTS

VIEW OR WORDWISE

View version 3 will recognise *W. as an abbreviation for the normal command *WORD, used when selecting VIEW. This is the same as Wordwise and Wordwise Plus use. If you have both word processors in your machine, which is chosen will depend on their relative priorities.



ANGELS AT 12 O'CLOCK

Three Flight Simulators

If you fancy flying with the Red Arrows or undertaking a sortie in a Harrier Jump Jet then one of these programs could change your life. Mike Williams has been trying out the controls.

Title : STRIKE FORCE HARRIER

Supplier : Mirrorsoft
Purnell Book Centre, Paulton,
Bristol BS18 5LQ.

Price : £9.95 (tape), £12.95 (disc)

Rating : *****

Title : JUMP JET

Supplier : Anirog Software,
8 High Street, Horley, Surrey.

Price : £9.95 (tape), £11.95 (disc)

Rating : ***

Title : RED ARROWS

Supplier : Database Publications
Europa House, 68 Chester Road,
Hazel Grove, Stockport SK7 5NY.

Price : £8.95 (tape), £11.95 (disc),

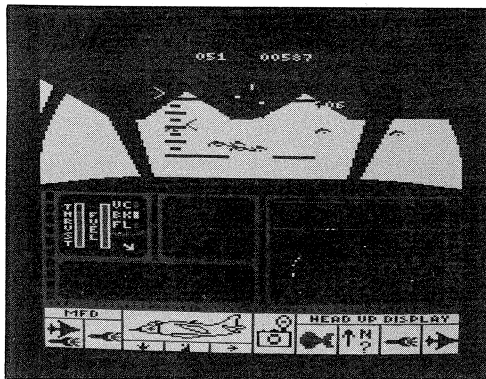
£12.95 (3" disc)

Rating : ****

STRIKE FORCE HARRIER - Mirrorsoft

This is one of two Harrier jump jet simulations to be released recently, and is already proving very popular in the best seller charts. The screen display is fairly typical of other aircraft simulations with various instruments and radar displays in the lower part of the screen, and external view out of the front of the aircraft in the top half.

The program uses either joystick or keyboard control. Basic flying skills are easy to master, and in some ways being a jump jet helps too. To take off, for example, you select downward thrust, increase power and the plane lifts off. Once you have gained a few hundred feet, change the thrust vector to 45 degrees and start moving forward. When you have gained enough speed, change the thrust vector to horizontal and the Harrier now behaves like an ordinary jet plane.



Landing is almost the reverse of this, but does require rather more care and skill, particularly in losing airspeed. The advantage of the jump jet is that you don't need to locate and land on a normal runway. You can touch down anywhere.

Once airborne, you can start practising your flying skills. The landscape shows small hillocks and mountains, and low level flying between mountains can prove quite exciting. Your plane is armed with bombs, to attack ground-based enemy targets, and Sidewinder air-to-air missiles for shooting down enemy aircraft. Luckily, the program provides a practice mode where the enemy doesn't fire back.

Once you feel you have mastered all the controls you can attempt your combat mission. This is to advance your ground support bases, knocking out any enemy forces on the way and ultimately locate and destroy the enemy HQ.

This is an excellent simulation, packed with interest and entertainment and all at just the right level of difficulty. It must certainly rank as one of the best aircraft simulations currently available.

JUMP JET - Anirog

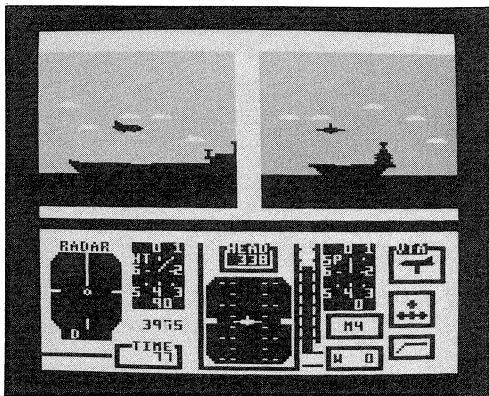
This is another Harrier jump jet simulation, but with a difference. This is the Sea Harrier, and your take off and landing point is the deck of an aircraft carrier. The plane can only be 'flown' using a suitable joystick, something which I think is regrettable, even if it might be claimed as more realistic.

As with Strike Force Harrier, the

bottom half of the screen display shows various cockpit instruments and radar. The top half of the screen shows one of three different views depending on the position of the aircraft. On take off, you have a bird's eye view of the deck of the carrier. From 50 to 200 feet, the top half of the screen is split showing a side and rear view of the carrier with the plane above. As before, you take off vertically and then progress gradually to normal horizontal flight. Once above 200 feet, the top half of the screen changes once more to provide a view of the sea and sky.

On returning to the carrier (assuming your navigation is good enough to find it) landing is the reverse of the above process, but again much more difficult. Unlike a landing on terra firma, you have to land on a moving runway!

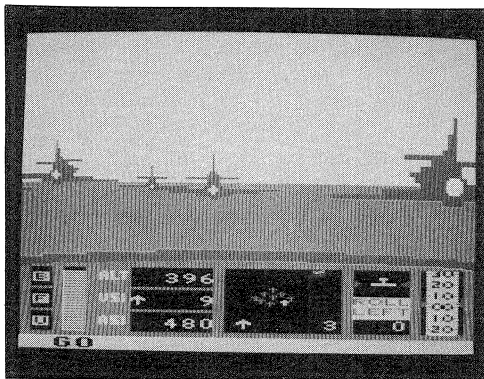
Your mission with Jump Jet is to seek out and destroy enemy aircraft with your missiles before returning to the carrier. You start with the rank of Flight Lieutenant and can progress up to Group Captain, with harder missions at each level.



Overall, I found both less enjoyment and less of interest in this program compared with the Mirrorsoft version. In my view, many potential users will find the skill levels required very difficult indeed to achieve, and this is inevitably a discouraging factor.

RED ARROWS - Database Software

The Red Arrows are a deservedly famous aerobatic team flying their all red Hawk aircraft. This first rate program from Database publications gives you the chance



to take part in a variety of aerobatic formations and manoeuvres.

In some respects, the actual flying is much easier to master as there are no take offs or landings to bother with. The plane can be flown with a joystick or entirely from the keyboard and you can select your own choice of keys. However, the keys are only listed on the screen, not in the booklet, so you need to note them down.

A practice mode allows you to try out all the different manoeuvres on your own before attempting the same thing in formation. With all my experience in flying everything from Spitfires, to 737 airliners to Harriers, I skipped this and went straight for formation flying, with predictably disastrous results. Starting off in V-shaped Big 9 formation, I was rapidly left behind as Red Leader accelerated away with the rest of the nine planes. Increase thrust and start catching up as the team moves into Diamond formation. The only problem now is that I'm going too fast and crash into the other planes! At least I am provided with a parachute and float safely back to earth. Pity about the others though.

The cockpit view of the other planes ahead in formation is really stunning, and makes quite a difference to the usual visual display, particularly in colour. After a few more repetitions with the same results I decide to take advantage of the demonstration mode, sit back in my armchair, and watch the Red Arrows put on a spectacular display just for me. Thoroughly recommended.



GAMES REVIEWS

by Daniel Gaster

Title : CITADEL
Supplier : Superior Software
Price : £9.95 (£11.95 on disc)
Rating : *****

Citadel is another in a line of graphic adventures which seem to be popular now. There will of course be compar-

isons with Castle Quest and Wizardore, although I believe that Citadel puts both of these in the shade. There are well over 100 rooms which form a huge fortress, an island with a temple as well as a pyramid maze and witch's house. The loading screen graphics and software speech synthesis are amazing as are the complexity of the rooms and the variety of their graphics. The object of the game is to collect five crystals that are scattered throughout the game. Only one is initially visible; the rest require long chains of objects to be

Title : GAMES DISC 3
Supplier : Blue Ribbon
Price : £9.95
Rating : ***

Blue Ribbon have released a number of new programs on one disc. All the programs can be bought separately on cassette for £2.50 each.

The first game is 'Astro Plumber' in which you walk or jet around a platform type screen sealing air leaks with a laser gun. 'Diamond Mine II' is the sequel to the popular 'Diamond Mine'. The nozzle of your vacuum pipe is guided around a maze sucking up diamonds. 'Ravage' is probably the best game on the disc. It is a 3D Starwars type space battle with nine levels of difficulty. 'Joey' is a kangaroo in a simple platform and ladder game with nine fairly challenging screens. A feature of this game is an aborigine who throws spears and boomerangs at Joey. The last

Title : MICROCOSM
Supplier : Firebird
Price : £3.00
Rating : ****

Another budget-price release from Firebird, this time in their 'Super Silver' range at £3.00 each. As always the instructions, packaging,

loading graphics, and multipart tune are of a very professional standard. The game involves flying around the screen with a jet-pack, carrying sections of ladder from the top to add to various other sections around the screen. Your aim is to make all the ladder sections into five complete ladders.

As well as this, an eye must be kept on the condition of the plants which you must feed with nutrients from time to time. Unfortunately you can only do this if you have kept the mutant insects at a

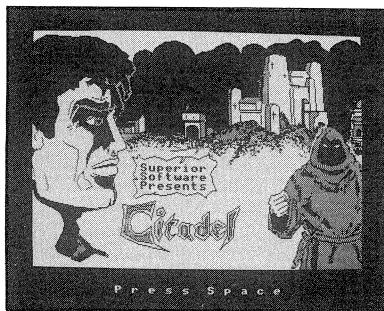
Title : STAR DRIFTER
Supplier : Firebird
Price : £2.50
Rating : ***

Another release from Firebird, this time an attempt to break into the graphic adventure market. The idea of the game is to explore a spaceship,

room by room, and to find clues as to the mysterious appearance of the whole fleet. The actual layout of the screen is very similar to Sabre-wulf, and in fact the whole game follows the same pattern.

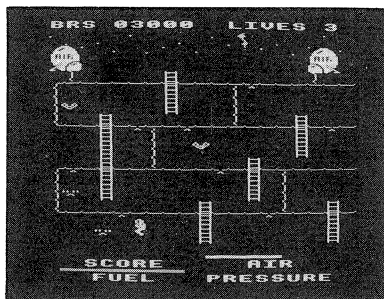
The main difference from Sabre-wulf is the ability to manipulate objects and carry them around. For instance, various keys are needed to pass through force fields scattered around the ship. Other objects like a radio, a gun (without which you cannot shoot monsters), and oxygenator unit are well defined on the screen.

manipulated in true adventure fashion before they can be found. Citadel combines arcade and adventure games so that clear thinking and nimble fingers are needed to get anywhere. I can think of no faults in the game and in my opinion Citadel is easily the best arcade-adventure on the market. There is not nearly enough room here for me to mention all the special features in the game. As a further incentive Superior Software are offering 2 cash prizes for the first people to complete various parts of the game.



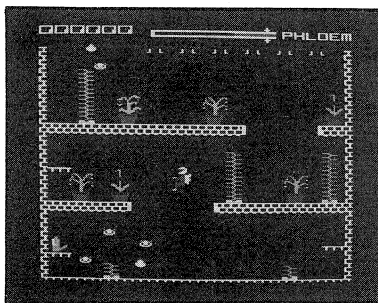
program is a drawing utility called 'Artist'. It has almost all the basic drawing functions (rubber-banding, filling, etc.) except for arc drawing (an annoying omission).

My main grumble about the games is the complete lack of keyboard instructions, only 2 of 5 games had them. Although most layouts can be guessed by trial and error it seems incredible that this omission could go unnoticed. Overall the package represents reasonable value for money and the standard of graphics is high.



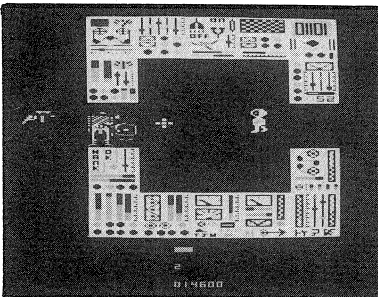
low enough level. In later screens, as well as the changing screen layouts, additional hazards are thrown in, such as spiders that appear and descend from the top of the screen.

This game is great value for money. I enjoyed playing it and found it very challenging without being frustrating. The number of things that need to be watched really keeps you on your toes. The standard of budget software is certainly rising.



Careful thinking is needed as only three objects can be carried at once. It is a tribute to the complexity of the game that it still needs to be mapped as there are 120 rooms in all.

The presentation of the game is up to Firebird's usual high standard. I found the game itself quite playable and even though it may not be quite so graphically stunning as other more expensive games, it is well worth buying in view of its low price.



After concentrating on Basic in past issues, Surac investigates some useful machine code routines to utilise the Beeb's internal timers. These give much more precise timing than Basic's TIME function.

Normally, these articles look at useful Basic techniques, but this month, we'll go into machine code. In particular, we will look at the timers in the Versatile Interface Adapter (VIA), the chip that, among other things, also drives the Printer and User Ports.

Sometimes, the 0.01 sec resolution which TIME provides is not accurate enough, even in Basic. The VIA, however, contains 2 timers which can be used to measure events to 1 microsecond. It also contains lots of other things, so first a look at what it offers.

6522 VIA.

The VIA would need several articles to do it justice but, briefly, has:

- Two 16-bit timers (T1 and T2) which can also be used as counters.
- Two 8-bit input/output (I/O) ports to communicate with the outside world.
- Two control lines for each port.
- An 8-bit serial I/O shift register.

One of the two I/O ports (Port A) is normally used by the printer. The second (Port B) is the User Port. Connections to both are accessible under the computer, and pinouts are

given on page 503 of the User Guide.

All the facilities provided by the VIA are programmable and many can be 'linked' - for example, a timer can control the shift register. Furthermore, the VIA can interrupt the CPU (Central Processing Unit - the 6502) as a result of many different, selectable, events.

The chip is programmed via sixteen 8-bit registers, which, for the user VIA in the BBC micro, are at addresses &FE60-&FE6F. Here are brief details of some of the more useful registers (the list is not complete):

<u>Address</u>	<u>Name</u>	<u>Abbreviation</u>
FE64	T1 Counter - Low byte	T1CL
FE65	T1 Counter - Hi byte	T1CH
FE66	T1 Latch - Low byte	T1LL
FE67	T1 Latch - Hi byte	T1LH
FE6B	Auxiliary Control Reg.	ACR
FE6E	Interrupt Enable Reg.	IER

While you can write to and read from all the registers, the effects are not always obvious. For instance, writing to T1CL actually puts the data into T1LL, the same result as writing directly to the latter. We won't use the second timer but it is similar to, but simpler than, T1. For full details of the chip, I must refer you to data sheets or the Advanced User Guide chapter 22.

CONTROLLING THE TIMER

Timer 1 is a 16-bit register which counts down at 1 Mhz. It thus takes 65.536 ms to go through one full cycle. To set it up, first write the low byte of the desired value to either T1CL or T1LL. Either way, it goes to the latch, not the counter. Then write the high byte to T1CH, which automatically puts the byte into BOTH latch and counter and, simultaneously, moves the low byte from its latch to the counter. The 16-bit value immediately starts decrementing in the counter, but is preserved in the latches.

The ACR controls just how the timer works. Bit 7 (the MSB) of the ACR makes a link between Timer 1 and the highest-bit

line in Port B. If zero, the line ignores the timer's operation. If set to "1", the line swaps level ("toggles") every time that the timer reaches zero, allowing the VIA to generate a square wave.

Bit 6 dictates just how the clock runs. If "0", the timer counts down to zero and, if required, gives an interrupt. This gives a way of generating a single-shot time delay. Once it reaches zero, it carries on decrementing, although there are no more interrupts. If, however, bit 6 is "1", whenever the timer reaches zero, it re-loads itself from the latches T1LL and T1LH. Normally, it would also generate an interrupt each time to give a steady "heartbeat" (e.g. the Beeb's clock).

USING THE TIMER

To demonstrate T1, let's provide a way of zeroing and reading it, either from machine-code or from Basic. This will give a way of measuring times with a 1 microsecond resolution. Remember, though, that we can only time up to 65.536 ms - after that the timer starts again from zero and we lose track of the total time.

The routines must do 3 things. The timer has to be initialised to free-run, it has to be zeroed, and it has to be read. Zeroing and reading the timer should take place as quickly as possible. Since T1 counts down, we must invert the value in T1CL and T1CH to produce a more useful count up when we read the timer. Finally, to make the routines compatible with Basic, the value read should be put into 4 bytes, so that it can be read as an integer. Not surprisingly, we have to control T1 by machine code - here's the code:

```
10000 DEF PROCassemble
10010 REM ** Set up system constants
10020 t1cl=&FE64:t1ch=&FE65
10030 t1ll=&FE66:t1lh=&FE67
10040 acr=&FE6B:ier=&FE6E
10050 :
10060 FOR PASS=0 TO 3 STEP 3
10070 P%=&A00:REM** &D10 for cassette
10080 timebuff=P%
10090 P%=P%+4
10100 [OPT PASS
10110 \
10120 \Initialise Timer 1
10130 \
10140 .timinit
```

```
10150 \
10160 \Prevent timer interrupts
10170 LDA #&40:STA ier
10180 \Set T1 to run continually,
      with no re-load
10190 LDA acr:AND #&3F:STA acr
10200 \Set T1 latches
10210 LDA #&FF:STA t1ll:STA t1lh
10220 \Clear hi bytes of time buffer
10230 LDA #0
10240 STA timebuff+2:STA timebuff+3
10250 RTS
10260 \
10270 \Start timer running from &FFFF
10280 \
10290 .timzero
10300 \
10310 LDA #&FF:STA t1ch
10320 RTS
10330 \
10340 \Read timer 1
10350 \
10360 .readtime
10370 \
10380 \Get counter values quickly
10390 LDA t1cl:LDX t1ch
10400 \Invert value, because
      T1 counts down
10410 EOR #&FF:STA timebuff
10420 TXA:EOR #&FF:STA timebuff+1
10430 RTS
10440 ]
10450 NEXT
10460 ENDPROC
```

There are 3 short routines involved. The first, 'timinit', is only used at the start, and sets the VIA running in the correct mode. Timer interrupts are cancelled and the latches set to an initial value of &FFFF. Why not zero, to count down from there? Simple - if you use zero, the chip does not run properly. The two highest bytes of the 4-byte time buffer are also zeroed.

The second routine, 'timzero', simply writes &FF to T1CH. At this, the &FF from T1LL is automatically loaded into T1CL and the timer starts counting down again.

The timer is read by 'readtime', the third routine, which pulls the 2 bytes of the current value into the A and X registers of the 6502. This is the quickest way of getting the time; once they have been read, the bytes are inverted (to count up) and saved in the buffer.

THREE MODEMS

Peter Rochford, well known for his interest in communications, has been trying out three of the latest modems for the Beeb.

Product : DATACHAT 1223
Price : £89.95
: £99.95 incl. software
Supplier : GEC Telecommunications Ltd.
P.O. Box 53,
Coventry CV3 1HJ.
Phone : 0203-446331

Product : MAGIC MODEM
Price : £99.95
Supplier : Datastar
Unicom House,
182 Royal College Street,
London NW1 9NN.
Phone : 01-482-1711

Product : LE MODEM
Price : £105.80
Supplier : Watford Electronics
Jessa House,
250 Lower High Street,
Watford.
Phone : 0923-37774

All prices are inclusive of VAT and p&p.

In BEEBUG Vol.3 No.9 we examined five modems and several terminal software packages for use with the BBC micro. The prices of the modems alone, at that time, ranged from around £80 to over £200. We now take a look at three recently launched modems, each costing around £100 and supplied complete with terminal software for the BBC. Can you really get a competent comms package for that price?

GENERAL INFORMATION

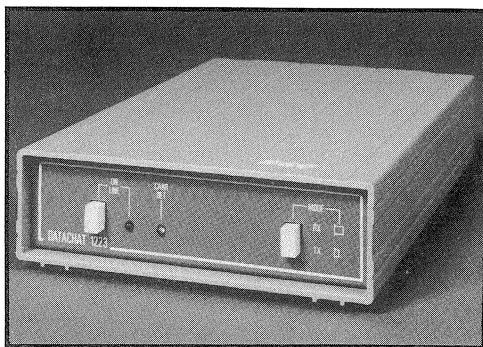
All three modems in this group review connect to the telephone line via a 2 metre lead terminating with the standard PST plug. Le Modem and the Datachat both

have a PST socket on the rear panel for connection of the displaced telephone, but the Magic Modem will need a two-way adaptor.

Connection to the computer in all cases is by a supplied lead that plugs into a DIN socket at the rear of the modem, and to the Beeb's RS423. Datachat is the only modem in this review that has BABT approval at the time of writing.

GEC DATACHAT 1223

The Datachat 1223 is a V23 modem (operating at 1200/75 baud and 75/1200 baud) making it suitable, with appropriate software, for Prestel, Telecom Gold, user-to-user file transfer and for many bulletin boards (BBs). It is also possible to use the Datachat in a 1200/1200 baud half duplex mode, the turnaround at both ends being achieved by software control.



The Datachat has an excellent standard of construction and is housed in an extruded aluminium case finished in beige. The brown front panel features two push switches, one for on/off-line and the other for manual switching of transmit and receive baud rates. Two LEDs indicate on-line and presence of carrier.

Terminal software supplied with the Datachat comes on disc although cassette users can obtain a ROM version that has slightly less features. The 40/80 column scrolling text mode provides most of the features that you could want for accessing BBs and Telecom Gold, but the viewdata mode provides little more than communication and saving of screens. There is no software download facility in viewdata mode which rules out that use with Micronet or Prestel Education.

Although the user-to-user facilities are good and worked well in practice, the error-corrected file transfer is not compatible with the widely used Xmodem standard and so makes it of limited use.

VERDICT

The Datachat 1223 is a well-engineered no-frills modem for those who need only V23 operation, but there is no way that I can recommend the supplied software for serious comms use. GEC can supply the Datachat for £10 less without software and you could then use it with Pace's Commstar for example.

THE MAGIC MODEM

The Magic Modem provides V21 and V23 operation, along with auto-dialling using the supplied software. A further option is 1200/1200 half duplex with the the line turnround achieved under software control.

The modem is housed in a beige plastic case with a black front panel that features a rotary selector switch for baud rate selection, and a bank of five LEDs for power, carrier detect, transmit data, receive data and transmission direction. The standard of construction of the Magic Modem is good but it does have a rather cheap look about it due to the gaudy yellow and black front panel design. The unit was supplied with an excellent manual that has plenty of information for beginner and expert alike.



The supplied software for the Magic Modem is on a 16K ROM called Commpanion, written by Soft Machinery. The manual is comprehensive and written in a manner that is easy to follow.

Using Commpanion and the Magic Modem is simplicity itself. Commands are entered from the Beeb's function keys and there is no need for a function key strip as a pull

down on-screen menu is available both on and off-line.

The viewdata terminal provides a host of facilities such as frame tagging, frame saving and loading, OS commands, call using the auto-dial, logoff, frame printing, telesoftware download and a frame editor for creating off-line mailbox frames.

The frame editor deserves extra mention as it really is superb. It has so many options, but thanks to a pull down menu, help is always at hand without having to search through the manual. There is also a pixel editor available which makes the production of complex mode 7 graphics much easier.

The scrolling terminal side of Commpanion is every bit as good as the viewdata side and has plenty of facilities such as call using the auto-dialler, OS commands, Xon/Xoff, spooling to disc, printing of data, sending of text files, and uploading/downloading. Again, these are all selected via the Beeb's function keys, with an on-screen pull-down menu for help. Any of the BBC screen modes can be used and if desired, a separate window can be defined for input whilst incoming data is displayed in the other. Great for MUD fans!

Error corrected file transfer is in the popular Xmodem format and files can be up/downloaded either to a memory buffer or direct to the current filing system.

VERDICT

This package was simply a delight to use. The Magic Modem performed perfectly during the review and I have no reason to believe that it would not continue to do so. The only criticism I can level at the modem is the short mains lead, and the lack of rubber feet on the case that causes it to skid around when using the selector switch!

The Commpanion ROM, in my opinion, is by far the best terminal software yet for the BBC and lack of space has permitted me to give you only a small glimpse of what it offers. Dedicated comms enthusiasts will find it has a wealth of facilities all of which can be 'tweaked' to suit their own needs via the ROM's 'Configure' command. Newcomers to comms and those who

want an easy-to-use terminal will be equally happy. I should point out to readers who already have a modem, that Soft Machinery market a ROM called Commsoft that is identical to Companion but without the Magic Modem auto dialling section.

LE MODEM

Watford Electronics must have produced more add-ons for the Beeb than any other manufacturer. Their latest is Le Modem, a V21 and V23 auto-dial, auto-answer modem.

It is housed in a black plastic case, with a front panel that sports just a bank of four LEDs. These are to indicate power, transmit, receive, and on line. Control of the modem is achieved by the supplied software via the Beeb's 1 Mhz BUS and connection is by a 1 metre 34-way ribbon cable that terminates inside the modem.

Le Modem has a fine standard of construction and looks smart. The accompanying manual is very good indeed and has some excellent additional information for those new to comms.

The supplied software is on a 16K ROM and a utility disc. The ROM software is divided into three main sections. Le Key is accessed by 'star' commands, and used for configuring the modem and selecting the various services. It allows the user to access Prestel for example by simply typing *CALL PRESTEL, the software taking care of selection of baud rates and selecting viewdata mode. There are other commands in the ROM that work at a lower level and these can be used from Basic to write your own programs to repeat dial an engaged number for example.

Le Prestel Terminal is the ROM's viewdata mode and features telesoftware downloading, an off-line mailbox editor, frame tagging, frame printing and saving/loading of screens.

L'Emulator is an 80 column scrolling text terminal that can be used for BBs and Telecom Gold. Much use is made of the function keys for selecting a variety of options such as Xon/Xoff, printing of data, changing screen and text colour, and spooling and sending of text files. Error corrected file transfer is provided for, using the usual Xmodem format.



Le Modem has the nice feature of monitoring the telephone line through the Beeb's own speaker. This is great for checking the progress of your call and is muted when you go on-line. I did find though that you can still hear the modem's tones at a low volume level when online, and I found this irritating.


VERDICT

In the main, I was favourably impressed with this package. The hardware is well designed and manufactured and worked perfectly during the course of this review. One point I am unhappy about though, is that whilst the modem is powered and connected, whenever Break is pressed a relay in the modem clatters on and off. It shouldn't cause problems and doesn't seize the telephone line, but is a bit disconcerting.

As for the software, it certainly has plenty of facilities and proved easy to get to grips with. I felt though that in certain areas it did lack the finesse of some terminal software I have used.

If you do consider buying this package, it is worth remembering that the scrolling terminal works only in 80 column and could be a problem for those with TVs or low resolution colour monitors.

CONCLUSIONS

Both the Magic Modem and Le Modem can be recommended, with perhaps the former having the edge with regard to its software. The Datachat is an excellent piece of hardware, unfortunately let down by its software, unless you are prepared to pay more for something like Commstar. 

Writing Your Own Compiler

(Part 3)

Davis Pilling concludes his series on writing a compiler, and gives some practical advice on its use.

The syntax analysis section of the compiler reduces a source code program to a series of procedure calls each of which represents an operation for a stack-based pseudo machine. Eventually, these procedures must produce real 6502 code. The accompanying listing, contains the code generating procedures for the compiler. Since all variables are assumed to be 5 byte floating point, it is convenient to use the numerical routines in the Basic ROM when possible.

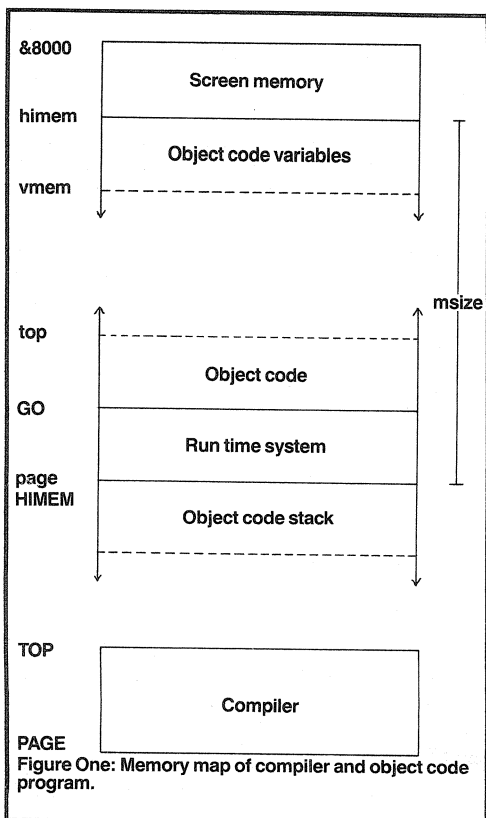


Figure One: Memory map of compiler and object code program.

RUN-TIME SYSTEM

A small piece of machine code, is needed as an interface between the ROM routines and the code generated by the compiler. This "run-time support system" also provides a few simple operating system routines in a more convenient form. PROCruntime (lines 4980 to 5930) assembles the run-time system.

Figure one is a memory map of the compiler and the object code program. This features a number of important variables situated at the beginning of the (first part of the) compiler listing. They are as follows:

`msize`: amount of memory reserved for the object code, run time system and variables.
`page`: lowest location used by object code
`top`: highest location used by object code
`vmem`: lowest location used by the variables of the compiled program.
`himem`: highest location that can be used by the object code.
`GO`: execution address of the object code.

The user has control over 'msize' and 'himem'. When the compiler is run, it sets its own HIMEM equal to `himem-msize`, reserving a chunk of memory for the object program. The variable 'page' is then set equal to HIMEM and a call of PROCruntime produces the run-time system at this address. The compiler then generates code above this. Every time a new variable is found in the program, memory is allocated to it from 'himem' down decreasing the value of 'vmem'.

The stack sets off from 'page' and works its way downwards towards the compiler. The top of stack element is kept in the Basic ROM's floating point accumulator 1 (FPA1). As numbers are popped and pushed, the contents of FPA1 are transferred to the stack proper. The stack pointer, is held by the zero page locations &4B and &4C. There are a few run-time routines which are important for managing the stack. For example 'push' will push FPA1 onto the stack and similarly the routine 'pull' transfers a number from the stack to FPA1. Finally, 'drop' removes a number from the stack.

CODE GENERATION

An example of a code generating procedure is PROCADD (line 4430). The

code produced by this first calls 'add' which is a ROM routine that adds the variable pointed to by (&A) (the stack pointer) to FPA1, and leaves the result in FPA1. A JSR of 'drop' removes the remaining operand from the stack. Thus the "add the top two stack elements" process is executed.

In the last part it was described how the SOUND statement is compiled by PROCsound which deals with it by calling PROCexp four times. This produces code that leaves the four parameters of the sound statement on the stack. Finally, PROCsound calls PROCsND which generates the statement JSR snd. 'snd' is a run-time routine which takes four elements off the stack and uses the OSWORD call on them to make a sound.

One problem which has not been mentioned is finding the addresses of things like GOTOs, GOSUBs and PROCs. Obviously, they compile into JMPs and JSRs, but when the compiler sees a statement like GOTO 2000 or PROCZONK, it has no way of knowing the object code address which corresponds to line 2000 or PROCZONK, unless it has already compiled line 2000 or seen a DEFPROCZONK statement. What happens is that the compiler generates code without these addresses. When compilation is finished, the code must be "fixed up" by having the addresses filled in. This is done by PROCFIX (line 4340).

Procedures are straightforward because when the DEFPROC statement for the procedure is met, the corresponding object code address can be put in the symbol table. For GOTOs and GOSUBs you may remember that every time one was found, the corresponding line number was inserted in string form in the symbol table. The first thing that PROCFIX does, is to scan the symbol table and look for GOTOs and GOSUBs. If it finds one, then PROCFIX scans the table of line numbers and object code addresses, and when the associated line number is found, copies its address into the symbol table. The final problem to be faced is finding the locations in the object code to which these addresses should be written.

To do this, the following idea is used. A JMP or JSR instruction takes three bytes, and if the address is not known the three bytes are used as follows: first

byte symbol table number of the object; second and third bytes pointer to the previous unknown address. The pointer to the first unknown address is held by the variable jm%. A linked list of unknown addresses is thus constructed. PROCJM (line 4260) inserts new items in the list. Finally, PROCFIX scans the list and fills in the addresses using the contents of the symbol table.

ERROR DETECTION AND REPORTING

The final aspect of compiler design, is error detection. Errors can be found in all stages of compilation. Conventional compilers try to continue after an error has been found. However, in a micro environment, a more acceptable (and much easier) approach is for the compiler to return the user to the program editor as soon as an error is found.

To keep things simple, only two types of errors are reported by our compiler. First syntax errors where the compiler has found something unexpected. These are handled by calling PROCERR in line 4960. Secondly, there are many cases where it is known that a certain token must be found next. For example, the expressions in a SOUND statement must be separated by commas. In this situation, PROCC (line 4920) is used. Looking through the syntax analysis section of the compiler, you can often see things like PROCC(comma). PROCC checks if the current token returned by the lexical analyser is equal to its argument, and if it is ends. Otherwise, there is an error in the program because the expected token is missing, so the procedure prints the message "missing" and then the token.

Run-time errors are handled automatically by the Basic ROM routines. There is, however, no attempt to check for errors in the fix-up process. Thus, if in a compiled program, you invoke a procedure without defining it, your program will crash.

USING THE COMPILER

Some thought must be given to using the compiler. To begin with, decide on the program you are going to compile and enter it in the compiler between lines 10 and 1000 in the way described in part one. There are a number of parameters that can be set in the compiler as described below, but you may prefer initially to use just the default values in the program listing.

These variables appear in the first few lines of part one of the compiler.

The first decision to be made is the value of 'himem'. This must be set less than or equal to the minimum value of HIMEM used by the compiled program. What this does is to ensure that the object code fits under screen memory. Thus if your program uses modes 4 & 5 himem=&5800; for mode 6 himem=&6000, and for mode 7 himem=&7C00. On a standard BBC, there is insufficient memory for programs that use modes 0 to 3. However, if you have a shadow RAM board or B+, then 'himem' can always have the mode 7 value and programs for all modes may be compiled. Unfortunately, only small programs can be compiled for use in modes 4 and 5, and disc users should move the compiler down in memory first (set PAGE=&1200 or use the move down routine listed on page 46 of this issue).

Next, you must decide how much memory to allocate for the object code and set 'msize' accordingly. Usually the object code will be 5 times as long as the source code and 600 bytes should be added to allow for the run time system.

Finally, it is necessary to select values for the variables 'ns' and 'nl' (lines 1110). These control the size of the symbol and line number tables. 'nl' must be greater than the number of lines in your program and 'ns' greater than the number of "symbols". Each variable, procedure name, and line GOTOed or GOSUBed counts as a "symbol". In addition, the fix-up technique limits 'ns' to less than 256. When a program is compiled, it is necessary to trade off the sizes of 'ns' and 'nl' against 'msize'. If you get 'ns' and 'nl' too small the compiler will end with a Basic error message. If 'msize' is too big then you will get the "No room" error. Finally, making 'msize' too small will give no error message but instead, the object code will overwrite first its variables and then screen memory. Having set up himem, 'msize', 'ns' and 'nl' in this way and readied your program for compilation, type RUN and stand back.

The first thing you should see, is the run-time system being assembled. The screen will then clear and split into two parts. In the top half you can watch the source code being scanned whilst in the lower half the corresponding assembly

language program can be seen. If any errors are found, the source code will break at the point that causes trouble.

If there are no problems, the compiler will print up some statistics including the lengths of the source and object code programs and the amount of memory used for storing variables and then ask you if you want to execute the code produced. If you answer 'Y' it will clear the screen and execute the code. When the code ends, the compiler finishes. You can run the machine code again by typing 'CALL GO' in immediate mode.

Because of the way the syntax of the language was defined, there is a limitation on the use of keywords like DEFPROC-ENDPROC. Briefly, DEFPROC-ENDPROC, REPEAT-UNTIL, WHILE-WEND and FOR-NEXT are treated as delimiters (like brackets), which means that you can only use them in matched pairs. This restriction does, however, prevent disastrous occurrences like returning from a non-existent procedure.

The object code can be saved and used separately from the compiler. To do this, instead of answering 'Y' to the "Execute code" prompt, press Escape and type:

```
PRINT~page,~top,~GO
```

then type:

```
*SAVE "filename"
```

followed by the three hex numbers. The resulting file can be *RUN. The object code can be used with Basic programs by resetting HIMEM to the value 'page', *LOADING the code file and CALLING 'GO'.

The code produced by the compiler is not optimal. Often, as you watch the assembly language going by, you will see things like JSR push, JSR pull. Obviously programs would be faster and smaller if inefficiencies like this were prevented. Such code optimisation is a large field: it can range from simply producing more efficient code for a program to actually modifying the source code.

Having seen how a simple Basic compiler is written there is nothing to stop you applying the same principles to the construction of a C or Pascal compiler, or inventing your own language.

In addition to the book mentioned in the last part, two other good books on compiler design are:

Understanding and Writing Compilers
by R. Bornat, Macmillan (1979)
and at a higher level:
Principles of Compiler Design by
A.V. Aho and J.D. Ullman,
Addison-Wesley (1977)

PROGRAM NOTE

The final part of the compiler, listed below should be merged with parts 1 and 2 published previously to form the complete program. See page 402 of the User Guide for more information on merging programs. The whole compiler is also included on the magazine cassette/disc for this month.

```

4220 REM Section #3
4230 REM Code generation and run time
4240 REM system.
4250 :
4260 DEFPROCJM(S%)
4270 ?P%=S%:P%?=jm MOD256
4280 P%?2=jm DIV256:jm=P%:P%=P%+3
4290 ENDPROC
4300 DEFPROCCLFIX
4310 LI%=VALst$(SN%):I%=-1:REPEATI%=I%+
1
4320 UNTIL LI%=lino(I%):sa(SN%)=lina(I%
)
4330 ENDPROC
4340 DEFPROCCLFIX
4350 IFjm=0 ENDPROC
4360 SN%=0:REPEAT
4370 IFstp(SN%)=lino ORstp(SN%)=gosub P
ROCLFIX
4380 SN%=SN%+1:UNTILst$(SN%)=""
4390 REPEAT P%=jm:jm=P%?1+256*P%?2
4400 IFstp(?P%)=proc PROCJSR(sa(?P%)) E
LSEIFstp(?P%)=gosub PROCJSR(sa(?P%)) ELS
E PROCJMP(sa(?P%))
4410 UNTIL jm=0
4420 ENDPROC
4430 DEFPROCADD:[JSRadd:JSRdrop:]ENDPRO
C
4440 DEFPROCSSUB:[JSRsub:JSRdrop:]ENDPRO
C
4450 DEFPROC MUL:[JSRmul:JSRdrop:]ENDPRO
C
4460 DEFPROC DIV:[JSRdiv:JSRdrop:]ENDPRO
C
4470 DEFPROC GTT:[JSRgtx:]ENDPROC
4480 DEFPROC LTT:[JSRltx:]ENDPROC
4490 DEFPROC GEQ:[JSRgex:]ENDPROC
4500 DEFPROC LEQ:[JSRlex:]ENDPROC
4510 DEFPROC EQU:[JSReqx:]ENDPROC
4520 DEFPROC NEQ:[JSRnex:]ENDPROC
4530 DEFPROC NEG:[JSRneg:]ENDPROC
4540 DEFPROC LDI:[JSRldi:]ENDPROC
4550 DEFPROC STI:[JSRsti:]ENDPROC
4560 DEFPROC VDU:[JSRvdx:]ENDPROC
4570 DEFPROC VDD:[JSRvdd:]ENDPROC
4580 DEFPROC INK:[JSRink:]ENDPROC
4590 DEFPROC RTS:[RTS:]ENDPROC
4600 DEFPROC END:[RTS:]ENDPROC
4610 DEFPROC PRI:[JSRpri:]ENDPROC
4620 DEFPROC VDL(L%):[LDA#L%:JSRswrch:]
ENDPROC
4630 DEFPROC SND:[JSRsnd:]ENDPROC
4640 DEFPROC JMP(A%):[JMPA%:]ENDPROC
4650 DEFPROC JSR(A%):[JSRA%:]ENDPROC
4660 DEFPROC JMC(A%):[JSRsig:PHP:JSRpull
:PLP:BNE P%+5:JMPA%:]ENDPROC
4670 DEFPROC LDA(A%):[JSRpush:JSRS4B:LDA
#A%MOD256:STA&4B:LDA#A%DIV256:STA&4C:JSR
ldf:JSRR4B:]ENDPROC
4680 DEFPROC STA(A%):[JSRS4B:LDA#A%MOD25
6:STA&4B:LDA#A%DIV256:STA&4C:JSRstf:JSRR
4B:JSRpull:]ENDPROC
4690 DEFPROC LCL(Q%):[JSRldl:]
4700 J%=3+?&4A2+256*?&4A3
4710 IP%=IJ%:P%?4=J%?4:P%=P%+5
4720 ENDPROC
4730 DEFPROC SQR:[JSRsqr:]ENDPROC
4740 DEFPROC SIN:[JSRsin:]ENDPROC
4750 DEFPROC COS:[JSRcos:]ENDPROC
4760 DEFPROC RND:[JSRrnz:]ENDPROC
4770 DEFPROC PI:[JSRpiz:]ENDPROC
4780 DEFPROC INP:[JSRinz:]ENDPROC
4790 DEFPROC INZ:[JSRinz:]ENDPROC
4800 DEFPROC NXT(A%,FA%)
4810 [JSR push:JSRS4B
4820 LDA#A%DIV256:STA&4C
4830 LDA#A%MOD256:STA&4B
4840 JSRnxx
4850 BCC P%+5:JMP FA%
4860 JSR drop:JSR pull
4870 ]ENDPROC
4880 DEFPROC CPS(NP%):[JSRpush:LDA#(NP%-1
)*5:JSRswpoi:]ENDPROC
4890 DEFPROC PP(A%):[LDX#A%MOD256:LDY#A%
DIV256:JSRswop:]ENDPROC
4900 DEFPROC PSX:[JSRpull:RTS:]ENDPROC
4910 DEFPROC PF(NP%,PS%):[JSRPS:LDA#(NP
%-1)*5:JSRpull:]ENDPROC
4920 DEFPROC(A%):IFT%=A% ENDPROC
4930 PROC S1:PRINT"" missing ";
4940 IFA%>32 ANDA%<lino THEN CALL TK E
LSEIFA%=id PRINT"id / variable";ELSEIFA%
=1 PRINT"expression"; ELSEIFA%=lino PRIN
T"line number";
4950 GOTO4970
4960 DEFPROC ERR:PROC S1:PRINT"" syntax
error!"
4970 PRINT":END
4980 DEFPROC runtime
4990 oswrch=&FFEE:osbyte=&FFF4
5000 osord=&FFF1
5010 FOR I%=0TO3 STEP 3
5020 P%=HIMEM
5030 [ OPT I%

```

```

5040 .push SEC:LDA&4B:SBC#5:STA&4B
5050 LDA&4C:SBC#0:STA&4C
5060 JSR stf:RTS
5070 .pull JSR ldf
5080 .drop CLC:LDA&4B:ADC#5:STA&4B
5090 LDA#0:ADC&4C:STA&4C:RTS
5100 .pulln CLC:ADC&4B:STA&4B:LDA#0
5110 ADC&4C:STA&4C:JSR pull:RTS
5120 .S4B LDA&4B:STA&80:LDA&4C:STA&81:R
TS
5130 .R4B LDA&80:STA&4B:LDA&81:STA&4C:R
TS
5140 .vdx JSRf2i:LDA&2A:JSRswrch:JSRp
u
ll:RTS
5150 .vdd JSRvdx:LDA&2B:JSRswrch:RTS
5160 .snd LDA#7:STA&78
5170 .sl JSR f2i:LDX#78:LDA&2B:STA &70,
X
5180 DEX:LDA&2A:STA &70,X:DEX:STX&78
5190 PHP:JSR pull:PLP
5200 BPL sl:LDX#&70:LDY#0:LDA#7
5210 JSRswrd:RTS
5220 .gtx JSRcpf:BCC ldfl:BEQ ldfl:BNE
ldtr
5230 .ltx JSRcpf:BCC ldtr:BCS ldfl
5240 .nex JSRcpf:BEQ ldfl:BNE ldtr
5250 .eqx JSRcpf:BEQ ldtr:BNE ldfl
5260 .gex JSRcpf:BCS ldtr:BCC ldfl
5270 .lex JSRcpf:BCC ldtr:BEQ ldtr:BCS
ldfl
5280 .ldfl LDA#0:JSRA2f:JSRdrop:RTS
5290 .ldtr LDA#255:JSRA2f:JSRdrop:RTS
5300 .pri JSRS4B:LDA#0:STA&15:LDY#&FF:J
SRf2s:LDX#0
5310 .pl LDA&600,X:JSRswrch:INX:CPX&36
:BNE pl:JSRR4B:JSRpull:RTS
5320 .ldl JSRpush
5330 JSRS4B:PLA:CLC:ADC#1:STA&4B
5340 PLA:ADC#0:STA&4C:JSRldf
5350 LDA&4B:CLC:ADC&4:TAx
5360 LDA&4C:ADC#0:PHA:TXA:PHA
5370 JSRR4B:RTS
5380 .prx PLA:STA&70:PLA:STA&71:LDY#0
5390 .prl INC &70:BNE pr2:INC&71
5400 .pr2 LDA(&70),Y:CMP#13:BEQ prx
5410 JSRswrch:JMP prl
5420 .prx LDA&71:PHA:LDA&70:PHA:RTS
5430 .sqz JSRS4B:JSRsqx:JSRR4B:RTS
5440 .siz JSRS4B:JSRsix:JSRR4B:RTS
5450 .coz JSRS4B:JSRcox:JSRR4B:RTS

```

```

5460 .piz JSRpush:JSRS4B:JSRpix:JSRR4B:
RTS
5470 .rnz JSRS4B:JSRf2i:JSRrxn:JSRi2f:J
SRR4B:RTS
5480 .ldi JSRf2i:LDY#0:LDA(&2A),Y
5490 JSR A2f:RTS
5500 .sti JSRf2i:LDA&2A:PHA
5510 JSR pull:JSRf2i
5520 LDY#0:PLA:STA(&2A),Y:JSR pull:RTS
5530 .ink JSRf2i:LDX&2A:LDY&2B
5540 LDA#&81:JSR osbyte
5550 TYA:CMP#&FF:BEQ IT
5560 CMP#&1B:BEQ IE
5570 .IR TXA:JSR A2f:RTS
5580 .IE LDA#&7E:JSR osbyte:LDY#&FF
5590 .IT TYA:JSR A2f:RTS
5600 .nxx JSRadd:JSRstf
5610 JSRR4B:JSRdrop
5620 JSRcpf:PHP:JSRR4B:JSRpull:JSRs
ig
5630 BMI FM
5640 PLP:BCC FL:BCS FL
5650 .FM PLP:BEQ F2:BCC F2:CLC:BCC FL
5660 .F2 SEG:FL:RTS
5670 .izz LDA#0:STA&78:STA&79:RTS
5680 .inz JSRpush:LDA&78:BNE i2
5690 LDA&79:BEQ i9:LDA#63:JSRswrch
5700 .i9 LDA#1:STA&79:JSR ins:STY&78
5710 .i2 LDX#0
5720 .i3 LDA&600,X:CMP#44:BEQ fc
5730 CPX&78:BEQ fc:INX:BNE i3
5740 .fc STX&36:JSR asc2
5750 BMI i4:JSR i2f
5760 .i4 LDX#0:LDY&36:INX
5770 .i6 LDA&600,Y:STA&600,X
5780 CPY&78:BEQ i5
5790 INY:INX:JMP i6
5800 .i5 LDA&78:SEC:SBC&36:STA&78
5810 RTS
5820 .swpoi CLC:ADC&4B:STA&7A:LDA#0
5830 ADC&4C:STA&7B:RTS
5840 .swop STX&7C:STY&7D:LDY#0:LDX#5
5850 .swp1 LDA(&7A),Y:PHA:LDA(&7C),Y
5860 STA(&7A),Y:PLA:STA(&7C),Y
5870 INY:DEX:BNE swp1
5880 LDA&7A:SEC:SBC#5:STA&7A
5890 LDA&7B:SBC#0:STA&7B:RTS
5900 .GO LDA#HIMEM MOD256:STA&4B:STA &4
5910 LDA#HIMEM DIV256:STA&4C:STA &5
5920 ]:NEXT
5930 ENDPROC

```



← 35

You can use these routines either directly in a machine code program, or from Basic (e.g. 'CALL readtime'). In Basic, though, you do need to allow for the CALLs' overhead. If you try:

```
CALL timzero:CALL readtime
```

```
PRINT !timebuff
```

you will find that it takes around 950 ms

to make the 2 CALLs. If you have the magazine disc or cassette, there is a short demo program to show this effect.

Next month, we'll build on this article to see how to use Timer 1 and interrupts to set up an extra Basic clock which you can read to 1 ms.



1st course

Using the EVAL Function (Part 2)

Mike Williams concludes his look at the EVAL function by examining some of its more unexpected applications.

In last month's article on using EVAL I gave an example of a program to select numbers (from a list) to be displayed on the screen. The program would only cope with simple tests, such as

">75", and I left you with the problem of how to extend the program to cope with more complex tests such as ">25 AND <75". The way I have chosen is as follows.

We will allow both AND and OR in our tests and insist that they are separated by spaces from the rest (as above). In principle, the program will need to scan through the test string, looking for the next space and extracting the characters up to that space. Thus, in the example string, the program would first extract ">25 ", then "AND " and lastly "<75 ", though to make this work properly we would have to add a space at the end of the test string to start with. This ensures that each identifiable part of the string can be detected. Lines 160 to 200 of last month's program should be replaced by those listed below.

```
160 REPEAT
170 CLS:test$=""
180 PRINT"Select which numbers"
190 INPUT t$:t$=t$+" "
191 REPEAT
192 I=INSTR(t$," ")
193 t1$=LEFT$(t$,I):t$=MID$(t$,I+1)
194 IF INSTR("AND OR ",t1$) THEN test$=test$+t1$ ELSE test$=test$+"number(I)+"
t1$
195 UNTIL t$=""
200 PRINT"Test string:"test$'
```

If either AND or OR are detected they are simply appended to the expanded test string in test\$, otherwise "number(I)" is appended followed by the current part of the test string. Thus the test:

>25 AND <75
becomes the expanded test:

number(I)>25 AND number(I)<75
and it is this (in string form) which is evaluated by the EVAL function. You can try other tests, including several AND and OR operators and see what happens. The new version of the program has been altered to display the expanded test string for you.

Now let us take a look at some of the less obvious uses of the EVAL function. Again, the examples may appear a little artificial in isolation.

Sometimes when writing a program using, say, several different arrays, a need arises for the program to select one of these arrays in order to perform some function upon it. Often, it feels as if the solution would be to store the name of the required array in another variable. We could then program the micro to carry out the required operation on the array whose name is stored in this variable. Unfortunately, neither BBC Basic nor most other versions of Basic allow this, but EVAL gives us the next best thing.

Consider the following example. The weights of 100 boys and of 100 girls are stored in the two arrays called BOYS() and GIRLS(). We are going to write a short program to allow a user to select either the boys or the girls and calculate their average weight. The program to do this is listed below, but you might first like to try and think how you might program this.

In lines 110 to 150 two sets of random weights are created in the two arrays, BOYS() and GIRLS(). The program then asks the user to select either "BOYS" or "GIRLS". This is assigned to the variable sex\$ which is used by the EVAL function at line 220 to select the correct array.

Without using EVAL, you would have to have one set of instructions to find the average weight of the boys, and another very similar set to find the average weight of the girls. You would also have


```

10 REM EVAL3
100 MODE 7
110 DIM BOYS(100),GIRLS(100)
120 FOR I=1 TO 100
130 BOYS(I)=70+RND(20)
140 GIRLS(I)=70+RND(30)
150 NEXT I
160 :
170 REPEAT
180 CLS
190 INPUT"Choose BOYS or GIRLS:" sex$
200 total=0
210 FOR I=1 TO 100
220 total=total+EVAL(sex$+"(I)")
230 NEXT I
240 average=total/100
250 PRINT"The average weight of 100 "
+sex$"is ";average;" pounds."
260 PRINT"Press any key to continue";
:G=GET
270 UNTIL FALSE
280 END

```

to use an IF-THEN statement to test the input and then jump to the correct routine. In fact, EVAL makes the whole program much shorter.

Sometimes it is not an array or variable that has to be selected but an operation (+ - * /). If the operation is first assigned to a variable, then EVAL keeps the programming short. For example:

```

math$="number1"+op$+"number2"
ans=EVAL(math$)

```

will add, subtract, multiply or divide the two numbers stored as number1 and number2 depending on which of the four operators is assigned to the variable op\$.

To conclude this look at the EVAL function, I'd like to show you another very useful example, part of which was used in the BEEBUG Filer program (Vol.4 Nos.6 to 8). Sorting a list of numbers is something that most programmers try their hand at sooner or later. Here's a way, using EVAL of course, to write a reasonably short program to sort an array either numerically or alphabetically, and in ascending or descending order. The sorting technique itself is a simple Bubble sort described in various Workshops (BEEBUG Vol.4 No.7). The only assumption is that all the data is held in string form and converted to numeric format when required. This is the program EVAL4.

```

10 REM EVAL4
100 MODE 7
110 DIM data$(100)
120 FOR I=1 TO 100
130 data$(I)=STR$(RND(999))
140 NEXT I
150 :
160 REPEAT:CLS
170 PRINT"String (S) search or a "
180 INPUT"numeric (N) search? " t1$
190 PRINT"Ascending (A) or "
200 INPUT"descending (D) order? " t2$
210 IF t1$="S" THEN t1$="" ELSE t1$="V
AL"
220 IF t2$="A" THEN t2$("<" ELSE t2$="
">"
230 test$=t1$+(data$(J))"+t2$+t1$+(d
ata$(J-1))"
240 PRINT"Please wait"
250 :
260 FOR I=1 TO 99
270 FOR J=100 TO I+1 STEP -1
280 IF EVAL(test$) THEN PROCswap
290 NEXT J,I
300 :
310 FOR I=1 TO 100:PRINT data$(I):NEXT
320 PRINT"Press any key to continue";:
G=GET
330 UNTIL FALSE
340 END
350 :
1000 DEF PROCswap
1010 temp$=data$(J)
1020 data$(J)=data$(J-1)
1030 data$(J-1)=temp$
1040 ENDPROC

```

The answers to the questions in lines 170 to 240 determine whether the VAL function is used to convert from string to numeric format, and whether > or < is used when comparing two elements together. The two variables t1\$ and t2\$ are set either to "" or "VAL" in the first case, and "<" or ">" in the second. The values of t1\$ and t2\$ are then used to create a complete test string at line 230.

The actual sorting takes place in lines 260 to 290 with EVAL evaluating the test string already created. To illustrate the whole process, one hundred random numeric values are assigned to a string array at the start of the program, and the results of the sort displayed at the end. The advantage of using EVAL is that the one sorting routine can be used to sort both numeric and string data, and in ascending or descending order.

BURGER TIME

Jonathan Temple, the author of several recent games, has come up with another mouth-watering challenge. Can you make all the beefburgers, while avoiding some rather nasty sausages, and become chef of the year?

Burger time features that famous chef Mr. Gordon Blue, whose kitchens have been over-run by deranged sausages bent on world domination! Whilst they chase him the chef must assemble four burgers in each kitchen...can you help him to complete his task?

Four burgers, each split into four parts - meat, lettuce and two halves of bun - are suspended at various points in the kitchen. You must knock them down into the cups below to form complete burgers by walking over them (rather unhygienic!)

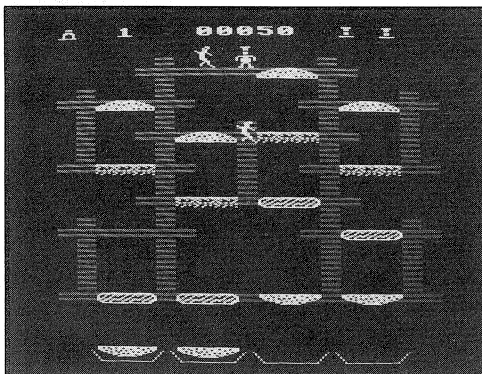
Meanwhile you will be chased by two sausages which can be paralysed for a while by throwing pepper over them. You can only carry one pepper pot at a time, and this is indicated at the top left-hand of the screen. Additional pepper pots appear at the top of the playing area.

After the second screen a strawberry-flavoured milkshake appears, which can be collected for bonus points.

The keys to use are 'Z' and 'X' for left and right, and '*' and '?' for up and down. 'Return' is used to throw the pepper - it will 'fire' in the last horizontal direction you moved.

In addition you can pause while playing by pressing 'P', and continue with 'C'. When paused you can use 'Q' and 'S' to turn the sound on and off, and 'F' to finish the game.

Since the program is fairly long and uses mode 2 graphics, users with disc systems (or any user with PAGE set higher than &E00) will need to include these few lines at the beginning of the program:



```
1 IF PA.<&E01 THEN 10
2 *K,0 *T.|MF.A%=0TO (TOP-PA.)S.4:A%!&E00
  =A%!PA.:N.|MPA.=&E00|MO.|MDEL.1,4|MRUN
  |M
3 *FX138,0,128
4 END
```

Please remember to save the program before using this routine, as pressing Break after relocating the program may cause it to be corrupted.

```
10 REM PROGRAM BURGER TIME
20 REM VERSION C1.2
30 REM AUTHOR J.Temple
40 REM BEEBUG MARCH 1986
50 REM PROGRAM SUBJECT TO COPYRIGHT
60 :
100 MODE 7:ON ERROR GOTO 3930
110 PROCinit:PROCchars
120 PROCenvs:PROCTitle
130 REPEAT:MODE 2
140 PROCsetvars:REPEAT
150 REPEAT:PROCscreen
160 REPEAT:PROCitems
170 N%=N% EOR 1
180 PROCsausages:PROCchef
190 UNTIL E%
200 IFE%=1 PROCKilled
210 UNTIL Z%=0 OR E%=2
220 IFE%=2 PROCnext
230 UNTIL Z%=0:PROCend
240 MODE 7:PROChall
250 UNTIL FALSE
260 :
270 DEFPROCchef
1010 A%=X%:B%=Y%
1020 IF INKEY=56 PROCpause
1030 IF INKEY=73 IFPOINT(X%,Y%-16)=8 IF
X%MOD64=0 Y%=Y%+16:SOUND18,-10,30,1:GOTO
1080
```

```

1040 IF INKEY-105 IFPOINT(X%,Y%-80)=8 I
FX%MOD64=0 Y%=Y%-16:SOUND18,-10,30,1:GOT
O1080
1050 IF INKEY-98 IFPOINT(X%-32,Y%-68)>0
Y%=(Y%DIV32)*32+28:X%=X%-32:IX%=-32:SOU
ND18,-10,15,1
1060 IF INKEY-67 IFPOINT(X%+64,Y%-68)>0
Y%=(Y%DIV32)*32+28:X%=X%+32:IX%=32:SOUN
D18,-10,15,1
1070 IF INKEY-74 IFPH% PROCthrow
1080 IF X%<>A% OR Y%<>B% MOVE A%,B%:PRI
NT M$:MOVE X%,Y%:PRINT M$
1090 L%=POINT(X%,Y%-64):M%=POINT(X%+56,
Y%-64)
1100 IFL%<>8 IFM%<>8 IFL%>0 IFM%>0 PROC
drop
1110 IFX%=576 IFY%=956 PROCcheck
1120 ENDPROC
1130 :
1140 DEFPROCpause
1150 REPEAT L%=GET AND &DF
1160 IF L%=81 THEN *FX210,1
1170 IF L%=83 THEN *FX210,0
1180 UNTIL L%=67 OR L%=70
1190 IF L%=70 E%=1:Z%=1
1200 ENDPROC
1210 :
1220 DEFPROCthrow
1230 SOUND16,-12,6,2
1240 Q%=X%:R%=Y%-16:PX%=IX%:PH%=0
1250 PROCscore(0):TP%=1
1260 MOVE Q%,R%:VDU18,3,3,241
1270 ENDPROC
1280 :
1290 DEFPROCcheck
1300 IFV%+W%=0 ENDPROC
1310 IFV% IFPH%=0 V%=0:PH%=1:MOVE576,92
4:VDU18,3,7,229:PROCscore(50):SOUND19,3,
75,6
1320 IFW% W%=0:MOVE576,956:PRINTG$:PROC
score(250):SOUND19,3,75,6
1330 ENDPROC
1340 :
1350 DEFPROCsausages
1360 IFP%(N%) P%(N%)=(P%(N%)+1)MOD PT%:
IFP%(N%) ENDPROC
1370 J%=J%(N%):K%=K%(N%)
1380 L%=L%(N%):M%=M%(N%)
1390 IFM% PROCcladder ELSE PROCwall
1400 J%(N%)=J%(N%)+L%(N%)
1410 K%(N%)=K%(N%)+M%(N%)
1420 MOVE J%,K%:PRINT S$
1430 IFABS(J%-X%)<64 IFABS(K%-Y%)<64 E%
=1
1440 MOVE J%(N%),K%(N%):PRINT S$
1450 IFABS(J%(N%)-X%)<64 IFABS(K%(N%)-Y
%)<64 E%=1
1460 ENDPROC
1470 :
1480 DEFPROCcladder

```

```

1490 IFM%=32 IFPOINT(J%,K%)=0 PROCselec
t:ENDPROC
1500 IFM%=-32 IFPOINT(J%,K%-96)=0 PROCs
elect:ENDPROC
1510 IFPOINT(J%-8,K%-64)=8 IFRND(N%*2+2
)<2 OR K%=Y% PROCselect
1520 ENDPROC
1530 :
1540 DEFPROCselect
1550 M%(N%)=0
1560 IFJ%=0 OR J%<X% L%(N%)=64 ELSE L%(
N%)=-64
1570 ENDPROC
1580 :
1590 DEFPROCwall
1600 IFPOINT(J%,K%)=8 IFY%>K% IFY%-K%>2
24 OR RND(3)=1 OR X%=J% L%(N%)=0:M%(N%)
=32:ENDPROC
1610 IFPOINT(J%,K%-96)=8 IFY%<K% IFK%-Y
%>224 OR RND(3)=1 OR X%=J% L%(N%)=0:M%(N
%)=-32:ENDPROC
1620 IFPOINT(J%-32,K%-68)<1 OR POINT(J%
+64,K%-68)<1 L%(N%)=-L%(N%)
1630 ENDPROC
1640 :
1650 DEFPROCdrop
1660 A%=(X%-128)DIV256*256+128
1670 T%=((A% DIV128)-1)/2
1680 F%=92:IFY%>284 IFT%=1 ORT%=2 F%=188
1690 B%=Y%-64:U%=4-(Y%-F%)/192
1700 C%=B%(T%,U%):B%(T%,U%)=0
1710 MOVE A%,B%:PRINT B$(C%)
1720 MOVE A%,B%-32:PRINT B$(C%)
1730 L%=B%-160:IF T%=1 OR T%=2 IFU%=2 L
%=B%-256
1740 IF U%=3 PROCadd
1750 SOUND 17,2,150,12
1760 FOR M%=B%-32 TO L% STEP -32
1770 MOVE A%,M%:PRINT B$(C%)
1780 MOVE A%,M%-32:PRINT B$(C%)
1790 NEXT
1800 IF U%<3 IF B%(T%,U%+1)>0 U%=U%+1:D
%=B%(T%,U%):B%(T%,U%)=C%:C%=D%:B%=M%:GOT
O1710
1810 IF U%<3 B%(T%,U%+1)=C%
1820 ENDPROC
1830 :
1840 DEFPROCadd
1850 L%=B%-(4-(B%(T%,4)))*32
1860 PROCscore(25)
1870 B%(T%,4)=B%(T%,4)+1:F%=0
1880 FOR H%=0 TO 3:F%=F%+B%(H%,4)
1890 NEXT:IF F%=16 IF E%=0 E%=2
1900 ENDPROC
1910 :
1920 DEFPROCitems
1930 IF TP% PROCthrowing
1940 IFV%+W%>0 PROCactive:ENDPROC
1950 IFPH%=0 IFRND(500-PT%*3)=1 MOVE 57
6,924:VDU18,3,7,229:V%=1:SOUND19,4,125,20

```

```

1960 IFP%>2 IFRND(100)=1 MOVE 576,956:P
RINT G$:W%=1:SOUND19,4,125,20
1970 ENDPROC
1980 :
1990 DEFPROCactive
2000 IFV% V%=V%+1:IFV%=80 MOVE576,924:V
DU18,3,7,229:V%=FALSE:SOUND16,-12,6,2
2010 IFW% W%=W%+1:IFW%=80 MOVE576,956:P
RINT G$:W%=FALSE:SOUND16,-12,6,2
2020 ENDPROC
2030 :
2040 DEFPROCthrowing
2050 MOVE Q%,R%:VDU18,3,3,241
2060 Q%=Q%+PX%
2070 TP%=TP%+1:IFTP%=10 OR Q%<0 OR Q%>1
216 TP%=0:ENDPROC
2080 MOVE Q%,R%:VDU 241
2090 IF ABS(J%(0)-Q%)<64 IFABS(K%(0)-R%
)<64 P%(0)=1:SOUND19,3,100,3:PROCscore(1
00)
2100 IF ABS(J%(1)-Q%)<64 IFABS(K%(1)-R%
)<64 P%(1)=1:SOUND19,3,100,3:PROCscore(1
00)
2110 ENDPROC
2120 :
2130 DEFPROCscore(N%)
2140 S%=S%+N%:VDU4,17,7,31,0,0
2150 PRINT TAB(3);P%;TAB(7,0);LEFT$( "00
00",5-LEN(STR$(S%)))+STR$(S%);" " ;STRI
NG$(Z%-1,CHR$237+" ")
2160 VDU31,0,0,32+197*PH%,5
2170 ENDPROC
2180 :
2190 DEFPROCKilled
2200 Z%=Z%-1:PH%=1:SOUND 0,1,100,2
2210 TIME=0:REPEAT UNTIL TIME>100
2220 ENDPROC
2230 :
2240 DEFPROCend
2250 *FX 15,0
2260 VDU4,28,4,13,15,11,12,26,5
2270 PROCprint("GAME OVER",352,636,1,3)
2280 TIME=0:REPEAT UNTIL TIME>200
2290 ENDPROC
2300 :
2310 DEFPROCchall
2320 N%=-1:FOR L%=7 TO 0 STEP -1
2330 IF S%>S%(L%) N%=L%
2340 NEXT
2350 IF N%>-1 PROCcongrats
2360 PROCdisplay("Hall of Fame")
2370 PRINTTAB(7,23);CHR$131"Press SPACE
BAR to play";
2380 REPEAT UNTIL GET=32
2390 ENDPROC
2400 :
2410 DEFPROCnext
2420 P%=P%+1:IF P%=2 Z%=Z%+1
2430 FOR N%=1 TO (300+P%*100) DIV25
2440 PROCscore(25):SOUND0,-10,5,1
2450 FOR D=1 TO 50:NEXT,
2460 PT%=PT%-30:IFPT%<30 PT%=30
2470 RESTORE 2530:N%=81:*FX15,0
2480 FOR L%=1 TO 10:READ A%,D%:N%=N%+A%
2490 SOUND1,-10,N%,D%
2500 SOUND2,-5,N%+48,D%:NEXT
2510 ENDPROC
2520 :
2530 DATA 0,4,8,4,8,4,4,4,8,8,-12,8,4,8
,-12,8,8,8,-16,8
2540 :
2550 DEFPROCcongrats
2560 LOCAL A%,L%,X%,Y%
1290 FOR L%=6 TO N% STEP -1
2580 S%(L%+1)=S%(L%):H$(L%+1)=H$(L%)
2590 NEXT
2600 S%(N%)=S%:H$(N%)=""
2610 PROCdisplay("Congratulations!")
2620 X%=&70:VDU 31,21,6+N%*2
2630 !&70=&200C0A00:??&74=127:CALL&FFF1
2640 H$(N%)=LEFT$(S&A00,12)
2650 ENDPROC
2660 :
2670 DEFPROCdisplay(M$)
2680 VDU12,31,4,6,136
2690 PROClarge(14,0,131,"SCORES")
2700 FOR L%=3 TO 4
2710 PRINTTAB ((40-LEN(M$))/2-3,L%);CHR$
141;CHR$134;M$
2720 NEXT
2730 FOR L%=0 TO 7
2740 PRINTTAB(5,6+L%*2);L%+1;" ... ";S%
(L%);TAB(17,6+L%*2);"... ";H$(L%)
2750 NEXT
2760 ENDPROC
2770 :
2780 DEFPROCprint(T$,X,Y,A,B)
2790 GCOL 0,A:MOVE X,Y:PRINT T$
2800 GCOL 0,B:MOVE X-8,Y-4:PRINT T$
2810 ENDPROC
2820 :
2830 DEFPROCsetvars
2840 Z%=3:S%=0:P%=1:PH%=1:PT%=120
2850 VDU 19,8,4,0;
2860 FOR L%=9 TO 15
2870 VDU 19,L%,L%-8,0;:NEXT
2880 ENDPROC
2890 :
2900 DEFPROCscreen
2910 VDU 4,12,23;10,32;0;0;0;
2920 PROCscore(0):VDU4,17,8
2930 W$=STRING$(7,CHR$234)
2940 FOR Y%=7 TO 25 STEP 6
2950 PRINTTAB(0,Y%);W$;TAB(12,Y%);W$
2960 IFY%=25 N%=Y% ELSE N%=Y%-3
2970 PRINTTAB(4,N%);STRING$(11,CHR$234)
2980 NEXT
2990 FOR Y%=6 TO 13
3000 VDU31,1,Y%,234,31,17,Y%,234:NEXT
3010 FOR Y%=18 TO 25

```

```

3020 VDU31,1,Y%,234,31,17,Y%,234:NEXT
3030 FOR Y%=9 TO 16
3040 VDU31,9,Y%,234:NEXT
3050 FOR Y%=3 TO 25
3060 VDU31,5,Y%,234,31,13,Y%,234:NEXT
3070 VDU 5:GCOL 0,6
3080 FOR X%=104 TO 872 STEP 256
3090 MOVE X%,48:DRAW X%+32,16
3100 DRAW X%+200,16:DRAW X%+232,48
3110 NEXT
3120 FOR Y%=220 TO 796 STEP 192
3130 U%=5-(Y%-28)/192
3140 FOR N%=0 TO 3:B%(N%,4)=0
3150 B%(N%,U%-1)=U%:NEXT
3160 N%=Y%:IF U%<4 N%=Y%+96
3170 MOVE 128,Y%:PRINT B$(U%)
3180 MOVE 384,N%:PRINT B$(U%)
3190 MOVE 640,N%:PRINT B$(U%)
3200 MOVE 896,Y%:PRINT B$(U%)
3210 NEXT
3220 E%=0:G%=0:V%=0:W%=0:TP%=0:IX%=0
3230 X%=576:Y%=284:MOVE X%,Y%:PRINT M$
3240 RESTORE 3320
3250 FOR N%=0 TO 1
3260 READ J%(N%),K%(N%),L%(N%)
3270 M%(N%)=0:P%(N%)=0
3280 MOVE J%(N%),K%(N%):PRINT S$
3290 NEXT:N%=0
3300 ENDPROC
3310 :
3320 DATA 320,572,64,832,572,-64
3330 :
3340 DEFPROCinit
3350 DIM B%(3,4),B$(4),J%(1),K%(1),L%(1)
),M%(1),P%(1),S%(7),H$(7)
3360 S%=0:C$=CHR$18+CHR$3
3370 B$(0)=C$+CHR$8+STRING$(3,CHR$234)
3380 B$(1)=C$+CHR$3+CHR$227+CHR$225+CHR
$228
3390 B$(2)=C$+CHR$2+STRING$(3,CHR$230)
3400 B$(3)=C$+CHR$1+CHR$232+CHR$231+CHR
$233
3410 B$(4)=C$+CHR$3+CHR$224+CHR$225+CHR
$226
3420 M$=C$+CHR$7+CHR$237+CHR$10+CHR$8+C
HR$238
3430 S$=C$+CHR$1+CHR$235+CHR$10+CHR$8+C
HR$236
3440 G$=C$+CHR$5+CHR$239+CHR$10+CHR$8+C
HR$240
3450 FOR L%=0 TO 7
3460 S$(L%)=2500-L%*250
3470 H$(L%)="GORDON BLUE":NEXT
3480 ENDPROC
3490 :
3500 DEFPROCchars
3510 VDU23,224,-1,214,-5,110,59,15,3,0
3520 VDU23,225,-1,223,-5,175,125,215,12
6,-1

```

```

3530 VDU23,226,-1,119,223,-6,108,240,19
2,0
3540 VDU23,227,0,3,14,59,119,221,-9,-1
3550 VDU23,228,0,192,112,220,186,239,18
1,-1
3560 VDU23,229,28,0,28,54,34,34,34,62
3570 VDU23,230,-1,146,73,130,84,1,148,3
3
3580 VDU23,231,-1,73,146,36,73,146,36,-
1
3590 VDU23,232,63,73,146,164,201,146,10
0,63
3600 VDU23,233,-4,74,147,37,73,147,38,2
52
3610 VDU23,234,-1,-1,0,0,-1,-1,0,0
3620 VDU23,235,12,20,24,152,120,52,50,4
8
3630 VDU23,236,48,56,24,12,20,36,34,1
3640 VDU23,237,60,24,24,0,24,24,0,126
3650 VDU23,238,153,189,189,60,24,36,36,
102
3660 VDU23,239,1,2,4,-1,137,145,-1,-1
3670 VDU23,240,191,223,94,110,60,24,44,
94
3680 VDU23,241,0,146,73,2,84,161,20,66
3690 ENDPROC
3700 :
3710 DEFPROCenvs
3720 ENVELOPE 1,1,0,0,0,0,0,0,90,-1,-2,
-3,97,97
3730 ENVELOPE 2,3,-1,-1,-1,70,50,1,100,
-1,-1,-10,101,5
3740 ENVELOPE 3,133,8,4,8,3,1,1,126,0,0
,-10,126,0
3750 ENVELOPE 4,1,0,0,0,0,0,0,126,-1,0,
-5,126,0
3760 ENDPROC
3770 :
3780 DEFPROCtitle
3790 VDU 23;10,32;0;0;0;0;
3800 PROClarge(11,4,130,"B U R G E R")
3810 PROClarge(13,8,131,"T I M E")
3820 PRINT'TAB(8);CHR$134;"by Jonathan
Temple"
3830 TIME=0:REPEAT UNTIL TIME>500
3840 ENDPROC
3850 :
3860 DEFPROClarge(V%,W%,C%,M$)
3870 LOCAL A%
3880 FOR A%=W% TO W%+1
3890 PRINTTAB(V%,A%)CHR$C%;CHR$141;M$
3900 NEXT
3910 ENDPROC
3920 :
3930 MODE 7:REPORT:PRINT" at line ";ERL
3940 *FX 15,1
3950 END

```



BEEBUG MAGAZINE is produced
by BEEBUG Publications Ltd.

Editor: Mike Williams
Assistant Editor: Geoff Bains
Production Assistant:
Yolanda Turuelo
Technical Assistant: Alan Webster
Secretary: Debbie Sinfield
Managing Editor: Lee Calcraft
Additional thanks are due to
Sheridan Williams, Adrian Calcraft,
John Yale and Tim Powys-Lybbe.

All rights reserved. No part of this
publication may be reproduced
without prior written permission of
the Publisher. The Publisher cannot
accept any responsibility, whatso-
ever for errors in articles, programs,
or advertisements published. The
opinions expressed on the pages of
this journal are those of the authors
and do not necessarily represent
those of the Publisher, BEEBUG
Publications Limited.

BEEBUG Publications Ltd (c) 1986

Editorial Address

BEEBUG
PO BOX 50,
Holywell Hill,
St. Albans AL1 3YS

CONTRIBUTING TO BEEBUG PROGRAMS AND ARTICLES

We are always seeking good quality
articles and programs for publica-
tion in BEEBUG. All contributions
used are paid for at up to £40 per
page, but please give us warning of
anything substantial that you
intend to write. A leaflet, 'Notes
of Guidance for Contributors' is
available on receipt of an A5 (or
larger) SAE.

In the case of material longer than
a page, we would prefer this to be
submitted on cassette or disc in
machine readable form using
"Wordwise", "View", or other
means, but please ensure an
adequate written description of
your contribution is also included.
If you use cassette, please include a
backup copy at 300 baud.

HINTS

There are prizes of £5 and £10 for
the best hints each month, plus one
of £15 for a hint or tip deemed to
be exceptionally good.

Please send all editorial material to
the editorial address above. If you
require a reply it is essential to
quote your membership number
and enclose an SAE.

SUBSCRIPTIONS

Send all applications for membership, subscription renewals, subscription
queries and orders for back issues to the subscriptions address.

MEMBERSHIP SUBSCRIPTION RATES

£ 6.90 6 months (5 issues) UK ONLY)

£12.90 UK - 1 year (10 issues)

£19.00 Europe,

£22.00 Middle East

£24.00 Americas & Africa,

£26.00 Elsewhere

BACK ISSUES
(Members only)

Vol	Single issues	Volume sets (10 issues)
1	90p	£8
2	£1	£9
3	£1.20	£11
4	£1.20	—

Please add the cost of post and packing as shown:

DESTINATION	First issue	Each subsequent issue
UK	30p	10p
Europe	70p	20p
Elsewhere	£1.50	50p

All overseas items are sent airmail (please send a sterling cheque). We will
accept official UK orders but please note that there will be a £1 handling
charge for orders under £10 that require an invoice. Note that there is no
VAT on magazines.

Back issues are for members only, so it is ESSENTIAL to quote your
membership number with your order. Please note that the BEEBUG
Reference Card and BEEBUG supplements are not supplied with back
issues.

Subscriptions, Back Issues &
Software Address

BEEBUG
PO BOX 109
St. Johns Road
High Wycombe HP10 8NP

Hotline for queries and software orders

St. Albans (0727) 40303
Manned Mon-Fri 9am-4.30pm

24hr Answerphone Service for Access and
Barclaycard orders, and subscriptions
Penn (049481) 6666

If you require members' discount on software it is essential to quote
your membership number and claim the discount when ordering.

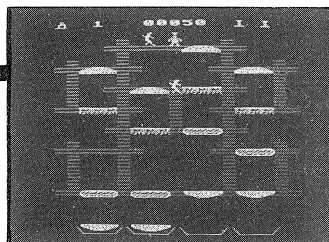
Magazine Cassette/Disc

MARCH 1986 CASSETTE DISC CONTENTS

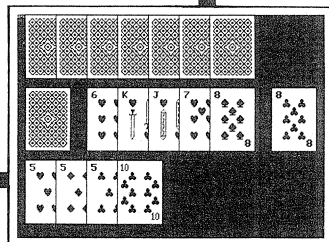
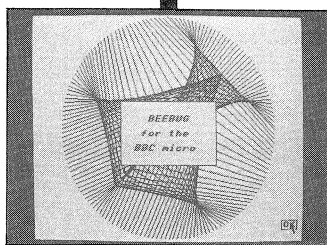
WINDOWS AND ICONS — all the routines plus demonstration program.
RAMSAFE — a utility to provide dynamic backup of your programs to sideways RAM.
PRESERVING SCREENS — the professional way to switch screens.
BASIC COMPILER — the complete program combining parts 1, 2 and 3.
BEEBUG WORKSHOP — fast timer routines for the Beeb.
FIRST COURSE — three more useful examples of the use of the EVAL function.
BURGER TIME — foil the mad sausages to complete the burgers in time.

EXTRA FEATURES THIS MONTH
BLACK JACK — a stunning implementation of this popular card game complete with full instructions.
MAGSCAN — data for this issue of BEEBUG (Vol. 4 No. 9).

Burger Time



Windows and Icons



Black Jack

All this for £3.00 (cass) £4.75 (disc) +50p p&p.

Back issues (disc since Vol. 3 No. 1, cass since Vol. 1 No. 10) available at the same prices.

Subscription rates	DISC UK	CASS UK	DISC O'seas	CASS O'seas
6 months (5 issues)	£25.50	£17	£30	£20
12 months (10 issues)	£50	£33	£56	£39

Prices are inclusive of VAT and postage as applicable. Sterling only please.

Cassette subscriptions can be commuted to disc subscription on receipt of £1.70 per issue of the subscription left to run.

All subscription and individual orders to
BEEBUG, PO BOX 109, St. Johns Road, High Wycombe HP10 8NP

Why Pay £2.50 for a FREE Demo Disc?

What You Get With Your £2.50 Free Disc

1. A stunning demonstration of many programs from the Beebugsoft range of software. Hear for yourself how good Studio 8 is. Watch Icon Master and Toolkit Plus at work. See the results of Hershey characters and Romlil. And much more.
2. If you decide to purchase any of our programs once you have seen the demonstrations, send us the disc label to receive a discount of £3.00 off the members discounted price of any one item of software from the latest 20 page full colour Beebugsoft catalogue. Hence the disc is free!
3. Also on the disc is a free arcade style machine code game. Blast the monsters with "Grid Runner".
4. A special code-breaker program is included on the disc. Issues of User, up to July 1986 will include special code numbers; type these numbers into your code-breaker program to see if you are one of the lucky winners for that month. Each winner will receive the Beebugsoft program of their choice.
5. Once you have finished using our Free Disc, you have a top quality disc which you may re-format and use for your own purposes.
6. SAVE A FURTHER £1.00. Combine your order for the Demo Disc with an order for Beebugsoft software and you may immediately deduct £1.00 from the order.

You Can't Go Wrong! Just fill in your name and address below and send it to us with £2.50.

This month's code-breaker number is R23249

This offer is limited to one disc per household/institution

Please send me a "free" disc.
I enclose a cheque for £2.50/Please debit my Barclaycard/Access No.
OR I also enclose an order for software, please charge only £1.50

Name

Address

Specify 40 or 80 Track

Send to:
Beebugsoft,
PO Box 109,
High Wycombe,
Bucks., HP10 8NP